# Practical File
# Of
## “Programming Java”


**Submitted in partial Fulfillment for the award of degree of Master of Computer Application**
**(BATCH:2023-25)**

**Submitted by:**
Prof. Harish Dutt
Associate Professor-USCS

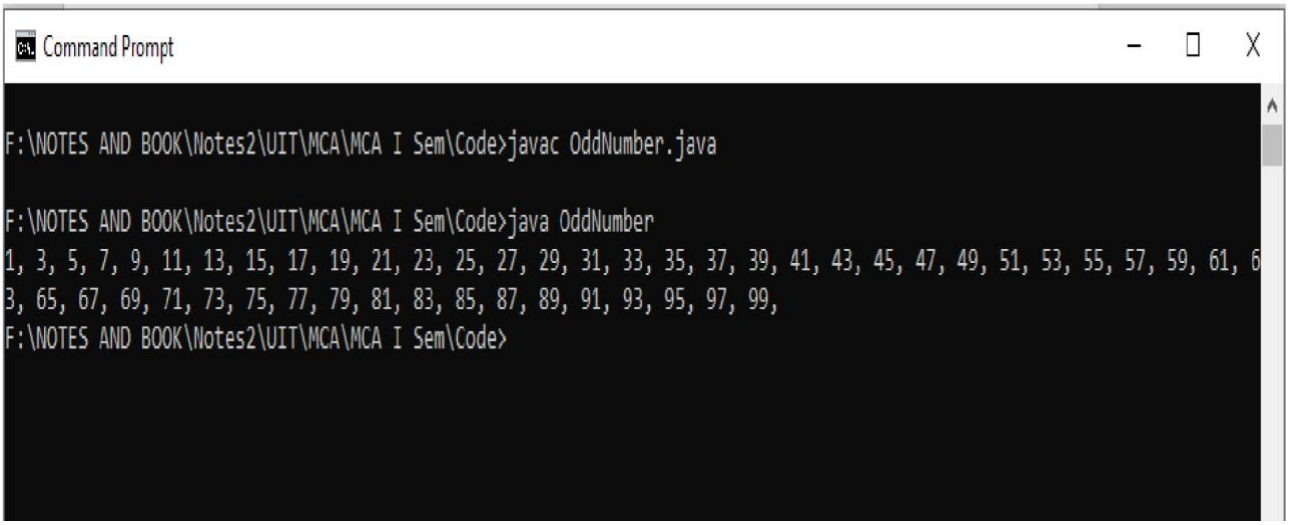
**Submitted by**:
Mohammad Ashaq
Roll No: 37
University Roll No: 232000125

# INDEX

**Practical 1: Write a java program to print all odd numbers between 1 to 100.**

```
class OddNumber
{
	public static void main(String[] args)
	{
		for(int i=0;i<=100;i++)
		{
			if(i%2!=0)
			{
				System.out.print(i+", ");
			}
		}
	}
}
```

**Output**



**Practical 2: Create A Simple Calculator Using Various Operators and Switch Case to Perform Various Arithmetic Operations in Java.**

```
import java.util.Scanner;
public class Calculator {

   public static void main(String arg[])
   {
     float v1=0,v2=0, sum, mul, div, sub;
float rus;       int choice;
     String str;

     Scanner sc;
do{
          System.out.println("------------- CALCULATOR ------------- \n");
          System.out.println("1 --->  SUM \n");
          System.out.println("2 --->  SUBTRACT \n");
          System.out.println("3 --->  Multiply \n");
          System.out.println("4 --->  Devide \n");
          System.out.println("5 --->  Exit \n");
System.out.println("Enter Choice : ");
sc=new Scanner(System.in);             choice=
sc.nextInt();
          if(choice != 5)
          {
            System.out.println("Enter First value : ");
v1=sc.nextFloat();
            System.out.println("Enter Second value : ");
v2=sc.nextFloat();
          }

      switch(choice)
      {
case 1:
         {
               sum= v1+v2;
               System.out.println("Sum is = "+ sum);
break;
         }
case 2:
{
               sub= v1-v2;
               System.out.println("Substraction is = "+ sub);
break;
         }
case 3:
{
               mul= v1*v2;
               System.out.println("Multiply is = "+ mul);
break;
```

```
        }
case 4:
{
            div= v1/v2;
            System.out.println("Devision is = "+ div);
break;
        }
case 5:
        {
            System.exit(0);
            break;
        }
default:
        {
           System.out.println("You choice is wrong");
        }

      }
       System.out.println("Do you want to continue if Yes then press 'y' or if No then press 'n'  :  ");
str = sc.next();

     }while(str.charAt(0)=='y'||str.charAt(0)=='Y');
     }

}
```

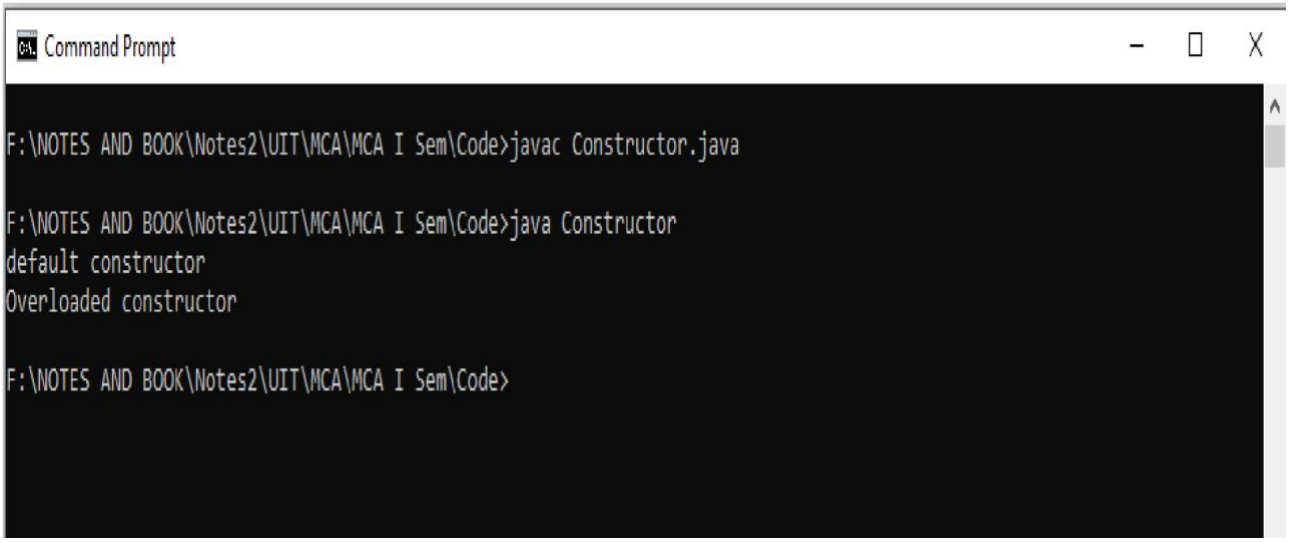
## Output

```
------------- CALCULATOR -------------

1 --->  SUM

2 --->  SUBTRACT

3 --->  Multiply

4 --->  Devide

5 --->  Exit

Enter Choice :
2
Enter First value :
34
Enter Second value :
22
Substraction is = 12.0
Do you want to continue if Yes then press 'y' or if No then press 'n'  :
y
------------- CALCULATOR -------------

1 --->  SUM

2 --->  SUBTRACT

3 --->  Multiply

4 --->  Devide

5 --->  Exit

Enter Choice :
5
```

**Practical 3: Write a java program to implement constructor overloading.**

```
class A {
        A()
        {
        System.out.println("default constructor");
         }
         A(int b)
        {
                System.out.println("Overloaded constructor");
        }
}
class Constructor
{
        public static void main(String[] args)
        {
                A ob = new A();
        A ob1 = new A(10);
        }
}
```

**Output**



**Practical 4: Write a program of sum of the given number.**

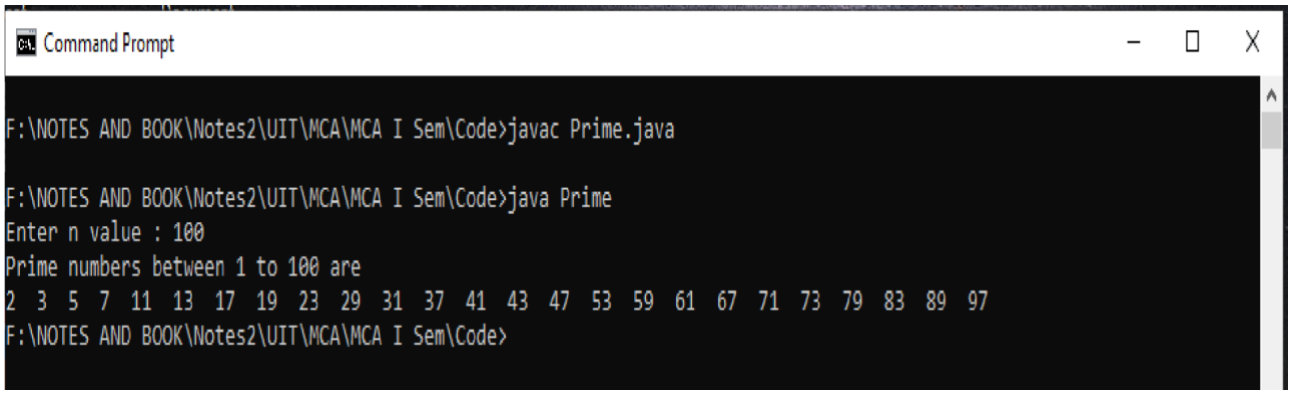
```
import java.util.Scanner; class
SumOfDigits
{
	public static void main(String arg[])
	{
		long n,sum;
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter a number ");
		 n=sc.nextLong();
		for(sum=0 ;n!=0 ;n/=10)
		{
		sum+=n%10;
		}
		System.out.println("Sum of digits of a number is "+sum);
	}
}
```

**Output**

**Practical 5: Write a program to print a prime number between 1 to 100 or given number.**

```
import java.util.Scanner; class
Prime
{
        public static void main(String arg[])
        {
        int i,count;
          System.out.print("Enter n value : ");
        Scanner sc=new Scanner(System.in);
int n=sc.nextInt();
          System.out.println("Prime numbers between 1 to "+n+" are ");
for(int j=2;j<=n;j++)
        {
        count=0;
        for(i=1;i<=j;i++)
        {
          if(j%i==0)
          {
              count++;
          }
        }
        if(count==2)
            System.out.print(j+"  ");
        }
        }
}
```

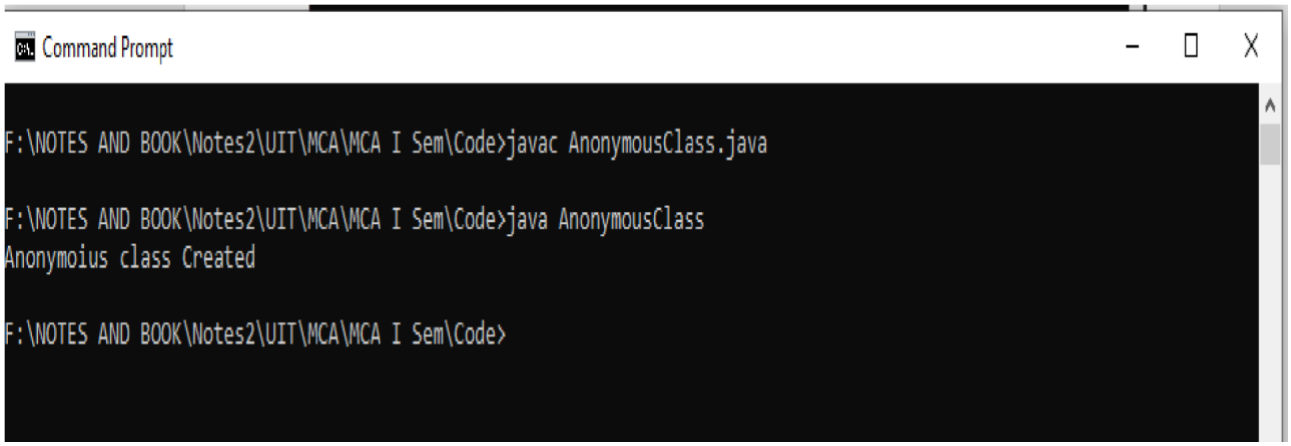
**Output**



**Practical 6: Write a java program to implement anonymous class.**

```
class AnonymousClass
{
        public static void main(String[] args)
        {
            abc ob = new abc()
                {
                        public void show()
                        {
                                System.out.println("Anonymoius class Created");
                        }
                    };
                ob.show();
        }
}
interface abc
{
   public void show();
}
```

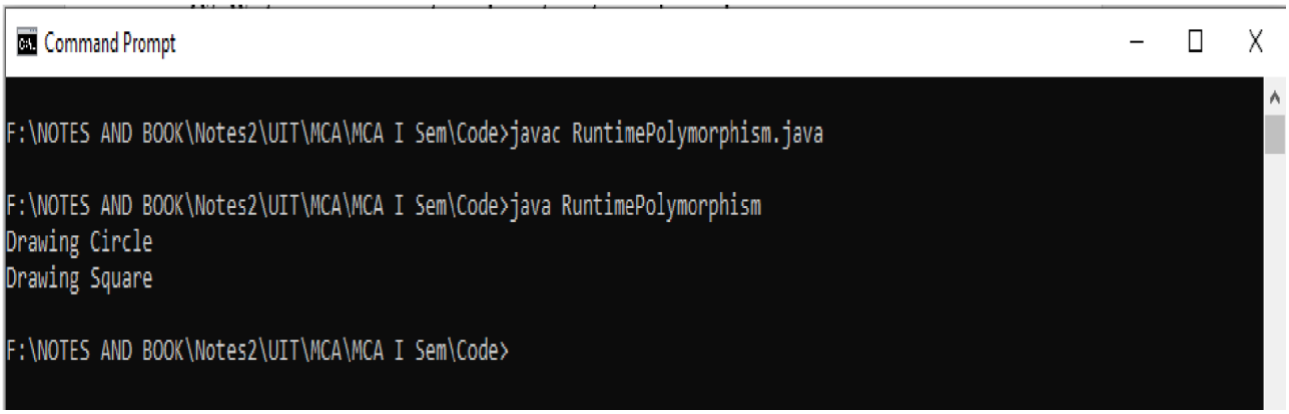
**Output**



**Practical 7: Write a java program to implement run time polymorphism.**

```
class Shape
{
  void draw()
  {
        System.out.println("Drawing Shape");
  } }
class Circle extends Shape
{
        void draw()
        {
              System.out.println("Drawing Circle");
        }
}
class Square extends Shape
{     void
draw()
          {
              System.out.println("Drawing Square");
          }
}
public class RuntimePolymorphism
{
    public static void main(String[] args)
          {
            Shape s;              s
= new Circle();
s.draw();
              s = new Square();
s.draw();
        }
}
```

**Output**

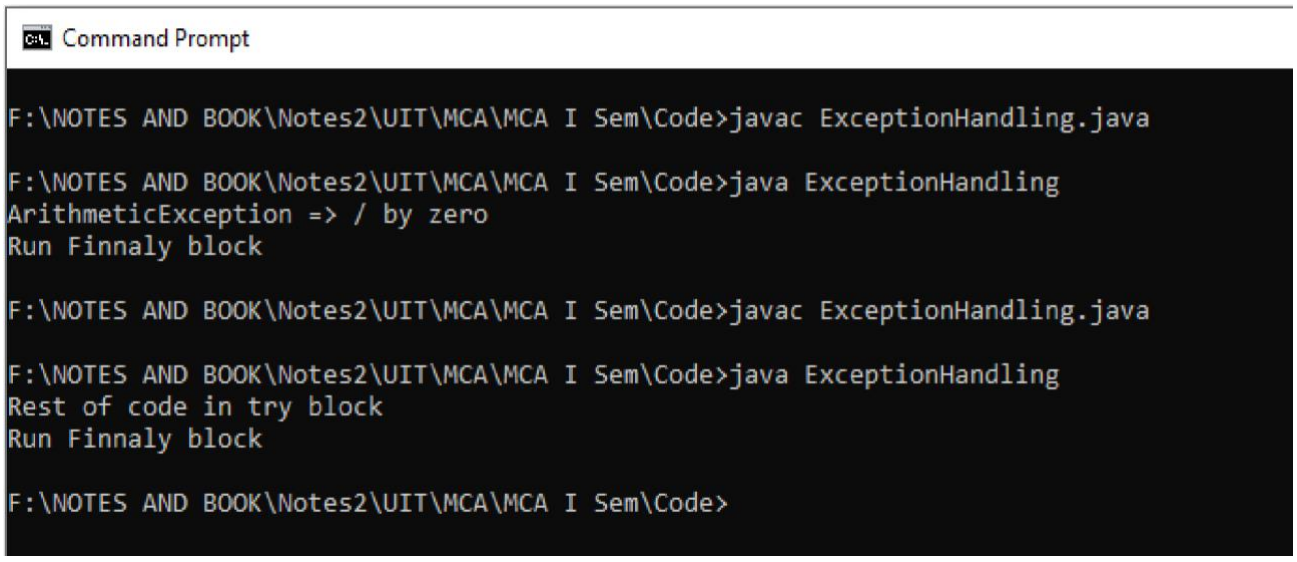


## Practical 8

**:** Write a program in java to handles the exception caused by dividing by zero.

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

     int n = 5 / 2;
     System.out.println("Rest of code in try block");
    }

   catch (ArithmeticException e) {
     System.out.println("ArithmeticException => " + e.getMessage());
   }
          finally{
                        System.out.println("Run Finnaly block");
                }
  }
}
```

**Output**

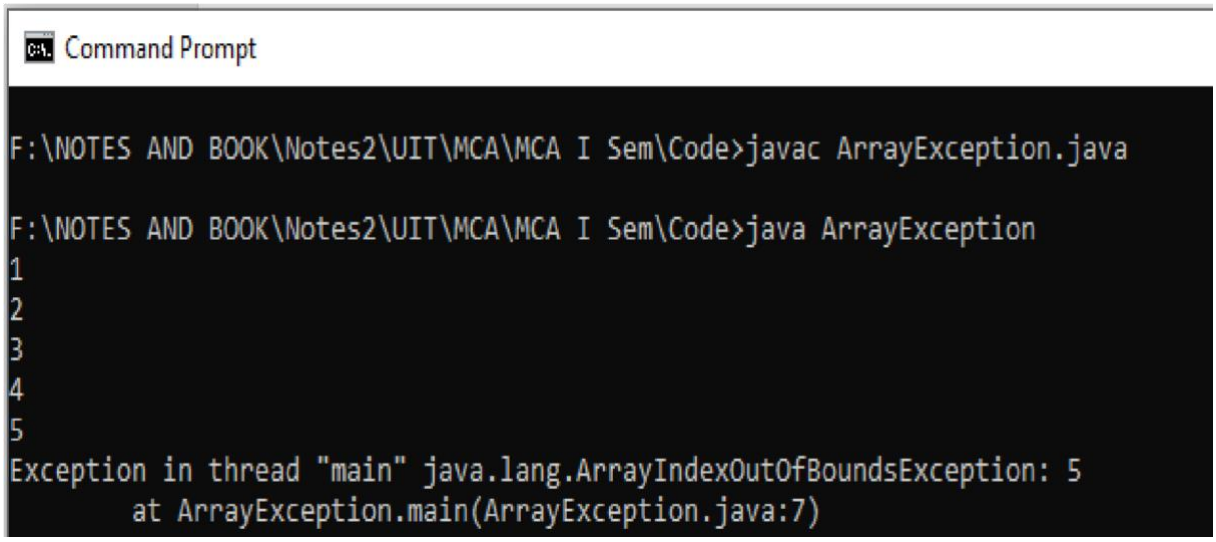


**Practical 9**

**:** Write a program in java to handles the ArrayIndexOutOfBoundsException.

```
public class ArrayException {     public
static void main(String[] args)
   {
      int ar[] = { 1, 2, 3, 4, 5 };

      for (int i = 0; i <= ar.length; i++)
         System.out.println(ar[i]);
   }
}
```

**Output**



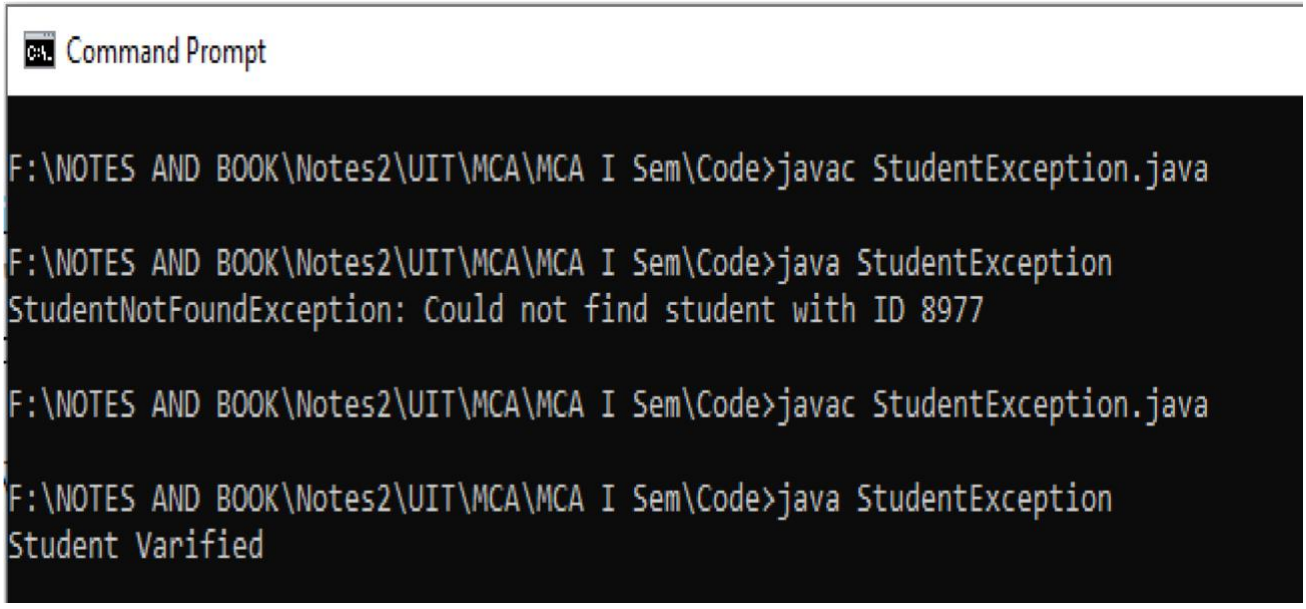

**:** Write a program in java to create your own exception.

```
class StudentNotFoundException extends Exception {
StudentNotFoundException(String message) {
                super(message);
        }
}
class StudentVarification {
        StudentVarification find(String studentID) throws StudentNotFoundException {
if (studentID.equals("8979")) {                 return new StudentVarification();
```

**Practical 10**

```
        } else {
  throw new StudentNotFoundException("Could not find student with ID " + studentID);
                }
        }
}
public class StudentException {
        public static void main(String[] args) {
                StudentVarification varify = new StudentVarification();
                try {
                        StudentVarification student = varify.find("8979");
                        System.out.println("Student Varified");

                } catch (StudentNotFoundException ex) {
                        System.err.println(ex);
                }
        }
}
```

**Output**

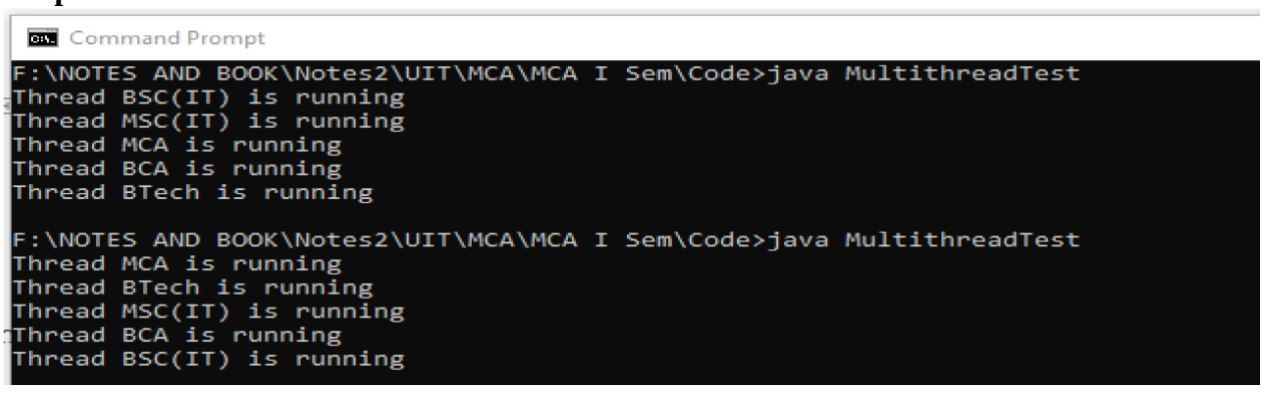


**Practical 11**

**:** Write a program in java for multithreading creation by extending Thread class.

```
class Multithreading extends Thread {
   public void run()
   {
      try {
         System.out.println("Thread " + Thread.currentThread().getName()+ " is running");
         Thread.sleep(10000);
      }
      catch (Exception e) {
         System.out.println("Exception is caught");
      }
   }
}
public class MultithreadTest {
   public static void main(String[] args)
   {
      Thread t1,t2,t3,t4,t5;

                  t1=new Multithreading();
                  t2=new Multithreading();
                  t3=new Multithreading();
                  t4=new Multithreading();
                  t5=new Multithreading();

                  t1.setName("MCA");
                  t2.setName("BCA");
                  t3.setName("BSC(IT)");
                  t4.setName("MSC(IT)");
                  t5.setName("BTech");

                  t1.start();
                  t2.start();
                  t3.start();
                  t4.start();
                  t5.start();
    }
}
```

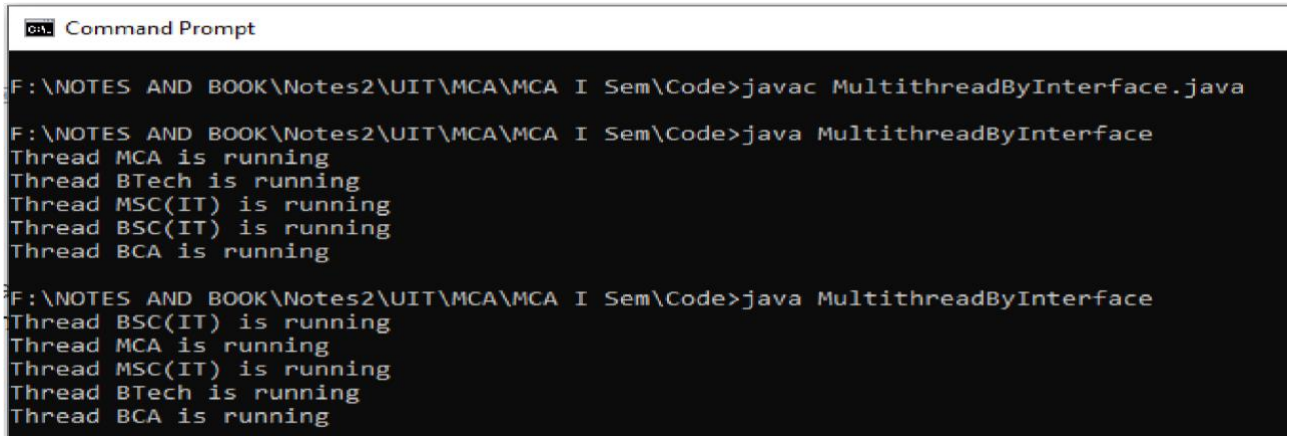
**Output**



**Practical 12:** Write a program in java for multithreading creation by implementing Runnable interface.

```
class Multithreading implements Runnable
{
public void run()
    {
      try {
         System.out.println("Thread " + Thread.currentThread().getName()+ " is running");
         Thread.sleep(10000);
      }
      catch (Exception e) {
         System.out.println("Exception is caught");
      }
   }
}
public class MultithreadByInterface {
   public static void main(String[] args)
   {
      Thread t1,t2,t3,t4,t5;
t1=new Thread(new Multithreading());
t2=new Thread(new Multithreading());
t3=new Thread(new Multithreading());
t4=new Thread(new Multithreading());
t5=new Thread(new Multithreading());

                  t1.setName("MCA");
t2.setName("BCA");            t3.setName("BSC(IT)");
         t4.setName("MSC(IT)");
                  t5.setName("BTech");

                  t1.start();
t2.start();                t3.start();
         t4.start();
                  t5.start();
   }
}
```

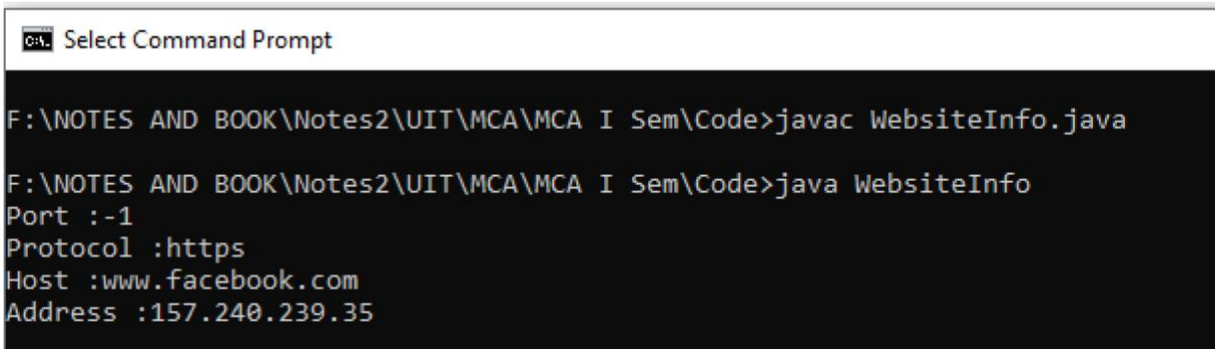
**Output**



**Practical 13: Write a program to access a website details such as its IP address, protocols and port number.**

```
import java.net.URL; import
java.net.InetAddress; public
class WebsiteInfo
{
  public static void main(String[] args)
  {
        try
        {
              URL url = new URL("https://www.facebook.com/");
              InetAddress in = InetAddress.getByName("www.facebook.com");
              System.out.println("Port :" + url.getPort());
              System.out.println("Protocol :" + url.getProtocol());
              System.out.println("Host :" + url.getHost());
              System.out.println("Address :" + in.getHostAddress());
              }
              catch (Exception ex)
              {
                    System.err.println(ex);
              }
        }
}
```

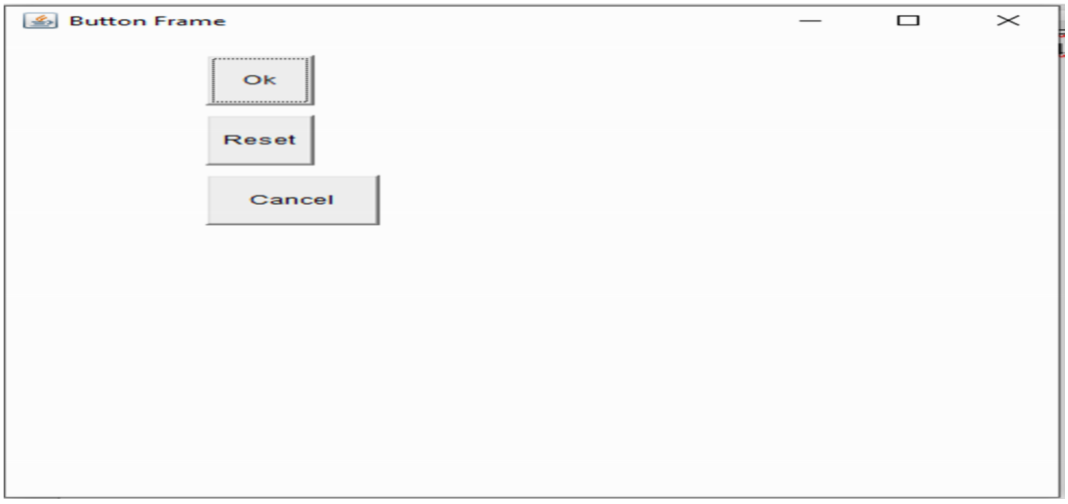
**Output**



**Practical 14. Write a program to create three buttons with caption OK, RESET, CANCEL.**

```
import java.awt.*; import
java.awt.event.WindowEvent;     import
java.awt.event.WindowListener;  import
java.awt.event.*;

public class AwtFrame extends Frame
{
        AwtFrame()
        {
          Button b1=new Button("Ok");
b1.setBounds(100,50,50,50);
          add(b1);

          Button b2=new Button("Reset");
b2.setBounds(100,110,50,50);
          add(b2);

          Button b3=new Button("Cancel");
b3.setBounds(100,170,80,50);
          add(b3);

                addWindowListener (new WindowAdapter() {
                        public void windowClosing (WindowEvent e) {
dispose();
                }
      });
          setTitle("Button Frame");
          setSize(500,500);
setLayout(null);
          setVisible(true);
        }
        public static void main(String a[])
        {
          new AwtFrame();
        }
}
```

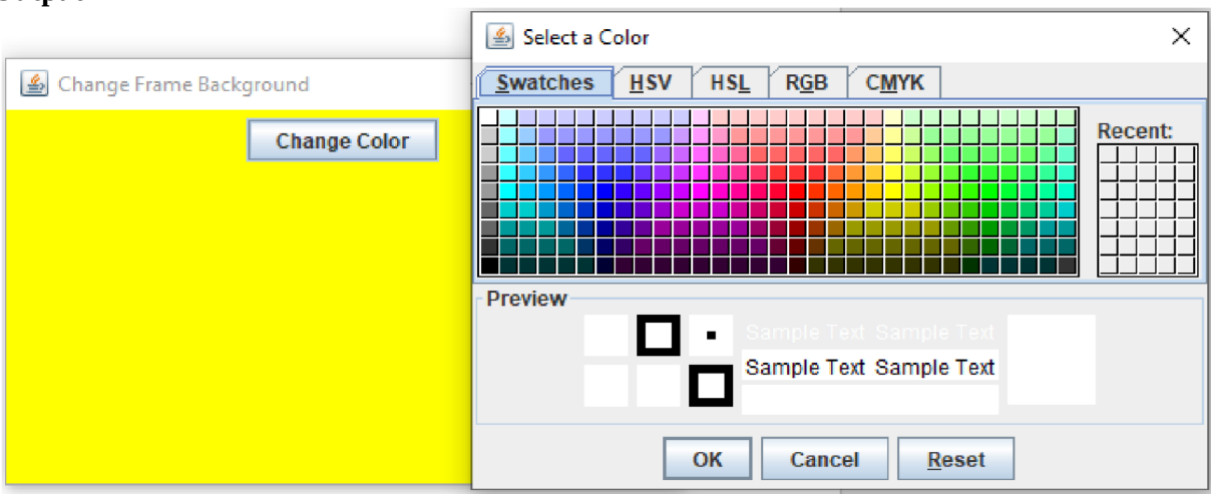
**Output**



**Practical 15 : Write a program in java to change the background color of frame on click of button.**

```
import javax.swing.*;
import java.awt.*; import
java.awt.event.*;
class Frame_Color implements ActionListener
{
   static JFrame frame;
   public static void main(String args[])
   {

        frame = new JFrame("Change Frame Background");
frame.setSize(400,400);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.getContentPane().setBackground(Color.white);     frame.setLayout(new
FlowLayout());



        Frame_Color obj = new Frame_Color();
        //Create a button
        JButton button = new JButton("Change Color");
button.addActionListener(obj);   frame.add(button);
frame.setVisible(true);
   }
   //Function to create color dialog box and change color
public void actionPerformed(ActionEvent e)
   {

        //Create a color dialog box

        JColorChooser color_box= new JColorChooser();
        Color color=color_box.showDialog(frame,"Select a Color",Color.white);

        //Change background color of frame

        frame.getContentPane().setBackground(color);
   }
}
```

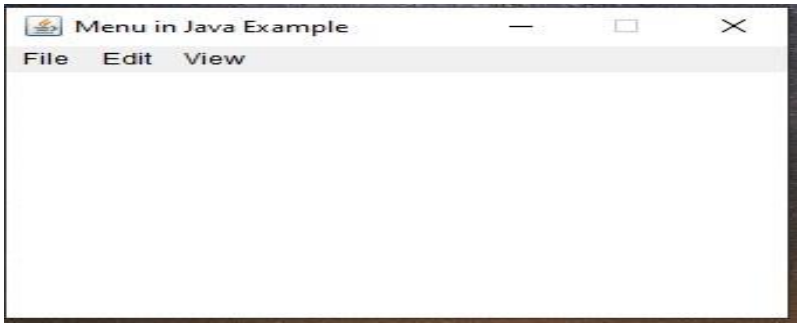
**Output**



**Practical 16: Write a program in java to create a menu bar in frame.**

```
import java.awt.*; import
java.awt.event.WindowEvent;     import
java.awt.event.WindowListener;  import
java.awt.event.*;  class MenuExample
extends Frame
 {
       MenuExample()
      {
          MenuBar menuBar = new MenuBar();
setMenuBar(menuBar);
          Menu menuFile = new Menu("File");
          Menu menuEdit = new Menu("Edit");
          Menu menuView = new Menu("View");

          menuBar.add(menuFile);
menuBar.add(menuEdit);
          menuBar.add(menuView);

          MenuItem itemOpen = new MenuItem("Open");
          MenuItem itemSave = new MenuItem("Save");
          MenuItem itemExit = new MenuItem("Exit");

          menuFile.add(itemOpen);
menuFile.add(itemSave);
          menuFile.add(itemExit);

          MenuItem itemcopy = new MenuItem("Copy");
          menuEdit.add(itemcopy);

                              addWindowListener (new WindowAdapter() {
          public void windowClosing (WindowEvent e) {
             System.exit(0);
                              }});
     }
 }
  class MenuBarExample
  {
         public static void main(String args[])
        {
            MenuExample frame = new MenuExample();
            frame.setTitle("Menu in Java Example");
            frame.setSize(350,250);
frame.setResizable(false);
            frame.setVisible(true);
        }
  }
```

**Output**



**Practical 17: Create A Java Applet That Works As A Simple Calculator. Use A Grid Layout to Arrange Buttons For The Digits and For The +,-,*, % Operations. Add A Text Field to Display The Result. Handle Any Possible Exceptions Like Divide By Zero.**

```
import java.awt.*; import
java.awt.event.*; import
java.applet.*; import
javax.swing.*;
/*
<applet code="MyCalculator" width=300 height=300>
</applet>
*/
public class MyCalculator extends Applet implements ActionListener {
int num1,num2,result;
   TextField T1;
   Button NumButtons[]=new Button[10];
Button Add,Sub,Mul,Div,clear,EQ;
   char Operation;
   Panel nPanel,CPanel,SPanel;
   public void init() {
nPanel=new Panel();
T1=new TextField(30);
      nPanel.setLayout(new FlowLayout(FlowLayout.CENTER));
nPanel.add(T1);        CPanel=new Panel();
      CPanel.setBackground(Color.white);
      CPanel.setLayout(new GridLayout(5,5,3,3));
      for(int i=0;i<10;i++) {
            NumButtons[i]=new Button(""+i);
         }
      Add=new Button("+");
      Sub=new Button("-");
      Mul=new Button("*");
Div=new Button("/");
clear=new Button("clear");
EQ=new Button("=");
T1.addActionListener(this);
for(int i=0;i<10;i++) {
         CPanel.add(NumButtons[i]);
      }
      CPanel.add(Add);
      CPanel.add(Sub);
      CPanel.add(Mul);
      CPanel.add(Div);
      CPanel.add(EQ);
      SPanel=new Panel();
      SPanel.setLayout(new FlowLayout(FlowLayout.CENTER));
      SPanel.setBackground(Color.yellow);
SPanel.add(clear);        for(int
i=0;i<10;i++) {
         NumButtons[i].addActionListener(this);
```

```
        }
        Add.addActionListener(this);
        Sub.addActionListener(this);
        Mul.addActionListener(this);
Div.addActionListener(this);
clear.addActionListener(this);
EQ.addActionListener(this);         this.setLayout(new
BorderLayout());
add(nPanel,BorderLayout.NORTH);
add(CPanel,BorderLayout.CENTER);
        add(SPanel,BorderLayout.SOUTH);
    }
    public void actionPerformed(ActionEvent ae) {
String str=ae.getActionCommand ();         char
ch=str.charAt(0);         if(Character.isDigit(ch))
T1.setText(T1.getText()+str);         else
if(str.equals("+")){
              num1=Integer.parseInt (T1.getText());
              Operation='+';
              T1.setText ("");
           }
if(str.equals("-")){
              num1=Integer.parseInt(T1.getText());
              Operation='-';
              T1.setText("");
           }
if(str.equals("*")){
              num1=Integer.parseInt(T1.getText());
              Operation='*';
              T1.setText("");
           }
if(str.equals("/")){
              num1=Integer.parseInt(T1.getText());
              Operation='/';
              T1.setText("");
           }
           if(str.equals("%")){
              num1=Integer.parseInt(T1.getText());
              Operation='%';
              T1.setText("");
           }            if(str.equals("=")) {
num2=Integer.parseInt(T1.getText());
              switch(Operation)
              {
                 case '+':result=num1+num2;
                       break;                    case
'-':result=num1-num2;
break;                    case
'*':result=num1*num2;
break;                    case '/':try {
                        result=num1/num2;
```

```
                    }
                    catch(ArithmeticException e) {
                       result=num2;
                       JOptionPane.showMessageDialog(this,"Divided by zero");
                    }
break;
            }
            T1.setText(""+result);
         }
         if(str.equals("clear")) {
            T1.setText("");
         }
      }
}
```

**Output**

**Practical 18: Create a java program to create a registration form and perform basic sql commands such as create, insert, update and delete.**

```
package mca.i.sem;

import java.sql.PreparedStatement;
import java.sql.ResultSet; import
java.sql.SQLException; import
java.sql.Statement; import
java.util.logging.Level; import
java.util.logging.Logger;
import javax.swing.JOptionPane;

public class Registration extends javax.swing.JFrame {

   public Registration() {
      initComponents();
   }

   private void initComponents() {

      jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();       jLabel3 =
new javax.swing.JLabel();       jLabel4 = new
javax.swing.JLabel();       jLabel5 = new
javax.swing.JLabel();       jLabel6 = new
javax.swing.JLabel();       jLabel7 = new
javax.swing.JLabel();       s = new
javax.swing.JCheckBox();       ss = new
javax.swing.JCheckBox();       graduation = new
javax.swing.JCheckBox();       pg = new
javax.swing.JCheckBox();       jLabel8 = new
javax.swing.JLabel();       jScrollPane1 = new
javax.swing.JScrollPane();       address = new
javax.swing.JTextArea();       insert = new
javax.swing.JButton();       update = new
javax.swing.JButton();       delete = new
javax.swing.JButton();       search = new
javax.swing.JButton();       s_id = new
javax.swing.JTextField();       s_name = new
javax.swing.JTextField();       s_father_name = new
javax.swing.JTextField();       s_mother_name = new
javax.swing.JTextField();       s_contact = new
javax.swing.JTextField();       delete1 = new
javax.swing.JButton();       jLabel9 = new
javax.swing.JLabel();

      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setResizable(false);
      getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
```

```
jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 36)); // NOI18N
jLabel1.setForeground(new java.awt.Color(0, 255, 204));  jLabel1.setText("Student
Registration Form");  getContentPane().add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(165, 6, 410,  39));

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel2.setForeground(new java.awt.Color(0, 255, 204));  jLabel2.setText("Student Name");
getContentPane().add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 100, 170, -
1));

jLabel3.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel3.setForeground(new java.awt.Color(0, 255, 204));  jLabel3.setText("Student ID");
getContentPane().add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 70, 140, -
1));

jLabel4.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel4.setForeground(new java.awt.Color(0, 255, 204));  jLabel4.setText("Father Name");
getContentPane().add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 140,  140, -
1));

jLabel5.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel5.setForeground(new java.awt.Color(0, 255, 204));  jLabel5.setText("Mother Name");
getContentPane().add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 180,  150, -
1));

jLabel6.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel6.setForeground(new java.awt.Color(0, 255, 204));  jLabel6.setText("Contact Name");
getContentPane().add(jLabel6, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 220, -1, 1));

jLabel7.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel7.setForeground(new java.awt.Color(0, 255, 204));  jLabel7.setText("Student
Qualification");  getContentPane().add(jLabel7, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 260, 240, -1));

s.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
s.setForeground(new java.awt.Color(255, 255, 0));
s.setText("10 th");
s.setOpaque(false);  getContentPane().add(s, new
org.netbeans.lib.awtextra.AbsoluteConstraints(40, 290, 100, -1));

ss.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
ss.setForeground(new java.awt.Color(255, 255, 0));  ss.setText("12 th");
ss.setOpaque(false);
getContentPane().add(ss, new org.netbeans.lib.awtextra.AbsoluteConstraints(140, 290, 100,
1));

graduation.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
graduation.setForeground(new java.awt.Color(255, 255, 0));
graduation.setText("Graduation");  graduation.setOpaque(false);
getContentPane().add(graduation,  new
   org.netbeans.lib.awtextra.AbsoluteConstraints(230,  290, 170, -1));
```

```
        pg.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
pg.setForeground(new java.awt.Color(255, 255, 0));         pg.setText("Post
Graduation");         pg.setOpaque(false);
        getContentPane().add(pg, new org.netbeans.lib.awtextra.AbsoluteConstraints(410, 290, 220, -
          1));

        jLabel8.setFont(new java.awt.Font("Times New Roman", 1, 24)); // NOI18N
jLabel8.setForeground(new java.awt.Color(0, 255, 204));         jLabel8.setText("Address");
        getContentPane().add(jLabel8, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 340,
          110, -1));

        address.setColumns(20);
        address.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
address.setRows(5);
        jScrollPane1.setViewportView(address);

        getContentPane().add(jScrollPane1, new org.netbeans.lib.awtextra.AbsoluteConstraints(110,
          330, 496, 57));

        insert.setBackground(new java.awt.Color(153, 153, 0));
insert.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
insert.setForeground(new java.awt.Color(204, 0, 0));
insert.setText("Insert Record");         insert.setOpaque(false);
        insert.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
insertActionPerformed(evt);
            }
        });
        getContentPane().add(insert, new org.netbeans.lib.awtextra.AbsoluteConstraints(40, 410, -1, -
          1));

        update.setBackground(new java.awt.Color(153, 153, 0));
update.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
update.setForeground(new java.awt.Color(204, 0, 0));
update.setText("Update Record");         update.setOpaque(false);
update.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
updateActionPerformed(evt);
            }
        });
        getContentPane().add(update, new org.netbeans.lib.awtextra.AbsoluteConstraints(190, 410, -1,
          -1));

        delete.setBackground(new java.awt.Color(153, 153, 0));
delete.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
delete.setForeground(new java.awt.Color(204, 0, 0));
delete.setText("Delete Record");         delete.setOpaque(false);
delete.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
deleteActionPerformed(evt);
            }
```

```
        });
        getContentPane().add(delete, new org.netbeans.lib.awtextra.AbsoluteConstraints(350, 410, -1,
         -1));

        search.setBackground(new java.awt.Color(153, 153, 0));
search.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
search.setForeground(new java.awt.Color(204, 0, 0));
search.setText("Search");         search.setOpaque(false);
search.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
searchActionPerformed(evt);
            }
        });
        getContentPane().add(search, new org.netbeans.lib.awtextra.AbsoluteConstraints(440, 56, 160,
         -1));

        s_id.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
        getContentPane().add(s_id, new org.netbeans.lib.awtextra.AbsoluteConstraints(180, 60, 254, -
         1));

        s_name.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
        getContentPane().add(s_name, new org.netbeans.lib.awtextra.AbsoluteConstraints(180, 100,
         420, -1));

        s_father_name.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
        getContentPane().add(s_father_name, new org.netbeans.lib.awtextra.AbsoluteConstraints(180,
         140, 420, -1));

        s_mother_name.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
getContentPane().add(s_mother_name,new org.netbeans.lib.awtextra.AbsoluteConstraints(180,
180, 420, -1));

        s_contact.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N
        getContentPane().add(s_contact, new org.netbeans.lib.awtextra.AbsoluteConstraints(180, 220,
         420, -1));

        delete1.setBackground(new java.awt.Color(153, 153, 0));
        delete1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
        delete1.setForeground(new java.awt.Color(204, 0, 0));
delete1.setText("Clear");         delete1.setOpaque(false);
delete1.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEvent evt) {
delete1ActionPerformed(evt);
            }
        });
        getContentPane().add(delete1, new org.netbeans.lib.awtextra.AbsoluteConstraints(500, 410,
         109, -1));

        jLabel9.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Image/wood.png"))); //
         NOI18N
```

```
        getContentPane().add(jLabel9, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 640,
         460));

        pack();
        setLocationRelativeTo(null);
    }
private void updateActionPerformed(java.awt.event.ActionEvent evt) {


        Statement stmt;

        if(s_id.getText().equals("")|| s_name.getText().equals("") || s_father_name.getText().equals("")
|| s_mother_name.getText().equals("") ||

                s_contact.getText().equals("") || address.getText().equals("")){

        JOptionPane.showMessageDialog(null, "All Fields are mendatory to fill");

        }

        else if(!s.isSelected() || !ss.isSelected() || !graduation.isSelected() || !pg.isSelected())

        {

           JOptionPane.showMessageDialog(null, "All Fields are mendatory to fill");

        }

else


        {

          int id= Integer.parseInt(s_id.getText());

          String query="UPDATE studentregistration set
S_Name='"+s_name.getText()+"',Father_Name='"+s_father_name.getText()+"',Mother_Name='"+s
_mother_name.getText()+"',Contact='"+s_contact.getText()+"', High_School='"+"YES"+ "' ,
Senior_Secondary='"+"YES"+

             "' , Graduation='"+"YES"+"', Post_Graduation='"+"YES"+"',
Address='"+address.getText()+"' where S_ID="+id;


//'",Father_Name='"+s_father_name.getText()+"',Mother_Name='"+s_mother_name.getText()+
//'",Contact='"+s_contact.getText()+"', High_School='"+"YES"+ "' , Senior_Secondary='"+"YES"+

             //"' , Graduation='"+"YES"+"', Post_Graduation='"+"YES"+"',
Address='"+address.getText()+


          LoadDriver ld=new LoadDriver();

try {
```

```
        stmt= ld.con.createStatement();
int i= stmt.executeUpdate(query);
if(i>0)
                JOptionPane.showMessageDialog(null, "Data Updated");
else
                JOptionPane.showMessageDialog(null, "Error : Data not Updated");


            //JOptionPane.showMessageDialog(null, "All Fields fill " + s.isSelected());
        } catch (SQLException ex) {
            Logger.getLogger(Registration.class.getName()).log(Level.SEVERE, null, ex);
JOptionPane.showMessageDialog(null, ex);
        }
    }
  }

   private void insertActionPerformed(java.awt.event.ActionEvent evt) {
if(s_id.getText().equals("")|| s_name.getText().equals("") || s_father_name.getText().equals("")
|| s_mother_name.getText().equals("") ||
            s_contact.getText().equals("") || address.getText().equals("")){
      JOptionPane.showMessageDialog(null, "All Fields are mendatory to fill");
      }
      else if(!s.isSelected() || !ss.isSelected() || !graduation.isSelected() || !pg.isSelected())
      {
         JOptionPane.showMessageDialog(null, "All Fields are mendatory to fill");
      }
else  {


        String query="INSERT INTO studentregistration VALUES('"+s_id.getText() +"',
'"+s_name.getText()+"', '"+s_father_name.getText()+"',
'"+s_mother_name.getText()+"','"+s_contact.getText()+"', 'YES', 'YES', 'YES',
'YES','"+address.getText()+"')";
```

```
        LoadDriver ld=new LoadDriver();
try {
            Statement stmt= ld.con.createStatement();
int r= stmt.executeUpdate(query);               if(r>0)
                JOptionPane.showMessageDialog(null, "Data saved");
else
                JOptionPane.showMessageDialog(null, "Error : Data not saved");



            //JOptionPane.showMessageDialog(null, "All Fields fill " + s.isSelected());
        } catch (SQLException ex) {
            //Logger.getLogger(RegistrationFrame.class.getName()).log(Level.SEVERE, null, ex);
JOptionPane.showMessageDialog(null, ex);
        }
    }
  }

   private void searchActionPerformed(java.awt.event.ActionEvent evt) {
      String query="SELECT * FROM studentregistration where S_ID="'+s_id.getText()+"'";
LoadDriver ld=new LoadDriver();               try{
            Statement stmt= ld.con.createStatement();
            ResultSet rs= stmt.executeQuery(query);


            if(rs.next()){
             s_name.setText(rs.getString("S_Name"));
s_father_name.setText(rs.getString("Father_Name"));
s_mother_name.setText(rs.getString("Mother_Name"));
s_contact.setText(rs.getString("Contact"));               address.setText(rs.getString("Address"));
if(rs.getString("High_School").equals("YES"))                   s.setSelected(true);
             if(rs.getString("Senior_Secondary").equals("YES"))
ss.setSelected(true);
```

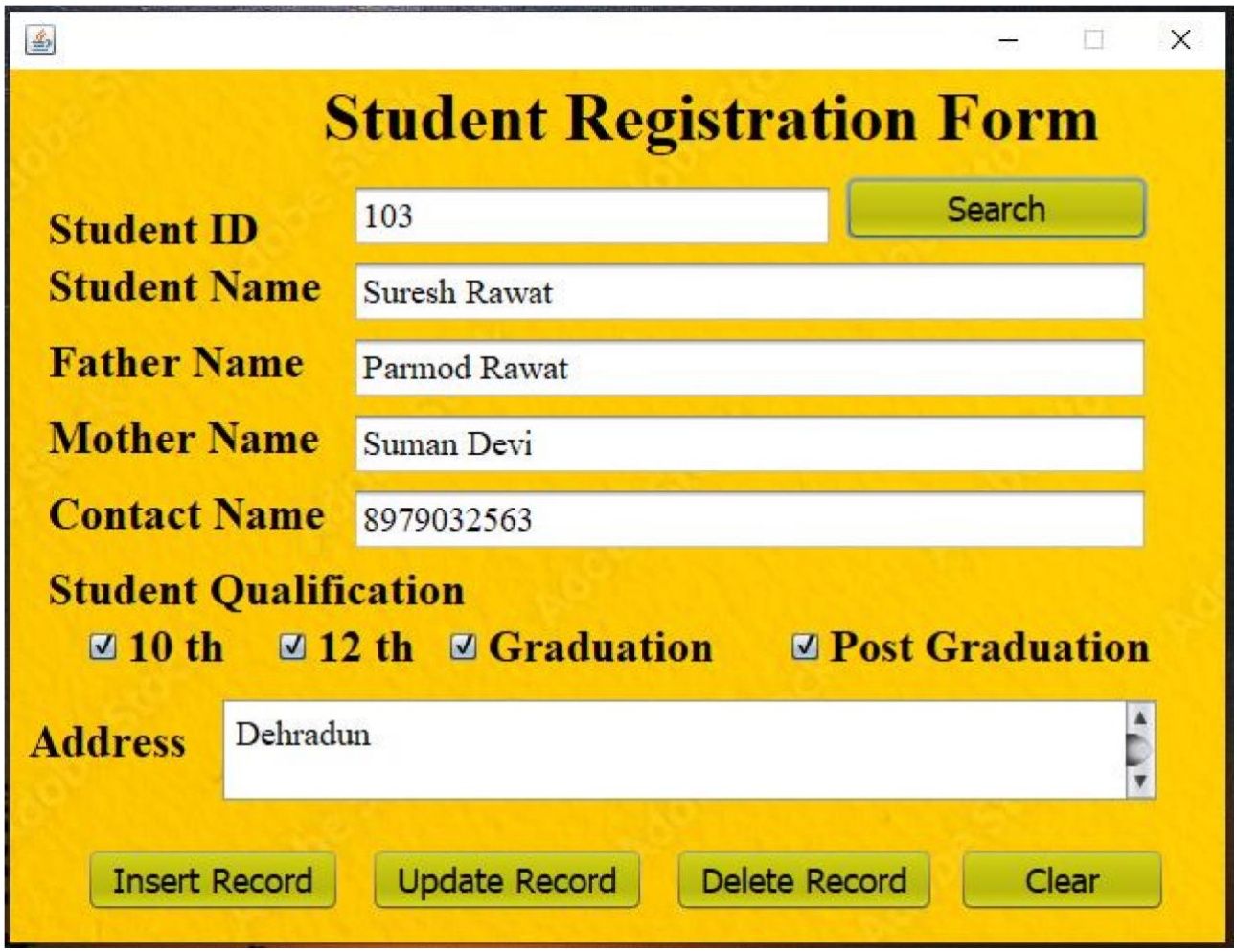
```
            if(rs.getString("Graduation").equals("YES"))
graduation.setSelected(true);

            if(rs.getString("Post_Graduation").equals("YES"))
pg.setSelected(true);

            }



            //JOptionPane.showMessageDialog(null, "All Fields fill " + s.isSelected());
        } catch (Exception ex) {
            //Logger.getLogger(RegistrationFrame.class.getName()).log(Level.SEVERE, null, ex);
JOptionPane.showMessageDialog(null, ex);

        }
    }
    private void deleteActionPerformed(java.awt.event.ActionEvent evt) {


        PreparedStatement pstmt;
        try {
            pstmt = new LoadDriver().con.prepareStatement("delete from studentregistration  where
            S_ID=?");
          int id= Integer.parseInt(s_id.getText());
pstmt.setInt(1, id);


        int i=pstmt.executeUpdate();
if(i>0)

        JOptionPane.showMessageDialog(null, " records deleted");
else

         JOptionPane.showMessageDialog(null, " records not found");


        } catch (SQLException ex) {
Logger.getLogger(Registration.class.getName()).log(Level.SEVERE, null, ex);
        }
```

```
    }

    private void delete1ActionPerformed(java.awt.event.ActionEvent evt) {
s_id.setText("");        s_name.setText("");        s_father_name.setText("");
s_mother_name.setText("");        s_contact.setText("");        address.setText("");
s.setSelected(false);        ss.setSelected(false);        graduation.setSelected(false);
pg.setSelected(false);

    }
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */                    try    {                                for
(javax.swing.UIManager.LookAndFeelInfo          info          :
javax.swing.UIManager.getInstalledLookAndFeels())        {
if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
break;
            }
          }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Registration.class.getName()).log(java.util.logging.Level.SEVE
RE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Registration.class.getName()).log(java.util.logging.Level.SEVE
RE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Registration.class.getName()).log(java.util.logging.Level.SEVE
RE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Registration.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
public void run() {
                new Registration().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
private javax.swing.JTextArea address;     private
javax.swing.JButton delete;     private
javax.swing.JButton delete1;     private
javax.swing.JCheckBox graduation;     private
javax.swing.JButton insert;     private
javax.swing.JLabel jLabel1;     private
javax.swing.JLabel jLabel2;     private
javax.swing.JLabel jLabel3;     private
javax.swing.JLabel jLabel4;     private
javax.swing.JLabel jLabel5;     private
javax.swing.JLabel jLabel6;     private
javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;     private
javax.swing.JLabel jLabel9;     private
javax.swing.JScrollPane jScrollPane1;     private
javax.swing.JCheckBox pg;     private
javax.swing.JCheckBox s;     private
javax.swing.JTextField s_contact;     private
javax.swing.JTextField s_father_name;     private
javax.swing.JTextField s_id;     private
```

```
javax.swing.JTextField s_mother_name;     private
javax.swing.JTextField s_name;     private
javax.swing.JButton search;     private
javax.swing.JCheckBox ss;     private
javax.swing.JButton update;

    // End of variables declaration
}
```

**Output**



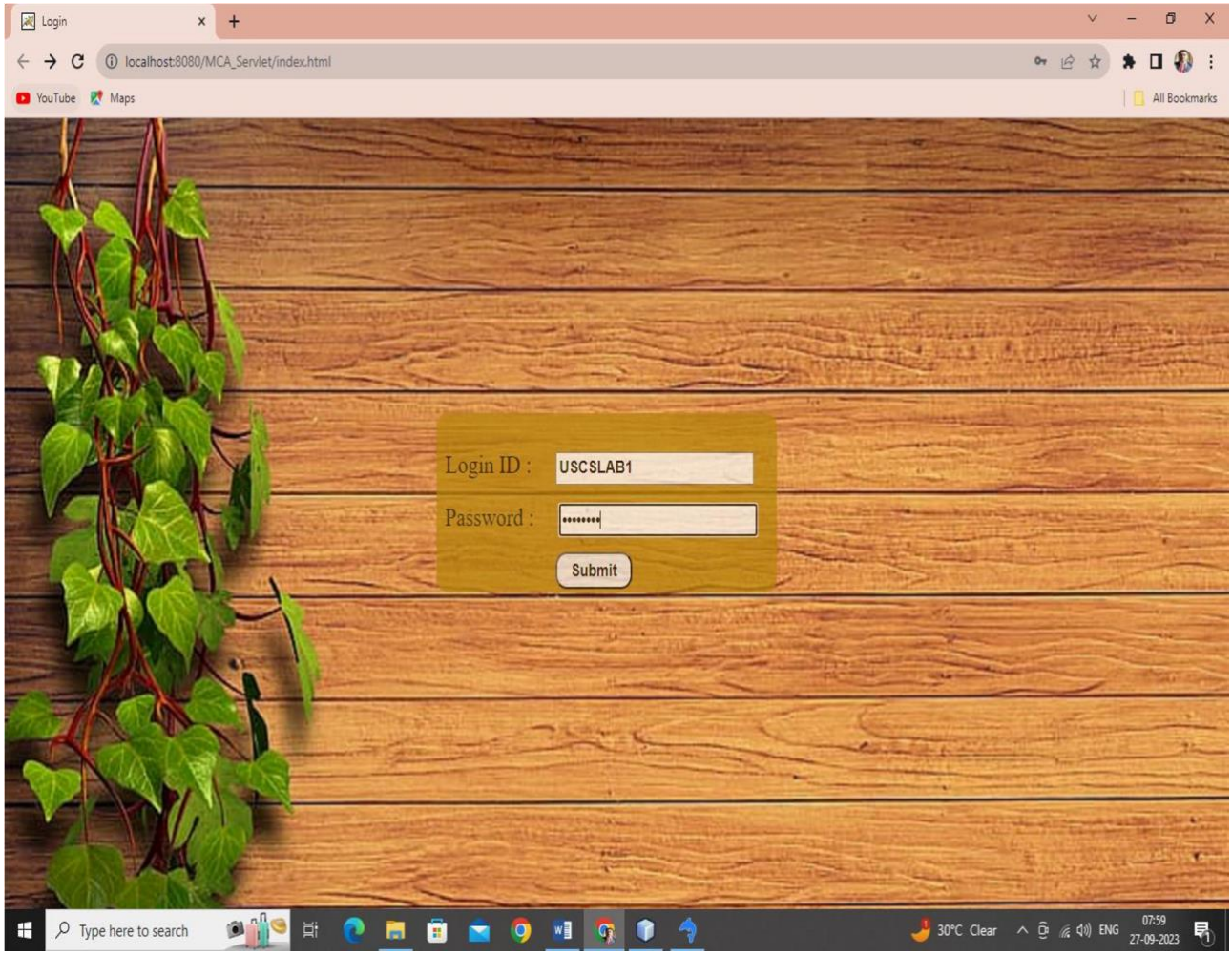
**Practical 19: Create a program in java servlet to verify the login ID and password from database, if it is correct show message login successful, else correct login ID and password.**

**Index.html**

```
<!DOCTYPE html>
<html>
   <head>
      <title>Login</title>
      <meta charset="UTF-8">
      <meta name="viewport" content="width=device-width, initial-scale=1.0">

      <style>
label{

            font-size: 24px;

            margin: 0px 10px;

}
input{

            width: 225px;
height: 25px;
            margin: 0px 15px;
         }
div {
         opacity: 0.7;

         margin: 280px 500px;
         width: 400px;
height: 170px;
         background:darkgoldenrod;
border-radius: 13px;
         }
.button{
           border-radius: 13px;
height: 35px;              width:
90px;             margin: 15px
140px;

}
body{

            background-image: url('wood3.png');
         }            input{
font-size: 16px;
            font-weight: bold;
         }

      </style>
   </head>
```

```
  <body>
    <form  method="GET" action="AccessData">
      <div>
        <br><br><label >Login ID :</label>
        <input type="text" placeholder="Enter Your ID" name="id"><br><br>
      <label >Password :</label>
      <input type="password" placeholder="Enter Your password" name="password"><br>

      <input class=button type="Submit" value="Submit">

      </div>


    </form>

  </body>
</html>
```

**Output**




**AccessData.java**

```
import java.io.IOException; import
java.io.PrintWriter; import java.sql.ResultSet;
import java.sql.SQLException; import
```

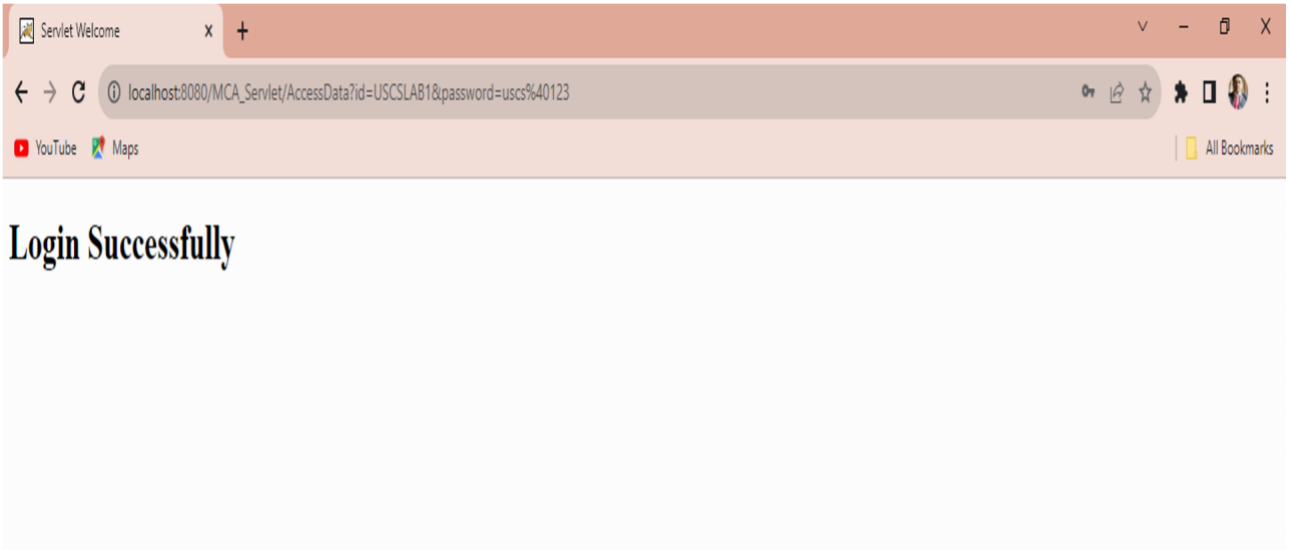
```
java.sql.Statement; import
java.util.logging.Level; import
java.util.logging.Logger; import
javax.servlet.RequestDispatcher; import
javax.servlet.ServletException; import
javax.servlet.http.HttpServlet; import
javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class AccessData extends HttpServlet {

  protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");          PrintWriter out =
response.getWriter();
      String id=request.getParameter("id");
      String p=request.getParameter("password");

      String query="select * from login where S_ID='"+id+"' and Password= '"+p+"'";
      Statement stmt;
try {
         stmt=new LoadDriver().con.createStatement();
         ResultSet rs= stmt.executeQuery(query);

         if(rs.next())
{
           RequestDispatcher rd=request.getRequestDispatcher("Welcome");
rd.forward(request, response);
         }
else           {
          //RequestDispatcher rd=request.getRequestDispatcher("Sorry");
          // rd.forward(request, response);
            response.sendRedirect("Sorry");
         }

      } catch (SQLException ex) {
         Logger.getLogger(AccessData.class.getName()).log(Level.SEVERE, null, ex);
      }


   }
}
```

**Welcome.java**

```
import java.io.IOException; import
java.io.PrintWriter; import
javax.servlet.ServletException; import
javax.servlet.http.HttpServlet; import
javax.servlet.http.HttpServletRequest; import
javax.servlet.http.HttpServletResponse;
```

```
public class Welcome extends HttpServlet {


   protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
       try (PrintWriter out = response.getWriter()) {
           /* TODO output your page here. You may use following sample code. */
out.println("<!DOCTYPE html>");
           out.println("<html>");
out.println("<head>");
           out.println("<title>Servlet Welcome</title>");
out.println("</head>");            out.println("<body>");
           out.println("<h1> Login Successfully</h1>");
           out.println("</body>");
           out.println("</html>");
       }
   }

}
```

**Output**




**Sorry.java**

```
import java.io.IOException; import
java.io.PrintWriter; import
javax.servlet.ServletException; import
javax.servlet.http.HttpServlet; import
javax.servlet.http.HttpServletRequest; import
javax.servlet.http.HttpServletResponse;


public class Sorry extends HttpServlet {
```

```
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
       try (PrintWriter out = response.getWriter()) {
           /* TODO output your page here. You may use following sample code. */
out.println("<!DOCTYPE html>");
           out.println("<html>");
out.println("<head>");
           out.println("<title>Servlet Sorry</title>");
out.println("</head>");            out.println("<body>");
           out.println("<h1>Sorry : Re-enter your login ID and Password</h1>");
out.println("</body>");
           out.println("</html>");
       }
   }
}
```

**Output**