==========JAVA Questions==========

Q1/-What is a function?

-A function is a self-defined block which is used for calculation, printing the result or to perform some specific operation.

Q2/- Why we pass argument/parameter to the function?

- We pass parameter to a function for getting more information regarding that function. If we don't pass parameter then it may represent partial information.

Q3/- Why functions are called Method in java?

- In some languages, we have a support to write a function inside a class as well as outside of the class.
- If we write a function inside a class then it is called Method.
- In java language, We can't write a function outside of the class, It must be inside the class only that the reason functions are Method in java .

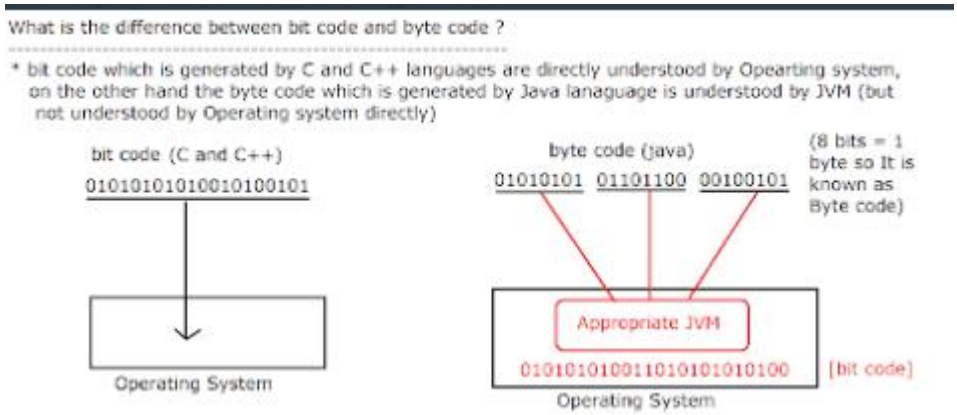
Q4/- What is platform independency in java?

- C and C++ programs are platform dependent programs that means the .exe file created on one machine will not be executed on another machine if the system configuration is different.
- That is the reason C and C++ programs are not suitable for website development.
- The role of java compiler :
- ---
- 1) Syntax verification.
- 2) Verify the compatibility issues (L.H.S = R.H.S)
- 3) Will Convert Source code into byte code.
- Java is a platform independent language. Whenever we write a java program, the extension of java program must be .java.
- Now this .java file we submit to java compiler (javac) for compilation process. After successful compilation the compiler will generate a very special byte code file i.e .class file (also known as bytecode). Now this .class file we submit to JVM for execution purpose.
- The role of JVM is to load and execute the .class file. Here JVM plays a major role because It converts the .class file into appropriate machine code instruction (Operating System format) so java becomes platform independent language and it is highly suitable for website development.

Q4/-JVM platform independency or dependency and why in javaf?

- We have different JVM for different Operating System that means JVM is platform dependent technology where as Java is platform independent technology.
- JVM internally contains an interpreter so it executes the code line by line. It is written in 'C' langugae hance platform dependent.
- Note : All the browsers internally contain JVM are known as JEB (Java Enabled Browsers) browser.

Q5/-What is the difference between bit code and byte code?



- bit code which is generated by C and C++ languages are directly understood by Opearting system, on the other hand the byte code which is generated by Java lanaguage is understood by JVM (but not understood by Operating system directly).

Q6/-What is the difference between JDK, JRE, JVM and JIT compiler?

- It stands for Java Development kit. Internally It contains JRE and JVM. It is a developer version that means by using JDK we can develop and execute our java program.
- In order to develop and execute the java program, It provides so many JDK tools which are as follows :
- a) javac : Java compiler, responsible for compile our java program.
- b) java : Java launcher, responsible for executing the java program.
- c) jdb : java debugger, Used to debug the code internally
- d) javadoc : Java documentation, Used to generate documentation
- e) javap : Java profiler, Used to get the details of .class file
- f) jconsole : Java Console, used to display the output in the console
-

- Note : If we want to verify the version of java which is installed in my system then we can use the following command :

- java -version

JRE

- JRE stands for (Java Runtime Environment) Internally it contains JVM + class libraries. It is a client version that means by using JRE we can execute (we nm
- This JRE folder is not available from java 11V so, from java 11V we can directly execute our java program by using the following command(No need to compile)
- java FileName.java [We can execute Directly]

JVM

- It stands for Java Virtual Machine. The main purpose of JVM to load and execute the .class file. JVM plays a major role because It converts the .class file into appropriate machine code instruction (byte code to bit code). JVM is also responsible to load the .class file, Verify .class file, execute the code line by line by using Interpreter.

JIT COMPILER

- As we know, Our interpreter is slow in nature so to boost up the java execution, we have JIT (Just In time) compiler support.
- It holds the repeated code instruction and native code instruction, It will directly provide these two instrution at time
- of line by line execution so our interpreter executes the code
- in more efficient way hance the overall execution becomes very fast.

Q7/-What is the difference between Compiler and Interpreter

What is the difference between compiler (javac) and Interpreter (JVM) :

Compiler	JVM
1) Scans the entire program once.	1) Scans the program line by line.
2) It will display all the errors and warnings at a time.	2) It will display the runtime error one by one not at a time.
3) Debugging is difficult.	3) Debugging is easy.
4) It generates byte code after successful compilation so to hold the byte code, separate memory is reqd	4) It does not generate any other file, Actually It will execute the code line by line and concurrently It will generate the output so separate memory is not reqd
5) After successful compilation, we can delete the .java file.	5) We can't delete .class file due to line by line execution
6) After successful compilation, Execution is fast	6) Interpreter is slow in execution because if we make a mistake at line number 5 then after resolving the issue it will again execute from line number 1
7) C, C++, Java, C# and so on, these languages are using compiler	7) Java, JavaScript, Python, Visual basic these languages are using Interpreter.

Q8/- What is the difference between statically typed and dynamically typed language ?

- statically typed language :
- ---
- The languages where data type is compulsory before initialization of a variable are called Statically typed language.
- In these languages, once we define the type of the variable then it will hold same kind of value till the end of the program.
- Example of statically typed languages :
- C,C++, JSE, C# and so on.
- Dynamically typed language :
- ---
- The languages where Data type is not compulsory, It is optional to initialize the variable are called Dynamically Typed Language.
- In these languages we can provide all different kinds of values to the variable during the execution of the program.
- Example of dynamically typed languages :
- Visual Basic, Javascript, Python and so on

Q9/- Flavors of Java and What are those?

- In java, We have total 4 flavors :
- 1) JSE (Java Standard Edition) [Stand-alone Application]
- 2) JEE (Java Enterprise Edition) [Websites OR web application]
- 3) JME (Java Micro Edition) [Mobile Apps OR Android Appln]
- 4) JavaFX (It is used to Develop GUI Application) [outdated]

Q10/- What is Comments in Java ?

- Comments are used to enhance the readability of the code. It is ignored by the Compiler.

Q11/- Description of main() method ?

- public :
- It is an access modifier in java which defines the accessibility level of main method. Our main method must be declared as public otherwise JVM can't access our main method so the execution of the program will not be started.
- If we don't declare our main method with public access modifier then code will compile but it will not be executed by JVM.

- static :
- As of now, We have only two types of methods in java.
- 1) static method (Object is not required)
- 2) non static method (Object is required)
- If we declare any method with static keyword then it is called static method. In order to call static method, Object is not required. We can call the static method by using the following two approaches
- Case 1 :
- ---
- If a static method is declared in the same class where main method is available then we can directly call the static method with the help of method name as shown in the program.
-
- public class StaticMethodDemo1
- {
- public static void main (String [] args)

```
    {
        System.out.println("Hello Batch 42 and 43!");
        m1();
    }

    public static void m1() //static Method
    {
        System.out.println("M1 static Method");
    }
}
```

Case 2 :

If a static method is declared in another class (not in the class where main method is available) then to call the static method class name is required that means we can't call the static method directly as shown in the program below :

```
class Hello
{
    public static void greet()
    {
        System.out.println("Hello Everyone!!!");
    }
}

public class StaticMethodDemo2
{
    public static void main(String[] args)
    {
        Hello.greet();
        System.out.println("Hello Batch 42 and 43!!!!");
    }
}
```

- Our main method must be declared as static so, JVM will call the main method with the help of class name because main is a static method.
-
- If we don't declare our main method as static then code will compile but it will not be executed by JVM.

void :

- It is a keyword in java. We should write void before the name of the method so that particular method will not return any kind of value.
- If we put the return type of the method as void then there is no communication between one module to another module.[It is one way communication]
- If we don't write void or any other kind of return type before the main method then code WILL NOT compile because java compiler has provided syntax rule that every method must contain return type.
- Note : If main method is not declared with void then program will NOT BE executed.
- Note : Without return type we can't define a method in java [Syntax Rule]

main() method :

- It is a user-defined method because a user is responsible to define some logic inside the main method.
- main() method is very important method because every program execution will start from main() method only, as well as the execution of the program ends with main() method only.
- String [] args :
- ---
- String is a predefined class in java available in java.lang package (just like header file) and args is an array variable
- of type String so, it can hold multiple values.

System.out.println() :

- It is an output statement in java, By using this statement we can print different types of values on the console.
- In this statement System is a predefined class available in java.lang pacakage.
- out is a reference variable of type PrintStream class available in java.io package.
- println() is a predefined method available in PrintStream class.

Q12/ Can we write multiple main method with same name ?

- Yes, We can write multiple methods with same name but parameter must be different otherwise code will not compile.
- Note :- We can also write multiple main methods with different parameter but JVM will always execute the main method which takes String [] args (String array) as a parameter as shown in the program below.

Q13/-Why the main method of java accepts String array as a parameter ?

- String is a collection of alpha-numeric character so it can accept all different kind of values. Java software people has
- provided String array as a parameter so it can ACCEPT MULTIPLE VALUES OF DIFFERENT TYPE, that means providing more wider scope to accept hetrogeneous types of values.

Q14/-What is CommandLine Argument in java?

- If we pass any argument to the main method then it is called Command Line Argument.
- By using command line argument, We can pass some value at runtime.

Q15/-How to convert a String value into corresponding integer type ?

- If we want to convert any String value into corresponding int type then Java software people has provided a predefined class called Integer class available in java.lang package.
- This Integer class contains a predefined static method called parseInt(String x) through which we can convert a String value into corresponding int type.

- int p = Integer.parseInt(args[0]);

Q16/-How to write a final and static variable ?

- While writing the final and static variable we should follow snake_case naming convention.
- Example :
- Integer.MIN_VALUE [MIN_VALUE is final and static variable]
- Integer.MAX_VALUE [MAX_VALUE is final and static variable]

Q17/- What is Tokens In java and types ?

- Token is the smallest unit of the program which is identified by the compiler.
- Without token we can't complete statement or expression in java.
- Token is divided into 5 types in java
- 1) Keyword
- 2) Identifier
- 3) Literal
- 4) Punctuators (Seperators)
- 5) Operator

- Keyword
- A keyword is a predefined word whose meaning is already defined by the compiler.
- In java all the keywords must be in lowercase only.
- A keyword we can't use as a name of the variable, name of the class or name of the method.
- true, false and null look like keywords but actually they are literals.
- As of now, we have 67+ keywords in java.

Identifiers :

- A name in java program by default considered as identifiers

- Assigned to variable, method, classes to uniquely identify them.
- We can't use keyword as an identifier.

Ex:-

```
class Fan
{
  int coil  ;

  void switchOn()
  {
  }
}
```

Here Fan(Name of the class), coil (Name of the variable) and switchOn(Name of the Method) are identifiers.

Rules for defining an identifier :

---

1) Can consist of uppercase(A-Z), lowercase(a-z), digits(0-9), $ sign, and underscore (_)
2) Begins with letter, $, and _
3) It is case sensitive
4) Cannot be a keyword
5) No limitation of length

Literals:

Any constant which we are assigning to variable is called Literal.
In java we have 5 types of Literals:
1) Integral Literal
2) Floating Point Literal
3) Boolean Literal
4) Character Literal
5) String Literal

Q18/- Is java pure Object Oriented language or not ?

- No, Java is not a pure object oriented langauge because it is accepting primary data type, Actually any language which accepts primary data type is not a pure object oriented language.
-
- Only Objects are moving in the network but not the primary data type so java has introduced Wrapper class concept to convert the primary data types into corresponding wrapper object.
-
- Primary Data type Wrapper Object
- byte : Byte
- short : Short
- int : Integer
- long : Long
- float : Float
- double : Double
- char : Character
- boolean : Boolean
-
-
- Note : Apart from these 8 data types, Everything is an object in java so, if we remove all these 8 primitive data types then java will become pure OOP language.

Q18/-How to find out the minimum value, maximum value as well as size of different data types:

- The Warpper classes like Byte, Short, Integer and Long has provided predefined static and final variables to represent minimum value, maximum value as well as size of the respective data type.

```
        System.out.println(Byte.MIN_VALUE);
        System.out.println(Byte.MAX_VALUE);
        System.out.println(Byte.SIZE);
        System.out.println("------------------------------");

        System.out.println(Short.MIN_VALUE);
        System.out.println(Short.MAX_VALUE);
        System.out.println(Short.SIZE);
        System.out.println("------------------------------");

        System.out.println(Integer.MIN_VALUE);
        System.out.println(Integer.MAX_VALUE);
        System.out.println(Integer.SIZE);
        System.out.println("------------------------------");

        System.out.println(Long.MIN_VALUE);
        System.out.println(Long.MAX_VALUE);
        System.out.println(Long.SIZE);
        System.out.println("------------------------------");

        System.out.println(Double.MIN_VALUE);
        System.out.println(Double.MAX_VALUE);
        System.out.println(Double.SIZE);
```

O/P

```
-----------
 f-128
127
8------------------------------
-32768
32767
16
------------------------------
-2147483648
2147483647
32
------------------------------
-9223372036854775808
9223372036854775807
64
------------------------------
4.9E-324
1.7976931348623157E308
64
```

Q19/-Integer class has provided the following static methods to convert decimal to octal, hexadecimal and binary.

- 1) public static String toBinaryString(int x) : Will convert the decimal into binary in String format.
- 2) public static String toOctalString(int x) : Will convert the decimal into octal in String format.
- 3) public static String toHexString(int x) : Will convert the decimal into hexadecimal in String format.

```
int x = 15;
        //Decimal to octal
        System.out.println(Integer.toOctalString(x));

        int y = 2781;
        //decimal to hexadecimal
        System.out.println(Integer.toHexString(y));

        int z = 5;
        //decimal to Binary
        System.out.println(Integer.toBinaryString(z));
```

Q20/- What is var keyword in java ?

- Java has provided var keyword from java 10V. By using var keyword we can directly assign any type of value to that particular variable.
- Using var keyword, Initialization is compulsory in the same line (Compiler will get the information of variable type based on the value).
- var keyword is allowed only for local variable.
- EX__
- var x = 100; //Initialization is compulsory here only
-
- //x = "NIT"; //Invalid
-
- x = 900;

Q21/- What Is Local Variable ?

- If a variable is declared inside the body of method OR block OR Constructor then it is called Local /Automatic/ Stack/ temporary variable.
- Example :
- ---------
- public void accept()
- {
- int x = 100; //x is a local variable
- }
-
- A local variable must be initialized by the developer before
- use because local variable does not have any default value.
- We can't apply any kind of modifier on local variable except final.
- public void accept()
- {
- final int x = 100; //final is a valid modifier
- }
- As far as it's scope is concerned, It must be used within the same method body only that menas we can't use local variable outside of the method body/block/constructor.
- All local variables are the part of the method body, all the methods are executed in a special memory in java called Stack Memory so local variables are the part of Stack memory.
- A local variable must be pre-declared and initialized before use.
- public void m1()
- {
- System.out.println(x); //error
- int x = 100;
- }

Q22/- Why we can't use local variables outside of the method OR body OR Constructor ?

- In java, Every methods are executed in a special memory called
- Stack Memory.
- Stack Memory works on LIFO (Last In First Out) basis.
- In java, Whenever we call a method then a separate Stack Frame will be created for each and every method.[03-FEB]
- Once the method execution is over then the corresponding method Stack frame will also be deleted from Stack Area, that is the reason we can't use local variable outside of the method.
- Each stack frame contains 3 parts :
- 1) Local Variable Array
- 2) Frame Data
- 3) Operand Stack

Q23/-How CharAt Is Works ?

- Note : charAt(int indexPosition) is a predefined non static method of String class, which is used to retrieve a character from the given String.

Q24/-JAVA FEATURES ?

- Features Of java :

  What are important features of Java ?
  Java: mainly its open source i.e., its free of cost to use java to develop applications.
  Till Java 10 it was completely free of cost but, from Java 11 onwards its both open source as well as licensed.

  Now we have:
  i. Open JDK
  ii.Oracle JDK

1. Simple:Its easy and simple to learn and develop projects.

Because all the concepts used in java are simple as the syntaxes used in java are easy to learn.

It doesnot use lots of confusing concepts
such as Pointers(used in C),multiple inheritance(C++) etc.
And above to all, it has lots of predefined libraries due to which development of project becomes faster and easy.

2. Secured: There is a special feature inbuilt i.e., Byte Code Verifier in JVM. Its responsibility is to check whether the code contains any malicious code or not. If it finds anything suspicious then the code is not allowed to be exceuted.Due to this the system is safe.

It also provides access modifier 'private' , which when used with an element, the element will not be allowed to access outside the class.

3. Object Oriented: Java is a class based program which means every program will be written by starting the program with class as the programs
in java are basically for representing real world objects in programming
world.And its possible using class.A class represents the blueprint of objects.

4. Byte Coded: Java programs are compiled to achieve platform independent
type apps.

5. Interpreted: Byte codes are generally dependent on the current OS but
interpreter takes it and generates native code for OS to perform action on commands present in Byte codes.
AS like java other PL are also interpreted PL.such as
C#.NET, python, javascript, PHP etc.

6. Portable(Platfrom Independent): Application of java developed in one
platform can work in other platforms also.We dont have to rewrite the
complete source code, we just need the byte code to be shared.

7. MultiThreaded: It supports to develop applications which can handle
multiple tasks parallely at a time.In java each task can have its
separate thread to complete its execution independent of any other tasks.

8. Robust(Strong): It has a fail safe system, which means whenever an
abnormal or unexpected condition is arised then the program terminates
only after saving the given data and there is no loss of data.

C is not type checked, but java is strictly type checked.
You can't have loss of data.
Exception handling will support in making the application continue
the execution even after any exception may arise.

9.High Performance: Java program execution is faster because of multithreading
and a special feature from Java 1.3 i.e JIT Compiler.

10.Distributed: We can develop applications which can be accessible through
worldwide remotely. It have special technologies such as :
RMI,XML,EJB,WebServices,REST API,JSONs.....to be conitnued in Adv.java

11. Dynamic: We can add new features to application without affecting theold features.
Beacuse of OOP concepts:

1. Inheritance
2. Encapsulation
3. Abstraction
4. Polymorphism.

12.Architecture Neutral: Whatever result you get in one platform the same you can get in another platform also(memory size is fixed for all OS and processors).
In C 'int' has different size from one OS to another OS.
but in java its always same in all OS.

13. Garbage Collected: It manages the memory by clearing the unused objects.We dont have to destroy objects, they are automatically destroyed by JVM,by using one of its resource called as Garbage Collector.

Q26/-How to convert values predefine methods ?

```
-    public static void main(String[] args) {
-        //  DECIMAL TO OCTAL
-        int decimal = 10;
-        String octal = Integer.toOctalString(decimal);
-        }
-    }
```

Q27/-What are the Non static methods in scanner class

- 1) public String next() : Used to read a single word.
- 2) public String nextLine() : Used to read a multiple words or complete line.
- 3) public byte nextByte() : Used to read byte value.
- 4) public short nextShort() : Used to read short value.
- 5) public int nextInt() : Used to read int value.
- 6) public long nextLong() : Used to read long value.
- 7) public float nextFloat() : Used to read float value.
- 8) public double nexDouble() : Used to read double value.
- 9) public boolean nextBoolean() : Used to read boolean value.
- 10) public char next().charAt(0) : Used to read a single character

Q28/-What is new Keyword ?

- It is a keyword as well as operator in java.
- It is used to provide default value for non static data member.
- It is also useful to create the object with the help of Constructor.

Q29/-How to sort Array data ?

- import java.util.Arrays; (using this package use sort array)
- Arrays.sort(int []arr); //For sorting int array
- Arrays.sort(Object []arr) //For sorting String array
- ==========

```
package Array;
import java.util.Arrays;
public class Short_INT__Array {

   public static void main(String[] args) {


         int num []= {10,22,2,33,9};
          Arrays.sort(num);
//       System.out.println(num);

         for (int num1 :  num)
         {
               System.out.println(num1);
         }}

}
-------------------------
Opposite Long to Short
import java.util.Arrays;
import java.util.Collections;

public class Main {
    public static void main(String[] args) {
        Integer[] num = {5, 2, 9, 1, 6};
        Arrays.sort(num, Collections.reverseOrder());
        System.out.println(Arrays.toString(num));
    }
}
```

Q30/-In java, Can we hold hetrogeneous types of data using array ?

- Yes, by using Object array we can hold hetrogeneous type of data but we can't peform sorting operation using Arrays.sort(), It will generate
- java.lang.ClassCastException

```
-
- public class ForEachDemo6
- {
-     public static void main(String[] args)
-     {
-         Object []arr = {12, 23.90, 'A', true, "NIT", new String("A")};
-
-         // Arrays.sort(arr);  java.lang.ClassCastException
-
-         for(Object obj : arr)
-         {
-             System.out.println(obj);
-         }
-     }
-
- }
```

Q31/-In java, Is it possible to write multiple classes in single .java file ?

- Yes, We can write multiple classes but in a single .java file,
- we should have only one public class and that class must be file name (FileName.java).