

ADVANCE JAVA LAB FILE
(BTCSE 606)

LAB REPORT

Submitted by

HUSSAIN QUDSIA EJAZ

ENROLL. NO. 2016-310-029

in partial fulfilment for the award of the degree of

B.Tech (Computer Science)

Under the guidance of

MR. SHAH IMRAN ALAM



**Department of Computer Science & Engineering
School of Engineering Sciences & Technology
JAMIA HAMDARD**

New Delhi-110062

SL. NO.	TOPIC	DATE	PgNo.	SIGN
1.	Write a Dynamic Web Project to greet user.	21.1.19		
2.	Write a Dynamic Web Project to greet user by the username entered.	22.1.19		
3.	Write a Dynamic Web Project to display the interests selected by the user using checkboxes.	28.1.19		
4.	Write a Dynamic Web Project to implement visit counter using inti method, config object and context object.	4.2.19		
5.	Write a Dynamic Web Project to implement persistent visit counter using text file.	11.2.19		
6.	Write a Dynamic Web Project to implement session management using session variables.	18.2.19		
7.	Write a Dynamic Web Project to set cookies and send them to client via response header.	25.2.19		
8.	Write a Dynamic Web Project to read the cookies sent to the client.	26.2.19		
9.	Write a Dynamic Web Application using JSP to implement: a)Directives b)Scriptlet c) Expression d)Declaration	4.3.19		
10.	Write a Dynamic Web Application using JSP to implement MVC without DataBase	11.3.19		
11.	Write a Dynamic Web Application using JSP to implement MVC with DataBase	12.3.19		
12.	Write a java program to implement 5 simple steps to connect a database	18.3.19		
13.	Write a java program to connect to a database which handles exceptions.	19.3.19		
14.	Write a java program to connect to a database to execute Insert, Update and Delete Queries using Statement	25.3.19		
15.	Write a java program to connect to a database and execute Insert Prepared Statement	26.3.19		
16.	Write a java program to connect to database and execute Delete Prepared Statement	2.4.19		
17.	Write a java program to connect to database and execute Update Prepared Statement	8.4.19		
18.	Write java program to connect to database to implement Transaction	9.4.19		
19.	Write a java program to connect to database to implement Batch Update	15.4.19		
20.	Write a java program to implement Remote Method Invocation.	16.4.19		

1. Write a Dynamic Web Project to greet user.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Experiment 1</title>
</head>
<body>
<h1>Please Enter Your Name</h1>
<form action = 'StaticServlet'>
<br>Name:<input type = 'text' value = 'Name'>
<input type = 'submit' value = 'send'>
</form>

</body>
</html>
```

SERVLET Code:

```
package com.qh.pkg;

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

/**
 * Servlet implementation class FirstStaticServlet
 */
public class FirstStaticServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     *

```

```

    * Default constructor.
    */
public FirstStaticServlet() {
    // TODO Auto-generated constructor stub
}

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    response.getWriter().append("Welcome to my First Dynamic Page");
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}
}

```

XML Code:

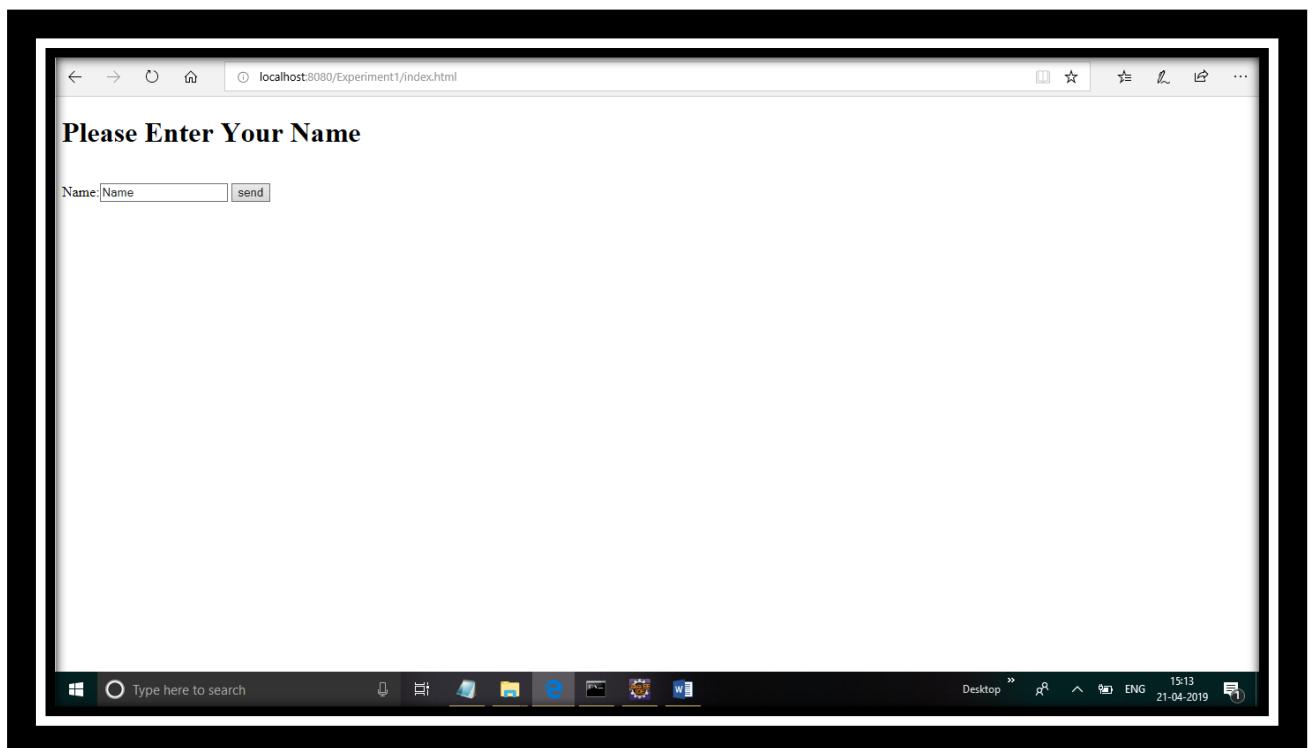
```

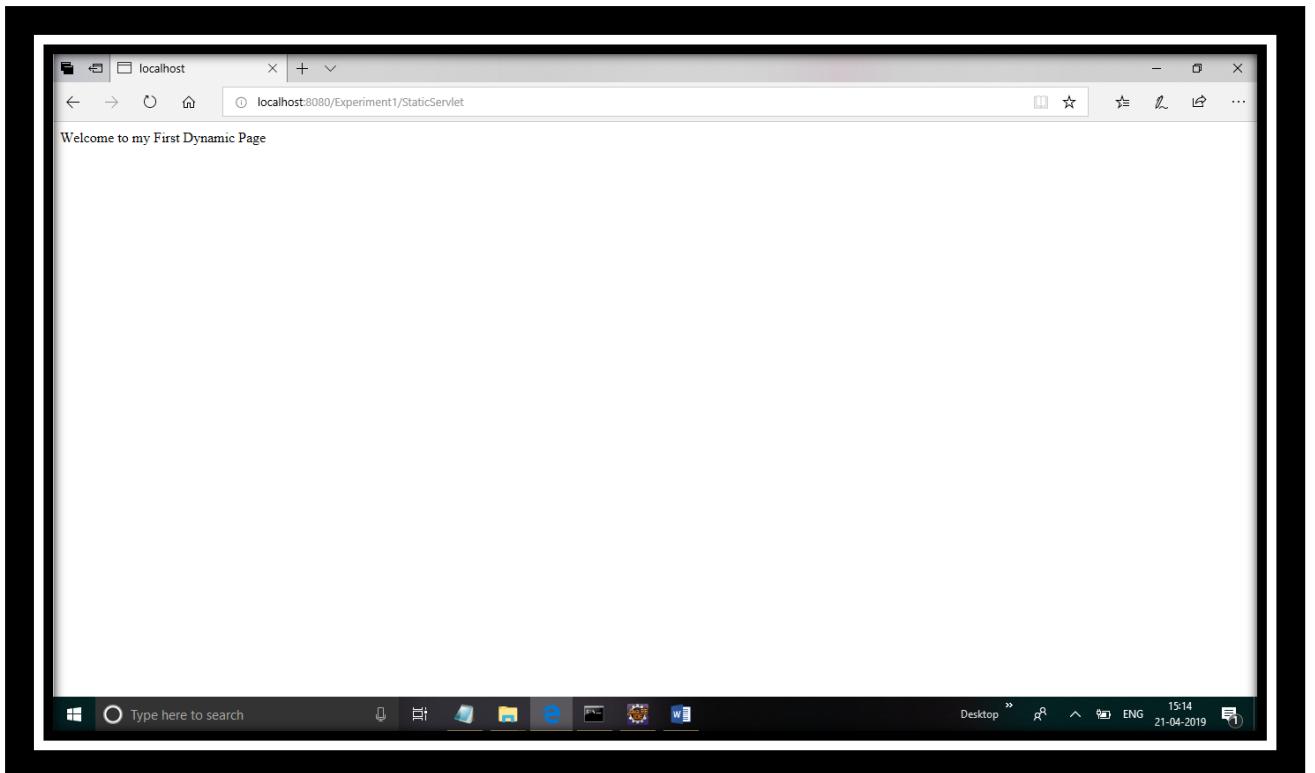
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">

```

```
<display-name>Experiment1</display-name>
<welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
</welcome-file-list>
<servlet>
    <description></description>
    <display-name>FirstStaticServlet</display-name>
    <servlet-name>FirstStaticServlet</servlet-name>
    <servlet-class>com.qh.pkg.FirstStaticServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>FirstStaticServlet</servlet-name>
    <url-pattern>/StaticServlet</url-pattern>
</servlet-mapping>
</web-app>
```

OUTPUT:





2. Write a Dynamic Web Project to greet user by the username entered.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Experiment 2</title>
</head>
<body>
<h1>Welcome to Dynamic Page</h1>
<form action="HelloServlet">
    <br>
    Name:<input type="text" name="user_name" size="20px">
    <br>
    <br>
    <input type="submit" value="submit">
</form>

</body>
</html>
```

SERVLET Code:

```
package com.qh2.pkg;

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

/**
 * Servlet implementation class SecondServlet
 */
public class SecondServlet extends HttpServlet {

```

```

private static final long serialVersionUID = 1L;

/**
 * Default constructor.
 */
public SecondServlet() {
    // TODO Auto-generated constructor stub
}

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    {
        // reading the user input
        String color= request.getParameter("user_name");
        PrintWriter out = response.getWriter();
        out.println (
            "<!DOCTYPE html>\n" +
            "<html> \n" +
            "  <head> \n" +
            "    <title>Welcome to my Dynamic Web Page </title> \n" +
            "  </head> \n" +
            "  <body> \n" +
            "    <font size=\"12px\">" + "Welcome to my page: " + color +
            "    </font> \n" +
            "  </body> \n" +
            "</html>"
    );
}

```

```

    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
    */

    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}

```

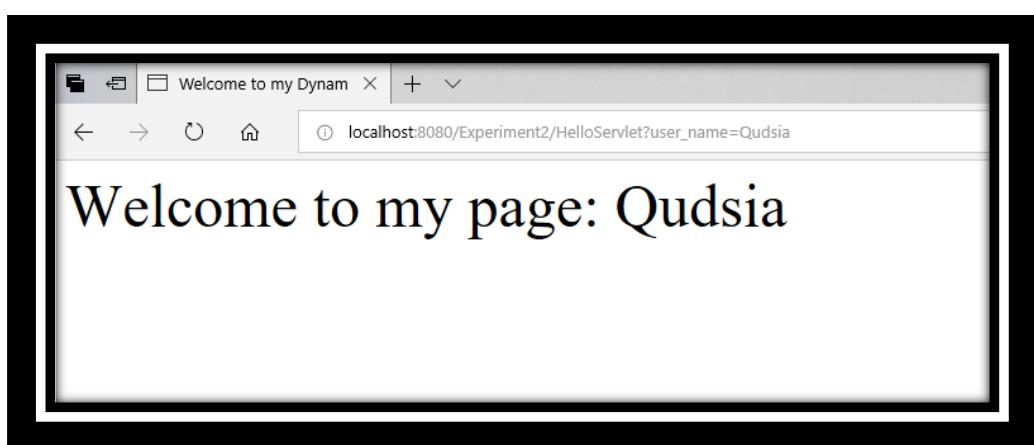
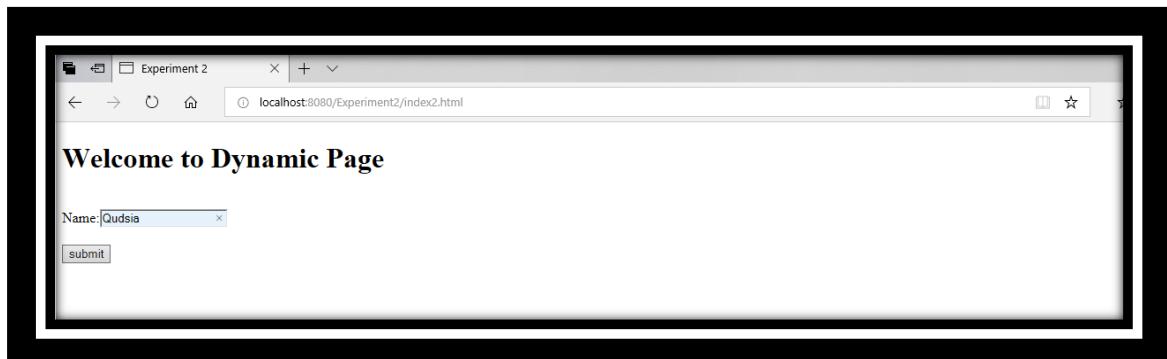
XML Code:

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
    <display-name>Experiment2</display-name>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
        <welcome-file>default.html</welcome-file>
        <welcome-file>default.htm</welcome-file>
        <welcome-file>default.jsp</welcome-file>
    </welcome-file-list>
    <servlet>
        <description></description>
        <display-name>hello</display-name>
        <servlet-name>SecondServlet</servlet-name>
        <servlet-class>com.qh2.pkg.SecondServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>SecondServlet</servlet-name>
        <url-pattern>/HelloServlet</url-pattern>
    </servlet-mapping>
</web-app>

```

OUTPUT:



3. Write a Dynamic Web Project to display the interests selected by the user using checkboxes.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Experiment 3</title>
</head>
<body>

<h1>Welcome to third Experiment</h1>

<form action ='MyThirdServlet' method = 'post'>
Username:<input type = 'text' value = 'Your Name' name = 'user_name'><br>
Age:<input type = 'text' value = 'Your Age' name = 'user_age'><br>

<input type = 'checkbox' name = 'interest' value = 'Reading'>Reading<br>
<input type = 'checkbox' name = 'interest' value = 'Riding'>Riding<br>
<input type = 'checkbox' name = 'interest' value = 'Sports'>Sports<br>
<input type = 'checkbox' name = 'interest' value = 'Programming'>Programming<br>
<input type = "submit" value = "submit">

</form>

</body>
</html>
```

SERVLET Code:

```
package com.qh3.pkg;

import java.io.IOException;
import java.io.PrintWriter;

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

/**
```

```

 * Servlet implementation class MyThirdServlet
 */
public class MyThirdServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Default constructor.
     */
    public MyThirdServlet() {
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
     * response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);

    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
     * response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.setContentType("text/html");
        try {
            PrintWriter out = response.getWriter();

```

```

// for name

String username = request.getParameter("user_name");
String userage = request.getParameter("user_age");
int u = Integer.parseInt(userage);

//for interest checkbox

String select[] = request.getParameterValues("interest");

// for dynamic page

out.println("<html>");
out.println("<head>");
out.println("<title>Newpage</title>");
out.println("</head>");
out.println("<body>");
out.println("<h1>"+"Welcome " +username + " to my page" + "</h1>");
for(String value : select)
{
    out.println("<h2>" + "Your Interests are:" +value);
}
out.println("<h2>" + "Your age is:" +userage );
out.println("<br>");
out.println("<br>");
out.println("</body>");
out.println("</body>");
out.println("</html>");

// condition for age

if(u > 20)
{
    out.println("You are eligible");
}
else
{
    out.println("You are not eligible");
}

```

```

        }

        out.close();

    }catch(Exception exception)
    {
        exception.printStackTrace();
    }

}

}

```

XML Code:

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
    <display-name>Experiment3'</display-name>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
        <welcome-file>default.html</welcome-file>
        <welcome-file>default.htm</welcome-file>
        <welcome-file>default.jsp</welcome-file>
    </welcome-file-list>
    <servlet>
        <description></description>
        <display-name>MyThirdServlet</display-name>
        <servlet-name>MyThirdServlet</servlet-name>
        <servlet-class>com.qh3.pkg.MyThirdServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>MyThirdServlet</servlet-name>
        <url-pattern>/MyThirdServlet</url-pattern>
    </servlet-mapping>
</web-app>

```

OUTPUT:

The screenshot shows a web browser window titled "Experiment 3". The address bar displays "localhost:8080/Experiment3/index3.html". The page content is titled "Welcome to third Experiment". It contains a form with the following fields:

- Username: Qudsia
- Age: 21
- Reading (unchecked)
- Riding (checked)
- Sports (checked)
- Programming (checked)

A submit button is also present.

The screenshot shows a web browser window titled "Newpage". The address bar displays "localhost:8080/Experiment3/MyThirdServlet". The page content is titled "Welcome Qudsia to my page" and displays the following information:

- Your Interests are: Riding
- Your Interests are: Sports
- Your Interests are: Programming
- Your age is: 21
- You are eligible

4. Write a Dynamic Web Project to implement visit counter using init method, config object and context object.

XML Code:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
  <display-name>Experiment4</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
  <servlet>
    <description></description>
    <display-name>FourthServlet</display-name>
    <servlet-name>FourthServlet</servlet-name>
    <servlet-class>com.qh4.pkg.FourthServlet</servlet-class>
    <init-param>
      <param-name>initial</param-name>
      <param-value>0</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>FourthServlet</servlet-name>
    <url-pattern>/FourthServlet</url-pattern>
  </servlet-mapping>
  <context-param>
    <param-name>admin</param-name>
    <param-value>qudsiahussain@gmail.com</param-value>
  </context-param>
</web-app>
```

SERVLET Code:

```
package com.qh4.pkg;

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

```

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

public class FourthServlet extends HttpServlet {

    int count;

    public void init(ServletConfig config) throws ServletException{
        super.init(config);
        String initial = config.getInitParameter("initial");
        try {
            count = Integer.parseInt(initial);
        }catch(Exception e)
        {
            count = 0;
        }
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
     response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.setContentType("text/plain");
        PrintWriter out = response.getWriter();

        count++;
        out.println("Qudsia's Servlet has been accessed");
        out.println(count + " :times");
    }
}

```

```

        out.println("For any queries send mail to: "
+getServletContext().getInitParameter("admin"));

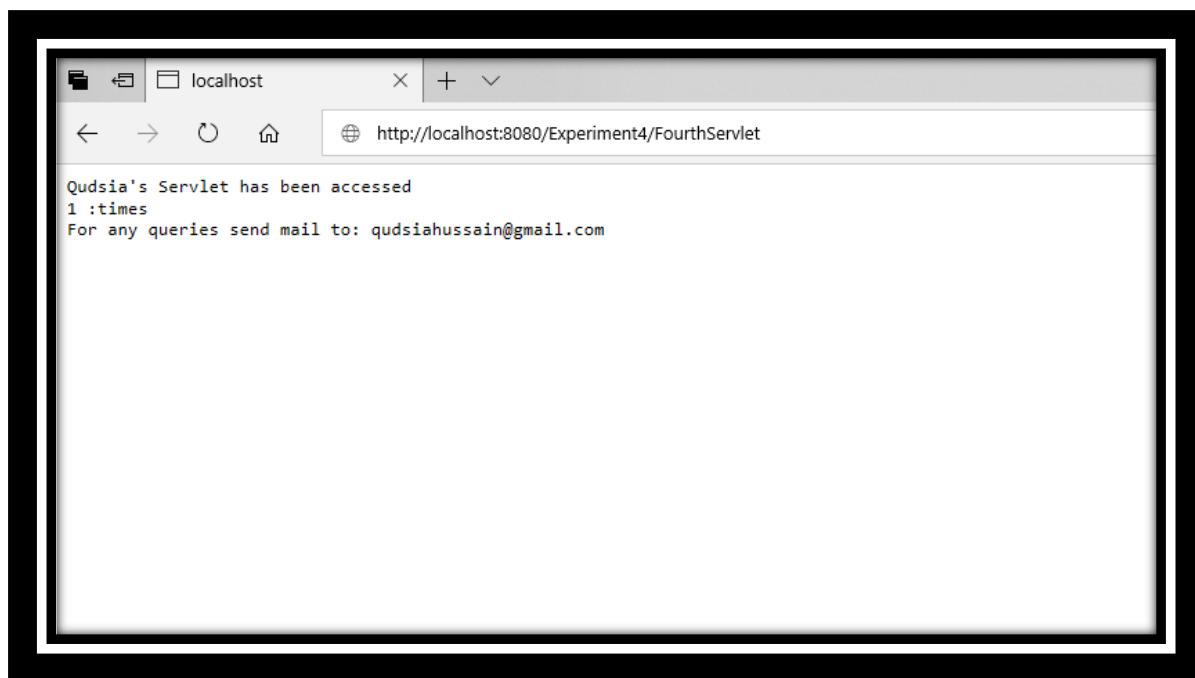
    }

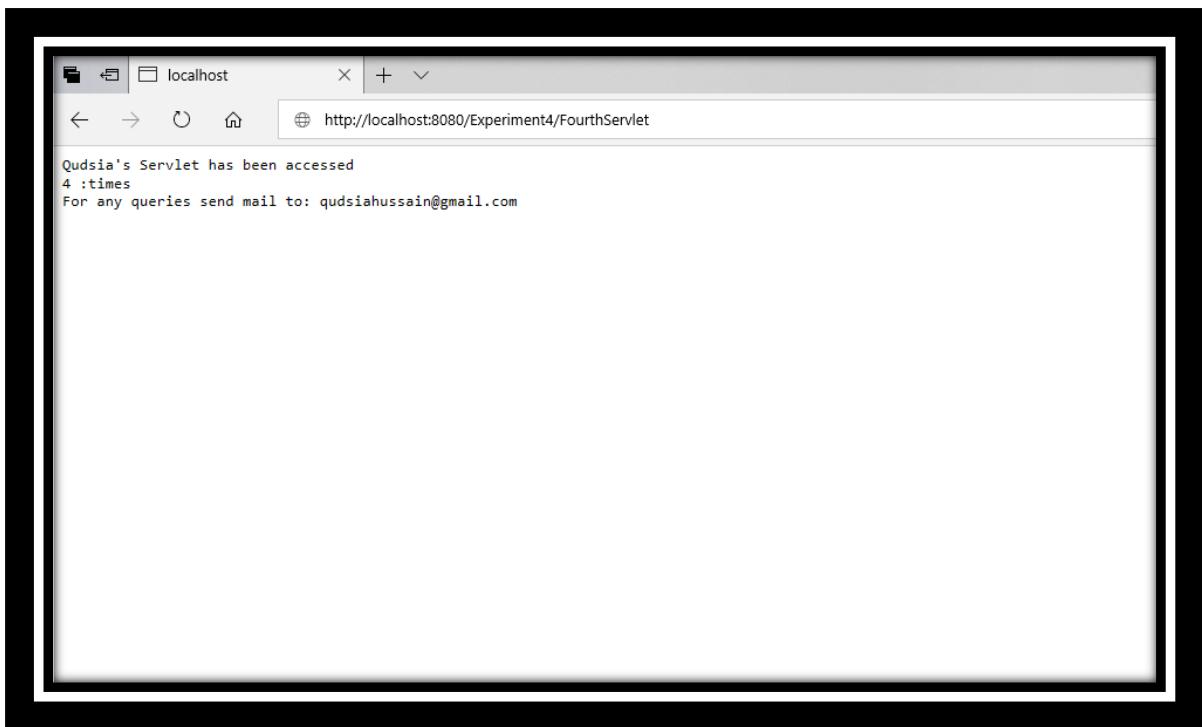
/***
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

}

```

OUTPUT:





5. Write a Dynamic Web Project to implement persistent visit counter using text file.

XML Code:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
  <display-name>Experiment5</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
  <servlet>
    <description></description>
    <display-name>FifthServlet</display-name>
    <servlet-name>FifthServlet</servlet-name>
    <servlet-class>com.exp5.pkg.FifthServlet</servlet-class>
    <init-param>
      <param-name>initial</param-name>
      <param-value>0</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>FifthServlet</servlet-name>
    <url-pattern>/FifthServlet</url-pattern>
  </servlet-mapping>
  <context-param>
    <param-name>admin</param-name>
    <param-value>qudsiahussain@gmail.com</param-value>
  </context-param>
</web-app>
```

SERVLET Code:

```
package com.exp5.pkg;

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

```

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

/**
 * Servlet implementation class FifthServlet
 */
public class FifthServlet extends HttpServlet {

    int count;

    public void init(ServletConfig config) throws ServletException{
        super.init(config);

        /**
         * Default constructor.
         */
        try {
            FileReader filereader = new FileReader("e://Experiment5.txt");
            BufferedReader bufferedreader =  new BufferedReader(filereader);
            String initial = bufferedreader.readLine();
            count = Integer.parseInt(initial);
            return;
        }catch(Exception e) {}

        String initial = getInitParameter("initial");
        try {
            count = Integer.parseInt(initial);
            return;
        }catch(Exception e) {}
        count = 0;

    }
}

```

```

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    response.setContentType("text/plain");
    PrintWriter out = response.getWriter();

    count++;
    out.println("Qudsia's Servlet has been accessed");
    out.println(count + " :times");
    out.println("For any queries send mail to: "
+getServletContext().getInitParameter("admin"));

}

public void destroy()
{
    saveState();
}

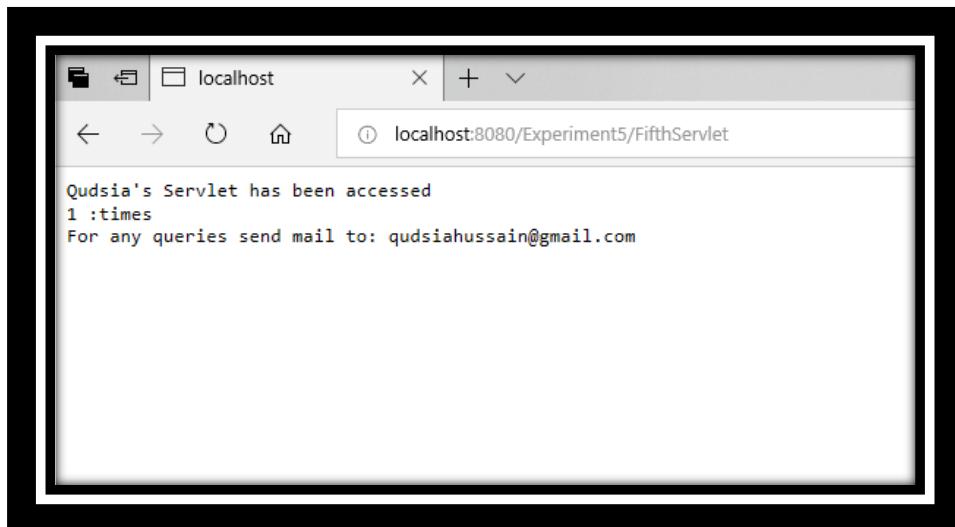
private void saveState() {
    try {
        FileWriter fileWriter = new
FileWriter("e://Experiment5.txt");
        String initial = Integer.toString(count);
        fileWriter.write(initial,0,initial.length());
        fileWriter.close();
        return;
    }catch(Exception e) {}

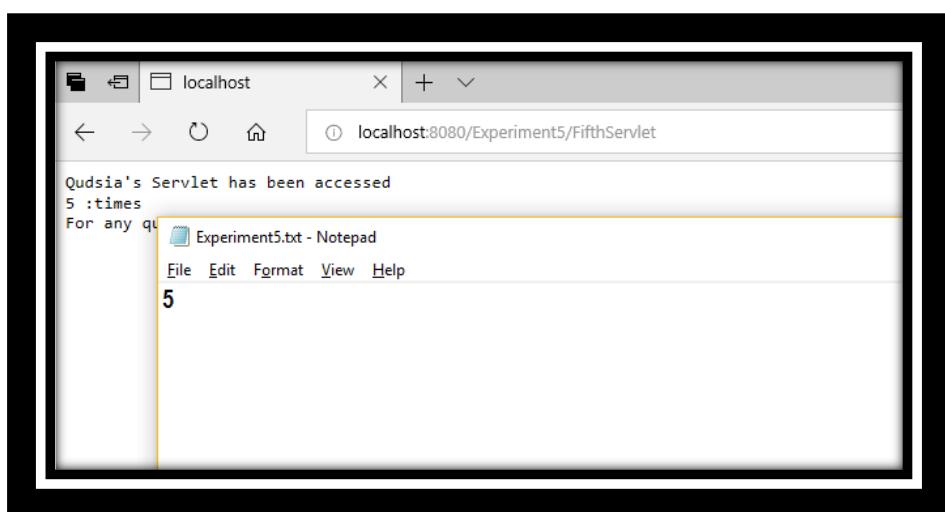
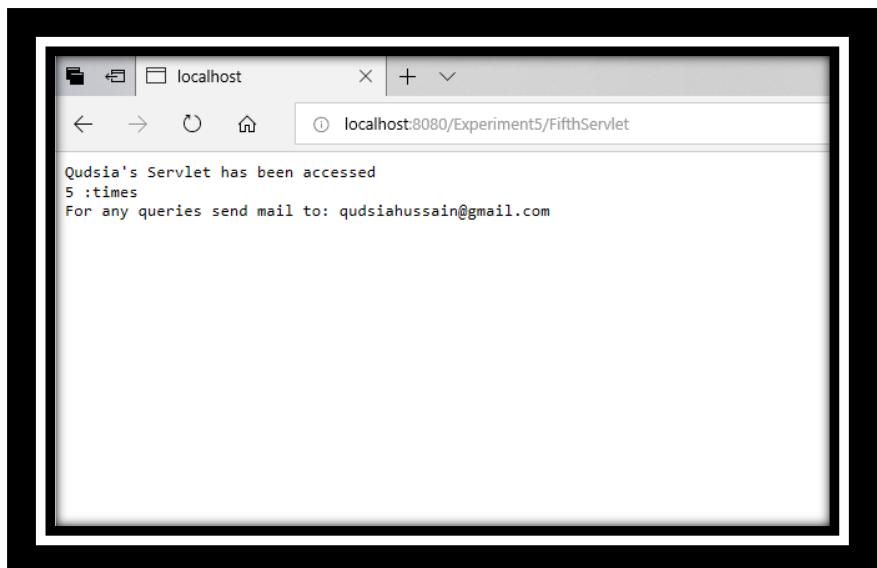
}

```

```
/**  
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse  
response)  
*/  
  
protected void doPost(HttpServletRequest request, HttpServletResponse  
response) throws ServletException, IOException {  
    // TODO Auto-generated method stub  
    doGet(request, response);  
}  
  
}
```

OUTPUT:





6. Write a Dynamic Web Project to implement session management using session variables.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="SixthServlet" method='get'>
First Name:<input type="text" name="first_name"><br>
Last Name:<input type="text" name="last_name"><br>
login Id:<input type="text" name="name"><br>
Password:<input type="password" name="password"><br>
<input type="submit" value="Login">
</form>
</body>
</html>
```

SERVLET Code:

```
package com.exp6.pkg;

import java.io.*;
import java.util.*;

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

/**
 * Servlet implementation class SixthServlet
 */
public class SixthServlet extends HttpServlet {

```

```

private static final long serialVersionUID = 1L;

/**
 * Default constructor.
 */
public SixthServlet() {
    // TODO Auto-generated constructor stub
}

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    // Create a session object if it is already not created.
    HttpSession session = request.getSession();

    // Get session creation time.
    Date createTime = new Date(session.getCreationTime());
    // Get last access time of this web page.
    Date lastAccessTime =
        new Date(session.getLastAccessedTime());
    String firstName=null;
    String lastName=null;
    String title = "Welcome Back to my website";
    Integer visitCount = Integer.valueOf("1");
    String visitCountKey = new String("visitCount");
    String userIDKey= new String("userID");
    String userIDValue;

    // Check if this is new comer on your web page.
    if (session.isNew()){
        title = "Welcome to my website";

```

```

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

        session.setAttribute(userIDKey, userIDValue);

        firstName=request.getParameter("first_name");

        lastName=request.getParameter("last_name");

        session.setAttribute("first_name", firstName);

        session.setAttribute("last_name", lastName);

        session.setAttribute("User Name",firstName+ " "+lastName);

    } else {

        firstName=(String)session.getAttribute("first_name");

        lastName=(String)session.getAttribute("last_name");

        visitCount = (Integer)session.getAttribute(visitCountKey);

        visitCount = visitCount + 1;

        userIDValue = (String)session.getAttribute(userIDKey);

    }

    session.setAttribute(visitCountKey, visitCount);

    // Set response content type

    response.setContentType("text/html");

    PrintWriter out = response.getWriter();

    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=\"#f0f0f0\">\n" +

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

"+session.getAttribute("User Name")+ "</h1>\n" +

        "<h2 align=\"center\">Session Infomation</h2>\n" +

        "<h5 align=\"right\">"+<a href=\"logout\">Logout</a>\r\n" +

        "<form action=\"SixthServlet\" align=\"center\" >\n" +

```

```

    "<table border=\"1\" align=\"center\"\>\n" +
    "<tr bgcolor=\"#949494\"\>\n" +
    "  <th>Session info</th><th>value</th></tr>\n" +
    "<tr>\n" +
    "  <td>id</td>\n" +
    "  <td>" + session.getId() + "</td></tr>\n" +
    "<tr>\n" +
    "  <td>Creation Time</td>\n" +
    "  <td>" + createTime +
    "  </td></tr>\n" +
    "<tr>\n" +
    "  <td>Time of Last Access</td>\n" +
    "  <td>" + lastAccessTime +
    "  </td></tr>\n" +
    "<tr>\n" +
    "  <td>User ID</td>\n" +
    "  <td>" + userIDValue +
    "  </td></tr>\n" +
    "  <td>Number of visits</td>\n" +
    "  <td>" + visitCount+
    "  </td></tr>\n" +
    "<tr>\n" +
    "  <td>First Name : </td>\n" +
    "  <td>" + firstName + "</td></tr>\n" +
    "<tr>\n" +
    "  <td>Last Name : </td>\n" +
    "  <td>" + lastName + "</td></tr>\n" +
    "<tr>\n" +
    "  <td> </td>\n" +
    "  <td>" + "<input type=\"submit\" value=\"Re-View Session
data and info\" />\n" + "</td></tr>\n" +
    "</table>\n" +
    "</form>\n"

```

```

        "</body>\n"+
        "</html>");

    //session.invalidate();
    //session.setMaxInactiveInterval(60*60);
    // session.getMaxInactiveInterval();
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

}

```

XML Code:

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
<display-name>Experiment6</display-name>
<welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
</welcome-file-list>
<servlet>
    <description></description>
    <display-name>SixthServlet</display-name>

```

```

<servlet-name>SixthServlet</servlet-name>
<servlet-class>com.exp6.pkg.SixthServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>SixthServlet</servlet-name>
<url-pattern>/SixthServlet</url-pattern>
</servlet-mapping>
</web-app>

```

OUTPUT:

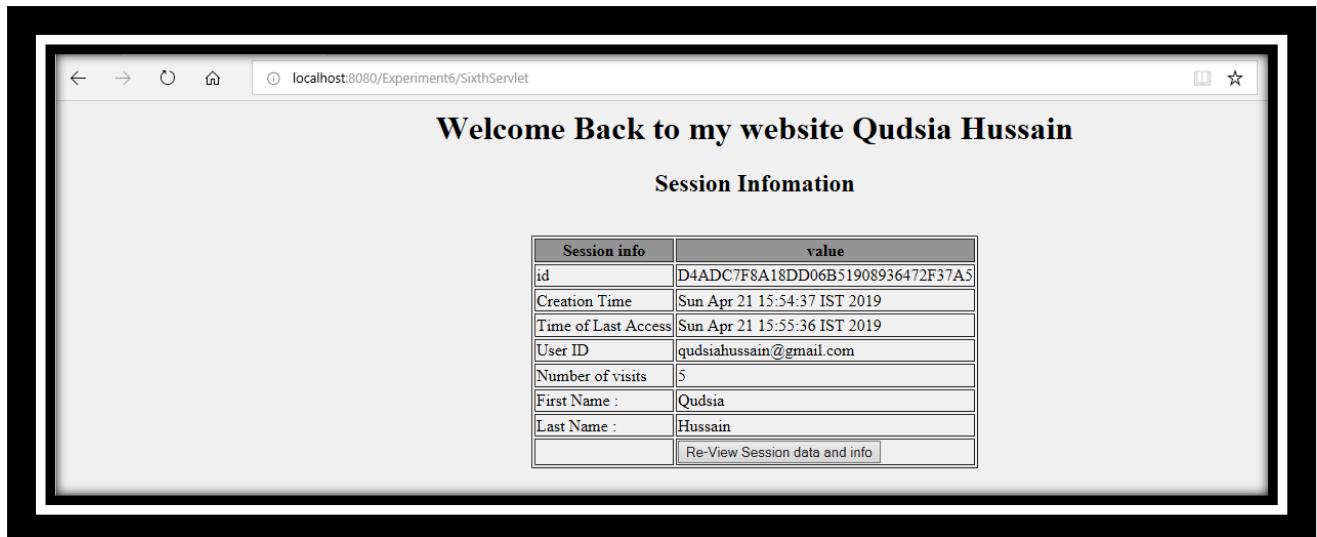
localhost:8080/Experiment6/index6.html

First Name: Qudsia
Last Name: Hussain
login Id: siahussain@gmail.com
Password: *****

Welcome to my website Qudsia Hussain

Session Infomation

Session info	value
id	D4ADC7F8A18DD06B51908936472F37A5
Creation Time	Sun Apr 21 15:54:37 IST 2019
Time of Last Access	Sun Apr 21 15:54:37 IST 2019
User ID	qudsiahussain@gmail.com
Number of visits	1
First Name :	Qudsia
Last Name :	Hussain
	Re-View Session data and info



7. Write a Dynamic Web Project to set cookies and send them to client via response header.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Experiment 7</title>
</head>
<body>
<form action = "SeventhServlet" method = "get">
Name:<input type = "text" name = "user_name">
First Number:<input type = "text" name = "num1">
Second Number:<input type = "text" name = "num2">
<input type = "submit" value = "Add">
</form>
</body>
</html>
```

SERVLET Code:

```
package com.exp7.pkg;

import java.io.IOException;
import java.io.*;

import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class SeventhServlet
 */
public class SeventhServlet extends HttpServlet {
```

```

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

    int i = Integer.parseInt(request.getParameter("num1"));

    int j = Integer.parseInt(request.getParameter("num2"));

    int k = i + j;

    Cookie add = new Cookie("k",k + "");

    response.addCookie(add);

    String name = request.getParameter("user_name");
    PrintWriter out = response.getWriter();
    out.println("Hello " +name);
    out.println("The Result is:<br>" +k);

}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

}

```

XML Code:

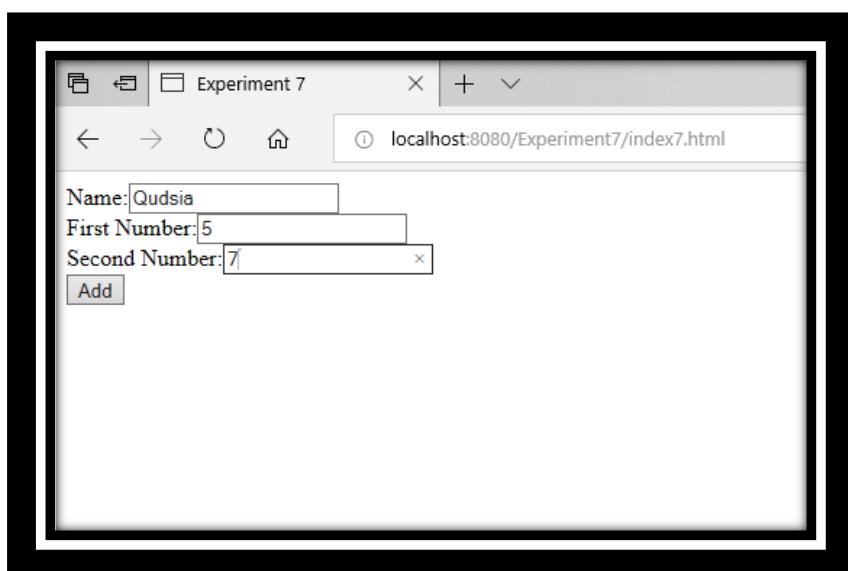
```

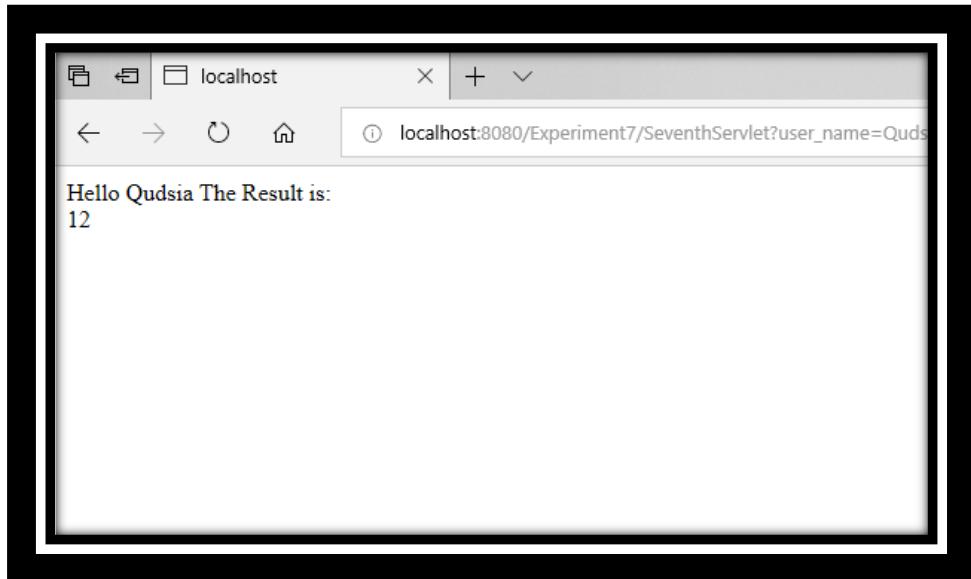
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"

```

```
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
<display-name>Experiment7</display-name>
<welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
</welcome-file-list>
<servlet>
    <description></description>
    <display-name>SeventhServlet</display-name>
    <servlet-name>SeventhServlet</servlet-name>
    <servlet-class>com.exp7.pkg.SeventhServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>SeventhServlet</servlet-name>
    <url-pattern>/SeventhServlet</url-pattern>
</servlet-mapping>
</web-app>
```

OUTPUT:





8. Write a Dynamic Web Project to read the cookies sent to the client.

HTML Code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action = "EightServlet" method = "get">
Name:<input type = "text" name = "user_name">
<br>
First Number:<input type = "text" name = "num1">
<br>
Second Number:<input type = "text" name = "num2">
<br>
<input type = "submit" value = "Add and Square">
</form>
</body>
</html>
```

SERVLET Code:

```
#1
package com.exp8.pkg;

import java.io.IOException;
import java.io.*;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class EightServlet
 */
@WebServlet("/EightServlet")
```

```

public class EightServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * Default constructor.
     */
    public EightServlet() {
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
     * response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        int i = Integer.parseInt(request.getParameter("num1"));

        int j = Integer.parseInt(request.getParameter("num2"));

        int k = i + j;

        Cookie add = new Cookie("k",k + "");

        response.addCookie(add);
        response.sendRedirect("EightServlet2");
    }

}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
 * response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

```

```
// TODO Auto-generated method stub
doGet(request, response);
}

}
```

#2

```
package com.exp8.pkg;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class EightServlet2
 */
@WebServlet("/EightServlet2")
public class EightServlet2 extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public EightServlet2() {
        super();
        // TODO Auto-generated constructor stub
    }
}
```

```

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

    int k = 0;
    Cookie cookie[] = request.getCookies();

    for (Cookie c : cookie)
    {
        if(c.getName().equals("k"))
        {
            k = Integer.parseInt(c.getValue());
        }
    }

    k = k * k;
    String name = request.getParameter("user_name");

    PrintWriter out = response.getWriter();
    out.println("Result is:" +k);
    out.println("Hello " +name);

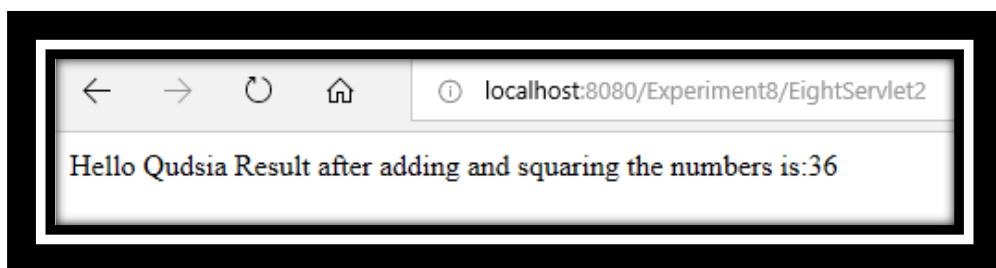
} @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
*/
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}

```

}

OUTPUT:

A screenshot of a web browser window. The title bar says "Insert title here". The address bar shows "localhost:8080/Experiment". The form contains three text input fields: "Name: Qudsia", "First Number: 3", and "Second Number: 3". Below the inputs is a button labeled "Add and Square".



9. Write a Dynamic Web Application using JSP to implement

- a) Directives b) Scriptlet c) Expression d) Declaration

#Directives:

JSP Code:

#File 1 → index.jsp

```
<%@page contentType="text/html"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Page Attributes </title>
</head>
<body>
<form action="Directive.jsp">
<h1>Enter the value of n1 and n2: </h1>
Number1: <input type="number" name="n1"/><br/>
Number2 <input type="number" name="n2"/><br/>
<input type="submit"/>
</form>
</body>
</html>
```

#File 2 → Directives.jsp

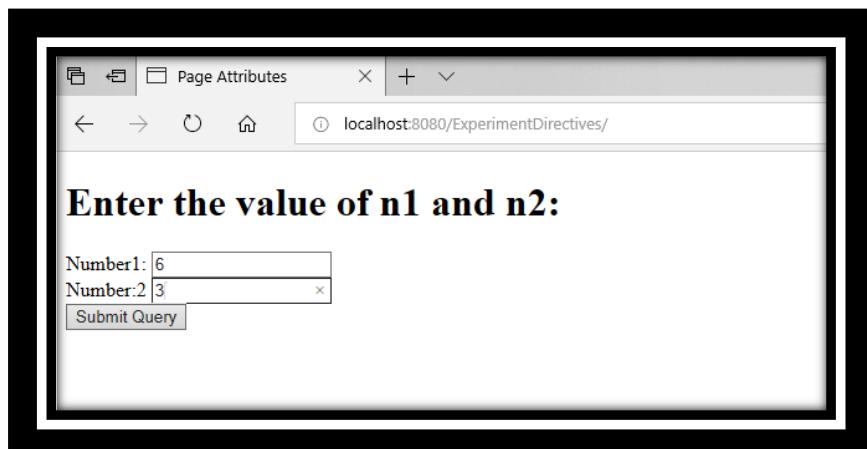
```
<%@ page contentType="text/html"
    pageEncoding="ISO-8859-1"%>
<%@ page import="java.util.*" %>
<%@ page info="composed by DSATM" %>
<%@ page language="java"%>
<%@ page buffer="16kb" %>
<%@ page autoFlush="true" %>
<%@ page isThreadSafe="true" %>
<%@ page errorPage="error.jsp" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Page Attributes</title>
</head>
<h2>Welcome to Qudsia's JSP Directives Page</h2>
<h2>Usage of Import Attributes </h2>
<h2>Today's Date is: <%=new Date()%></h2>
<h2>To See the use of Error page enter n2 value zero and click submit </h2>
<% int n1=Integer.parseInt(request.getParameter("n1"));
int n2=Integer.parseInt(request.getParameter("n2"));
%>
<h2>Value of n1/n2 = <%=n1/n2 %></h2>
</body>
```

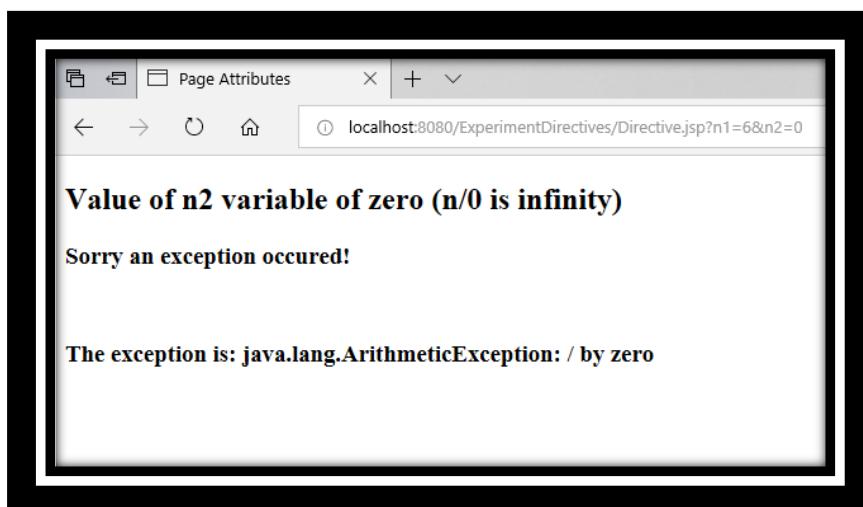
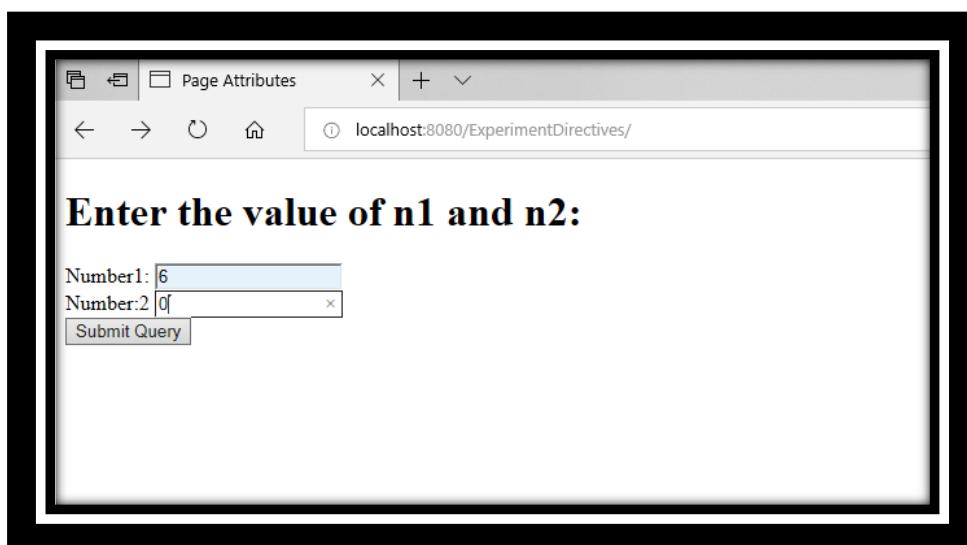
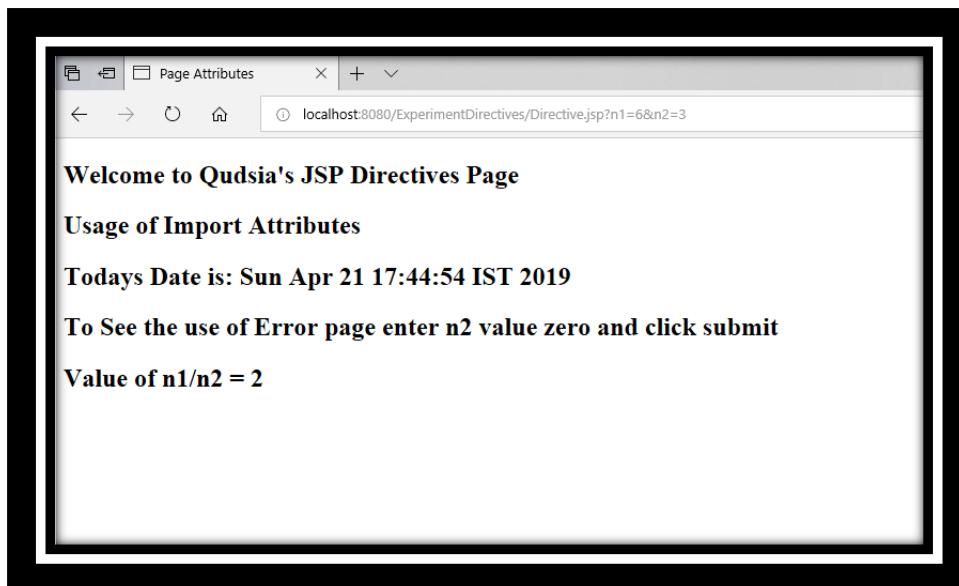
```
</html>
```

#File 3 → error.jsp

```
<%@page contentType="text/html"
   pageEncoding="ISO-8859-1"%>
<%@ page isErrorPage="true" %>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Page Attributes</title>
</head>
<body>
<h2>Value of n2 variable of zero (n/0 is infinity)</h2>
<h3> Sorry an exception occurred!</h3><br/>
<h3> The exception is: <%= exception%></h3>
</body>
</html>
```

OUTPUT:





#Scriptlets, Expression and Declaration:

JSP Code:

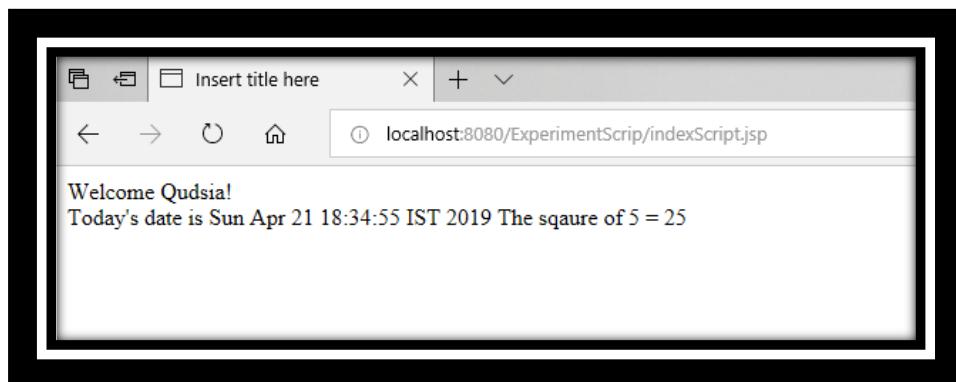
#Scriptlet

```
<%@ page language="java" import = "java.util.Calendar" contentType="text/html;
charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
Welcome Qudsia!
<br>
Today's date is
<%--Scriptlet --%>
<%
int x =5;

out.println(Calendar.getInstance().getTime());
out.println("The square of 5 = "+x*x);

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

OUTPUT:



#Expression

JSP Code:

```

<%@ page language="java" import = "java.util.Calendar" contentType="text/html;
charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
Welcome Qudsia!
<br>
Today's date is
<%--Scriptlet --%>
<%
int x =5;

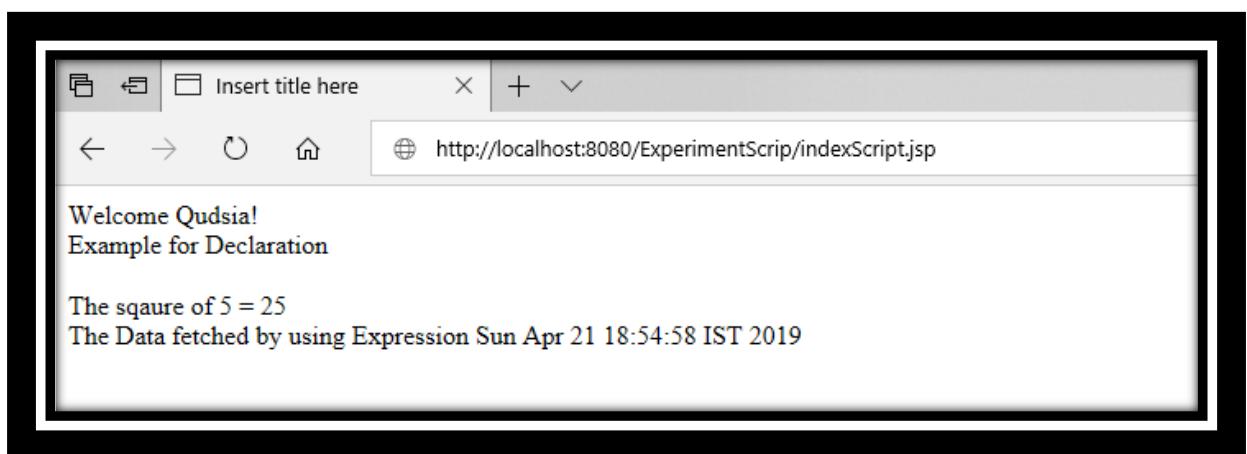
out.println(Calendar.getInstance().getTime());
out.println("The square of 5 = "+x*x);
%>
<%--Expressions --%>
The Data fetched by using Expression
<%= Calendar.getInstance().getTime()

%>

</body>
</html>

```

OUTPUT:



#Declaraton

JSP Code:

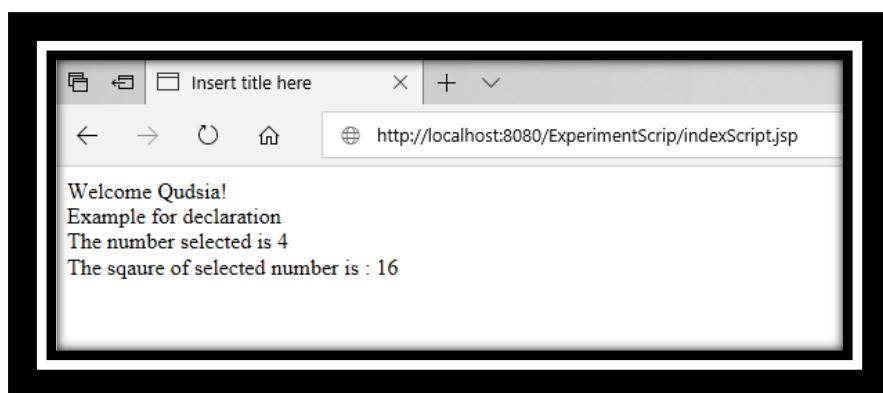
```

<%@ page language="java" import = "java.util.*" contentType="text/html;
charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>

```

```
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
Welcome Qudsia!
<br>
Example for declaration
<br>
<%!
public int square(int x)
{
return x*x;
}
%>
<%
out.println("The number selected is 4");
%>
<br>
<%
out.println("The sqaure of selected number is :");
out.println(+square(4));
%>
</body>
</html>
```

OUTPUT:



10. Write a Dynamic Web Application to implement MVC without Database.

#SERVLET CODE

LoginControl

```
package com.qh.servlets.pkg;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.DispatcherType;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class LoginControl
 */
@WebServlet("/LoginControl")
public class LoginControl extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LoginControl() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     */

    
```

```

        * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
    */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
*/
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    response.setContentType("text/html");

    String name = request.getParameter("name");
    String password = request.getParameter("password");
    PrintWriter out = response.getWriter();

    LoginBean bean = new LoginBean();
    bean.setUserName(name);
    bean.setPassword(password);
    request.setAttribute("bean", bean);

    boolean status=LoginValidator.validate(bean);

    if(status)
    {
        RequestDispatcher rd =
request.getRequestDispatcher("loginsuccess.jsp");
        rd.forward(request, response);
    }
    else {

```

```

        RequestDispatcher
rd=request.getRequestDispatcher("loginerror.jsp");
        rd.forward(request, response);
    }
}
}

```

LoginBean

```

package com.qh.servlets.pkg;

public class LoginBean {
    private String username,password;

    public String getUserName()
    {
        return username;
    }

    public void setUserName(String name)
    {
        this.username = name;
    }

    public String getPassword()
    {
        return password;
    }
    public void setPassword(String password)
    {
        this.password = password;
    }
}

```

LoginValidator

```

package com.qh.servlets.pkg;

public class LoginValidator {
    public static boolean validate(LoginBean bean) {
        if(bean.getPassword().equals("admin")) {
            return true;
        }
        else {
            return false;
        }
    }
}

```

#HTML CODE

INDEX

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>MVC Without DB</title>
</head>
<body>

<form action = "LoginControl" method = "post">
You are successfully logged in!
<br>
<br>
Welcome Qudsia
</form>
</body>
</html>
```

INDEX ERROR

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<p>Sorry! username or password incorrect</p>
<%@ include file = "index.jsp">

%>

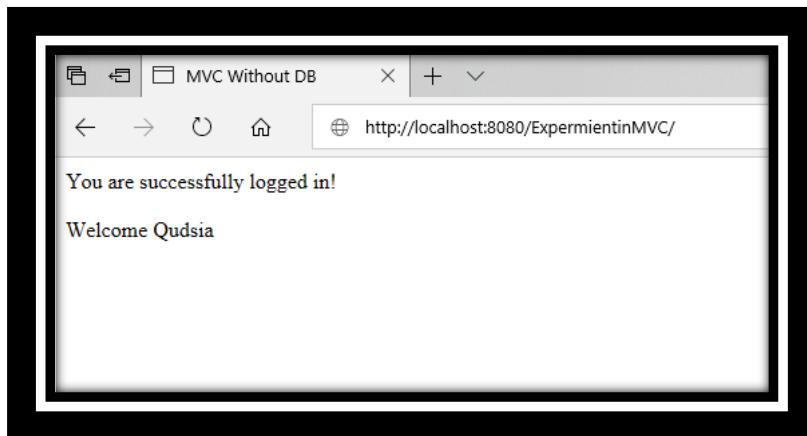
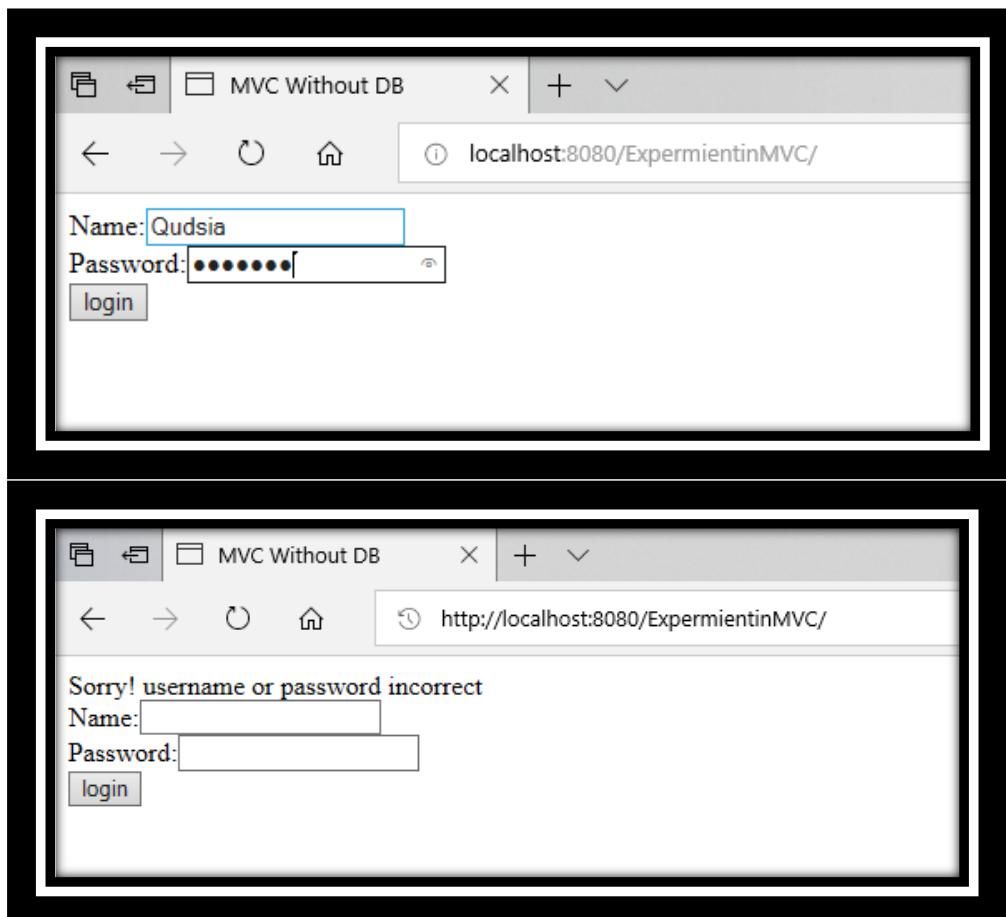
</body>
</html>
```

INDEX SUCCESS

```
<%@ page language="java" import="com.qh.servlets.pkg.LoginBean"
    contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<p>You are successfully logged in!</p>
<%
LoginBean bean=(LoginBean)request.getAttribute("bean");
out.print("Welcome," +bean.getUserName());
%>
```

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

OUTPUT:



11. Write a Dynamic Web Application to implement MVC with Database.

#SERVLET CODE

LoginControl

```
package com.qh.servlets.pkg;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.Dispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class LoginControl
 */
@WebServlet("/LoginControl")
public class LoginControl extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LoginControl() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**

```

```

        * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
    */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
}

/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
*/
protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    response.setContentType("text/html");

    String name = request.getParameter("name");
    String password = request.getParameter("password");
    PrintWriter out = response.getWriter();

    LoginBean bean = new LoginBean();
    bean.setUserName(name);
    bean.setPassword(password);
    request.setAttribute("bean", bean);

    boolean status=LoginValidator.validate(bean);

    if(status)
    {
        RequestDispatcher rd =
request.getRequestDispatcher("loginsuccess.jsp");
        rd.forward(request, response);
    }
    else {

```

```

        RequestDispatcher
rd=request.getRequestDispatcher("loginerror.jsp");
        rd.forward(request, response);
    }
}
}

```

LoginBean

```

package com.qh.servlets.pkg;

public class LoginBean {
    private String username,password;

    public String getUserName()
    {
        return username;
    }

    public void setUserName(String name)
    {
        this.username = name;
    }

    public String getPassword()
    {
        return password;
    }
    public void setPassword(String password)
    {
        this.password = password;
    }
}

```

LoginValidator

```

package com.qh.servlets.pkg;

public class LoginValidator {
    public static boolean validate(LoginBean bean) {
        String password = null;
        LoginDao dao = new LoginDao();
        Password = dao.fetchPassword(bean.getUserName());
        If(password!=null && bean.getPassword().equals(password)){
            Return true;
        }
        else {
            return false;
        }
    }
}

```

LoginDao

```
public class fetchPassword(String username) {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {

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

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

            stmt = con.createStatement();

            int count=stmt.executeUpdate("insert into usertable
values(11,'aaa1','aaa1pwd')");
            count=stmt.executeUpdate("update usertable set
UserPassword='helloabcde2' where UserName='Rubina'");
            count=stmt.executeUpdate("delete from usertable where
UserName='hussain'");

Scanner scan = new Scanner(System.in);
String sql=scan.nextLine();
boolean result=stmt.execute(sql);
if(result==true) {
    ResultSet rs=stmt.getResultSet();
    while (rs.next())
        System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));

} else {
    System.out.println("the number of records effected are
= "+stmt.getUpdateCount());
}

con.close();
} catch (SQLException e) {
    System.out.println(" Inside catch SQLException block");
    System.out.println(e.getCause());

    e.printStackTrace();
} catch (ClassNotFoundException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
}
```

```
    }
}
```

#HTML CODE

INDEX

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>MVC Without DB</title>
</head>
<body>

<form action = "LoginControl" method = "post">
You are successfully logged in!
<br>
<br>
Welcome Qudsia
</form>
</body>
</html>
```

INDEX ERROR

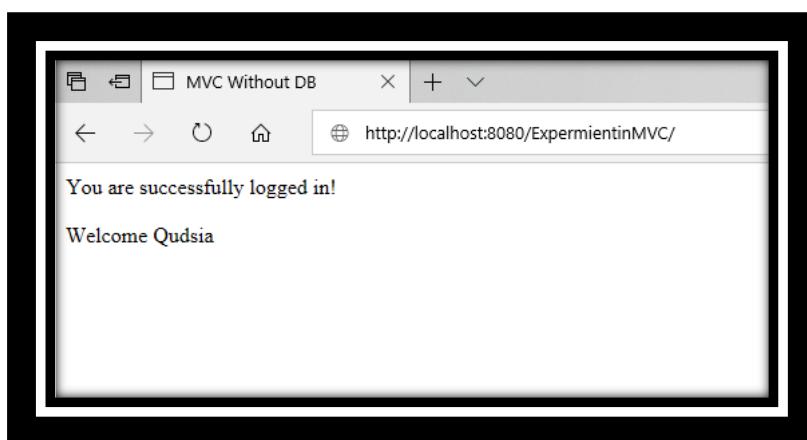
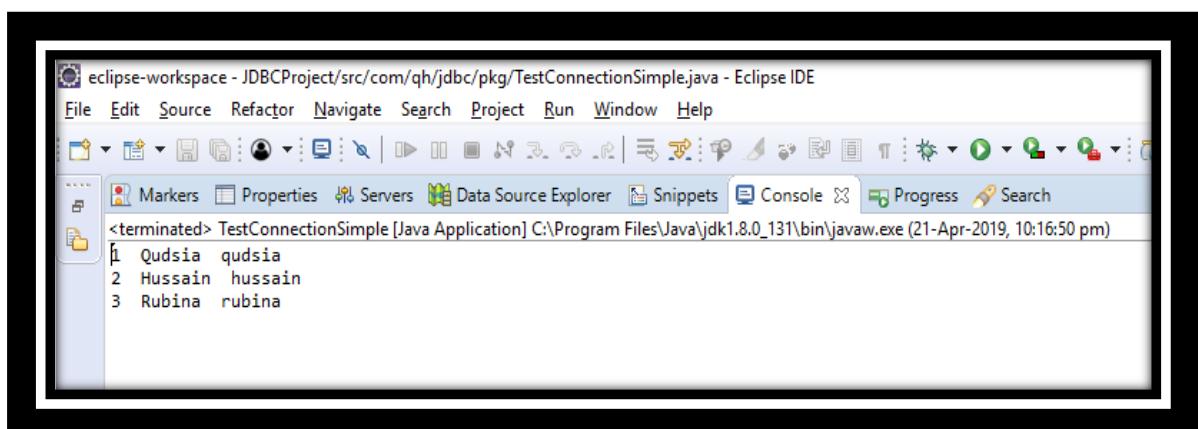
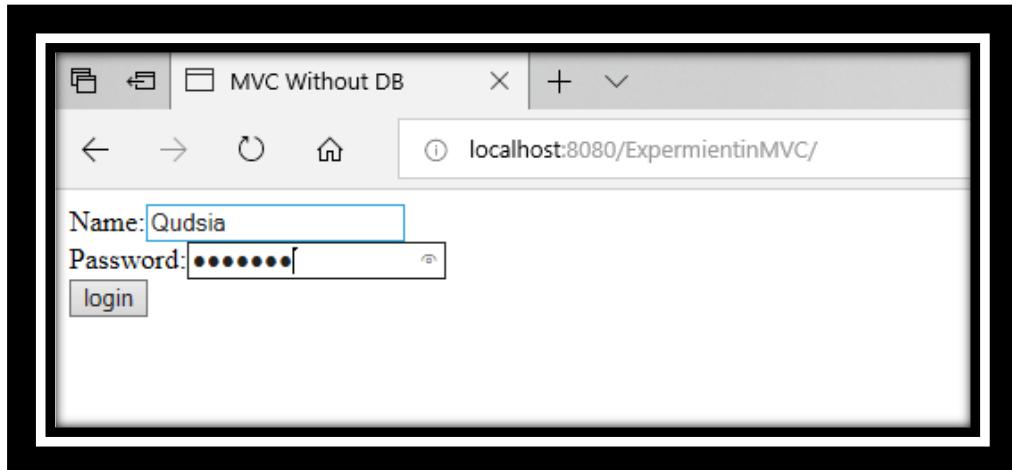
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<p>Sorry! username or password incorrect</p>
<%@ include file = "index.jsp"%
%>

</body>
</html>
```

INDEX SUCCESS

```
<%@ page language="java" import="com.qh.servlets.pkg.LoginBean"
contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
```

```
<p>You are successfully logged in!</p>
<%
LoginBean bean=(LoginBean)request.getAttribute("bean");
out.print("Welcome," +bean.getUserName());
%>
</body>
</html>
```



12. Write a java program to implement 5 simple steps to connect a database

#Simple Connection with DataBase

Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class TestConnectionSimple {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {
            //1. Register the Driver class
            Class.forName("com.mysql.cj.jdbc.Driver");

            //2. Get the Connection Object
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mytestdb", "root",
                                "root");

            //3. Create the Statement
            stmt = con.createStatement();

            //4. Execute the query
            ResultSet rs = stmt.executeQuery("select * from user_table");

            //Process the result
            while (rs.next())
                System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));

            //5. Close the connection object
            con.close();

        } catch (SQLException e) {
            System.out.println(" Inside catch SQLException block");
            System.out.println(e.getCause());
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

```
}
```

OUTPUT:

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree view is open, showing databases like librarymanage, librarymanagement, mytestdb, sakila, sys, test, and world. Under the mytestdb database, there is a 'Tables' node which contains a 'usertable'. On the right, a SQL editor window displays the query: `SELECT * FROM mytestdb.usertable;`. Below the query, the results are shown in a 'Result Grid' table:

User Id	User Name	User Password
1	Qudsia	qudsia
2	Hussain	hussain
3	Rubina	rubina
*	NULL	NULL

The screenshot shows the Eclipse IDE interface with the title bar 'eclipse-workspace - JDBCProject/src/com/qh/jdbc/pkg/TestConnectionSimple.java - Eclipse IDE'. The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations. The 'Console' tab is selected, showing the output of a Java application named 'TestConnectionSimple [Java Application]'. The output text is:
<terminated> TestConnectionSimple [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (21-Apr-2019, 10:16:50 pm)
1 Qudsia qudsia
2 Hussain hussain
3 Rubina rubina

13. Write a java program to connect to a database which handles exceptions.

Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Scanner;

public class StatementDemo {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {
            //1. Register the Driver class
            Class.forName("com.mysql.cj.jdbc.Driver");

            //2. Get the Connection Object
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mytestdb", "root",
                           "root");

            //3. Create the Statement
            stmt = con.createStatement();

            int count=stmt.executeUpdate("insert into usertable
values(11,'aaa1','aaa1pwd')");
            count=stmt.executeUpdate("update usertable set
UserPassword='helloabcde2' where UserName='Rubina'");
            count=stmt.executeUpdate("delete from usertable where
UserName='hussain'");

            //5. Close the connection object
            Scanner scan = new Scanner(System.in);
            String sql=scan.nextLine();
            boolean result=stmt.execute(sql);
            if(result==true) {
                ResultSet rs=stmt.getResultSet();
                while (rs.next())
                    System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

```

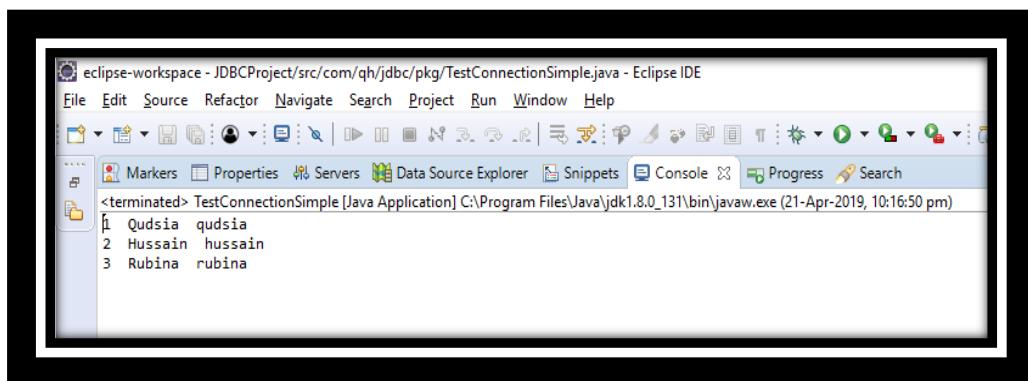
        }else {
            System.out.println("the number of records effected are
= "+stmt.executeUpdate());
        }

        con.close();
    } catch (SQLException e) {
        System.out.println(" Inside catch SQLException block");
        System.out.println(e.getCause());

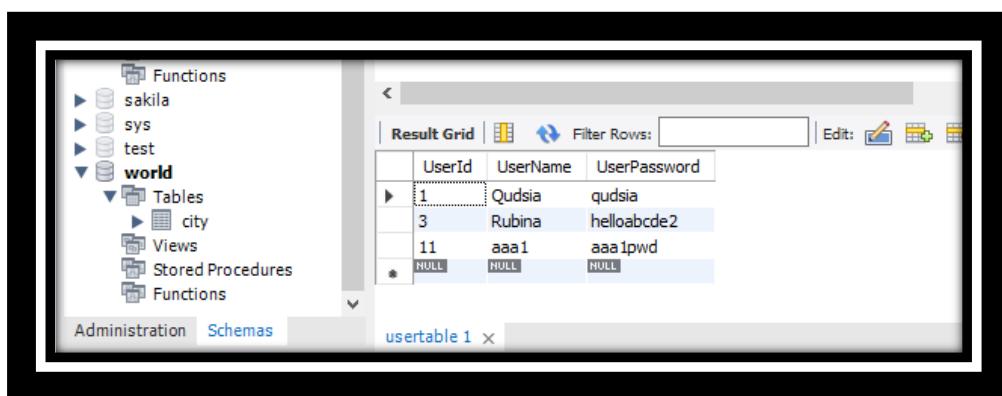
        e.printStackTrace();
    } catch (ClassNotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}

```

OUTPUT:



1 Qudsia qudsia
2 Hussain hussain
3 Rubina rubina



UserId	UserName	UserPassword
1	Qudsia	qudsia
3	Rubina	helloabcde2
11	aaa1	aaa1pwd
*	NULL	NULL

14. Write a java program to connect to a database to execute Insert, Update and Delete Queries using Statement Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Scanner;

public class StatementDemo {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {
            //1. Register the Driver class
            Class.forName("com.mysql.cj.jdbc.Driver");

            //2. Get the Connection Object
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mytestdb", "root",
                           "root");

            //3. Create the Statement
            stmt = con.createStatement();

            int count=stmt.executeUpdate("insert into usertable
values(11,'aaa1','aaa1pwd')");
            count=stmt.executeUpdate("update usertable set
UserPassword='helloabcde2' where UserName='Rubina'");
            count=stmt.executeUpdate("delete from usertable where
UserName='hussain'");

            //5. Close the connection object
            Scanner scan = new Scanner(System.in);
            String sql=scan.nextLine();
            boolean result=stmt.execute(sql);
            if(result==true) {
                ResultSet rs=stmt.getResultSet();
                while (rs.next())
                    System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));

            }else {
                System.out.println("the number of records effected are
= "+stmt.getUpdateCount());
            }
        }
    }
}
```

```

        con.close();
    } catch (SQLException e) {
        System.out.println(" Inside catch SQLException block");
        System.out.println(e.getCause());

        e.printStackTrace();
    } catch (ClassNotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}
}

```

OUTPUT:

eclipse-workspace - JDBCProject/src/com/qh/jdbc/pkg/TestConnectionSimple.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Markers Properties Servers Data Source Explorer Snippets Console Progress Search

<terminated> TestConnectionSimple [Java Application] C:\Program Files\Java\jdk1.8.0_131\bin\javaw.exe (21-Apr-2019, 10:16:50 pm)

1 Qudsia qudsia
2 Hussain hussain
3 Rubina rubina

	UserId	UserName	UserPassword
1	Qudsia	qudsia	
3	Rubina	helloabcde2	
11	aaa1	aaa1pwd	
*	NULL	NULL	

15. Write a java program to connect to a database and execute Insert Prepared Statement

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.Scanner;

class InsertPrepared {
    public static void main(String args[]) {
        try {
            Class.forName("com.mysql.jdbc.Driver");

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

            PreparedStatement stmt = con.prepareStatement("insert into
usertable values(?, ?, ?)");

            boolean exit = false;
            int i = 0;
            int records = 0;
            ResultSet rs = stmt.executeQuery("select * from usertable");
            rs.last();
            i = rs.getRow();
            Scanner scan = new Scanner(System.in);
            while (!exit) {

                System.out.println("Enter record " + (i + 2) + ":" +
username, password");
                stmt.setInt(1, i + 2);
                stmt.setString(2, scan.next());
                stmt.setString(3, scan.next());
                records = records + stmt.executeUpdate();
                System.out.println("Enter c to continue or e to exit");
                i++;
                if (scan.next().equalsIgnoreCase("e")) {
                    exit = true;
                }
            }

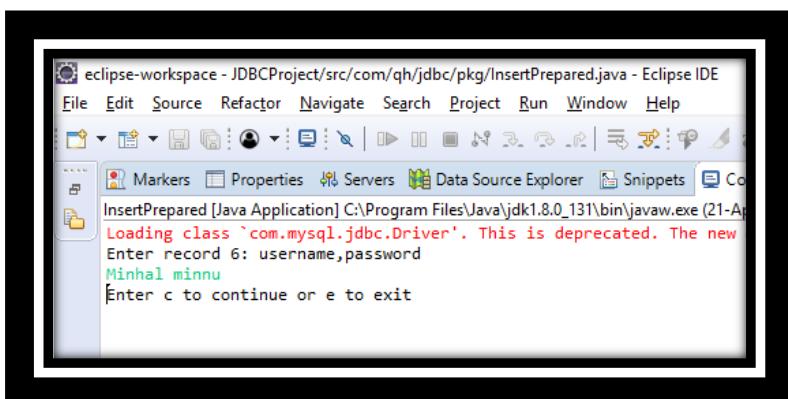
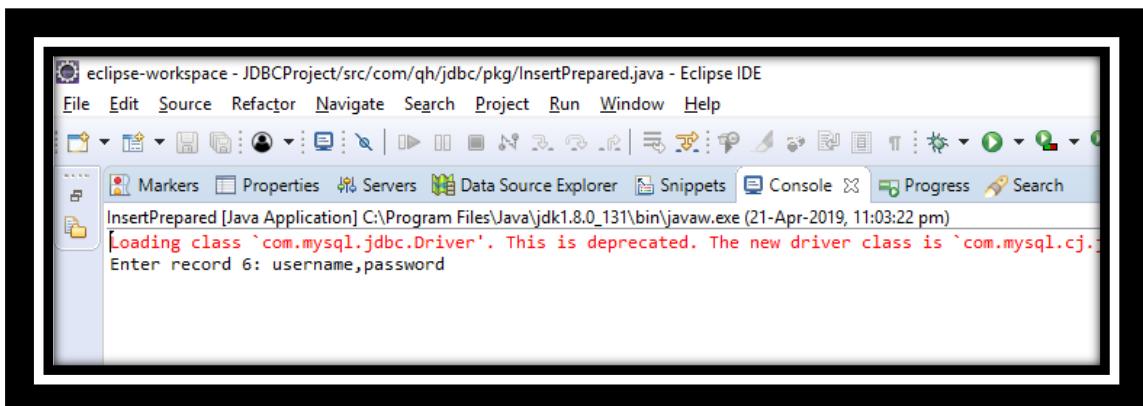
            // int i=stmt.executeUpdate();
            System.out.println(records + " records inserted");

            rs = stmt.executeQuery("select * from usertable");
            System.out.println("Present Table state  -----" +
-----");
            // Process the result
            while (rs.next())
                System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));
        }
    }
}
```

```
        con.close();

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

OUTPUT:



16. Write a java program to connect to database and execute Update Prepared Statement Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.Statement;

public class UpdatePrepared {
    public static void main(String args[]){
        try{
            Class.forName("com.mysql.jdbc.Driver");

            Statement stmt1;

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

            PreparedStatement stmt=con.prepareStatement("update usertable
set UserPassword=? where UserName=?");

            stmt.setString(1,"qudsia");
            stmt.setString(2,"hussain");

            int i=stmt.executeUpdate();
            System.out.println(i+" records updated");

            con.close();

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

    }
}
```

OUTPUT:

The screenshot shows the MySQL Workbench interface. On the left, the database tree displays the 'world' schema under the 'Tables' category. On the right, a 'Result Grid' window is open, showing a table with three columns: UserId, UserName, and UserPassword. The data in the grid is as follows:

	UserId	UserName	UserPassword
▶	1	Qudsia	qudsia
▶	3	Rubina	helloabcde2
▶	4	aaa1	aaa1pwd
▶	6	Minhal	minnu
▶	11	aaa1	aaa1pwd

This screenshot is similar to the one above, showing the MySQL Workbench interface with the 'world' schema selected. The 'Result Grid' window shows the same table data. A note 'usertable 1' is visible at the bottom of the grid area.

	UserId	UserName	UserPassword
▶	1	hussain	qudsia
▶	3	Rubina	helloabcde2
▶	4	aaa1	aaa1pwd
▶	6	Minhal	minnu
▶	11	aaa1	aaa1pwd

usertable 1 ×

17. Write a java program to connect to database and execute Delete Prepared Statement

#Delete Prepared Statement

Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.util.Scanner;

public class DeletePrepared {
    public static void main(String args[]){
        try{
            Class.forName("com.mysql.jdbc.Driver");

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

            PreparedStatement stmt=con.prepareStatement("delete from
usertable where UserName=?");

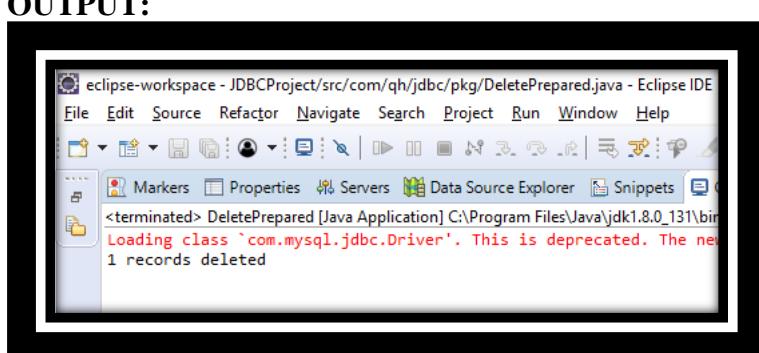
            stmt.setString(1,"Minhal");
            int i=stmt.executeUpdate();
            System.out.println(i+" records deleted");

            con.close();

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

    }
}
```

OUTPUT:



The screenshot shows a database management interface. On the left, there is a tree view of databases:

- Stored Procedures
- Functions
- sakila
- sys
- test
- world** (selected)
- Tables
 - city
- Views
- Stored Procedures
- Functions

On the right, a result grid displays data from the 'world' database:

User Id	User Name	User Password
1	hussain	qudsia
3	Rubina	helloabcde2
4	aaa1	aaa1pwd
11	aaa1	aaa1pwd
NULL	NULL	NULL

18. Write java program to connect to database to implement Transaction

#Transaction

Java Code:

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class TransactionDemo {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {
            //1. Register the Driver class
            Class.forName("com.mysql.jdbc.Driver");

            // Get the Connection Object
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mytestdb", "root",
                           "root");

            // Create the Statement
            stmt = con.createStatement();

            con.setAutoCommit(false);
            int count=stmt.executeUpdate("insert into usertable
values(17,'ab1','abcdpasswd')");
            count+=stmt.executeUpdate("insert into usertable
values(16,'ab2','uvwxyzpasswd')");

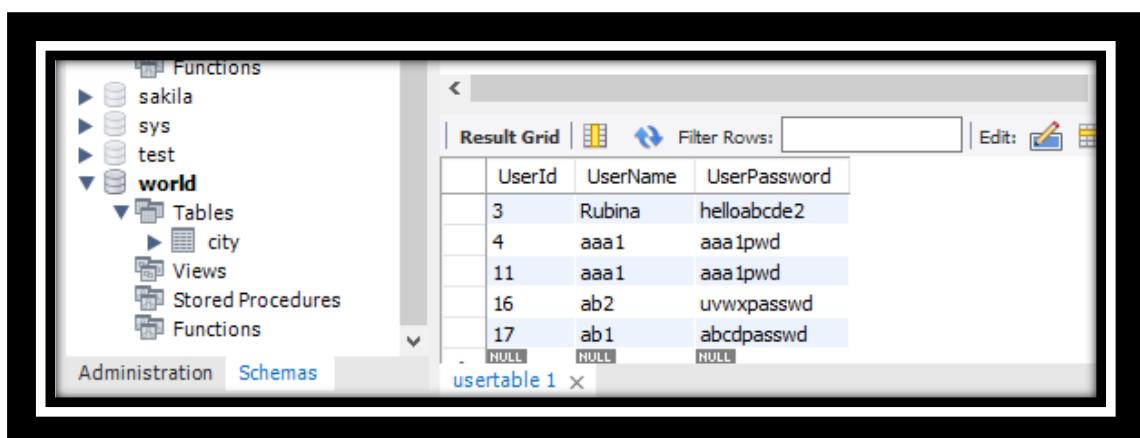
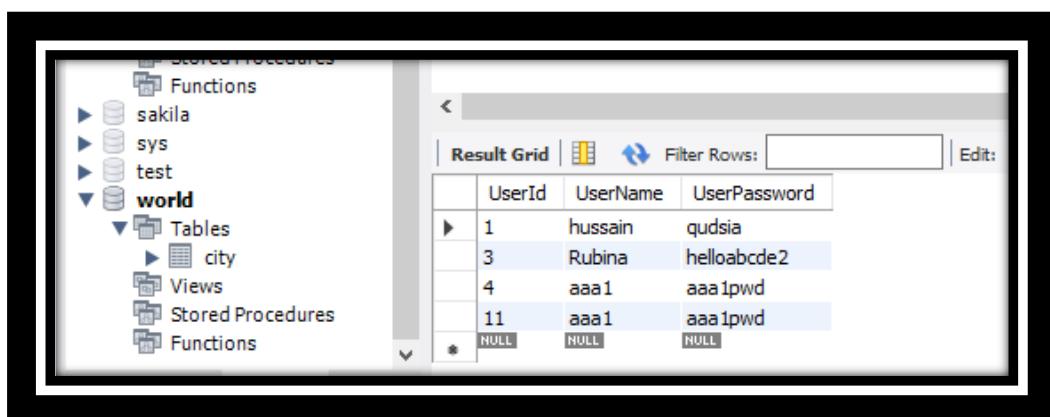
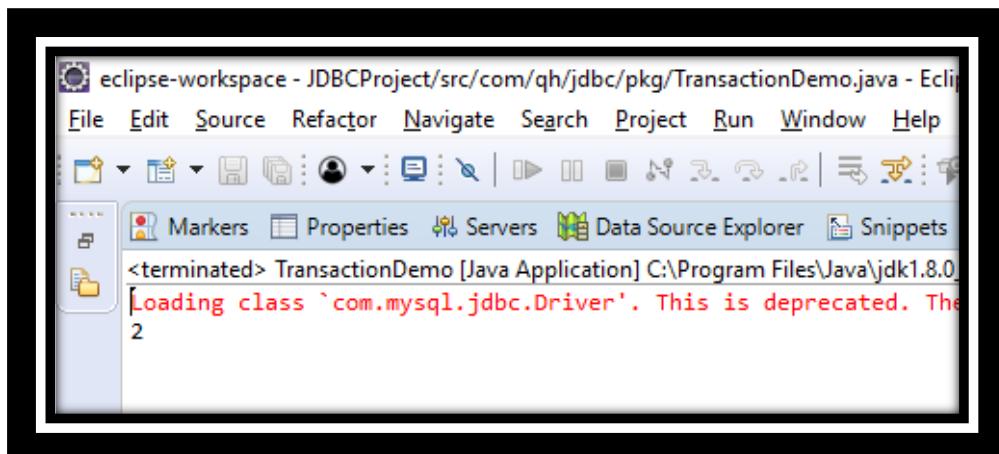
            System.out.println(count);

            con.commit();

            con.close();
        } catch (Exception e) {
            System.out.println(e);
            e.printStackTrace();
            try {
                con.rollback();
                rs=stmt.executeQuery("select * from usertable");
                while (rs.next())
                    System.out.println(rs.getInt(1) + " " +
rs.getString(2) + " " + rs.getString(3));
            } catch (SQLException e1) {
                // TODO Auto-generated catch block
                e1.printStackTrace();
            }
        }
    }
}
```

```
        }  
    }  
}
```

OUTPUT:



19. Write a java program to connect to database to implement Batch Update

#Batch

```
package com.qh.jdbc.pkg;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class BatchDemo {
    static ResultSet rs;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection con=null;
        Statement stmt=null;
        try {
            //1. Register the Driver class
            Class.forName("com.mysql.jdbc.Driver");

            // Get the Connection Object
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mytestdb", "root",
                           "root");

            // Create the Statement
            stmt = con.createStatement();

            String sql="insert into usertable
values(18,'abcde1','abcdepasswd')";
            stmt.addBatch(sql);

            stmt.addBatch("insert into usertable
values(19,'abcde2','abcdepasswd')");

            stmt.executeBatch();

stmt.addBatch("insert into usertable values(19,'abcde3','abcdepasswd')");
stmt.addBatch("insert into usertable values(20,'abcde4','abcdepasswd')");

            stmt.executeBatch();

            //Process the result
            con.close();
        } catch (SQLException e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

```
}
```

OUTPUT:

The screenshot shows the MySQL Workbench interface. On the left, the 'Schemas' tree view is expanded to show the 'world' schema, which contains tables like 'city', 'Views', 'Stored Procedures', and 'Functions'. Below the tree are tabs for 'Administration' and 'Schemas', with 'Schemas' currently selected. To the right is a 'Result Grid' window titled 'Result Grid' with a 'Filter Rows:' input field and an 'Edit' button. The grid displays data from a table with columns 'UserId', 'UserName', and 'UserPassword'. The data is as follows:

	UserId	UserName	UserPassword
	3	Rubina	helloabcde2
	4	aaa1	aaa1pwd
	11	aaa1	aaa1pwd
	16	ab2	uvwxyzpasswd
	17	ab1	abcdpasswd
	NULL	NULL	NULL

This screenshot shows the same MySQL Workbench interface as the previous one, but with more data in the result grid. The 'world' schema is selected in the tree view. The 'Result Grid' window now displays data from a table with columns 'UserId', 'UserName', and 'UserPassword'. The data is as follows:

	UserId	UserName	UserPassword
	1	hussain	qudsia
	3	Rubina	helloabcde2
	4	aaa1	aaa1pwd
	6	abcde1	abcdepasswd
	7	abcde2	abcdepasswd
	8	abcde3	abcdepasswd
	9	abcde4	abcdepasswd
	11	aaa1	aaa1pwd
	16	ah?	uvwxyznasswd

20. Write a Java Program to implement RMI

#Adder.java

```
import java.rmi.*;  
public interface Adder extends Remote{  
  
    public int add(int x,int y) throws RemoteException;  
}
```

#AdderRemote.java

```
import java.rmi.*;  
import java.rmi.server.*;  
  
public class AdderRemote extends UnicastRemoteObject implements Adder{  
  
    AdderRemote() throws RemoteException{  
        super();  
    }  
  
    public int add(int x,int y){  
        System.out.println("Parameters received for addition are "+x  
            +" and "+y);  
        System.out.println("Returning back sum="+ (x+y));  
        return x+y;  
    }  
  
}
```

#MyClient.java

```
import java.rmi.*;  
import java.util.Scanner;
```

```

public class MyClient{

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

Adder stub=(Adder) Naming.lookup("rmi://localhost:5000/Adder_remote");
//System.out.println(stub.add(34,82));
Scanner scan=new Scanner(System.in);

while(true){
System.out.println("Enter two numbers to add");
int x1=scan.nextInt();
int x2=scan.nextInt();

System.out.println("Sum="+stub.add(x1,x2));
}

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

}

```

#MyServer.java

```

import java.rmi.*;
import java.rmi.registry.*;

public class MyServer{

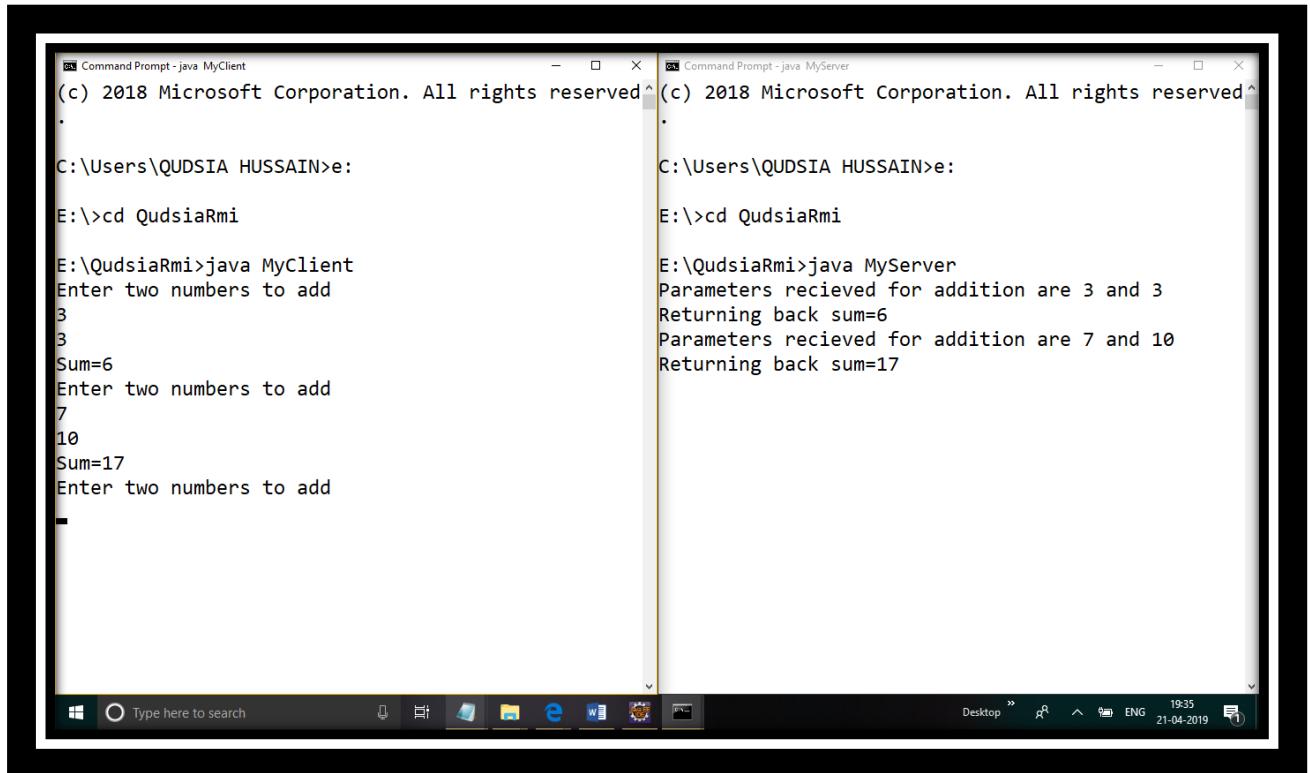
public static void main(String args[]){
try{

Adder stub=new AdderRemote();

```

```
Naming.rebind("rmi://localhost:5000/Adder_remote",stub);  
  
}catch(Exception e){System.out.println(e);}  
}  
}
```

OUTPUT:



The screenshot shows two Windows Command Prompt windows side-by-side. The left window, titled 'Command Prompt - java MyClient', displays the client's interaction with the server. The right window, titled 'Command Prompt - java MyServer', shows the server's responses to the client's requests.

Left Window (MyClient):

```
(c) 2018 Microsoft Corporation. All rights reserved.  
.  
C:\Users\QUDSIA HUSSAIN>e:  
E:>>cd QudsiaRmi  
E:\QudsiaRmi>java MyClient  
Enter two numbers to add  
3  
3  
Sum=6  
Enter two numbers to add  
7  
10  
Sum=17  
Enter two numbers to add  
-
```

Right Window (MyServer):

```
(c) 2018 Microsoft Corporation. All rights reserved.  
.  
C:\Users\QUDSIA HUSSAIN>e:  
E:>>cd QudsiaRmi  
E:\QudsiaRmi>java MyServer  
Parameters received for addition are 3 and 3  
Returning back sum=6  
Parameters received for addition are 7 and 10  
Returning back sum=17
```


SL. NO.	TOPIC	DATE	PgNo.	SIGN
1.	Write a Dynamic Web Project to greet user.		1	
2.	Write a Dynamic Web Project to greet user by the username entered.			
3.	Write a Dynamic Web Project to display the interests selected by the user using checkboxes.			
4.	Write a Dynamic Web Project to implement visit counter using inti method, config object and context object.			
5.	Write a Dynamic Web Project to implement persistent visit counter using text file.			
6.	Write a Dynamic Web Project to implement session management using session variables.			
7.	Write a Dynamic Web Project to set cookies and send them to client via response header.			
8.	Write a Dynamic Web Project to read the cookies sent to the client.			
9.	Write a Dynamic Web Application using JSP to implement: a)Directives b)Scriptlet c) Expression d)Declaration			
10.	Write a Dynamic Web Application using JSP to implement MVC without DataBase			
11.	Write a Dynamic Web Application using JSP to implement MVC with DataBase			
12.	Write a java program to implement 5 simple steps to connect a database			
13.	Write a java program to connect to a database which handles exceptions.			
14.	Write a java program to connect to a database to execute Insert, Update and Delete Queries using Statement			
15.	Write a java program to connect to a database and execute Insert Prepared Statement			
16.	Write a java program to connect to database and execute Delete Prepared Statement			
17.	Write a java program to connect to database and execute Update Prepared Statement			
18.	Write java program to connect to database to implement Transaction			
19.	Write a java program to connect to database to implement Batch Update			
20.	Write a java program to implement Remote Method Invocation.			