ALA PRACTICAL 1

AIM : Write a Client-server application using UDP Protocol in which client sends a message to server. Server will convert that message into upper case and returns the result back to client.

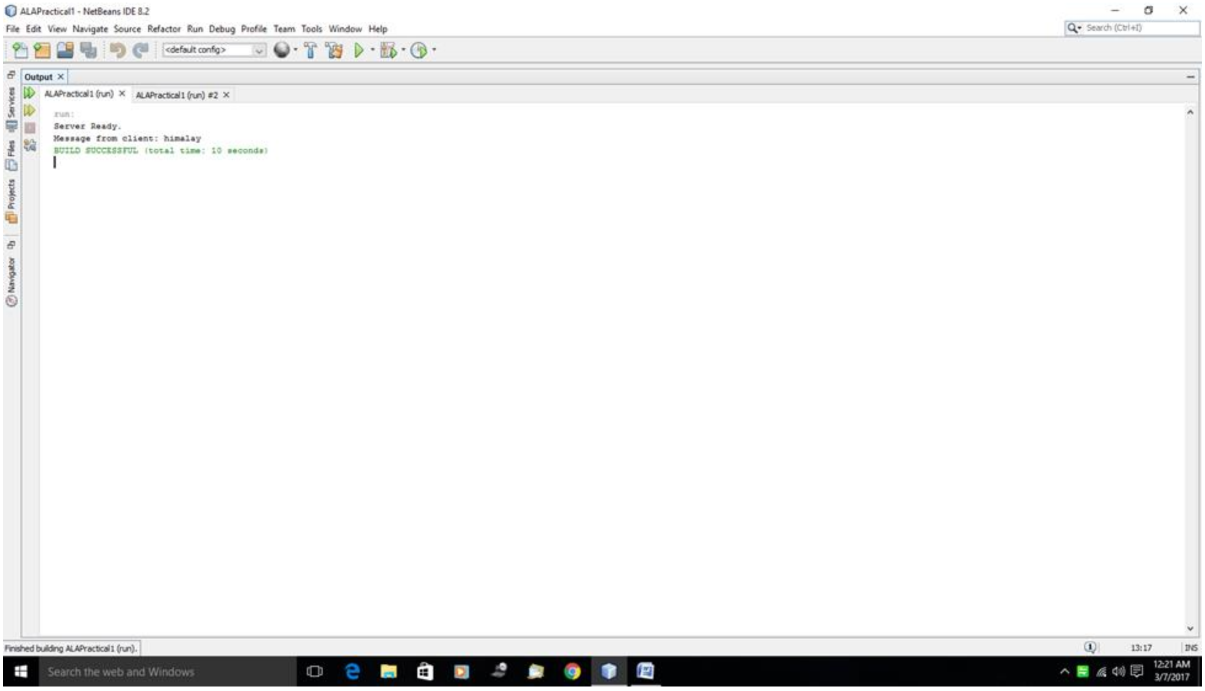
server.java

```
package alapractical1;

import java.io.*;

import java.net.*;

import java.util.*;

public class server

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

        try

        {       DatagramSocket ds=new DatagramSocket(3333);

                System.out.println("Server Ready.");

                byte buf[]=new byte[1000];

                DatagramPacket dp=new DatagramPacket(buf,buf.length);

                ds.receive(dp);

                String str=new String(dp.getData());

                System.out.println("Message from client: "+str);

                String s=str.toUpperCase();

                byte msg[]=s.getBytes();

                DatagramPacket dp1=new
        DatagramPacket(msg,msg.length,dp.getAddress(),dp.getPort());

                ds.send(dp1);

                ds.close();

        }

        catch(Exception e)

        {       System.out.println(e);    }
        }
}
```

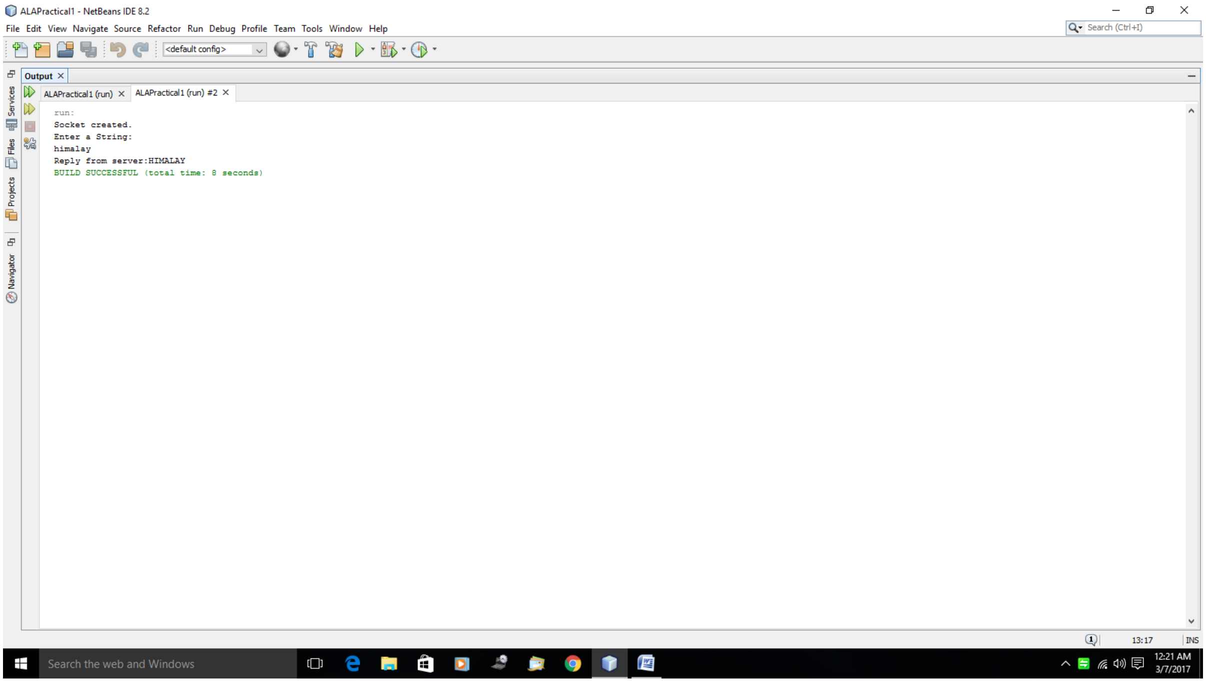
client.java

```
package alapractical1;

import java.io.*;

import java.net.*;

import java.util.*;

public class client

{       public static void main(String [] args)

        {       DatagramSocket ds=null;

                Scanner sc=new Scanner(System.in);

                try

                {       ds=new DatagramSocket();

                        System.out.println("Socket created.");

                        System.out.println("Enter a String: ");

                        String s="";

                        s=sc.nextLine();

                        byte[] m=s.getBytes();

                        InetAddress ip=InetAddress.getLocalHost();

                        int serverport=3333;

                        DatagramPacket request=new DatagramPacket(m,m.length,ip,3333);

                        ds.send(request);

                        byte[] buffer=new byte[1000];

                        DatagramPacket reply=new DatagramPacket(buffer,buffer.length);

                        ds.receive(reply);

                        System.out.println("Reply from server"+":"+new String(reply.getData()));

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

ALA Practical 1 Output


ALAPractical1 - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Search (Ctrl+I)
Output
ALAPractical1 (run)
ALAPractical1 (run) #2
Services
Files
Projects
Navigator
Finished building ALAPractical1 (run).
13:17
INS
Search the web and Windows
12:21 AM
3/7/2017

```
run:
Server Ready.
Message from client: himalay
BUILD SUCCESSFUL (total time: 10 seconds)
```


ALAPractical1 - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Search (Ctrl+I)
Output
ALAPractical1 (run)
ALAPractical1 (run) #2
Services
Files
Projects
Navigator
13:17
INS
Search the web and Windows
12:21 AM
3/7/2017

```
run:
Socket created.
Enter a String:
himalay
Reply from server:HIMALAY
BUILD SUCCESSFUL (total time: 8 seconds)
```

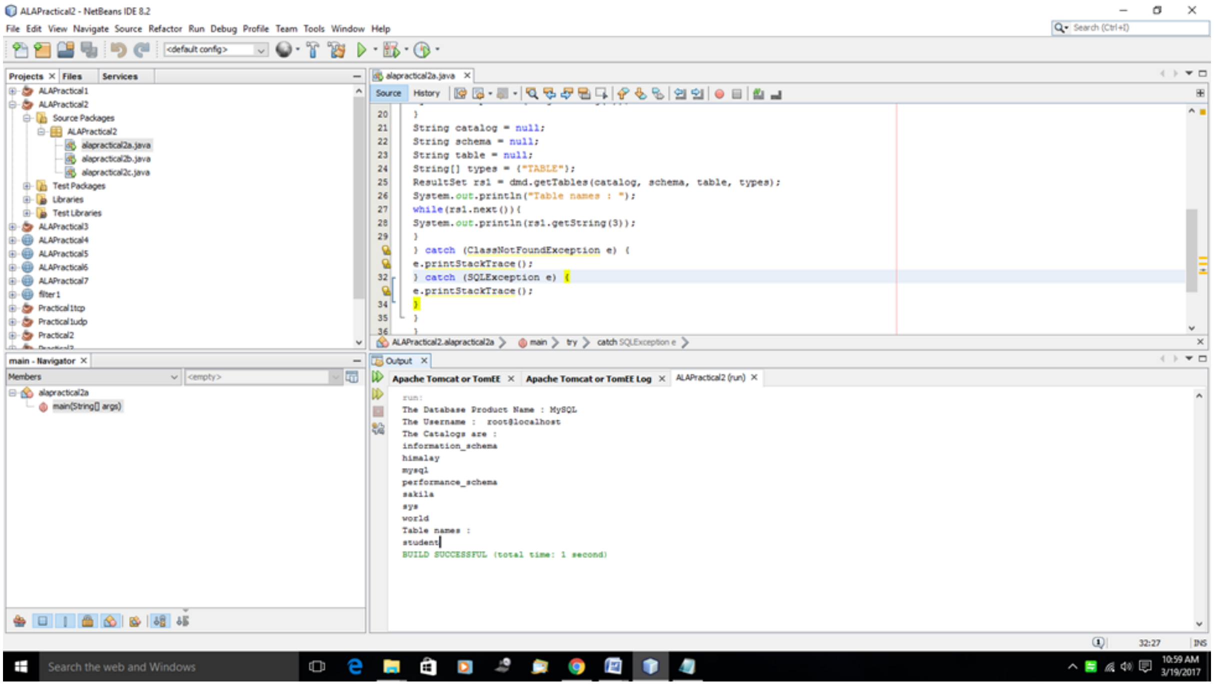
ALA PRACTICAL 2A

AIM : Learn how to process the database metadata.

```
package ALAPractical2;

import java.sql.*;

public class alapractical2a

{		public static void main(String[] args)

		{		try

				{		Class.forName("com.mysql.jdbc.Driver");

						String username = "root";

						String password = "himalaypatel";

						Connection c = DriverManager.getConnection("jdbc:mysql://localhost:3306/himalay",username,password);

						DatabaseMetaData dmd = c.getMetaData();

						System.out.println("The Database Product Name : "+dmd.getDatabaseProductName());

						System.out.println("The Username :  "+dmd.getUserName());

						System.out.println("The Catalogs are : ");

						ResultSet rs = dmd.getCatalogs();

						while(rs.next())

						{		System.out.println(rs.getString(1));		}

						String catalog = null;

						String schema = null;

						String table = null;

						String[] types = {"TABLE"};

						ResultSet rs1 = dmd.getTables(catalog, schema, table, types);

						System.out.println("Table names : ");

						while(rs1.next())
```

```
                        {           System.out.println(rs1.getString(3));       }
                }
                catch (ClassNotFoundException e)
                {           e.printStackTrace();        }
                catch (SQLException e)
                {           e.printStackTrace();        }
        }
}
```

ALA PRACTICAL 2A OUTPUT



```
}
String catalog = null;
String schema = null;
String table = null;
String[] types = {"TABLE"};
ResultSet rs1 = dmd.getTables(catalog, schema, table, types);
System.out.println("Table names : ");
while(rs1.next()){
System.out.println(rs1.getString(3));
}
} catch (ClassNotFoundException e) {
e.printStackTrace();
} catch (SQLException e) {
e.printStackTrace();
}
}
}
```

```
run:
The Database Product Name : MySQL
The Username :  root@localhost
The Catalogs are :
information_schema
himalay
mysql
performance_schema
sakila
sys
world
Table names :
student
BUILD SUCCESSFUL (total time: 1 second)
```

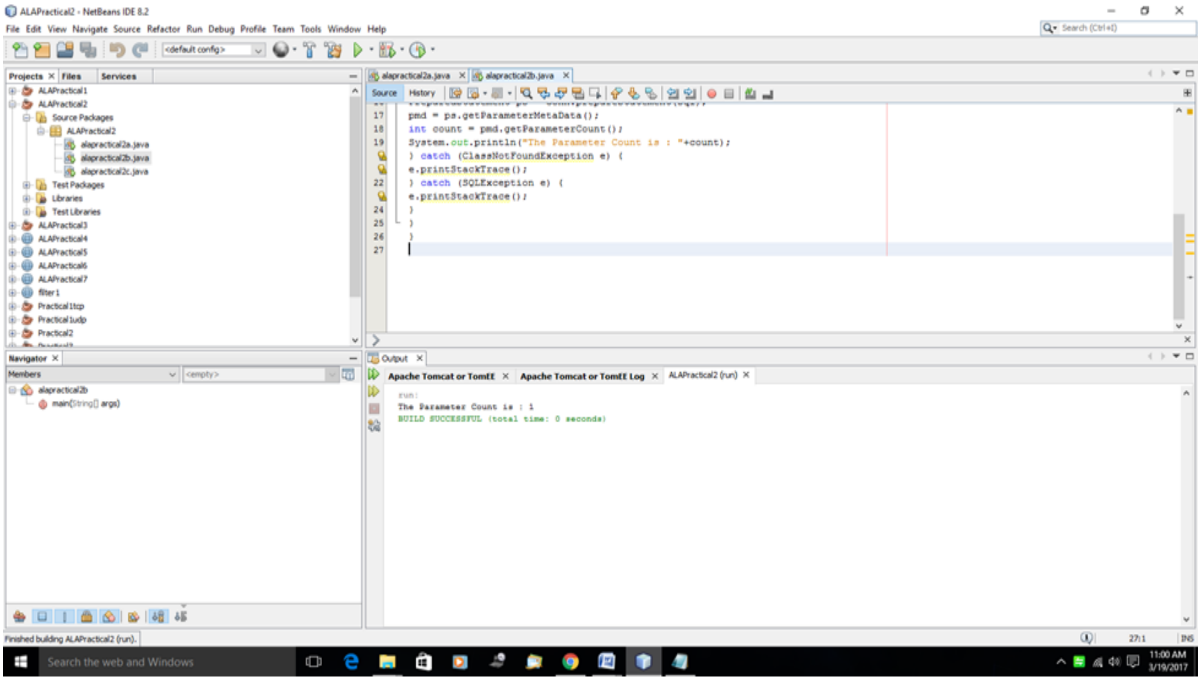
ALAPractical2 - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
alapractical2a.java
Source History
ALAPractical2.alapractical2a > main > try > catch SQLException e
Output
Apache Tomcat or TomEE
Apache Tomcat or TomEE Log
ALAPractical2 (run)

ALA PRACTICAL 2B

AIM : Learn how to get metadata of a query parameter.

```
package ALAPractical2;

import java.sql.*;

public class alapractical2b {

public static void main(String[] args)

{       try

        {       Class.forName("com.mysql.jdbc.Driver");

                String username = "root";

                String password = "himalaypatel";

                Connection conn = 
        DriverManager.getConnection("jdbc:mysql://localhost:3306/himalay",username,password);

                ParameterMetaData pmd;

                String sql = "select * from Student where Semester=? ";

                PreparedStatement ps = conn.prepareStatement(sql);

                pmd = ps.getParameterMetaData();

                int count = pmd.getParameterCount();

                System.out.println("The Parameter Count is : "+count);

        }

        catch (ClassNotFoundException e)

        {       e.printStackTrace();    }

        catch (SQLException e)

        {       e.printStackTrace();    }

        }

}
```

ALA PRACTICAL 2B OUTPUT



```
pmd = ps.getParameterMetaData();
int count = pmd.getParameterCount();
System.out.println("The Parameter Count is : "+count);
} catch (ClassNotFoundException e) {
e.printStackTrace();
} catch (SQLException e) {
e.printStackTrace();
}
}
}
```

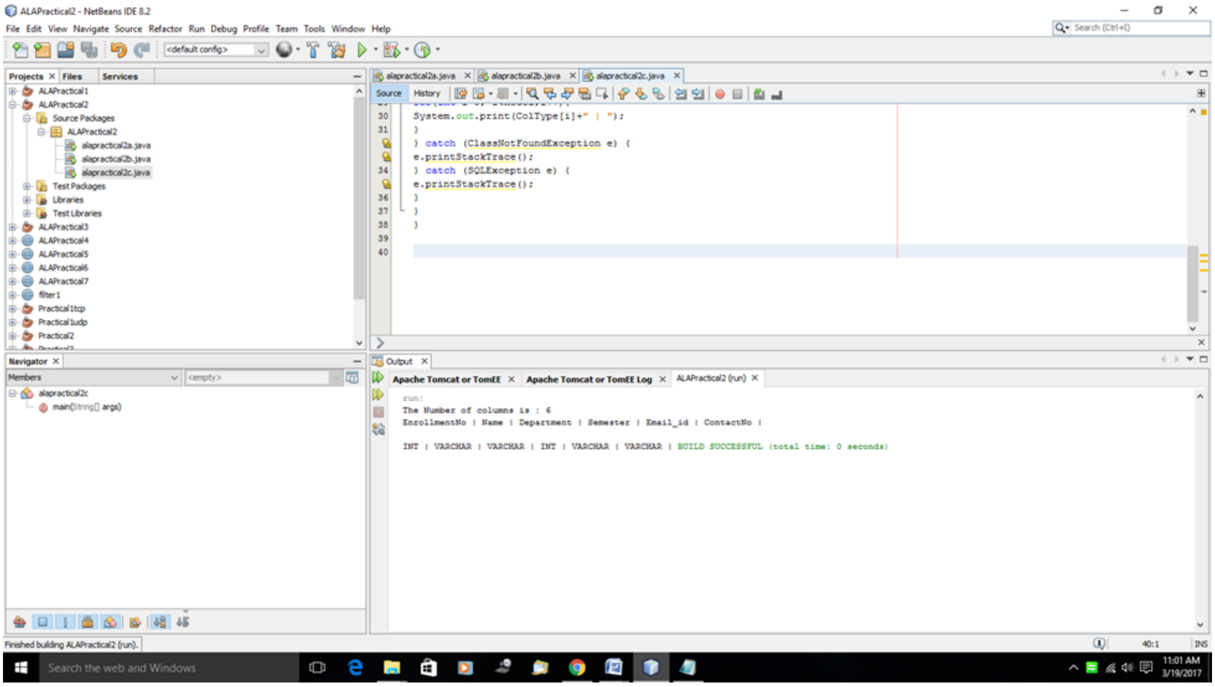
```
run:
The Parameter Count is : 1
BUILD SUCCESSFUL (total time: 0 seconds)
```

ALA PRACTICAL 2C

AIM : Learn how to get metadata from the results of a query.

```
package ALAPractical2;

import java.sql.*;

public class alapractical2c

{       public static void main(String[] args)

        {       try

                {       Class.forName("com.mysql.jdbc.Driver");

                        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/himalay","root","himalaypatel");

                        String query = "select * from Student";

                        Statement s = con.createStatement();

                        Query(query);

                        java.sql.ResultSetMetaData rmd = rs.getMetaData();

                        int noCol = rmd.getColumnCount();

                        Number of columns is : "+noCol);

                        String ColName[] = new String[noCol];

                        for(int i=0;i<noCol;i++)

                        {       ColName[i] = rmd.getColumnName(i+1);        }

                        String ColType[] = new String[noCol];

                        for(int i=0;i<noCol;i++)

                        {       ColType[i] = rmd.getColumnTypeName(i+1);    }

                        for(int i=0; i<noCol;i++)

                        {       System.out.print(ColName[i]+" | ");     }

                        System.out.println("\n");

                        for(int i=0; i<noCol;i++)

                        {       System.out.print(ColType[i]+" | ");     }
                }
                catch (ClassNotFoundException e)
                {         e.printStackTrace();      }
                catch (SQLException e)
                {         e.printStackTrace();      }
        }
}
```

ALA PRACTICAL 2C OUTPUT



ALAPractical2 - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
alapractical2a.java
alapractical2b.java
alapractical2c.java
Source History
Output
Apache Tomcat or TomEE
Apache Tomcat or TomEE Log
ALAPractical2 (run)

```
System.out.print(ColType[i]+" | ");
}
} catch (ClassNotFoundException e) {
e.printStackTrace();
} catch (SQLException e) {
e.printStackTrace();
}
}
}
```

```
run:
The Number of columns is : 6
EnrollmentNo | Name | Department | Semester | Email_id | ContactNo |

INT | VARCHAR | VARCHAR | INT | VARCHAR | VARCHAR | BUILD SUCCESSFUL (total time: 0 seconds)
```

Finished building ALAPractical2 (run).

ALA PRACTICAL 3

AIM :Write a servlet that shows all request headers and response headers.

index.html

```
<html>

<head>

<title>Request and Response Headers</title>

<meta charset="UTF-8">

        <meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

        <a href="Servlet1">Get Headers</a>

</body>

</html>
```

Servlet1.java (for displaying Request Headers)

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

import java.util.*;

public class Servlet1 extends HttpServlet

{       protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

        {       doPost(request,response);       }

        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

        {       response.setContentType("text/html");

                PrintWriter out=response.getWriter();

                try
                {         String title = "HTTP Request Headers";

                          String docType = "<!doctype html public \"-//w3c//dtd html 4.0 " +

                 "transitional//en\">\n";

                          out.println(docType +  "<html>\n" +  "<head><title>" + title +
                "</title></head>\n"+ "<body bgcolor=\"orange\">\n" + "<h1 align=\"center\">" +
                title + "</h1>\n" + "<table width=\"100%\" border=\"1\" align=\"center\">\n" +
                          "<tr bgcolor=\"yellow\">\n" + "<th>Header Name</th><th>Header
                Value(s)</th>\n"+ "</tr>\n");

        }

        catch(Exception e)

        {         e.printStackTrace();       }

                Enumeration headerNames = request.getHeaderNames();

                while(headerNames.hasMoreElements())

                {         String paramName = (String)headerNames.nextElement();

                          out.print("<tr><td>" + paramName + "</td>\n");

                          String paramValue = request.getHeader(paramName);

                          amValue + "</td></tr>\n");

                }

                response.setContentType("text/html");

                out.println("</table>");

                out.println("<a href="+"Servlet3"+">Get Response Headers</a>");

                out.println("\n</body></html>");

        }

}
```

Servlet2.java (for displaying Response Headers)

```
import java.io.*;

import javax.servlet.*;

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

public class Servlet2 extends HttpServlet

{           protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

            {           doPost(request,response);         }

            protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

            {           response.setContentType("text/html");

                        PrintWriter out=response.getWriter();

                        try

                        {           out = response.getWriter();

                                    String title = "HTTP Response Headers";

                                    String docType = "<!doctype html public \"-//w3c//dtd html 4.0 " +
                        "transitional//en\">\n";

                                    out.println(docType + "<html>\n" + "<head><title>" + title +
                        "</title></head>\n"+ "<body bgcolor=\"green\">\n" + "<h1 align=\"center\">" +
                        title + "</h1>\n" + "<table width=\"100%\" border=\"1\" align=\"center\">\n" + "<tr
                        bgcolor=\"purple\">\n" + "<th>Header Name</th><th>Header Value(s)</th>\n"+
                        "</tr>\n");

            }

            catch(Exception e)

            {           e.printStackTrace();       }

            Collection<String> respHeader= response.getHeaderNames();

            <String> headerNames = respHeader.iterator();

            while(headerNames.hasNext())

            {           String paramName = headerNames.next();

                        out.print("<tr><td>" + paramName + "</td>\n");

                        String paramValue = response.getHeader(paramName);

                        out.println("<td> " + paramValue + "</td></tr>\n");

            }
            out.println("</table>");

            out.println("<a href="+"Servlet1"+">Get Request Headers</a>");

            out.println("\n</body></html>");

            }

}
```

Servlet3.java (for adding Response headers)

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

import java.util.*;

public class Servlet3 extends HttpServlet

{           protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

            {           doPost(request,response);         }

            protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException

            {           response.setHeader("Himalay", "Patel");

                        response.setHeader("BrowserUsed", "Chrome");

                        RequestDispatcher dispatcher =
            getServletContext().getNamedDispatcher("Servlet2");

                        dispatcher.forward(request, response);

            }

}
```

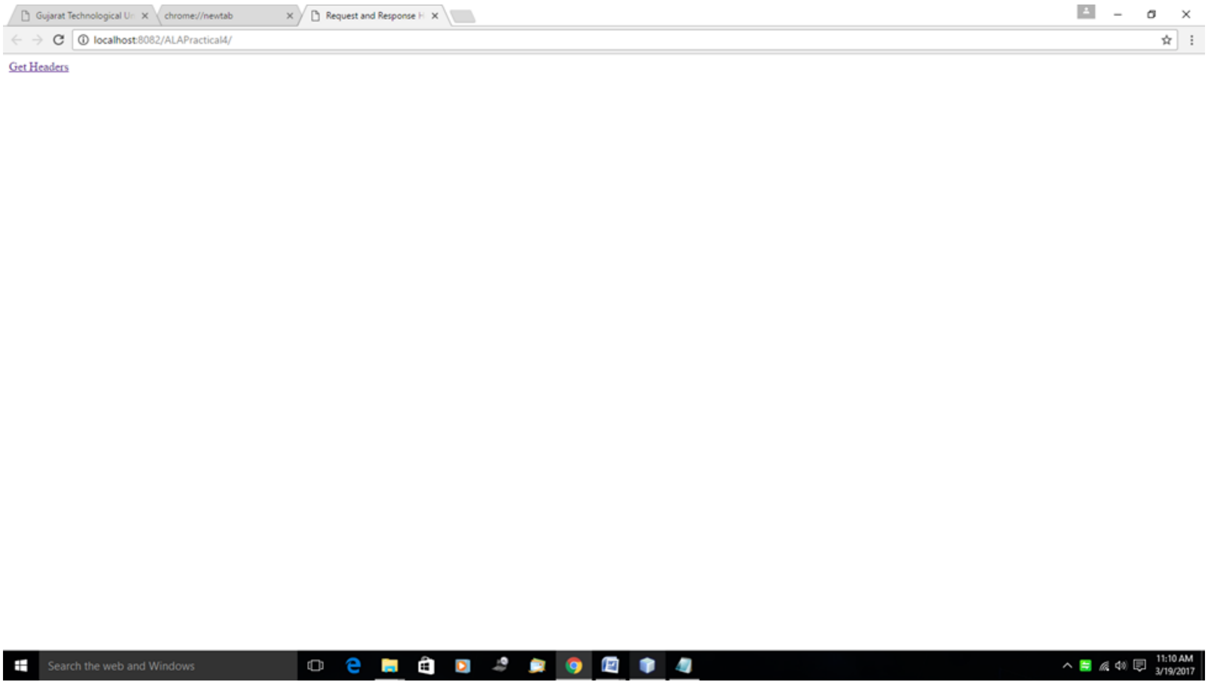
web.xml

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

<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

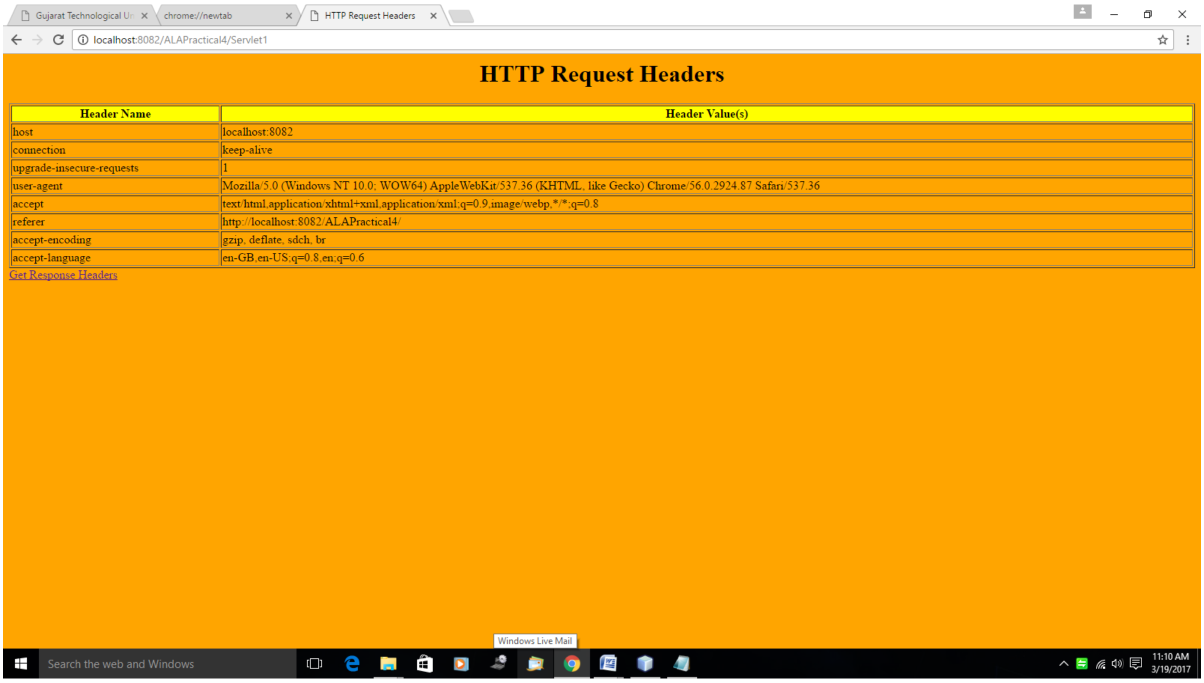
```
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
<servlet>
        <servlet-name>Servlet1</servlet-name>
        <servlet-class>Servlet1</servlet-class>
</servlet>
<servlet>
        <servlet-name>Servlet2</servlet-name>
        <servlet-class>Servlet2</servlet-class>
</servlet>
 <servlet>
        <servlet-name>Servlet3</servlet-name>
        <servlet-class>Servlet3</servlet-class>
</servlet>
<servlet-mapping>
        <servlet-name>Servlet1</servlet-name>
        <url-pattern>/Servlet1</url-pattern>
</servlet-mapping>
<servlet-mapping>
        <servlet-name>Servlet2</servlet-name>
        <url-pattern>/Servlet2</url-pattern>
</servlet-mapping>
 <servlet-mapping>
        <servlet-name>Servlet3</servlet-name>
        <url-pattern>/Servlet3</url-pattern>
</servlet-mapping>
<welcome-file-list>
        <welcome-file>index.html</welcome-file>
```

```
</welcome-file-list>

</web-app>
```

ALA PRACTICAL 3 OUTPUT



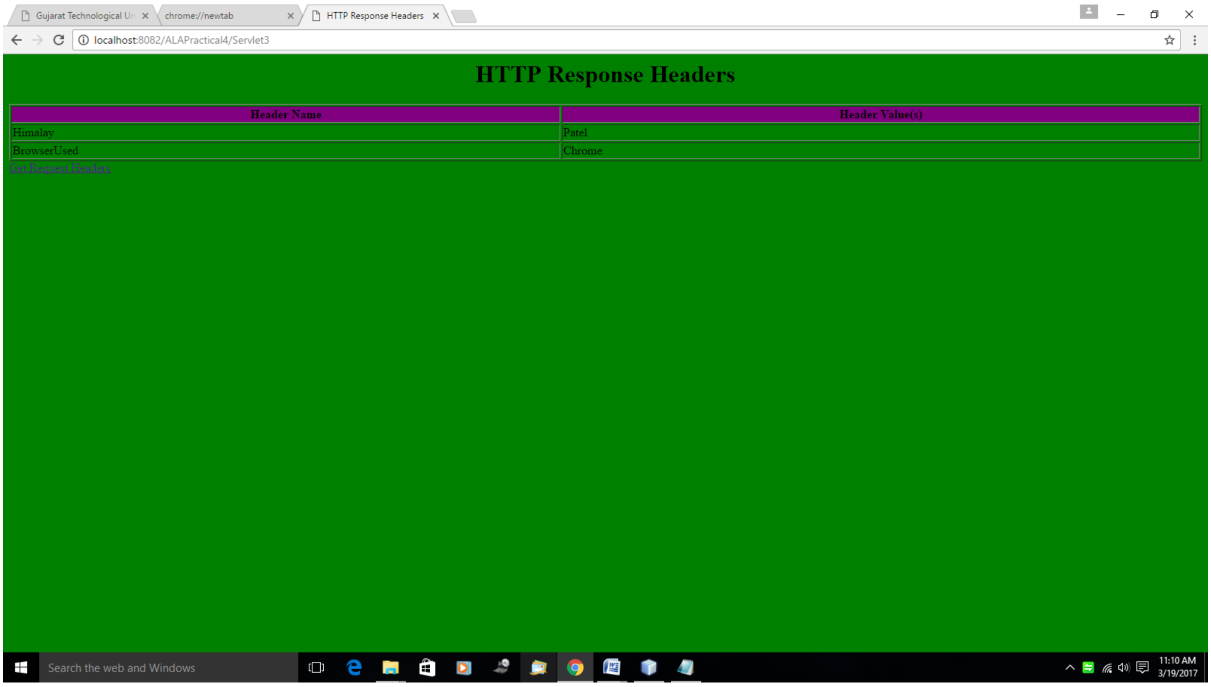
Get Headers



HTTP Request Headers

Header Name	Header Value(s)
host	localhost:8082
connection	keep-alive
upgrade-insecure-requests	1
user-agent	Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/56.0.2924.87 Safari/537.36
accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
referer	http://localhost:8082/ALAPractical4/
accept-encoding	gzip, deflate, sdch, br
accept-language	en-GB,en-US;q=0.8,en;q=0.6

Get Response Headers



HTTP Response Headers

Header Name	Header Value(s)
Himalay	Patel
BrowserUsed	Chrome

Get Request Headers

ALA PRACTICAL 4

AIM : Write a servlet that instructs the browser to reconnect every five seconds. Display the time (print new java.util.Date()) on each connection. (response header)

index.html

```
<html>

<head>

<title>Time</title>

<meta charset="UTF-8">

        <meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

        <a href="Servlet1">Time</a>

</body>

</html>
```

Servlet1.java (for getting and displaying time)

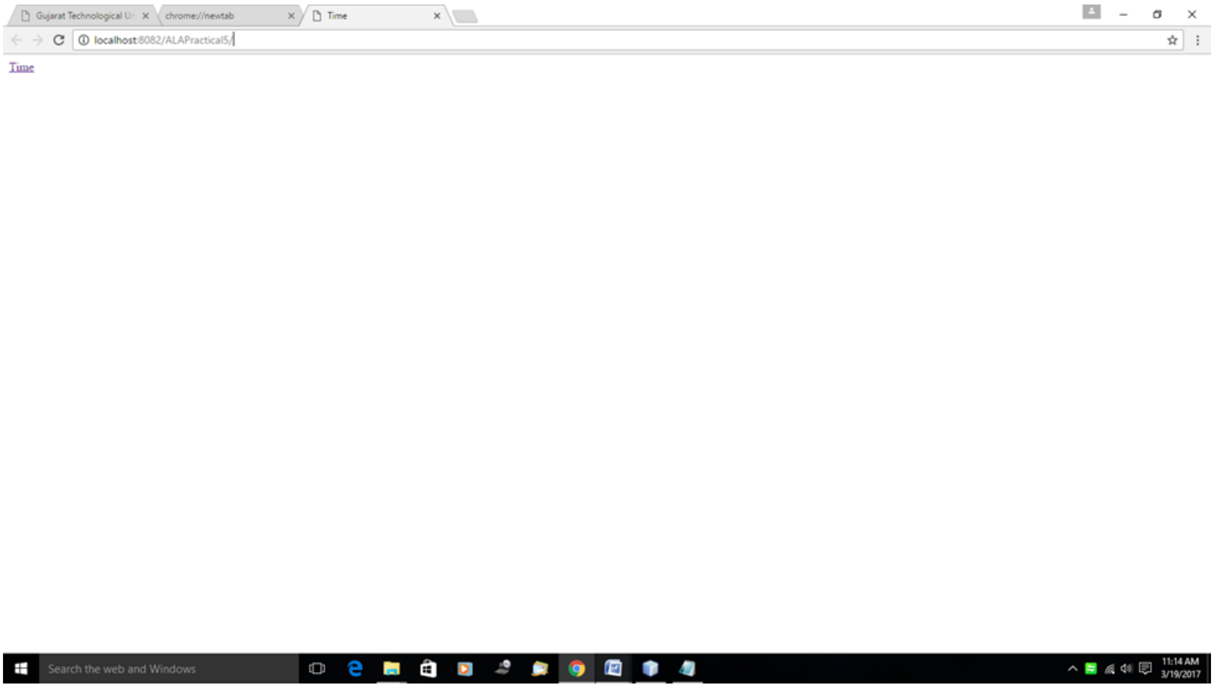
```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

import java.util.*;

public class Servlet1 extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doPost(request,response);

}

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setIntHeader("Refresh", 5);
response.setContentType("text/html");

Calendar calendar = new GregorianCalendar();

String am_pm;

int hour = calendar.get(Calendar.HOUR);

int minute = calendar.get(Calendar.MINUTE);

int second = calendar.get(Calendar.SECOND);

if(calendar.get(Calendar.AM_PM) == 0)

am_pm = "AM";

else

am_pm = "PM";

String CT = hour+":"+ minute +":"+ second +" "+ am_pm;

PrintWriter out = response.getWriter();

String title = "Auto Refresh Webpage";

String docType = "<!doctype html public \"-//w3c//dtd html 4.0 " +

        "transitional//en\">\n";

              out.println(docType +

              "<html>\n" +

              "<head><title>" + title + "</title></head>\n"+

              "<body bgcolor=\"orange\">\n" +

              "<h1 align=\"center\">" + title + "</h1>\n" +

              "<p>Current Time is: " + CT + "</p>\n");

       }

}
```

web.xml

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

<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

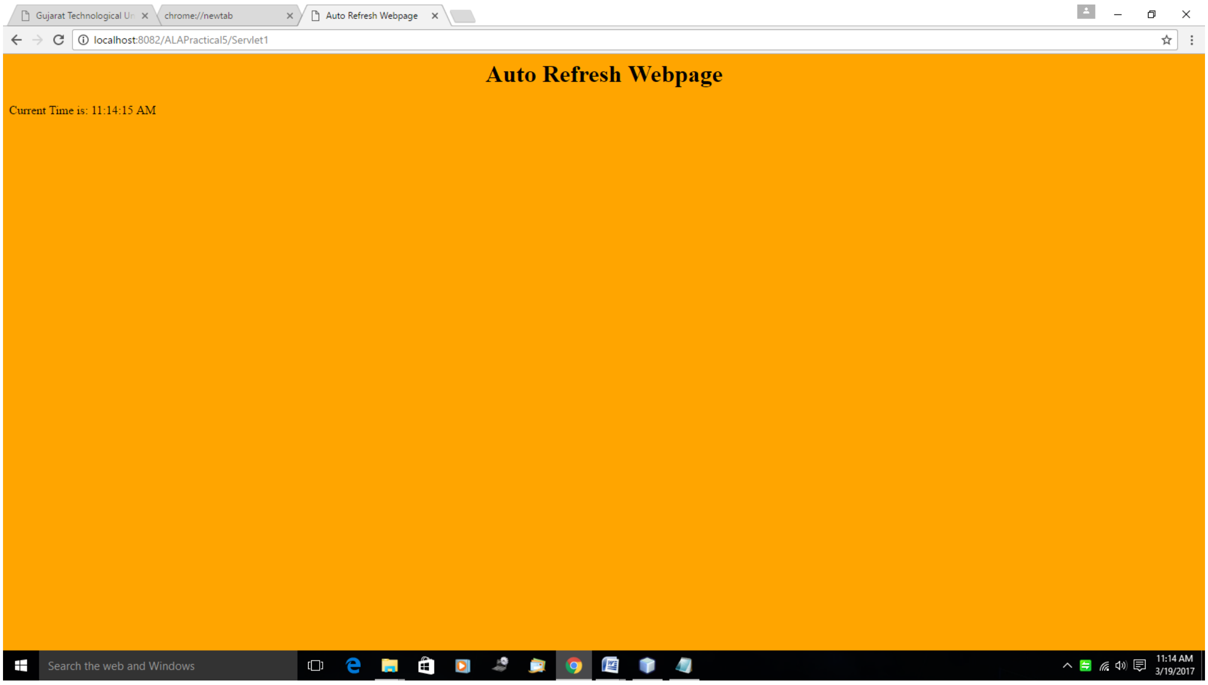
```
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
    <servlet>
        <servlet-name>Servlet1</servlet-name>
        <servlet-class>Servlet1</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>Servlet1</servlet-name>
        <url-pattern>/Servlet1</url-pattern>
    </servlet-mapping>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

ALA PRACTICAL 4 OUTPUT



localhost:8082/ALAPractical5/

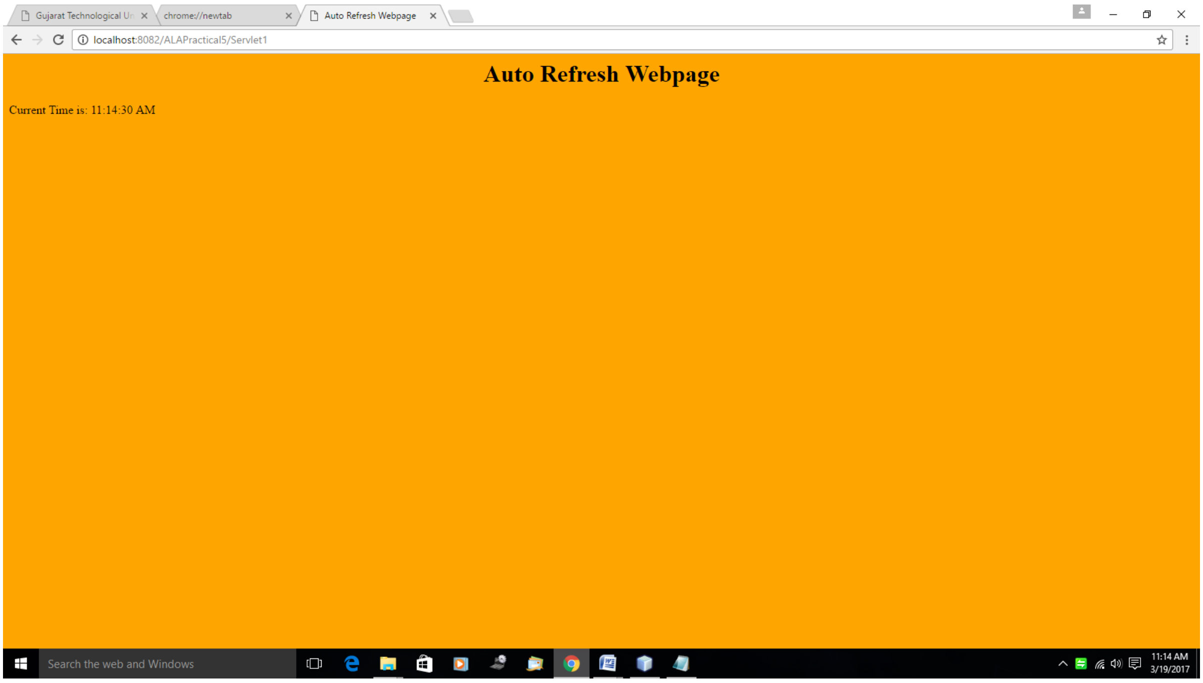
Time



Auto Refresh Webpage

Current Time is: 11:14:15 AM

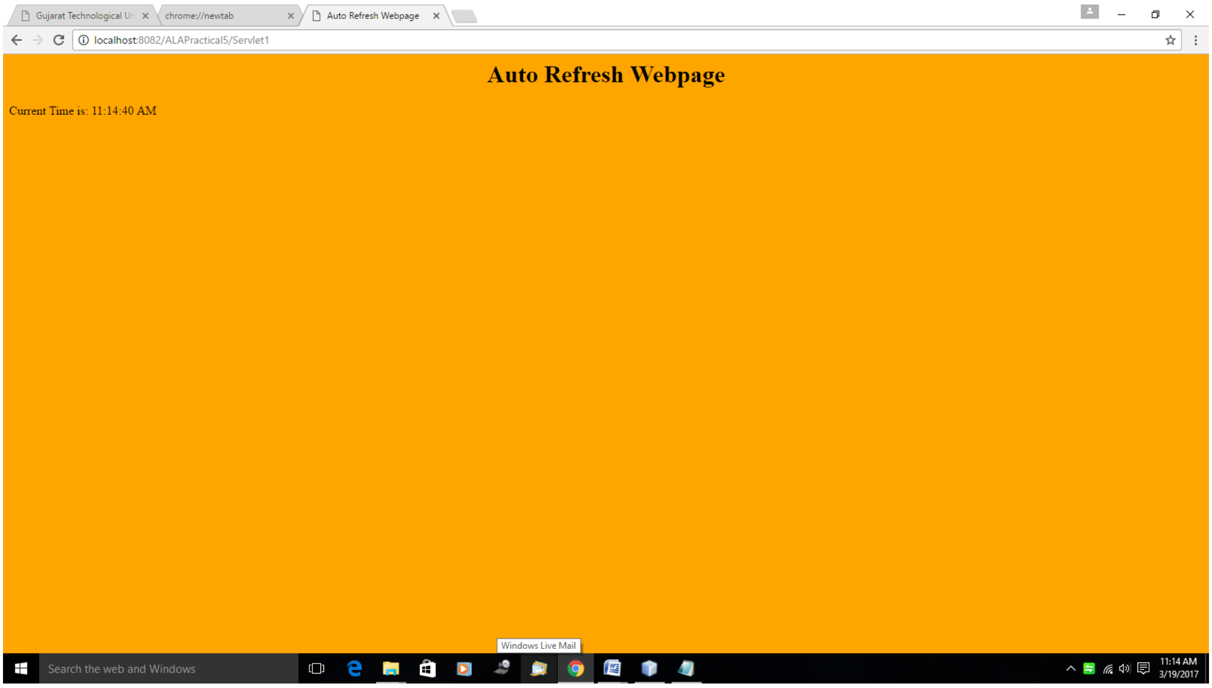
After 15 seconds-



Auto Refresh Webpage

Current Time is: 11:14:30 AM

After 25 seconds-



Auto Refresh Webpage

Current Time is: 11:14:40 AM

ALA PRACTICAL 5

AIM : Invoke the method from within template text using JSP elements (example o/p - Sum of 3 and 4 is 7.) Print above line to console as well using JSP elements.

index.html

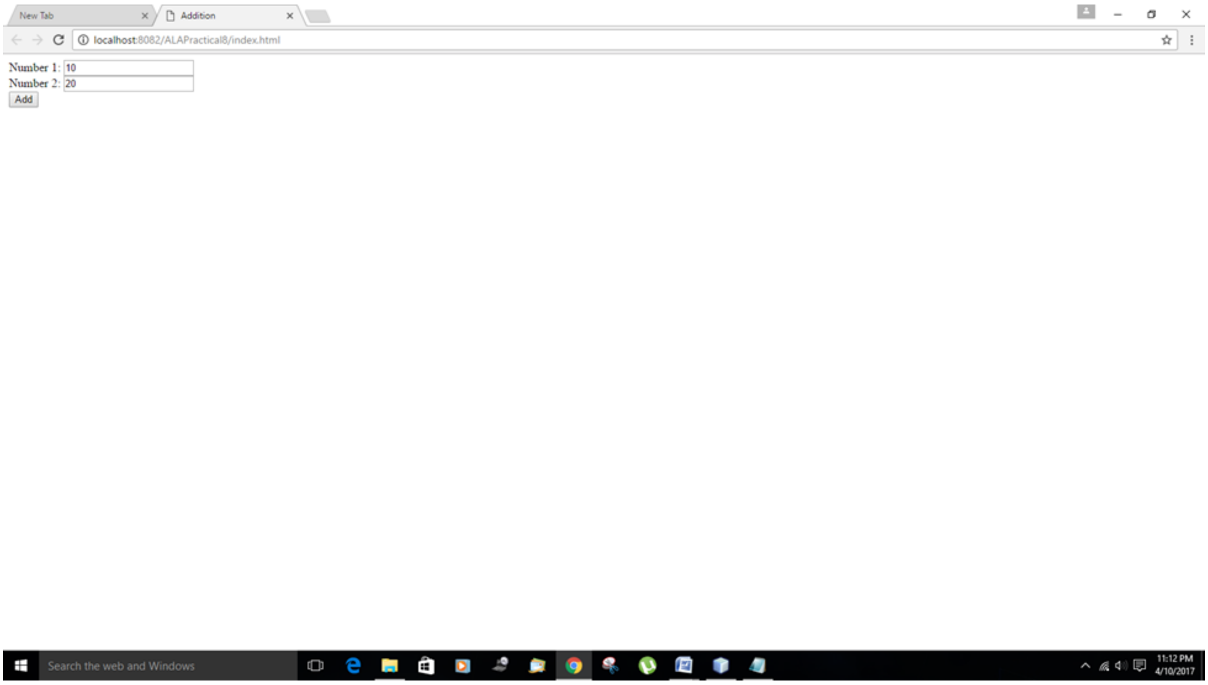
```
<html>

  <head>

    <title>Addition</title>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

  </head>

  <body>

    <form action="AddJSP.jsp" method="post">

      Number 1: <input type="text" name="n1"><br>

      Number 2: <input type="text" name="n2"><br>

      <input type="submit" value="Add">

    </form>

  </body>

</html>
```

AddJSP.jsp

```
<%--

  Document   : AddJSP

  Created on : Mar 24, 2017, 1:11:52 PM

  Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>
```

```
   <head>

      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

      <title>Addition of 2 numbers</title>

   </head>

   <body>

  <%

         int x= Integer.parseInt(request.getParameter("n1"));

         int y=Integer.parseInt(request.getParameter("n2"));

      %>

     <%!

         int add(int a,int b)

         {

            return a+b;

         }

      %>

  <%= "Addition of " + x + " and " +  y  + " is " + add(x,y)%>

    </body>

</html>
```

ALA PRACTICAL 5 OUTPUT



New Tab
Addition
localhost:8082/ALAPractical8/index.html
Search the web and Windows
11:12 PM
4/10/2017

Number 1: 10

Number 2: 20

Add

ALA PRACTICAL 6

AIM : Write a servlet that sends the first 9 requests to Welcome Page, the next request to Reject Page, and then repeats. That is, every tenth request should get sent to Reject Page and the rest should get sent to Welcome Page.

index.html

```
<html>

<head>

<title>Multiple Requests</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<a href="Servlet1">Start</a>

</body>

</html>
```

Servlet1.java (to redirect/dispatch requests)

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

public class Servlet1 extends HttpServlet {

public int count;

public void init()

{

count=1;

}

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {
```

```
doPost(request,response);

}

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out=response.getWriter();

if(count!=10){

out.println("<html><body><h2>Count is " +count+ " </h2></body></html>");

RequestDispatcher rd= request.getRequestDispatcher("welcome.html");

rd.include(request,response);

count++;

}

else{

out.println("<html><body><h2>Count is "+count+" </h2></body></html");

RequestDispatcher rd= request.getRequestDispatcher("reject.html");

rd.include(request,response);

count=1;

}

out.println("<html><body><a href=\"Servlet1\">Again</a></body></html");

}

}
```

welcome.html

```
<html>

<head>

<title>Welcome</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
</head>

<body>

   <h2>Welcome Page</h2>

</body>

</html>
```

reject.html

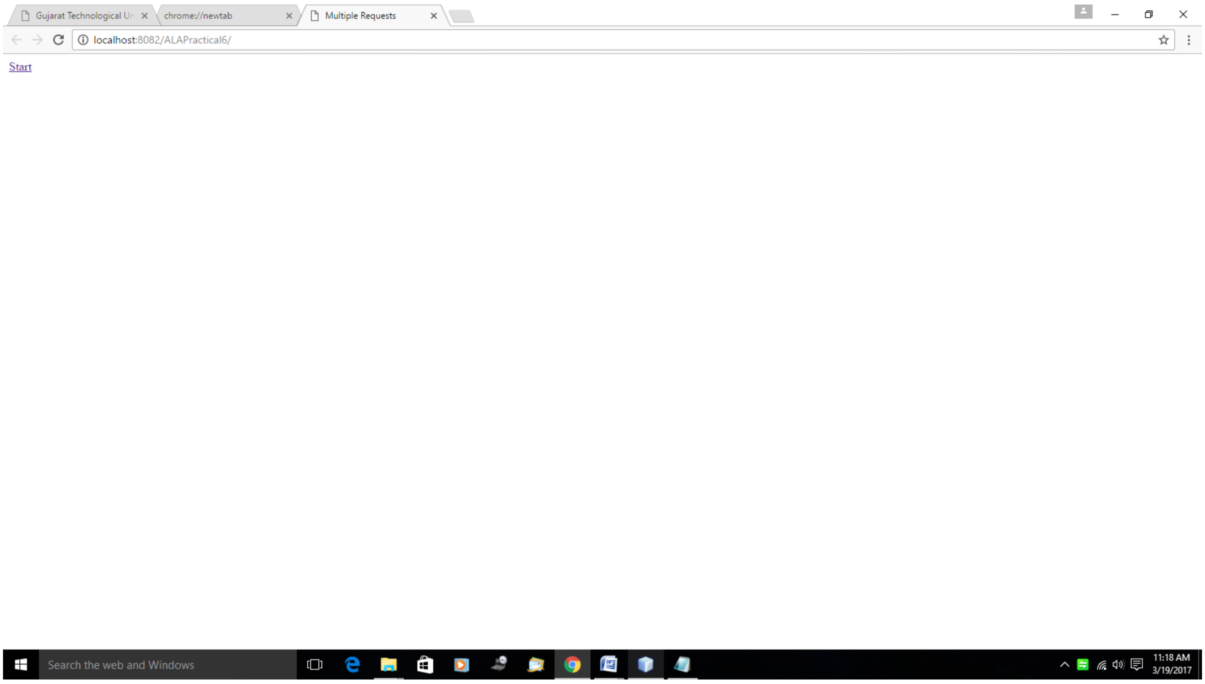
```
<html>

<head>

<title>Reject</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

   <h2>Reject Page</h2>

</body>

</html>
```

web.xml

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

<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">

   <servlet>

      <servlet-name>Servlet1</servlet-name>

      <servlet-class>Servlet1</servlet-class>

   </servlet>

   <servlet-mapping>
```

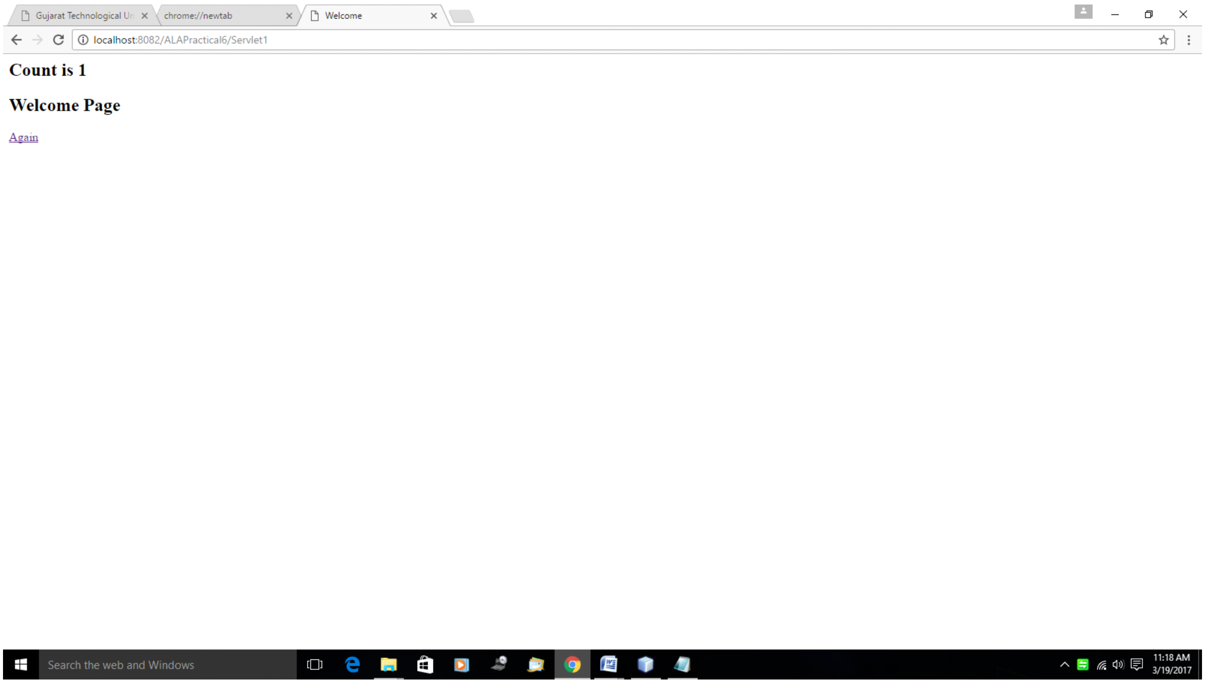
```
        <servlet-name>Servlet1</servlet-name>

        <url-pattern>/Servlet1</url-pattern>

    </servlet-mapping>

    <welcome-file-list>

        <welcome-file>index.html</welcome-file>

    </welcome-file-list>

</web-app>
```

ALA PRACTICAL 6 OUTPUT



Start

Initially, count=1



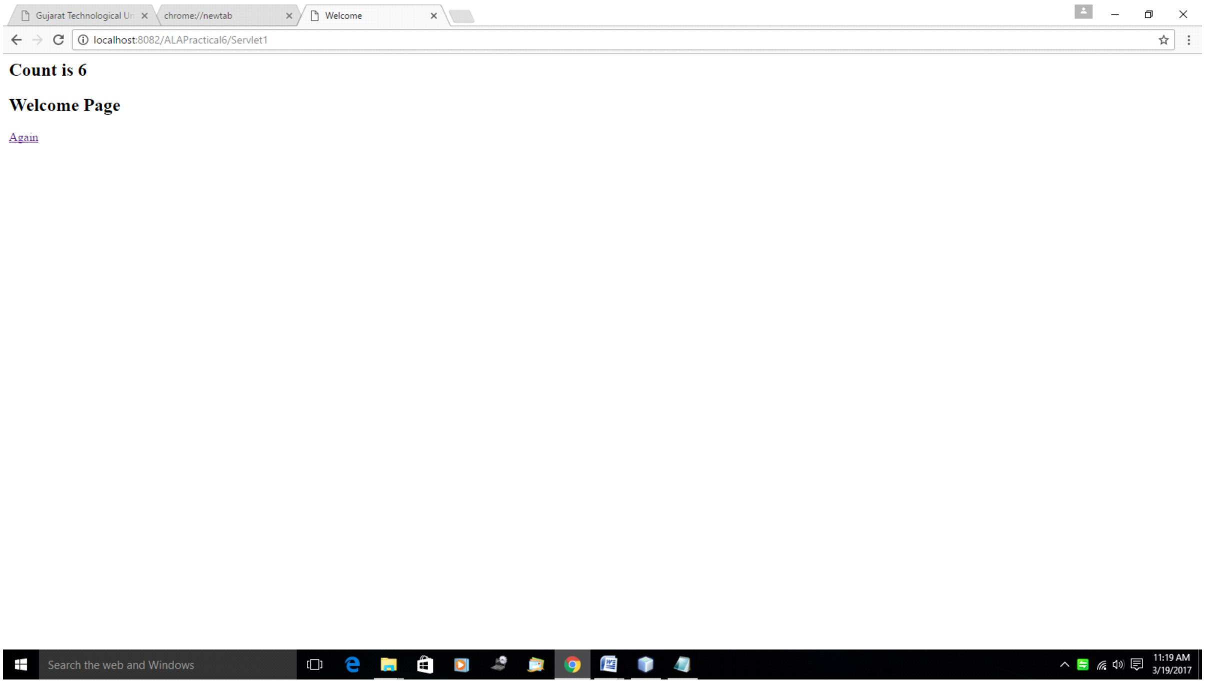
Gujarat Technological Un
chrome://newtab
Welcome
localhost:8082/ALAPractical6/Servlet1
Search the web and Windows
11:18 AM
3/19/2017

Count is 1

Welcome Page

Again

After 6 requests-



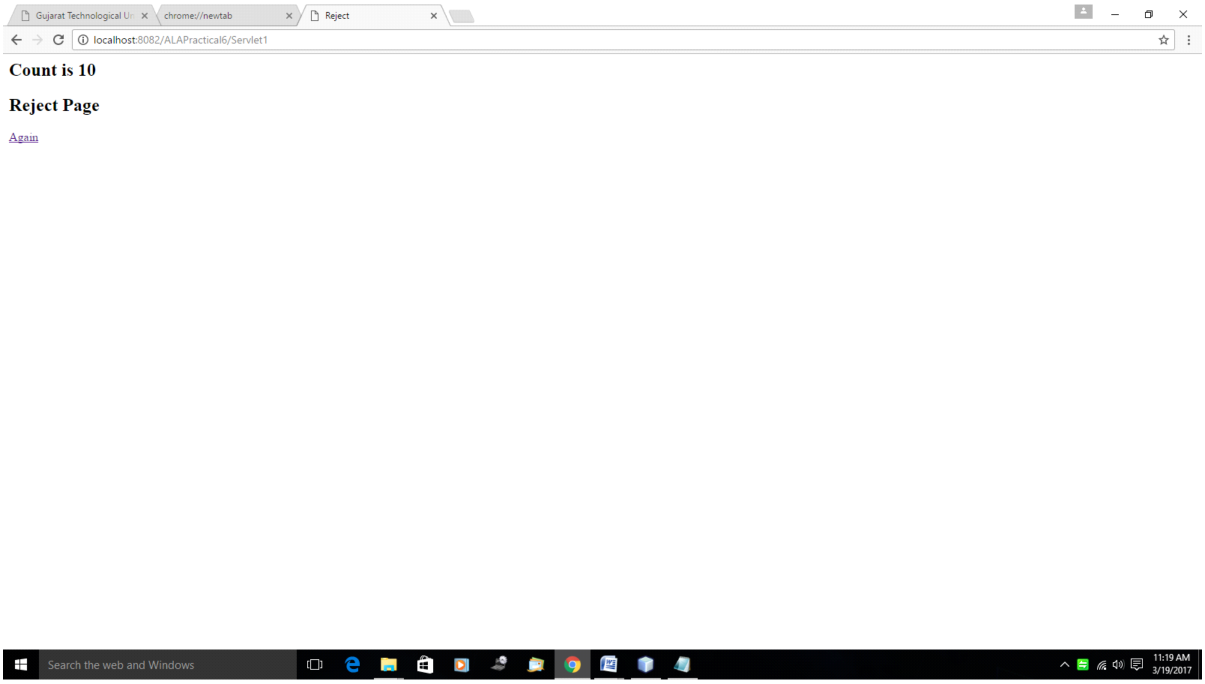
Gujarat Technological Un
chrome://newtab
Welcome
localhost:8082/ALAPractical6/Servlet1
Search the web and Windows
11:19 AM
3/19/2017

Count is 6

Welcome Page

Again

10th request (Reject page)-

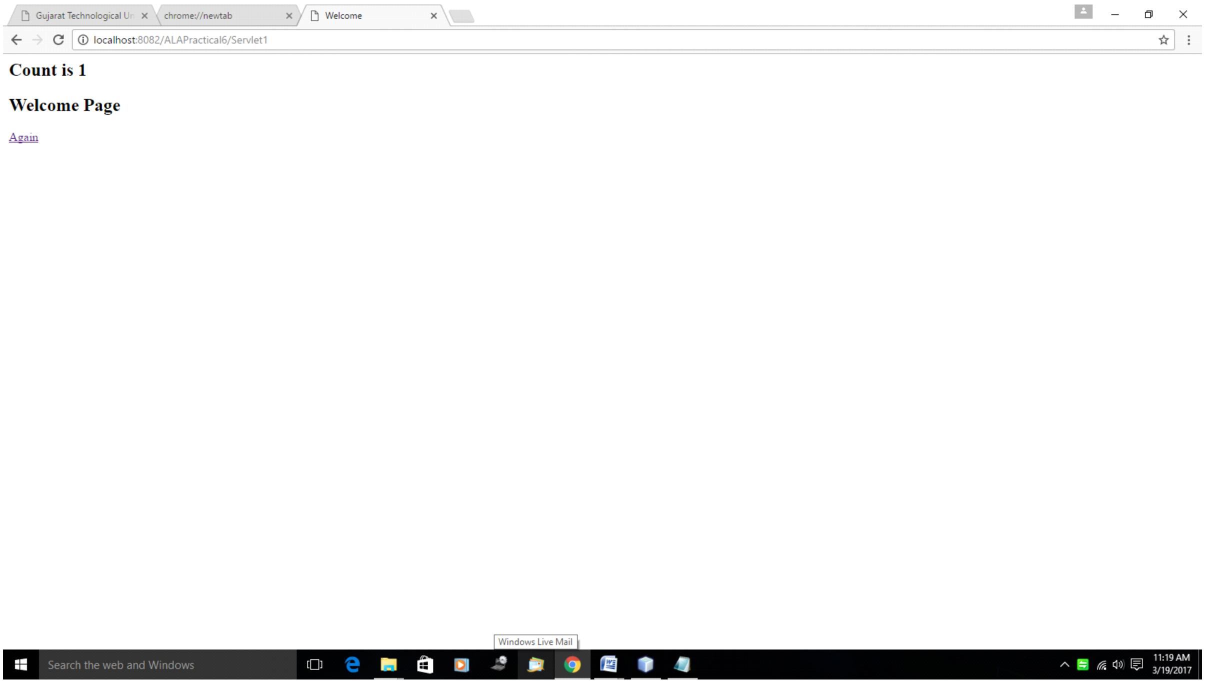

Gujarat Technological Ur
chrome://newtab
Reject
localhost:8082/ALAPractical6/Servlet1
Search the web and Windows
11:19 AM
3/19/2017

Count is 10

Reject Page

Again

After 10 requests (again to Welcome page)



Count is 1

Welcome Page

Again

ALA PRACTICAL 7

AIM :JSP Scripting Elements

Do the following with only JSPs (and no servlets) and using JSP scripting elements

Create a method that take two int numbers and return their sum as int.

Invoke the method from within template text using JSP elements (example o/p - Sum of 3 and 4 is 7.) Print above line to console as well using JSP elements.

index.html

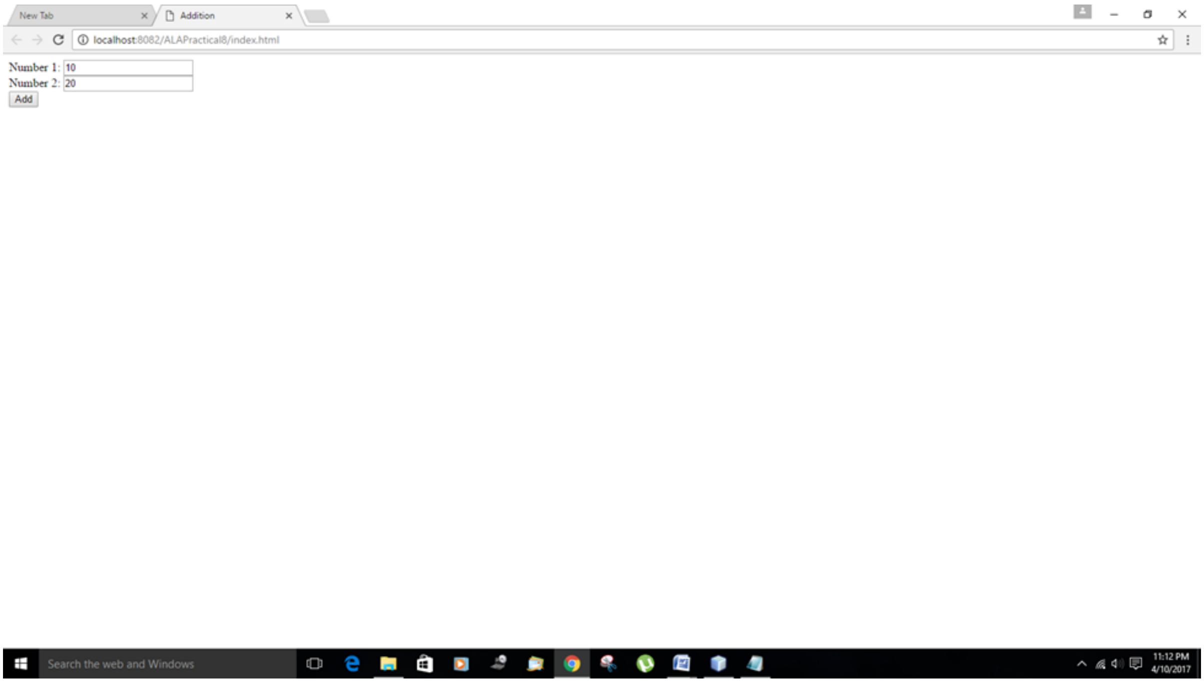
```
<html>

    <head>

        <title>Addition</title>

        <meta charset="UTF-8">

        <meta name="viewport" content="width=device-width, initial-scale=1.0">

    </head>

    <body>

        <form action="AddJSP.jsp" method="post">

            Number 1: <input type="text" name="n1"><br>

            Number 2: <input type="text" name="n2"><br>

            <input type="submit" value="Add">

        </form>

    </body>

</html>
```

AddJSP.jsp

```
<%--

    Document   : AddJSP

    Created on : Mar 24, 2017, 1:11:52 PM

    Author     : Himalay

--%>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Addition of 2 numbers</title>
    </head>
    <body>
   <%
            int x= Integer.parseInt(request.getParameter("n1"));
            int y=Integer.parseInt(request.getParameter("n2"));
        %>
       <%!
            int add(int a,int b)
            {
                return a+b;
            }
        %>
   <%= "Addition of " + x + " and " +  y  + " is " + add(x,y)%>
     </body>
</html>
```

ALA PRACTICAL 7 OUTPUT



New Tab
Addition
localhost:8082/ALAPractical8/index.html
Search the web and Windows
11:12 PM
4/10/2017

Number 1: 10
Number 2: 20
Add

ALA PRACTICAL 8

AIM : JSP Forms and Standard Actions

Create a JSP web form to submit the name and age of an employee.

Form should be submitted to a JSP file.

Create a bean class called Employee for storing the name and age of the employee

Within the JSP to which form is submitted, do the following using only standard actions-

Declare an instance of the Employee JavaBean

Populate the Employee object with values from form submission

Retrieve the values and display it with proper template text:

Employee name is John

Employee age is 30

index.jsp

```
<%--

   Document   : index

   Created on : Mar 24, 2017, 1:59:58 PM

   Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

   <head>

      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

   <title>Employee form</title>

   </head>

   <body>

      <form action="nextJSP.jsp">

         Name: <input type="text" name="name">

         Age: <input type="text" name="age">
```

```
            <input type="submit" value="Go">

        </form>

    </body>

</html>
```

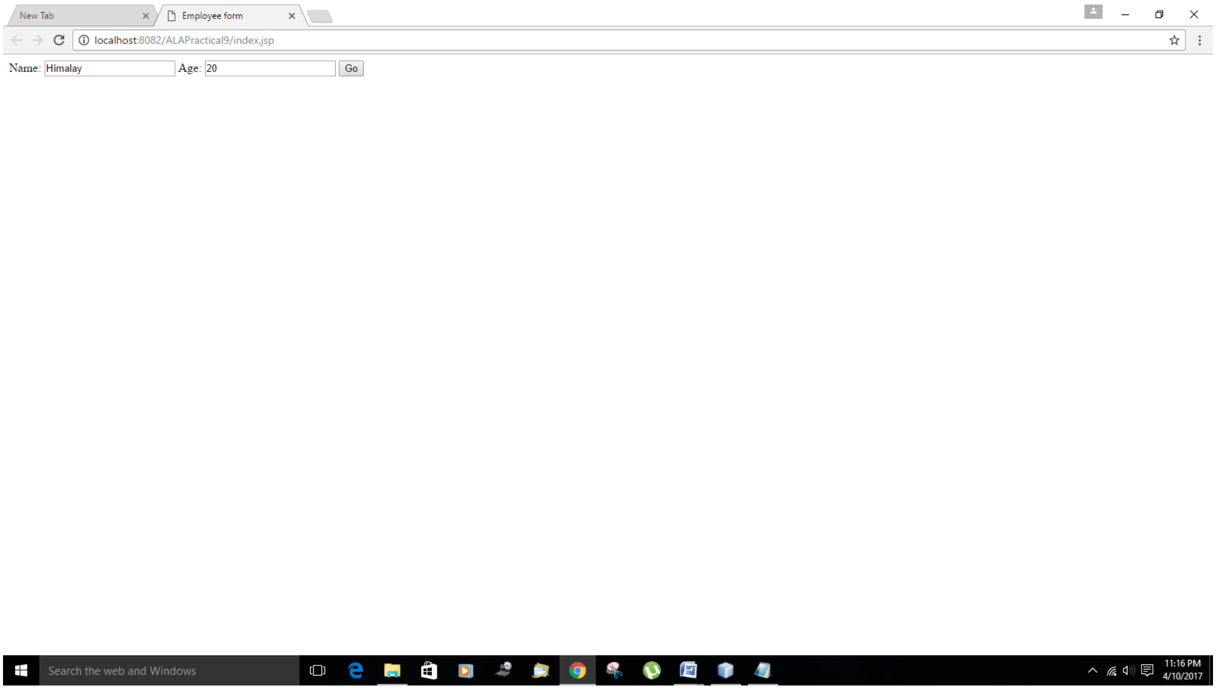
nextJSP.jsp

```
<%--

    Document   : nextJSP

    Created on : Mar 24, 2017, 2:01:48 PM

    Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

    <head>

        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

        <title>Welcome</title>

    </head>

    <body>

      <jsp:useBean id="emp" class="test.Employee" scope="request">

      <jsp:setProperty name="emp" property="*" />

      <jsp:getProperty property="name" name="emp"/><br>

      <jsp:getProperty property="age" name="emp"/><br>

      </jsp:useBean>

    </body>

</html>
```

test/Employee.java

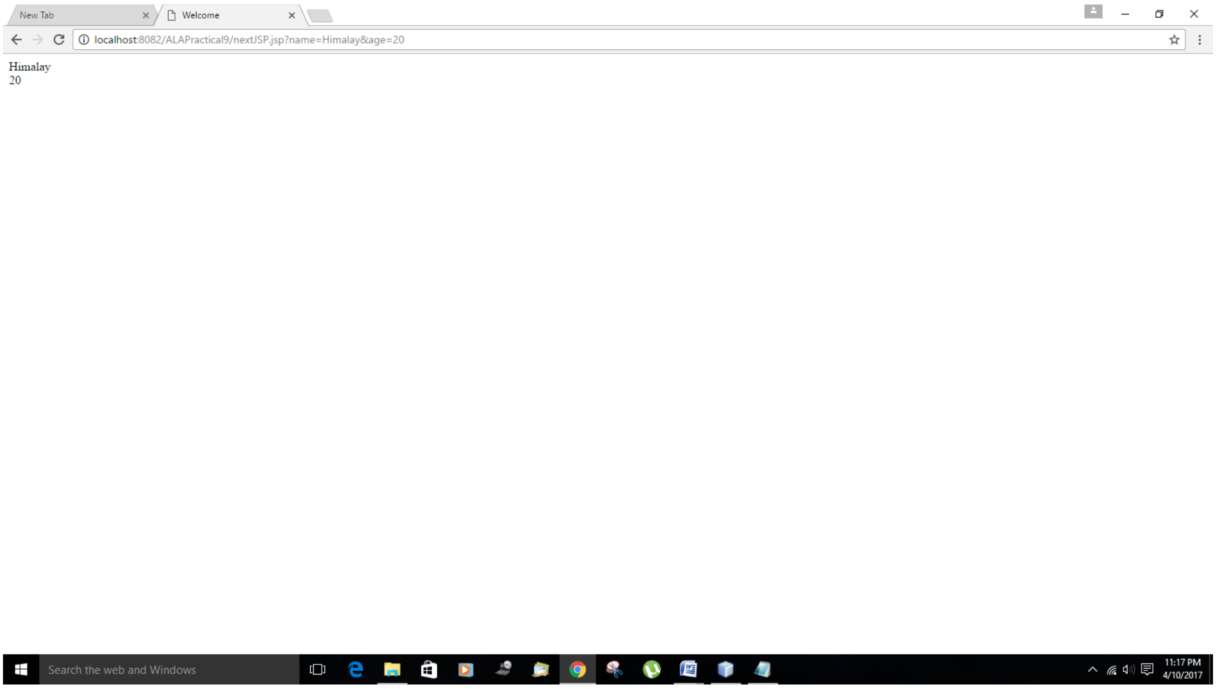
```
package test;

public class Employee {

public Employee(){}

private String name;

private int age;

public void setName(String str){

name=str;

}

public void setAge(int x){

age=x;

}

public String getName(){

   return name;

}

public int getAge(){

   return age;

}

}
```

ALA PRACTICAL 8 OUTPUT



localhost:8082/ALAPractical9/index.jsp

Name: Himalay Age: 20 Go



localhost:8082/ALAPractical9/nextJSP.jsp?name=Himalay&age=20

Himalay
20

ALA PRACTICAL 9

AIM : Create two simple JSP files.

A JSP web form that will submit data to the second JSP

Second JSP may simply print the values of the form submission

index.jsp

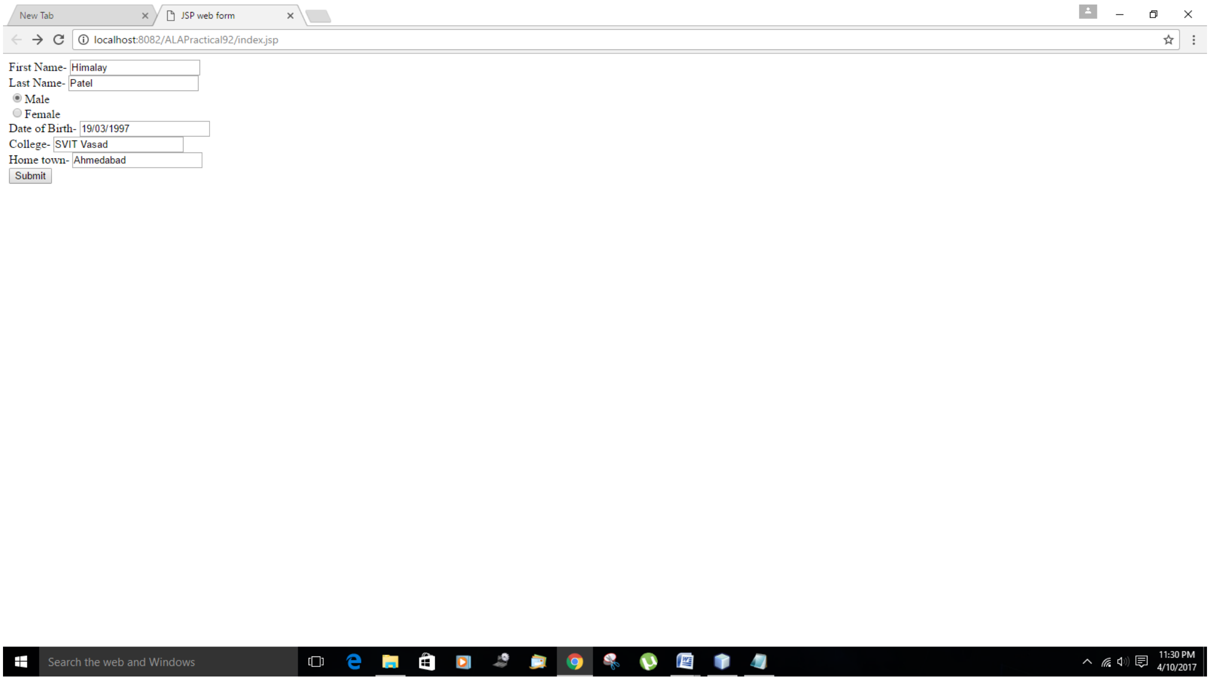
```
<%--

   Document   : index

   Created on : Apr 4, 2017, 11:19:23 AM

   Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP web form</title>

</head>

<body>

<form name="ff" method="post" action="show.jsp">

   First Name- <input type="text" name="fName" /><br>

   Last Name- <input type="text" name="lName" /><br>

   <input type="radio" name="gender" value="Male"/>Male<br>

   <input type="radio" name="gender" value="Female"/>Female<br>

   Date of Birth- <input type="text" name="dob" /><br>

   College- <input type="text" name="college" /><br>

   Home town- <input type="text" name="home" /><br>

   <input type="submit" name="submit" value="Submit" />
```

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

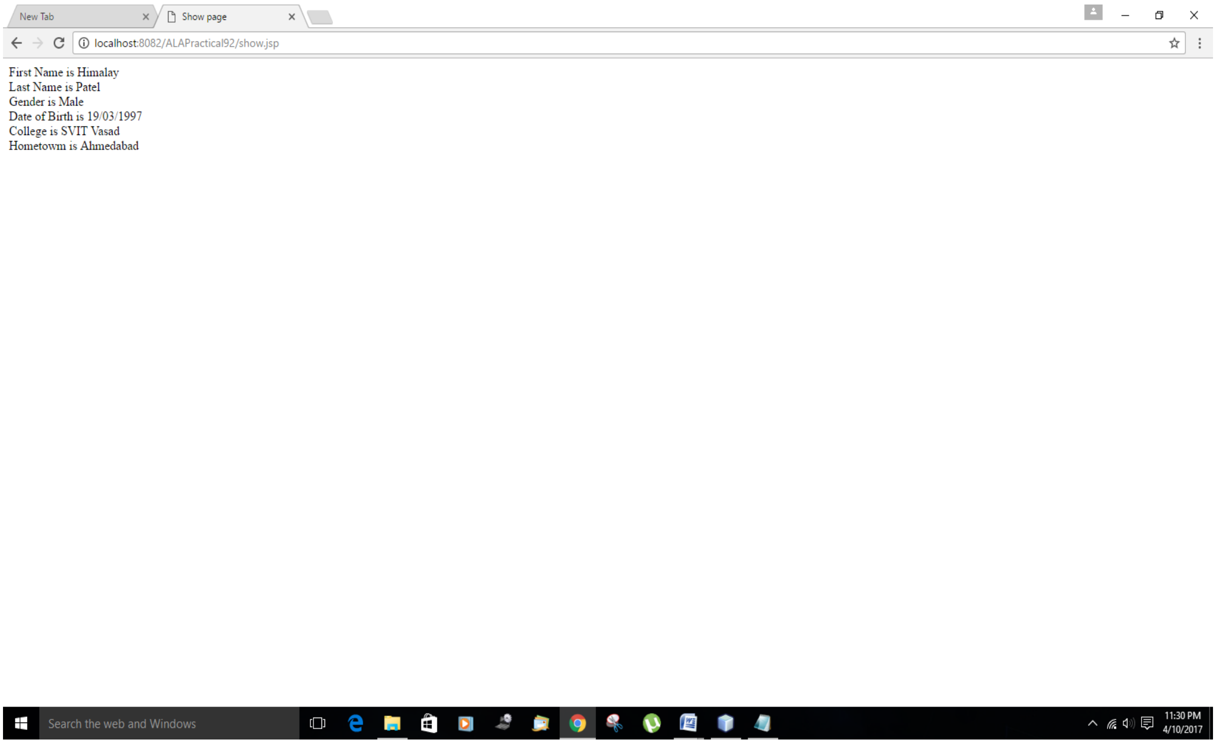
show.jsp

```
<%--
   Document   : show
   Created on : Apr 4, 2017, 11:24:09 AM
   Author     : Himalay
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Show page</title>
</head>
<body>
   <%
   String fname=request.getParameter("fName");
   String lname=request.getParameter("lName");
   String gender=request.getParameter("gender");
   String dob=request.getParameter("dob");
   String college=request.getParameter("college");
   String home=request.getParameter("home");
   %>
<%= "First Name is " + fname + "<br>" %>
<%= "Last Name is " + lname + "<br>" %>
<%= "Gender is " + gender + "<br>" %>

<%= "Date of Birth is " + dob + "<br>" %>

<%= "College is " + college + "<br>" %>

<%= "Hometowm is " + home + "<br>" %>

</body>

</html>
```

ALA PRACTICAL 9 OUTPUT


New Tab
JSP web form
localhost:8082/ALAPractical92/index.jsp
Search the web and Windows
11:30 PM
4/10/2017

First Name- Himalay
Last Name- Patel
● Male
○ Female
Date of Birth- 19/03/1997
College- SVIT Vasad
Home town- Ahmedabad
Submit


New Tab
Show page
localhost:8082/ALAPractical92/show.jsp
Search the web and Windows
11:30 PM
4/10/2017

```
First Name is Himalay
Last Name is Patel
Gender is Male
Date of Birth is 19/03/1997
College is SVIT Vasad
Hometowm is Ahmedabad
```

ALA PRACTICAL 10 (Use JSTL SQL taglib)

AIM : Write a JSP which insert the details of a customer who register with the web site by using registration form.

insert.html

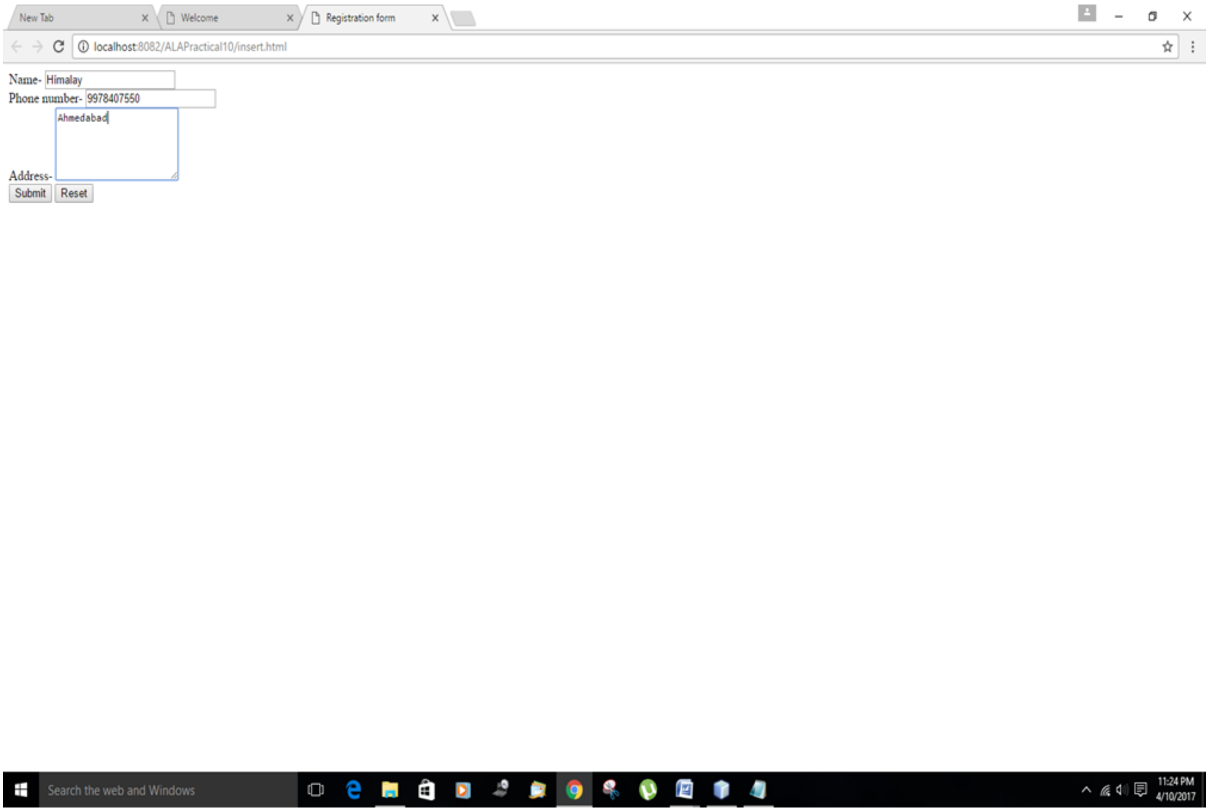
```
<!DOCTYPE html>

<html>

<head>

<title>Registration form</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<form name="f1" method="post" action="insert.jsp">

    Name- <input type="text" name="name" /><br>

    Phone number- <input type="text" name="phone" /><br>

    Address- <textarea rows="5" name="address"></textarea><br>

    <input type="submit" name="submit" />

    <input type="reset" name="reset" />

</form>

</body>

</html>
```

insert.jsp

```
<%--

    Document   : insert

    Created on : Apr 4, 2017, 8:06:37 AM

    Author     : Himalay

--%>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix='c'%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix='sql'%>

<sql:setDataSource var="dataSource" driver="com.mysql.jdbc.Driver"

url="jdbc:mysql://localhost:3306/himalay" user="root" password="himalaypatel" />

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Insert details</title>

</head>

<body>

<%

        String name= request.getParameter("name");

        String phone= request.getParameter("phone");

        String address= request.getParameter("address");

%>

<sql:update dataSource="${dataSource}" var="result">

insert into customer(name,phone,address) values (?,?,?);

<sql:param value="${param.name}" />

<sql:param value="${param.phone}" />

<sql:param value="${param.address}" />

</sql:update>

<h2>Data inserted successfully.</h2>

</body>

</html>
```

ALA PRACTICAL 10 OUTPUT

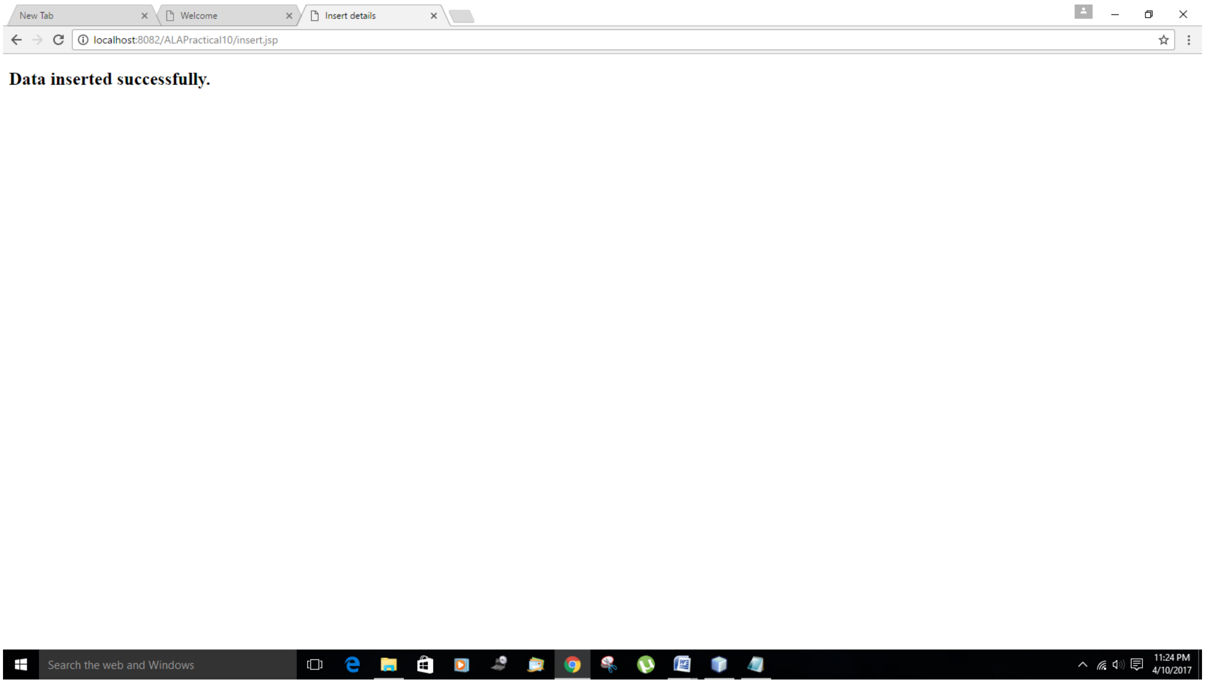


localhost:8082/ALAPractical10/insert.html

Name- Himalay

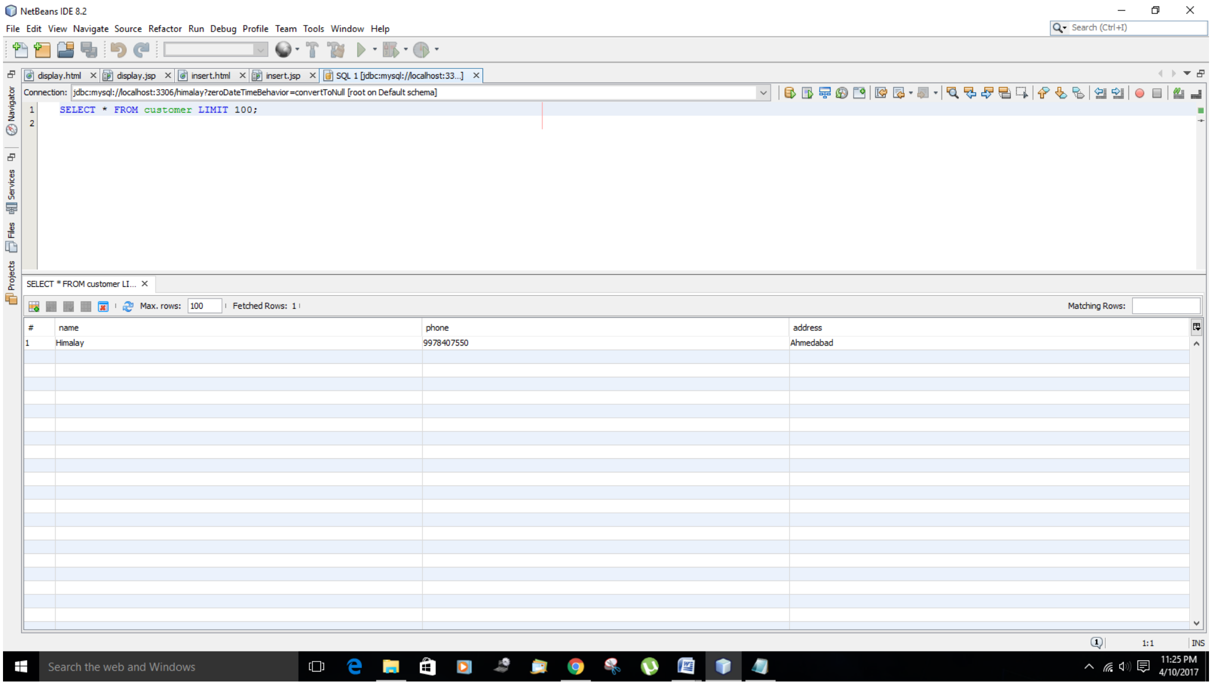
Phone number- 9978407550

Address- Ahmedabad



localhost:8082/ALAPractical10/insert.jsp

Data inserted successfully.



NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
display.html
display.jsp
insert.html
insert.jsp
SQL 1 [jdbc:mysql://localhost:33...]
Navigator
Services
Files
Projects
Search the web and Windows
11:25 PM
4/10/2017

Connection: jdbc:mysql://localhost:3306/himalay?zeroDateTimeBehavior=convertToNull [root on Default schema]

```
SELECT * FROM customer LIMIT 100;
```

SELECT * FROM customer LI...

Max. rows: 100 Fetched Rows: 1 Matching Rows:

#	name	phone	address
1	Himalay	9978407550	Ahmedabad

ALA PRACTICAL 11 (Use JSTL SQL taglib)

AIM :Write a JSP which displays customer details in tabular form by iterating through the database customer table.

display.html

```
<html>

<head>

<title>Display details</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<form name="f2" method="post" action="display.jsp">

<input type="submit" name="submit" value="Display details of customer table" />

</form>

</body>

</html>
```

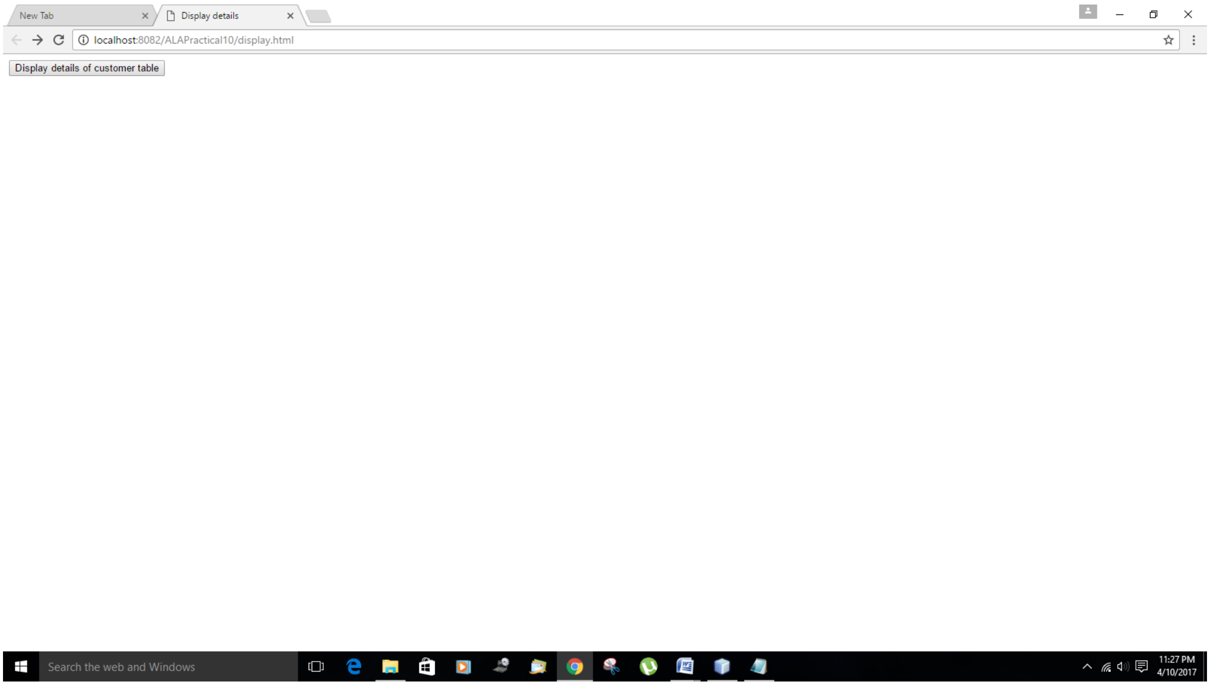
display.jsp

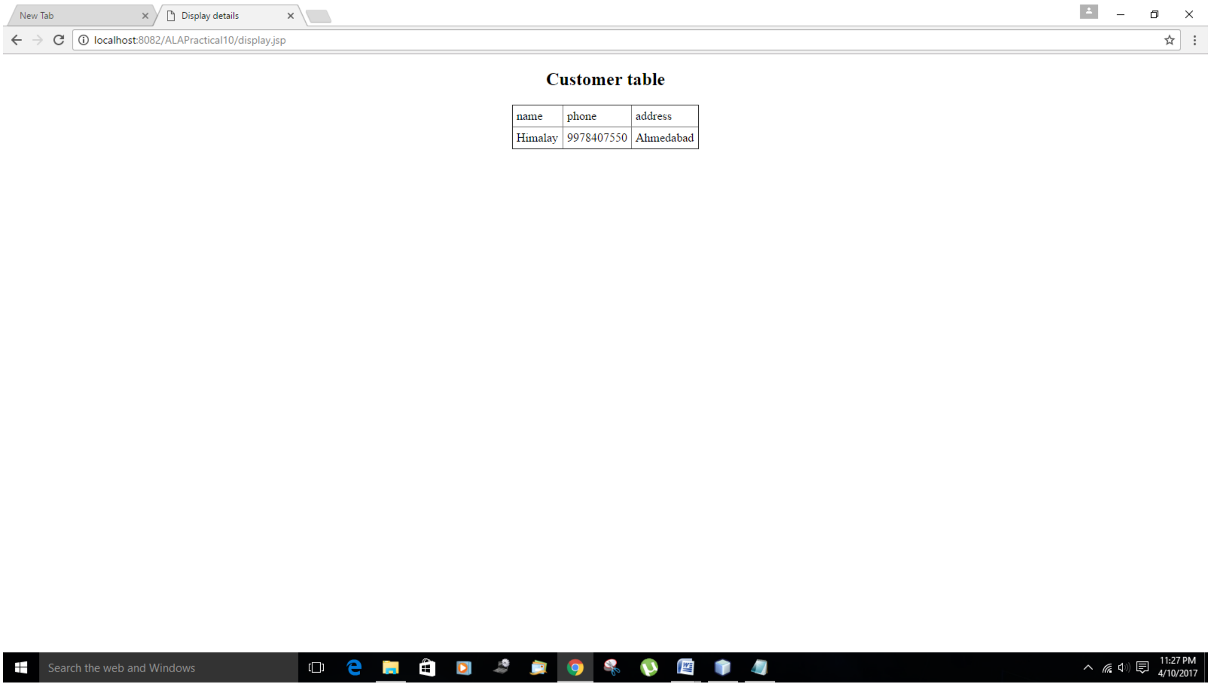
```
<%--

    Document   : display

    Created on : Apr 4, 2017, 8:38:17 AM

    Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix='c'%>

<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix='sql'%>

<sql:setDataSource var="dataSource" driver="com.mysql.jdbc.Driver"

url="jdbc:mysql://localhost:3306/himalay" user="root" password="himalaypatel" />
```

```
<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Display details</title>

<style type="text/css">

table{

border-collapse:collapse;

}

</style>

</head>

<body>

<sql:query var="customers" dataSource="${dataSource}">

select * from customer;

</sql:query>

<center><h2>Customer table</h2></center>

<table border="1" cellspacing="3" cellpadding="5" align="center">

<tr>

<c:forEach var="column" items="${customers.columnNames}" >

<td><c:out value="${column}" /></td>

</c:forEach>

</tr>

<c:forEach var="row" items="${customers.rows}">

<tr>

<td><c:out value="${row.name}" /></td>

<td><c:out value="${row.phone}" /></td>

<td><c:out value="${row.address}" /></td>
```

```
</tr>

</c:forEach>

</table>

</body>

</html>
```

ALA PRACTICAL 11 OUTPUT


New Tab
Display details
localhost:8082/ALAPractical10/display.html
Display details of customer table
Search the web and Windows
11:27 PM
4/10/2017


New Tab
Display details
localhost:8082/ALAPractical10/display.jsp
Search the web and Windows
11:27 PM
4/10/2017

Customer table

name	phone	address
Himalay	9978407550	Ahmedabad

ALA PRACTICAL 12

AIM : Create a servlet that makes an array of Name objects, which have firstName and lastName properties. Have your JSP page make an HTML table with first names in the left table cell and last names in the right table cell. Use the JSP 2.0 expression language as well as JSTL.

index.html

```
<html>

<head>

<title>Name objects</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<form name="f" method="post" action="Servlet1">

<button type="submit">Display Names in tabular form</button>

</form>

</body>

</html>
```

test/Servlet1.java

```
package test;

import test.Name;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.*;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

public class Servlet1 extends HttpServlet {

   @Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doPost(request,response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

Name[] n;

n = new Name[3];

for(int i=0;i<3;i++){

Name m=new Name();

n[i]=m;

}

n[0].setFirstName("Himalay");

n[0].setLastName("Patel");

n[1].setFirstName("Parva");

n[1].setLastName("Patel");

n[2].setFirstName("Kaival");

n[2].setLastName("Kothari");

request.setAttribute("names",n);

RequestDispatcher rd= request.getRequestDispatcher("go.jsp");

rd.forward(request,response);
}

}
```

test/Name.java

```
package test;

public class Name {

private String firstName;

private String lastName;

public Name(){ }

public void setFirstName(String fname){

firstName=fname;

}

public void setLastName(String lname){

lastName=lname;

}

public String getFirstName(){

return firstName;

}

public String getLastName(){

return lastName;

}

}
```

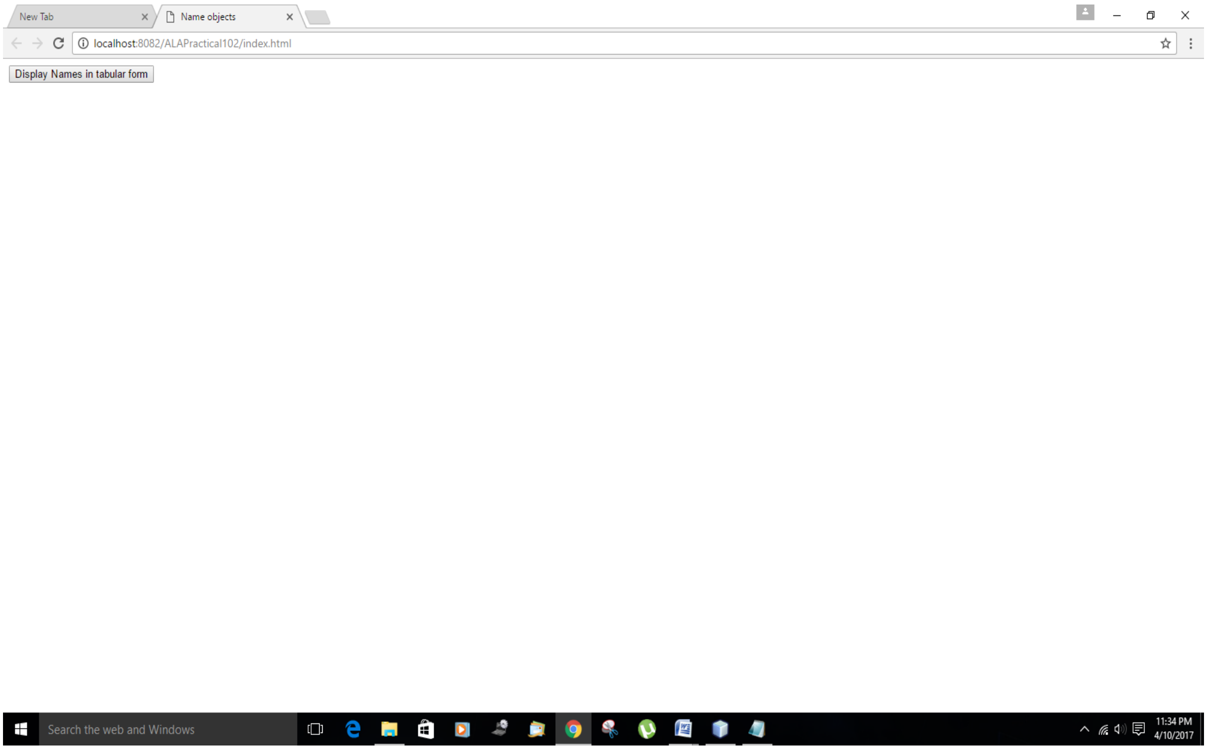
go.jsp

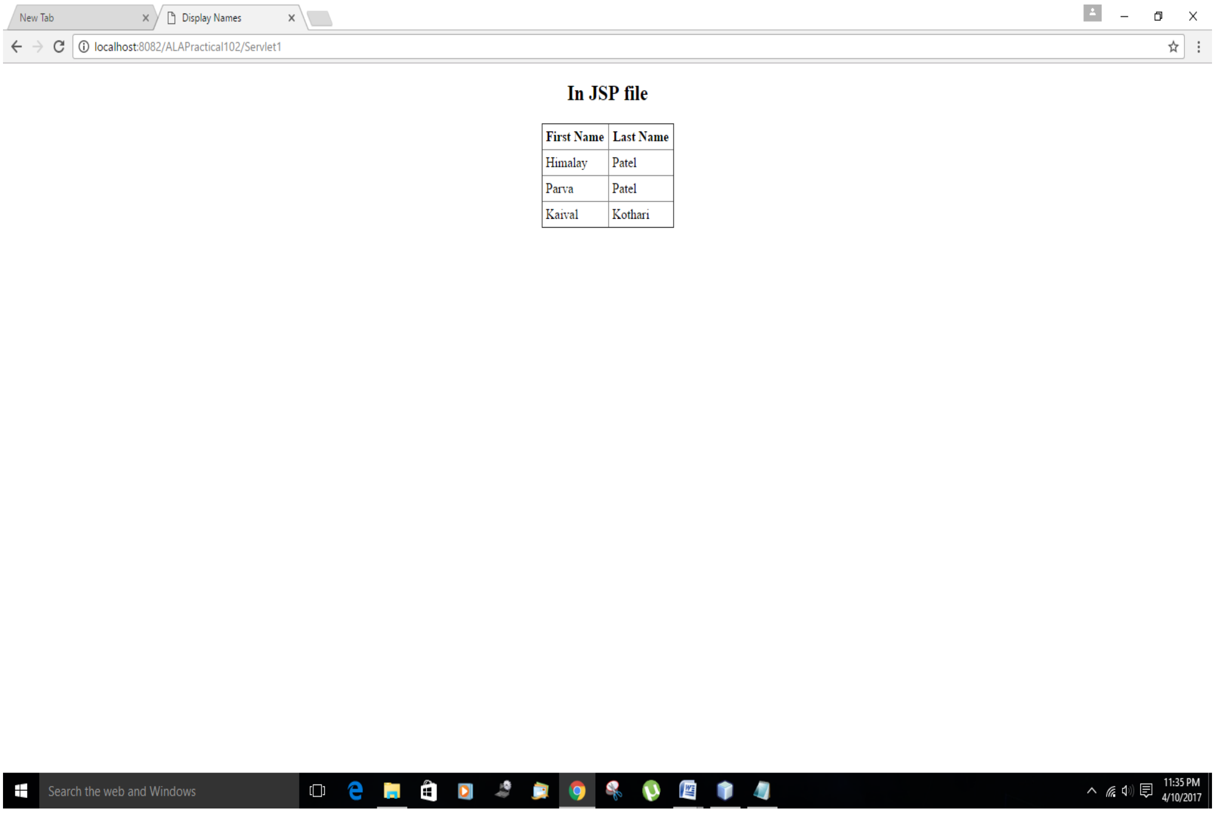
```
<%--

    Document   : go

    Created on : Apr 4, 2017, 11:55:28 AM

    Author     : Himalay
```

```
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix='c'%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Display Names</title>
<style type="text/css">
table{
border-collapse:collapse;
}
</style>
</head>
<body>
<center><h2>In JSP file</h2></center>
<center>
  <table border="1" cellspacing="3" cellpadding="5">
  <tr>
  <th>First Name</th>
  <th>Last Name</th>
  </tr>
  <c:forEach var="n" items="${names}">
  <tr>
  <td><c:out value="${n.firstName}" /></td>
  <td><c:out value="${n.lastName}" /></td>
  </tr>
```

```
    </c:forEach>

    </table>

</center>

</body>

</html>
```

ALA PRACTICAL 12 OUTPUT


New Tab
Name objects
localhost:8082/ALAPractical102/index.html
Display Names in tabular form
Search the web and Windows
11:34 PM
4/10/2017


New Tab
Display Names
localhost:8082/ALAPractical102/Servlet1
Search the web and Windows
11:35 PM
4/10/2017

In JSP file

First Name	Last Name
Himalay	Patel
Parva	Patel
Kaival	Kothari

ALA PRACTICAL 13

AIM : Using JSTL XML taglib

Write a JSP which parses an XML file containing Customer details like Name, Age, phoneno, email.. Displays the result in tabular form.

student.xml

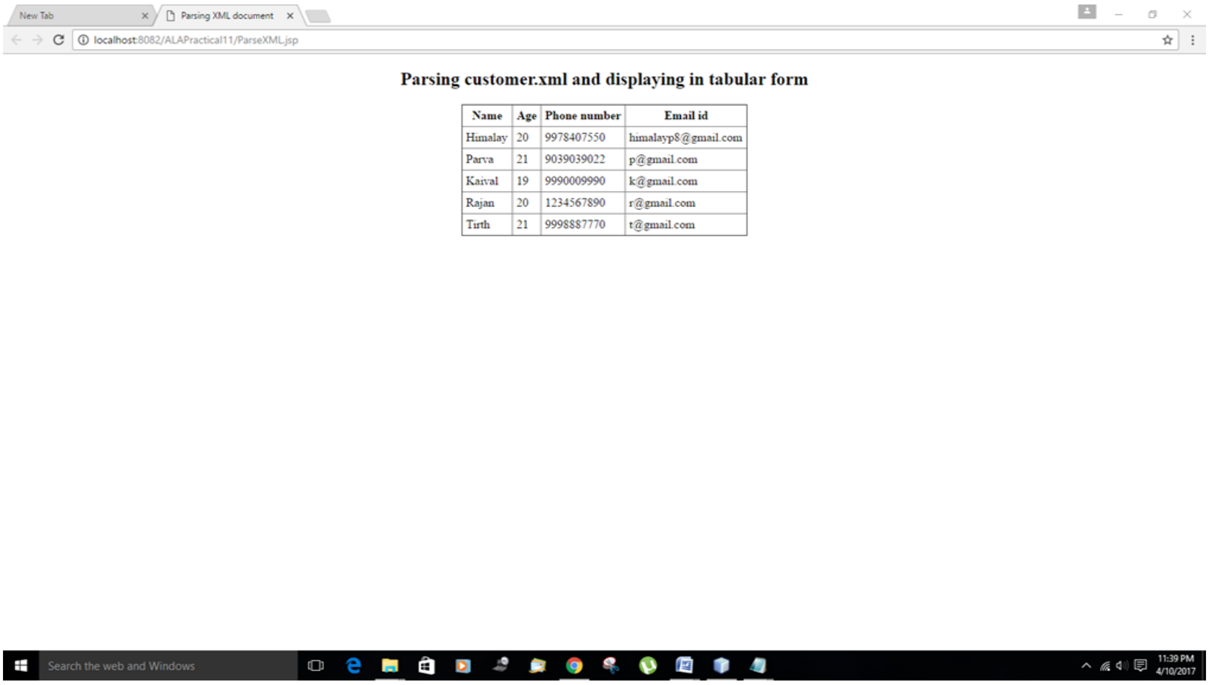
```
<?xml version="1.0" encoding="UTF-8"?>

<customers>

<customer>

   <name>Himalay</name>

   <age>20</age>

   <phoneno>9978407550</phoneno>

   <email>himalayp8@gmail.com</email>

</customer>

<customer>

   <name>Parva</name>

   <age>21</age>

   <phoneno>9039039022</phoneno>

   <email>p@gmail.com</email>

</customer>

<customer>

   <name>Kaival</name>

   <age>19</age>

   <phoneno>9990009990</phoneno>

   <email>k@gmail.com</email>

</customer>

<customer>

   <name>Rajan</name>

   <age>20</age>
```

```
    <phoneno>1234567890</phoneno>

    <email>r@gmail.com</email>

</customer>

<customer>

    <name>Tirth</name>

    <age>21</age>

    <phoneno>9998887770</phoneno>

    <email>t@gmail.com</email>

</customer>

</customers>
```

ParseXML.jsp

```
<%--

    Document   : ParseXML

    Created on : Apr 4, 2017, 4:37:27 PM

    Author     : Himalay

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix='c'%>

<%@taglib uri="http://java.sun.com/jsp/jstl/xml" prefix='x'%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Parsing XML document</title>

<style type="text/css">

table{

border-collapse:collapse;
}
</style>
</head>
<body>
<c:import var="customerInfo" url="http://localhost:7777/practicals/customer.xml"/>
<x:parse xml="${customerInfo}" var="output"/>
<center><h2>Parsing customer.xml and displaying in tabular form</h2></center>
<center><table border="1" cellspacing="3" cellpadding="5">
<tr>
<th>Name</th>
<th>Age</th>
<th>Phone number</th>
<th>Email id</th>
</tr>
<x:forEach select="$output/customers/customer" var="xx">
<tr>
<td><x:out select="$xx/name" /></td>
<td><x:out select="$xx/age" /></td>
<td><x:out select="$xx/phoneno" /></td>
<td><x:out select="$xx/email" /></td>
</tr>
</x:forEach>
</table></center>
</body>
</html>
```

ALA PRACTICAL 13 OUTPUT



New Tab
Parsing XML document
localhost:8082/ALAPractical11/ParseXML.jsp
Search the web and Windows
11:39 PM
4/10/2017

Parsing customer.xml and displaying in tabular form

Name	Age	Phone number	Email id
Himalay	20	9978407550	himalayp8@gmail.com
Parva	21	9039039022	p@gmail.com
Kaival	19	9990009990	k@gmail.com
Rajan	20	1234567890	r@gmail.com
Tirth	21	9998887770	t@gmail.com

ALA PRACTICAL 14

AIM : Make a JSF page which takes input for two numbers and has command buttons for different operation. When any button is clicked that operation is performed and result will be displayed in a text box. Should show an error message if any number is missing.

index.xhtml

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

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"

xmlns:h="http://xmlns.jcp.org/jsf/html">

<h:head>

<title>Calculator using JSF</title>

</h:head>

<h:body>

<h2><h:outputText value="Simple Calculator using JSF" /></h2>

<h:form id="f1">

<h:outputText value="Enter a-"/>

<h:inputText value="#{test.a}" /><br></br>

<h:outputText value="Enter b-"/>

<h:inputText value="#{test.b}" /><br></br>

<h:commandButton action="#{test.add}" value="Add" /><br></br>

<h:commandButton action="#{test.subtract}" value="Subtract" /><br></br>

<h:commandButton action="#{test.multiply}" value="Multiply" /><br></br>

<h:commandButton action="#{test.divide}" value="Divide" /><br></br>

<h:outputText value="Answer is-"/>

<h:inputText value="#{test.result}" /><br></br>

<h:outputText value="#{test.error}"/><br></br>
```

```
</h:form>

</h:body>

</html>
```

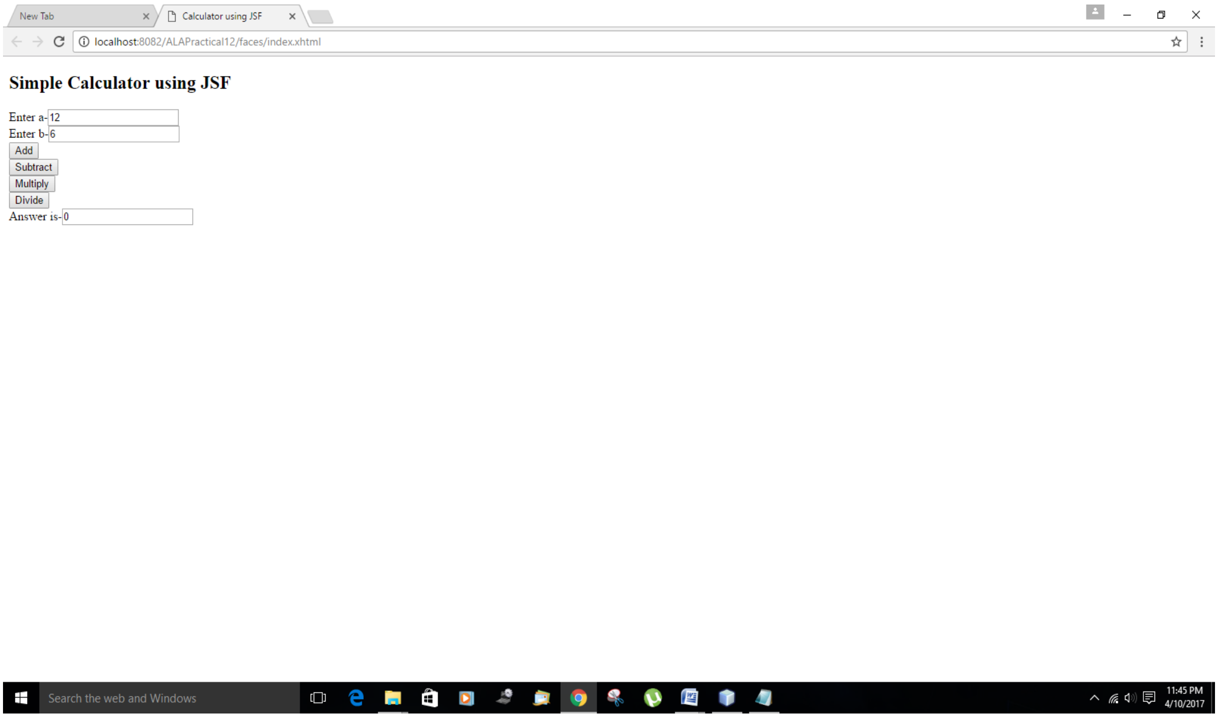
bean/test.java

```
package bean;

import javax.faces.bean.ManagedBean;

import javax.faces.bean.RequestScoped;

@ManagedBean

@RequestScoped

public class test {

private String a;

private String b;

private int result;

private String error;

public test(){ }

public String getA(){

return a;

}

public void setA(String str){

a=str;

}

public String getB(){

return b;

}

public void setB(String str){

b=str;

}
```

```
public int getResult(){

return result;

}

public void setResult(int str){

result=str;

}

public String getError(){

return error;

}

public void setError(String str){

error=str;

}

public void add(){

if(a.equals("") || b.equals("")){

error="One of the two numbers is missing.";

result=-9999999;

return;

}

result=Integer.parseInt(a)+Integer.parseInt(b);

}

public void subtract(){

if(a.equals("") || b.equals("")){

error="One of the two numbers is missing.";

result=-9999999;

return;

}

result=Integer.parseInt(a)-Integer.parseInt(b);
}

public void multiply(){

if(a.equals("") || b.equals("")){

error="One of the two numbers is missing.";

result=-9999999;

return;

}

result=Integer.parseInt(a)*Integer.parseInt(b);

}

public void divide(){

if(a.equals("") || b.equals("")){

error="One of the two numbers is missing.";

result=-9999999;

return;

}

result=Integer.parseInt(a)/Integer.parseInt(b);

}

}
```

ALA PRACTICAL 14 OUTPUT

Initially,

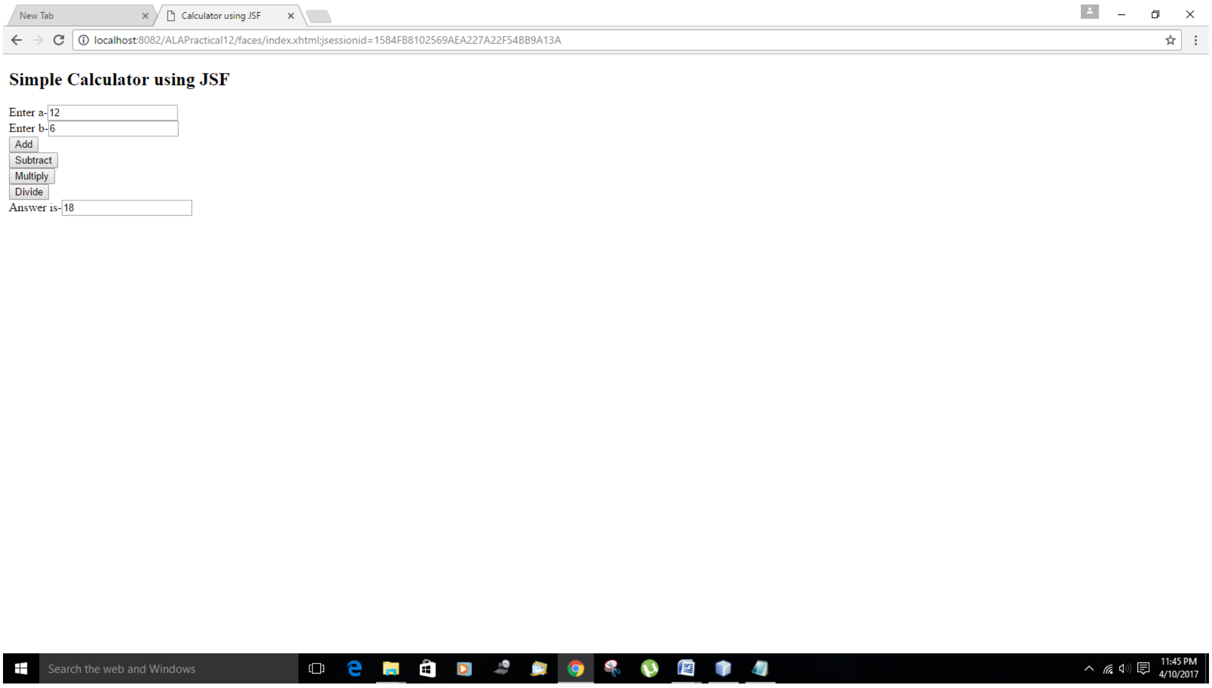


New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml
Search the web and Windows
11:45 PM
4/10/2017

Simple Calculator using JSF

Enter a-12
Enter b-6
Add
Subtract
Multiply
Divide
Answer is-0

Add,

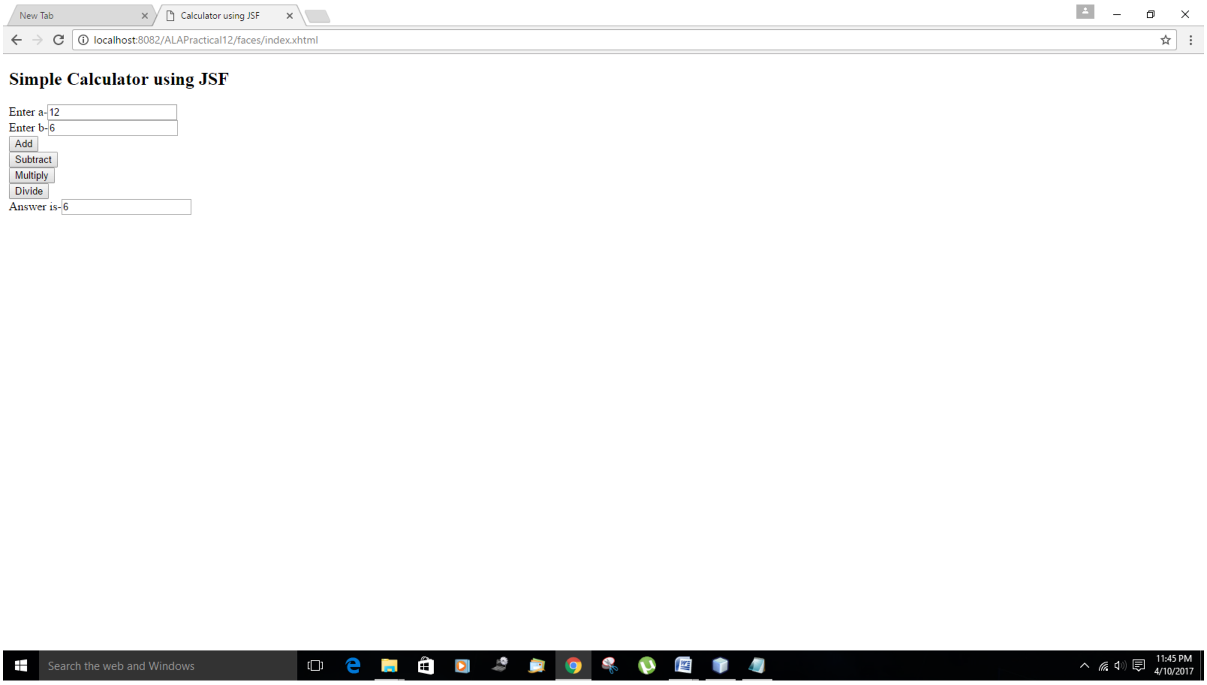


New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml;jsessionid=1584FB8102569AEA227A22F54BB9A13A
Search the web and Windows
11:45 PM
4/10/2017

Simple Calculator using JSF

Enter a-12
Enter b-6
Add
Subtract
Multiply
Divide
Answer is-18

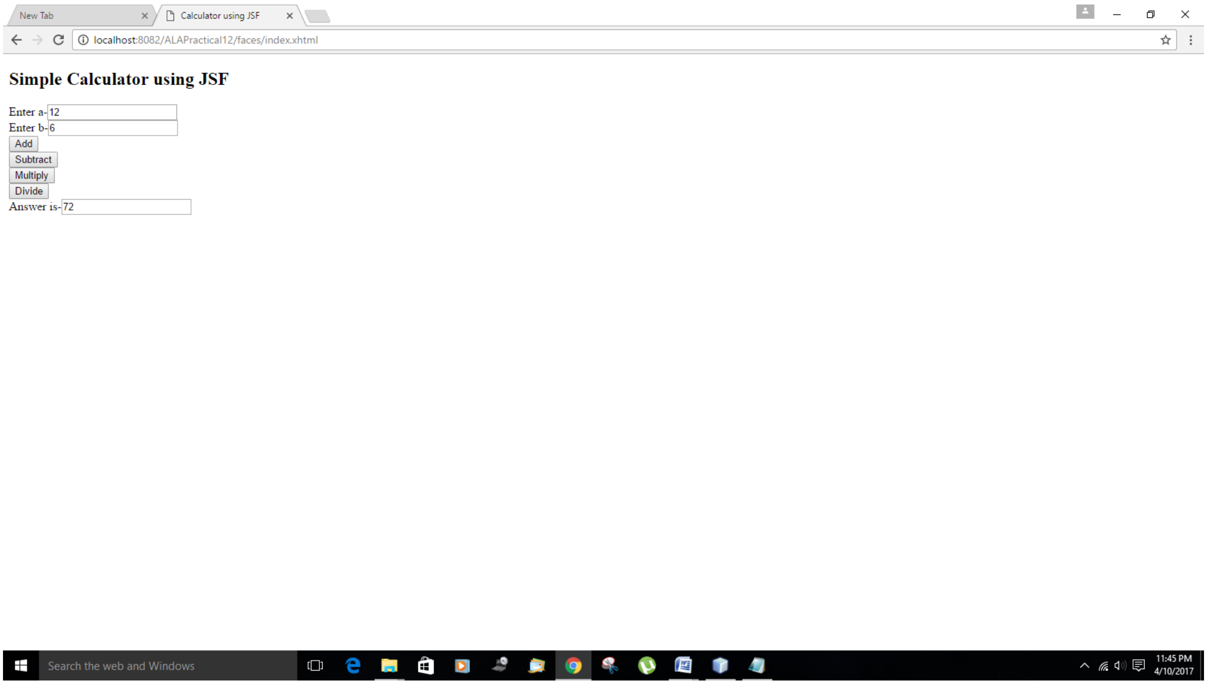
Subtract,


New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml
Search the web and Windows
11:45 PM
4/10/2017

Simple Calculator using JSF

Enter a- 12
Enter b- 6
Add
Subtract
Multiply
Divide
Answer is- 6

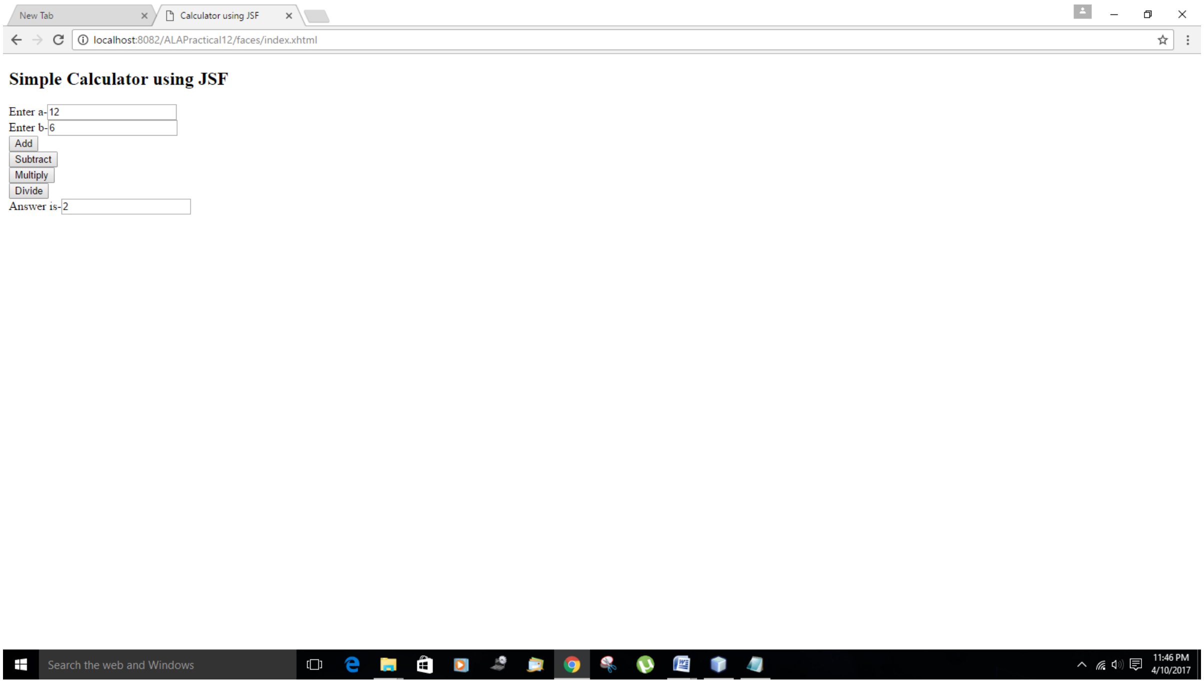
Multiply,


New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml
Search the web and Windows
11:45 PM
4/10/2017

Simple Calculator using JSF

Enter a- 12
Enter b- 6
Add
Subtract
Multiply
Divide
Answer is- 72

Divide,

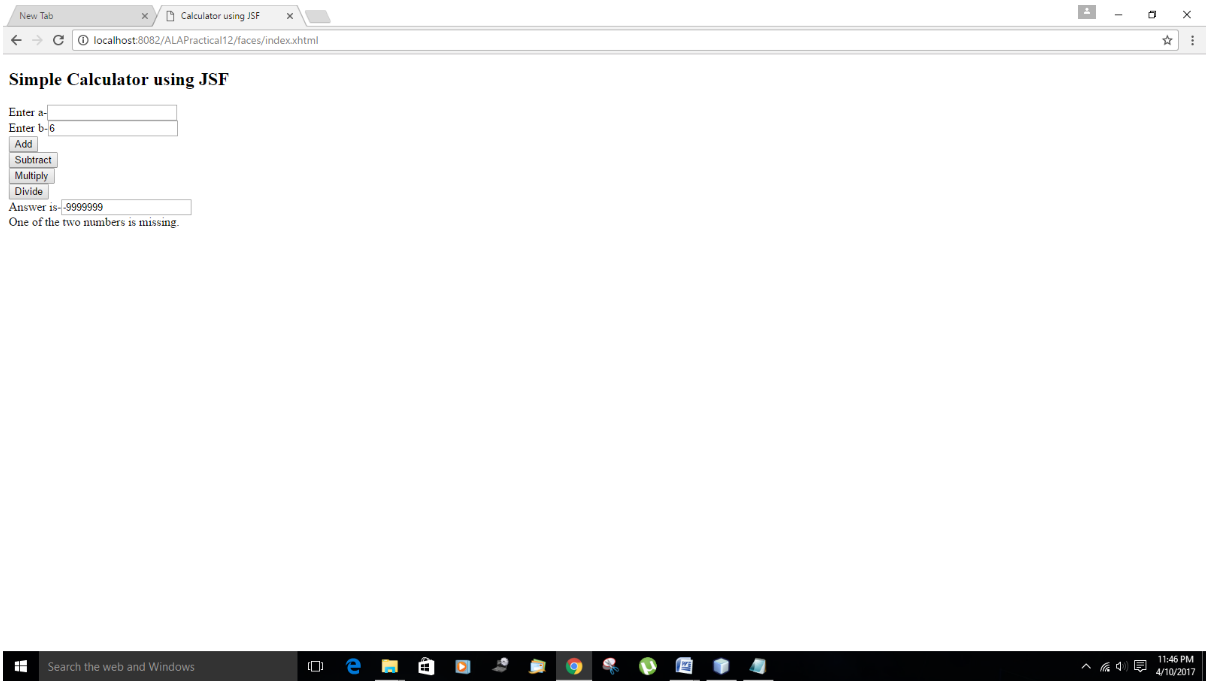


New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml
Search the web and Windows
11:46 PM
4/10/2017

Simple Calculator using JSF

Enter a- 12
Enter b- 6
Add
Subtract
Multiply
Divide
Answer is- 2

Number missing,



New Tab
Calculator using JSF
localhost:8082/ALAPractical12/faces/index.xhtml
Search the web and Windows
11:46 PM
4/10/2017

Simple Calculator using JSF

Enter a-
Enter b- 6
Add
Subtract
Multiply
Divide
Answer is- -9999999
One of the two numbers is missing.