**1.What is JVM?**

The Java interpreter along with the runtime environment required to run the Java application in called as Java virtual machine(JVM)

**2. What is the most important feature of Java?**

Java is a platform independent language.

**3. 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).

**4. 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.

**5. What is the base class of all classes?**

java.lang.Object

**6. What are the access modifiers in Java?**

There are 3 access modifiers. Public, protected and private, and the default one if no identifier is specified is called friendly, but programmer cannot specify the friendly identifier explicitly.

**7. What is are packages?**

A package is a collection of related classes and interfaces providing access protection and namespace management.

**8. What is meant by Inheritance and what are its advantages?**

 Inheritance is the process of inheriting all the features from a class. The advantages of inheritance are reusability of code and accessibility of variables and methods of the super class by subclasses.

**9. What is the difference between superclass and subclass?**

 A super class is a class that is inherited whereas sub class is a class that does the inheriting.

**10. What is an abstract class?**

An abstract class is a class designed with implementation gaps for subclasses to fill in and is deliberately incomplete.

**11. What are the states associated in the thread?**

Thread contains ready, running, waiting and dead states.

**12. What is synchronization?**

Synchronization is the mechanism that ensures that only one thread is accessed the resources at a time.

**13. What is deadlock?**

When two threads are waiting each other and can’t precede the program is said to be deadlock.

**14. What is an applet?**

Applet is a dynamic and interactive program that runs inside a web page displayed by a java capable browser

**16. How do you set security in applets?**

using setSecurityManager() method

**18. What is JDBC?**

JDBC is a set of Java API for executing SQL statements. This API consists of a set of classes and interfaces to enable programs to write pure Java Database applications.

**23. What are Encapsulation, Inheritance and Polymorphism?**

 Encapsulation is the mechanism that binds together code and data it manipulates and keeps both safe from outside interference and misuse. Inheritance is the process by which one object acquires the properties of another object. Polymorphism is the feature that allows one interface to be used for general class actions.

**24. What is the use of bin and lib in JDK?**

Bin contains all tools such as javac, appletviewer, awt tool, etc., whereas lib contains API and all packages.

**25. What is method overloading and method overriding?**

Method overloading: When a method in a class having the same method name with different arguments is said to be method overloading. Method overriding : When a method in a class having the same method name with same arguments is said to be method overriding.

**26. What is the difference between this() and super()?**

this() can be used to invoke a constructor of the same class whereas super() can be used to invoke a super class constructor.

**28. What is URL?**

URL stands for Uniform Resource Locator and it points to resource files on the Internet. URL has four components: http://www. address. com:80/index.html, where http - protocol name, address - IP address or host name, 80 - port number and index.html - file path.

**32. What are checked exceptions?**

Checked exception are those which the Java compiler forces you to catch. e.g. IOException are checked Exceptions.

**33. What are runtime exceptions?**

Runtime exceptions are those exceptions that are thrown at runtime because of either wrong input data or because of wrong business logic etc. These are not checked by the compiler at compile time.

**34. What is the difference between error and an exception?**

An error is an irrecoverable condition occurring at runtime. Such as OutOfMemory error. These JVM errors and you can not repair them at runtime. While exceptions are conditions that occur because of bad input etc. e.g. FileNotFoundException will be thrown if the specified file does not exist. Or a NullPointerException will take place if you try using a null reference. In most of the cases it is possible to recover from an exception (probably by giving user a feedback for entering proper values etc.).

**35. What is the purpose of finalization?**

The purpose of finalization is to give an unreachable object the opportunity to perform any cleanup processing before the object is garbage collected. For example, closing a opened file, closing a opened database Connection.

**38. What is mutable object and immutable object?**

If a object value is changeable then we can call it as Mutable object. (Ex., StringBuffer, …) If you are not allowed to change the value of an object, it is immutable object. (Ex., String, Integer, Float, …)

**39. What is the purpose of Void class?**

The Void class is an uninstantiable placeholder class to hold a reference to the Class object representing the primitive Java type void.

**40. What is JIT and its use?**

Really, just a very fast compiler… In this incarnation, pretty much a one-pass compiler — no offline computations. So you can’t look at the whole method, rank the expressions according to which ones are re-used the most, and then generate code. In theory terms, it’s an on-line problem.

**41. What is nested class?**

If all the methods of a inner class is static then it is a nested class.

**42. What is HashMap and Map?**

Map is Interface and Hashmap is class that implements that.

**43. What are different types of access modifiers?**

public: Any thing declared as public can be accessed from anywhere. private: Any thing declared as private can’t be seen outside of its class. protected: Any thing declared as protected can be accessed by classes in the same package and subclasses in the other packages. default modifier : Can be accessed only to classes in the same package.

**44. What is the difference between Reader/Writer and InputStream/Output Stream?**

The Reader/Writer class is character-oriented and the InputStream/OutputStream class is byte-oriented.

**46. What is Constructor?**

A constructor is a special method whose task is to initialize the object of its class.  
 It is special because its name is the same as the class name.  
They do not have return types, not even void and therefore they cannot return values.  
 They cannot be inherited, though a derived class can call the base class constructor.     
 Constructor is invoked whenever an object of its associated class is created.

**47. What is an Iterator ?**  
The Iterator interface is used to step through the elements of a Collection.  
Iterators let you process each element of a Collection.  
Iterators are a generic way to go through all the elements of a Collection no matter how it is organized.  
Iterator is an Interface implemented a different way for every Collection.

**48. What is the List interface?**

The List interface provides support for ordered collections of objects.  
Lists may contain duplicate elements.

**49. What is memory leak?**

A memory leak is where an unreferenced object that will never be used again still hangs around in memory and doesnt get garbage collected.

**50. What is the difference between the prefix and postfix forms of the ++ operator?**

The prefix form performs the increment operation and returns the value of the increment operation. The postfix form returns the current value all of the expression and then performs the increment operation on that value.

**51. What is the difference between a constructor and a method?**

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.

**52. What will happen to the Exception object after exception handling?**

Exception object will be garbage collected.

**53. Difference between static and dynamic class loading.**

Static class loading: The process of loading a class using new operator is called static class loading. Dynamic class loading: The process of loading a class at runtime is called dynamic class loading.  
Dynamic class loading can be done by using Class.forName(….).newInstance().

**55. What is JSP?**

JSP is a technology that returns dynamic content to the Web client using HTML, XML and JAVA elements. JSP page looks like a HTML page but is a servlet. It contains Presentation logic and business logic of a web application.

**56. What is the purpose of apache tomcat?**

Apache server is a standalone server that is used to test servlets and create JSP pages. It is free and open source that is integrated in the Apache web server. It is fast, reliable server to configure the applications but it is hard to install. It is a servlet container that includes tools to configure and manage the server to run the applications. It can also be configured by editing XML configuration files.

**58. Briefly explain daemon thread.**

Daemon thread is a low priority thread which runs in the background performs garbage collection operation for the java runtime system.

**59. What is a native method?**

A native method is a method that is implemented in a language other than Java.

**60. Explain different way of using thread?**

A Java thread could be implemented by using Runnable interface or by extending the Thread class. The Runnable is more advantageous, when you are going for multiple inheritance.

**62. What kind of thread is the Garbage collector thread?**

It is a daemon thread.

**63. What are the different ways to handle exceptions?**

There are two ways to handle exceptions,  
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.

**64. How many objects are created in the following piece of code?**

MyClass c1, c2, c3;  
c1 = new MyClass ();  
c3 = new MyClass ();  
Answer: Only 2 objects are created, c1 and c3. The reference c2 is only declared and not initialized.

**65.What is UNICODE?**

Unicode is used for internal representation of characters and strings and it uses 16 bits to represent each other.

‘

**What is the difference between creating String as new() and literal?**

When we create string with new() Operator, it’s created in heap and not added into string pool while String created using literal are created in String pool itself which exists in PermGen area of heap.

String s = new String("Test");  
   
does not  put the object in String pool , we need to call String.intern() method which is used to put  them into String pool explicitly. its only when you create String object as String literal e.g. String s = "Test" Java automatically put that into String pool.

**What is difference between StringBuffer and StringBuilder in Java ?**

Classic Java questions which some people thing tricky and some consider very easy. StringBuilder in Java is introduced in Java 5 and only difference between both of them is that Stringbuffer methods are synchronized while StringBuilder is non synchronized. for more differences.

**Can you write code for iterating over hashmap in Java**

**What will happen if you call return statement or System.exit on try or catch block ? will finally block execute?**

This is a very *popular tricky Java question* and its tricky because many programmer think that finally block always executed. This question challenge that concept by putting return statement in try or catch block or calling System.exit from try or catch block. Answer of this tricky question in Java is that finally block will execute even if you put return statement in try block or catch block but finally block won't run if you call System.exit form try or catch.

**Can you override private or static method in Java ?**

Another popular Java tricky question, As I said method overriding is a good topic to ask trick questions in Java.  Anyway, you can not override private or static method in Java, if you create similar method with same return type and same method arguments that's called method hiding.

**What will happen if we put a key object in a HashMap which is already there ?**

This tricky Java questions is part of How HashMap works in Java, which is also a popular topic to create confusing and tricky question in Java. well if you put the same key again than it will replace the old mapping because HashMap doesn't allow duplicate keys.

**If a method throws NullPointerException in super class, can we override it with a method which throws RuntimeException?**

One more tricky Java questions from overloading and overriding concept. Answer is you can very well throw super class of RuntimeException in overridden method but you can not do same if its checked Exception

**Can you access non static variable in static context?**

Another tricky Java question from Java fundamentals. No you can not access static variable in non static context in Java.

**What if the main method is declared as private?**

The program compiles properly but at runtime it will give “Main method not public.” message.

**What is meant by pass by reference and pass by value in Java?**

Pass by reference means, passing the address itself rather than passing the value. Pass by value means passing a copy of the value.

**What is Byte Code?**

**Or**

**What gives java it’s “write once and run anywhere” nature?**

All Java programs are compiled into class files that contain bytecodes. These byte codes can be run in any platform and hence java is said to be platform independent.

**Expain the reason for each keyword of public static void main(String args[])?**

public- main(..) is the first method called by java environment when a program is executed so it has to accessible from java environment. Hence the access specifier has to be public.

static: Java environment should be able to call this method without creating an instance of the class , so this method must be declared as static.

void: main does not return anything so the return type must be void

The argument String indicates the argument type which is given at the command line and arg is an array for string given during command line.

**What are the differences between == and .equals() ?**

**Or**

**what is difference between == and equals**

**Or**

**Difference between == and equals method**

**Or**

**What would you use to compare two String variables – the operator == or the method equals()?**

**Or**

**How is it possible for two String objects with identical values not to be equal under the == operator?**

The == operator compares two objects to determine if they are the same object in memory i.e. present in the same memory location. It is possible for two String objects to have the same value, but located in different areas of memory.

== compares references while .equals compares contents. The method public boolean equals(Object obj) is provided by the Object class and can be overridden. The default implementation returns true only if the object is compared with itself, which is equivalent to the equality operator == being used to compare aliases to the object. String, BitSet, Date, and File override the equals() method. For two String objects, value equality means that they contain the same character sequence. For the Wrapper classes, value equality means that the primitive values are equal.

**How to convert String to Number in java program?**

The valueOf() function of Integer class is is used to convert string to Number. Here is the code example:  
String numString = “1000″;  
int id=Integer.valueOf(numString).intValue();

**What is the difference between a while statement and a do statement?**

A while statement (pre test) checks at the beginning of a loop to see whether the next loop iteration should occur. A do while statement (post test) checks at the end of a loop to see whether the next iteration of a loop should occur. The do statement will always execute the loop body at least once.

**Describe the principles of OOPS.**

There are three main principals of oops which are called Polymorphism, Inheritance and Encapsulation.

**Explain the Inheritance principle.**

Inheritance is the process by which one object acquires the properties of another object. Inheritance allows well-tested procedures to be reused and enables changes to make once and have effect in all relevant places

**What is implicit casting?**

Implicit casting is the process of simply assigning one entity to another without any transformation guidance to the compiler. This type of casting is not permitted in all kinds of transformations and may not work for all scenarios.

Example

int i = 1000;

long j = i; //Implicit casting

**What is explicit casting?**

Explicit casting in the process in which the complier are specifically informed to about transforming the object.

Example

long i = 700.20;

int j = (int) i; //Explicit casting

**When is static variable loaded? Is it at compile time or runtime? When exactly a static block is loaded in Java?**

Static variable are loaded when classloader brings the class to the JVM. It is not necessary that an object has to be created. Static variables will be allocated memory space when they have been loaded. The code in a static block is loaded/executed only once i.e. when the class is first initialized. A class can have any number of static blocks. Static block is not member of a class, they do not have a return statement and they cannot be called directly. Cannot contain this or super. They are primarily used to initialize static fields.

**Can I have multiple main methods in the same class?**

We can have multiple overloaded main methods but there can be **only one main method with the following signature :**

public static void main(String[] args) {}

No the program fails to compile. The compiler says that the main method is already defined in the class.

Top 10 Java Coding Questions For Test Automation Developers.

Question-1: Write Code To Filter Duplicate Elements From An Array And Print As A List?

package simple.test;

import java.util.ArrayList;

import java.util.HashSet;

import java.util.List;

import java.util.Set;

public class findDuplicates {

public static void main(String[] args) {

ArrayList<String> list = new ArrayList<String>();

// Form a list of numbers from 0-9.

for (int i = 0; i < 10; i++) {

list.add(String.valueOf(i));

}

// Insert a new set of numbers from 0-5.

for (int i = 0; i < 5; i++) {

list.add(String.valueOf(i));

}

System.out.println("Input list : " + list);

System.out.println("\nFiltered duplicates : " + processList(list));

}

public static Set<String> processList(List<String> listContainingDuplicates) {

final Set<String> resultSet = new HashSet<String>();

final Set<String> tempSet = new HashSet<String>();

for (String yourInt : listContainingDuplicates) {

if (!tempSet.add(yourInt)) {

resultSet.add(yourInt);

}

}

return resultSet;

}

}

Question-2: Write Code To Sort The List Of Strings Using Java Collection?

package simple.test;

import java.util.Arrays;

public class sortStrings {

public static void main(String[] args) throws Exception {

String[] inputList = { "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul",

"aug", "Sep", "Oct", "nov", "Dec" };

// Display input un-sorted list.

System.out.println("-------Input List-------");

showList(inputList);

// Call to sort the input list.

Arrays.sort(inputList);

// Display the sorted list.

System.out.println("\n-------Sorted List-------");

showList(inputList);

// Call to sort the input list in case-sensitive order.

System.out.println("\n-------Sorted list (Case-Sensitive)-------");

Arrays.sort(inputList, String.CASE\_INSENSITIVE\_ORDER);

// Display the sorted list.

showList(inputList);

}

public static void showList(String[] array) {

for (String str : array) {

System.out.print(str + " ");

}

System.out.println();

}

}

Question-3: Write A Function To Reverse A Number In Java?

package simple.test;

public class invertNumber {

public long doInvert(long number) {

long invert = 0;

while (number != 0) {

invert = (invert \* 10) + (number % 10);

number = number / 10;

}

return invert;

}

public static void main(String args[]) {

long lnum = 654321;

invertNumber input = new invertNumber();

System.out.println("Input value : " + lnum);

System.out.println("Inverted value : " + input.doInvert(lnum));

}

}

Further Reading:

💡 Top 20 Selenium Coding Tips for Software Testers.

Question-4: Write A Method To Check Prime No. In Java?

package simple.test;

import java.util.Scanner;

public class findPrime {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.print("Enter an int value : ");

int input = scan.nextInt();

if (checkPrime(input)) {

System.out.println("Input value " + input + " is a prime number.");

} else {

System.out.println("Input value " + input

+ " is not a prime number.");

}

}

public static boolean checkPrime(int n) {

if (n <= 1) {

return false;

}

for (int i = 2; i < Math.sqrt(n); i++) {

if (n % i == 0) {

return false;

}

}

return true;

}

}

Question-5: Write A Java Program To Find Out The First Two Max Values From An Array?

package simple.test;

public class findTwoMaxValue {

public void GetTwoMaxValues(int[] nums) {

int maxOne = 0;

int maxTwo = 0;

for (int n : nums) {

if (maxOne < n) {

maxTwo = maxOne;

maxOne = n;

} else if (maxTwo < n) {

maxTwo = n;

}

}

System.out.println("Max1 - " + maxOne);

System.out.println("Max2 - " + maxTwo);

}

public static void main(String[] args) {

int list[] = { 15, 24, 48, 21, 43, 11, 79, 93 };

findTwoMaxValue max = new findTwoMaxValue();

max.GetTwoMaxValues(list);

}

}

Question-6: Write A Java Program To Find The Longest Substring From A Given String Which Doesn’t Contain Any Duplicate Characters?

package simple.test;

import java.util.HashSet;

import java.util.Set;

public class findSubstr {

private Set<String> stringSet = new HashSet<String>();

private int lstringSet = 0;

public Set<String> findStr(String input) {

// Reset instance data.

stringSet.clear();

lstringSet = 0;

// Set a boolean flag on each char's ASCII value.

boolean[] flag = new boolean[256];

int j = 0;

char[] inputCharArr = input.toCharArray();

for (int i = 0; i < inputCharArr.length; i++) {

char c = inputCharArr[i];

if (flag[c]) {

extractSubString(inputCharArr, j, i);

for (int k = j; k < i; k++) {

if (inputCharArr[k] == c) {

j = k + 1;

break;

}

flag[inputCharArr[k]] = false;

}

} else {

flag[c] = true;

}

}

extractSubString(inputCharArr, j, inputCharArr.length);

return stringSet;

}

private String extractSubString(char[] inputArr, int start, int end) {

StringBuilder sb = new StringBuilder();

for (int i = start; i < end; i++) {

sb.append(inputArr[i]);

}

String subStr = sb.toString();

if (subStr.length() > lstringSet) {

lstringSet = subStr.length();

stringSet.clear();

stringSet.add(subStr);

} else if (subStr.length() == lstringSet) {

stringSet.add(subStr);

}

return sb.toString();

}

public static void main(String a[]) {

findSubstr substr = new findSubstr();

System.out

.println("Actual Strings ------------ | ---- Longest Non-Repeated Strings");

System.out.println("Software\_Programmer"

+ " | " + substr.findStr("Software\_Programmer"));

System.out.println("Software\_Developer\_In\_Test"

+ " | " + substr.findStr("Software\_Developer\_In\_Test"));

System.out.println("developers\_write\_unit\_tests"

+ " | " + substr.findStr("developers\_write\_unit\_tests"));

System.out.println("javajavbasp.net"

+ " | " + substr.findStr("javajavbasp.net"));

}

}

Question-7: Write Java Code To Get Rid Of Multiple Spaces From A String?

package simple.test;

import java.util.StringTokenizer;

public class removeExtraSpaces {

public static void main(String args[]){

String input = "Try to remove extra spaces.";

StringTokenizer substr = new StringTokenizer(input, " ");

StringBuffer sb = new StringBuffer();

while(substr.hasMoreElements()){

sb.append(substr.nextElement()).append(" ");

}

System.out.println("Actual string: " + input);

System.out.println("Processed string: " + sb.toString().trim());

}

}

Question-8: Write Java Code To Identify A Number As Palindrome?

package simple.test;

import java.io.BufferedReader;

import java.io.InputStreamReader;

public class identifyPalindrome {

public static void main(String[] args) {

try {

BufferedReader object = new BufferedReader(new InputStreamReader(

System.in));

System.out.println("Input number");

int inputValue = Integer.parseInt(object.readLine());

int n = inputValue;

int rev = 0;

System.out.println("Input value is : ");

System.out.println(" " + inputValue);

for (int i = 0; i <= inputValue; i++) {

int r = inputValue % 10;

inputValue = inputValue / 10;

rev = rev \* 10 + r;

i = 0;

}

System.out.println("Post reversal : " + " ");

System.out.println(" " + rev);

if (n == rev) {

System.out.print("Input value is a palindrome.");

} else {

System.out.println("Input value is not a palindrome.");

}

} catch (Exception e) {

System.out.println("Out of Range.");

}

}

}

Question-9: Write Java Code To Swap Two Numbers Without Using A Temporary Variable?

package simple.test;

public class smartSwapping {

public static void main(String args[]) {

int numX = 10;

int numY = 20;

System.out.println("Pre-swapping state:");

System.out.println("numX value: " + numX);

System.out.println("numY value: " + numY);

System.out.println("");

numX = numX + numY;

numY = numX - numY;

numX = numX - numY;

System.out.println("Post-swapping state:");

System.out.println("numX value: " + numX);

System.out.println("numY value: " + numY);

}

}

Question-10: Write A Java Program To Demonstrate String Reverse With And Without StringBuffer Class?

package simple.test;

public class invertString {

public String invertWithStringBuffer(String str) {

StringBuffer buffer = new StringBuffer(str);

buffer.reverse();

return buffer.toString();

}

public String invertWithoutStringBuffer(String str) {

int length = str.length();

String original = str;

String invert = "";

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

invert = invert + original.charAt(i);

}

return invert;

}

public static void main(String[] args) {

invertString invertStr = new invertString();

System.out.println("Inverted String with StringBuffer class: "

+ invertStr.invertWithStringBuffer("987654321"));

System.out.println("");

System.out.println("Inverted String without StringBuffer class: "

+ invertStr.invertWithoutStringBuffer("kjihgfedcba"));

}

}