

## Lesson 01 Demo 04

### Session Tracking in JSP

**Objective:** To understand and implement session tracking in JSP (Java Server Pages)

**Tools required:** Eclipse IDE

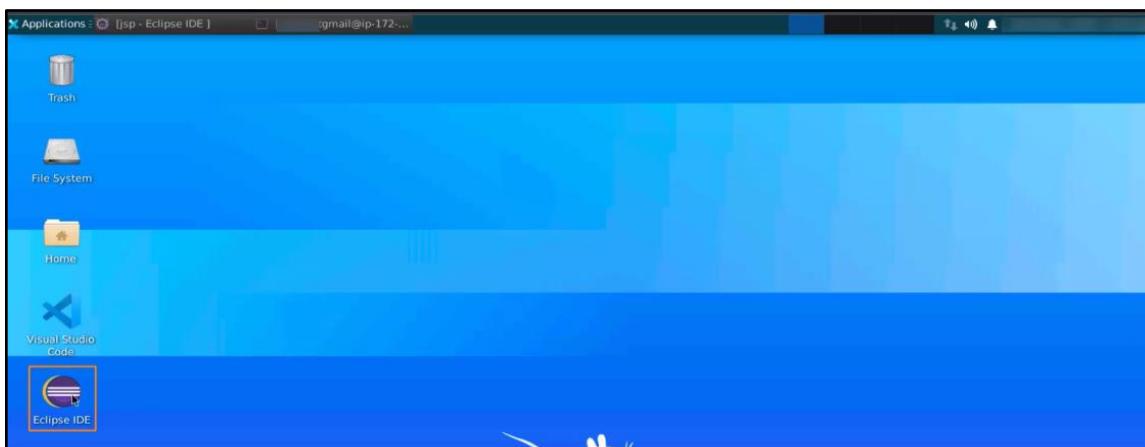
**Prerequisites:** None

#### Steps to be followed:

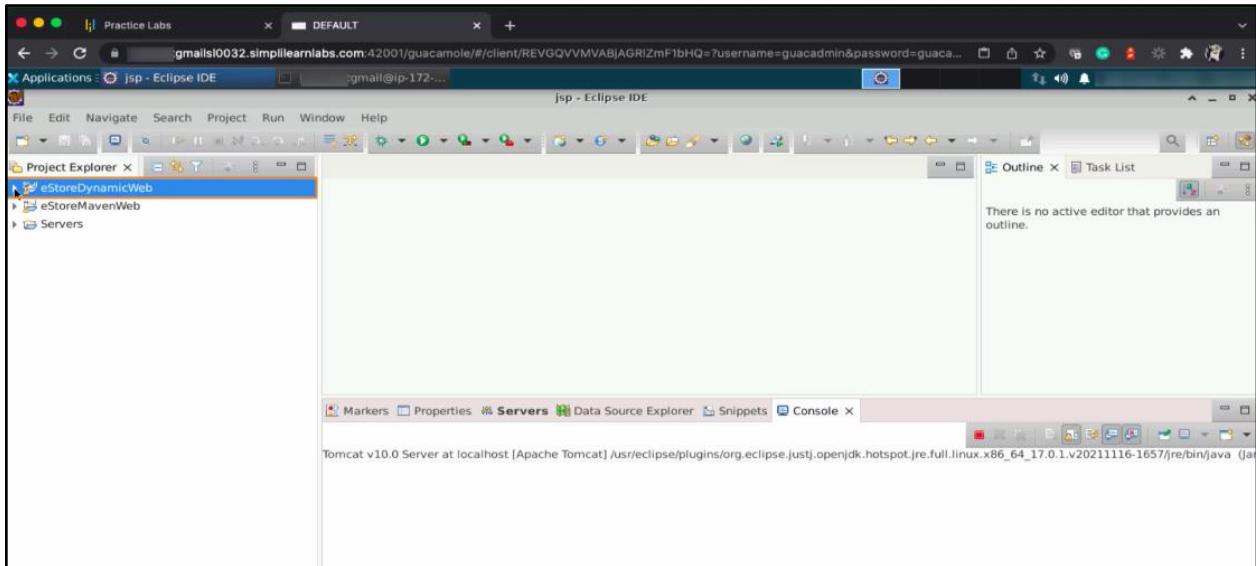
1. Create a new JSP file
2. Forward the page to the **home.jsp** page
3. Add cookies in the **login.jsp**
4. Navigate to **login.jsp** and add attributes
5. Write an operation to read the data from the URL
6. Write an operation for hidden field data
7. Write an operation for the HTTP session object

#### Step 1: Create a new JSP file

##### 1.1 Open Eclipse IDE

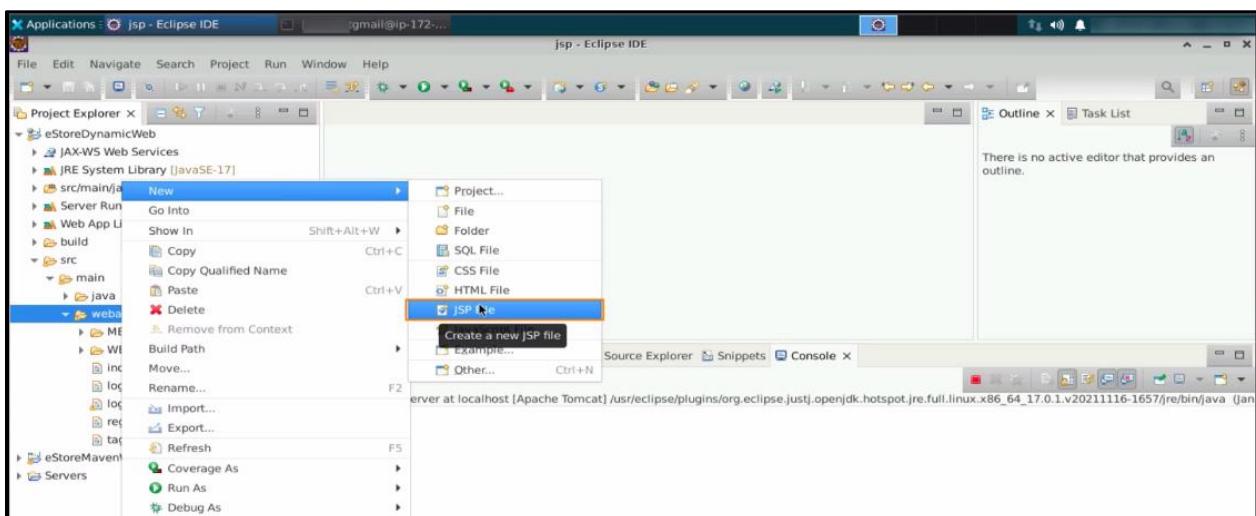


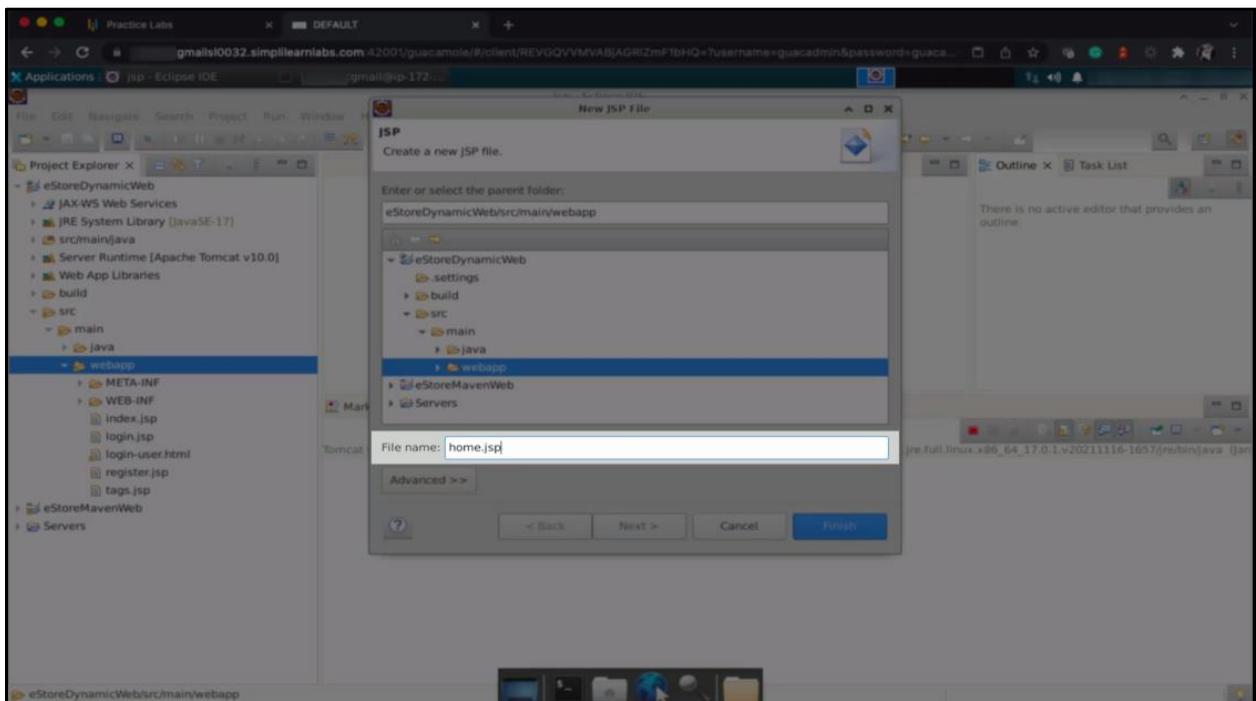
## 1.2 Open the eStoreDynamicWeb project



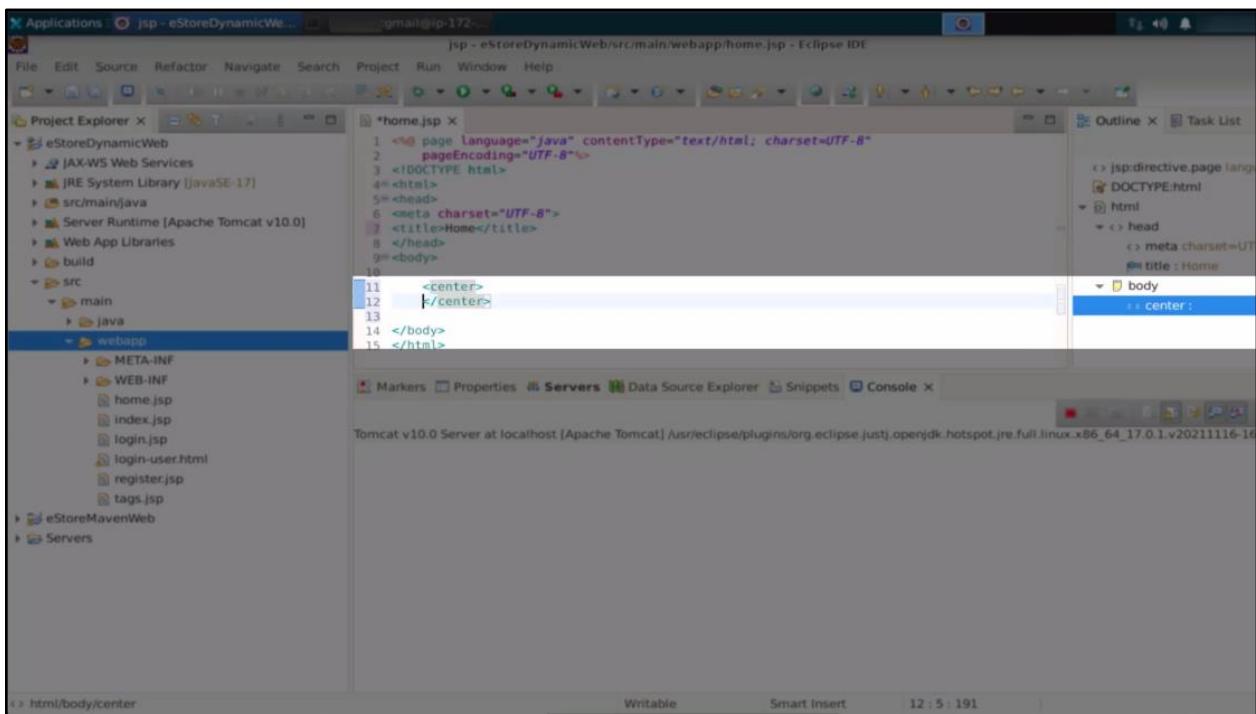
**Note:** Please refer to the previous demo on how to create the **eStoreDynamicWeb** project

## 1.3 Create a new JSP file and name the file **home.jsp**





#### 1.4 Inside the JSP file, create a <center> tag



### 1.5 Inside the <center> tag, create an <h3> tag with Welcome to Home text

```

<%@ page language="java" contentType="text/html; charset=UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Home</title>
</head>
<body>
<center>
<h3>Welcome to Home</h3>
</center>
</body>

```

### 1.6 Then, add a <jsp:forward> tag to forward the request to home.jsp

```

<%@page import="java.sql.ResultSet"%>
<%@page import="com.estore.db.DB"%>
<%@page import="java.sql.PreparedStatement"%>
<%@page import="com.estore.model.User"%>
<%@page language="java" contentType="text/html; charset=UTF-8"%>
<%@page pageEncoding="UTF-8"%>

User user = new User();
user.email = request.getParameter("txtEmail");
user.password = request.getParameter("txtPassword");

DB db = DB.getDB();

String sql = "select * from Users where email = ? and password = ?";
PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
preparedStatement.setString(1, user.email);
preparedStatement.setString(2, user.password);

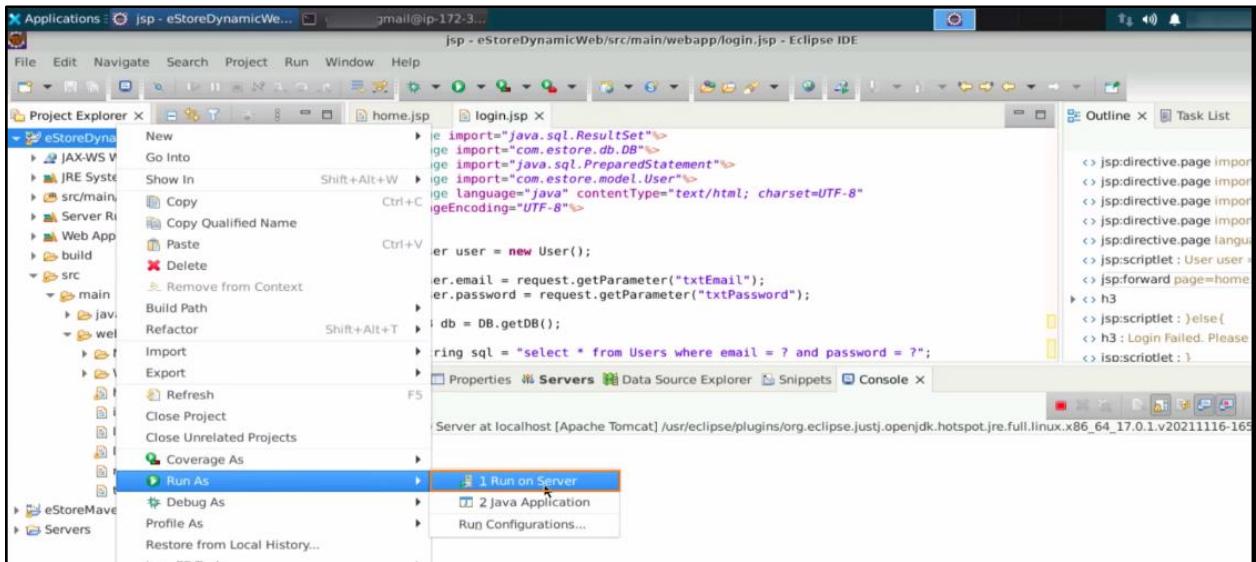
ResultSet set = db.executeQuery(preparedStatement);

if(set != null && set.next()){
    user.name = set.getString("name"); // we are getting the name form the DB
}

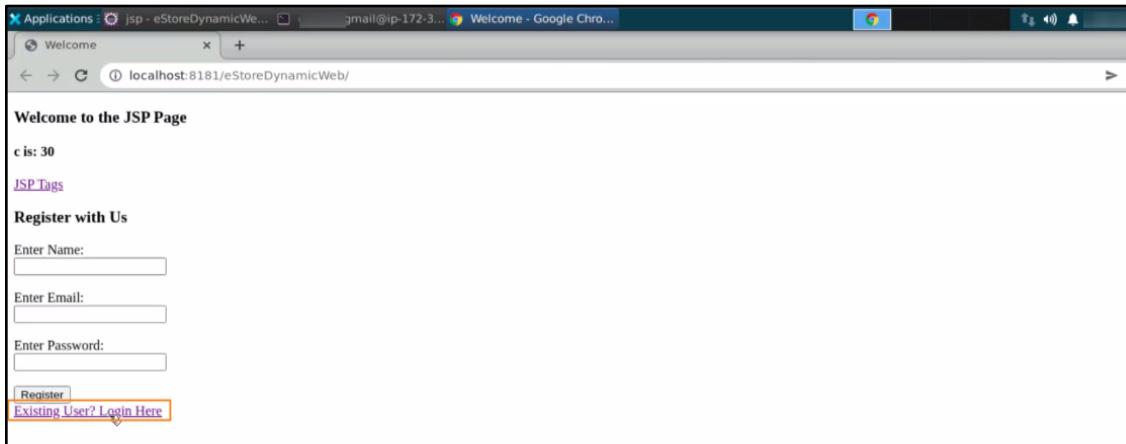
<jsp:forward page="home.jsp"/>
<h3>Thank Your For Logging in: <=> user.name <=></h3>
<h3>Login Failed. Please try again!</h3>

```

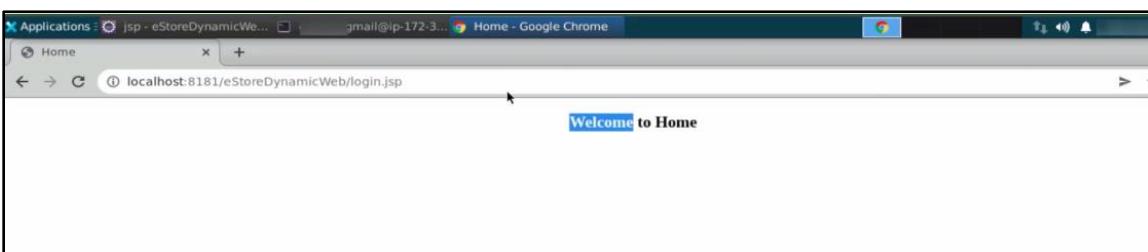
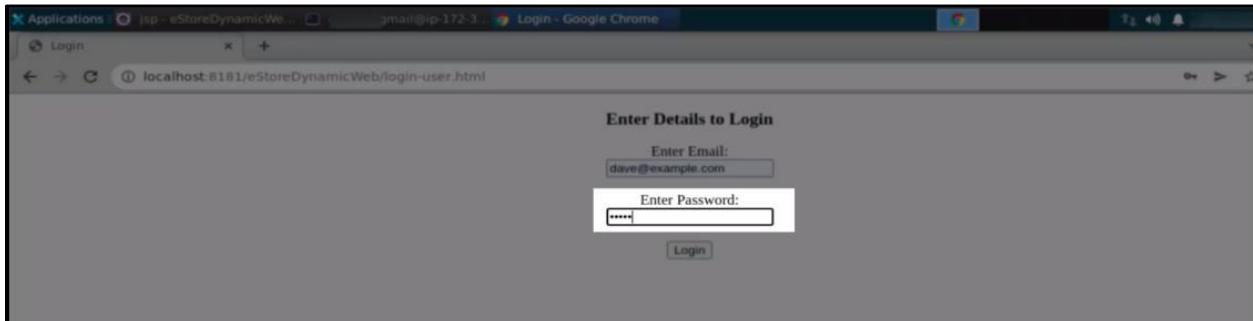
## 1.7 Save the file and run the code



## 1.8 Click on Existing User? Login Here



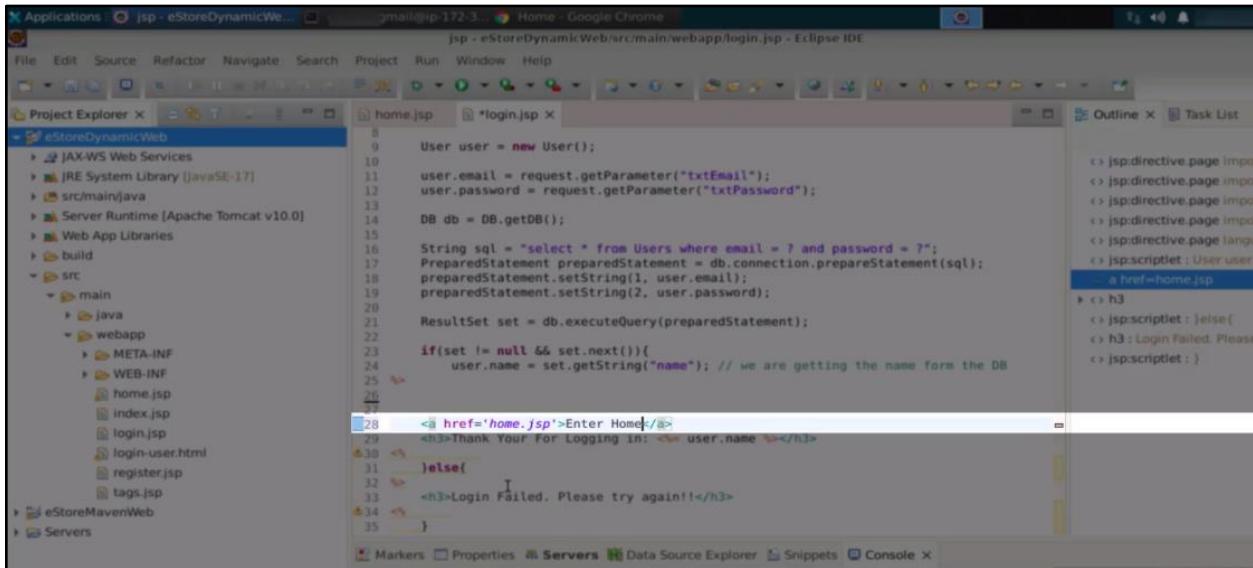
1.9 Enter the email address and password and click **Login**



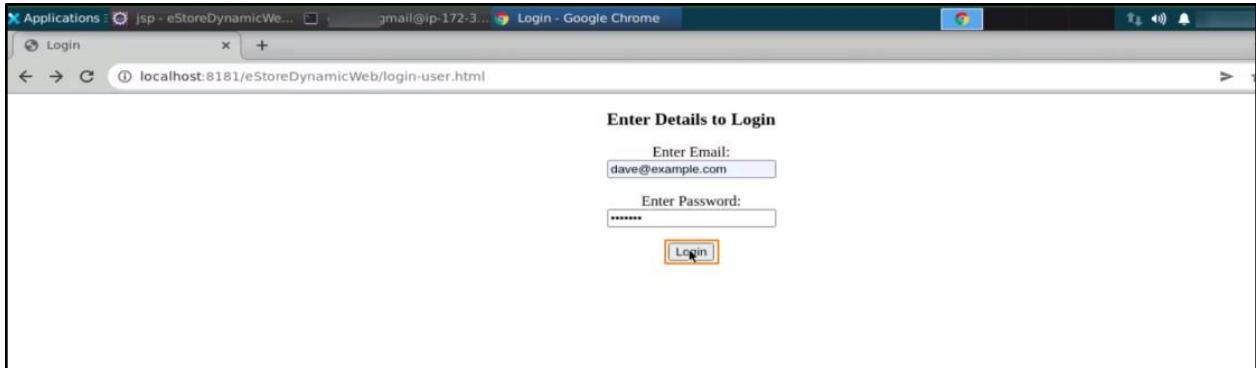
You can see the output as **Welcome to Home.**

## **Step 2: Forward the page to the home.jsp page**

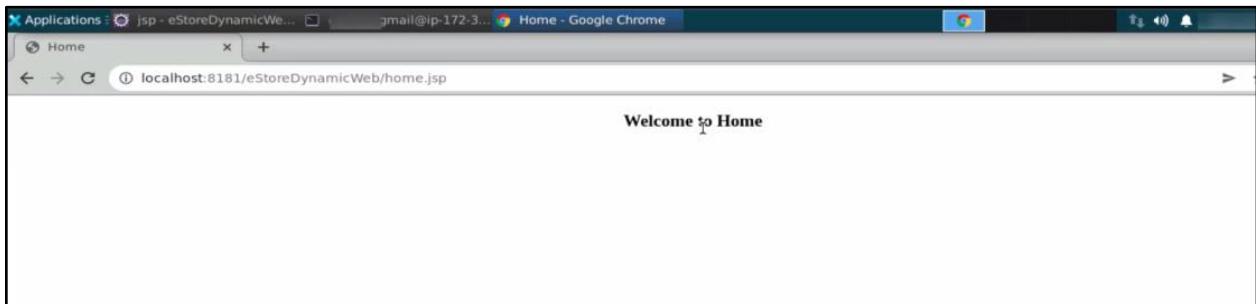
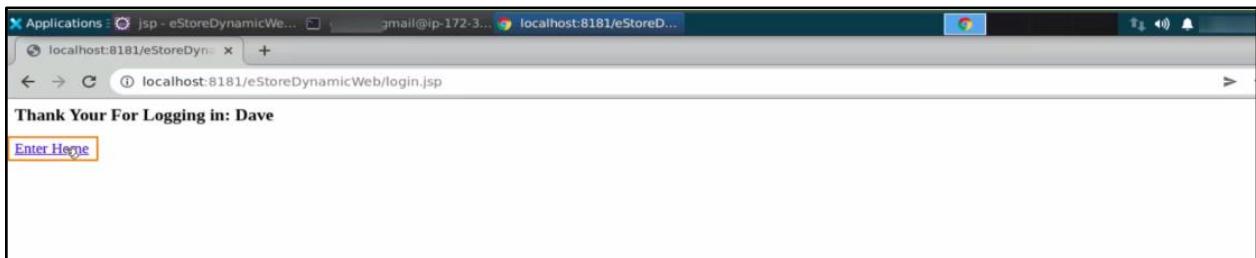
2.1 In the **login.jsp** file, use an anchor tag **<a>** to forward the page to the **home.jsp** page and set the **href** attribute of the anchor tag to **home.jsp**



2.2 Go back to the browser and click on the **Login** link



2.3 Click on the **Enter Home** link



You will see the output **Welcome to Home** displayed on the page.

## Step 3: Add cookies in the login.jsp

3.1 In the **login.jsp** file, add cookies to store user login information. This can be done using JSP scriptlets or JSP tags.

jsp - eStoreDynamicWeb/src/main/webapp/login.jsp - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

home.jsp login.jsp x

```
1 <%@page import="java.sql.ResultSet"%>
2 <%@page import="com.estore.db.DB"%>
3 <%@page import="java.sql.PreparedStatement"%>
4 <%@page import="com.estore.model.User"%>
5 <%@page language="java" contentType="text/html; charset=UTF-8"
6 pageEncoding=UTF-8%>
7<%>
8
9 User user = new User();
10
11 user.email = request.getParameter("txtEmail");
12 user.password = request.getParameter("txtPassword");
13
14 DB db = DB.getDB();
15
16 String sql = "select * from Users where email = ? and password = ?";
17 PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
18 preparedStatement.setString(1, user.email);
19 preparedStatement.setString(2, user.password);
20
21 ResultSet set = db.executeQuery(preparedStatement);
22
23 if(set != null && set.next()){
24     user.name = set.getString("name"); // we are getting the name form the DB
25
26     // Cookies
27     Cookie cookie1 = new Cookie("keyName", user.name);
28     Cookie cookie2 = new Cookie("keyEmail", user.email);
29
30     response.addCookie(cookie1);
31     response.addCookie(cookie2);
32 }
33
34 <h3>Thank Your For Logging in: <%> user.name <%></h3>
35 <a href='home.jsp'>Enter Home</a>
36<%>
37 }else{
38 }
```

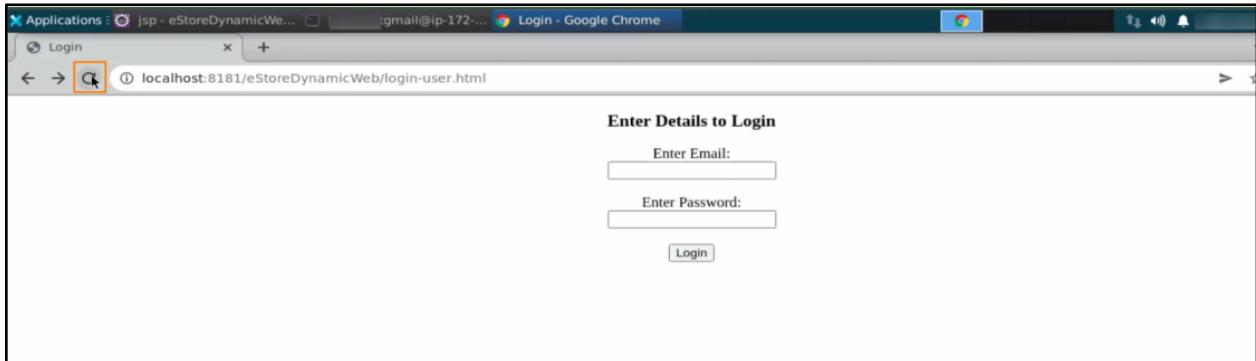
3.2 Navigate back to the **home.jsp** page and add a tag to display the cookies. Use an expression tag `<%= %>` to retrieve and display the cookie values.

Jsp - eStoreDynamicWeb/src/main/webapp/home.jsp - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

```
home.jsp x login.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2 pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4<html>
5<head>
6 <meta charset="UTF-8">
7 <title>Home</title>
8 </head>
9<body>
10
11<center>
12   <h3>Welcome to Home</h3>
13
14
15   Cookie[] cookies = request.getCookies();
16   for(Cookie cookie : cookies){
17
18     cookie.getName() | cookie.getValue() <br>
19
20   }
21
22
23
24 </center>
25
26 </body>
27 </html>
```

### 3.3 Refresh the page in the browser

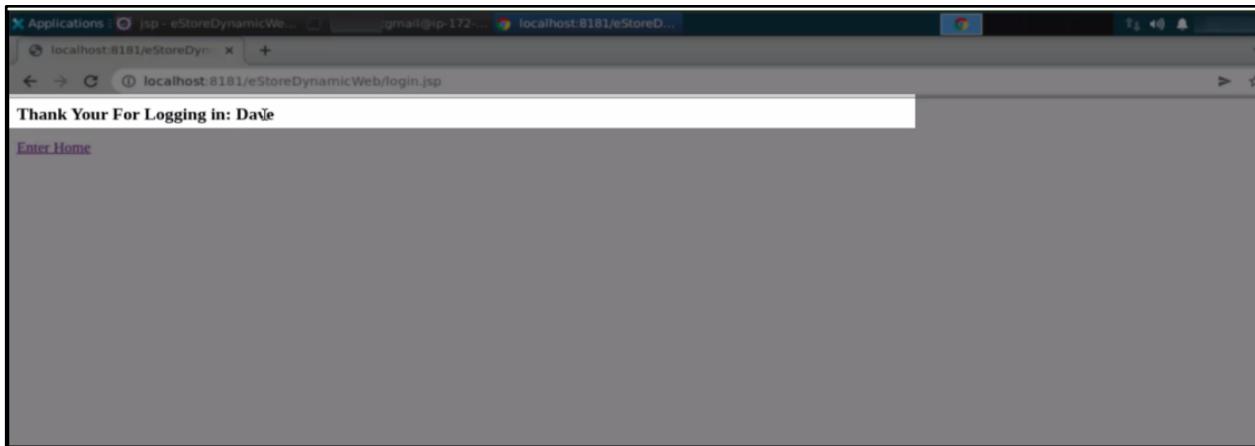


### 3.4 Enter your email address and password in the Login form



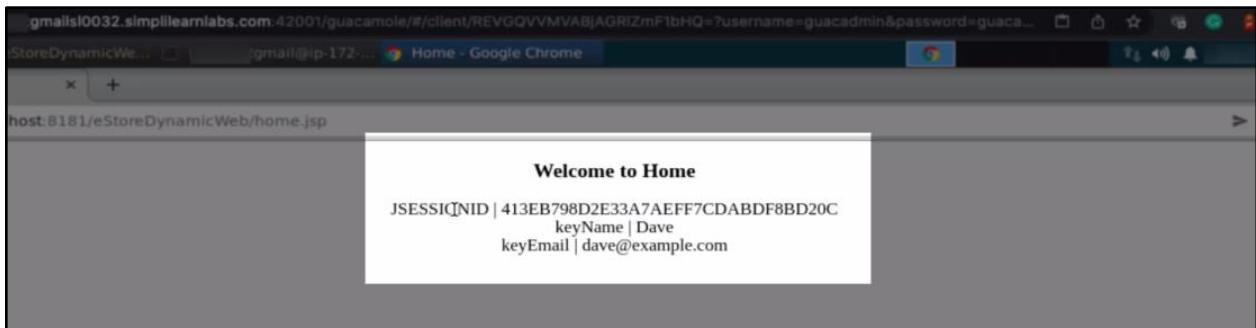
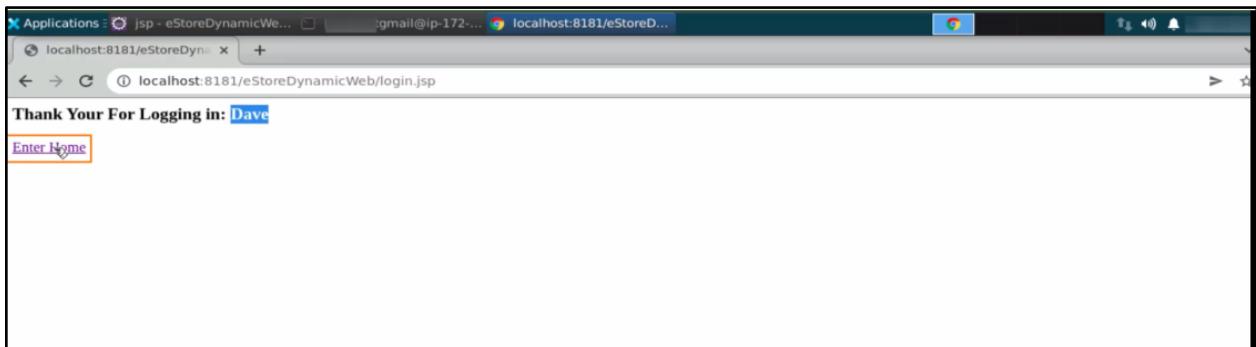
### 3.5 Click on the **Login** button





You will see the output **Thank You for Logging in, [Username]** displayed on the page, where **[Username]** is the name retrieved from the cookie.

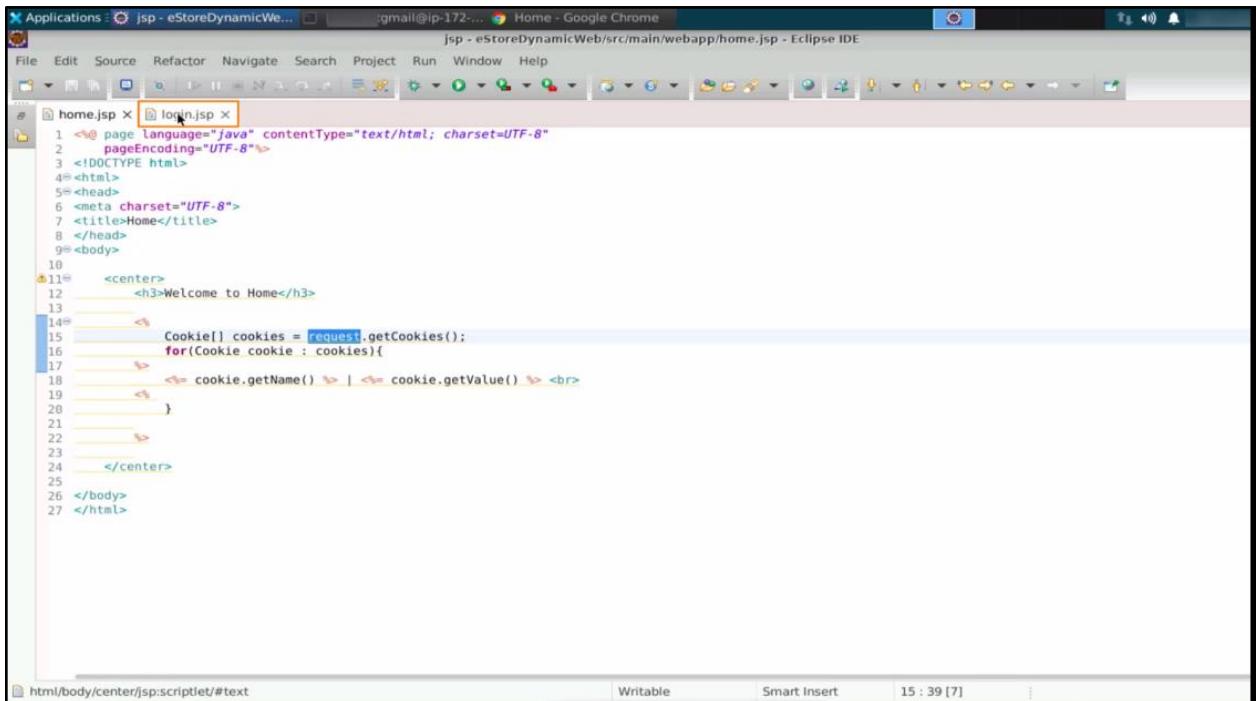
### 3.6 Click the **Enter Home** link



You will see the user details displayed on the page.

## Step 4: Navigate to login.jsp and add attributes

### 4.1 Go back to Eclipse IDE and open the login.jsp file



The screenshot shows the Eclipse IDE interface with the title bar "Applications : Jsp - eStoreDynamicWe...". The workspace contains two files: "home.jsp" and "login.jsp". The "login.jsp" file is currently selected and its content is displayed in the editor:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Home</title>
</head>
<body>
<center>
<h3>Welcome to Home</h3>
<br>
Cookie[] cookies = request.getCookies();
for(Cookie cookie : cookies){
<br>
    cookie.getName() | cookie.getValue() <br>
}
</center>
</body>
</html>
```

The code includes Java scriptlets and HTML tags. A blue selection bar highlights the loop from line 14 to line 20. The status bar at the bottom shows "html/body/center/jsp:scriptlet/#text", "Writable", "Smart Insert", and "15 : 39 [7]".

### 4.2 Modify the anchor tag used for forwarding to include attributes for the user's name and email

The screenshot shows the Eclipse IDE interface with two files open:

- home.jsp**:  
A JSP file containing Java code. It includes imports for User, DB, and PreparedStatement. It retrieves email and password from request parameters, initializes a User object, and executes a database query to find a matching user. If found, it sets session attributes for name and email, creates two cookies (keyName and keyEmail), and adds them to the response. If no user is found, it displays an error message.

```
# home.jsp  login.jsp x
6  pageEncoding="UTF-8">
7<%>
8  User user = new User();
9  user.email = request.getParameter("txtEmail");
10 user.password = request.getParameter("txtPassword");
11
12 DB db = DB.getDB();
13
14 String sql = "select * from Users where email = ? and password = ?";
15 PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
16 preparedStatement.setString(1, user.email);
17 preparedStatement.setString(2, user.password);
18
19 ResultSet set = db.executeQuery(preparedStatement);
20
21 if(set != null && set.next()){
22     user.name = set.getString("name"); // we are getting the name from the DB
23
24     // Cookies
25     Cookie cookie1 = new Cookie("keyName", user.name);
26     Cookie cookie2 = new Cookie("keyEmail", user.email);
27
28     response.addCookie(cookie1);
29     response.addCookie(cookie2);
30 }
31
32 >>
33
34 <h3>Thank You For Logging in: <%=user.name%</h3>
35 <a href='home.jsp?name=<%=user.name%>&email=<%=user.email%>'>Enter Home</a>
36
37 }else{
38
39     <h3>Login Failed. Please try again!!</h3>
40 }
41
42
43 >
```
- login.jsp**:  
A JSP file containing a form for logging in. It has input fields for email and password, and a submit button.

```
<form action="login.jsp" method="post">
    Email: <input type="text" name="txtEmail" />
    Password: <input type="password" name="txtPassword" />
    <br/>
    <input type="submit" value="Log In" />

```

These attributes can be passed as query parameters in the URL.

#### 4.3 Set the response type of the anchor tag to **String** using the **response** attribute

```

    6  pageEncoding="UTF-8">
    7<%>
    8  User user = new User();
    9  user.email = request.getParameter("txtEmail");
   10 user.password = request.getParameter("txtPassword");
   11 DB db = DB.getDB();
   12 String sql = "select * from Users where email = ? and password = ?";
   13 PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
   14 preparedStatement.setString(1, user.email);
   15 preparedStatement.setString(2, user.password);
   16 ResultSet set = db.executeQuery(preparedStatement);
   17 if(set != null && set.next()){
   18     user.name = set.getString("name"); // we are getting the name form the DB
   19
   20     // Cookies
   21     Cookie cookie1 = new Cookie("keyName", user.name);
   22     Cookie cookie2 = new Cookie("keyEmail", user.email);
   23
   24     response.addCookie(cookie1);
   25     response.addCookie(cookie2);
   26
   27     String response = "<a href='home.jsp?name="+user.name+"&email="+user.email+">Enter Home</a>" +
   28         "<br>";
   29     out.print(response);
   30
   31     <h3>Thank Your For Logging in: <b><%= user.name %></b></h3>
   32     <a href='home.jsp?name=<%=user.name%>&email=<%=user.email%>'>Enter Home</a>
   33 }
   34 else{
   35     <h3>Login Failed. Please try again!!</h3>
   36 }
   37
   38 <h3>Thank Your For Logging in: <b><%= user.name %></b></h3>
   39 <a href='home.jsp?name=<%=user.name%>&email=<%=user.email%>'>Enter Home</a>
   40 }
   41
   42 <h3>Login Failed. Please try again!!</h3>
   43

```

#### 4.4 Add a print statement to print the **response** on the console

```

    6  pageEncoding="UTF-8">
    7<%>
    8  User user = new User();
    9  user.email = request.getParameter("txtEmail");
   10 user.password = request.getParameter("txtPassword");
   11 DB db = DB.getDB();
   12 String sql = "select * from Users where email = ? and password = ?";
   13 PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
   14 preparedStatement.setString(1, user.email);
   15 preparedStatement.setString(2, user.password);
   16 ResultSet set = db.executeQuery(preparedStatement);
   17 if(set != null && set.next()){
   18     user.name = set.getString("name"); // we are getting the name form the DB
   19
   20     // Cookies
   21     Cookie cookie1 = new Cookie("keyName", user.name);
   22     Cookie cookie2 = new Cookie("keyEmail", user.email);
   23
   24     response.addCookie(cookie1);
   25     response.addCookie(cookie2);
   26
   27     String response = "<a href='home.jsp?name="+user.name+"&email="+user.email+">Enter Home</a>" +
   28         "<br>";
   29     out.print(response);
   30
   31     <h3>Thank Your For Logging in: <b><%= user.name %></b></h3>
   32     <a href='home.jsp?name=<%=user.name%>&email=<%=user.email%>'>Enter Home</a>
   33 }
   34 else{
   35     <h3>Login Failed. Please try again!!</h3>
   36 }
   37
   38 <h3>Thank Your For Logging in: <b><%= user.name %></b></h3>
   39 <a href='home.jsp?name=<%=user.name%>&email=<%=user.email%>'>Enter Home</a>
   40 }
   41
   42 <h3>Login Failed. Please try again!!</h3>
   43

```

4.5 Go back to the browser and refresh the page

The screenshot shows a Google Chrome window with a login form. The title bar says 'Applications : jsp - eStoreDynamicWe... : gmail@ip-172-... Login - Google Chrome'. The address bar shows 'localhost:8181/eStoreDynamicWeb/login-user.html'. The form is titled 'Enter Details to Login'. It contains two text input fields: 'Enter Email:' with the value 'dave@example.com' and 'Enter Password:' with the value '\*\*\*\*\*'. A 'Login' button is at the bottom.

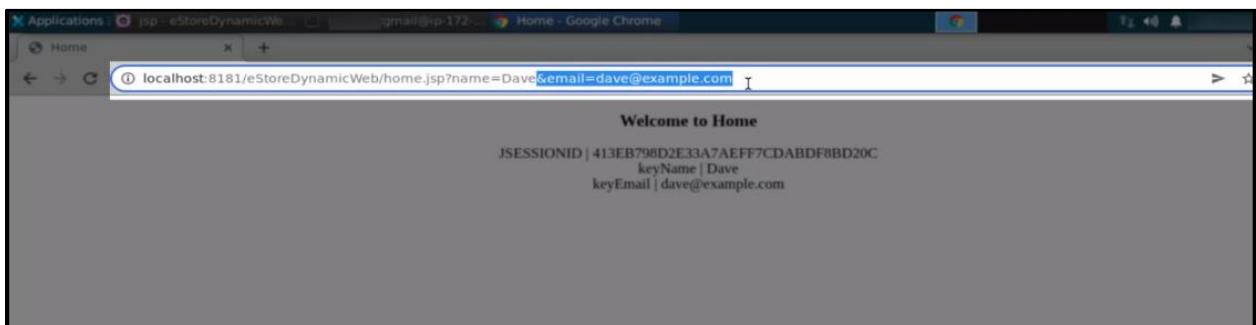
4.6 Enter your email address and password in the login form

The screenshot shows the same Google Chrome window with the login form. The 'Login' button at the bottom of the form is now highlighted with a yellow border, indicating it is the active button.

4.7 Click the **Login** button

The screenshot shows the same Google Chrome window after the 'Login' button was clicked. The page has refreshed, and the 'Login' button is no longer highlighted. The rest of the form and browser interface remain the same.

#### 4.8 Click on the Enter Home link



You will see the email ID passed in the URL.

### Step 5: Write an operation to read the data from the URL

#### 5.1 Go back to the **home.jsp** page

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
%>
<!DOCTYPE html>
<meta charset="UTF-8">
<title>Home</title>
<body>
<center>
<h3>Welcome to Home</h3>
<br>
Cookie[] cookies = request.getCookies();
for(Cookie cookie : cookies){
    cookie.getName() | cookie.getValue() <br>
}
</center>
</body>
</html>

```

5.2 Write an operation to read the data from the URL using JSP scriptlets or JSP tags.

Retrieve the query parameters from the URL and display them on the page

```

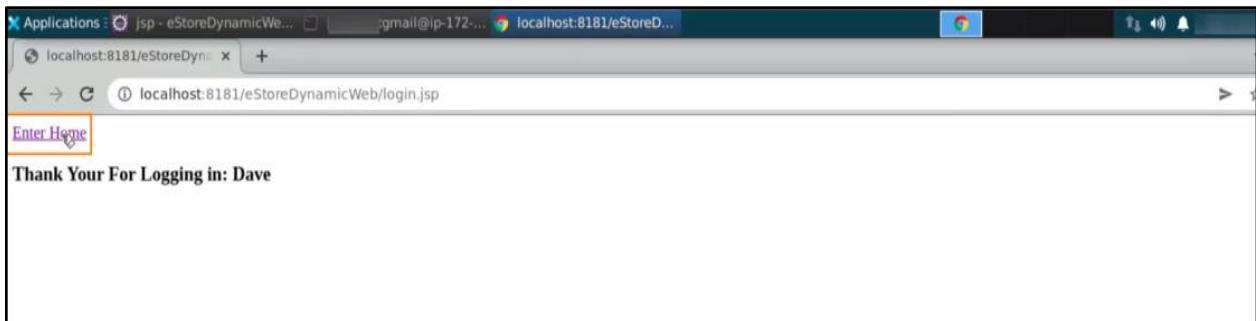
<%@ page language="java" contentType="text/html; charset=UTF-8"
%>
<!DOCTYPE html>
<meta charset="UTF-8">
<title>Home</title>
<body>
<center>
<h3>Welcome to Home</h3>
<br>
Cookie[] cookies = request.getCookies();
for(Cookie cookie : cookies){
    cookie.getName() | cookie.getValue() <br>
}
<br>
String name = request.getParameter("name");
String email = request.getParameter("email");
<br>
Name: <%= name %> <br>
Email: <%= email %>
</center>
</body>
</html>

```

### 5.3 Go back and click the **Login** link

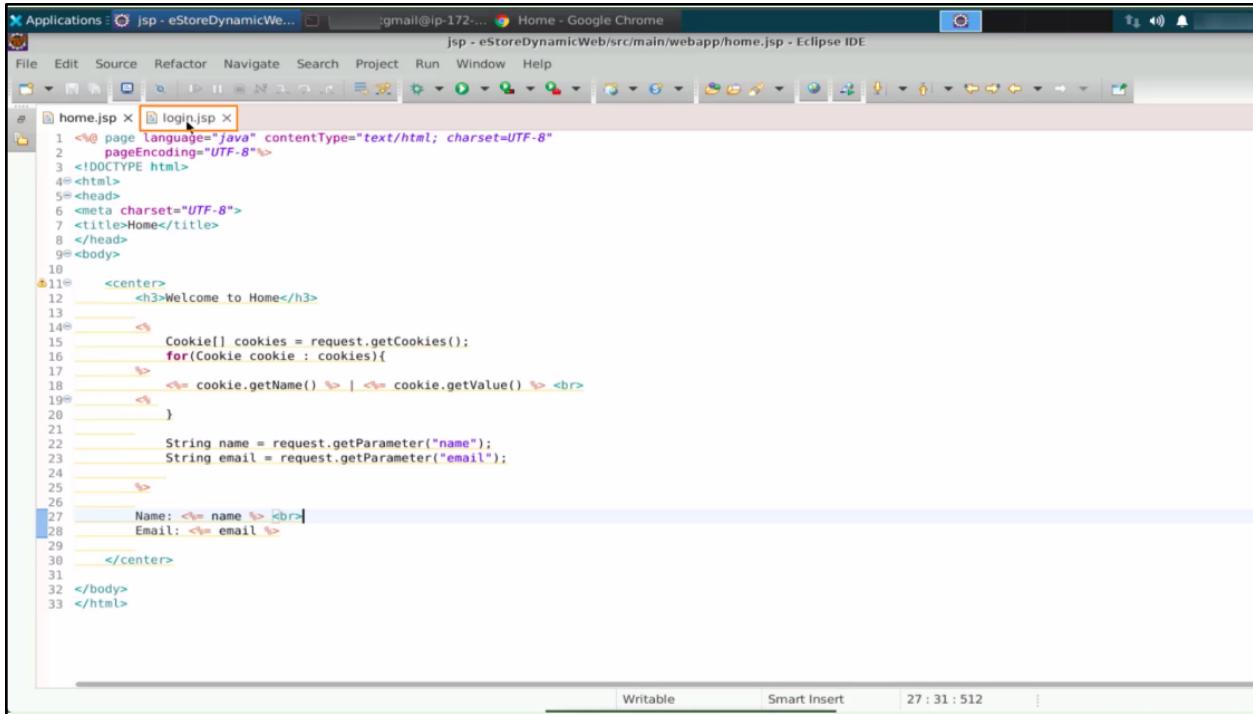
A screenshot of a Google Chrome browser window. The address bar shows the URL `localhost:8181/eStoreDynamicWeb/login-user.htm`. The page title is "Login". The content area displays a form titled "Enter Details to Login" with fields for "Enter Email:" containing `dave@example.com` and "Enter Password:" containing `*****`. A "Login" button is at the bottom, with a mouse cursor hovering over it.

### 5.4 Click the **Enter Home** link



You will see the user credentials displayed in the URL.

## 5.5 Navigate back to the project and go to the **login.jsp** file



The screenshot shows the Eclipse IDE interface with the "login.jsp" file open in the editor. The file contains JSP code with some lines highlighted in blue, indicating they are selected or part of a block. The code includes Java scriptlets and JSTL tags for reading cookies and displaying user information.

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2 %>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Home</title>
8 </head>
9 <body>
10
11 <center>
12 <h3>Welcome to Home</h3>
13
14
15     Cookie[] cookies = request.getCookies();
16     for(Cookie cookie : cookies){
17
18         cookie.getName() | cookie.getValue() <br>
19
20     }
21
22     String name = request.getParameter("name");
23     String email = request.getParameter("email");
24
25
26     Name: <%= name %<br>
27     Email: <%= email %>
28
29
30 </center>
31
32 </body>
33 </html>
```

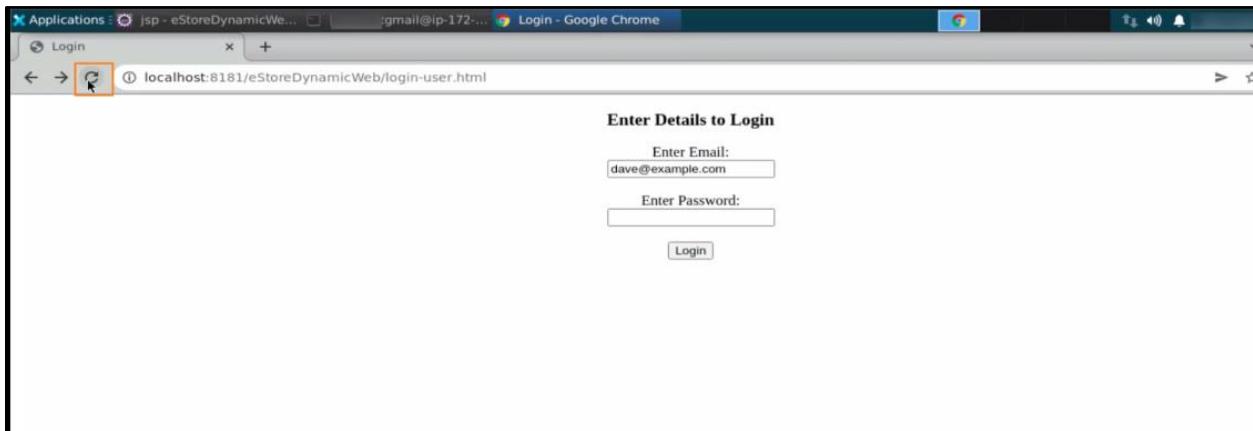
## 5.6 Create another response object

```

9     User user = new User();
10    user.email = request.getParameter("txtEmail");
11    user.password = request.getParameter("txtPassword");
12
13    DB db = DB.getDB();
14
15    String sql = "select * from Users where email = ? and password = ?";
16    PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
17    preparedStatement.setString(1, user.email);
18    preparedStatement.setString(2, user.password);
19
20    ResultSet set = db.executeQuery(preparedStatement);
21
22    if(set != null && set.next()){
23        user.name = set.getString("name"); // we are getting the name from the DB
24
25        // Cookies
26        Cookie cookie1 = new Cookie("keyName", user.name);
27        Cookie cookie2 = new Cookie("keyEmail", user.email);
28
29        response.addCookie(cookie1);
30        response.addCookie(cookie2);
31
32        //String res = "<a href='home.jsp?name="+user.name+"&email='"+user.email+">Enter Home</a>";
33        String res = "<form action='home.jsp' method='post'>
34            +<input type='hidden' name='name' value='"+user.name+"' /><br>
35            +<input type='hidden' name='email' value='"+user.email+"' /><br>
36            +<input type='submit' value='Enter Home' /><br>
37        +</form>";
38
39        out.print(res);
40
41    <h3>Thank Your For Logging in: <%= user.name %</h3>
42
43
44    }else{
45
46

```

5.7 Go back to the browser and refresh the page



### 5.8 Enter your email address and password in the login form

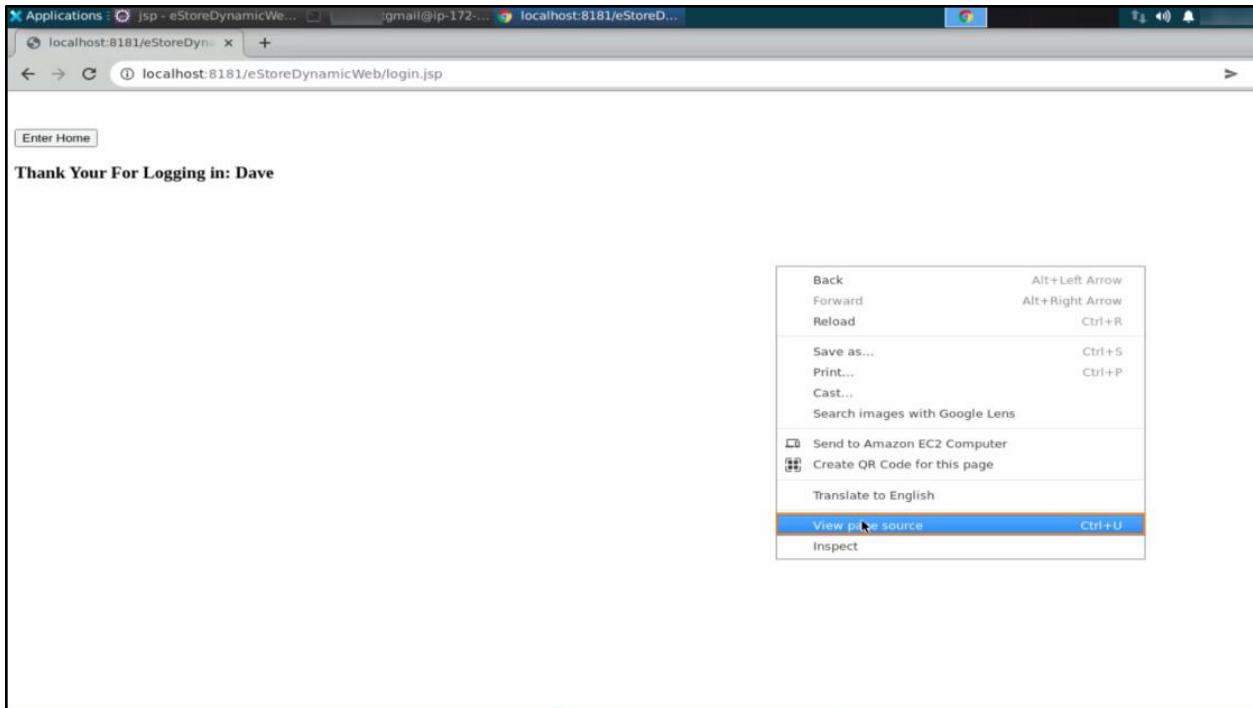
The screenshot shows a Google Chrome window with the title bar "Applications : jsp - eStoreDynamicWe...". The address bar shows "localhost:8181/eStoreDynamicWeb/login-user.html". The main content area has a heading "Enter Details to Login". It contains two input fields: "Enter Email:" with the value "dave@example.com" and "Enter Password:" with the value "\*\*\*\*\*". Below the fields is a "Login" button.

### 5.9 Click the **Login** button

The screenshot shows a Google Chrome window with the title bar "Applications : jsp - eStoreDynamicWe...". The address bar shows "localhost:8181/eStoreDynamicWeb/login-user.html". The main content area has a heading "Enter Details to Login". It contains two input fields: "Enter Email:" with the value "dave@example.com" and "Enter Password:" with the value "\*\*\*\*\*". The "Login" button is highlighted with a yellow border.

### 5.10 Click the **Enter Home** link

The screenshot shows a Google Chrome window with the title bar "Applications : jsp - eStoreDynamicWe...". The address bar shows "localhost:8181/eStoreDynamicWeb/login.jsp". The main content area displays the message "Thank You For Logging in: Dave". Below the message is a button labeled "Enter Home", which is highlighted with a yellow border.

5.11 Right-click the page and select **View page source**

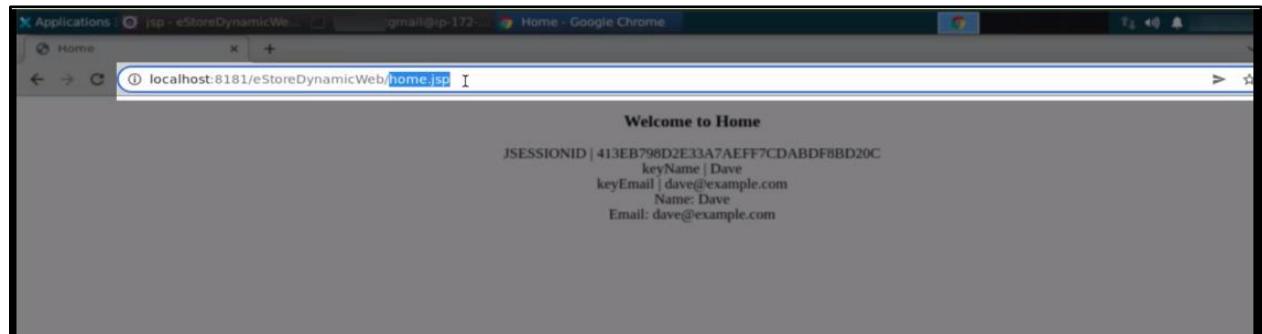
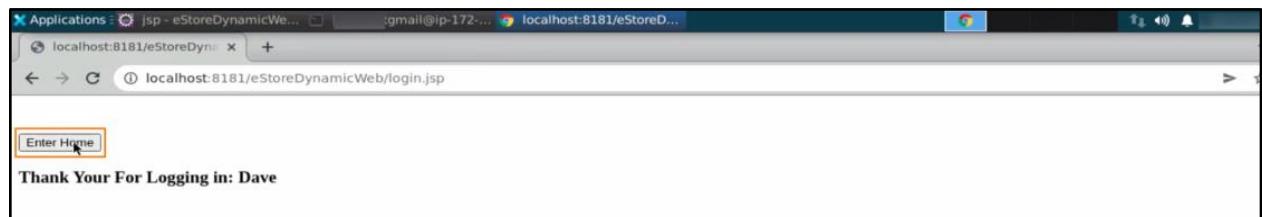
```

1
2
3
4
5
6
7
8
9
10
<form action='home.jsp' method='post'><input type='hidden' name='name' value='Dave' /><br><input type='hidden' name='email' value='dave@example.com' /><br><input type='submit' value='Enter Home' />
<h3>Thank Your For Logging in: Dave</h3>

```

You will see the email ID displayed as the value in the HTML source code.

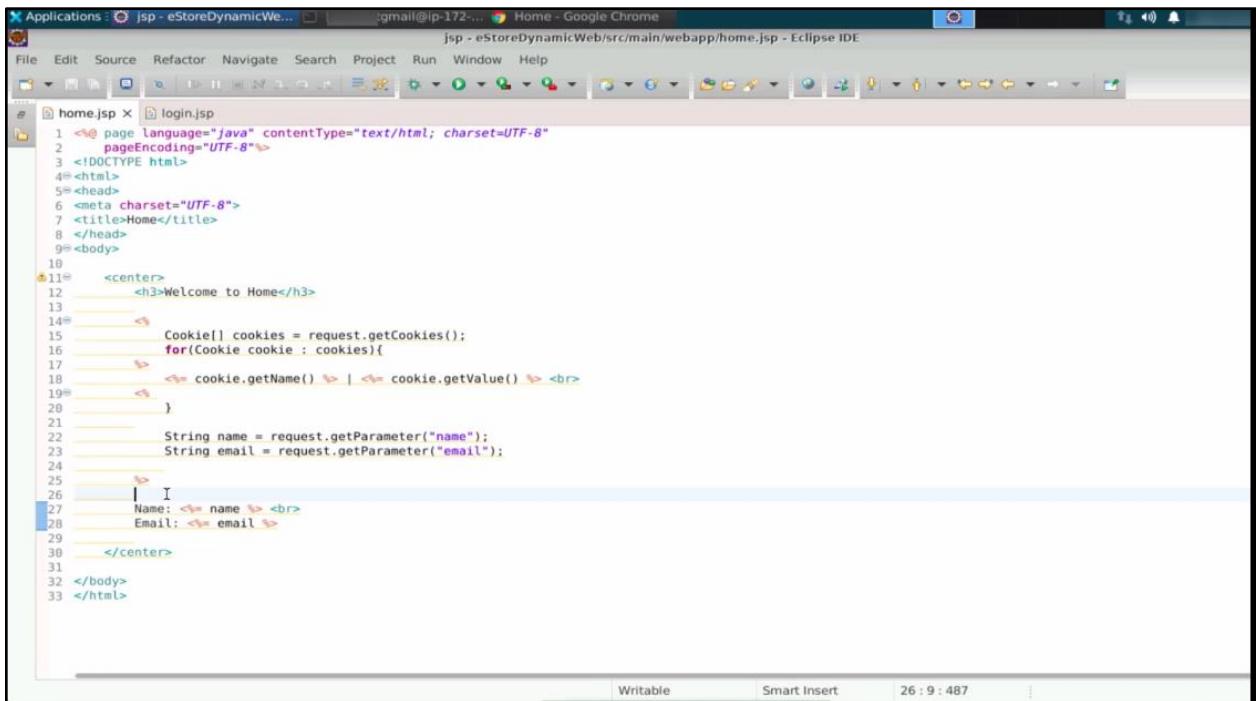
### 5.1 Go back and click on the **Enter Home** link again



You will see **home.jsp** in the URL.

## Step 6: Write an operation for hidden field data

### 6.1 Go back to the **home.jsp** page



The screenshot shows the Eclipse IDE interface with the **home.jsp** file open in the editor. The code is as follows:

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2 pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Home</title>
8 </head>
9 <body>
10
11 <center>
12   <h3>Welcome to Home</h3>
13
14
15   Cookie[] cookies = request.getCookies();
16   for(Cookie cookie : cookies){
17     cookie.getName() | cookie.getValue() <br>
18   }
19
20
21   String name = request.getParameter("name");
22   String email = request.getParameter("email");
23
24
25   Name: <input type="text" value="<%= name %>"> <br>
26   Email: <input type="text" value="<%= email %>">
27
28
29 </center>
30
31
32 </body>
33 </html>
```

### 6.2 Write an operation to read and display the data from the hidden field in the form

```

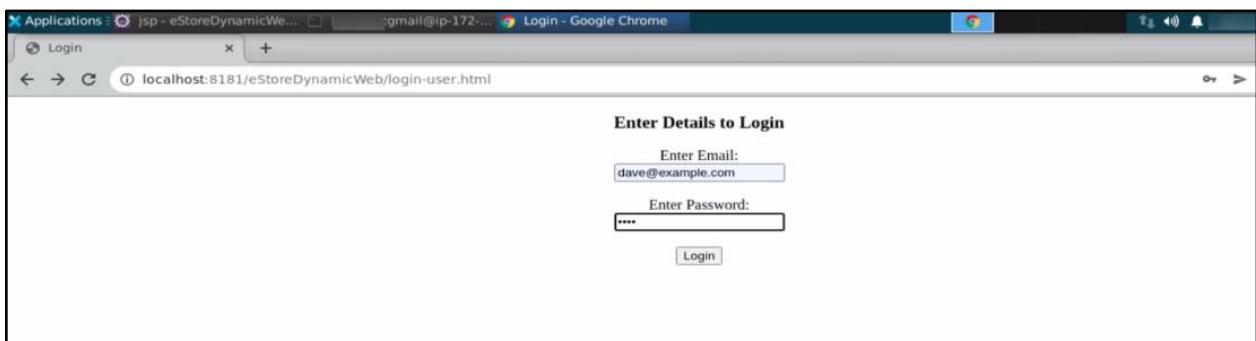
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2 pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Home</title>
8 </head>
9 <body>
10
11 <center>
12 <h3>Welcome to Home</h3>
13
14 <%
15 Cookie[] cookies = request.getCookies();
16 for(Cookie cookie : cookies){
17
18 <%= cookie.getName() %> | <%= cookie.getValue() %> <br>
19
20 }
21
22 String name = request.getParameter("name");
23 String email = request.getParameter("email");
24
25 String txtName = request.getParameter("txtName");
26 String txtEmail = request.getParameter("txtEmail");
27
28 <h3>URL Rewriting</h3>
29 Name: <%= name %> <br>
30 Email: <%= email %>
31
32 <h3>Hidden From Filed Data</h3>
33 Name: <%= txtName %> <br>
34 Email: <%= txtEmail %>
35
36
37 </center>
38

```

6.3 Go back to the browser and refresh the page



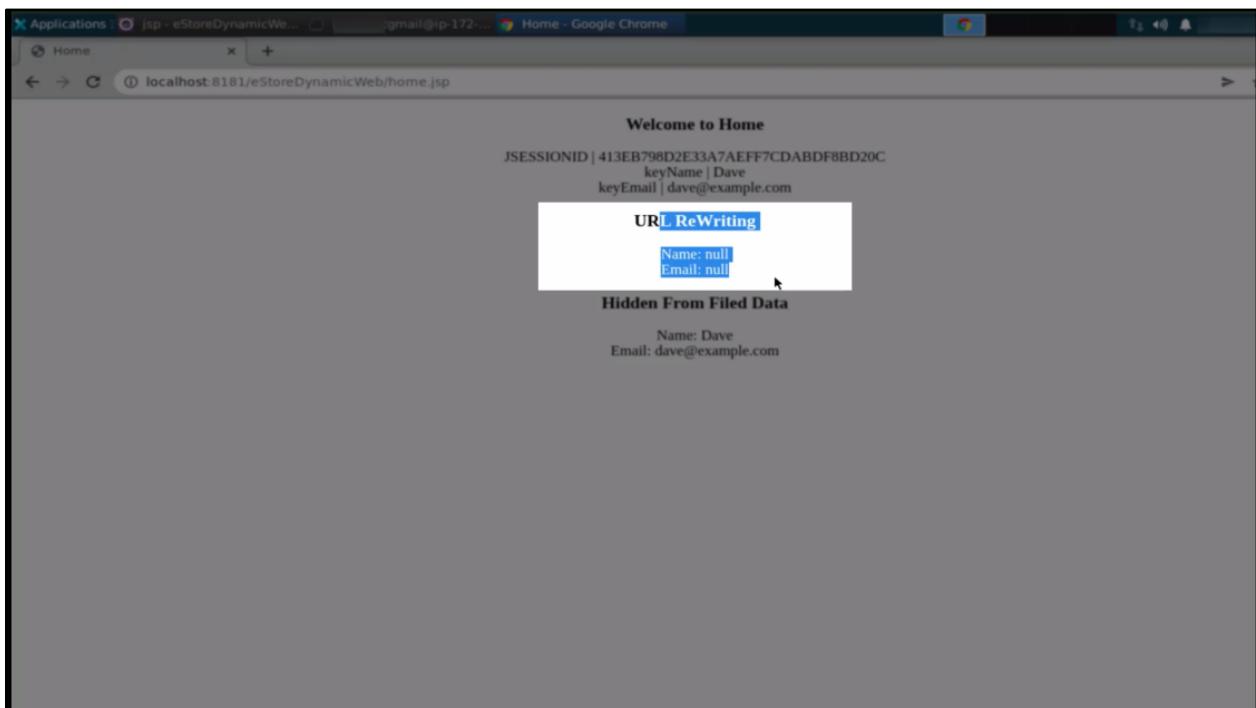
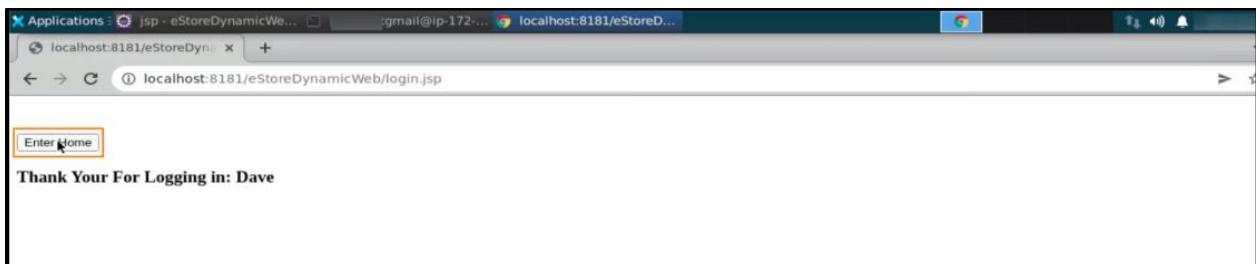
6.4 Enter your email address and password in the login form



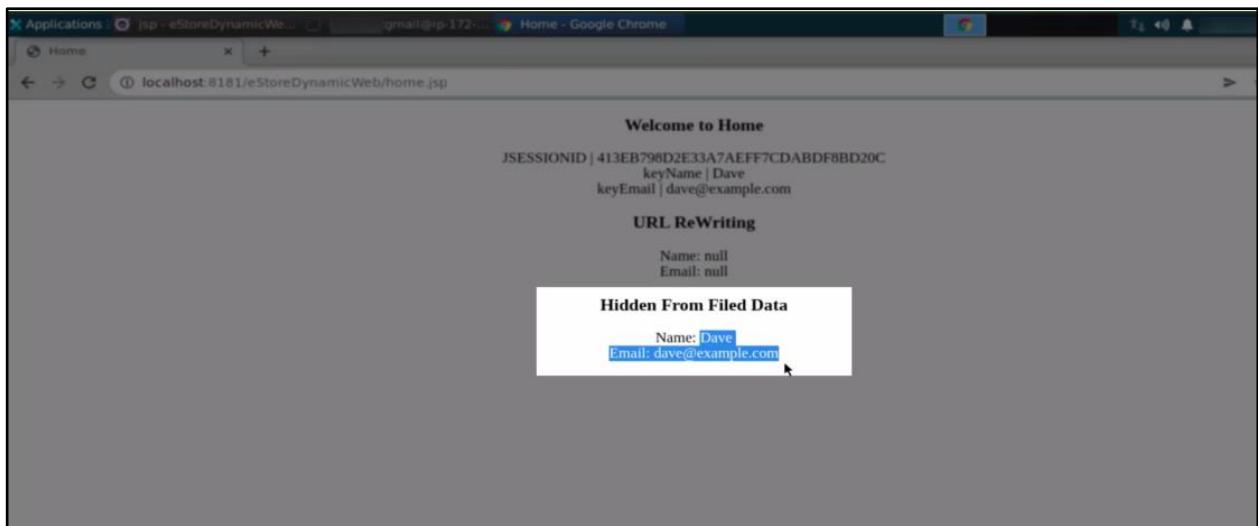
## 6.5 Click on the **Login** button

The screenshot shows a web browser window with the title bar "Applications : jsp - eStoreDynamicWe...". The address bar shows "localhost:8181/eStoreDynamicWeb/login-user.html". The main content area is titled "Enter Details to Login". It contains two input fields: "Enter Email:" with the value "dave@example.com" and "Enter Password:" with the value "\*\*\*\*\*". Below these fields is a "Login" button.

## 6.6 Click on the **Enter Home** link



Observe if there is any data in the URL rewriting.



You will see the data printed from the hidden field.

## Step 7: Write an operation for the HTTP session object

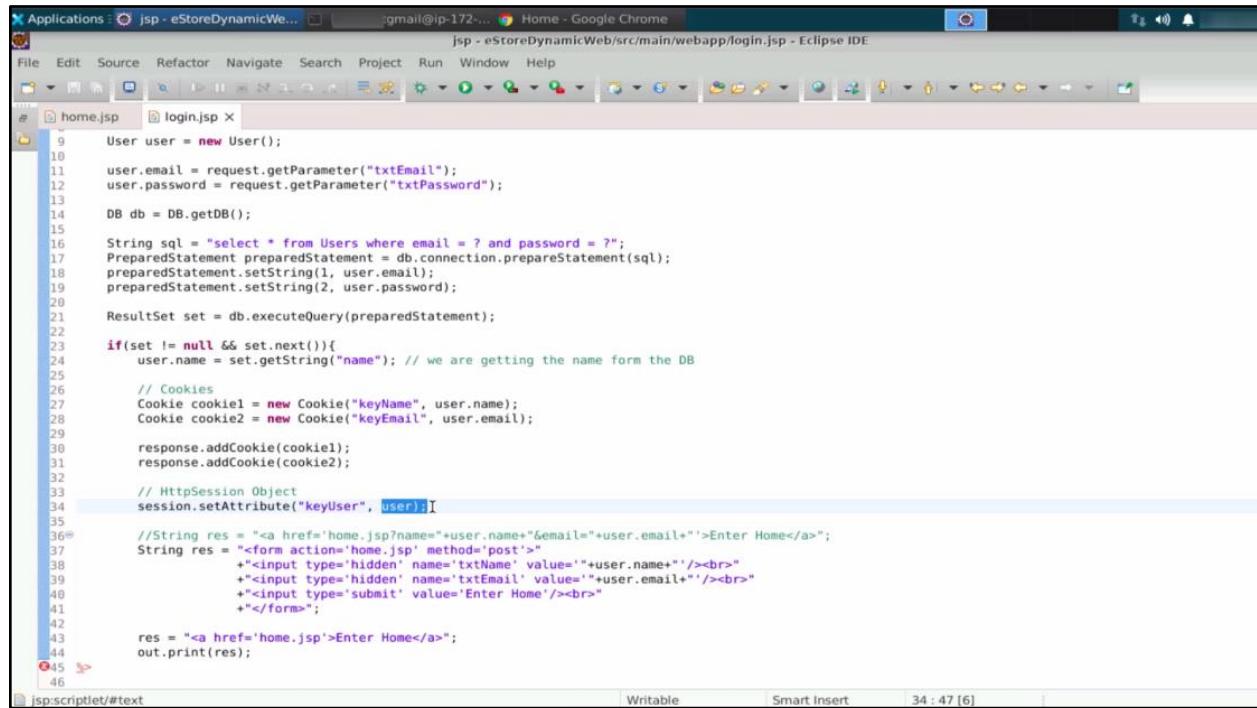
### 7.1 Go back to the login.jsp page

```

File Edit Source Refactor Navigate Search Project Run Window Help
File Edit Source Refactor Navigate Search Project Run Window Help
home.jsp login.jsp
9     User user = new User();
10    user.email = request.getParameter("txtEmail");
11    user.password = request.getParameter("txtPassword");
12
13    DB db = DB.getDB();
14
15    String sql = "select * from Users where email = ? and password = ?";
16    PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
17    preparedStatement.setString(1, user.email);
18    preparedStatement.setString(2, user.password);
19
20    ResultSet set = db.executeQuery(preparedStatement);
21
22    if(set != null && set.next()){
23        user.name = set.getString("name"); // we are getting the name from the DB
24
25        // Cookies
26        Cookie cookie1 = new Cookie("keyName", user.name);
27        Cookie cookie2 = new Cookie("keyEmail", user.email);
28
29        response.addCookie(cookie1);
30        response.addCookie(cookie2);
31
32        //String res = "<a href='home.jsp?name="+user.name+"&email='"+user.email+">Enter Home</a>";
33        String res = "<form action='home.jsp' method='post'>
34            +<input type='hidden' name='txtName' value='"+user.name+"' /><br>
35            +<input type='hidden' name='txtEmail' value='"+user.email+"' /><br>
36            +<input type='submit' value='Enter Home' /><br>
37            +</form>";
38
39        out.print(res);
40
41        <h3>Thank Your For Logging in: <= user.name </h3>
42
43
44    }else{
45
46

```

7.2 Write an operation to retrieve and display values from the HTTP session object. This can be done using JSP scriptlets or JSP tags.



```

  Applications : jsp - eStoreDynamicWe... | gmail@ip-172... | Home - Google Chrome | jsp - eStoreDynamicWeb/src/main/webapp/login.jsp - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
# home.jsp login.jsp x
9 User user = new User();
10 user.setEmail = request.getParameter("txtEmail");
11 user.setPassword = request.getParameter("txtPassword");
12
13 DB db = DB.getDB();
14
15 String sql = "select * from Users where email = ? and password = ?";
16 PreparedStatement preparedStatement = db.connection.prepareStatement(sql);
17 preparedStatement.setString(1, user.getEmail());
18 preparedStatement.setString(2, user.getPassword());
19
20 ResultSet set = db.executeQuery(preparedStatement);
21
22 if(set != null && set.next()){
23     user.name = set.getString("name"); // we are getting the name from the DB
24
25     // Cookies
26     Cookie cookie1 = new Cookie("keyName", user.name);
27     Cookie cookie2 = new Cookie("keyEmail", user.getEmail());
28
29     response.addCookie(cookie1);
30     response.addCookie(cookie2);
31
32     // HttpSession Object
33     session.setAttribute("keyUser", user);
34 }
35
36//String res = "<a href='home.jsp?name="+user.name+"&email='"+user.getEmail()+">Enter Home</a>";
37 String res = "<form action='home.jsp' method='post'>
38             +<input type='hidden' name='txtName' value='"+user.name+"'/><br>
39             +<input type='hidden' name='txtEmail' value='"+user.getEmail()+"'/><br>
40             +<input type='submit' value='Enter Home'/><br>
41             +</form>";
42
43 res = "<a href='home.jsp>Enter Home</a>";
44 out.print(res);
45
46
  
```

jsp:scriptlet/#text      Writable      Smart Insert      34 : 47 [6]

7.3 Go to the **home.jsp** page and write an operation to display values from the HTTP session object

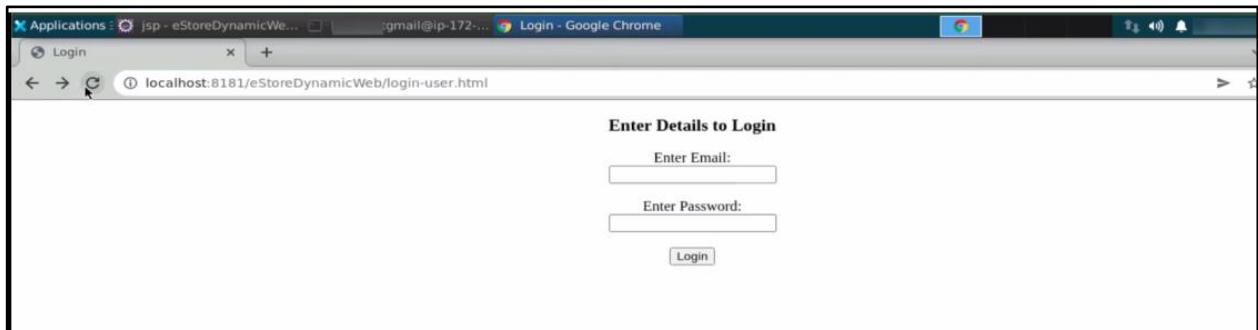
```

File Edit Source Refactor Navigate Search Project Run Window Help
File Edit Source Refactor Navigate Search Project Run Window Help
home.jsp x login.jsp
9 </head>
10<body>
11
12<center>
13     <h3>Welcome to Home</h3>
14
15
16     Cookie[] cookies = request.getCookies();
17     for(Cookie cookie : cookies){
18
19         cookie.getName() | cookie.getValue() <br>
20
21     }
22
23     String name = request.getParameter("name");
24     String email = request.getParameter("email");
25
26     String txtName = request.getParameter("txtName");
27     String txtEmail = request.getParameter("txtEmail");
28
29
30     <h3>URL Rewriting</h3>
31     Name: name <br>
32     Email: email <br>
33
34     <h3>Hidden From Field Data</h3>
35     Name: txtName <br>
36     Email: txtEmail <br>
37
38     <h3>Http Session</h3>
39
40     User user = (User)session.getAttribute("keyUser");
41
42     Name: user.name <br>
43     Email: user.email <br>
44     Password: user.password <br>
45
46 </center>

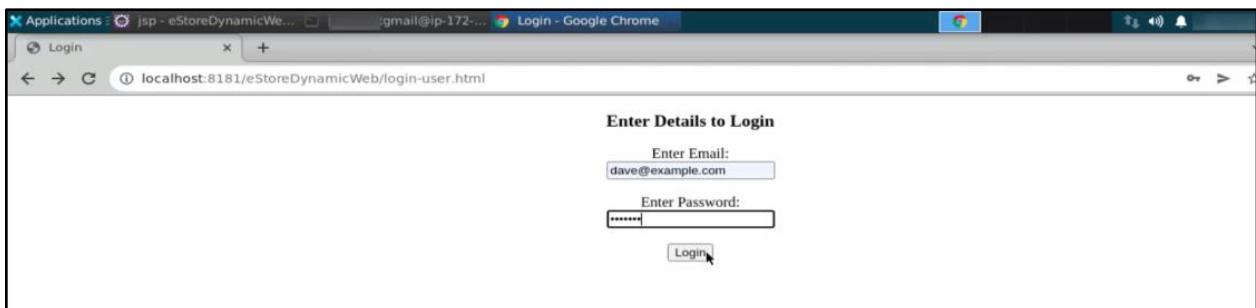
```

Writable Smart Insert 44 : 32 : 973

#### 7.4 Go back to the browser and refresh the page



#### 7.5 Enter your email address and password in the login form



## 7.6 Click on the **Login** button

The screenshot shows a web browser window with a single tab labeled "Login". The URL in the address bar is "localhost:8181/eStoreDynamicWeb/login-user.html". The page content is titled "Enter Details to Login". It contains two input fields: one for "Enter Email:" with the value "dave@example.com" and another for "Enter Password:" with the value "\*\*\*\*\*". Below these fields is a "Login" button, which is currently being clicked, as indicated by the yellow border around it.

## 7.7 Click on the **Enter Home** link

The screenshot shows a web browser window with a single tab labeled "localhost:8181/eStoreDynamicWeb/login.jsp". The URL in the address bar is "localhost:8181/eStoreDynamicWeb/login.jsp". The page content displays the message "Thank You For Logging in: Dave". Above this message, there is a link labeled "Enter Home", which is highlighted with a red box.

The screenshot shows a web browser window with a single tab labeled "Home". The URL in the address bar is "localhost:8181/eStoreDynamicWeb/home.jsp". The page content includes several sections: "Welcome to Home", "JSESSIONID | 413EB798BD2E33A7AEFF7CDABDF8BD20C", "keyName | Dave", "keyEmail | dave@example.com", "URL ReWriting", "Hidden From Field Data", and "Http Session". The "Http Session" section is highlighted with a white box. It contains the values "Name: Dave", "Email: dave@example.com", and "Password: dave123".

You will see the values from the HTTP session displayed on the page.

Following these steps, you have successfully implemented session tracking in JSP (Java Server Pages).