**Name: Vijeta S. Karad**

**Batch\_No: 6523.**

**Entrollment\_No: EBEON1221522057.**

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

Q.4] Process of initialization

i. Write a program to create a class ‘Insect’ having two instance variables ‘i’, ’j’ of int type where ‘i’ is initialized with value of 1.

ii. Implement the default constructor for this class which displays the values of the instance variables in the console and initialize ‘j’ with value 1.

iii. Now declare a variable x1 as following

private static int x1 = print("static Insect.x1 initialized");

Where print (String) method is a static method which prints its parameter String in the console and return int type (here returns 0);

iv. Write a program to create class ‘Beetle’ which extends the above class ‘Insect’, having a private instance variable k initialized as following

private int k = print("Beetle.k initialized");

v. This class ‘Beetle’ also implements default constructor which actually prints the value of ‘k’ and ‘j’.

vi. Now declare a private static variable ‘x’ of int type and initialize it as following.

private static int x2 = print("static Beetle.x2 initialized");

vii. Write a test class by creating a class ‘TestInitialization’ which implements main() method, Inside main method create a new instance of ‘Beetle’ class.

Expected Output:

static Insect.x1 initialized

static Beetle.x2 initialized

i = 1, j = 0

Beetle.k initialized

k = 0

j = 1

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

--->

class Insect

{

int i = 1;

int j;

Insect()

{

System.out.println("i = " + i);

System.out.println("j = " + j);

j = 1;

}

private static int x1 = print("static Insect.x1 initialized");

private static int print(String string)

{

System.out.println(string);

return 0;

}

}

class Beetle extends Insect

{

private int k = print("Beetle.k initialized");

private static int x2 = print("static Beetle.x2 initialized");

private static int print(String string)

{

System.out.println(string);

return 0;

}

public int getK()

{

return k;

}

public void setK(int k)

{

this.k = k;

}

}

public class TestInitialization

{

public static void main(String[] args)

{

Beetle betle = new Beetle();

System.out.println("k = " + betle.getK());

System.out.println("j = " + betle.j);

}

}

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

Output -

static Insect.x1 initialized

static Beetle.x2 initialized

i = 1

j = 0

Beetle.k initialized

k = 0

j = 1

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