Java-Success.com

Industrial strength Java Career Companion

search here …   Go

Home | **Java FAQs** | **600+ Java Q&As** | **Career** | **Tutorials** | **Member** | **Why?**

Can u Debug? | Java 8 ready? | Top X | Productivity Tools | Judging Experience?

♥ Q1 – Q5 Groovy basics interview Q&A

Posted on July 20, 2015 by Arulkumaran Kumaraswamipillai

Q1. What are the key features of Groovy?
A1. Groovy is a powerful, optionally typed and dynamic language aimed at rapid development.

Groovy = Java – verbosity
+ Dynamic Typing (mostly)
+ Closures (instead of Anonymous classes) -> Reusuable & assignable piece of code
+ Domain Specific Language (DSL)
+ Builders
+ Meta Programming
+ Groovy Development Kit
+ Operators can be overloaded

Q2. What are the differences between a statically typed language like Java and a dynamically typed language like Groovy?
A2. **1)** Statically typed languages perform type checking at

600+ Full Stack Java Interview Questions Answered ♥Free ♦FAQs [Mouse Hover for full title text]

open all | close all

compile time, whereas dynamically-typed languages perform type checking at **runtime**.

2) Statically-typed languages require you to declare the data types of your variables before you use them, while dynamically-typed languages do not.

```
1  // Java example
2  int num = 5; // int at compile-time
3  double num2 = 23.0; //double at compile-time
```

```
1  // Groovy example
2  def num = 5;
3  def num2 = 5.0;
4
5  def result = 0;
6
7  println "num=$num and num2=$num2" //at runtime
8
9  result = num2/num //double divided by int
10
11 println "result=$result"  //result is double at
12
13 def result2 = num/num2 //int divided by double
14
15 println "result2=$result2"  //result is int at
```

Here is another Groovy example with "handleAnimal(anyAnimal)" method that takes any objects with the "sound()" method. If you pass an object that does not have a sound() method, it will throw aqn error like "groovy.lang.MissingMethodException: No signature of method: java.lang.Object.sound() is applicable for argument types: ……"

```
1  //define a Dog class
2  class Dog {
3
4     def sound(){
5        println "wow wow"
6     }
7  }
8
9  //Define a Cat class
10 class Cat {
11
12    def sound(){
13       println "Meow meow"
14    }
15 }
```

```
16
17 //dynamically takes at run time any object type
18 def handleAnimal(anyAnimal){
19     anyAnimal.sound();
20 }
21
22 def dog = new Dog();
23 def cat = new Cat();
24
25 handleAnimal(dog);   // prints "Wow wow"
26 handleAnimal(cat);   // prints "Meow meow"
```

Q3. Does dynamically typed means Groovy is a weakly typed language like JavaScript?

Q3. Dynamically typed languages such as Groovy don't perform type checking at code editing time or compile time. But, if you mistreat an object as a wrong type, you'll get a **runtime** exception.

Q4. Can you explain closure and currying in Groovy with a simple code snippet?

Q4. A closure in Groovy is an open, anonymous, block of code that can take arguments, return a value and be assigned to a variable. A closure may reference variables declared in its surrounding scope. A closure is like the **lambda expressions** in Java 8, but the difference being that the Groovy closure contain free variables which are defined outside of its surrounding scope.

When you **curry( )** a closure, you're asking the parameters to be **prebound**. This is known as the **curried closure**. Here is a very simple example

```
1 def sumThreeNumbers = {num1, num2, num3 -> print
2
3 sumThreeNumbers(1,2,3) //invoke the closure 1+2+3
4
5 def num1Bound = sumThreeNumbers.curry(1) // 1 is
6 num1Bound(4,5); // 1 + (4+5) = 10
7
8 def num1And2Bound = num1Bound.curry(3); // num1 &
9 num1And2Bound(7); // (1+3) + 7 = 11
```

Note: Closure is one of the backbones of Groovy programming. You must know it well. 20 Groovy closure interview questions & answers with examples.

Q5. What is a GDK or Groovy Development Kit?

Q5. Groovy enhances the JDK with convenience methods, of which quite a few make extensive use of closures. GDK has its packages like groovy.lang, groovy.sql, etc on top of the JDK.

GDK provides additional methods to the JDK to make it more groovy. The Groovy JDK is a collection of extensions to classes available in the standard Java SDK such as Object. For example, the standard Java JDK provides a "File.delete()" method whereas the GDK provides an extension to the standard File class by adding a method "**deleteDir()**". deleteDir() behaves as recursive delete (i.e "rm -rf" in unix) by deleting a directory even if it contains other files and directories.

The GDK's extended String class has numerous convenience methods such as "String.count(String)" method that counts the number of occurrences of the String passed as a parameter. The "String.contains(String)" is another handy method.

One of the noticeable additions to **the primitive type wrappers like Character, Integer**, and so on, is the overloaded operator mapping methods like plus() for "+" operator and next() for the operator ++. These are very useful in creating DSLs. The Number superclass from which the Integer, Double, etc extend has been enhanced with iterating methods like times(..), step(..), upto(..) and downto(..). The times(…) and other enhanced methods take closures as parameters. For example,

```
1 public void times(Closure closure)
```

Groovy adds a number of methods at the **java.lang.Object level**, like dump(). The enhanced Object also supports methods for metaprogramming to dynamically access properties and invoke methods. There are other methods in the enhanced Object class like

1) boolean any(Closure closure) –&gt Iterates over the contents of an object or collection, and checks whether a predicate is valid for at least one element.

2) Object asType(Class clazz) –> Converts the given array to either a List, Set, or SortedSet.

3) List collect(Closure transform) –> Iterates through this aggregate Object transforming each item into a new value using the transform closure, returning a list of transformed values.

e.g.

```
1 def myList = [1, 2, 3, 4];
2 def resultList = myList.collect({item -> item * 2
3 println resultList //[2, 4, 6, 8]
```

4) Object find(Closure closure) –> Finds the first value matching the closure condition

5) Collection findAll(Closure closure) –> Finds all items matching the closure condition.

```
1 def myList = [1, 2, 3, 4];
2 def resultList = myList.findAll({item -> item % 2
3 println resultList //[2, 4]
```

6) Object inject(Object initialValue, Closure closure) –> Iterates through the given Object, passing in the initial value to the closure along with the first item.

```
1 (1..5).inject(0, {sum,number -> sum = sum + numbe
```

Check the API documentation for more methods.

Popular Member Posts

01: ♦ 15+ Hibernate basics Q1 – Q7 interview
questions & answers

1,769 views

♦ 11 Spring boot interview questions & answers

934 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview
Questions & Answers

831 views

01: ♦ 15 Beginner level Java multi-threading interview
Q&A

534 views

001A: ♦ 7+ Java integration styles & patterns
interview questions & answers

430 views

18 Java scenarios based interview Questions and
Answers

395 views

♦ 7 Java debugging interview questions & answers

323 views

01: ♦ 15 Ice breaker questions asked 90% of the time
in Java job interviews with hints

303 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview
Questions and Answers

300 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions
& answers

297 views

Bio	Latest Posts

About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java
developer in 3 yrs. Contracting since
2003, and attended 150+ Java job
interviews, and often got 4 - 7 job offers

to choose from. It pays to prepare. Published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. join my LinkedIn Group to get notified of special offers. User & expert **reviews**.

‹   ♦ 20 Groovy closure interview questions & answers with examples

♦ 9 Groovy meta programming interview questions & answers   ›

Posted in Groovy Interview Q&A