How to create a suite in "TestNG"

Create to testng class like-

First Class is "Test1.java"

```
import org.testng.Assert;
import org.testng.annotations.Test;

public class Test1
{
    String message = "Shailesh";
    MessageUtil messageUtil = new MessageUtil(message);

    @Test
    public void testPrintMessage()
    {
        System.out.println("Inside testPrintMessage()");
        Assert.assertEquals(message, messageUtil.printMessage());
    }
}
```

Second Class is "Test2.java"

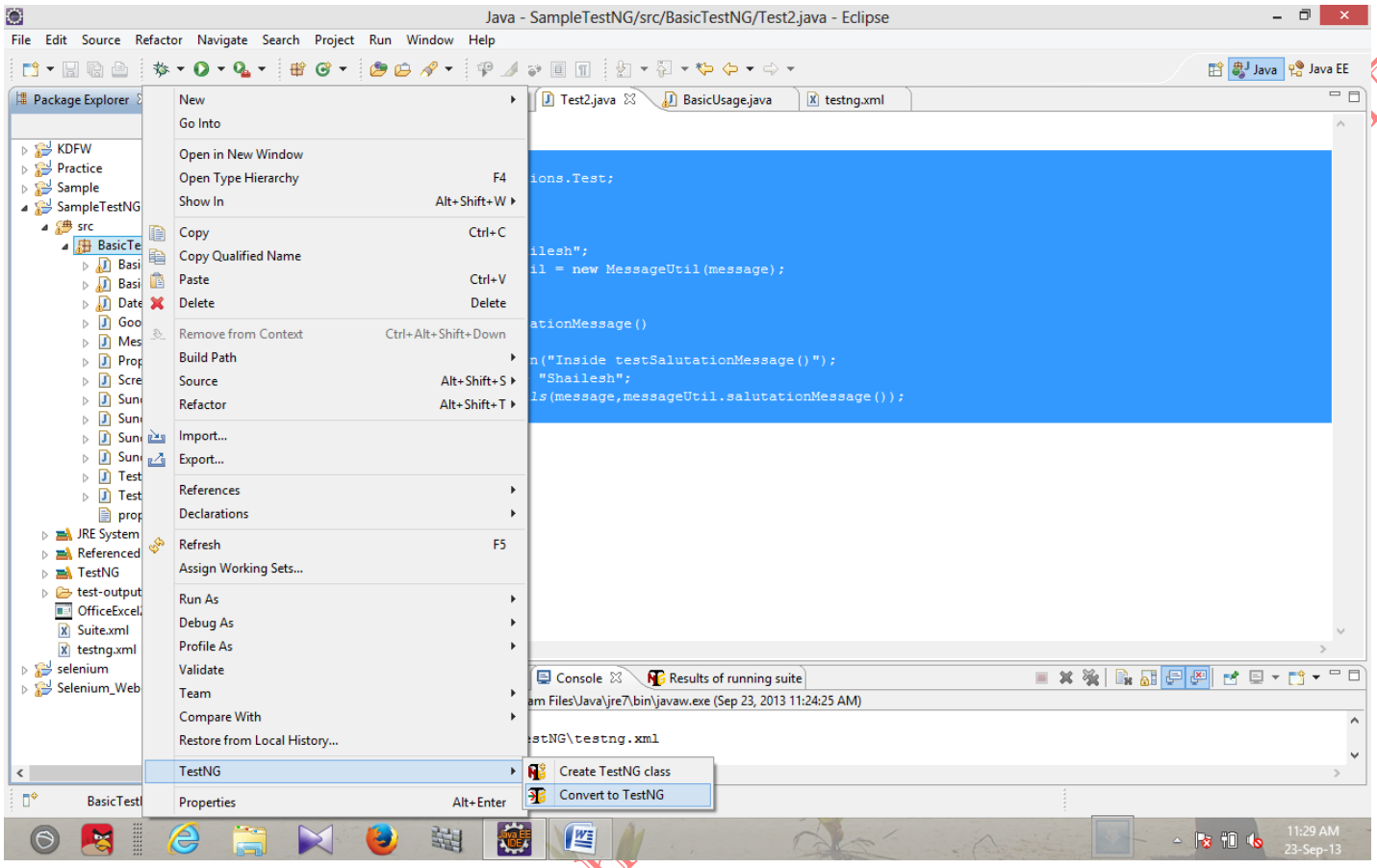
```
import org.testng.Assert;
import org.testng.annotations.Test;

public class Test2
{
    String message = "Shailesh";
    MessageUtil messageUtil = new MessageUtil(message);

    @Test
    public void testSalutationMessage()
    {
        System.out.println("Inside testSalutationMessage()");
        message = "Hi!" + "Shailesh";
        Assert.assertEquals(message,messageUtil.salutationMessage());
    }
}
```

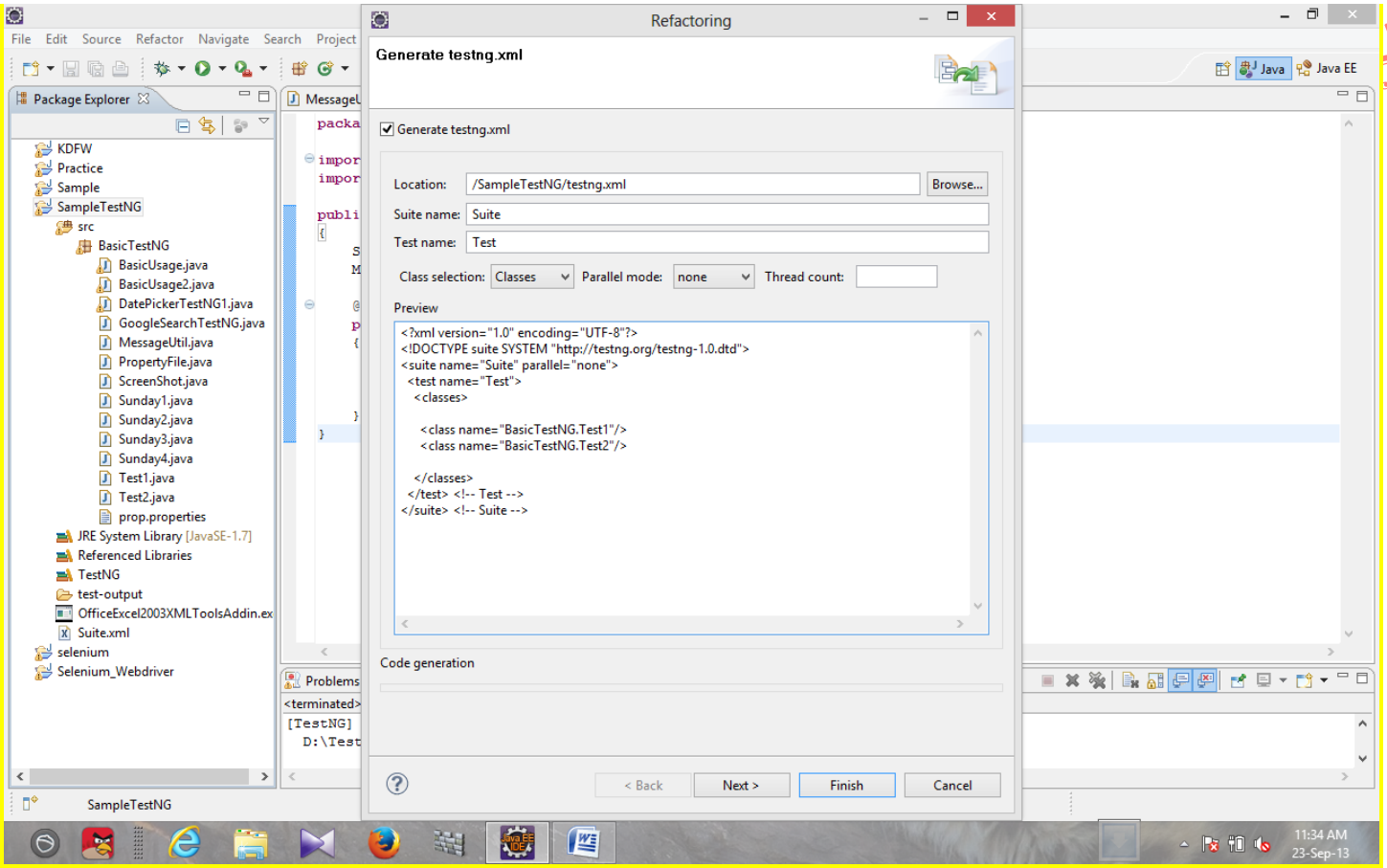
Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404067

Now right click on your project and select TestNG then Click on "Convert to TestNG"

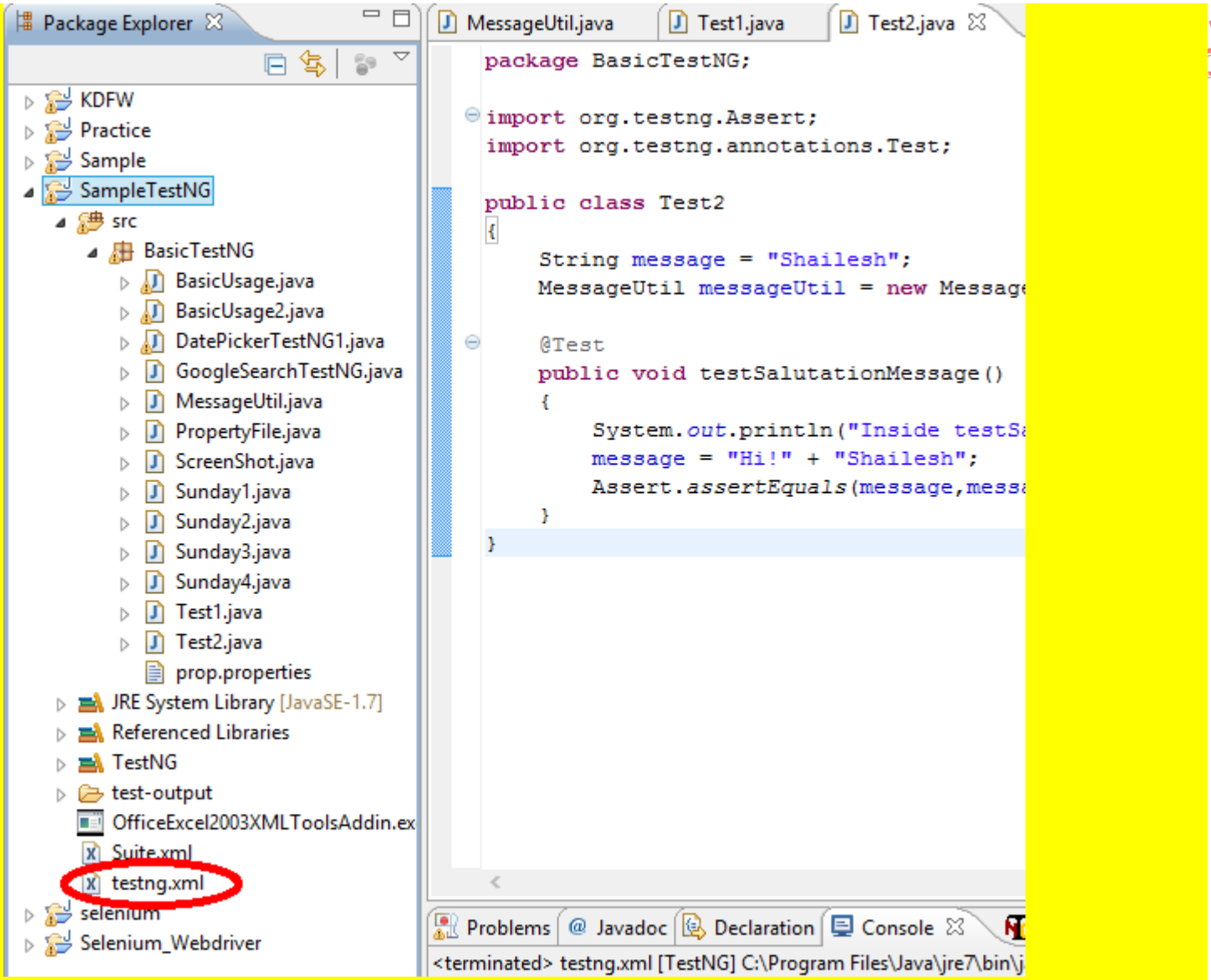


Shailesh Kumar - (shailesh

Select only those classes which you want to include in “Suite” and then click on “finish”

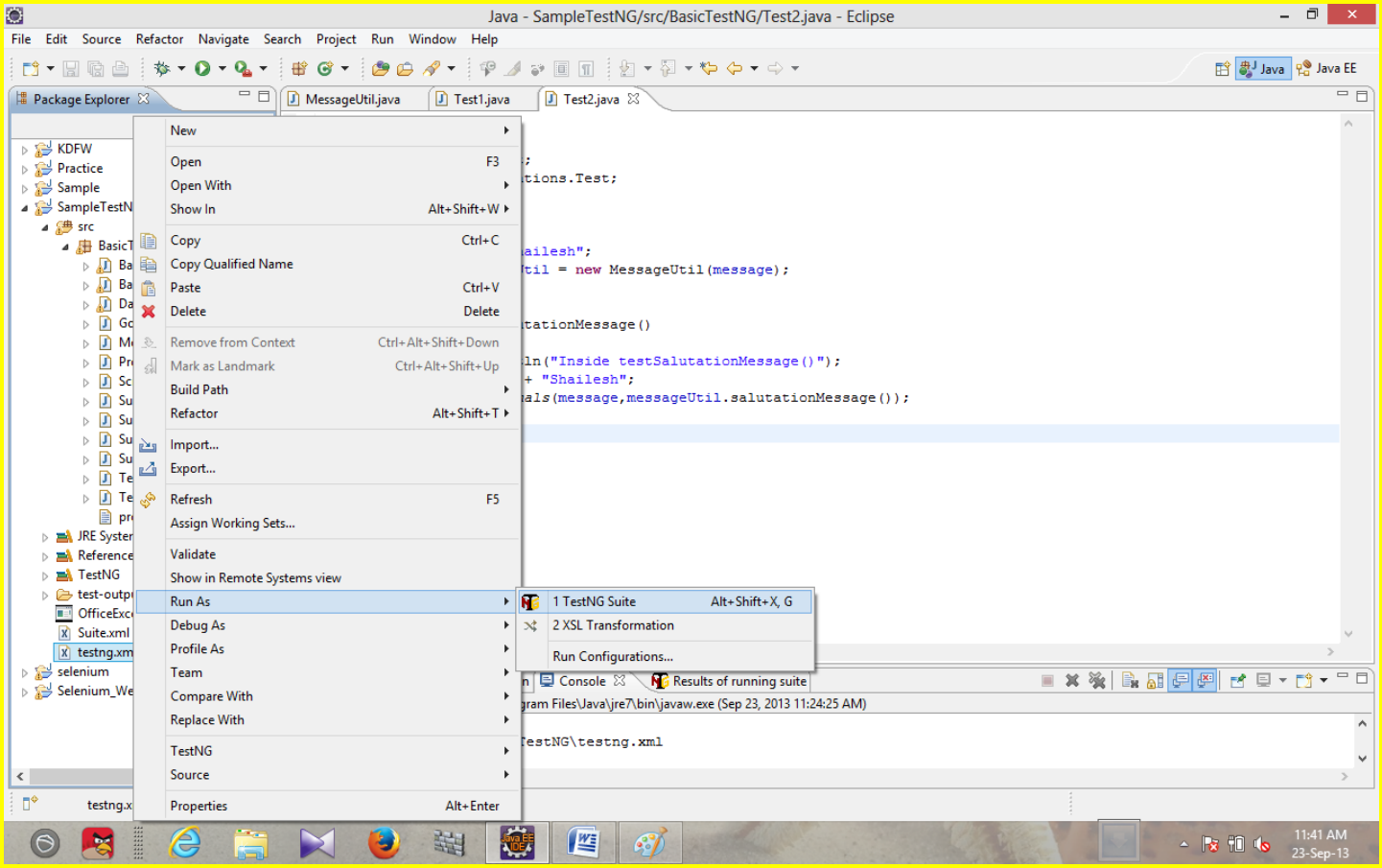


After click on finish you will get a new “testing.xml” under your project like this-



Shailesh Kum

Now right click on “testing.xml” and “Run AS“ as “TestNG Suite ”



After doing this you can see the output on “console window ” as well as “TestNG Window”

TestNG Interview Questions

Q: What is TestNG?

A: TestNG is an open source automated testing framework; where NG of TestNG means Next Generation. TestNG is similar to JUnit (especially JUnit 4), but its not a JUnit extension. Its inspired by JUnit. It is designed to be better than JUnit, especially when testing integrated classes.

Q: What are the features of TestNG?

A: Features of TestNG are:

- Annotations.
- TestNG uses more Java and OO features.
- Supports testing integrated classes (e.g., by default, no need to create a new test class instance for every test method).
- Separate compile-time test code from run-time configuration/data info.
- Flexible runtime configuration.
- Introduces 'test groups'. Once you have compiled your tests, you can just ask TestNG to run all the "front-end" tests, or "fast", "slow", "database", etc...
- Supports Dependent test methods, parallel testing, load testing, partial failure.
- Flexible plug-in API.
- Support for multi threaded testing.

Q: What are the advantages of TestNG over Junit?

A: Advantages of TestNG over Junit are:

- In Junit we have to declare @BeforeClass and @AfterClass which is a constraint where as in TestNG there is no constraint like this.
- Additional Levels of setUp/tearDown level are available in TestNG like @Before/AfterSuite,@Before/AfterTest and @Before/AfterGroup
- No Need to extend any class in TestNG.
- There is no method name constraint in TestNG as in Junit. You can give any name to the test methods in TestNG.
- In TestNG we can tell the test that one method is dependent on another method where as in Junit this is not possible. In Junit each test is independent of another test.
- Grouping of testcases is available in TestNG where as the same is not available in Junit.
- Execution can be done based on Groups. For ex. If you have defined many cases and segregated them by defining 2 groups as Sanity and Regression. Then if you only want to execute the "Sanity" cases then just tell TestNG to execute the "Sanity" and TestNG will automatically execute the cases belonging to the "Sanity" group

Q: What are the basic steps involved in writing TestNG tests?

A: Writing a test in TestNG basically involves following steps:

- Write the business logic of your test and insert TestNG annotations in your code.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404063157

- Add the information about your test (e.g. the class name, the groups you wish to run, etc...) in a testng.xml file or in build.xml.
- Run TestNG.

Q: Give examples of some of the annotations supported by TestNG.

A: TestNG supports the following annotations:

@BeforeSuite, @AfterSuite, @BeforeClass, @AfterClass, @BeforeTest, @AfterTest, @BeforeGroups, @AfterGroups, @BeforeMethod, @AfterMethod, @DataProvider, @Factory, @Listeners, @Parameters, @Test.

Q: What are the benefits of using annotations?

A: Following are some of the benefits of using annotations:

- TestNG identifies the methods it is interested in by looking up annotations. Hence, method names are not restricted to any pattern or format.
- We can pass additional parameters to annotations.
- Annotations are strongly typed, so the compiler will flag any mistakes right away.
- Test classes no longer need to extend anything (such as TestCase, for JUnit 3).

Q: What are the different ways in which TestNG can be invoked?

A: You can invoke TestNG in several different ways:

- Using Eclipse
- With ant
- From the command line
- Using IntelliJ's IDEA

Q: Give a an example to invoke TestNG from command line.

A: Assuming that you have TestNG in your class path, the simplest way to invoke TestNG is as follows:

```
java org.testng.TestNG testng1.xml [testng2.xml testng3.xml ...]
```

Q: What is testng.xml file used for?

A: File *testng.xml* captures your entire testing in XML. This file makes it easy to describe all your test suites and their parameters in one file, which you can check in your code repository or e-mail to coworkers. It also makes it easy to extract subsets of your tests or split several runtime configurations (e.g., testng-database.xml would run only tests that exercise your database).

Q: What is test suite?

A: A **Test suite** is a collection of test cases that are intended to test a behavior or set of behaviors of software program. In TestNG, we cannot define a suite in testing source code, but it is represented by one XML file as

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

suite is the feature of execution. This also allows flexible configuration of the *tests* to be run. A suite can contain one or more tests and is defined by the <suite> tag.

<suite> is a root tag of your testng.xml. It describes a test suite, which in turn is made of several <test> sections.

Q: How can you disable a test in TestNG?

A: Annotation *@Test(enabled = false)* helps to disable the test case which you want to ignore.

Q: What is group test?

A: Group tests permits you dispatch methods into proper portions and preform sophisticated groupings of test methods. Not only can you declare those methods that belong to groups, but you can also specify groups that contain other groups. Then, TestNG can be invoked and asked to include a certain set of groups (or regular expressions) while excluding another set. This gives you maximum flexibility in how you partition your tests and doesn't require you to recompile anything if you want to run two different sets of tests back to back.

Q: How to you specify a group in testng.xml?

A: Groups are specified in your testng.xml file using the <groups> tag. It can be found either under the <test> or <suite> tag. Groups specified in the <suite> tag apply to all the <test> tags underneath.

Q: What is exception test?

A: TestNG provides a option of tracing the Exception handling of code. You can test whether a code throws desired exception or not. The **expectedExceptions** parameter is used along with @Test annotation. Now, let's see *@Test(expectedExceptions)* in action.

Q: What is dependency test?

A: Sometimes, you may need to invoke methods in a Test case in a particular order or you want to share some data and state between methods. This kind of dependency is supported by TestNG as it supports the declaration of explicit dependencies between test methods.

TestNG allows you to specify dependencies either with:

- Using attributes *dependsOnMethods* in @Test annotations OR
- Using attributes *dependsOnGroups* in @Test annotations.

Q: What is difference between *dependsOnGroups* and *dependsOnMethods*?

A: Following are the differences:

- On using groups, we are no longer exposed to refactoring problems. As long as we don’t modify the dependsOnGroups or groups attributes, our tests will keep running with the proper dependencies set up.
- Whenever a new method needs to be added in the dependency graph, all we need to do is put it in the right group and make sure it depends on the correct group. We don’t need to modify any other method.

Q: What is parametric testing?

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

A: In most cases, you'll come across a scenario where the business logic requires a hugely varying number of tests. *Parameterized tests* allow developers to run the same test over and over again using different values.

TestNG lets you pass parameters directly to your test methods in two different ways:

- With testng.xml
- With Data Providers

Q: How do you pass parameters with testng.xml?

A: We define the simple parameters in the testng.xml file and then reference those parameters in source files.

Q:What does it mean to pass parameters using dataproviders?

A: When you need to pass complex parameters or parameters that need to be created from Java (complex objects, objects read from a property file or a database, etc...), in such cases parameters can be passed using Dataproviders. A Data Provider is a method annotated with *@DataProvider*. This annotation has only one string attribute: its name. If the name is not supplied, the Data Provider's name automatically defaults to the method's name. A Data Provider returns an array of objects.

Q: How can you run the JUnit tests using TestNG?

A: Put JUnit library on the TestNG classpath, so it can find and use JUnit classes, change your test runner from JUnit to TestNG in Ant and then run TestNG in "mixed" mode. This way you can have all your tests in the same project, even in the same package, and start using TestNG. This approach also allows you to convert your existing JUnit tests to TestNG incrementally. Also define property *junit="true"* in the <test> tag of the testng.xml.

Q: What are different ways in which you can generate the reports of TestNg results?

A: There are two main ways to generate a report with TestNG:

- **Listeners :** For implementing a listener class, the class has to implement the *org.testng.ITestListener* interface. These classes are notified at runtime by TestNG when the test starts, finishes, fails, skips, or passes.
- **Reporters :** For implementing a reporting class, the class has to implement an *org.testng.IReporter* interface. These classes are called when the whole suite run ends. The object containing the information of the whole test run is passed to this class when called.

Kumar - (shailesh24x7@yahoo.com) - 9404065157

Java Interview Questions

1. **What is the most important feature of Java?**
   Java is a platform independent language.

2. **What do you mean by platform independence?**
   Platform independence means that we can write and compile the java code in one platform (eg Windows) and can execute the class in any other supported platform eg (Linux,Solaris,etc).

3. **Are JVM's platform independent?**
   JVM's are not platform independent. JVM's are platform specific run time implementation provided by the vendor.

4. **What is a JVM?**
   JVM is Java Virtual Machine which is a run time environment for the compiled java class files.

5. **What is the difference between a JDK and a JVM?**
   JDK is Java Development Kit which is for development purpose and it includes execution environment also. But JVM is purely a run time environment and hence you will not be able to compile your source files using a JVM.

6. **What is a pointer and does Java support pointers?**
   Pointer is a reference handle to a memory location. Improper handling of pointers leads to memory leaks and reliability issues hence Java doesn't support the usage of pointers.

7. **What is the base class of all classes?**
   java.lang.Object

8. **Does Java support multiple inheritance?**
   Java doesn't support multiple inheritance.

9. **Is Java a pure object oriented language?**
   Java uses primitive data types and hence is not a pure object oriented language.

10. **Are arrays primitive data types?**
    In Java, Arrays are objects.

11. **What is difference between Path and Classpath?**
    Path and Classpath are operating system level environment variales. Path is used define where the system can find the executables(.exe) files and classpath is used to specify the location .class files.

12. **What are local variables?**
    Local varaiables are those which are declared within a block of code like methods. Local variables should be initialised before accessing them.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

13. **What are instance variables?**
    Instance variables are those which are defined at the class level. Instance variables need not be initialized before using them as they are automatically initialized to their default values.

14. **How to define a constant variable in Java?**
    The variable should be declared as static and final. So only one copy of the variable exists for all instances of the class and the value can't be changed also.
    static final int PI = 2.14; is an example for constant.

15. **Should a main method be compulsorily declared in all java classes?**
    No not required. main method should be defined only if the source class is a java application.

16. **What is the return type of the main method?**
    Main method doesn't return anything hence declared void.

17. **Why is the main method declared static?**
    main method is called by the JVM even before the instantiation of the class hence it is declared as static.

18. **What is the arguement of main method?**
    main method accepts an array of String object as arguement.

19. **Can a main method be overloaded?**
    Yes. You can have any number of main methods with different method signature and implementation in the class.

20. **Can a main method be declared final?**
    Yes. Any inheriting class will not be able to have it's own default main method.

21. **Does the order of public and static declaration matter in main method?**
    No it doesn't matter but void should always come before main().

22. **Can a source file contain more than one Class declaration?**
    Yes a single source file can contain any number of Class declarations but only one of the class can be declared as public.

23. **What is a package?**
    Package is a collection of related classes and interfaces. package declaration should be first statement in a java class.

24. **Which package is imported by default?**
    java.lang package is imported by default even without a package declaration.

25. **Can a class declared as private be accessed outside it's package?**
    Not possible.

26. **Can a class be declared as protected?**
    A class can't be declared as protected. only methods can be declared as protected.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

27. **What is the access scope of a protected method?**
    A protected method can be accessed by the classes within the same package or by the subclasses of the class in any package.

28. **What is the purpose of declaring a variable as final?**
    A final variable's value can't be changed. final variables should be initialized before using them.

29. **What is the impact of declaring a method as final?**
    A method declared as final can't be overridden. A sub-class can't have the same method signature with a different implementation.

30. **I don't want my class to be inherited by any other class. What should i do?**
    You should declared your class as final. But you can't define your class as final, if it is an abstract class. A class declared as final can't be extended by any other class.

31. **Can you give few examples of final classes defined in Java API?**
    java.lang.String,java.lang.Math are final classes.

32. **How is final different from finally and finalize?**
    final is a modifier which can be applied to a class or a method or a variable. final class can't be inherited, final method can't be overridden and final variable can't be changed.
    finally is an exception handling code section which gets executed whether an exception is raised or not by the try block code segment.
    finalize() is a method of Object class which will be executed by the JVM just before garbage collecting object to give a final chance for resource releasing activity.

33. **Can a class be declared as static?**
    No a class cannot be defined as static. Only a method,a variable or a block of code can be declared as static.

34. **When will you define a method as static?**
    When a method needs to be accessed even before the creation of the object of the class then we should declare the method as static.

35. **What are the restriction imposed on a static method or a static block of code?**
    A static method should not refer to instance variables without creating an instance and cannot use "this" operator to refer the instance.

36. **I want to print "Hello" even before main is executed. How will you acheive that?**
    Print the statement inside a static block of code. Static blocks get executed when the class gets loaded into the memory and even before the creation of an object. Hence it will be executed before the main method. And it will be executed only once.

37. **What is the importance of static variable?**
    static variables are class level variables where all objects of the class refer to the same variable. If one object changes the value then the change gets reflected in all the objects.

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

38. **Can we declare a static variable inside a method?**
    Static varaibles are class level variables and they can't be declared inside a method. If declared, the class will not compile.

39. **What is an Abstract Class and what is it's purpose?**
    A Class which doesn't provide complete implementation is defined as an abstract class. Abstract classes enforce abstraction.

40. **Can a abstract class be declared final?**
    Not possible. An abstract class without being inherited is of no use and hence will result in compile time error.

41. **What is use of a abstract variable?**
    Variables can't be declared as abstract. only classes and methods can be declared as abstract.

42. **Can you create an object of an abstract class?**
    Not possible. Abstract classes can't be instantiated.

43. **Can a abstract class be defined without any abstract methods?**
    Yes it's possible. This is basically to avoid instance creation of the class.

44. **Class C implements Interface I containing method m1 and m2 declarations. Class C has provided implementation for method m2. Can i create an object of Class C?**
    No not possible. Class C should provide implementation for all the methods in the Interface I. Since Class C didn't provide implementation for m1 method, it has to be declared as abstract. Abstract classes can't be instantiated.

45. **Can a method inside a Interface be declared as final?**
    No not possible. Doing so will result in compilation error. public and abstract are the only applicable modifiers for method declaration in an interface.

46. **Can an Interface implement another Interface?**
    Intefaces doesn't provide implementation hence a interface cannot implement another interface.

47. **Can an Interface extend another Interface?**
    Yes an Interface can inherit another Interface, for that matter an Interface can extend more than one Interface.

48. **Can a Class extend more than one Class?**
    Not possible. A Class can extend only one class but can implement any number of Interfaces.

49. **Why is an Interface be able to extend more than one Interface but a Class can't extend more than one Class?**
    Basically Java doesn't allow multiple inheritance, so a Class is restricted to extend only one Class. But an Interface is a pure abstraction model and doesn't have inheritance hierarchy like classes(do remember that the base class of all classes is Object). So an Interface is allowed to extend more than one Interface.

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

50. **Can an Interface be final?**
Not possible. Doing so so will result in compilation error.

51. **Can a class be defined inside an Interface?**
Yes it's possible.

52. **Can an Interface be defined inside a class?**
Yes it's possible.

53. **What is a Marker Interface?**
An Interface which doesn't have any declaration inside but still enforces a mechanism.

54. **Which OO Concept is achieved by using overloading and overriding?**
Polymorphism.

55. **If i only change the return type, does the method become overloaded?**
No it doesn't. There should be a change in method arguements for a method to be overloaded.

56. **Why does Java not support operator overloading?**
Operator overloading makes the code very difficult to read and maintain. To maintain code simplicity, Java doesn't support operator overloading.

57. **Can we define private and protected modifiers for variables in interfaces?**
No

58. **What is Externalizable?**
Externalizable is an Interface that extends Serializable Interface. And sends data into Streams in Compressed Format. It has two methods, writeExternal(ObjectOuput out) and readExternal(ObjectInput in)

59. **What modifiers are allowed for methods in an Interface?**
Only public and abstract modifiers are allowed for methods in interfaces.

60. **What is a local, member and a class variable?**
Variables declared within a method are "local" variables. Variables declared within the class i.e not within any methods are "member" variables (global variables). Variables declared within the class i.e not within any methods and are defined as "static" are class variables

61. **What is an abstract method?**
An abstract method is a method whose implementation is deferred to a subclass.

62. **What value does read() return when it has reached the end of a file?**
The read() method returns -1 when it has reached the end of a file.

63. **Can a Byte object be cast to a double value?**
No, an object cannot be cast to a primitive value.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

64. **What is the difference between a static and a non-static inner class?**
A non-static inner class may have object instances that are associated with instances of the class's outer class. A static inner class does not have any object instances.

65. **What is an object's lock and which object's have locks?**
An object's lock is a mechanism that is used by multiple threads to obtain synchronized access to the object. A thread may execute a synchronized method of an object only after it has acquired the object's lock. All objects and classes have locks. A class's lock is acquired on the class's Class object.

66. **What is the % operator?**
It is referred to as the modulo or remainder operator. It returns the remainder of dividing the first operand by the second operand.

67. **When can an object reference be cast to an interface reference?**
An object reference be cast to an interface reference when the object implements the referenced interface.

68. **Which class is extended by all other classes?**
The Object class is extended by all other classes.

69. **Which non-Unicode letter characters may be used as the first character of an identifier?**
The non-Unicode letter characters $ and _ may appear as the first character of an identifier

70. **What restrictions are placed on method overloading?**
Two methods may not have the same name and argument list but different return types.

71. **What is casting?**
There are two types of casting, casting between primitive numeric types and casting between object references. Casting between numeric types is used to convert larger values, such as double values, to smaller values, such as byte values. Casting between object references is used to refer to an object by a compatible class, interface, or array type reference.

72. **What is the return type of a program's main() method?**
void.

73. **If a variable is declared as private, where may the variable be accessed?**
A private variable may only be accessed within the class in which it is declared.

74. **What do you understand by private, protected and public?**
These are accessibility modifiers. Private is the most restrictive, while public is the least restrictive. There is no real difference between protected and the default type (also known as package protected) within the context of the same package, however the protected keyword allows visibility to a derived class in a different package.

75. **What is Downcasting ?**
Downcasting is the casting from a general to a more specific type, i.e. casting down the hierarchy

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

76. **What modifiers may be used with an inner class that is a member of an outer class?**
A (non-local) inner class may be declared as public, protected, private, static, final, or abstract.

77. **How many bits are used to represent Unicode, ASCII, UTF-16, and UTF-8 characters?**
Unicode requires 16 bits and ASCII require 7 bits. Although the ASCII character set uses only 7 bits, it is usually represented as 8 bits. UTF-8 represents characters using 8, 16, and 18 bit patterns. UTF-16 uses 16-bit and larger bit patterns.

78. **What restrictions are placed on the location of a package statement within a source code file?**
A package statement must appear as the first line in a source code file (excluding blank lines and comments).

79. **What is a native method?**
A native method is a method that is implemented in a language other than Java.

80. **What are order of precedence and associativity, and how are they used?**
Order of precedence determines the order in which operators are evaluated in expressions. Associatity determines whether an expression is evaluated left-to-right or right-to-left

81. **Can an anonymous class be declared as implementing an interface and extending a class?**
An anonymous class may implement an interface or extend a superclass, but may not be declared to do both.

82. **What is the range of the char type?**
The range of the char type is 0 to 2^16 - 1.

83. **What is the range of the short type?**
The range of the short type is -(2^15) to 2^15 - 1.

84. **Why isn't there operator overloading?**
Because C++ has proven by example that operator overloading makes code almost impossible to maintain.

85. **What does it mean that a method or field is "static"?**
Static variables and methods are instantiated only once per class. In other words they are class variables, not instance variables. If you change the value of a static variable in a particular object, the value of that variable changes for all instances of that class. Static methods can be referenced with the name of the class rather than the name of a particular object of the class (though that works too). That's how library methods like System.out.println() work. out is a static field in the java.lang.System class.

86. **Is null a keyword?**
The null value is not a keyword.

87. **Which characters may be used as the second character of an identifier,but not as the first character of an identifier?**

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

The digits 0 through 9 may not be used as the first character of an identifier but they may be used after the first character of an identifier.

88. **Is the ternary operator written x : y ? z or x ? y : z ?**
It is written x ? y : z.

89. **How is rounding performed under integer division?**
The fractional part of the result is truncated. This is known as rounding toward zero.

90. **If a class is declared without any access modifiers, where may the class be accessed?**
A class that is declared without any access modifiers is said to have package access. This means that the class can only be accessed by other classes and interfaces that are defined within the same package.

91. **Does a class inherit the constructors of its superclass?**
A class does not inherit constructors from any of its superclasses.

92. **Name the eight primitive Java types.**
The eight primitive types are byte, char, short, int, long, float, double, and boolean.

93. **What restrictions are placed on the values of each case of a switch statement?**
During compilation, the values of each case of a switch statement must evaluate to a value that can be promoted to an int value.

94. **What is the difference between a while statement and a do statement?**
A while statement checks at the beginning of a loop to see whether the next loop iteration should occur. A do statement checks at the end of a loop to see whether the next iteration of a loop should occur. The do statement will always execute the body of a loop at least once.

95. **What modifiers can be used with a local inner class?**
A local inner class may be final or abstract.

96. **When does the compiler supply a default constructor for a class?**
The compiler supplies a default constructor for a class if no other constructors are provided.

97. **If a method is declared as protected, where may the method be accessed?**
A protected method may only be accessed by classes or interfaces of the same package or by subclasses of the class in which it is declared.

98. **What are the legal operands of the instanceof operator?**
The left operand is an object reference or null value and the right operand is a class, interface, or array type.

99. **Are true and false keywords?**
The values true and false are not keywords.

100. **What happens when you add a double value to a String?**
The result is a String object.

Shailesh Kumar - (shailesh24k7@yahoo.com) - 9404065157

101. **What is the diffrence between inner class and nested class?**
When a class is defined within a scope od another class, then it becomes inner class. If the access modifier of the inner class is static, then it becomes nested class.

102. **Can an abstract class be final?**
An abstract class may not be declared as final

103. **What is numeric promotion?**
Numeric promotion is the conversion of a smaller numeric type to a larger numeric type, so that integer and floating-point operations may take place. In numerical promotion, byte, char, and short values are converted to int values. The int values are also converted to long values, if necessary. The long and float values are converted to double values, as required

104. **What is the difference between a public and a non-public class?**
A public class may be accessed outside of its package. A non-public class may not be accessed outside of its package.

105. **To what value is a variable of the boolean type automatically initialized?**
The default value of the boolean type is false

106. **What is the difference between the prefix and postfix forms of the ++ operator?**
The prefix form performs the increment operation and returns the value of the increment operation. The postfix form returns the current value all of the expression and then performs the increment operation on that value.

107. **What restrictions are placed on method overriding?**
Overridden methods must have the same name, argument list, and return type. The overriding method may not limit the access of the method it overrides. The overriding method may not throw any exceptions that may not be thrown by the overridden method.

108. **What is a Java package and how is it used?**
A Java package is a naming context for classes and interfaces. A package is used to create a separate name space for groups of classes and interfaces. Packages are also used to organize related classes and interfaces into a single API unit and to control accessibility to these classes and interfaces.

109. **What modifiers may be used with a top-level class?**
A top-level class may be public, abstract, or final.

110. **What is the difference between an if statement and a switch statement?**
The if statement is used to select among two alternatives. It uses a boolean expression to decide which alternative should be executed. The switch statement is used to select among multiple alternatives. It uses an int expression to determine which alternative should be executed.

111. **What are the practical benefits, if any, of importing a specific class rather than an entire package (e.g. import java.net.* versus import java.net.Socket)?**
It makes no difference in the generated class files since only the classes that are actually used are

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

referenced by the generated class file. There is another practical benefit to importing single classes, and this arises when two (or more) packages have classes with the same name. Take java.util.Timer and javax.swing.Timer, for example. If I import java.util.* and javax.swing.* and then try to use "Timer", I get an error while compiling (the class name is ambiguous between both packages). Let's say what you really wanted was the javax.swing.Timer class, and the only classes you plan on using in java.util are Collection and HashMap. In this case, some people will prefer to import java.util.Collection and import java.util.HashMap instead of importing java.util.*. This will now allow them to use Timer, Collection, HashMap, and other javax.swing classes without using fully qualified class names in.

112. **Can a method be overloaded based on different return type but same argument type ?**
No, because the methods can be called without using their return type in which case there is ambiquity for the compiler

113. **What happens to a static var that is defined within a method of a class ?**
Can't do it. You'll get a compilation error

114. **How many static init can you have ?**
As many as you want, but the static initializers and class variable initializers are executed in textual order and may not refer to class variables declared in the class whose declarations appear textually after the use, even though these class variables are in scope.

115. **What is the difference between method overriding and overloading?**
Overriding is a method with the same name and arguments as in a parent, whereas overloading is the same method name but different arguments

116. **What is constructor chaining and how is it achieved in Java ?**
A child object constructor always first needs to construct its parent (which in turn calls its parent constructor.). In Java it is done via an implicit call to the no-args constructor as the first statement.

117. **What is the difference between the Boolean & operator and the && operator?**
If an expression involving the Boolean & operator is evaluated, both operands are evaluated. Then the & operator is applied to the operand. When an expression involving the && operator is evaluated, the first operand is evaluated. If the first operand returns a value of true then the second operand is evaluated. The && operator is then applied to the first and second operands. If the first operand evaluates to false, the evaluation of the second operand is skipped.

118. **Which Java operator is right associative?**
The = operator is right associative.

119. **Can a double value be cast to a byte?**
Yes, a double value can be cast to a byte.

120. **What is the difference between a break statement and a continue statement?**
A break statement results in the termination of the statement to which it applies (switch, for, do, or while). A continue statement is used to end the current loop iteration and return control to the loop statement.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9440065157

121. **Can a for statement loop indefinitely?**
Yes, a for statement can loop indefinitely. For example, consider the following: for(;;) ;

122. **To what value is a variable of the String type automatically initialized?**
The default value of an String type is null.
What is the difference between a field variable and a local variable?
A field variable is a variable that is declared as a member of a class. A local variable is a variable that is declared local to a method.

123. **How are this() and super() used with constructors?**
this() is used to invoke a constructor of the same class. super() is used to invoke a superclass constructor.

124. **What does it mean that a class or member is final?**
A final class cannot be inherited. A final method cannot be overridden in a subclass. A final field cannot be changed after it's initialized, and it must include an initializer statement where it's declared.

125. **What does it mean that a method or class is abstract?**
An abstract class cannot be instantiated. Abstract methods may only be included in abstract classes. However, an abstract class is not required to have any abstract methods, though most of them do. Each subclass of an abstract class must override the abstract methods of its superclasses or it also should be declared abstract.

126. **What is a transient variable?**
transient variable is a variable that may not be serialized.

127. **How does Java handle integer overflows and underflows?**
It uses those low order bytes of the result that can fit into the size of the type allowed by the operation.

128. **What is the difference between the >> and >>> operators?**
The >> operator carries the sign bit when shifting right. The >>> zero-fills bits that have been shifted out.

129. **Is sizeof a keyword?**
The sizeof operator is not a keyword.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

Some Additional Question for Java

1)What is OOPs?
Ans: Object oriented programming organizes a program around its data,i.e.,objects and a set of well defined interfaces to that data.An object-oriented program can be characterized as data controlling access to code.
2)what is the difference between Procedural and OOPs?
Ans: a) In procedural program, programming logic follows certain procedures and the instructions are executed one after another. In OOPs program, unit of program is object, which is nothing but combination of data and code.
b) In procedural program,data is exposed to the whole program whereas in OOPs program,it is accessible with in the object and which in turn assures the security of the code.
3)What are Encapsulation, Inheritance and Polymorphism?
Ans: Encapsulation is the mechanism that binds together code and data it manipulates and keeps both safe from
outside interference and misuse.
Inheritance is the process by which one object acquires the properties of another object.
Polymorphism is the feature that allows one interface to be used for general class actions.
4)What is the difference between Assignment and Initialization?
Ans: Assignment can be done as many times as desired whereas initialization can be done only once.
5)What are Class, Constructor and Primitive data types?
Ans: Class is a template for multiple objects with similar features and it is a blue print for objects. It defines a type of object according to the data the object can hold and the operations the object can perform.
Constructor is a special kind of method that determines how an object is initialized when created.
Primitive data types are 8 types and they are:
byte, short, int, long
float, double
boolean
char
6)What is an Object and how do you allocate memory to it?

Ans: Object is an instance of a class and it is a software unit that combines a structured set of data with a set of operations for inspecting and manipulating that data. When an object is created using new operator, memory is allocated to it.
7)What is the difference between constructor and method?
Ans: Constructor will be automatically invoked when an object is created whereas method has to be called explicitly.
8)What are methods and how are they defined?
Ans: Methods are functions that operate on instances of classes in which they are defined. Objects can communicate with each other using methods and can call methods in other classes.
Method definition has four parts. They are name of the method, type of object or primitive type the method
returns, a list of parameters and the body of the method. A method's signature is a combination of the first

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

three parts mentioned above.
9)What is the use of bin and lib in JDK?
Ans: Bin contains all tools such as javac, appletviewer, awt tool, etc., whereas lib contains API and all packages.
10)What is casting?
Ans: Casting is used to convert the value of one type to another.
11)How many ways can an argument be passed to a subroutine and explain them?
Ans: An argument can be passed in two ways. They are passing by value and passing by reference.
Passing by value: This method copies the value of an argument into the formal parameter of the subroutine.
Passing by reference: In this method, a reference to an argument (not the value of the argument) is passed to the parameter.
12)What is the difference between an argument and a parameter?
Ans: While defining method, variables passed in the method are called parameters. While using those methods, values passed to those variables are called arguments.
13)What are different types of access modifiers?
Ans: public: Any thing declared as public can be accessed from anywhere.
private: Any thing declared as private can't be seen outside of its class.
protected: Any thing declared as protected can be accessed by classes in the same package and subclasses in the other packages.
default modifier : Can be accessed only to classes in the same package.
14)What is final, finalize() and finally?
Ans: final : final keyword can be used for class, method and variables.

A final class cannot be subclassed and it prevents other programmers from subclassing a secure class to invoke insecure methods.
A final method can' t be overridden
A final variable can't change from its initialized value.
finalize( ) : finalize( ) method is used just before an object is destroyed and can be called just prior to
garbage collection.
finally : finally, a key word used in exception handling, creates a block of code that will be executed after a
try/catch block has completed and before the code following the try/catch block.
The finally block will execute whether or not an exception is thrown.
for example, if a method opens a file upon exit, then you will not want the code that closes the file to be bypassed by the exception-handling mechanism. This finally keyword is designed to address this contingency

Sir (shailesh24x7@yahoo.com) - 9404065157

Selenium - Web Driver Interview Questions

Selenium WebDriver

1. What is selenium webdriver?
2. How to write the tests in selenium webdriver?
3. How to configure selenium webdriver in eclipse?
4. What are the prerequisites to run selenium webdriver?
5. What is the difference between selenium 1.0 and webdriver?
6. What are the advantages of selenium webdriver?
7. What are the disadvantages of selenium webdriver over selenium 1.0?
8. How to handle multiple windows in selenium webdriver?
9. How to navigate with browser buttons in selenium webdriver?
10. Which are the locators used for recognizing the objects in selenium webdriver?
11. How to run the tests in internet explorer using selenium webdriver?
12. How to run the tests in firefox using selenium webdriver?
13. How to run the tests in google chrome using selenium webdriver?
14. How to run the tests without a browser or with HTML unit driver in selenium webdriver?
15. How to run Selenium 1.0 tests in webdriver?
16. How to convert selenium 1.0 tests to webdriver tests?
17. What is webdriver backed selenium?
18. When to use web driver backed selenium?
19. Which version of selenium IDE supports webdriver?
20. How to invoke an application in webdriver?
21. Which of Selenium IDE commands not supported in webdriver?
22. Where to download selenium webdriver?
23. What is implicit and explicit wait in selenium webdriver?

==========

1: What is Selenium 2.0? I have heard this buzz word many times.

Answer: Selenium 2.0 is a consolidation of two web testing tools:
a) Selenium RC and
b) Web Driver,
It claims to give best of both words: Selenium and Web Driver.

2: Why are two tools being combined as Selenium 2.0?

Answer: Selenium 2.0 promises to give much cleaner API than Selenium RC and at the same time not being restricted by JavaScript Security restriction like same origin policy.

3: So everyone is going to use Selenium 2.0?

Answer: Well no, for example if you are using Selenium Perl client driver than there is no similar offering from Selenium 2.0 and you would have to stick to Selenium 1.0 till there is similar library available for Selenium 2.0

4: So how do I specify my browser configurations with Selenium 2.0?

Answer: Selenium 2.0 offers following browser/mobile configuration:
a) Android Driver,
b) Chrome Driver,

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

c) EventFiringWebDriver,
d) Firefox Driver,
e) HtmlUnitDriver,
f) InternetExplorerDriver,
g) IPhoneDriver,
h) IPhoneSimulatorDriver,
i) RemoteWebDriver
And all of them have been implemented from interface WebDriver.To be able to use any of these drivers, you need to instantiate their corresponding classes.

5: How is Selenium 2.0 configuration different from Selenium 1.0?

Answer: In case of Selenium 1.0, you need Selenium jar file pertaining to one library; for example, in case of Java you need Java client driver and also Selenium server jar file. With Selenium 2.0, you need language Binding (i.e. Java, C#, etc) and Selenium server jar if you are using Remote Control or Remote Web Driver.

6. Selenium Benefits over Web Driver.

Selenium supports many browsers and many languages, Web Driver needs native implementations for each new language/browser combo.
Very mature and complete API
Currently (Sept 2010) supports JavaScript alerts and confirms better

Benefits of Web Driver Compared to Selenium

• Native automation faster and a little less prone to error and browser configuration
• Does not Requires Selenium-RC Server to be running
• Access to headlessHTMLUnit can allow really fast tests
• Great API

7. What is Web Driver?

Web Driver uses a different underlying framework from Selenium's javascript Selenium-Core. It also provides an alternative API with functionality not supported in Selenium-RC. WebDriver does not depend on a JavaScript core embedded within the browser; therefore it is able to avoid some long-running Selenium limitations.
WebDriver's goal is to provide an API that establishes
A well-designed standard programming interface for web-app testing.
• Improved consistency between browsers.
• Additional functionality addressing testing problems not well-supported in Selenium 1.0.
The Selenium developers strive to continuously improve Selenium. Integrating WebDriver is another step in that process. The developers of Selenium and of WebDriver felt they could make significant gains for the Open Source test automation community be combining forces and merging their ideas and technologies. Integrating WebDriver into Selenium is the current result of those efforts.

8.When to Use WebDriver?

One should use WebDriver when requiring improved support for
* Mult-browser testing including improved functionality for browsers not well-supported by Selenium-1.0.
* Handling multiple frames, multiple browser windows, popups, and alerts.
* Page navigation.
* Drag-and-drop.
* AJAX-based UI elements.

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

Advantages of web driver compared to selenium.

1) Support for iPhone and Android testing
2) Implementation of listeners - a much awaited feature
3) Better features for Ajax testing.
4) You can easily simulate clicking on front and back button of browser.
5) You can extract objects in bulk like QTP. For ex - extract all links of page. With RC this was a big hastle
6) Unlike RC you don't have to start a server in webdriver.
7) You can simulate movement of a mouse using selenium.
8) Tabs and pops are more or less the same. RC can also handle and Webdriver can also handle.
9) You can find coordinates of any object using Webdriver.
10) You have classes in Webdriver which help you to simulate key press events of keyboard.
10) Keyword driven framework is very easy to build in webdriver.

9.Which browsers does WebDriver support?

The existing drivers are the ChromeDriver, InternetExplorerDriver, FirefoxDriver, OperaDriver and HtmlUnitDriver. For more information about each of these, including their relative strengths and weaknesses, please follow the links to the relevant pages. There is also support for mobile testing via the AndroidDriver and IPhoneDriver.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

Selenium Interview Questions

Q1. What is Selenium?
Ans. Selenium is a set of tools that supports rapid development of test automation scripts for web based applications. Selenium testing tools provides a rich set of testing functions specifically designed to fulfil needs of testing of a web based application.

Q2. What are the main components of Selenium testing tools?
Ans. Selenium IDE, Selenium RC and Selenium Grid

Q3. What is Selenium IDE?
Ans. Selenium IDE is for building Selenium test cases. It operates as a Mozilla Firefox add on and provides an easy to use interface for developing and running individual test cases or entire test suites. Selenium-IDE has a recording feature, which will keep account of user actions as they are performed and store them as a reusable script to play back.

Q4. What is the use of context menu in Selenium IDE?
Ans. It allows the user to pick from a list of assertions and verifications for the selected location.

Q5. Can tests recorded using Selenium IDE be run in other browsers?
Ans. Yes. Although Selenium IDE is a Firefox add on, however, tests created in it can also be run in other browsers by using Selenium RC (Selenium Remote Control) and specifying the name of the test suite in command line.

Q6. What are the advantage and features of Selenium IDE?
Ans. 1. Intelligent field selection will use IDs, names, or XPath as needed
2. It is a record & playback tool and the script format can be written in various languages including C#, Java, PERL, Python, PHP, HTML
3. Auto complete for all common Selenium commands
4. Debug and set breakpoints
5. Option to automatically assert the title of every page
6. Support for Selenium user-extensions.js file

Q7. What are the disadvantage of Selenium IDE tool?
Ans. 1. Selenium IDE tool can only be used in Mozilla Firefox browser.
2. It is not playing multiple windows when we record it.

Q8. What is Selenium RC (Remote Control)?
Ans. Selenium RC allows the test automation expert to use a programming language for maximum flexibility and extensibility in developing test logic. For example, if the application under test returns a result set and the automated test program needs to run tests on each element in the result set, the iteration / loop support of programming language's can be used to iterate through the result set, calling Selenium commands to run tests on each item.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

Selenium RC provides an API and library for each of its supported languages. This ability to use Selenium RC with a high level programming language to develop test cases also allows the automated testing to be integrated with the project's automated build environment.

Q9. What is Selenium Grid?
Ans. Selenium Grid in the selenium testing suit allows the Selenium RC solution to scale for test suites that must be run in multiple environments. Selenium Grid can be used to run multiple instances of Selenium RC on various operating system and browser configurations.

Q10. How Selenium Grid works?
Ans. Selenium Grid sent the tests to the hub. Then tests are redirected to an available Selenium RC, which launch the browser and run the test. Thus, it allows for running tests in parallel with the entire test suite.

Q 11. What you say about the flexibility of Selenium test suite?
Ans. Selenium testing suite is highly flexible. There are multiple ways to add functionality to Selenium framework to customize test automation. As compared to other test automation tools, it is Selenium's strongest characteristic. Selenium Remote Control support for multiple programming and scripting languages allows the test automation engineer to build any logic they need into their automated testing and to use a preferred programming or scripting language of one's choice. Also, the Selenium testing suite is an open source project where code can be modified and enhancements can be submitted for contribution.

Q12. What test can Selenium do?
Ans. Selenium is basically used for the functional testing of web based applications. It can be used for testing in the continuous integration environment. It is also useful for agile testing

Q13. What is the cost of Selenium test suite?
Ans. Selenium test suite a set of open source software tool, it is free of cost.

Q14. What browsers are supported by Selenium Remote Control?
Ans. The test automation expert can use Firefox, IE 7/8, Safari and Opera browsers to run tests in Selenium Remote Control.

Q15. What programming languages can you use in Selenium RC?
Ans. C#, Java, Perl, PHP, Python, Ruby

Q16. What are the advantages and disadvantages of using Selenium as testing tool?
Ans. Advantages: Free, Simple and powerful DOM (document object model) level testing, can be used for continuous integration; great fit with Agile projects.

Disadvantages: Tricky setup; dreary errors diagnosis; can not test client server applications.

Q17. What is difference between QTP and Selenium?

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

Ans. Only web applications can be testing using Selenium testing suite. However, QTP can be used for testing client server applications. Selenium supports following web browsers: Internet Explorer, Firefox, Safari, Opera or Konqueror on Windows, Mac OS X and Linux. However, QTP is limited to Internet Explorer on Windows.

QTP uses scripting language implemented on top of VB Script. However, Selenium test suite has the flexibility to use many languages like Java, .Net, Perl, PHP, Python, and Ruby.

Q18. What is difference between Borland Silk test and Selenium?
Ans. Selenium is completely free test automation tool, while Silk Test is not. Only web applications can be testing using Selenium testing suite. However, Silk Test can be used for testing client server applications. Selenium supports following web browsers: Internet Explorer, Firefox, Safari, Opera or Konqueror on Windows, Mac OS X and Linux. However, Silk Test is limited to Internet Explorer and Firefox.

Silk Test uses 4Test scripting language. However, Selenium test suite has the flexibility to use many languages like Java, .Net, Perl, PHP, Python, and Ruby.

What is Selenium?
Selenium is a suite of tools for browser automation. It is composed of "IDE", a recording and playback mechanism, "WebDriver" and "RC" which provide APIs for browser automation in a wide variety of languages, and "Grid", which allows many tests using the APIs to be run in parallel. It works with most browsers, including Firefox from 3.0 up to 7, Internet Explorer 8, Google Chrome, Safari and Opera 11.5

- Describe technical problems that you had with Selenium tool?

As with any other type of test automation tools like SilkTest, HP QTP, Watir, Canoo Webtest, Selenium allows to record, edit, and debug tests cases. However there are several problems that seriously affect maintainability of recorded test cases, occasionally Quality Assurance Engineers complain that it takes more time to maintain automated test cases than to perform manual testing; however this is an issue with all automated testing tools and most likely related to improper testing framework design. Another problem is complex ID for an HTML element. If IDs is auto-generated, the recorder test cases may fail during playback. The work around is to use XPath to find required HTML element. Selenium supports AJAX without problems, but QA Tester should be aware that Selenium does not know when AJAX action is completed, so ClickAndWait will not work. Instead QA tester could use pause, but the snowballing effect of several 'pause' commands would really slow down total testing time of test cases. The best solution would be to use waitForElement.

- What test can Selenium do?

Selenium could be used for the functional, regression, load testing of the web based applications. The automation tool could be implemented for post release validation with continuous integration tools like Jenkins, Hudson, QuickBuild or CruiseControl.

- What is the price of Selenium license per server?

Selenium is open source software, released under the Apache 2.0 license and can be downloaded and used without charge.

- How much does Selenium license cost per client machine?

Selenium is open source software, released under the Apache 2.0 license and can be downloaded and used without charge.

- Where to download Selenium?

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

Selenium can be downloaded and installed for free from seleniumhq.org

- What is the latest version of Selenium components?

The latest versions are Selenium IDE 1.3.0, Selenium Server (formerly the Selenium RC Server) 2.9.0, Selenium Client Drivers Java 2.9.0, Selenium Client Drivers C# 2.9.0, Selenium Client Drivers Ruby 2.8.0, Selenium Client Drivers Python 2.9, Selenium Grid 1.0.8.

- What is Selenium IDE?

Selenium IDE is a Firefox add-on that records clicks, typing, and other actions to make a test cases, which QA Tester can play back in the Firefox browser or export to Selenium RC. Selenium IDE has the following features: record/play feature, debugging with step-by-step and breakpoints, page abstraction functionality, an extensibility capability allowing the use of add-ons or user extensions that expand the functionality of Selenium IDE

- What are the limitations of Selenium IDE?

Selenium IDE has many great features and is a fruitful and well-organized test automation tool for developing test cases, in the same time Selenium IDE is missing certain vital features of a testing tool: conditional statements, loops, logging functionality, exception handling, reporting functionality, database testing, re-execution of failed tests and screenshots taking capability. Selenium IDE doesn't for IE, Safari and Opera browsers.

What does SIDE stand for?

Selenium IDE. It was a very tricky interview question.

- What is Selenium Remote Control (RC) tool?

Selenium Remote Control (RC) is the powerful solution for test cases that need more than simple browser actions and linear execution. Selenium-RC allows the developing of complex test scenarios like reading and writing files, querying a database, and emailing test reports. These tasks can be achieved by tweaking test cases in your preferred programming language.

- What are the advantages using Selenium as testing tool?

If QA Tester would compare Selenium with HP QTP or Micro Focus SilkTest, QA Engineer would easily notice tremendous cost savings for Selenium. In contrast to expensive SilkTest license or QTP license, Selenium automation tool is absolutely free. It means that with almost no investment in purchasing tools, QA Team could easily build the state of the art test automation infrastructure. Selenium allows developing and executing test cases in various programming languages including .NET, Java, Perl, RubyPython, PHP and even HTML. This is a great Selenium advantage, most likely your software developers already know how to develop and maintain C# or Java code, so they transfer coding techniques and best practices to QA team. Selenium allows simple and powerful DOM-level testing and in the same time could be used for testing in the traditional waterfall or modern Agile environments. Selenium would be definitely a great fit for the continuous integration tools Jenkins, Hudson, CruiseControl, because it could be installed on the server testing box, and controlled remotely from continuous integration build.

- What is Selenium Grid?

Selenium Grid extends Selenium RC to distribute your tests across multiple servers, saving you time by running tests in parallel.

- What is Selenium WebDriver?

Selenium WebDriver is a tool for writing automated tests of websites. It is an API name and aims to mimic the behaviour of a real user, and as such interacts with the HTML of the application. Selenium WebDriver is the successor of Selenium Remote Control which has been officially deprecated.

- How many browsers are supported by Selenium IDE?

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

Test Engineer can record and playback test with Selenium IDE in Firefox.

- Can Selenium test an application on iPhone's Mobile Safari browser?

Selenium should be able to handle Mobile Safari browser. There is experimental Selenium IPhone Driver for running tests on Mobile Safari on the iPhone, iPad and iPod Touch.

- Can Selenium test an application on Android browser?

Selenium should be able to handle Android browser. There is experimental Selenium Android Driver for running tests in Android browser.

- What are the disadvantages of using Selenium as testing tool?

Selenium weak points are tricky setup; dreary errors diagnosis; tests only web applications

- How many browsers are supported by Selenium Remote Control?

QA Engineer can use Firefox 7, IE 8, Safari 5 and Opera 11.5 browsers to run actuall tests in Selenium RC.

- How many programming languages can you use in Selenium RC?

Several programming languages are supported by Selenium Remote Control - C# Java Perl PHP Python Ruby

- How many testing framework can QA Tester use in Selenium RC?

Testing frameworks aren't required, but they can be helpful if QA Tester wants to automate test cases. Selenium RC supports Bromine, JUnit, NUnit, RSpec (Ruby), Test::Unit (Ruby), TestNG (Java), unittest (Python).

- How to developer Selenium Test Cases?

Using the Selenium IDE, QA Tester can record a test to comprehend the syntax of Selenium IDE commands, or to check the basic syntax for a specific type of user interface. Keep in mind that Selenium IDE recorder is not clever as QA Testers want it to be. Quality assurance team should never consider Selenium IDE as a "record, save, and run it" tool, all the time anticipate reworking a recorded test cases to make them maintainable in the future.

- What programming language is best for writing Selenium tests?

The web applications may be written in Java, Ruby, PHP, Python or any other web framework. There are certain advantages for using the same language for writing test cases as application under test. For example, if the team already have the experience with Java, QA Tester could always get the piece of advice while mastering Selenium test cases in Java. Sometimes it is better to choose simpler programming language that will ultimately deliver better success. In this case QA testers can adopt easier programming languages, for example Ruby, much faster comparing with Java, and can become become experts as soon as possible.

- Have you read any good books on Selenium?

There are several great books covering Selenium automation tool, you could check the review at Best Selenium Books: Top Recommended page

- Do you know any alternative test automation tools for Selenium?

Selenium appears to be the mainstream open source tool for browser side testing, but there are many alternatives. Canoo Webtest is a great Selenium alternative and it is probably the fastest automation tool. Another Selenium alternative is Watir, but in order to use Watir QA Tester has to learn Ruby. One more alternative to Selenium is Sahi, but is has confusing interface and small developers community.

- Compare HP QTP vs Selenium?

When QA team considers acquiring test automation to assist in testing, one of the most critical decisions is what technologies or tools to use to automate the testing. The most obvious approach will be to look to the software market and evaluate a few test automation tools. Read Selenium vs QTP comparison

- Compare Borland Silktest vs Selenium?

Check Selenium vs SilkTest comparison

- How to test Ajax application with Selenium

Ajax interview questions could be tough for newbie in the test automation, but will be easily cracked by Selenium Tester

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

with a relevant experience. Read the detailed approach at Testing Ajax applications with Selenium in the right way

- How can I learn to automate testing using Selenium?

Don't be surprised if the interviewer asks you to describe the approach for learning Selenium. This interviewer wants to hear how you can innovative software test automation process the company. Most likely they are looking for software professional with a good Selenium experience, who can do Selenium training for team members and get the team started with test automation. I hope this Selenium tutorial will be helpful in the preparation for this Selenium interview question.

1. What do you know about Selenium?

Selenium is a suite of tools for web automation testing.
Selenium first came to life in 2004 when Jason Huggins was testing an internal application at ThoughtWorks.
Selenium was a tremendous tool, it wasn't without it's drawbacks. Because of its Javascript based automation engine and the security limitations browsers apply to Javascript, different things became impossible to do.
Selenium Suite of projects include:
Selenium IDE
Selenium Core
Selenium 1 (known as. Selenium RC or Remote Control)
Selenium 2 (known as. Selenium Webdriver)
Selenium-Grid

2. What are the technical challenges with selenium?
View
As you know Selenium is a free ware open source testing tool. There are many challenges with Selenium.
1. Selenium Supports only web based applications
2. It doesn't support any non web based (Like Win 32, Java Applet, Java Swing, .Net Client Server etc) applications
3. When you compare selenium with QTP, Silk Test, Test Partner and RFT, there are many challenges in terms of maintainability of the test cases
4. Since Selenium is a freeware tool, there is no direct support if one is in trouble with the support of applications
5. There is no object repository concept in Selenium, so maintainability of the objects is very high
6. There are many challenges if one have to interact with Win 32 windows even when you are working with Web based applications
7. Bitmap comparison is not supported by Selenium
8. Any reporting related capabilities, you need to depend on third party tools
9. You need to learn any one of the native language like (.Net, Java, Perl, Python, PHP, Ruby) to work efficiently with the scripting side of selenium

3. What are the test types supported by Selenium?
Selenium could be used for testing the web based applications. The test types can be supported are:
1. functional,
2. regression,
3. load testing
The automation tool could be implemented for post release validation with continuous integration tools like:
1. Jenkins,
2. Hudson,

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

3. QuickBuild
4. CruiseCont

4. What are the capabilities of Selenium IDE?
Selenium IDE (Integrated Development Environment) works similar to commercial tools like QTP, Silk Test and Test Partner etc. The below mentioned points describes well about Selenium IDE.
1. Selenium IDE is a Firefox add-on.
2. Selenium IDE can support recording the clicks, typing, and other actions to make a test cases.
3. Using Selenium IDE A Tester can play back the test cases in the Firefox browser
4. Selenium IDE supports exporting the test cases and suites to Selenium RC.
5. Debugging of the test cases with step-by-step can be done
6. breakpoint insertion is possible
7. Page abstraction functionality is supported by Selenium IDE
8. Selenium IDE can supports an extensibility capability allowing the use of add-ons or user extensions that expand the functionality of Selenium IDE
4. What are the capabilities of Selenium IDE?
Selenium IDE (Integrated Development Environment) works similar to commercial tools like QTP, Silk Test and Test Partner etc. The below mentioned points describes well about Selenium IDE.
1. Selenium IDE is a Firefox add-on.
2. Selenium IDE can support recording the clicks, typing, and other actions to make a test cases.
3. Using Selenium IDE A Tester can play back the test cases in the Firefox browser
4. Selenium IDE supports exporting the test cases and suites to Selenium RC.
5. Debugging of the test cases with step-by-step can be done
6. breakpoint insertion is possible
7. Page abstraction functionality is supported by Selenium IDE
8. Selenium IDE can supports an extensibility capability allowing the use of add-ons or user extensions that expand the functionality of Selenium IDE
6. Which are the browsers supported by Selenium IDE?
Selenium IDE supports only one browser Mozilla Firefox. The versions supported as of now are:
Mozilla Firefox 2.x
Mozilla Firefox 3.x
The versions not supported as of now are:
earlier versions of Mozilla Firefox 2.x
Mozilla Firefox 4.x
7. How to execute a single line command from Selenium IDE?
Single line command from Selenium IDE can be executed in two ways
1. Right click on the command in Selenium IDE and select "Execute This Command"
2. Select the command in Selenium IDE and press "X" key on the keyboard
8. How to insert a start point in Selenium IDE?
Start point Selenium IDE can be set in two ways
1. Right click on the command in Selenium IDE and select "Set / Clear Start Point"
2. Select the command in Selenium IDE and press "S" key on the keyboard
3. You can have only one start point
4. If you have already set one start point and you selected other command as start point. Then the first start point will

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

be removed and the new start point will be set
1. Right click on the command in Selenium IDE and select "Inert New Comment"
2. If you want to comment an existing line. You need to follow the below mentioned steps.
a. Select the source tab in IDE
b. Select the line which you want to comment
c. Assume that if you want to comment a open command you need to write like below mentioned code

```
<tr>
<!--
<td>open&l/td>
<td>/node/304/edit&l/td>
<td></td>
-->
</tr>
```

9. How to insert a comment in Selenium IDE?
Comments in Selenium IDE can be set in two ways
1. Right click on the command in Selenium IDE and select "Inert New Comment"
2. If you want to comment an existing line. You need to follow the below mentioned steps.
a. Select the source tab in IDE
b. Select the line which you want to comment
c. Assume that if you want to comment a open command you need to write like below mentioned code

```
<tr>
<!--
<td>open&l/td>
<td>/node/304/edit&l/td>
<td></td>
-->
</tr>
```

10. How to insert a break point in Selenium IDE?
Break point can be set in two ways in Selenium IDE
1. Right click on the command in Selenium IDE and select "Toggle Break Point"
2. Select the command in Selenium IDE and press "B" key on the keyboard
3. If you want to clear the break point once again Spress "B" key on the keyboard
4. You can set multiple break points in Selenium IDE
11. How to debug the tests in Selenium IDE?
To debug or execute the test cases line by line. Follow the below mentioned steps
1. Insert a break point (see the question to know more How to insert a break point in Selenium IDE?)from the location where you want to execute step by step
2. Run the test case
3. execution will be paused at the given break point
4. Click on the step (Blue) button to continue with the next statement
5. Click on Run button, to continue executing all the commands at a time
12. How to export the tests from Selenium IDE to Selenium RC in different languages?

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404065157

From selenium IDE the test cases can be exported into the languages
1. .Net,
2. Java,
3. Perl,
4. Python,
5. PHP,
6. Ruby
The below mentioned steps can explain how to export the test cases
1. Open the test case from Selenium IDE
2. Select File -> Export Test Case As
13. How to export Selenium IDE Test Suite to Selenium RC Suites?
From selenium IDE the test suites can be exported into the languages as mentioned below
1. .Net,
2. Java,
3. Perl,
4. Python,
5. PHP,
6. Ruby
The below mentioned steps can explain how to export the test suites
1. Open the test case from Selenium IDE
2. Select File -> Export Test Suite As
14. Which is the command used for displaying the values of a variable into the output console or log?
The command used for displaying the values of a variable into the output console or log - echo
If you want to display a constant string. The below mentioned command can be used
echo <constant string>
ex: echo "The sample message"
If you want to display the value of a variable it can be written like below
echo ${<variable name>>
ex: echo ${var1}
Note: Here var1 is the variable
15. Which are the browsers supported by Selenium RC?
Supported browsers for Selenium RC include:
1. *firefox
2. *mock
3. *firefoxproxy
4. *pifirefox
5. *chrome
6. *iexploreproxy
7. *iexplore
8. *firefox3
9. *safariproxy
10. *googlechrome
11. *konqueror
12. *firefox2

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

13. *safari
14. *piiexplore
15. *firefoxchrome
16. *opera
17. *iehta
18. *custom

Note: Any third party browser is supported with *custom followed by the complete path of the browser with executable

16. Which are the Operating Systems supported by Selenium?

Selenium IDE

Works in Firefox 2+ Start browser, run tests Run tests

Operating Systems Supported:

1. Windows,
2. OS X
3. Linux
4. Solaris
5. Others whichever supports Firefox 2+

Selenium Remote Control

Used for starting browser and run tests

Operating Systems Supported:

1. Windows,
2. OS X
3. Linux
4. Solaris
5. Others

Selenium Core

Used for running tests

Operating Systems Supported:

1. Windows,
2. OS X
3. Linux
4. Solaris
5. Others

17. What is Selenium RC?

View

Selenium-RC is the solution for tests that need a little more than just simple browser actions and a linear execution. Selenium-RC leverages the full power of programming languages, creating tests that can do things like read and write external files, make queries to a database, send emails with test reports, and practically anything else a user can do with a normal application.

You will want to use Selenium-RC whenever your test requires logic not supported by running a script from Selenium-IDE

18. Why Selenium RC is used?

Selenium-IDE does not directly support:

1. condition statements
2. iteration

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

3. logging and reporting of test results
4. error handling, particularly unexpected errors
5. database testing
6. test case grouping
7. re-execution of failed tests
8. test case dependency
9. capture screenshots on test failures

The reason behind why Selenium-IDE does not support the above mentioned requirements is IDE supports only HTML language. Using HTML language we cannot achieve the above mentioned requirements. Because HTML does not support conditional, looping and external source connectives.

To overcome the above mentioned problems Selenium RC is used.

Since Selenium RC supports the languages .Net, Java, Perl, Python, PHP, and Ruby. In these languages we can write the programme to achieve the IDE issues

19. Which are the languages supported by Selenium RC?

View

The languages supported by Selenium RC

1. .Net,
2. Java (Junt 3, Junt 4, TestNG, Groovy)
3. Perl,
4. Python,
5. PHP,
6. Ruby

20. What is Selenium Grid?

View

Selenium Grid is part of Selenium suite of projects. Selenium Grid transparently distribute your tests on multiple machines so that you can run your tests in parallel, cutting down the time required for running in-browser test suites. This will dramatically speeds up in-browser web testing, giving you quick and accurate feedback you can rely on to improve your web application.

21. What is Selenium WebDriver or Google WebDriver or Selenium 2.0?

View

WebDriver uses a different underlying framework from Selenium's javascript Selenium-Core. It also provides an alternative API with functionality not supported in Selenium-RC. WebDriver does not depend on a javascript core embedded within the browser, therefore it is able to avoid some long-running Selenium limitations.

WebDriver's goal is to provide an API that establishes

- A well-designed standard programming interface for web-app testing.
- Improved consistency between browsers.
- Additional functionality addressing testing problems not well-supported in Selenium 1.0.

The Selenium developers strive to continuously improve Selenium. Integrating WebDriver is another step in that process. The developers of Selenium and of WebDriver felt they could make significant gains for the Open Source testautomation community be combining forces and merging their ideas and technologies. Integrating WebDriver intoSelenium is the current result of those efforts.

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

22. What are the capabilities of Selenium WebDriver or Google WebDriver or Selenium 2.0?
One should use WebDriver when requiring improved support for
• Mult-browser testing including improved functionality for browsers not well-supported by Selenium-1.0.
• Handling multiple frames, multiple browser windows, popups, and alerts.
• Page navigation.
• Drag-and-drop.
• AJAX-based UI elements.
nderlying framework from Selenium's javascript Selenium-Core. It also provides an alternative API with functionality not supported in Selenium-RC. WebDriver does not depend on a javascript core embedded within the browser, therefore it is able to avoid some long-running Selenium limitations.
WebDriver's goal is to provide an API that establishes
• A well-designed standard programming interface for web-app testing.
• Improved consistency between browsers.
• Additional functionality addressing testing problems not well-supported in Selenium 1.0.
The Selenium developers strive to continuously improve Selenium. Integrating WebDriver is another step in that process. The developers of Selenium and of WebDriver felt they could make significant gains for the Open Source testautomation community be combining forces and merging their ideas and technologies. Integrating WebDriver intoSelenium is the current result of those efforts.
24. What is the architecture of Selenium Grid?
The below mentioned theory explains about the setup of Selenium Grid with architecture and how it works.
Selenium Grid builds on the traditional Selenium setup, taking advantage of the following properties:
* The Selenium test, the application under test, and the remote control/browser pair do not have to be co-located. They communicate through HTTP, so they can all live on different machines.
* The Selenium tests and the web application under test are obviously specific to a particular project. Nevertheless, neither the Selenium remote control nor the browser is tied to a specific application. As a matter of fact, they provide a capacity that can easily be shared by multiple applications and multiple projects.
Consequently, if only we could build a distributed grid of Selenium Remote Controls, we could easily share it across builds, applications, projects - even potentially across organizations. Of course we would also need to address the scalability issues as described earlier when covering the traditional Selenium setup. This is why we need a component in charge of:
* Allocating a Selenium Remote Control to a specific test (transparently)
* Limiting the number of concurrent test runs on each Remote Control
* Shielding the tests from the actual grid infrastructure
Selenium Grid calls this component the Selenium Hub.
* The Hub exposes an external interface that is exactly the same as the one of a traditional Remote Control. This means that a test suite can transparently target a regular Remote Control or a Selenium Hub with no code change. It just needs to target a different IP address. This is important as it shields the tests from the gridinfrastructure (which you can scale transparently). This also makes the developer's life easier. The same test can be run locally on a developer machine, or run on a heavy duty distributed grid as part of a build – without ever changing a line of code.
* The Hub allocates Selenium Remote Controls to each test. The Hub is also in charge of routing the Selenese requests from the tests to the appropriate Remote Control as well as keeping track of testing sessions.
* When a new test starts, the Hub puts its first request on hold if there is no available Remote Control in the grid providing the appropriate capabilities. As soon as a suitable Remote Control becomes available, the Hub will serve the

Shailesh Kumar - (shailesh247@yahoo.com) - 9404065157

request. For the whole time, the tests do not have to be aware of what is happening within the grid; it is just waiting for an HTTP response to come back.

25. Does Selenium support mobile internet testing?

View

Selenium supports Opera. And opera is used in most of the Smart phones. So whichever Smart phone supports opera, selenium can be used to test. So, one can use Selenium RC to run the tests on mobiles.

For more details on supported browsers of Selenium Which are the browsers supported by Selenium RC?

26. Does Selenium support Google Android Operating System?

View

Yes, Selenium Web Driver or Google Web Driver or Selenium 2.0 supports Android Operating System. There are several libraries written to support Android Operating System.

Fri, 07/15/2011 - 04:35 — Visitor

selenium android driver testing

try this .... this is cool ... ;)

```
public class OneTest extends TestCase {
public void testGoogle() throws Exception {
AndroidDriver driver = new AndroidDriver();
// And now use this to visit Google
driver.get("http://google.com");
Thread.sleep(15000);
// Find the text input element by its name
WebElement element = driver.findElement(By.name("q"));
// Enter something to search for
element.sendKeys("hotmail");
//driver.findElement(By.name("q")).sendKeys("niloy.cit");
// Now submit the form. WebDriver will find the form for us from the element
element.submit();
//driver.findElement(By.name("btnG")).click();
// Check the title of the page
System.out.println("Page title is: " + driver.getTitle());
driver.quit();
}
}
```

how i can give command to a already run webdriver to open an application or websight in emulator or real device ??

27. What are the types of text patterns available in Selenium?

There are three types of patterns available in Selenium

1. globbing
2. regular expressions
3. exact

28. How to use regular expressions in Selenium?

Regular expressions in Selenium IDE can be used with the keyword - regexp: as a prefix to the value and patterns needs

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

to be included for the expected values.
For example if you want to use the regular expression for a command
Command: verifyText
Target: //font/font/b/font[1]
Value: Flight Confirmation # 2011-05-02451
in the above example Flight Confirmation is continuously changing each time you run the test case. So this can be written with a regular expression as mentioned below
Command: verifyText
Target: //font/font/b/font[1]
Value: regexp:Flight Confirmation # [0-9]{4}-[0-9]{2}-[0-9]{5,10}
29. What are the regular expression patterns available in Selenium?
Selenium regular expression patterns offer the same wide array of special characters that exist in JavaScript. Below are a subset of those special characters
PATTERN MATCH
. any single character
[ ] character class: any single character that appears inside the brackets
* quantifier: 0 or more of the preceding character (or group)
+ quantifier: 1 or more of the preceding character (or group)
? quantifier: 0 or 1 of the preceding character (or group)
{1,5} quantifier: 1 through 5 of the preceding character (or group)
| alternation: the character/group on the left or the character/group on the right
( ) grouping: often used with alternation and/or quantifier

30. What is Selenese?
Selenium set of commands which are used for running the test are called as Selenese.
There are three types of Selenese, those are:
1. Actions - used for performing the operations and interactions with the target elements
2. Assertions - used as check points
3. Accessors - used for storing the values in a variable

31. How do you add check points or verification points in Selenium?
check points or verification points are known as Assertions in Selenium. The keywords with below mentioned prefix will be used for adding check points or verification points.
1. verify
2. assert
3. waitFor
32. What is Assertion in Selenium?
Assertion is nothing but a check or verification point.
Assertion verifies the state of the application conforms to what is expected.
Examples include “make sure the page title is X” and “verify that this checkbox is checked.
33. What are the types of Assertions there in Selenium?
Selenium Assertions can be used in 3 modes:
1) assert - When an “assert” fails, the test will be aborted. If you are executing test suite, the next state case will start

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

2) verify - When a “verify” fails, the test will continue execution, logging the failure.
3) waitFor - “waitFor” commands wait for some condition to become true (which can be useful for testing Ajaxapplications). They will succeed immediately if the condition is already true. However, they will fail and halt the test if the condition does not become true within the current timeout setting

34. When to use Assert, Verify and WaitFor in Selenium?
1) assert - If the expected value is mandatory to continue with the next set of steps we will use Assert. As Assert aborts the test, if the expected value doesn't match. It is good to use for any mandatory checks.
2) verify - If the expected value is optional to continue with the next set of steps we will use Verify. As Verify continues executing with the next set of steps, if the expected value doesn't match. It is good to use for any optional checks.
3) waitFor - If your test needs to wait, if the expected value is not matching we use waitFor. We normally use waitFor for AJAX kind of controls loading within a page

35. What is an Accessor in Selenium?
Accessor is one of the type of Selenese.
I. Accessors are used for storing the value of a target in a variable.
Ex:
1) storeTitle - Stores the title of a window in a variable
2) storeText - Stores the target element text in a variable
II. Accessors are also used for evaluating the result and storing the result in a variable
Ex: storeTextPresent - Evaluates whether the text is present in the current window. If the text is present stores true in the variable else stores false
Ex: storeEementPresent - Evaluates whether the element is present in the current window. If the element is present stores true in the variable else stores false
36. When to use Accessors in Selenium?
View
Accessors are mostly used for storing the value in a variable.
The variable can be used for following reasons:
1) To get the value from an element and comparing with some dynamic value
2) To take a logical decision to execute the test steps
ex: if the value of the variable true execute step1 and step2 else step3 and step4
3) To execute some statements in a loop based on the value returned by the element
37. How to capture bitmaps in Selenium?
View
Bitmaps are captured using the Selenium set of commands. There are two modes of capturing the bitmaps
1) Capture the bitmap for the entire page - it captures the browser main page area of AUT
2) Capture the bitmap for the screen shots - it captures the entire screen shot like the print scree that you give from your keyboard
Selenium doesn't support bitmap capturing for an element on AUT.
38. Which are the commands used for capturing the bitmaps?
captureEntirePageScreenshot
Saves the entire contents of the current window canvas to a PNG file. Contrast this with the captureScreenshot

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157

command, which captures the contents of the OS viewport (i.e. whatever is currently being displayed on the monitor), and is implemented in the RC only. Currently this only works in Firefox when running in chrome mode, and in IE non-HTA using the EXPERIMENTAL "Snapsie" utility. The Firefox implementation is mostly borrowed from the Screengrab! Firefox extension. Please see captureEntirePageScreenshot for more details

captureEntirePageScreenshotAndWait

Saves the entire contents of the current window canvas to a PNG file. Contrast this with the captureScreenshot command, which captures the contents of the OS viewport (i.e. whatever is currently being displayed on the monitor), and is implemented in the RC only. Currently this only works in Firefox when running in chrome mode, and in IE non-HTA using the EXPERIMENTAL "Snapsie" utility. The Firefox implementation is mostly borrowed from the Screengrab! Firefox extension. Please see

captureEntirePageScreenshotAndWait for details.

Note: This command runs with only mozilla firefox when you run the tests from RC. Other browsers it will not support

39. What is the difference between captureEntirePageScreenshot and CaptureScreenShot?

View

captureEntirePageScreenshot

1. This captures the AUT web page only
2. This supports only mozilla firefox
3. Accepts two arguments. one is the file name to be saved and other argument is back ground color

CaptureScreenShot

1. This captures the System screen shot
2. This supports all the browsers when you run from Selenium RC
3. Accepts one argument. That is the file name to be saved.

41. What are the limitations of Selenium IDE

View

The limitations of Selenium IDE are:

1) Selenium IDE uses only HTML language
2) Conditional or branching statements execution like using of if, select statements is not possible
3) Looping statements using is not possible directly in Selenium HTML language in ide
4) Reading from external files like .txt, .xls is not possible
5) Reading from the external databases is not possible with ide
6) Exceptional handling is not there
7) A neat formatted Reporting is not possible with ide

To eliminate the above issues we use Selenium RC

What is Xpath ? When would you use them ?

Xpath is a way to navigate in XML document

Direct child denoted by /

Relative child by //

ID,class,names can be used Xpath

//input[@name='q']

//input[@id='ls-ib']

//input[@class='lst']

Use of contains : If only part of ID/class/name is constant then contains can be used

//input[contains(@id,'lst-ib')]

Shailesh Kumar (shailesh24x7@yahoo.com) - 9404066515

What is faster in Xpath,CSS Selector,ID,name ?
CSS selector is fastest

Selenium RC

What are main components of Selenium RC

- Client Libraries
- Selenium Server : which acts proxy between browser and AUT (Application under test)

Why do I need Selenium Server
To tackle Same origin Policy
How to start Selenium Server
java -jar <selenium-server>.jar
java -jar <selenium-server>.jar -port 5555
How browser started in selenium ?
Selenium = new DefaultSelenium(serverHost,serverport,browser,appURL)
Selenium.start()
browser can be *firefox,*iexplore etc
How do you use recovery scenario with Selenium ?
That is used through programming language like through Exception Handling in java

Can I execute Javascript from my tests ?
Selenium.getEval(windows.document.images.length)
What are Heightened Privilege Browsers ?
Firefox and IE have browser modes which are not restricted by same origin policy
firefox - chrome
IE - iehta

Selenium GRID

How does Selenium Gird Works?

- Uses combination of Selenium RC's server
- One RC server work as Hub while other are slaves
- when there is request for specific configuration for test execution then hub looks for a free RC Slave server

Steps for working in Selenium GRID

- Download jar files of Selenium Grid
- ant should be setup on system to be able to work with Grid
- Navigate to distribution directory and issue below command: **ant launch-hub**
- Above command will start hub on 4444 locally. This can be verified by navigating to URL : **http://localhost:4444/console**
- Start Selenium Grid slave server on same machine of hub as : **ant launch-remote-control**
- To start GRID RC Slaves on different machine as below

ant -Dport=<port> -Dhost=<hostname> -DhubURL=<huburl> launch-remote-control

hostname is the host where RC slave is running and huburl is the url of machine where grid hub is running

How do you specify environment while starting grid slave ?
ant -Denvironment="safari on Mac" launch-remote-control
How I use machine configuration in my selenium tests

Selenium = new DefaultSelenium("locahost",4444,'safari on Mac'**,'http://yahoo.com');**

Shailesh Kumar (shailesh247@yahoo.com) - 9404065157

WebDriver(Selenium 2.0)

What is Selenium 2.0 ?
Selenium RC + WebDriver
Browser Configuration with Selenium 2.0 ?
FirefoxDriver, InternetExplorerDriver, HTMLUnitDriver
How is Selenium 2.0 configuration different from Selenium 1.0 ?
Selenium 1.0 : selenium jar file for java client and Selenium server jar file
Selenium 2.0 : Language binding jar and selenium server jar only if you are using Remote Control or Remote WebDriver
Code example of setting Selenium 2.0 ?

```
WebDriver webdriver= new FirefixDriver() ;
webdriver.get("url");
webdriver.findElement(By.name("q")).sendKeys("test");
```

What all different element locators in Selenium 2.0 ?

```
driver.findElement(By.id("HTMLid"));
driver.findElement(By.name("HTMLname"));
driver.findElement(By.cssSelector("cssSelector"));
driver.findElement(By.xpath("XpathLocator"));
```

Shailesh Kumar - (shailesh24x7@yahoo.com) - 9404065157