

## Web tech lab – 5

- Aditya Madwal
- 2200290120011

### 1. Creating a Cookie with User IDs and Passwords

#### **CreateCookieServlet.java:**

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

public class CreateCookieServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // Create a cookie with user IDs and passwords
        Cookie userCookie = new
        Cookie("userCredentials",
"user1:pwd1,user2:pwd2,user3:pwd3,user4:pwd4");
        userCookie.setMaxAge(60 * 60 * 24); // Set cookie to expire in 1
        day response.addCookie(userCookie);

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h2>Cookie Created Successfully</h2>");
        out.println("</body></html>");
    }
}
```

### 2. Authenticating User IDs and Passwords from Login Form

#### **LoginServlet.java:**

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

public class LoginServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {
        // Get user ID and password from request
    }
}
```

```
String userID = request.getParameter("userID");  
String password =  
request.getParameter("password");
```

```

// Get cookies from the request
Cookie[] cookies =
request.getCookies(); String
userCredentials = null;

// Find the cookie with user credentials
if (cookies != null) {
    for (Cookie cookie : cookies) {
        if (cookie.getName().equals("userCredentials")) {
            userCredentials = cookie.getValue();
            break;
        }
    }
}

// Authenticate user
boolean authenticated = false;
if (userCredentials != null) {
    String[] users = userCredentials.split(",");
    for (String user : users) {
        String[] credentials = user.split(":"); if
        (credentials[0].equals(userID) &&
credentials[1].equals(password)) {
            authenticated = true;
            break;
        }
    }
}

// Generate response
response.setContentType("text/html");
PrintWriter out = response.getWriter();
out.println("<html><body>");
if (authenticated) {
    out.println("<h2>Login Successful</h2>");
} else {
    out.println("<h2>Login Failed</h2>");
}
out.println("</body></html>");
}
}

```

### 3. Creating a Simple Login Form

**login.html:**

```

<!DOCTYPE html>
<html>

```

```
<head>
  <title>Login</title>
</head>
<body>
  <h2>Login Form</h2>
  <form action="LoginServlet" method="post">
    User ID: <input type="text" name="userID"><br>
```

```

        Password: <input type="password" name="password"><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>

```

## Explanation

1. **CreateCookieServlet.java:**
  - Creates a cookie named userCredentials with the values user1:pwd1,user2:pwd2,user3:pwd3,user4:pwd4.
  - Adds the cookie to the response with a lifespan of one day.
2. **LoginServlet.java:**
  - Retrieves the user ID and password entered in the login form.
  - Reads the cookies from the request and finds the userCredentials cookie.
  - Splits the cookie value to extract user credentials and checks if the provided user ID and password match any stored credentials.
  - Returns a success or failure message based on the authentication result.
3. **login.html:**
  - A simple HTML form that posts the user ID and password to LoginServlet for authentication.

Make sure to map these servlets to appropriate URLs in your web.xml:

## web.xml:

```

<web-app>
    <servlet>
        <servlet-name>CreateCookieServlet</servlet-name>
        <servlet-class>CreateCookieServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>CreateCookieServlet</servlet-name>
        <url-pattern>/createCookie</url-pattern>
    </servlet-mapping>

    <servlet>
        <servlet-name>LoginServlet</servlet-name>
        <servlet-class>LoginServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>LoginServlet</servlet-name>
        <url-pattern>/LoginServlet</url-pattern>
    </servlet-mapping>
</web-app>

```