Week 1 & 2

Design static web pages for home page that includes hyperlinks for registration page, login page and forgot password pages. Use form elements to create required web pages for the applications considered.

Scenario 1: Design Home page that comprises of 3 Frames. Top frame consists of Logo and title of the web page. Left frame comprises of links to different web pages and Right frame used to display the content of web pages.
Scenario 2: Left frame has links to Login page, Registration page, Contact us etc…
Scenario 3: Login page has username and password fields along with submit button, forgot password and sign-up hyperlinks.
Scenario 4: Registration page has username, password, confirm password, email-id, Mobile Number, date of birth, Address, Gender fields, submit button etc…

Description:

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It is often assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript. HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset. The window is divided into frames in a similar way the tables are organized: into rows and columns.

Program:

Index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>
</head>
<frameset rows="13%,87%">
   <frame name="top" src="top_frame.html"/>
   <frameset cols="20%,80%">
      <frame name="left" src="left_frame.html"/>
      <frame name="right" src="right_frame.html"/>
   </frameset>
</frameset>
</html>
```

Title.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
   <meta charset="utf-8">
   <title></title>
</head>
  <body class="top">
      <!--  <img style="width: 30px;height:30px;"src="images/logo.jpg" alt="">  -->
     <div class="flex">
       <img src="logo.jpg" width='50px' height='50px' /> <h1><b>GVP STUDENT MANAGEMENT
SYSTEM</b></h1>
```

```
    </div>
  </body>
</html>
```

Bottom.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>h</title>
  </head>
  <frameset cols="15%,85%">
    <frame name="left" src="left.html">
    <frame name="right" src="right.html">
  </frameset>
</html>
```

Left.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
</head>
  <body class="left">

      <a href="registration.html" target="right" class="btn">registration</a><br><br>
      <a href="login.html" target="right" class="btn">login</a><br><br>
      <a href="contact.html" target="right" class="btn">contact us</a><br>
  </body>
</html>
```

- **Create an empty file right.html it is where the href of left frame targets.**

Registration.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
</head>
  <body class="forms">
    <div class="titles">
      <h1 align="center">SIGN UP</h1>
    </div>

    <form align="center" onsubmit="validateForm()" >
      <label >Username:</label><br>
      <input type="text" id="uname" name="uname"><br>
      <label >Password:</label><br>
      <input type="password" id="pwd" name="pwd"><br>
      <label >Confirm Password:</label><br>
      <input type="password" id="cpwd" name="pwd"><br>
      <label>Email:</label><br>
      <input type="text" id="email" name="email"><br>
```

```
    <label>PhoneNo:</label><br>
    <input type="number" id="phn" name="phn"><br>
    <label>DateOfBirth</label><br>
    <input type="date" id="dob" name="dob"><br>
    <label>address:</label><br>
    <input type="text" id="add" name="add"><br>

    <label>gender</label><br>
    <input type="radio" id="gender" name="gender" value="Male">
    <label>Male</label><br>
    <input type="radio" id="gender" name="gender" value="Female">
    <label>Female</label><br>
    <input type="radio" id="gender" name="gender" value="Others">
    <label>Others</label><br><br>

    <input type="submit"name="let's go" value="let's go" class="btn"><br>
    <label><a href="login.html">login</a></label>
   </form>
  </body>
</html>
```

Login.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body class="forms">
    <div class="titles">
      <h1 align="center">LOG IN</h1>
    </div>
    <form align="center" id="myForm">
      <label>Username:</label><br>
      <input type="text" id="uname" name="uname"><br>
      <label >Password</label><br>
      <input type="password" id="pwd" name="pwd"><br><br>
      <input type="submit" onclick="clickevent()" id="submit" value="login" class="btn"><br>
      <label><a href="forpass.html">forgot password</a></label><br>
      <label><a href="registration.html" >signup</a></label>
    </form>
  </body>
</html>
```

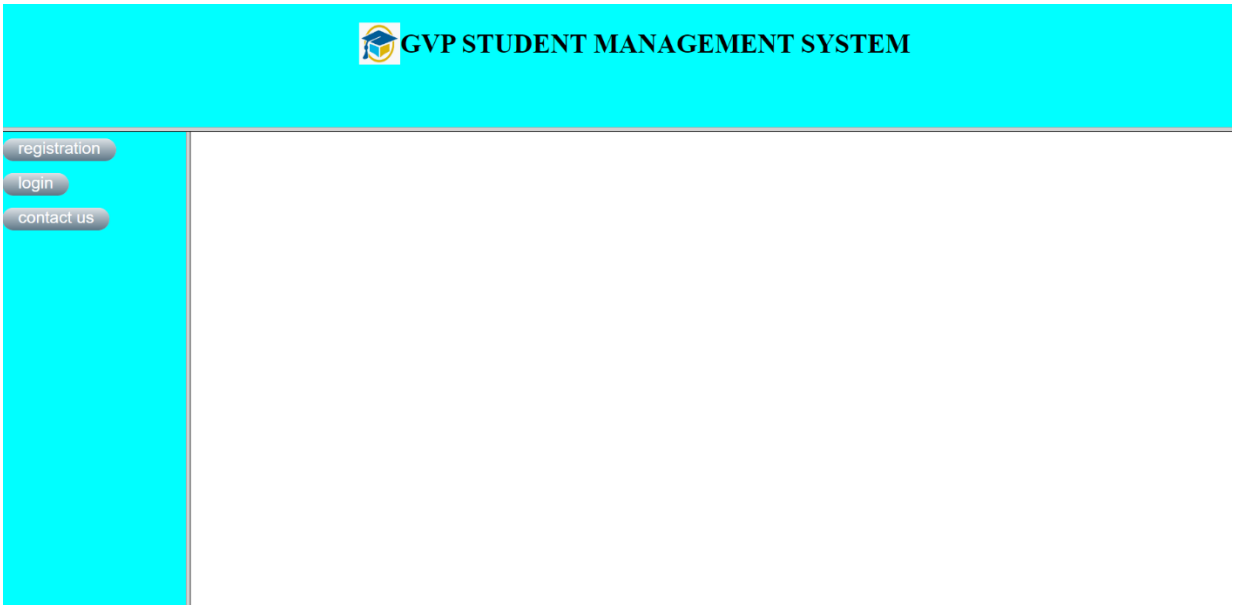
Contact.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body class="para">
    <div class="titles">
      <h1 align="center">CONTACT US</h1>
    </div>
    <p align="center"><b>
```

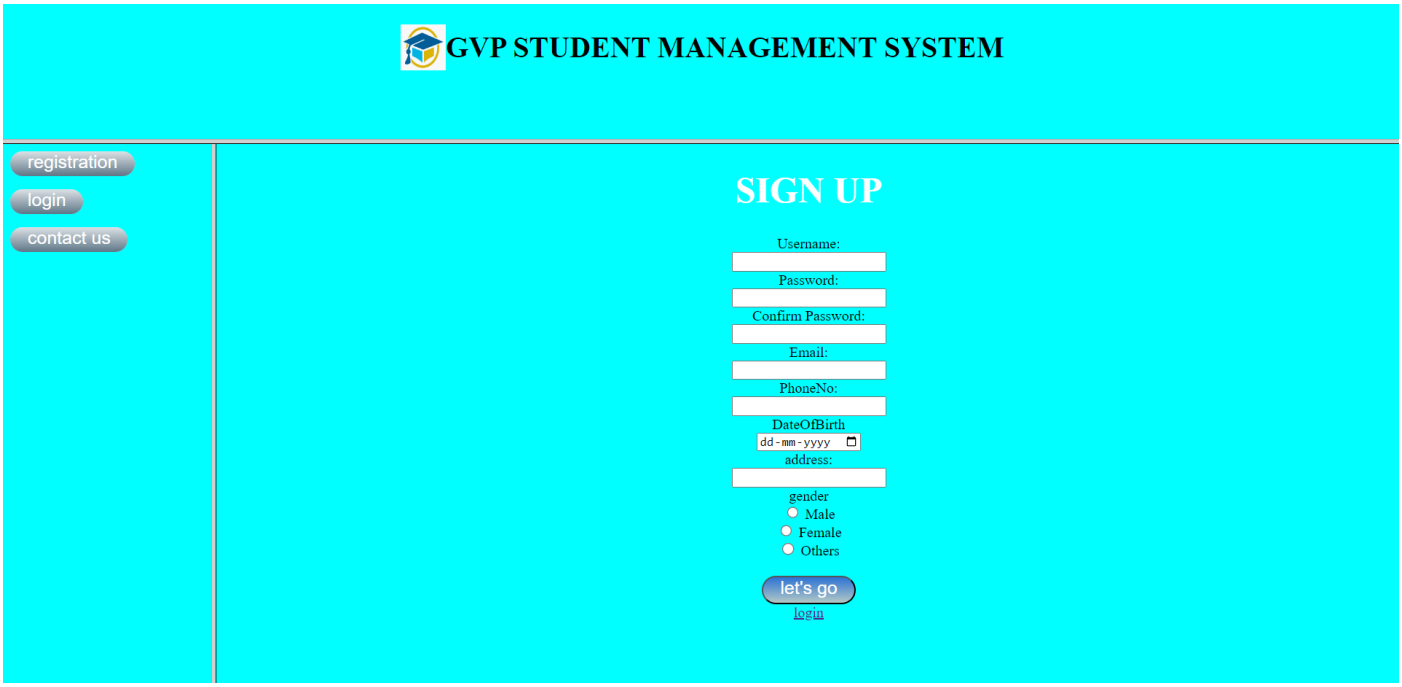
```
Gayatri vidya parishad college of engineering <br>
Madhurwada, Vishakapatnam - 530048 <br>
Andhra Prdesh <br>
India <br>
Telephone No: +91 8688597858 <br>
Email: <a href="mailto:gvpce@yahoo.com">gvpce@yahoo.com</a>
</b></p>
</body></html>
```

Output:

1) **All the three frames are created**

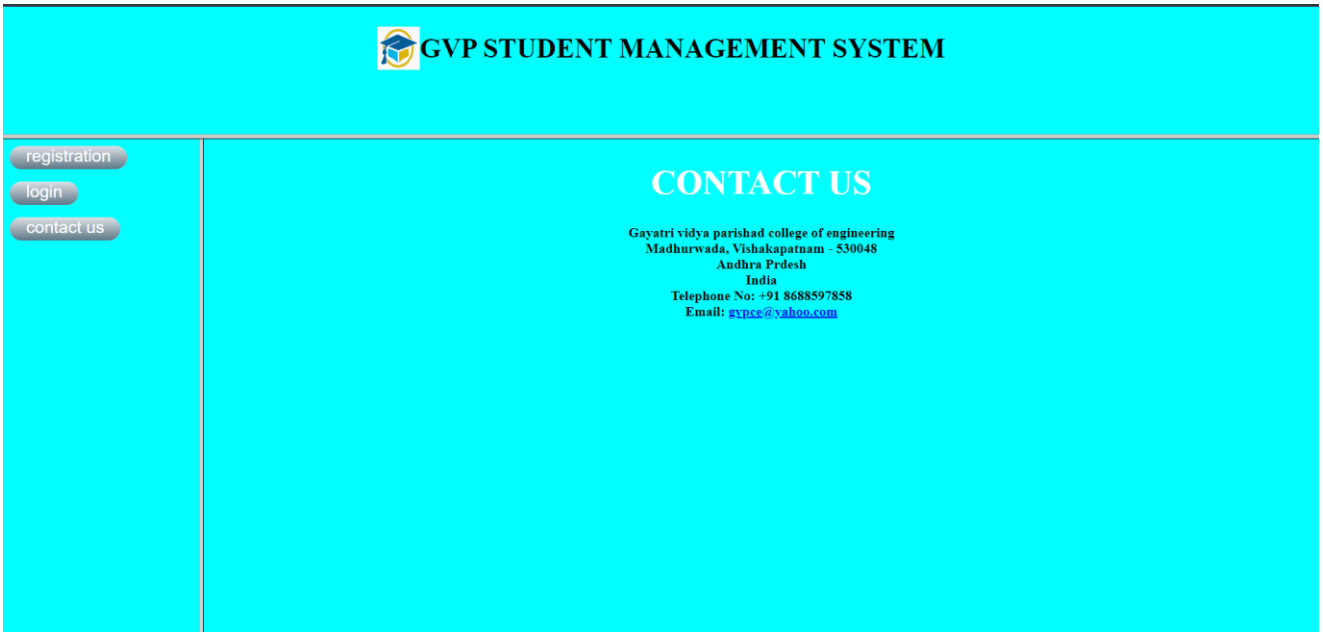


2) **Registration page:**



3) Login page:

4) Contact page:



Week 3

Apply styles using CSS to the web pages designed in Week 1 & 2.

Scenario 1: Apply styles to web pages using inline, embedded, and external style sheets.

Scenario 2: Use various background, text, font and colour styles.

Scenario 3: Apply boxes and columns styles.

Scenario 4: Use positioning and floating styles.

Program:

Here in all the codes we are implementing internal CSS as it is written in the head section of the html code

Title.html:

```
<style >
    .top{
      background-color: #00ffff;
    }
    .flex{
      display:flex;
      align-items:center;
      justify-content: center;
    }
  </style>
```

Left.html:

```
<style>
    .left{
      background-color: #00FFFF;
    }
    .btn {
      background: #dde2e6;
      background-image: -webkit-linear-gradient(top, #dde2e6, #5e7787);
      background-image: -moz-linear-gradient(top, #dde2e6, #5e7787);
      background-image: -ms-linear-gradient(top, #dde2e6, #5e7787);
      background-image: -o-linear-gradient(top, #dde2e6, #5e7787);
      background-image: linear-gradient(to bottom, #dde2e6, #5e7787);
      -webkit-border-radius: 28;
      -moz-border-radius: 28;
      border-radius: 28px;
      font-family: Arial;
      color: #ffffff;
      font-size: 20px;
      padding: 0px 19px 5px 19px;
      text-decoration: none;
    }

    .btn:hover {
      background: #3cb0fd;
      background-image: -webkit-linear-gradient(top, #3cb0fd, #3498db);
      background-image: -moz-linear-gradient(top, #3cb0fd, #3498db);
      background-image: -ms-linear-gradient(top, #3cb0fd, #3498db);
```

```
      background-image: -o-linear-gradient(top, #3cb0fd, #3498db);
      background-image: linear-gradient(to bottom, #3cb0fd, #3498db);
      text-decoration: none;
    }
  </style>
```

Login.html:

```
<style>
  .titles{
    text-align: center;
    color: white;
    padding:0px;
    font-size: 20px;
    margin-top: 0px;
  }
  .forms{
    background-color: #00ffff;
  }
  .btn {
      background: #2d73d4;
      background-image: -webkit-linear-gradient(top, #2d73d4, #b3c9c6);
      background-image: -moz-linear-gradient(top, #2d73d4, #b3c9c6);
      background-image: -ms-linear-gradient(top, #2d73d4, #b3c9c6);
      background-image: -o-linear-gradient(top, #2d73d4, #b3c9c6);
      background-image: linear-gradient(to bottom, #2d73d4, #b3c9c6);
      -webkit-border-radius: 28;
      -moz-border-radius: 28;
      border-radius: 28px;
      font-family: Arial;
      color: #ffffff;
      font-size: 20px;
      padding: 0px 19px 5px 19px;
      text-decoration: none;
    }

    .btn:hover {
      background: #c727d9;
      background-image: -webkit-linear-gradient(top, #c727d9, #7bb8bd);
      background-image: -moz-linear-gradient(top, #c727d9, #7bb8bd);
      background-image: -ms-linear-gradient(top, #c727d9, #7bb8bd);
      background-image: -o-linear-gradient(top, #c727d9, #7bb8bd);
      background-image: linear-gradient(to bottom, #c727d9, #7bb8bd);
      text-decoration: none;
    }
  </style>
```

Registration.html:

```
<style>
.titles{
  text-align: center;
  color: white;
```

```
  padding:0px;
  font-size: 20px;
  margin-top: 0px;
}
.forms{
  background-color: #00ffff;
}
.btn {
  background: #2d73d4;
  background-image: -webkit-linear-gradient(top, #2d73d4, #b3c9c6);
  background-image: -moz-linear-gradient(top, #2d73d4, #b3c9c6);
  background-image: -ms-linear-gradient(top, #2d73d4, #b3c9c6);
  background-image: -o-linear-gradient(top, #2d73d4, #b3c9c6);
  background-image: linear-gradient(to bottom, #2d73d4, #b3c9c6);
  -webkit-border-radius: 28;
  -moz-border-radius: 28;
  border-radius: 28px;
  font-family: Arial;
  color: #ffffff;
  font-size: 20px;
  padding: 0px 19px 5px 19px;
  text-decoration: none;
}

.btn:hover {
  background: #c727d9;
  background-image: -webkit-linear-gradient(top, #c727d9, #7bb8bd);
  background-image: -moz-linear-gradient(top, #c727d9, #7bb8bd);
  background-image: -ms-linear-gradient(top, #c727d9, #7bb8bd);
  background-image: -o-linear-gradient(top, #c727d9, #7bb8bd);
  background-image: linear-gradient(to bottom, #c727d9, #7bb8bd);
  text-decoration: none;
}
</style>
```

Contact.html:

```
<style>
      .titles{
        text-align: center;
        color: white;
        padding:0px;
        font-size: 20px;
        margin-top: 0px;
      }
    .para{
      background-color:#00ffff;
    }
    </style>
```

Week 4

Perform the client-side data validations using JavaScript for Scenario 3 and 4 in Week 1 & 2.
Scenario 1: In login form, define username and password constraints and ensure that the credentials follow them.
Scenario 2: In registration form, username must be of atleast 6 characters. Password must be of atleast 8 characters and follow password constraints. Password and confirm password fields must match with each other. E-mail id must be of the form "yourname@domain.com". Mobile number must be of 10 digits only and starting digit must be any number from 6-9 etc…

Program:

Login.html:

```
<script type="text/javascript">
   function clickevent(){
     var uname=document.getElementById("uname").value;
     var pwd=document.getElementById("pwd").value;
     if(uname==""||pwd==""){
       alert("all fields are necessary");
       return false;
     }
     if(pwd.length<8){
       alert("password length must be greater than 8");
       return false;
     }
     alert("login successful");
     return true;
   }
 </script>
```

Registration.html:

```
<script type="text/javascript">
function validateForm() {
  var uname = document.getElementById("uname").value;
  var pwd = document.getElementById("pwd").value;
  var cpwd = document.getElementById("cpwd").value;
  var email = document.getElementById("email").value;
  var phn = document.getElementById("phn").value;
  var dob = document.getElementById("dob").value;
  var add = document.getElementById("add").value;
  var gender = document.querySelector('input[name="gender"]:checked');
  if (uname == "" || pwd == "" || cpwd == "" || email == "" || phn == "" || dob == "" || add == "" || gender == null) {
    alert("All fields are required");
    return false;
  } else if (pwd != cpwd) {
    alert("Passwords do not match");
    return false;
  } else if (!/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {
    alert("Invalid email address");
    return false;
  } else if (!/^\d{10}$/.test(phn)) {
    alert("Invalid phone number");
    return false;
  } else {
    return true;
  }
}
</script>
```

Week-5

Create XML documents for chosen applications and validate using DTD and schema. Also render the content of XML document using XSL.

Scenario 1: XML document must have attributes and elements so that they can be validated against DTD/Schema.

Scenario 2: Check the data types of variables declared in XML document using Schema.

Scenario 3: Display the details of data contained in XML document in a table using XSL.

Program:

Data_table.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="table.xsl"?>
<!DOCTYPE student
[
<!ELEMENT student (s)*>
<!ELEMENT s (name,branch,age,city)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT branch (#PCDATA)>
<!ELEMENT age (#PCDATA)>
<!ELEMENT city (#PCDATA)>
]>
 <student xmlns:xsi=”http://www.w3.org/2001/XMLSchema-instance” xsi:schemaLocation="Schem.xsd">
  <s>
   <name> Divyank Singh Sikarwar </name>
   <branch> CSE</branch>
   <age>18</age>
   <city> Agra </city>
  </s>
  <s>
   <name> Aniket Chauhan </name>
   <branch> CSE</branch>
   <age> 20</age>
   <city> Shahjahanpur </city>
  </s>
  <s>
   <name> Simran Agarwal</name>
   <branch> CSE</branch>
   <age> 23</age>
   <city> Buland Shar</city>
  </s>
  <s>
   <name> Abhay Chauhan</name>
   <branch> CSE</branch>
   <age> 17</age>
   <city> Shahjahanpur</city>
  </s>
  <s>
   <name> Himanshu Bhatia</name>
   <branch> IT</branch>
```

```
  <age> 25</age>
  <city> Indore</city>
 </s>
</student>
```

Schem.xsd:

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="person">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="firstname" type="xs:string"/>
      <xs:element name="lastname" type="xs:string"/>
              <xs:element name="rollno" type="xs:integer"/>
              <xs:element name="dob" type="xs:date"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>
```

Table.xsl:

```
<?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>
 <body>
  <h1 align="center">Students' Basic Details</h1>
   <table border="3" align="center" >
   <tr>
    <th>Name</th>
    <th>Branch</th>
    <th>Age</th>
    <th>City</th>
   </tr>
    <xsl:for-each select="student/s">
   <tr>
    <td><xsl:value-of select="name"/></td>
    <td><xsl:value-of select="branch"/></td>
    <td><xsl:value-of select="age"/></td>
    <td><xsl:value-of select="city"/></td>
   </tr>
    </xsl:for-each>
    </table>
  </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

Week 6

Create necessary tables for the application chosen using JDBC, establish the database connectivity.
Scenario 1: Establish the connectivity using JDBC drivers.
Scenario 2: Use create and select statements.
Scenario 3: Use insert, update and delete queries.

Program:

Main.java:

```
import java.sql.*;

public class Main {

    public static void main(String[] args) {
        String jdbcUrl = "jdbc:mysql://localhost:3306/mydb";
        String dbUsername = "root";
        String dbPassword = "21131";

        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection conn = DriverManager.getConnection(jdbcUrl, dbUsername, dbPassword);

            // Scenario 1: Use create and select statements
            Statement stmt = conn.createStatement();

            // Create a new table
            String createTableQuery = "CREATE TABLE users (id INT PRIMARY KEY, name VARCHAR(50), email
VARCHAR(50))";
            stmt.executeUpdate(createTableQuery);

            // Insert some data into the table
            String insertQuery = "INSERT INTO users (id, name, email) VALUES (1, 'John Doe', 'john.doe@example.com')";
            stmt.executeUpdate(insertQuery);

            // Select data from the table
            String selectQuery = "SELECT * FROM users";
            ResultSet rs = stmt.executeQuery(selectQuery);
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                String email = rs.getString("email");
                System.out.println(id + " " + name + " " + email);
            }

            // Scenario 2: Use insert, update and delete queries
            // Insert a new user into the table
            String insertUserQuery = "INSERT INTO users (id, name, email) VALUES (2, 'Jane Smith', 'jane.smith@example.com')";
            stmt.executeUpdate(insertUserQuery);

            // Update the email address of a user
            String updateUserQuery = "UPDATE users SET email = 'john.doe@gmail.com' WHERE id = 1";
            stmt.executeUpdate(updateUserQuery);

            // Delete a user from the table
            String deleteUserQuery = "DELETE FROM users WHERE id = 2";
            stmt.executeUpdate(deleteUserQuery);
```

```
            // Display the updated table contents
            rs = stmt.executeQuery(selectQuery);
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                String email = rs.getString("email");
                System.out.println(id + " " + name + " " + email);
            }

            conn.close();
        } catch (SQLException e) {
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
    }
}
```

Week 7 & 8

Create the necessary servlets for the application chosen.

Scenario 1: Check the authenticity of the login details with the information available in database. If he is a valid user it must redirect to site resources otherwise it should stay in same page with invalid username/password message.
Scenario 2: Insert the details of the registration page into the database. If registration is successful it must display "Registration is successful".
Scenario 3: Update the password field in the database.

Program:

LoginServlet.java:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class LoginServlet extends HttpServlet {
   public void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
      // Retrieve the username and password from the request parameters
      String username = request.getParameter("username");
      String password = request.getParameter("password");

      // Connect to the database
      String jdbcUrl = "jdbc:mysql://localhost:3306/mydb";
      String dbUsername = "root";
      String dbPassword = "password";
      try {
         Class.forName("com.mysql.jdbc.Driver");
         Connection conn = DriverManager.getConnection(jdbcUrl, dbUsername, dbPassword);

         // Check if the user is valid by querying the database
         String query = "SELECT * FROM users WHERE username = ? AND password = ?";
         PreparedStatement stmt = conn.prepareStatement(query);
         stmt.setString(1, username);
         stmt.setString(2, password);
         ResultSet rs = stmt.executeQuery();

         // If the user is valid, redirect to the site resources
         if (rs.next()) {
            response.sendRedirect("resources.jsp");
         } else {
            // If the user is not valid, stay in the same page with an error message
            request.setAttribute("errorMessage", "Invalid username or password");
            RequestDispatcher dispatcher = request.getRequestDispatcher("login.jsp");
            dispatcher.forward(request, response);
         }
      } catch (SQLException e) {
         e.printStackTrace();
      } catch (ClassNotFoundException e) {
         e.printStackTrace();
      } }
}
```

RegistrationServlet.java:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class RegistrationServlet extends HttpServlet {
   public void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
      // Retrieve the registration details from the request parameters
      String username = request.getParameter("username");
      String password = request.getParameter("password");
      String email = request.getParameter("email");

      // Connect to the database
      String jdbcUrl = "jdbc:mysql://localhost:3306/mydb";
      String dbUsername = "root";
      String dbPassword = "password";
      try {
         Class.forName("com.mysql.jdbc.Driver");
         Connection conn = DriverManager.getConnection(jdbcUrl, dbUsername, dbPassword);

         // Insert the registration details into the database
         String query = "INSERT INTO users (username, password, email) VALUES (?, ?, ?)";
         PreparedStatement stmt = conn.prepareStatement(query);
         stmt.setString(1, username);
         stmt.setString(2, password);
         stmt.setString(3, email);
         int result = stmt.executeUpdate();

         // If registration is successful, display a success message
         if (result == 1) {
            request.setAttribute("successMessage", "Registration is successful");
            RequestDispatcher dispatcher = request.getRequestDispatcher("registration.jsp");
            dispatcher.forward(request, response);
         }
      } catch (SQLException e) {
         e.printStackTrace();
      } catch (ClassNotFoundException e) {
         e.printStackTrace();
      }}
   }
```

Web.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app >
<servlet>
<servlet-name>servlet1</servlet-name>
<servlet-class>Login</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>servlet1</servlet-name>
<url-pattern>/Login</url-pattern>
</servlet-mapping>
</web-app>
```

Week 9 & 10

Create the necessary JSP's for the application chosen.

Scenario 1: Use Scriptlets, JSP Action tags and Expressions.

Scenario 2: Perform the same scenarios as done in Week 7&8 using JSP Elements.

Program:

Login.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
	<title>Login</title>
</head>
<body>
	<h1>Login</h1>
	<%
	if (request.getMethod().equals("POST")) {
		// Get the parameters from the request
		String username = request.getParameter("username");
		String password = request.getParameter("password");

		// Establish a database connection
		String jdbcUrl = "jdbc:mysql://localhost:3306/mydb";
		String dbUsername = "root";
		String dbPassword = "password";
		Connection connection = DriverManager.getConnection(jdbcUrl, dbUsername, dbPassword);

		// Prepare the SQL statement to retrieve the user with the given username and password
		String sql = "SELECT * FROM users WHERE username = ? AND password = ?";
		PreparedStatement statement = connection.prepareStatement(sql);
		statement.setString(1, username);
		statement.setString(2, password);

		// Execute the SQL statement and check if a user was found
		ResultSet result = statement.executeQuery();
		if (result.next()) {
			// User was found, redirect to the home page
			response.sendRedirect("home.jsp");
		} else {
			// User was not found, display an error message
			out.println("<p>Invalid username or password</p>");
		}

		// Close the database connection and statement
		result.close();
		statement.close();
		connection.close();
	}
	%>
	<form action="login.jsp" method="post">
		<label for="username">Username:</label>
```

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

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

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

Registration.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
        <title>Registration</title>
</head>
<body>
<h1>Registration</h1>
        <%
        if (request.getMethod().equals("POST")) {
            // Get the parameters from the request
            String username = request.getParameter("username");
            String password = request.getParameter("password");
            String email = request.getParameter("email");

            // Establish a database connection
            String jdbcUrl = "jdbc:mysql://localhost:3306/mydb";
            String dbUsername = "root";
            String dbPassword = "password";
            Connection connection = DriverManager.getConnection(jdbcUrl, dbUsername, dbPassword);

            // Prepare the SQL statement to insert the user into the database
            String sql = "INSERT INTO users (username, password, email) VALUES (?, ?, ?)";
            PreparedStatement statement = connection.prepareStatement(sql);
            statement.setString(1, username);
            statement.setString(2, password);
            statement.setString(3, email);

            // Execute the SQL statement and check if registration was successful
            int result = statement.executeUpdate();
            if (result == 1) {
                // Registration was successful, display a success message
                out.println("<p>Registration is successful</p>");
            } else {
                // Registration was not successful, display an error message
                out.println("<p>Registration failed</p>");
            }

            // Close the database connection and statement
            statement.close();
            connection.close();
        }
        %>
    <form action="register.jsp" method="post">
```

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

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

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

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

</html>
```

Week 12 & 13

Embed the jQuery features for the application chosen.

Scenario 1: Perform the Scenarios of Week 4 using JQUERY ready function.

Scenario 2: Use the get and post methods for server-side communication.

Program:

Scenario 1:

For login:

```
$(document).ready(function() {
  $("#login-form").submit(function() {
    var uname = $("#uname").val();
    var pwd = $("#pwd").val();

    if(uname == "" || pwd == "") {
      alert("All fields are necessary");
      return false;
    }

    if(pwd.length < 8) {
      alert("Password length must be greater than 8");
      return false;
    }

    alert("Login successful");
    return true;
  });
});
```

For registration:

```
$(document).ready(function() {
  $("#submitBtn").click(function(event) {
    event.preventDefault(); // prevent form from submitting

    var uname = $("#uname").val();
    var pwd = $("#pwd").val();
    var cpwd = $("#cpwd").val();
    var email = $("#email").val();
    var phn = $("#phn").val();
    var dob = $("#dob").val();
    var add = $("#add").val();
    var gender = $('input[name="gender"]:checked');

    if (uname == "" || pwd == "" || cpwd == "" || email == "" || phn == "" || dob == "" || add == "" || gender.length == 0) {
      alert("All fields are required");
    } else if (pwd != cpwd) {
      alert("Passwords do not match");
    } else if (!/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {
      alert("Invalid email address");
    } else if (!/^\d{10}$/.test(phn)) {
      alert("Invalid phone number");
    } else {
      $("#myForm").submit(); // submit the form if validation passes
    }
  });
});
```

Scenario 2:

For login:

```
<!DOCTYPE html>
<html>
<head>
	<title>Login Form</title>
	<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
	<script>
		$(document).ready(function() {
			$("#login-form").submit(function(event) {
				event.preventDefault(); // prevent default form submission behavior
				var form = $(this);
				var url = form.attr("action");
				var username = form.find("#username").val();
				var password = form.find("#password").val();
				$.post(url, {username: username, password: password}, function(data) {
					if (data.success) {
						window.location.href = "home.jsp"; // redirect to home page on
successful login
					} else {
						alert("Invalid username or password"); // display error message on
failed login
					}
				});
			});
		});
	</script>
</head>
<body>
	<h1>Login Form</h1>
	<form id="login-form" action="loginServlet" method="post">
		<label for="username">Username:</label>
		<input type="text" name="username" id="username" required><br>
		<label for="password">Password:</label>
		<input type="password" name="password" id="password" required><br>
		<input type="submit" value="Login">
	</form>
</body>
</html>
```

For registration:

```
<!DOCTYPE html>
<html>
<head>
	<title>Registration Form</title>
	<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
	<script>
		$(document).ready(function() {
			$("#registration-form").submit(function(event) {
				event.preventDefault(); // prevent default form submission behavior
				var form = $(this);
				var url = form.attr("action");
				var username = form.find("#username").val();
				var password = form.find("#password").val();
				var email = form.find("#email").val();
				$.post(url, {username: username, password: password, email: email}, function(data)
					{
```

```
                                        if (data.success) {
                                                alert("Registration successful!"); // display success message on
successful registration
                                                form[0].reset(); // clear form fields
                                        } else {
                                                alert("Registration failed!"); // display error message on failed
registration
                                        }
                                });
                        });
                });
        </script>
</head>
<body>
        <h1>Registration Form</h1>
        <form id="registration-form" action="registrationServlet" method="post">
                <label for="username">Username:</label>
                <input type="text" name="username" id="username" required><br>
                <label for="password">Password:</label>
                <input type="password" name="password" id="password" required><br>
                <label for="email">Email:</label>
                <input type="email" name="email" id="email" required><br>
                <input type="submit" value="Register">
        </form>
</body>
</html>
```