**Interview Questions & Answer for Java (modified 09/21/16 )**

**1 Q: What are the four fundamental concepts of java?**

a. Abstraction  
b. Inheritance  
c. Polymorphism  
d. Encapsulation

**2 Q: What is Class?**

Class is a blueprint of its object.  It has state and behavior.(by state we mean variable and by behavior we mean method.)

its describe state and behavior of that object.

**3 Q: What is variable?**

Variable is name of memory location. example: int a(variable name)=5;

**4 Q: What is local and global variable?**

Local variable: Local variables are declared in methods, constructors, or blocks. Global variable declared outside the method inside the class. Global variable also known as Instance variable.

**5 Q: What is object?**

Object is an instance of a class.

**6 Q: What is method?**

Collection of statements which grouped together to perform an operation.

**7 Q: What is datatype?**

Data type is a set of data with having predefined characteristics. Data type also gives some space in memory.

There are two kinds of data type:

1: Primitive

Ex: byte, short, int, long, float, double, character, Boolean .

2: Referenced / Non-primitive:

 Ex:String, Integer

**103 Q: What are wrapper classes?**

Ans: Java provides specialized classes corresponding to  each of the primitive data types. These are called wrapper classes.

They are example: Integer, Character, Double, Boolean etc.

**8 Q: What is javac and jvm, jdk jre ?**

Javac means java compiler which compile java code jvm means java virtual machine which execute class file or machine language in order to get result of program. Jdk means java development kit which helps to build the program. Jre means java runtime environment which helps to run project.

**Difference between Default and public access modifier.**

Public is open from any where

Default:-

In same package default is public

But in different package default private.

**9 Q: What is modifier ? How many type of modifier are there and what are those ?**

A modifier is a keyword placed in a class, method or variable declaration that changes how it operates.

Modifier are two kinds:

**1: access modifier:** It gives us access level of class, methods, variable. There are some different access level based on package and to create an object or inherite .

 access modifier are 4 kind.

 a. default  b. public   c. protected  d. private

 Creating an object in same package:

 public and default can access. Protected and private can not access.

 By inheritance in same package **:** all access modifier can access except private.

How to access private variable:

 By using getter setter method(getter mean read, setter mean write)

**example:**

private int a=8;

 int getA(){

return a;

 }

 int setA(int x){

a=x;

 return a;

 }

**2. Non-access modifier:**

Non access modifier are three kinds:

 a. final

b. static

c. abstract

**Example:**

**a. Final:**

If you declare a class as a final you can not inheritance but you can create an object. if you declare variable as a final you just can use it but you can not change. if you declare method as a final you can not override.

**b. static:**

you cannot declare static in a class level.

 if you declare variable as a static that will work with change value and you can access that with class name. if you declare method as a static you can overload that method but you can not override. you can call that with class name and only can take static property.

**c. abstract:**

If you declare class as a abstract you can not create object but u can inherit.  abstract class can take abstract and non abstract method

**Access Modifier:** Access modifier works with access level. That means if we want to give access to others or not then we use access modifier Rules: For same package rules are the same when we create an object or inherit In different package if we create an object then we can only access public. If we inherit/extends we can access public and protected Public: same package: Yes different package: Yes Protected: same package: Yes different package: Yes (only when we inherit/extends) Private: same package: No different package: No Default: same package: Yes different package: No

**10 Q:** D**ifference between instance and class variable?**

1. When you declare a variable with static keyword in a class level that called class variable.
2. When you declare a variable in a class level without static keyword is called instance variable.

**11 Q: What is static variable ? Tell me the benefit of static variable?**

When you use static keyword before any variable that is called static variable. you cannot declare static in a class level. if you declare variable as a static that will work with change value and you can access that with class name. if you declare method as a static you can overload that method but you can not override. you can call that with class name and only can take static property.

Static variable makes your program memory efficient (i.e it saves memory).

**12 Q: How to divided a string?**

String s="how are you";

String s1[]=s.split(" ");

**13 Q: How to compare between two string?**

String s="how";

String s1="are";  
s.compareTo(s1);

it will return int value -

   0- if both are equal

  (+) positive int - first object is greater than the second one

   (-) negative int - first object is less than the second one

**14 Q: What is the difference between string and string-buffer?**

The main difference between String and StringBuffer is string is immutable and StringBuffer is mutable

**15 Q: What is the syntax for For Each loop / Enhanced Loop?**

When we do not know the size / length (condition) then we should use For Each loop:

int a[]={4,6,5,9,8};

for(int x:a)

{

}

For loop- from JDK 1.00

Enhance/ for each loop- from JDK 5.00

**16 Q: What is array? Why we need an array?**

Array is a container which hold same type of value and fixed size. When you need more than one value in a same variable .

**17 Q: Can u tell me how to get max or minimum number from an array?**

int a[]={4,76,4,8,9};

int max=a[0];

for(int i=1;i<a.length;i++){

if(a[i]>max)//if u just say a[i]<max tahole

minimum number paben.

{

max=a[i];

}

}

syso(max)

**18 Q: How to get a private variable in another class?**

By using getter setter method(getter mean read, setter mean write)

**Example:**

private int a=8;  
  
int getA()  
{  
  
return a;  
  
}  
  
int setA(int x)  
{  
  
a=x;  
  
return a;  
  
}

**19 Q: How to get an array as a descending order or ascending ?**

Ans:int a[]={6,8,7,9,4};

Arrays.sort(a);  
  
for(int i=0;i<a.length;i++)  
{  
  
    syso(a[i]);  
  
}  
  
for(int i=a.length-1;i>=0;i--)  
{  
  
    syso(a[i]);  
  
}

**20 Q : What is the most important feature of Java?**

Ans : Java is a platform independent language.

**21 Q: 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) because of JVM which actually helps to do that.

**22 Q: What is a JVM?**

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

**23 Q: 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.

**24 Q: What is the base class of all classes ?**

Ans: Object

Few methods of Object class : finalize(), notify(), notifyAll(), equals(), wait(), wait(x), wait(x, y), getClass(), hashcode()

**25 Q: Does Java support multiple inheritance ?**

Ans: Java doesn't support multiple inheritance.

**26 Q: Is Java a pure object oriented language?**

Java uses primitive data types and can not inherite more than a class that why java is not a pure object oriented language.

**27 Q: Are arrays primitive data types ?**

No, In Java Arrays are objects.

**30 Q: 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.

**Example:**

static final int MAX\_LENGTH = 50; is an example for constant.

**31 Q: Should a main() method be compulsorily declared in all java classes?**

It is not required for all classes but we need a class with main() method at least once to execute our project.

**32 Q: What is the return type of the main() method?**

Main() method doesn't return anything hence declared void.

**33 Q: Why is the main() method declared static?**

main() is called by the JVM even before the instantiate of the class hence it is declared as static.

**34 Q: What is the argument of main() method?**

main() method accepts an array of String object as argument.

**35 Q: Does the order of public and static declaration matter in main() method?**

No. It doesn't matter but void should always come before main().

**36 Q: 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.

**37 Q: What is a package?**

Package is a collection of related classes and interfaces.

**38 Q: Which package is imported by default?**

java.lang package is imported by default even without a package declaration.

**39 Q: Can a class declared as private or protected?**

Not possible.

**40 Q: 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.

**41 Q: 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.

**42 Q: I don't want my class to be inherited by any other class. What should i do?**

You have to declared your class as final because final class cannot be inherited.

**43 Q: Can you give few examples of final classes defined in Java API?**

java.lang.String,  java.lang.Math are final classes.

**44 Q: 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.

**45 Q: Can a class be declared as static?**

We can not declare top level class as static, but only inner class can be declared static.

**Example:**

public class Test

{

static class InnerClass

{

//is called inner class

}

}

**\*\*\*\***

**46 Q: When will you define a method as static and what is restriction?**

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.

A static method should not refer to instance variables without creating an instance and cannot use "this" operator to refer the instance.

**47 Q: 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 and can be called by class name.

**48 Q: Can we declare a static variable inside a method?**

Static variables are class level variables and they can't be declared inside a method. If declared, the class will not compile.

**49 Q: 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 and can not create an object of that class.

This is basically to avoid instance creation of the class.

**50 Q: Can an abstract class be declared final?**

No

**51 Q: What is use of a abstract variable?**  
  
   Ans: Variables can't be declared as abstract.   
  
  
**52 Q: Can a abstract class be defined without any abstract methods?**

Yes it's possible. This is basically to avoid instance creation of the class.

**53 Q: Lets say you have a class which implement a interface but class did not implements all methods then is that possible you can create an object of that class?**

No because regular class can’t take unimplemented methods so that you have to declared that class as abstract. As we know abstract class can’t be instantiated.

**54 Q: Can a method inside a Interface be declared as final?**

No not possible. public and abstract are the only applicable modifiers for method declaration in an interface.

**55 Q: Can an Interface implement another Interface?**

No not possible.

**56 Q: Can an Interface extend another Interface?**

Yes an Interface can inherit another Interface, for that matter an  Interface can extend more than one Interface.

**57 Q: 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.

**58 Q: Can an Interface be declared as a final?**

Ans: No.

**59 Q: Can a class be defined inside an Interface?**  
  
  Ans: Yes it's possible.  
  
**60 Q: Can an Interface be defined inside a class?**  
  
  Ans: it's possible.  
  
**61 Q: Which object oriented Concept is achieved by using overloading and overriding?**  
  
  Ans: Polymorphism.  
  
**62 Q: What modifiers are allowed for methods in an Interface?**  
  
  Ans: Only public and abstract modifiers are allowed for methods in interfaces.  
  
**63 Q: What is an abstract method?**  
  
   Ans: An abstract method is a method which is unimplemented.  
  
**64 Q: What is the difference between a static and a non-static inner class? ( Same as # 75 )**  
  
  Ans: Nested classes are divided into two categories: static and non-static. Nested classes that are declared static are simply called static nested classes. Non-static nested classes are called inner classes.

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 Q: What is the % operator?**  
 Ans: It is referred to as the remainder operator. It returns the remainder of dividing the first

**66 Q: What restrictions are placed on method overloading?**  
  
  Ans: Method name must be same, parameter must be different, access modifier and return type may be different, and it happens in the same class.

**What are the restrictions placed on method overriding?**  
  
 Ans : 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.  
  
**67 Q: What is casting?**  
  
Ans: **Casting** really means is taking an Object of one particular type and “turning it into” another Object type. This process is called **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.  
  
  
**68 Q: If a variable is declared as private, where may the variable be accessed?**  
  
  Ans : A private variable may only be accessed within the class in which it is declared. If you want to give access to another class use getter and setter method. getter mean read and setter mean write  
  
**69 Q: What does it mean that a method or field is "static"?**  
  
  Ans : Static variable is called class variable . Static variable always work with change value. Static variable u can call with class name means without creating an object. 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). Static method can be overload but u can not override.  
  
  
**70 Q: Does a class inherit the constructors of its super class?**  
  
  Ans : A class does not inherit constructors from any of its super classes.  
  
  
**71 Q:** **What restrictions are placed on the values of each case of a switch statement ?**  
  
  Ans : During compilation, the values of each case of a switch statement must evaluate to a value that can be promoted to an int value.  
  
**72 Q: What is the difference between a while statement and a do while statement ?**  
  
  Ans : A while statement checks condition at the beginning of a loop  to see whether the  loop will be execute or not.

A do while statement checks at the end of a loop to see whether the condition is right or not. The do while statement will always execute the body of a loop at least once.  
  
**73 Q: When does the compiler supply a default constructor for a class ?**  
  
   Ans : The compiler supplies a default constructor for a class if no other  constructors are provided.  
  
**74 Q: What is the difference between inner class and nested class? ( Same as # 64)**  
  
  Ans: Nested classes are divided into two categories: static and non-static. Nested classes that are declared static are simply called static nested classes. Non-static nested classes are called inner classes.

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.  
  
**75 Q: Can an abstract class be final?**  
  
  Ans: An abstract class must not be declared as final.  
  
**76 Q: What is the difference between a public and a non-public class?**  
  
  Ans: A public class may be accessed outside of its package. A non-public class may not be accessed outside of its package.  
  
**77 Q: To what value is a variable of the boolean type automatically initialized?**  
  
  Ans: The default value of the boolean type is false.  
  
**78. What are the restrictions placed on method overriding?**  
  
 Ans: 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.  
  
**79. What modifiers may be used with a top-level class?**  
  
  Ans: A top-level or regular class may be public, abstract, or final.  
  
**80. What is the difference between an if statement and a switch statement?**  
  
  Ans: 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.  
  
**81.What happens to a static variable that is defined within a method of a class ?**  
  
  Ans: You can not declared static variable in a method.  
  
**82. What is the difference between the Boolean & operator and the && operator?**  
  
   Ans: &  is bitwise. && is logical.

 & evaluates both sides of the operation. a&b- it will consider the both side and    
 && evaluates the left side of the operation, A&&B-  if it's  true, it continues and evaluates the right side.  
  
**83.What is break and continue keyword in loop?**

Ans: 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.  
  
**84. Can a for statement loop indefinitely?**  
  
   Ans: Yes, a for statement can loop indefinitely. For example, consider the following: for(;;);  
  
**85. To what value is a variable of the String type automatically initialized?**  
  
  Ans: The default value of an String type is null.  
  
**86. How are this() and super() used with constructors?**  
  
   Ans: this() is used to invoke a constructor of the same class. super() is used to invoke a superclass constructor.

**87. What does it mean that a class or member is final?**  
  
  Ans : 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.  
  
**88. What does it mean that a method or class is abstract?**  
  
  Ans : 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 super classes or it also should be declared  abstract.  
  
**89. What is a transient variable?**  
  
  Ans: Transient variable is a variable that may not be serialized.  
  
**90. What is the difference between a constructor and a method?**  
  
  Ans: A constructor is a member function of a class that is used to create objects of that class. It has the same name as the class itself, has no return type, and is invoked using the new operator.

A method is an ordinary member function of a class. It has its own name, a return type (which may be void), and is invoked using the dot operator.  
  
**91. What is the purpose of garbage collection in Java, and when is it used?**  
  Ans: The purpose of garbage collection is to identify and discard objects that are no longer needed by a program so that their resources can be reclaimed and reused.

A Java object is subject to garbage collection when it becomes unreachable to the program in which it is used.

**Q: What is Abstraction in java ?**

Ans:  Abstraction: Abstraction is a processes that hiding the implementation details and showing some essentials details.

Abstraction in Java or Object oriented programming is a way to segregate implementation from an interface and one of the five fundamentals along with Encapsulation, Inheritance, Polymorphism, Class and Object.  
  
**92 Q: What is an abstract class?**  
  
  Ans: Abstract class must be extended/ subclass (to be useful). It serves as a template. A class that is abstract may not be instantiated ( i e. you may not call its constructor), abstract class may contain static data. Any class with an abstract method is automatically abstract itself, and must be declared as such. A class may be declared abstract even if it has no abstract methods. This prevents it from being instantiated.  
**94 Q: What is an Iterator?**  
  
  Ans : Iterator is a cursor and interface of collections framework, it is used to get data from table one by one. It has 2 methods Next and hasNext. hasNext() helps to verify if the next element is present and Next() is use to retrieve the element.

**95 Q: What is static in java?**  
  
  Ans : Static means one per class, not one for each object no matter how many instance of a class might exist. This means that we can use them without creating an instance of a class. Static variable work with change value. Static methods can be overload but can’t be override and we can call with class name. Static method can’t take non static variable or method (property).

Static keyword we can’t use in top class but we can use in nested class.  
  
**96 Q. What is final class?**  
  
  Ans: Final class can not be inherited by sub class. Only u can instantiate.  
  
**97 Q: Do I need to import java.lang package any time? Why ?**  
  
  Ans: No. It is by default loaded internally by the JVM.  
  
**98 Q: What are Checked and UnChecked Exception?**  
  
  Ans: The exceptions which are checked by compiler for smooth execution of the programme at runtime are called  checked exception.  
  Example: FileNotFoundException, NoSuchElementException  etc  In the Case of checked exceptions compiler will check whether we are handling exception if programmer not handling then we will get compile time error.

The exceptions which are not checked by compiler are called unchecked exception Example: StringIndexOutOfBoundsException, ArithmaticException, NuulPointException.  
 In the case of unchecked exception compiler won't check whether programmer handling exception or not.  
  
**99 Q: What is the default value of an object reference declared as an instance variable?**  
  
  Ans: The default value will be null unless we define it explicitly.  
  
**100. Primitive data types are passed by reference or pass by value?**  
  
  Ans : Primitive data types are passed by value.  
  
**101 Q: What is serialization?**  
  
  Ans : Serialization is a mechanism by which we can save the state of an object by converting it to a byte stream.  
  
**102 Q: What is the common usage of serialization?**  
  
  Ans: Whenever an object is to be sent over the network, objects need to be serialized. Moreover if the state of an object is to be saved, objects need to be serialized.  
  
**104 Q: What are runtime exceptions?**  
  
  Ans: Runtime exceptions are those exceptions that are not warn by compiler but it thrown at runtime. Ex:StackOverflowException ,MemoryoutException ,ArithmaticException  
  
**106 Q: What are the different ways to handle exceptions?**  
  
There are two ways to handle exceptions,  
    
  Ans:1. By wrapping the desired code in a try block followed by a catch block to catch the exceptions. and  
  
  2. List the desired exceptions in the throws clause of the method and let the caller of the method handle those exceptions.  
  
**107 Q: Is it necessary that each try block must be followed by a catch block?**  
  
  Ans: It is not necessary that each try block must be followed by a catch block. It should be followed by either a catch block or a finally block. And whatever exceptions are likely to be thrown should be declared in the throws clause of the method.

**108 Q: If I write return at the end of the try block, will the finally block still execute?**  
  
  Ans : Yes even if we write return as the last statement in the try block and no exception occurs, the finally block will execute. The finally block will execute and then the control return.  
  
**109 Q: What is synchronization and why is it important?**  
  
  Ans : Synchronization is a process which helps for muti-threading means project can take more than one thread or request at the same time.

If we don't have this concept in a project we can not handle more than one request at same time.

**110 Q: What are the steps in the JDBC connection?**  
  
  Ans:While making a JDBC connection we go through the following steps   
  
@Test  
public void jdbc() throws ClassNotFoundException {  
Connection con = null;  
Statement st = null;  
ResultSet rs = null;  
String url = "jdbc:<mysql://localhost:3306/testdb>";  
String user = "root";  
String password = "";  
try {  
Class.forName("com.mysql.jdbc.Driver"); // to initialize the driver  
con = DriverManager.getConnection(url, user, password); // create connection in to data base  
st = con.createStatement(); // to create statement  
rs = st.executeQuery("SELECT VERSION()"); // to get result set  
if (rs.next()) {  
System.out.println(rs.getString(1)); // to print the 1st value from table  
}  
} catch (SQLException ex) {  
Logger lgr = Logger.getLogger(JDBCConnection.class.getName());  
lgr.log(Level.SEVERE, ex.getMessage(), ex);  
} finally {  
try {  
if (rs != null) {  
rs.close();  
}  
if (st != null) {  
st.close();  
}  
if (con != null) {  
con.close();  
}  
} catch (SQLException ex) {  
System.out.println("......");  
}  
}  
}

**\*\*\*111 Q: How does a try statement determine which catch clause should be used to handle an exception?**  
  
  Ans : When an exception is thrown within the body of a try statement, the catch clauses of the try statement are examined in the order in which they appear. The first catch clause that is capable of handling the exception is executed. The remaining catch clauses are ignored.

**112 Q: Is String a primitive data type in Java?**  
  
Ans: No. String is not a primitive data type in Java, even though it is one of the most extensively used object. Strings in Java are instances ( object )of String class defined in java.lang package.  
  
**113 Q: What happens if you don't initialize an instance variable of any of the primitive types in Java?**  
  
Ans : Java by default initializes it to the default value for that primitive type. Thus an int will be initialized to 0(zero), a boolean will be initialized to false.  
  
**114 Q: What is HashMap and Map?**  
  
Ans : Map is an Interface and Hashmap is the class that implements Map.  
  
**116 Q: Difference between Vector and ArrayList?**  
  
Ans: Vector is synchronized whereas arraylist is not.

\*\*\*\*\*

Throw

Throws   
**117 Q : How to handle exception in java?**  
  
Ans : By using Try Catch block we can handle exception. Basic syntax of Try Catch block and combination of Try Catch block are explained below –  
  
**First combination:**  
  
Try  
  
{  
            Any statement;  
  
}  
  
catch ( Any Exception Class     Any variable name )  
  
{  
            syso(“print whatever you want”);  
}  
  
    **Second Combination:**  
Try  
{  
            Any statement;  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
.  
.  
.  
  
catch (Exception(-parent class of exception)     Any variable name)  
{  
            syso(“print whatever you want”);  
}

**Third combination:**  
  
Try  
  
{  
            Any statement;  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
.  
.  
.  
  
catch (Exception(-parent class of exception)    Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
  
finally  
  
{  
            statements;  
}  
  
**Fourth combination**  
  
  
Try  
  
{  
            Any statement;  
}  
  
finally  
  
{  
            statements;  
}  
  
\*\* finally will always execute no matter what.  
  
  
**118:What is the difference between error and exception?**  
Ans:

Any departure from the expected behavior of the system or program, which stops the working of the system is an error.

Compile time error is normally called an error.

Any error or problem which programmer can handle and continue to work normally is an exception.

Runtime error is called exception.

Error means is major problem which developer can not handle such as system error.

Exception means the problem which developer can handle by developer can handle using try catch block or throws keyword.   
  
**119 Q: What is difference between method overloading and overriding?**  
  
Ans : Overloading: Overloading happen in a same class. Same method name with different parameter (size/type). Access modifier and return type may be changed. Compile time polymorphism. Or Static Polymorphism.  
  
Overriding: Overriding happens between parent class and child class. The body will be changed but the access modifier, method name, return type and parameter cannot be changed. Run time polymorphism. Or Dynamic Polymorphism.  
  
**120 Q: what is the difference between final finally and finalize?**  
  
Ans: 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 not be overridden and final variable can not be changed.

Finally is an exception handling code section which gets executed whether an exception is raised or not by try block code segment.

Finalize() is a method of object class which will be executed by the JVM just before the garbage collecting object to give a final chance for resource releasing activity.

As I said earlier final keyword can be used along with variable, method and Class in Java. If you make a variable final, you can not change it's value, it will act like a constant. final variables are initialized at the time of creation except in case of blank final variable which is initialized in Constructor. If you make a method final in Java, you can not override it in sub class . If you make a class final means it can not be sub classed. Making a class final automatically makes all its method final and this is sometime required due to security reason, final keyword also help to write Immutable classes which are critical for designing thread-safe multi-threading system and reducing amount of synchronization. Now let's see What is finally in Java? As I said finally is used for exception handling along with try and catch. As per Java programming language’s rule, for exception handling you at least need either catch or finally block. finally block has special advantage over catch that its guaranteed to be executed despite whether Exception is thrown or not, this makes it, an ideal place to close system resource e.g.InputStream or OutputStream, which is required to release scarce file descriptor. Closing streams, network connection, database connection in finally block is good coding practice in Java. By the way from Java 7 you can use try with resource block to close resource automatically. Since finally is guaranteed to be executed on most cases,  finally block always execute, except in case of JVM dies i.e. calling System.exit() .   
Now let’s see What is finalize() method, finalize() is called by Garbage collection thread just before collecting eligible Objects. This is the last chance for object to perform any cleanup but since its not guaranteed that whether finalize() will be called, its bad practice to keep resource till finalize call. Though you can build a safety net on finalize by double checking scarce resources. See 10 points on finalize method to know more about specific points of finalize(). So, final, finally and finalize all are different keyword, they are used for different purpose. only similarity between them is that they are a Java programming language keyword, other than that final, finalize and finally are completely different than each other.  
  
**121 Q: What is difference between array and array-list?**  
  
Ans: Arrays can contain primitive or Objects   
  
whereas ArrayList can contain only Objects.   
  
Arrays are fixed size whereas Array List size is dynamic or resizable  
  
Arrays doesn’t provide a lot of features like Array List, such as addAll, removeAll, iterator etc.

Array take same datatype as a value but ArrayList take Different type of value.

Array work with primitive datatype but ArrayList works only with object type.  
  
  
**122 : What is interface ? why we need interface?**  
  
Ans : Interface is like a class but not a class; except we can only declare methods and variables in Interface, we cannot actually implement the method in Interface. In other words, Interface is basically a collection of abstract/unimplemented method.  
  
Ans- When test scenario is available but acceptance criteria and requirement specifications are not finalized.

**Interface features:**  
  
    Class implements the interface  
    One class can implement more than one interface at a time.  
    A class can extend one class with implement more than one interface   
  
 one implement at a time.  
    Interface extends more than one interface.  
    Interface cannot implement another interface.  
    Interface centralizes all classes.  
  
**123 Q: When a class must be abstract class? When we need abstract class?**  
  
Ans : An abstract class is a class with collections of implemented and unimplemented method, which cannot be instantiated (instantiated means we can’t create object for abstract class)  
  
Even there is only one unimplemented method in a class then that class is known as Abstract class.  
  
When we have to keep a method as unimplemented in a class- we have to write that method as abstract method, then the whole class will be Abstract class.  
  
**124 Q: How to execute constructor of Abstract class?**  
  
 Ans: We need to extend the abstract class, where we have to create a constructor - through this constructor by the help of keyword ‘super’ we can reach parent class. Then we have to create object in another class then the constructor of extended class will be executed; this is how the constructor of Abstract class will be executed.  
  
**125 Q: What is difference between interface and abstract?**  
  
Ans: Abstraction in Java or Object oriented programming is a way to segregate implementation from an interface and one of the five fundamentals along with Encapsulation, Inheritance, Polymorphism, Class, and Object.

|  |
| --- |
| 1) Abstract class can **have abstract and non-abstract** methods. | | Interface can have **only abstract** methods. |
| 2) Abstract class **doesn't support multiple inheritance**. | | Interface **supports multiple inheritance**. |
| 3) Abstract class **can have final, non-final, static and non-static variables**. | | Interface has **only static and final variables**. |
| 4) Abstract class **can have static methods, main method and constructor**. | | Interface **can't have static methods, main method or constructor**. |
| 5) Abstract class **can provide the implementation of interface**. | | Interface **can't provide the implementation of abstract class**. |
| 6) The **abstract keyword** is used to declare abstract class. | | The **interface keyword** is used to declare interface. |
| 7) **Example:** public abstract class Shape{ public abstract void draw()  {  } } | | **Example:** public interface Drawable{ void draw()  {  } } |
| 8) If you add new method to abstract class, you can provide default implementation of it. So you don’t need to change your current code | | If you add new method to interface, you have to change the classes which are implementing that interface |
| 9) Abstract classes are almost same as java classes except you can not instantiate it | | Different from normal java class |
| 10) It is faster than interface | | Interface is somewhat slower as it takes some time to find implemented method in class |

**126 Q : What is polymorphism?**

Ans : Polymorphism is the ability of an object to take many forms.

Polymorphism is the different forms of an object/class.

Polymorphism  is a concept which allows u to give the ability an object in different form. This relation is called is a relationship with different form means in order to perform this concept you have to inherite the parent class first then u can use only. It include method overload and override concept as well.  
  
There are two types of polymorphism in java:   
  
compile time polymorphism and runtime polymorphism. We can perform polymorphism in java by method overloading and method overriding.

Ex:class Animal        class dog extends Animal       class cow extends Animal        class smoke{ Animal a=new Dog():  Animal a1=new Cow();}

{

}                                   {

                                      }

**127:What is encapsulation?**  
  
Ans : Encapsulation is a mechanism that binds together code and the data it manipulates and keeps both safe from outside the class. It is also called data hiding.

you can give access if you want using getter and setter method.  
  
**128 Q: What is inheritance ? is that possible java can inherit more than one class?**  
  
Ans : In inheritance is the process by which one object acquires the properties of another object  Inheritance is a mechanism wherein a new class is derived from existing class (oop concept). A class can’t inherit more than one class.  
  
**129 Q : What is difference between list and set?**  
  
Ans : difference between them is that List allows duplicates and maintain insertion order while Set doesn't allow duplicates and doesn't maintain any order.   
  
**130 Q : What is difference between Hashmap and hash table?**

 There are many differences between HashMap and Hashtable classes that are given below.

|  |
| --- |
| 1) HashMap is **non synchronized**. It is not-thread safe and can't be shared between many threads without proper synchronization code. | Hashtable is **synchronized**. It is thread-safe and can be shared with many threads. |
| 2) HashMap **allows one null key and multiple null values**. | Hashtable **doesn't allow any null key or value**. |
| 3) HashMap is a **new class introduced in JDK 1.2**. | Hashtable is a **legacy class**. |
| 4) HashMap is **fast**. | Hashtable is **slow**. |
| 5) We can make the HashMap as synchronized by calling this code Map m = Collections.synchronizedMap(hashMap); | Hashtable is internally synchronized and can't be unsynchronized. |
| 6) HashMap is **traversed by Iterator**. | Hashtable is **traversed by Enumerator and Iterator**. |
| 7) Iterator in HashMap is **fail-fast**. | Enumerator in Hashtable is **not fail-fast**. |
| 8) HashMap inherits **AbstractMap** class. | Hashtable inherits **Dictionary** class. |

**131 Q : What is difference between array list and linkedlist?**

**Ans : Difference between ArrayList and LinkedList**

ArrayList and LinkedList both implements List interface and maintains insertion order. Both are non synchronized classes.

But there are many differences between ArrayList and LinkedList classes that are given below.

|  |
| --- |
| 1) ArrayList internally uses **dynamic array** to store the elements. | | LinkedList internally uses **doubly linked list** to store the elements. |
| 2) Adding and retrival the element ArrayList is best choice. | | To remove element Linkedlist is best choice |
| 3) ArrayList class can **act as a list** only because it implements List only. | | LinkedList class can **act as a list and queue** both because it implements List and Deque interfaces. |
| 4) ArrayList is **better for storing and accessing** data. | | LinkedList is **better for manipulating** data. |

**132 Q: What is difference between Enumeration and Iterator?**  
  
Ans: Enumeration is twice as fast as Iterator and uses very less memory. Enumeration is very basic and fits to basic needs. But Iterator is much safer as compared to Enumeration because it always denies other threads to modify the collection object which is being iterated by it. Enumeration works in only List interface from collection framework and it can not remov e any element. But in Iterator work in whole collection framework and it can remove the element if needed.

**133 Q: How to generate random number using java?**  
  
Ans:int random = (int)(Math.random() \* 5000 + 1);  
  
**134.Lets say you have a string(String s="how are you") how can remove all space between word?**  
Ans:String s="How are u";

   s.replace(" ","");

first parameter takes old character and second parameter takes new character.  
  
**135. What is constructor ? When u need constructor ? can you overload or override a constructor?**  
   
Ans : Constructor is a block of code which execute at the time of object creation.

To construct something means lets say we have to change a value of a variable then we can use constructor.

Constructor name always same as class name. we can have constructor as many as we want in a class. But we have to change parameter type or parameter size. To change the parameter size or type we can overload a constructor.

We can’t override a constructor in java.  
  
**136.What is regular expression in java?**  
  
Ans: A regular expression is a special sequence of characters that helps us match or find other strings or sets of strings, using a specialized syntax held in a pattern. They can be used to search, edit, or manipulate text and data.  
  
Here is some expression:  
  
\A    Matches beginning of string.  
\Z    Matches end of string. If a newline exists, it matches just before newline.  
\z    Matches end of string.  
\G    Matches point where last match finished.  
\n    Back-reference to capture group number   
  
"n"  
\b    Matches word boundaries when outside brackets. Matches backspace (0x08) when inside brackets.  
\B    Matches non word boundaries.  
\n, \t, etc.    Matches newlines, carriage returns, tabs, etc.  
\Q    Escape (quote) all characters up to \E  
\E    Ends quoting begun with \Q  
  
  
**137.What is**[**java.io?**](http://java.io/)  
  
    Ans:[Java.io](http://java.io/) is a package in java where you can get FileInputStream and FileOutputStream etc.  
  
**138 Q: If you create a object it showing an error what could be the reason ?**  
  
    Ans : Mainly it could be Constructor parameter did not match.  
**139.What is Java Collections Framework? List out some benefits of Collections framework?**  
  
   Ans : Collection framework basically work for data structure. There are some limitation in regular data type and array concept in java such as array is fixed size and it takes only same data type. To overcome this limitation collection framework take place where we can store our data in different format like ArrayList class can take duplicate value , HashSet class can take only unique value and Hashmap class can take only key-value structure.

:Collection framework have some interface like List, Set, Map, Iterator etc , Collections interfaces implements different classes like arrayList ,vector, LinkedHashSet , HashMap, Hashtable  etc. **it has algorithm concept as well**.

Java Collections have come through a long way with usage of Generics and Concurrent Collection classes for thread-safe operations. It also includes blocking interfaces and their implementations in java concurrent package. Some of the benefits of collections framework are Reduced development effort by using core collection classes rather than implementing our own collection classes. Code quality is enhanced with the use of well tested collections framework classes. Reduced effort for code maintenance by using collection classes shipped with JDK. Reusability and Interoperability  
  
**140 Q: What is Big-O notation? Give some examples?**  
  
    Ans : The Big-O notation describes the performance of an algorithm in terms of number of elements in a data structure.

Since Collection classes are actually data structures, we usually tend to use Big-O notation to chose the collection implementation to use based on time, memory and performance. Example 1: ArrayList get(index i) is a constant-time operation and doesn’t depend on the number of elements in the list.So it’s performance in Big-O notation is O(1).  
  
Example 2: A linear search on array or list performance is O(n) because we need to search through entire list of elements to find the element.  
  
  
**141 Q: What is Exception handling?**  
  
   Ans : The exception handling in java is one of the powerful mechanisms to handle the run time errors so that the normal flow of the application can be maintained.  
  
When there will be error the exception handling will print message and move on to the next phase to maintain the normal flow of that application.

There are two type of exception :

1.Checked or compile time exception

2.Unchecked or Run Time exception.

**\*\* What is parameterized method ?**

Ans : If we want to get any value inside any method from outside then we will have to create a parameterized method.

\*\* What is return type method ?

Ans : If we want to give any value to others we have to give it through return , so the method has to be return type.

**Updated Interview Questions**

142 Q: If you declare a variable locally without assigning a value then what will be the default value ?

Ans: For int 0, for String null, for boolean false.

143 Q: What is the difference between notify and notifyAll ?

Ans: Both notify() and notifyAll()  are used to notify waiting threads.

**notify**() wakes up the first thread that called wait() on the same object.

**notifyAll**() wakes up all the threads that called wait() on the same object.

144 Q: What is the difference between equals and hascode...... Fix

**Ans:** Both equals() and hashCode() are defined in java.lang.Object class and there default implementation is based upon Object information e.g. default equals() method return true, if two objects are exactly same i.e. they are pointing to same memory address, while default implementation of hashcode method return int and implemented as native method.

Similar default implementation of toString() method, returns type of class, followed by memory address in hex String.

 I want a brief definition about the equals() , "==" and hashCode(). If i run following code means the output will be "true false 2420395 2420395". But i had understand that equals() method compares the string and "==" compares the reference. But in output the hashCcode() method prints the reference number for both strings as same then why the "==" returns "false".

String str = "Name";

String str1 = new String("Name");

if(str.equals(str1))

System.out.println("true");

else

System.out.println("false");

if(str==str1)

System.out.println("true");

else

System.out.println("false");

System.out.println(str.hashCode());

System.out.println(str1.hashCode());

}

………………………………………………………………………………….

## equals () only compare string it's does not check reference of string

but '==' check reference and data both

in 1st case String str = "Name"; only one object is created but in

2nd case Two object is created

String str1 = new String("Name");

then reference are not same of both string that means it returns false

145 Q: How many class can you declare in the same source file ?

Ans: As many as we want

146 Q: Difference between inner and nested class ?

Ans:

147 Q: Which modifier you can use in top class level ?

Ans: Only abstract & final

148 Q: Difference between Collection and Collections ?

Ans:

149 Q: Difference between Vector and Arraylist ?

Ans: Vector is synchronized but arrayalist is not synchronized

150 Q: How to get unique value from a list of data ?

Ans: HashSet

151 Q: Difference between encapsulation and private variable ?

Ans:

152 Q: What is Abstraction in java ?

Ans:

Abstraction in Java or Object oriented programming is a way to segregate implementation from an interface and one of the ~~five~~ fundamentals along with Encapsulation, Inheritance, Polymorphism, Class, and Object.

153 Q: How can you Abstract class constructor ?

Ans: Constructor can not be abstract

154 Q: What is Synchronization and multithreading ?

Ans:

155 Q: What is serialization ?

Serialization is a process in which current state of Object will be saved in stream of bytes.

As byte stream create is platform neutral hence once objects created in one system can be deserialized in other platform.

**What is the use of Serialization**

As written above serialization will translate the Object state to Byte Streams. This Byte stream can be used for different purpose.

* Write to Disk
* Store in Memory
* Sent byte stream to other platform over network
* Save byte stream in DB(As BLOB)

156 Q: When does finalize method gets called ?

Ans:

157 Q: What is exception ? difference between checked and unchecked exception ?

Ans:

158 Q: What is constant variable in java and some regular expression in java ?

Ans:

159 Q: What is iterator and enum? Difference between enumeration and iterator ?

Ans:

Iterator is a interface of collections framework, it is used to get data from table. It has 2 methods Next and hasNext. HasNext() helps to verify if the next element is present and Next() is use to retrieve the element.

An enum is essentially a constrained set of values.

|  |  |
| --- | --- |
| Enumeration | Iterator |
| 1. Enumeration does not have a remove() method.  2. Enumeration is less safe .  3. Enumeration uses less memory | 1. Iterator has a remove() method.  2.Iterator is much safe.  3.Iterator uses more memory |

That’s why Enumerator is twice as fast as iterator.

160 Q: How to sort a String ? How to print print a String in reverse ?

Ans: **import** java.util.Arrays;

**import** org.testng.annotations.Test;

**public** **class** SortArray

{

@Test

**public** **void** m()

{

String original="edcba";

**char**[] chars=original.toCharArray();

Arrays.*sort*(chars);

String sorted=**new** String(chars);

System.***out***.println(sorted);

}

}

161 Q: Can you tell me some of the methods name in Object class ?

Ans:  finalize(), notify(), notifyAll(), wait().

162 Q: Do you know what are the current version of all the tools you are using ?

Ans:  JDK 8.0

Eclipse 4.0.4

Selenium-Server-Standalone 3.3.0

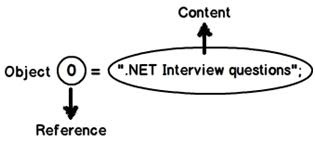
TestNg 6.9.4

163 Q: What is the difference between equals and == sign ?

Ans: The equals method compares the objects.

The == binary operator compares memory addresses.

object o = ".NET Interview questions";

[](http://1.bp.blogspot.com/-lSqEBTeEsE4/UXpXc_CTkaI/AAAAAAAAEek/tadFT10Tqck/s1600/ob1.jpg)

“==” compares if the object references are the same while “.Equals()” compares if the contents are the same.

So if you run the below code, both “==” and “.Equals()” returns true because content as well as references are the same.

164 Q: What is System.out.println ?

Ans:  System is a final class in java.lang package.

Out is a static member (variable) of PrintStream class.

Println is a method of PrintSteam class under package java.lang.

165 Q: Can you override a constructor ?

Ans: Constructor can only be overload.

166 Q:  Where do you use overload and overriding in WebDriver ?

### Ans: **Example for method overload in WebDriver.**

Frame(string), frame(int), frame(WebElement).

167 Q: Why do we need inheritence in Java ?

Ans:

168 Q: How many way you can handle exception ?

Ans:  2 ways

1. By wrapping the desired code in a try block ( By using try catch block ) followed by a catch block to catch the exceptions. And
2. List the desired exceptions in the throws clause of the method and let the caller of the method handle those exceptions.

170 Q: How to divide, reverse, sort, remove character and cont a String ?

Ans:

**class** ReverseString

{

**public** **static** **void** main(String args[])

{

StringBuffer a = **new** StringBuffer("Java programming is fun");

System.out.println(a.reverse());

}

}

171 Q:  What is Transient variable ?

Ans: Transient variable is a variable that may not be serialized .

It is a process of from data to binary code & from binary code to object.

172 Q: Difference between Set and list ? Compare and CompareTo ? HashMap and HashTable ?

Ans:

173 Q: Difference between Array and ArrayList ?

Ans:

174 Q: How to reverse a String using Framework ?

Ans:

 String string="whatever";  
 String reverse = new StringBuffer(string).reverse().toString();

 System.out.println(reverse);

175 Q:

Ans:

**176 Q: What is the difference between throw and throws in Java ?**

Ans: Java throw keyword is followed by instance

Java throw keyword is used in method

Java throw keyword is used to explicitly throw an exception

…………………………………………………………………..

Java throws keyword is followed by class

Java throws keyword is used in method signature

**throw** keyword is used to throw our own exception in the program. **throw new** Exception\_class **throws** is to declare an exception within a method.

e.g suppose some programmer defines a method and he knows that it may causes exception, but he/she don't want to handle that exception, then this method defined with the **throws** keyword. But when we call this function, it should be call with **try** and **catch** block to maintain the normal flow of the program

**177 Q: what operator we can use to replace if else condition in java?**

**Ans:**May be ternary operator .

What Is JavaFX?

What is Mojo in Maven?

what is mojo in maven?

Ans.

A Mojo is really just a goal in Maven 2, and plug-ins consist of any number of goals (Mojos). Mojos can be defined as annotated Java classes or Beanshell script. A Mojo specifies metadata about a goal: a goal name, which phase of the lifecycle it fits into, and the parameters it is expecting.MOJO is a play on POJO (Plain-old-Java-object), substituting "Maven" for "Plain"

What is javaFX?

Ans.**JavaFX** is a software platform for creating and delivering desktop applications, as well as rich internet applications (RIAs) that can run across a wide variety of devices. **JavaFX** is intended to replace Swing as the standard GUI library for Java SE, but both will be included for the foreseeable future.

|  |  |
| --- | --- |
| Epted111 | [StringBuffer](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuffer.html) is synchronized, [StringBuilder](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuilder.html) is not. |
|  |  |
| 574down vote | StringBuilder is faster than StringBuffer because it's not synchronized. |

Basically, StringBuffer methods are synchronized while StringBuilder are not.



\*\*\* What is the difference between encapsulation & data hidden ?

Ans:

|  |  |
| --- | --- |
| **Encapsulation** | **Data hiding** |
| 1. Data Encapsulation is a process | Data hiding is both a process and a technique |
| 1. Data in data encapsulation is either **public or private** | Data in data hiding is considered as **private and non accessible** |
| 1. Data encapsulation is one of the mechanisms of data hiding | Data hiding is having data encapsulation as one of the mechanisms |
| 1. Data encapsulation’s concern is for consolidation which is related to data’s hiding goal less complexity in data. | Data encompasses not only less data complexity but also data protection and security. |
| 1. Focus of data encapsulation is on the data inside the capsule. | Data hiding is concerned with restrictions and allowance in terms of access and use. |
| 1. Data encapsulation is the process of combining and assembling elements in order to create a new entity and the wrapping of private data into a class. | Data hiding is the process of hiding the details of an object or function. |

~~In Data hidding it is possible to use another class by getter & setter method~~

~~In Data encapsulation it is not possible to use another class by getter & setter~~

31 What is desired capability and what is firefox profile,firefox binary class ?

Ans:  DesiredCapabilities helps to set properties for the WebDriver

It is used to perform a parallel execution on different machine configurations.

desired capability is a parameterized method which is used in for cross browsing & parallel test as well.

**Parameters**

*binary*

Type: [OpenQA.Selenium.Firefox.FirefoxBinary](https://seleniumhq.github.io/selenium/docs/api/dotnet/html/T_OpenQA_Selenium_Firefox_FirefoxBinary.htm)  
A [FirefoxBinary](https://seleniumhq.github.io/selenium/docs/api/dotnet/html/T_OpenQA_Selenium_Firefox_FirefoxBinary.htm) object representing the operating system environmental settings used when running Firefox.

*profile*

Type: [OpenQA.Selenium.Firefox.FirefoxProfile](https://seleniumhq.github.io/selenium/docs/api/dotnet/html/T_OpenQA_Selenium_Firefox_FirefoxProfile.htm)  
A [FirefoxProfile](https://seleniumhq.github.io/selenium/docs/api/dotnet/html/T_OpenQA_Selenium_Firefox_FirefoxProfile.htm) object representing the profile settings when used in starting Firefox.

\*What is data provider explain

A DataProvider is data feeder method defined in a class that supplies a test method with data.

We may hook up any test method with a DataProvider and make the test method execute once for every row of our test data.

**What is Test Strategy?**

Test strategy is defined as a set of guiding principle that enlightens test design & regulates how testing needs to be done.

* Components of Test strategy includes- objectives and scope, documentation formats, test processes, team reporting structure, client communication strategy etc.

\*32 what is proxy and session id?

Server that stands between an external network (such as Internet) and an organization's internal (private) networks and serves as a firewall.

It prevents external users from directly accessing the internal information resources, or even knowing their location. All external requests for information are intercepted by the proxy server and checked for their validity, and only authorized requests are passed on to the internal server. This security, however, comes at the cost of inconvenience to genuine users and slower performance.  
2. Server that stores frequently requested information (documents or files) from elsewhere in the network and makes it available to the users.  
  
  
Read more: <http://www.businessdictionary.com/definition/proxy-server.html>

A **session ID** is a unique number that a Web site's server assigns a specific user for the duration of that user's visit (**session**).

The **session ID** can be stored as a cookie, form field, or URL (Uniform Resource Locator). Some Web servers generate **session**IDs by simply incrementing static numbers.

33 what is google analytics

**Google Analytics** is a [freemium](https://en.wikipedia.org/wiki/Freemium) [web analytics](https://en.wikipedia.org/wiki/Web_analytics) service offered by [Google](https://en.wikipedia.org/wiki/Google) that tracks and reports website traffic

**1.What is maven?**

Ans : maven is a project building tool.

In order to run a maven project after class name must be test(ex:bittechTest) and test class must be in src/test/java folder.

To get jar file from maven you have to must provide three things in pom.xml(group id,artifact id and version).

maven project can be run from command line also

For maven project you dont have to install or add in your project externally.

**2.What is .m2 folder in maven?**

Ans: This is folder where store all jar file or dependencies in local machine

**3.how to run one single method from command line?**

Ans: mvn test className#methodName test.

**4.what are command u use in maven?**

Ans: mvn test, mvn clean test, mvn compile

**5. What is plugin?**

**Ans:** In computing, a plug-in (or plugin, add-in, addin, add-on, addon, or extension) is a software component that adds a specific feature to an existing computer program.

plugins are where much of the real action is performed, plugins are used to: create jar files, create war files, compile code, unit test code, create project documentation, and so on.

**6.what is profile?**

Ans: Profiles are specified using a subset of the elements available in the POM ...

‎Project basedir (profiles.xml) .

\*35 what is maven life cycle

**What is maven life cycle?**

Ans : Validate - Validate the project is correct and all necessary information is available.

compile - compile the source code of the project

test - test the compiled source code using a suitable unit testing framework. These tests should not require the code be packaged or deployed

package - take the compiled code and package it in its distributable format, such as a JAR.

~~integration-test - process and deploy the package if necessary into an environment where integration tests can be run~~

verify - run any checks to verify the package is valid and meets quality criteria.

install - install the package into the local repository, for use as a dependency in other projects locally.

deploy - done in an integration or release environment ( build environment ), copies the final package to the remote repository for sharing with other developers and projects.

\*\*\* What is .m2 folder

.m2 folder is consider as local repository. And it is created automatically after first time execution of a mvn project. When a mvn project is run first time then it gets all the dependencies & necessary files from maven server/maven repository and all of these we get from the second time execution from local repository / local machine that is .m2 folder.

POM Contains -Profile , properties , all JARs, ,dependencies, all plugIn .

Maven repository.com

\*\*\*Step Defination folder will be created in the Src/test/java folder /SmokesTest

\*\*\*Folder.feature file in the project lebel

58 what is page object model describe it

Page object model is just a design pattern not frame work.

As we will be working as pages ( pages will be pure java classes )

Based on Application behavior we will be creating separate pages and will store all the locator and respective method to use them.

62 tell me some automation challenges

1. Reusability of code from POM ( Project Object model )
2. Handling the dynamic element ( Xpath )
3. Handling the element which are in the frame
4. Ajax Handling
5. Flash play ( coustome )
6. Non GUI commands
7. Dynamically.
8. Video element (Streming that means current live video)
9. Capsa

47.Difference between testng listener and webDriver listener?

Ans:

**TestNg Listner :** It shows the nature of activity of the test

& logging the test case activity.

**webDriver Listener:** Which kind of activity shows by the driver

& logging the driver activity.

**What is JavaScriptExecutor?**  
  
[JavaScriptExecutor](http://selenium.googlecode.com/git/docs/api/java/org/openqa/selenium/JavascriptExecutor.html) is an interface which provides mechanism to execute Javascript through selenium driver.

It provides “executescript” & "executeAsyncScript" methods, to run JavaScript in the context of the currently selected frame or window.

Why we use it?  
Sometimes in the application to find or locate an element selenium driver does not provide better performance in that case / for that reason we can use [JavaScriptExecutor](http://selenium.googlecode.com/git/docs/api/java/org/openqa/selenium/JavascriptExecutor.html) for better performance.

To enhance the capabilities of the existing scripts by performing javascript injection into our application under test.  
  
In simple words  “Javascript can be executed within the browser with the help of JavaScriptExecutor.”  
  
Package:-  
  
import org.openqa.selenium.JavascriptExecutor;

FRD- Function Requared Document

BRD- Business Requared Document

SME-Subject Matter Expart

What is the difference betn executescript” & "executeAsyncScript"?

**What is team velocity in Agile ?**

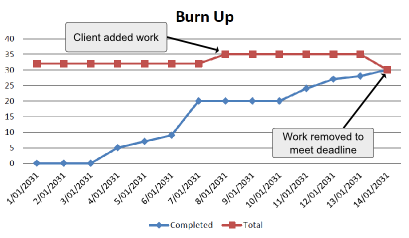
Velocity is a metric that predicts how much work an [Agile software development](http://searchsoftwarequality.techtarget.com/definition/agile-software-development) team can successfully complete within a two-week [sprint](http://searchsoftwarequality.techtarget.com/definition/Scrum-sprint) (or similar time-boxed period).

Velocity is a useful planning tool for estimating how fast work can be completed and how long it will take to complete a project. The metric is calculated by reviewing work the team successfully completed during previous sprints; for example, if the team completed 10 [stories](http://searchsoftwarequality.techtarget.com/definition/story) during a two-week sprint and each story was worth 3 [story points](http://whatis.techtarget.com/definition/story-point), then the team's velocity is 30 story points per sprint.

**What is Burn up ?**

A burn up chart tracks progress towards a projects completion. In the simplest form of burn up chart there are two lines on the chart:

1. A total work line (the project scope line)
2. A work completed line



A burnup chart clearly shows both completed work and project scope. The project will be completed when the lines meet.

The vertical axis is amount of work, and is measured in units customized to your own project. Some common units are number of tasks, estimated hours or story points (in agile project management methodologies). The horizontal axis is time, usually measured in days.

At each day you can see the amount of work completed and the total amount of work. The distance between the two lines is thus the amount of work remaining. When the two lines meet, the project will be complete. This is a powerful measure of how close you are to completion of the project, similar to a [burn down chart](http://www.clariostechnology.com/productivity/blog/whatisaburndownchart)

**What is Burn down ?**

A burn down chart is a visual representation of the amount of work that still needs to be completed before the end of a project. A burn down chart has a Y axis (work) and an X axis (time). Ideally, the chart illustrates a downward trend as the amount of work still left to do over time "burns down" to zero.

Q 5. What java version are you using? ---------JDK 8, Java 8.0

? Q 6. What class? give an real life example ?

Ans: Every Human can do things such as walking, speaking etc. this is known as method, walking is the same for everyone, but when I walk, not everyone walk. The act of walking only affects the Humaninstance which does this, but still it's defined by the Human class

Q 7. How many class can have in one source file ? How many can be public?

Ans: As many as we want. Public will be only one.

Q 9. What happen if u create a class as final ? OR

What is Customizing imutable class ?

When we declare a class is final then we can not inherit that class. We can only create object of that class.

If we declare a class is final then all the method belongs to the class is also final this procedure is called Customizing imutable class.

Q 20. what is volatile variable ?

Ans: When a field is declared volatile, the compiler and runtime are put on notice that this variable is shared and that operations on it should not be reordered with other memory operations.

Volatile variables are not cached in registers or in caches where they are hidden from other processors, so a read of a volatile variable always returns the most recent write by any thread.

Q 36. Difference between = and == sign?

Ans: You use "=" when you're assigning a value to a variable:

a = 7

== is used to comparing two objects.

Also

"==" is used to check some condition, for example:

if a == 7:

print a

which literally means "**if a equals 7**, print it"

Q 37. difference between == sign and equals method?

== is used to comparing two objects.

equals () compare the objects what they content

equals () only compare string it's does not check reference of string

but '==' check reference and data both

Q 38. Difference between equals and hashcode method?

**Ans:** Both equals() and hashCode() are defined in java.lang.Object class and there default implementation is based upon Object information e.g. default equals() method return true, if two objects are exactly same i.e. they are pointing to same memory address, while default implementation of hashcode method return int and implemented as native method.

Serialization: It is a one kind of conversion .The process by which object From java supported form to file or network/transportation supported form.

Deserealization : It is also a kind of conversion .The process by which object are convert to readable/usable format from the transportation format.

FileOutputStram

FileInputStram

**How can we create a thread ?**

# How to create thread

There are two ways to create a thread:

1. By extending Thread class
2. By implementing Runnable interface.

### **Thread class:**

|  |
| --- |
| Thread class provide constructors and methods to create and perform operations on a thread. Thread class extends Object class and implements Runnable interface. |

### **Commonly used Constructors of Thread class:**

|  |
| --- |
| * Thread() * Thread(String name) * Thread(Runnable r) * Thread(Runnable r,String name) |

### **Starting a thread:**

|  |
| --- |
| **start() method** of Thread class is used to start a newly created thread. It performs following tasks:   * A new thread starts(with new call stack). * The thread moves from New state to the Runnable state. * When the thread gets a chance to execute, its target run() method will run. |

### **1) Java Thread Example by extending Thread class**

1. **class** Multi **extends** Thread{
2. **public** **void** run(){
3. System.out.println("thread is running...");
4. }
5. **public** **static** **void** main(String args[]){
6. Multi t1=**new** Multi();
7. t1.start();
8. }
9. }

### **2) Java Thread Example by implementing Runnable interface**

1. **class** Multi3 **implements** Runnable{
2. **public** **void** run(){
3. System.out.println("thread is running...");
4. }
6. **public** **static** **void** main(String args[]){
7. Multi3 m1=**new** Multi3();
8. Thread t1 =**new** Thread(m1);
9. t1.start();
10. }
11. }

Output:thread is running...

|  |
| --- |
| If you are not extending the Thread class,your class object would not be treated as a thread object.So you need to explicitely create Thread class object.We are passing the object of your class that implements Runnable so that your class run() method may execute. |

What is exception in Java ?

# What Is an Exception?

The term exception is short hand for the phrase "exceptional event."

**Definition:** An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.

\*\*\*\*Legacy class all are synchronized.

? Angular js

? Single tens

Concorent hashmap : Synchronized HashMap is called Concorent HashMap

Hasing concept

Underlying data structure is

Hashtable

Unique value

Iterator is used in array alternate of for loop

Enhance loop can be used in array

Regular loop is not use in array

**Differences Between HashMap & ConcurrentHashMap**

|  |  |
| --- | --- |
| **HashMap** | **ConcurrentHashMap** |
| 1. HashMap allows only one null key & can take multiple null values. | 1. ConcurrentHashMap does not allow any null key & null values. |
| 1. HashMap is not synchronized . | 2. ConcurrentHashMap is synchronized. |
| 1. HashMap is not threadsafe. | 3. ConcurrentHashMap is threadsafe. |
| 1. In multiple threaded environment HashMap is usually faster than ConcurrentHashMap . As          only single thread can access the certain portion of the Map and thus reducing the performance .       While in HashMap any number of threads can access the code at the same time . | It is actually Extended version of Hashtable |

Q 1. Why java is object oriented programing langauge?

Q 2. Why java is not pure object oriented language?

Q 3. Why Java is platform independent language?

Q 4. Difference between jdk and jvm?

Q 5. What java version are you using? ----JDK 8, Java 1.8

In console type java –version.

java version "1.8.0\_91"

? Q 6. What class? give an real life example ?

Ans : Every Human can do things such as walking, speaking etc. this is known as method, walking is the same for everyone, but when I walk, not everyone walk. The act of walking only affects the Humaninstance which does this, but still it's defined by the Human class

Q 7. How many class can have in one source file ? How many can be public?

Ans: As many as we want. Public will be only one

Q 8. What modifier can take a class?

Q 9. What happen if u create a class as final ?

Q 10. what is the difference between inner class and nested class?

Q 11. Is inner class can take protected and private? Yes

Q 12. In order to create a class is there a constructor must need to create or not?

Q 13. Difference between class variable and instance variable?

Q 14. What is default value for int, String and Boolean ?

Q 15. Is that possible to declare a static variable inside method?

Q 16. If you declare a final variable is there any way u can assign value ?

Yes we can assign by creating constructor of that class.

Q 17. How to declare constant variable?

Q 18. How to get private variable in another class?

Q 19. What is transient variable ?

Serialization: It is a one kind conversion .The process by which object From java supported form to file or network/transportation supported form.

Deserealization : It is also a kind conversion .The process by which object are convert to readable/usable format from the transportation format.

FileOutputStram

FileInputStram

Q 20. what is volatile variable ?

Ans: Volatile keyword in java is used as an indicator to java compiler and thread that do not cache value of this variable and always read it from the main memory.

When a field is declared volatile, the compiler and runtime are put on notice that this variable is shared and that operations on it should not be reordered with other memory operations.

Volatile variables are not cached in registers or in caches where they are hidden from other processors, so a read of a volatile variable always returns the most recent write by any thread.

Q 21. How many primitive data type? What are they?

Q 22. What is wrapper class?

Q 23. Is that possible main method could overloaded and overridden? if yes why?

Ans : Yes, you can overload main method in Java. But the program doesn't execute the overloaded main method when you run your program, you have to call the overloaded main method from the actual main method.

that means main method acts as an entry point for the java interpreter to start the execute of the application. Where as a loaded main need to be called from main.

You cannot override a static method. You have just hide the main method MainOverride class.

MAIN is a class method (since its static by definition). Hence, it does not makes sense to "override" it (or for that matter any static method).

Q 24. Why main method is void static and public?

Q 25. What is args in main method?

args is an array name whose data type is string.

In public static void **main**(String **args**[]) **args** is an array of console line **argument** whose data type is String.

**in this array, we can store various string arguments by invoking them at the command line**

as shown below: java myProgram Shaan Royal then Shaan and Royal will be stored in the array as **arg**[0]="Shaan"; **arg**[1]= “Royal”

Q 26. What is System.out.println in java?

System is a final class in java.lang package .

Println is a method of printStream class.

Println prints the argument passed to the standard console and a new line.

Out – is a static member field of system class and is of type printStream. Its access specifier are public & final.

Q 27. What is static in java and explain?

Q 28. difference between default and private modifier?

In the same package by object creation or by inherite Private modifier is not accessible. Even in different package by object creation or by inherite Private modifier is not accessible.

On the other hand in the same package default is public But in different package default is private.

Q 29. What is immutable class and Object ? How to create customized immutable class ?

Ans) **Immutable class** is a **class** which once created, it's contents can not be changed. **Immutable** objects are the objects whose state can not be

changed once constructed. e.g. String **class**.

**To create immutable class in java, you have to do following steps.**

1. Declare the class as final so it can't be extended.
2. Make all fields private so that direct access is not allowed.
3. Don't provide setter methods for variables.
4. Make all mutable fields final so that it's value can be assigned only once.

Final class

Q 30. Is static method can be overload and override? If not why?

Ans: We **cannot override static methods**. **Static methods** are belongs to class, not belongs to object. Inheritance will not be applicable for class members.

Q 31. Why static method can not take non static variable or method

Ans: Static fields and methods are connected to the class itself and not its instances. If you have a class A, a 'normal' method b and a static method c and make an instance a of your class, the calls to A.c() and a.b() are valid. Method c() has so no idea, which instance is connected, so it cannot use non-static fields.

Q 32. Difference between method overload and override?

Compile time polymorphism, static polymorphism// Run time polymorphism or Dynamic polymorphism.

Q 33. What is the base class of Java ?

Ans : Object

Q 34. Why java can not inherit more than one class ?

Because of diamond problem /Ambiguity problem classes in java are not organized by hierarchy structure.

Class A Class B Class C

{ { {

show() show()

} } }

Now if We inherit A & B in C like Class C extends A, extends B

{

C c=new C();

}

Now we are getting show() method from both Class A & B

If we call c.show() in this case which method will response ?

This is called **ambiguity** problem.

Q 35. What is finalize method does?

Q 36. Difference between = and == sign?

Ans: You use "=" when you're assigning a value to a variable:

a = 7

== is used to comparing two objects.

Also

"==" is used to check some condition, for example:

if a == 7:

print a

which literally means "**if a equals 7**, print it"

Q 37. difference between == sign and equals method?

equals () only compare string it's does not check reference of string

but '==' check reference and data both

Q 38. Difference between equals and hashcode method?

**Ans:** Both equals() and hashCode() are defined in java.lang.Object class and there default implementation is based upon Object information e.g. default equals() method return true, if two objects are exactly same i.e. they are pointing to same memory address, while default implementation of hashcode method return int and implemented as native method.

Q 39. What is constructor ?

Q 40. Difference between method and constructor?

Q 41. Difference this() and super()?

Q 42. Difference between this and super keyword?

Q 43. What is purpose to use parent class as a datatype for creating object of child class?

Child class are getting value from the parent class or not

Because of we want to use the characteristics of parent class in the child class.

……………………………………………………………………………………………………

1. What is abstraction in Java?

**Ans:**  Abstraction in Java or Object oriented programming is a way to segregate implementation from an interface and one of the ~~five~~ fundamentals along with Encapsulation, Inheritance, Polymorphism, Class, and Object.  
2. How to divided a string?

@Test

**public** **void** divide()

{

String s="How are you";

String s1[]=s.split(" ");

**for**(**int** i=0; i<s1.length;i++)

{

System.***out***.println(s1[i]);

}

}  
3. How to reverse a string?

@Test

**public** **void** Reverse()

{

String s=**new** String("How are you");

StringBuffer sb=**new** StringBuffer(s);

sb.reverse();

System.***out***.println(sb.reverse());

}

String word = "How are you";

for (int i = (word.length() - 1); i >= 0; i--)

{

System.out.print(word.charAt(i))

}

4. How to reverse a string by word?

@Test

**public** **void** wordWiseReverse()

{

String s="How are you";

String[] sp=s.split(" ");

**for**(**int** i=sp.length-1; i>=0; i--)

{

System.***out***.print(sp[i].concat(" "));

}

}

5. How to remove special character from a string ?

@Test

**public** **void** removeSpecialCharacter()

{

String s="+\*1@2^3!4A<n>a?y%e$t~5/6";

String s1=s.replaceAll("[+\*^!@<>?%$~/]","");

{

System.***out***.println(s1);

}

}  
6. How to find duplicate character from a string ?

@Test

**public** **void** duplicateCharacterFind()

{

String s="americaww";

**char** [] c=s.toCharArray();

System.***out***.println(c[3]);

**for**(**int** i=0; i<=c.length-1; i++)

{

**for**(**int** j=i+1; j<=c.length-1; j++)

**if**(c[i]==c[j])

{

System.***out***.println(c[j]);

}

}

}

7. How to get second ( index ) of a from "united state of america"

@Test

**public** **void** removeSpecialCharacter()

{

String s="united state of america";

**int** s1=s.indexOf('a');

System.***out***.println(s1);

**int** s2=s.indexOf('a', s1+1);

System.***out***.println(s2);

}  
8. Difference between string and stringbuffer?

Ans : [See -14](Interview%20Questions%20Java.docx)  
9. Difference between stringbuffer and stringbuilder?

|  |  |
| --- | --- |
| StringBuffer | StringBuilder |
| StringBuffer is synchronized i.e. thread safe. It means two threads can't call the methods of StringBuffer simultaneously. | StringBuilder is not-synchronized i.e. not thread safe. It means two threads can call the methods of StringBuilder |
| StringBuffer is less efficient than StringBuilder. | StringBuilder is more efficient than StringBuffer. |

10 [Q.is](http://Q.is) given string is polyndrom or not ?

@Test

**public** **void** palindrome()

{

String variable ="stats"; //write a string name

StringBuffer rev = **new** StringBuffer(variable).reverse();

System.***out***.println(rev);

String strRev = rev.toString();

//System.out.println(strRev);

**if**(variable.equals(strRev)) //Check the condition

{

System.***out***.println("True");

}

**else**

{

System.***out***.println("False");

}

}  
11. What is maven? What is maven lifecycle?

\*How to make a jar file:

right click on project--->click export--->select jarfile-->give jar name--->select location–>finish

12. What are the common command use in maven?

To run maven project in command line you have to write mvn test(if you want to give option which browser will be run then you will write mvn test -Dmoga=f).

Ans: mvn test, mvn clean test, mvn compile

13. What contains pom.xml?

POM Contains -Profile , properties , all JARs, ,dependencies, all plugIn

Maven repository.com

14. What is .m2 folder?

.m2 folder is consider as local repository. And it is created automatically after first time execution of a mvn project. When a mvn project is run first time then it gets all the dependencies & necessary files from maven server/maven repository and all of these we get from the second time execution from local repository / local machine that is .m2 folder.

15. Describe page object model?

Page object model is just a design pattern not frame work.

As we are working as pages ( pages will be pure java classes )

Based on Application behavior we will be creating separate pages and will store all the locator and respective method to use them.

16. How was your cucumber framework ?

[See cucumber](../Desktop/GitRippo/Class10/target/cucumber.json)

17. How to create feature file?

Right Click on the project->New ->File….Type file name must with extention .feature.

? 18. What are the cucumber option used?

**Features:** Features Options helps Cucumber to locate the Feature file in the project folder structure.

All we need to do is to specify the folder path and Cucumber will automatically find all the ‘.features‘ extension files in the folder.

It can be defined like:

@RunWith(Cucumber.class)

@CucumberOptions( features = “Feature“)

public class RunYoursTest

{

// This class will be empty

}

Or if the feature file is in the deep folder structure-

@RunWith(Cucumber.class)

@CucumberOptions(

features = “src/test/features“

)

public class RunYoursTest

{

// This class will be empty

}

**Glue:**It is almost the same think as Features Option but the only difference is that it helps Cucumber to locate the Step Definition file.

Whenever Cucumber encounters a Step, it looks for a Step Definition inside all the files present in the folder mentioned in Glue Option.  
It can be defined like-

@RunWith(Cucumber.class)

@CucumberOptions(

features = “Feature“

glue = “stepDefinition“

)

public class RunYoursTest

{

// This class will be empty

}

Or if the Step Definition file is in the deep folder structure-

@RunWith(Cucumber.class)

@CucumberOptions(

features = “src/test/features“

glue = “src/test/stepDeinition“

)

public class RunYoursTest

{

// This class will be empty

}

Difference between scenario outline and Background key word ?

The **Scenario Outline** steps provide a template which is never directly run. A **Scenario Outline** is run once for each row in the Examples section ~~beneath it~~ (except for the first header row). The **Scenario Outline** uses placeholders, which are contained within < > in the **Scenario Outline's** steps.

**Background :** Whenever we have something common then I put them under the Background key word.

It is just like the junit annotations @Before we put all those common functionality under that annotation.

?Difference between scenario outline and Datatable ?

The **Scenario Outline** steps provide a template which is never directly run. A **Scenario Outline** is run once for each row in the Examples section ~~beneath it~~ (except for the first header row). The **Scenario Outline** uses placeholders, which are contained within < > in the **Scenario Outline's** steps.

A **Data Table**, similar to Microsoft Excel, helps the testers to create data-driven test cases that can be used to run an Action multiple times. There are two types of **Datatables** − Local **DataTable** − Each action has its own private **datatable**, also known as local **datatable**, which is can also be accessed across actions.

**[[[**Data Tables

Data Tables are handy for passing a list of values to a step definition:

Given the following users exist:

| name | email | twitter |

| Aslak | aslak@cucumber.io | @aslak\_hellesoy |

| Julien | julien@cucumber.io | @jbpros |

| Matt | matt@cucumber.io | @mattwynne |

Just like [Doc Strings](https://cucumber.io/docs/reference#doc-strings), Data Tables will be passed to the [Step Definition](https://cucumber.io/docs/reference#step-definitions) as the last argument.

The type of this argument will be DataTable. See the API docs for more details about how to access the rows and cells.**]]]**

\*\*What is hook in Cucumber ?

There are two types of hook in cucumber

1. Before - @Before- This is going to execute before scenario start
2. After -@After- This is going to execute when all the steps of a scenario done their execution.

20. How to install cucumber ?

First we have to open eclipse. Then go to help menu. Install new software , paste <http://cucumber.github.com/cucumber-eclipse/update-site> link in work with text box then next Finish.

21. What is array? is all array is object or not?

[See Java](Interview%20Questions%20Java.docx) 16.

Array is a container which hold same type of value and fixed size. When you need more than one value in a same variable .

22. How to copy one array to another array?

@Test

**public** **void** copyArray()

{

**int** a[] = { 1, 2, 3, 4, 5, 6 };

**int** b[] = **new** **int**[a.length];

**for** (**int** i = 0; i < a.length; i++)

{

a[i] = b[i];

System.***out***.println(b[i]);

}

}

23. How to get max number from an array?

@Test

**public** **void** max()

{

**int** a[]={4,76,4,8,9};

**int** max=a[0];

**for**(**int** i=1;i<a.length;i++)

{

**if**(a[i]>max)

{

max=a[i];

}

}

System.***out***.println(max);

}  
24. How to find duplicate number from an array?

@Test

**public** **void** findDuplicateValue()

{

**int** a[]={2,6,6,6,9,9,4,9,3};

**for** (**int** i=0; i<=a.length-1 ; i++)

{

//System.out.println("Value "+a[i]);

**for**(**int** j=i+1; j<=a.length-1; j++)

{

//System.out.println("2nd Value "+a[j]);

**if**(a[i]==a[j])

{

System.***out***.println(a[i]);

}

}

}

}

}  
25. How to print an array in descending order?

@Test

**public** **void** decsending()

{

**int** a[]={6,8,7,9,4};

Arrays.*sort*(a);

**for**(**int** i=a.length-1;i>=0;i--)

{

System.***out***.println(a[i]);

}

}  
26. Difference between while and do while loop?

Ans : A while statement checks condition at the beginning of a loop  to see whether the  loop will be execute or not.

A do while statement checks at the end of a loop to see whether the condition is right or not. The do while statement will always execute the body of a loop at least once.  
  
27. Difference between for loop and enhanced loop ?

1.When we know the size of a data structure . Then we can use for loop

2. For loop can be used for both incremental and decremental use.

3. Structure for For loop is

for( initialization ; condition ; increment/decrement)

{

Statement/action.

}

1.When we do not know the size / length (condition) then we should use For Each loop:

2.For Each loop structure

int a[]={4,6,5,9,8};

for(int x:a)

{

}

3. enhanced loop can not be decremental.

**28. How switch statement work ?**

When several/multiple options are available then it is not recommended to use nested if else because it reduces readability. To handle this requirement we should go for switch statement.

Switch statement is constructed by some rules . Such as

We can not use all data types in switch statement there are some restrictions also. Only the allowed argument/data types are { byte, short, char, int , String and enum} also.

Case labels: 1. It should be constant expression

2. The value should be in the range of switch argument type.

3. Duplicate case labels are not allowed.

Fall through inside a switch : ( falling down the execution)

Default case : within the switch default at most once ( i.e o times or 1 times )

Default case will only be executed if there are no cases were matches or executed. Default case can be written in anywhere within the switch. But recommended in the last position.

**Syntax**

Switch(x) allowed type 1.4 version

{

Case 1:

Action-1

Break;

Case 2:

Action-2

Break;

Case 3:

Action-3

Break;

.

.

.

.

.

.

Case n :

Action-n

Break;

Default :

Default Action

}

[[[ The allowed argument types version { byte, short, char, int } until 1.4 version but from 1.5 versions corresponding Rapper classes and enum type also allowed. From 1.7 version String types are allowed.

|  |  |  |
| --- | --- | --- |
| 1.4v | 1.5v | 1.7v |
| Byte | Byte |  |
| Short | Short |  |
| Char | Character | String |
| Int | Integer |  |
|  | Enum |  |

In switch statement { Karli braces are mandatory

}

both case & default are optional.

Also empty switch statement is valid in java.

Inside switch every statement should be under some case or defult i.e. independent statement are not allowed inside.

Every case level should be constant expression required not variable.

Both switch argument and case level can be expression but case level should be constant expression.

Duplicate expression are not allowed

except switch Karli braces are optional

Case labels: 1. It should be constant expression

2. The value should be in the range of switch argument type.

3. Duplicate case labels are not allowed.

Fall through inside a switch : ( falling down the execution)

Default case : within the switch default at most once ( i.e o times or 1 times )

Default case will only be executed if there are no cases were matches or executed. Default case can be written in anywhere within the switch. But recommended in the last case. **]]]**

29. How to handle exception?

Ans : [See -168](Interview%20Questions%20Java.docx)

There are two ways to handle exceptions,  
    
Ans:1. By wrapping the desired code in a try block followed by a catch block to catch the exceptions. and  
  
  2. List the desired exceptions in the throws clause of the method and let the caller of the method handle those exceptions.  
  
30. Difference between throw and throws clause?

Ans : See -176

Ans: **throw** keyword is used to throw our own exception in the program. **throw new** Exception\_class **throws** is to declare an exception within a method.

e.g suppose some programmer defines a method and he knows that it may causes exception, but he/she don't want to handle that exception, then this method defined with the **throws** keyword. But when we call this function, it should be call with **try** and **catch** block to maintain the normal flow of the program

31. What  are the combination in try catch and finally block?

Ans : See -117  
32. difference between final ,finally and finalize ?  
33.Can we use more than one catch block? if yes what is the sequence ?

Ans : Yes See -117

Ans : By using Try Catch block we can handle exception. Basic syntax of Try Catch block and combination of Try Catch block are explained below –  
  
**First combination:**  
  
Try  
  
{  
            Any statement;  
  
}  
  
catch ( Any Exception Class     Any variable name )  
  
{  
            syso(“print whatever you want”);  
}  
  
    **Second Combination:**  
Try  
{  
            Any statement;  
}  
  
catch (Any Exception Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
catch (Any Exception Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
.  
.  
.  
  
catch (Exception(-parent class of exception)     Any variable name)  
{  
            syso(“print whatever you want”);  
}

**Third combination:**  
  
Try  
  
{  
            Any statement;  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
catch (AnyException Class     Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
.  
.  
.  
  
catch (Exception(-parent class of exception)    Any variable name)  
  
{  
            syso(“print whatever you want”);  
}  
  
  
finally  
  
{  
            statements;  
}  
  
**Fourth combination**  
  
  
Try  
  
{  
            Any statement;  
}  
  
finally  
  
{  
            statements;  
}  
  
\*\* finally will always execute no matter what.  
**? 34. What is System.exit()?**

exit() is used to terminate the program.

System.exit(0)- 0 represents that the program ended successfully. It is a normal termination.

System.exit(1)- Any number greater than zero 0 represents program execution failed . it is the abnormal termination

Number less than zero represents program execution error. it is the abnormal termination

Java class is not maintained by hierarchy structure.

Interface can not implement a class.

add() using object

add() using index

addAll()

remove()

removeAll()

contains()

containsAll()

size()

isEmpty()

get()

set()

clear()

Environment

Docer / like Linux

Puffet

Stack

Abstraction: Abstraction is a processes that hiding the implementation details and showing some essentials details.

Step to reproduce

Test Bed.

What is Backlock

In JIRA it’s a portion /area /place . where new and incomplete ticket

throw keyword is used only for throwable objects.

We can use throws keyword to delegate responsibility of exception handling to the color (it may be another method or jvm) then color method is responsible to handle that exception.

Throws keyword required only for Checked exception.

Throws keyword required only to convence compiler.

Uses of throws keyword doesn’t prevent abnormal termination of the program.

InteruptedException.