MS SQL Server Joins Job Interview Questions And Answers

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 Joins 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 Joins 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 Joins category. To ensure quality, each submission is checked by our team, before it becomes live. This MS SQL Server Joins 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 Joins Interview Questions And Answers Guide.

Question - 1:

What is cross join in SQL server joins?

Ans:

A cross join that does not have a WHERE clause produces the Cartesian product of the tables involved in the join. The size of a Cartesian product result set is the number of rows in the first table multiplied by the number of rows in the second table. The common example is when company wants to combine each product with a pricing table to analyze each product at each price.

View All Answers

Question - 2:

What is Inner Join in SQL server joins?

Ans:

A join that displays only the rows that have a match in both joined tables is known as inner Join. This is the default type of join in the Query and View Designer.

View All Answers

Question - 3:

What is Outer Join in SQL server joins?

Ans:

A join that includes rows even if they do not have related rows in the joined table is an Outer Join. You can create three different outer join to specify the unmatched rows to be included:
* Left Outer Join
* Right Outer Join
* Full Outer Join

View All Answers

Question - 4:

What is Left Outer Join in SQL server joins?

Ans:

In Left Outer Join all rows in the first-named table i.e. "left" table, which appears leftmost in the JOIN clause are included. Unmatched rows in the right table do not appear.

View All Answers

Question - 5:

Explain dataWarehousing in SQL server?

Ans:

* Subject-oriented, meaning that the data in the database is organized so that all the data elements relating to the same real-world event or object are linked together.
* Time-variant, meaning that the changes to the data in the database are tracked and recorded so that reports can be produced showing changes over time.
* Non-volatile, meaning that data in the database is never over-written or deleted, once committed, the data is static, read-only, but retained for future reporting.
* Integrated, meaning that the database contains data from most or all of an organization's operational applications, and that this data is made consistent.

View All Answers

Question - 6:

Explain Identity in SQL Server?

Ans:

Identity (or Auto-Number) is a column that automatically generates numeric values. A start and increment value can be set, but most DBA leave these at 1. A GUID

column also generates numbers; the value of this cannot be controlled. Identity/GUID columns do not need to be indexed.

View All Answers

Question - 7:

What is multi-statement table-value user-defined function?

Ans:

A Multi-Statement Table-Value user-defined function returns a table and is also an exceptional alternative to a view as the function can support multiple T-SQL statements to build the final result where the view is limited to a single SELECT statement. Also, the ability to pass parameters into a TSQL select command or a group of them gives us the capability to in essence create a parameterized, non-update-able view of the data in the underlying tables. Within the create function command you must define the table structure that is being returned. After creating this type of user-defined function, It can be used in the FROM clause of a T-SQL command unlike the behavior found when using a stored procedure which can also return record sets.

View All Answers

Question - 8:

What is inline table-value user-defined function?

Ans:

An Inline Table-Value user-defined function returns a table data type and is an exceptional alternative to a view as the user-defined function can pass parameters into a T-SQL select command and in essence provide us with a parameterized, non-update-able view of the underlying tables.

View All Answers

Question - 9:

What is scalar user-defined function?

Ans:

A Scalar user-defined function returns one of the scalar data types. Text, ntext, image and time stamp data types are not supported. These are the type of user-defined functions that most developers are used to in other programming languages. You pass in 0 to many parameters and you get a return value.

View All Answers

Question - 10:

Explain user defined functions?

Ans:

User-Defined Functions allow defining its own T-SQL functions that can accept 0 or more parameters and return a single scalar data value or a table data type.

View All Answers

Question - 11:

Can you please explain the difference between primary keys and foreign keys?

Ans:

Primary keys are the unique identifiers for each row. They must contain unique values and cannot be null. Due to their importance in relational databases, Primary keys are the most fundamental of all keys and constraints. A table can have only one Primary key. Foreign keys are both a method of ensuring data integrity and a manifestation of the relationship between tables.

View All Answers

Question - 12:

What is self join in SQL server joins?

Ans:

This is a particular case when one table joins to itself, with one or two aliases to avoid confusion. A self join can be of any type, as long as the joined tables are the same. A self join is rather unique in that it involves a relationship with only one table. The common example is when company has a hierarchal reporting structure whereby one member of staff reports to another. Self Join can be Outer Join or Inner Join.

View All Answers

Question - 13:

What is full outer join in SQL server joins?

Ans:

In Full Outer Join all rows in all joined tables are included, whether they are matched or not.

View All Answers

Question - 14:

What is right outer join in SQL server joins?

Ans:

In Right Outer Join all rows in the second-named table i.e. "right" table, which appears rightmost in the JOIN clause are included. Unmatched rows in the left table are not included.

View All Answers

Question - 15:

What are the properties of sub-query?

Ans:

A subquery is a SELECT statement that is nested within another T-SQL statement. A subquery SELECT statement if executed independently of the T-SQL statement, in which it is nested, will return a resultset. Meaning a subquery SELECT statement can standalone and is not depended on the statement in which it is nested. A subquery SELECT statement can return any number of values, and can be found in, the column list of a SELECT statement, a FROM, GROUP BY, HAVING, and/or ORDER BY clauses of a T-SQL statement. A Subquery can also be used as a parameter to a function call. Basically a subquery can be used anywhere an expression can be used.

View All Answers

Question - 16:

Explain sub-query?

Ans:

Sub-queries are often referred to as sub-selects, as they allow a select statement to be executed arbitrarily within the body of another SQL statement. A sub-query is executed by enclosing it in a set of parentheses. Sub-queries are generally used to return a single row as an atomic value, though they may be used to compare values against multiple rows with the IN keyword.

View All Answers

Question - 17:

Can you please explain the difference between Function and Stored Procedure?

Ans:

UDF can be used in the SQL statements anywhere in the WHERE/HAVING/SELECT section where as Stored procedures cannot be. UDFs that return tables can be treated as another rowset. This can be used in JOINs with other tables. Inline UDF's can be thought of as views that take parameters and can be used in JOINs and other Rowset operations.

View All Answers

Question - 18:

Explain collation?

Ans:

Collation refers to a set of rules that determine how data is sorted and compared. Character data is sorted using rules that define the correct character sequence, with options for specifying case sensitivity, accent marks, kana character types and character width.

View All Answers

Question - 19:

We need to perform what steps in the following order to work with a cursor?

Ans:

* Declare cursor
* Open cursor
* Fetch row from the cursor
* Process fetched row
* Close cursor
* Deallocate cursor

View All Answers

Question - 20:

Explain Cursor as data base object?

Ans:

Cursor is a database object used by applications to manipulate data in a set on a row-by- row basis, instead of the typical SQL commands that operate on all the rows in the set at one time.

View All Answers

Question - 21:

Explain Linked Server in SQL?

Ans:

Linked Servers is a concept in SQL Server by which we can add other SQL Server to a Group and query both the SQL Server dbs using T-SQL Statements. With a linked server, you can create very clean, easy to follow, SQL statements that allow remote data to be retrieved, joined and combined with local data. Stored Procedure sp_addlinkedserver, sp_addlinkedsrvlogin will be used add new Linked Server.

View All Answers

Question - 22:

Explain index in SQL server?

Ans:

An index is a physical structure containing pointers to the data. Indices are created in an existing table to locate rows more quickly and efficiently. It is possible to create an index on one or more columns of a table, and each index is given a name. The users cannot see the indexes; they are just used to speed up queries. Effective indexes are one of the best ways to improve performance in a database application. A table scan happens when there is no index available to help a query. In a table scan SQL Server examines every row in the table to satisfy the query results. Table scans are sometimes unavoidable, but on large tables, scans have a terrific impact

on performance.

Question - 23:

Explain view in SQL server?

Ans:

A simple view can be thought of as a subset of a table. It can be used for retrieving data, as well as updating or deleting rows. Rows updated or deleted in the view are updated or deleted in the table the view was created with. It should also be noted that as data in the original table changes, so does data in the view, as views are the way to look at part of the original table. The results of using a view are not permanently stored in the database. The data accessed through a view is actually constructed using standard T-SQL select command and can come from one to many different base tables or even other views.

Question - 24:

Explain Nested Trigger in SQL?

Ans:

A trigger can also contain INSERT, UPDATE and DELETE logic within itself, so when the trigger is fired because of data modification it can also cause another data modification, thereby firing another trigger. A trigger that contains data modification logic within itself is called a nested trigger.

Question - 25:

Explain triggers in SQL?

Ans:

A trigger is a SQL procedure that initiates an action when an event (INSERT, DELETE or UPDATE) occurs. Triggers are stored in and managed by the DBMS. Triggers are used to maintain the referential integrity of data by changing the data in a systematic fashion. A trigger cannot be called or executed; DBMS automatically fires the trigger as a result of a data modification to the associated table. Triggers can be viewed as similar to stored procedures in that both consist of procedural logic that is stored at the database level. Stored procedures, however, are not event-drive and are not attached to a specific table as triggers are. Stored procedures are explicitly executed by invoking a CALL to the procedure while triggers are implicitly executed. In addition, triggers can also execute stored procedures.

Question - 26:

Explain Stored Procedure?

Ans:

A stored procedure is a named group of SQL statements that have been previously created and stored in the server database. Stored procedures accept input parameters so that a single procedure can be used over the network by several clients using different input data. And when the procedure is modified, all clients automatically get the new version. Stored procedures reduce network traffic and improve performance. Stored procedures can be used to help ensure the integrity of the database.
* e.g. sp_helpdb, sp_renamedb, sp_depends etc.

Question - 27:

What is DKNF in normalization form?

Ans:

Domain-Key Normal Form A model free from all modification anomalies is said to be in DKNF.

Question - 28:

What is ONF in normalization form?

Ans:

Optimal Normal Form A model limited to only simple (elemental) facts, as expressed in Object Role Model notation.

Question - 29:

What is 5NF in normalization form?

Ans:

Isolate Semantically Related Multiple Relationships There may be practical constrains on information that justify separating logically related many-to-many relationships.

Question - 30:

What is 4NF in normalization form?

Ans:

Isolate Independent Multiple Relationships No table may contain two or more 1:n or n:m relationships that are not directly related.

Question - 31:

What is BCNF normalization form?

Ans:

Boyce-Codd Normal Form If there are non-trivial dependencies between candidate key attributes, separate them out into distinct tables.

Question - 32:

What is 3NF normalization form?

Ans:

Eliminate Columns Not Dependent On Key If attributes do not contribute to a description of the key, remove them to a separate table. All attributes must be directly dependent on the primary key.

Question - 33:

What is 2NF normalization form?

Ans:

Eliminate Redundant Data If an attribute depends on only part of a multi-valued key, remove it to a separate table.

Question - 34:

What is 1NF normalization form?

Ans:

Eliminate Repeating Groups Make a separate table for each set of related attributes, and give each table a primary key. Each field contains at most one value from its attribute domain.

Question - 35:

What is De-normalization?

Ans:

De-normalization is the process of attempting to optimize the performance of a database by adding redundant data. It is sometimes necessary because current DBMSs implement the relational model poorly. A true relational DBMS would allow for a fully normalized database at the logical level, while providing physical storage of data that is tuned for high performance. De-normalization is a technique to move from higher to lower normal forms of database modeling in order to speed up database access.

Question - 36:

Explain the properties of the Relational tables?

Ans:

* Values are atomic.
* Column values are of the same kind.
* Each row is unique.
* The sequence of columns is insignificant.
* The sequence of rows is insignificant.
* Each column must have a unique name.

Question - 37:

Explain RDBMS?

Ans:

Relational Data Base Management Systems (RDBMS) are database management systems that maintain data records and indices in tables. Relationships may be created and maintained across and among the data and tables. In a relational database, relationships between data items are expressed by means of tables. Inter-dependencies among these tables are expressed by data values rather than by pointers. This allows a high degree of data independence. An RDBMS has the capability to recombine the data items from different files, providing powerful tools for data usage.

Question - 38:

What is FULL OUTER JOIN?

Ans:

A full outer join will combine results of both left and right outer join. Hence the records from both tables will be displayed with a NULL for missing matches from either of the tables.
Example: To display employees who have a bonus and to display bonus even if he is not an employee.
Select * From employee FULL OUTER JOIN bonus ON employee.bonusID=bonus.bonusID

Question - 39:

What is RIGHT OUTER JOIN?

Ans:

A right outer join or a right join returns results from the table mentioned on the right of the join irrespective of whether it finds matches or not. If the ON clause matches 0 records from table on the left, it will still return a row in the result-but with NULL in each column.
Example: To display Bonus irrespective of whether they are an employee or not.
Select * From employee RIGHT OUTER JOIN bonus ON employee.bonusID=bonus.bonusID

Question - 40:

What is LEFT OUTER JOIN?

Ans:

A left outer join or a left join returns results from the table mentioned on the left of the join irrespective of whether it finds matches or not. If the ON clause matches 0 records from table on the right, it will still return a row in the result-but with NULL in each column.
Example: To display employees irrespective of whether they have got bonus.
Select * From employee LEFT OUTER JOIN bonus ON employee.bonusID=bonus.bonusID

Question - 41:

What is OUTER JOIN?

Ans:

In An outer join, rows are returned even when there are no matches through the JOIN criteria on the second table.

Question - 42:

What is inner join? Explain with an example?

Ans:

INNER JOIN: Inner join returns rows when there is at least one match in both tables.
Syntax:
SELECT column_name(s) FROM table_name1 INNER JOIN table_name2 ON table_name1.column_name=table_name2.column_name
Example: To display records of an employee who got an appraisal.
SELECT employee.firstname, appraisal.amount FROM employee INNER JOIN appraisal ON employee.id = appraisal.employee.id;

Question - 43:

What is Hash join?

Ans:

A hash join algorithm can only produce equi-joins. The database system pre-forms access to the tables concerned by building hash tables on the join-attributes.

Question - 44:

What is Merge join?

Ans:

Merge join If both join relations come in order, sorted by the join attribute(s), the system can perform the join trivially, thus: It can consider the current group of tuples from the inner relation which consists of a set of contiguous tuples in the inner relation with the same value in the join attribute. For each matching tuple in the current inner group, add a tuple to the join result. Once the inner group has been exhausted, advance both the inner and outer scans to the next group.

Question - 45:

Explain Nested Join?

Ans:

In nested joins, for each tuple in the outer join relation, the system scans the entire inner-join relation and appends any tuples that match the join-condition to the result set.

Question - 46:

What is a self join in SQL Server?

Ans:

Two instances of the same table will be joined in the query.

Question - 47:

What are different Types of Join?

Ans:

A join is typically used to combine results of two tables. A Join in SQL can be:-
Inner joins
Outer Joins
Left outer joins
Right outer joins
Full outer joins
Inner join

An inner join looks for matching records taken from one table from another.
A left outer join limits results to the table in left of JOIN.
A right outer join limits results to the table in right of JOIN.
Full outer joins are the combination of left and right outer joins

View All Answers

Question - 48:

Can you explain different types of joins?

Ans:

Types of joins: INNER JOINs, OUTER JOINs, CROSS JOINs.
OUTER JOINs are further classified as LEFT OUTER JOINS, RIGHT OUTER JOINS and FULL OUTER JOINS.

View All Answers

Question - 49:

Explain a join?

Ans:

Joins are used in queries to explain how different tables are related.
Joins also let you select data from a table depending upon data from another table.

View All Answers

MS SQL Server Most Popular & Related Interview Guides

1 : **SQL Server Database Administrator Interview Questions and Answers.**

2 : **SQL Server Service Broker Interview Questions and Answers.**

3 : **SQL Server Locks Interview Questions and Answers.**

4 : **SQL Server Constraints Interview Questions and Answers.**

5 : **MS SQL Data Mining Interview Questions and Answers.**

6 : **SSIS Interview Questions and Answers.**

7 : **SQL Server Architecture Interview Questions and Answers.**

8 : **MS SQL Server Indexes Interview Questions and Answers.**

9 : **MS SQL Server 2008 Interview Questions and Answers.**

10 : **SQL Database Concepts 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