

## SEND MAIL

### PROGRAM:

```
import java.util.*;
import javax.mail.*;
import javax.mail.internet.*;

public class SendMail {
    public static void main(String[] args) {
        //Get the Properties and Create a default session
        Properties prop = System.getProperties();
        prop.setProperty("mail.server.com", "127.0.0.1");
        Session session = Session.getDefaultInstance(prop);

        try {
            //Set the mail headers
            MimeMessage msg = new MimeMessage(session);
            msg.setFrom(new InternetAddress("karthik1@mailserver.com"));
            msg.addRecipient(Message.RecipientType.TO, new
InternetAddress("karthik2@mailserver.com"));
            msg.setSubject("First Mail");

            //Create the mime body and attachments
            MimeBodyPart msgBody = new MimeBodyPart();
            msgBody.setContent("Hello World", "text/html");

            MimeBodyPart attFile = new MimeBodyPart();
            attFile.attachFile("RecvMail.java");

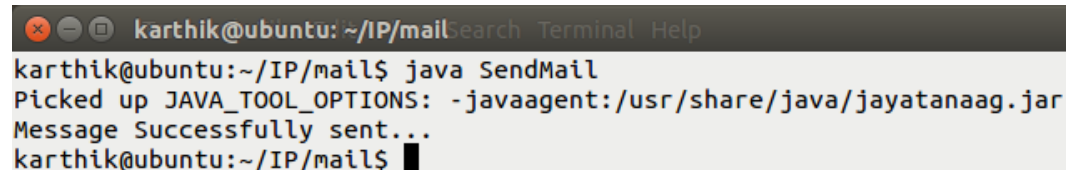
            Multipart partMsg = new MimeMultipart();
            partMsg.addBodyPart(msgBody);
            partMsg.addBodyPart(attFile);

            msg.setContent(partMsg);

            Transport.send(msg);

            System.out.println("Message Successfully sent...");
        } catch (Exception e) { e.printStackTrace(); }
    }
}
```

### OUTPUT:

A terminal window titled 'karthik@ubuntu: ~/IP/mail' with a search bar and 'Terminal Help' text. The command 'java SendMail' has been executed. The output shows 'Picked up JAVA\_TOOL\_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar' followed by 'Message Successfully sent...' on a new line. The prompt 'karthik@ubuntu:~/IP/mail\$' is visible at the bottom.

```
karthik@ubuntu:~/IP/mail$ java SendMail
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
Message Successfully sent...
karthik@ubuntu:~/IP/mail$
```

## RECEIVE MAIL

### PROGRAM:

```
import java.util.*;
import javax.mail.*;

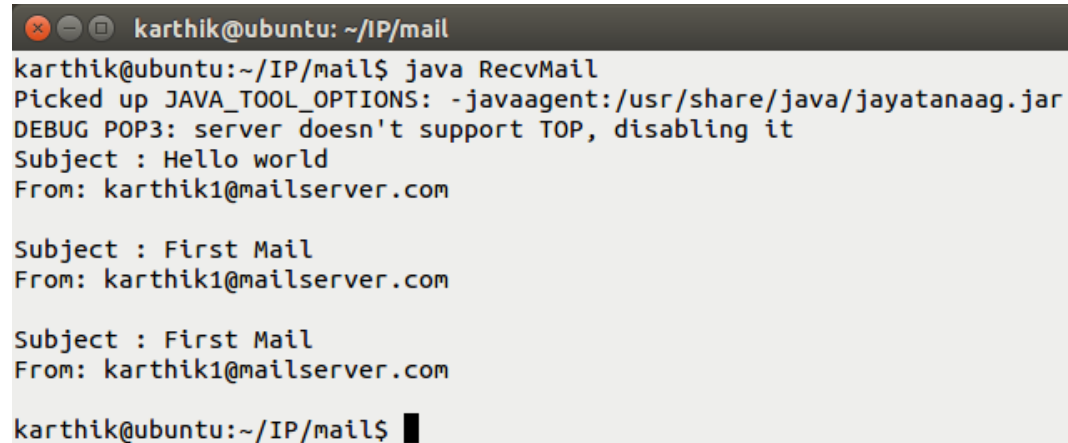
public class RecvMail {
    public static void main(String[] args) throws Exception{
        Properties prop = System.getProperties();
        prop.put("mail.pop3.host", "127.0.0.1");

        Session session = Session.getDefaultInstance(prop);
        Store store = session.getStore("pop3");
        store.connect("karthik2@mailserver.com", "root");

        Folder inbox = store.getFolder("INBOX");
        inbox.open(Folder.READ_ONLY);
        for(Message msg: inbox.getMessages()) {
            System.out.println("Subject : " + msg.getSubject());
            System.out.println("From: " + msg.getFrom()[0]);
            System.out.println();
        }
        inbox.close(false);

        store.close();
    }
}
```

### OUTPUT:



```
karthik@ubuntu: ~/IP/mail
karthik@ubuntu:~/IP/mail$ java RecvMail
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
DEBUG POP3: server doesn't support TOP, disabling it
Subject : Hello world
From: karthik1@mailserver.com

Subject : First Mail
From: karthik1@mailserver.com

Subject : First Mail
From: karthik1@mailserver.com
karthik@ubuntu:~/IP/mail$
```

### RESULT:

THE ABOVE PROGRAM SUCCESSFULLY SENT AND RECEIVED MAILS.

**FORM SERVLET COMMUNICATION:****PROGRAM:****FORM SERVLET.JAVA**

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

public class FormServlet extends HttpServlet {
    public String setRow(String fName, String fValue) {
        String row =
            "<tr>" +
                "<td>" + "<b>" + fName + "</b>" + "</td>" +
                "<td>" + fValue + "</td>" +
            "</tr>";
        return row;
    }

    @Override
    public void doGet(HttpServletRequest req, HttpServletResponse resp) {
        String[] fieldNames = {
            "pName",
            "pTitle",
            "pJob",
            "pContact",
            "pBday",
            "pMobile",
            "pEmail",
            "pMarried"
        };

        String responseText = "<html><body><table>";
        for(int i=0; i<fieldNames.length; i++) {
            responseText += setRow(fieldNames[i], req.getParameter(fieldNames[i]));
        }
        responseText += "</body></html></table>";

        try {
            resp.setContentType("text/html");
            PrintWriter out = resp.getWriter();
            out.println(responseText);
        } catch (Exception e) {e.printStackTrace();}
    }
}

/*http://localhost:8080/FormServlet/?
pName=jbhjbhbj
&pTitle=hbjbhjb
&pJob=bhjb
&pContact=9487907066
&pBday=ds&pID=98
&pMobile=9487907066
&pEmail=fhjdkfh%40jkh.com
&pMarried=Married#*/
```

**WEB.XML**

```

<web-app>
  <servlet>
    <servlet-name>FormServlet</servlet-name>
    <servlet-class>FormServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>FormServlet</servlet-name>
    <url-pattern>/FormServlet</url-pattern>
  </servlet-mapping>
  <welcome-file-list>/index.html</welcome-file-list>
</web-app>

```

**INDEX.HTML**

```

<!DOCTYPE html>
<html>
<head>
  <title>Form | Form Servlet Communication</title>
</head>
<body>
  <form name="pForm" action="FormServlet" method="get" onsubmit="return
validateForm()">
    <fieldset>
      <legend>Personal Information</legend>
      <table>
        <tr>
          <td>Name
          <td><input type="text" name="pName" title="Enter your name">
        </tr>
        <tr>
          <td>Title
          <td><input type="text" name="pTitle" title="Enter your title">
        </tr>
        <tr>
          <td>Job
          <td><input type="text" name="pJob" title="Enter your job">
        </tr>
        <tr>
          <td>Contact
          <td><input type="tel" name="pContact" title="Enter your contact's
phone number">
        </tr>
        <tr>
          <td>Birthday
          <td><input type="date" name="pBday" title="Enter your date of
birth">
        </tr>
        <tr>
          <td>ID
          <td><input type="number" name="pID" title="Enter your Person ID">
        </tr>
        <tr>
          <td>Mobile
          <td><input type="tel" name="pMobile" title="Enter your mobile
number">
        </tr>
        <tr>
          <td>Email

```

```

        <td><input type="email" name="pEmail" title="Enter your e-mail
id">
    </tr>
    <tr>
        <td>Married
        <td>
            <input type="radio" name="pMarried" value="Married" title="If
your are married">Married</input><br>
            <input type="radio" name="pMarried" value="Single" title="If
otherwise">Single</input>
        </td>
    </tr>
    <tr>
        <td> <input type="submit" value="submit" />
    </tr>
</table>
</fieldset>
</form>
</body>
</html>

<script type="text/javascript">
    function validateForm() {
        var form = document.forms["pForm"];

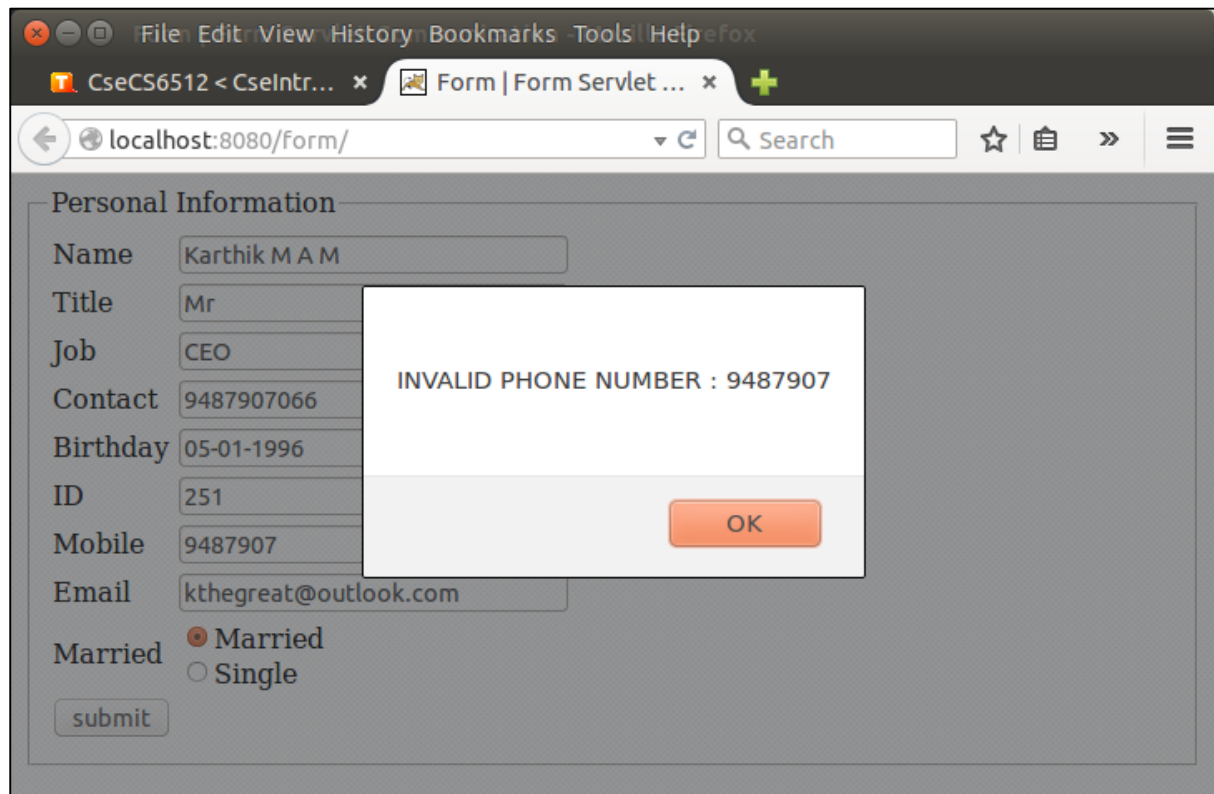
        var mobilePattern = new RegExp("[0-9]{10}");
        var emailPattern = new RegExp("[a-zA-Z0-9_]+[a-zA-Z0-9_]+[.][a-zA-Z]+");

        for(var i=1; i<form.length - 1; i++) {
            if (form[i].value == "") {
                window.alert("ALL FIELDS ARE MANDATORY");
                return false;
            }

            switch (form[i].type) {
                case "text": {
                    break;
                }
                case "tel": {
                    if (mobilePattern.test(form[i].value) == false) {
                        window.alert("INVALID PHONE NUMBER : " + form[i].value);
                        return false;
                    }
                    break;
                }
                case "email": {
                    if (emailPattern.test(form[i].value) == false) {
                        window.alert("INVALID MAIL : " + form[i].value);
                        return false;
                    }
                    break;
                }
                case "radio": {
                    var selected = false;
                    for(var j=0; j<form.length; j++) {
                        if(form[j].name == form[i].name && form[j].checked) {
                            selected = true;
                        }
                    }
                    if (!selected) {
                        window.alert("SELECT THE OPTIONS : " + form[i].name);
                    }
                }
            }
        }
    }

```

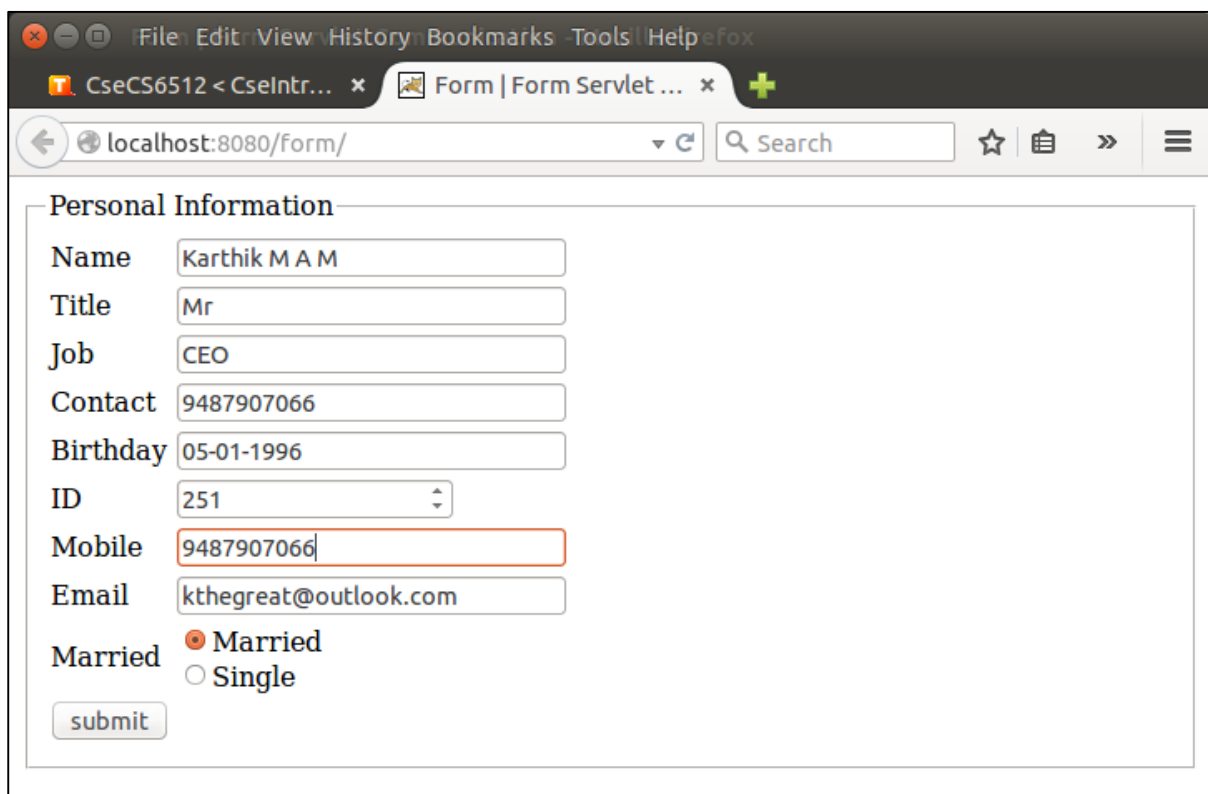
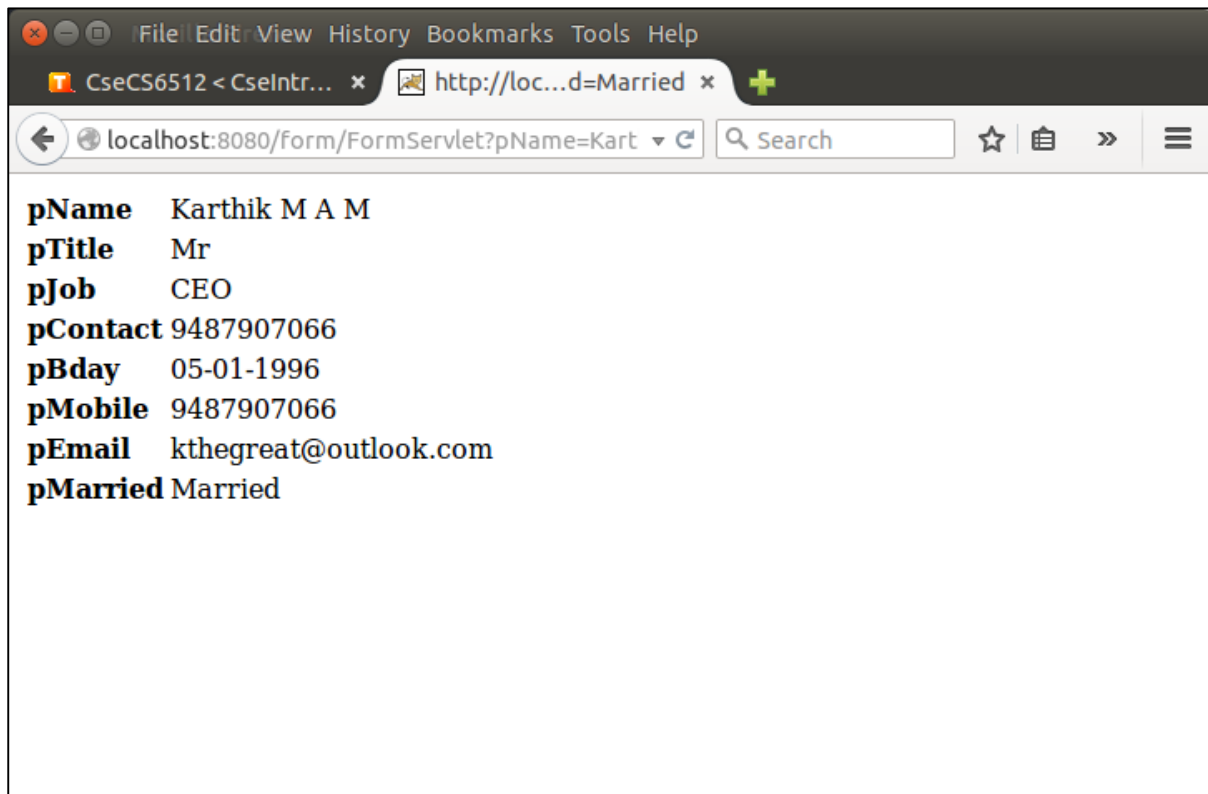
```
        return false;
    }
}
return true;
}
</script>
```

**OUTPUT:**

The screenshot shows a web browser window with the address bar displaying `localhost:8080/form/`. The page title is "Form | Form Servlet ...". The form is titled "Personal Information" and contains the following fields:

- Name: Karthik M A M
- Title: Mr
- Job: CEO
- Contact: 9487907066
- Birthday: 05-01-1996
- ID: 251
- Mobile: 9487907
- Email: kthegreat@outlook.com
- Married: ☒ Married ☐ Single

A modal dialog box is displayed over the form, showing the error message: "INVALID PHONE NUMBER : 9487907". The dialog has an "OK" button.

**RESULT:**

THE ABOVE PROGRAM SUCCESSFULLY DEMONSTRATED THE FORM SERVLET COMMUNICATION.

## STUDENT INFORMATION SYSTEM

### PROGRAM:

#### INDEX.HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Student Information System</title>
</head>
<body>
  <h4><center>Student Information System</center></h4>
  <form name="detailsForm" action="StudentSystem" method="get">
    <table>
      <tr>
        <td><button name="submit" value="Find"
onclick="buttonClick()">Search</button></td>
        <td><button name="submit" value="Add"
onclick="buttonClick()">Add</button></td>
        <td><button name="submit" value="Edit"
onclick="buttonClick()">Edit</button></td>
        <td><button name="submit" value="Delete"
onclick="buttonClick()">Delete</button></td>
      </tr>
    </table>
    <table>
      <tr>
        <td><b>ID</b></td>
        <td><input type="text" name="id" /></td>
      </tr>
      <tr>
        <td><b>Name</b></td>
        <td><input type="text" name="name" /></td>
      </tr>
      <tr>
        <td><b>Roll</b></td>
        <td><input type="text" name="roll" /></td>
      </tr>
      <tr>
        <td><b>Branch</b></td>
        <td><input type="text" name="branch" /></td>
      </tr>
      <tr>
        <td><b>Year</b></td>
        <td><input type="text" name="year" /></td>
      </tr>
      <tr>
        <td><b>Semester</b></td>
        <td><input type="text" name="semester" /></td>
      </tr>
      <tr>
        <td><b>DOB</b></td>
        <td><input type="text" name="dob" /></td>
      </tr>
      <tr>
        <td><b>Address</b></td>
        <td><input type="text" name="address" /></td>
      </tr>
    </table>
  </form>
</body>
</html>
```



```

        </table>
    </form>
</body>
</html>

<script type="text/javascript">
</script>

```

## WEB.XML

```

<web-app>
    <servlet>
        <servlet-name>StudentSystem</servlet-name>
        <servlet-class>StudentSystem</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>StudentSystem</servlet-name>
        <url-pattern>/StudentSystem</url-pattern>
    </servlet-mapping>
    <welcome-file-list>/index.html</welcome-file-list>
</web-app>

```

## STUDENT SYSTEM.JAVA

```

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

public class StudentSystem extends HttpServlet {
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/db";
    static final String USER = "root";
    static final String PASS = "2309";

    private String id;
    private String name;
    private String roll;
    private String branch;
    private String year;
    private String semester;
    private String dob;
    private String address;
    private String submit;

    public void doGet(HttpServletRequest req, HttpServletResponse res) {
        id = req.getParameter("id");
        name = req.getParameter("name");
        roll = req.getParameter("roll");
        branch = req.getParameter("branch");
        year = req.getParameter("year");
        semester = req.getParameter("semester");
        dob = req.getParameter("dob");
        address = req.getParameter("address");
        submit = req.getParameter("submit");

        try {
            Connection con = null;

```

```
PreparedStatement stmt = null;

Class.forName(JDBC_DRIVER);

con = DriverManager.getConnection(DB_URL, USER, PASS);
switch(submit) {
    case "Add": {
        stmt = con.prepareStatement("insert into student values(?, ?, ?,
?, ?, ?, ?, ?);");

        stmt.setString(1, id);
        stmt.setString(2, name);
        stmt.setString(3, roll);
        stmt.setString(4, branch);
        stmt.setString(5, year);
        stmt.setString(6, semester);
        stmt.setString(7, dob);
        stmt.setString(8, address);

        stmt.execute();
        break;
    }
    case "Edit": {
        stmt = con.prepareStatement("update student set name = ?, roll =
?, branch = ?, year = ?, semester = ?, dob = ?, address = ? where id = ?");

        stmt.setString(8, id);
        stmt.setString(1, name);
        stmt.setString(2, roll);
        stmt.setString(3, branch);
        stmt.setString(4, year);
        stmt.setString(5, semester);
        stmt.setString(6, dob);
        stmt.setString(7, address);

        stmt.execute();
        break;
    }
    case "Delete": {
        stmt = con.prepareStatement("delete from student where id=?");

        stmt.setString(1, id);

        stmt.execute();
        break;
    }
    default: {
        break;
    }
}

stmt = con.prepareStatement("select * from student where id=?");
stmt.setString(1, id);

ResultSet result = stmt.executeQuery();

if(result.next()) {
    id = result.getString(1);
    name = result.getString(2);
    roll = result.getString(3);
}
```

```

        branch = result.getString(4);
        year = result.getString(5);
        semester = result.getString(6);
        dob = result.getString(7);
        address = result.getString(8);
    }

    res.getWriter().printf(htmlDoc, id, name, roll, branch, year, semester,
dob, address, submit + " Successful!");
    } catch(Exception e) { }
}

static public String htmlDoc =
"<!DOCTYPE html>" +
" +
"
<html>
" +
"
<head>
" +
"
    <title>Student Information System</title>" +
"
</head>" +
"
<body>
" +
"
    <h4><center>Student Information System</center></h4>" +
    "<form name=\"detailsForm\" action=\"StudentSystem\" method=\"get\">"
    " +
    "<table>"
    " +
    "
    <tr>
    " +
    "
        <td><button name=\"submit\" value=\"Find\"
onclick=\"buttonClick()\">Search</button></td>" +
    "
        <td><button name=\"submit\" value=\"Add\"
onclick=\"buttonClick()\">Add</button></td>" +
    "
        <td><button name=\"submit\" value=\"Edit\"
onclick=\"buttonClick()\">Edit</button></td>" +
    "
        <td><button name=\"submit\" value=\"Delete\"
onclick=\"buttonClick()\">Delete</button></td>" +
    "
    </tr>" +
    "
    </table>" +
    "<table>"
    " +
    "
    <tr>
    " +
    "
        <td><b>ID</b></td>" +
    "
        <td><input type=\"text\" name=\"id\" value=\"%s\" /></td>" +

```

```

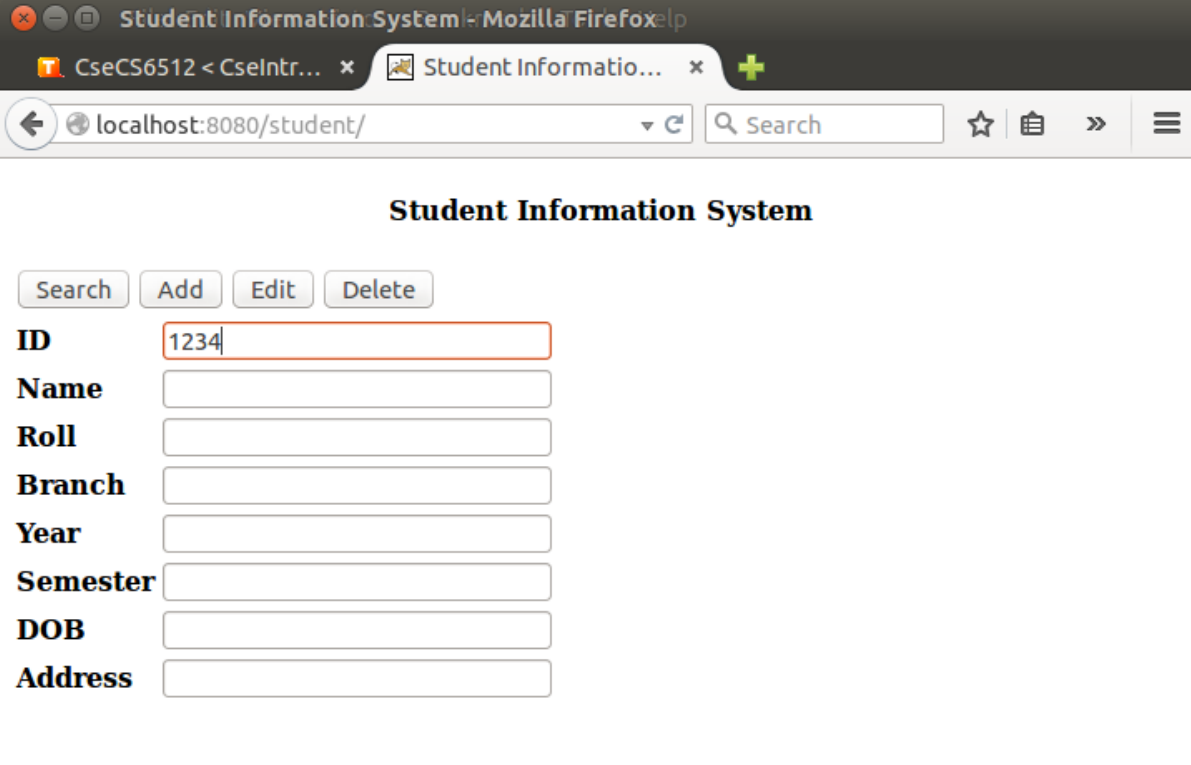
"
</tr>" +
"
<tr>
" +
"
    <td><b>Name</b></td>" +
"
    <td><input type=\"text\" name=\"name\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>Roll</b></td>" +
"
    <td><input type=\"text\" name=\"roll\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>Branch</b></td>" +
"
    <td><input type=\"text\" name=\"branch\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>Year</b></td>" +
"
    <td><input type=\"text\" name=\"year\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>Semester</b></td>" +
"
    <td><input type=\"text\" name=\"semester\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>DOB</b></td>" +
"
    <td><input type=\"text\" name=\"dob\" value=\"%s\" /></td>" +
"
</tr>" +
"
<tr>
" +
"
    <td><b>Address</b></td>" +
"

```

```

        <td><input type=\"text\" name=\"address\" value=\"%s\" /></td>\" +
        \"
    </tr>\" +
    \"
</table>\" +
    \"<p>%s</p>\" +
    \"
</form>\" +
    \"
</body>\" +
    \"
</html>\";
}

```

**OUTPUT:**


The screenshot shows a web browser window titled "Student Information System" with the URL "localhost:8080/student/". The page displays a form titled "Student Information System" with four buttons: "Search", "Add", "Edit", and "Delete". Below the buttons, there are input fields for the following fields:

Field	Value
ID	1234
Name	
Roll	
Branch	
Year	
Semester	
DOB	
Address	

File Edit View History Bookmarks Tools Help

CseCS6512 < CseIntr... Student Informatio... +

localhost:8080/student/StudentSystem?submit: Search ☆ 📁 » ☰

### Student Information System

Search Add Edit Delete

**ID** 1

**Name** Karthik

**Roll** 1234

**Branch** CSE

**Year** 3

**Semester** 5

**DOB** 05-01-1996

**Address** Erode

Add Successful!

File Edit View History Bookmarks Tools Help

CseCS6512 < CseIntr... Student Informatio... +

localhost:8080/student/StudentSystem?submit: Search ☆ 📁 » ☰

### Student Information System

Search Add Edit Delete

**ID** 1

**Name** Karthik

**Roll** 1234

**Branch** CSE

**Year** 3

**Semester** 5

**DOB** 05-01-1996

**Address** Erode

Find Successful!

File Edit View History Bookmarks Tools Help

CseCS6512 < CseIntr... x Student Informatio... x +

localhost:8080/student/StudentSystem?submit: Search ☆ » ≡

### Student Information System

Search Add Edit Delete

**ID**

**Name**

**Roll**

**Branch**

**Year**

**Semester**

**DOB**

**Address**

Edit Successful!

```

root@ubuntu: ~# Edit View Search Terminal Help
mysql> select * from student;
+-----+-----+-----+-----+-----+-----+-----+-----+
+
+ id | name | roll | branch | year | semester | dob | address
+-----+-----+-----+-----+-----+-----+-----+-----+
+
+ 2 | Karupiah | 1235 | CSE | 3 | 5 | 05-01-1996 | Chennai
+
+ 3 | Dharma | 1236 | CSE | 3 | 5 | 05-01-1996 | Chennai
+
+ 4 | Aniruth | 1237 | CSE | 3 | 5 | 05-01-1996 | Chennai
+
+ 5 | Akshay | 1237 | CSE | 3 | 5 | 05-01-1996 | Chennai
+
+ 1 | Karthik M A M | 1234 | CSE | 3 | 5 | 05-01-1996 | Erode
+
+ 1 | Karthik M A M | 1234 | CSE | 3 | 5 | 05-01-1996 | Erode
+-----+-----+-----+-----+-----+-----+-----+-----+
+
6 rows in set (0.00 sec)

mysql>

```

**Student Information System**

**ID**   
**Name**   
**Roll**   
**Branch**   
**Year**   
**Semester**   
**DOB**   
**Address**

Delete Successful!

```

root@ubuntu: ~# edit View Search Terminal Help

mysql> select * from student;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | name   | roll | branch | year | semester | dob       | address |
+----+-----+-----+-----+-----+-----+-----+-----+
| 2  | Karupiah | 1235 | CSE    | 3    | 5        | 05-01-1996 | Chennai |
| 3  | Dharma  | 1236 | CSE    | 3    | 5        | 05-01-1996 | Chennai |
| 4  | Aniruth | 1237 | CSE    | 3    | 5        | 05-01-1996 | Chennai |
| 5  | Akshay  | 1237 | CSE    | 3    | 5        | 05-01-1996 | Chennai |
+----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>

```

**RESULT:**

**THE STUDENT INFO SYSTEM WAS SUCCESSFULLY CREATED.**



## APPLET TO SERVLET COMMUNICATION

### PROGRAM:

#### APPLET CLASS.JAVA

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

public class AppletClass extends Applet implements ActionListener{
    private TextField txtUser = new TextField(10);
    private TextField txtPass = new TextField(10);
    private TextField txtResult = new TextField(10);
    private Button btnSend = new Button("Send");

    public void init() {
        add(txtUser);
        add(txtPass);
        add(btnSend);
        add(txtResult);

        btnSend.addActionListener(this);
        txtResult.setEditable(false);
    }

    public void actionPerformed(ActionEvent e) {
        try {
            URL url = new URL(getCodeBase(), "ServletClass");

            URLConnection con = url.openConnection();
            con.setDoInput(true);
            con.setDoOutput(true);
            con.setUseCaches(false);
            con.setRequestProperty("Content-Type", "application/x-java-serialized-
object");

            ObjectOutputStream out = new ObjectOutputStream(con.getOutputStream());
            out.writeObject(txtUser.getText());
            out.writeObject(txtPass.getText());
            out.flush();
            out.close();

            ObjectInputStream in = new ObjectInputStream(con.getInputStream());
            txtResult.setText((String) in.readObject());
        } catch (Exception ex) { txtResult.setText(ex.getMessage()); }
    }
}
```

#### SERVLET CLASS.JAVA

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

public class ServletClass extends HttpServlet {
    public void doPost(HttpServletRequest req, HttpServletResponse res) {
        try {
```

```
ObjectInputStream in = new ObjectInputStream(req.getInputStream());
String user = (String) in.readObject();
String pass = (String) in.readObject();

String result = "Failure";
if("karthik".equals(user) && "karthik".equals(pass)) {
    result = "Success";
}

res.setContentType("application/x-java-serialized-object");
ObjectOutputStream out = new ObjectOutputStream(res.getOutputStream());
out.writeObject(result);
out.flush();
out.close();
} catch(Exception e) {}
}
}
```

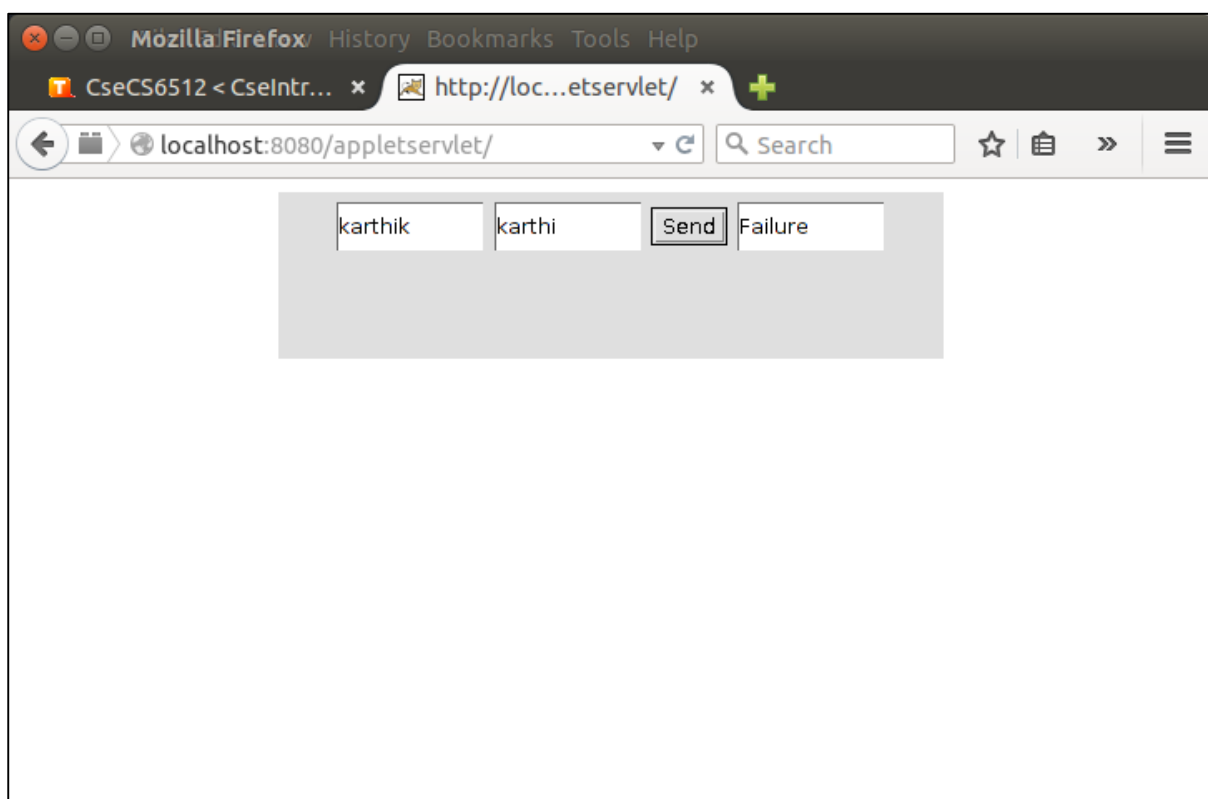
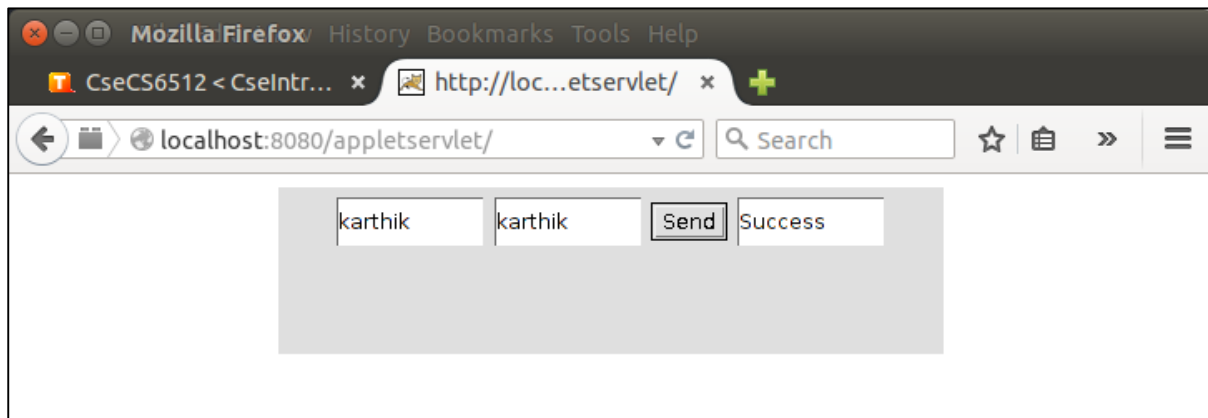
### INDEX.HTML

```
<!DOCTYPE html>
<html>
  <body>
    <center>
      <applet code="AppletClass" width="400" height="100"></applet>
    </center>
  </body>
</html>
```

### WEB.XML

```
<web-app>
  <servlet>
    <servlet-name>ServletClass</servlet-name>
    <servlet-class>ServletClass</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>ServletClass</servlet-name>
    <url-pattern>/ServletClass</url-pattern>
  </servlet-mapping>
  <welcome-file-list>/index.html</welcome-file-list>
</web-app>
```

### OUTPUT:



**RESULT:**

**THE APPLET TO SERVLET COMMUNICATION WAS ESTABLISHED.**

**SESSION VISIT COUNTER:****PROGRAM:****INDEX.JSP**

```
<%@ page import="java.io.*" %>

<!DOCTYPE html>
<html>
<head>
    <title>Session Tracking</title>
</head>
<body>
    <center>
        <%
            int visitCount = 0;
            String visitCountKey = "visitCount";

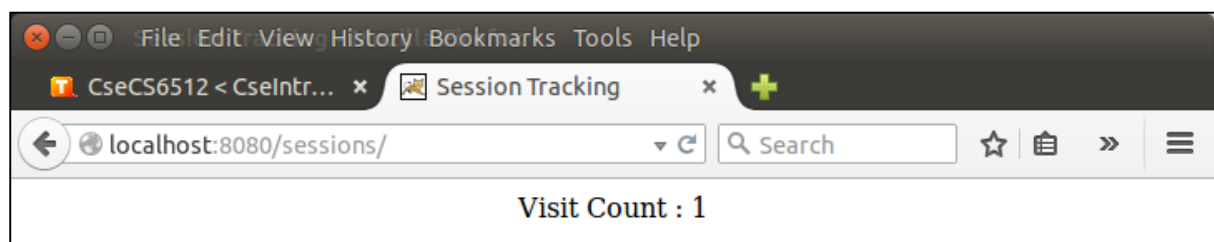
            if(session.isNew()) {
                session.setAttribute(visitCountKey, visitCount);
            }

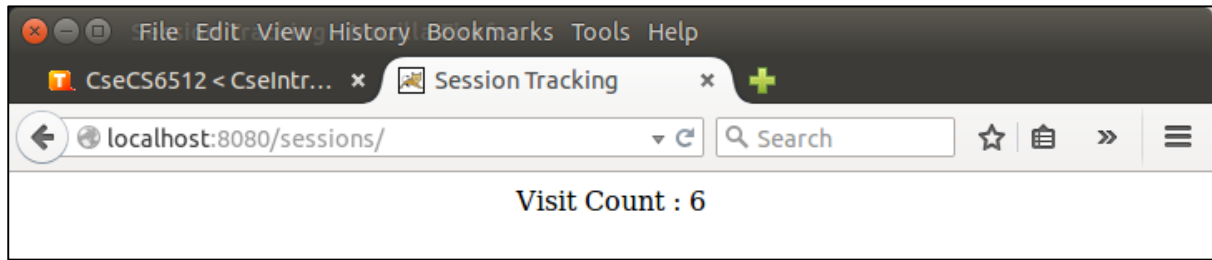
            visitCount = (int)session.getAttribute(visitCountKey) + 1;
            session.setAttribute(visitCountKey, visitCount);
            %>
        <% out.println("Visit Count : " + visitCount); %>
    </center>
</body>
</html>
```

**WEB.XML**

```
<web-app>
    <servlet>
        <servlet-name>index</servlet-name>
        <jsp-file>/index.jsp</jsp-file>
    </servlet>
    <servlet-mapping>
        <servlet-name>index</servlet-name>
        <url-pattern>/index.jsp</url-pattern>
    </servlet-mapping>

    <welcome-file-list>index.jsp</welcome-file-list>
</web-app>
```

**OUTPUT:**



**RESULT:**

**THE HIT COUNTER USING JSP SESSIONS WAS SIMULATED.**

**ONLINE EXAM USING JSP****PROGRAM:****INDEX.JSP**

```

<%@ page import="java.sql.*,java.io.*"%>

<%
    String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    String DB_URL = "jdbc:mysql://localhost/db";
    String USER = "root";
    String PASS = "2309";
%>

<!DOCTYPE html>
<html>
<head>
    <title>Questions</title>
</head>
<body>
    <h3><center>Online Test</h3></center>
    <%
        try {
            Connection con = null;
            ResultSet questions = null;
            ResultSet options = null;

            Class.forName(JDBC_DRIVER);
            con = DriverManager.getConnection(DB_URL, USER, PASS);

            questions = con.prepareStatement("select * from questions").executeQuery();
            options = con.prepareStatement("select * from options").executeQuery();

            out.println("<form action='results.jsp' method='get'>
                ");
            while(questions.next()) {
                out.println("<b>
                    " + questions.getString(1) + ". " + questions.getString(2) + " <b>
                        <br>");

                    for(int i=0; i<3; i++) {
                        options.next();
                        out.println("<input type='radio' name='" +
questions.getString(1) + "' + \"value='\" + options.getString(2) + \"'>");
                        out.println((i + 1) + ". " + options.getString(2) +
"<br>");
                        out.println("</input>");
                    }

                    out.println("<br>");
                }
                out.println("<input type='submit' value='Check Results'>");
                out.println("
            </form>");
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    %>

```

```
</body>
</html>
```

## RESULTS.JSP

```
<%@ page import="java.sql.*,java.io.*" %>

<%
    String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    String DB_URL = "jdbc:mysql://localhost/db";
    String USER = "root";
    String PASS = "2309";
%>

<!DOCTYPE html>
<html>
<head>
    <title>Results</title>
</head>
<body>
    <%
        Connection con = null;
        ResultSet answers = null;

        Class.forName(JDBC_DRIVER);
        con = DriverManager.getConnection(DB_URL, USER, PASS);

        answers = con.prepareStatement("select * from answers;").executeQuery();

        int score = 0;
        while(answers.next()) {
            String userAns = request.getParameter(answers.getString(1));
            if(userAns != null && userAns.equals(answers.getString(2))) {
                score++;
            }
        }
    %>
    <center>
        <h4><%= "Your Score is " + score %></h4>
    </center>
</body>
</html>
```

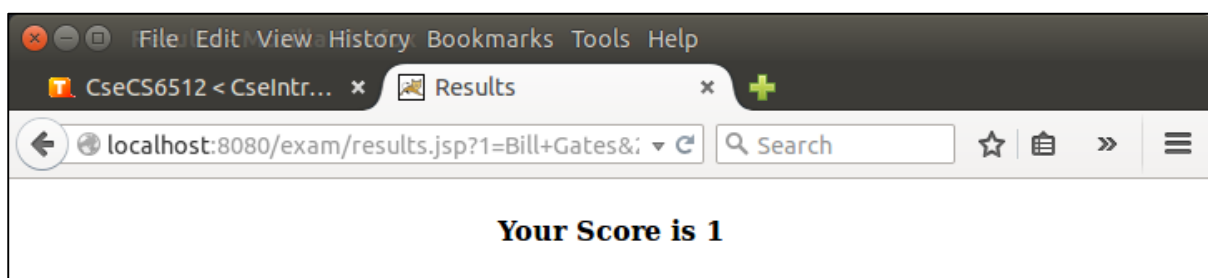
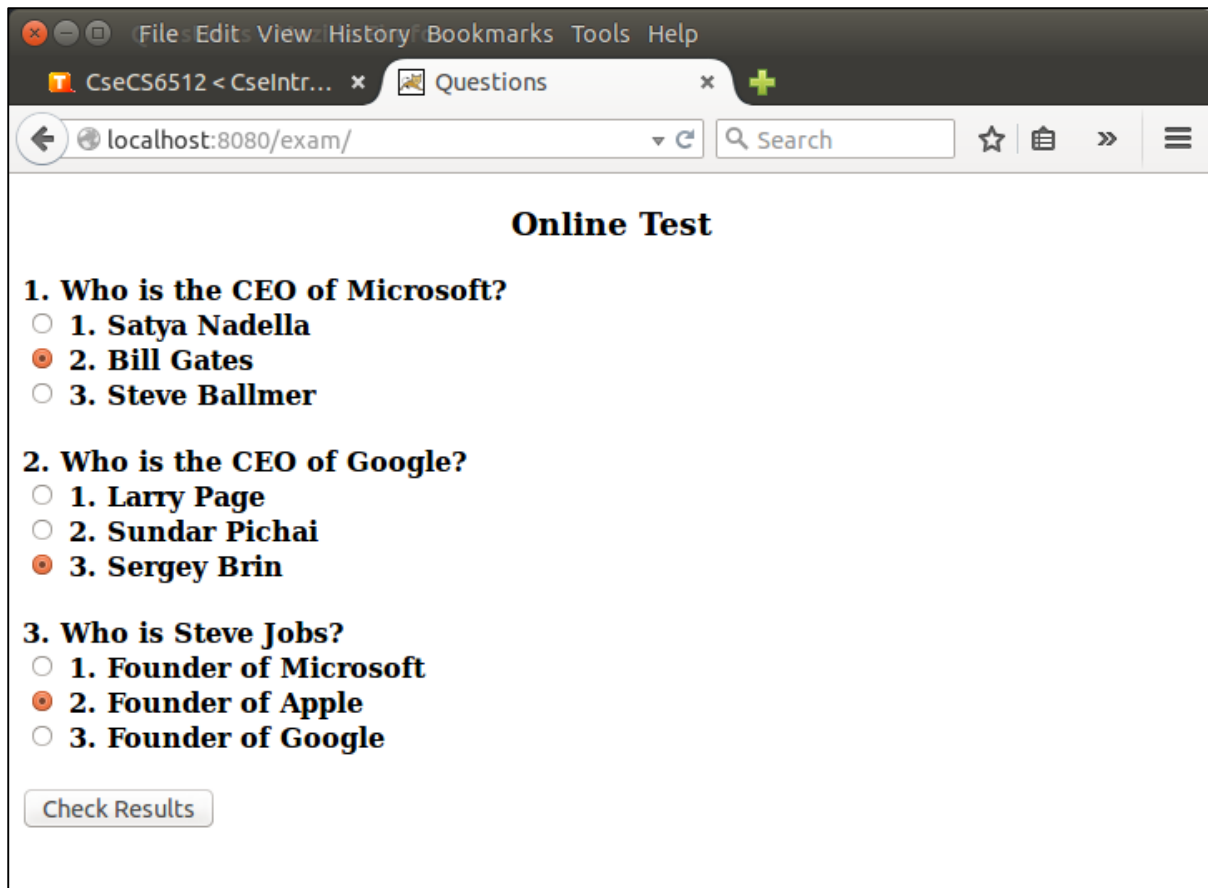
## WEB.XML

```
<web-app>
    <servlet>
        <servlet-name>Questions</servlet-name>
        <jsp-file>/index.jsp</jsp-file>
    </servlet>
    <servlet-mapping>
        <servlet-name>Questions</servlet-name>
        <url-pattern>/index.jsp</url-pattern>
    </servlet-mapping>

    <servlet>
        <servlet-name>Results</servlet-name>
        <jsp-file>/results.jsp</jsp-file>
    </servlet>
```

```
<servlet-mapping>
  <servlet-name>Results</servlet-name>
  <url-pattern>/results.jsp</url-pattern>
</servlet-mapping>

<welcome-file-list>/index.jsp</welcome-file-list>
</web-app>
```

**OUTPUT:****RESULT:**

THE ONLINE EXAM WEB-APP WAS SUCCESSFULLY CREATED.



## XML DOM

### PROGRAM:

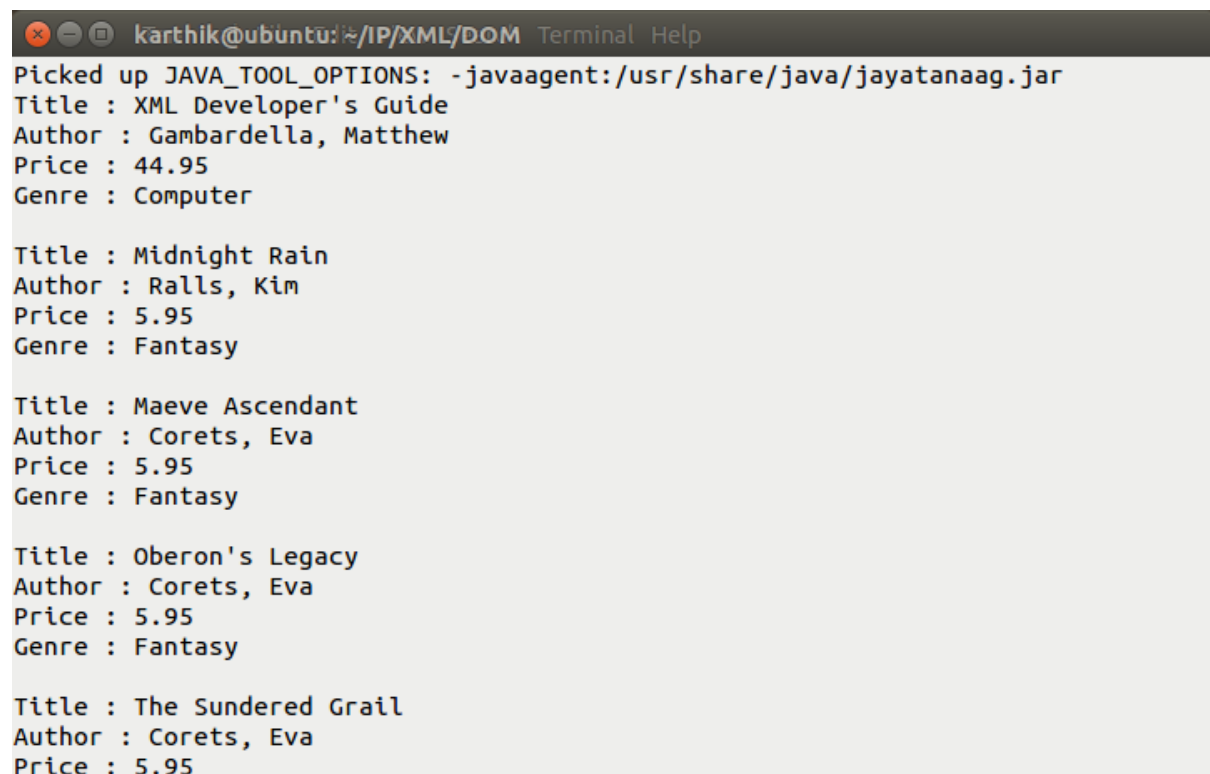
```
import java.io.*;
import javax.xml.parsers.*;
import org.w3c.dom.*;

public class XmlParser {
    public static void main(String[] args) {
        try {
            File xmlFile = new File("sample.xml");

            Document xmlDoc = DocumentBuilderFactory
                .newInstance()
                .newDocumentBuilder()
                .parse(xmlFile);
            xmlDoc.getDocumentElement().normalize();

            NodeList nodeList = xmlDoc.getElementsByTagName("book");
            for(int i=0; i<nodeList.getLength(); i++) {
                Element element = (Element)nodeList.item(i);
                System.out.println("Author : " + element
                    .getElementsByTagName("author")
                    .item(0)
                    .getTextContent());
            }
        } catch (Exception e) { e.printStackTrace(); }
    }
}
```

### OUTPUT:

A terminal window titled 'karthik@ubuntu: ~/IP/XML/DOM Terminal Help'. The output shows the results of parsing an XML file. It lists five books with their details: Title, Author, Price, and Genre. The first book is 'XML Developer's Guide' by Gambardella, Matthew, priced at 44.95, in the Computer genre. The other four books are 'Midnight Rain', 'Maeve Ascendant', 'Oberon's Legacy', and 'The Sundered Grail', all priced at 5.95 and in the Fantasy genre.

```
karthik@ubuntu: ~/IP/XML/DOM Terminal Help
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/share/java/jayatanaag.jar
Title : XML Developer's Guide
Author : Gambardella, Matthew
Price : 44.95
Genre : Computer

Title : Midnight Rain
Author : Ralls, Kim
Price : 5.95
Genre : Fantasy

Title : Maeve Ascendant
Author : Corets, Eva
Price : 5.95
Genre : Fantasy

Title : Oberon's Legacy
Author : Corets, Eva
Price : 5.95
Genre : Fantasy

Title : The Sundered Grail
Author : Corets, Eva
Price : 5.95
```

```
karthik@ubuntu:~/IP/XML/DOM Terminal Help
Title : Lover Birds
Author : Randall, Cynthia
Price : 4.95
Genre : Romance

Title : Splish Splash
Author : Thurman, Paula
Price : 4.95
Genre : Romance

Title : Creepy Crawlies
Author : Knorr, Stefan
Price : 4.95
Genre : Horror

Title : Paradox Lost
Author : Kress, Peter
Price : 6.95
Genre : Science Fiction

Title : Microsoft .NET: The Programming Bible
Author : O'Brien, Tim
Price : 36.95
Genre : Computer

Price : 4.95
Genre : Horror

Title : Paradox Lost
Author : Kress, Peter
Price : 6.95
Genre : Science Fiction

Title : Microsoft .NET: The Programming Bible
Author : O'Brien, Tim
Price : 36.95
Genre : Computer

Title : MSXML3: A Comprehensive Guide
Author : O'Brien, Tim
Price : 36.95
Genre : Computer

Title : Visual Studio 7: A Comprehensive Guide
Author : Galos, Mike
Price : 49.95
Genre : Computer

karthik@ubuntu:~/IP/XML/DOM$
```

**RESULT:**

**THE XML WAS PARSED USING DOM.**

**XML XSLT:****PROGRAM:****SELECT A GENRE**

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <table>
          <tr>
            <th>Title</th>
            <th>Genre</th>
          </tr>
          <xsl:for-each select="catalog/book[genre='Computer']">
            <xsl:sort select="title"/>
            <tr>
              <td><xsl:value-of select="title"/></td>
              <td><xsl:value-of select="genre"/></td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

**SELECT A PRICE RANGE**

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <table>
          <tr>
            <th>Title</th>
            <th>Price</th>
          </tr>
          <xsl:for-each select="catalog/book">
            <xsl:if test="not(price > 50) and (price > 10)">
              <tr>
                <td><xsl:value-of select="title"/></td>
                <td><xsl:value-of select="price"/></td>
              </tr>
            </xsl:if>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

**SORT AND DISPLAY**

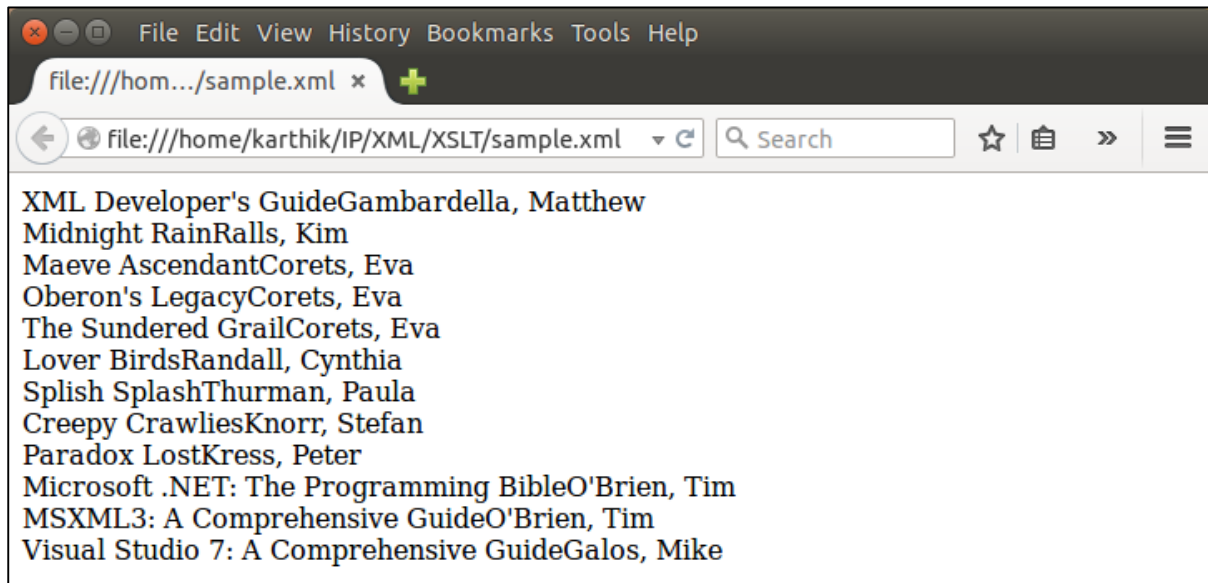
```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <table>
          <tr>
            <th>Title</th>
            <th>Author</th>
          </tr>
          <xsl:for-each select="catalog/book">
            <xsl:sort select="title"/>
            <tr>
              <td><xsl:value-of select="title"/></td>
              <td><xsl:value-of select="author"/></td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

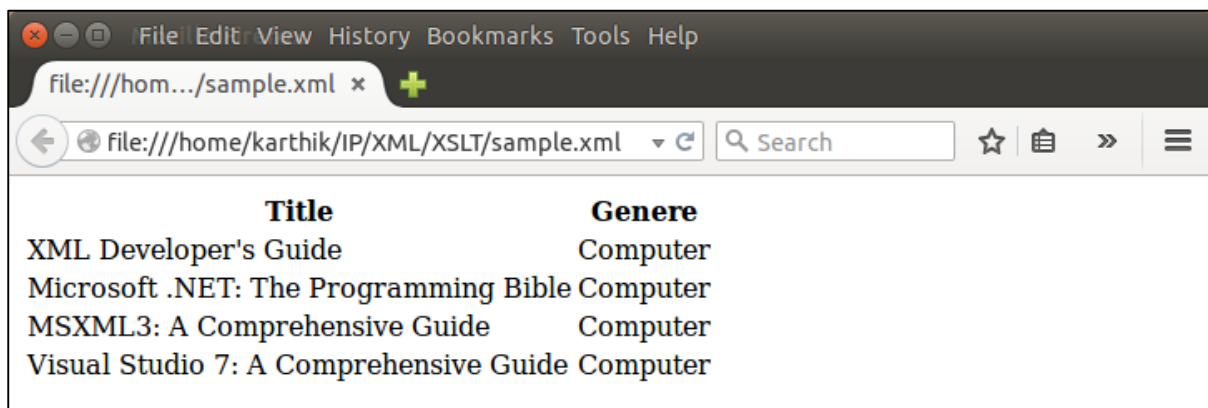
### CONCATENATION OF AUTHOR AND TITLE

```
<?xml version="1.0" encoding="UTF-8"?>

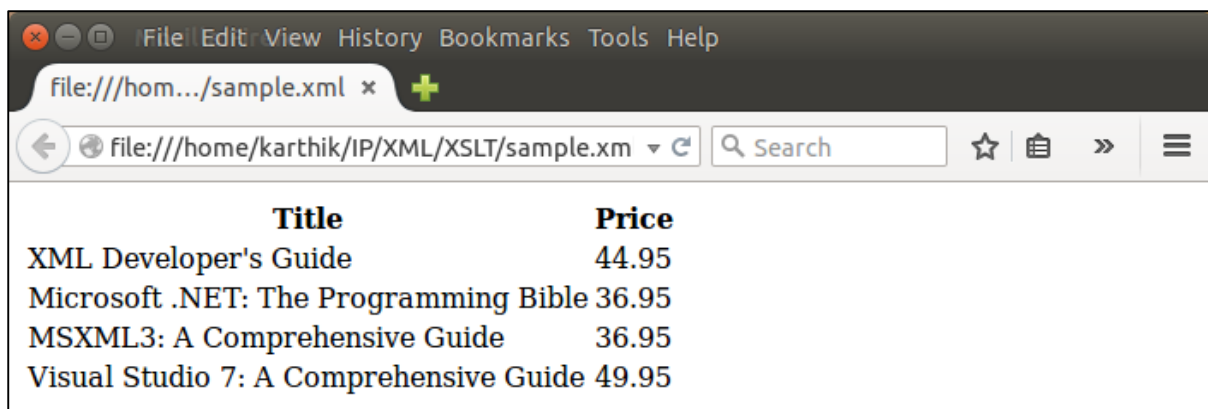
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <xsl:for-each select="catalog/book">
          <xsl:value-of select="title"/>
          <xsl:value-of select="author"/>
          <br/>
        </xsl:for-each>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

**OUTPUT:****CONCAT.XSL**


XML Developer's Guide	Gambardella, Matthew
Midnight Rain	Ralls, Kim
Maeve Ascendant	Corets, Eva
Oberon's Legacy	Corets, Eva
The Sundered Grail	Corets, Eva
Lover Birds	Randall, Cynthia
Splish Splash	Thurman, Paula
Creepy Crawlies	Knorr, Stefan
Paradox Lost	Kress, Peter
Microsoft .NET: The Programming Bible	O'Brien, Tim
MSXML3: A Comprehensive Guide	O'Brien, Tim
Visual Studio 7: A Comprehensive Guide	Galos, Mike

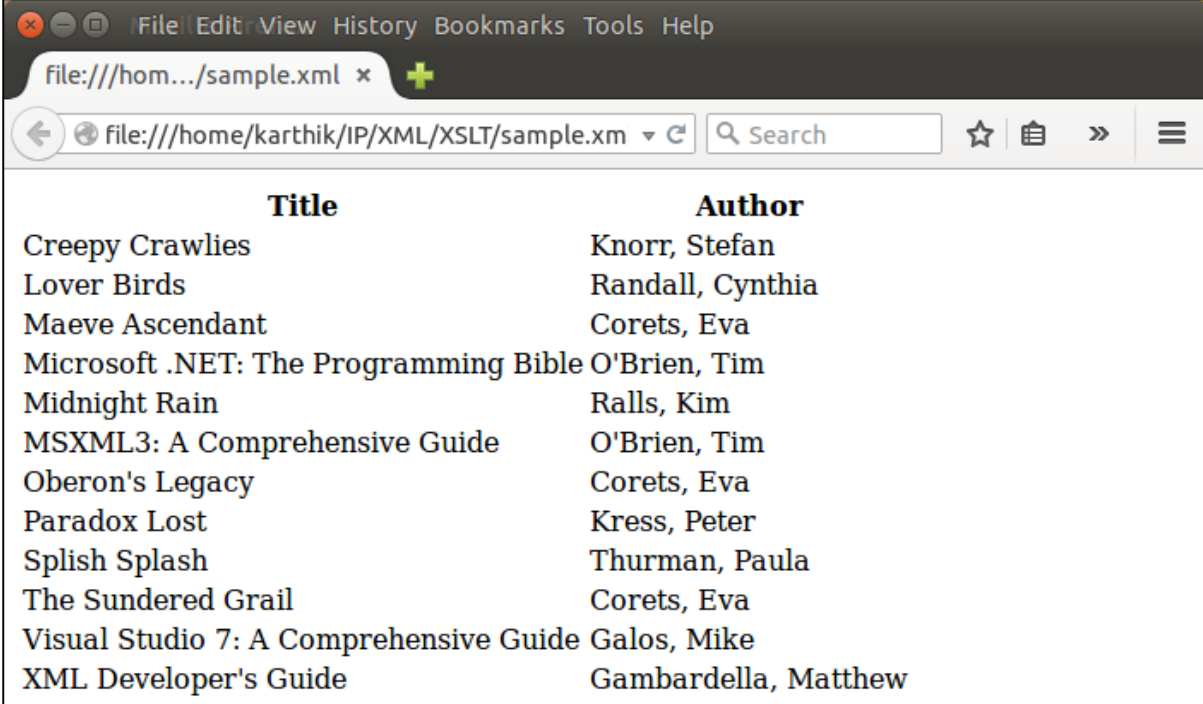
**GENRE.XSL**


Title	Genre
XML Developer's Guide	Computer
Microsoft .NET: The Programming Bible	Computer
MSXML3: A Comprehensive Guide	Computer
Visual Studio 7: A Comprehensive Guide	Computer

**RANGE.XSL**


Title	Price
XML Developer's Guide	44.95
Microsoft .NET: The Programming Bible	36.95
MSXML3: A Comprehensive Guide	36.95
Visual Studio 7: A Comprehensive Guide	49.95

## SORT.XSL



Title	Author
Creepy Crawlies	Knorr, Stefan
Lover Birds	Randall, Cynthia
Maeve Ascendant	Corets, Eva
Microsoft .NET: The Programming Bible	O'Brien, Tim
Midnight Rain	Ralls, Kim
MSXML3: A Comprehensive Guide	O'Brien, Tim
Oberon's Legacy	Corets, Eva
Paradox Lost	Kress, Peter
Splish Splash	Thurman, Paula
The Sundered Grail	Corets, Eva
Visual Studio 7: A Comprehensive Guide	Galos, Mike
XML Developer's Guide	Gambardella, Matthew

## TABLE.XSL



Title	Artist
XML Developer's Guide	Gambardella, Matthew
Midnight Rain	Ralls, Kim
Maeve Ascendant	Corets, Eva
Oberon's Legacy	Corets, Eva
The Sundered Grail	Corets, Eva
Lover Birds	Randall, Cynthia
Splish Splash	Thurman, Paula
Creepy Crawlies	Knorr, Stefan
Paradox Lost	Kress, Peter
Microsoft .NET: The Programming Bible	O'Brien, Tim
MSXML3: A Comprehensive Guide	O'Brien, Tim
Visual Studio 7: A Comprehensive Guide	Galos, Mike

## RESULT:

THE XSL STYLESHEET WAS USED TO DISPLAY A FORMATTED XML DOCUMENT.