Page no.: ____/
Date: __/__/____/

practical-2

a) write a program to create a class and implement a default, overloaded and ~~copy~~ parameterized constructor.

i) for default constructor :-

program -

```
class TestConstructor
{
int a, b;
TestConstructor()
{
System.out.println("Value Default Constructor!!!
                                    ");
a=10;
b=20;
System.out.println("value of a="+a);
System.out.println("value of b="+b);
}
}
class mainConstructor
{
public static void main(String[] args)
{
Test Constructor obj = new TestConstructor();
}
}
```

Output - c:\Java ass>javac mainConstructor.java
c:\Java ass>java mainConstructor

Default Constructor !!!
value of a=10
value of b=20

ii) for paramaterized constructor.

program—

```
clay TestConstructor
{
 int a,b;
 TestConstructor(int x, int y)
 {
System.out.println("paramaterized constructor!!!");
 a=x;
 b=y;
System.out.println("value of a="+a);
System.out.println("value of b="+b);
}
}
clas mainConstructor
{
 public static void main(String [] args)
 {
TestConstructor obj = new TestConstructor(10,20);
 }
};
```

Output-

```
C:\java all> javac mainConstructor.java
C:\java all> java mainConstructor
Paramaterized constructor !!!
value of a=10
value of b=20
```

Page no.: ____/
Date: ____/____/____/

b) write a program to create a clay and implement the concepts of method overloading.

i) program to different data type of arguments-

program-

```
clay Calculate
{
void sum (int a, int b)
{
System.out.println ("Sum is:"+(a+b));
}
void sum (float a, float b)
{
System.out.println ("Sum is:" + (a+b));
}
public static void main (String [] args)
{
Calculate cal = new Calculate ();
cal.sum (8,5);
cal.sum (4.6f, 3.8f)
}
}
```

Output- C:\Java an> javac Calculate.java
C:\Java an> java Calculate
Sum is: 13
Sum is: 8.4

ii) program to different number of arguments-

program-

```
class Demo
{
void multiply(int l, int b)
{
System.out.println("Result is:"+(l*b));
}
void multiply(int l, int b, int h)
{
System.out.println("Result is:"+(l*b*h));
}
public static void main(String [] args)
{
Demo ar = new Demo();
ar.multiply(3,5);
ar.multiply(4,6,2);
}
}
```

Output-

```
C:\Java all>javac Demo.java
C:\Java all>java Demo
  Result is: 40
  Result is: 48
```

Page no.: ____/

Date: ____/____/____/

c) write a program to create class and implement the concepts of static methods.

i) program- for static variable to create a class.-

program-

```
public class StaticVariableExample {
    int x, y;
    static int z;
StaticVariableExample(int x, int y) {
    this.x = x;
    this.y = y;
    }
    public void show()
    {
    int a;
System.out.println("Inside Show method");
System.out.println("x="+x+"y="+y+"z="+z);
    }
    public static void main(String [] args)
    {
StaticVariableExample obj_1 = new StaticVariableExample(10,20);
StaticVariableExample obj_2 = new StaticVariableExample(100,200);
 obj_1.show();
StaticVariableExample.z = 1000;
 Obj_2.show();
}
}
```

Output- C:\Java all> javac StaticVariableExample.java
C:\Java all> java StaticVariableExample
Inside show method
x=10 y=10 z=20
Inside show method
x=100 y=200 z=1000

ii) program for static method-

Program-

```
class Student {
    int rollno;
    String name;
    static String college = "ITS";
static void change() {
    college = "BBDIT";
}
Student(int r, String n) {
    rollno = r;
    name = n;
}
void display() {
System.out.println(rollno + " " + name + " "
                    + college);
}
}

public class TestStaticMethod {
  public static void main(String[] args) {
```

```
Student.change();
Student S1 = new Student(111, "Akash");
Student S2 = new Student(222, "Shyam");
Student S3 = new Student(333, "Akhilesh");
  S1.display();
  S2.display();
  S3.display();
 }
}
```

Output-

```
c:\Java all> Javac TestStaticMethod.java
c:\Java all> java TestStaticMethod
111 Akash BBDIT
222 Shyam BBDIT
333 Akhilesh BBDIT
```