**UNIVERSITY SCHOOL OF INFORMATION & COMMUNICATION TECHNOLOGY**

**GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY**

**Sector 16 C, Dwarka, Delhi, 110078**



**IT-620 Front End Design Techniques**

**Submitted To: Submitted By:**

Dr. Juhee Arora Ritika

7316404524

MCA (SE)

Section -2

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **S No.** | **Program** | **Page No** | **Sign** |
| 1 | What is frontend? Write a note on HTML, CSS, Javascript | 3 |  |
| 2 | Using HTML tags, design your own curriculum vitae. | 5 |  |
| 3 | Using HTML tags, redesign the home page of GGSIP University | 10 |  |
| 4 | Write XML program to display student profile having age like student roll no., name, age, semester, email id, phone number, department name, apply and validate using DTD. | 14 |  |
| 5 | write a javascript program to check whether an input number is palindrome number or not | 17 |  |
| 6 | Write a Generic Servlet program to display your own Enrollment number and Name using Apache Tomcat server. | 19 |  |
| 7 | Write a HTTP Servlet program to display all the HTTP Request Header parameters. | 20 |  |
| 8 | Write a HTTP Servlet program to create a Cookie. | 22 |  |
| 9 | Write a JDBC Program to fetch the employees records from the Employee table designed in MS Access/MySQL. The Table should have fields like: Employee ID, Name, DOB, Address, Department, DOJ, Position, etc. | 24 |  |
| 10 | Create your portfolio website. | 28 |  |

**Question 1**

**What is frontend? Write a note on HTML, CSS, Javascript.**

The *frontend* of a website or application refers to everything that users see and interact with directly in their web browser. It encompasses the visual layout, design, and interactive elements-such as text, images, buttons, navigation menus, and forms-that make up the user interface (UI). Frontend development, also known as client-side development, is the process of building these user-facing components using three core technologies: HTML, CSS, and JavaScript.

Frontend developers are responsible for translating design concepts into functional code, ensuring that websites are visually appealing, responsive, accessible, and provide a seamless user experience across devices[5](https://multishoring.com/blog/what-is-front-end-development/).

HTML (HyperText Markup Language)

* Definition: HTML is the standard markup language used to create and structure the content of web pages.
* Role: It provides the skeleton or framework of a webpage by defining elements such as headings, paragraphs, links, images, and lists.
* Syntax: HTML uses tags (e.g., <h1>, <p>, <div>) to mark up content. Most tags come in pairs: an opening tag and a closing tag.
* Example:

<!DOCTYPE html>

<html>

<head>

<title>Sample Page</title>

</head>

<body>

<h1>Hello, World!</h1>

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

</body>

</html>

* Features: HTML is easy to learn, widely supported, and forms the foundation for integrating other technologies like CSS and JavaScript.

CSS (Cascading Style Sheets)

* Definition: CSS is a rule-based language used to control the visual appearance and layout of HTML elements on a webpage.
* Role: It dictates styles such as colors, fonts, spacing, positioning, and responsive behaviors, making web pages aesthetically pleasing and consistent.
* Syntax: CSS rules consist of a selector (which HTML element to style) and a set of property-value pairs inside curly braces.
* Example:

h1 {

color: red;

font-size: 2.5cm;

}

p {

color: blue;

background: lightgray;

}

* Features: CSS allows developers to separate content (HTML) from presentation, making websites easier to maintain and update.

JavaScript

* Definition: JavaScript is a scripting or programming language that enables dynamic behavior and interactivity on web pages.
* Role: It allows developers to create features such as animations, form validation, interactive maps, real-time updates, and more.
* Usage: JavaScript code can be embedded directly in HTML files using the <script> tag or linked externally.
* Example:

<script>

function showMessage() {

alert(“Hello, JavaScript!”);

}

</script>

<button onclick=”showMessage()”>Click Me</button>

* Features: JavaScript is essential for modern web development, enabling complex interactions and seamless user experiences.
* Together, HTML, CSS, and JavaScript form the backbone of frontend development, allowing developers to build websites and applications that are structured, visually engaging, and interactive.

**Question 2  
Using HTML tags, design your own curriculum vitae.**

<!DOCTYPE html>

<html lang=”en”>

<head>

  <meta charset=”UTF-8”>

  <title>My Curriculum Vitae</title>

  <style>

    body { font-family: Arial, sans-serif; margin: 40px; background: #f9f9f9; }

    .container { background: #fff; padding: 30px; border-radius: 8px; max-width: 700px; margin: auto; box-shadow: 0 2px 8px rgba(0,0,0,0.1);}

    h1, h2 { color: #2c3e50; }

    h1 { border-bottom: 2px solid #3498db; padding-bottom: 10px; }

    .section { margin-bottom: 30px; }

    ul { margin: 0; padding-left: 20px; }

    .contact { font-size: 1.1em; }

  </style>

</head>

<body>

  <div class=”container”>

    <h1>Ritika</h1>

    <p class=”contact”>

      Email: ritikabhadani23@gmail.com | Phone: +91-7827480672<br>

      Address:  New Delhi, India

    </p>

    <div class=”section”>

      <h2>Career Objective</h2>

      <p>

        Motivated and detail-oriented software developer with a passion for creating efficient and user-friendly web applications. Seeking a challenging position to contribute technical skills and grow professionally.

      </p>

    </div>

    <div class=”section”>

      <h2>Education</h2>

      <ul>

        <li>

            <strong>MCA(SE)</strong><br>

            Guru Gobind Singh Indraprastha University, New Delhi<br>

            2024 – 2026 | SGPA: 7.7/10

          </li>

        <li>

          <strong>B.Sc(hons.) Computer Science</strong><br>

          University of Delhi, New Delhi<br>

          2019 – 2024 | CGPA: 8.568/10

        </li>

        <li>

          <strong>Senior Secondary (XII), CBSE</strong><br>

          Army Public School,Delhi Cantt. <br>

          2018-19 | Percentage: 81%

        </li>

      </ul>

    </div>

    <div class=”section”>

      <h2>Technical Skills</h2>

      <ul>

        <li>Programming Languages: C, C++, Java, Python</li>

        <li>Web Technologies: HTML, CSS, JavaScript, React</li>

        <li>Database: MySQL, MongoDB</li>

        <li>Tools: Git, VS Code</li>

        <li>Other: MS Excel, MS Power BI, Tableau</li>

      </ul>

    </div>

    <div class=”section”>

      <h2>Projects</h2>

      <ul>

        <li>

          <strong>BREAST CANCER DETECTION USING MACHINE LEARNING</strong></li>

         Tools: Python, Scikit-learn, Pandas, Jupyter Notebook

            Techniques: Logistic Regression, Data Preprocessing, Model Evaluation

            Developed a machine learning model to classify tumors as benign or malignant using the Breast Cancer

            Wisconsin dataset.

             Performed data cleaning, feature scaling, and exploratory data analysis to prepare the dataset for

            training.

             Implemented and compared multiple classification algorithms including Logistic Regression.

             Evaluated model performance using metrics such as accuracy, confusion matrix, and ROC-AUC score.

             Achieved high prediction accuracy and demonstrated strong understanding of end-to-end ML workflow.

<https://github.com/Ritika168/Breast-Cancer-Detection-Using-ML>

            <li>

          <strong>SHAUK FORK RESTRO</strong></li>Developed a static responsive web application catering to a restaurant’s online presence. Key features

            implemented: Menu section, Top Products showcase, Blog section, Newsletter Subscription, Location

            Integration, Countdown Timer.

             Technologies Used: HTML, CSS, Vanilla JavaScript

<https://github.com/Ritika168/ShaukForkRestro>

      </ul>

    </div>

    <div class=”section”>

      <h2>Achievements</h2>

      <ul>

        <li>Participated in SHECODE HACKATHON 2k25 organized by AIT PUNE</li>

        <li>Participated in CODERED HACKATHON 2k25 organized by AIT PUNE</li>

        <li>Active Team Member of TEDX GGSIPU since SEPTEMBER 2024 </li>

      </ul>

    </div>

    <div class=”section”>

      <h2>Personal Details</h2>

      <ul>

        <li>Date of Birth: 16th August 2001</li>

        <li>Languages Known: English, Hindi</li>

        <li>Nationality: Indian</li>

      </ul>

    </div>

    <div class=”section”>

      <h2>Declaration</h2>

      <p>

        I hereby declare that the information provided above is true to the best of my knowledge.

      </p>

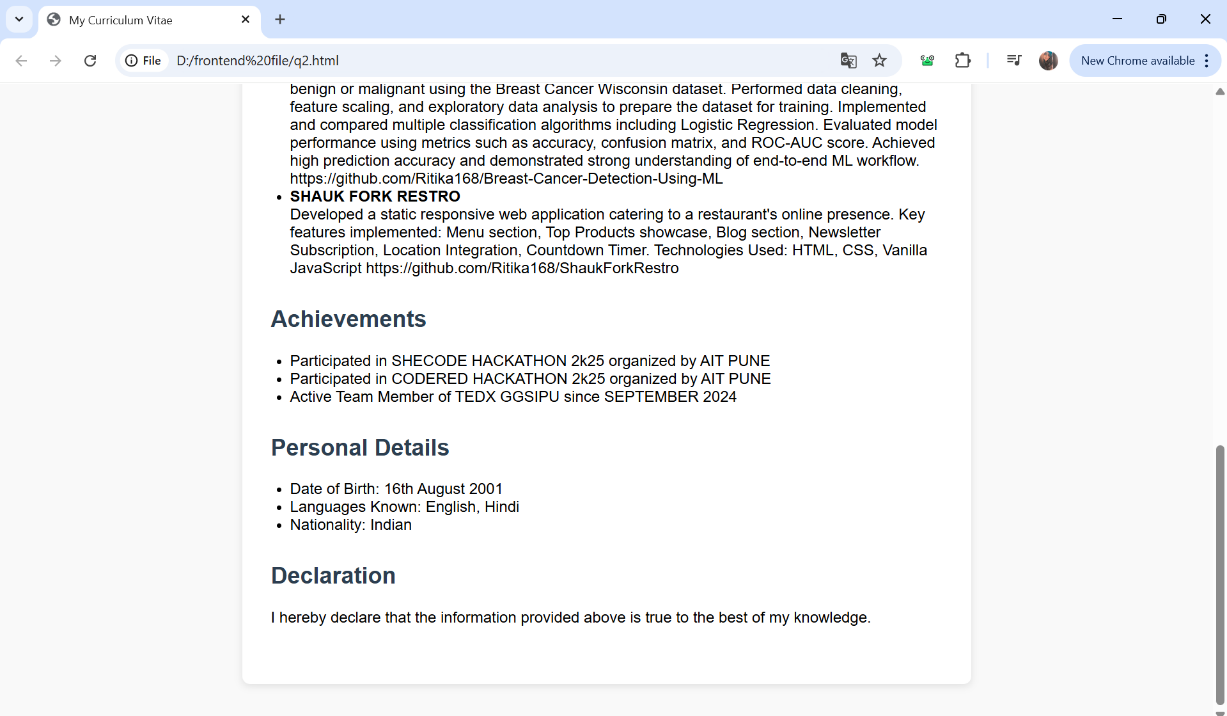
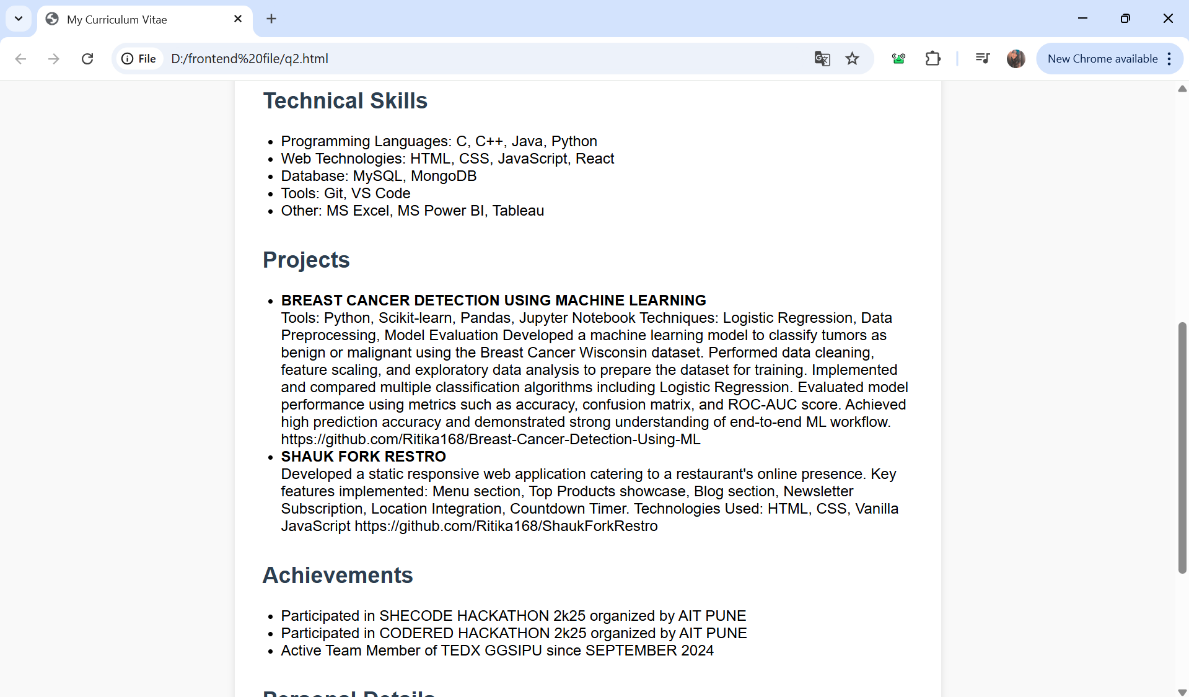
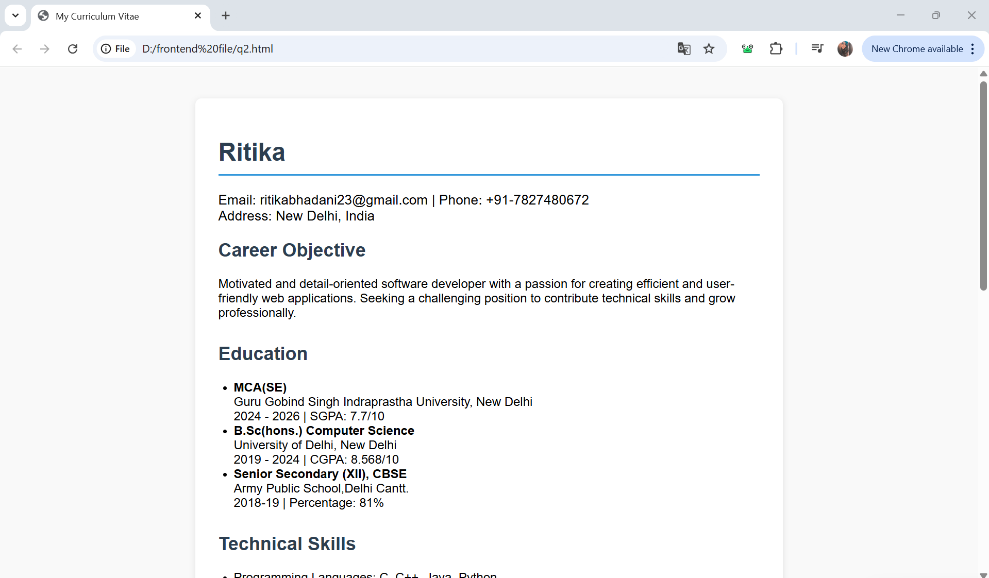
    </div>

  </div>

</body>

</html>

**OUTPUT:**

****

**Question 3**

**Using HTML tags, redesign the home page of GGSIP University**

<!DOCTYPE html>

<html lang=”en”>

<head>

  <meta charset=”UTF-8”>

  <title>Guru Gobind Singh Indraprastha University</title>

  <style>

    body { font-family: Arial, sans-serif; background: #f4f8fb; margin: 0; }

    header { background: #003366; color: #fff; padding: 20px 0; text-align: center; }

    nav { background: #0055a5; display: flex; justify-content: center; }

    nav a { color: #fff; text-decoration: none; padding: 15px 20px; display: block; }

    nav a:hover { background: #003366; }

    .container { max-width: 1100px; margin: 30px auto; background: #fff; padding: 30px; border-radius: 8px; box-shadow: 0 2px 8px rgba(0,0,0,0.08);}

    .row { display: flex; gap: 30px; flex-wrap: wrap; }

    .column { flex: 1; min-width: 250px; }

    h2 { color: #003366; border-bottom: 2px solid #0055a5; padding-bottom: 6px; }

    ul { padding-left: 20px; }

    .quick-links, .news-events { background: #eef4fa; padding: 15px; border-radius: 6px; margin-bottom: 20px; }

    footer { background: #003366; color: #fff; text-align: center; padding: 15px 0; margin-top: 40px; }

    @media (max-width: 800px) {

      .row { flex-direction: column; }

    }

  </style>

</head>

<body>

  <header>

    <h1>Guru Gobind Singh Indraprastha University</h1>

    <p>(A State University established by Govt. of NCT of Delhi)</p>

  </header>

  <nav>

    <a href=”#”>Home</a>

    <a href=”#”>Admissions</a>

    <a href=”#”>Academics</a>

    <a href=”#”>Examinations</a>

    <a href=”#”>Administration</a>

    <a href=”#”>Students</a>

    <a href=”#”>Notices</a>

    <a href=”#”>Contact</a>

  </nav>

  <div class=”container”>

    <div class=”row”>

      <div class=”column”>

        <h2>Admissions</h2>

        <ul>

          <li><a href=”#”>Admissions 2025 Main Page</a></li>

          <li><a href=”#”>CET 2025 Notices</a></li>

          <li><a href=”#”>Admission Regulatory Committee</a></li>

          <li><a href=”#”>Refund 2024</a></li>

        </ul>

        <h2>Academics</h2>

        <ul>

          <li><a href=”#”>List of Programs</a></li>

          <li><a href=”#”>Faculty Profile</a></li>

          <li><a href=”#”>Scheme/Syllabus</a></li>

          <li><a href=”#”>Schools / Centres</a></li>

          <li><a href=”#”>Library</a></li>

        </ul>

      </div>

      <div class=”column”>

        <h2>News & Events</h2>

        <div class=”news-events”>

          <ul>

            <li><a href=”#”>Convocation 2025 Announced</a></li>

            <li><a href=”#”>Seminar/Conference Schedule</a></li>

            <li><a href=”#”>Placement Cell Updates</a></li>

            <li><a href=”#”>NAD Notices</a></li>

          </ul>

        </div>

        <h2>Examinations</h2>

        <ul>

          <li><a href=”#”>Examination Notices</a></li>

          <li><a href=”#”>Examination Datesheet</a></li>

          <li><a href=”#”>Examination Results</a></li>

          <li><a href=”#”>Exam Regulations / Forms</a></li>

        </ul>

      </div>

      <div class=”column”>

        <h2>Quick Links</h2>

        <div class=”quick-links”>

          <ul>

            <li><a href=”#”>Tenders</a></li>

            <li><a href=”#”>Jobs & Opportunities</a></li>

            <li><a href=”#”>RTI Act 2005</a></li>

            <li><a href=”#”>Annual Reports</a></li>

            <li><a href=”#”>Alumni Portal</a></li>

            <li><a href=”#”>Student Grievance Committee</a></li>

            <li><a href=”#”>Moocs-IPU</a></li>

          </ul>

        </div>

        <h2>Contact</h2>

        <p>

          Sector 16C, Dwarka, New Delhi – 110078 <br>

          Phone: 011-25302170 <br>

          Email: [info@ipu.ac.in](mailto:info@ipu.ac.in)

        </p>

      </div>

    </div>

  </div>

  <footer>

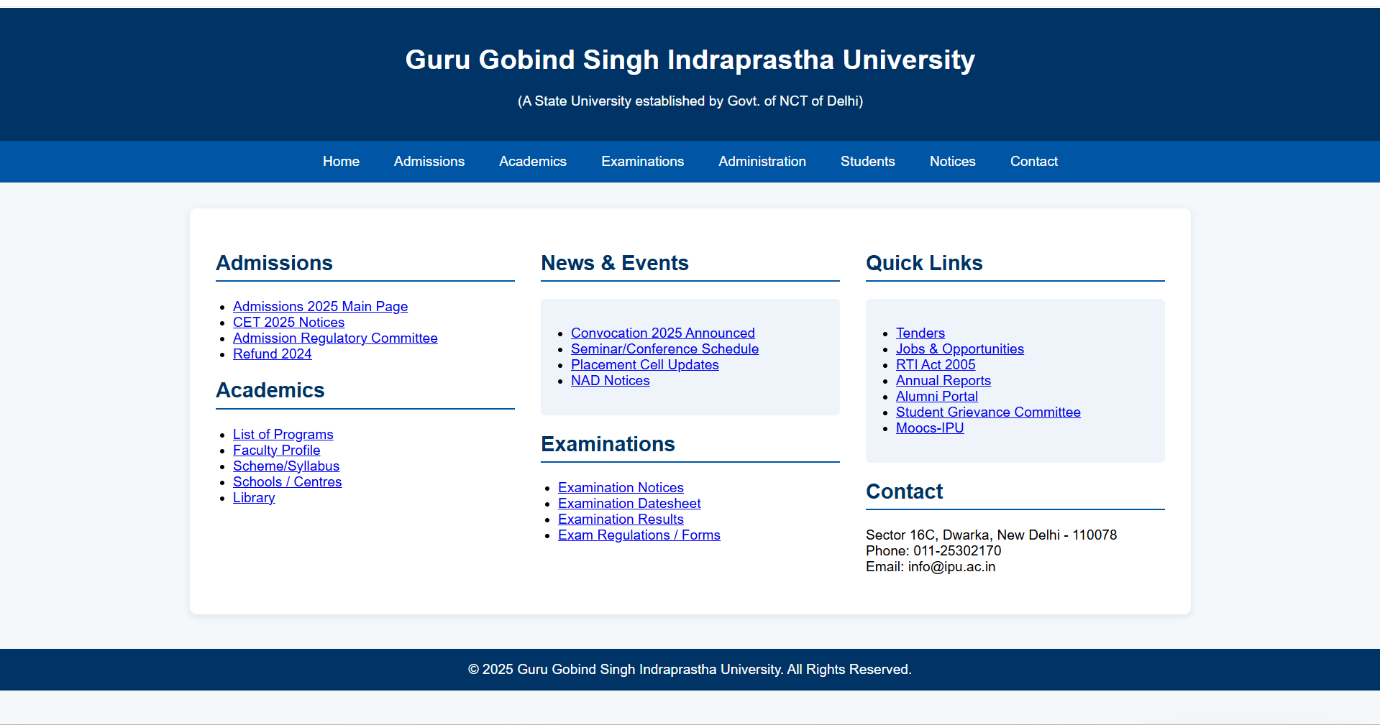
    &copy; 2025 Guru Gobind Singh Indraprastha University. All Rights Reserved.

  </footer>

</body>

</html>

**OUTPUT:**



**Question 4**

**Write XML program to display student profile having age like student roll no., name, age, semester, email id, phone number, department name, apply and validate using DTD.**

**XML**

<?xml version=”1.0” encoding=”UTF-8”?>

<!DOCTYPE students SYSTEM “q4.dtd”>

<?xml-stylesheet type=”text/css” href=”student.css”?>

<students>

    <student>

        <rollno>2023CS101</rollno>

        <name>Rahul Sharma</name>

        <age>20</age>

        <semester>4</semester>

        <email>[riya.sharma@uni.edu</email](mailto:)>

        <phone>+91-9876543210</phone>

        <department>Computer Science</department>

    </student>

    <student>

        <rollno>2023EC102</rollno>

        <name>Priya Patel</name>

        <age>21</age>

        <semester>3</semester>

        <email>[riya.patel@uni.edu</email](mailto:)>

        <phone>+91-9876543211</phone>

        <department>Electronics</department>

    </student>

</students>

**CSS**

students {

    font-family: Arial, sans-serif;

    margin: 20px;

    display: block;

}

student {

    display: block;

    border: 1px solid #ccc;

    padding: 15px;

    margin-bottom: 20px;

    border-radius: 5px;

    background-color: #f9f9f9;

}

rollno {

    color: #2c3e50;

    font-weight: bold;

    display: block;

    margin-bottom: 5px;

}

name {

    color: #2980b9;

    font-size: 1.2em;

    display: block;

    margin-bottom: 10px;

}

age, semester, email, phone, department {

    display: block;

    margin-left: 20px;

    color: #34495e;

}

department {

    color: #27ae60;

    font-weight: bold;

    margin-top: 10px;

}

**DTD**

<!ELEMENT students (student+)>

<!ELEMENT student (rollno, name, age, semester, email, phone, department)>

<!ELEMENT rollno (#PCDATA)>

<!ELEMENT name (#PCDATA)>

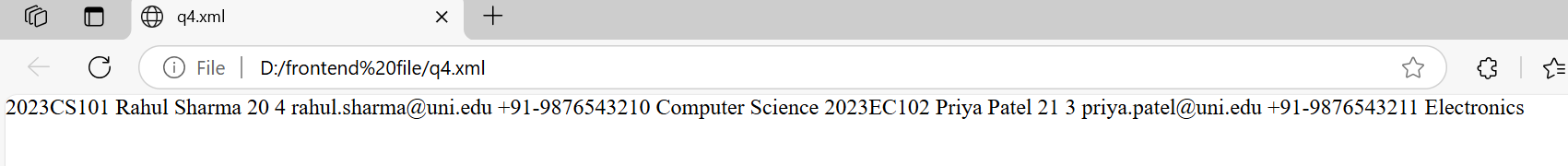
<!ELEMENT age (#PCDATA)>

<!ELEMENT semester (#PCDATA)>

<!ELEMENT email (#PCDATA)>

<!ELEMENT phone (#PCDATA)>

<!ELEMENT department (#PCDATA)>

**OUTPUT:**

**Question 5**

**Write a Javascript program to check whether an input number is palindrome number or not.**

<!DOCTYPE html>

<html>

<head>

  <title>Palindrome Number Checker</title>

  <style>

    body { font-family: Arial, sans-serif; margin: 40px; }

    input[type=”number”] { padding: 5px; font-size: 1em; }

    button { padding: 6px 14px; font-size: 1em; }

    #result { margin-top: 15px; font-weight: bold; }

  </style>

</head>

<body>

  <h2>Palindrome Number Checker</h2>

  <input type=”number” id=”num” placeholder=”Enter a number” />

  <button onclick=”checkPalindrome()”>Check</button>

  <div id=”result”></div>

  <script>

    function checkPalindrome() {

      let num = document.getElementById(‘num’).value;

      let originalNum = num;

      // Remove minus sign for negative numbers

      num = Math.abs(num).toString();

      let reversedNum = num.split(‘’).reverse().join(‘’);

      let resultDiv = document.getElementById(‘result’);

      if (num === reversedNum) {

        resultDiv.textContent = originalNum + “ is a palindrome number.”;

      } else {

        resultDiv.textContent = originalNum + “ is NOT a palindrome number.”;

      }

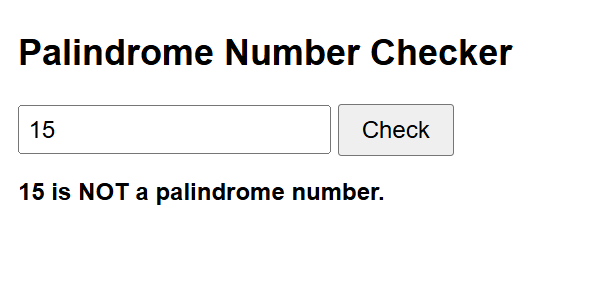
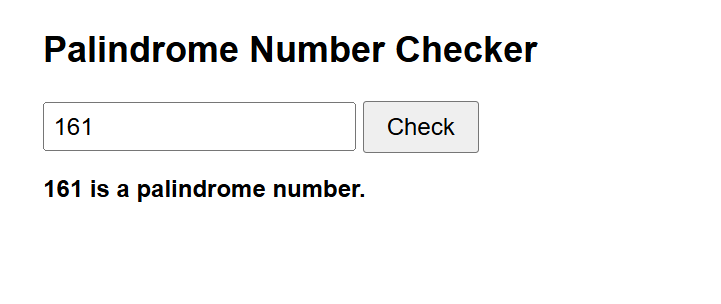
    }

  </script>

</body>

</html>

**OUTPUT:**



**Question 6**

**Write a Generic Servlet program to display your own Enrollment number and Name using Apache Tomcat server.**

package enrollment;

import java.io.IOException;

import javax.servlet.GenericServlet;

import javax.servlet.ServletException;

import javax.servlet.ServletRequest;

import javax.servlet.ServletResponse;

public class MyInfoServlet extends GenericServlet {

private static final long serialVersionUID = 1L;

@Override

public void service(ServletRequest request, ServletResponse response) throws ServletException,

IOException {

// Set content type to HTML response.setContentType("text/html");

// Output enrollment number and name response.getWriter().println("<html>");

// response.getWriter().println("<head><title>My Info</title></head>");

// response.getWriter().println("<body>");

response.getWriter().println("<h1>Enrollment Number: 07316404524</h1>");

response.getWriter().println("<h1>Name: Ritika </h1>");

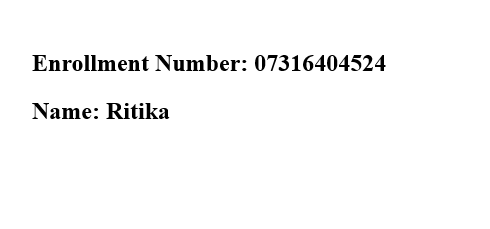
response.getWriter().println("</body>");

response.getWriter().println("</html>");

}

}

**OUTPUT:**

****

**Question 7**

**Write a HTTP Servlet program to display all the HTTP Request Header parameters.**

package parameter;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Enumeration;

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("/DisplayHeadersServlet")

public class DisplayHeadersServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* \*

\*

\* @param request\* @param response\* @throws ServletException\* @throws

\* IOException

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

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

PrintWriter out = response.getWriter();

out.println("<html>");

out.println("<head>");

out.println("<title>HTTP Request Headers</title>");

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

out.println("<body>");

out.println("<h1>HTTP Request Headers</h1>");

out.println("<table border=\"1\">");

out.println("<tr><th>Header Name</th><th>Header Value</th></tr>");

Enumeration<String> headerNames = request.getHeaderNames();

if (headerNames != null) {

while (headerNames.hasMoreElements()) {

String headerName = headerNames.nextElement();

Enumeration<String> headers = request.getHeaders(headerName);

while (headers.hasMoreElements()) {

String headerValue = headers.nextElement();

out.println("<tr><td>" + headerName + "</td><td>" + headerValue + "</td></tr>");

}

}

} else {

out.println("<tr><td colspan=\"2\">No headers present in the request.</td></tr>");

}

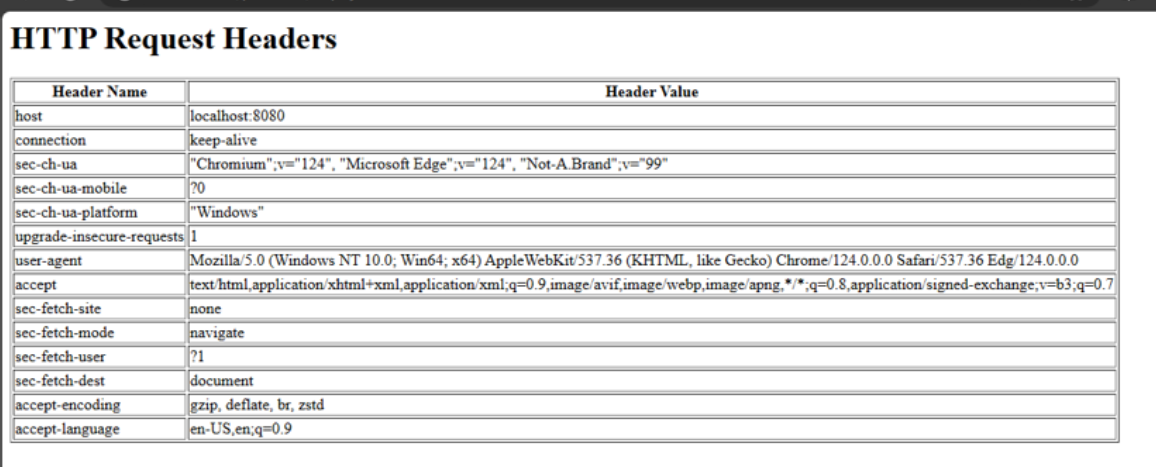
out.println("</table>");

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

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

}

}

**OUTPUT:**

**Question 8**

**Write a HTTP Servlet program to create a Cookie.**

package ck;

import java.io.IOException;

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("/CreateCookieServlet")

public class cookie extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\*

\* @param request

\* @param response

\* @throws ServletException

\* @throws IOException

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Create a new cookie

Cookie cookie = new Cookie("myCookie", "cookieValue");

// Set cookie max age (in seconds) cookie.setMaxAge(3600); // 1 hour (3600

// seconds)

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

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

response.getWriter().println("<html>");

response.getWriter().println("<head><title>Cookie Created</title></head>");

response.getWriter().println("<body>");

response.getWriter().println("<h1>Cookie Created Successfully</h1>");

response.getWriter().println(

"<p>A cookie named 'myCookie' has been created with value 'cookieValue' and max age of 1

hour.</p>");

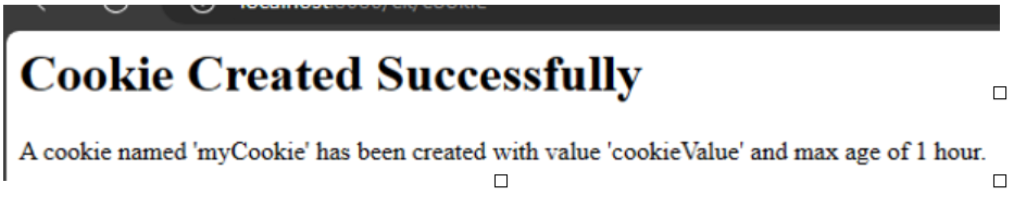
response.getWriter().println("</body>");

response.getWriter().println("</html>");

}

}

**OUTPUT:**



**Question 9**

**Write a JDBC Program to fetch the employees records from the Employee table designed in MS Access/MySQL. The Table should have fields like: Employee ID, Name, DOB, Address, Department, DOJ, Position, etc.**

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

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Employee Records</title>

</head>

<body>

<h1>Employee Records</h1>

Click here to Fetch Employee Records from Your Employee Database<br><br>

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

<input type="submit" value="Fetch Employees">

</form>

<%

java.sql.ResultSet resultSet = (java.sql.ResultSet) request.getAttribute("resultSet"); if(resultSet != null) {

%>

<table border="1"><br><br>

<tr>

<th>Employee ID</th>

<th>Employee Name</th>

<th>DOB</th>

<th>Address</th>

<th>Department</th>

<th>DOJ</th>

<th>Position</th>

<!-- Add more columns as needed -->

</tr>

<% while (resultSet.next()) { %>

<tr>

<td><%= resultSet.getInt("Employee\_id") %></td>

<td><%= resultSet.getString("Employee\_name") %></td>

<td><%= resultSet.getString("DOB") %></td>

<td><%= resultSet.getString("address") %></td>

<td><%= resultSet.getString("department") %></td>

<td><%= resultSet.getString("DOJ") %></td>

<td><%= resultSet.getString("position") %></td>

<!-- Add more columns as needed -->

</tr>

<% } %>

</table>

<% } %>

</body>

</html>

getEmployee.java

package records; import java.io.\*;

import java.sql.\*;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.\*;

@WebServlet("/FetchEmployees")

public class FetchEmployee extends HttpServlet {

private static final long serialVersionUID = 1L;

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws

ServletException, IOException {

Connection connection = null; Statement statement = null; ResultSet resultSet = null;

try {

// Load the JDBC driver Class.forName("com.mysql.cj.jdbc.Driver");

// Establish connection String url;

url = "jdbc:mysql://localhost:3306/employee";

String username = "root"; // Replace with your MySQL username String password = "12345"; // Replace

with your MySQL password connection = DriverManager.getConnection(url, username, password);

// Create SQL query

String sql = "SELECT \* FROM employee”;

// Execute query

statement = connection.createStatement(); resultSet = statement.executeQuery(sql);

// Send data to index.jsp for display request.setAttribute("resultSet", resultSet);

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

} catch (ClassNotFoundException | SQLException e) {

// Log the exception for debugging purposes log("Error fetching employee records", e);

// Send user-friendly error message response.setContentType("text/html"); PrintWriter out =

response.getWriter();

out.println("<h2>Error: Unable to fetch employee records.</h2>");

out.println("<p>Please try again later or contact the system administrator for assistance.</p>");

} finally {

// Close resources try {

if (resultSet != null) { resultSet.close();

}

if (statement != null) { statement.close();

}

if (connection != null) { connection.close();

}

} catch (SQLException e) {

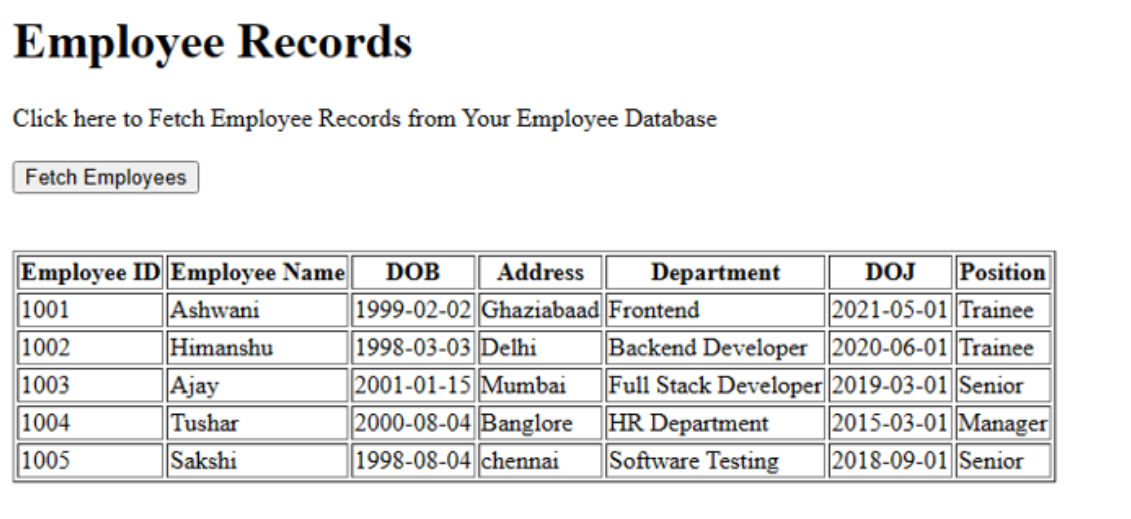
}

}

}

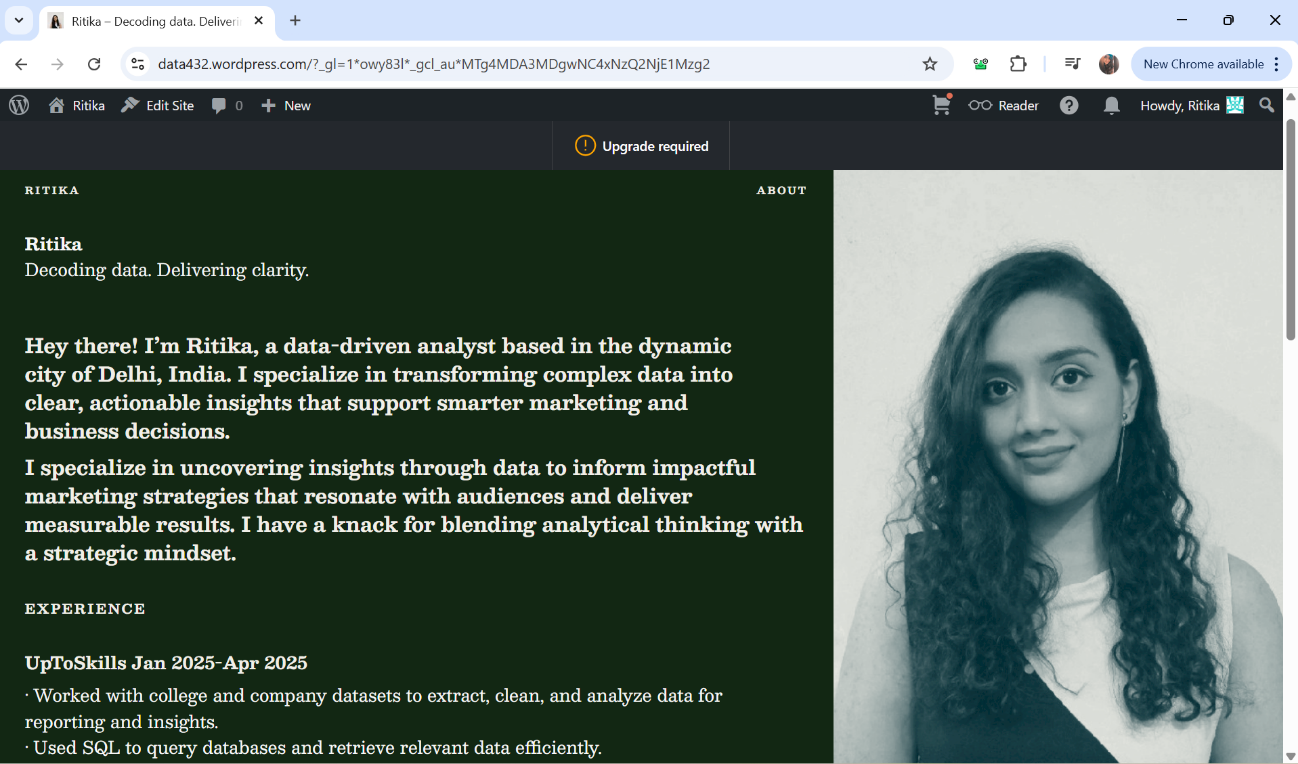
}

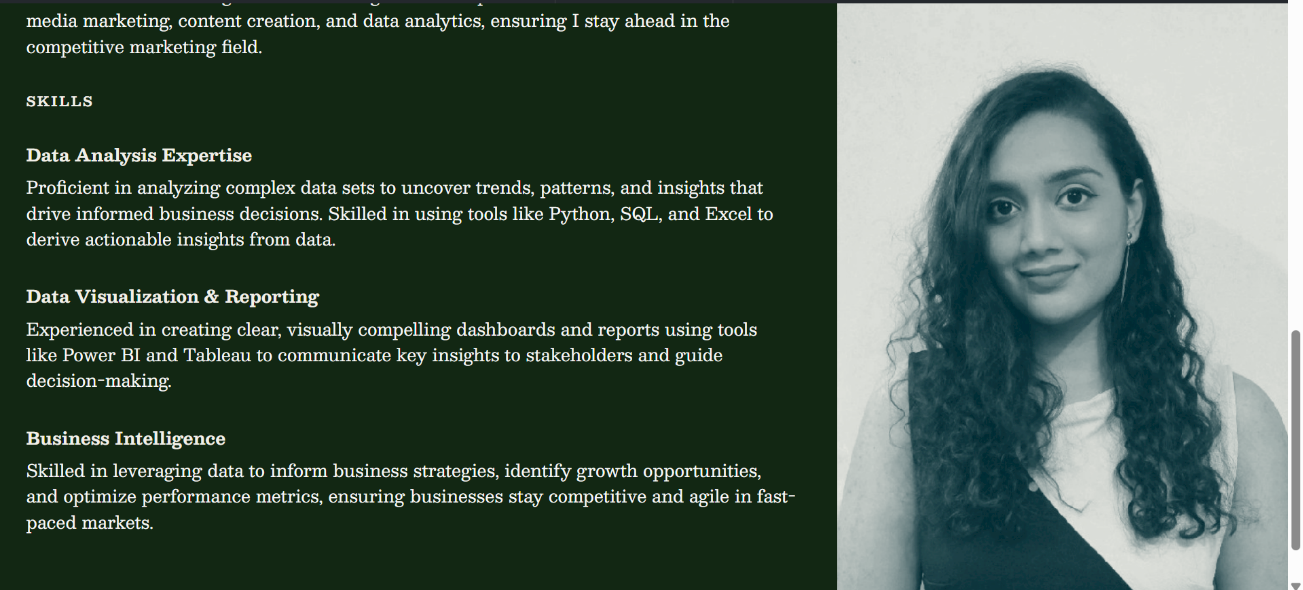
**OUTPUT:**

****

**Question 10**

**Create your portfolio website.**

****

****