ASSIGNMENT 9

1. What is Inheritance in Java?

Ans: In programming language inheritance is a process where one class can access the properties of another existing class.
It achieved by using extends keywords.
There are having different type of inheritance
1.single inheritance
2.multi level inheritance
3.hierichical inheritance
4.multiple inheritance
In java multiple inheritance not supported.

2. What is superclass and subclass??
Ans: super class is also know as parent class .
A class where sub class inherit the properties is called as super class.

Sub class is also called as child class .
The class that access the properties of parent calass is called as subclass.

3. How is Inheritance implemented/achieved in Java?
Ans: In java for achieving inheritance our parent class should not be as private access modifier.
And we must use the keyword extends for inherit the parent class.

4. What is polymorphism?
Ans: polymorphism made by combing two Greek words
1.poly :- it means many
2.morphism:- many forms
In the concept of polymorphism one entity play different role.
It is mainly done when a programmer use override method in java.

5. Differentiate between method overloading and method overriding.
Ans:

Method Overloading	Method Overriding
1. In method overloading we must use the same name of function. 2. It is the process where we reuse the same method name as present in the class previously But it play different role. 3. In method overloading we can change access modifier. 4. In method overrloadingit is not mendtory to put the same signature of method while overloading here all method must contain different no. of parameter. 5. Error caught on runtime	1. In method overriding we must use the same name of function. 2. It is the process where we reuse the same method name as present in the class previously But it play different role. 3. In method overriding we can not change access modifier. 4. In method overriding we must put the same signature of method while overriding. 5. Error visible on compile time

6. **What is an abstraction explained with an Example?**
   Ans: when in java a method of parent class is using by other child class then that method should as abstract method.
   An abstract class must have minimum one abstract method but it is not mandatory , if any method is marked as abstract then it is not mandatory to make that class also abstract.
   If marked any method as abstract then that method not have its implementation. and that not accessible
   Example:
   Abstract class name
   {
   Abstract void method name();
   }

7. **What is the difference between an abstract method and final method in Java?Explain with an example**
   Ans: Abstract method is incomplete method while final method is complete method.
   If any class or method mark as abstract then for modification we can only us override method . means it can be override.
   But once we use final method for any entity then we cant modify to those . modification I not allowed here.

8. **What is the final class in Java?**

   Ans: If a class is marked as final in java then that class will be not participate in inheritance .

   if we try to do then it will throw **“compile time error”.**

   Once we mark a class as final then its conform that it complete , means that class is not abstract class.

   Example

```
final class A
{
void m1(){
System.out.println("m1-A");
}
}
// throw compilation error
class B extends A
{
void m1()
{
}
}

public class Test_FinalAbstract{
public static void main(String[] args){
B a=new B();
a.m1();
}
}
```

9. Differentiate between abstraction and encapsulation.

   Ans:

ABSTRACTION	ENCAPSULATION
1. When we make any class as abstract then we cant inherit them. 2. Its purpose is to gaining information . 3. It solve the issue at design time 4. It focus on external lookout. 5. In abstraction we use abstract class and interface to hide its code complexity	1. Encapsulation isfeatures of oops using that we can hide a code and data into single unit. 2. It use three access modifier according to need. 1.public, 2.privat, 3.protected. 3. when we use encapsulation then lines of code get reduce.

10. **Difference between Runtime and compile time polymorphism explain with an example**

Ans:

Runtime polymorphism	**compile time polymorphism**
1. When polymorphism exist at run time then its called ans runtime polymorphism 2. Its called as dynamic binding , late binding , overriding as well. 3. It is more flexible because all thing execute at runtime time 4. Method overriding is runtime polymorphism ,it hold the same name same parameter or signature.	1. When polymorphism exist at run time then its called ans runtime polymorphism. 2. It called a static,early or overloading binding. 3. It is less flexible because all thing execute at compile time. 5. Method overloading is compile time polymorphism ,it can not hold the same name same parameter or signature. It have different parameter or signature .