1. Embed an image map picture(India map) on a Web page that provides different links to other Web pages(different states) and show the all the related information depending on where a user clicks on the image.

```
<!DOCTYPE html>
<html>
<head>
  <title>India Map Image Map</title>
</head>
<body>
  <h1>India Map</h1>
  <img src="path_to_your_india_map_image.jpg" alt="India Map" usemap="#indiaMap">
  <map name="indiaMap">
    <area shape="rect" coords="x1,y1,x2,y2" href="link_to_state1.html" alt="State 1">
    <area shape="rect" coords="x1,y1,x2,y2" href="link_to_state2.html" alt="State 2">
    <!-- Add more 'area' elements for other states -->
  </map>
</body>
</html>
```

2. Create an webpage to embed a human body image, identify and display all the related information about the human body parts (head,eye,nose,finger etc) based on the user clicks on the human body image map.

```
<!DOCTYPE html>
<html>
<head>
  <title>Human Body Map</title>
</head>
<body>
```

```
<h1>Human Body Map</h1>

<img src="path_to_your_human_body_image.jpg" alt="Human Body" usemap="#bodyMap">

<map name="bodyMap">

    <area shape="rect" coords="x1,y1,x2,y2" href="#" alt="Head" onclick="showInfo('head')">

    <area shape="rect" coords="x1,y1,x2,y2" href="#" alt="Eyes" onclick="showInfo('eyes')">

    <area shape="rect" coords="x1,y1,x2,y2" href="#" alt="Nose" onclick="showInfo('nose')">

    <area shape="rect" coords="x1,y1,x2,y2" href="#" alt="Fingers" onclick="showInfo('fingers')">

    <!-- Add more 'area' elements for other body parts -->

</map>

<div id="infoContainer"></div>

<script>

    function showInfo(bodyPart) {

        var infoContainer = document.getElementById("infoContainer");

        infoContainer.innerHTML = "";


        if (bodyPart === "head") {

            infoContainer.innerHTML = "<h2>Head Information</h2><p>Information about the head...</p>";

        } else if (bodyPart === "eyes") {

            infoContainer.innerHTML = "<h2>Eyes Information</h2><p>Information about the eyes...</p>";

        } else if (bodyPart === "nose") {

            infoContainer.innerHTML = "<h2>Nose Information</h2><p>Information about the nose...</p>";

        } else if (bodyPart === "fingers") {
```

infoContainer.innerHTML = "<h2>Fingers Information</h2> Information about the fingers... ";

}

// Add more 'else if' statements for other body parts

// You can also redirect to separate pages for each body part instead of showing the information directly using window.location.href = "your_page_url";

}

</script>

</body>

</html>

3. Add an embedded style sheet to the html document that represent hotel menu items.The style sheet should contain the rules for displaying all non vegetarian menu items in {color:red;fontsize: smaller;font-weight:bold} and vegetarian menu items in {color:green;font-size:larger}.

```
<!DOCTYPE html>
<html>
<head>
    <title>Hotel Menu</title>
    <style>
        /* Embedded Style Sheet */
        .non-veg {
            color: red;
            font-size: smaller;
            font-weight: bold;
        }

        .veg {
            color: green;
            font-size: larger;
        }
    </style>
</head>
<body>
    <!-- Your HTML content for the menu goes here -->
</body>
</html>
```

```html
<body>
    <h1>Hotel Menu</h1>

    <ul>
        <li class="non-veg">Chicken Curry</li>
        <li class="veg">Vegetable Biryani</li>
        <li class="veg">Paneer Tikka</li>
        <li class="non-veg">Fish Fry</li>
    </ul>

    <!-- More HTML content for the menu if needed -->

</body>
```

4. Create external style sheet using different subset of the style rules with the following specification
   Body{font-family:arial, Helvetica, sans-serif} li {font-weight:bold}
   h1{text decoration:underline}ul{margin-left:20px}
   Then write a complete html document to apply all the above rules to the suitable application domain.

```css
/* style.css */

body {
  font-family: Arial, Helvetica, sans-serif;
}

li {
  font-weight: bold;
}

h1 {
  text-decoration: underline;
}

ul {
  margin-left: 20px;
}
```

```html
<!DOCTYPE html>
<html>
<head>
    <title>Application Domain</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <!-- Your HTML content for the application domain goes here -->
</body>
</html>
```

```html
<body>
    <h1>Welcome to the Application Domain</h1>

    <ul>
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
    </ul>

    <p>This is a sample paragraph.</p>

    <!-- More HTML content for the application domain if needed -->

</body>
```

5. Develop a web page that allows the user to enter all the details of the passenger(name, age, email id , gender).Write a client side scripting code to validate the email id ,age and gender, where email id should consists of the special symbol @and period(.),where age between 1 to 100 and gender is male or female

```
<!DOCTYPE html>

<html>

<head>

   <title>Passenger Details</title>

   <script>

      function validateForm() {

         var name = document.getElementById("name").value;

         var age = document.getElementById("age").value;
```

```
        var email = document.getElementById("email").value;
        var gender = document.getElementById("gender").value;

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

        // Age validation
        if (isNaN(age) || age < 1 || age > 100) {
            alert("Please enter a valid age between 1 and 100.");
            return false;
        }

        // Gender validation
        if (gender !== "male" && gender !== "female") {
            alert("Please select a valid gender.");
            return false;
        }

        // Form is valid
        return true;
    }
  </script>
</head>
```

```
<body>
    <h1>Passenger Details</h1>

    <form onsubmit="return validateForm()">
        <label for="name">Name:</label>
        <input type="text" id="name" required><br><br>

        <label for="age">Age:</label>
        <input type="number" id="age" required><br><br>

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

        <label for="gender">Gender:</label>
        <select id="gender" required>
            <option value="">Select Gender</option>
            <option value="male">Male</option>
            <option value="female">Female</option>
        </select><br><br>

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

6. Create a form that gets input of several lines of text and a search character. On clicking the search button in the client form invoke suitable script function to display the number of occurrences of the character in the text using suitable string methods.

```
<!DOCTYPE html>
<html>
<head>
    <title>Character Occurrence Counter</title>
    <script>
        function countOccurrences() {
            var text = document.getElementById("text").value;
            var searchChar = document.getElementById("searchChar").value;
            var count = 0;
            // Using string methods to count occurrences
            for (var i = 0; i < text.length; i++) {
                if (text.charAt(i) === searchChar) {
                    count++;
                }
            }
            // Displaying the result
            var result = "Number of occurrences of '" + searchChar + "' in the text: " + count;
            document.getElementById("result").innerText = result;
        }
    </script>
</head>
<body>
    <h1>Character Occurrence Counter</h1>
    <form>
```

```
        <label for="text">Enter Text:</label>
        <textarea id="text" rows="4" cols="50"></textarea><br><br>
        <label for="searchChar">Search Character:</label>
        <input type="text" id="searchChar"><br><br>
        <button type="button" onclick="countOccurrences()">Search</button>
    </form>
    <div id="result"></div>
</body>
</html>
```

7. Create an address book servlet that allows the user to register for several mailing lists. The address book contains first name, designation, address1, address2, City, State, emailid. When emailid is given, it return an HTML page with all the information.

```
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;
public class AddressBookServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        // Retrieve the email ID from the request parameter
        String emailId = request.getParameter("emailid");
        // TODO: Retrieve the information from the address book based on the email ID
        // Prepare the HTML response
        response.setContentType("text/html");
```

```
        PrintWriter out = response.getWriter();
        // Generate the HTML page with the retrieved information
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Address Book</title>");
        out.println("</head>");
        out.println("<body>");
        // TODO: Output the retrieved information in the HTML page
        out.println("</body>");
        out.println("</html>");
    }
}
```

8. (i) write a servlet program to display a 'Wish you happy birth day' message on the client system.(20 marks).
   ii. Create a dynamic HTML form that allows the user to enter the name of the city. While submission of this form, the servlet retrieves and responses the weather report for the user entered city.(60 marks)

   i)

```
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;

public class BirthdayServlet extends HttpServlet {
   @Override
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      out.println("<html>");
      out.println("<head>");
      out.println("<title>Birthday Greeting</title>");
```

```
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Wish you a Happy Birthday!</h1>");
        out.println("</body>");
        out.println("</html>");
    }
}
```

ii)

```
<!DOCTYPE html>
<html>
<head>
    <title>Weather Report</title>
</head>
<body>
    <h1>Weather Report</h1>

    <form action="WeatherServlet" method="POST">
        <label for="city">City:</label>
        <input type="text" id="city" name="city" required>
        <button type="submit">Get Weather</button>
    </form>
</body>
</html>
```

```
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;

public class WeatherServlet extends HttpServlet {
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        // Retrieve the city entered by the user
        String city = request.getParameter("city");
        // TODO: Retrieve the weather report for the entered city
        // Prepare the HTML response
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
```

```
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Weather Report</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Weather Report for " + city + "</h1>");
        // TODO: Output the weather report in the HTML page
        out.println("</body>");
        out.println("</html>");
    }
}
```

9. Write a Java Server Page program to find the simple interest .Test this by a client program to get the form data inputs (Principal amount, rate of interest,periods) and display the result(SI=(P*N*R)/100).

```
<%@ page language="java" %>
<html>
<head>
    <title>Simple Interest Calculator</title>
</head>
<body>
    <h1>Simple Interest Calculator</h1>

    <form action="calculate.jsp" method="post">
        <label for="principal">Principal Amount:</label>
        <input type="number" id="principal" name="principal" required><br><br

        <label for="rate">Rate of Interest:</label>
        <input type="number" id="rate" name="rate" required><br><br>

        <label for="periods">Periods (in years):</label>
        <input type="number" id="periods" name="periods" required><br><br>

        <button type="submit">Calculate</button>
    </form>

    <%-- Display the result if available --%>
    <c:if test="${not empty result}">
        <h2>Result:</h2>
        <p>Simple Interest: ${result}</p>
    </c:if>
</body>
</html>
```

10. Create an XML to represent the BOOKS catalog that has the following elements(TITLE,ISBN NO,AUTHOR,PUBLISHER,PRICE).Display the book details styledwith XSLT.

jsp Copy code

```jsp
<%@ page language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%
    // Retrieve the form data
    double principal = Double.parseDouble(request.getParameter("principal"));
    double rate = Double.parseDouble(request.getParameter("rate"));
    int periods = Integer.parseInt(request.getParameter("periods"));

    // Calculate the simple interest
    double simpleInterest = (principal * rate * periods) / 100;

    // Store the result as a request attribute
    request.setAttribute("result", simpleInterest);
%>

<jsp:forward page="interest.jsp" />
```

11. Create an Extensible mark up language to represent the students mark information of a class. Create a webpage to display all the students consolidated mark statement with pass (green color) or fail (red color)using XSLT.

Xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<class>
  <student rollno="1">
    <name>John</name>
    <marks>
      <maths>90</maths>
      <science>80</science>
```

```
    <english>75</english>
   </marks>
  </student>
  <student rollno="2">
   <name>Jane</name>
   <marks>
    <maths>75</maths>
    <science>85</science>
    <english>80</english>
   </marks>
  </student>
  <student rollno="3">
   <name>Bob</name>
   <marks>
    <maths>60</maths>
    <science>50</science>
    <english>45</english>
   </marks>
  </student>
</class>
```

XSLT:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
```

```
<head>
  <title>Consolidated Mark Statement</title>
  <style>
    .pass {color: green;}
    .fail {color: red;}
  </style>
</head>
<body>
  <h1>Consolidated Mark Statement</h1>
  <table border="1">
    <tr>
      <th>Roll No.</th>
      <th>Name</th>
      <th>Maths</th>
      <th>Science</th>
      <th>English</th>
      <th>Status</th>
    </tr>
    <xsl:for-each select="class/student">
      <tr>
        <td><xsl:value-of select="@rollno"/></td>
        <td><xsl:value-of select="name"/></td>
        <td><xsl:value-of select="marks/maths"/></td>
        <td><xsl:value-of select="marks/science"/></td>
        <td><xsl:value-of select="marks/english"/></td>
        <td>
          <xsl:choose>
```

```
        <xsl:when test="marks/maths &gt;= 40 and marks/science &gt;= 40 and
      marks/english &gt;= 40">
          <span class="pass">Pass</span>
        </xsl:when>
        <xsl:otherwise>
          <span class="fail">Fail</span>
        </xsl:otherwise>
      </xsl:choose>
    </td>
   </tr>
  </xsl:for-each>
 </table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

12. Create a program in AJAX using Javascript, DHTML and the XMLHttpRequest object to perform a GET or POST and return a result without reloading the HTML page.

```html
<!DOCTYPE html>
<html>
<head>
  <title>AJAX GET Request Example</title>
  <script>
    function performAJAXRequest() {
      // Create XMLHttpRequest object
      var xhr = new XMLHttpRequest();

      // Define the callback function to handle the response
      xhr.onreadystatechange = function() {
        if (xhr.readyState === 4 && xhr.status === 200) {
          // Get the response data
          var response = xhr.responseText;

          // Display the response data
          var resultContainer = document.getElementById("result");
          resultContainer.innerHTML = response;
        }
      };

      // Open and send the GET request
      xhr.open("GET", "https://api.example.com/data", true);
      xhr.send();
    }
  </script>
</head>
<body>
  <h1>AJAX GET Request Example</h1>
  <button onclick="performAJAXRequest()">Make GET Request</button>
  <div id="result"></div>
</body>
</html>
```

13. Create a HTML application with frames, links, tables and other tags for a Super market and also do the necessary client side validation using Java script. Document the functional requirements you are considering.

```
<!DOCTYPE html>
<html>
<head>
  <title>Supermarket</title>
  <script>
    function validateForm() {
      // Get the form elements
      var firstName = document.getElementById("firstName").value;
      var lastName = document.getElementById("lastName").value;
      var email = document.getElementById("email").value;
      // Check for empty fields
      if (firstName === "" || lastName === "" || email === "") {
        alert("Please fill out all fields.");
        return false;
      }
      // Check for valid email format
      if (!email.includes("@") || !email.includes(".")) {
        alert("Please enter a valid email address.");
        return false;
      }
      // Form is valid
      return true;
```

```
    }
  </script>
</head>
<frameset cols="20%, 80%">
  <frame src="navigation.html">
  <frame src="products.html">
</frameset>
<noframes>
  <body>
    <h1>Supermarket</h1>
    <p>This website requires frames. Please upgrade your browser.</p>
  </body>
</noframes>
</html>
```

14. Create a web page containing details about your college using DHTML. Use at least five hyperlinks. Also embed images in the web page you are creating. Document the functional requirements you are considering.

```
<!DOCTYPE html>
<html>
<head>
  <title>College Details</title>
  <style>
    .image {
      max-width: 100%;
      height: auto;
    }
  </style>
```

```
</head>
<body>
  <h1>College Name</h1>
  <h2>About Us</h2>
  <p>College details, including location, history, mission, and vision.</p>
  <h2>Departments</h2>
  <ul>
    <li><a href="departments.html">Department 1</a></li>
    <li><a href="departments.html">Department 2</a></li>
    <li><a href="departments.html">Department 3</a></li>
  </ul>
  <h2>Faculty</h2>
  <p>Information about the faculty members and their expertise.</p>
  <h2>Admissions</h2>
  <p>Admission process and requirements for prospective students.</p>
  <h2>Contact</h2>
  <p>Contact details, including address, phone number, and email.</p>
  <h2>Events</h2>
  <p>Upcoming events and activities in the college.</p>

  <h2>Gallery</h2>
  <img src="college_logo.png" alt="College Logo" class="image">
  <img src="campus_photo.jpg" alt="Campus Photo" class="image">
  <img src="academic_activity.jpg" alt="Academic Activity" class="image">
</body>
</html>
```

15. Create a web page containing details about the placement cell in your college using DHTML. Use at least five hyperlinks. Also embed images in the web page you are creating. Document the functional requirements you are considering.

```
<!DOCTYPE html>
<html>
<head>
  <title>Placement Cell</title>
  <style>
    .image {
      max-width: 100%;
      height: auto;
    }
  </style>
</head>
<body>
  <h1>Placement Cell</h1>
  <h2>About Us</h2>
  <p>Details about the role and responsibilities of the placement cell in your college.</p>
  <h2>Services</h2>
  <ul>
    <li><a href="job_opportunities.html">Job Opportunities</a></li>
    <li><a href="internships.html">Internships</a></li>
    <li><a href="alumni_testimonials.html">Alumni Testimonials</a></li>
    <li><a href="training_programs.html">Training Programs</a></li>
    <li><a href="contact.html">Contact Us</a></li>
  </ul>
```

```
  <h2>Achievements</h2>
  <p>Highlight the successful placements and notable achievements of the placement cell.</p>
  <h2>Image Gallery</h2>
  <img src="placement_event1.jpg" alt="Placement Event 1" class="image">
  <img src="placement_event2.jpg" alt="Placement Event 2" class="image">
  <img src="placement_event3.jpg" alt="Placement Event 3" class="image">
  <h2>Testimonials</h2>
  <blockquote>
    <p>"The placement cell has been instrumental in shaping my career. I am grateful for their guidance and support." - John Doe, Class of 2022</p>
    <p>"Thanks to the placement cell, I secured my dream job. Their efforts are commendable." - Jane Smith, Class of 2021</p>
  </blockquote>
</body>
</html>
```

16. Create a web page using cascading style sheets for an Automobile Company which markets and sells cars. Use at least five hyperlinks. Also embed images in the web page you are creating. Document the functional requirements you are considering.

```
<!DOCTYPE html>
<html>
<head>
  <title>Automobile Company</title>
  <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
```

```
<header>
  <h1>Welcome to Automobile Company</h1>
  <nav>
    <ul>
      <li><a href="car_models.html">Car Models</a></li>
      <li><a href="special_offers.html">Special Offers</a></li>
      <li><a href="financing_options.html">Financing Options</a></li>
      <li><a href="customer_reviews.html">Customer Reviews</a></li>
      <li><a href="contact.html">Contact</a></li>
    </ul>
  </nav>
</header>
<section id="about">
  <h2>About Us</h2>
  <p>Information about the automobile company, its history, mission, vision, and core values.</p>
</section>
<section id="car-models">
  <h2>Car Models</h2>
  <div class="car">
    <img src="car1.jpg" alt="Car 1">
    <h3>Car Model 1</h3>
    <p>Features, specifications, and pricing details of Car Model 1.</p>
  </div>
  <div class="car">
    <img src="car2.jpg" alt="Car 2">
    <h3>Car Model 2</h3>
```

```
    <p>Features, specifications, and pricing details of Car Model 2.</p>
  </div>
  <div class="car">
    <img src="car3.jpg" alt="Car 3">
    <h3>Car Model 3</h3>
    <p>Features, specifications, and pricing details of Car Model 3.</p>
  </div>
 </section>
 <footer>
  <p>&copy; 2023 Automobile Company. All rights reserved.</p>
 </footer>
</body>
</html>
```

17. Develop a web application for an Inventory Control System using JSP. Use cookies as part of the web application and perform appropriate validations. Document the functional requirements you are considering.

index.jsp

```
<!DOCTYPE html>
<html>
<head>
 <title>Login Page</title>
</head>
<body>
 <h1>Login</h1>
 <form action="login.jsp" method="post">
  <label for="email">Email:</label>
```

```
    <input type="text" id="email" name="email" required><br>
    <label for="password">Password:</label>
    <input type="password" id="password" name="password" required><br>
    <input type="submit" value="Login">
  </form>
</body>
</html>
```

login.jsp

```
<%@ page language="java" %>
<%@ page import="java.util.Map" %>
<%@ page import="java.util.HashMap" %>

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

  // Perform authentication and validation
  // Assume user credentials are stored in a map for demonstration purposes
  Map<String, String> users = new HashMap<>();
  users.put("user1@example.com", "password1");
  users.put("user2@example.com", "password2");

  if (users.containsKey(email) && users.get(email).equals(password)) {
    // User authenticated, set a cookie to remember login status
    Cookie loginCookie = new Cookie("loggedInUser", email);
```

```
    loginCookie.setMaxAge(60 * 60); // Set cookie expiry time (in seconds)
    response.addCookie(loginCookie);

    // Redirect to the dashboard page
    response.sendRedirect("dashboard.jsp");
  } else {
    // Invalid login, redirect back to login page with an error message
    response.sendRedirect("index.jsp?error=1");
  }
%>
```

18. Create an XML document for the following relations for a company database application:

EMPLOYEE (ENO, NAME, GENDER, DOB, DOJ, DESIGNATION, BASIC, DEPT_NO, PAN, SENO) PAN – Permanent Account Number ,SENO – Supervisor Employee Number DEPARTMENT (DEPT_NO, NAME, MENO),MENO - Manager Employee Number PROJECT (PROJ_NO, NAME, DEPT_NO) WORKSFOR (ENO, PROJ_NO, DATE_WORKED, HOURS)

Each department has a manager managing it. There are also supervisors in each department who supervise a set of employees. A department can control any number of projects. But only one department can control a project. An employee can work on any number of distinct projects on a given day. The primary keys are underlined.

For the XML document created, create a style sheet and display the document in a browser.

19. Create an XML document for the following relations for a banking application:
CUSTOMER (CID, CNAME) ACCOUNT (ANO, ATYPE, BALANCE, CID)
An account can be a savings account or a current account. Check ATYPE in ‘S’ or ‘C’. A customer can have both types of accounts.
TRANSACTION (TID, ANO, TTYPE, TDATE, TAMOUNT)
TTYPE CAN BE ‘D’ OR ‘W’ D- Deposit; W – Withdrawal
The primary keys are underlined.
b. For the XML document created, create a style sheet and display the document in a browser.

20. Create a HTML application with frames, links, tables and other tags for INDIA Map. State the assumptions you make (Social logic you are taking into consideration).

```
<!DOCTYPE html>
<html>
<head>
  <title>India Map</title>
</head>
<body>
  <h1>India Map</h1>

  <frameset rows="*, 100">
    <frame src="india_map.html" name="map_frame">
    <frame src="" name="details_frame">
  </frameset>

  <noframes>
    <body>
      <h2>Your browser does not support frames.</h2>
      <p>You can still view the map <a href="india_map.html">here</a>.</p>
    </body>
  </noframes>
</body>
</html>
```

21. Create a HTML page with frames, links, tables and other tags for highlighting the facilities in the Department in your College. State the assumptions you make (business logic you are taking into consideration).

html Copy code

```
<!DOCTYPE html>
<html>
<head>
  <title>Department of Computer Science Facilities</title>
</head>
<body>
  <h1>Department of Computer Science Facilities</h1>

  <frameset cols="200, *">
    <frame src="facilities_menu.html" name="menu_frame">
    <frame src="facilities.html" name="details_frame">
  </frameset>

  <noframes>
    <body>
      <h2>Your browser does not support frames.</h2>
      <p>You can still view the facilities <a href="facilities.html">here</a
    </body>
  </noframes>
</body>
</html>
```

22. Design a table and alter that table's column properties using an external and internal CSS file.

```html
<!DOCTYPE html>
<html>
<head>
  <title>Table with CSS</title>
  <link rel="stylesheet" type="text/css" href="external.css">
  <style>
    .column {
      background-color: lightblue;
      color: white;
      font-weight: bold;
      text-align: center;
    }
  </style>
</head>
<body>
  <table>
    <tr>
      <th class="column">Name</th>
      <th class="column">Age</th>
      <th class="column">City</th>
    </tr>
    <tr>
      <td>John</td>
      <td>25</td>
      <td>New York</td>
    </tr>
    <tr>
      <td>Jane</td>
      <td>30</td>
      <td>London</td>
    </tr>
  </table>
</body>
</html>
```

Stop generating

```css
.column {
  background-color: lightgreen;
  color: black;
  text-align: center;
  font-size: 16px;
}
```

23. Create a User Registration form with First Name, Last name, Address, City, State, Country,Pincode, Username and Password fields for a General login webpage and satisfy the following criteria:
Create a validate ( ) function that does the following:
(a) Checks that the First Name, Last Name, City, Country, Username, and Password1 fields arefilled out.
(b) Checks that the Pincode is exactly 6 numeric's.

(c) Checks that the state is exactly two characters.
(d) Checks that the email is a valid email address.
• false if email has fewer than 6 characters
• false if email does not contain an @ symbol
• false if email does not contain a period (.)
• true otherwise

```
<!DOCTYPE html>
<html>
<head>
  <title>User Registration Form</title>
  <script>
    function validate() {
      var firstName = document.getElementById("firstName").value;
      var lastName = document.getElementById("lastName").value;
      var city = document.getElementById("city").value;
      var country = document.getElementById("country").value;
      var pincode = document.getElementById("pincode").value;
      var state = document.getElementById("state").value;
      var email = document.getElementById("email").value;
      var password1 = document.getElementById("password1").value;

      if (firstName === "" || lastName === "" || city === "" || country === "" || email ===
"" || password1 === "") {
        alert("Please fill out all required fields.");
        return false;
      }

      if (pincode.length !== 6 || isNaN(pincode)) {
        alert("Pincode should be a 6-digit numeric value.");
        return false;
      }

      if (state.length !== 2) {
        alert("State should be exactly two characters.");
        return false;
      }

      if (email.length < 6 || email.indexOf("@") === -1 || email.indexOf(".") === -1) {
        alert("Please enter a valid email address.");
        return false;
      }

      return true;
    }
  </script>
</head>
<body>
  <h2>User Registration</h2>
```

```
    <form onsubmit="return validate()">
      <label for="firstName">First Name:</label>
      <input type="text" id="firstName" required><br>

      <label for="lastName">Last Name:</label>
      <input type="text" id="lastName" required><br>

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

      <label for="city">City:</label>
      <input type="text" id="city" required><br>

      <label for="state">State:</label>
      <input type="text" id="state" maxlength="2" required><br>

      <label for="country">Country:</label>
      <input type="text" id="country" required><br>

      <label for="pincode">Pincode:</label>
      <input type="text" id="pincode" pattern="[0-9]{6}" required><br>

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

      <label for="username">Username:</label>
      <input type="text" id="username" required><br>

      <label for="password1">Password:</label>
      <input type="password" id="password1" required><br>

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

24. Design a Job Registration form/Sign Up form using HTML. [The Form should contains Textbox[name], Radio button [sex], Text area[comments], List Items[country],File[Browse], LabelField, Password field[password] and check box.

```
<!DOCTYPE html>
<html>
<head>
  <title>Job Registration Form</title>
</head>
<body>
  <h2>Job Registration Form</h2>
  <form>
    <label for="name">Name:</label>
```

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

    <label>Sex:</label>
    <label for="male">Male</label>
    <input type="radio" id="male" name="sex" value="male">
    <label for="female">Female</label>
    <input type="radio" id="female" name="sex" value="female"><br>

    <label for="comments">Comments:</label>
    <textarea id="comments"></textarea><br>

    <label for="country">Country:</label>
    <select id="country">
      <option value="usa">USA</option>
      <option value="uk">UK</option>
      <option value="canada">Canada</option>
      <option value="australia">Australia</option>
    </select><br>

    <label for="resume">Resume:</label>
    <input type="file" id="resume"><br>

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

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

    <label for="terms">I agree to the terms and conditions:</label>
    <input type="checkbox" id="terms" required><br>

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

25. Create a page with JavaScript to do the following. These can all be on one page.
    (a) Prompt the user for their name.
    (b) Use a pop-up box to welcome the user by name.
    (c) Display the current date on the page in the following format: April 30, 2014. Do not display the time. Do not "hard code" the date; if I load the page tomorrow, I should get a different date than if I load it today.
    (d) Display the last modified date of the document.
    (e) Put some useful advice, on any subject, in the status line of the browser.

```html
<!DOCTYPE html>
<html>
<head>
  <title>JavaScript Page</title>
  <script>
    // Prompt the user for their name
    var name = prompt("Please enter your name:");

    // Use a pop-up box to welcome the user by name
    alert("Welcome, " + name + "!");

    // Display the current date in the specified format
    var currentDate = new Date();
    var options = { month: "long", day: "numeric", year: "numeric" };
    var formattedDate = currentDate.toLocaleDateString("en-US", options);
    document.getElementById("date").textContent = formattedDate;

    // Display the last modified date of the document
    document.getElementById("lastModified").textContent = document.lastModifi

    // Set the status line of the browser with useful advice
    window.status = "Remember to stay hydrated and take breaks regularly!";
  </script>
</head>
<body>
  <h1>JavaScript Page</h1>
  <p>Today's Date: <span id="date"></span></p>
  <p>Last Modified: <span id="lastModified"></span></p>
</body>
</html>
```

26. Develop a web application for a Super Market Billing System using ASP / JSP pages. Use cookies as part of the web application. State the assumptions you make (business logic you are taking into consideration).
    Note: Your application must access a database.

27. Create a HTML application with frames, links, tables and other tags that will provide information about the services provided in Veena Multi Specialty Hospital. State the assumptions you make (business logic you are taking into consideration).

Certainly! Here's an example of an HTML application with frames, links, tables, and other tags to provide information about the services provided in Veena Multi Specialty Hospital:

html Copy code

```
<!DOCTYPE html>
<html>
<head>
  <title>Veena Multi Specialty Hospital</title>
</head>
<frameset rows="15%, 85%">
  <frame src="header.html" noresize>
  <frameset cols="20%, 80%">
    <frame src="menu.html" noresize>
    <frame src="content.html" name="content">
  </frameset>
</frameset>
</html>
```

1. header.html:

html Copy code

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* Add any necessary styles for the header */
  </style>
</head>
<body>
  <h1>Veena Multi Specialty Hospital</h1>
  <!-- Add hospital logo if available -->
</body>
</html>
```

2. menu.html:

```html
<!DOCTYPE html>
<html>
<head>
  <style>
    /* Add any necessary styles for the menu */
  </style>
</head>
<body>
  <h2>Services</h2>
  <ul>
    <li><a href="content.html" target="content">Cardiology</a></li>
    <li><a href="content.html" target="content">Orthopedics</a></li>
    <li><a href="content.html" target="content">Oncology</a></li>
    <!-- Add more services as needed -->
  </ul>
</body>
</html>
```

3. content.html:

```html
<!DOCTYPE html>
<html>
<head>
  <title>Veena Multi Specialty Hospital</title>
</head>
<body>
  <h2>Cardiology</h2>
  <p>Veena Multi Specialty Hospital provides comprehensive cardiology servic
  <ul>
    <li>Diagnosis and treatment of heart diseases</li>
    <li>Cardiac surgeries</li>
    <li>Interventional cardiology procedures</li>
    <!-- Add more information about cardiology services -->
  </ul>
</body>
</html>
```

28. Create a program to change the content of the web page using AJAX.

HTML:

```html
<!DOCTYPE html>
<html>
<head>
  <title>AJAX Example</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script src="script.js"></script>
</head>
<body>
  <h1 id="content">Initial Content</h1>
  <button id="changeButton">Change Content</button>
</body>
</html>
```

JavaScript (script.js):

```javascript
$(document).ready(function() {
  // Handle button click event
  $('#changeButton').click(function() {
    // Make an AJAX request to fetch new content
    $.ajax({
      url: 'newcontent.html', // URL of the file containing the new content
      method: 'GET',
      dataType: 'html',
      success: function(response) {
        // Update the content on the page
        $('#content').html(response);
      },
      error: function() {
        console.log('Error loading new content.');
      }
    });
  });
});
```

29. a) Write a java servlets to keep track of visit count of the page.
b) Write a servlet program to create a cookie to store and retrieve the user information in a login page.

a)

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

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

   protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      // Get the session object
      HttpSession session = request.getSession(true);

      // Get the visit count from the session or initialize it to 1
      Integer visitCount = (Integer) session.getAttribute("visitCount");
      if (visitCount == null) {
         visitCount = 1;
      } else {
         visitCount++;
      }

      // Store the updated visit count in the session
      session.setAttribute("visitCount", visitCount);

      // Display the visit count to the user
      out.println("<h1>Visit Count: " + visitCount + "</h1>");

      out.close();
   }
}
```

b)

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
```

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

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

   protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      // Get the username and password from the login form
      String username = request.getParameter("username");
      String password = request.getParameter("password");

      // Check the username and password for authentication
      if (username.equals("admin") && password.equals("password")) {
         // Create a cookie to store the username
         Cookie usernameCookie = new Cookie("username", username);

         // Set the cookie's maximum age to 1 day (in seconds)
         usernameCookie.setMaxAge(24 * 60 * 60);

         // Add the cookie to the response
         response.addCookie(usernameCookie);

         out.println("<h1>Login Successful!</h1>");
      } else {
         out.println("<h1>Login Failed!</h1>");
      }

      out.close();
   }
}
```

30. Write a Html form to enter the salary details of an employee (Name,Empid,Designation,Basic pay, DA is 10% of basic pay, HRA is 3% of basic pay). Write a servlet to calculate the net salary of employee. Display the entire details of the employee.

```
<!DOCTYPE html>
<html>
<head>
  <title>Employee Salary Details</title>
</head>
<body>
  <h1>Employee Salary Details</h1>
  <form action="SalaryServlet" method="post">
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" required><br><br>

    <label for="empId">Employee ID:</label>
    <input type="text" id="empId" name="empId" required><br><br>

    <label for="designation">Designation:</label>
    <input type="text" id="designation" name="designation" required><br><br>

    <label for="basicPay">Basic Pay:</label>
    <input type="number" id="basicPay" name="basicPay" required><br><br>

    <input type="submit" value="Calculate Net Salary">
  </form>
</body>
</html>
```

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

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

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
```

```
        // Retrieve form data
        String name = request.getParameter("name");
        String empId = request.getParameter("empId");
        String designation = request.getParameter("designation");
        double basicPay = Double.parseDouble(request.getParameter("basicPay"));

        // Calculate DA and HRA
        double da = 0.1 * basicPay;
        double hra = 0.03 * basicPay;

        // Calculate net salary
        double netSalary = basicPay + da + hra;

        // Display the employee details and net salary
        out.println("<h2>Employee Details:</h2>");
        out.println("<p>Name: " + name + "</p>");
        out.println("<p>Employee ID: " + empId + "</p>");
        out.println("<p>Designation: " + designation + "</p>");
        out.println("<p>Basic Pay: $" + basicPay + "</p>");
        out.println("<p>DA: $" + da + "</p>");
        out.println("<p>HRA: $" + hra + "</p>");
        out.println("<p>Net Salary: $" + netSalary + "</p>");

        out.close();
    }
}
```

31. Create a PHP form that collects user input for their name, email, and phone number. Perform form validation to ensure that all fields are filled out and that the email is valid. Insert the data into a MySQL database table called "users". Finally, display a message to the user indicating whether the insertion was successful or not.

```
<!DOCTYPE html>
<html>
<head>
  <title>User Registration</title>
</head>
<body>
  <?php
  // Define database connection variables
  $servername = "localhost";
  $username = "your_username";
  $password = "your_password";
  $database = "your_database";

  // Create database connection
  $conn = new mysqli($servername, $username, $password, $database);

  // Check connection
  if ($conn->connect_error) {
```

```
  die("Connection failed: " . $conn->connect_error);
 }

 // Initialize variables
 $name = $email = $phone = "";
 $message = "";

 if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // Validate form fields
  if (empty($_POST["name"]) || empty($_POST["email"]) || empty($_POST["phone"])) {
   $message = "Please fill in all fields.";
  } elseif (!filter_var($_POST["email"], FILTER_VALIDATE_EMAIL)) {
   $message = "Invalid email format.";
  } else {
   // Sanitize form data
   $name = sanitize($_POST["name"]);
   $email = sanitize($_POST["email"]);
   $phone = sanitize($_POST["phone"]);

   // Insert data into the database
   $sql = "INSERT INTO users (name, email, phone) VALUES ('$name', '$email', '$phone')";

   if ($conn->query($sql) === TRUE) {
    $message = "Data inserted successfully.";
   } else {
    $message = "Error inserting data: " . $conn->error;
   }
  }
 }

 // Function to sanitize form data
 function sanitize($data) {
  $data = trim($data);
  $data = stripslashes($data);
  $data = htmlspecialchars($data);
  return $data;
 }

 // Close the database connection
 $conn->close();
 ?>

 <h1>User Registration Form</h1>
 <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);
?>">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required><br><br>

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

```
  <label for="phone">Phone:</label>
  <input type="tel" id="phone" name="phone" required><br><br>

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

 <p><?php echo $message; ?></p>
</body>
</html>
```

32. Create a PHP form that collects user input for their address, city, and state. Perform form validation to ensure that all fields are filled out and that the state is a valid two-letter abbreviation. Insert the data into a MySQL database table called "addresses". Finally, display a message to the user indicating whether the insertion was successful or not.

```
<!DOCTYPE html>
<html>
<head>
 <title>Address Form</title>
</head>
<body>
 <?php
 // Define database connection variables
 $servername = "localhost";
 $username = "your_username";
 $password = "your_password";
 $database = "your_database";

 // Create database connection
 $conn = new mysqli($servername, $username, $password, $database);

 // Check connection
 if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
 }

 // Initialize variables
 $address = $city = $state = "";
 $message = "";

 if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // Validate form fields
  if (empty($_POST["address"]) || empty($_POST["city"]) || empty($_POST["state"])) {
   $message = "Please fill in all fields.";
  } elseif (!preg_match("/^[A-Za-z]{2}$/", $_POST["state"])) {
   $message = "Invalid state format. State should be a valid two-letter abbreviation.";
  } else {
```

```
    // Sanitize form data
    $address = sanitize($_POST["address"]);
    $city = sanitize($_POST["city"]);
    $state = sanitize($_POST["state"]);

    // Insert data into the database
    $sql = "INSERT INTO addresses (address, city, state) VALUES ('$address', '$city',
'$state')";

    if ($conn->query($sql) === TRUE) {
      $message = "Data inserted successfully.";
    } else {
      $message = "Error inserting data: " . $conn->error;
    }
  }
 }

 // Function to sanitize form data
 function sanitize($data) {
  $data = trim($data);
  $data = stripslashes($data);
  $data = htmlspecialchars($data);
  return $data;
 }

 // Close the database connection
 $conn->close();
 ?>

 <h1>Address Form</h1>
 <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);
?>">
  <label for="address">Address:</label>
  <input type="text" id="address" name="address" required><br><br>

  <label for="city">City:</label>
  <input type="text" id="city" name="city" required><br><br>

  <label for="state">State (2-letter abbreviation):</label>
  <input type="text" id="state" name="state" maxlength="2" required><br><br>

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

 <p><?php echo $message; ?></p>
</body>
</html>
```

33. Create a web application that displays a list of famous people. Allow the user to click on a person to view their biography.

Index.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Famous People List</title>
  <style>
    ul {
      list-style-type: none;
    }
    li {
      margin-bottom: 10px;
      cursor: pointer;
    }
  </style>
</head>
<body>
  <h1>Famous People</h1>
  <ul id="people-list">
    <li onclick="showBiography('Albert Einstein')">Albert Einstein</li>
    <li onclick="showBiography('Marie Curie')">Marie Curie</li>
    <li onclick="showBiography('Nelson Mandela')">Nelson Mandela</li>
    <li onclick="showBiography('Ada Lovelace')">Ada Lovelace</li>
    <li onclick="showBiography('Leonardo da Vinci')">Leonardo da Vinci</li>
  </ul>

  <div id="biography-container" style="display: none;">
    <h2 id="biography-title"></h2>
    <p id="biography-content"></p>
  </div>

  <script src="script.js"></script>
</body>
</html>
```

Script.js

```
function showBiography(person) {
  // Hide the people list
  document.getElementById("people-list").style.display = "none";

  // Show the biography container
  document.getElementById("biography-container").style.display = "block";

  // Update the biography title and content
  var titleElement = document.getElementById("biography-title");
  var contentElement = document.getElementById("biography-content");
```

```
  // Fetch the biography from a server or use hardcoded data
  var biography = getBiography(person); // Function to fetch biography data

  titleElement.textContent = person;
  contentElement.textContent = biography;
}
```

34. Create an XML document containing the details of 4 products and save it on the server. Write an Ajax program that takes a product code as input and returns the product's details by parsing the XML document.

```
<%@ page import="java.sql.*" %>
<%@ page import="javax.sql.DataSource" %>
<%@ page import="javax.naming.InitialContext" %>
<%@ page import="javax.naming.Context" %>
<%@ page import="java.util.*" %>
<%
// Declare necessary variables
Connection conn = null;
Statement stmt = null;
ResultSet rs = null;
int totalEmployees = 0;
double averageSalary = 0.0;
double highestSalary = 0.0;

try {
  // Get the database connection from the DataSource
  Context initContext = new InitialContext();
  Context envContext = (Context) initContext.lookup("java:/comp/env");
  DataSource dataSource = (DataSource) envContext.lookup("jdbc/your_datasource");

  // Establish the database connection
  conn = dataSource.getConnection();

  // Execute SQL queries
  stmt = conn.createStatement();

  // Retrieve the total number of employees
  rs = stmt.executeQuery("SELECT COUNT(*) AS total FROM employees");
  if (rs.next()) {
    totalEmployees = rs.getInt("total");
  }

  // Retrieve the average salary of all employees
  rs = stmt.executeQuery("SELECT AVG(salary) AS average FROM employees");
  if (rs.next()) {
    averageSalary = rs.getDouble("average");
  }
```

```
  // Retrieve the highest salary among all employees
  rs = stmt.executeQuery("SELECT MAX(salary) AS highest FROM employees");
  if (rs.next()) {
    highestSalary = rs.getDouble("highest");
  }
} catch (Exception e) {
  // Handle any errors that occur during the database connection or query execution
  e.printStackTrace();
} finally {
  // Close the database resources
  if (rs != null) rs.close();
  if (stmt != null) stmt.close();
  if (conn != null) conn.close();
}
%>

<!DOCTYPE html>
<html>
<head>
  <title>Employee Information</title>
</head>
<body>
  <h1>Employee Information</h1>
  <table>
    <tr>
      <th>Total Employees</th>
      <th>Average Salary</th>
      <th>Highest Salary</th>
    </tr>
    <tr>
      <td><%= totalEmployees %></td>
      <td><%= averageSalary %></td>
      <td><%= highestSalary %></td>
    </tr>
  </table>
</body>
</html>
```

35. Assume you have a database schema that contains a table called "employees" with the following columns: employee_id (int, primary key),first_name (varchar), last_name (varchar), salary (double)
Write a JSP program that connects to the database using JDBC and retrieves the following information:

    1. The total number of employees in the database.
    2. The average salary of all employees.
    3. The highest salary among all employees.

    Once you have retrieved this information, create a JSP page that displays this information in a table format.

Databaseconnecion.java

```
import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class DatabaseConnection {

   private static final String URL = "jdbc:mysql://localhost:3306/your_database_name";

   private static final String USERNAME = "your_username";

   private static final String PASSWORD = "your_password";

   public static Connection getConnection() throws SQLException {

      return DriverManager.getConnection(URL, USERNAME, PASSWORD);

   }

   public static void closeConnection(Connection connection) {

      if (connection != null) {

         try {

            connection.close();

         } catch (SQLException e) {
```

```
            e.printStackTrace();
        }
    }
}
```

Employees.jsp

```
<%@ page import="java.sql.*" %>
<%@ page import="DatabaseConnection" %>
<%
    Connection connection = null;
    PreparedStatement statement = null;
    ResultSet resultSet = null;
    try {
        // Establish database connection
        connection = DatabaseConnection.getConnection();

        // Retrieve total number of employees
        statement = connection.prepareStatement("SELECT COUNT(*) FROM employees");
        resultSet = statement.executeQuery();
        resultSet.next();
        int totalEmployees = resultSet.getInt(1);
```

```
        statement = connection.prepareStatement("SELECT AVG(salary) FROM
         employees");

        resultSet = statement.executeQuery();

        resultSet.next();

        double averageSalary = resultSet.getDouble(1);



        // Retrieve highest salary

        statement = connection.prepareStatement("SELECT MAX(salary) FROM
         employees");

        resultSet = statement.executeQuery();

        resultSet.next();

        double highestSalary = resultSet.getDouble(1);

%>



<!DOCTYPE html>

<html>

<head>

   <title>Employee Information</title>

   <style>

      table {

         border-collapse: collapse;

         width: 400px;

      }



      th, td {

         border: 1px solid black;
```

```
            padding: 8px;
        }

        th {
            background-color: #f2f2f2;
        }
    </style>
</head>
<body>
    <h1>Employee Information</h1>
    <table>
        <tr>
            <th>Total Employees</th>
            <td><%= totalEmployees %></td>
        </tr>
        <tr>
            <th>Average Salary</th>
            <td><%= averageSalary %></td>
        </tr>
        <tr>
            <th>Highest Salary</th>
            <td><%= highestSalary %></td>
        </tr>
    </table>
</body>
```

```
</html>


<%
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        // Close database resources
        DatabaseConnection.closeConnection(connection);
        if (statement != null) {
            statement.close();
        }
        if (resultSet != null) {
            resultSet.close();
        }
    }
%>
```

A small inspiration from bhalaji