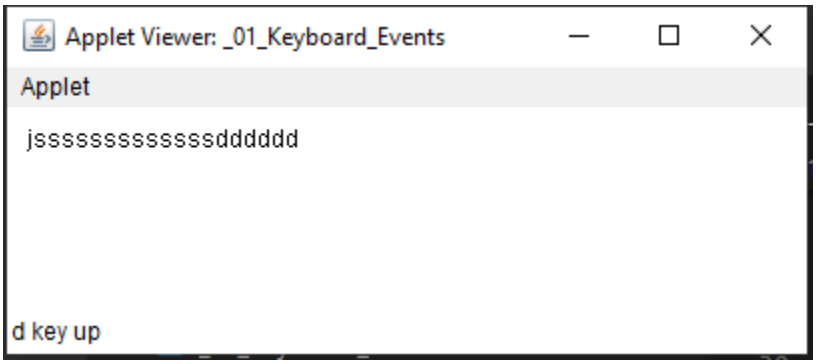
1. Program for AWT KeyEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

/*
<applet code="_01_Keyboard_Events" width="400" height="100" > </applet>
*/

// In case of Applet need to declare class Public else applet will not be able to access class
// KeyListener is from awt.event.KeyListener
public class _01_Keyboard_Events extends Applet implements KeyListener{

String msg = "";
// init is from java.applet.Applet
public void init(){
addKeyListener(this);
requestFocus();    //*** what is this doing ***
}

@Override
public void keyPressed(KeyEvent e) {
// showStatus was from java.applet.Applet
```

```
showStatus(e.getKeyChar() + " key down");
}

@Override
public void keyReleased(KeyEvent e) {
showStatus(e.getKeyChar() + " key up");
}

// repaint was from awt.Component.repaint
@Override
public void keyTyped(KeyEvent e) {
// showStatus(e.getKeyChar() + " key typed");

msg += e.getKeyChar();
repaint();
}

// output text you typing
public void paint(Graphics g){
g.drawString( msg, 10, 20 );
}

}
```

Output:



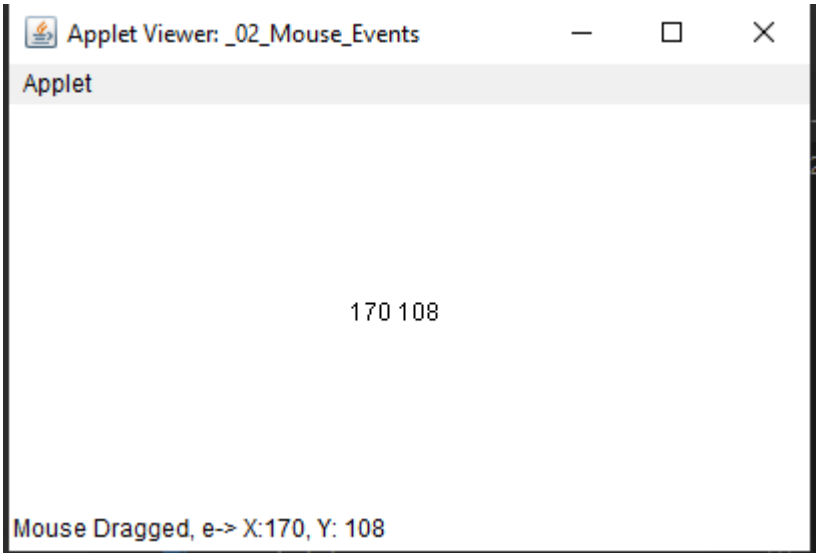
2. Program for AWT MouseEvent, MouseMotionEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="_02_Mouse_Events" height="200" width="400" ></applet>
*/

public class _02_Mouse_Events extends Applet implements MouseListener,
MouseMotionListener {

int X, Y;

public void init(){
addMouseListener(this);
```

```
addMouseMotionListener(this);
}

@Override
public void mouseClicked(MouseEvent e) {
showStatus("Mouse clicked");
}

@Override
public void mouseEntered(MouseEvent e) {
showStatus("Mouse Entered");
}

@Override
public void mouseExited(MouseEvent e) {
showStatus("Mouse Exited");
}

// *****  ?  *****
@Override
public void mousePressed(MouseEvent e) {
showStatus("Mouse Pressed");
}
```

```
@Override
public void mouseReleased(MouseEvent e) {
showStatus("Mouse Release");
}

@Override
public void mouseDragged(MouseEvent e) {
showStatus("Mouse Dragged, e-> X:" + e.getX() + ", Y: " + e.getY());
X = e.getX();
Y = e.getY();
repaint();
}

@Override
public void mouseMoved(MouseEvent e) {
showStatus("Mouse Moved");
}

public void paint(Graphics g){
g.drawString(X +" "+ Y, X, Y);
}
}
```

Output:

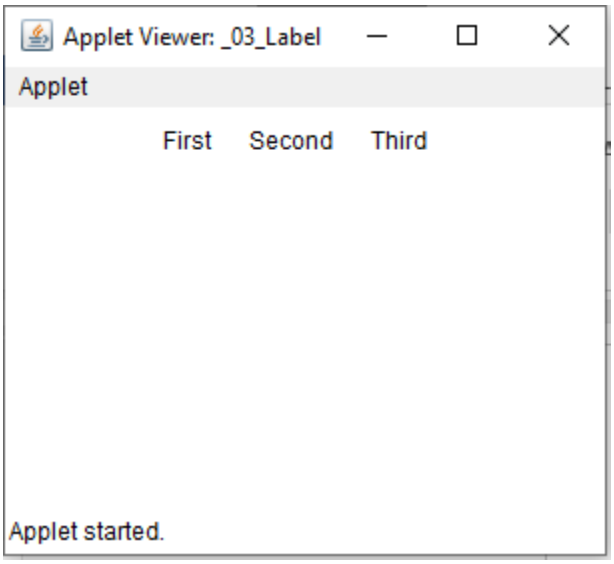


3. Program for AWT Labels.

```
import java.awt.*;
import java.applet.*;

// <applet code="_03_Label" height="300"width="300" > </applet>
public class _03_Label extends Applet {
Label l1, l2, l3;
public void init(){
l1 = new Label("Shubham");
l2 = new Label("Dahiya");
l3 = new Label("HIM");
```

```
// adding to the window of applet
add(l1);
add(l2);
add(l3);
}
}
```

Output:



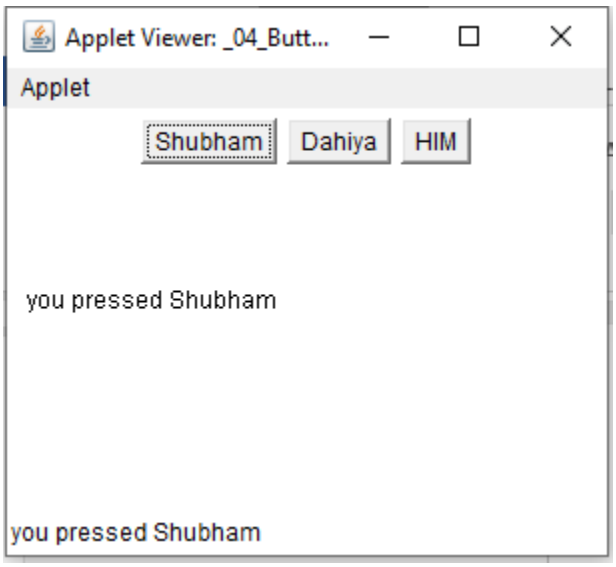
4. Program for AWT Button & ActionEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
```

```
// <applet code="_04_Button" height="200"width="300" > </applet>

public class _04_Button extends Applet implements ActionListener {
Button b1, b2, b3;
String msg = "";
public void init(){
b1 = new Button("Shubham");
b2 = new Button("Dahiya");
b3 = new Button("HIM");

// adding to applet
add(b1);
add(b2);
add(b3);

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
}
@Override
public void actionPerformed(ActionEvent e) {
msg = "you pressed " + e.getActionCommand();

showStatus("you pressed " + e.getActionCommand() );
```

```
repaint();
}

public void paint(Graphics g){
g.drawString(msg, 10, 100);
}
}
```

Output:



5. Program for AWT TextField.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
```

```
// <applet code="_05_TextField" height="200"width="400" > </applet>

public class _05_TextField extends Applet implements ActionListener{
TextField name, password;
Label l1, l2;
String s1= "", s2 ="";
public void init(){
l1 = new Label("name", Label.LEFT);
l2 = new Label("password", Label.RIGHT);

name= new TextField(12);
password= new TextField(12);

add(l1);
add(name);
add(l2);
add(password);

name.addActionListener(this);
password.addActionListener(this);
}

@Override
```

```
public void actionPerformed(ActionEvent e) {
s1 = "name:" + e.getActionCommand();
s2 = "Password: " +e.getActionCommand();
repaint();
}

public void paint(Graphics g){
g.drawString(s1, 10, 100);
g.drawString(s2, 10, 120);
}
}
```

Output:

6. Program for AWT List.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

// <applet code="_06_List" height="200"width="500" > </applet>

public class _06_List extends Applet implements ActionListener {
List grocery, dev_field;
public void init(){
grocery = new List(5, true);
dev_field = new List(4, false );

grocery.add("apple");
grocery.add("Mango");
grocery.add("Grapes");
grocery.add("Banana");
grocery.add("Pineapple");
grocery.add("Strawberry");
grocery.add("Pulse");
grocery.add("Ladyfinger");
grocery.add("Potato");
grocery.add("Tomato");
```

```
grocery.select(2);

dev_field.add("Web Development");
dev_field.add("Android Development");
dev_field.add("iOS Development");
dev_field.select(0);

add(grocery);
add(dev_field);

grocery.addActionListener(this);
dev_field.addActionListener(this);
}
@Override
public void actionPerformed(ActionEvent e) {
repaint();
}

public void paint(Graphics g){
String gr = "Current selcted groceries are: ";

int idx[] = grocery.getSelectedIndexes();
for(int i:idx)
gr+=grocery.getItem(i) +",";
```

```
String f = "Currently selected dev_field ";
f += dev_field.getSelectedItem();

g.drawString(gr, 10, 100);
g.drawString(f, 10, 120);

}
}
```

Output:

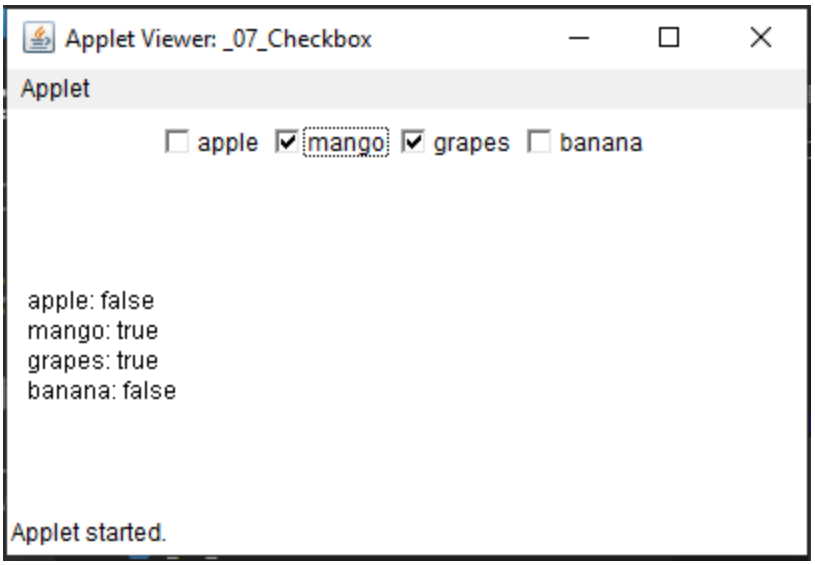
7. Program for AWT Checkbox & ItemEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

// <applet code="_07_Checkbox" height="200"width="400" > </applet>

public class _07_Checkbox extends Applet implements ItemListener {
Checkbox apple, mango, grapes, banana;
public void init(){
apple = new Checkbox("apple");
mango = new Checkbox("mango");
grapes = new Checkbox("grapes", null, true);
banana = new Checkbox("banana");

add(apple);
add(mango);
add(grapes);
add(banana);

apple.addItemListener(this);
mango.addItemListener(this);
grapes.addItemListener(this);
```

```
banana.addItemListener(this);
}
@Override
public void itemStateChanged(ItemEvent e) {
repaint();
}

public void paint(Graphics g){
String ap = "apple: ";
ap += apple.getState();
g.drawString(ap, 10, 100);

String ma = "mango: ";
ma += mango.getState();
g.drawString(ma, 10, 115);

String gr = "grapes: ";
gr += grapes.getState();
g.drawString(gr, 10, 130);

String ba = "banana: ";
ba += banana.getState();
g.drawString(ba, 10, 145);
}
```

}

Output:



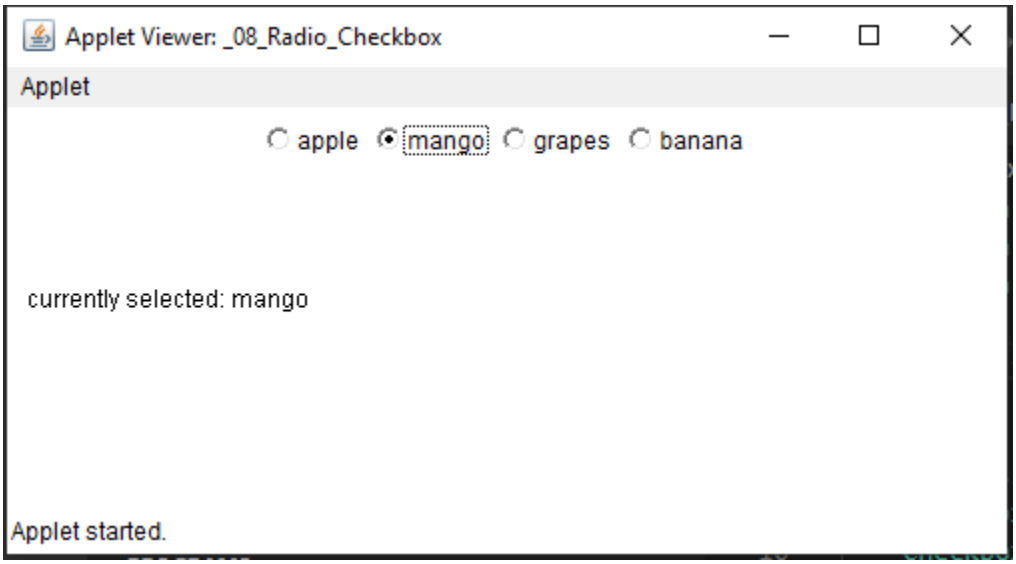
8. Program for AWT RadioButton & ItemEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

// <applet code="_08_Radio_Checkbox" height="200"width="500" > </applet>

public class _08_Radio_Checkbox extends Applet implements ItemListener{
Checkbox apple, mango, grapes, banana;
CheckboxGroup group;
public void init(){
```

```
group = new CheckboxGroup();

apple = new Checkbox("apple", group, false);
mango = new Checkbox("mango", group, false);
grapes = new Checkbox("grapes", group, true);
banana = new Checkbox("banana", group, false);

add(apple);
add(mango);
add(grapes);
add(banana);

apple.addItemListener(this);
mango.addItemListener(this);
grapes.addItemListener(this);
banana.addItemListener(this);
}
@Override
public void itemStateChanged(ItemEvent e) {
repaint();
}

public void paint(Graphics g){
```

```
String str = "currently selected: ";
str += group.getSelectedCheckbox().getLabel();

g.drawString(str, 10, 100);
}
}
```

Output:



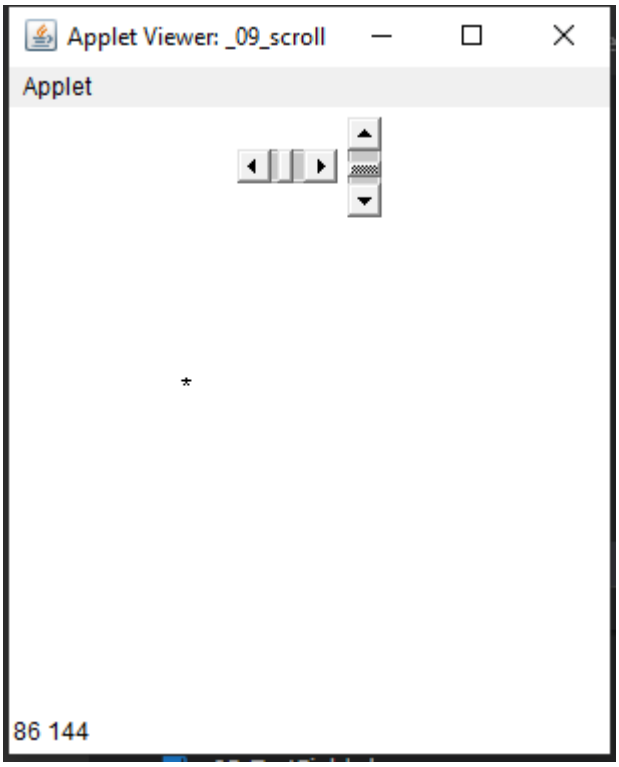
9. Program for AWT Scroll & AdjustmentEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

// <applet code="_09_scroll" height="300"width="300" > </applet>
```

```
public class _09_scroll extends Applet implements AdjustmentListener {
Scrollbar hsb, vsb;
int x = 0, y = 0;
public void init(){
vsb = new Scrollbar(Scrollbar.VERTICAL,0, 70, 0, 300);
hsb = new Scrollbar(Scrollbar.HORIZONTAL,0, 70, 0, 300);

vsb.setBounds(300, 300, 500, 300);
hsb.setBounds(300, 300, 500, 300);

add(hsb);
add(vsb);

vsb.addAdjustmentListener(this);
hsb.addAdjustmentListener(this);
}
@Override
public void adjustmentValueChanged(AdjustmentEvent e) {
// if( e.get)
// x = e.getValue();
repaint();
}
```

```
public void paint(Graphics g){
x = hsb.getValue();
y = vsb.getValue();
showStatus(x + " " + y);

g.drawString("*", x, y);
}
}
```

Output:

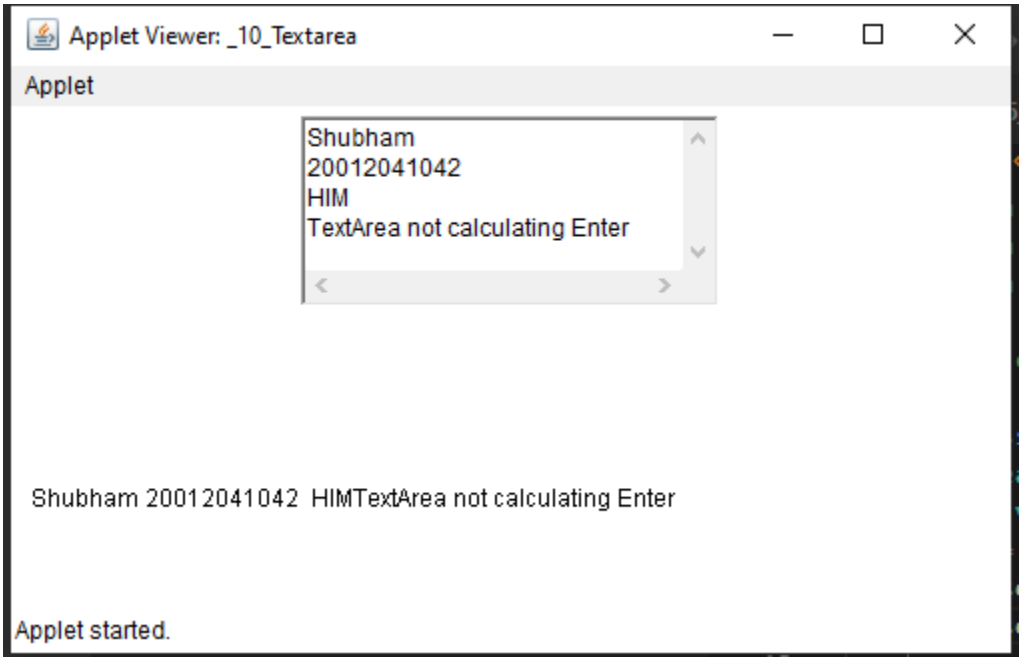


10. Program for AWT TextArea & TextEvent.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;;

// <applet code="_10_Textarea" height="250"width="500" > </applet>

public class _10_Textarea extends Applet implements TextListener{
TextArea a;
public void init(){
a = new TextArea("Hello start with this");
a.setColumns(27);
a.setRows(5);

add(a);
a.addTextListener(this);

}

@Override
public void textValueChanged(TextEvent e) {
repaint();
}
```

```
public void paint(Graphics g){
String s = a.getText();
g.drawString(s, 10, 200);
}

}
```

Output:



Swings Program

11. Program for Swing JLabel

```
import javax.swing.*;
class _11_JLabel{
JFrame jframe;

_11_JLabel(){
jframe = new JFrame("My First Frame in Swing - Shubham");

jframe.setSize(300, 400);
jframe.setVisible(true);

JLabel label1 = new JLabel("Shubham Dahiya - 20012041042");

jframe.add(label1);
}

public static void main(String[] args){
SwingUtilities.invokeLater(new Runnable(){
public void run(){
new _11_JLabel();
}
});
```

```
}
}
```

Output:

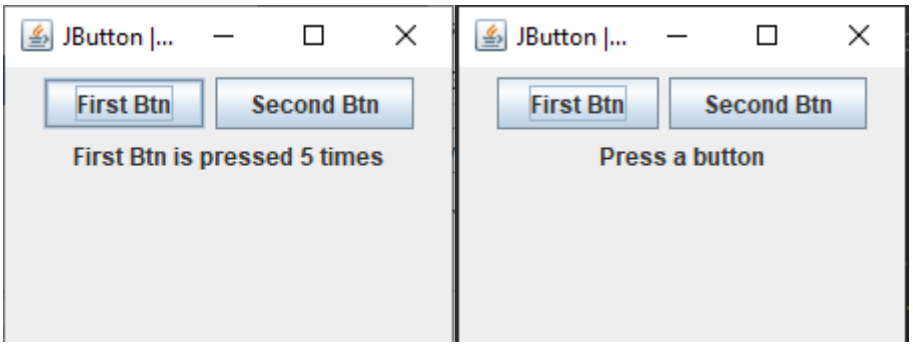
12. Program for Swing JButton

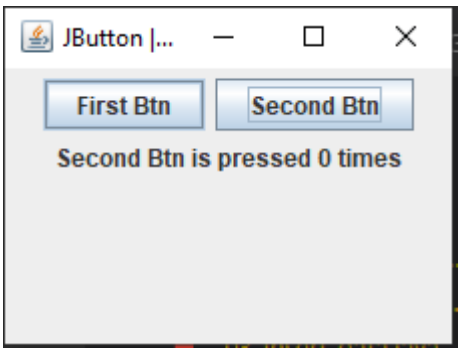
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class _12_JButton implements ActionListener {
JFrame jframe;
JButton btn1, btn2;
JLabel jLabel;
_12_JButton(){
jframe = new JFrame("JButton | Shubham");
jframe.setSize(500, 400);
```

```
jframe.setVisible(true);
jframe.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

jframe.getContentPane().setLayout(new FlowLayout());

btn1 = new JButton("First Btn");
btn2 = new JButton("Second Btn");
btn1.addActionListener(this);
btn2.addActionListener(this);

btn1.setBounds(10, 20, 100, 40);
btn2.setBounds(40, 20, 100, 40);
btn2.setSize(50, 10);

jLabel = new JLabel("Press a button");
jframe.add(btn1);
jframe.add(btn2);
jframe.add(jLabel);
}

int a = 0, b = 0;

@Override
public void actionPerformed(ActionEvent e) {
```

```
if( e.getActionCommand().equals("First Btn")){
jLabel.setText(e.getActionCommand() + " is pressed " + a++ + " times");
}
if( e.getActionCommand().equals("Second Btn")){
jLabel.setText(e.getActionCommand() + " is pressed " + b++ + " times");
}
}

public static void main(String[] args) {
SwingUtilities.invokeLater(new Runnable(){
public void run(){
new _12_JButton();
}
});
}
}
```

Output:





13. Program for Swing JCheckBox

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
public class _13_JCheckBox extends JFrame
{
public _13_JCheckBox()
{
//creating JCheckBox.
JCheckBox jcb = new JCheckBox("First");
//adding JCheckBox to frame.
add(jcb);

jcb = new JCheckBox("Second");
add(jcb);

jcb = new JCheckBox("Dahiya");
```

```
add(jcb);


setLayout(new FlowLayout());
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setSize(400, 100);
setVisible(true);
}
public static void main(String[] args)
{
new _13_JCheckBox();
}
}
```

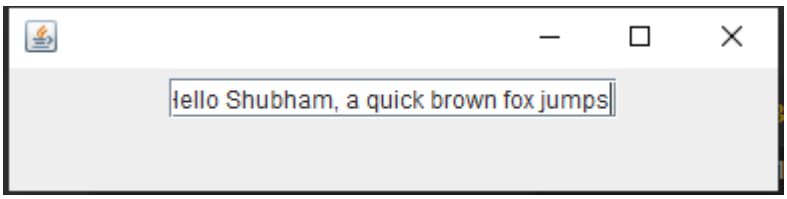
Output:

14. Program for Swing JTextField

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
```

```
public class _14_JTextField extends JFrame
{
public _14_JTextField()
{
//creating JTextField.
JTextField jtf = new JTextField(20);
//adding JTextField to frame.
add(jtf);
setLayout(new FlowLayout());
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setSize(400, 100);
setVisible(true);
}
public static void main(String[] args)
{
new _14_JTextField();
}
}
```

Output:

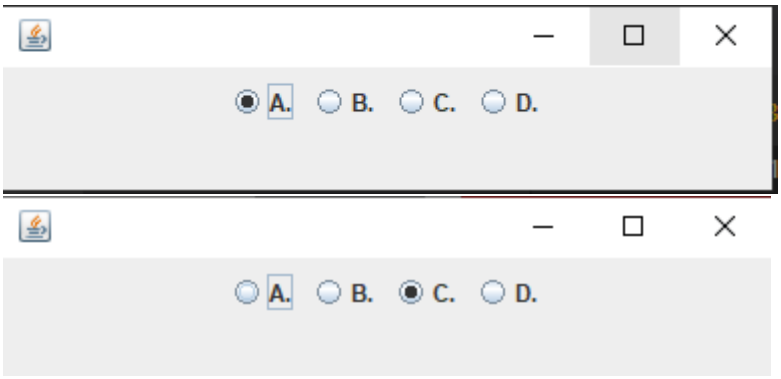


15. Program for Swing JRadioButton

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
public class _15_JRadioButton extends JFrame
{
public _15_JRadioButton()
{
//creating JRadioButton.
JRadioButton jcb = new JRadioButton("A.");
//adding JRadioButton to frame.
add(jcb);

jcb = new JRadioButton("B.");
add(jcb);

jcb = new JRadioButton("C.");
add(jcb);

jcb = new JRadioButton("D.");
add(jcb);

setLayout(new FlowLayout());
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
setSize(400, 100);
setVisible(true);
}
public static void main(String[] args)
{
new _15_JRadioButton();
}
}
```

Output:



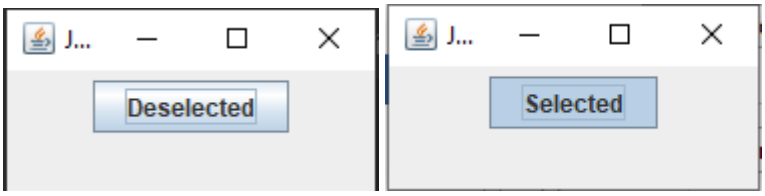
16. Program for Swing JToggleButton

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class _16_JToggleButton extends JFrame implements ItemListener{
```

```
private JToggleButton btn;
public _16_JToggleButton(){

setTitle("JToggleButton Example");
setLayout(new FlowLayout());
setJToggleButton();
setAction();
setSize(200, 100);
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

private void setJToggleButton() {
btn = new JToggleButton("Deselected");
add(btn);
}
private void setAction() {
btn.addItemListener(this);
}
public void itemStateChanged(ItemEvent eve) {
if (btn.isSelected())
btn.setText("Selected");
else
btn.setText("Deselected");
```

```
}

public static void main(String[] args) {
new _16_JToggleButton();
}
}
```

Output:



17. Program for Swing JComboBox

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class _17_JComboBox extends JFrame implements ItemListener {

// frame
static JFrame frame;
// combobox
```

```
static JComboBox combobox;
// label
static JLabel l1, l2;

public static void main(String[] args)
{
// create a new frame
frame = new JFrame("frame");

// create an object
_17_JComboBox obj = new _17_JComboBox();

// set the layout of the frame
frame.setLayout(new FlowLayout());

// array of strings containing languages
String s1[] = { "Java", "PHP", "Python", "C++", "Ruby" };

// create a checkbox
combobox = new JComboBox(s1);

// add ItemListener
combobox.addItemListener(obj);
```

```
// create labels
l1 = new JLabel("What is your favorite language? ");
l2 = new JLabel("[Java]");

// set the text color
l2.setForeground(Color.blue);

// create a new panel
JPanel p = new JPanel();

// add combobox and labels to the panel
p.add(l1);
p.add(combobox);
p.add(l2);

// add panel to frame
frame.add(p);

// set the frame size
frame.setSize(400, 200);

frame.show();
}
public void itemStateChanged(ItemEvent e)
```

```
{
// check if the state of the combobox is changed
if (e.getSource() == combobox) {

l2.setText(" ["+combobox.getSelectedItem()+"]");
}
}
}
```

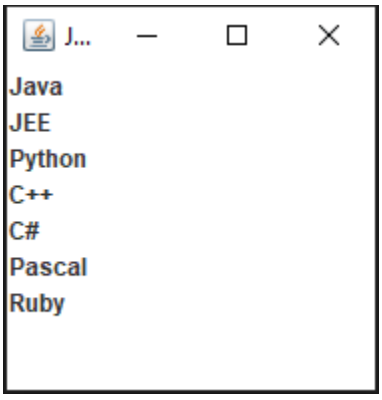
Output:

18. Program for Swing JList

```
import javax.swing.*;

public class _18_JList extends JFrame{
private JList<String> langages;

public _18_JList()
{
//create the model and add elements
DefaultListModel<String> model = new DefaultListModel<>();
model.addElement("Java");
model.addElement("JEE");
model.addElement("Python");
model.addElement("C++");
model.addElement("C#");
model.addElement("Pascal");
model.addElement("Ruby");

//create the list of languages
langages = new JList<>(model);
add(langages);

this.setTitle("JList Example");
this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
this.setSize(200,200);
this.setLocationRelativeTo(null);
this.setVisible(true);
}

public static void main(String[] args)
{
SwingUtilities.invokeLater(new Runnable()
{
@Override
public void run()
{
new _18_JList();
}
});
}
}
```

Output:



Collection Framework

19. Program for ArrayList

```
import java.util.ArrayList;
public class _19_ArrayList {
public static void main(String args[]) {
ArrayList<Integer> arrlist1 = new ArrayList<Integer>(5);

//add(E e)
arrlist1.add(12);
arrlist1.add(20);
arrlist1.add(45);
System.out.println("Printing list1:"+arrlist1);
ArrayList<Integer> arrlist2 = new ArrayList<Integer>(5);
arrlist2.add(25);
arrlist2.add(30);
arrlist2.add(31);
System.out.println("\nPrinting list2:"+arrlist2);

//addAll(int index, Collection<? extends E> c)
arrlist1.addAll(arrlist2);
System.out.println("\nPrinting all the elements"+arrlist1);

//contains(Object o)
```

```
System.out.print("\nIs 30 present in the arraylist: ");
System.out.println(arrlist1.contains(30));

// indexOf(Object o)
int pos =arrlist1.indexOf(45);
System.out.println("\nThe element 45 is at index : " + pos);

//isEmpty()
System.out.println("\nIs ArrayList Empty: "+arrlist1.isEmpty());
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> javac _19_ArrayList.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> java _19_ArrayList
Printing list1:[12, 20, 45]

Printing list2:[25, 30, 31]

Printing all the elements[12, 20, 45, 25, 30, 31]

Is 30 present in the arraylist: true

The element 45 is at index : 2

Is ArrayList Empty: false
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection>
```

20. Program for LinkedList

```
import java.util.*;

public class _20_LinkedList{
public static void main(String args[]){
//linked list declaration syntax:
LinkedList <String> list = new LinkedList<String> ();

// addFirst, addLast, add :
list.addFirst("I");
list.addFirst("Hello!");
list.addLast("am");
list.add("a");
list.addLast("Linked list");
list.addLast("by");
list.add("using");
list.addLast("Classes");
list.addLast("Framework");

// print list and return size():
System.out.println(list);
System.out.println(list.size());

// remove() :
```

```
list.removeFirst();
list.remove(6);
list.remove(6);
list.remove(5);
list.removeLast();

System.out.println(list);
System.out.println(list.size());
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> javac _20_LinkedList.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> java _20_LinkedList
[Hello!, I, am, a, Linked list, by, using, Classes, Framework]
9
[I, am, a, Linked list]
4
```

21. Program for Stack

```
import java.util.Stack;

public class _21_Stack {

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

```
Stack<Integer> stack = new Stack();


stack.push(1);

stack.push(2);

stack.push(3);


stack.pop();


System.out.println("peek: " + stack.peek());

System.out.println("empty: " + stack.empty());

System.out.println("search 2: " + stack.search(2));

System.out.println("search 4: " + stack.search(4));


stack.pop();

stack.pop();

System.out.println("empty: " + stack.empty());


}

}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> java _21_Stack
peek: 2
empty: false
search 2: 1
search 4: -1
empty: true
```

22. Program for HashSet

```
import java.util.*;

public class _22_HashSet {
public static void main(String[] args) {
// Hashset Declaratin and creating object.
Set<String> hashSet = new HashSet<String>();

// Adding elements to it
hashSet.add("White");
hashSet.add("Pink");
hashSet.add("Blue");
hashSet.add("Green");
hashSet.add("Yellow");

// Adding duplicates
hashSet.add("White");
hashSet.add("White");
hashSet.add("White");
hashSet.add("Yellow");

// Iterating HashSet to print its values.
```

```
Iterator<String> it = hashSet.iterator();


while (it.hasNext()) {
System.out.println(it.next());
}
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> javac _22_HashSet.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\02_collection> java _22_HashSet
White
Pink
Blue
Yellow
Green
```

JDBC

23. Program for JDBC Connection

```
import java.util.*;
import java.sql.*;

class _01_connection{
public static void main(String[] args) {
try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");
System.out.println("conn successful");

} catch (Exception e) {
System.out.println( "db not connected \nError is: " + e.getMessage());
}
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> javac _01_connection.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _01_connection
conn successful
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

24. Program for JDBC create table

```
import java.util.*;
import java.sql.*;

class _02_create_table{
public static void main(String[] args) {
try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
System.out.println("conn successful");

Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");
Statement stmt = conn.createStatement();

// stmt.executeUpdate("create table HIM(SName String, SAge number, SCourse
string )");
stmt.executeUpdate("create table HIM(SName text, SAge number, SCourse text
)");
System.out.println("HIM table created");

} catch (Exception e) {
System.out.println( "Error is: " + e.getMessage());
}
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _02_create_table
conn successful
HIM table created
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

25. Program for JDBC insert data

```
import java.util.*;

import java.sql.*;


class _03_insert_data{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

try {

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

System.out.println("conn successful");


Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");

// PreparedStatement ps = conn.prepareStatement("insert into HIM (SName,
SAge, SCourse) values('Shubham', ?, 'BCA')");

// PreparedStatement ps = conn.prepareStatement("insert into HIM
values('Shubham', 20, 'BCA')");

PreparedStatement ps = conn.prepareStatement("insert into HIM values( ?, ? , ?)
");
```

```
System.out.println("Enter name:");
String name = sc.next();
System.out.println("Enter age:");
int age = sc.nextInt();
System.out.println("Enter course:");
String course = sc.next();

ps.setString(1, name);
ps.setInt(2, age);
ps.setString(3, course);

// ps.setInt(1, 20);
ps.executeUpdate();

System.out.println("Data inserted in table");
conn.close();      // -----> ********** most Important **********

} catch (Exception e) {
System.out.println( "db not connected \nError is: " + e.getMessage());
}
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _03_insert_data
conn successful
Enter name:
Shubham
Enter age:
20
Enter course:
BCA
Data inserted in table
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

26. Program for JDBC select table

```
import java.util.*;
import java.sql.*;


class _04_select_table{
public static void main(String[] args) {
try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
System.out.println("conn successful");


Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");


Statement stmt = conn.createStatement();
ResultSet rs  = stmt.executeQuery("select * from HIM");


System.out.println("records in HIM table are: ");
```

```
while(rs.next()){


System.out.print( rs.getString(1) +"\t");

System.out.print( rs.getString(2) +"\t");

System.out.println( rs.getString(3) );

}


System.out.println("--- end ---");


} catch (Exception e) {

System.out.println( "db not connected \nError is: " + e.getMessage());

}

}

}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> javac _04_select_table.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _04_select_table
conn successful
records in HIM table are:
Shubham 20.0    BCA
Vipul   18.0    MCA
Ankit   21.0    MBA
Yash    22.0    BBA
--- end ---
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

27. Program for JDBC Update Table

```
import java.sql.*;
public class _05_update_table {
public static void main(String[] args) {

try{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");

Statement stmt = conn.createStatement();
stmt.executeUpdate("update HIM set SName='Yashi',SCourse='MCA' where
SAge=22 ");
conn.close();

System.out.println("Update successful");
}
catch(Exception e){
System.out.println("exception is: " + e.getMessage());
}

}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> javac _05_update_table.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _05_update_table
Update successful
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _04_select_table
conn successful
records in HIM table are:
Shubham 20.0    BCA
Vipul   18.0    MCA
Ankit   21.0    MBA
Yashi   22.0    MCA
--- end ---
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

28. Program for JDBC Delete Table

```
import java.sql.*;


public class _06_delete_table {
public static void main(String[] args) {
try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection conn = DriverManager.getConnection("jdbc:odbc:shubham");


Statement stmt = conn.createStatement();
stmt.executeUpdate("delete from Him where Sname='Shubham'");


conn.close();
System.out.println("Delete Successful");
```

```
} catch (Exception e) {
System.out.println(e.getMessage());
}
}
}
```

Output:

```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> javac _06_delete_table.java
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _06_delete_table
Delete Successful
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

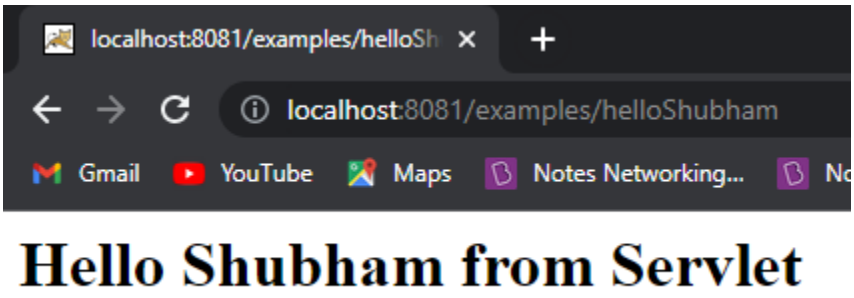
```
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc> java _04_select_table
conn successful
records in HIM table are:
Vipul   18.0    MCA
Ankit   21.0    MBA
Yashi   22.0    MCA
--- end ---
PS S:\BCA\5th Sem\Advance Java\Practical\Programs\03_jdbc>
```

Servlets

30. Program for Hello Servlet

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class _01_HelloServlet extends HttpServlet {

public void doGet(HttpServletRequest req, HttpServletResponse res) throws
ServletException, IOException{

res.setContentType("text/html");
PrintWriter out = res.getWriter();

out.println("<html>");
out.println("<body>");
out.println("<h1> Hello Shubham from Servlet");
out.println("</h1>");
out.println("</body>");
out.println("</html>");
}

}
```

Output:



31. Program for Reading Servlet Parameters

Html file

```
<html>
<body>
<center>
<form name="form1" method="post"
action="http://localhost:8081/examples/PostParametersServlet">
<table>
<tr>
 <td><B>Employee</td>
 <td><input type=textbox name="u_name" size="30" placeholder="Enter name
here..."></td>
</tr>
<tr>
 <td><B>Phone</td>
```

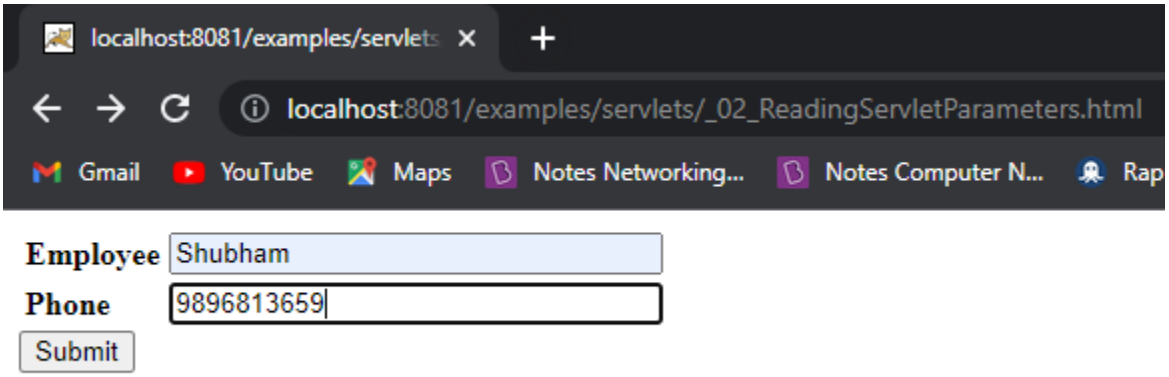
```
 <td><input type=textbox name="u_password" size="30" placeholder="Enter phone no. here..."></td>
</tr>
</table>
<input type=submit value="Submit">
</form>
</center>
</body>
</html>
```

Java file

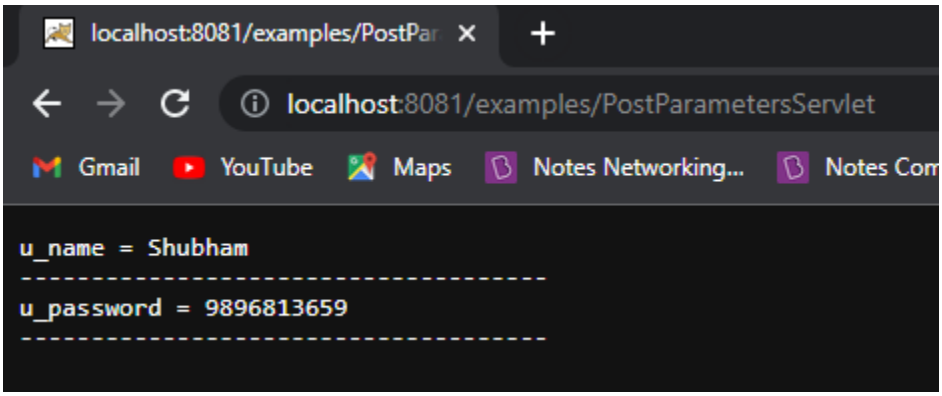
```
import java.io.*;
import javax.servlet.*;
import java.util.*;

public class _02_ReadiingServletParameters extends GenericServlet {
public void service(ServletRequest req, ServletResponse res) throws ServletException, IOException {
PrintWriter pw = res.getWriter();
Enumeration e = req.getParameterNames();
while(e.hasMoreElements()){
String name = (String) e.nextElement();
String value = (String) req.getParameter(name);

pw.println( name + " = " + value );
```

```
pw.println( "----------------------------------" );
}

pw.close();
}
}
```

Output:



onSubmit:

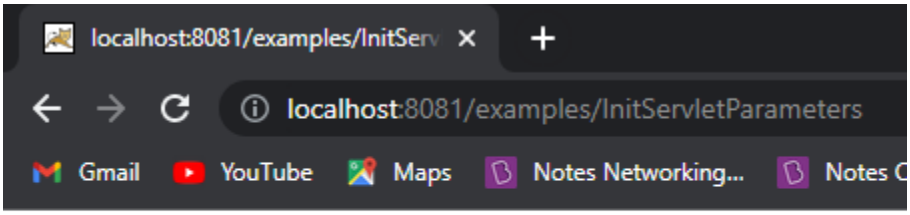


32. Program for Reading Initialization Parameter

```
import java.io.*;
import java.util.*;
import javax.servlet.*;

public class _03_ReadingInitalizationParameters extends GenericServlet{
public void service (ServletRequest req, ServletResponse res) throws
ServletException, IOException {
res.setContentType("text/html");
PrintWriter pw = res.getWriter();

ServletConfig config = getServletConfig();

String name = (String) config.getInitParameter("name");
String roll = (String) config.getInitParameter("roll_no");

// ---------- or ----------

// String name = (String) getInitParameter("name");
// String roll = (String) getInitParameter("roll_no");

pw.println("<b> Name: </b>" + name);
pw.println("<b> Roll No. </b>: " + roll);

pw.close();
}
```

}

Output:



Name: Shubham Dahiya **Roll No.** : 20012041042

33. Program for Handling Http Get Request

Html file

```
<html>
<body>
<center>
<form name="form"
 action="http://localhost:8081/examples/HttpPostRequest">
<B>State:</B>
<select name="state" size="1">
<option value="J & K">J & K</option>
<option value="Delhi">Delhi</option>
<option value="Punjab">Punjab</option>
<option value="Haryana">Haryana</option>
```

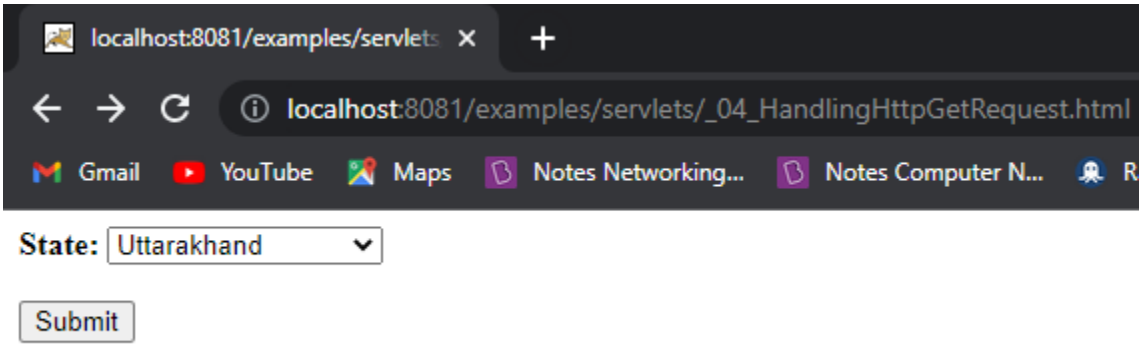
```
<option value="Gujarat">Gujarat</option>
<option value="Bihar">Bihar</option>
<option value="UttarPradesh">UttarPradesh</option>
<option value="Uttarakhand">Uttarakhand</option>
<option value="Hyderabad">Hyderabad</option>
<option value="Goa">Goa</option>
<option value="Mumbai">Mumbai</option>
<option value="Sikkim">Sikkim</option>
<option value="Arunachal Pradesh">Arunachal Pradesh</option>
</select>
<br><br>
<input type=submit value="Submit">
</form>
</body>
</html>
```

Java file

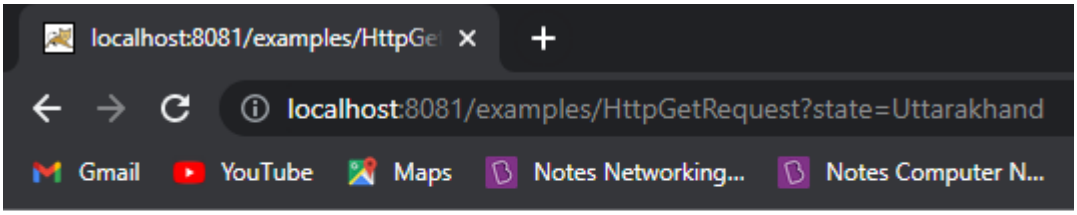
```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class _04_HandlingHttpGetRequest extends HttpServlet{
public void doGet(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException{
```

```
PrintWriter pw = res.getWriter();

res.setContentType("text/html");

String state = req.getParameter("state");
pw.println("<h1> The selected state is: </h1>");
pw.println( "<b><u>" + state + "</u></b>");

pw.close()
}
}
```

Output:



onSubmit:



The selected state is:

Uttarakhand

34. Program for Handling Http Post Request

Html file

```
<html>
<body>
<center>
<form name="form"
 method="post"
 action="http://localhost:8081/examples/HttpPostRequest">
<B>State:</B>
<select name="state" size="1">
<option value="J & K">J & K</option>
<option value="Delhi">Delhi</option>
<option value="Punjab">Punjab</option>
<option value="Haryana">Haryana</option>
<option value="Gujarat">Gujarat</option>
<option value="Bihar">Bihar</option>
```

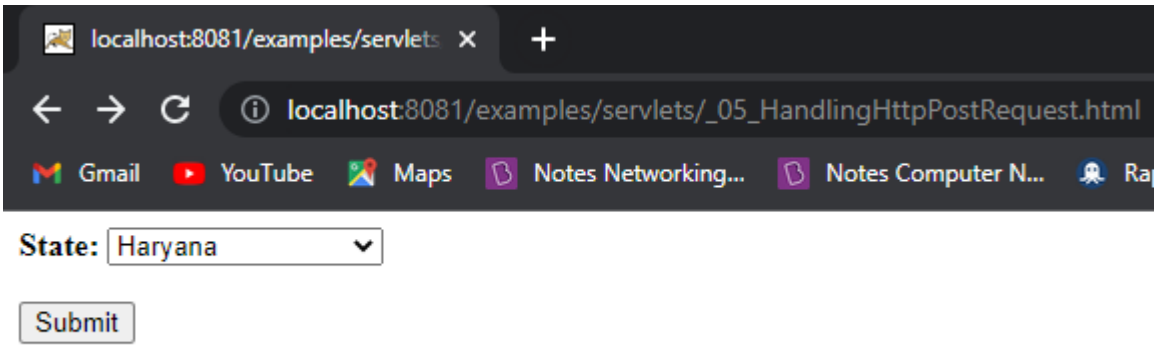
```
<option value="UttarPradesh">UttarPradesh</option>
<option value="Uttarakhand">Uttarakhand</option>
<option value="Hyderabad">Hyderabad</option>
<option value="Goa">Goa</option>
<option value="Mumbai">Mumbai</option>
<option value="Sikkim">Sikkim</option>
<option value="Arunachal Pradesh">Arunachal Pradesh</option>
</select>
<br><br>
<input type=submit value="Submit">
</form>
</body>
</html>
```

Java file

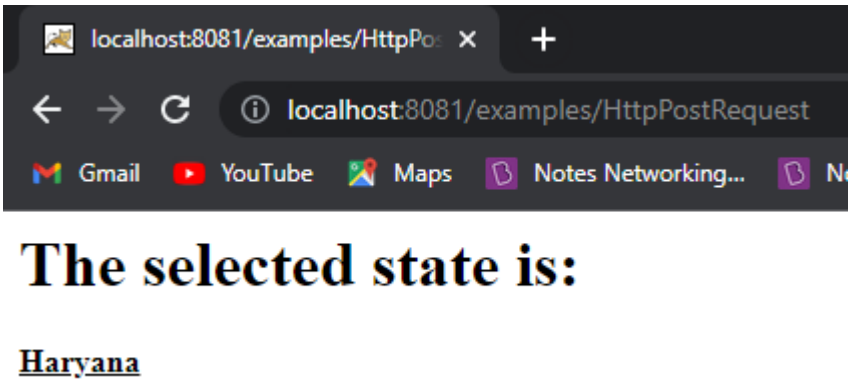
```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class _05_HandlingHttpPostRequest extends HttpServlet{
public void doPost(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException{
PrintWriter pw = res.getWriter();
```

```
res.setContentType("text/html");

String state = req.getParameter("state");
pw.println("<h1> The selected state is: </h1>");
pw.println( "<b><u>" + state + "</u></b>");

pw.close();
}
}
```

Output:



onSubmit:

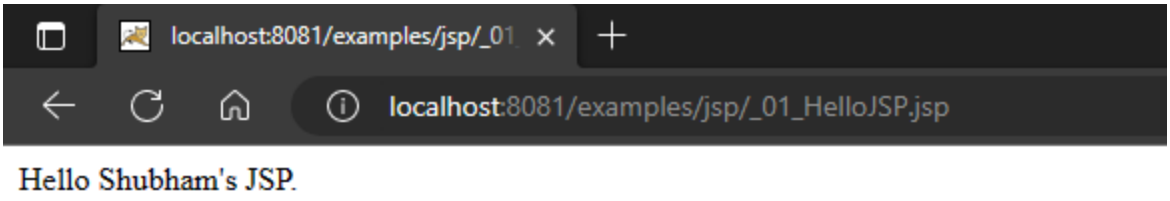


JSP

35. Program for Hello JSP

```
<html>
<body>
<% out.println("Hello Shubham's JSP."); %>
</body>
</html>
```

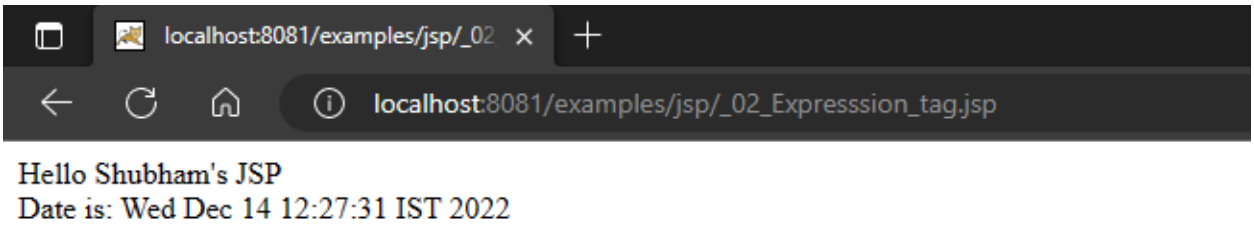
Output:



36. Program for JSP Expression tag

```
<html>
<body>
<%="Hello Shubham's JSP\n" %>
<br>
Date is: <%=java.util.Calendar.getInstance().getTime() %>
</body>
</html>
```

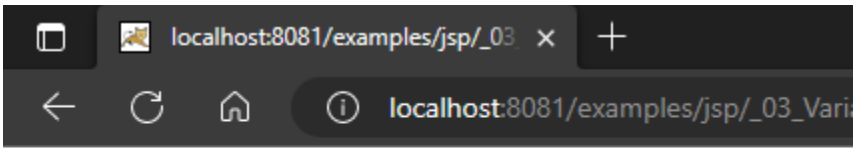
Output:



37. Program for Variable Declaration in JSP

```
<html>
<body>
<%!
int num = 10;
String s = "Shubham Dahiya, HIM, 20012041042";
%>

num is: <%= num %>
<br>

String s is: <%= s %>
</body>
</html>
```

Output:

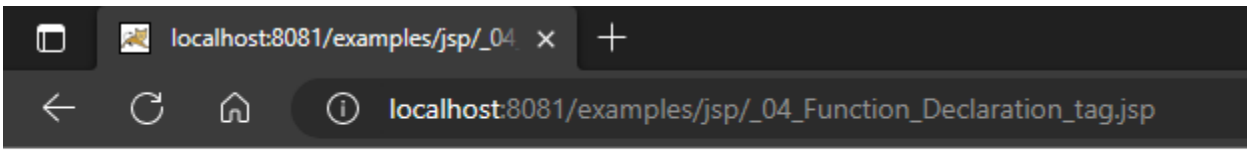


num is: 10
String s is: Shubham Dahiya, HIM, 20012041042

38. Program for Function Declaration in JSP

```
<html>
<body>
<%!
int cube(int n){
   return n*n*n;
}
%>
<br>
Cube of 6 is <%= cube(6) %>
</body>
</html>
```

Output:



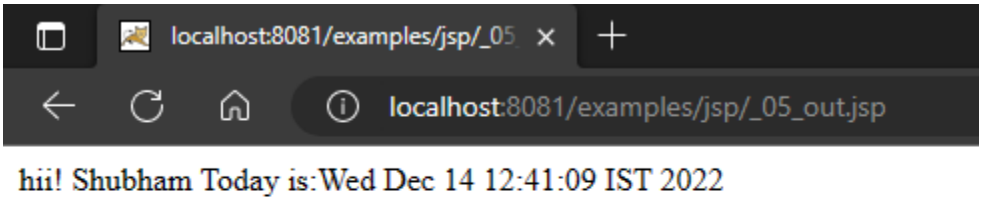
Cube of 6 is 216

Implicit Objects in JSP

39. Program for out Implicit Object in JSP

```
<html>
<body>
<% out.println("hii! Shubham"); %>
<% out.print("Today is:"+java.util.Calendar.getInstance().getTime()); %>
</body>
</html>
```

Output:



40. Program for request implicit object in JSP

_06_index.html

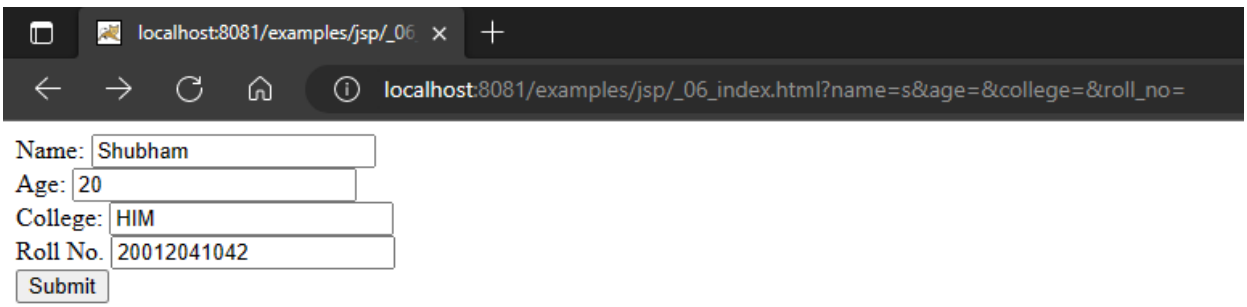
```
<html>
<body>
<form action="_06_request.jsp">
Name: <input type="text" name="name">
<br>
```

```
Age: <input type="number" name="age">
<br>
College: <input type="text" name="college">
<br>
Roll No. <input type="text" name="roll_no">
<br>
<input type="submit">
</form>
</body>
</html>
```

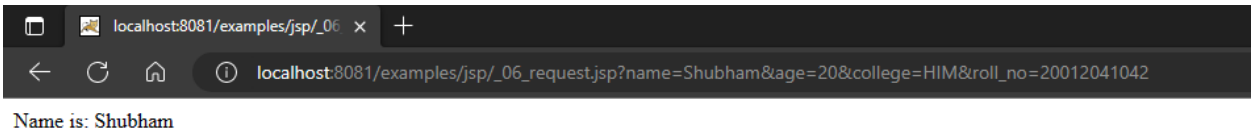
_06_request.jsp

```
<html>
<body>
Name is: <%= request.getParameter("name") %>
</body>
</html>
```

Output:



onSubmit:



41. Program for response Implicit Object in JSP

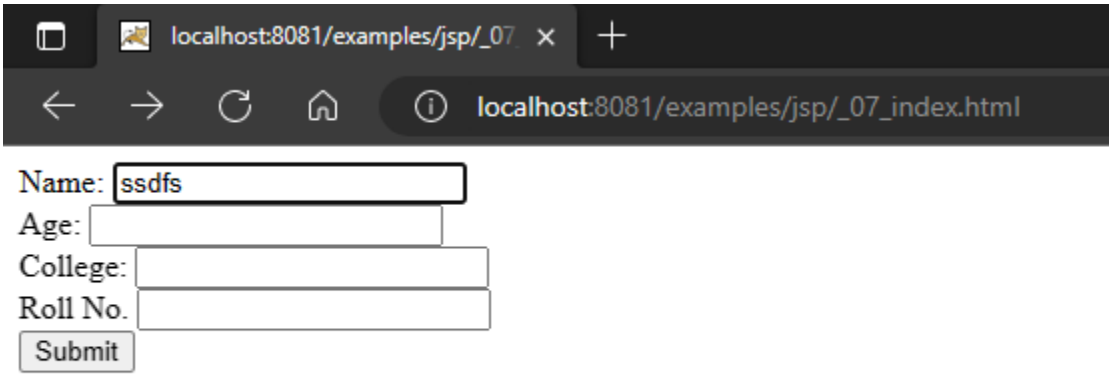
_07_index.html

```
<html>
<body>
<form action="_07_response.jsp">
Name: <input type="text" name="name">
<br>
Age: <input type="number" name="age">
<br>
College: <input type="text" name="college">
<br>
Roll No. <input type="text" name="roll_no">
<br>
<input type="submit">
</form>
</body>
</html>
```

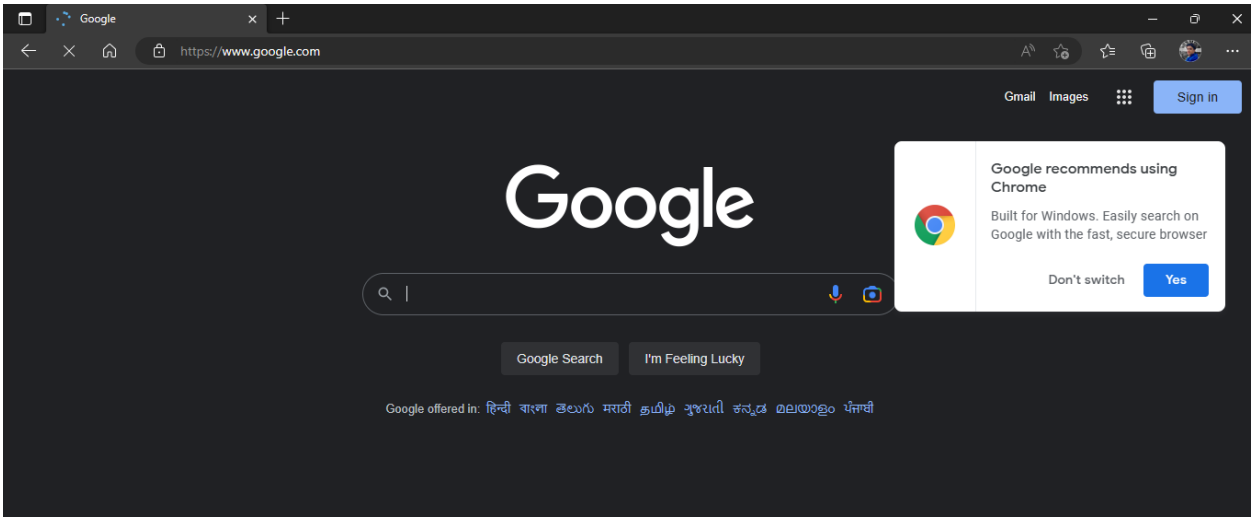
_07_response.jsp

```
<%
response.sendRedirect("http://www.google.com");
 %>
```

Output:



onSubmit



42. Program for config Implicit Object in JSP

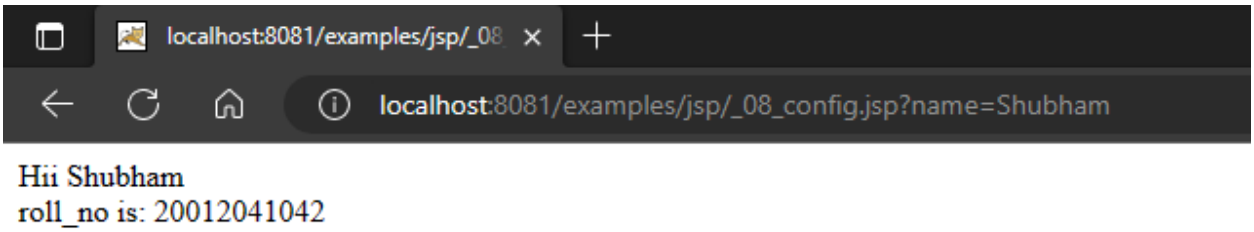
_08_index.html

```
<html>
<body>
<form action="_08_config.jsp">
Name: <input type="text" name="name">
<br>
<input type="submit">
</form>
</body>
</html>
```

_08_config.jsp

```
<%
out.println("Hii " + request.getParameter("name"));
%>

<br>

<%
String roll_no = config.getInitParameter("roll_no");
out.println("roll_no is: "+ roll_no);
%>
```

Output:


localhost:8081/examples/jsp/_08
localhost:8081/examples/jsp/_08_config.jsp?name=Shubham
Hii Shubham
roll_no is: 20012041042