

Name : Janvi Chauhan

Roll No : 03

Course : MSC(CS) - 6

Subject : Distributed Application Development Practical

=====

[Q-1]

Create a servlet that gets submitted values and calculate compound Interest.

Note: Write JavaScript to validate the input fields.

Formula $A = P (1 + R/(n*100))^{(nt)}$

A = the future value of the investment, including interest. P = the principal investment amount (the initial deposit or loan amount). R = the annual interest rate

(decimal). n = the number of times that interest is compounded per unit t. t = the time the money is invested for, where t = no. of months/12 (E.g. 1year and 4

months so $t = 16/12 = 1.333$). Note: 1) Validate all inputs using JavaScript 2) Assumptions can be made wherever necessary

=====

=== index.jsp ===

<%--

Document : index

Created on : 16-Mar-2024, 5:49:06 pm

Author : ansh

--%>

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

<!DOCTYPE html>

```
<html>

<head>

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

  <title>Compound Interest</title>

  <script>

    function validateForm() {

var amount = document.getElementById('amount').value;

var rate = document.getElementById('rate').value;

var year = document.getElementById('year').value;

var months = document.getElementById('months').value;

var interval = document.getElementById('interval').value;


// Validate Principal Amount

if (amount === "" || isNaN(amount) || parseFloat(amount) <= 0) {

  alert("Please enter a valid principal amount.");

  return false;

}


// Validate Interest Rate

if (rate === "" || isNaN(rate) || parseFloat(rate) <= 0) {

  alert("Please enter a valid interest rate.");

  return false;

}


// Validate Years

if (year === "" || isNaN(year) || parseInt(year) < 0) {

  alert("Please enter a valid number of years.");

  return false;

}


// Validate Months
```

```

if (months === "" || isNaN(months) || parseInt(months) < 0) {
    document.getElementById('months').value = 0;
    return false;
}

// Validate Compound Interval
if (interval === "" || isNaN(interval) || parseInt(interval) <= 0) {
    alert("Please select a compound interval.");
    return false;
}

// Form is valid
return true;
}

</script>
</head>
<body>
    <h1>Calculate Compund Interest</h1>
    <table>
        <form action="InterestServlet.ansh" onsubmit="return validateForm()">
            <tr>
                <td>Principal Amount :</td>
                <td><input type="number" name="amount" id="amount"></td>
            </tr>
            <tr>
                <td>Interest Rate :</td>
                <td><input type="number" name="rate" id="rate"></td>
            </tr>
            <tr>
                <td>Years :</td>

```

```

        <td><input type="number" name="year" id="year"></td>

        <td>Months :</td>

        <td><input type="number" name="months" id="months"></td>
    </tr>

    <tr>

        <td>Compound Interval</td>

        <td>
            <select name="interval" id="interval">
                <option selected disabled>Select Interval</option>
                <option value="365">Daily (365/yr)</option>
                <option value="360">Daily (360/yr)</option>
                <option value="52">Weekly (52/yr)</option>
                <option value="26">Bi-Weekly (26/yr)</option>
                <option value="24">Half-Monthly (24/yr)</option>
                <option value="12">Monthly (12/yr)</option>
                <option value="4">Quarterly (4/yr)</option>
                <option value="2">Half-Yearly (2/yr)</option>
                <option value="1">Yearly (1/yr)</option>
            </select>
        </td>
    </tr>

    <tr>

        <td><input type="submit" value="Submit" name="submit"></td>
    </tr>
</form>
</table>
</body>
</html>

```

```

=== InterestServlet ===

```

```

/*

```

* Click <nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt> to change this license

* Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this template

*/

```
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;
```

```
/**
```

```
 *
```

```
 * @author ansh
```

```
 */
```

```
public class InterestServlet extends HttpServlet {
```

```
    /**
```

```
     * Processes requests for both HTTP GET and POST
```

```
     * methods.
```

```
     *
```

```
     * @param request servlet request
```

```
     * @param response servlet response
```

```
     * @throws ServletException if a servlet-specific error occurs
```

```
     * @throws IOException if an I/O error occurs
```

```
     */
```

```
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
```

```
        throws ServletException, IOException {
```

```
        response.setContentType("text/html;charset=UTF-8");
```

```
        try (PrintWriter out = response.getWriter()) {
```

```

/* TODO output your page here. You may use following sample code. */

double amount,rate,year,month,interval;

amount = Double.parseDouble(request.getParameter("amount"));
rate = Double.parseDouble(request.getParameter("rate"));
year = Double.parseDouble(request.getParameter("year"));
month = Double.parseDouble(request.getParameter("months"));
interval = Double.parseDouble(request.getParameter("interval"));

double newRate = rate / 100;

double t = interval / 12;

double totalTimeInMonths = year * 12 + month;

double compoundInterest = amount * Math.pow(1 + newRate / totalTimeInMonths,
totalTimeInMonths * t);

out.println("<!DOCTYPE html>");
out.println("<html>");
out.println("<head>");
out.println("<title>Servlet InterestServlet</title>");
out.println("</head>");
out.println("<body>");
out.println("<h1>The Compund Interest is " + compoundInterest + "</h1>");
out.println("</body>");
out.println("</html>");

}
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs

```

```

    * @throws IOException if an I/O error occurs
    */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }
} // </editor-fold>

```

```
}
```

```
=====
```

[Q-2] :Write servlet which prints the name and value for all its init parameters, details of HTTP request headers, client/browser, and server.

Note: Assumptions can be made wherever necessary.

```
=====
```

```
public class NewServlet extends HttpServlet {  
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)  
        throws ServletException, IOException {  
        response.setContentType("text/html;charset=UTF-8");  
        try (PrintWriter out = response.getWriter()) {  
  
            out.println("<h2>Init Parameters:</h2>");  
            out.println("<ul>");  
            Enumeration<?> initParameterNames = getServletConfig().getInitParameterNames();  
            while (initParameterNames.hasMoreElements()) {  
                String paramName = (String) initParameterNames.nextElement();  
                String paramValue = getServletConfig().getInitParameter(paramName);  
                out.println("<li>" + paramName + ": " + paramValue + "</li>");  
            }  
  
            out.println("<h2>HTTP Request Headers:</h2>");  
            out.println("<ul>");  
            Enumeration<?> headerNames = request.getHeaderNames();
```



```

while (headerNames.hasMoreElements()) {
    String headerName = (String) headerNames.nextElement();
    String headerValue = request.getHeader(headerName);
    out.println("<li>" + headerName + ": " + headerValue + "</li>");
}

out.println("<h2>Client/Browser Details:</h2>");
out.println("<ul>");
out.println("<li>User Agent: " + request.getHeader("User-Agent") + "</li>");
out.println("<li>Remote Address: " + request.getRemoteAddr() + "</li>");
out.println("<li>Remote Host: " + request.getRemoteHost() + "</li>");
out.println("<li>Remote Port: " + request.getRemotePort() + "</li>");
out.println("</ul>");

out.println("<h2>Server Details:</h2>");
out.println("<ul>");
out.println("<li>Server Name: " + request.getServerName() + "</li>");
out.println("<li>Server Port: " + request.getServerPort() + "</li>");
out.println("</ul>");

out.close();
}
}

```

Output:-

Init Parameters:

a: 1

HTTP Request Headers:

host: localhost:8080

connection: keep-alive

cache-control: max-age=0

sec-ch-ua: "Brave";v="123", "Not:A-Brand";v="8", "Chromium";v="123"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Windows"

upgrade-insecure-requests: 1

user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36

accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8

sec-gpc: 1

accept-language: en-GB,en

sec-fetch-site: none

sec-fetch-mode: navigate

sec-fetch-user: ?1

sec-fetch-dest: document

accept-encoding: gzip, deflate, br, zstd

cookie: JSESSIONID=40509A9A30B2D0621F5A77399EA8FEE4

Client/Browser Details:

User Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36

Remote Address: 0:0:0:0:0:0:1

Remote Host: 0:0:0:0:0:0:1

Remote Port: 49903

Server Details:

Server Name: localhost

Server Port: 8080

=====

=====

[Q-3]

Design the registration page/form and write JavaScript to validate the following fields of the Registration page.

1. First Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Last Name (Name should contain alphabets and the length should not be less than 6 characters).
3. Password (Password should not be less than 6 characters in length).
4. Conform Password (Password and conform password must be same).
5. E-mail (should not contain any invalid and must follow the standard pattern of email and it is unique and required field)
6. Mobile Number (Phone number should contain 10 digits only).
7. Address (should not be Empty).
8. Select user role (admin, registered user, guest etc.) from dropdown (required field)
8. Create a table user table to store the above details.
9. Write a servlet to insert data into the table

Note: Assumptions can be made wherever necessary.

=====

=====

JSP File

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>
    <script>
```

```
function validateForm() {  
  
    var firstName = document.getElementById("firstName").value;  
    var lastName = document.getElementById("lastName").value;  
    var password = document.getElementById("password").value;  
    var confirmPassword = document.getElementById("confirmPassword").value;  
    var email = document.getElementById("email").value;  
    var mobile = document.getElementById("mobile").value;  
    var address = document.getElementById("address").value;  
    var role = document.getElementById("role").value;  
  
    if (!/^[a-zA-Z]{6,}$/ .test(firstName)) {  
        alert("First name should contain alphabets only and be at least 6 characters long.");  
        return false;  
    }  
  
    if (!/^[a-zA-Z]{6,}$/ .test(lastName)) {  
        alert("Last name should contain alphabets only and be at least 6 characters long.");  
        return false;  
    }  
  
    if (password.length < 6) {  
        alert("Password should be at least 6 characters long.");  
        return false;  
    }  
  
    if (password !== confirmPassword) {  
        alert("Password and Confirm Password do not match.");  
        return false;  
    }  
  
    if (!/^\S+@\S+\.\S+$/ .test(email)) {
```

```
        alert("Invalid email address.");
        return false;
    }
}
```

```
if (!/^\\d{10}$/.test(mobile)) {
    alert("Mobile number should contain 10 digits only.");
    return false;
}
```

```
if (address.trim() === "") {
    alert("Address should not be empty.");
    return false;
}
```

```
if (role === "") {
    alert("Please select a user role.");
    return false;
}
```

```
    return true;
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<h2>Registration Form</h2>
```

```
<form onsubmit="return validateForm()" action="Register" method="post">
```

```
<label for="firstName">First Name:</label>
```

```
<input type="text" id="firstName" name="firstName" required><br>
```

```
<label for="lastName">Last Name:</label>
```

```
<input type="text" id="lastName" name="lastName" required><br>
```

```
<label for="password">Password:</label>
<input type="password" id="password" name="password" required><br>

<label for="confirmPassword">Confirm Password:</label>
<input type="password" id="confirmPassword" name="confirmPassword" required><br>

<label for="email">Email:</label>
<input type="email" id="email" name="email" required><br>

<label for="mobile">Mobile Number:</label>
<input type="text" id="mobile" name="mobile" required><br>

<label for="address">Address:</label>
<textarea id="address" name="address" required></textarea><br>

<label for="role">Select User Role:</label>
<select id="role" name="role" required>
  <option value="">--Select Role--</option>
  <option value="admin">Admin</option>
  <option value="registered user">Registered User</option>
  <option value="guest">Guest</option>
</select><br>

<input type="submit" value="Submit">
</form>
</body>
</html>
```

Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

        String firstName = request.getParameter("firstName");
        String lastName = request.getParameter("lastName");
        String password = request.getParameter("password");
        String email = request.getParameter("email");
        String mobile = request.getParameter("mobile");
        String address = request.getParameter("address");
        String role = request.getParameter("role");

        Class.forName("com.mysql.jdbc.Driver");
        Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/anshYadav",
            "root", "");
        Statement state = conn.createStatement();

        String sql = "INSERT INTO users (firstName, lastName, password, email, mobile, address, role)
VALUES
('"+firstName+"','"+lastName+"','"+password+"','"+email+"','"+mobile+"','"+address+"','"+role+"')";

        int cnt = state.executeUpdate(sql);
        if(cnt<=0)
            throw new SQLException();
        else
            out.println("Data inserted Successfully.");
    }
}

```

=====

=====

[Q-4]

1. Use user table (username, password, role) Role - (admin, registered user, guest etc.)
2. Design a good-looking login page which accepts and submits username as email and password.
3. Write JavaScript to validate inputs.
3. Write servlet to verify whether the user exists or not.
4. if exist, redirect to user's home page.

Note: Assumptions can be made wherever necessary

=====

=====

JSP File

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
  <script>
    function validateForm() {
      var email = document.getElementById("username").value.trim();
      var password = document.getElementById("password").value.trim();

      if (email === "" || password === "") {
        document.getElementById("error").innerHTML = "Username and password are required.";
        return false;
      }

      return true;
    }
  </script>
</head>
<body>
  <div>
    <input type="text" id="username" value="Email Address" />
    <input type="password" id="password" value="Password" />
    <input type="button" value="Login" />
  </div>
  <div id="error"></div>
</body>
</html>
```



```

    }
</script>
</head>
<body>
    <form action="Login" method="post" onsubmit="return validateForm()">
        <h2>Login</h2>
        <div id="error" class="error"></div>
        <input type="email" id="email" name="email" placeholder="Email" required><br>
        <input type="password" id="password" name="password" placeholder="Password"
required><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>

```

Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

        String email = request.getParameter("email");
        String password = request.getParameter("password");

        Class.forName("com.mysql.jdbc.Driver");
        Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/anshYadav",
"root", "");
        Statement st = conn.createStatement();

        String sql = "SELECT * FROM users WHERE email = '"+email+"' AND password =
 '"+password+"'";

```

```
ResultSet rs = st.executeQuery(sql);
```

```
if (rs.next()) {  
    String role = rs.getString("role");  
    switch (role) {  
        case "admin" -> response.sendRedirect("admin.jsp");  
        case "registered user" -> response.sendRedirect("user.jsp");  
        default -> response.sendRedirect("guest.jsp");  
    }  
} else {  
    out.println("<h2>Invalid username or password.</h2>");  
}  
}  
}
```

```
=====
```

[Q-5] Write a CRUD application using servlet(s).

```
=====
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>  
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>CRUD</title>  
</head>
```

```

<body>

    <form action="CRUD" method="post">

        <h2>CRUD</h2>

        <input type="email" id="email" name="email" placeholder="Email" required><br>

        <input type="password" id="password" name="password" placeholder="Password"
required><br>

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

    </form>

</body>

</html>

```

Insert Servlet

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/anshYadav",
"root", "");
        Statement st = con.createStatement();

        String s1 = request.getParameter("username");
        String s2 = request.getParameter("password");

        String sql1 = "INSERT INTO crud (username, password) VALUES ('" + s1 + "', '" + s2 + "')";
        int ab = st.executeUpdate(sql1);

        if (ab > 0) {
            out.println("Data inserted Successfully.");
        } else {

```

```

        out.println("Failed to insert data.");
    }

    }

}

```

Update Servlet

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

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

        Statement st = con.createStatement();

        String s1 = request.getParameter("username");
        String s2 = request.getParameter("password");

        String sql = "UPDATE crud SET password = '" + s2 + "' WHERE username = '" + s1 + "'";
        int rs=st.executeUpdate(sql);
        if(rs>0){
            out.println("data update");
        }
        else{
            out.println("Not");
        }

    }

}

```

Delete Servlet

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

        Class.forName("com.mysql.jdbc.Driver");
        java.sql.Connection con =
        DriverManager.getConnection("jdbc:mysql://localhost:3306/anshYadav", "root", "");
        Statement st = con.createStatement();

        String s1 = request.getParameter("username");

        String sql = "DELETE FROM crud WHERE username = '" + s1 + "'";
        int rs = st.executeUpdate(sql);

        if(rs>0){
            out.println("Delete");
        }
        else{
            out.println("Not");
        }
    }
}
```

Read Servlet

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
```

```

try (PrintWriter out = response.getWriter()) {

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

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

    Statement st = con.createStatement();

    String sql = "SELECT * FROM crud";
    ResultSet rs= st.executeQuery(sql);

    out.println("<html><body>");
    out.println("<h2>User Data:</h2>");
    out.println("<table border='1'>");
    out.println("<tr><th>Username</th><th>Password</th></tr>");

    while (rs.next()) {
        String username = rs.getString("username");
        String password = rs.getString("password");

        out.println("<tr><td>" + username + "</td><td>" + password + "</td></tr>");
    }
}
}

```

```

=====
=====

```

[Q-6] A web application using servlet that takes a name as input and on submit it shows a hello <name> page where name is taken from the request. It shows the start time

at the right top corner of the page, displays all session information, displays the number of times a client has accessed it and provides a logout button. On clicking this

button, it should show a logout page with Thank You <name > message with the duration of usage (hint: Use session to store name and time).

=====

JSP File

<%--

Document : index

Created on : 18 Apr 2024, 4:10:44 pm

Author : Ansh

--%>

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

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Welcome</title>

</head>

<body>

<h2>Welcome</h2>

<form action="hello" method="post">

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

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

</form>

</body>

</html>

Hello Servlet

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

        HttpSession session = request.getSession();

        session.setAttribute("name", name);
        session.setAttribute("startTime", new Date());

        Integer accessCount = (Integer) session.getAttribute("accessCount");
        if (accessCount == null) {
            accessCount = 0;
        }
        accessCount++;
        session.setAttribute("accessCount", accessCount);

        out.println("Name"+session.getAttribute("name"));
        out.println("Start Time"+session.getAttribute("startTime"));
        out.println("Access Count"+session.getAttribute("accessCount"));

        out.println("<html>");
        out.println("<head><title>Hello " + name + "</title></head>");
        out.println("<body>");
        out.println("<form action='Logout' method='post'>");
        out.println("<input type='submit' value='Logout'>");
        out.println("</form>");
```



```

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

    }
}

```

Logout Servlet

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

        HttpSession session = request.getSession();

        String name = (String) session.getAttribute("name");
        Date startTime = (Date) session.getAttribute("startTime");

        long duration = new Date().getTime() - startTime.getTime();
        long durationSeconds = duration / 1000;

        out.println("<html>");
        out.println("<head><title>Logout</title></head>");
        out.println("<body>");
        out.println("<h2>Thank You " + name + "</h2>");
        out.println("<p>Duration of usage: " + durationSeconds + " seconds</p>");
        out.println("</body>");
        out.println("</html>");

        session.invalidate();
    }
}

```

}

Output:-

Name Ansh Yadav Start Time Thu Apr 18 17:23:34 IST 2024 Access Count 1

Thank You Ansh

Duration of usage: 16 seconds

=====

[Q-7] Design and develop JSP application to demonstrate 1. JSP Scripting elements 2. JSP Directives

3. JSP Implicit Objects 4. JSP Action tags

=====

=== WEB.XML ===

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

```
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
```

```
<servlet>
```

```
<servlet-name>AboutServlet</servlet-name>
```

```
<jsp-file>/about.jsp</jsp-file>
```

```
</servlet>
```

```
<servlet>
```

```
<servlet-name>IndexServlet</servlet-name>
```

```
<jsp-file>/index.jsp</jsp-file>
```

```
</servlet>
```

```
<servlet>
```

```
<servlet-name>LoginServlet</servlet-name>
<jsp-file>/login.jsp</jsp-file>
</servlet>
<servlet>
  <servlet-name>RegisterServlet</servlet-name>
  <jsp-file>/register.jsp</jsp-file>
</servlet>
<servlet>
  <servlet-name>OtherServlet</servlet-name>
  <servlet-class>OtherServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AboutServlet</servlet-name>
  <url-pattern>/about</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>LoginServlet</servlet-name>
  <url-pattern>/login</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>RegisterServlet</servlet-name>
  <url-pattern>/register</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>IndexServlet</servlet-name>
  <url-pattern>/home</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>OtherServlet</servlet-name>
  <url-pattern>/OtherServlet</url-pattern>
</servlet-mapping>
```

```
<session-config>
  <session-timeout>
    30
  </session-timeout>
</session-config>
<welcome-file-list>
  <welcome-file>index.jsp</welcome-file>
</welcome-file-list>
</web-app>
```

```
=== /link/footer.jspf ===
```

```
<!-- any content can be specified here e.g.: --%>
```

```
<%@ page pageEncoding="UTF-8" %>
```

```
<style>
```

```
/* Static Styles */
```

```
footer {
```

```
  background-color: #333;
```

```
  color: #fff;
```

```
  padding: 20px 0;
```

```
  text-align: center;
```

```
  position: fixed;
```

```
  bottom: 0;
```

```
  width: 100%;
```

```
}
```

```
/* Media Query for smaller screens */
```

```
@media (max-width: 768px) {
```

```
  footer {
```

```
    position: static;
```

```
  }
```

```
}
```

```
</style>

<footer>

  <p>&copy; 2024 Your Website. All rights reserved.</p>

</footer>
```

```
=== /link/header.jspf ===
```

```
<%-- any content can be specified here e.g.: --%>
```

```
<%@ page pageEncoding="UTF-8" %>
```

```
<style>

  ul {

    list-style-type: none;

    margin: 0;

    padding: 0;

    background-color: #333; /* Background color for the navigation bar */

    overflow: hidden;

  }

  li {

    float: left;

  }

  li a {

    display: block;

    color: white;

    text-align: center;

    padding: 14px 16px;

    text-decoration: none;

  }

  li a:hover {

    background-color: #555; /* Background color on hover */

  }

</style>

<ul class="navbar">
```

```
<li><a href="home">Home</a></li>
<li><a href="login">Login</a></li>
<li><a href="register">Register</a></li>
<li><a href="about">About</a></li>
</ul>
```

```
=== /role/admin.jsp ===
```

```
<%--
```

```
Document : admin
```

```
Created on : 17-Mar-2024, 2:19:00 pm
```

```
Author : ansh
```

```
--%>
```

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

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

```
<title>Admin Page</title>
```

```
</head>
```

```
<body>
```

```
<%@include file="../links/header.jspf" %>
```

```
<h3>Admin Page</h3>
```

```
<%@include file="../links/footer.jspf" %>
```

```
</body>
```

```
</html>
```

```
=== /role/guest.jsp ===
```

```
<%--
```

```
Document : guest
```

Created on : 17-Mar-2024, 2:19:18 pm

Author : ansh

--%>

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

<!DOCTYPE html>

<html>

<head>

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

<title>Guest Page</title>

</head>

<body>

<%@include file="../links/header.jspf" %>

<h3>Guest Page!</h3>

<%@include file="../links/footer.jspf" %>

</body>

</html>

=== /role/user.jsp ===

<%--

Document : user

Created on : 17-Mar-2024, 2:19:04 pm

Author : ansh

--%>

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

<!DOCTYPE html>

<html>

<head>

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

<title>User Page</title>

```
</head>

<body>

    <%@include file="../links/header.jspf" %>

    <h3>User Page!</h3>

    <%@include file="../links/footer.jspf" %>

</body>

</html>
```

=== about.jsp ===

```
<%--

Document   : about

Created on : 17-Mar-2024, 11:48:43 am

Author    : ansh

--%>
```

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

<!DOCTYPE html>

<html>

    <head>

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

        <title>About Page</title>

    </head>

    <body>

        <%@include file="links/header.jspf" %>

        <h2>About Page!</h2>

        <div>

            <h3>List Of Pages</h3>

            <div>Home Page</div>

            <div>Login Page</div>

            <div>Registration Page</div>

            <div>Admin Page</div>
```



```
<div>User Page</div>

<div>Guest Page</div>

</div>

<%@include file="links/footer.jspf" %>

</body>

</html>
```

=== index.jsp ===

```
<%--
```

Document : index

Created on : 17-Mar-2024, 11:45:26 am

Author : ansh

```
--%>
```

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

```
<%@page import="java.util.Date" %>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

```
<title>Home Page</title>
```

```
</head>
```

```
<body>
```

```
<%@include file="links/header.jspf" %>
```

```
<h1>Student Login And Registration Form!</h1>
```

```
<div>
```

```
<p><i>Client's IP Address: </i><%= request.getRemoteAddr() %></p>
```

```
<p><i>Session ID: </i><%= session.getId() %></p>
```

```
<p><i>Application Context Path: </i><%= application.getContextPath() %></p>
```

```
<p><i>Server Path: </i><%= request.getServletPath()%></p>
```

```
</div>
```

```

<%@include file="links/footer.jspf" %>

<%
    if(session.getAttribute("mail")!=null){
        out.println(session.getAttribute("mail"));
        out.println(new Date(session.getCreationTime()));
        out.println("<button onclick=\"handleClick()\">Log Out</button>");
    }
%>

<script>

function handleClick() {
    // Redirect to the other servlet page
    alert("Logout successful");

    window.location.href = "OtherServlet"; // Change "OtherServlet" to the actual URL of your
servlet
}

</script>

</body>

</html>

```

=== login.jsp ===

```

<%--
    Document : register
    Created on : 17-Mar-2024, 11:48:35 am
    Author : ansh
--%>

```

```

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

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

<%@page import="java.util.Date" %>

<%@page import="javax.servlet.http.HttpSession" %>

<!DOCTYPE html>

```

```
<html>

<head>

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

  <title>Login Page</title>

  <script>
function validateForm() {

    var password = document.getElementById('password').value.trim();
    var email = document.getElementById('mail').value.trim();


    // Validate Password
    if (password === "") {
        alert("Please enter a password.");
        return false;
    }


    // Validate Email
    var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    if (!email.match(emailRegex)) {
        alert("Please enter a valid email address.");
        return false;
    }

    return true;
}
</script>
</head>

<body>

  <%@include file="links/header.jspf" %>

  <h2>Login Page!</h2>

  <table>

    <form method="post" action="" onsubmit="return validateForm();">

      <tr>
```

```

        <td>E-mail :</td>

        <td><input type="mail" name="mail" id="mail" ></td>
    </tr>

    <tr>

        <td>Password :</td>

        <td><input type="password" name="password" id="password" ></td>
    </tr>


    <tr>

        <td></td>

        <td><input type="submit" value="Sign In" name="submit"></td>
    </tr>
</form>
</table>
<%@include file="links/footer.jspf" %>
</body>
</html>
<%
    if(request.getMethod().equalsIgnoreCase("post")){
        String password = request.getParameter("password");
        String email = request.getParameter("mail");

        if (password.isEmpty() || email.isEmpty()){
            out.println("<script>alert('All fields are required.');"</script>");
        } else {
            Class c = Class.forName("com.mysql.jdbc.Driver");

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

            Statement statement = connect.createStatement();

            String sql = "SELECT * FROM userData WHERE mail='"+email+"' and
password='"+password+"'";

            ResultSet data = statement.executeQuery(sql);

```

```

while(data.next()){

    String role = data.getString("role");
    if(role.equals("admin"))
    {
        session.setAttribute("mail", email);
        session.setAttribute("startTime", new Date());
    }
    %>
    <%= role%>
    <jsp:include page="roles/admin.jsp"/>
<%

    }else if(role.equals("user"))
    {
        session.setAttribute("mail", email);
        session.setAttribute("startTime", new Date());
    }
    %>
    <jsp:forward page="roles/user.jsp"/>
<%

    }else if(role.equals("guest"))
    {
        session.setAttribute("mail", email);
        session.setAttribute("startTime", new Date());
    }
    %>
    <jsp:forward page="roles/guest.jsp"/>
<%

    }

}

}
%>

=== register.jsp ===

```

```
<%--
```

```
Document : register
```

```
Created on : 17-Mar-2024, 11:48:35 am
```

```
Author : ansh
```

```
--%>
```

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

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

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

```
<title>Register Page</title>
```

```
<script>
```

```
function validateForm() {
```

```
    var Name = document.getElementById('name').value.trim();
```

```
    var password = document.getElementById('password').value.trim();
```

```
    var confirmPassword = document.getElementById('confirm_password').value.trim();
```

```
    var email = document.getElementById('mail').value.trim();
```

```
    var role = document.getElementById('role').value;
```

```
    // Validate First Name
```

```
    if (Name === "") {
```

```
        alert("Please enter your first name.");
```

```
        return false;
```

```
    }
```

```
    // Validate Password
```

```
    if (password === "") {
```

```
        alert("Please enter a password.");
```

```
        return false;
```

```

    }

    // Validate Confirm Password
    if (confirmPassword !== password) {
        alert("Passwords do not match.");
        return false;
    }

    // Validate Email
    var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    if (!email.match(emailRegex)) {
        alert("Please enter a valid email address.");
        return false;
    }

    // Validate Role
    if (role === null || role === "") {
        alert("Please select a role.");
        return false;
    }

    // Form is valid
    return true;
}

</script>
</head>
<body>
    <%@include file="links/header.jspf" %>
    <h2>Register Page!</h2>
    <table>
        <form method="post" action="" onsubmit="return validateForm();">

```

```

<tr>

    <td>Name :</td>

    <td><input type="text" name="name" id="name"></td>

</tr>

<tr>

    <td>Password :</td>

    <td><input type="password" name="password" id="password" ></td>

</tr>

<tr>

    <td>Confirm Password :</td>

    <td><input type="password" name="confirm_password" id="confirm_password" ></td>

</tr>

<tr>

    <td>E-mail :</td>

    <td><input type="mail" name="mail" id="mail" ></td>

</tr>

<tr>

    <td>Role :</td>

    <td><select name="role" id="role" >

        <option selected disabled>Select User</option>

        <option value="admin">Admin User</option>

        <option value="user">Register User</option>

        <option value="guest">Guest User</option>

    </select>

    </td>

</tr>

<tr>

    <td></td>

    <td><input type="submit" value="Register User" name="submit"></td>

</tr>

```



```

        </form>

    </table>

    <%@include file="links/footer.jspf" %>

</body>

</html>

<%
    if(request.getMethod().equalsIgnoreCase("post")){
        String name = request.getParameter("name");
        String password = request.getParameter("password");
        String confirmpassword = request.getParameter("confirm_password");
        String email = request.getParameter("mail");
        String role = request.getParameter("role");

        if (name.isEmpty() || password.isEmpty() || confirmpassword.isEmpty() || email.isEmpty() ||
role == null || role.isEmpty()) {
            out.println("<script>alert('All fields are required.');

```

%>

=====

[Q-8]

1. use user table that create in previous exercise.
2. Create a good-looking login form which accepts and submits username and password.
3. Validate all inputs using JavaScript.
4. Incorporate JSP Scripting elements, JSP Directives, JSP Implicit Objects, and JSP Action tags in your development.
4. Write JSP page to verify whether the user is exists or not and perform following,
 1. If user does not exist, back to login page.
 2. If user is exists and user-role is admin then redirect to "admin.jsp"
 3. Note: Assumptions can be made wherever necessary. For redirection use <jsp:forward/>

=====

JSP File

<%--

Document : index

Created on : 18 Apr 2024, 4:10:44 pm

Author : Ansh

--%>

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

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

```

<title>Login</title>

<script>
    function validateForm() {
        var email = document.getElementById("username").value.trim();
        var password = document.getElementById("password").value.trim();

        if (email === "" || password === "") {
            document.getElementById("error").innerHTML = "Username and password are required.";
            return false;
        }

        return true;
    }
</script>
</head>
<body>
    <form action="Login" method="post" onsubmit="return validateForm()">
        <h2>Login</h2>
        <div id="error" class="error"></div>
        <input type="email" id="email" name="email" placeholder="Email" required><br>
        <input type="password" id="password" name="password" placeholder="Password"
required><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>

```

Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {

```

```

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

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

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

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

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

    Statement st = conn.createStatement();

    String sql = "SELECT * FROM users WHERE email = '"+email+"' AND password =
 '"+password+"'";

    ResultSet resultSet = st.executeQuery(sql);

    if (resultSet.next()) {

        String userRole = resultSet.getString("role");

        if (userRole.equals("admin")) {

            request.getRequestDispatcher("admin.jsp").forward(request, response);

        } else if (userRole.equals("registered user")) {

            request.getRequestDispatcher("user.jsp").forward(request, response);

        }

        else if (userRole.equals("guest")){

            request.getRequestDispatcher("guest.jsp").forward(request,response);

        }

    } else {

        request.getRequestDispatcher("index.jsp").forward(request, response);

    }

}

}

```

=====

=====

[Q-9]

1. Create a medicine search form. Users can search medicine by medicine name or manufacturer name.

2. Create a medicine table (id, medicine-name, medicine-detail, manufacturer-name, batch-no, manufacturing month and year, expiry month and year).

=====

=====

JSP File

<%--

Document : index

Created on : 18 Apr 2024, 4:10:44 pm

Author : Ansh

--%>

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

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Medicine Search</title>

</head>

<body>

<h2>Medicine Search</h2>

```
<form action="medicine" method="get">

    <label for="searchTerm">Search by Medicine Name:</label>

    <input type="text" id="searchTerm" name="searchTerm" required>

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

</form>

</body>

</html>
```

Servlet File

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

        String Name = request.getParameter("searchTerm");

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

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

        Statement st = conn.createStatement();

        String sql = "SELECT * FROM medicines WHERE medicine_name LIKE '"+Name+"'";
        ResultSet resultSet = st.executeQuery(sql);

        out.println("<html>");
        out.println("<head><title>Search Results</title></head>");
        out.println("<body>");
        out.println("<h2>Search Results</h2>");
        out.println("<table border='1'>");
```

```

        out.println("<tr><th>ID</th><th>Medicine Name</th><th>Medicine  
Detail</th><th>Manufacturer Name</th><th>Batch No</th><th>Manufacturing  
Date</th><th>Expiry Date</th></tr>");

        while (resultSet.next()) {

            int id = resultSet.getInt("id");

            String medicineName = resultSet.getString("medicine_name");

            String medicineDetail = resultSet.getString("medicine_detail");

            String manufacturerName = resultSet.getString("manufacturer_name");

            String batchNo = resultSet.getString("batch_no");

            int manufacturingMonth = resultSet.getInt("manufacturing_month");

            int manufacturingYear = resultSet.getInt("manufacturing_year");

            int expiryMonth = resultSet.getInt("expiry_month");

            int expiryYear = resultSet.getInt("expiry_year");


            String manufacturingDate = String.format("%02d/%04d", manufacturingMonth,  
manufacturingYear);

            String expiryDate = String.format("%02d/%04d", expiryMonth, expiryYear);


            out.println("<tr>");

            out.println("<td>" + id + "</td>");

            out.println("<td>" + medicineName + "</td>");

            out.println("<td>" + medicineDetail + "</td>");

            out.println("<td>" + manufacturerName + "</td>");

            out.println("<td>" + batchNo + "</td>");

            out.println("<td>" + manufacturingDate + "</td>");

            out.println("<td>" + expiryDate + "</td>");

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

        }

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

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

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

```

```
conn.close();
```

```
}
```

```
}
```

=====

=====

[Q-10]

1. Create Book table (bookId, title, author, price, Quantity, ISBN, publisher, edition year, catalogueId) and catalogue (catalogueId, title, description).

2. Create CRUD application using JSP(s).

=====

=====

Insert JSP File

```
<%--
```

```
Document : index
```

```
Created on : 18 Apr 2024, 4:10:44 pm
```

```
Author : Ansh
```

```
--%>
```

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

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Book</title>
```

```
</head>
```

```
<body>
```

```
    <h2>Book</h2>
```



```

<form action="crud" method="post">

    <input type="text" id="title" name="title" placeholder="Title" required><br><br>

    <input type="text" id="author" name="author" placeholder="Author" required><br><br>

    <input type="text" id="price" name="price" placeholder="Price" required><br><br>

    <input type="text" id="quantity" name="quantity" placeholder="Quantity" required><br><br>

    <input type="text" id="isbn" name="isbn" placeholder="ISBN" required><br><br>

    <input type="text" id="publisher" name="publisher" placeholder="Publisher"
required><br><br>

    <input type="text" id="editionYear" name="editionYear" placeholder="Edition Year"
required><br><br>

    <input type="text" id="catalogueId" name="catalogueId" placeholder="Catalogue ID"
required><br><br>

    <input type="text" id="catalogueTitle" name="catalogueTitle" placeholder="Catalogue Title"
required><br><br>

    <input type="text" id="catalogueDescription" name="catalogueDescription"
placeholder="Catalogue Description" required><br><br>

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

</form>

</body>

</html>

```

Insert Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

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

        Statement st = con.createStatement();

```

```

String s2 = request.getParameter("title");

String s3 = request.getParameter("author");

String s4 = request.getParameter("price");

String s5 = request.getParameter("quantity");

String s6 = request.getParameter("isbn");

String s7 = request.getParameter("publisher");

String s8 = request.getParameter("editionYear");

String s9 = request.getParameter("catalogueId");

String s10 = request.getParameter("catalogueTitle");

String s11 = request.getParameter("catalogueDescription");


String sql1 = "INSERT INTO books ( title, author, price, quantity, ISBN, publisher, edition_year, catalogueId) VALUES ('"+s2+"','"+s3+"','"+s4+"','"+s5+"','"+s6+"','"+s7+"','"+s8+"','"+s9+"')";

String sql2 = "INSERT INTO catalogues (catalogueId, title, description) VALUES ('"+s9+"','"+s10+"','"+s11+"')";


int ab = st.executeUpdate(sql1);

int cb = st.executeUpdate(sql2);


if (ab > 0 && cb > 0) {

    out.println("Data inserted Successfully.");

} else {

    out.println("Failed to insert data.");

}

}

```

Update JSP File

<%--

Document : index

Created on : 18 Apr 2024, 4:10:44 pm

Author : Ansh

--%>

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

<!DOCTYPE html>

<html>

<head>

 <title>Book</title>

</head>

<body>

 <h2>Book</h2>

 <form action="crud" method="post">

 <input type="text" id="title" name="title" placeholder="Title" required>

 <input type="text" id="author" name="author" placeholder="Author" required>

 <input type="text" id="price" name="price" placeholder="Price" required>

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

 </form>

</body>

</html>

Update Servlet File

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException, ClassNotFoundException, SQLException {
```

```
    response.setContentType("text/html;charset=UTF-8");
```

```
    try (PrintWriter out = response.getWriter()) {
```

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

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

```
        Statement st = con.createStatement();
```

```

String s2 = request.getParameter("title");

String s3 = request.getParameter("author");

String s4 = request.getParameter("price");


String sql = "UPDATE books SET author = '" + s3 + "' , price = '" + s4 + "' WHERE title = '" + s2 + 
'''';

int rs=st.executeUpdate(sql);

if(rs>0){
    out.println("Data update");
}
else{
    out.println("Not Update");
}

}

}

```

Read JSP File

```

<%--
    Document   : index
    Created on : 18 Apr 2024, 4:10:44 pm
    Author    : Ansh
--%>

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

<!DOCTYPE html>

<html>

<head>

    <title>Book</title>

```

```

</head>
<body>
    <h2>Book</h2>
    <form action="crud" method="post">
        <input type="text" id="title" name="title" placeholder="Title" required><br><br>
        <input type="submit" value="CRUD">
    </form>
</body>
</html>

```

Read Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

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

        Statement st = con.createStatement();

        String s2 = request.getParameter("title");

        String sql = "SELECT * FROM books WHERE title = '"+ s2 +"'";

        ResultSet rs= st.executeQuery(sql);

        out.println("<html><body>");
        out.println("<h2>Book Data:</h2>");
        out.println("<table border='1'>");

```

```
out.println("<tr><th>Title</th><th>Author</th><th>Price</th><th>Quantity</th><th>ISBN</th><th>Publisher</th><th>Edition Year</th><th>Catalogue ID</th></tr>");
```

```
while (rs.next()) {  
    String title = rs.getString("title");  
    String author = rs.getString("author");  
    String price = rs.getString("price");  
    String quantity = rs.getString("quantity");  
    String isbn = rs.getString("ISBN");  
    String publisher = rs.getString("publisher");  
    String editionYear = rs.getString("edition_year");  
    String catalogueId = rs.getString("catalogueId");  
  
    out.println("<tr><td>" + title + "</td><td>" + author + "</td><td>" + price + "</td><td>" +  
quantity + "</td><td>" + isbn + "</td><td>" + publisher + "</td><td>" + editionYear + "</td><td>" +  
catalogueId + "</td></tr>");  
}
```

Delete JSP File

```
<%--
```

Document : index

Created on : 18 Apr 2024, 4:10:44 pm

Author : Ansh

```
--%>
```

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

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```

<title>Book</title>
</head>
<body>
  <h2>Book</h2>
  <form action="crud" method="post">
    <input type="text" id="title" name="title" placeholder="Title" required><br><br>
    <input type="submit" value="CRUD">
  </form>
</body>
</html>

```

Delete Servlet File

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {

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

        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/anshYadav",
"root", "");
        Statement st = con.createStatement();

        String s2 = request.getParameter("title");

        String sql = "DELETE FROM books WHERE title = '" + s2 + "'";
        int rs = st.executeUpdate(sql);

        if(rs>0){
            out.println("Delete");
        }
    }
}

```

```
        else{
            out.println("Not");
        }
    }
}
```

Database

```
CREATE TABLE books (
    bookId INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(255) NOT NULL,
    price DECIMAL(10, 2) NOT NULL,
    quantity INT NOT NULL,
    ISBN VARCHAR(50) NOT NULL,
    publisher VARCHAR(255) NOT NULL,
    edition_year INT NOT NULL,
    catalogueId INT
);

CREATE TABLE catalogues (
    catalogueId INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    description TEXT
);
```


=====

[Q-11]

=====

=====

[Q-12]

Write XML file (planner.xml) according to the following information.

- planner as root element.
- planner has one or more year element.
- year contains one attribute 'value' and one or more child element date.
- date have two attribute 'month' and 'day' and one child element note.

Write DTD that enforce above rules on XML

=====

Planner.XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE planner SYSTEM "planner.dtd">
<planner>
  <year value="2024">
    <date month="January" day="1">
      <note>New Year's Day</note>
```

```

    </date>

    <date month="February" day="14">
        <note>Valentine's Day</note>
    </date>
</year>
</planner>

```

Planner.DTD

```

<!ELEMENT planner (year+)>
<!ELEMENT year (date+)>
<!ATTLIST year value CDATA #REQUIRED>
<!ELEMENT date (note)>
<!ATTLIST date month CDATA #REQUIRED
              day CDATA #REQUIRED>
<!ELEMENT note (#PCDATA)>

```

```

=====
=====

```

[Q-13]

1. Write a DTD that will define the following structure for documents of type studentlist

- o A studentlist element consists of one or more student entries.

- o A student entry consists of the student's enrollmentno, name, the study_ program in which the student is enrolled and enrolment_ year.

- o A student has an enrollmentno by which they can be uniquely identified.
- o A student's name consists of three elements, the firstname, the middlename which is optional, and the lastname.
- o study_program contain values like "BSc", "MSc", "MSW", and "MBA" which is required.
- o enrolment_year is four-digit numbers like 2017 which is required.

2. Write XML file of a studentlist document and validate with your DTD.

```
=====
=====
```

Studentlist.XML

```
<!DOCTYPE studentlist SYSTEM "studentlist.dtd">
```

```
<studentlist>
```

```
  <student>
```

```
    <enrollmentno>101</enrollmentno>
```

```
    <name>
```

```
      <firstname>AnshJYadav</firstname>
```

```
      <lastname>Hirapara</lastname>
```

```
    </name>
```

```
    <study_program>BSc</study_program>
```

```
    <enrolment_year>2021</enrolment_year>
```

```
  </student>
```

```
  <student>
```

```
    <enrollmentno>102</enrollmentno>
```

```
    <name>
```

```
      <firstname>Trupal</firstname>
```

```
      <lastname>Lathiya</lastname>
```

```
    </name>
```

```
    <study_program>MSc</study_program>
```

```
    <enrolment_year>2020</enrolment_year>
```

```
  </student>
```

```
</studentlist>
```

Studentlist.DTD

<!ELEMENT studentlist (student+)>

<!ELEMENT student (enrollmentno, name, study_program, enrolment_year)>

<!ELEMENT enrollmentno (#PCDATA)>

<!ELEMENT name (firstname, lastname)>

<!ELEMENT firstname (#PCDATA)>

<!ELEMENT lastname (#PCDATA)>

<!ELEMENT study_program (#PCDATA)>

<!ELEMENT enrolment_year (#PCDATA)>

=====

[Q-14]

1. Write a Schema that will define the following structure for documents of type studentlist

- o A studentlist element consists of one or more student entries.

- o A student entry consists of the student's enrollmentno, name, the study_program in which the student is enrolled and enrolment_year.

- o A student has an enrollmentno by which they can be uniquely identified.

- o A student's name consists of three elements, the firstname, the middlename which is optional, and the lastname.

- o study_program contain values like are "BSc", "MSc", "MSW", and "MBA" which is required.

o enrolment_year is four-digit numbers like 2017 which is required.

2. Write XML file of a studentlist document and validate with your DTD.

=====

Studentlist.XML

```
<studentlist xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="studentlist.xsd">
  <student>
    <enrollmentno>101</enrollmentno>
    <name>
      <firstname>AnshJYadav</firstname>
      <lastname>Hirapara</lastname>
    </name>
    <study_program>BSc</study_program>
    <enrolment_year>2019</enrolment_year>
  </student>
  <student>
    <enrollmentno>102</enrollmentno>
    <name>
      <firstname>Trupal</firstname>
      <lastname>Lathiya</lastname>
    </name>
    <study_program>MSc</study_program>
    <enrolment_year>2020</enrolment_year>
  </student>
</studentlist>
```

XMLSchema-instance

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="NameType">
    <xs:sequence>
      <xs:element name="firstname" type="xs:string"/>
      <xs:element name="lastname" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="StudentType">
    <xs:sequence>
      <xs:element name="enrollmentno" type="xs:string"/>
      <xs:element name="name" type="NameType"/>
      <xs:element name="study_program" type="xs:string"/>
      <xs:element name="enrolment_year" type="xs:gYear"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="studentlist">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="student" type="StudentType" minOccurs="1"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

=====

=====

[Q-15]

1. Write an XML Schema that will define the following structure for documents of type phonebook

- o • A phonebook is a root element and that consists of one or more department elements.
- o • A department element consists of the deptname, deptphone, deptemail, and one or more employee elements.
- o • A employee element consist of empid, empname, empphone, and empemail.
- o • An empid by which they can be uniquely identified.

2. Write XML file of a phonebook document and validate with your XML Schema.

=====

=====

Phonebook.XML

```
<phonebook xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
  xsi:noNamespaceSchemaLocation="phonebook.xsd">
```

```
  <department>
```

```
    <deptname>HR</deptname>
```

```
    <deptphone>123-456-7890</deptphone>
```

```
    <deptemail>hr@example.com</deptemail>
```

```
    <employee>
```

```
      <empid>101</empid>
```

```
      <empname>John Doe</empname>
```

```
      <empphone>987-654-3210</empphone>
```

```
      <empemail>john.doe@example.com</empemail>
```

```
</employee>
<employee>
  <empid>102</empid>
  <empname>Jane Smith</empname>
  <empphone>987-654-3211</empphone>
  <empemail>jane.smith@example.com</empemail>
</employee>
</department>
</phonebook>
```

XMLSchema-instance

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
<xs:complexType name="EmployeeType">
  <xs:sequence>
    <xs:element name="empid" type="xs:string"/>
    <xs:element name="empname" type="xs:string"/>
    <xs:element name="empphone" type="xs:string"/>
    <xs:element name="empemail" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="DepartmentType">
  <xs:sequence>
    <xs:element name="deptname" type="xs:string"/>
    <xs:element name="deptphone" type="xs:string"/>
    <xs:element name="deptemail" type="xs:string"/>
    <xs:element name="employee" type="EmployeeType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```



```

        </xs:sequence>
    </xs:complexType>

    <xs:element name="phonebook">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="department" type="DepartmentType" minOccurs="1"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

</xs:schema>

```

```

=====
=====

```

[Q-16] Design and develop JSP based application using JSTL tags.

```

=====
=====

```

```

<%--
    Document   : index
    Created on : 27-Mar-2024, 2:48:19 pm
    Author    : ansh
--%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%--<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>--%>
<html>
<head>

```

```
<title>Simple Calculator</title>
</head>
<body>
  <h2>Simple Calculator</h2>
  <form action="" method="post">
    <input type="text" name="num1" placeholder="Enter number 1" /><br />
    <input type="text" name="num2" placeholder="Enter number 2" /><br />
    <button type="submit" name="operation" value="add">Add</button>
    <button type="submit" name="operation" value="subtract">Subtract</button>
    <button type="submit" name="operation" value="multiply">Multiply</button>
    <button type="submit" name="operation" value="divide">Divide</button>
  </form>
  <c:if test="${not empty param.operation}">
    <h3>Result:</h3>
    <c:choose>
      <c:when test="${param.operation eq 'add'}">
        <c:set var="result" value="${param.num1 + param.num2}" />
      </c:when>
      <c:when test="${param.operation eq 'subtract'}">
        <c:set var="result" value="${param.num1 - param.num2}" />
      </c:when>
      <c:when test="${param.operation eq 'multiply'}">
        <c:set var="result" value="${param.num1 * param.num2}" />
      </c:when>
      <c:when test="${param.operation eq 'divide'}">
        <c:choose>
          <c:when test="${param.num2 ne 0}">
            <c:set var="result" value="${param.num1 / param.num2}" />
          </c:when>
          <c:otherwise>
            <c:set var="result" value="Cannot divide by zero" />
          </c:otherwise>
        </c:choose>
      </c:when>
    </c:choose>
  </c:if>
</body>
```

```
        </c:otherwise>
    </c:choose>
</c:when>
</c:choose>
<c:out value="${result}" />
</c:if>
</body>
</html>
```