

MS SQL Server Job Interview Questions And Answers



Interview Questions Answers

<https://interviewquestionsanswers.org/>

About Interview Questions Answers

Interview Questions Answers . ORG is an interview preparation guide of thousands of Job Interview Questions And Answers, Job Interviews are always stressful even for job seekers who have gone on countless interviews. The best way to reduce the stress is to be prepared for your job interview. Take the time to review the standard interview questions you will most likely be asked. These interview questions and answers on MS SQL Server will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts.

If you find any **question or answer** is incorrect or incomplete then you can **submit your question or answer** directly with out any registration or login at our website. You just need to visit [MS SQL Server Interview Questions And Answers](#) to add your answer click on the *Submit Your Answer* links on the website; with each question to post your answer, if you want to ask any question then you will have a link *Submit Your Question*; that's will add your question in MS SQL Server category. To ensure quality, each submission is checked by our team, before it becomes live. This [MS SQL Server Interview preparation PDF](#) was generated at **Wednesday 29th November, 2023**

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter for latest Jobs and interview preparation guides.
<https://twitter.com/InterviewQA>

If you need any further assistance or have queries regarding this document or its material or any of other inquiry, please do not hesitate to contact us.

Best Of Luck.

Interview Questions Answers.ORG Team
<https://InterviewQuestionsAnswers.ORG/Support@InterviewQuestionsAnswers.ORG>



MS SQL Server Interview Questions And Answers Guide.

Question - 1:

PHP MSSQL - How To Create an Identity Column?

Ans:

Many tables require an ID column to assign a unique ID number for each row in the table. For example, if you have a table to hold forum member profiles, you need an ID number to identify each member. To allow SQL Server to automatically assign a new ID number for each new record, you can define the ID column with IDENTITY attribute as shown in the following sample script:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "CREATE TABLE ggl_users (
    . " id INTEGER IDENTITY NOT NULL"
    . ", name VARCHAR(80) NOT NULL"
    . ", email VARCHAR(80) NULL"
    . ", time DATETIME NULL"
    . ")";
$res = mssql_query($sql, $con);
if (!$res) {
    print("Table creation failed with error:
");
    print(" ".mssql_get_last_message().
");
} else {
    print("Table ggl_users created.
");
}
mssql_close($con);
?>
```

[View All Answers](#)

Question - 2:

PHP MSSQL - How To Query Multiple Tables Jointly?

Ans:

If you want to query information stored in multiple tables, you can use the SELECT statement with a WHERE condition to make an inner join. Assuming that you have 3 tables in a forum system: "users" for user profile, "forums" for forums information, and "posts" for postings, you can query all postings from a single user with a script as shown below:

```
<?php
$con = mssql_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$userID = 101;
$sql = "SELECT posts.subject, posts.time, users.name,
    . " forums.title"
    . " FROM posts, users, forums"
    . " WHERE posts.userID = ".$userID
    . " AND posts.userID = users.id"
    . " AND posts.forumID = forums.id";
$res = mssql_query($sql, $con);
while ($row = mssql_fetch_array($res)) {
    print($row['subject'].", ".$row['time'].", "
        . $row['name'].", ".$row['title'].
");
}
mssql_free_result($res);
mssql_close($con);
?>
```

[View All Answers](#)



Question - 3:

PHP MSSQL - How To Perform Key Word Search in Tables?

Ans:

The simplest way to perform key word search is to use the SELECT statement with a LIKE operator in the WHERE clause. The LIKE operator allows you to match a text field with a keyword pattern specified as '%keyword%', where (%) represents any number of any characters. Any single quote (') in the keyword needs to be protected by replacing them with two single quotes ('). The tutorial exercise below shows you how to search for records whose "notes" contains "e":

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$key = "e";
$key = str_replace("'", "", $key);
$sql = "SELECT id, url, notes FROM ggl_links"
      . " WHERE notes LIKE '%" . $key . "%'";
$res = mssql_query($sql, $con);
while ($row = mssql_fetch_array($res)) {
    print($row['id'], ", ", $row['url'], ", "
      . $row['notes'], "\n");
}
mssql_free_result($res);
mssql_close($con);
?>
```

If you run this script, you will get something like this:

```
102, www.GlobalGuideLine.com, Nice site.
202, www.yahoo.com, It's another search engine!
301, netsc
```

[View All Answers](#)

Question - 4:

PHP MSSQL - How To Display a Past Time in Days, Hours and Minutes?

Ans:

You have seen a lots of Websites are displaying past times in days, hours and minutes. If you want to do this yourself, you can use the DATEDIFF() SQL function. The following tutorial exercise shows you how to use DATEDIFF() to present a past time in days, hours, and minutes:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$submit_time = "2007-05-29 04:09:49";
$sql = "SELECT 'Posted '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, '$submit_time',"
      . " GETDATE())/(24*60))"
      . " + ' days, '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, '$submit_time',"
      . " GETDATE())%(24*60)/60)"
      . " + ' hours, and '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, '$submit_time',"
      . " GETDATE())%60)"
      . " + ' minutes ago.'";
print(
  $sql
);
$res = mssql_query($sql, $con);
if (!$res) {
    print("SQL statement failed with error:
");
    print(" ".mssql_get_last_message().
");
} else
```

[View All Answers](#)

Question - 5:

PHP MSSQL - How To Include Date and Time Values in SQL Statements?

Ans:

If you want to provide date and time values in a SQL statement, you should write them in the format of "yyyy-mm-dd hh:mm:ss", and quoted with single quotes ('). The tutorial exercise below shows you two INSERT statements. The first one uses a hard-code date value. The second one uses the date() function to return a date value representing current date and time.

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$notes = "Added long time ago!";
$time = "1999-01-01 01:02:03";
$sql = "INSERT INTO ggl_links (id, url, notes, time)"
      . " VALUES ("
      . " 301, 'netscape.com', '" . $notes . "', '" . $time . "';"
      if (!mssql_query($sql, $con)) {
```



```
print("SQL statement failed with error:
");
print(" ".mssql_get_last_message().
");
} else {
print("1 rows inserted.
");
}
$notes = "Added today!";
$time = date("Y-m-d H:i:s");
$sql = "INSERT INTO ggl_links (id, url, notes, time)"
. " VALUES ("
. " 302, 'myspace.com', '$notes', '$time.'";
if (!$mssql_query
```

[View All Answers](#)

Question - 6:

PHP MSSQL - How To Include Text Values in SQL Statements?

Ans:

Text values in SQL statements should be quoted with single quotes ('). If the text value contains a single quote ('), it should be protected by replacing it with two single quotes ("). In SQL language syntax, two single quotes represents one single quote in string literals. The tutorial exercise below shows you two INSERT statements. The first one will fail, because it has an un-protected single quote. The second one will be ok, because a str_replace() is used to replace (') with ("):

[View All Answers](#)

Question - 7:

PHP MSSQL - How To Delete Existing Rows in a Table?

Ans:

If you want to remove a row from a table, you can use the DELETE statement with a WHERE clause to identify the row. The following sample script deletes one row:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "DELETE FROM ggl_links WHERE id = 1102";
$res = mssql_query($sql,$con);
if (!$res) {
print("SQL statement failed with error:
");
print(" ".mssql_get_last_message().
");
} else {
$number_of_rows = mssql_rows_affected($con);
print("$number_of_rows rows deleted.
");
}
mssql_close($con);
?>
```

If you run this script, you will get something like this:

1 rows deleted.

If you run it again, no rows will be deleted. And you will get something like this:

0 rows deleted.

[View All Answers](#)

Question - 8:

PHP MSSQL - How To Update Existing Rows in a Table?

Ans:

Updating existing rows in a table requires to run the UPDATE statement with a WHERE clause to identify the row. The following sample script updates one row with two new values:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "UPDATE ggl_links SET notes='Nice site.', counts=8"
. " WHERE id = 102";
$res = mssql_query($sql,$con);
if (!$res) {
print("SQL statement failed with error:
");
print(" ".mssql_get_last_message().
");
} else {
$number_of_rows = mssql_rows_affected($con);
print("$number_of_rows rows updated.
");
}
mssql_close($con);
?>
```



If you run this script, you will get something like this:
1 rows updated.

[View All Answers](#)

Question - 9:

PHP MSSQL - How To Loop through Returning Rows?

Ans:

The best way to query tables and loop through returning rows is to run a SELECT statement with the mssql_query() function, catch the returning object as a result set, and loop through the result with mssql_fetch_array() function in a while loop as shown in the following sample PHP script:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "SELECT id, url, time FROM ggl_links";
$res = mssql_query($sql,$con);
while ($row = mssql_fetch_array($res)) {
    print($row['id'].", ".$row['url'].", ".$row['time']. "
");
}
mssql_free_result($res);
mssql_close($con);
?>
```

[View All Answers](#)

Question - 10:

PHP MSSQL - What Is a Result Set Object Returned by mssql_query()?

Ans:

A result set object is a logical representation of data rows returned by mssql_query() function on SELECT statements. Every result set object has an internal pointer used to identify the current row in the result set. Once you get a result set object, you can use the following functions to retrieve detail information:

- * mssql_free_result(\$res) - Closes this result set object.
- * mssql_num_rows(\$res) - Returns the number rows in the result set.
- * mssql_num_fields(\$res) - Returns the number fields in the result set.
- * mssql_fetch_row(\$res) - Returns an array contains the next row indexed by field positions. The internal pointer is moved to the next row too.

[View All Answers](#)

Question - 11:

PHP MSSQL - How To Get the Number of Affected Rows?

Ans:

If you insert multiple rows with a single INSERT statement, you can use the mssql_rows_affected() function to find out how many rows were inserted. mssql_rows_affected(\$connection) returns the number of affected rows of the last INSET, UPDATE or DELETE statement.

The following tutorial script shows you report back the number of rows inserted properly:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "INSERT INTO ggl_links
. " SELECT id+1000, REVERSE(url), notes, counts, time"
. " FROM ggl_links WHERE id > 1000";
$res = mssql_query($sql,$con);
if (!$res) {
    print("SQL statement failed with error:
");
    print(" ".mssql_get_last_message(). "
");
} else {
    $number_of_rows = mssql_rows_affected($con);
    print("$number_of_rows rows inserted.
");
}
mssql_close($con);
```

If you run this script, you should get:
2 rows inserted

[View All Answers](#)

Question - 12:

PHP MSSQL - How To Insert Multiple Rows with a subquery?

Ans:

If want to insert rows into a table based on data rows from other tables, you can use a subquery inside the INSERT statement as shown in the following script example:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "INSERT INTO ggl_links
. " SELECT id+1000, REVERSE(url), notes, counts, time"
. " FROM ggl_links";
```



```
$res = mssql_query($sql,$con);
if (!$res) {
    print("SQL statement failed with error:
");
    print(" ".mssql_get_last_message().
");
} else {
    print("Multiple rows inserted.
");
}
mssql_close($con);
```

If you run this script, the table should have 4 rows now. And you will get:
Multiple rows inserted

[View All Answers](#)

Question - 13:

PHP MSSQL - How To Insert Data with NULL Values?

Ans:

There are two ways to provide NULL value to a column in an INSERT statement:

- * Include the column in the statement, but specify keyword NULL as the value.
- * Exclude the column from the statement.

The following tutorial exercise inserts two rows. Both of them have NULL values. But they were inserted differently:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "INSERT INTO ggl_links
    (id, url, notes, counts, time)
    VALUES (101, 'www.GlobalGuideLine.com',
    NULL, NULL, NULL)";
$res = mssql_query($sql,$con);
if (!$res) {
    print("SQL statement failed with error:
");
    print(" ".mssql_get_last_message().
");
} else {
    print("One data row inserted.
");
}
$sql = "INSERT INTO ggl_links (id, url) VALUES (
    102, 'www.GlobalGuideLine.com')";
$res = mssql_query($sql,$con);
print("Another data row inserted.
");
mssql_close($con);
?>
```

If you run this script, you will get:

One data row inserted.

Another data row inserted.

[View All Answers](#)

Question - 14:

PHP MSSQL - How To Make a Column Nullable?

Ans:

Based on the testing result from the previous tutorial you can find out that there is a big difference in the column definition when running CREATE TABLE statement with mssql_query():

* If CREATE TABLE is executed through mssql_query() and "NULL/NOT NULL" keyword is omitted in column definition, mssql_query() will assume NOT NULL.

* If CREATE TABLE is executed directly on SQL Server and "NULL/NOT NULL" keyword is omitted in column definition, SQL Server will use NULL as the default.

Now you have to modify the CREATE TABLE statement to create "ggl_links" again by adding NULL to columns: notes, counts, and time:

[View All Answers](#)

Question - 15:

PHP MSSQL - How To Insert Data into an Existing Table?

Ans:

If you want to insert a row of data into an existing table, you can use the INSERT INTO statement as shown in the following sample script:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "INSERT INTO ggl_links (id, url) VALUES (
    101, 'www.GlobalGuideLine.com')";
$res = mssql_query($sql,$con);
if (!$res) {
    print("SQL statement failed with error:
");
}
```



```
print(" ".mssql_get_last_message()."  
");  
} else {  
    print("One data row inserted.  
");  
}  
mssql_close($con);  
?>
```

If you run this script, unfortunately, you will get an error:

SQL statement failed with error:

The statement has been terminated.

So what is wrong with the statement? The error message does not give any details. You need to take this statement to SQL Server Management Studio to try it:

USE GlobalGuideLineDatabase

GO

```
INSERT INTO ggl_links (id, url) VALUES (  
    101, 'www.GlobalGuideLine.com')
```

GO

Msg 515, Level 16, State 2, Line 1

Cannot insert the value NULL into column 'notes',

table 'GlobalGuideLineDatabase.dbo.ggl_links'

[View All Answers](#)

Question - 16:

PHP MSSQL - How To Drop an Existing Table?

Ans:

If you need to delete a table created before, you can run the DROP TABLE SQL statement using the mssql_query() function, as shown in the following sample script:

```
<?php  
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');  
mssql_select_db('GlobalGuideLineDatabase', $con);  
# dropping an existing table  
$sql = "DROP TABLE ggl_links";  
$res = mssql_query($sql,$con);  
print("Table ggl_links dropped.  
");  
# creating a new table  
$sql = "CREATE TABLE ggl_links ("  
    . " id INT NOT NULL"  
    . ", url VARCHAR(80) NOT NULL"  
    . ", notes VARCHAR(1024)"  
    . ", counts INT"  
    . ", time DATETIME"  
    . ")";  
$res = mssql_query($sql,$con);  
print("Table ggl_links created.  
");  
mssql_close($con);  
?>
```

If you run this script, "ggl_links" will be dropped and created again:

Table ggl_links dropped.

Table ggl_links created.

[View All Answers](#)

Question - 17:

PHP MSSQL - How To Create a New Table?

Ans:

If you want to create a table in the SQL Server database, you can run the CREATE TABLE SQL statement using the mssql_query() function, as shown in the following sample script:

```
<?php  
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');  
mssql_select_db('GlobalGuideLineDatabase', $con);  
# creating a new table  
$sql = "CREATE TABLE ggl_links ("  
    . " id INT NOT NULL"  
    . ", url VARCHAR(80) NOT NULL"  
    . ", notes VARCHAR(1024)"  
    . ", counts INT"  
    . ", time DATETIME"  
    . ")";  
$res = mssql_query($sql,$con);  
if (!$res) {  
    print("Table creation failed with error:  
");  
    print(" ".mssql_get_last_message()."  
");  
} else {  
    print("Table ggl_links created.  
");  
}
```




```
mssql_close($con);
```

```
?>
```

If you run this script for the first time and there is no existing table called `ggl_links` in the database, you will get:

Table `ggl_links` created.

If you run it again, you will get:

Table creation failed with error:

There is already an object named '`ggl_links`' in the database.

[View All Answers](#)

Question - 18:

How To List All Field Names in the Result Set using `mssql_field_name()`?

Ans:

The result set object returned by a `SELECT` statement also contains column (field) names, lengths and types. You can use `mssql_field_name()`, `mssql_field_length()` and `mssql_field_type()` to get those information. The tutorial exercise below shows a good example:

```
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "SELECT * FROM sys.objects"
      . " WHERE type_desc='USER_TABLE'";
$res = mssql_query($sql, $con);
print("Result set columns:");
```

```
");
```

```
for ($i=0; $i
```

Run this PHP script, you will see a list columns (fields):

Result set columns:

```
name, char, 255
object_id, int, 4
principal_id, int, 4
schema_id, int, 4
parent_object_id, int, 4
type, char, 2
type_desc, char, 120
create_date, datetime, 8
modify_date, datetime, 8
is_ms_shipped, bit, 1
is_published, bit, 1
is_schema_published, bit, 1
```

[View All Answers](#)

Question - 19:

How To Retrieve Field Values using `mssql_result()`?

Ans:

Once the result set is captured in an object, you can think of it as a "table" with rows and columns (fields). You can use `mssql_result()` to retrieve the value of any given row and column (field) with this formats:

```
$value = mssql_result($res, $row, $column);
```

#- \$row is the row number, starting with 0

#- \$column is the column number, starting with 0

The tutorial PHP script below shows you how to list tables in the database with multiple field values:

```
<?php
```

```
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = "SELECT * FROM sys.objects"
      . " WHERE type_desc='USER_TABLE'";
$res = mssql_query($sql, $con);
print("User Tables:");
```

```
");
```

```
for ($i=0; $i
```

If you run this script, you will get something like:

User Tables:

```
ggl_rates, 85575343, USER_TABLE
ggl_team, 165575628, USER_TABLE
ggl_random, 821577965, USER_TABLE
ggl_links_indexed, 1061578820, USER_TABLE
ggl_links, 1093578934, USER_TABLE
ggl_links_copy, 1253579504, USER_TABLE
tipBackup2, 2121058592, USER_TABLE
```

[View All Answers](#)

Question - 20:

How To Loop through Result Set Objects using `mssql_fetch_array()`?

Ans:

If the returning output of a query statement is captured in a result set object, you can use `mssql_fetch_array()` to loop through each row in the output.

The tutorial PHP script below shows you how to list tables in the database:

```
<?php
```

```
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
```



```
$sql = "SELECT * FROM sys.objects"
      " WHERE type_desc='USER_TABLE'";
$res = mssql_query($sql, $con);
print("User Tables:
");
while ($row = mssql_fetch_array($res)) {
    print("  ".$row['name']."
");
}
mssql_free_result($res);
mssql_close($con);
?>
```

If you run this script, you will get something like:

```
User Tables:
ggl_rates
ggl_team
ggl_random
ggl_links_indexed
ggl_links
ggl_links_copy
tipBackup2
```

[View All Answers](#)

Question - 21:

How To Retrieve Error Messages using mssql_get_last_message()?

Ans:

When you call mssql_query() to execute a SQL statement, and the execution failed on the SQL Server, you can use mssql_get_last_message() function to retrieve the error messages.

The tutorial script below shows you a good example:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
# dropping a table
$sql = 'DROP TABLE fyi.center';
$res = mssql_query($sql, $con);
if (!$res) {
    print("Execution failed:
");
    print("  Error: ".mssql_get_last_message().
");
} else {
    print("Execution was successful.
");
}
mssql_close($con);
?>
```

[View All Answers](#)

Question - 22:

How To Execute a SQL Statement using mssql_query()?

Ans:

Once you have created a connection object, you can use the mssql_query() function to send a SQL statement to the SQL Server linked to the connection object for execution.

Here is a simple PHP script that creates a new schema and a new table:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
# creating a new schema
$sql = 'CREATE SCHEMA ggl';
mssql_query($sql, $con);

# creating a new table
$sql = 'CREATE TABLE globalguideline (name VARCHAR(80))';
mssql_query($sql, $con);
mssql_close($con);
?>
```

If you run this script for the first time, it will execute those two statements correctly for you. But if you run it again, you will see some warning messages:

Warning: mssql_query(): message: There is already an object named 'ggl' in the database. (severity 16)

in C: estglobalguideline.php on line 7

Warning: mssql_query(): message: CREATE SCHEMA failed due to previous errors. (severity 16)

in C: estglobalguideline.php on line 7

The messages are very clear and easy to understand.

[View All Answers](#)

Question - 23:



How To Select an Existing Database using mssql_select_db()?

Ans:

The first thing after you have created a connection object to the SQL Server is to select the database where your tables are located, by using the mssql_select_db() function. If your MSSQL server is offered by your Web hosting company, they will assign an empty database to you and provide you the database name. You should use this name to select this empty database as your current database. The following script shows you how to select a database called "GlobalGuideLineDatabase".

To test the mssql_select_db() function, try the following script:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
mssql_select_db('GlobalGuideLineDatabase', $con);
$sql = 'SELECT * FROM sys.tables';
if ($res = mssql_query($sql, $con)) {
    print(mssql_num_rows($res) . " tables in database.
");
} else {
    print("SQL failed.
");
}
mssql_close($con);
?>
```

You will get something like this:
10 tables in database.

[View All Answers](#)

Question - 24:

How To Disconnect from a SQL Server using mssql_close()?

Ans:

When you call mssql_connect(), it will return an object representing the connection to the SQL Server. This connection object will be used by subsequent MSSQL function calls. When you are done with this connection, you should close it to free up resources by calling the mssql_close() function.

The tutorial script below shows you how to call mssql_connect() and mssql_close() to connect and disconnect to the SQL Server:

```
<?php
$con = mssql_connect('LOCALHOST','sa','GlobalGuideLine');
if (!$con) {
    print("Connection failed with error.
");
} else {
    print("The SQL Server connection object is ready.
");
}
mssql_close($con);
?>
```

If you run this script and get this output: "The SQL Server connection object is ready", your connection to the SQL Server are working.

Note that 'sa' and 'GlobalGuideLine' used in this script are system administrator login name and password. You may use any other login name and password defined on the SQL Server.

[View All Answers](#)

Question - 25:

What Are Commonly Used MSSQL Functions in PHP?

Ans:

If you look at the PHP 5 manual, you will see a group of functions listed under the Microsoft SQL Server Functions section. The commonly used functions are:

mssql_connect ? Open MS SQL server connection mssql_close ? Close MS SQL Server connection mssql_select_db ? Select MS SQL database mssql_query ? Send MS SQL query mssql_num_rows ? Gets the number of rows in result mssql_fetch_row ? Get row as enumerated array mssql_fetch_assoc ? Returns an associative array of the current row in the result mssql_fetch_array ? Fetch a result row as an associative array, a numeric array, or both mssql_fetch_object ? Fetch row as object mssql_free_result ? Free result memory mssql_num_fields ? Gets the number of fields in result mssql_field_name ? Get the name of a field mssql_field_type ? Gets the type of a field mssql_field_length ? Get the length of a field mssql_fetch_field ? Get field information mssql_get_last_message ? Returns the last message from the server mssql_rows_affected ? Returns the number of records affected by the query
Some of the functions will be discussed in this tutorial collection.

[View All Answers](#)

Question - 26:

How To Connect PHP with Different Port Numbers?

Ans:

You know that SQL Server could be configured to accept connections with different TCP/IP port numbers. See other tutorial collections on how to view and configure SQL Server TCP/IP protocol.

If you installed SQL Server 2005 Express Edition with default settings, it should be running with instance name of "SQLEXPRESS" and port number of "1269" You can use the mssql_connect() function to connect to the server in 3 ways:

```
$con = mssql_connect('LOCALHOST','login','pass');
$con = mssql_connect('LOCALHOST\SQLEXPRESS','login','pass');
$con = mssql_connect('LOCALHOST,1269','login','pass');
```

Other ways of entering the server name and port number will not work. The PHP manual has this statement: "servername - The MS SQL server. It can also include a port number. e.g. hostname:port." The example is really for non-Windows systems. Try the following testing PHP script to find out:

[View All Answers](#)

Question - 27:



Where to Find ntwdlib.dll Version 2000.80.194.0?

Ans:

You know that the copy of ntwdlib.dll version 2000.2.8.0 included in Windows binary version of PHP 5.2.3 does not work with MS SQL Server 2005. You need to get a new version of ntwdlib.dll.

One way to get a free copy of ntwdlib.dll is to download MS SQL Server 2000 SP4 from Microsoft Website. It contains a copy of ntwdlib.dll version 2000.80.194.0 in the system directory.

Copy ntwdlib.dll from SQL Server 2000 SP4 package to C:

twdblib.dll-new. Check the properties of this file, you will see the version is 2000.80.194.0. Enter this single-statement script, fyi-center.php, for testing:

```
<?php
    mssql_connect('LOCALHOST','sa','GlobalGuideLine');
?>
```

[View All Answers](#)

Question - 28:

What Happens If ntwdlib.dll Is Missing on Your Machine?

Ans:

In order to install the proper version of ntwdlib.dll on your machine, you need to find out how many copies of ntwdlib.dll do you have on your machine and what are the version numbers on those copies.

1. Go to Start > Search > All files and folders. Enter ntwdlib.dll in the "All or part of the file name:" field and click Search. You should see only one copy:

ntwdlib.dll C:\php

Check the properties of this copy, you will see its version is 2000.2.8.0.

2. Rename C:\php

twdblib.dll to C:\php

twdblib.dll-old. Run this script:

```
<?php mssql_connect('LOCALHOST','sa','GlobalGuideLine');
?>
```

You see this error message dialog box and an error in the command window:

Missing ntwdlib

PHP Warning: PHP Startup: Unable to load dynamic library

'./extphp_mssql.dll' - The specified module could not be found.

in Unknown on line 0

Fatal error: Call to undefined function mssql_connect()

in C:\estglobalguideline.php on line 2

This proves that there is only one copy of ntwdlib.dll on your machine. Read the next tutorial to see how to download the correct version of ntwdlib.dll for SQL Server 2005.

[View All Answers](#)

Question - 29:

What Is Wrong with SQL Server Client Library DLL, ntwdlib.dll?

Ans:

The second requirement to access SQL Server with PHP scripts is the SQL Server Client Library DLL, ntwdlib.dll. The good news is that the Windows binary version of PHP 5.2.3 has ntwdlib.dll included in the PHP home directory.

But the bad news is that the included copy of ntwdlib.dll is not compatible with SQL Server 2005. What makes it worse is that no specific error messages is returned from the PHP engine, php_mssql.dll, ntwdlib.dll, or the SQL Server. All you get is "Unable to connect to server: LOCALHOST", and you know that SQL Server is running properly on LOCALHOST.

If you read the PHP manual, it does mention that "copying ntwdlib.dll from winntsystem32 on the server (SQL Server machine) to winntsystem32 on the PHP box".

This instruction is too short and causes the following confusions:

- * Windows binary version of PHP 5.2.3 already has a copy of ntwdlib.dll included. But that copy is not good for SQL Server 2005.

- * There is no ntwdlib.dll on the SQL Server 2005 Express Edition machine.

- * You don't need to place ntwdlib.dll in winntsystem32. Placing it in PHP home directory is fine.

[View All Answers](#)

Question - 30:

How to Turn on the MSSQL API Module in PHP?

Ans:

If you want to access SQL Server database with PHP scripts, the first thing you need to do is to turn on the MSSQL API Module as shown in this tutorial:

1. Prepare a single line script to test the MSSQL API Module:

```
<?php
    mssql_connect('LOCALHOST','sa','GlobalGuideLine');
?>
```

Run the script, you should get:

Fatal error: Call to undefined function mssql_connect()

in C:\estglobalguideline.php on line 2

2. If you C:\php\php.ini does not exist, copy C:\php\php.ini-dist to C:\php\php.ini. Open C:\php\php.ini and remove (;) on the extension=php_mssql.dll setting:

;extension=php_mssql.dll

extension=php_mssql.dll

Run the script again, you may get:

PHP Warning: PHP Startup: Unable to load dynamic library

'C:\php\php_mssql.dll' - The specified module could not be found.

in Unknown on line 0

Fatal error: Call to undefined function mssql_connect()

in C:\estglobalguideline.php on line 2

3. Open C:\php\php.ini and change the extension_dir setting. The default setting is "./", but all extension module DLL files are in ./ext.



[View All Answers](#)

Question - 31:

What Do You Need to Connect PHP to SQL Server?

Ans:

If you want to access MS SQL Server in your PHP script, you need to make sure that:

1. MSSQL API module (extension) is installed and turned on in your PHP engine. If you installed the Windows binary version of PHP 5.2.3, MSSQL API module is included but not turned yet. See the next tutorial on turning on MSSQL API module.

If you are getting errors like: "Fatal error: Call to undefined function mssql_connect()", you know that MSSQL API module is not turned on correctly. See the next tutorial on turning on the MSSQL API module.

2. MS SQL Client Tools to be installed on the system where PHP is installed. The PHP manual also suggested an alternative, which is to get a copy of the SQL Server Client Library DLL, ntdbllib.dll. Again Windows binary version of PHP 5.2.3 contains a copy of ntdbllib.dll.

If you are getting errors like: "Unable to load dynamic library './extphp_mssql.dll' - The specified module could not be found", you know that the DLL, ntdbllib.dll, is not installed correctly. See the next tutorial on installing ntdbllib.dll.

[View All Answers](#)

Question - 32:

PHP ODBC - How To Create an Identity Column?

Ans:

Many tables require an ID column to assign a unique ID number for each row in the table. For example, if you have a table to hold forum member profiles, you need an ID number to identify each member. To allow SQL Server to automatically assign a new ID number for each new record, you can define the ID column with IDENTITY attribute as shown in the following sample script:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "CREATE TABLE ggl_users ("
      . " id INTEGER IDENTITY NOT NULL"
      . ", name VARCHAR(80) NOT NULL"
      . ", email VARCHAR(80)"
      . ", time DATETIME"
      . ")";
$res = odbc_exec($con, $sql);
if (!$res) {
    print("Table creation failed with error:
");
    print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
    print("Table ggl_users created.
");
}
odbc_close($con);
?>
```

If you run this script, a new table will be created with ID column defined as an identity column. Its value will be auto-assigned each time a new record is inserted with an auto-incremented number. The sample script below inserts two records with ID values assigned by SQL Server:

If you run this script, you will get something

[View All Answers](#)

Question - 33:

PHP ODBC - How To Query Multiple Tables Jointly?

Ans:

If you want to query information stored in multiple tables, you can use the SELECT statement with a WHERE condition to make an inner join. Assuming that you have 3 tables in a forum system: "users" for user profile, "forums" for forums information, and "posts" for postings, you can query all postings from a single user with a script as shown below:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$userID = 101;
$sql = "SELECT posts.subject, posts.time, users.name, "
      . " forums.title"
      . " FROM posts, users, forums"
      . " WHERE posts.userID = ".$userID
      . " AND posts.userID = users.id"
      . " AND posts.forumID = forums.id";
$res = odbc_exec($con, $sql);
while ($row = odbc_fetch_array($res)) {
    print($row['subject'].", ".$row['time'].", "
      . $row['name'].", ".$row['title'].
");
}
odbc_free_result($res);
odbc_close($con);
?>
```

[View All Answers](#)

Question - 34:



PHP ODBC - How To Perform Key Word Search in Tables?

Ans:

The simplest way to perform key word search is to use the SELECT statement with a LIKE operator in the WHERE clause. The LIKE operator allows you to match a text field with a keyword pattern specified as '%keyword%', where (%) represents any number of any characters. Any single quote (') in the keyword needs to be protected by replacing them with two single quotes ('). The tutorial exercise below shows you how to search for records whose "notes" contains "e":

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$key = "e";
$key = str_replace("'", "", $key);
$sql = "SELECT id, url, notes FROM ggl_links"
      . " WHERE notes LIKE '%" . $key . "%'";
$res = odbc_exec($con, $sql);
while ($row = odbc_fetch_array($res)) {
    print($row['id'], ", ", $row['url'], ", "
        . $row['notes'], "
    );
}
odbc_free_result($res);
odbc_close($con);
?>
```

If you run this script, you will get something like this:

102, www.GlobalGuideLine.com, Nice site.

202, www.google.com, It's another search engine!

301, yahoo.com, Added long time ago!

302, mspace.com, Added today!

[View All Answers](#)

Question - 35:

PHP ODBC - How To Display a Past Time in Days, Hours and Minutes?

Ans:

You have seen a lots of Websites are displaying past times in days, hours and minutes. If you want to do this yourself, you can use the DATEDIFF() SQL function. The following tutorial exercise shows you how to use DATEDIFF() to present a past time in days, hours, and minutes:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$submit_time = "2007-05-29 04:09:49";
$sql = "SELECT 'Posted '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, 'submit_time',"
      . " GETDATE()))/(24*60))"
      . " + ' days, '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, 'submit_time',"
      . " GETDATE()))%(24*60)/60)"
      . " + ' hours, and '"
      . " + CONVERT(VARCHAR(40),"
      . " DATEDIFF(minute, 'submit_time',"
      . " GETDATE()))%60)"
      . " + ' minutes ago.'";
print("
$sql
");
$res = odbc_exec($con, $sql);
if (!$res) {
    print("SQL statement failed with error:
");
    print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
    odbc_fetch_row($res);
    $message
```

[View All Answers](#)

Question - 36:

PHP ODBC - How To Include Date and Time Values in SQL Statements?

Ans:

If you want to provide date and time values in a SQL statement, you should write them in the format of "yyyy-mm-dd hh:mm:ss", and quoted with single quotes ('). The tutorial exercise below shows you two INSERT statements. The first one uses a hard-code date value. The second one uses the date() function to return a date value.

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$notes = "Added long time ago!";
$time = "1999-01-01 01:02:03";
$sql = "INSERT INTO ggl_links (id, url, notes, time)"
      . " VALUES ("
      . " 301, 'netscape.com', '" . $notes . "', '" . $time . "')";
if (!odbc_exec($con, $sql)) {
    print("SQL statement failed with error:
```



```
");
print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
print("1 rows inserted.
");
}
$notes = "Added today!";
$time = date("Y-m-d H:i:s");
$sql = "INSERT INTO ggl_links (id, url, notes, time)"
. " VALUES ("
. " 302, 'myspace.com', '". $notes."', '". $time.'"");
```

[View All Answers](#)

Question - 37:

PHP ODBC - How To Include Text Values in SQL Statements?

Ans:

Text values in SQL statements should be quoted with single quotes ('). If the text value contains a single quote ('), it should be protected by replacing it with two single quotes ("). In SQL language syntax, two single quotes represents one single quote in string literals.

The tutorial exercise below shows you two INSERT statements. The first one will fail, because it has an un-protected single quote. The second one will be ok, because a str_replace() is used to replace (') with ("):

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GGL');
$notes = "It's a search engine!";
$sql = "INSERT INTO ggl_links (id, url, notes) VALUES ("
. " 201, 'www.google.com', '". $notes.'"");
if (!odbc_exec($con, $sql)) {
print("SQL statement failed with error:
");
print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
print("1 rows inserted.
");
}
$notes = "It's another search engine!";
$notes = str_replace("'", "", $notes);
$sql = "INSERT INTO ggl_links (id, url, notes) VALUES ("
. " 202, 'www.yahoo.com', '". $notes.'"");
```

[View All Answers](#)

Question - 38:

PHP ODBC - How To Delete Existing Rows in a Table?

Ans:

If you want to remove a row from a table, you can use the DELETE statement with a WHERE clause to identify the row. The following sample script deletes one row:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "DELETE FROM ggl_links WHERE id = 1102";
$res = odbc_exec($con, $sql);
if (!$res) {
print("SQL statement failed with error:
");
print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
$number_of_rows = odbc_num_rows($res);
print("$number_of_rows rows deleted.
");
}
odbc_close($con);
?>
```

If you run this script, you will get something like this:

1 rows deleted.

If you run it again, no rows will be deleted. And you will get something like this:

0 rows deleted.

[View All Answers](#)

Question - 39:

PHP ODBC - How To Update Existing Rows in a Table?

Ans:

Updating existing rows in a table requires to run the UPDATE statement with a WHERE clause to identify the row. The following sample script updates one row with two new values:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "UPDATE ggl_links SET notes='Nice site.', counts=8"
```




```
. " WHERE id = 102";
$res = odbc_exec($con, $sql);
if (!$res) {
    print("SQL statement failed with error:
");
    print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
    $number_of_rows = odbc_num_rows($res);
    print("$number_of_rows rows updated.
");
}
odbc_close($con);
?>
```

If you run this script, you will get something like this:
1 rows updated.

[View All Answers](#)

Question - 40:

PHP ODBC - How To Loop through Returning Rows?

Ans:

The best way to query tables and loop through returning rows is to run a SELECT statement with the odbc_exec() function, catch the returning object as a result set, and loop through the result with odbc_fetch_array() function in a while loop as shown in the following sample PHP script:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "SELECT id, url, time FROM ggl_links";
$res = odbc_exec($con, $sql);
while ($row = odbc_fetch_array($res)) {
    print($row['id'].", ".$row['url'].", ".$row['time'].
");
}
odbc_free_result($res);
odbc_close($con);
?>
```

Using odbc_fetch_array() is better than other fetch functions, because it allows you to access field values by field names. If you run this script, you will see all rows from the ggl_links table are printed on the screen:

```
101.globalguideline.com,
102.globalguideline.com/sql,
1101.globalguideline.com/html,
1102.globalguideline.com/xml,
2101.globalguideline.com/xslt,
2102.globalguideline.com/seo,
```

Don't forget to call odbc_free_result(\$res). It is important to free up result set objects as soon as you are done with them.

[View All Answers](#)

Question - 41:

PHP ODBC - What Is a Result Set Object Returned by odbc_exec()?

Ans:

A result set object is a logical representation of data rows returned by odbc_exec() function on SELECT statements. Every result set object has an internal pointer used to identify the current row in the result set. Once you get a result set object, you can use the following functions to retrieve detail information:

- * odbc_free_result(\$res) - Closes this result set object.
- * odbc_num_rows(\$res) - Returns the number rows in the result set.
- * odbc_num_fields(\$res) - Returns the number fields in the result set.
- * odbc_fetch_row(\$res) - Moving the internal pointer to the next row.
- * odbc_fetch_array(\$res) - Returns an array contains the next row indexed by field names. The internal pointer is moved to the next row too.
- * odbc_fetch_object(\$res) - Returns an object representing the next row. The internal pointer is moved to the next row too.
- * odbc_field_len(\$res, \$i) - Returns an array contains lengths of all fields in the last row returned.
- * odbc_field_name(\$res, \$i) - Returns the name of the field of the specified index.
- * odbc_result(\$i) - Returns the value of the field specified by its position from the current row.
- * odbc_result(\$name) - Returns the value of the field specified by its name from the current row.

[View All Answers](#)

Question - 42:

PHP ODBC - How To Get the Number of Affected Rows?

Ans:

If you insert multiple rows with a single INSERT statement, you can use the odbc_num_rows() function to find out how many rows were inserted. odbc_num_rows(\$result_set) returns the number of affected rows based on the result set object returned by the last INSERT, UPDATE or DELETE statement.

The following tutorial script shows you report back the number of rows inserted properly:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "INSERT INTO ggl_links
    . " SELECT id+1000, REVERSE(url), notes, counts, time"
    . " FROM ggl_links WHERE id > 1000";
$res = odbc_exec($con, $sql);
if (!$res) {
    print("SQL statement failed with error:
```




```
");  
print(odbc_error($con).": ".odbc_errormsg($con)."  
");  
} else {  
    $number_of_rows = odbc_num_rows($res);  
    print("$number_of_rows rows inserted.  
");  
}  
odbc_close($con);
```

If you run this script, you should get:

2 rows inserted

[View All Answers](#)

Question - 43:

PHP ODBC - How To Insert Multiple Rows with a subquery?

Ans:

If want to insert rows into a table based on data rows from other tables, you can use a subquery inside the INSERT statement as shown in the following script example:

```
<?php  
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');  
$sql = "INSERT INTO ggl_links"  
    . " SELECT id+1000, REVERSE(url), notes, counts, time"  
    . " FROM ggl_links";  
$res = odbc_exec($con, $sql);  
if (!$res) {  
    print("SQL statement failed with error:  
");  
    print(odbc_error($con).": ".odbc_errormsg($con)."  
");  
} else {  
    print("Multiple rows inserted.  
");  
}  
odbc_close($con);
```

If you run this script, the table should have 4 rows now. And you will get:

Multiple rows inserted

[View All Answers](#)

Question - 44:

PHP ODBC - How To Insert Data into an Existing Table?

Ans:

If you want to insert a row of data into an existing table, you can use the INSERT INTO statement as shown in the following sample script:

```
<?php  
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');  
$sql = "INSERT INTO ggl_links (id, url) VALUES ("  
    . " 101, 'GlobalGuideLine.com')";  
$res = odbc_exec($con, $sql);  
if (!$res) {  
    print("SQL statement failed with error:  
");  
    print(odbc_error($con).": ".odbc_errormsg($con)."  
");  
} else {  
    print("One data row inserted.  
");  
}  
$sql = "INSERT INTO ggl_links (id, url) VALUES ("  
    . " 102, 'GlobalGuideLine.com')";  
$res = odbc_exec($con, $sql);  
print("One data row inserted.  
");  
odbc_close($con);  
?>
```

If you run this script, two data rows should be inserted into the table. And you will get:

One data row inserted.

One data row inserted.

[View All Answers](#)

Question - 45:

PHP ODBC - How To Create a New Table?

Ans:

If you want to create a table in the database connected through a ODBC DSN, you can run the CREATE TABLE SQL statement using the odbc_exec() function, as shown in the following sample script:

```
<?php  
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');  
# creating a new table
```



```
$sql = "CREATE TABLE ggl_links ("
. " id INT NOT NULL"
. ", url VARCHAR(80) NOT NULL"
. ", notes VARCHAR(1024)"
. ", counts INT"
. ", time DATETIME"
. ")";
$res = odbc_exec($con, $sql);
if (!$res) {
    print("Table creation failed with error:
");
    print(odbc_error($con).": ".odbc_errormsg($con).
");
} else {
    print("Table ggl_links created.
");
}
odbc_close($con);
?>
```

[View All Answers](#)

Question - 46:

How To List All Columns in a Table using odbc_columns()?

Ans:

If you want to get a list of all columns in a table, you can use the odbc_columns() function, which can actually be used to list all columns in all tables and views in the database. The syntax of odbc_columns() is:

```
$result_set = odbc_columns($connection_object,
    $qualifier, # database name for SQL Server
    $schema, # schema name for SQL Server
    $table_name, # table or view name for SQL Server
    $column_name # column name
)
```

#- The returning result set contains 5 fields:

```
#- TABLE_QUALIFIER, TABLE_SCHEMA, TABLE_NAME, COLUMN_NAME
#- DATA_TYPE, TYPE_NAME, PRECISION, LENGTH, SCALE, RADIX
#- NULLABLE, REMARKS
```

The schema, table_name and column_name arguments accept search patterns ('%' to match zero or more characters and '_' to match a single character).

The tutorial example below shows you how to get a list of columns in a table called ggl_rates in the current database, GlobalGuideLineDatabase, which is hard coded in the DSN definition:

[View All Answers](#)

Question - 47:

How To List All Tables in the Database using odbc_tables()?

Ans:

If you want to get a list of all tables in the database, you can use the odbc_tables() function, which can actually be used to list all tables and views in the database. The syntax of odbc_tables() is:

```
$result_set = odbc_tables($connection_object,
    $qualifier, # database name for SQL Server
    $owner, # schema name for SQL Server
    $name, # table or view name for SQL Server
    $type # valid type names are TABLE and VIEW
)
```

#- The returning result set contains 5 fields:

```
#- TABLE_QUALIFIER, TABLE_OWNER, TABLE_NAME, TABLE_TYPE,
#- REMARKS
```

The owner and name arguments accept search patterns ('%' to match zero or more characters and '_' to match a single character).

The tutorial example below shows you how to get a list of tables in the current database, GlobalGuideLineDatabase, which is hard coded in the DSN definition:

[View All Answers](#)

Question - 48:

How To Create Prepared Statements using odbc_prepare()?

Ans:

If you have a SQL statement that need to executed repeatedly many times with small changes, you can create a prepared statement object with parameters so it can be executed more efficiently.

There are two functions you need to use prepare and execute a prepared statement object:

```
$statement_object = odbc_prepare($connection,
    $statement_string);
```

#- The \$statement_string may have parameters represented

#- by "?".

```
$result_set = odbc_execute($statement_object $array);
```

#- The \$array is used to supply values to parameters

#- defined in the statement object.

The tutorial PHP script below shows you how to insert 3 rows into a table with a prepared statement object with 2 parameters:

[View All Answers](#)



Question - 49:

How To Loop through Result Set Objects using `odbc_fetch_row()`?

Ans:

If the returning output of a query statement is captured in a result set object, you can use `odbc_fetch_row()` to loop through each row in the output. The tutorial PHP script below shows you how to list tables in the database:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = "SELECT * FROM sys.objects"
      " WHERE type_desc='USER_TABLE'";
$res = odbc_exec($con, $sql);
print("User Tables:
");
while (odbc_fetch_row($res)) {
    print(" ".odbc_result($res,'name')."
");
}
odbc_free_result($res);
odbc_close($con);
?>
```

If you run this script, you will get something like:

```
User Tables:
ggl_rates
ggl_team
ggl_random
ggl_links_indexed
ggl_links
ggl_links_copy
tipBackup2
```

[View All Answers](#)

Question - 50:

How To Receive Returning Result from a Query?

Ans:

When you execute a SQL SELECT statement with the `odbc_exec()` function, you can capture the returning result with a result set object with the following syntax:

```
$result_set = odbc_exec($sql_statement);
```

#- The returning value could be a Boolean value FALSE,

#- if the execution failed.

Data rows and field values in the result set object can be retrieved using other ODBC functions as shown in the tutorial PHP script below:

```
<?php
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');
$sql = 'SELECT GETDATE()';
$res = odbc_exec($con, $sql);
odbc_fetch_row($res);
$date = odbc_result($res,1);
print("Database current time: ". $date ."
");
odbc_close($con);
?>
```

If you run this script, you will get something like this:

```
Database current time: 2007-06-02 22:07:05.110
```

[View All Answers](#)

Question - 51:

How To Turn Off Warning Messages during PHP Execution?

Ans:

If you don't want see warning messages generated from the PHP engine when executing PHP scripts, you can change the `error_reporting` setting in the `php.ini` configuration file.

Open `php.ini` and change the following lines:

```
;error_reporting = E_ALL & ~E_NOTICE
```

```
error_reporting = E_ALL & ~E_WARNING
```

Now run the script in the previous tutorial again, you will not see the warning messages from the PHP engine:

Execution failed:

```
State: S0002
```

```
Error: [Microsoft][ODBC SQL Server Driver][SQL Server]
```

```
Cannot drop the table 'GlobalGuideLine', because it does not exist or you do not have permission.
```

[View All Answers](#)

Question - 52:

How To Retrieve Error Messages using `odbc_errormsg()`?

Ans:

When you call `odbc_exec()` to execute a SQL statement, and the execution failed on the SQL Server, you can use `odbc_error()` and `odbc_errormsg()` to retrieve the error code and error messages.

The tutorial script below shows you a good example:



```
<?php
$con = odbc_connect('gg1_SQL_SERVER','sa','GlobalGuideLine');
# dropping a table
$sql = 'DROP TABLE fyi.center';
$res = odbc_exec($con, $sql);
if (!$res) {
    print("Execution failed:
");
    print("  State: ".odbc_error($con).
");
    print("  Error: ".odbc_errormsg($con).
");
} else {
    print("Execution was successful.
");
}
odbc_close($con);
?>
```

If you run this script for the first time, you will get this output:

Execution was successful.

If you run this script again, the SQL statement will fail on the SQL Server, and you will get:

```
Warning: odbc_exec(): SQL error: [Microsoft]
[ODBC SQL Server Driver][SQL Server]
Cannot drop the table 'fyi.center', because
it does not exist or you do not have permission.,
SQL state S0002 in SQLExecDirect in C: estggl_center.php
on line 6
Execution failed:
  State: S0002
  Error: [Microsoft][ODBC SQL
```

[View All Answers](#)

Question - 53:

How To Execute a SQL Statement using odbc_exec()?

Ans:

Once you have created an ODBC connection object, you can use the odbc_exec() function to send a SQL statement to the SQL Server linked to the connection object for execution.

Here is a simple PHP script that creates a new schema and a new table:

```
<?php
$con = odbc_connect('gg1_SQL_SERVER','sa','GlobalGuideLine');
# creating a new schema
$sql = 'CREATE SCHEMA fyi';
odbc_exec($con, $sql);

# creating a new table
$sql = 'CREATE TABLE fyi.center (name VARCHAR(80))';
odbc_exec($con, $sql);
odbc_close($con);
?>
```

If you run this script for the first time, it will execute those two statements correctly for you. But if you run it again, you will see some warning messages:

```
Warning: odbc_exec(): SQL error: [Microsoft]
[ODBC SQL Server Driver][SQL Server]
There is already an object named 'fyi' in the database.,
SQL state S0001 in SQLExecDirect in C: estggl_center.php
on line 6
Warning: odbc_exec(): SQL error: [Microsoft]
[ODBC SQL Server Driver][SQL Server]
There is already an object named 'center' in the database.,
SQL state S0001 in SQLExecDirect in C: estggl_center.php
on line 10
The messages are very clear and easy to understand.
```

[View All Answers](#)

Question - 54:

How To List All DSN Entries on Your Local Machine using odbc_data_source()?

Ans:

If you are interested to know what DSN entries are available on your local machine, you can use odbc_data_source(\$con, SQL_FETCH_FIRST) and odbc_data_source(\$con, SQL_FETCH_NEXT) in a loop to list all DSN entries defined on your local machine. The tutorial script below shows a good example:

```
<pre><?php
$con = odbc_connect('gg1_SQL_SERVER','sa','GlobalGuideLine');
if (!$con) {
    print("There is a problem with the connection.
");
} else {
    print("The ODBC connection object is ready.
");
    $list = odbc_data_source($con, SQL_FETCH_FIRST);
    while ($list) {
```



```
foreach ($list as $key => $value) {  
    print($key . " = " . $value . "  
");  
}  
$list = odbc_data_source($con, SQL_FETCH_NEXT);  
}  
odbc_close($con);  
}  
?></pre>
```

[View All Answers](#)

Question - 55:

How To Connect to a SQL Server using odbc_connect()?

Ans:

If you have an ODBC DSN (Data Source Name) created linking to a SQL Server, you are ready to connect to the SQL Server through the DSN with ODBC functions. There is no changes needed in the php.ini configuration file.

The tutorial script below shows you how to call odbc_connect() and odbc_close() to connect and disconnect to the SQL Server through the DSN name "ggl_SQL_SERVER":

```
<?php  
$con = odbc_connect('ggl_SQL_SERVER','sa','GlobalGuideLine');  
if (!$con) {  
    print("There is a problem with SQL Server connection.  
");  
}  
} else {  
    print("The SQL Server connection object is ready.  
");  
}  
odbc_close($con);  
}  
?>
```

If you run this script and get this output: "The SQL Server connection object is ready", your PHP environment and ODBC connection to the SQL Server are working. Note that 'sa' and 'GlobalGuideLine' used in this script are system administrator login name and password. You may use any other login name and password defined on the SQL Server.

[View All Answers](#)

Question - 56:

How To Test ODBC DSN Connection Settings?

Ans:

Assuming you have followed other GlobalGuideline.com tutorials and created an ODBC DSN called "ggl_SQL_SERVER", and planning to use it your PHP scripts, you should test this ODBC connection first as shown in this tutorial:

1. Go to Control Panel > Administrative Tools.
2. Run Data Sources (ODBC). The ODBC Data Source Administrator window shows up.
3. Click "System DSN" tab, select "ggl_SQL_SERVER", and click "Configure..." button. The Microsoft SQL Server DSN Configuration wizard window shows up.
4. Review the first screen and click Next.
5. Review the second screen, enter the password as "GlobalGuideLine", and click Next.
6. Review the third screen and click Next.
7. Review the fourth screen and click Finish.
8. Review the confirmation screen and click "Test Data Source...". You should see the test result as:

Attempting connection

Connection established

Verifying option settings

Disconnecting from server

TESTS COMPLETED SUCCESSFULLY!

Your ggl_SQL_SERVER ODBC DSN is tested and ready to use.

[View All Answers](#)

Question - 57:

What Are Commonly Used ODBC Functions in PHP?

Ans:

If you look at the PHP 5 manual, you will see a group of functions listed under the ODBC Functions (Unified) section. The commonly used ODBC functions are:

- * odbc_connect ? Establish an ODBC connection.
- * odbc_data_source ? Returns information about a current connection.
- * odbc_close ? Close an ODBC connection.
- * odbc_exec ? Prepare and execute a SQL statement.
- * odbc_fetch_row ? Fetch a row - moving the pointer to the a new row in a result object.
- * odbc_result ? Get the value of a specific field from the current row of a result object.
- * odbc_fetch_array ? Fetch a result row as an associative array.
- * odbc_fetch_object ? Fetch a result row as an object.
- * odbc_num_rows ? Number of rows in a result.
- * odbc_field_name ? Get the name of a specified field index.
- * odbc_field_type ? Get the data type of a specified field index.
- * odbc_next_result ? Checks if multiple results are available.
- * odbc_free_result ? Free resources associated with a result object.
- * odbc_prepare ? Prepares a statement for execution.

[View All Answers](#)



Question - 58:

What Are the Requirements to Use ODBC Connections in PHP Scripts?

Ans:

If you are planning to use ODBC connections to access SQL Server databases in PHP scripts, you need to check the following requirements:

- * The PHP engine must support ODBC functions. If you install PHP 5.2.2 from The PHP Group, the ODBC functions are already included in the default installation.
- * The SQL Server must have TCP/IP protocol enabled for a specific port.
- * The SQL Server Browser Service must be running on the SQL server machine.
- * An ODBC DSN must be created on machine where you are running PHP scripts.
- * The ODBC DSN connection must be tested to make sure it is working.

You need to following other tutorials provided by GlobalGuideLine.com to install PHP engine, install SQL Server, configure TCP/IP protocol, start SQL Server Browser Service, create and test ODBC DSN connections.

[View All Answers](#)

Question - 59:

How Can Windows Applications Connect to SQL Servers via ODBC?

Ans:

One way of connecting a windows application to a SQL Server is to use ODBC drivers. The requirements to do this is summarized here:

- * The SQL Server must have TCP/IP protocol enabled with a specific port number.
- * The SQL Server Browser Service must be running on the server machine.
- * A ODBC DSN must be created to represent the connection to SQL Server on the specified port number.
- * The local windows application must be able to support ODBC API calls.

The diagram below shows how MS Access can connect to a SQL Server through the ODBC driver:

SQL Server ODBC Connection

[View All Answers](#)

Question - 60:

How To Connect MS Access to SQL Servers through ODBC?

Ans:

Once you got a DSN defined in the ODBC manager that connects to your SQL Server, you can connect a normal MS Access document to the Oracle server, and link an Access table to a SQL Server table. The tutorial below gives you a good example:

- * Start MS Access with a new database file.
- * Go to File menu.
- * Select Get External Data.
- * Select Import.... The Import dialog box shows up.
- * Select Files of type: ODBC Database(). The Select Data Source dialog box shows up.
- * Click the Machine Data Source tab. You should see the DSN name "ggl_SQL_SERVER" you defined earlier.
- * Select "ggl_SQL_SERVER".
- * Enter User Name: sa.
- * Enter Password: GlobalGuideLine.

Click the OK button to continue. You should see a list of tables available for you to import from the SQL Server as shown in the picture below:

MS Access importing tables via ODBC

Select the table you are interested in and click OK. You should the selected table being imported from the SQL Server to MS Access.

[View All Answers](#)

Question - 61:

How To Configure and Test ODBC DSN Settings?

Ans:

Continue from the previous tutorial. Click Next after you have finished changing the port number. The ODBC Data Source Administrator will try to connect to the SQL server through the specified port number and login information.

If the connection is successful, you will see the next screen asking for selecting default database.

1. Check "Change the default database to:" check box, and select "GlobalGuideLineDatabase" as the default database for this DSN setting. See picture bellow:

ODBC DSN Default Database Selection

2. Click Next to see the last screen of the "Create a New Data Source to SQL Server" wizard.
3. Change nothing on the last screen and click Finish button. The confirmation screen shows up.
4. Click "Test Data Source..." button. You should see the test result as:

Attempting connection

Connection established

Verifying option settings

Disconnecting from server

TESTS COMPLETED SUCCESSFULLY!

Your ggl_SQL_SERVER ODBC DSN is ready to use.

[View All Answers](#)

Question - 62:

How To Configure ODBC DSN with Different Port Numbers?

Ans:

If your SQL Server is not using the default port number, like 1269, you need to set the port number to the correct value during the ODBC DSN creation process, as show in this tutorial:

1. Start ODBC Data Source Administrator and click System DSN tab.
2. Click Add button, select SQL Server and click Finish button.
3. Enter the following and click Next:

Name: ggl_SQL_SERVER



Description: GlobalGuideLine.com SQL Server

Server: LOCALHOST

4. Select the radio button on "With SQL Server authentication using a login ID and password entered by the user. Also enter in Login ID field: "sa" and in Password field: "GlobalGuideLine".

5. Click "Client Configuration..." button. The "Edit Network Library Configuration" dialog box shows up.

Check and uncheck "Dynamically determine port" and enter "1269" as the Port Number. See picture below:

ODBC DSN Port Number Setting

Click OK and continue with the next tutorial to finish up ODBC DSN creation.

[View All Answers](#)

Question - 63:

How To Verify the Port Number of the SQL Server?

Ans:

When applications use TCP/IP for network communication, you need to know the port number where the server is listening for connect request from the client.

If you want to connect to the SQL Server through the TCP/IP, you must know on which port number the SQL Server is listening for connection requests. Follow this tutorial to find and verify the port number of your SQL Server.

1. Go to Start > All Programs > Microsoft SQL Server 2005 > Configuration Tools > SQL Server Configuration Manager.

2. Double click on "SQL Server 2005 Network Configuration" to see the list of SQL Server instances.

3. Click on "Protocols for SQLEXPRESS". You will see a list of protocols.

4. Right-mouse click on TCP/IP and select the "Properties" command. The properties dialog box shows up.

Click the IP Address tab. The server IP address and port number used by the SQL Server will be displayed. You should see something like:

IP Address: 127.0.0.1

Ports: 1269

Note that 1433 may also be used as the default port number by your SQL Server.

[View All Answers](#)

Question - 64:

How To Enable TCP/IP Protocol on a SQL Server?

Ans:

By default, the TCP/IP protocol is turned off when a SQL Server is installed to reduce security risk. But if you want applications to connect and access the SQL Server, you need to enable the TCP/IP protocol on the server by following this tutorial:

1. Go to Start > All Programs > Microsoft SQL Server 2005 > Configuration Tools > SQL Server Configuration Manager.

2. Double click on "SQL Server 2005 Network Configuration" to see the list of SQL Server instances.

3. Click on "Protocols for SQLEXPRESS". You will see a list of protocols.

4. Right-mouse click on TCP/IP and select the "Enable" command. See the picture below:

TCP/IP Protocol Enabled

5. Click on "SQL Server 2005 Services". You will see two services.

6. Right-mouse click on "SQL Server (SQLEXPRESS)" and select restart to finish the TCP/IP protocol setting change.

Your SQL Server is ready to accept network connection now.

[View All Answers](#)

Question - 65:

How To Start SQL Server Browser Service?

Ans:

SQL Server Browser Service is installed as part of the SQL Server. But it is turned off by default to reduce the security risk. If you want start SQL Server Browser Service to allow the SQL Server to accept network connections, you need to follow the steps below:

1. Go to Control Panel > Administrative Tools.

2. Double click on "Services". The Services window shows up.

3. Double click on "SQL Server Browser". The properties dialog box shows up.

4. Change the "Startup Type" from Disabled to Automatic. Then click the Start button.

The "SQL Server Browser" service should be running now.

[View All Answers](#)

Question - 66:

What Are the Requirements on SQL Server Network Connections?

Ans:

By default, SQL Server 2005 Express Edition is installed only one connection protocol enabled:

* Shared Memory - SQL Server connection protocol for applications that are running on the same machine where the SQL Server is running. For example, if you are running SQLCMD tool on the SQL Server machine, it will use the "Shared Memory" protocol.

If you want application that are running remotely to connect and access the SQL Server you need enable the "TCP/IP" protocol.

For applications that are running on the same machine as the SQL Server, but they do not support "Shared Memory" protocol, you also need to enable the "TCP/IP" protocol for those applications. ODBC Manager is an application that requires the "TCP/IP" protocol to connect to the SQL Server.

In order for the SQL Server to accept network connections, you also need to run SQL Server Browser Service on the SQL Server machine.

In summary, there are two requirements for a SQL Server to accept network connections:

* Start SQL Server Browser Service on the SQL Server machine.

* Enable the TCP/IP protocol support on the SQL Server.

[View All Answers](#)

Question - 67:

Why Are You Getting Errors When Creating a New ODBC DSN?

Ans:



Continue from the previous tutorial. After clicking Next on the SQL login information screen, ODBC manager will try to connect to the SQL Server with the login information you provided.

After a period of waiting time, you may get error message box saying that:

Connection failed:

SQLState: '01000'

SQL Server Error: 53

[Microsoft][ODBC SQL Server Driver][DBNETLIB]

ConnectionOpen(Connect()).

Connection failed:

SQLState: '08001'

SQL Server Error: 17

[Microsoft][ODBC SQL Server Driver][DBNETLIB]

SQL Server does not exist or access denied.

See the picture below:

ODBC DSN Connection Failed

Three possible reasons for the failing:

* Wrong server name - You provided an incorrect server name.

* SQL Server not configured to take a network connection - You need to check the SQL Server configuration.

* Wrong login name or password - You provided incorrect login name or password.

The first and third reasons are easy to validate and correct. The second reason requires further investigation. Continue with the next tutorial to configure your SQL Server to take a network connection.

[View All Answers](#)

Question - 68:

How To Provide Login Information for a New ODBC DSN?

Ans:

Continue from the previous tutorial. After clicking Next on the first screen of the "Create a New Data Source to SQL Server" wizard, you should see the second screen asking you to select SQL Server login type, login name and password.

Select the radio button on "With SQL Server authentication using a login ID and password entered by the user. Also enter in Login ID field: "sa" and in Password field: "GlobalGuideLine". See the picture below:

ODBC DSN Login Information

Remember that "sa" is the system administrator login name, you probably should use a less privileged login name here. "GlobalGuideLine" must be the correct password defined in the SQL Server for "sa" login name.

Click Next and continue with the next tutorial to finish up creating a new DSN.

[View All Answers](#)

Question - 69:

How To Define the Name and Server for a new DSN?

Ans:

Continuing from the previous tutorial, on the first screen of the "Create a New Data Source to SQL Server" wizard, you should enter 3 fields: Name, Description, and Server as suggested below:

Name: ggl_SQL_SERVER

Description: GlobalGuideLine.com SQL Server

Server: LOCALHOST

See the picture below:

ODBC DSN setting for SQL Server

Note that the name and description can be decided by yourself. But the server must be the network machine name where the SQL Server is running. In this example, LOCALHOST is the machine name for your local system.

Click Next and continue with the next tutorial to finish up creating a new DSN.

[View All Answers](#)

Question - 70:

How To Add a New DSN with the ODBC Driver for SQL Server?

Ans:

Assuming that the ODBC driver for SQL Server has been installed as part of the Windows system, the next step of setting up ODBC connection to SQL Server is to create a new DSN (Data Source Name) with the ODBC Data Source Administrator:

* Go to Control Panel.

* Go to Administrative Tools.

* Run Data Sources (ODBC). The ODBC Data Source Administrator window shows up.

* Go to System DSN tab.

* Click the Add button.

You should a list of all ODBC drivers installed on your system. SQL Server ODBC driver should be on the list. If you look at the picture below, you will see the SQL Server ODBC driver with version 2000.85.1117.00, file SQLSRV32.DLL, date 8/4/2004:

ODBC Driver for SQL Server

Select "SQL Server" from the ODBC driver list, and click "Finish" button. The "Create a New Data Source to SQL Server" wizard window shows up. Continue with the next tutorial to finish up creating a new DSN.

[View All Answers](#)

Question - 71:

What Is Open Database Communication (ODBC)?

Ans:

ODBC, Open Database Communication, a standard API (application program interface) developed by Microsoft for Windows applications to communicate with database management servers.



If you want to access a database server through an ODBC driver from an application program, you need to meet the following requirements:

- * An ODBC driver specifically designed for the database server.
- * An ODBC DSN (Data Source Name) - an ODBC configuration representing the ODBC driver and database server.
- * An ODBC API (Application Programming Interface) for your application program to interact with database server through the ODBC driver.

For example, if you want to a SQL Server in a PHP script through an ODBC driver, you need to make sure that:

- * An ODBC driver for MS SQL Server - The ODBC driver is installed as part of the Windows system.
- * An ODBC DSN - You need to create one yourself using the ODBC configuration tool.
- * An ODBC API in PHP language - The API is installed as part of the PHP engine.

[View All Answers](#)

Question - 72:

How To Delete an Existing Database User?

Ans:

If you don't want to keep a database user any more, you should delete the user by using the "DROP USER" statement. This tutorial exercise shows how to delete "Dba_User":

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
DROP USER Dba_User;
GO
-- List all user names
SELECT name, sid, type, type_desc
FROM sys.database_principals WHERE type = 'S';


```

name sid type type_desc

dbo 0x01 S SQL_USER
guest 0x00 S SQL_USER
INFORMATION_SCHEMA NULL S SQL_USER
sys NULL S SQL_USER
User "Dba_User" has been deleted now.
```


```

[View All Answers](#)

Question - 73:

How To Change the Name of a Database User?

Ans:

If you want to change the name of an existing database user, you can use the "ALTER USER" statement as shown in the tutorial exercise below:

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
ALTER USER ggl_User WITH NAME = Dba_User;
GO
-- List all user names
SELECT name, sid, type, type_desc
FROM sys.database_principals WHERE type = 'S';
GO


```

name sid type type_desc

dbo 0x01 S SQL_USER
guest 0x00 S SQL_USER
INFORMATION_SCHEMA NULL S SQL_USER
sys NULL S SQL_USER
Dba_User 0x5EB8701EAEBAA74F86F... S SQL_USER
The of the last user changed from "ggl_User" to "Dba_User".
```


```

[View All Answers](#)

Question - 74:

How To Verify a User name with SQLCMD Tool?

Ans:

The quickest way to verify a user name in a database is probably to use the SQLCMD tool. You can connect to the server, select the database, and check which user name is linked the current login name as shown below.

Start a command window and enter the following command:

```
C:>SQLCMD -S LOCALHOSTSQUEXPRESS -U ggl_Login -P IYF
1> USE GlobalGuideLineDatabase;
2> GO
Changed database context to 'GlobalGuideLineDatabase'.
1> PRINT User_Name();
2> GO
ggl_User
This shows user "ggl_User" in database "GlobalGuideLineDatabase" is linked to login name "ggl_Login".
```

[View All Answers](#)

Question - 75:

How To Find the Login Name Linked to a Given User Name?

Ans:



If you know a user name in a database and you want to find out which login name is linked this user name, you need to check the Security ID (SID) of the user name based on the following rules:

- * Each login name is associated a unique SID.
- * When a user name is linked to a login name, the login name's SID is copied to the user name.

So the login name linked to a user name must have the SID as the user name. The tutorial exercise below shows you how to find the login name that is linked to the user name "ggl_User":

```
-- Login with sa
USE GlobalGuideLineDatabase;
GO


```

SELECT u.name AS User_Name, l.name AS Login_Name, u.sid
FROM sys.server_principals l,
 sys.database_principals u
WHERE l.sid = u.sid
 AND u.name = 'ggl_User';
GO
User_Name Login_Name sid

ggl_User ggl_Login 0x5EB8701EAEBAA74F86FCF5BD8E37B8C5</pre>
(1 row(s) affected)
```


```

[View All Answers](#)

Question - 76:

How To List All User Names in a Database?

Ans:

If you want to see a list of all user names defined in a database, you can use the system view, sys.database_principals as shown in this tutorial exercise:

```
-- Login with sa
-- Select a database
USE GlobalGuideLineDatabase;
GO
-- List all user names
SELECT name, sid, type, type_desc
FROM sys.database_principals WHERE type = 'S';
GO


```

name sid type type_desc

dbo 0x01 S SQL_USER
guest 0x00 S SQL_USER
INFORMATION_SCHEMA NULL S SQL_USER
sys NULL S SQL_USER
ggl_User 0x5EB8701EAEBAA74F86F... S SQL_USER</pre>
(5 row(s) affected)
```


```

As you can see, there are 5 user names defined in "GlobalGuideLineDatabase". 4 are defined by SQL Server during the database creation process. Only the last one "ggl_User" was defined by you in the previous tutorial.

[View All Answers](#)

Question - 77:

How To Create a User Name in a Database?

Ans:

User names are security principals at the database level. If you want to allow a login name to access a specific database, you need to create a user name in that database and link it to the login name.

Creating a user name can be done by using the "CREATE USER" statement as shown in this tutorial exercise:

```
-- Login with "sa"
-- Create a login
CREATE LOGIN ggl_Login WITH PASSWORD = 'TYF'
GO
-- Select a database
USE GlobalGuideLineDatabase;
GO
-- Create a user and link it to a login
CREATE USER ggl_User FOR LOGIN ggl_Login;
GO
Login name "ggl_Login" should be able to access database "GlobalGuideLineDatabase" through user name "ggl_User".
```

[View All Answers](#)

Question - 78:

How To Delete a Login Name in MS SQL Server?

Ans:

If you don't want to keep a login name any more, you should delete it by using the "DROP LOGIN" statement as shown in this tutorial example:

```
-- Login with "sa"
DROP LOGIN Db_Login;
GO
Command(s) completed successfully.
-- View login names
SELECT name, sid, type, type_desc FROM sys.server_principals
WHERE type = 'S';
```



```
GO
<pre>name      sid                type type_desc
-----
sa      0x01                S  SQL_LOGIN
ggl_DBA 0x348AF32B3B58CB40B67A5F5B7086B96E S  SQL_LOGIN</pre>
```

[View All Answers](#)

Question - 79:

How To Disable a Login Name in MS SQL Server?

Ans:

If you want temporarily disable a login name, you can use the "ALTER LOGIN" statement with a DISABLE keyword. If you want to enable it later on, you can use the ENABLE keyword. The tutorial exercise below shows how to disable and enable login name "Dba_Login":

```
-- Login with "sa"
-- Disable a login
ALTER LOGIN ggl_Login DISABLE;
-- View login status
SELECT name, type, type_desc, is_disabled
FROM sys.server_principals
WHERE type = 'S';
GO
name      type type_desc is_disabled
-----
sa      S  SQL_LOGIN 0
ggl_DBA  S  SQL_LOGIN 0
Dba_Login S  SQL_LOGIN 1
Now try to login with the disabled login name:
C:>SQLCMD -S LOCALHOSTSQLEXPRESS -U Dba_Login -P IYF
Msg 18470, Level 14, State 1, Server LOCALHOSTSQLEXPRESS
Login failed for user 'Dba_Login'. Reason: The account is
disabled.
```

C:>
Run the statements below to enable login name "Dba_Login":

```
-- Login with "sa"
-- Enable a login
ALTER LOGIN ggl_Login ENABLE;
```

[View All Answers](#)

Question - 80:

How To Change a Login Name in MS SQL Server?

Ans:

If you want to change a login name, you can use the "ALTER LOGIN" statement as shown in this tutorial example:

```
-- Login with "sa"
-- Change login name
ALTER LOGIN ggl_Login WITH NAME = Dba_Login;
GO
-- View login names
SELECT name, sid, type, type_desc FROM sys.server_principals
WHERE type = 'S';
GO
<pre>name      sid                type type_desc
-----
sa      0x01                S  SQL_LOGIN
ggl_DBA 0x348AF32B3B58CB40B67A5F5B7086B96E S  SQL_LOGIN
Dba_Login 0x5EB8701EAEBA74F86FCF5BD8E37B8C5 S  SQL_LOGIN</pre>
```

[View All Answers](#)

Question - 81:

How To Change the Password of a Login Name in MS SQL Server?

Ans:

If a developer lost the password of his or her login name, you can reset the password with the "ALTER LOGIN" statement as shown in this tutorial example:

```
-- Login with sa
ALTER LOGIN ggl_DBA WITH PASSWORD = 'globalguideline';
GO
Command(s) completed successfully.
ggl_DBA's password is changed now. Try it with the SQLCMD tool:
C:>SQLCMD -S LOCALHOSTSQLEXPRESS -U ggl_DBA -P globalguideline
1> QUIT
```

You can also use the "ALTER LOGIN" to change your own password.

[View All Answers](#)

Question - 82:

How To List All Login Names on the MS SQL Server?

Ans:



If you want to see a list of all login names defined on the server, you can use the system view, sys.server_principals as shown in this tutorial exercise:

```
-- Login with sa
SELECT name, sid, type, type_desc FROM sys.server_principals
WHERE type = 'S';
GO


```
name sid type type_desc

sa 0x01 S SQL_LOGIN
ggl_DBA 0x348AF32B3B58CB40B67A5F5B7086B96E S SQL_LOGIN
ggl_Login 0x5EB8701EAEBA74F86FCF5BD8E37B8C5 S SQL_LOGIN
```



So you have two login names on the server at this moment.


```

[View All Answers](#)

Question - 83:

How To Verify a Login name with SQLCMD Tool?

Ans:

The quickest way to verify a login name on a SQL Server is probably to use the SQLCMD tool, which takes the server name, login name and password in a single command line. Here is how to try it yourself:

Start a command window and enter the following command:

```
C:>SQLCMD -S LOCALHOSTSQLEXPRESS -U ggl_DBA -P ABD_LGG
```

```
1> PRINT Suser_Sname();
```

```
2> GO
```

```
ggl_DBA
```

```
1> QUIT
```

```
C:>
```

This shows ggl_DBA is a valid login and the password is correct. Now try this command:

```
C:>SQLCMD -S LOCALHOSTSQLEXPRESS -U ggl_DEV -P ABD_LGG
```

```
Msg 18456, Level 14, State 1, Server LOCALHOSTSQLEXPRESS, Line 1
```

```
Login failed for user 'ggl_DEV'.
```

```
C:>
```

This shows ggl_DEV is not a valid login or the password is incorrect.

[View All Answers](#)

Question - 84:

How To Create a New Login Name in MS SQL Server?

Ans:

In previous tutorials, it is assumed that you use the "sa" (System Administrator) login name to connect to your SQL Server. But that is not what a normal developer uses to connect to the server, since "sa" has the ALL permissions granted. You need to create new login names and grant less permissions to them, and give them to developers.

To create a new login name, you can use the "CREATE LOGIN" statement in a simple syntax like this:

```
CREATE LOGIN login_name WITH PASSWORD = 'password'
```

To run "CREATE LOGIN" statement, you need to connect to the server with a privileged login name like "sa". See the tutorial example below:

```
-- Login with 'sa'
```

```
-- Create new login names
```

```
CREATE LOGIN ggl_DBA WITH PASSWORD = 'ABD_LGG'
```

```
GO
```

```
Command(s) completed successfully.
```

```
CREATE LOGIN ggl_Login WITH PASSWORD = 'LGG'
```

```
GO
```

[View All Answers](#)

Question - 85:

What Is the Security Principal at the Database Level That Represents Your Session?

Ans:

Security principal identifies who you are when you interact with the SQL Server. What can do you at the database solely depends on the security principal that represents you. So it is very important to know your security principal at the database level.

What is the database level security principal of your session? The answer is simple - the user name in the current database that has been mapped to your login name.

For example, if you connects to the server with the login name "sa", and sets "GlobalGuideLineDatabase" as the current database, SQL Server will map "sa" to a user name defined in "GlobalGuideLineDatabase" based on some logics. That mapped user name is your security principal for "GlobalGuideLineDatabase" at the database level.

On a Unix system, there is a nice command called "whoami" which returns your security principal on the system. SQL Server provides a similar function called User_Name(), which returns your security principal (a user name mapped to the login name) at the database level. Try it by following this tutorial example:

```
PRINT Suser_Sname();
```

```
GO
```

```
sa
```

```
USE GlobalGuideLineDatabase;
```

```
GO
```

```
-- Find out your security principal at the database level
```

```
PRINT User_Name();
```

```
GO
```

```
dbo
```

This example shows you that "dbo" is user name in "GlobalGuideLineDatabase" that mapped to

[View All Answers](#)

Question - 86:



What Is the Security Principal at the Server Level That Represents Your Session?

Ans:

Security principal identifies who you are when you interact with the SQL Server. What can do you at the server level solely depends on the security principal that represents you. So it is very important to know your security principal at the server level.

What is the server level security principal of your session? The answer is simple - the login name you used to connect to the server to start the session. For example, if you connects to the server with the login name "sa", then your security principal is "sa" at the server level.

On a Unix system, there is a nice command called "whoami" which returns your security principal on the system. SQL Server provides a similar function called Suser_Sname(), which returns your security principal (a login name) at the server. Try it by following this tutorial example:

-- Find out your security principal at the server level

```
PRINT Suser_Sname();
```

```
GO
```

```
sa
```

This example shows you that you used "sa" as the login name to connect to the server.

[View All Answers](#)

Question - 87:

What Are Security Principals Used in SQL Server 2005?

Ans:

SQL Server 2005 supports several basic security principals located at different levels:

- * Windows-Level Principals: Windows Local Login and Windows Network Domain Login - Used to control accesses to SQL Server instances.

- * SQL Server-Level Principal: SQL Server Login. - Used to control accesses to SQL Server instances.

- * Database-Level Principal: Database User. - Used to control accesses to database instances.

To access a SQL Server instance, you must use a Windows login or a SQL Server login previously created in that server instance. Once you logged in, you are represented by this login name as your security principal at the server level.

If you select a specific database in the server to use, SQL Server will search that database for a previously user that has been mapped to your login name. If that user is located, you are represented by this user name as your security principal at the database level.

[View All Answers](#)

Question - 88:

What Is the Security Model Used in SQL Server 2005?

Ans:

SQL Server 2005 uses a very standard security model involves 3 concepts:

- * Securables - Entities representing resources that need to be secured. For example, a database table is a securable.

- * Principals - Entities representing users that request accesses to resources. For example, a login user is a principal.

- * Permissions - Types of accesses associated with securables. Permissions can be granted to or revoked from principals. For example, "Update" is a permission associated a securable, table "R". "Update" on "R" can be granted to a principal, user "U". Now user "U" will get "Update" access on table "R".

In simple terms, a principal answers the security question of "Who are you?"; a securable answers the security question of "What is your target object?"; a permission answers the security question of "You are allowed to perform this action on this target object".

SQL Server 2005 supports multiple securables and multiple principals organized into hierarchical structures.

[View All Answers](#)

Question - 89:

How To Replace Given Values with NULL using NULLIF()?

Ans:

Sometime you want to hide certain values by replacing them with NULL values. SQL Server offers you a nice function called NULLIF() to do this:

NULLIF(expression, value)

-- Returns NULL if "expression" equals to value"

-- Returns "expression", otherwise

NULLIF() can be viewed as the reverse function of ISNULL(). The tutorial script below shows you a good example of using NULLIF():

```
USE GlobalGuideLineDatabase;
```

```
GO
```

```
SELECT id, counts FROM ggl_links;
```

```
GO
```

```
<pre>id      counts
```

```
-----
```

```
101      NULL
```

```
102      8
```

```
1101     NULL
```

```
202      NULL
```

```
2101     NULL
```

```
2102     NULL
```

```
301      NULL
```

```
302      NULL</pre>
```

-- converting NULL to 0

```
UPDATE ggl_links SET counts=ISNULL(counts,0);
```

```
GO
```

[View All Answers](#)

Question - 90:

How To Replace NULL Values in Expressions using ISNULL()?

Ans:

As you learned from previous tutorials, NULL values presented in expressions will cause the final results to be NULL. Sometimes, you want NULL values to be



replaced with some default values, like 0, "", or 'NULL', so that expressions can be evaluated properly.
SQL Server offers a built-in function called ISNULL() to help you replacing NULL values in expressions:
ISNULL(expression, replacement)

-- Returns "expression", if it is not NULL

-- Returns "replacement", if "expression" is NULL

The tutorial example below shows you how to replace possible NULL values in @middle_initial:

USE GlobalGuideLineDatabase;

GO

CREATE PROCEDURE welcome

@first_name VARCHAR(20),

@middle_initial VARCHAR(1),

@last_name VARCHAR(20)

AS

PRINT 'Hello '+@first_name

+ ' '+@middle_initial

+ ' '+@last_name;

PRINT 'Hello '+@first_name

+ ' '+ISNULL(@middle_initial,"")

+ ' '+@last_name;

GO

EXEC welcome 'John', 'W', 'King';

GO

Hello John W King

Hello John W King

EXEC welcome 'John', NULL, 'King';

GO

Hello John King

The first PRINT statement in the second test returns a blank line because

[View All Answers](#)

Question - 91:

What Happens If NULL Values Are Involved in Boolean Operations?

Ans:

If NULL values are involved in Boolean operations, the result will vary depending on the operator type. For AND operator, FALSE takes precedence over NULL.

The result can be summarized in a table below:

```
<pre>AND  TRUE  FALSE  NULL
```

```
TRUE  true   false  null
```

```
FALSE  false  false  false
```

```
NULL   null   false  null</pre>
```

For OR operator, TRUE takes precedence over NULL. The result can be summarized in a table below:

```
<pre>OR    TRUE  FALSE  NULL
```

```
TRUE  true   true   true
```

```
FALSE  true   false  null
```

```
NULL   true   null   null</pre>
```

The tutorial script below shows you that NULL AND FALSE returns FALSE:

IF 0=NULL AND 0=1 PRINT 'NULL AND FALSE returns TRUE'

ELSE PRINT 'NULL AND FALSE does not returns TRUE'

GO

NULL AND FALSE does not returns TRUE

IF NOT (0=NULL AND 0=1)

PRINT 'NULL AND FALSE returns FALSE'

ELSE PRINT 'NULL AND FALSE does not returns FALSE'

GO

NULL AND FALSE returns FALSE

[View All Answers](#)

Question - 92:

What Happens If NULL Values Are Involved in Comparison Operations?

Ans:

If NULL values are involved in comparison operations, the result will be Boolean NULL values. This behavior is very interesting because you would expect a comparison operation returns only one of the two values: TRUE and FALSE. But SQL Server may return you a third value: NULL.

The tutorial script below proves that "1>NULL" is not returning TRUE or FALSE. It is returning a third value, NULL:

IF 1>NULL PRINT '1>NULL is returning TRUE'

ELSE PRINT '1>NULL is not returning TRUE'

GO

1>NULL is not returning TRUE

IF NOT 1>NULL PRINT '1>NULL is returning FALSE'

ELSE PRINT '1>NULL is not returning FALSE'

GO

1>NULL is not returning FALSE

Another test proves that "'GGL'=NULL" is not returning TRUE or FALSE. It is returning a third value, NULL:

IF 'GGL'=NULL PRINT "'GGL'=NULL returns TRUE'

ELSE PRINT "'GGL'=NULL does not return TRUE'

GO

'GGL'=NULL does not return TRUE

IF NOT 'GGL'=NULL PRINT "'GGL'=NULL returns FALSE'

ELSE PRINT "'GGL'=NULL does not return FALSE'



GO
'GGL'=NULL does not return FALSE

[View All Answers](#)

Question - 93:

What Happens If NULL Values Are Involved in Bitwise Operations?

Ans:

If NULL values are involved in bitwise operations, the result will be binary NULL values. The following tutorial script shows you some good examples:

```
SELECT 1 | NULL;
GO
-----
NULL
SELECT 707 & NULL;
GO
-----
NULL
SELECT ~NULL;
GO
-----
NULL
```

[View All Answers](#)

Question - 94:

What Happens If NULL Values Are Involved in Datetime Operations?

Ans:

If NULL values are involved in datetime operations, the result will be datetime NULL values. The following tutorial script shows you some good examples:

```
USE GlobalGuideLineDatabase;
GO
SELECT GETDATE()+NULL;
GO
-----
NULL
SELECT DATEDIFF(DAY, GETDATE(), NULL);
GO
-----
NULL
```

[View All Answers](#)

Question - 95:

What Happens If NULL Values Are Involved in String Operations?

Ans:

If NULL values are involved in string operations, the result will be string NULL values. The following tutorial script shows you some good examples:

```
SELECT 'GlobalGuideLine'+NULL;
GO
-----
NULL
SELECT LEN(NULL);
GO
-----
NULL
SELECT REVERSE(NULL);
GO
-----
NULL
```

[View All Answers](#)

Question - 96:

What Happens If NULL Values Are Involved in Arithmetic Operations?

Ans:

If NULL values are involved in arithmetic operations, the result will be numeric NULL values. The following tutorial script shows you some good examples:

```
SELECT 7+NULL;
GO
-----
NULL
SELECT 10.02*NULL;
GO
-----
NULL
SELECT 4.988E+10/NULL;
GO
-----
NULL
```

[View All Answers](#)



Question - 97:

How To Assign NULL Values to Variables or Columns?

Ans:

The rule for assigning NULL values to variables or table columns is simple: Use keyword "NULL" directly as normal values. Specifically,

- * "NULL" can be used in SET statements to assign NULL values to variables.
- * "NULL" can be used in SET clauses in UPDATE statements.
- * "NULL" can be used in value lists in INSERT statements.
- * "NULL" can be used in parameter lists when calling stored procedures or functions.

The tutorial script below gives you some good examples:

```
USE GlobalGuideLineDatabase;
GO
-- assign NULL values to variables
DECLARE @birth_date DATETIME;
SET @birth_date = NULL;
SELECT @birth_date;
GO
-----
NULL
-- assign NULL values to columns
UPDATE ggl_links SET notes = NULL;
GO
(8 row(s) affected)
-- assign NULL values to parameters
EXEC sp_help NULL;
GO
Name
-----
ggl_links_dump
ggl_links_top
ggl_links_view
...
```

[View All Answers](#)

Question - 98:

What Are NULL Values in MS SQL Server?

Ans:

A NULL value is a special value that represents an unknown value. SQL Server supports NULL values with the following features:

- * All data types used for table columns support NULL values. In another word, NULL values can be stored in database tables.
- * Individual table columns may be defined to not allow NULL values. In this case, you can not assign NULL values to those columns.
- * "NULL" is a keyword that represent a NULL value in expressions.
- * NULL values can be used directly in SET (assignment) statements.
- * If NULL values are involved in an arithmetic operation, the result will be a numeric NULL.
- * If NULL values are involved in a string operation, the result will be a string NULL.
- * If NULL values are involved in a datetime operation, the result will be a datetime NULL.
- * If NULL values are involved in a bitwise operation, the result will be a binary NULL.
- * If NULL values are involved in a comparison operation, the result will be a Boolean NULL.
- * If NULL values are involved in a Boolean operation, the result could be TRUE, FALSE, or NULL.
- * To test NULL values, you need to use two special operators, IS NULL and IS NOT NULL.
- * Special functions are available to handle NULL values, like ISNULL(), and NULLIF()

[View All Answers](#)

Question - 99:

How To Create a Dynamic Cursor with the DYNAMIC Option?

Ans:

If the underlying table is changed after the cursor is opened, should the changes be reflected in the cursor result set? The answer is based on the update option used when creating the cursor. SQL SERVER supports two update options:

1. STATIC - The result set will be a static copy created when the OPEN statement is executed. Subsequent updates on the underlying tables will not affect the result set. STATIC is the default option.
2. SCROLL - The result set will be dynamically updated each time when a FETCH statement is executed. In another word, the result set always reflects the latest changes on the underlying tables.

The tutorial script below gives you a good example of how dynamic cursors work:

```
USE GlobalGuideLineDatabase;
GO
DECLARE @ggl_cursor CURSOR;
SET @ggl_cursor = CURSOR FOR
    SELECT id, url, notes, counts, time FROM ggl_links;
OPEN @ggl_cursor;
DECLARE @id INT, @url VARCHAR(80), @notes VARCHAR(80),
        @counts INT, @time DATETIME;
FETCH NEXT FROM @ggl_cursor INTO @id, @url, @notes,
        @counts, @time;
```

[View All Answers](#)

Question - 100:

How To Create a Scrollable Cursor with the SCROLL Option?



Ans:

SQL Server offers two scrolling option on cursors:

1. FORWARD_ONLY - The cursor can only be scrolled forward with "FETCH NEXT" statements. In another word, you can only loop through the cursor from the first row to the last row. FORWARD_ONLY is the default option.
2. SCROLL - The cursor can be scrolled back and forth with "FETCH NEXT", "FETCH PRIOR", and other fetch options.

The tutorial example below creates a cursor with the SCROLL option so that the result set can be looped through backward:

```
DECLARE @ggl_cursor CURSOR;
SET @ggl_cursor = CURSOR SCROLL FOR
    SELECT id, url, notes, counts, time
    FROM ggl_links ORDER BY id;
OPEN @ggl_cursor;
DECLARE @id INT, @url VARCHAR(80), @notes VARCHAR(80),
    @counts INT, @time DATETIME;
FETCH LAST FROM @ggl_cursor INTO @id, @url, @notes,
    @counts, @time;
WHILE @@FETCH_STATUS = 0 BEGIN
    PRINT CONVERT(CHAR(5),ISNULL(@id,0))
    +CONVERT(CHAR(18),ISNULL(@url,'NULL'))
    +CONVERT(CHAR(20),ISNULL(@notes,'NULL'))
    +CONVERT(CHAR(4),ISNULL(@counts,0))
    +CONVERT(CHAR(11),ISNULL(@time,'2007'));
    FETCH PRIOR FROM @ggl_cursor INTO @id, @url, @notes,
        @counts, @time;
END
CLOSE @ggl_cursor;
```

[View All Answers](#)

Question - 101:

How To Declare and Use Cursor Variables?

Ans:

There are two ways to representing a cursor:

1. A cursor name - A static name representing a cursor object. A cursor name should be linked to a cursor object in the DECLARE statement.
2. A cursor variable name - A variable name pointing to a cursor object. A cursor variable name should be declared with the CURSOR data type. It should be then assigned with a cursor object using the SET statement.

The tutorial exercise below shows you how to declare a cursor variable and assign a cursor object to it:

```
USE GlobalGuideLineDatabase;
GO
-- declare a cursor variable
DECLARE @ggl_cursor CURSOR;
-- assign a cursor object
SET @ggl_cursor = CURSOR FOR
    SELECT id, url, notes, counts, time FROM ggl_links;
```

[View All Answers](#)

Question - 102:

How To Loop through the Result Set with @@FETCH_STATUS?

Ans:

The FETCH statement only returns one row from the result set. If you want to return all rows, you need to put the FETCH statement in a loop. A simple way to stop the loop to check the system variable @@FETCH_STATUS, which returns a status code of the last FETCH statement:

```
@status = @@FETCH_STATUS;
-- Returns 0, the fetch was successful
-- Returns -1, the fetch failed or end of result set reached
-- Returns -2, the row fetched is missing
```

The tutorial exercise below shows how @@FETCH_STATUS is used to make WHILE loop over the result set in a cursor:

```
USE GlobalGuideLineDatabase;
GO
```

[View All Answers](#)

Question - 103:

How To Transfer Data from a Cursor to Variables with a "FETCH" Statement?

Ans:

By default, a FETCH statement will display the fetched row on the client program window. If you want to transfer the output data to variables, you can specify an INTO clause with a list of variables that matches the list of fields in the result set.

The tutorial exercise below shows you a good example of using the FETCH statement to transfer one row of output data from the result set to variables:

```
USE GlobalGuideLineDatabase;
GO
DECLARE ggl_cursor CURSOR FOR
    SELECT id, url, notes, counts, time FROM ggl_links;
OPEN ggl_cursor;
DECLARE @id INT, @url VARCHAR(80), @notes VARCHAR(80),
    @counts INT, @time DATETIME;
FETCH NEXT FROM ggl_cursor INTO @id, @url, @notes,
    @counts, @time;
PRINT 'id = '+CONVERT(VARCHAR(20),ISNULL(@id,0));
PRINT 'url = '+ISNULL(@url,'NULL');
```



```
PRINT 'notes = '+ISNULL(@notes,'NULL');
PRINT 'counts = '+CONVERT(VARCHAR(20),ISNULL(@counts,0));
PRINT 'time = '+CONVERT(VARCHAR(20),ISNULL(@time,'2007'));
CLOSE ggl_cursor;
DEALLOCATE ggl_cursor;
GO
id = 101
url = globalguideline.com
notes = NULL
counts = 0
time = Jan 1 2007 12:00AM
```

[View All Answers](#)

Question - 104:

How To Fetch the Next Row from a Cursor with a "FETCH" Statement?

Ans:

When the result set is ready in a cursor, you can use a FETCH statement to retrieve one row from the result set in the same format as a SELECT statement. The FETCH statement has the following formats:

```
FETCH NEXT FROM cursor_name;
FETCH PRIOR FROM cursor_name;
FETCH FIRST FROM cursor_name;
FETCH LAST FROM cursor_name;
FETCH ABSOLUTE n FROM cursor_name;
FETCH RELATIVE n FROM cursor_name;
```

The tutorial exercise below shows you how FETCH statements are used to retrieve the first row and the second row back from a cursor:

```
USE GlobalGuideLineDatabase;
GO
DECLARE ggl_cursor CURSOR FOR
  SELECT * FROM ggl_links;
OPEN ggl_cursor;
FETCH NEXT FROM ggl_cursor;
FETCH NEXT FROM ggl_cursor;
CLOSE ggl_cursor;
DEALLOCATE ggl_cursor;
GO
<pre>id url          notes    counts time
-----
101 globalguideline.com NULL     NULL  NULL
(1 row(s) affected)
id url          notes    counts time
-----
102 globalguideline.com/sql Nice site. 8     NULL</pre>
(1 row(s) affected)
```

[View All Answers](#)

Question - 105:

How To Execute the Cursor Queries with "OPEN" Statements?

Ans:

Once a cursor is declared, you need to execute the query attached to the cursor so that the result set returned from the query can be accessed through the cursor. To execute the cursor query, you should use the OPEN statement as in this format:

```
OPEN cursor_name;
```

When you are done with using the result set attached to a cursor, you should close the result set to free up server resources.

The tutorial example below shows you how to open and close a cursor:

```
USE GlobalGuideLineDatabase;
GO
DECLARE ggl_cursor CURSOR FOR
  SELECT * FROM ggl_links;
OPEN ggl_cursor;
-- result set is ready to use
-- other statements
CLOSE ggl_cursor;
DEALLOCATE ggl_cursor;
GO
```

[View All Answers](#)

Question - 106:

How To Declare a Cursor with "DECLARE ... CURSOR" in MS SQL Server?

Ans:

If you want to use a cursor to represent the result set of a query, you need to define a cursor name with a SELECT sub-statement using the "DECLARE ... CURSOR" statement using the following syntax format:

```
DECLARE cursor_name CURSOR FOR
  SELECT ...;
```

Note that the DECLARE statement will not actually execute the SELECT sub-statement. It only attaches the SELECT sub-statement to the cursor.

A cursor name should be deallocated to free up server resources if the cursor is not needed any more.

The tutorial example below shows you how to declare and deallocate a cursor.

```
USE GlobalGuideLineDatabase;
```



```
GO
DECLARE ggl_cursor CURSOR FOR
  SELECT * FROM ggl_links;
-- other statements
DEALLOCATE ggl_cursor;
GO
```

[View All Answers](#)

Question - 107:

What Are Cursors in MS SQL Server?

Ans:

A cursor is a special data type that represents a result set returned by a SELECT query statement. There are several notes about cursor you need to remember:

- * Cursor data type can not be used to define table columns.
- * Cursor data type is used on store procedures, functions, and triggers to help you loop through result sets returned by queries.
- * Cursor data type can be used to define cursor variables.
- * There are special Transact-SQL statements dedicated to work with cursors variables: OPEN, FETCH, CLOSE, and DEALLOCATE.
- * Cursor variables can be passed as procedure or function parameters.
- * There is a special function, CURSOR_STATUS(), for check cursor statuses.

[View All Answers](#)

Question - 108:

Can You Create a Logon Trigger in SQL Server 2005 Express Edition?

Ans:

Can you create a logon trigger in SQL Server 2005 Express Edition? The answer is no. LOGON is not a supported event type in Express Edition. The script below shows you the error message when you try to create a logon trigger:

```
CREATE TRIGGER welcome ON ALL SERVER
AFTER LOGON
AS
  PRINT 'Welcome to GlobalGuideLineDatabase database!';
GO
Msg 1084, Level 15, State 1, Procedure welcome, Line 2
'LOGON' is an invalid event type.
Or may be there is a configuration setting that disables the logon event type.
```

[View All Answers](#)

Question - 109:

Can You Roll Back the DDL Statement in a Trigger?

Ans:

Can you roll back the DDL statement in a trigger? The answer is yes. Since the DDL statement that fires the trigger and the statements defined inside the trigger are all executed as a single statement batch, you can add a ROLLBACK statement in the trigger to rollback the entire batch.

```
USE GlobalGuideLineDatabase;
GO
CREATE TRIGGER drop_rollback ON DATABASE AFTER DROP_TABLE AS PRINT 'Drop table is not allowed!';
ROLLBACK;
GO
DROP TABLE ggl_users;
GO
Drop table is not allowed!
Msg 3609, Level 16, State 2, Line 2
The transaction ended in the trigger. The batch has been aborted.
```

This trigger is powerful. It will stop you from dropping any tables in GlobalGuideLineDatabase database.

[View All Answers](#)

Question - 110:

How To Create a DDL Trigger using "CREATE TRIGGER" Statements?

Ans:

A DDL trigger is defined to handle a DDL statement event, like create, alter and drop tables, views, indexes, etc. DDL triggers can be used to generate warning messages on database object changes. The format of creating a DDL trigger should be:

```
CREATE TRIGGER trigger_name ON DATABASE
AFTER ddl_event_types
AS
  statements
GO
-- ddl_event_types are keywords like:
-- CREATE_TABLE, ALTER_TABLE, DROP_TABLE, ...
Below is a simple example of creating a DDL trigger to generate messages on ALTER_TABLE events:
USE GlobalGuideLineDatabase;
GO
CREATE TRIGGER ddl_message ON DATABASE
AFTER ALTER_TABLE
AS
  PRINT 'Someone is changing tables!';
GO
ALTER TABLE ggl_users
  ALTER COLUMN id INT NOT NULL;
GO
Someone is changing tables!
```

[View All Answers](#)



Question - 111:

How To Override DML Statements with Triggers?

Ans:

Sometime, you may want to implement some business logics in a DML trigger to cancel the DML statement. For example, you may want to check the new email address format provided by the UPDATE statement. If the email address is invalid, you to cancel the UPDATE statement.

There is no easy way to cancel the DML statement in a DML trigger. But there is easy way to override the DML statement with an "INSTEAD OF" trigger. SQL Server supports 2 options (3 keywords) on when the defined trigger will be fired:

- * AFTER - Trigger fired after the DML statement executed successfully.

- * INSTEAD OF - Trigger fired instead of the DML statement execution, allowing the trigger to decide whether or not, and how, to execute the statement.

- * FOR - Same as AFTER.

The tutorial exercise below shows you how define an "INSTEAD OF" trigger on ggl_users to validate email addresses:

```
CREATE TRIGGER check_email ON ggl_users
INSTEAD OF UPDATE
AS
DECLARE @count INT;
SELECT @count = COUNT(*) FROM INSERTED
WHERE email NOT LIKE '%_@_%';
IF @count = 0
    UPDATE ggl_users SET email=i.email
    FROM INSERTED AS i
    WHERE ggl_users.id = i.id
ELSE
    PRINT 'Invalid email(s) found.';
GO
-- invalid email
UPDATE ggl_users SET email='john.king'
WHERE name
```

[View All Answers](#)

Question - 112:

What Happens to a Trigger with Multiple Affected Rows?

Ans:

If there is only one row affected by a DML statement, we know that the DML trigger will be executed once. But how many times the DML trigger will be executed if the DML statement resulted multiple affected rows? The answer is still one.

In the case of multiple affected rows, both INSERTED and DELETED tables will contain multiple rows.

If you want your trigger to report all affected rows, you need to write a loop

```
select * from ggl_users;
-- reporting the first affected row only
UPDATE ggl_users SET email=name;
GO
Email changed from NULL to John King
(5 row(s) affected)
-- reporting all affected rows
ALTER TRIGGER update_user ON ggl_users
AFTER UPDATE
AS
SELECT 'Email changed from '
+ ISNULL(d.email,'NULL')
+ ' to '
+ ISNULL(i.email,'NULL')
FROM INSERTED AS i, DELETED AS d
WHERE i.id = d.id;
GO
UPDATE ggl_users SET email=REVERSE(name);
GO


```
<pre>-----
Email changed from Marc Kumar to ramuK craM
Email changed from Roy Bush to hsuB yoR
Email changed from Jack Gate to etaG kcaJ
Email changed from Nancy Greenberg to grebneerG ycnaN
Email changed from John King to gniK nhoJ</pre>
(5 row(s) affected)
```


```

[View All Answers](#)

Question - 113:

How To Access the Deleted Record of an Event?

Ans:

When a DML event occurs, SQL Server will prepare a temporary table called "DELETED", which contains the old record of the affected row, which is:

- * A copy of the deleted row for a DELETE statement.

- * A copy of the row to be updated for an UPDATE statement.

- * Empty for an INSERT statement.

The tutorial exercise below shows you how to improve the trigger, update_user, to report email changes on table, ggl_users, with both old and new emails:

```
USE GlobalGuideLineDatabase;
GO
ALTER TRIGGER update_user ON ggl_users
AFTER UPDATE
AS
```



```
DECLARE @new VARCHAR(80);
DECLARE @old VARCHAR(80);
SELECT @new = email FROM INSERTED;
SELECT @old = email FROM DELETED;
PRINT 'Email changed from ' + @old + ' to ' + @new;
GO
UPDATE ggl_users SET email='king@ggl'
WHERE name = 'John King';
GO
Email changed from smith@GlobalGuideline to master@GlobalGuideline
(1 row(s) affected)
INSERTED and DELETED are working as expected. The reported message is getting better.
```

[View All Answers](#)

Question - 114:

How To Access the Inserted Record of an Event?

Ans:

When a DML event occurs, SQL Server will prepare a temporary table called "INSERTED", which contains the new record of the affected row, which is:

- * A copy of the inserted row for an INSERT statement.
- * A copy of the updated row for an UPDATE statement.
- * Empty for a DELETE statement.

The tutorial exercise below shows you how to create a trigger, update_user, to report email changes on table, ggl_users:

```
USE GlobalGuideLineDatabase;
GO
DISABLE TRIGGER dml_message ON ggl_users;
GO
CREATE TRIGGER update_user ON ggl_users
AFTER UPDATE
AS
DECLARE @new VARCHAR(80);
SELECT @new = email FROM INSERTED;
PRINT 'Email changed to ' + @new;
GO
UPDATE ggl_users SET email='smith@GlobalGuideline'
WHERE name = 'John King';
GO
Email changed to smith@GlobalGuideline
(1 row(s) affected)
```

As you can see, the INSERTED table is helpful, if you want the trigger to perform specific logics on the affected rows.

[View All Answers](#)

Question - 115:

How To See the Event List of an Existing Trigger using sys.trigger_events?

Ans:

If what are the DML events an existing trigger is handling, you can use the catalog view, sys.trigger_events. You need to join sys.trigger_events and sys.triggers to get a better list as shown in this tutorial example:

```
USE GlobalGuideLineDatabase
GO
SELECT t.name, e.type, e.type_desc
FROM sys.trigger_events AS e, sys.triggers AS t
WHERE e.object_id = t.object_id
GO


```

name type type_desc

dml_message 1 INSERT
dml_message 2 UPDATE
dml_message 3 DELETE
new_user 1 INSERT
(4 row(s) affected)

```



The list clearly shows that dml_message handles 3 events: INSERT, UPDATE and DELETE.


```

[View All Answers](#)

Question - 116:

How To Create a Trigger for INSERT Only?

Ans:

The trigger, dml_message, provided in previous tutorials was defined to handle all 3 types of DML statements, INSERT, UPDATE, and DELETE.

If you do not want the trigger to handle all 3 types of DML statements, you can list only 1 or 2 of the statement keywords. For example, the following SQL script defines a trigger that only handle the INSERT statement events:

```
USE GlobalGuideLineDatabase
GO
CREATE TRIGGER new_user ON ggl_users
AFTER INSERT
AS
PRINT 'Time: ' + CONVERT(VARCHAR(12), GETDATE())
+ ' New users added.';
GO
```



```
INSERT INTO ggl_users (name) VALUES ('Marc MHI');
GO
Time: Jul 1 2007
Records are inserted, updated, or deleted in ggl_users
Time: Jul 1 2007 New users added.
(1 row(s) affected)
```

[View All Answers](#)

Question - 117:

How To Disable Triggers using "DISABLE TRIGGER"?

Ans:

If want to stop the execution of an existing trigger temporarily, you can use the "DISABLE TRIGGER" statement to disable it. The disabled trigger will be kept in the database.

If you want to resume the execution of a disabled trigger, you can use the "ENABLE TRIGGER" statement to enable it.

The tutorial exercise below shows you how to disable and enable triggers:

```
USE GlobalGuideLineDatabase
GO
-- disabling a trigger
DISABLE TRIGGER dml_message ON ggl_users;
GO
INSERT INTO ggl_users (name) VALUES ('MHI Gate');
GO
(1 row(s) affected)
-- enabling a trigger
ENABLE TRIGGER dml_message ON ggl_users;
GO
INSERT INTO ggl_users (name) VALUES ('Roy MHI');
GO
Time: Jul 1 2007
Records are inserted, updated, or deleted in ggl_users
(1 row(s) affected)
```

[View All Answers](#)

Question - 118:

How To Get the Definition of a Trigger Back?

Ans:

If you want get the definition of an existing trigger back from the SQL Server, you can use the catalog view called sys.sql_modules, which stores definitions of views, stored procedures, and triggers.

The sys.sql_modules holds trigger definitions identifiable by the object id of each trigger. The tutorial exercise below shows you how to retrieve the definition of trigger, "dml_message" by joining sys.sql_modules and sys.triggers:

```
USE GlobalGuideLineDatabase;
GO
SELECT m.definition
FROM sys.sql_modules m, sys.triggers t
WHERE m.object_id = t.object_id
AND t.name = 'dml_message';
GO
definition
-----
CREATE TRIGGER dml_message ON ggl_users
AFTER INSERT, UPDATE, DELETE
AS
PRINT 'Time: '+CONVERT(VARCHAR(12),GETDATE());
PRINT 'Records are inserted, updated,'
+ ' or deleted in ggl_users';
(1 row(s) affected)
```

[View All Answers](#)

Question - 119:

How To Delete Existing Triggers using "DROP TRIGGER"?

Ans:

If you don't want to use a trigger any more, you should delete it from the database by using the "DROP TRIGGER" statement as shown in tutorial example:

```
USE GlobalGuideLineDatabase;
GO
DROP TRIGGER new_user;
GO
SELECT * FROM sys.triggers
GO


```
name object_id parent_id type type_desc

dml_message 690101499 674101442 TR SQL_TRIGGER
Trigger, new_user, is deleted now.
```


```

[View All Answers](#)

Question - 120:



How To Modify Existing Triggers using "ALTER TRIGGER"?

Ans:

If you want to make changes to an existing trigger, you could use the "ALTER TRIGGER" statements to refine the trigger again. The tutorial exercise below shows you how to modify the trigger defined in a previous tutorial:

```
USE GlobalGuideLineDatabase;
GO
ALTER TRIGGER dml_message ON ggl_users
AFTER INSERT, UPDATE, DELETE
AS
    PRINT 'Time: '+CONVERT(VARCHAR(12),GETDATE());
    PRINT 'Records are inserted, updated,'
    + ' or deleted in ggl_users';
GO
UPDATE ggl_users SET email='john@ggl' WHERE id = 1;
GO
Time: Jul 1 2007
Records are inserted, updated, or deleted in ggl_users
An extra printing statement is added the trigger.
```

[View All Answers](#)

Question - 121:

How To List All Triggers in the Database with sys.triggers in MS SQL Server?

Ans:

If you want to list all triggers defined in the current database, you can use the catalog view, sys.triggers, as shown in the following tutorial example:

```
USE GlobalGuideLineDatabase;
GO
CREATE TRIGGER new_user ON ggl_users
AFTER INSERT
AS
    PRINT 'New users added.';
GO
SELECT * FROM sys.triggers
GO


```
name object_id parent_id type type_desc

dml_message 690101499 674101442 TR SQL_TRIGGER
new_user 706101556 674101442 TR SQL_TRIGGER
```



The result shows that there are 2 triggers defined in GlobalGuideLineDatabase.


```

[View All Answers](#)

Question - 122:

How To Test a DML Trigger in MS SQL Server?

Ans:

To test a DML trigger defined on a table, you just need to execute several INSERT, UPDATE and DELETE statements on that table as shown in this tutorial example:

```
USE GlobalGuideLineDatabase;
GO
INSERT INTO ggl_users (name) VALUES ('GGL Admin');
GO
Records are inserted, updated, or deleted in ggl_users
(1 row(s) affected)
UPDATE ggl_users SET email='root@ggl'
WHERE name = 'GGL Admin';
GO
Records are inserted, updated, or deleted in ggl_users
(1 row(s) affected)
DELETE FROM ggl_users WHERE name = 'GGL Admin';
GO
Records are inserted, updated, or deleted in ggl_users
(1 row(s) affected)
The trigger, dml_message, is working as expected.
```

[View All Answers](#)

Question - 123:

How To Create a DML Trigger using CREATE TRIGGER Statements?

Ans:

A DML trigger is a trigger declared to handle a DML event, which occurs when an INSERT, UPDATE or DELETE statement is executed. If you want to create a DML trigger, you should use the "CREATE TRIGGER" statement in the following format:

```
CREATE TRIGGER trigger_name ON table_name
AFTER INSERT, UPDATE, DELETE
AS
    statements
GO
```

The tutorial exercise below shows you a very simple DML trigger defined on the ggl_users table. It does nothing but printing a simple static message.

USE GlobalGuideLineDatabase;



```
GO
CREATE TRIGGER dml_message ON ggl_users
AFTER INSERT, UPDATE, DELETE
AS
    PRINT 'Records are inserted, updated,'
    + ' or deleted in ggl_users';
GO
Command(s) completed successfully.
A simple DML trigger is defined on ggl_users now.
```

[View All Answers](#)

Question - 124:

How To Create a Simple Table to Test Triggers in MS SQL Server?

Ans:

If you want to follow other tutorial examples included in this collection, you need to run this SQL script to create a simple table called ggl_users:

```
USE GlobalGuideLineDatabase;
GO
DROP TABLE ggl_users;
GO
CREATE TABLE ggl_users (
    id INTEGER IDENTITY NOT NULL,
    name VARCHAR(80) NOT NULL,
    email VARCHAR(80) NULL,
    password VARCHAR(32) NULL
);
INSERT INTO ggl_users (name) VALUES ('John King');
INSERT INTO ggl_users (name) VALUES ('Nancy Greenberg');
GO
ggl_users is created now with 2 records.
```

[View All Answers](#)

Question - 125:

What Are the Basic Features of a Trigger in MS SQL Server?

Ans:

Since a SQL Server trigger is a really an event handler, it has the following basic features similar to event handlers in other programming languages:

- * Event Type - It must be declared to handle a specific event, like a DELETE event.
 - * Object Scope - It must be declared to handle events in a specific database object scope, like a specific table.
 - * Statement Body - It must have a statement body, a batch of statements to be executed when the specified event occurs in specified database object scope. An event handler (trigger) with an empty statement body is useless.
 - * Access of Event Attributes - It must have access to some attributes of the event, so it can have different logics for different instances of the event.
- For example, you can implement a trigger to send a security alert message to each user whenever his or her password is changed. This trigger should have the following features:
- * Event Type - It must be declared to handle the UPDATE event.
 - * Object Scope - It must be declared to handle the UPDATE event on the user password table only.
 - * Statement Body - It must have a statement body to determine if the password is really changed or not. If it is changed, send an email to the user's email address.
 - * Access of Event Attributes - It must have access to some attributes of the event instance, like the old value and the new value of the password, and the user email address.

[View All Answers](#)

Question - 126:

What Are Triggers in MS SQL Server?

Ans:

A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server. A trigger is really an event handler. SQL Server allows users to create triggers (event handlers) for 3 types of events:

- * DML Event - Occurs when a DML (Data Manipulation Language) statement: INSERT, UPDATE or DELETE, is executed.
- * DDL Event - Occurs when a DDL (Data Definition Language) statement: CREATE, ALTER, or DROP, is executed.
- * Logon Event - Occurs when a user logs in to the Server.

There are 3 different types of triggers (event handlers) based on the types of events they are triggered by:

- * DML Trigger - Executes in response to a DML event.
- * DDL Trigger - Executes in response to a DDL event.
- * Logon Trigger - Executes in response to a logon event.

[View All Answers](#)

Question - 127:

How To Drop an Existing Schema in MS SQL Server?

Ans:

If you want to delete a schema, you need to move all objects out of that schema, then use the "DROP SCHEMA" statement to delete the schema. The tutorial exercise below shows you how to drop schema "ggl":

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
-- Drop failed because schema is not empty
DROP SCHEMA ggl;
```



```
GO
Msg 3729, Level 16, State 1, Line 1
Cannot drop schema 'ggl' because it is being referenced
  by object 'DF__ggl_links__creat__4316F928'.
-- Move one table out
ALTER SCHEMA dbo TRANSFER ggl.ggl_links;
GO
-- Delete one table
DROP TABLE ggl.test;
GO
-- Dropped ok
DROP SCHEMA ggl;
GO
Command(s) completed successfully.
```

[View All Answers](#)

Question - 128:

What Happens If You Are Trying to Access a Schema Not Owned by You?

Ans:

In general, if you are trying to access an object in schema owned by another database user, you will get a "permission denied" error, unless that you have been granted access permission to that object explicitly. Here is a tutorial example showing you the permission error:

```
-- Login with "ggl_login"
USE GlobalGuideLineDatabase;
GO
PRINT User_Name();
GO
ggl_User
SELECT COUNT(*) FROM dbo.ggl_random;
GO
Msg 229, Level 14, State 5, Line 1
SELECT permission denied on object 'ggl_random',
  database 'GlobalGuideLineDatabase', schema 'dbo'.
```

[View All Answers](#)

Question - 129:

How To Change the Ownership of a Schema in MS SQL Server?

Ans:

If you want to change the owner of a schema, you can use the "ALTER AUTHORIZATION" statement using the following syntax:

```
ALTER AUTHORIZATION ON SCHEMA::schema_name TO user_name
The following tutorial example shows you how to change ownership of schema "ggl" to "ggl_user":
```

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
ALTER AUTHORIZATION ON SCHEMA::ggl TO ggl_user
GO
SELECT s.name, u.name AS owner
  FROM sys.schemas s, sys.database_principals u
 WHERE s.principal_id = u.principal_id;
GO


```
<pre>name owner

dbo dbo
ggl ggl_User
guest guest
...</pre>
```


```

[View All Answers](#)

Question - 130:

Who Is the Owner of a Schema in MS SQL Server?

Ans:

When you create a schema in a database, SQL Server will assign a owner (a database user) to this schema. If your login name is mapped to the owner of a schema at the database level, you have the full permission on all objects in this schema.

The following tutorial exercise shows you how to see who is the owner of a schema:

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
SELECT s.name, u.name AS owner
  FROM sys.schemas s, sys.database_principals u
 WHERE s.principal_id = u.principal_id;
GO


```
<pre>name owner

dbo dbo
ggl dbo
guest guest
...</pre>
```


```



The last query shows that schame "ggl" is owned by "dbo".

[View All Answers](#)

Question - 131:

What Is the Default Schema of Your Login Session in MS SQL Server?

Ans:

When you login to a SQL Server and select a database to use, SQL Server will assign your login session a default schema. The schema name can be omitted when you refer to objects in the default schema. Here is what you should remember about default schema:

- * The default schema of your login session in the current database is the default schema assigned to the current database level principal - database user.
- * If you are referring to an object in the default schema, you do not need to specify the schema name.
- * If you are referring to an object outside the default schema, you must specify the schema name.

The tutorial exercise below shows you how to verify your default schema:

```
-- Login with "ggl_login"
USE GlobalGuideLineDatabase;
GO
Changed database context to 'GlobalGuideLineDatabase'.
PRINT User_Name();
GO
ggl_User
SELECT name, default_schema_name
FROM sys.database_principals WHERE type = 'S';
GO


```
name default_schema_name

dbo dbo
guest guest
INFORMATION_SCHEMA NULL
sys NULL
ggl_User dbo
```


```

The last query shows that the default schema for "ggl_login" in "GlobalGuideLineDatabase" is &qu

[View All Answers](#)

Question - 132:

How To List All Objects in a Given Schema?

Ans:

If you are wonder what objects are stored in a given schema as an object container, you can use view "sys.objects" to get a list of all objects in a schema. The tutorial exercise shows you how to list all objects in schema "ggl" and "dbo":

```
-- Login with 'sa'
USE GlobalGuideLineDatabase;
GO
-- What is "ggl"?


```
SELECT o.name, o.schema_id, o.type_desc
FROM sys.objects o, sys.schemas s
WHERE o.schema_id = s.schema_id
AND s.name = 'ggl';
GO
name schema_id type_desc

test 5 USER_TABLE
ggl_links 5 USER_TABLE
UQ_ggl_links_4222D4EF 5 UNIQUE_CONSTRAINT
DF_ggl_links_creat_4316F928 5 DEFAULT_CONSTRAINT
SELECT o.name, o.schema_id, o.type_desc
FROM sys.objects o, sys.schemas s
WHERE o.schema_id = s.schema_id
AND s.name = 'dbo';
GO
```


```

[View All Answers](#)

Question - 133:

How To Transfer an Existing Table from One Schema to Another Schema in MS SQL Server?

Ans:

If you want to move an existing table from one schema to another schema, you can use the "ALTER SCHEMA ... TRANSFER ..." statement as shown in the tutorial exercise below:

```
-- Login with "sa"
USE GlobalGuideLineDatabase;
GO
-- Confirming that "ggl_links" is in "dbo"
SELECT t.name AS table_name, t.type_desc,
s.name AS schema_name
FROM sys.tables t, sys.schemas s
WHERE t.schema_id = s.schema_id
AND t.name = 'ggl_links';
GO


```
table_name type_desc schema_name

```


```



```
ggl_links USER_TABLE dbo</pre>
-- Moving a table to a new schema
ALTER SCHEMA ggl TRANSFER ggl_links;
GO
-- Confirming that "ggl_links" is moved to "ggl"
SELECT t.name AS table_name, t.type_desc,
       s.name AS schema_name
FROM sys.tables t, sys.schemas s
WHERE t.schema_id = s.schema_id
AND t.name = 'ggl_links';
GO
<pre>table_name type_desc schema_name
-----
ggl_links USER_TABLE ggl</pre>
The last query confirms that table "ggl_links" is now in schema "ggl".
```

[View All Answers](#)

Question - 134:

How To Create a New Table in a Given Schema?

Ans:

When you create a new table, you can specify in which schema you want this table to be located by prefixing the table name with the schema name. In the tutorial example below, a new table "test" is created in schema "ggl":

```
USE GlobalGuideLineDatabase;
GO
CREATE TABLE ggl.test (id INT);
GO
Command(s) completed successfully.
SELECT t.name AS table_name, t.type_desc,
       s.name AS schema_name
FROM sys.tables t, sys.schemas s
WHERE t.schema_id = s.schema_id
AND t.name = 'test';
GO
table_name type_desc schema_name
-----
test USER_TABLE ggl
The last query confirms that table "test" is inside schema "ggl".
```

[View All Answers](#)

Question - 135:

How To List All Schemas in a Database?

Ans:

If you want see all existing schemas in the current database, you can use view sys.schemas as shown in the example below:

```
USE GlobalGuideLineDatabase;
GO
SELECT * FROM sys.schemas;
GO
<pre>name                schema_id principal_id
-----
dbo                  1          1
guest                2          2
INFORMATION_SCHEMA  3          3
sys                  4          4
ggl                  5          1
db_owner             16384      16384
db_accessadmin       16385      16385
db_securityadmin     16386      16386
db_ddladmin          16387      16387
db_backupoperator    16389      16389
db_datareader        16390      16390
db_datawriter        16391      16391
db_denydatareader    16392      16392
db_denydatawriter    16393      16393
(14 row(s) affected)</pre>
All schemas, except "ggl", in the list were created by SQL Server.
```

[View All Answers](#)

Question - 136:

How To Create a New Schema in a Database?

Ans:

If you want to create a new schema in an existing database, you can use the "CREATE SCHEMA" statement as shown in the tutorial example below:

```
USE GlobalGuideLineDatabase;
GO
CREATE SCHEMA ggl;
GO
Command(s) completed successfully.
```



A new schema called "ggl" has been created in "GlobalGuideLineDatabase" database. "ggl" is an empty schema at this moment since no objects has been moved into "ggl" yet.

[View All Answers](#)

Question - 137:

What Is a Schema in MS SQL Server 2005?

Ans:

A schema is a container of database objects with the following interesting related rules:

- * A schema may contain different object types, like tables, indexes, views, procedures, functions, etc.
- * A database user can be assigned with a default schema.
- * Object names must be prefixed with schema names when referencing schemas outside your default schema.
- * Every schema has a single owner (a database user). A database user may own multiple schemas.
- * If you login name is mapped to the owner of a schema, you have full permissions on this schema.
- * To drop a schema, you need to empty the schema first.

[View All Answers](#)

Question - 138:

How To Create an Multi-Statement Table-Valued Function?

Ans:

To create a multi-statement table-valued function, you need to define a temporary table as the returning table in the function. INSERT statements should be used to insert data into the returning table.

The tutorial exercise below shows you a simple example of how to build a temporary table and make it as the returning table of a function:

USE GlobalGuideLineDatabase;

GO

<pre>

```
CREATE FUNCTION Yearly_Stats(
    @start_year INT, @end_year INT)
RETURNS @stats TABLE (year INT,
    min INT, max INT, counts INT)
AS BEGIN
    DECLARE @year INT;
    SET @year = @start_year;
    WHILE @year <= @end_year BEGIN
        INSERT INTO @stats
            SELECT @year AS year, MIN(counts) AS min,
                MAX(counts) AS max, COUNT(*) AS counts
            FROM fyi_links WHERE DATEPART(YEAR, created) = @year;
        SET @year = @year + 1;
    END;
    RETURN;
END
GO
</pre>
```

[View All Answers](#)

Question - 139:

How To Create an Inline Table-Valued Function?

Ans:

To create an inline table-valued function, you need to use the "RETURNS TABLE" clause in the "CREATE FUNCTION" statement. There should be no function body, except for a RETURN statement with a SELECT subquery:

An inline table-valued function can be viewed as a select statement with parameters, see the example showing in this tutorial exercise:

USE GlobalGuideLineDatabase;

GO

```
CREATE FUNCTION Top_Links(@level INT)
RETURNS TABLE
AS
RETURN (SELECT * FROM ggl_links WHERE counts > @level);
GO
SELECT counts, id, url FROM Top_Links(999900) ORDER BY counts DESC;
GO
```

```
<pre>counts    id      url
-----
999966    36470   dgqnv qd toqcoupuxortasdtzvc
999953    12292   qebmw ywe q kza wskxqns j
999943    6192    p o qi akk hk od
999923    79161   kv g g
999920    19124   p zoio
999909    90930   xq x y r
(6 row(s) affected)</pre>
```

[View All Answers](#)

Question - 140:

How Many Ways to Create Table-Valued Functions?

Ans:



SQL Server supports two syntax of creating table-valued functions:

1. Inline Table-valued Functions - A table-valued function created with a single SELECT statement:

```
CREATE FUNCTION function_name(  
    @parameter_1 data_type,  
    @parameter_2 data_type,  
    ...  
    @parameter_n data_type  
)  
RETURNS TABLE  
AS  
RETURN (select_statement);
```

2. Multi-statement Table-valued Functions - A table-valued function created with a local temporary table and a statement block:

```
CREATE FUNCTION function_name(  
    @parameter_1 data_type,  
    @parameter_2 data_type,  
    ...  
    @parameter_n data_type  
)  
RETURNS @table_variable_name TABLE (  
    column_definition_list)  
AS BEGIN  
    statement_1;  
    statement_2;  
    ...  
    statement_n;  
    RETURN  
END
```

[View All Answers](#)

Question - 141:

How Many Categories of Functions based Their Return Modes?

Ans:

SQL Server supports 2 categories of user defined functions based on their return modes:

1. Scalar-valued Functions - A function that returns a single value. Scalar-valued functions can be used in scalar expressions. Below are some scalar-valued functions:

```
PRINT GETDATE();  
GO  
May 19 2007 1:26PM  
PRINT 'URL reversed: '+REVERSE('globalguideline.com');  
GO  
URL reversed: moc.enilediuglabolg
```

2. Table-valued Functions - A function that returns data in rows and columns like a table. Table-valued functions can be used in table expressions like the FROM clause of SELECT statements Below are some scalar-valued functions:

```
SELECT * FROM fn_helpcollations() WHERE name LIKE 'French_CI%'  
GO  
name
```

```
-----  
French_CI_AI  
French_CI_AI_WS  
French_CI_AI_KS  
French_CI_AI_KS_WS  
French_CI_AS  
French_CI_AS_WS  
French_CI_AS_KS  
French_CI_AS_KS_WS  
(8 row(s) affected)
```

[View All Answers](#)

Question - 142:

How To Provide Default Values to Function Parameters?

Ans:

If you add a parameter when creating a stored procedure, you can provide a default value so that the execution statement is not required to pass input value to this parameter:

* To define a default value to a parameter when creating the function, you should use this format: "@parameter_name data_type = default_value".

* To use the default value of a parameter when executing the function, you should use the keyword DEFAULT as the input value for that parameter.

The tutorial exercise below shows you how provide default values to function parameters:

```
CREATE FUNCTION Age_In_Days (  
    @birth_date DATETIME,  
    @today DATETIME = NULL  
)  
RETURNS INT  
AS BEGIN  
    IF @today IS NULL SET @today = GETDATE();  
    RETURN DATEDIFF(DAY, @birth_date, @today);  
END;  
GO  
-- Default value is used  
PRINT 'Age in days: '+STR(  
    dbo.Age_In_Days('01-Jan-2007', DEFAULT));
```



```
GO
Age in days: 138
-- Default value is not used
PRINT 'Age in days: '+STR(
dbo.Age_In_Days('01-Jan-2007', '11-May-2007'));
GO
Age in days: 130
-- Can not skip input values
-- even for parameters with default values
PRINT 'Age in days: '+STR(
    dbo.Age_In_Days('01-Jan-2007'));
GO
Msg 313, Level 16, State 2, Line 1
An in
```

[View All Answers](#)

Question - 143:

Can You Pass Expressions to Function Parameters?

Ans:

Can you pass expressions to stored procedure parameters? The answer is yes. When executing functions, input values can be written as expressions. But the resulting value data type must match the parameter. The tutorial exercise below shows you how input values should be specified:

```
CREATE FUNCTION Area(@radius REAL)
RETURNS REAL
AS BEGIN
    RETURN 3.14*@radius*@radius;
END;
GO
-- Input value data matches the parameter
PRINT 'Area of a circle: '+STR(dbo.Area(1.5),9,3);
GO
Area of a circle: 7.065
-- Input value data does not match the parameter
PRINT 'Area of a circle: '+STR(dbo.Area('1.5'),9,3);
GO
Msg 102, Level 15, State 1, Line 3
Incorrect syntax near 'Area'.
-- Expressions are allowed
PRINT 'Area of a circle: '+STR(dbo.Area(1.0+0.5),9,3);
GO
Area of a circle: 7.065
```

[View All Answers](#)

Question - 144:

How To Provide Values to User Defined Function Parameters?

Ans:

If a user defined function is created with parameters, you need pass values to those parameters when calling the function with one of two formats listed below:

expression... function_name(value_1, value_2, ... value_n)...

The tutorial exercise below shows you how to pass values to function parameters:

```
DROP FUNCTION Welcome;
GO
CREATE FUNCTION Welcome(@url VARCHAR(40))
RETURNS VARCHAR(40)
AS BEGIN
    RETURN 'Welcome to '+@url;
END;
GO
PRINT 'Hi there, '+dbo.Welcome('GlobalGuideLine.com');
GO
Hi there, Welcome to GlobalGuideLine.com
PRINT 'Hi there, '+dbo.Welcome('GlobalGuideLine.com');
GO
Hi there, Welcome to GlobalGuideLine.com
```

[View All Answers](#)

Question - 145:

How To Create User Defined Functions with Parameters?

Ans:

Very often, you need to create a function with one or more parameters so that the function can be more generic. You only supply values to those parameters at the time of executing the function.

User defined functions with parameters can be created with the following syntax:

```
CREATE FUNCTION function_name (
    @parameter_1 data_type,
    @parameter_2 data_type,
    ...
    @parameter_n data_type
```




```

)
RETURNS data_type
AS BEGIN
    statement_1;
    statement_2;
    ...
    statement_n;
END;

```

The following tutorial exercise shows you how to create a function with one parameter called @url:

```

USE GlobalGuideLine;
GO
DROP FUNCTION Welcome;
GO
CREATE FUNCTION Welcome(@url VARCHAR(40))
    RETURNS VARCHAR(40)
    AS BEGIN
        RETURN 'Welcome to '+@url;
    END;
GO
PRINT 'Hi there, '+dbo.Welcome('GlobalGuideLine.com');
GO
Hi there, Welcome to GlobalGuideLine.com

```

[View All Answers](#)

Question - 146:

How To Modify an Existing User Defined Function?

Ans:

If you find a mistake in an existing function previously created, you can drop (delete) it and create it again correctly. But dropping a function may affect other database objects who are depending on this function.

So the best way to correct a mistake in an existing function is to use the "ALTER FUNCTION" statement as shown in the following tutorial example:

```

-- Modifying an existing function
ALTER FUNCTION Sundays()
    RETURNS INT
    AS BEGIN
        DECLARE @date DATETIME;
        DECLARE @count INT;
        SET @date = '2006-12-31';
        SET @count = 0;
        WHILE DATEPART(YEAR, @date) < 2008 BEGIN
            SET @date = DATEADD(DAY, 1, @date);
            IF DATENAME(WEEKDAY, @date) = 'Sunday'
                SET @count = @count + 1;
        END;
        RETURN @count;
    END;
GO

```

Command(s) completed successfully.

Do you know what correction has been made on this function?

[View All Answers](#)

Question - 147:

How To Get the Definition of a User Defined Function Back?

Ans:

If you want get the definition of an existing user defined function back from the SQL Server, you can use the system view called sys.sql_modules, which stores definitions of functions, stored procedures, and views.

The sys.sql_modules holds user defined function definitions identifiable by the object id of each function. The tutorial exercise below shows you how to retrieve the definition of stored procedure, "Sundays" by joining sys.sql_modules and sys.objects:

```

SELECT m.definition
    FROM sys.sql_modules m, sys.objects o
    WHERE m.object_id = o.object_id
    AND o.name = 'Sundays';
GO

```

```

definition
-----
CREATE FUNCTION Sundays()
    RETURNS INT
    AS BEGIN
        DECLARE @date DATETIME;
        DECLARE @count INT;
        SET @date = '2006-12-31';
        SET @count = 0;
        WHILE DATEPART(YEAR, @date) <= 2008 BEGIN
            SET @date = DATEADD(DAY, 1,
(1 row(s) affected)

```

[View All Answers](#)

Question - 148:



How To Generate CREATE FUNCTION Script on an Existing Function?

Ans:

If you want to know how an existing user defined function was created, you can use SQL Server Management Studio to automatically generate a "CREATE FUNCTION" script. The following tutorial shows you how to do this:

1. Run SQL Server Management Studio and connect to SQL server.
2. On the Object Explorer window, follow the object tree: Databases > GlobalGuideLineDatabase > Programmability > Functions > Scalar-valued Functions > dbo.Sundays.
3. Click right mouse button on dbo.Sundays. The context menu shows up.
4. Select "Script Function as" > "CREATE to" > "New Query Editor Window". The following script will be displayed:

```
USE [GlobalGuideLineDatabase]
GO
/***** Object: UserDefinedFunction [dbo].[Sundays]
    Script Date: 05/19/2007 23:24:36 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
<pre>CREATE FUNCTION [dbo].[Sundays]()
RETURNS INT
AS BEGIN
    DECLARE @date DATETIME;
    DECLARE @count INT;
    SET @date = '2006-12-31';
    SET @count = 0;
    WHILE DATEPART(YEAR, @date) <= 2008 BEGIN
        SET @date = DATEADD(DAY, 1, @date);
        IF DATENAME(WEEKDAY, @date) = 'Sunday'
            SET @count = @count + 1;
    END;
    RETURN @count;
END;
</pre>
```

[View All Answers](#)

Question - 149:

How To Drop an Existing User Defined Function in MS SQL Server?

Ans:

If you have an existing user defined function that you don't want to use it anymore, you should delete it from the SQL Server by using the "DROP FUNCTION" statement as shown in the tutorial example below:

```
USE GlobalGuideLineDatabase;
GO
DROP FUNCTION Welcome;
GO
Command(s) completed successfully.
SELECT * FROM sys.objects WHERE type_desc LIKE '%FUNCTION';
GO
<pre>name      object_id  schema_id  type_desc
-----
Sundays  2117582582 1          FN SQL_SCALAR_FUNCTION
(1 row(s) affected)</pre>
User defined function "Welcome" is no longer in the database.
```

[View All Answers](#)

Question - 150:

How To List All User Defined Functions in the Current Database?

Ans:

If you want to see a list of all user defined functions in your current database, you can use the system view, sys.objects as shown in this tutorial exercise:

```
USE GlobalGuideLineDatabase;
GO
-- Number of Sundays in this year
CREATE FUNCTION Sundays()
RETURNS INT
AS BEGIN
    DECLARE @date DATETIME;
    DECLARE @count INT;
    SET @date = '2006-12-31';
    SET @count = 0;
    WHILE DATEPART(YEAR, @date) <= 2008 BEGIN
        SET @date = DATEADD(DAY, 1, @date);
        IF DATENAME(WEEKDAY, @date) = 'Sunday'
            SET @count = @count + 1;
    END;
    RETURN @count;
END;
GO
SELECT * FROM sys.objects WHERE type_desc LIKE '%FUNCTION';
GO
```



```
<pre>name    object_id  schema_id  type type_desc
```

```
Welcome 2085582468 1      FN  SQL_SCALAR_FUNCTION
Sundays 2117582582 1      FN  SQL_SCALAR_FUNCTION
```

```
(2 row(s) affected)</pre>
```

sys.objects contains all types of objects in the current database. You need select only the FUNCTION object type.

[View All Answers](#)

Question - 151:

How To Use User Defined Functions in Expressions?

Ans:

An user defined function must return a value, which can be used in any expression as long as the return value data type matches the expression.

To execute a user defined function and use its return value in an expression, you just need to enter the schema name and the function name as a value in the expression. The tutorial exercise below shows you how use a user defined function in an expression:

```
-- Calling a function without schema name
```

```
PRINT 'Hi there, '+Welcome();
```

```
GO
```

```
Msg 195, Level 15, State 10, Line 1
```

```
'Welcome' is not a recognized built-in function name.
```

```
-- Calling a function with schema name
```

```
PRINT 'Hi there, '+dbo.Welcome();
```

```
GO
```

```
Hi there, Welcome to globalguideline.com
```

[View All Answers](#)

Question - 152:

How To Create a Simple User Defined Function in MS SQL Server?

Ans:

If you want to create a simple user defined function, you can use the "CREATE FUNCTION" command with a statement block in a simple format as shown in below:

```
CREATE FUNCTION function_name()
```

```
RETURNS data_type
```

```
AS BEGIN
```

```
    statement_1;
```

```
    statement_2;
```

```
    ...
```

```
    statement_n;
```

```
    RETURN expression;
```

```
END;
```

```
GO
```

The following tutorial exercise shows you how to create a simple user defined function:

```
USE GlobalGuideLineDatabase;
```

```
GO
```

```
CREATE FUNCTION Welcome()
```

```
RETURNS VARCHAR(40)
```

```
AS BEGIN
```

```
    RETURN 'Welcome to globalguideline.com';
```

```
END;
```

```
GO
```

```
PRINT dbo.Welcome();
```

```
GO
```

```
Welcome to globalguideline.com
```

[View All Answers](#)

Question - 153:

What Are the Differences between User Defined Functions and Stored Procedures?

Ans:

Differences between user defined functions and stored procedures are:

- * Stored procedures does not return any data and they can not be used in expressions.
- * User defined functions does return data and they can be used in expressions.
- * Stored procedures only takes data constants as input parameters.
- * User defined functions can take data constants and expressions as input parameters.
- * Stored procedures support output parameters.
- * User defined functions do not support output parameters.
- * Stored procedures take a parameter's default value, if it is not specified in the calling statement.
- * User defined functions take a parameter's default value, if it is specified with the keyword DEFAULT in the calling statement.
- * Stored procedures are called with no parenthesis to include parameters.
- * User defined functions are called with parenthesis to include parameters. Empty parenthesis is needed if no parameters.
- * Stored procedures can be created locally and temporarily.
- * User defined functions can not be created locally and temporarily.

[View All Answers](#)

Question - 154:

What Are User Defined Functions in MS SQL Server?

Ans:



A user defined function is a collection of Transact-SQL statements that stored in the SQL Server. A user defined function will return data when executed. A user defined function works in the same way as a system function. It can be used as a scalar expression or a temporary table depending on the turning data format. A system function is provided as part of the SQL Server. But a user defined function must be created and managed by yourself. If you are using a group of statements to calculate the same data repeatedly in different places, you should consider to create a user defined function for that group of statements. User defined functions are also called functions.

[View All Answers](#)

Question - 155:

Can Another User Execute Your Local Temporary Stored Procedures?

Ans:

Can another user execute your local temporary stored procedures? The answer is no.

To test this out, continue with the exercise from the previous tutorial. Keep that user session running and open a new client session with sqlcmd.exe. Then run the tutorial script below:

```
-- Executing permanent procedure
-- created by another user session
EXECUTE Hello 'globalguideline.com';
GO
Welcome to globalguideline
-- Executing local temporary procedure
-- created by another user session
EXECUTE #Hello 'globalguideline.com';
GO
Msg 2812, Level 16, State 62, Line 1
Could not find stored procedure '#Hello'.
```

[View All Answers](#)

Question - 156:

How To Create a Local Temporary Stored Procedure?

Ans:

A local temporary stored procedure is a special stored procedure that:

- * Is created like a normal (permanent) stored procedure with the name prefixed with a number sign (#).
- * Are only valid in the same client session where it was created.
- * Will be deleted when creating session is terminated.

This tutorial exercise here creates two stored procedures, one is permanent and the other is local temporary:

```
DROP PROCEDURE Hello;
DROP PROCEDURE #Hello;
GO
CREATE PROCEDURE Hello
    @url nvarchar(40)
AS
    PRINT 'Welcome to ' + REVERSE(@url);
GO
CREATE PROCEDURE #Hello
    @url nvarchar(40)
AS
    PRINT 'Welcome to ' + @url;
GO
EXECUTE Hello 'globalguideline.com';
GO
Welcome to globalguideline
EXECUTE #Hello 'globalguideline.com';
GO
Welcome to globalguideline.com
```

[View All Answers](#)

Question - 157:

How To Receive Output Values from Stored Procedures?

Ans:

If an output parameter is defined in a stored procedure, the execution statement must provide a variable to receive the output value in the format: "@variable_name OUTPUT" or "@parameter_name = @variable_name OUTPUT". The following tutorial exercise gives you a good example:

```
-- Using @variable format
DECLARE @message VARCHAR(40);
EXECUTE diff_in_days
    '01-Jan-2007',
    '11-May-2007',
    @message OUTPUT;
PRINT @message;
GO
May 11, 2007 - Jan 01, 2007 = 130
-- Using @parameter = @variable format
DECLARE @message VARCHAR(40);
EXEC diff_in_days
    @start_date='01-Jan-2007',
    @end_date='11-May-2007',
    @days = @message OUTPUT;
```



```
PRINT @message;
GO
May 11, 2007 - Jan 01, 2007 =      130
```

[View All Answers](#)

Question - 158:

How To Define Output Parameters in Stored Procedures?

Ans:

Sometime a stored procedure not only want to take input values from the calling statement batch, but it also want to send output values back to the calling statement batch. This can be done by defining output parameters in the CREATE PROCEDURE statement.

To define an output parameter, you should use this format: "@parameter_name data_type OUTPUT", as shown in the following tutorial exercise:

```
DROP PROCEDURE diff_in_days;
GO
-- Defining an output parameter
CREATE PROCEDURE diff_in_days
    @start_date DATETIME,
    @end_date DATETIME = '19-May-2007',
    @days VARCHAR(40) OUTPUT
AS BEGIN
    SET @days = CONVERT(VARCHAR(20),@end_date,107)
    + ' - '
    + CONVERT(VARCHAR(20),@start_date,107)
    + ' = '
    + STR(DATEDIFF(DAY, @start_date, @end_date));
END;
GO
Command(s) completed successfully.
EXEC diff_in_days
    @start_date='01-Jan-2007'
```

```
GO
Msg 201, Level 16, State 4, Procedure diff_in_days, Line 0
Procedure or Function 'diff_in_days' expects
parameter '@days', which was not supplied.
```

[View All Answers](#)

Question - 159:

How To Provide Default Values to Stored Procedure Parameters?

Ans:

If you add a parameter when creating a stored procedure, you can provide a default value so that the execution statement is not required to pass input value to this parameter.

To provide a default value to a parameter, you should use this format: "@parameter_name data_type = default_value". The tutorial exercise below shows you how provide default values to stored procedure parameters:

```
DROP PROCEDURE diff_in_days;
GO
CREATE PROCEDURE diff_in_days
    @start_date DATETIME,
    @end_date DATETIME = '19-May-2007'
AS BEGIN
    PRINT CONVERT(VARCHAR(20),@end_date,107)
    + ' - '
    + CONVERT(VARCHAR(20),@start_date,107)
    + ' = '
    + STR(DATEDIFF(DAY, @start_date, @end_date));
END;
GO
-- Default value is used
EXEC diff_in_days
    @start_date='01-Jan-2007';
GO
May 19, 2007 - Jan 01, 2007 =      138
-- Default value is not used
EXEC diff_in_days
    @start_date='01-Jan-2007',
    @end_date='11-May-2007';
GO
May 11, 2007 - Jan 01, 2007 =      130
-- Input value must be supplied for a parameter
-- without a default value
EXEC diff_in_days
    @end_date='11-May-2007';
GO
Msg 201, Level 16, State 4, Procedure diff_in_days, Line 0
```

[View All Answers](#)

Question - 160:

Can You Pass Expressions to Stored Procedure Parameters?



Ans:

Can you pass expressions to stored procedure parameters? The answer is no.

When executing stored procedures, all input values must be entered as data literals, which can be specified within single quotes ('), or without them if they cause no confusion. The tutorial exercise below shows you how input values should be specified:

```
CREATE PROCEDURE area_of_circle @radius REAL
AS BEGIN
    PRINT 'Radius = ' + STR(@radius,9,3);
    PRINT 'Area = ' + STR(3.14*@radius*@radius,9,3);
END;
GO
-- Input value without quotes
EXEC area_of_circle 1.5;
GO
Radius =    1.500
Area =     7.065
-- Input value with quotes
EXEC area_of_circle '1.5';
GO
Radius =    1.500
Area =     7.065
-- Expressions are not allowed
EXEC area_of_circle 1.0+0.5;
GO
Msg 102, Level 15, State 1, Line 1
Incorrect syntax near '+'.
```

[View All Answers](#)

Question - 161:

What Are the Advantages of Passing Name-Value Pairs as Parameters?

Ans:

When calling a stored procedure defined with parameters, you can pass values to those parameters in two ways:

- * Passing only values in the same order as parameters defined in the stored procedure.
- * Passing name-value pairs in any order.

The advantages of passing name-value pairs to stored procedure parameters are:

- * Makes the calling statement more readable - You know which value is passed to which parameter.
- * Makes it possible to pass values in an order different than how parameters are defined.

The tutorial exercise shows you some good examples of passing name-value pairs as parameters:

```
CREATE PROCEDURE diff_in_days
    @start_date DATETIME,
    @end_date DATETIME
AS BEGIN
    PRINT CONVERT(VARCHAR(20),@end_date,107)
    + ' - '
    + CONVERT(VARCHAR(20),@start_date,107)
    + ' = '
    + STR(DATEDIFF(DAY, @start_date, @end_date));
END;
GO
EXEC diff_in_days
    '01-Jan-2007',
    '19-May-2007';
GO
May 19, 2007 - Jan 01, 2007 =    138
EXEC diff_in_days
    @start_date='01-Jan-2007',
    @end_date='19-May-2007';
GO
May 19, 2007 - Jan 01, 2007 =    138
-- Name-value pairs can be given in any order
EXEC diff_in_days
```

[View All Answers](#)

Question - 162:

How To Provide Values to Stored Procedure Parameters in MS SQL Server?

Ans:

If a stored procedure is created with parameters, you need pass values to those parameters when calling the stored procedure with one of two formats listed below:

```
-- Passing values only
EXEC procedure_name value_1, value_2, ... value_n;
-- Passing name-value pairs
EXEC procedure_name
    @parameter_1 = value_1,
    @parameter_2 = value_2,
    ...
    @parameter_n = value_n;
```

The tutorial exercise below shows 2 ways to pass values to stored procedure parameters:

```
DROP PROCEDURE Hello;
GO
CREATE PROCEDURE Hello
```



```
@url nvarchar(40)
AS
PRINT 'Welcome to ' + @url;
GO
EXEC Hello 'globalguideline.com';
GO
Welcome to globalguideline.com
EXEC Hello @url='globalguideline.com';
GO
Welcome to globalguideline.com
```

[View All Answers](#)

Question - 163:

How To Create Stored Procedures with Parameters in MS SQL Server?

Ans:

Very often, you need to create a stored procedure with one or more parameters. You only supply values to those parameters at the time of executing the stored procedure.

Stored procedures with parameters can be created with the following syntax:

```
CREATE PROCEDURE procedure_name
    @parameter_1 datatype,
    @parameter_2 datatype,
    ...
    @parameter_n datatype
```

```
AS
statement_1;
statement_2;
...
statement_n;
```

GO

The following tutorial exercise shows you how to create a stored procedure with one parameter called @url:

```
USE GlobalGuideLineDatabase;
GO
DROP PROCEDURE Hello;
GO
CREATE PROCEDURE Hello
    @url nvarchar(40)
```

```
AS
PRINT 'Welcome to ' + @url;
GO
EXEC Hello 'globalguideline.com';
GO
Welcome to globalguideline.com
```

[View All Answers](#)

Question - 164:

How To Modify an Existing Stored Procedure in MS SQL Server?

Ans:

If you find a mistake in an existing stored procedure previously created, you can drop (delete) it and create it again correctly. But dropping a stored procedure may affect other database objects who are depending on this stored procedure.

So the best way to correct a mistake in an existing stored procedure is to use the "ALTER PROCEDURE" statement as shown in the following tutorial example:

```
USE GlobalGuideLineDatabase;
```

GO

```
-- Finding a mistake - the last line is wrong
```

```
SELECT m.definition
FROM sys.sql_modules m, sys.procedures p
WHERE m.object_id = p.object_id
AND p.name = 'ShowFaq';
```

GO

```
definition
```

```
-----
CREATE PROCEDURE ShowFaq AS BEGIN
PRINT 'Number of questions:';
SELECT COUNT(*) FROM Faq;
PRINT 'First 5 questions:'
SELECT TOP 5 * FROM Faq;
```

END;

```
CREATE TABLE Faq (Question VARCHAR(80));
```

```
(1 row(s) affected)
```

```
-- Modifying the stored procedure
```

```
ALTER PROCEDURE ShowFaq AS BEGIN
PRINT 'Number of questions:';
SELECT COUNT(*) FROM Faq;
PRINT 'First 5 questions:'
SELECT TOP 5 * FROM Faq;
```

END;

GO

Command(s) completed successfully.



[View All Answers](#)

Question - 165:

How To Get the Definition of a Stored Procedure Back?

Ans:

If you want get the definition of an existing stored procedure back from the SQL Server, you can use the system view called sys.sql_modules, which stores definitions of views and stored procedures.

The sys.sql_modules holds stored procedure definitions identifiable by the object id of each view. The tutorial exercise below shows you how to retrieve the definition of stored procedure, "ShowFaq" by joining sys.sql_modules and sys.procedures:

```
USE GlobalGuideLineDatabase;
GO
SELECT m.definition
FROM sys.sql_modules m, sys.procedures p
WHERE m.object_id = p.object_id
AND p.name = 'ShowFaq';
GO
definition
```

```
-----
CREATE PROCEDURE ShowFaq AS BEGIN
PRINT 'Number of questions:';
SELECT COUNT(*) FROM Faq;
PRINT 'First 5 questions:'
SELECT TOP 5 * FROM Faq;
END;
CREATE TABLE Faq (Question VARCHAR(80));
(1 row(s) affected)
```

[View All Answers](#)

Question - 166:

How To Generate CREATE PROCEDURE Script on an Existing Stored Procedure?

Ans:

If you want to know how an existing stored procedure was created, you can use SQL Server Management Studio to automatically generate a "CREATE PROCEDURE" script. The following tutorial shows you how to do this:

1. Run SQL Server Management Studio and connect to SQL server.
2. On the Object Explorer window, follow the object tree: Databases > GlobalGuideLineDatabase > Programmability > Stored Procedures > dbo.ShowFaq.
3. Click right mouse button on dbo.ShowFaq. The context menu shows up.
4. Select "Script Stored Procedure as" > "CREATE to" > "New Query Editor Window". The following script will be displayed:

```
USE [GlobalGuideLineDatabase]
GO
/***** Object: StoredProcedure [dbo].[ShowFaq]
Script Date: 05/19/2007 21:31:35 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[ShowFaq] AS BEGIN
PRINT 'Number of questions:';
SELECT COUNT(*) FROM Faq;
PRINT 'First 5 questions:'
SELECT TOP 5 * FROM Faq;
END;
CREATE TABLE Faq (Question VARCHAR(80));
```

[View All Answers](#)

Question - 167:

How To End a Stored Procedure Properly in MS SQL Server?

Ans:

Where the end of the "CREATE PROCEDURE" statement structure? The answer is simple, the end of the statement batch.

Even if you are using a "BEGIN ... END" statement block, the stored procedure structure is not going to end at the end of the statement block. It will continue to the end of the statement batch, usually the GO command. The tutorial exercise gives you a good example:

```
USE GlobalGuideLineDatabase;
GO
DROP PROCEDURE ShowFaq;
DROP TABLE Faq;
GO
-- How this statement batch will be executed?
CREATE PROCEDURE ShowFaq AS BEGIN
PRINT 'Number of questions:';
SELECT COUNT(*) FROM Faq;
PRINT 'First 5 questions:'
SELECT TOP 5 * FROM Faq;
END;
CREATE TABLE Faq (Question VARCHAR(80));
GO
EXEC ShowFaq;
GO
```



Number of questions:

Msg 208, Level 16, State 1, Procedure ShowFaq, Line 3

Invalid object name 'Faq'.

What happened here was that the "CREATE TABLE" statement was not executed. It was included as part of the stored procedure "ShowFaq". This is why you were getting the error "Invalid object name 'Faq'."

[View All Answers](#)

Question - 168:

How To Create a Stored Procedure with a Statement Block in MS SQL Server?

Ans:

If you are creating a stored procedure with multiple statements, it's better to use "BEGIN ... END" to group all statements into a single statement block. The tutorial exercise below shows you some good examples:

USE GlobalGuideLineDatabase;

GO

</pre>

CREATE PROCEDURE Show AS BEGIN

SELECT name, type_desc FROM sys.tables;

SELECT name, type_desc FROM sys.views;

SELECT name, type_desc FROM sys.procedures;

END;

GO

Command(s) completed successfully.

EXEC Show;

GO

| name | type_desc |
|------|-----------|
|------|-----------|

| | |
|------------|------------|
| ggl_random | USER_TABLE |
|------------|------------|

| | |
|-------------------|------------|
| ggl_links_indexed | USER_TABLE |
|-------------------|------------|

| | |
|-----------|------------|
| ggl_links | USER_TABLE |
|-----------|------------|

| | |
|----------------|------------|
| ggl_links_copy | USER_TABLE |
|----------------|------------|

| | |
|------|-----------|
| name | type_desc |
|------|-----------|

| | |
|---------------|------|
| ggl_links_top | VIEW |
|---------------|------|

| | |
|----------------|------|
| ggl_links_dump | VIEW |
|----------------|------|

| | |
|----------------|------|
| ggl_links_view | VIEW |
|----------------|------|

| | |
|------|-----------|
| name | type_desc |
|------|-----------|

| | |
|-------|----------------------|
| Hello | SQL_STORED_PROCEDURE |
|-------|----------------------|

| | |
|------|----------------------|
| date | SQL_STORED_PROCEDURE |
|------|----------------------|

| | |
|------|----------------------|
| Show | SQL_STORED_PROCEDURE |
|------|----------------------|

[View All Answers](#)

Question - 169:

How To Drop an Existing Stored Procedure in MS SQL Server?

Ans:

If you have an existing procedure that you don't want to use it anymore, you should delete it from the SQL Server by using the "DROP PROCEDURE" statement as shown in the tutorial example below:

USE GlobalGuideLineDatabase;

GO

DROP PROCEDURE datetime;

GO

Command(s) completed successfully.

SELECT * FROM sys.procedures;

GO

<pre>Name object_id schema_id type type_desc

| | | | | |
|-------|------------|---|---|----------------------|
| Hello | 1621580815 | 1 | P | SQL_STORED_PROCEDURE |
|-------|------------|---|---|----------------------|

| | | | | |
|------|------------|---|---|----------------------|
| date | 1653580929 | 1 | P | SQL_STORED_PROCEDURE |
|------|------------|---|---|----------------------|

(2 row(s) affected)

Stored procedure "datetime" is no longer in the database.

[View All Answers](#)

Question - 170:

How To List All Stored Procedures in the Current Database using MS SQL Server?

Ans:

If you want to see a list of stored procedures in your current database, you can use the system view, sys.procedures as shown in this tutorial exercise:

USE GlobalGuideLineDatabase;

GO

SELECT * FROM sys.procedures;

GO

<pre>Name object_id schema_id type type_desc

| | | | | |
|-------|------------|---|---|----------------------|
| Hello | 1621580815 | 1 | P | SQL_STORED_PROCEDURE |
|-------|------------|---|---|----------------------|

| | | | | |
|------|------------|---|---|----------------------|
| date | 1653580929 | 1 | P | SQL_STORED_PROCEDURE |
|------|------------|---|---|----------------------|

| | | | | |
|----------|------------|---|---|----------------------|
| datetime | 1669580986 | 1 | P | SQL_STORED_PROCEDURE |
|----------|------------|---|---|----------------------|

(3 row(s) affected)



[View All Answers](#)

Question - 171:

How To Execute a Stored Procedure in MS SQL Server?

Ans:

If you want execute a stored procedure created previously, you can use the EXECUTE statement in the following formats:

```
EXEC procedure_name;
```

```
EXECUTE procedure_name;
```

The key word EXEC is actually optional. So you can execute a stored procedure by just entering the procedure name as the statement. See examples in the following tutorial exercise:

```
USE GlobalGuideLineDatabase;
```

```
GO
```

```
-- create a quick procedure
```

```
CREATE PROCEDURE date AS
```

```
    PRINT CONVERT(VARCHAR(20),GETDATE(),107);
```

```
GO
```

```
-- execute with EXEC
```

```
EXEC date;
```

```
GO
```

```
May 19, 2007
```

```
-- execute with EXEC
```

```
date;
```

```
GO
```

```
May 19, 2007
```

```
-- using a reserved keyword as procedure name
```

```
CREATE PROCEDURE datetime AS PRINT GETDATE();
```

```
GO
```

```
datetime;
```

```
GO
```

```
May 19, 2007 11:35PM
```

Looks like SQL Server allows you to reserved keywords as stored procedure names

[View All Answers](#)

Question - 172:

How To Create a Simple Stored Procedure in MS SQL Server?

Ans:

If you want to create a simple stored procedure with no input and output parameters, you can use the "CREATE PROCEDURE" command with a statement batch in a simple format as shown in below:

```
</pre>CREATE PROCEDURE procedure_name AS
```

```
    statement_1;
```

```
    statement_2;
```

```
    ...
```

```
    statement_n;
```

```
GO
```

The following tutorial exercise shows you how to create a simple stored procedure:

```
USE GlobalGuideLineDatabase;
```

```
GO
```

```
CREATE PROCEDURE Hello AS
```

```
    SELECT 'Welcome to:';
```

```
    SELECT ' GlobalGuideLine.com';
```

```
GO
```

```
Command(s) completed successfully.
```

```
EXEC Hello;
```

```
GO
```

```
-----
```

```
Welcome to;
```

```
(1 row(s) affected)
```

```
-----
```

```
GlobalGuideLine.com
```

```
(1 row(s) affected)</pre>
```

[View All Answers](#)

Question - 173:

How To Create an Index on a View?

Ans:

If you need to search and sort data in a view with a large number of row, you may want to create an index on the view to speed up your search process.

The tutorial exercise below shows you how to create a unique clustered index on a view.

```
DROP VIEW ggl_links_view;
```

```
GO
```

```
CREATE VIEW ggl_links_view (ID, UrlReversed)
```

```
    AS SELECT id, REVERSE(url)
```

```
    FROM ggl_links_copy WHERE counts > 1000;
```

```
GO
```

```
CREATE UNIQUE CLUSTERED INDEX date_string
```

```
    ON ggl_links_view (ID);
```



GO

Cannot create index on view 'ggl_links_view'
because the view is not schema bound.

```
ALTER VIEW ggl_links_view (ID, UrlReversed)
WITH SCHEMABINDING
AS SELECT id, REVERSE(url)
FROM dbo.ggl_links_copy WHERE counts > 1000;
```

GO

```
CREATE UNIQUE CLUSTERED INDEX date_string
ON ggl_links_view (ID);
```

GO

```
EXEC SP_HELP ggl_links_view;
```

GO

```
<pre>index_name  index_description      index_keys
```

```
date_string clustered, unique located on PRIMARY ID</pre>
```

[View All Answers](#)

Question - 174:

How To Bind a View to the Schema of the Underlying Tables?

Ans:

By default, views are not bound to the schema of the underlying tables. This means that SQL Server will allow you to change underlying table's schema any time. For example, you can drop the underlying table while keep the view. Of course, this will make the view not valid any more.

If you don't want anyone to change underlying table's schema once a view has been defined, you can create a binding view with the "WITH SCHEMABINDING" clause to bind the view to the schema of underlying tables. A binding view has the following features:

- * Changing of underlying table's schema is not allowed as long as there exists one binding view.
- * Indexes can be created only on binding views.

The tutorial exercise below shows you how to create a binding with "WITH SCHEMABINDING":

```
DROP VIEW ggl_links_view;
GO
CREATE VIEW ggl_links_view (ID, DateString, CountUrl)
WITH SCHEMABINDING
AS SELECT id, CONVERT(VARCHAR(16), created, 107),
    CONVERT(VARCHAR(20), counts)+' - '+url
FROM ggl_links_copy WHERE counts > 1000;
```

GO

```
Msg 4512, Level 16, State 3, Procedure ggl_links_view,
Line 3
```

```
Cannot schema bind view 'ggl_links_view' because name
'ggl_links_copy' is invalid for schema binding.
Names must be in two-part format and an object
cannot
```

[View All Answers](#)

Question - 175:

How Column Data Types Are Determined in a View?

Ans:

When you define a view, its columns are defined through a list of expressions in the underlying SELECT statement. Their data types will be determined implicitly by the expressions.

For example, if the column expression is a column name of a underlying table, the data type of the view column will be the same of the underlying table column.

If the column expression is a function, the data type of the view column will be the function return data type.

If the column expression is an operation, the data type of the view column will be the expression result data type.

The following tutorial exercise shows you some examples of view column data types:

```
DROP VIEW ggl_links_view;
GO
CREATE VIEW ggl_links_view (ID, DateString, CountUrl) AS
SELECT id, CONVERT(VARCHAR(16), created, 107),
    CONVERT(VARCHAR(20), counts)+' - '+url
FROM ggl_links WHERE counts > 1000
GO
```

[View All Answers](#)

Question - 176:

How To Assign New Column Names in a View?

Ans:

By default, column names in a view are provided by the underlying SELECT statement.

But sometimes, the underlying SELECT statement can not provide names for output columns that specified as expressions with functions and operations. In this case, you need to assign new names for the view's columns. The tutorial exercise below creates a view to merge several table columns into a single view column with a format called CSV (Comma Separated Values):

```
CREATE VIEW ggl_links_dump AS
SELECT CONVERT(VARCHAR(20), id)
    + ',' + CONVERT(VARCHAR(20), counts)
    + ',' + url + ''
FROM ggl_links WHERE counts > 1000
```

GO

```
Msg 4511, Level 16, State 1, Procedure ggl_links_dump,
```



Line 2

Create View or Function failed because no column name was specified for column 1.

[View All Answers](#)

Question - 177:

Can We Delete Data from a View?

Ans:

Can you delete data in a view? The answer is no.

But if the question is "Can you delete data from the underlying table through view?" The answer is then yes. SQL Server will allow you to delete data from the underlying table through a view. The tutorial exercise below is a good example:

```
DELETE FROM ggl_links_top WHERE id = 100001;
GO
SELECT * FROM ggl_links_top;
GO
36470 999966 dgqnvmy pyjqd toqcoupuxortasdtzvcae jonfb
12292 999953 qebmw v qqmywe q kza wskxqns jnb
6192 999943 p o qisvrakk hk od
SELECT TOP 1 * FROM ggl_links ORDER BY counts DESC;
GO


```
<pre>id url

36470 dgqnvmy pyjqd toqcoupuxortasdtzvcae jonfb ...</pre>
```


```

[View All Answers](#)

Question - 178:

Can We Update Data in a View?

Ans:

The answer is no.

But if the question is "Can you update data in the underlying table through view?" The answer is then yes. SQL Server will allow you to update data in the underlying table through a view. The tutorial exercise below is a good example:

```
<pre>UPDATE ggl_links_top SET url = REVERSE(url)
WHERE id = 100001;
GO
SELECT * FROM ggl_links_top;
GO
id counts url
-----
100001 1000001 globalguideline.com
36470 999966 dgqnvmy pyjqd toqcoupuxortasdtzvcae jonfb
12292 999953 qebmw v qqmywe q kza wskxqns jnb
SELECT TOP 1 * FROM ggl_links ORDER BY counts DESC;
GO
id url notes counts created
-----
100001 globalguideline.com NULL 1000001 2007-05-19</pre>
```

[View All Answers](#)

Question - 179:

Can We Insert Data into a View?

Ans:

The answer is no.

But if the question is "Can you insert data into the underlying table through view?" The answer is then yes. SQL Server will allow you to insert data into the underlying table through a view with a condition:

* The insert columns must be limited to columns of a single underlying table.

The tutorial exercise below shows you how to insert data into a underlying table through a view:

```
USE GlobalGuideLineDatabase;
GO


```
<pre>ALTER VIEW ggl_links_top AS
SELECT TOP 3 id, counts, url FROM ggl_links
WHERE counts > 100
ORDER BY counts DESC;
GO
INSERT INTO ggl_links_top
VALUES(100001, 1000001, 'globalguideline.com');
GO
SELECT * FROM ggl_links_top;
GO
id counts url

100001 1000001 globalguideline.com
36470 999966 dgqnvmy pyjqd toqcoupuxortasdtzvcae jonfb
12292 999953 ggl wskxqns job
SELECT TOP 1 * FROM ggl_links ORDER BY counts DESC;
GO
id url notes counts created
```


```



100001 globalguideline.com NULL 1000001 2007-05-19</pre>

[View All Answers](#)

Question - 180:

How To Modify the Underlying Query of an Existing View?

Ans:

If you have an existing view, and want to change the underlying SELECT statement, you can use the "ALTER VIEW ..." statement to redefine the view. The tutorial exercise below shows you how modify an existing view:

USE GlobalGuideLineDatabase;

GO

<pre>

ALTER VIEW ggl_links_top AS

SELECT TOP 3 id, counts, url FROM ggl_links

WHERE counts > 100

ORDER BY counts DESC;

GO

SELECT * FROM ggl_links_top;

GO

id counts url

36470 999966 dgqnvmy pyjqd toqcoupuxortasdtzvcae jonfb

12292 999953 qebma wskxqns jnb

6192 999943 p o qisv</pre>

[View All Answers](#)

Question - 181:

Can You Use ORDER BY When Defining a View?

Ans:

Sometimes you want the data in a view to be sorted and try to use the ORDER BY clause in the SELECT statement to define the view.

But SQL Server will not allow you to use ORDER BY to define a view without the TOP clause. The tutorial exercise below shows you what error you will get when using ORDER BY in a CREATE VIEW statement:

USE GlobalGuideLineDatabase;

GO

CREATE VIEW ggl_links_top AS

SELECT id, counts, url FROM ggl_links

WHERE counts > 100

ORDER BY counts DESC;

GO

Msg 1033, Level 15, State 1, Procedure ggl_links_top, Line 4

The ORDER BY clause is invalid in views, inline functions,

derived tables, subqueries, and common table expressions,

unless TOP or FOR XML is also specified.

CREATE VIEW ggl_links_top AS

SELECT TOP 100 id, counts, url FROM ggl_links

WHERE counts > 100

ORDER BY counts DESC;

GO

SELECT TOP 3 * FROM ggl_links_top;

GO

<pre>id counts url

36470 999966 dgqnvmy onfb

12292 999953 qebmw v qkxqns jnb

6192 999943 p o qik od</pre>

The view seems to be sorted correctly.

[View All Answers](#)

Question - 182:

What Happens If You Delete a Table That Is Used by a View?

Ans:

Assuming that you have a table which is used by a view, and you try to delete that table. SQL Server will let you delete the table without any trouble.

But that view will become invalid. The tutorial exercise below shows you what happens to the view, when the underlying table is deleted:

USE GlobalGuideLineDatabase;

GO

SELECT * INTO ggl_links_copy

FROM ggl_links WHERE counts > 0;

GO

CREATE VIEW ggl_links_view AS

SELECT * FROM ggl_links_copy;

GO

SELECT COUNT(*) FROM ggl_links_view;

GO

50015

DROP TABLE ggl_links_copy;

GO



```
SELECT COUNT(*) FROM ggl_links_view;
GO
Msg 208, Level 16, State 1, Line 1
Invalid object name 'ggl_links_copy'.
Msg 4413, Level 16, State 1, Line 1
Could not use view or function 'ggl_links_view'
because of binding errors.
```

[View All Answers](#)

Question - 183:

How to Create a View using Data from Another View?

Ans:

Can You Create a View with Data from Another View? The answer is yes. A view can be used as a table to build other views. The tutorial exercise below shows you how to create a view using data from another view:

```
<pre>USE AdventureWorksLT;
GO
CREATE VIEW SalesOrderTop AS
    SELECT SalesOrderNumber, TotalDue, CompanyName
    FROM SalesOrderView
    WHERE TotalDue > 10000.0
GO
SELECT TOP 10 * FROM SalesOrderTop;
GO
SalesOrderNumber TotalDue    CompanyName
-----
SO71780          42452.6519  Nearby Cycle Shop
SO71782          43962.7901  Professional Sales and Service
SO71783          92663.5609  Eastside Department Store
SO71784          119960.824  Action Bicycle Specialists
SO71796          63686.2708  Extreme Riding Supplies
SO71797          86222.8072  Riding Cycles
SO71832          39531.6085  Closest Bicycle Store
SO71845          45992.3665  Trailblazing Sports
SO71858          15275.1977  Thrilling Bike Tours
SO71897          14017.9083  Paints and Solvents Company</pre>
```

[View All Answers](#)

Question - 184:

How to Create a View with Data from Multiple Tables?

Ans:

Can You Create a View with Data from Multiple Tables? The answer is yes. A view can be created with a SELECT statement to join data from multiple tables.

It is a common practice to normalize data into multiple tables. Then using a view to de-normalize them into a single output.

The tutorial exercise below shows you how to create a view to normalize data from two tables SalesOrderHeader and Customer in the sample database AdventureWorksLT.

```
USE AdventureWorksLT;
GO
CREATE VIEW SalesOrderView AS
    SELECT o.SalesOrderNumber, o.OrderDate, o.TotalDue,
           c.FirstName, c.LastName, c.CompanyName
    FROM SalesLT.SalesOrderHeader o, SalesLT.Customer c
    WHERE o.CustomerID = c.CustomerID
GO
```

[View All Answers](#)

Question - 185:

How To Get the Definition of a View Out of the SQL Server?

Ans:

If you want get the definition of an existing view back from the SQL Server, you can use the system view called sys.sql_modules, which stores definitions of views and procedures.

The sys.sql_modules holds view definitions identifiable by the object id of each view. The tutorial exercise below shows you how to retrieve the definition of view, "ggl_link_view" by joining sys.sql_modules and sys.views:

```
USE GlobalGuideLineDatabase;
GO
<pre>SELECT m.definition
    FROM sys.sql_modules m, sys.views v
    WHERE m.object_id = v.object_id
    AND v.name = 'ggl_links_top';
GO
definition
-----
CREATE VIEW ggl_links_top (LinkText) AS
    SELECT CONVERT(VARCHAR(20),id)
           + ' - ' + CONVERT(VARCHAR(20),counts)
           + ' - ' + url
    FROM ggl_links WHERE counts > 1000
(1 row(s) affected)</pre>
```




[View All Answers](#)

Question - 186:

How To Generate CREATE VIEW Script on an Existing View?

Ans:

If you want to know how an existing view was created, you can use SQL Server Management Studio to automatically generate a "CREATE VIEW" script. The following tutorial shows you how to do this:

1. Run SQL Server Management Studio and connect to SQL server.
2. On the Object Explorer window, follow the object tree: Databases > GlobalGuideLineDatabase > Views > dbo.ggl_links_top.
3. Click right mouse button on dbo.ggl_links_top. The context menu shows up.
4. Select "Script Table as" > "CREATE to" > "New Query Editor Window". The following script will be displayed:

```
USE [GlobalGuideLineDatabase]
GO
<pre>/***** Object: View [dbo].[ggl_links_top]
Script Date: 05/19/2007 15:07:27 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[ggl_links_top] AS
SELECT id, counts, url FROM ggl_links
WHERE counts > 100;</pre>
```

[View All Answers](#)

Question - 187:

How To Get a List of Columns in a View using the "sp_help" Stored Procedure?

Ans:

Another way to get a list of columns from a view is to use the "sp_help" stored procedure. "sp_help" returns more than just a list of columns. It returns: the view information, the column information, the identity column, the row GUID column. The tutorial exercise below shows you what you will get with sp_help:

```
EXEC SP_HELP ggl_links_top;
```

```
GO
<pre>Name          Owner Type Created_datetime
-----
ggl_links_top  dbo   view  2007-05-19 13:43:46.983
```

```
Column_name Type  Computed Length Prec Scale Nullable
-----
id          int   no      4      10  0   yes
counts     int   no      4      10  0   yes
url         varchar no      80             no
```

Identity

No identity column defined.

RowGuidCol

No rowguidcol column defined.</pre>

[View All Answers](#)

Question - 188:

How To Get a List of Columns in a View using the "sp_columns" Stored Procedure?

Ans:

If you have an existing table, but you don't remember what are the columns defined in the view, you can use the "sp_columns" stored procedure to get a list of all columns of the specified view. The following tutorial script shows you a good example:

```
EXEC SP_COLUMNS ggl_links_top;
```

```
GO
```

```
<pre>TABLE_OWNER TABLE_NAME COLUMN_NAME TYPE_NAME LENGTH
```

```
-----
dbo      ggl_links_top id      int      4
dbo      ggl_links_top counts  int      4
dbo      ggl_links_top url     varchar  80</pre>
(3 row(s) affected)
```

The "sp_columns" stored procedure returns a long list of properties for each column of the specified view.

[View All Answers](#)

Question - 189:

How To Get a List of Columns in a View using "sys.columns" in MS SQL Server?

Ans:

If you have an existing view, but you don't remember what are the columns defined in the view, you can use the "sys.columns" system view to get a list of all columns of all views in the current database.

In order to a list of columns of a single view, you need to join sys.columns and sys.views as shown in the tutorial example below:

```
SELECT * FROM sys.columns c, sys.views v
WHERE c.object_id = v.object_id
```



```
AND t.name = 'ggl_links_top'
GO


```
<pre>object_id name column_id user_type_id max_length

1205579333 id 1 56 4
1205579333 counts 2 56 4
1205579333 url 3 167 80</pre>
```


```

You can see the column names easily from the sys.columns view. But you can only see the column type IDs. This requires another join to get the column type names. You may try the "sp_columns" stored procedure to get a better list of columns.

[View All Answers](#)

Question - 190:

How To Drop Existing Views from a Database in MS SQL Server?

Ans:

If you don't need a specific view any more, you can use the DROP VIEW statement to delete it from the database. The following tutorial exercise shows you how to delete the view, ggl_links_view:

```
USE GlobalGuideLineDatabase;
GO
SELECT * FROM sys.views;
GO
name          object_id  schema_id type type_desc
-----
ggl_links_view 1189579276 1      V  VIEW
ggl_links_top  1205579333 1      V  VIEW
(2 row(s) affected)
DROP VIEW ggl_links_view;
GO
SELECT * FROM sys.views;
GO


```
<pre>name object_id schema_id type type_desc

ggl_links_top 1205579333 1 V VIEW
(1 row(s) affected)</pre>
```


```

[View All Answers](#)

Question - 191:

How To See Existing Views in MS SQL Server?

Ans:

If you want to know how many views you have created in a database, you use the system view called sys.views to get a list of views defined in the current database. The tutorial exercise shows you how many views in database GlobalGuideLineDatabase:

```
USE GlobalGuideLineDatabase;
GO
CREATE VIEW ggl_links_view AS
  SELECT * FROM ggl_links;
GO
SELECT * FROM sys.views;
GO


```
<pre>name object_id schema_id type type_desc

ggl_links_view 1189579276 1 V VIEW
ggl_links_top 1205579333 1 V VIEW
(2 row(s) affected)</pre>
```


```

[View All Answers](#)

Question - 192:

How To Create a View on an Existing Table in MS SQL Server?

Ans:

If you want to a view on an existing table, you can use the CREATE VIEW statement in a simple syntax:

CREATE VIEW view_name AS SELECT ...
The tutorial exercise below shows you how to create a view to represent sub set of data stored in ggl_links table:

```
USE GlobalGuideLineDatabase;
GO
CREATE VIEW ggl_links_top AS
  SELECT id, counts, url FROM ggl_links
  WHERE counts > 100;
GO
SELECT TOP 10 * FROM ggl_link_top;
GO


```
<pre>id counts url

7600 237946 eyfndw jdt lee ztejyx l q jdh k
19437 222337 eypx u x
55924 1877 eyq ntohxe i rtnlu riwaskzp cucoa dva c
63742 121330 ezdaeh mmgmo vaai meytbjiv f jixfsdjw pw
92455 945262 ezlmyenrw dyeb
36391 41386 f
```


```



```
87433 977726 f
7180 559314 f kqbqlej s xixuuregg lh r dqqvqsstxw
2252 702033 f bkh jy sqrkttuoarxmfp idqyhyy tme d
1228 146283 f m asukh</pre>
```

[View All Answers](#)

Question - 193:

What Are Views in MS SQL Server?

Ans:

A view is a database object that represents the data in one or more tables in the same structure as a separate table. Here are some basic rules about views:

- * Tables store real data.
- * Views do not store real data.
- * Views must have underlying tables to provide data.
- * Each view is based on a single SELECT statement to control what data to collect from tables, and how data should be represented.
- * View's columns can be mapped directly to columns in underlying tables.
- * View's columns can be created expressions based multiple columns in underlying tables.
- * Views can be used in same way as tables in queries.

[View All Answers](#)

Question - 194:

How To Recreate an Existing Index in MS SQL Server?

Ans:

If you want to change the definition of an existing index, you can use the "DROP INDEX" statement to drop the index first. Then use the "CREATE INDEX" statement to create it again with the new definition.

But you can also combine those two statements into one:

CREATE INDEX ... WITH (DROP_EXISTING = ON)

The tutorial exercise below recreates ggl_links_url with a change to index columns:

USE GlobalGuideLineDatabase;

GO

CREATE INDEX ggl_links_url ON ggl_links_indexed (url, counts)

WITH (DROP_EXISTING = ON);

GO

SP_HELP ggl_links_indexed;

GO

```
<pre>index_name      index_description      index_keys
-----
```

```
ggl_links_counts  nonclustered located on PRIMARY  counts
```

```
ggl_links_url     nonclustered located on PRIMARY  url, counts</pre>
```

[View All Answers](#)

Question - 195:

How To Rebuild All Indexes on a Single Table?

Ans:

If you have several indexes on a single table and want to rebuild all of them, you may use the "ALTER INDEX ALL ON table_name REBUILD" statement as shown in the tutorial exercise below:

UPDATE ggl_links_indexed

SET url = REVERSE(url), counts = -counts

WHERE id <= 50000;

GO

(50000 row(s) affected)

SELECT i.index_id, i.name, s.avg_fragmentation_in_percent

FROM sys.dm_db_index_physical_stats (

DB_ID(N'GlobalGuideLineDatabase'),

OBJECT_ID(N'ggl_links_indexed'),

DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i

WHERE s.object_id = i.object_id

AND s.index_id = i.index_id;

GO

```
0      NULL      0.574712643678161
```

```
2      ggl_links_url  85.750315258512
```

```
3      ggl_links_counts 84.040404040404
```

ALTER INDEX ALL ON ggl_links_indexed REBUILD;

GO

SELECT i.index_id, i.name, s.avg_fragmentation_in_percent

FROM sys.dm_db_index_physical_stats (

DB_ID(N'GlobalGuideLineDatabase'),

OBJECT_ID(N'ggl_links_indexed'),

DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i

WHERE s.object_id = i.object_id

AND s.index_id = i.index_id;

GO<pre>

```
0      NULL      0.574712643678161
```

```
2      ggl_links_url  0.12987012987013
```

```
3      ggl_links_counts 0.44843</pre>
```

[View All Answers](#)



Question - 196:

How To Rebuild Indexes with ALTER INDEX ... REBUILD?

Ans:

When an index is defragmented to a large percentage, like > 30%, you can use the "ALTER INDEX ... REBUILD" statement to rebuild the index. Here is a tutorial exercise on rebuilding indexes:

```
UPDATE ggl_links_indexed SET url = REVERSE(url)
WHERE id <= 50000;
GO
(50000 row(s) affected)
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
FROM sys.dm_db_index_physical_stats (
    DB_ID(N'GlobalGuideLineDatabase'),
    OBJECT_ID(N'ggl_links_indexed'),
    DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
WHERE s.object_id = i.object_id
AND s.index_id = i.index_id;
GO
0      NULL      0.574712643678161
2      ggl_links_url  85.0142045454545
3      ggl_links_counts 0.448430493273543
ALTER INDEX ggl_links_url ON ggl_links_indexed REBUILD;
GO
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
FROM sys.dm_db_index_physical_stats (
    DB_ID(N'GlobalGuideLineDatabase'),
    OBJECT_ID(N'ggl_links_indexed'),
    DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
WHERE s.object_id = i.object_id
AND s.index_id = i.index_id;
GO<pre>
0      NULL      0.574712643678161
2      ggl_links_url  0
3      ggl_links_counts 0.448430493273543</pre>
```

[View All Answers](#)

Question - 197:

How To Defragment Indexes with ALTER INDEX ... REORGANIZE?

Ans:

When an index is defragmented to a small percentage, like < 30%, you can use the "ALTER INDEX ... REORGANIZE" statement to defragment the index. Here is a tutorial exercise on defragmenting indexes:

```
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
FROM sys.dm_db_index_physical_stats (
    DB_ID(N'GlobalGuideLineDatabase'),
    OBJECT_ID(N'ggl_links_indexed'),
    DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
WHERE s.object_id = i.object_id
AND s.index_id = i.index_id;
GO
0      NULL      0.574712643678161
2      ggl_links_url  84.053862508859
3      ggl_links_counts 0.448430493273543
ALTER INDEX ggl_links_url ON ggl_links_indexed REORGANIZE;
GO
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
FROM sys.dm_db_index_physical_stats (
    DB_ID(N'GlobalGuideLineDatabase'),
    OBJECT_ID(N'ggl_links_indexed'),
    DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
WHERE s.object_id = i.object_id
AND s.index_id = i.index_id;
GO
0      NULL      0.574712643678161
2      ggl_links_url  1.87590187590188
3      ggl_links_counts 0.448430493273543
The fragmentation level has been reduced from 84.05% to 1.88%.
```

[View All Answers](#)

Question - 198:

How To Defragment Table Indexes?

Ans:

When a table index is fragmented to a certain percentage, you need to defragment the index to maintain its performance level. There are 3 ways to defragment:

1. "ALTER INDEX index_name ON table_name REORGANIZE" - Defragmenting the specified index performed in online mode. No locks applied on affected table. Used for indexes with a small fragmentation percentage.
2. "ALTER INDEX index_name ON table_name REBUILD" - Defragmenting the specified index performed in offline mode by default. It can be performed in online mode. Used for indexes with a large fragmentation percentage.
3. "CREATE INDEX ... WITH (DROP_EXISTING = ON)" - Re-creating the specified index. The definition of the index can be changed.



[View All Answers](#)

Question - 199:

What Causes Index Fragmentation?

Ans:

Index fragmentation is usually caused by deleting of existing rows or updating existing values of the indexed column. Inserting new rows should not cause any index fragmentation.

This tutorial exercise shows you how update statements of 50000 rows on the table "ggl_link_indexed" with 100000 rows caused the index fragmented 84%:

USE GlobalGuideLineDatabase;

GO

```
SELECT COUNT(*) FROM ggl_links_indexed;
```

GO

100000

```
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
```

```
FROM sys.dm_db_index_physical_stats (
```

```
DB_ID(N'GlobalGuideLineDatabase'),
```

```
OBJECT_ID(N'ggl_links_indexed'),
```

```
DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
```

```
WHERE s.object_id = i.object_id
```

```
AND s.index_id = i.index_id;
```

GO

```
0      NULL      0.574712643678161
```

```
2      ggl_links_url      0
```

```
3      ggl_links_counts 0
```

```
UPDATE ggl_links_indexed SET url = REVERSE(url)
```

```
WHERE id <=50000;
```

GO

(50000 row(s) affected)

```
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
```

```
FROM sys.dm_db_index_physical_stats (
```

```
DB_ID(N'GlobalGuideLineDatabase'),
```

```
OBJECT_ID(N'ggl_links_indexed'),
```

```
DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
```

```
WHERE s.object_id = i.object_id
```

```
AND s.index_id = i.index_id;
```

GO

[View All Answers](#)

Question - 200:

What Is Index Fragmentation in MS SQL Server?

Ans:

Index fragmentation is a phenomena where index contents are no longer stored continuously in the storage. When index contents become scattered in the storage, fragmented, performance on index will degrade.

If you want to see the fragmentation level of an index, you can use the system function called sys.dm_db_index_physical_stats() in the following format:

```
SELECT * FROM sys.dm_db_index_physical_stats(
```

```
database_id, table_id, index_id, DEFAULT, DEFAULT
```

```
)
```

The tutorial exercise below shows you how to view the fragmentation level of all indexes on table "ggl_links_indexed":

USE GlobalGuideLineDatabase

GO

```
SELECT COUNT(*) FROM ggl_links_indexed;
```

GO

100000

```
SELECT i.index_id, i.name, s.avg_fragmentation_in_percent
```

```
FROM sys.dm_db_index_physical_stats (
```

```
DB_ID(N'GlobalGuideLineDatabase'),
```

```
OBJECT_ID(N'ggl_links_indexed'),
```

```
DEFAULT, DEFAULT, DEFAULT) s, sys.indexes i
```

```
WHERE s.object_id = i.object_id
```

```
AND s.index_id = i.index_id;
```

GO

[View All Answers](#)

Question - 201:

What Is the Impact on Other User Sessions When Creating Indexes?

Ans:

If you are creating a new index on a table with existing data, all existing rows will be indexed as part of the CREATE INDEX statement. If the table is large, the indexing process could take some time. The impact of this indexing process on other user sessions is based whether SQL server is using the Offline mode or Online mode.

By default, SQL Server performs indexing operations in Offline mode, where table locks are applied for the duration of the index operation. An offline index operation that creates, rebuilds, or drops a clustered index, or rebuilds or drops a nonclustered index, acquires a Schema modification (Sch-M) lock on the table. This prevents all user access to the underlying table for the duration of the operation. An offline index operation that creates a nonclustered index acquires a Shared (S) lock on the table. This prevents updates to the underlying table but allows read operations, such as SELECT statements.

SQL Server Enterprise Edition supports indexing operations in Online mode, where other user sessions will not be impacted.

However, SQL Server Express Edition does not support the Online mode. If you try it, you will get an error as shown below:

```
CREATE INDEX ggl_links_url ON ggl_links_indexed (url)
```



WITH (ONLINE = ON);
GO
Online index operations can only be performed in Enterprise edition of SQL Server.

[View All Answers](#)

Question - 202:

What Happens If You Add a New Index to Large Table?

Ans:

An index can be added when you create a new table. New rows will be indexed as they are inserted into the table. But you can also add a new index to an existing table with the same CREATE INDEX statement. The existing rows will be indexed as part of the CREATE INDEX statement. If you add a new index to an existing table with a large number of rows. The CREATE INDEX statement could take some time to finish. See the tutorial exercise below:

```
USE GlobalGuideLineDatabase
GO
-- Drop indexes if needed
DROP INDEX ggl_links_indexed.ggl_links_url;
DROP INDEX ggl_links_indexed.ggl_links_counts;
GO
SELECT COUNT(*) FROM ggl_links_indexed;
GO
100000
-- Create two indexes
DECLARE @start_time DATETIME, @end_time DATETIME;
SET @start_time = GETDATE();
CREATE INDEX ggl_links_url ON ggl_links_indexed (url);
CREATE INDEX ggl_links_counts ON ggl_links_indexed (counts);
SET @end_time = GETDATE();
PRINT 'Milliseconds used: '+CONVERT(VARCHAR(20),
    DATEDIFF(MILLISECOND, @start_time, @end_time));
GO
-- First time
Milliseconds used: 12626
-- Second time
Milliseconds used: 11763
-- Third time
Milliseconds used: 13890
You can see creating two indexes on a table of 100000 rows costs about 12 seconds.
```

[View All Answers](#)

Question - 203:

Does Index Speed Up SELECT Statements?

Ans:

If you want to see the impact of indexes on SELECT statements, you can run the same SELECT statement on "ggl_links" and "ggl_links_indexed" tables. See the tutorial exercise below:

```
USE GlobalGuideLineDatabase;
GO
-- Run SELECT on the table without indexes
DECLARE @start_time DATETIME, @end_time DATETIME;
SET @start_time = GETDATE();
SELECT TOP 3 counts, url FROM ggl_links
    WHERE url LIKE 'a%'
    ORDER BY counts DESC;
SET @end_time = GETDATE();
PRINT 'Milliseconds used: '+CONVERT(VARCHAR(20),
    DATEDIFF(MILLISECOND, @start_time, @end_time));
GO


```
<pre>counts url

999417 a ihgu migox mlqvi gvs n nmabwdk iehu ezjyv n 1
999008 agqbr gkjwfpjvp z c cq k inen j ppjsu x iuk uhp
998471 a yiu squqco eih
</pre>
Milliseconds used: 46
```


```

[View All Answers](#)

Question - 204:

Does Index Slows Down INSERT Statements?

Ans:

If you want to see the impact of indexes on INSERT statements, you can repeat the same insert script on the table "ggl_links" of the same structure with two indexes: one non-clustered index on column "url" and one non-clustered index on column "counts". See the tutorial exercise below:

```
USE GlobalGuideLineDatabase
GO
-- Drop the old table, if needed
DROP TABLE ggl_links_indexed;
GO
```



```
-- Create a table
CREATE TABLE ggl_links_indexed (
    id INT,
    url VARCHAR(80) NOT NULL,
    notes VARCHAR(1024),
    counts INT,
    created DATETIME NOT NULL DEFAULT(getdate())
);
GO
-- Create two indexes
CREATE INDEX ggl_links_url ON ggl_links_indexed (url);
CREATE INDEX ggl_links_counts ON ggl_links_indexed (counts);
GO
-- Empty the table if needed
DELETE FROM ggl_links_indexed;
GO
-- Performance test of INSERT
DECLARE @start_time DATETIME, @end_time DATETIME;
SET @start_time = GETDATE();
INSERT INTO ggl_links_indexed
    SELECT id, rand_string, REVERSE(rand_string),
        rand_integer, rand_datetime
    FROM ggl_random
SET @end_time = GETDATE();
PRINT 'Milliseconds used: '+CONVERT(VARCHAR(20),
    DATEDIFF(MILLISECOND, @start_time, @end_time));
GO
```

[View All Answers](#)

Question - 205:

How To Create a Large Table with Random Data for Index Testing in MS SQL Server?

Ans:

If you want to see how index can be used to improve data search performance, you have to build some large tables, which requires large amount of random data. This tutorial exercise helps you to build a large table with pure random data:

USE GlobalGuideLineDatabase;

GO

-- Drop the old table, if needed

DROP TABLE ggl_random;

GO

-- Create a table with primary key

CREATE TABLE ggl_random (

id INT,

rand_integer INT,

rand_number numeric(18,9),

rand_datetime DATETIME,

rand_string VARCHAR(80)

);

GO

[View All Answers](#)

Question - 206:

What Is the Difference Between Clustered and Non-Clustered Indexes in MS SQL Server?

Ans:

SQL Server 2005 supports two types of indexes: clustered index and non-clustered index. Here are the main differences between them:

- * One table can only have only one clustered index.
- * One table can only have many non-clustered index.
- * A clustered index requires no separate storage than the table storage. It forces the rows to be stored sorted on the index key.
- * A non-clustered index requires separate storage than the table storage to store the index information.
- * A table with a clustered index is called clustered table. Its rows are stored in a B-Tree structure sorted.
- * A table without any clustered indexes is called non-clustered table. Its rows are stored in heap structure unsorted.
- * The default index created as part of the primary key column is a clustered index.

[View All Answers](#)

Question - 207:

Does the UNIQUE Constraint Create an Index?

Ans:

If you add the UNIQUE constraint on a column, SQL Server will automatically add a non-clustered index for that column. The tutorial exercise below shows you the index created as part of the UNIQUE column, "id", of "ggl_links":

USE GlobalGuideLineDatabase;

GO

-- Drop the old table, if needed

DROP TABLE ggl_links;

GO

-- Create a table with a UNIQUE constraint

CREATE TABLE ggl_links (

id INT UNIQUE,



```
url VARCHAR(80) NOT NULL,
notes VARCHAR(1024),
counts INT,
created DATETIME NOT NULL DEFAULT(getdate())
);
```

```
GO
-- Create an index for column "url"
CREATE INDEX ggl_links_url ON ggl_links (url);
GO
```

```
<pre>-- View indexes
EXEC SP_HELP ggl_links;
GO
index_name          index_description      keys
-----
ggl_links_url       nonclustered located      url
                    on PRIMARY
UQ__ggl_links__4222D4EF nonclustered, unique
                    key located on PRIMARY  id</pre>
```

Notice that the index created as part of the UNIQUE constraint is named by SQL Server as "UQ__ggl_links__4222D4EF".

[View All Answers](#)

Question - 208:

Is the PRIMARY KEY Column of a Table an Index in MS SQL Server?

Ans:

If you define a primary key on a table, an index for the primary key column will be created by default. The tutorial exercise below shows you the index created as part of the primary key column of "ggl_links":

```
USE GlobalGuideLineDatabase;
GO
-- Drop the old table, if needed
DROP TABLE ggl_links;
GO
-- Create a table with primary key
CREATE TABLE ggl_links (
  id INT PRIMARY KEY,
  url VARCHAR(80) NOT NULL,
  notes VARCHAR(1024),
  counts INT,
  created DATETIME NOT NULL DEFAULT(getdate())
);
```

```
GO
-- Create an index for column "url"
CREATE INDEX ggl_links_url ON ggl_links (url);
GO
```

```
-- View indexes
EXEC SP_HELP ggl_links;
GO
<pre>index_name          index_description      keys
-----
ggl_links_url       nonclustered located      url
                    on PRIMARY
PK__ggl_links__239E4DCF clustered, unique, primary
                    key located on PRIMARY  id</pre>
```

Notice that the index created as part of the primary key is named by SQL Server as "PK__ggl_links__239E4DCF".

[View All Answers](#)

Question - 209:

How To Drop Existing Indexes in MS SQL Server?

Ans:

For some reason, if you want remove an existing index, you can use the DROP INDEX statement with following syntax:

```
CREATE INDEX table_name.index_name
The tutorial exercise below shows you how to remove the index "ggl_links_id":
USE GlobalGuideLineDatabase;
```

```
GO
SELECT * FROM sys.indexes WHERE object_id = (
  SELECT object_id FROM sys.tables WHERE name = 'ggl_links'
);
```

```
GO
<pre>object_id name          index_id type_desc  is_unique
-----
421576540 NULL          0      HEAP      0
421576540 ggl_links_id 2      NONCLUSTERED 0
421576540 ggl_links_url 3      NONCLUSTERED 0</pre>
```

```
DROP INDEX ggl_links.ggl_links_id;
GO
SELECT * FROM sys.indexes WHERE object_id = (
  SELECT object_id FROM sys.tables WHERE name = 'ggl_links'
);
```



GO

```
<pre>object_id name          index_id type_desc  is_unique
-----
421576540 NULL            0      HEAP           0
421576540 ggl_links_url 3      NONCLUSTERED 0</pre>
```

[View All Answers](#)

Question - 210:

How To View Existing Indexes on an Given Table using sys.indexes?

Ans:

Another way to view existing indexes defined for a given table is to use the system view called "sys.indexes". The tutorial exercise shows you how many indexes were defined from the previous tutorial on table "ggl_links":

USE GlobalGuideLineDatabase;

GO

```
SELECT * FROM sys.indexes WHERE object_id = (
    SELECT object_id FROM sys.tables WHERE name = 'ggl_links'
);
```

GO

```
<pre>object_id name          index_id type_desc  is_unique
-----
421576540 NULL            0      HEAP           0
421576540 ggl_links_id 2      NONCLUSTERED 0
421576540 ggl_links_url 3      NONCLUSTERED 0</pre>
```

The extra line in the query result is not a real index at this moment. It will be explained in another tutorial.

[View All Answers](#)

Question - 211:

How To View Existing Indexes on an Given Table using SP_HELP?

Ans:

If you want to know how many indexes have been defined for a given table, you can use the SP_HELP built-in stored procedure in the following syntax:

```
EXEC SP_HELP table_name
```

-- Returns all database objects related the given table

The tutorial exercise shows you how many indexes were defined from the previous tutorial on table "ggl_links":

```
EXEC SP_HELP ggl_links;
```

GO

...

```
<pre>
```

```
index_name  index_description          index_keys
-----
ggl_links_id nonclustered located on PRIMARY id
ggl_links_url nonclustered located on PRIMARY url
...</pre>
```

[View All Answers](#)

Question - 212:

How To Create an Index on an Existing Table in MS SQL Server?

Ans:

If you want to an index on an existing table, you can use the CREATE INDEX statement in a simple syntax:

```
CREATE INDEX index_name ON table_name (column_name)
```

The tutorial exercise below shows you how to add an index for the "id" column of the "ggl_links" table:

USE GlobalGuideLineDatabase;

GO

-- Drop the old table, if needed

```
DROP TABLE ggl_links;
```

GO

-- Create a fresh new table

```
CREATE TABLE ggl_links (
```

```
    id INT NOT NULL,
```

```
    url VARCHAR(80) NOT NULL,
```

```
    notes VARCHAR(1024),
```

```
    counts INT,
```

```
    created DATETIME NOT NULL DEFAULT(getdate())
```

```
);
```

GO

-- Create an index for "id"

```
CREATE INDEX ggl_links_id ON ggl_links (id);
```

GO

-- Create an index for "url"

```
CREATE INDEX ggl_links_url ON ggl_links (url);
```

GO

[View All Answers](#)

Question - 213:

What Are Indexes in MS SQL Server?



Ans:

An index is a secondary database object associated with a table to improve the retrieval performance of rows from that table.

An index can be defined for a single column or multiple columns of a given table. If an index is defined on a single column of a table, the index can be viewed as ordered map between all values in that column and their row locations in the table. For examples, if you have table with a column called "company_num" and you created an index for that column. The contents of the table and the index may look like what is presented in the following picture:

Table Index Diagram

As shown in the above picture, locating the rows of company_num = 17 can be done much faster through the index.

[View All Answers](#)

Question - 214:

How to add an address record into AdventureWorksLT?

Ans:

To find out if we can add data into AdventureWorksLT or not, you can try to add an address record into the "SalesLT.Address" in AdventureWorksLT:

```
USE AdventureWorksLT
```

```
GO
```

```
INSERT SalesLT.Address (AddressLine1, City, StateProvince,
```

```
CountryRegion, PostalCode)
```

```
VALUES('1 Main Street', 'Java', 'Oracle',
```

```
'GGL', 'Center')
```

```
GO
```

```
SELECT * FROM SalesLT.Address
```

```
WHERE CountryRegion = 'GGL'
```

```
GO
```

```
AddressID/AddressLine1/AddressLine2/City/StateProvince
```

```
/CountryRegion/PostalCode/rowguid
```

```
/ModifiedDate
```

```
11384/1 Main Street/NULL/Java/Oracle
```

```
/GGL/Center/6073DFAE-9803-4B4F-B60E-D9742C0EED1D
```

```
/2007-05-19 19:24:44.140
```

[View All Answers](#)

Question - 215:

How AdventureWorksLT tables are related?

Ans:

There are 12 user tables defined in AdventureWorksLT:

```
USE AdventureWorksLT
```

```
GO
```

```
SELECT s.name, t.name, t.type_desc
```

```
FROM sys.tables t, sys.schemas s
```

```
WHERE t.schema_id = s.schema_id
```

```
ORDER BY s.name
```

```
GO
```

```
<pre>s.name t.name t.type_desc
```

```
dbo BuildVersion USER_TABLE
```

```
dbo ErrorLog USER_TABLE
```

```
SalesLT Address USER_TABLE
```

```
SalesLT Customer USER_TABLE
```

```
SalesLT CustomerAddress USER_TABLE
```

```
SalesLT Product USER_TABLE
```

```
SalesLT ProductCategory USER_TABLE
```

```
SalesLT ProductDescription USER_TABLE
```

```
SalesLT ProductModel USER_TABLE
```

```
SalesLT ProductModelProductDescription USER_TABLE
```

```
SalesLT SalesOrderDetail USER_TABLE
```

```
SalesLT SalesOrderHeader USER_TABLE</pre>
```

Here is an ER (Entity Relation) diagram presented on Jasmin's photo gallery:

AdventureWorksLT ER Diagram.

[View All Answers](#)

Question - 216:

How to attach AdventureWorksLT physical files to the server?

Ans:

After installed the sample database AdventureWorksLT, you need to attach it to your SQL server to make it available by follow this tutorial:

```
EXEC sp_attach_db @dbname=N'AdventureWorksLT',
```

```
@filename1=N'C:\Program Files\Microsoft SQL Server'
```

```
+ 'MSSQL.1\MSSQLData\AdventureWorks_Data.mdf',
```

```
@filename2=N'C:\Program Files\Microsoft SQL Server'
```

```
+ 'MSSQL.1\MSSQLData\AdventureWorks_Log.ldf'
```

```
GO
```

```
USE AdventureWorksLT
```

```
GO
```

```
SELECT TOP 10 CustomerID, FirstName, LastName, Phone
```

```
FROM SalesLT.Customer
```

```
GO
```

```
<pre>CustomerID FirstName LastName Phone
```



```
1 Orlando Gee 245-555-0173
2 Keith Harris 170-555-0127
3 Donna Carreras 279-555-0130
4 Janet Gates 710-555-0173
5 Lucy Harrington 828-555-0186
6 Rosmarie Carroll 244-555-0112
7 Dominic Gash 192-555-0173
10 Kathleen Garza 150-555-0127
11 Katherine Harding 926-555-0159
12 Johnny Caprio 112-555-0191</pre>
```

Looks like the sample database AdventureWorksLT is ready for you play.

[View All Answers](#)

Question - 217:

How to download and install the scaled-down database AdventureWorksLT?

Ans:

If you want to practice you DBA skills with the scaled-down version of the sample database AdventureWorksLT provided by Microsoft, you should follow this tutorial to download and install it first:

1. Go to the SQL Server 2005 Samples and Sample Databases download page.

2. Go to the x86 section in the Instructions section, Click "AdventureWorksLT.msi -- 2,251 KB" to download the sample file. Save the download file to c: emp.

3. Double click on the downloaded file: c: empAdventureWorksLT.msi. The installation setup window shows up. Follow the instructions to finish the installation.

When the installation is done, two physical database files are installed in the data directory: C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQLData:

AdventureWorksLT_Data.mdf 5,120 KB

AdventureWorksLT_Log.ldf 2,048 KB

[View All Answers](#)

Question - 218:

What Samples and Sample Databases Are Provided by Microsoft?

Ans:

In order to help you to learn SQL Server, Microsoft provides several free sample scripts and sample databases.

- * SqlServerSamples.msi - 25,469 KB: Sample scripts.

- * AdventureWorksDB.msi - 28,053 KB: Sample OLTP database: AdventureWorks, case-sensitive collation version.

- * AdventureWorksDBCI.msi - 29,177 KB: Sample OLTP database: AdventureWorks, case-insensitive collation version.

- * AdventureWorksBI.msi - 7,393 KB: Sample data warehouse database: AdventureWorks, case-sensitive collation version.

- * AdventureWorksBICI.msi - 16,764 KB: Sample data warehouse database: AdventureWorks, case-insensitive collation version.

- * AdventureWorksLT.msi - 2,251 KB: The scaled-down sample database AdventureWorks for beginners.

[View All Answers](#)

Question - 219:

How To Skip Remaining Statements in a Loop Block Using CONTINUE Statements?

Ans:

If you want to skip the remaining statements in a loop block, you can use the CONTINUE statement.

The tutorial exercise below shows you how to use a CONTINUE statement to skip the remaining statements and continue the next iteration:

-- Printing first 7 Sundays in 2000

```
DECLARE @date DATETIME;
```

```
DECLARE @count INT;
```

```
SET @date = '1999-12-31';
```

```
SET @count = 0;
```

```
WHILE DATEPART(YEAR, @date) < 2001 BEGIN
```

```
    SET @date = DATEADD(DAY, 1, @date);
```

```
    IF @count = 7 BREAK;
```

```
    IF DATENAME(WEEKDAY, @date) <> 'Sunday' CONTINUE;
```

```
    PRINT CONVERT(VARCHAR(40), @date, 107);
```

```
    SET @count = @count + 1;
```

```
END
```

```
GO
```

```
Jan 02, 2000
```

```
Jan 09, 2000
```

```
Jan 16, 2000
```

```
Jan 23, 2000
```

```
Jan 30, 2000
```

```
Feb 06, 2000
```

```
Feb 13, 2000
```

[View All Answers](#)

Question - 220:

How To Stop a Loop Early with BREAK Statements in MS SQL Server?

Ans:

If you want to stop a WHILE loop early, you can use the BREAK statement in the loop statement block.

The tutorial exercise below shows you how to use a BREAK statement to stop a WHILE loop early:

-- Counting number of days in 2000

```
DECLARE @date DATETIME;
```

```
DECLARE @count INT;
```



```
SET @date = '2000-01-01';
SET @count = 0;
WHILE 1=1 BEGIN
    IF DATEPART(YEAR, @date) > 2000 BREAK;
    SET @count = @count + 1;
    SET @date = DATEADD(DAY, 1, @date);
END
SELECT @count;
366
-- 2000 is a leap year!
```

[View All Answers](#)

Question - 221:

How To Use "BEGIN ... END" Statement Structures in MS SQL Server?

Ans:

"BEGIN ... END" statement structure is used to group multiple statements into a single statement block, which can be used in other statement structures as a single statement. For example, a statement block can be used in an "IF ... ELSE ..." statement structure as a single statement.

The tutorial exercise below shows you how to use "BEGIN ... END" statement structures to place multiple statements into an "IF ... ELSE" statement structure:

```
DECLARE @site_name VARCHAR(40);
SET @site_name = 'SQA';
IF @site_name = 'DBA'
    BEGIN
        PRINT 'Dropping table: dba_links';
        DROP TABLE dba_links;
    END
ELSE IF @site_name = 'SQA'
    BEGIN
        PRINT 'Dropping table: sq_a_links';
        DROP TABLE sq_a_links;
    END
ELSE
    PRINT 'Unknown site name: '+@site_name;
GO
Dropping table: sq_a_links
```

[View All Answers](#)

Question - 222:

How To Use "IF ... ELSE IF ... ELSE ..." Statement Structures in MS SQL Server?

Ans:

"IF ... ELSE IF ... ELSE ..." statement structure is used to select one of the specified statements to be executed based on specified Boolean conditions. Here is the syntax of "IF ... ELSE IF ... ELSE ..." statement structure:

```
IF condition_1 statement_1;
ELSE IF condition_2 statement_2;
```

```
...
```

```
ELSE IF condition_n statement_n;
```

```
ELSE statement_o;
```

```
-- Executes statement_x is
```

```
if condition_x results in Boolean TRUE
```

The tutorial exercise below shows you how to use an IF ... ELSE statement structure to selectively execute one of the CREATE TABLE statements:

```
USE GlobalGuideLineDatabase
```

```
GO
```

```
DECLARE @site_name VARCHAR(40);
```

```
SET @site_name = 'SQA';
```

```
IF @site_name = 'DBA'
```

```
    CREATE TABLE dba_links (url VARCHAR(256));
```

```
ELSE IF @site_name = 'SQA'
```

```
    CREATE TABLE sq_a_links (url VARCHAR(256));
```

```
ELSE
```

```
    PRINT 'Unknown site name: '+@site_name;
```

```
GO
```

```
Command(s) completed successfully.
```

```
SELECT name FROM sys.tables WHERE name LIKE '%links';
```

```
GO
```

```
sq_a_links
```

[View All Answers](#)

Question - 223:

What Are Logical/Boolean Operations in MS SQL Server?

Ans:

Logical (Boolean) operations are performed on Boolean values with logical operators like 'AND', 'OR', or 'NOT'. Logical operations return Boolean values. SQL Server 2005 supports the following logical operations:

- * AND - Returns TRUE if both operands are TRUE.
- * OR - Returns TRUE if one of the operands is TRUE.
- * NOT - Returns TRUE if the only operand is FALSE.

Logical operations are commonly used to combine Boolean values resulted from comparison operations. The following tutorial exercise shows you a good example:



```
DECLARE @income MONEY;
DECLARE @marriage_status VARCHAR(10);
SET @income = 55000.00;
SET @marriage_status = 'Single';
SELECT CASE WHEN
    (@marriage_status = 'Married' AND @income < 65000)
    OR (@marriage_status = 'Single' AND @income < 35000)
    THEN 'Qualified for the benefit.'
    ELSE 'Not qualified for the benefit.'
END;
GO
Not qualified for the benefit.
```

[View All Answers](#)

Question - 224:

How To Test Values Returned by a Subquery with the IN Operator?

Ans:

Normally, the comparison operator IN is used against a list of specific values as in the format of: "test_value IN (value_1, value_2, ..., value_n)". But you can also replace the list of values by a subquery as the following formats:

```
test_value IN (SELECT column FROM ...)
-- Returns TRUE if the test_value equals to one of
the values returned from the subquery
test_value NOT IN (SELECT column FROM ...)
-- Returns TRUE if the test_value does not equal to any of
the values returned from the subquery
```

The following tutorial exercise shows you two examples of IN operators. The sample database AdventureWorksLT provided by Microsoft is used.

```
USE adventureWorksLT
GO
SELECT COUNT(*) FROM SalesLT.Customer c
WHERE c.CustomerID IN (
    SELECT s.CustomerID FROM SalesLT.SalesOrderHeader s
)
GO
32
SELECT COUNT(*) FROM SalesLT.Customer c
WHERE c.CustomerID NOT IN (
    SELECT s.CustomerID FROM SalesLT.SalesOrderHeader s
)
GO
408
```

[View All Answers](#)

Question - 225:

How To Test Subquery Results with the EXISTS Operator?

Ans:

EXISTS is a special operator used to test subquery results. EXISTS can be used in two ways:

```
EXISTS (SELECT ...)
-- Returns TRUE if the specified subquery has one or more rows returned.
NOT EXISTS (SELECT ...)
-- Returns TRUE if the specified subquery no rows returned.
```

The following tutorial exercise shows you two examples of EXISTS operators. The sample database AdventureWorksLT provided by Microsoft is used.

```
USE AdventureWorksLT
GO
-- Number of customers with orders
SELECT COUNT(*) FROM SalesLT.Customer c
WHERE EXISTS (
    SELECT * FROM SalesLT.SalesOrderHeader s
    WHERE s.CustomerID = c.CustomerID
)
GO
32
-- Number of customers without orders
SELECT COUNT(*) FROM SalesLT.Customer c
WHERE NOT EXISTS (
    SELECT * FROM SalesLT.SalesOrderHeader s
    WHERE s.CustomerID = c.CustomerID
)
GO
408
```

[View All Answers](#)

Question - 226:

How To Use Wildcard Characters in LIKE Operations in MS SQL Server?

Ans:

Wildcard character '%' can be used in the pattern string for the LIKE operator to match any string of zero or more characters. The following example uses '%Sport% Store%' to search all company names that has a partial word 'Sport' and ended with the word 'Store'. The sample database AdventureWorksLT provided by Microsoft is used.



```
USE adventureWorksLT
GO
SELECT c.CompanyName FROM SalesLT.Customer c
WHERE c.CompanyName LIKE '%Sport% Store'
GO
CompanyName
-----
```

Specialty Sports Store
Camping and Sports Store
Vigorous Sports Store
Our Sporting Goods Store
Sports Store
Sports Products Store

[View All Answers](#)

Question - 227:

What To Perform Pattern Match with the LIKE Operator?

Ans:

Pattern match is a very important operation for search records base on character string columns. SQL Server 2005 offers the LIKE operator to perform pattern match operations in two formats:

target_string LIKE pattern

-- Returns TRUE

if the target string matches the pattern

target_string NOT LIKE pattern

-- Returns TRUE

if the target string does not match the pattern

Pattern match is a powerful operation. But you need to remember several rules:

- * Pattern may contain predefined wildcard characters, similar to Unix Regular Expressions. But they are not the same.
- * '%' is a wildcard character that matches any string of zero or more characters.
- * '_' is a wildcard character that matches any single character.
- * '[' is a wildcard character that matches any single character.
- * '[abc]' is a wildcard character that matches any character listed inside the brackets.
- * '[a-c]' is a wildcard character that matches any character in the range defined in the brackets.
- * '[^abc]' is a wildcard character that matches any character not listed inside the brackets.
- * '[^a-c]' is a wildcard character that matches any character not in the range defined in the brackets.

[View All Answers](#)

Question - 228:

Can Binary Strings Be Converted into NUMERIC or FLOAT Data Types?

Ans:

Can binary strings be converted into numeric or float data types? The answer is no. Binary strings can not be converted implicitly or explicitly into NUMERIC, DECIMAL, REAL, or FLOAT data types. The tutorial exercise gives you some examples of errors when converting binary strings to NUMERIC or FLOAT data types:

-- Implicit conversion to NUMERIC

```
SELECT 0x66 + 0.44;
```

GO

Msg 8114, Level 16, State 5, Line 1

Error converting data type varbinary to numeric.

-- Explicit conversion to NUMERIC

```
SELECT CONVERT(NUMERIC(9,2), 0x66) + 0.44;
```

GO

Msg 8114, Level 16, State 5, Line 1

Error converting data type varbinary to numeric.

-- Implicit conversion to REAL

```
DECLARE @real REAL;
```

```
SET @real = 0x66;
```

Msg 206, Level 16, State 2, Line 2

Operand type clash: varbinary is incompatible with real

-- Implicit conversion to FLOAT

```
DECLARE @float FLOAT(53);
```

```
SET @float = 0x66;
```

Msg 206, Level 16, State 2, Line 2

Operand type clash: varbinary is incompatible with float

[View All Answers](#)

Question - 229:

How To Convert Binary Strings into Integers in MS SQL Server?

Ans:

Binary strings and integers are convertible implicitly and explicitly. But there several rules you need to remember:

- * Binary strings will be implicitly converted into an integer data type, if it is involved in an arithmetical operation with another integer data type operand.
- * Binary strings will be implicitly converted into an integer data type, if it is assigned to a variable, a column or a parameter of an integer data type.
- * Binary strings will be explicitly converted into integer data types using CAST() and CONVERT() functions.
- * When converting binary strings that have more bytes than the target data type size, bytes on the left hand side will be truncated.
- * When converting binary strings that have less bytes than the target data type size, 0x00 will be padded on the left hand side.

Examples showing in the tutorial exercise below will help you remembering those rules.

```
SELECT 0x66 + 44
```




```
GO
146
DECLARE @integer INT;
SET @integer = 0x66;
SELECT @integer + 44
GO
146
SELECT CAST(0x66 AS INT) + 44
GO
146
SELECT CONVERT(INT, 0x66) + 44
GO
146
-- Only last 4 bytes are used for INT conversion
SELECT 0x7700000066 + 44
GO
146
-- 8 bytes will be used for BIGINT conversion
SELECT 0x7700000066 + CONVERT(BIGINT,44)
```

[View All Answers](#)

Question - 230:

Can Binary Strings Be Used in Arithmetical Operations?

Ans:

Can binary strings be used in arithmetical operations? The answer is yes. But there are two simple rules you need to remember:

- * If an arithmetical operation has one binary string operand and one integer data type operand, the binary string operand will be converted to a integer data type to match the other operand. The operation will be performed as an integer operation.

- * A + operator with two binary strings will be performed as binary string concatenation.

- * A -, *, or / operator with two binary strings will be performed as binary string concatenation.

The tutorial exercise below shows you some good examples:

```
SELECT 0x66 + 44
GO
146
SELECT 0x66 - 44
GO
58
SELECT 0x66 * 44
GO
4488
SELECT 0x66 / 44
GO
2
SELECT 0x66 + 0x44
GO
0x6644
```

[View All Answers](#)

Question - 231:

How To Concatenate Two Binary Strings Together?

Ans:

SQL Server 2005 allows to concatenate two binary strings into a single string with the (+) operator. The following tutorial exercise shows you some binary string concatenation examples:

-- Concatenating two binary string literals

```
SELECT 0x57656C636F6D6520746F20
+ 0x46594963656E7465722E636F6D;
```

GO

```
0x57656C636F6D6520746F2046594963656E7465722E636F6D
```

-- Watch out: This is not a binary string concatenation

```
SELECT '0x57656C636F6D6520746F20'
+ '0x46594963656E7465722E636F6D';
```

GO

```
0x57656C636F6D6520746F200x46594963656E7465722E636F6D
```

-- Concatenating two binary strings

```
SELECT CONVERT(VARBINARY(40),'Welcome to ')
+ CONVERT(VARBINARY(40),'GlobalGuideLine.com');
```

GO

```
0x57656C636F6D6520746F2046594963656E7465722E636F6D
```

-- Binary strings can not be concatenated

with character strings

```
SELECT 'Welcome to '
+ 0x46594963656E7465722E636F6D;
```

GO

Msg 402, Level 16, State 1, Line 1

The data types varchar and varbinary are incompatible in the add operator.

[View All Answers](#)



Question - 232:

How To Locate and Take Substrings with CHARINDEX() and SUBSTRING() Functions?

Ans:

Transact-SQL is not a language designed for manipulating strings, but it does have two simple functions to locate and take substrings: CHARINDEX() and SUBSTRING(). The tutorial exercise below assumes two given strings: 'Pages: 18' and 'Words: 3240'. The objective is to calculate the number of words per page. Read the script below to see how this is done by using CHARINDEX() and SUBSTRING() functions:

```
DECLARE @sPages VARCHAR(40), @sWords VARCHAR(40);
SET @sPages = 'Pages: 18';
SET @sWords = 'Words: 3240';
SET @sPages = SUBSTRING(@sPages, CHARINDEX(':', @sPages)+1, 20);
SET @sWords = SUBSTRING(@sWords, CHARINDEX(':', @sWords)+1, 20);
PRINT 'Number of words per page: '
      + CONVERT(VARCHAR(20), CONVERT(INT, @sWords)/CONVERT(INT, @sPages));
GO
Number of words per page: 180
If you are a PHP developer, you can get this done in a much quick way.
```

[View All Answers](#)

Question - 233:

How To Insert New Line Characters into Strings?

Ans:

If you want to break a string into multiple lines, you need to insert new line characters into the string. With some client tools like SQL Server Management Studio, it is not so easy to insert a new line character. One work around is to use the CHAR(int) function to generate new line character and other special characters with their code values:

- * CHAR(9) - Generates the tab character.
- * CHAR(10) - Generates the line feed (new line) character.
- * CHAR(13) - Generates the carriage return character.

```
The tutorial examples below gives you a good example
PRINT 'Welcome to '+CHAR(10)+'GlobalGuideLine.com';
PRINT CHAR(10);
PRINT 'Current date and time is '
      +CONVERT(VARCHAR(20), GETDATE());
GO
Welcome to
GlobalGuideLine.com
Current date and time is May 19 2007 7:30PM
```

[View All Answers](#)

Question - 234:

What Are the Character String Functions Supported by SQL Server 2005?

Ans:

SQL Server 2005 supports 23 character string functions:

- * ASCII(char) - Returning the code value of a non-Unicode character.
- * CHAR(int) - Returning the non-Unicode character of a code value.
- * CHARINDEX(word, string, start_location) - Returning the location of the searched "word" in a string.
- * DIFFERENCE(string1, string2) - Returning an integer value that indicates the difference between the SOUNDEX values of two strings.
- * LEFT(string, length) - Returning a substring of "length" character from the left hand side.
- * LEN(string) - Returning the number of characters in the string.
- * LOWER(string) - Returning the same string with all upper case characters converted to lower case.
- * LTRIM(string) - Returning the same string with leading spaces removed.
- * NCHAR(int) - Returning the Unicode character of a code value.
- * PATINDEX(pattern, string) - Returning the location of the "pattern" in a string.
- * QUOTENAME(string, quote) - Returning the same string enclosed in "quote".

[View All Answers](#)

Question - 235:

How To Convert a Unicode Strings to Non-Unicode Strings?

Ans:

Since Unicode character set is different than code page based (non-Unicode) character set, converting Unicode strings to non-Unicode strings may result in wrong characters or missing characters. So you should avoid converting Unicode strings to non-Unicode strings. If you really want to, there are 3 ways to convert a Unicode string to a non-Unicode string:

- * Implicit conversion by assignment operations - When a Unicode string is assigned to a variable, a column, or a parameter of a non-Unicode string data type, SQL Server will implicitly convert the Unicode string to a non-Unicode string.

- * Explicit conversion using the CAST() function - A Unicode string can be explicitly converted to non-Unicode string using the CAST(Unicode_string AS VARCHAR(size)) function.

- * Explicit conversion using the CONVERT() function - A Unicode string can be explicitly converted to non-Unicode string using the CONVERT(VARCHAR(size), Unicode_string) function.

Some numeric data type conversion examples are provided in the tutorial exercise below:

```
-- Implicit conversion by an assignment operation
DECLARE @regcode VARCHAR(40);
SET @regcode = N'Some Unicode characters: '
      + NCHAR(9733)+NCHAR(9734)+NCHAR(9792)+NCHAR(9794);
SELECT @regcode;
GO
```



Some Unicode characters: ????

[View All Answers](#)

Question - 236:

What Happens When Unicode Strings Concatenate with Non-Unicode Strings?

Ans:

If a Unicode string NVARCHAR is concatenated with a non-Unicode string VARCHAR, SQL Server will implicitly convert the non-Unicode string to Unicode string for concatenation.

```
DECLARE @regcode VARCHAR(40);
DECLARE @unicode NVARCHAR(40);
SET @regcode = 'Some Unicode characters: '
SET @unicode = NCHAR(9733)+NCHAR(9734)+NCHAR(9792)
+NCHAR(9794);
SELECT @regcode + @unicode;
SELECT DATALENGTH(@regcode);
SELECT DATALENGTH(@unicode);
SELECT DATALENGTH(@regcode + @unicode);
Some Unicode characters: ????
```

25

8

58

Note that the non-Unicode string @regcode has been converted to a Unicode string. The number of bytes of @regcode changed from 25 to 50. With 8 bytes from @unicode, the number of bytes of the concatenated string becomes 58.

[View All Answers](#)

Question - 237:

How To Concatenate Two Character Strings Together?

Ans:

Concatenating two character strings together is most commonly used string operation. SQL Server 2005 allows to concatenate two character strings into a single string with the (+) operator. The following tutorial exercise shows you some string concatenation examples:

```
DECLARE @site VARCHAR(40);
SET @site = 'GlobalGuideLine.com';
SELECT 'Welcome to '+@site;
SELECT 'Current date and time is '
+CONVERT(VARCHAR(20), GETDATE());
GO
Welcome to GlobalGuideLine.com
Current date and time is May 19 2007 5:18PM
DECLARE @start INT, @end INT, @total INT;
SET @start = 21;
SET @end = 30;
SET @total = 728;
SELECT 'Search result '
+ CONVERT(VARCHAR(20),@start)
+ ' - '
+ CONVERT(VARCHAR(20),@end)
+ ' of '
+ CONVERT(VARCHAR(20),@total);
GO
```

Search result 21 - 30 of 728

[View All Answers](#)

Question - 238:

How To Generate Random Numbers with the RAND() Function in MS SQL Server?

Ans:

Random numbers are very useful for generating test data, passwords, or other security related data. SQL Server 2005 offers you the random number generator function RAND in two format:

* RAND(seed) - Starting a new sequence of random numbers based on the given integer "seed" and returning the first random number in FLOAT(53) from the sequence.

* RAND() - Returning the next random number in FLOAT(53) from the current sequence. If there has been no current sequence, SQL Server will start a new sequence with a random "seed".

Note that calling RAND(seed) with the same seed will start the same sequence and return the same number. To avoid this repeating pattern, you should always call RAND() without any seed and let the server to randomly pickup a sequence. The tutorial exercise below shows some good examples on how to generate random numbers:

```
SELECT RAND(100), RAND(), RAND(); -- new sequence
SELECT RAND(100), RAND(), RAND(); -- same sequence again
SELECT RAND(), RAND(), RAND();
SELECT RAND(), RAND(), RAND();
GO
0.715436657367485 0.28463380767982 0.0131039082850364
0.715436657367485 0.28463380767982 0.0131039082850364
0.28769876521071 0.100505471175005 0.292787286982702
0.868829058415689 0.370366365964781 0.58334760467751
```

[View All Answers](#)



Question - 239:

How To Round a Numeric Value To a Specific Precision?

Ans:

Sometimes you need to round a numeric value to a specific precision. For example, you may want to round values in your financial statement to the precision of 1000.00. This can be done by the ROUND() function with the following syntax:

ROUND(value, precision, type)

value: The input value to be rounded.

precision: The location of the precision digit relative to the decimal point.

type: 0 - Round to nearest value;

1 - Truncate to a lower value.

The tutorial exercise below gives some good examples of how to use the ROUND() function:

```
SELECT ROUND(1234.5678, 0, 0);
```

```
SELECT ROUND(1234.5678, -3, 0);
```

```
SELECT ROUND(1234.5678, -4, 0);
```

```
SELECT ROUND(1234.5678, 3, 0);
```

```
SELECT ROUND(1234.5678, 3, 1);
```

```
GO
```

```
1235.0000
```

```
1000.0000
```

```
0.0000
```

```
1234.5680
```

```
1234.5670
```

[View All Answers](#)

Question - 240:

How To Convert Numeric Values to Integers in MS SQL Server?

Ans:

Sometimes you need to round a numeric value into an integer. SQL Server 2005 offers you a number of ways to do this:

- * FLOOR(value) - Returning the largest integer less than or equal to the input value. The returning data type is the same as the input value.

- * CEILING(value) - Returning the smallest integer greater than or equal to the input value. The returning data type is the same as the input value.

- * ROUND(value, 0, 0) - Returning the integer most close to the input value. The returning data type is the same as the input value.

- * CAST(value AS INT) - Returning the largest integer less than or equal to the input value. The returning data type is INT.

- * CONVERT(INT, value) - Returning the largest integer less than or equal to the input value. The returning data type is INT.

The tutorial exercise below gives some good examples of converting numeric values to integers:

```
<pre> SELECT FLOOR(1234.5678);
```

```
SELECT CEILING(1234.5678);
```

```
SELECT ROUND(1234.5678, 0, 0);
```

```
SELECT CAST(1234.5678 AS INT);
```

```
SELECT CONVERT(INT, 1234.5678);
```

```
GO
```

```
1234
```

```
1235
```

```
1235.0000
```

```
1234
```

```
1234</pre>
```

[View All Answers](#)

Question - 241:

What Are the Mathematical Functions Supported by SQL Server 2005?

Ans:

SQL Server 2005 supports 23 mathematical functions: ABS, ACOS, ASIN, ATAN, ATN2, CEILING, COS, COT, DEGREES, EXP, FLOOR, LOG, LOG10, PI, POWER, RADIANS, RAND, ROUND, SIGN, SIN, SQRT, SQUARE, and TAN. The return data types of mathematical functions are determined by two rules:

- * Arithmetical functions: ABS, CEILING, DEGREES, FLOOR, POWER, RADIANS, and SIGN are returning the same data type as the input values. No implicit data type conversions are performed.

- * Other mathematical functions: COS, COT, EXP, LOG, LOG10, SIN, SQRT, SQUARE and TAN are taking only FLOAT data type as input and returning FLOAT data type. Implicit data type conversion will be performed if you are calling these functions with a non-FLOAT data type.

The tutorial exercise gives you some good examples on how to use mathematical functions:

```
<pre> -- ABS returns the same data type as the input
```

```
DECLARE @x FLOAT(53);
```

```
DECLARE @y NUMERIC(9,2);
```

```
DECLARE @z INT;
```

```
SET @x = -12345.123456789E+20;
```

```
SET @y = -12345.12;
```

```
SET @z = -12345
```

```
SELECT ABS(@x);
```

```
SELECT ABS(@y);
```

```
SELECT ABS(@z);
```

```
GO
```

```
1.2345123456789E+24
```

```
12345.12
```

```
12345</pre>
```

[View All Answers](#)

Question - 242:



What Happens When Converting Big Values to NUMERIC Data Types?

Ans:

If you are converting a numeric expression to a NUMERIC data type and the value is too big for the storage size, you will get an arithmetic overflow error as shown in the following examples:

-- Overflow error on implicit conversion

```
DECLARE @x FLOAT(24);
```

```
DECLARE @y NUMERIC(5,2);
```

```
SET @x = 12345.12E+00;
```

```
SET @y = @x;
```

```
GO
```

Msg 8115, Level 16, State 6, Line 4

Arithmetic overflow error converting real to data type numeric.

-- Overflow error on explicit conversions

```
DECLARE @x FLOAT(24);
```

```
SET @x = 12345.12E+00;
```

```
SELECT CAST(@x AS NUMERIC(5,2));
```

```
SELECT CONVERT(NUMERIC(5,2),@x);
```

```
GO
```

Msg 8115, Level 16, State 6, Line 3

Arithmetic overflow error converting real to data type numeric.

Msg 8115, Level 16, State 6, Line 4

Arithmetic overflow error converting real to data type numeric.

[View All Answers](#)

Question - 243:

What Happens When Converting Big Values to Integers?

Ans:

If you are converting a numeric expression to an integer data type and the value is too big for integer storage size, you will get an arithmetic overflow error as shown in the following examples:

-- Overflow error on implicit conversion

```
DECLARE @x FLOAT(24);
```

```
DECLARE @y TINYINT;
```

```
SET @x = 12345.12E+00;
```

```
SET @y = @x;
```

```
GO
```

Msg 232, Level 16, State 1, Line 4

Arithmetic overflow error for type tinyint, value = 12345.120117.

-- Overflow error on explicit conversions

```
DECLARE @x FLOAT(24);
```

```
SET @x = 12345.12E+00;
```

```
SELECT CAST(@x AS TINYINT);
```

```
SELECT CONVERT(TINYINT, @x);
```

```
GO
```

Msg 232, Level 16, State 1, Line 4

Arithmetic overflow error for type tinyint, value = 12345.120117.

Msg 232, Level 16, State 1, Line 5

Arithmetic overflow error for type tinyint, value = 12345.120117.

[View All Answers](#)

Question - 244:

How To Convert Character Strings into Numeric Values?

Ans:

Sometimes you need to convert numeric values enclosed in character strings back to numeric values by using the CONVERT() function. When converting character strings to values with CONVERT(), you need to remember two rules:

- * Leading and trailing space characters are ignored.

- * The input string must represent a numeric literal that matches the target numeric data type.

- * If the input string does not represent a numeric literal that matches the target numeric data type, you will get a conversion error.

The tutorial exercise below shows you how to use the CONVERT() function to convert strings to values:

```
SELECT CONVERT(NUMERIC(10,5), '12345.12');
```

```
GO
```

12345.12000

-- Input does not match the target data type

```
SELECT CONVERT(INT, '12345.12');
```

```
GO
```

Msg 245, Level 16, State 1, Line 1

Conversion failed when converting the varchar value '12345.12' to data type int.

-- Input does not match the target data type

```
SELECT CONVERT(NUMERIC(10,5), '12345.12E+00');
```

```
GO
```

Msg 8114, Level 16, State 5, Line 1



Error converting data type varchar to numeric.

-- Double conversion works

```
SELECT CONVERT(NUMERIC(10,5),
    CONVERT(FLOAT(24), ' 12345.12E+00 '));
GO
12345.12012
SELECT CONVERT(FLOAT(24), ' 12345.12E+00 ');
```

[View All Answers](#)

Question - 245:

How To Convert Numeric Expression Data Types using the CONVERT() Function??

Ans:

If you want to convert the data type of a numeric expression to a new data type, you can use the CONVERT(data_type, expression) function. The tutorial exercise below shows you how to use the CONVERT() function:

-- FLOAT converted to NUMERIC by CONVERT()

```
DECLARE @pi FLOAT(24);
SET @pi = 3.141592E+00;
SELECT CONVERT(NUMERIC(5,2), @pi);
GO
3.14
```

-- FLOAT converted to NUMERIC by CONVERT()

```
DECLARE @x FLOAT(24);
SET @x = 12345.12E+00;
SELECT CONVERT(NUMERIC(10,5), @x);
GO
12345.12012
```

-- FLOAT converted to INT by CONVERT()

```
DECLARE @x FLOAT(24);
SET @x = 12345.12E+00;
SELECT CONVERT(INT, @x);
GO
12345
```

[View All Answers](#)

Question - 246:

How To Convert Numeric Expression Data Types using the CAST() Function?

Ans:

If you want to convert the data type of a numeric expression to a new data type, you can use the CAST(expression AS data_type) function. The tutorial exercise below shows you how to use the CAST() function:

-- FLOAT converted to NUMERIC by CAST()

```
DECLARE @pi FLOAT(24);
SET @pi = 3.141592E+00;
SELECT CAST(@pi AS NUMERIC(5,2));
GO
3.14
```

-- FLOAT converted to NUMERIC by CAST()

```
DECLARE @x FLOAT(24);
SET @x = 12345.12E+00;
SELECT CAST(@x AS NUMERIC(10,5));
GO
12345.12305
```

-- FLOAT converted to INT by CAST()

```
DECLARE @x FLOAT(24);
SET @x = 12345.12E+00;
SELECT CAST(@x AS INT);
GO
12345
```

[View All Answers](#)

Question - 247:

How To Convert Numeric Expression Data Types by Assignment Operations?

Ans:

An assignment operation is used to assign an expression to a variable, a column, or a parameter. If the data type of the expression does not match the data type of the receiving variable, column, or parameter, SQL Server will perform an implicit data type conversion on the expression. Note that implicit data type conversion during assignment operation can convert a higher rank data type to a lower rank data type, which may result in losing data during the conversion. The tutorial exercise shows you some good examples: -- INT converted to NUMERIC DECLARE @i INT; DECLARE @d NUMERIC(9,3); SET @i = 123; SET @d = @i; SELECT @d; GO 123.000 -- INT converted to NUMERIC DECLARE @i INT; DECLARE @d NUMERIC(9,3); SET @i = 123; SET @d = @i; SELECT @d; GO 123.000 DECLARE @pi FLOAT(24); DECLARE @dp NUMERIC(5,2); SET @pi = 3.1415927E+00; SET @dp = @pi; SELECT @dp; GO 3.14

[View All Answers](#)

Question - 248:

How To Convert a Numeric Expression from One Data Type to Another?

Ans:



There are 4 ways to convert a numeric expression from one data type to another data type:

- * Implicit conversion by arithmetic operations - When arithmetic operations are performed on expressions of different data types, implicit data type conversion will be performed before the arithmetic operation on the expression with a lower data type rank.

- * Implicit conversion by assignment operations - When a numeric expression is assigned to variable, column, or parameter of different data type, the expression will be converted to match the data type of the variable, column, or parameter.

- * Explicit conversion using the CAST() function - A numeric expression can be explicitly converted to different data type using the CAST(expression AS data_type) function.

- * Explicit conversion using the CONVERT() function - A numeric expression can be explicitly converted to different data type using the CONVERT(data_type, expression) function.

Some numeric data type conversion examples are provided in the tutorial exercise below:

-- Implicit conversion by an arithmetic operation

-- INT converted to NUMERIC

```
DECLARE @d NUMERIC(9,6);
```

```
SET @d = 1.0;
```

```
SELECT @d/3;
```

```
GO
```

```
0.33333333
```

-- Implicit conversion by an assignment operation

-- NUMERIC converted to INT

```
DECLARE @i INT;
```

```
DECLARE @d NUMERIC(9,3);
```

```
SET @d = 123.456;
```

```
SET @i = @d;
```

```
SE
```

[View All Answers](#)

Question - 249:

What Is an Expression in MS SQL Server?

Ans:

A numeric expression is a combination of identifiers, values, and operators that SQL Server 2005 can evaluate to obtain a numeric value. A simple expression could be a constant, a function, a column name, a variable, or a subquery without any operators. Complex expressions can be constructed by joining other expressions with operators. The following tutorial exercise shows you some expression examples:

```
DECLARE @site VARCHAR(40);
```

```
SET @site = 'GlobalGuideLine.com';
```

```
SELECT 'Welcome'; -- Expression: constant
```

```
SELECT @site; -- Expression: variable
```

```
SELECT GETDATE(); -- Expression: function
```

```
SELECT 'Welcome to '+@site; -- Expression with an operator
```

```
GO
```

```
Welcome
```

```
GlobalGuideLine.com
```

```
2007-05-19 18:42:09.077
```

```
Welcome to GlobalGuideLine.com
```

```
DECLARE @rate NUMERIC(5,2);
```

```
DECLARE @deposit MONEY;
```

```
DECLARE @value MONEY;
```

```
SET @rate = 5.25;
```

```
SET @deposit = 4000.00;
```

-- Expression with multiple operators.

```
SET @value = @deposit*(1.0+@rate/100)*(1.0+@rate/100);
```

```
PRINT @value;
```

```
GO
```

```
4431.03
```

[View All Answers](#)

Question - 250:

What Are the Underflow and Overflow Behaviors on FLOAT Literals?

Ans:

If you enter a floating number that is too big or too small for the FLOAT data type, SQL Server 2005 will behave as:

- * FLOAT(24) Underflow: If a floating number is too small for FLOAT(24), it will be stored as 0 without any warning.

- * FLOAT(24) Overflow: If a floating number is too big for FLOAT(24), you will get an arithmetic overflow error.

- * FLOAT(53) Underflow: If a floating number is too small for FLOAT(53), it will be stored as 0 with a warning.

- * FLOAT(53) Overflow: If a floating number is too big for FLOAT(53), you will get a value-out-of-range error.

The tutorial exercise below some good underflow and overflow examples:

-- Single precision underflow without warning

```
DECLARE @x REAL; -- FLOAT(24)
```

```
SET @x = 9.234568E-39;
```

```
SELECT @x;
```

```
GO
```

```
0
```

-- Single precision overflow error

```
DECLARE @x REAL; -- FLOAT(24)
```

```
SET @x = 9.234568E+39;
```

```
GO
```

```
Msg 232, Level 16, State 2, Line 2
```

```
Arithmetic overflow error for type real,
```

```
value = 923456800000000040000000000000000000000000000.000000.
```

-- Double precision underflow with warning



```
DECLARE @x FLOAT(53);
SET @x = 9.23456789012346E-309
SELECT @x;
GO
Warning: the floating point value '9.23456789012346E-309'
is too small. It will be interpreted as 0.
0
-- Double precision overflow error
```

[View All Answers](#)

Question - 251:

How REAL and FLOAT Literal Values Are Rounded?

Ans:

By definition, FLOAT(n) should store the mantissa of the floating number in n bits. For example, FLOAT(16) should have a precision one-byte less than FLOAT(24). However, SQL Server 2005 only supports two precisions for floating numbers:

- * Single Precision: FLOAT(24) or REAL, stored in 4 bytes, giving about 7 digits of precision, covering all types from FLOAT(1) to FLOAT(24),
- * Double Precision: FLOAT(53), stored in 8 bytes, giving about 15 digits of precision, covering all types from FLOAT(25) to FLOAT(53).

The tutorial exercise below shows you some different precision and rounding examples:

```
-- FLOAT(1) works like FLOAT(24)
DECLARE @x FLOAT(1)
SET @x = 9.234567890E+10;
SELECT @x;
GO
9.234568E+10 -- 7 digits precision
-- Single precision with rounding
DECLARE @x REAL; -- FLOAT(24)
SET @x = 9.234567890E+10;
SELECT @x;
GO
9.234568E+10 -- 7 digits precision
-- FLOAT(25) works like FLOAT(53)
DECLARE @x FLOAT(25);
SET @x = 9.2345678901234567890E+100;
SELECT @x;
GO
9.23456789012346E+100 -- 15 digits precision
-- Double precision with rounding
DECLARE @x FLOAT(53);
SET @x = 9.2345678901234567890E+100;
SELECT @x;
GO
9.23456789012346E+100 -- 15 digits precision
```

[View All Answers](#)

Question - 252:

How Extra Digits Are Handled with NUMERIC Data Type Literals?

Ans:

Exact numeric data types defined with NUMERIC(p,s) has two limits defined by two parameters: p (precision) and s (scale):

* Maximum number of digits of the integer part (digits before the decimal point) is defined as p-s. If this limit is passed, SQL Server will give you an arithmetic overflow error.

* Maximum number of digits of the decimal part (digits after the decimal point) is defined as s. If this limit is passed, SQL Server will perform a round operation.

The tutorial exercise below gives an example of arithmetic overflow errors and rounding operations.

```
-- Exact numeric value
DECLARE @x NUMERIC(9,2); -- NUMERIC(p,s)
SET @x = 1234567.12;
SELECT @x;
GO
1234567.12
-- Overflow error: p-s limit passed
DECLARE @x NUMERIC(9,2); -- NUMERIC(p,s)
SET @x = 123456789.12;
GO
Msg 8115, Level 16, State 8, Line 2
Arithmetic overflow error converting numeric to data type
numeric.
-- Rounding on extra decimal digits: s limit passed
DECLARE @x NUMERIC(9,2); -- NUMERIC(p,s)
SET @x = 1234567.12345;
SELECT @x;
GO
1234567.12
```

[View All Answers](#)

Question - 253:

What Happens If an Integer Is Too Big for INT Data Type?

Ans:



If you are entering an INT data type literal with representing an integer value too big for INT data type to store, the SQL Server will give you an arithmetic overflow error. The same error will happen on BIGINT, INT, SMALLINT, and TINYINT data types. Remember that INT data types uses 4 bytes to an integer value between -2³¹ (-2,147,483,648) to 2³¹-1 (2,147,483,647). The tutorial exercise below gives an example of arithmetic overflow errors.

-- INT value in the range

```
DECLARE @x INT;
SET @x = 2147483647;
SELECT @x;
GO
```

2147483647

-- INT value overflow

```
DECLARE @x INT;
SET @x = 2147483648;
```

Msg 8115, Level 16, State 2, Line 2

Arithmetic overflow error converting expression to data type int.

[View All Answers](#)

Question - 254:

What Are Out-of-Range Errors with Date and Time Literals?

Ans:

When you enter data and time literals, you may get out-of-range errors due to two common mistakes:

- * The date value is a valid calendar date, but it is not in the range covered by DATETIME data type: from January 1, 1753, to December 31, 9999.
- * The date value is not a valid calendar date, for example: 30-Feb-2007.
- * The time value does not represent a valid time, for example: 24:55:07.233.

The tutorial exercise below shows you some data and time out-of-range errors:

-- Invalid date

```
DECLARE @x DATETIME;
SET @x = '30-Feb-2007 22:55:07.233';
GO
```

Msg 242, Level 16, State 3, Line 2

The conversion of a char data type to a datetime data type resulted in an out-of-range datetime value.

-- Date below the DATETIME limits

```
DECLARE @x DATETIME;
SET @x = '19-May-1752 22:55:07.233';
GO
```

Msg 242, Level 16, State 3, Line 2

The conversion of a char data type to a datetime data type resulted in an out-of-range datetime value.

-- Invalid time

```
DECLARE @x DATETIME;
SET @x = '19-May-2007 24:55:07.233';
GO
```

Msg 242, Level 16, State 3, Line 2

The conversion of a char data type to a datetime data type resulted in an out-of-range datetime value.

[View All Answers](#)

Question - 255:

What Happens If Time-Only Values Are Provided as Date and Time Literals?

Ans:

If only time value is provided in a data and time literal, the SQL Server will pad the date value with a zero, representing the base date, January 1, 1900. The tutorial exercise below gives you some good examples:

-- 'hh:mi:ss.mmm' format

```
DECLARE @x DATETIME;
SET @x = '22:55:07.233';
SELECT @x;
GO
```

1900-01-01 22:55:07.233

-- 'hh:mi:ss.mmmAM/PM' format

```
DECLARE @x DATETIME;
SET @x = '10:55:07.233PM';
SELECT @x;
GO
```

1900-01-01 22:55:07.233

-- 'hh:miAM/PM' format

```
DECLARE @x DATETIME;
SET @x = '10:55PM';
SELECT @x;
GO
```

1900-01-01 22:55:00.000

[View All Answers](#)

Question - 256:

What Happens If Date-Only Values Are Provided as Date and Time Literals?

Ans:

If only date value is provided in a data and time literal, the SQL Server will pad the time value with a zero, or '00:00:00.000', representing the midnight time of the



day. The tutorial exercise below gives you some good examples:

```
-- 'mm/dd/yyyy' format
DECLARE @x DATETIME;
SET @x = '05/19/2007';
SELECT @x;
GO
2007-05-19 00:00:00.000
-- 'mm.dd.yy' format
DECLARE @x DATETIME;
SET @x = '05.19.07';
SELECT @x;
GO
2007-05-19 00:00:00.000
-- 'yyyy-mm-dd' format
DECLARE @x DATETIME;
SET @x = '2007-05-19';
SELECT @x;
GO
2007-05-19 00:00:00.000
-- 'mon dd, yyyy' format
DECLARE @x DATETIME;
SET @x = 'May 19, 2007';
SELECT @x;
GO
2007-05-19 00:00:00.000
-- 'dd-mon-yyyy' format
DECLARE @x DATETIME;
SET @x = '19-May-2007';
SELECT @x;
GO
2007-05-19 00:00:00.000
```

[View All Answers](#)

Question - 257:

Why I Can Not Enter 0.001 Second in Date and Time Literals in MS SQL Server?

Ans:

If you enter milliseconds in data and time literals, they will be rounded up to 10/3 milliseconds increments, because DATETIME data type uses 4 bytes to store the time of the day. A 4-byte integer can only give an accuracy of one three-hundredth second, or 3.33 milliseconds. So if you enter a time with 0.001 second, it will be rounded to 0.000 second. The tutorial exercise below gives you some good examples of how milliseconds are rounded by the SQL Server.

```
-- No rounding
DECLARE @x DATETIME;
SET @x = '2007-05-19 22:55:07.233';
SELECT @x;
GO
2007-05-19 22:55:07.233
-- Rounded down to 0.000
DECLARE @x DATETIME;
SET @x = '2007-05-19 22:55:07.001';
SELECT @x;
GO
2007-05-19 22:55:07.000
-- Rounded up to 0.003
DECLARE @x DATETIME;
SET @x = '2007-05-19 22:55:07.002';
SELECT @x;
GO
2007-05-19 22:55:07.003
-- Rounded up to 0.007
DECLARE @x DATETIME;
SET @x = '2007-05-19 22:55:07.006';
SELECT @x;
GO
2007-05-19 22:55:07.007
```

[View All Answers](#)

Question - 258:

How To Enter Date and Time Literals in MS SQL Server?

Ans:

Date and time literals are entered as character strings enclosed in single quotes ('). The string must be in one of the recognizable data and time formats. Some of the most commonly used formats are given in the following tutorial exercise:

```
-- Default format for query output
DECLARE @x DATETIME;
SET @x = '2007-05-19 22:55:07.233';
SELECT @x;
GO
2007-05-19 22:55:07.233
-- Default for string print output
DECLARE @x DATETIME;
```



```
SET @x = 'May 19 2007 10:55PM';
SELECT @x;
GO
2007-05-19 22:55:00.000
-- Europe default format
DECLARE @x DATETIME;
SET @x = '19-May-2007 22:55:07.233';
SELECT @x;
GO
2007-05-19 22:55:07.233
-- ISO8601 standard format
DECLARE @x DATETIME;
SET @x = '2007-05-19T22:55:07.233';
SELECT @x;
GO
2007-05-19 22:55:07.233
```

[View All Answers](#)

Question - 259:

How To Enter Binary String Literals in MS SQL Server?

Ans:

Binary string long values are normally generated by client applications by reading input channels, like image files. But sometimes, you may need to enter some short binary strings as literals. Binary string literals are entered as a string of bytes expressed in HEX numbers and prefixed with (0x). Input strings will be truncated or padded to fit the storage size of fixed length binary string data type. The padding bytes will be the zero byte: 0x00. The tutorial exercise shows you good examples of truncating and padding fixed length binary strings:

```
-- Size matches
DECLARE @x BINARY(8);
SET @x = 0x2605260626402642;
PRINT @x;
GO
0x2605260626402642
-- Truncated
DECLARE @x BINARY(4);
SET @x = 0x2605260626402642;
PRINT @x;
GO
0x26052606
-- Padded
DECLARE @x BINARY(12);
SET @x = 0x2605260626402642;
PRINT @x;
GO
0x260526062640264200000000
-- No padding on variable length data type
DECLARE @x VARBINARY(8);
SET @x = 0x; -- Empty binary string
PRINT @x;
GO
0x
-- Padding on fixed length data type
DECLARE @x BINARY(8);
SET @x = 0x; -- Empty binary strings
PRINT @x;
GO
0x0000000000000000
```

[View All Answers](#)

Question - 260:

How To Enter Unicode Character String Literals in MS SQL Server?

Ans:

Unicode characters are multi-byte characters. They are very hard to be entered as string literals, because it requires:

- * The SQL client tool to support Unicode characters. The command line tool 'sqlcmd' does not support Unicode. But SQL Server Management Studio does support Unicode.

- * A special Unicode character input tool to allow you to enter multi-byte Unicode characters.

But if you know the code value of a Unicode character, you can enter it with the help of the NCHAR(int) function. For example:

```
-- A Unicode string of some interesting characters
PRINT NCHAR(9733)+NCHAR(9734)+NCHAR(9792)+NCHAR(9794);
GO
```

Run the sample statement with SQL Server Management Studio, you should see some Unicode characters.

[View All Answers](#)

Question - 261:

How Fixed Length Strings Are Truncated and Padded?

Ans:

When the length of the input string does not match the storage size of the fixed length string data type CHAR(n). SQL Server will:

- * If the input string of CHAR(n) has less than n bytes, it will be padded with space characters to become n bytes.



* If the input string of CHAR(n) has more than n bytes, it will be truncated to n bytes.

The tutorial exercise shows you good examples of truncating and padding fixed length character strings:

```
-- Length matches the data type size
DECLARE @msg CHAR(36);
SET @msg = 'Welcome to GlobalGuideLine.com SQL Server!';
PRINT ('+'@msg+');
GO
(Welcome to GlobalGuideLine.com SQL Server!)
-- Length is bigger than the data type size - truncated
DECLARE @msg CHAR(24);
SET @msg = 'Welcome to GlobalGuideLine.com SQL Server!';
PRINT ('+'@msg+');
GO
(Welcome to GlobalGuideLine.com)
-- Length is smaller than the data type size - padded
DECLARE @msg CHAR(46);
SET @msg = 'Welcome to GlobalGuideLine.com SQL Server!';
PRINT ('+'@msg+');
GO
(Welcome to GlobalGuideLine.com SQL Server!)
```

[View All Answers](#)

Question - 262:

How To Find Out What Is the Default Collation in a Database?

Ans:

The default collation of a database comes from the server if you are not using the COLLATE clause in the CREATE DATABASE statement. If you are not using the COLLATE clause for character string column, it will use the default collation from the database. Using this logic, you can find out the default collation in a database or server very easily. See the following tutorial exercise:

```
CREATE DATABASE GlobalGuideLineDatabase
GO
CREATE TABLE faq (Name VARCHAR(80))
GO
SP_HELP FAQ
GO
Column_name ... Collation
Name ... SQL_Latin1_General_CP1_CI_AS
```

Now we know that the default collation for SQL Server 2005 is SQL_Latin1_General_CP1_CI_AS.

[View All Answers](#)

Question - 263:

What Happens If Strings Are Casted into Wrong Code Pages in MS SQL Server?

Ans:

In SQL Server, different collations may use different code pages. For example:

- * Albanian_CI_AI_KS_WS - Albanian, Code page 1250.
- * Arabic_CI_AS_KS_WS - Arabic, Code page 1256.
- * French_CI_AI - French, Code page 1252.
- * Korean_Wansung_BIN - Korean-Wansung, Code page 949.
- * SQL_Latin1_General_CP1250_CI_AS - Latin1-General, Code page 1250.

If you are casting a string of characters from one code page to a different code page, some character will be converted to similar. For example

```
PRINT 'Fran?is: e??a?o?';
-- The default code page
PRINT 'Fran?is: e??a?o?'
COLLATE French_CI_AI; -- Code page 1252
PRINT 'Fran?is: e??a?o?'
COLLATE Polish_CI_AS; -- Code page 1250
PRINT 'Fran?is: e??a?o?'
COLLATE Cyrillic_General_CI_AS; -- Code page 1256
GO
Fran?is: e??a?o?
Fran?is: e??a?o?
Fran?is: e?e-aa-o?
Francais: eeee-aa-oo
How find out the default Collation?
```

[View All Answers](#)

Question - 264:

How To Specify the Collation for a Character Data Type in MS SQL Server?

Ans:

If you do not want to use the default collation provided by the SQL Server, you can use the "COLLATE collation_name" clause to specify a different collation to be used at different levels:

- * Database Level - Used in CREATE DATABASE or ALTER DATABASE statements to set a new default collation for the entire database.
- * Table Column Level - Used in CREATE TABLE or ALTER TABLE statements to override the database default collation for this column.
- * Character String Expression Level - Used at the end of the expression to cast the expression to the specified collation.

[View All Answers](#)



Question - 265:

What Is a Collation in MS SQL Server?

Ans:

A collation is a set of rules defining a character set and its sorting rules. SQL Server support a large number of built-in collations. For example:

- * Albanian_CI_AI_KS_WS - Albanian, case-insensitive (CI), accent-insensitive (AI), kanatype-sensitive (KS), width-sensitive (WS).
- * Arabic_CI_AS_KS_WS - Arabic, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive.
- * French_CI_AI - French, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive.
- * Korean_Wansung_BIN - Korean-Wansung, binary sort.
- * SQL_Latin1_General_CP1250_CI_AS - Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive.

If you want to get a list of all collations support on the SQL Server, run this statement:

```
SELECT * FROM fn_helpcollations()
GO
```

[View All Answers](#)

Question - 266:

How To Write Character String Constants or Literals in MS SQL Server?

Ans:

Character string literals are used everywhere in Transact-SQL statements. You need to learn some important rules on writing character string literals:

- * Character string literals must be enclosed in single quotes as 'Welcome to GlobalGuideLine.com!'.
- * An empty string must be expressed as ''.
- * Single quote characters inside a string must be expressed as two single quotes. For example: 'It's Sunday!'.
- * Character strings use non-default character code pages must use the COLLATE clause to provide the code page through a collate name. For example, 'Fran?is' COLLATE French_CI_AS.

[View All Answers](#)

Question - 267:

What Is a Constant or Literal in MS SQL Server?

Ans:

A constant, or literal, is a symbol that represents a specific value of a specific data type. Constants or literals are used commonly as the default values for table columns, variables, and parameters. The format of constant or literal depends on the data type of the value it represents. There are x formats used to specify literals in SQL Server:

- * Integer Number Literals - Strings of numbers prefixed with (+/-) as the sign if needed. They are used for integer data types. For example: -2255.
- * Decimal Number Literals - Strings of numbers with a single decimal point and an optional prefix sign (+/-). They are used for DECIMAL data type. For example: 1894.1204.
- * Floating Point Number Literals - Numeric values represented in the scientific notation. They are used for FLOAT data type. For example: 9.22337e+010.
- * Character String Literals - Strings of characters enclosed in single quotes. They are used for character string data types. . For example: 'Welcome to GlobalGuideLine.com!'.
- * Unicode Character String Literals - Strings of Unicode characters enclosed in single quotes with a prefix of (N). They are used for Unicode character string data types. For example: N'Welcome to GlobalGuideLine.com'.

[View All Answers](#)

Question - 268:

What Are the Differences between DECIMAL and FLOAT in MS SQL Server?

Ans:

DECIMAL and FLOAT are both used to store numerical values. But they have the following main differences:

- * DECIMAL(p,s) stores values with the decimal point fixed at the position of s (scale) digits from the right. The total number of decimal digits is also fixed as p (precision).
- * FLOAT(n) stores values with the decimal point floating based on the value. The number of bits used to store the mantissa part is fixed as n.
- * If the input value of DECIMAL(p,s) has more digits after the decimal point than the scale s, the value will be rounded to the scale s.
- * If the input value of FLOAT(n) has more total digits (mantissa) than what n bits can store, the value will be rounded to fit the storage size.
- * If the input value of DECIMAL(p,s) has more digits before the decimal point than p-s, SQL Server will give you an over-flow error.
- * If the input value of FLOAT(n) is too big that the exponential part goes over the positive limit, SQL Server will give you an over-flow error.
- * If the input value of FLOAT(n) is too small that the exponential part goes over the negative limit, SQL Server will give you an under-flow error.

[View All Answers](#)

Question - 269:

What Are the Differences between CHAR and VARCHAR in MS SQL Server?

Ans:

CHAR and VARCHAR are both used to store code page based character strings. But they have the following main differences:

- * CHAR(n) stores character strings with a fixed length, n bytes, storage format.
- * VARCHAR(n) stores character strings with a variable length, less than n bytes, storage format.
- * If the input string of CHAR(n) has less than n bytes, it will be padded with space characters to become n bytes.
- * If the input string of VARCHAR(n) has less than n bytes, it will not be padded with any characters.
- * If the input string of CHAR(n) has more than n bytes, it will be truncated to n bytes.
- * If the input string of VARCHAR(n) has more than n bytes, it will also be truncated to n bytes.

[View All Answers](#)

Question - 270:

What Are the Differences between CHAR and NCHAR in MS SQL Server?

Ans:



Both CHAR and NCHAR are fixed length data types. But they have the following main differences:

- * CHAR stores characters based on the code page with 1 byte per character most of the time.
- * NCHAR stores characters using Unicode character set with 2 bytes per character most of the time.
- * CHAR(n) specifies a length of n bytes by default.
- * NCHAR(n) specifies a length of n characters by default.

[View All Answers](#)

Question - 271:

What Are Binary String Data Types in MS SQL Server?

Ans:

Binary string data types are used to store binary strings. SQL Server 2005 supports the following binary string data types:

- * BINARY - Binary strings with a fixed length of n bytes defined as BINARY(n). The maximum length is 8,000 bytes.
- * VARBINARY - Binary strings with a variable length of n bytes defined as VARBINARY(n). The maximum length is 8,000 bytes. VARBINARY has a synonym of BINARY VARYING. VARBINARY also has special form as VARBINARY(MAX), which can store up to $2^{31}-1$ bytes.
- * IMAGE - Binary strings with a variable length up to $2^{31}-1$ (2,147,483,647) bytes.

[View All Answers](#)

Question - 272:

What Are Unicode Character String Data Types in MS SQL Server?

Ans:

Unicode character string data types are used to store Unicode character strings. SQL Server 2005 supports the following Unicode character string data types:

- * NCHAR - Unicode character strings with a fixed length of n characters defined as NCHAR(n). The maximum length is 4,000 characters. NCHAR has two synonyms of NATIONAL CHAR and NATIONAL CHARACTER.
- * VARCHAR - Unicode character strings with a variable length of n characters defined as VARCHAR(n). The maximum length is 4,000 characters. NVARCHAR has two synonyms of NATIONAL CHAR VARYING and NATIONAL CHARACTER VARYING. NVARCHAR also has special form as NVARCHAR(MAX), which can store up to $2^{31}-1$ bytes.
- * NTEXT - Unicode character strings with a variable length up to $2^{31}-1$ (2,147,483,647) bytes. NTEXT is equivalent to NVARCHAR(MAX). NTEXT has a synonym of NATIONAL TEXT.

[View All Answers](#)

Question - 273:

What Are Character String Data Types in MS SQL Server?

Ans:

Character string data types are used to store code page based character strings. SQL Server 2005 supports the following character string data types:

- * CHAR - Code page based character strings with a fixed length of n bytes defined as CHAR(n). The maximum length is 8,000 bytes. CHAR has a synonym of CHARACTER.
- * VARCHAR - Code page based character strings with a variable length of n bytes defined as VARCHAR(n). The maximum length is 8,000 bytes. VARCHAR has two synonyms of CHAR VARYING and CHARACTER VARYING. VARCHAR also has special form as VARCHAR(MAX), which can store up to $2^{31}-1$ bytes.
- * TEXT - Code page based character strings with a variable length up to $2^{31}-1$ (2,147,483,647) bytes. TEXT is equivalent to VARCHAR(MAX).

Here are some good examples of character string values:

```
PRINT 'Hello!'; -- CHAR(10)
PRINT 'Hello!'; -- VARCHAR(10)
GO
```

[View All Answers](#)

Question - 274:

What Are Date and Time Data Types in MS SQL Server?

Ans:

Date and time data types are used to store an instances of calendar dates and times. SQL Server 2005 supports the following date and time data types:

- * DATETIME - Date and time values stored in 8 bytes with 4 bytes for the date and 4 bytes for the time. DATETIME values are in the range of January 1, 1753 to December 31, 9999 with a precision of 3.33 milliseconds.
- * SMALLDATETIME - Date and time values stored in 4 bytes with 2 bytes for the date and 2 bytes for the time. SMALLDATETIME values are in the range of January 1, 1900 to June 6, 2079 with a precision of 1 minute.

Note that SQL Server does not support DATE and TIME data types separately. Here are some good examples of date and time values:

```
PRINT '2107-05-19 22:52:51.607'; -- DATETIME
PRINT '2007-05-19 22:52:00'; -- SMALLDATETIME
GO
```

[View All Answers](#)

Question - 275:

What Are Approximate Numeric Data Types in MS SQL Server?

Ans:

Approximate numeric data types are used to store numeric values with floating decimal points. SQL Server 2005 supports the following approximate numeric data types:

- * FLOAT - Floating point values with a fixed number of bits n for the mantissa part defined as FLOAT(n). FLOAT values are in the range of $-1.79E+308$ to $-2.23E-308$, 0 and $2.23E-308$ to $1.79E+308$.
- * REAL - Same as FLOAT(24), also called single precision floating point numbers.

The DOUBLE PRECISION data type is supported as a synonym of FLOAT(53). Here are some good examples of approximate numeric values:

```
PRINT 9.22337203685e+010; -- FLOAT(53)
PRINT 9.22337e+010; -- FLOAT(24)
```




GO

[View All Answers](#)

Question - 276:

What Are Exact Numeric Data Types in MS SQL Server?

Ans:

Exact numeric data types are used to store numeric values with exact precisions and scales. SQL Server 2005 supports the following exact numeric data types:

- * BIGINT - 8-byte integers in the range of -2^{63} (-9,223,372,036,854,775,808) to $2^{63}-1$ (9,223,372,036,854,775,807).
- * INT - 4-byte integers in the range of -2^{31} (-2,147,483,648) to $2^{31}-1$ (2,147,483,647).
- * SMALLINT - 2-byte integers in the range of -2^{15} (-32,768) to $2^{15}-1$ (32,767).
- * TINYINT - 1-byte integers in the range of 0 to 255.
- * BIT - 1-bit integers with 2 values: 0 and 1.
- * DECIMAL - Decimal numbers with a fixed precision p and a fixed scale s defined as DECIMAL(p,s). DECIMAL values are in the range of $-10^{38}+1$ to $10^{38}-1$. DECIMAL has a synonym of DEC.
- * NUMERIC - Same as DECIMAL.
- * MONEY - Currency values stored in 8 bytes in the range of -922,337,203,685,477.5808 to 922,337,203,685,477.5807
- * SMALLMONEY - Currency values stored in 4 bytes in the range of - 214,748.3648 to 214,748.3647

Here are some good examples of exact numeric values:

```
PRINT 372036854775808; -- BIGINT
PRINT 147483648; -- INT
PRINT 2768; -- SMALLINT
PRINT 250; -- TINYINT
PRINT 1; -- BIT
PRINT 12748.3648; -- DECIMAL(9,2)
PRINT 337203685477.58; -- MONEY
PRINT 748.36; -- SMALLMONEY
GO
```

[View All Answers](#)

Question - 277:

How Many Categories of Data Types Used by SQL Server?

Ans:

A data type is an attribute that specifies the type of data that the object can hold: integer data, character data, monetary data, date and time data, binary strings, and so on. In SQL Server 2005, each column, local variable, expression, and parameter has a related data type. Data types in SQL Server 2005 are organized into the following categories:

- * Exact Numerics - BIGINT, INT, SMALLINT, TINYINT, BIT, DECIMAL, NUMERIC, MONEY, and SMALLMONEY.
- * Approximate Numerics - FLOAT, and REAL.
- * Date and Time - DATETIME and SMALLDATETIME.
- * Character Strings - CHAR, VARCHAR, and TEXT.
- * Unicode Character Strings - NCHAR, NVARCHAR, and NTEXT.
- * Binary Strings - BINARY, VARBINARY, and IMAGE.
- * Other Data Types - CURSOR, SQL_VARIANT, TABLE, UNIQUEIDENTIFIER, TIMESTAMP, and XML.

[View All Answers](#)

Question - 278:

How To Create User Messages with PRINT Statements in MS SQL Server?

Ans:

Normally, messages are generated by SQL Server and returned to the client applications associated with the execution result of statement batches. But you can define your messages and ask the SQL Server to your client session with the PRINT statement, as shown in this tutorial exercise:

```
C:>sqlcmd -S localhostsqlexpress -U sa -P GlobalGuideLine
1> PRINT 'What time is it?';
2> SELECT GETDATE();
3> GO
```

What time is it?

```
-----
2007-05-19 21:51:23.797
(1 rows affected)
```

```
1> PRINT 'What time is it?';
2> PRINT 'It's ' + CAST(GETDATE() AS NVARCHAR(30));
3> GO
```

What time is it?

It's May 19 2007 9:55PM

Note that the SELECT statement returns query result, which is very different than messages returned by the PRINT statement.

[View All Answers](#)

Question - 279:

How To Use GO Command in "sqlcmd"?

Ans:

"sqlcmd" is a command line client application to run Transact-SQL statements on a target SQL Server. When "sqlcmd" is started and connected to a SQL Server, it will start a new batch and prompt you to enter the first statement of the batch. You can enter one or more statements in one or more lines to form a Transact-SQL statement batch. To end a batch of statements and send it to the SQL Server for execution, you need to enter the GO command. The following "sqlcmd" tutorial session sends two batches to the SQL Server:

```
C:>sqlcmd -S localhostsqlexpress -U sa -P GlobalGuideline
```



```
1> SELECT getdate();
2> SELECT getdates();
3> SELECT getdate();
4> GO
Msg 195, Level 15, State 10, Line 2
'getdates' is not a recognized built-in function name.
1> SELECT getdate();
2> SELECT getdate();
3> SELECT getdate();
4> GO
```

```
-----
2007-05-19 22:55:07.233
(1 rows affected)
-----
```

```
2007-05-19 22:55:07.233
(1 rows affected)
-----
```

```
2007-05-19 22:55:07.233
(1 rows affected)
1>QUIT
C:
```

[View All Answers](#)

Question - 280:

What Happens to a Statement Batch If There Is a Compilation Error?

Ans:

If a statement batch has multiple statements, and one of them has compilation error, all statements in the batch will not be executed. The tutorial exercise below gives you some good examples:

```
SELECT getdate();
SELECT getdates();
SELECT getdate();
GO
Msg 195, Level 15, State 10, Line 2
'getdates' is not a recognized built-in function name.
```

As you can see, the compilation error on the second statement stops the execution of all statements.

[View All Answers](#)

Question - 281:

What Is a Transact-SQL Statement Batch in MS SQL Server?

Ans:

A batch is a group of one or more Transact-SQL statements sent at the same time from an application to SQL Server 2005 for execution. SQL Server compiles the statements of a batch into a single executable unit, called an execution plan. The statements in the execution plan are then executed one at a time. How to define and send a batch is depending on the client application you are using. For example:

- # You need to use the GO command in "sqlcmd" tool to end a batch and send it to SQL server.
- # SQL Server Management Studio also uses GO command to separate batches when multiple statements are selected for execution.
- # In ADO, a batch is defined as all the statements entered in the CommandText property of a Command object.

[View All Answers](#)

Question - 282:

How To Enter Comments in Transact-SQL Statements?

Ans:

There are 2 ways to enter comments within Transact-SQL statements:

- * Starting comments with two dashes "--": Everything between "--" and the end of the line is treated as a comment.
- * Entering comments between "/*" and "*/": Everything between "/*" and "*/" is treated as a comment.

Here are some good examples of how to enter comments:

```
/* The following statement
creates a table
called ggl_links */
CREATE TABLE ggl_links (
  id INTEGER PRIMARY KEY, -- the primary key
  url VARCHAR(80) NOT NULL, -- address of the link
  notes VARCHAR(1024),
  counts INT, -- number of clicks
  created DATETIME NOT NULL DEFAULT(getdate())
);
GO
-- Get rid of this table
DROP TABLE ggl_links;
GO
```

[View All Answers](#)

Question - 283:

How To Start and End Transact-SQL Statements?

Ans:



There are simple rule about writing Transact-SQL statements:

- * A Transact-SQL statement should be started with a pre-defined statement keyword.
- * A Transact-SQL statement should be ended with a new line (/n) or a semicolon (;).
- * A Transact-SQL statement can be entered in a single line or multiple lines.
- * Transact-SQL statement keywords are case-insensitive.
- * Extra white space characters are ignored within Transact-SQL statements.

Here are some good examples of Transact-SQL statements:

```
CREATE TABLE ggl_links (id INTEGER PRIMARY KEY,
url VARCHAR(80) NOT NULL,
notes VARCHAR(1024),
counts INT,
created DATETIME NOT NULL DEFAULT(getdate()));
```

```
SeLeCT 'Welcome to GlobalGuideLine.com SQL Server!';
Print 'Come visit this page again!'
```

[View All Answers](#)

Question - 284:

What Is a Transact-SQL Statement?

Ans:

A Transact-SQL statement is a basic execution unit of the Transact-SQL language. Here are some examples of Transact-SQL statements:

- * "CREATE TABLE ...;" - A data definition language statement that defines a table.
- * "INSERT INTO ...;" - A data manipulation language statement that inserts data into a table.
- * "SELECT * FROM ...;" - A query statement that returns data from a table.
- * "DROP TABLE ...;" - A data definition language statement that deletes a table.
- * "DECLARE @name data_type;" - A declaration statement that defines a local variable.
- * "SET @name = expression;" - An assignment statement that assigns a value to a local variable.
- * "BEGIN TRANSACTION;" - A transaction management statement that marks the beginning of a transaction.
- * "ROLLBACK TRANSACTION;" - A transaction management statement that rolls back all changes of a transaction.

[View All Answers](#)

Question - 285:

What Is Transact-SQL Language?

Ans:

According to Wikipedia, Transact-SQL, sometimes abbreviated T-SQL, is Microsoft's and Sybase's proprietary extension to the SQL language. Microsoft's implementation ships in the Microsoft SQL Server product. Sybase uses the language in its Adaptive Server Enterprise, the successor to Sybase SQL Server. In order to make it more powerful, SQL has been enhanced with additional features such as:

- * Control-of-flow language - Like BEGIN and END, BREAK, CONTINUE, GOTO, IF and ELSE, RETURN, WAITFOR, and WHILE statements.
- * Local variables - Like DECLARE and SET statements
- * Various support functions for string processing, date processing, mathematics, etc. - Like SIN(), ASIN(), LOG(), etc.
- * Improvements to DELETE and UPDATE statements - Both the DELETE and UPDATE statements allow a FROM clause to be added which allows joins to be included.

[View All Answers](#)

Question - 286:

What Is SQL Language?

Ans:

SQL, SEQUEL (Structured English Query Language), is a language for RDBMS (Relational Database Management Systems). During the 1970s, a group at IBM's San Jose research center developed a database system "System R" based upon Codd's model. Structured English Query Language ("SEQUEL") was designed to manipulate and retrieve data stored in System R. The acronym SEQUEL was later condensed to SQL. SQL was adopted as a standard by ANSI (American National Standards Institute) in 1986 and ISO (International Organization for Standardization) in 1987. Since then, SQL standard has gone through a number of revisions:

- * 1986, SQL-86 or SQL-87 - First published by ANSI. Ratified by ISO in 1987.
- * 1989, SQL-89 - Minor revision.
- * 1992, SQL-92 or SQL2 - Major revision (ISO 9075).
- * 1999, SQL:1999 or SQL3 - Added regular expression matching, recursive queries, triggers, non-scalar types and some object-oriented features.
- * 2003, SQL:2003 - Introduced XML-related features, window functions, standardized sequences and columns with auto-generated values (including identity-columns).
- * 2006, SQL:2006 - ISO/IEC 9075-14:2006 defines ways in which SQL can be used in conjunction with XML.

[View All Answers](#)

Question - 287:

Can Group Functions Be Used in the ORDER BY Clause in MS SQL Server?

Ans:

If the query output is aggregated as groups, you can sort the groups by using group functions in the ORDER BY clause. The following statement returns the maximum "counts" in each group, determined by a unique combination of tag and year. The group output is sorted by the maximum "counts" in each group in ascending order:

```
SELECT tag, YEAR(created), MAX(counts)
FROM ggl_links GROUP BY tag, YEAR(created)
ORDER BY MAX(counts)
GO<pre>
tag year(created) max(counts)
DEV 2006 120
DBA 2006 390
```



```
DEV 2004 439
SQA 2007 728
SQA 2003 828
DBA 2005 960
DBA 2007 972
```

[View All Answers](#)

Question - 288:

Can Multiple Columns Be Used in SQL GROUP BY Clause in MS SQL Server?

Ans:

If you want to break your output into smaller groups, you can specify multiple column names or expressions in the GROUP BY clause. Output in each group must satisfy a specific combination of the expressions listed in the GROUP BY clause. The more columns or expressions entered in the GROUP BY clause, the smaller the groups will be. The tutorial exercise below shows you how to break data into groups per "tag" and per year when they were created. Then the group function COUNT(*) is applied on each group:

```
SELECT tag, YEAR(created), COUNT(*)
FROM ggl_links GROUP BY tag, YEAR(created)
```

```
GO
tag year(created) count(*)
SQA 2003 1
DEV 2004 1
DBA 2005 1
DBA 2006 1
DEV 2006 1
DBA 2007 1
SQA 2007 1
```

So there is only one row in each group.

[View All Answers](#)

Question - 289:

How To Count Duplicated Values in a Column in MS SQL Server?

Ans:

If you have a column with duplicated values, and you want to know what are those duplicated values are and how many duplicates are there for each of those values, you can use the "GROUP BY ... HAVING" clause as shown in the following example. It returns how many duplicated first names in the ggl_team table:

```
SELECT first_name, COUNT(*) FROM ggl_team
GROUP BY first_name HAVING COUNT(*) > 1
```

```
GO
first_name count(*)
John 5
```

So we have "John" as duplicated first name 5 times in the ggl_team table.

[View All Answers](#)

Question - 290:

How To Apply Filtering Criteria at Group Level with The HAVING Clause in MS SQL Server?

Ans:

Let's say you have divided the query output into multiple groups with the GROUP BY clause. Now you are only interested in some of the groups, not all the groups. If you want to filter out some groups from the query, you can apply filtering criteria at the group level by using the HAVING clause inside the GROUP BY clause with this syntax:

```
SELECT group_level_fields FROM source_tables
WHERE search_condition
GROUP BY group_by_expression
HAVING group_filtering_condition
```

Since group_filtering_condition applies to groups, only group level expressions can be used in group_filtering_condition. The following tutorial exercise gives you some good examples of HAVING clause:

```
SELECT tag, COUNT(*), MIN(created), AVG(counts)
FROM ggl_links
GROUP BY tag HAVING AVG(counts) > 300
```

```
GO
tag COUNT(*) MIN(created) AVG(counts)
DBA 3 2005-01-01 774
SQA 2 2003-01-01 778
```

```
SELECT tag, COUNT(*), MIN(created), AVG(counts)
FROM ggl_links
GROUP BY tag
HAVING AVG(counts) > 300 AND tag = 'DBA'
```

```
GO
tag COUNT(*) MIN(created) AVG(counts)
DBA 3 2005-01-01 774
```

Note that the more criteria you have in the HAVING clause, the less groups you will get.

[View All Answers](#)

Question - 291:

How To Divide Query Output into Multiple Groups with the GROUP BY Clause in MS SQL Server?

Ans:



Sometimes, you want to divide the query output into multiple groups, and apply group functions on each individual groups. Dividing query output into multiple groups can be done with the GROUP BY clause. Here is the syntax of a SELECT statement with a GROUP BY clause.

```
SELECT group_level_fields FROM source_tables
```

```
WHERE search_condition
```

```
GROUP BY group_by_expression
```

* group_by_expression - An list of columns to be used as the group criteria - Rows that have the same combination of values of there columns form a single group.

* group_level_fields - An list of selection expressions that can be evaluated at the group level.

The final output of the SELECT statement is the resulting values of group_level_fields for each group. The following script gives you a good GROUP BY example with a single column as the group_by_expression. In this case, rows with the same value of this column will be considered as a single group.

```
SELECT tag, COUNT(*), MAX(counts), MIN(created)
```

```
FROM ggl_links GROUP BY tag
```

```
GO<pre>
```

```
tag    COUNT(*) MAX(counts) MIN(created)
```

```
DBA    3      972      2005-01-01
```

```
DEV    2      439      2004-01-01
```

```
SQA    2      828      2003-01-01</pre>
```

Notice that, column "tag" can also be used in group_level_fields, because it is used as the group_by_expression, and becomes a constant for any given group.

[View All Answers](#)

Question - 292:

Can Group Functions Be Mixed with Non-group Selection Fields in MS SQL Server?

Ans:

If a group function is used in the SELECT clause, all other selection fields must be group level fields. Non-group fields can not be mixed with group fields in the SELECT clause. The script below gives you an example of invalid SELECT statements with group and non-group selection fields mixed in a SELECT statement:

```
SELECT COUNT(*), url FROM ggl_links
```

```
GO
```

```
Msg 8120, Level 16, State 1, Line 1
```

```
Column 'ggl_links.url' is invalid in the select list because
```

```
it is not contained in either an aggregate function or the
```

```
GROUP BY clause.
```

```
SELECT 2*COUNT(*), 2*counts FROM ggl_links
```

```
GO
```

```
Msg 8120, Level 16, State 1, Line 1
```

```
Column 'ggl_links.counts' is invalid in the select list
```

```
because it is not contained in either an aggregate function
```

```
or the GROUP BY clause.
```

In these examples, COUNT(*) is a group field and "url"/"2*counts" is a non-group field. The error message also tells that "url"/"counts" is not an aggregate function (group function).

[View All Answers](#)

Question - 293:

How To Use Group Functions in the SELECT Clause in MS SQL Server?

Ans:

If group functions are used in the SELECT clause, all rows that meet the criteria defined in the WHERE clause will be treated as a single group. The group functions will be apply all rows in that group as a whole. The final output of the SELECT statement is the resulting values of the group functions, not the rows in the group.

Here are two good examples of using group functions :

```
SELECT COUNT(*), MAX(counts), MIN(created)
```

```
FROM ggl_links
```

```
GO
```

```
COUNT(*) MAX(counts) MIN(created)
```

```
7      972      2003-01-01
```

```
SELECT COUNT(*), MAX(counts), MIN(created)
```

```
FROM ggl_links WHERE tag = 'DBA'
```

```
GO
```

```
COUNT(*) MAX(counts) MIN(created)
```

```
3      972      2005-01-01
```

In first case, the group contains all the rows in table ggl_links, because is no WHERE clause. In the second case, the group contains only 3 rows because of the WHERE clause tag = 'DBA'.

[View All Answers](#)

Question - 294:

What Are Group Functions in Query Statements in MS SQL Server?

Ans:

Group functions are functions applied to a group of rows. Examples of group functions are:

* COUNT(*) - Returns the number of rows in the group.

* MIN(exp) - Returns the minimum value of the expression evaluated on each row of the group.

* MAX(exp) - Returns the maximum value of the expression evaluated on each row of the group.

* AVG(exp) - Returns the average value of the expression evaluated on each row of the group.

* SUM(exp) - Returns the sum value of the expression evaluated on each row of the group.

* COUNTDISTINCT(exp) - Returns the number of distinct values of the expression evaluated on each row of the group.

[View All Answers](#)

Question - 295:

How To Filter Out Duplications in the Returning Rows in MS SQL Server?



Ans:

If there are duplications in the returning rows, and you want to remove the duplications, you can use the keyword DISTINCT in the SELECT clause. The DISTINCT applies to the combination of all data fields specified in the SELECT clause. The tutorial exercise below shows you how DISTINCT works:

```
CREATE TABLE ggl_team (first_name VARCHAR(8),
last_name VARCHAR(8))
GO
INSERT INTO ggl_team VALUES ('John', 'Gate')
GO
INSERT INTO ggl_team VALUES ('John', 'Russell')
GO
INSERT INTO ggl_team VALUES ('John', 'Seo')
GO
INSERT INTO ggl_team VALUES ('John', 'Gate')
GO
INSERT INTO ggl_team VALUES ('James', 'Gate')
GO
INSERT INTO ggl_team VALUES ('Peter', 'Gate')
GO
INSERT INTO ggl_team VALUES ('John', 'Gate')
GO
```

[View All Answers](#)

Question - 296:

Can SELECT Statements Be Used on Views in MS SQL Server?

Ans:

Select (query) statements can be used on views in the same way as tables. The following tutorial exercise helps you creating a view and running a query statement on the view:

```
<pre>
CREATE VIEW myLinks AS SELECT * FROM ggl_links
WHERE url LIKE '%glo%'
GO
SELECT tag, counts, url, created
FROM myLinks ORDER BY counts DESC
GO
tag counts url created
DBA 972 globalguideline.com 2007-05-19
SQA 728 globalguideline.com/html 2007-05-19
DEV 120 globalguideline.com/sql 2006-04-30
</pre>
```

[View All Answers](#)

Question - 297:

How To Count Rows with the COUNT(*) Function in MS SQL Server?

Ans:

If you want to count the number of rows, you can use the COUNT(*) function in the SELECT clause. The following tutorial exercise shows you some good examples:

```
SELECT COUNT(*) FROM ggl_links
GO
7
SELECT COUNT(*) FROM ggl_links
WHERE url LIKE '%glo%'
GO
3
```

So there are 7 rows in total in table "ggl_links", and 3 rows that have 'glo' as part of their url names.

[View All Answers](#)

Question - 298:

How To Sort Query Output in Descending Order in MS SQL Server?

Ans:

If you want to sort a column in descending order, you can specify the DESC keyword in the ORDER BY clause. The following SELECT statement first sorts the "tag" in descending order, then sorts the "counts" in ascending order:

```
SELECT tag, counts, url, created
FROM ggl_links ORDER BY tag DESC, counts
GO
<pre>
tag counts url created
SQA 728 globalguideline.com/sql 2007-05-19
SQA 828 www.winrunner.com 2003-01-01
DEV 120 globalguideline.com/xml 2006-04-30
DEV 439 www.php.net 2004-01-01
DBA 390 www.mysql.com 2006-01-01
DBA 960 www.oracle.com 2005-01-01
DBA 972 globalguideline.com 2007-05-19
</pre>
```

[View All Answers](#)



Question - 299:

Can the Query Output Be Sorted by Multiple Columns in MS SQL Server?

Ans:

You can specifying multiple columns in the ORDER BY clause as shown in the following example statement, which returns rows sorted by "tag" and "counts" values:

```
SELECT tag, counts, url, created
FROM ggl_links ORDER BY tag, counts
```

GO

<pre>

| tab | counts | url | created |
|-----|--------|-------------------------|------------|
| DBA | 390 | www.mysql.com | 2006-01-01 |
| DBA | 960 | www.oracle.com | 2005-01-01 |
| DBA | 972 | globalguideline.com | 2007-05-19 |
| DEV | 120 | globalguideline.com/xml | 2006-04-30 |
| DEV | 439 | www.php.net | 2004-01-01 |
| SQA | 728 | globalguideline.com/sql | 2007-05-19 |
| SQA | 828 | www.winrunner.com | 2003-01-01 |

</pre>

[View All Answers](#)

Question - 300:

How To Sort the Query Output with ORDER BY Clauses in MS SQL Server?

Ans:

If you want the returning rows to be sorted, you can specify a sorting expression in the ORDER BY clause. The simplest sort expression is column name who's values will be sorted by. The following select statement returns rows sorted by the values in the "counts" column:

```
SELECT * FROM ggl_links ORDER BY counts
```

GO

<pre>

| id | url | notes | counts | created | tag |
|-----|--------------------------|-------|--------|------------|----------------|
| 101 | globalguideline.com | | NULL | 120 | 2006-04-30 DEV |
| 104 | www.mysql.com | | 390 | 2006-01-01 | DBA |
| 106 | www.php.net | | 439 | 2004-01-01 | DEV |
| 103 | globalguideline.com/html | | NULL | 728 | 2007-05-19 SQA |
| 107 | www.winrunner.com | | 828 | 2003-01-01 | SQA |
| 105 | www.oracle.com | | 960 | 2005-01-01 | DBA |
| 102 | globalguideline.com/sql | | NULL | 972 | 2007-05-19 DBA |

</pre>

The output is sorted by the values of the "counts" column.

[View All Answers](#)

Question - 301:

How To Add More Data to the Testing Table in MS SQL Server?

Ans:

If you want to continue with other tutorial exercises in this FAQ collection, you need to add more data to the testing table. Follow the script below to add a new column and more rows:

```
ALTER TABLE ggl_links ADD tag VARCHAR(8)
```

GO

```
UPDATE ggl_links SET tag = 'DEV' WHERE id = 101
```

GO

```
UPDATE ggl_links SET tag = 'DBA' WHERE id = 102
```

GO

```
UPDATE ggl_links SET tag = 'SQA' WHERE id = 103
```

GO

```
INSERT INTO ggl_links VALUES (104,
'www.mysql.com', '', '0', '2006-01-01', 'DBA')
```

GO

```
INSERT INTO ggl_links VALUES (105,
'www.oracle.com', '', '0', '2005-01-01', 'DBA')
```

GO

```
INSERT INTO ggl_links VALUES (106,
'www.php.net', '', '0', '2004-01-01', 'DEV')
```

GO

```
INSERT INTO ggl_links VALUES (107,
'www.winrunner.com', '', '0', '2003-01-01', 'SQA')
```

GO

```
UPDATE ggl_links
SET counts = CAST(LOG(id)*1000000 AS INT) % 1000
```

GO

[View All Answers](#)

Question - 302:

How To Select Some Specific Rows from a Table in MS SQL Server?

Ans:

If you don't want select all rows from a table, you can specify a WHERE clause to tell the query to return only the rows that meets the condition defined in the



WHERE clause. The WHERE clause condition is a normal Boolean expression. If any data from the table needs to be used in the Boolean expression, column names should be used to represent the table data.

The first select statement below only returns rows that have url names containing the letter "a". The second select statement returns no rows, because the WHERE clause results FALSE for all rows in the table.

```
SELECT * FROM ggl_links WHERE url LIKE '%s%'
GO
id url          notes counts created
102 globalguideline.com/sql NULL 0 2007-05-19
103 globalguideline.com/xslt NULL NULL 2007-05-19
SELECT * FROM ggl_links WHERE id < 100
GO
0 row
```

[View All Answers](#)

Question - 303:

How To Select Some Specific Columns from a Table in a Query in MS SQL Server?

Ans:

If you want explicitly tell the query to return some specific columns, you can specify the column names in the SELECT clause. The following select statement returns only three columns, "id", "created" and "url" from the table "ggl_links":

```
<pre>
SELECT id, created, url FROM ggl_links
id created url
101 2006-04-30 www.globalguideline.com
102 2007-05-19 www.globalguideline.com/html
103 2007-05-19 www.globalguideline.com/sql
</pre>
```

[View All Answers](#)

Question - 304:

How To Select All Columns of All Rows from a Table with a SELECT statement in MS SQL Server?

Ans:

The simplest query statement is the one that selects all columns of all rows from a single table: "SELECT * FROM tableName". The (*) in the SELECT clause tells the query to return all columns. The missing WHERE clause tells the query to return all rows in specified table. The tutorial exercise below returns all columns and all rows from table "ggl_links":

```
<pre>
SELECT * FROM ggl_links
id url          notes counts created
101 globalguideline.com NULL 0 2006-04-30
102 globalguideline.com/html NULL 0 2007-05-19
103 globalguideline.com/sql NULL NULL 2007-05-19
</pre>
```

[View All Answers](#)

Question - 305:

What Is a SELECT Query Statement in MS SQL Server?

Ans:

The SELECT statement is also called the query statement. It is the most frequently used SQL statement in any database application. SELECT statements allows you to retrieve data from one or more tables or views, with different selection criteria, grouping criteria and sorting orders.

A SELECT statement has the following basic syntax:

```
<pre>
SELECT select_list
FROM table_source
WHERE search_condition
GROUP BY group_by_expression
HAVING search_condition
ORDER BY order_by_expression
</pre>
```

Here is an example of a SELECT statement with all clauses mentioned above:

```
<pre>
SELECT SalesOrderID, SUM(LineTotal) AS TotalPrice
FROM SalesLT.SalesOrderDetail
WHERE ModifiedDate > '2004-05-01'
GROUP BY SalesOrderID
HAVING COUNT(*) > 30
ORDER BY TotalPrice DESC
</pre>
```

[View All Answers](#)

Question - 306:

How To Use ORDER BY with UNION Operators in MS SQL Server?

Ans:

If you need to sort the output from two queries grouped together with a UNION operator, you need to apply the ORDER BY clause at the group level, not at the subquery level.



Note that SQL Server and MySQL react differently to the ORDER BY clause used in a subquery of a UNION operator:

- * SQL Server will give error if ORDER BY is used inside a subquery of a UNION operator.
- * MySQL will ignore the ORDER BY clause inside a subquery of a UNION operator.

The following tutorial exercise shows you how to use ORDER BY clause with UNION operator:

```
(SELECT * FROM ggl_links WHERE tag = 'DBA'
ORDER BY created)
UNION
(SELECT * FROM ggl_links WHERE tag = 'DEV'
ORDER BY created)
GO
Msg 156, Level 15, State 1, Line 2
Incorrect syntax near the keyword 'ORDER'.
Msg 156, Level 15, State 1, Line 5
Incorrect syntax near the keyword 'ORDER'.
(SELECT * FROM ggl_links WHERE tag = 'DBA')
UNION
(SELECT * FROM ggl_links WHERE tag = 'DEV')
ORDER BY created
GO
```

[View All Answers](#)

Question - 307:

How To Use UNION to Merge Outputs from Two Queries Together in MS SQL Server?

Ans:

If you have two queries that returns the same row fields, you can merge their outputs together with the UNION operator. The following tutorial exercise shows you how to use the UNION operator:

```
SELECT * FROM ggl_links WHERE tag = 'DBA'
GO


```

id url notes counts created tag
102 globalguideline.com NULL 972 2007-05-19 DBA
104 www.mysql.com 390 2006-01-01 DBA
105 www.oracle.com 960 2005-01-01 DBA

```



```

SELECT * FROM ggl_links WHERE tag = 'DEV'
GO


```

id url          notes counts created tag
101 globalguideline.com/html NULL 120 2006-04-30 DEV
106 www.php.net 439 2004-01-01 DEV

```



```

SELECT * FROM ggl_links WHERE tag = 'DBA'
UNION
SELECT * FROM ggl_links WHERE tag = 'DEV'
GO


```

id url notes counts created tag
102 globalguideline.com NULL 972 2007-05-19 DBA
104 www.mysql.com 390 2006-01-01 DBA
105 www.oracle.com 960 2005-01-01 DBA
101 globalguideline.com/html NULL 120 2006-04-30 DEV
106 www.php.net 439 2004-01-01 DEV

```


```


```


```

[View All Answers](#)

Question - 308:

How To Return the Second 5 Rows in MS SQL Server?

Ans:

If you want to display query output in multiple pages with 5 rows per page, and the visitor wants to see the output for the second page, you need to display query output from row 6 to row 10. If you are using MySQL server, you can use the "LIMIT startRow maxRows".

But the LIMIT clause is not supported by the SQL server. And there seems to be no easy workaround. You may consider to return the top 10 rows, skip the first 5 rows, then keep the second 5 rows.

[View All Answers](#)

Question - 309:

How To Return the Top 5 Rows from a SELECT Query in MS SQL Server?

Ans:

If you want the query to return only the first 5 rows, you can use the "TOP 5" clause. The TOP clause takes one parameter to indicate how many top rows to return. The following statements returns the first 5 rows and 3 rows from the ggl_links:

```
SELECT TOP 5 id, url, counts, tag FROM ggl_links
ORDER BY counts DESC
GO


```

id url counts tag
102 www.globalguideline.com 972 DBA
105 www.google.com 960 DBA
```


```



```
107 www.yahoo.com      828 SQA
103 www.mysql.com      728 SQA
106 www.php.net        439 DEV
</pre>
SELECT TOP 3 id, url, counts, tag FROM ggl_links
ORDER BY counts DESC
GO
<pre>
id url counts tag
102 www.globalguideline.com 972 DBA
105 www.google.com 960 DBA
107 www.yahoo.com 828 SQA
</pre>
```

[View All Answers](#)

Question - 310:

How To Count Groups Returned with the GROUP BY Clause in MS SQL Server?

Ans:

If you use the COUNT(*) function on groups returned with the GROUP BY clause, it will count the number of rows within each group, not the number of groups. If you want to count the number of groups, you can put the GROUP BY query into a subquery and apply the COUNT(*) function on the main query as shown in the following tutorial exercise:

```
SELECT tag AS Category, YEAR(created) AS Year,
COUNT(*) AS Counts FROM ggl_links GROUP BY tag,
YEAR(created)
GO
<pre>
Category Year Counts
HTML 2003 1
XML 2004 1
CSS 2005 1
SQL 2006 1
SEO 2006 1
JavaScript 2007 1
JOBS 2007 1
</pre>
SELECT COUNT(*) FROM (
SELECT tag AS Category, YEAR(created) AS Year,
COUNT(*) AS Counts FROM ggl_links GROUP BY tag,
YEAR(created) ) groups
GO
7
```

[View All Answers](#)

Question - 311:

How To Use Subqueries in the FROM Clause in MS SQL Server?

Ans:

If you have a query returning many rows of data, and you want to perform another query on those rows, you can put the first query as a subquery in the FROM clause of the second query. A subquery used in this way become a temporary table, and you must provide a table alias name for the subquery as in "SELECT ... FROM (SELECT ...) aliasName". The following statement shows you how to use a subquery as base table for the main query:

```
SELECT * FROM (SELECT l.id, l.url, r.comment
FROM ggl_links l LEFT OUTER JOIN ggl_rates r
ON l.id = r.id) WHERE url LIKE '%er%'
GO
Msg 156, Level 15, State 1, Line 3
Incorrect syntax near the keyword 'WHERE'.
SELECT * FROM (SELECT l.id, l.url, r.comment
FROM ggl_links l LEFT OUTER JOIN ggl_rates r
ON l.id = r.id) s WHERE s.url LIKE '%er%'
GO
<pre>
101 www.globalguideline.com The best
102 www.globalguideline.com/html Well done
103 www.globalguideline.com/xml Thumbs up
107 www.globalguideline.com/sql NULL
</pre>
```

The error on the first query is caused by the missing alias name to name output of the subquery as a temporary table.

[View All Answers](#)

Question - 312:

How To Use Subqueries with the EXISTS Operators in MS SQL Server?

Ans:

A subquery can be used with the EXISTS operator as "EXISTS (subquery)", which returns true if the subquery returns one or more rows. The following statement is a good example of "EXISTS (subquery)". It returns rows from ggl_links table that there are rows existing in the ggl_rates table with the same id.

```
SELECT id, url, tag, YEAR(created) As year
FROM ggl_links WHERE EXISTS (
SELECT * FROM ggl_rates
```



```
WHERE ggl_rates.id = ggl_links.id)
```

```
GO
```

```
<pre>
```

```
id url tag Year
```

```
101 www.globalguideline.com main 2006
```

```
102 www.globalguideline.com/html DBA 2007
```

```
103 www.globalguideline.com/sql SQL 2007
```

```
</pre>
```

Note that the subquery uses columns from the source table of the outer query.

[View All Answers](#)

Question - 313:

How To Use Subqueries with the IN Operators in MS SQL Server?

Ans:

A subquery can be used with the IN operator as "expression IN (subquery)". The subquery should return a single column with one or more rows to form a list of values to be used by the IN operation. The following tutorial exercise shows you how to use a subquery with the IN operator. It returns all links with ids in the ggl_rates table.

```
<pre>
```

```
SELECT id, url, tag, YEAR(created) AS year
```

```
FROM ggl_links WHERE id IN (SELECT id FROM ggl_rates)
```

```
GO
```

```
id url tag Year
```

```
101 www.globalguideline.com main 2006
```

```
102 www.globalguideline.com/html HTMLA 2007
```

```
103 www.globalguideline.com/sql SQL 2007
```

```
SELECT id, url, tag, YEAR(created) AS year
```

```
FROM ggl_links
```

```
WHERE id IN (101, 102, 103, 204, 205, 206, 207)
```

```
GO
```

```
id url tag Year
```

```
101 www.globalguideline.com main 2006
```

```
102 www.globalguideline.com/html HTMLA 2007
```

```
103 www.globalguideline.com/sql SQL 2007
```

```
</pre>
```

As you can see, the subquery is equivalent to a list of values.

[View All Answers](#)

Question - 314:

What Is a Subquery in a SELECT Query Statement in MS SQL Server?

Ans:

A subquery is a SELECT statement used as part of the selection criteria of the main SELECT statement. The subquery specified in the WHERE clause will be evaluated repeated on each row of the selection base table. The output of the subquery will be used in the final evaluation of the criteria. Usually, subqueries are used in the following Boolean operations:

- * "expression IN (subquery)" - True if the expression matches one of the returned values from the subquery.
- * "expression NOT IN (subquery)" - True if the expression does not match any of the returned values from the subquery.
- * "EXISTS (subquery)" - True if the subquery returns one or more rows.
- * "NOT EXISTS (subquery)" - True if the subquery returns no rows.

[View All Answers](#)

Question - 315:

How To Name Query Output Columns in MS SQL Server?

Ans:

Each column in the query output has a default name. If you don't like the default name, you can specify a new name for any column in the query output by using the AS clause. The following statement shows you a good example:

```
SELECT tag AS Category, YEAR(created) AS Year,
```

```
COUNT(*) AS Counts FROM ggl_links
```

```
GROUP BY tag, YEAR(created) ORDER BY COUNT(*) DESC
```

```
GO
```

```
<pre>
```

```
Category Year Counts
```

```
HTML 2003 1
```

```
SQL 2004 1
```

```
SEO 2005 1
```

```
Jobs 2006 1
```

```
XML 2006 1
```

```
XSL 2007 1
```

```
JavaScript 2007 1
```

```
</pre>
```

[View All Answers](#)

Question - 316:

How To Write an Inner Join with the WHERE Clause in MS SQL Server?

Ans:



If you don't want to use the INNER JOIN ... ON clause to write an inner join, you can put the join condition in the WHERE clause as shown in the following query example:

```
SELECT l.id, l.url, r.comment
FROM ggl_links l, ggl_rates r WHERE l.id = r.id
GO
<pre>
id    url                comment
101   www.globalguideline.com    The best
102   www.globalguideline.com/html Well done
103   www.globalguideline.com/sql  Thumbs up</pre>
```

[View All Answers](#)

Question - 317:

How To Write a Query with a Full Outer Join in MS SQL Server?

Ans:

If you want to query from two tables with a full outer join, you can use the FULL OUTER JOIN ... ON clause in the FROM clause. The following query returns output with a full outer join from two tables: ggl_links and ggl_rates. The join condition is that the id in the ggl_links table equals to the id in the ggl_rates table:

```
SELECT l.id, l.url, r.comment FROM ggl_links l
FULL OUTER JOIN ggl_rates r ON l.id = r.id
GO
<pre>
id    url                comment
101   www.globalguideline.com    The best
102   www.globalguideline.com/html Well done
103   www.globalguideline.com/se  Thumbs up
104   www.google.com            NULL
105   www.yahoo.com             NULL
106   www.php.net               NULL
107   www.mysql.com             NULL
NULL  NULL                    Number 1
NULL  NULL                    Not bad
NULL  NULL                    Good job
NULL  NULL                    Nice tool</pre>
```

As you can see, a full outer join returns 3 groups of rows:

- * The rows from both tables that satisfy the join condition.
- * The rows from the first (left) table that do not satisfy the join condition.
- * The rows from the second (right) table that do not satisfy the join condition.

[View All Answers](#)

Question - 318:

How To Write a Query with a Right Outer Join in MS SQL Server?

Ans:

If you want to query from two tables with a right outer join, you can use the RIGHT OUTER JOIN ... ON clause in the FROM clause. The following query returns output with a right outer join from two tables: ggl_links and ggl_rates. The join condition is that the id in the ggl_links table equals to the id in the ggl_rates table:

```
SELECT l.id, l.url, r.comment FROM ggl_links l
RIGHT OUTER JOIN ggl_rates r ON l.id = r.id
GO
id    url                comment
101   www.globalguideline.com    The best
102   www.globalguideline.com/html Well done
103   www.globalguideline.com/sql  Thumbs up
NULL  NULL                    Number 1
NULL  NULL                    Not bad
NULL  NULL                    Good job
NULL  NULL                    Nice tool
```

Note that a right outer join may return extra rows from the second (right) table that do not satisfy the join condition. In those extra rows, columns from the first (left) table will be given null values.

The extra rows returned from the right outer join in this example represents rates that have no links in the above example.

[View All Answers](#)

Question - 319:

How To Write a Query with a Left Outer Join in MS SQL Server?

Ans:

If you want to query from two tables with a left outer join, you can use the LEFT OUTER JOIN ... ON clause in the FROM clause. The following query returns output with a left outer join from two tables: ggl_links and ggl_rates. The join condition is that the id in the ggl_links table equals to the id in the ggl_rates table:

```
SELECT l.id, l.url, r.comment FROM ggl_links l
LEFT OUTER JOIN ggl_rates r ON l.id = r.id
GO
id    url                comment
101   www.globalguideline.com    The best
102   www.globalguideline.com/html Well done
103   www.globalguideline.com/sql  Thumbs up
104   www.google.com            NULL
105   www.yahoo.com             NULL
106   www.php.net               NULL
```



107 www.mysql.com NULL

Note that a left outer join may return extra rows from the first (left) table that do not satisfy the join condition. In those extra rows, columns from the second (right) table will be given null values.

The extra rows returned from the left outer join in this example represents links that have no rates in the above example.

[View All Answers](#)

Question - 320:

How To Define and Use Table Alias Names in MS SQL Server?

Ans:

When column names need to be prefixed with table names, you can define table alias name and use them to prefix column names. To define an alias for a table name, just enter the alias name right after the original table name in the FROM clause as shown in the following select statement:

```
SELECT l.id, l.url, r.comment FROM ggl_links l
INNER JOIN ggl_rates r ON l.id = r.id
GO
id url comment
101 www.globalguideline.com The best
102 www.globalguideline.com/html Well done
103 www.globalguideline.com/sql Thumbs up
```

[View All Answers](#)

Question - 321:

How To Write a Query with an Inner Join in MS SQL Server?

Ans:

If you want to query from two tables with an inner join, you can use the INNER JOIN ... ON clause in the FROM clause. The tutorial exercise below creates another testing table and returns output with an inner join from two tables: ggl_links and ggl_rates. The join condition is that the id in the ggl_links table equals to the id in the ggl_rates table:

```
CREATE TABLE ggl_rates (id INTEGER,
comment VARCHAR(16))
GO
INSERT INTO ggl_rates VALUES (101, 'The best')
GO
INSERT INTO ggl_rates VALUES (102, 'Well done')
GO
INSERT INTO ggl_rates VALUES (103, 'Thumbs up')
GO
INSERT INTO ggl_rates VALUES (204, 'Number 1')
GO
INSERT INTO ggl_rates VALUES (205, 'Not bad')
GO
INSERT INTO ggl_rates VALUES (206, 'Good job')
GO
INSERT INTO ggl_rates VALUES (207, 'Nice tool')
GO
SELECT ggl_links.id, ggl_links.url,
ggl_rates.comment FROM ggl_links
INNER JOIN ggl_rates ON ggl_links.id = ggl_rates.id
GO
id url comment
101 www.globalguideline.com The best
102 www.globalguideline.com/html Well done
103 www.globalguideline.com/sql Thumbs up
```

Note that when multiple tables are used in a query, column names need to be prefixed with table names in case the same column name exists in both tables.

[View All Answers](#)

Question - 322:

How To Join Two Tables in a Single Query in MS SQL Server?

Ans:

Two tables can be joined together in a query in 4 ways:

- * Inner Join: Returns only rows from both tables that satisfy the join condition.
- * Left Outer Join: Returns rows from both tables that satisfy the join condition, and the rest of rows from the first (left) table.
- * Right Outer Join: Returns rows from both tables that satisfy the join condition, and the rest of rows from the second (right) table.
- * Full Outer Join: Returns rows from both tables that satisfy the join condition, the rest of rows from the first (left) table, and the rest of rows from the second (right) table.

[View All Answers](#)

Question - 323:

How To Delete All Rows with TRUNCATE TABLE Statement in MS SQL Server?

Ans:

If you want to delete all rows from a table, you have two options:

- * Use the DELETE statement with no WHERE clause.
- * Use the TRUNCATE TABLE statement.

The TRUNCATE statement is more efficient than the DELETE statement. The tutorial exercise shows you a good example of TRUNCATE statement:

```
SELECT COUNT(*) FROM ggl_rates
```



```
GO
5
TRUNCATE TABLE ggl_rates
GO
SELECT COUNT(*) FROM ggl_rates
GO
0
```

[View All Answers](#)

Question - 324:

How To Delete Multiple Rows with One DELETE Statement in MS SQL Server?

Ans:

You can delete multiple rows from a table in the same way as deleting a single row, except that the WHERE clause will match multiple rows. The tutorial exercise below deletes 3 rows from the ggl_links table:

```
-- view rows to be deleted
SELECT id, url, notes, counts FROM ggl_links
WHERE id > 300
GO
id url notes counts
801 www.globalguideline.com Wrong 1202
802 www.globalguideline.com/html Wrong 1204
803 www.globalguideline.com/sql Wrong 1206
-- delete multiple rows
DELETE FROM ggl_links WHERE id > 300
GO
(3 row(s) affected)
-- try to view the deleted row
SELECT id, url, notes, counts FROM ggl_links
WHERE id > 300
GO
no rows
```

[View All Answers](#)

Question - 325:

How To Delete an Existing Row with DELETE Statements in MS SQL Server?

Ans:

If you want to delete an existing row from a table, you can use the DELETE statement with a WHERE clause to identify that row. Here is good sample of DELETE statements:

```
-- insert a row for this test
INSERT INTO ggl_links (url, id)
VALUES ('www.google.com', 301)
GO
(1 row(s) affected)
-- view the inserted row
SELECT id, url, notes, counts FROM ggl_links
WHERE id = 301
GO
id url notes counts
301 www.google.com NULL NULL
-- delete one row
DELETE FROM ggl_links WHERE id = 301
GO
(1 row(s) affected)
-- try to view the deleted row
SELECT id, url, notes, counts FROM ggl_links
WHERE id = 301
no rows
Row with id of 301 is truly deleted.
```

[View All Answers](#)

Question - 326:

What Happens If the UPDATE Subquery Returns Multiple Rows in MS SQL Server?

Ans:

If a subquery is used in a UPDATE statement, it must return exactly one row for each row in the update table that matches the WHERE clause. If it returns multiple rows, SQL Server will give you an error message. To test this out, you can try the following tutorial exercise:

```
-- insert two rows to ggl_rates
INSERT INTO ggl_rates VALUES (0, 'Number 1')
GO
INSERT INTO ggl_rates VALUES (0, 'Number 2')
GO
-- make sure there are 2 match rows
SELECT * FROM ggl_rates WHERE id = 0
GO
id comment
0 Number 1
0 Number 2
```




```
-- update with subquery that returns 2 rows
UPDATE ggl_links SET notes = (
    SELECT comment FROM ggl_rates
    WHERE ggl_rates.id = ggl_links.id
)
WHERE id = 0
GO
Msg 512, Level 16, State 1, Line 1
Subquery returned more than 1 value. This is not permitted
    when the subquery follows =, !=, <, <=, >, >= or when
    the subquery is used as an expression.
The statement has been terminated.
It is clear that we are using subquery as an expression, and it must return 0 or 1 row. Otherwise, we will get an error.
```

[View All Answers](#)

Question - 327:

What Happens If the UPDATE Subquery Returns No Rows in MS SQL Server?

Ans:

If you use a subquery to assign new values in the SET clause in an UPDATE statement, and the subquery returns no rows for an outer row, SQL Server will provide a NULL value to the SET clause. The tutorial exercise below shows you a good example:

```
-- insert a new row
INSERT INTO ggl_links (id, url, notes)
VALUES (0, 'www.globalguideline.com', 'Number one')
GO
(1 row(s) affected)
-- view old values
SELECT * FROM ggl_links WHERE id = 0
GO
id url          notes counts created
0 www.globalguideline.com Number one NULL 2007-05-23
-- make sure there is no matching row in fyi_rates
SELECT * FROM ggl_rates WHERE id = 0
GO
0 rows
-- update a subquery returning no rows
UPDATE ggl_links SET notes = (
    SELECT comment FROM ggl_rates
    WHERE ggl_rates.id = ggl_links.id
)
WHERE id = 0
(1 row(s) affected)
-- view new values
SELECT * FROM ggl_links WHERE id = 0
GO
id url          notes counts created
0 www.globalguideline.com NULL NULL 2007-05-23
Column "notes" gets updated with NULL if there is no return rows in the subquery.
```

[View All Answers](#)

Question - 328:

How To Use Values from Other Tables in UPDATE Statements in MS SQL Server?

Ans:

If you want to update values in one table with values from another table, you can use a subquery as an expression in the SET clause. The subquery should return only one row for each row in the update table that matches the WHERE clause. The tutorial exercise below shows you a good example:

```
-- Create another table
CREATE TABLE ggl_rates (id INTEGER,
    comment VARCHAR(16))
Go
-- Insert some rows in the new table
INSERT INTO ggl_rates VALUES (101, 'The best')
Go
INSERT INTO ggl_rates VALUES (102, 'Well done')
Go
INSERT INTO ggl_rates VALUES (103, 'Thumbs up')
Go
-- Update ggl_links with values from ggl_rates
UPDATE ggl_links SET notes = (
    SELECT comment FROM ggl_rates
    WHERE ggl_rates.id = ggl_links.id
)
WHERE id > 0 AND id < 110
GO
(3 row(s) affected)
-- View the updated values
SELECT * FROM ggl_links
WHERE id > 0 AND id < 110
GO
id url          notes counts created
```



101 www.globalguideline.com The best 999 2006-04-30
102 www.globalguideline.com/html Well done 0 2007-05-19
103 www.globalguideline.com/sql Thumbs up NULL 2007-05-19

Note that if column names are confusing between the inner table and the outer table, you need to pref

[View All Answers](#)

Question - 329:

Is the Order of Columns in the SET Clause Important in MS SQL Server?

Ans:

The answer is NO. The order of columns in the SET clause of the UPDATE statement is NOT important. You probably already noticed from the previous tutorial. There is a BIG DIFFERENCE among SQL Server, MySQL and Oracle on update multiple columns with previous values:

- * SQL Server provides you the existing values from the database on columns names used in new value expressions. So the order of columns in the SET clause is NOT important

- * MySQL provides you the updated values on columns names used in new value expressions. So the order of columns in the SET clause is important.

- * Oracle provides you the existing values from the database on columns names used in new value expressions. So the order of columns in the SET clause is NOT important

Here is a good tutorial exercise:

```
SELECT * FROM ggl_links
```

-- Check the old values

```
SELECT * FROM ggl_links WHERE url = 'www.globalguideline.com'
```

```
GO
```

```
id url notes counts created
```

```
101 www.globalguideline.com Good. 999 2006-04-30
```

-- Update "id" before "counts"

```
UPDATE ggl_links SET id = id+200, counts = id*2
```

```
WHERE url = 'www.globalguideline.com'
```

```
GO
```

(1 row(s) affected)

[View All Answers](#)

Question - 330:

How to use old values to define new values in UPDATE statements in MS SQL Server?

Ans:

If a row matches the WHERE clause in a UPDATE statement, existing values in this row can be used in expressions to provide new values in the SET clause. Existing values are represented by column names in the expressions. The tutorial exercise below shows you a good example:

```
SELECT * FROM ggl_links WHERE id >= 500
```

```
GO
```

```
id url notes counts created
```

```
601 www.globalguideline.com Wrong 9 2006-04-30
```

```
602 www.globalguideline.com/html Wrong 9 2007-05-21
```

```
603 www.globalguideline.com/sql Wrong 9 2007-05-23
```

```
UPDATE ggl_links SET id = id+200, counts = id*2
```

```
WHERE id >= 500
```

```
GO
```

(3 row(s) affected)

```
SELECT * FROM ggl_links WHERE id >= 500
```

```
GO
```

```
id url notes counts created
```

```
801 www.globalguideline.com Wrong 1202 2006-04-30
```

```
802 www.globalguideline.com/html Wrong 1204 2007-05-19
```

```
803 www.globalguideline.com/sql Wrong 1206 2007-05-19
```

This statement increased values in the id column by 200. It also updated the counts column with the newly increased id value.

[View All Answers](#)

Question - 331:

How To Update Multiple Rows with One UPDATE Statement in MS SQL Server?

Ans:

If the WHERE clause in an UPDATE statement matches multiple rows, the SET clause will be applied to all matched rows. This rule allows you to update values on multiple rows in a single UPDATE statement. Here is a good example:

```
SELECT * FROM ggl_links WHERE id >= 500
```

```
GO
```

```
id url notes counts created
```

```
601 ww.globalguideline.com NULL 0 2006-04-30
```

```
602 ww.globalguideline.com/html NULL 0 2007-05-19
```

```
603 ww.globalguideline.com/sql NULL NULL 2007-05-19
```

```
SELECT * FROM ggl_links
```

```
UPDATE ggl_links SET counts = 9, notes = 'Wrong'
```

```
WHERE id >= 500
```

(3 row(s) affected)

```
SELECT * FROM ggl_links WHERE id >= 500
```

```
GO
```

```
id url notes counts created
```

```
601 ww.globalguideline.com Wrong 9 2006-04-30
```

```
602 ww.globalguideline.com/html Wrong 9 2007-05-19
```

```
603 ww.globalguideline.com/sql Wrong 9 2007-05-19
```

The UPDATE statement updated 3 rows with the same new values.



[View All Answers](#)

Question - 332:

How To Update Values in a Table with UPDATE Statements in MS SQL Server?

Ans:

If you want to update some values in one row or multiple rows in a table, you can use the UPDATE statement. The tutorial script below shows a good example:

```
SELECT * FROM ggl_links WHERE id = 101
GO
id url          notes counts created
101 www.globalguideline.com NULL 0 2006-04-30
UPDATE ggl_links SET counts = 999, notes = 'Good.'
WHERE id = 101;
```

```
GO
(1 row(s) affected)
SELECT * FROM ggl_links WHERE id = 101
GO
```

```
id url          notes counts created
101 www.globalguideline.com Good. 999 2006-04-30
```

As you can see, the SET clause takes column and value pairs to provide new values, while the WHERE clause defines which row to apply the update.

[View All Answers](#)

Question - 333:

How To Insert Multiple Rows with One INSERT Statement in MS SQL Server?

Ans:

If you want to insert multiple rows with a single INSERT statement, you can use a subquery instead of the VALUES clause. Rows returned from the subquery will be inserted the target table. The following tutorial exercise gives you a good example:

```
INSERT INTO ggl_links SELECT id+500, REVERSE(url),
notes, counts, created FROM ggl_links
GO
(3 row(s) affected)
SELECT * FROM ggl_links
GO
id url          notes counts created
101 www.globalguideline.com NULL 0 2006-04-30
102 www.globalguideline.com/html NULL 0 2007-05-19
103 www.globalguideline.com/sql NULL NULL 2007-05-19
601 www.globalguideline.com/seo NULL 0 2006-04-30
602 www.globalguideline.com/xml NULL 0 2007-05-19
603 www.globalguideline.com/JavaScript_Guide NULL NULL 2007-05-19
```

As you can see, "INSERT INTO ... SELECT ..." is powerful statement. you can use it build up data in tables quickly.

[View All Answers](#)

Question - 334:

What Happens If You Insert a Duplicate Key for the Primary Key Column in MS SQL Server?

Ans:

If your table has a primary key column, and you are trying to insert a new row with duplicate key value on the primary key column, you will get an error. The reason is simple - Primary key column does not allow duplicate values. The following tutorial exercise gives you a good example:

```
SELECT * FROM ggl_links
INSERT INTO ggl_links VALUES (101,
'www.globalguideline.com',
NULL,
0,
'2006-04-30')
GO
Msg 2627, Level 14, State 1, Line 1
Violation of PRIMARY KEY constraint
'PK__ggl_links__03317E3D'. Cannot insert duplicate
key in object 'dbo.ggl_links'.
```

The statement has been terminated.

You are getting this error, because value "101" has already been used by an existing row.

[View All Answers](#)

Question - 335:

How to provide column names in INSERT Statements in MS SQL Server?

Ans:

If you don't want to specify values for columns that have default values, or you want to specify values to columns in an order different than how they are defined, you can provide a column list in the INSERT statement. If a column is omitted in the column, SQL Server applies 3 rules:

- * If default value is defined for the column, that default value will be used.
- * If no default value is defined for the column and NULL is allowed, NULL will be used.
- * If no default value is defined for the column and NULL is not allowed, SQL Server will reject the insert statement with an error.

The following tutorial exercise gives you some good examples:

```
INSERT INTO ggl_links (url, id)
VALUES ('www.globalguideline.com',103)
```



```
GO
(1 row(s) affected)
INSERT INTO ggl_links (id) VALUES (110)
GO
Msg 515, Level 16, State 2, Line 1
Cannot insert the value NULL into column 'url',
table 'GlobalGuideLineDatabase.dbo.ggl_links'; column does
not allow nulls. INSERT fails.
The statement has been terminated.
SELECT * FROM ggl_links
GO
id url notes counts created
101 www.globalguideline.com NULL 0 2006-04-30
102 www.globalguideline.com NULL 0 2007-05-19
103 www.globalguideline.com NULL NULL 2007-05-19
```

[View All Answers](#)

Question - 336:

How To Use Column Default Values in INSERT Statements in MS SQL Server?

Ans:

If a column is defined with a default value in a table, you can use the key word DEFAULT in the INSERT statement to take the default value for that column. The following tutorial exercise gives a good example:

```
INSERT INTO ggl_links VALUES (102,
'www.globalguideline.com',
NULL,
0,
DEFAULT)
GO
(1 row(s) affected)
SELECT * FROM fyi_links
GO
id url notes counts created
101 www.globalguideline.com NULL 0 2006-04-30
102 www.globalguideline.com NULL 0 2007-05-19
The default value, getdate(), is used for "created" column, which gives the current date.
```

[View All Answers](#)

Question - 337:

How To Insert a New Row into a Table with "INSERT INTO" Statements in MS SQL Server?

Ans:

To insert a new row into a table, you can use the INSERT INTO statement with values specified for all columns as in the following syntax:

```
INSERT INTO table_name VALUES (list_of_values_of_all columns)
```

Note that the list of values of all columns must be specified in the same order as how columns are defined in the CREATE TABLE statement. The following tutorial example inserts a row into "ggl_links":

```
INSERT INTO ggl_links VALUES (101,
'www.globalguideline.com',
NULL,
0,
'2006-04-30')
GO
(1 row(s) affected)
SELECT * FROM ggl_links
GO
id url notes counts created
101 www.globalguideline.com NULL 0 2006-04-30
The values are stored in the new record nicely.
```

[View All Answers](#)

Question - 338:

How To Create a Testing Table with Test Data in MS SQL Server?

Ans:

If you want to practice DML statements, like INSERT, UPDATE and DELETE statements, you should create a testing table. The tutorial exercise shows you a good example:

```
CREATE TABLE ggl_links (id INTEGER PRIMARY KEY,
url VARCHAR(80) NOT NULL,
notes VARCHAR(1024),
counts INT,
created DATETIME NOT NULL DEFAULT(getdate()))
GO
SELECT c.column_id as seq, c.name, x.name as type,
c.max_length, c.is_nullable
FROM sys.columns c, sys.tables t, sys.systypes x
WHERE c.object_id = t.object_id
AND c.system_type_id = x.xtype
AND t.name = 'ggl_links'
ORDER BY c.column_id
```



GO

| seq | name | type | max_length | is_nullable |
|-----|---------|----------|------------|-------------|
| 1 | id | int | 4 | 0 |
| 2 | url | varchar | 80 | 0 |
| 3 | notes | varchar | 1024 | 1 |
| 4 | counts | int | 4 | 1 |
| 5 | created | datetime | 8 | 0 |

You should keep this table to practice other tutorial exercises presented in this collection.

[View All Answers](#)

Question - 339:

What Are DML (Data Manipulation Language) Statements in MS SQL Server?

Ans:

DML (Data Manipulation Language) statements are statements to change data values in database tables. There are 3 primary DML statements:

- * INSERT - Inserting new rows into database tables. For example "INSERT INTO gg1_links VALUES (101, 'www.globalguideline.com', NULL, 0, '2006-04-30')"
- inserts a single new row in the gg1_links table.
- * UPDATE - Updating existing rows in database tables.
- * DELETE - Deleting existing rows from database tables.

[View All Answers](#)

Question - 340:

How To Drop an Existing Table with "DROP TABLE" Statements in MS SQL Server?

Ans:

If you want to delete an existing table and its data rows, you can use the "DROP TABLE" statement as shown in the tutorial script below:

```
SELECT * FROM tipBackup
GO
id subject description create_date
1 Learn SQL Visit www.globalguideline.com 2006-07-01
DROP TABLE tipBackup
GO
SELECT * FROM tipBackup
GO
Msg 208, Level 16, State 1, Line 1
Invalid object name 'tipBackup'.
```

Be careful, when you use the "DROP TABLE" statement. All data rows are gone too.

[View All Answers](#)

Question - 341:

How to rename an existing table with the "sp_rename" stored procedure in MS SQL Server?

Ans:

If you have an existing table and you want to change the table name, you can use the "sp_rename ... 'OBJECT'" stored procedure. "sp_rename" allows you to change names of COLUMN, DATABASE, INDEX, USERDATATYPE, and OBJECT (including tables). The tutorial example below shows you how to rename a table:

```
sp_rename 'tip', 'faq', 'OBJECT'
GO
Caution: Changing any part of an object name could break
scripts and stored procedures.
SELECT name, type_desc, create_date FROM sys.tables
GO
```

| name | type_desc | create_date |
|-----------|------------|-------------------------|
| faq | USER_TABLE | 2007-05-19 23:05:43.700 |
| tipBackup | USER_TABLE | 2007-05-19 23:25:23.357 |

You can also rename a table with on the Object Explorer window of SQL Server Management Studio. See tutorials on rename table columns.

[View All Answers](#)

Question - 342:

How to change the data type of an existing column with "ALTER TABLE" statements in MS SQL Server?

Ans:

Sometimes, you may need to change the data type of an existing column. For example, you want increase the string length of a column. You can use the "ALTER TABLE ... ALTER COLUMN" statements in the following syntax:

ALTER TABLE table_name ALTER COLUMN column_name new_type

Here is a good example of change column data types:

```
-- Can not make a string column shorter
ALTER TABLE tip ALTER COLUMN subject VARCHAR(10)
GO
Msg 8152, Level 16, State 14, Line 1
String or binary data would be truncated.
The statement has been terminated.
```

```
-- Can make a string column longer
ALTER TABLE tip ALTER COLUMN subject VARCHAR(100)
GO
```

Command(s) completed successfully.

```
-- Can not change string to numeric
```

```
ALTER TABLE tip ALTER COLUMN subject NUMBER
```



GO

Msg 8114, Level 16, State 5, Line 1

Error converting data type varchar to numeric.

The statement has been terminated.

As you can see, the new date type must be compatible with the old data type in order for the "ALTER TABLE ... ALTER COLUMN" statement to work.

[View All Answers](#)

Question - 343:

How to rename an existing column with SQL Server Management Studio?

Ans:

If you are using SQL Server Management Studio, you can rename almost any data objects through the Object Explorer window. The tutorial example below shows you how to rename a column:

1. Run SQL Server Management Studio and connect to SQL server.
2. On the Object Explorer window, follow the object tree: Databases > GlobalGuideLineDatabase > Tables > dbo.tip > Columns > title.
3. Click right mouse button on "title". The context menu shows up.
4. Select "Rename", type "subject" over "title", and press Enter key. The column name will be changed.

[View All Answers](#)

Question - 344:

How to rename an existing column with the "sp_rename" stored procedure in MS SQL Server?

Ans:

If you have an existing column in a table and you want to change the column name, you can use the "sp_rename ... 'COLUMN'" stored procedure. "sp_rename" allows you to change names of COLUMN, DATABASE, INDEX, USERDATATYPE, and OBJECT. The tutorial example below shows you how to rename a column:

USE master

GO

sp_rename 'tip.subject', 'title', 'COLUMN'

GO

Msg 15248, Level 11, State 1, Procedure sp_rename, Line 213

Either the parameter @objname is ambiguous or the claimed

@objtype (COLUMN) is wrong.

USE GlobalGuideLineDatabase

GO

sp_rename 'tip.subject', 'title', 'COLUMN'

GO

Caution: Changing any part of an object name could break scripts and stored procedures.

SELECT id, title, description, author FROM tip

GO

id title description author

1 Learn SQL Visit www.globalguideline.com NULL

You are getting the first error because 'GlobalGuideLineDatabase' is not the current database.

[View All Answers](#)

Question - 345:

How To Add a New Column to an Existing Table with "ALTER TABLE ... ADD" in MS SQL Server?

Ans:

If you have an existing table with existing data rows, and want to add a new column to that table, you can use the "ALTER TABLE ... ADD" statement. The tutorial script below shows you a good example:

ALTER TABLE tip ADD author VARCHAR(40)

GO

sp_columns tip

GO

TABLE_OWNER TABLE_NAME COLUMN_TABLE TYPE_NAME ...

dbo tip id int ...

dbo tip subject varchar ...

dbo tip description varchar ...

dbo tip create_date datetime ...

dbo tip author datetime ...

SELECT * FROM tip

GO

id subject description create_date author

1 Learn SQL Visit www.globalguideline.com 2008-05-01 NULL

This SQL script added a new column called "author" to the "tip" table. NULL values were added to this column on all existing data rows.

[View All Answers](#)

Question - 346:

How to create new tables with "SELECT ... INTO" statements in MS SQL Server?

Ans:

Let's say you have a table with many data rows, now you want to create a backup copy of this table of all rows or a subset of them, you can use the "SELECT ... INTO" statement. The tutorial script below gives you a good example:

INSERT INTO tip VALUES (1, 'Learn SQL',



```
'Visit www.GlobalGuideLine.com',2006-05-01')
```

```
GO
```

```
SELECT * INTO tipBackup FROM tip
```

```
GO
```

```
(1 rows affected)
```

```
SELECT * FROM tipBackup
```

```
GO
```

```
id subject description create_date
```

```
1 Learn SQL Visit www.globalguideline.com 2008-05-01
```

```
sp_columns tipBackup
```

```
GO
```

```
TABLE_OWNER TABLE_NAME COLUMN_TABLE TYPE_NAME ...
```

```
dbo tipBackup id int ...
```

```
dbo tipBackup subject varchar ...
```

```
dbo tipBackup description varchar ...
```

```
dbo tipBackup create_date datetime ...
```

As you can see, the "SELECT ... INTO" statement created a table called "tipBackup" using the same column definitions as the "tip" table and copied all data rows into "tipBackup".

[View All Answers](#)

Question - 347:

How To Generate CREATE TABLE Script on an Existing Table in MS SQL Server?

Ans:

If you want to know how an existing table was created, you can use SQL Server Management Studio to automatically generate a "CREATE TABLE" script. The following tutorial shows you how to do this:

1. Run SQL Server Management Studio and connect to SQL server.
2. On the Object Explorer window, follow the object tree: Databases > GlobalGuideLineDatabase > Tables > dbo.tip.
3. Click right mouse button on dbo.tip. The context menu shows up.
4. Select "Script Table as" > "CREATE to" > "New Query Editor Window". The following script will be displayed:

```
USE [GlobalGuideLineDatabase]
```

```
GO
```

```
/****** Object: Table [dbo].[tip]
```

```
Script Date: 05/05/2008 11:34:57 *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[tip](
```

```
[id] [int] NOT NULL,
```

```
[subject] [varchar](80)
```

```
COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
```

```
[description] [varchar](256)
```

```
COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
```

```
[create_date] [datetime] NULL,
```

```
PRIMARY KEY CLUSTERED (
```

```
[id] ASC
```

```
)WITH (PAD_INDEX = OFF,
```

```
IGNORE_DUP_KEY = OFF) ON [PRIMARY]
```

```
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

[View All Answers](#)

Question - 348:

How To Get a List of Columns using the "sp_help" Stored Procedure in MS SQL Server?

Ans:

Another way to get a list of columns from a table is to use the "sp_help" stored procedure. "sp_help" returns more than just a list of columns. It returns: the table information, the column information, the identity column, the row GUID column, the primary key, indexes, and constraints. If you run "sp_help tip" in SQL Server Management Studio, you will see the result as shown in this picture:

sp_help Stored Procedure

[View All Answers](#)

Question - 349:

How To Get a List of Table Columns using the "sp_columns" Stored Procedure in MS SQL Server?

Ans:

If you have an existing table, but you don't remember what are the columns defined in the table, you can use the "sp_columns" stored procedure to get a list of all columns of the specified table. The following tutorial script shows you a good example:

```
sp_columns tip
```

```
GO
```

```
TABLE_OWNER TABLE_NAME COLUMN_TABLE TYPE_NAME ...
```

```
dbo tip id int ...
```

```
dbo tip subject varchar ...
```

```
dbo tip description varchar ...
```

```
dbo tip create_date datetime ...
```




The "sp_columns" stored procedure returns a long list of properties for each column of the specified table. Take a look at each of them.

[View All Answers](#)

Question - 350:

How To Get a List of Columns using the "sys.columns" View in MS SQL Server?

Ans:

If you have an existing table, but you don't remember what are the columns defined in the table, you can use the "sys.columns" system view to get a list of all columns of all tables in the current database.

In order to a list of columns of a single table, you need to join sys.columns and sys.tables as shown in the tutorial example below:

```
SELECT * FROM sys.columns c, sys.tables t
WHERE c.object_id = t.object_id
AND t.name = 'tip'
```

GO

| object_id | name | column_id | user_type_id | max_length |
|------------|-------------|-----------|--------------|------------|
| 2073058421 | id | 1 | 56 | 4 |
| 2073058421 | subject | 2 | 167 | 80 |
| 2073058421 | description | 3 | 167 | 256 |
| 2073058421 | create_date | 4 | 61 | 8 |

You can see the column names easily from the sys.columns view. But you can only see the column type IDs. This requires another join to get the column type names.

You may try the "sp_columns" stored procedure to get a better list of columns shown in the next tutorial.

[View All Answers](#)

Question - 351:

How To Get a List of All Tables with "sys.tables" View in MS SQL Server?

Ans:

If you want to see the table you have just created, you can use the "sys.tables" system view to get a list of all tables in the current database. The tutorial script gives you a good example:

```
SELECT name, type_desc, create_date FROM sys.tables
```

GO

| name | type_desc | create_date |
|------|------------|-------------------------|
| tip | USER_TABLE | 2007-05-19 23:05:43.700 |

The output shows that there is only one table in the current database.

[View All Answers](#)

Question - 352:

How to create new tables with "CREATE TABLE" statements in MS SQL Server?

Ans:

If you want to create a new table, you can use the "CREATE TABLE" statement. The following tutorial script shows you how to create a table called "tip":

```
CREATE TABLE tip (id INTEGER PRIMARY KEY,
subject VARCHAR(80) NOT NULL,
description VARCHAR(256) NOT NULL,
create_date DATETIME NULL)
```

GO

This script creates a testing table called "tip" with 4 columns in the current database.

[View All Answers](#)

Question - 353:

What are DDL (Data Definition Language) statements for tables in MS SQL Server?

Ans:

DDL (Data Definition Language) statements are statements to create and manage data objects in the database. There are three primary DDL statements to create and manage tables:

- * CREATE TABLE - Creating a new table.
- * ALTER TABLE - Altering the definition of an existing table.
- * DROP TABLE - Dropping an existing table.

[View All Answers](#)

Question - 354:

What is a database table?

Ans:

A table in database is a data object used to store data. Tables have the following features:

- * Data is stored in a table with a structure of rows and columns.
- * Columns must be pre-defined with names, types and constraints.
- * A table object may have other associated data objects like, constraints, triggers, indexes, and statistics.

For example, a table called Address may have columns defined to store different elements of an address like, street number, city, country, postal code, etc.

[View All Answers](#)

Question - 355:

What are system databases in MS SQL Server?

Ans:



System databases are created by the SQL Server itself during the installation process. System databases are used by the SQL server to help manage other user databases and client execution sessions. SQL Server 2005 Express Edition uses 4 system databases:

- * master - The brain of a SQL server - Stores server configuration, runtime information, and database metadata.
- * model - An empty database model - Used to clone new databases.
- * msdb - The background job scheduler - Used for background jobs and related tasks.
- * tempdb - The temporary database - Used by the server as a scratch pad.

[View All Answers](#)

Question - 356:

How to set database to be SINGLE_USER in MS SQL Server?

Ans:

Databases in SQL Server have three user access options:

- * MULTI_USER - All users that have the appropriate permissions to connect to the database are allowed. This is the default.
- * SINGLE_USER - One user at a time is allowed to connect to the database. All other user connections are broken.
- * RESTRICTED_USER - Only members of the db_owner fixed database role and dbcreator and sysadmin fixed server roles are allowed to connect to the database, but it does not limit their number.

You can use the "ALTER DATABASE" to change database user access options as shown in the tutorial below:

```
USE GlobalGuideLineDatabase
GO
ALTER DATABASE GlobalGuideLineDatabase SET SINGLE_USER
GO
Now connect to server with another client session and try:
USE GlobalGuideLineDatabase
GO
Msg 924, Level 14, State 1, Line 1
Database 'GlobalGuideLineDatabase' is already open and can only
have one user at a time.
Go back to the first session and re-set the database to MULTI_USER:
ALTER DATABASE GlobalGuideLineDatabase SET MULTI_USER
GO
```

[View All Answers](#)

Question - 357:

How to set database to be READ_ONLY in MS SQL Server?

Ans:

Databases in SQL Server have two update options:

- * READ_WRITE - Data objects are allowed to be queried and modified. This is the default.
- * READ_ONLY - Data objects are allowed to be queried, but not allowed to be modified.

You can use the "ALTER DATABASE" to change database update options as shown in the tutorial below:

```
USE GlobalGuideLineDB
GO
INSERT Links (Name) VALUES ('www.globalguideline.com')
GO
(1 rows affected)
ALTER DATABASE GlobalGuideLineDB SET READ_ONLY
GO
INSERT Links (Name) VALUES ('www.globalguideline.com')
GO
Msg 3906, Level 16, State 1, Server SQLEXPRESS, Line 1
Failed to update database "GlobalGuideLineDB" because
the database is read-only.
SELECT * FROM Links
GO
Name
www.globalguideline.com
ALTER DATABASE GlobalGuideLineDB SET READ_WRITE
GO
INSERT Links (Name) VALUES ('www.globalguideline.com')
GO
(1 rows affected)
As you can see from the output, inserting data into a table is not allowed if the database is in READ_ONLY mode.
```

[View All Answers](#)

Question - 358:

How to move database physical files in MS SQL Server?

Ans:

If you want to move database physical files to a new location, you can use the "ALTER DATABASE" statements to bring the database offline, and link it to the files at the new location. The following tutorial gives you a good example:

```
ALTER DATABASE GlobalGuideLine SET ONLINE
GO
USE GlobalGuideLine
GO
CREATE TABLE Links (Name NVARCHAR(32))
GO
ALTER DATABASE GlobalGuideLine SET OFFLINE
GO
```



Now it is safe to move the database physical files to a new location:

1. Run Windows Explorer
2. Create a new directory: c: empdata
3. Drag and drop c: empGlobalGuideLine.mdf to c: empdata
3. Drag and drop c: empGlobalGuideLine.mdf to c: empdata

Go back to the SQL client program and run:

```
ALTER DATABASE GlobalGuideLine
MODIFY FILE (NAME = GlobalGuideLineDB,
FILENAME = 'C: empdataGlobalGuideLineDB.mdf')
GO
```

[View All Answers](#)

Question - 359:

How to set a database state to OFFLINE in MS SQL Server?

Ans:

If you want to move database physical files, you should take the database offline by using the "ALTER DATABASE" statement with the following syntax:

```
ALTER DATABASE database_name SET OFFLINE
```

The following tutorial example will bring "GlobalGuideLine" offline:

```
ALTER DATABASE GlobalGuideLine SET OFFLINE
```

```
GO
```

```
SELECT name, state_desc from sys.databases
```

```
GO
```

```
name          state_desc
```

```
master        ONLINE
```

```
tempdb        ONLINE
```

```
model         ONLINE
```

```
msdb          ONLINE
```

```
GlobalGuideLine OFFLINE
```

```
USE GlobalGuideLine
```

```
GO
```

```
Msg 942, Level 14, State 4, Line 1
```

```
Database 'GlobalGuideLine' cannot be opened because
it is offline.
```

[View All Answers](#)

Question - 360:

What are database states in MS SQL Server?

Ans:

A database is always in one specific state. For example, these states include ONLINE, OFFLINE, or SUSPECT. To verify the current state of a database, select the state_desc column in the sys.databases catalog view. The following table defines the database states.

- * ONLINE - Database is available for access. The primary filegroup is online, although the undo phase of recovery may not have been completed.

- * OFFLINE - Database is unavailable. A database becomes offline by explicit user action and remains offline until additional user action is taken. For example, the database may be taken offline in order to move a file to a new disk. The database is then brought back online after the move has been completed.

- * RESTORING - One or more files of the primary filegroup are being restored, or one or more secondary files are being restored offline. The database is unavailable.

- * RECOVERING - Database is being recovered. The recovering process is a transient state; the database will automatically become online if the recovery succeeds. If the recovery fails, the database will become suspect. The database is unavailable.

[View All Answers](#)

Question - 361:

Why I am getting this error when renaming a database in MS SQL Server?

Ans:

If you are trying to rename a database that is in use, you will get an error message like this: "The database could not be exclusively locked to perform the operation."

Before renaming a database, you must stop all client sessions using this database. Otherwise, you will get an error as shown in this tutorial example:

1. Launch one instance of SQL Server Management Studio and run:

```
USE GlobalGuideLine
```

```
GO
```

2. Keep the first instance running and launch another instance of SQL Server Management Studio:

```
ALTER DATABASE GlobalGuideLine
```

```
MODIFY NAME = GlobalGuideLineDatabase
```

```
GO
```

```
Msg 5030, Level 16, State 2, Server LOCALHOSTSQLEXPRESS
```

```
The database could not be exclusively locked to perform
the operation.
```

Obviously, the first instance is blocking the "ALTER DATABASE" statement.

[View All Answers](#)

Question - 362:

How to rename databases in MS SQL Server?

Ans:

If don't like the name of a database, you can change it by using the "ALTER DATABASE" statement with the following syntax:

```
ALTER DATABASE database_name
```

```
MODIFY NAME = new_database_name
```



The tutorial example below shows you how change the database name from "GlobalGuideLineDatabase" to "GlobalGuideLine":

```
ALTER DATABASE GlobalGuideLineDatabase
```

```
    MODIFY NAME = GlobalGuideLine
```

```
GO
```

The database name 'GlobalGuideLine' has been set.

[View All Answers](#)

Question - 363:

How to create database with physical files specified in MS SQL Server?

Ans:

If you don't like the default behavior of the CREATE DATABASE statement, you can specify the physical database files with a longer statement:

```
CREATE DATABASE database_name
```

```
    ON (NAME = logical_data_name,
```

```
        FILENAME = physical_data_name,
```

```
        SIZE = x, MAXSIZE = y, FILEGROWTH = z)
```

```
    LOG ON (NAME = logical_log_name,
```

```
        FILENAME = physical_log_name,
```

```
        SIZE = x, MAXSIZE = y, FILEGROWTH = z)
```

For example, the following statement will create a database with database files located in the C: emp directory:

```
USE master
```

```
GO
```

```
DROP DATABASE GlobalGuideLineDatabase
```

```
GO
```

```
CREATE DATABASE GlobalGuideLineDatabase
```

```
    ON (NAME = GlobalGuideLineDatabase,
```

```
        FILENAME = 'C: empGlobalGuideLineDatabase.mdf',
```

```
        SIZE = 10MB, MAXSIZE = 50MB, FILEGROWTH = 5MB)
```

```
    LOG ON (NAME = GlobalGuidelineLog,
```

```
        FILENAME = 'C: empGlobalGuideLineDatabase.ldf',
```

```
        SIZE = 1MB, MAXSIZE = 5MB, FILEGROWTH = 1MB)
```

```
GO
```

```
SELECT type_desc, name, physical_name, size
```

```
    FROM sys.database_files
```

```
GO
```

```
type_desc name          physical_name          size
```

```
ROWS      GlobalGuideLineDatabase C: empGlobalGuideLineDatabase.mdf 1280
```

```
LOG       GlobalGuidelineLog C: empGlobalGuideLineDatabase.ldf 128
```

[View All Answers](#)

Question - 364:

Where is my database stored on the hard disk in MS SQL Server?

Ans:

If a database is created with simple CREATE DATABASE statement, the server will create two database files on the hard disk to store data and configuration information about that data bases:

- * database_name.mdf - SQL Server Database Primary Data File

- * database_name_log.ldf - SQL Server Database Transaction Log File

To find out the location of database files, you can query the "sys.database_files" view as shown in this tutorial example:

```
USE GlobalGuideLineDatabase
```

```
GO
```

```
SELECT type_desc, physical_name, size
```

```
    FROM sys.database_files
```

```
GO
```

```
type_desc physical_name          size
```

```
ROWS      c:Program FilesMicrosoft SQL Server
```

```
          MSSQL.1\MSSQLDATA\GlobalGuideLineDatabase.mdf 152
```

```
LOG       c:Program FilesMicrosoft SQL Server
```

```
          MSSQL.1\MSSQLDATA\GlobalGuideLineDatabase_log.LDF 63
```

Go verify these two files with Windows Explorer.

[View All Answers](#)

Question - 365:

How to get a list all databases on the SQL server?

Ans:

If you don't remember database names you have created, you can get a list of all databases on the server by query the "sys.databases" view as shown in this tutorial example:

```
CREATE DATABASE GlobalGuideLineDatabase
```

```
GO
```

```
SELECT name, database_id, create_date FROM sys.databases
```

```
GO
```

```
[name]    [database_id] [create_date]
```

```
master    1  2003-04-08 09:13:36.390
```

```
tempdb    2  2007-05-19 13:42:42.200
```

```
model     3  2003-04-08 09:13:36.390
```

```
msdb      4  2005-10-14 01:54:05.240
```

```
GlobalGuideLineDatabase 5  2007-05-19 20:04:39.310
```

As you can see, the newly created database is listed at the end of query result.



[View All Answers](#)

Question - 366:

Why I am getting this error when dropping a database in MS SQL Server?

Ans:

If you are trying to drop a database that is in use, you will get an error message like this: 'Cannot drop database "GlobalGuidelineData" because it is currently in use.' Before dropping a database, you must stop all client sessions using this database. If your own client session is using this database, you should set a different database as the current database as shown in this tutorial example:

```
CREATE DATABASE GlobalGuidelineData
GO
USE GlobalGuidelineData
GO
DROP DATABASE GlobalGuideLineDatabase
GO
Msg 3702, Level 16, State 4, Server LOCALHOSTS\SQLSERVER
Cannot drop database "GlobalGuideLineDatabase" because it is
currently in use.
USE master
GO
DROP DATABASE GlobalGuideLineDatabase
GO
```

[View All Answers](#)

Question - 367:

How to delete a database in MS SQL Server?

Ans:

If you created a database incorrectly, or you have a database that is not needed any more, you can delete it with the "DROP DATABASE" statement with this syntax:

```
DROP DATABASE database_name
For example, execute this statement:
DROP DATABASE YourDataBaseName
GO
```

The database "YourDataBaseName" created in the previous tutorial should be deleted from the SQL server.

Warning, if you delete a database, all tables and their data in that database will be deleted.

[View All Answers](#)

Question - 368:

How to set the current database in MS SQL Server?

Ans:

Once you are connected to the SQL Server, you should select a database to work with and set it as the current database using the "USE" statement with this syntax:

```
USE database_name
The following tutorial example shows you how to set "YourDataBaseName" as the current database, and create a table in "YourDataBaseName":
USE YourDataBaseName
GO
```

```
Changed database context to 'YourDataBaseName'.
CREATE TABLE Links (Name NVARCHAR(32))
GO
SELECT name, type_desc, create_date FROM sys.tables
GO
name type_desc create_date
Links USER_TABLE 2007-05-19 23:05:43.700
```

[View All Answers](#)

Question - 369:

What is the simplest way to create a new database in MS SQL Server?

Ans:

The simplest way to create a new database is to use the "CREATE DATABASE" statement with this syntax:

```
CREATE DATABASE database_name
For example, run this statement:
CREATE DATABASE YourDataBaseName
GO
```

A new database called "YourDataBaseName" should be created in the SQL server. Of course, YourDataBaseName database should be empty at this moment - no tables. But it should have some other data objects automatically created by the server.

[View All Answers](#)

Question - 370:

What is a database in MS SQL Server?

Ans:

A database is a logical container that contains a set of related database objects:

- * Tables - Storages of structured data.
- * Views - Queries to present data from tables.
- * Indexes - Sorting indexes to speed up searches.



- * Stored Procedures - Predefined SQL program units.
- * Users - Identifications used for data access control.
- * Other objects.

[View All Answers](#)

Question - 371:

How to delete database objects with "DROP" statements in MS SQL Server?

Ans:

To remove all database objects created by previous tutorials, you could just delete the database. However, in this tutorial, you will go through the steps to reverse every action you took doing the tutorial.

Removing permissions and objects - Before you delete objects, make sure you are in the correct database:

```
USE YourDataBaseName;
```

```
GO
```

Use the REVOKE statement to remove execute permission for Mary on the stored procedure:

```
REVOKE EXECUTE ON pr_Names FROM Mary;
```

```
GO
```

Use the DROP statement to remove permission for Mary to access the YourDataBaseName database:

```
DROP USER Mary;
```

```
GO
```

Use the DROP statement to remove permission for Mary to access this instance of SQL Server 2005:

```
DROP LOGIN [Mary];
```

```
GO
```

Use the DROP statement to remove the store procedure pr_Names:

```
DROP PROC pr_Names;
```

```
GO
```

Use the DROP statement to remove the view vw_Names:

```
DROP View vw_Names;
```

```
GO
```

Use the DELETE statement to remove all rows from the Products table:

```
DELETE FROM Products;
```

```
GO
```

Use the DROP statement to remove the Products table:

```
DROP Table Products;
```

```
GO
```

You cannot remove the YourDataBaseName database while you are in the database; therefore, first switch context to another database, and then use the DROP sta

[View All Answers](#)

Question - 372:

How to grant a permission in MS SQL Server using "GRANT EXECUTE" statements?

Ans:

This answer is about creating login and configure users for databases with Transact-SQL statements. Granting a user access to a database involves three steps. First, you create a login. The login lets the user connect to the SQL Server Database Engine. Then you configure the login as a user in the specified database. And finally, you grant that user permission to database objects. This answer shows you these three steps, and shows you how to create a view and a stored procedure as the object. This tutorial assumes that you are running SQL Server Management Studio Express.

As an administrator, you can execute the SELECT from the Products table and the vw_Names view, and execute the pr_Names procedure; however, Mary cannot. To grant Mary the necessary permissions, use the GRANT statement.

Procedure Title - Execute the following statement to give Mary the EXECUTE permission for the pr_Names stored procedure.

```
GRANT EXECUTE ON pr_Names TO Mary;
```

```
GO
```

[View All Answers](#)

Question - 373:

How to create a view and a stored procedure in MS SQL Server using "CREATE VIEW/PROCEDURE" statements?

Ans:

This answer is about creating login and configure users for databases with Transact-SQL statements. Granting a user access to a database involves three steps. First, you create a login. The login lets the user connect to the SQL Server Database Engine. Then you configure the login as a user in the specified database. And finally, you grant that user permission to database objects. This lesson shows you these three steps, and shows you how to create a view and a stored procedure as the object. This tutorial assumes that you are running SQL Server Management Studio Express.

Now that Mary can access the YourDataBaseName database, you may want to create some database objects, such as a view and a stored procedure, and then grant Mary access to them. A view is a stored SELECT statement, and a stored procedure is one or more Transact-SQL statements that execute as a batch.

Views are queried like tables and do not accept parameters. Stored procedures are more complex than views. Stored procedures can have both input and output parameters and can contain statements to control the flow of the code, such as IF and WHILE statements. It is good programming practice to use stored procedures for all repetitive actions in the database.

[View All Answers](#)

Question - 374:

How to create a user to access a database in MS SQL Server using "CREATE USER" statements?

Ans:

This answer is about creating login and configure users for databases with Transact-SQL statements. Granting a user access to a database involves three steps. First, you create a login. The login lets the user connect to the SQL Server Database Engine. Then you configure the login as a user in the specified database. And finally, you grant that user permission to database objects. This lesson shows you these three steps, and shows you how to create a view and a stored procedure as the object. This tutorial assumes that you are running SQL Server Management Studio Express.



Mary now has access to this instance of SQL Server 2005, but does not have permission to access the databases. She does not even have access to her default database YourDataBaseName until you authorize her as a database user.

To grant Mary access, switch to the YourDataBaseName database, and then use the CREATE USER statement to map her login to a user named Mary.

To create a user in a database - Type and execute the following statements (replacing computer_name with the name of your computer) to grant Mary access to the YourDataBaseName database.

```
USE [YourDataBaseName];
GO
CREATE USER [Mary] FOR LOGIN [computer_nameMary];
GO
Now, Mary has access to both SQL Server 2005 and the YourDataBaseName database.
```

[View All Answers](#)

Question - 375:

How to create a login account in MS SQL Server to access the database engine using "CREATE LOGIN" statements?

Ans:

Now this answer will teach you that how to create login and configure users for databases with Transact-SQL statements. Granting a user access to a database involves three steps. First, you create a login. The login lets the user connect to the SQL Server Database Engine. Then you configure the login as a user in the specified database. And finally, you grant that user permission to database objects. This lesson shows you these three steps, and shows you how to create a view and a stored procedure as the object. This tutorial assumes that you are running SQL Server Management Studio Express.

To access the Database Engine, users require a login. The login can represent the user's identity as a Windows account or as a member of a Windows group, or the login can be a SQL Server login that exists only in SQL Server. Whenever possible you should use Windows Authentication.

By default, administrators on your computer have full access to SQL Server. For this lesson, we want to have a less privileged user; therefore, you will create a new local Windows Authentication account on your computer. To do this, you must be an administrator on your computer. Then you will grant that new user access to SQL Server. The following instructions are for Windows XP Professional.

[View All Answers](#)

Question - 376:

How to read data in a table with "SELECT" statements?

Ans:

Now this part is for creating database objects with Transact-SQL statements. This Question shows you how to create a database, create a table in the database, and then access and change the data in the table. Because this Answer is an introduction to using Transact-SQL, it does not use or describe the many options that are available for these statements. This Guide assumes that you are running SQL Server Management Studio Express.

Use the SELECT statement to read the data in a table. The SELECT statement is one of the most important Transact-SQL statements, and there are many variations in the syntax. For this Answer, you will work with five simple versions.

To read the data in a table - Type and execute the following statements to read the data in the Products table.

```
-- The basic syntax for reading data from a single table
SELECT ProductID, ProductName, Price, ProductDescription
FROM dbo.Products
GO
```

You can use an asterisk to select all the columns in the table. This is often used in ad hoc queries. You should provide the column list in your permanent code so that the statement will return the predicted columns, even if a new column is added to the table later.

```
-- Returns all columns in the table
-- Does not use the optional schema, dbo
SELECT * FROM Products
GO
```

[View All Answers](#)

Question - 377:

How to insert and update data into a table with "INSERT" and "UPDATE" statements?

Ans:

Now you how to create a database, create a table in the database, and then access and change the data in the table. Because here is an introduction to using Transact-SQL, it does not use or describe the many options that are available for these statements. Now we assumes that you are running SQL Server Management Studio Express.

Now that you have created the Products table, you are ready to insert data into the table by using the INSERT statement. After the data is inserted, you will change the content of a row by using an UPDATE statement. You will use the WHERE clause of the UPDATE statement to restrict the update to a single row. The four statements will enter the following data.

```
ProductID ProductName Price ProductDescription
1 Clamp 12.48 Workbench clamp
50 Screwdriver 3.17 Flat head
75 Tire Bar Tool for changing tires
3000 3mm Bracket .52
```

The basic syntax is: INSERT, table name, column list, VALUES, and then a list of the values to be inserted. The two hyphens in front of a line indicate that the line is a comment and the text will be ignored by the compiler. In this case, the comment describes a permissible variation of the syntax.

[View All Answers](#)

Question - 378:

How to create new table with "CREATE TABLE" statements?

Ans:

This is the second tutorial of a quick lesson on creating database objects with Transact-SQL statements. This section shows you how to create a database, create a table in the database, and then access and change the data in the table. Because this section is an introduction to using Transact-SQL, it does not use or describe the many options that are available for these statements. This SQL Guide assumes that you are running SQL Server Management Studio Express.

To create a table, you must provide a name for the table, and the names and data types of each column in the table. It is also a good practice to indicate whether null



values are allowed in each column.

Most tables have a primary key, made up of one or more columns of the table. A primary key is always unique. The Database Engine will enforce the restriction that any primary key value cannot be repeated in the table.

[View All Answers](#)

Question - 379:

How to create new databases with "CREATE DATABASE" statements?

Ans:

This is the first SQL Questions Guide of a quick lesson on creating database objects with Transact-SQL statements. This section shows you how to create a database, create a table in the database, and then access and change the data in the table. Because this is an introduction to using Transact-SQL, it does not use or describe the many options that are available for these statements. This SQL Questions Guide assumes that you are running SQL Server Management Studio Express.

Like many Transact-SQL statements, the CREATE DATABASE statement has a required parameter: the name of the database. CREATE DATABASE also has many optional parameters, such as the disk location where you want to put the database files. When you execute CREATE DATABASE without the optional parameters, SQL Server uses default values for many of these parameters. This tutorial uses very few of the optional syntax parameters.

To create a database - In a Query Editor window, type but do not execute the following code:

```
CREATE DATABASE YourDataBaseName  
GO
```

[View All Answers](#)

Question - 380:

How to use Transact-SQL statements to access the database engine?

Ans:

Transact-SQL statements can be used to access the database engine directly. Here are some good tutorials provided by the SQL Server 2005 Books Online. See the SQL Server 2005 Tutorials > Database Engine Tutorials > Writing Transact-SQL Statements Tutorial section in the SQL Server 2005 Books Online document.

This SQL Questions Guide is intended for users who are new to writing SQL statements. It will help new users get started by reviewing some basic statements for creating tables and inserting data. This tutorial uses Transact-SQL, the Microsoft implementation of the SQL standard. This tutorial is intended as a brief introduction to the Transact-SQL language and not as a replacement for a Transact-SQL class. The statements in this tutorial are intentionally simple, and are not meant to represent the complexity found in a typical production database.

[View All Answers](#)

Question - 381:

How to run SQL Server 2005 Books Online on your local system?

Ans:

SQL Server 2005 Books Online can be accessed by a Web browser over the Internet. But you can also download it and read it on your local system. If you have downloaded and installed SQL Server 2005 Books Online package, you follow this tutorial to run it:

1. Click Start > Programs > Microsoft SQL Server 2005 > Documentation and Tutorials > Tutorials > SQL Server Tutorials. The SQL Server 2005 Books Online window shows up.
2. Click the plus sign (+) next to "SQL Server 2005 Tutorials in the Contents window".
3. Click the plus sign (+) next to "SQL Server Tools Tutorials".
4. Click "Lesson 1: Basic Navigation in SQL Server Management Studio". The book content shows up for you to read.

[View All Answers](#)

Question - 382:

How to run Queries with SQL Server Management Studio Express?

Ans:

1. Launch and connect SQL Server Management Studio Express to the local SQL Server 2005 Express.
2. Click on the "New Query" button below the menu line. Enter the following SQL statement in the query window:
SELECT 'Welcome to GlobalGuideLine.com Tips on SQL Server!'
3. Click the Execute button in the toolbar area. You should get the following in the result window:
Welcome to GlobalGuideLine.com Tips on SQL Server!

[View All Answers](#)

Question - 383:

How to download and install SQL Server 2005 Books Online?

Ans:

1. Go to the SQL Server 2005 Books Online download page.
2. Click the download button, the File Download box shows up. Save the download file to c: emp.
3. Double click on the downloaded file: c: empSqlServer2K5_BOL_Feb2007.msi. The installation setup window shows up. Follow the instructions to finish the installation.
4. When the installation is done. You will see a new entry in the Start menu: Start > Programs > Microsoft SQL Server 2005 > Documentation and Tutorials

[View All Answers](#)

Question - 384:

How to download and install Microsoft SQL Server Management Studio Express?

Ans:

Microsoft SQL Server Management Studio Express (SSMSE) is a free, easy-to-use graphical management tool for managing SQL Server 2005 Express Edition and SQL Server 2005 Express Edition with Advanced Services. If you want to download and install it to your system, follow this tutorial:

1. Go to SQL Server Management Studio Express home page.
2. Go to the "Files in This Download" section.



3. Click the Download button next to "SQLServer2005_SSMSEE.msi - 38.5 MB" And save it to c: emp directory.
4. Look at and compare the downloaded file properties with:
Name: SQLServer2005_SSMSEE.msi
Location: C: emp
Size: 40,364,032 bytes
5. Double click to install. The setup window shows up. Follow the instructions to finish the installation process.
6. When installation is done, you will see a new program menu entry as: Start > Programs > Microsoft SQL Server 2005 > SQL Server Management Studio Express

[View All Answers](#)

Question - 385:

How to connect SQL Server Management Studio Express to SQL Server 2005 Express?

Ans:

Once you have SQL Server 2005 Express installed and running on your local machine, you are ready to connect SQL Server Management Studio Express to the server:

Click Start > Programs > Microsoft SQL Server 2005 > SQL Server Management Studio Express to launch SQL Server Management Studio Express.

The "Connect to Server" box shows up. The Server Name field has a default value of "LOCALHOSTSQLEXPRESS". So don't change it. Select "SQL Server Authentication" as the Authentication. Enter enter "sa" as the Login, and "GlbaiGuideLine" as the Password.

SQL Server 2005 Connect Window.

Click the Connect button, you should see the SQL Server Management Studio Express window comes up.

[View All Answers](#)

Question - 386:

How do you know if SQL Server is running on your local system?

Ans:

After installing SQL Server 2006 Express Edition, it will be running on your local system quietly as a background process.

If you want to see this process is running, run Windows Task Manager. You should see a process called sqlservr.exe running in the process list:

sqlservr.exe 00 1,316 K

If you select sqlservr.exe and click the "End Process" button, SQL Server will be stopped.

If you can not find sqlservr.exe in the process list, you know that your SQL Server is running.

[View All Answers](#)

Question - 387:

What is sqlservr.exe - Process - SQL Server (SQLEXPRESS)?

Ans:

Process sqlservr.exe is the Microsoft SQL Server system service installed as part of the Microsoft SQL Server 2005 Express Edition.

mscorsvw.exe process and program file info:

CPU usage: 00%

Memory usage: 1,316K

Launching method: System Service

Directory: C:\Program Files\Microsoft SQL Server\MSSQL.1

MSSQLBinn

File name: sqlservr.exe

Description: SQL Server Windows NT

Size: 28,768,528 bytes

Date: Friday, October 14, 2005, 3:51:46 AM

Version: 2005.90.1399.0

Company name: Microsoft

System essential: No

Virus/Spyware/Adware: No

[View All Answers](#)

Question - 388:

How to install SQL Server 2005 Express Edition?

Ans:

Once you have downloaded SQL Server 2005 Express Edition, you should follow this tutorial to install it on your system:

1. Double click SQLEXPRESS.EXE. The setup window shows up.
2. Click Next to let the setup program to unpack all files from the downloaded file.
3. When unpack is down, the setup program will check all required programs on your system.
4. Then the setup program will start the installation process.
5. On the Authentication Mode window, click the radio button for Mixed Mode (Windows Authentication and SQL Server Authentication). And enter "GlbaiGuideLine" in the "Specify the sa logon password below:" fields.
6. Continue to finish the installation process.
7. When installation is done, you will see a new program menu entry as: Start > Programs > Microsoft SQL Server 2005 > Configuration Tools.

[View All Answers](#)

Question - 389:

What is mscorsvw.exe - Process - Microsoft .NET Framework NGEN?

Ans:

Process mscorsvw.exe is installed as a system service as part of the .NET Framework 2.0. You can disable it, if you are not using any applications that require .NET Framework 2.0.

mscorsvw.exe process and program file info:



CPU usage: 00%
Memory usage: 2,704K
Launching method: System Service
Directory: C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727
File name: mscorsvw.exe
Description: .NET Runtime Optimization Service
Size: 66,240 bytes
Date: Friday, September 23, 2005, 7:28:56 AM
Version: 2.0.50727.42
Company name: Microsoft
System essential: No
Virus/Spyware/Adware: No

[View All Answers](#)

Question - 390:

How to download and install Microsoft .NET Framework Version 2.0?

Ans:

.NET Framework Version 2.0 is required by many Microsoft applications like SQL Server 2005. If you want download and install .NET Framework Version 2.0, you should follow this tutorial:

1. Go to the Microsoft .NET Framework Version 2.0 Redistributable Package (x86) page.
2. Click the Download button. Save the download file dotnetfx.exe to c: emp.

3. Look at and compare the downloaded file properties with:

Name: dotnetfx.exe
Location: C: emp
Size: 23,510,720 bytes
Version: 2.0.50727.42

4. Close all IE (Internet Explorer) windows.

5. Double click the downloaded file: to c: empdotnetfx.exe. The Microsoft .NET Framework 2.0 Setup windows shows up. Follow the instructions to finish the installation.

[View All Answers](#)

Question - 391:

Why I am getting "The Microsoft .Net Framework 2.0 in not installed" message?

Ans:

When you try to install SQL Server 2005 Express Edition, you may get a Microsoft SQL Server 2005 Setup error box with this message: "The Microsoft .Net Framework 2.0 in not installed. Please install before running setup."

You are getting this error, because .NET Framework 2.0 is not installed on your system yet. Read the next tutorial to download and install .NET Framework 2.0.

[View All Answers](#)

Question - 392:

System Requirements for SQL Server 2005 Express Edition?

Ans:

The following system requirements cover the SQL Server 2005 Express Edition:

Processor

32-bit Processor of 600-megahertz (MHz) or faster

Operating System

Windows XP with Service Pack 2 or later

Windows 2000 Professional with SP4

Windows 2000 Server with Service Pack 4 or later

Windows Server 2003 Standard, or Enterprise SP1

Windows Server 2003 Web Edition SP1

Windows Small Business Server 2003 with SP1

Vista Home Basic and above

Framework

.NET Framework 2.0

Memory

512 megabytes (MB) or more recommended

Hard Disk

Approximately 425 MB of available hard-disk space

[View All Answers](#)

Question - 393:

How to download Microsoft SQL Server 2005 Express Edition?

Ans:

Microsoft SQL Server 2005 Express Edition is the free version of the Microsoft SQL Server 2005. If you are interested to try SQL Server 2005, you should follow this tutorial to download Microsoft SQL Server 2005 Express Edition:

1. Go to the Microsoft SQL Server 2005 Express Edition download page.

2. Go to the Files in This Download section, and click Download button next to the "SQLEXPRESS - 53.5 MB" file. The File Download box shows up.

3. Save the download file to C: emp directory. When the download is done, you should get the following file:

Name: SQLEXPRESS.EXE

Description: Microsoft SQL 2005 Server Express Edition

Location: C: emp



Size: 56,105,688 bytes

Version: 9.0.1399.6

[View All Answers](#)

Question - 394:

What is Microsoft SQL Server?

Ans:

Microsoft SQL Server is a relational database management system (RDBMS) developed by Microsoft. It runs on Windows systems and uses Transact-SQL as the query language.

Microsoft SQL Server release history:

- * 1993 - SQL Server 4.21 for Windows NT
- * 1995 - SQL Server 6.0, codenamed SQL95
- * 1996 - SQL Server 6.5, codenamed Hydra
- * 1999 - SQL Server 7.0, codenamed Sphinx
- * 1999 - SQL Server 7.0 OLAP, codenamed Plato
- * 2000 - SQL Server 2000 32-bit, codenamed Shiloh (version 8.0)
- * 2003 - SQL Server 2000 64-bit, codenamed Liberty
- * 2005 - SQL Server 2005, codenamed Yukon (version 9.0)
- * 2005 - SQL Server 2005 Express Edition, restricted free version

[View All Answers](#)

Databases Programming Most Popular & Related Interview Guides

- 1 : [RDBMS Interview Questions and Answers.](#)
- 2 : [SQL Interview Questions and Answers.](#)
- 3 : [SSRS Interview Questions and Answers.](#)
- 4 : [Sybase Interview Questions and Answers.](#)
- 5 : [Database Administrator \(DBA\) Interview Questions and Answers.](#)
- 6 : [Teradata Interview Questions and Answers.](#)
- 7 : [PostgreSQL Interview Questions and Answers.](#)
- 8 : [MongoDB Interview Questions and Answers.](#)
- 9 : [IBM DB2 Interview Questions and Answers.](#)
- 10 : [Data Modeling Interview Questions and Answers.](#)

Follow us on FaceBook

www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter

<https://twitter.com/InterviewQA>

For any inquiry please do not hesitate to contact us.

Interview Questions Answers.ORG Team

[https://InterviewQuestionsAnswers.ORG/
support@InterviewQuestionsAnswers.ORG](https://InterviewQuestionsAnswers.ORG/support@InterviewQuestionsAnswers.ORG)