**What is public static void main() ?**

**=>** public static void main() is entry point for the java program.

In public static void main()

Public is access modifer which gives access of code to the JVM.

Static means JVM can call these method without creating object . it can be call direct using classname.

Void is return type means main() method does not return any value .

And we are using Main() method is the entry point for java program .

And as argument we use Stirng [ ] array to accept value from command line client.

**Why java is so popular ?**

**=>** because of platform independent and object oriented programming language.

**Why we say that java is platform independent ?**

**=>** java is platform independent language because when we compile java code .class file is created ,means byte code is created , so these byte code is responsible to run on any JVM . so whtever operating sysytem are there if they having JVM install they can run these byte code.

So we can copy and paste code from one system to another system and we can run that byte code .

So , byte code is responsible to platform indendency.

**What is difference between heap and stack ?**

**=>** Generally object is created in heap memory . so in case of heap memory when we create object using new keyword all instance methods and instance variable get memory inside heap .

And in case of stack memory , genrally local variables (means variables inside methods ) and method parameter and method return values are stored in stack, when function is called its parameter and local variables are pushed into the stack and when the function returns they removed from the stack .

**Diffrence between equals() and == operator ?**

**=>** in java , equals() method is part of object class which is meant for content comaprison .

Means we compares value within the object .whereas

== operator meant for address comparison means we use == operator to comparres refrences.

Stirng s1 = new String(“shubh”);

String s2 = new String(“shubh”);

System.out.println(“by using equals methos : “ +s1.equals(s2) ); //true

System.out.println(“by using == operator “+ (s1 == s2) ); //false

**Why pointers are not used in java?**

**=>** pointers are not used in java because java is designed with memory saftey .

Suppose I am having variable and it is using pointer and if third person knows the address of variable then ofcourse that value can be accessed . but it should not allow .

So for security purpose java not use pointer .Instead of pointer java used references ., which are similer to pointers but added with saftey feature .

In java all objects are stored in heap area and accessed via references . which are managed by the system.

And also java uses garbage collector to automatically manage memory allocation and deallocation . which eliminates the need for explicit memory management.

**What is JIT ? and why we used?**

**=>** JIT stands for just in time compiler . It is part of Java runtime environment and gives fast performance to the java vertual machine.

It is used to convert java byte code into native machine code.

What is difference between local and instance variable with example?

=>

**What is methodoverloading and methodoverriding ?**

**=>** method overloading is the example of compile time polymorphism.

We can achieved method overloading by providing multiple methods with the same name but with different parameters in a same class . we can achieved it by changing numbers of parameters , or by changing data types of parametrs

Method overriding is the example of run time polymorphism .

Method overriding allow a subclass to provide specific implimentations of a method that is already defined in superclass . for method overriding method in the subclass must have the same name , return type and parameter as the method in superclass.

**What is key difference between java and c++ ?**

**=>** one of the biggest key difference in java and c++ is that java is platform independent ,

Meaning that java code can be run on any platform without recompile the code . while c++ code must be compiled for each platform separatly .

C++ suppports multiple inheritance but java does not support multiple inheritance .

Java has automatic garbage collector which meansprogrammer does not need to manually free memory , while in c++ programmer manually allocate and free memory.

C++ has pointers , which are used to manupulate memory directly , while java deos not have pointers.

Difference between final , finalize and finally ?

=>

**Explian type casting in java? Or what is casting in java ? and what is implicit casting ? explicit casting ?**

**=>** type casting means converting a variable of one data type into another .

There are two types of type casting in java

Implicit type casting and explicit type casting

Implicit type casting is done by compiler itself , in implicit type casting lower end data type is convert into higher end data type .

For example

Int a = 10 ;

Float b = a ; // int convert to float

And In explicit type casting highest data type is converted into lowest data type . here we do forcefully conversion in explicit type casting

For example :

float d = 10.03;

Int a = (int) d ; //float convert to int , it may leads to loss of data .

Explain call by value and call by refernce ?

=>

**Explain super keyword ?**

**=>** super keyword is used to access memebrs of the superclass .

If I want to access superclass variable in child class we can use super.superclass\_variable\_name .

And same with method if I overide superclass method in child class and we want to access super class method so I can use super.superclass\_method .

So same with constructor I can access parent class constructor from child class using super() .

**Explain this keyword ?**

**=>** this keyword is used to reference to the current object . it can be used in number of ways .

Means if I have local variable with the same name as instance variable . we can use this to refer to the instance variable .

And if I have multiple constructor in a class . we can use this to invoke another constructor from within costructor .

What are inner class and anonymous class ?

=>

**What is class and object ?**

**=>** class is template use to create object . or class is blue print which define objects of certain kinds .class is logical entity contains properties and behaviours .

Properties means variable and behaviours means methods .

For example : car , book , student

Suppose I have class student which has some properties like student name , roll no , address but don’t have any values associated with .

object is real world entity which has state , behavior and identity .

State means object has some data associated with it .

Behavior means it exhibit some actions .

Identity means it should be uniquly identifyble .

object is an instance of a class .In java we create objects by using new keword . and whenever we create objects , every instance members push into heap memory . and we can use them by using reference variable .

Example : my car , red book , ramesh student.

**What is abstraction ?**

**=>**  abstraction is one of the object oriented principle in java .

Abstaction is an act of represnting essential features and hiding internal implimentations details .

For example : I have one Account class , in that I have getBalance() method .

Here I will never highlights this method to end user , instead I give GUI screen and provide check balance button when user click this button that getBalance() method will execute .

In this way the user don’t know which code is executing he only see button .

That is abstaction , hide internal implimentation details and only showing essentials .

**What is default value of local variable and static variable ?**

**=>** Local variables are variables declared inside a method or block. They do not have a default value and must be initialized before they can be used. If a local variable is not initialized before it is used, a compile-time error will occur.

Static variables, on the other hand, are variables declared as static inside a class. They have default values depending on their data types:

byte, short, int, long: 0

float, double: 0.0

char: '\u0000' (null character)

boolean: false

Reference types (such as objects): null

**Is instance variable has default value ?**

**=>** Instance variables are variables declared inside a class, but outside of any method or block. They are initialized to default values when the object of the class is created.

They have default values depending on their data types:

byte, short, int, long: 0

float, double: 0.0

char: '\u0000' (null character)

boolean: false

Reference types (such as objects): null

**Define Data type ?**

**=>** Data type means what type of value is hold by the variable .In java we have two types of data type primitive data types and refrence type

Java have 8 no of primitive data type .

Byte ,int ,short ,long

Float , double

Char and boolean

And refrence type like object and Arrays .

Java also provides custom data type using classes and interfaces.

Define Exception handling concept in java?

=>

**Part 2 :**

**What is encapsulation ?**

**=>** encapsulation means binding data together into a single unit .

Lets have real life example there is word capsule , what capsule is doing , it wrapping data into single unit and provide protection to it by its outer cover .

So ,Similarly with programming language class is best example of encapsulation .

Class wraps data member and member functions into single unit and we provide protection using access modifirs like private and protected .

So , class is best way to bind data into single unit that is encapsulation .

**What is tightly encapsulation in java?**

**=>** Tightly encapsulation means if all the instance variables of the class declared as private then that class is said to be tightly encapsulated.

**If we declare data members as private then how we can access these data members outside the class ?**

**=>** If we declare data members as private then we can access these data members by using getter and setter methods .

**Explain Inheritance ?**

**=>** when we want to extend or reuse properties of parent class into child class we use inheritance . to impliment inheritance we use extends keyword . when a class extends another class it inherits all the non-private variables and methods of the superclass

For example :

Suppose I have class VEHICLE and it has data members like make and model\_No and I have another class CAR , it has same data members like like vehicle class and one extra data memebr like color , so I don’t required to write code again , I just need to extend class VEHICLE. like

Public class CAR extends VEHICLE { }

In this way all data members of class VEHICLE will inherited to class CAR. And by cretaing object of CAR class we can access properties of VEHICLE also.

This is possible because of inheritance .

We have 5 types of inheritance in programming : single level inheritance , multilevel inheritance , hierarchical inheritance , multple inheritance , and hybrid inheritance .

But java does not support multiple inheritance and hybrid inheritance .

**What is polymorphism ?**

**=>**  in polymorphism , poly means many and morph means forms ie. Having multiple forms is nothing but polymorphism .

For example : a person in class is a student and same person in market acts as a customer

Here person is same but its behavior is different . so we can say that it is polymorphism .

In java we have two types of polymorphism , compile time polymorphism and run time polymorphism

Method overloading is the example of compile time polymorphism .

And method overriding is the example of run time polymorphism .

**Can you explain static keyword in java ?**

**=>** static keyword used with variable or method or block . when we declare vaiable or method or block as static means they belongs to class itself .

When I declare variable or method as static , it means that there will be only one copy of that variable or methods share for all the instaces of that class .

We can access static variables and methods directly by using class name .

For example : I have static variable count in Test class so I can access It by using Test.count

and similerly with the method suppose I have add() method and which is static means syntax like public static int add() ; so this method I can access by using class name directly

As Test.add();

Also we have static block whatever code written in static block will be loaded at time of class loading , so static block will execute before main method .

**What is constructor ?**

**=>** Costructor is a special method is used to initialize member variable of invoking objects .

For example I have class student and in student class I have instance variables like name , email ,and rollno . so whenever we createing object that time we required constructor to intialise value for instance variable .

And in java if we are not creating any constructor compiler provides default constructor .means every class in java has default constructor .

In JAVA we have two types of constructor , default constructor and parameterised costructor.

**How do you allocate memory to object ?**

**=>** When we are creating object by using new keyword . so that time memory will be allocated .

What is difference between constructor and method ?

=> constructor are called automatically and method we have to call explicitly .

What are different types of access modifiers in java ?

=>

**What is garbage collector and how we can call it explicitly ?**

**=>** Garbage collector is built in java feature that automatically manages memory allocation and deallocation of objects in java program .

Unrefrence object are eligible for the garbage collector in java.

And we can call garbage collector in java by system.gc() method . but it does not guarenty that the garbage collector will run immediately .

What is finalize method ?

=>

What is var args method ?

=>

What is wrapper classes ?

=>

What are reference variables in java?

=> variables which are used to access objects are called reference variable .

Ex . Employee emp = new Employee() ;

Here emp is refrence variable .

How to do encapsulation in java?

=> by making instance variables private and defining getter and setter methods.

Will compiler creates a default constructor if I have a parameterised constructor in the class ?

=> no compiler will not create default constructor if I have a parameterised constructor in the class

**Can we override constructor in java ?**

**=>** constructor can’t be overriden in java . only methods can override.

**Can static method access instance variable ?**

**=>**no instance variable we can not access in static methods .

**How do we access instace variable and static variable in java ?**

**=>** instance variable and instance methods can be accessed using reference variable .

But to access static variable or static methods we use class name

**Can we override static method ?**

**=>**static method cant be override . if we have static method they are class level method .

**Why we use refrence variable ?**

**=>**objects are instance of class that resides in heap memory. Objects does not have any name , so to access object we use refrences.

Can we define static mthods inside interface?

=> we cant declare static methods inside interface .only instance methods are permitted in interface .and only public and abstract access modifers are allowed .

**Difference between java and c++ ?**

**=>** Java is platform independent. C++ is not platform independent.

Java is a pure Object Oriented Language (except for primitive variables). In C++, one can write structural programs without using classes and objects.

C++ has pointers (access to internal memory). Java has no concept called pointers.

In C++, programmer has to handle memory management. A programmer has to write code to remove an object from memory. In Java, JVM takes care of removing objects from memory using a process called Garbage Collection.

C++ supports Multiple Inheritance. Java does not support Multiple Inheritance.

**What is role of class loader ?**

**=>** when a program is executed JVM needs to load class . so JVM uses classLoader to find class.

There are three class loader

System class loader - loads all classes from classpath.

Extension class loader - loads class fro extension directory.

Bootstrap class loader - loads all java core files .

**What are string literal ?**

**=>** All string literal are stored in string constant pool. We create string literal using double quote.

String str = “Hello” ;

If we create any string literal , it check if it exists in the pool . if it exists it will not create new it only refers to same .

**Explian toString() method ?**

**=>** toString( ) method returns string representation of an object . if we does not override toString() method then it inherit toString() method from object class which returns string contaning class name , @ sign and object hashcode in hexadecimal format .