EXPERIMENT NO. 01

AIM: Write a program to show to show scope of variables in JAVA.

PROGRAM:

```
class Main
{
public static void main(String[] args)
{
int a=10;
int b=10;
int c=a+b;


  System.out.println("value of a="+a);
  System.out.println("value of b="+b);
  System.out.println("sum="+c);

}
}
```

```
value of a=10
value of b=10
sum=20


...Program finished with exit code 0
Press ENTER to exit console.
```

EXPERIMENT NO.02

AIM: Write a program to show Concept of CLASS in JAVA.

PROGRAM:

```
class Student{

int rollno;

String name;

void insertRecord(int r, String n){

rollno=r;

name=n;

}
void displayInformation(){System.out.println(rollno+" "+name);}

}

class TestStudent4{

public static void main(String args[]){

Student s1=new Student();

Student s2=new Student();

s1.insertRecord(111,"Karan");

s2.insertRecord(222,"Aryan");

s1.displayInformation();

s2.displayInformation();}

}
```

OUTPUT:

Compile by: javac TestStudent4.java

Run by: java TestStudent4

```
111 Karan
222 Aryan
```

EXPERIMENT NO.03

AIM: Write a program to show typecasting in java

PROGRAM:

Converting double into an int

```
class Main {
  public static void main(String[] args) {
    // create double type variable
    double num = 10.99;
    System.out.println("The double value: " + num);

    // convert into int type
    int data = (int)num;
    System.out.println("The integer value: " + data);
  }
}
```

OUTPUT:

```
The double value: 10.99
The integer value: 10
```

conversion from int to String

```
class Main {
  public static void main(String[] args) {
    // create int type variable
    int num = 10;
    System.out.println("The integer value is: " + num);

    // converts int to string type
    String data = String.valueOf(num);
    System.out.println("The string value is: " + data);
  }
}
```

OUTPUT:

```
The integer value is: 10
The string value is: 10
```

EXPERIMENT NO. 4

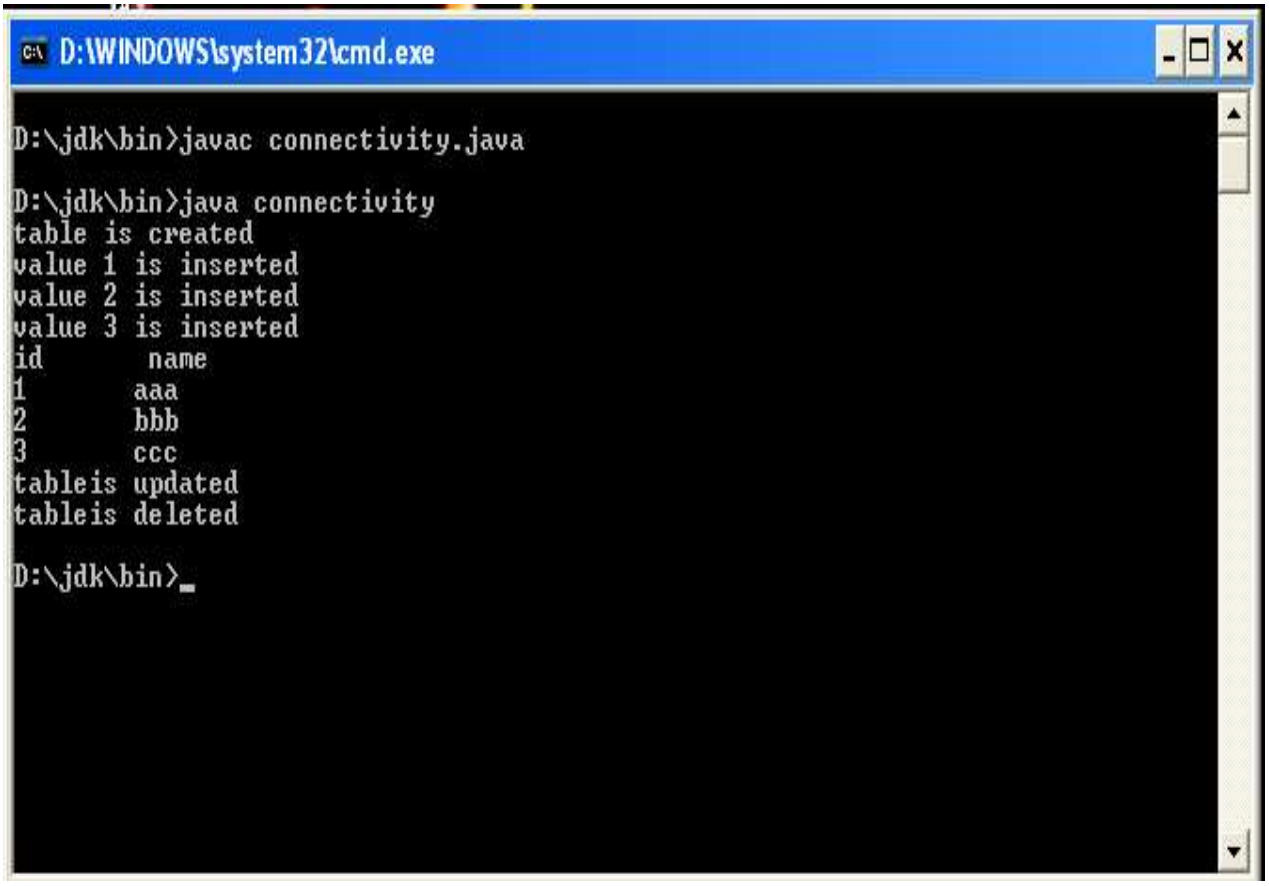
AIM: Write A Program To Perform Connectivity in JDBC (Java Database Connectivity)

PROGRAM:

```
import java.lang.*;
import java.io.*;
import java.sql.*;
public class connectivity
{
public static void main(String args[])
{

try { String str0=”Drop table student”;
String str1="create table student"+" (c_id integer ,"+ "c_name varchar(20))";

String str2="insert into student(c_id,c_name)values (1,'aaa')";
String str3="insert into student(c_id,c_name)values (2,'bbb')";
String str4="insert into student(c_id,c_name)values (3,'ccc')";
String str5="select * from student";
String str6="update student set c_id=5 where c_name='bbb'";
String str7="delete from student where c_id=5";
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:test","hb","");
Statement stmt=con.createStatement();

Stmt.execute(str0);
stmt.execute(str1);
System.out.println("table is created");
int count1=stmt.executeUpdate(str2);
```

```
System.out.println("value 1 is inserted");
int count2=stmt.executeUpdate(str3);
System.out.println("value 2 is inserted");
int count3=stmt.executeUpdate(str4);
System.out.println("value 3 is inserted");
ResultSet rs=stmt.executeQuery(str5);
System.out.println("id \t name");
while(rs.next())
{
String id=rs.getString("c_id");
String name=rs.getString("c_name");
System.out.print(id+"\t");
System.out.print(name+"\n");
System.out.print();
}
int count4=stmt.executeUpdate(str6);
System.out.println("tableis updated");
int count5=stmt.executeUpdate(str7);
System.out.println("tableis deleted");
con.close();

}
catch(Exception ex)
{
System.out.println("error occured"+ ex);

}
```

OUTPUT:


D:\WINDOWS\system32\cmd.exe
D:\jdk\bin>javac connectivity.java
D:\jdk\bin>java connectivity
table is created
value 1 is inserted
value 2 is inserted
value 3 is inserted
id name
1 aaa
2 bbb
3 ccc
tableis updated
tableis deleted
D:\jdk\bin>

EXPERIMENT NO. 5

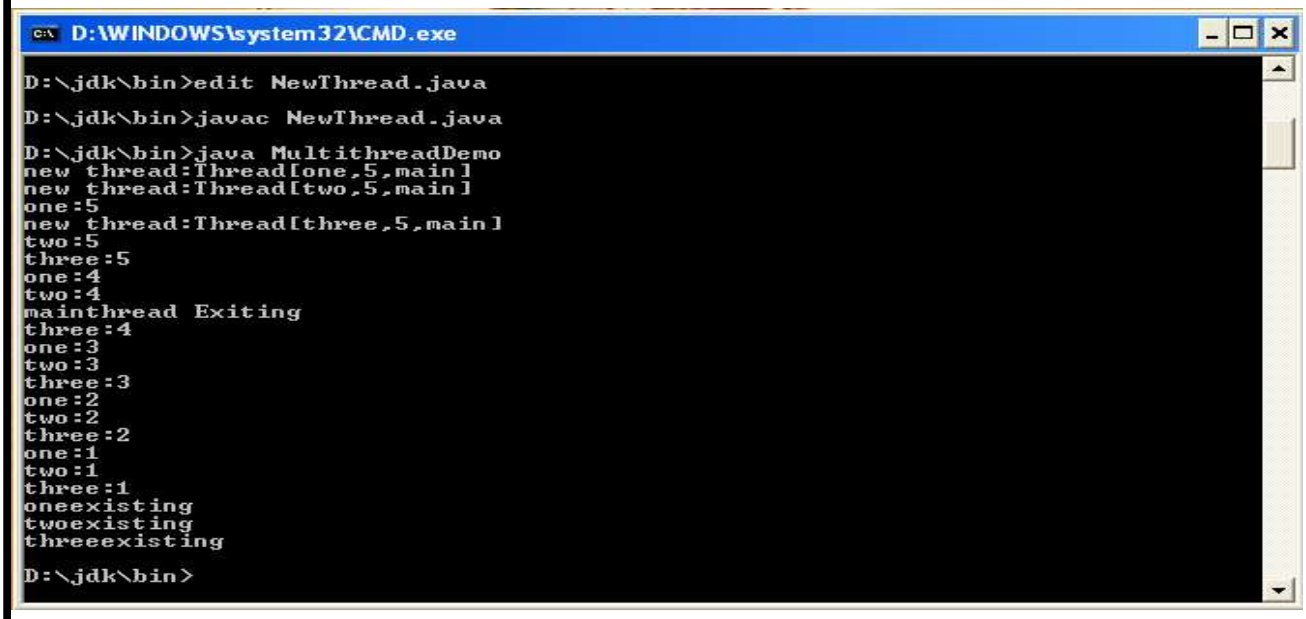
AIM: Write A Program To Perform Multithreading Operation

```
importjava.io

*;

importjava.net

.;

class NewThread implements Runnable

{

        String
        name;

        Thread t;

        NewThread(String threadname)

        {

                name=threadname;

                t=new

                Thread(this,name);

                System.out.println("new

                thread:"+t);t.start();

        }

        public void run()

        {

        try

        {

                for(int i=5;i>0;i--)
```

```
            {
                System.out.println(name
                +":"+i);
                Thread.sleep(1000);
            }
        }
        catch(InterruptedException e)
        {
            System.out.println(name+"Intrrupted");
        }
            System.out.println(name+"existing");
        }
}
class MultithreadDemo
{
public static void main(String args[]) throws Exception
{
new
NewThread("one");
new
NewThread("two");
new
NewThread("three")
;try
```

```
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{
System.out.println("mainthread interrupted");
}
System.out.println("mainthread Exiting");
}
}
```

OUTPUT:



EXPERIMENT NO. 6

AIM: Write a Program to show Inheritance .

```
class Animal {
  // field and method of the parent class
  String name;
  public void eat() {
    System.out.println("I can eat");
  }
}
// inherit from Animal
class Dog extends Animal {
// new method in subclass
  public void display() {
    System.out.println("My name is " + name);
  }
}
class Main {
  public static void main(String[] args) {
// create an object of the subclass
    Dog labrador = new Dog();

// access field of superclass
    labrador.name = "Rohu";
    labrador.display();

// call method of superclass
    // using object of subclass
    labrador.eat();
```

```
}

}
```

```
Output:  My name is Rohu
         I can eat
```

EXPERIMENT NO. 7

AIM: Write a program to show Access Specifiers (Public, Private, Protected) in JAVA.

Public -

```
//Animal java
// public class
public class Animal {
// public variable
public int legCount;

// public method
public void display() {
System.out.println("I am an animal.");
System.out.println("I have " + legCount + " legs.");
}
}

// Main.java
public class Main {
public static void main( String[] args ) {
// accessing the public class
Animal animal = new Animal();

// accessing the public variable
animal.legCount = 4;
// accessing the public method
animal.display();
}
}
```

OUTPUT: I AM AN ANIMAL

Protected

```
class Animal {
   // protected method
   protected void display() {
      System.out.println("I am an animal");
   }
}

class Dog extends Animal {
   public static void main(String[] args) {

      // create an object of Dog class
      Dog dog = new Dog();
       // access protected method
      dog.display();
   }
}
```

Output:: I am an animal

Private

```
class Data {
   private String name;

   // getter method
   public String getName() {
      return this.name;
   }
   // setter method
   public void setName(String name) {
      this.name= name;
   }
}
public class Main {
   public static void main(String[] main){
      Data d = new Data();

      // access the private variable using the getter and setter
      d.setName("Programiz");
      System.out.println(d.getName());
   }
}
```

EXPERIMENT NO. 8

AIM: Write a program to Hide a Class.

```
//Java program to demonstrate
// method Hiding in java

// Base Class
class Complex {
   public static void f1()
   {
      System.out.println(
         "f1 method of the Complex class is executed.");
   }
}

// class child extend Demo class
class Sample extends Complex {
   public static void f1()
   {
      System.out.println(
         "f1 of the Sample class is executed.");
   }
}
public class Main {

   public static void main(String args[])
   {
      Complex d1 = new Complex();
```

```
        // d2 is reference variable of class Demo that
        // points to object of class Sample
        Complex d2 = new Sample();

        // But here method will be call using type of
        // reference
        d1.f1();
        d2.f1();
    }
}
```

Output

f1 method of the Complex class is executed.

EXPERIMENT NO. 9

AIM: Write a program to show Polymorphism

```
class Polygon {

  // method to render a shape
  public void render() {
    System.out.println("Rendering Polygon...");
  }
}

class Square extends Polygon {

  // renders Square
  public void render() {
    System.out.println("Rendering Square...");
  }
}

class Circle extends Polygon {

  // renders circle
  public void render() {
    System.out.println("Rendering Circle...");
  }
}

class Main {
```

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

    // create an object of Square
    Square s1 = new Square();
    s1.render();

    // create an object of Circle
    Circle c1 = new Circle();
    c1.render();
  }
}
```

Output

```
Rendering Square...
Rendering Circle...
```

EXPERIMENT NO. 10

AIM: Write a program to demonstrate AWT.

```
1.  import java.awt.*;
2.
3.
4.  public class AwtProgram1 {
5.  public AwtProgram1()
6.      {
7.  Frame f = new Frame();
8.          Button btn=new Button("Hello World");
9.          btn.setBounds(80, 80, 100, 50);
10.         f.add(btn);         //adding a new Button.
11.         f.setSize(300, 250);        //setting size.
12.         f.setTitle("JavaTPoint");  //setting title.
13.         f.setLayout(null);   //set default layout for frame.
14.         f.setVisible(true);          //set frame visibility true.
15.     }
16.
17.
18. public static void main(String[] args) {
19. // TODO Auto-generated method stub
20.
21.         AwtProgram1 awt = new AwtProgram1();   //creating a frame.
22.     }
23. }
```

Output: