**Name:Faizan Ahmed Reg.no:FA20-BSE-029**

**CLASS: BSE7A COURSE:DESIGN PATTERN**

**LAB 1,2**

--------------------------------------------------------------------------------------------------------------------------------

Lab1:

Abstraction: example

abstract class Language {

// method of abstract class

public void display() {

System.out.println("This is Java Programming");

}

}

class Main extends Language {

public static void main(String[] args) {

// create an object of Main

Main obj = new Main();

// access method of abstract class

// using object of Main class

obj.display();

}

}

---------------------------------------------------------------------------------------------------------------------------------------

LAB2:

1. OVERLOADING of the Main method is Possible or not?

Answer:

In java overloading of the Main method is not possible.B/c you can’t have different version of the same main method .it always remains the same as of the “ public static void main(string[] args)”

Example:

class TestBird {

public static void main(String[] args) {

Animal myBird = new Bird();

myBird.label();

myBird.move();

myBird.eat();

}

}

-------------------------------------------------------------------------------------------------------------------------------------

1. What is return by the constructor and how u can identify it from declarations?

Answer:

Constructor in java doesn’t have a return type you can identify it having same name of class.In java it didn’t explicitly return a value as regular method gives.It only initializes an object when it creates.You can identify the constructor name as of the class.

Example:

Class Main{

Int x,y;

Public Main(){

X=5;

Y=12;}

public static void main(string[] args){

Main mymain=new Main();

System.out.println(mymain.x);

}

}

--------------------------------------------------------------------------------------------------------------------------------

1. can we create a program without main method? how many methods are allowed in Java?

Answer?

No, we can’t create a program without main Method. Whenever we create a JAVA program Main method will be the entry point for execution of a program “public static void main(string[] args)”. I think there is no fix limit on the number of methods Which is allowed in Java.

----------------------------------------------------------------------------------------------------------------------------

1. what are the six ways to use this keyword?

**Answer:**

* This is used to get the current object. E.g: ClassName Current Object = this;
* This is used to invoke current object's method.this.MyMethod ();
* this () is used to invoke current class constructor.e.gthis(3) (it will called another constructor with int parameter).
* this is passed as a parameter to a method call.**e.g Method(this);**
* This is passed as a parameter to a constructor.**e.g ClassName myClass=new ClassName(this);**
* this used to return the current object from the method.**e.g MyClass Object(){return this;}**

1. **prove that multiple inheritance in not supported in java?**

**Answer:**

In Java Multiple inheritance is not supported because **a class can’t extend more than one class like that Class extends** Car, Vigo{} reason is to not confuse let say **if class B extends from C and A and both have an same method name now Java complier can’t decide which method should inherit.**

**---------------------------------------------------------------------------------------------------------**

1. **when to use aggregation and not composition and vice versa?**

**Answer:**

**Use Composition When:**

* when we want one class to be made up of another class, and the part is essential to the whole.
* we can think of it as a stronger "part-of" relationship.
* Example: A car has an engine, and the engine is a crucial part of the car; it doesn't make sense without it.

**Use Aggregation When:**

* When we want one class to have another class as a part, but that part can exist on its own.
* we can think of it as a "has-a" relationship.
* Example: A car has an engine, but the engine can exist outside the car.

**-----------------------------------------------------------------------------------------------------------------------**

1. **how to override the static method?**

**Answer:**

You cannot override static methods in Java. because static methods are not associated with the instance of a class, but with the class itself. when a subclass inherits a static method from its parent class, it cannot modify the behavior of the static method in any way.

For example:

class Parent {

static void staticMethod() {

System.out.println("Static method in Parent class");

}

}

class Child extends Parent {

static void staticMethod() {

System.out.println("Static method in Child class");

}

}

public class Main {

public static void main(String[] args) {

Parent.staticMethod();

Child.staticMethod();

}

}

**------------------------------------------------------------------------------------------------------------------------**