

ANSWER ALL THE QUESTIONS

1) Explain about main method in Java

Main method is the entry point of any standalone Java application. The syntax is public static void main (String args []). Main method is public and static so that Java can access it without initializing the class. The input parameter is an array of String through which we can pass runtime arguments to the Java program.

2. Public: public is an access modifier which is used to specify who can access this method. Public means that this method will be accessible by any class.

Static: a keyword in Java which identifies it is class based i.e; it can be accessed without creating instance of a class.

Void: is a return type of method, Void defines the method which will not return any value.

Main: It is the name of the method which is searched by JVM as a starting point for an application with particular signature only. It is the method where main execution occur.

`String args []`: a parameter passed to the main method.

2. What are different control flow statements available in Java.

The Control flow statement in Java allow you to run a block of code when special condition was met.

(i) If Statement

If Statement in Java works exactly like in most programming languages. With you can choose to execute a specific block of code when predefined condition is met. The structure of if statement in Java looks like this

```
If {  
    Execution Code  
}
```

(ii) If else Statement

With this you can control what to do if condition is met and what to do otherwise.

```
If {  
    Execution Code  
}  
else {  
    Execution Code  
}
```

(iii) The Switch Statement

In Some Cases you can avoid Using
Multiple if in Code and make look better

Switch (Variable) {

Case Value 1

Statements

break;

Case value 2

Statements

break;

default (Optional)

Statement

break;

}

Q. What is the difference between break and Continue Statement.

Both break and Continue are the jump Statement, that transfer Control of Program to another Part of Program. Java Supports three Jump statements "break", "Continue", "Return". The main difference between break and Continue is of loop whereas Continue terminate Current Iteration and resumes the Control to next Iteration of loop.

Break:

- * It terminates the execution of remaining iteration of the loop.
- * break resumes the control of the program to the end of loop enclosing that break.
- * It causes only termination of loop.
- * break can be used with switch, & label.

Continue:

- * It terminates only the current iteration of loop.
- * Only stops the current iteration.
- * Continue can not be executed with switch() and label.

A) What is an array? How will you declare an array in Java?

- (a) Array is a collection of similar data types.
- (b) It cannot have different data type, it can hold both primitive types int (float, double) and object reference.
- (c) It is fixed in length i.e., static in nature.
- (d) Arrays are created on heap memory and not on stacks.

You can declare array in Java by the following way.

datatype[] array Variable name = new datatype [array].

For example for Int datatype you can declare an Int array as.

Int [] temp = new Int [276]

5. When will you get ArrayIndex Out of Bounds Exception?

ArrayOut of BoundException is thrown when an attempt is made to access the array with illegal index. For example, illegal index means if the index is either negative or greater than or equal to size of the array.

Public class Exception Example {

 Public static void main (String args []) {

 Int [] rollnumber = {23, 17, 20, 29, 30};

* Index below is greater than size of given array*

 Int element = roll number [6];

 System.out.println (element);

}

6. Define the syntax to create an object for a class.
What are the naming conventions to be followed while creating a class, method and a variable.
Explain with examples.

Create an object for a class:

<classname> reference name = new <constructorname();>;

Name convention:

Classname: should start with uppercase letter and be a noun eg:- String, Color, Button, System, Thread etc.,

Method Name: should start with lowercase letter and be a verb eg:- Main(), getText(), println() etc.,

Variable Name: should start with lowercase letter
eg: firstname, lastname etc.,

Ex: ChromeDriver driver = new ChromeDriver();

7) What is variable? How will you declare a variables in java?

A Variable is a container which holds the Value while the Java program is executed. A Variable is assigned with a data type. There are three types of variables in java:

local, instance and static

A variable is declared as below

public String name;

public int age;

A variable is initialized as below:

```
public String name = "Test Leaf";  
public String age = "10";
```

8. What is String in java? Is it Data Type?

String is a Java class (part of JDK) and it is not data type like int, boolean, char etc. It is a sequence of characters and enclosed with the double quotes ("").

For example:

```
String txt = "Test Leaf";
```

9. What are the different ways to create the string object in java?

There are two ways to create the string object by string literal and by new keyword.

i) String as literal like:

```
String s = "Test Leaf";
```

ii) String as Object like:

```
String s = new String ("Test Leaf");
```

10. What is the Difference between .Equals and ==

Both equals() and "==" operator in java are used to compare objects to check equality.

But there are certain differences between them.

- a) `.equals()` is a method and `=` is an operator.
- b) `=` operator is used for reference comparison (address comparison) and `.equals()` method for content comparison. i.e., `=` checks if both objects point to the same memory location whereas `.equals()` evaluates to the comparison of values in the objects.
- c) If a class does not override the `equals` method then by default it uses `equals(Object o)` method of the closest parent class that has overridden this method.

Eg:

```
public class Test {  
    public static void main(String[] args)  
    {  
        String s1 = new String("Test Leaf");  
        String s2 = new String("Test Leaf");  
        System.out.println(s1 == s2);  
        System.out.println(s1.equals(s2));  
    }  
}
```

Output:

false
true