Ruby 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 Ruby 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 Ruby 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 Ruby category. To ensure quality, each submission is checked by our team, before it becomes live. This Ruby 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

Ruby Interview Questions And Answers Guide.

Question - 1:

Explain the difference between nil and false in ruby?

Ans:

False is a boolean datatype
Nil is not a data type

Question - 2:

Explain the use of global variable $ in Ruby?

Ans:

If you declare one variable as global we can access any where, where as class variable visibility only in the class

Question - 3:

What is the use of load and require in ruby?

Ans:

Reuire() loads and processes the Ruby code from a separate file, including whatever classes, modules, methods, and constants are in that file into the current scope.
Load() performs the inclusion operation once, it reprocesses the code every time load is called.

Question - 4:

What is the use of super in Ruby Rails?

Ans:

* Ruby uses the super keyword to call the superclass implementation of the current method.
* Within the body of a method, calls to super acts just like a call to that original method.
* The search for a method body starts in the superclass of the object that was found to contain the original method.
    def url=(addr)
    super (addr.blank? || addr.starts_with?('http')) ? addr : http://#{addr}
    end

Question - 5:

Interpolation is a very important process in Ruby, comment.

Ans:

* Interpolation is the process of inserting a string into a literal.
* You can interpolate a string into a literal by placing a Hash (#) within {} open and close brackets.
* This refers to a new name to by referring to the copy of the original method.

Question - 6:

Describe the environment variables present in Ruby.

Ans:

RUBYOPT
Additional command-line options to Ruby; examined after real command-line options are parsed ($SAFE must be 0).
RUBYLIB
Additional search path for Ruby programs ($SAFE must be 0).
RUBYPATH

With -S option, search path for Ruby programs (defaults to PATH).
RUBYSHELL
Shell to use when spawning a process; if not set, will also check SHELL or COMSPEC. DLN_LIBRARY_PATH Search path for dynamically loaded modules.
RUBYLIB_PREFIX
(Windows only) Mangle the RUBYLIB search path by adding this prefix to each component.

View All Answers

Question - 7:

Explain the concepts and capabilities of garbage collection feature of Ruby.

Ans:

* Ruby is an object oriented language and every object oriented language tends to allocate many objects during execution of the program.
* Ruby deletes unallocated and unused objects automatically.
* This feature can be controlled by applying proper syntax and program through ruby.

View All Answers

Question - 8:

Describe class libraries in Ruby.

Ans:

* The Ruby standard library extends the foundation of the Ruby built-in library with classes and abstractions for a variety of programming needs, including network programming, operating system services, threads, and more.
* These classes provide flexible capabilities at a high level of abstraction, giving you the ability to create powerful Ruby scripts useful in a variety of problem domains.

View All Answers

Question - 9:

What is Rails?

Ans:

* Ruby on Rails is an open source web application framework for the Ruby programming language.
* It is used with an Agile development methodology that is used by web developers for rapid development.

View All Answers

Question - 10:

Explain about the programming language ruby?

Ans:

Ruby is the brain child of a Japanese programmer Matz. He created Ruby. It is a cross platform object oriented language. It helps you in knowing what your code does in your application. With legacy code it gives you the power of administration and organization tasks. Being open source, it did go into great lengths of development.

View All Answers

Question - 11:

Explain about class libraries in ruby?

Ans:

Ruby has a strong set of class libraries and it covers from a variety of domains such as thread programming, domains and data types. Also ruby is a new language and it also has additional libraries coming every day. Many of the new languages which do exist have huge libraries because of their age.

View All Answers

Question - 12:

Explain about portability?

Ans:

Ruby language can be ported to many platforms. Ruby programs can be ported to many platforms without any modification to the source code. This feature made the language very useful and highly used by many programmers worldwide. Some of the platforms used are DOS, UNIX, WINDOWS, etc.

View All Answers

Question - 13:

Explain about garbage collection feature of ruby?

Ans:

Ruby is an object oriented language and every object oriented language tends to allocate many objects during execution of the program. Ruby deletes unallocated and unused objects automatically. This feature can be controlled by applying proper syntax and program through ruby.

View All Answers

Question - 14:

Explain about the command line options?

Ans:

Ruby`s language is executed from the command line like most of the scripting languages. Programming and behavior language environment can be controlled from

the interpreter itself. Some of the commands which are used are as follows -d, -h, -e prog, -v, -T, -r lib, etc.

View All Answers

Question - 15:

Explain about environment variables present in ruby?

Ans:

Following are some of the environment variables used to control the behavior programming of ruby. While programming ENV object lists some of the current variables. RUBYLIB path searches for libraries. Make sure that you separate each path with colons. RUBYOPT passes command line options to Ruby interpreter. There are many more which can be obtained by searching the huge pool of library.

View All Answers

Question - 16:

Explain about operators in Ruby?

Ans:

Like all the modern languages Ruby supports many different operators to its credit. Most of the operators in Ruby are in fact methods due to the object oriented nature of the language. This feature gives Ruby an edge over its competitors. Semantics of these operators can be changed due to the object oriented nature of the language.

View All Answers

Question - 17:

Explain about the defined operator?

Ans:

Define operator defines whether a passed expression is defined or not. If the expression is defined it returns the description string or null if the expression is not defined. If a variable is defined it gets initialized. If method_call is defined as true then method also gets defined. This is also the same case with super and yield.

View All Answers

Question - 18:

Explain about methods?

Ans:

Methods in ruby basically perform two functions, named operation and the code present in the class which does a specific function. In Ruby all your algorithms live in methods which inturn is present on objects. Ruby does not have any provision for functions. Code present in Ruby is always a method of some object. Behind the scenes ruby gives you the flexibility to work with methods as functions if you are considering working with other languages.

View All Answers

Question - 19:

Explain about normal method class?

Ans:

This function calls a method and it can take any number of arguments and expr. Make sure that you put an asterisk or an ampersand before the expression. Last expr argument can be declared with a hash without any braces. If you want to increase the size of the array value then make sure that you put an asterisk before expression. "::" can be used to separate the class from methods.

View All Answers

Question - 20:

Explain about ruby names?

Ans:

Classes, variables, methods, constants and modules can be referred by ruby names. When you want to distinguish between various names you can specify that by the first character of the name. Some of the names are used as reserve words which should not be used for any other purpose. A name can be lowercase letter, upper case letter, number, or an underscore, make sure that you follow the name by name characters.

View All Answers

Question - 21:

Explain about variables?

Ans:

There are four different types of variables they are local, instance, global, and class. Variables can be used in the program without any declaration and they can contain data of any type. A local variable contains lower case characters followed by name characters. Instance variable starts with a @ sign followed by name characters.

View All Answers

Question - 22:

Explain about Class variable and global variable?

Ans:

A class variable starts with an @@ sign which is immediately followed by upper or lower case letter. You can also put some name characters after the letters which stand to be a pure optional. A class variable can be shared among all the objects of a class. A single copy of a class variable exists for each and every given class.
To write a global variable you start the variable with a $ sign which should be followed by a name character. Ruby defines a number of global variables which also

include other punctuation characters such as $_ and $-k.

View All Answers

Question - 23:

How does ruby deal with extremely large numbers?

Ans:

Unlike other programming languages ruby deals with extremely large numbers it doesn't have any barriers. There is no limit on the extent of limit of number usage. Ruby performs this function with two different classes they are fixnum and bignum. Fixnum represents easily managed small numbers and bignum represents big numbers. Ruby entirely handles the functioning of these two classes which leaves a programmer to concentrate on his arithmetic operations.

View All Answers

Question - 24:

Explain about Float, Dig and Max?

Ans:

Float class is used whenever the function changes constantly. It acts as a sub class of numeric. They represent real characters by making use of the native architecture of the double precision floating point.
Max is used whenever there is a huge need of Float.
Dig is used whenever you want to represent a float in decimal digits.

View All Answers

Question - 25:

Explain about interpolation?

Ans:

Interpolation is a very important process in Ruby. Interpolation is the process of inserting a string into a literal. There is only one way in which you can interpolate a string into a literal by placing a Hash (#) within { } open and close brackets. This refers to a new name to by referring to the copy of the original method.

View All Answers

Question - 26:

Explain about ruby code blocks?

Ans:

Ruby code blocks form an important part of ruby and are very fun to use. With the help of this feature you can place your code between do-end and you can associate them with method invocations and you can get an impression that they are like parameters. They may appear near to a source of the code and adjacent to a method call. The code is not executed during the program execution but it is executed when the context of its appearance is met or when it enters a method.

View All Answers

Computer Programming Most Popular & Related Interview Guides

1 : **Python Interview Questions and Answers.**

2 : **OOP Interview Questions and Answers.**

3 : **Software engineering Interview Questions and Answers.**

4 : **PHP Interview Questions and Answers.**

5 : **VBA (Visual Basic for Applications) Interview Questions and Answers.**

6 : **Visual Basic (VB) Interview Questions and Answers.**

7 : **Node.js Interview Questions and Answers.**

8 : **CMMI Interview Questions and Answers.**

9 : **Microsoft Foundation Class (MFC) Interview Questions and Answers.**

10 : **Lotus Notes 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