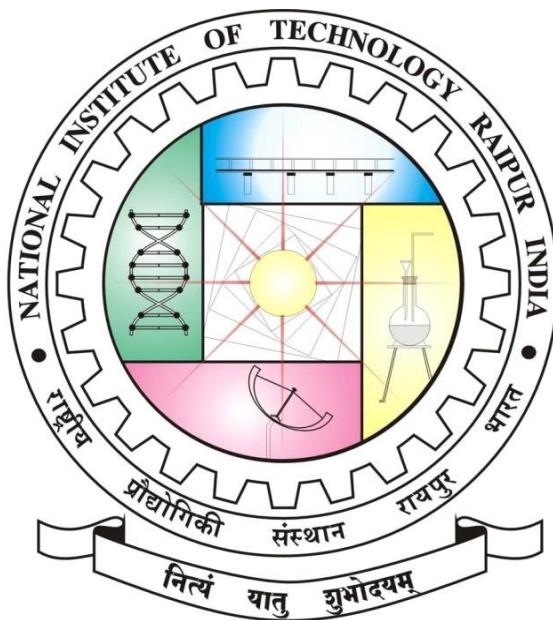


NATIONAL INSTITUTE OF TECHNOLOGY RAIPUR
DEPARTMENT OF COMPUTER APPLICATIONS



Lab Assignment Submission for
WEB TECHNOLOGY AND ADVANCE JAVA

Submitted to:

Dr. Shyam Sundar
Assistant Professor, NIT Raipur

Submitted by:

Name - Yash Shah
Roll - 21223099

Signature

Signature

INDEX

S. No.	PROGRAM
1	Design a static web site containing at least 5 pages using HTML and CSS.
2	Perform form-validation using javascript.
3	Create n (user-defined number) number of text inputs using javascript.
4	Create an XML document to store Book's records and access them using an HTML page.
5	Create a registration page and show the form-data on submission in another web page (using servlet/jsp).
6	Create a web page that stores student record into database (oracle/MySQL). Also create a show student-information page that shows those records.
7	Create a user registration page and login page. Also create a user homepage that should be redirected after successful login.
8	Use sessions to track the user across the website.
9	Store some cookie variables on user's computer and retrieve them in the next Http request.
10	Create a small single page application (SPA) using AngularJS.

1. Design a static web site containing at least 5 pages using HTML and CSS.

Page 1: index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pizza Hub</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css" />
    <link
      rel="stylesheet"
      href="https://fonts.googleapis.com/css?family=Amatic+SC" />
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <!-- Navbar -->
    <div class="w3-top w3-hide-small">
      <div
        class="w3-bar w3-xlarge w3-black w3-opacity w3-hover-opacity-off"
        id="myNavbar">
        <a href="index.html" class="w3-bar-item w3-button">HOME</a>
        <a href="menu.html" class="w3-bar-item w3-button">MENU</a>
        <a href="about.html" class="w3-bar-item w3-button">ABOUT</a>
        <a href="workinghours.html" class="w3-bar-item w3-button"
           >Working Hours</a>
        >
        <a href="contact.html" class="w3-bar-item w3-button">CONTACT</a>
      </div>
    </div>

    <!-- Header with image -->
    <header class="bgimg w3-display-container w3-grayscale-min" id="home">
      <div class="w3-display-bottomleft w3-padding">
        <span class="w3-tag w3-xlarge">Open from 10am to 12pm</span>
      </div>
      <div class="w3-display-middle w3-center">
        <span class="w3-text-white w3-hide-small" style="font-size: 100px"
              >thin<br />CRUST PIZZA</span>
        >
        <span
          class="w3-text-white w3-hide-large w3-hide-medium"
          style="font-size: 60px"
          ><b>thin<br />CRUST PIZZA</b></span>
        >
        <p>
          <a href="menu.html" class="w3-button w3-xxlarge w3-black"
             >Let me see the menu</a>
          >
        </p>
      </div>
    </header>
  </body>
</html>
```

Page 2: menu.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Pizza Hub</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css" />
    <link
        rel="stylesheet"
        href="https://fonts.googleapis.com/css?family=Amatic+SC" />
    <link rel="stylesheet" href="style.css" />
</head>
<body>
    <!-- Navbar (sit on top) -->
    <div class="w3-top w3-hide-small">
        <div
            class="w3-bar w3-xlarge w3-black w3-opacity w3-hover-opacity-off"
            id="myNavbar">
            <a href="index.html" class="w3-bar-item w3-button">HOME</a>
            <a href="menu.html" class="w3-bar-item w3-button">MENU</a>
            <a href="about.html" class="w3-bar-item w3-button">ABOUT</a>
            <a href="workinghours.html" class="w3-bar-item w3-button">Working
Hours</a>
            <a href="contact.html" class="w3-bar-item w3-button">CONTACT</a>
        </div>
    </div>

    <!-- Menu Container -->
    <div class="w3-container w3-black w3-padding-64 w3-xxlarge" id="menu">
        <div class="w3-content">
            <h1 class="w3-center w3-jumbo" style="margin-bottom: 64px">THE MENU</h1>
            <div class="w3-row w3-center w3-border w3-border-dark-grey">
                <a
                    href="javascript:void(0)"
                    onclick="openMenu(event, 'Pizza');"
                    id="myLink">
                    <div class="w3-col s4 tablink w3-padding-large w3-hover-red">
                        Pizza
                    </div>
                </a>
                <a href="javascript:void(0)" onclick="openMenu(event, 'Pasta');">
                    <div class="w3-col s4 tablink w3-padding-large w3-hover-red">
                        Salads
                    </div>
                </a>
                <a href="javascript:void(0)" onclick="openMenu(event, 'Starter');">
                    <div class="w3-col s4 tablink w3-padding-large w3-hover-red">
                        Starter
                    </div>
                </a>
            </div>
        </div>
        <!-- Pizza Menu -->
        <div id="Pizza" class="w3-container menu w3-padding-32 w3-white">
```

```
<h1>
    <b>Margherita</b>
    <span class="w3-right w3-tag w3-dark-grey w3-round">$12.50</span>
</h1>
<p class="w3-text-grey">
    Fresh tomatoes, fresh mozzarella, fresh basil
</p>
<hr />

<h1>
    <b>Formaggio</b>
    <span class="w3-right w3-tag w3-dark-grey w3-round">$15.50</span>
</h1>
<p class="w3-text-grey">
    Four cheeses (mozzarella, parmesan, pecorino, jarlsberg)
</p>
<hr />

<h1>
    <b>Chicken</b>
    <span class="w3-right w3-tag w3-dark-grey w3-round">$17.00</span>
</h1>
<p class="w3-text-grey">
    Fresh tomatoes, mozzarella, chicken, onions
</p>
<hr />

<h1>
    <b>Pineapple 'o' clock</b>
    <span class="w3-right w3-tag w3-dark-grey w3-round">$16.50</span>
</h1>
<p class="w3-text-grey">
    Fresh tomatoes, mozzarella, fresh pineapple, bacon, fresh basil
</p>
<hr />
</div>
```

```
<!-- Pasta Menu -->
```

```
<div id="Pasta" class="w3-container menu w3-padding-32 w3-white">
    <h1>
        <b>Lasagna</b> <span class="w3-tag w3-grey w3-round">Popular</span>
        <span class="w3-right w3-tag w3-dark-grey w3-round">$13.50</span>
    </h1>
    <p class="w3-text-grey">
        Special sauce, mozzarella, parmesan, ground beef
    </p>
    <hr />

    <h1>
        <b>Ravioli</b>
        <span class="w3-right w3-tag w3-dark-grey w3-round">$14.50</span>
    </h1>
    <p class="w3-text-grey">Ravioli filled with cheese</p>
    <hr />

    <h1>
```

```

        <b>Spaghetti Classica</b>
        <span class="w3-right w3-tag w3-dark-grey w3-round">$11.00</span>
    </h1>
    <p class="w3-text-grey">Fresh tomatoes, onions, ground beef</p>
    <hr />

    <h1>
        <b>Seafood pasta</b>
        <span class="w3-right w3-tag w3-dark-grey w3-round">$25.50</span>
    </h1>
    <p class="w3-text-grey">Salmon, shrimp, lobster, garlic</p>
</div>

<!-- Starters Menu -->

<div id="Starter" class="w3-container menu w3-padding-32 w3-white">
    <h1><b>Today's Soup</b> <span class="w3-tag w3-grey w3-round">Seasonal</span><span class="w3-right w3-tag w3-dark-grey w3-round">$5.50</span></h1>
    <p class="w3-text-grey">Ask the waiter</p>
    <hr>

    <h1><b>Bruschetta</b> <span class="w3-right w3-tag w3-dark-grey w3-round">$8.50</span></h1>
    <p class="w3-text-grey">Bread with pesto, tomatoes, onion, garlic</p>
    <hr>

    <h1><b>Garlic bread</b> <span class="w3-right w3-tag w3-dark-grey w3-round">$9.50</span></h1>
    <p class="w3-text-grey">Grilled ciabatta, garlic butter, onions</p>
    <hr>

    <h1><b>Tomozzarella</b> <span class="w3-right w3-tag w3-dark-grey w3-round">$10.50</span></h1>
    <p class="w3-text-grey">Tomatoes and mozzarella</p>
</div><br>

    </div>
</div>
</div>

<script>
    // Tabbed Menu
    function openMenu(evt, menuName) {
        var i, x, tablinks;
        x = document.getElementsByClassName("menu");
        for (i = 0; i < x.length; i++) {
            x[i].style.display = "none";
        }
        tablinks = document.getElementsByClassName("tablink");
        for (i = 0; i < x.length; i++) {
            tablinks[i].className = tablinks[i].className.replace(" w3-red",
            "");
        }
        document.getElementById(menuName).style.display = "block";
        evt.currentTarget.firstElementChild.className += " w3-red";
    }

```

```

document.getElementById("myLink").click();
</script>

</body>
</html>

```

Page 3: about.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Pizza Hub</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css" />
    <link
        rel="stylesheet"
        href="https://fonts.googleapis.com/css?family=Amatic+SC" />
    <link rel="stylesheet" href="style.css" />
</head>
<body>
    <!-- Navbar (sit on top) -->
    <div class="w3-top w3-hide-small">
        <div
            class="w3-bar w3-xlarge w3-black w3-opacity w3-hover-opacity-off"
            id="myNavbar">
            <a href="index.html" class="w3-bar-item w3-button">HOME</a>
            <a href="menu.html" class="w3-bar-item w3-button">MENU</a>
            <a href="about.html" class="w3-bar-item w3-button">ABOUT</a>
            <a href="workinghours.html" class="w3-bar-item w3-button">
                Working Hours</a>
            >
            <a href="contact.html" class="w3-bar-item w3-button">CONTACT</a>
        </div>
    </div>

    <!-- About Container -->
    <div
        class="w3-container w3-padding-64 w3-red w3-grayscale w3-xlarge"
        id="about">
        <div class="w3-content">
            <h1 class="w3-center w3-jumbo" style="margin-bottom: 64px">About</h1>
            <p>
                The Pizza Restaurant was founded in blabla by Mr. Italiano in lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
            </p>
            <p>
                <strong>The Chef?</strong> Mr. Italiano himself
            </p>

```

```

<p>We are proud of our interiors.</p>

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

```

Page 4: workinghours.html

```

<!DOCTYPE html>
<html>
<head>
<title>Pizza Hub</title>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css" />
<link
rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Amatic+SC" />
<link rel="stylesheet" href="style.css" />
</head>
<body>
<!-- Navbar (sit on top) -->
<div class="w3-top w3-hide-small">
<div
class="w3-bar w3-xlarge w3-black w3-opacity w3-hover-opacity-off"
id="myNavbar">
<a href="index.html" class="w3-bar-item w3-button">HOME</a>
<a href="menu.html" class="w3-bar-item w3-button">MENU</a>
<a href="about.html" class="w3-bar-item w3-button">ABOUT</a>
<a href="workinghours.html" class="w3-bar-item w3-button"
>Working Hours</a>
<br>
<a href="contact.html" class="w3-bar-item w3-button">CONTACT</a>
</div>
</div>

<div
class="w3-container w3-padding-64 w3-red w3-grayscale w3-xlarge"
id="about">
<div class="w3-content">
<h1><b>Opening Hours</b></h1>

<div class="w3-row">
<div class="w3-col s6">
<p>Mon & Tue CLOSED</p>
<p>Wednesday 10.00 - 24.00</p>
<p>Thursday 10:00 - 24:00</p>
</div>
<div class="w3-col s6">
<p>Friday 10:00 - 12:00</p>

```

```

        <p>Saturday 10:00 - 23:00</p>
        <p>Sunday Closed</p>
    </div>
</div>

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

```

Page 5: contact.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Pizza Hub</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css" />
    <link
        rel="stylesheet"
        href="https://fonts.googleapis.com/css?family=Amatic+SC" />
    <link rel="stylesheet" href="style.css" />
</head>
<body>
    <!-- Navbar (sit on top) -->
    <div class="w3-top w3-hide-small">
        <div
            class="w3-bar w3-xlarge w3-black w3-opacity w3-hover-opacity-off"
            id="myNavbar">
            <a href="index.html" class="w3-bar-item w3-button">HOME</a>
            <a href="menu.html" class="w3-bar-item w3-button">MENU</a>
            <a href="about.html" class="w3-bar-item w3-button">ABOUT</a>
            <a href="workinghours.html" class="w3-bar-item w3-button"
                >Working Hours</a>
            >
            <a href="contact.html" class="w3-bar-item w3-button">CONTACT</a>
        </div>
    </div>

    <!-- Contact -->
    <div
        class="w3-container w3-padding-64 w3-blue-grey w3-grayscale-min w3-xlarge">
        <div class="w3-content">
            <h1 class="w3-center w3-jumbo" style="margin-bottom: 64px">Contact</h1>
            <p>
                Find us at some address at some place or call us at 05050515-122330
            </p>
            <p>
                <span class="w3-tag">FYI!</span> We offer full-service catering for
                any event, large or small. We understand your needs and we will cater
            </p>
        </div>
    </div>

```

```

the food to satisfy the biggest criteria of them all, both look and
taste.
</p>
<p class="w3-xxlarge">
  <strong>Reserve</strong> a table, ask for today's special or just send
  us a message:
</p>
<form action="/action_page.php" target="_blank">
  <p>
    <input
      class="w3-input w3-padding-16 w3-border"
      type="text"
      placeholder="Name"
      required
      name="Name" />
  </p>
  <p>
    <input
      class="w3-input w3-padding-16 w3-border"
      type="number"
      placeholder="How many people"
      required
      name="People" />
  </p>
  <p>
    <input
      class="w3-input w3-padding-16 w3-border"
      type="datetime-local"
      placeholder="Date and time"
      required
      name="date"
      value="2020-11-16T20:00" />
  </p>
  <p>
    <input
      class="w3-input w3-padding-16 w3-border"
      type="text"
      placeholder="Message \ Special requirements"
      required
      name="Message" />
  </p>
  <p>
    <button class="w3-button w3-light-grey w3-block" type="submit">
      SEND MESSAGE
    </button>
  </p>
</form>
</div>
</div>
</body>
</html>
```

style.css

```
body, html {
  height: 100%
```

```

}
body,h1,h2,h3,h4,h5,h6 {
    font-family: "Amatic SC", sans-serif
}
.menu {
    display: none
}
.bgimg {
    background-repeat: no-repeat;
    background-size: cover;
    background-image:
url("https://res.cloudinary.com/dxjyi2glj/image/upload/v1680420521/pizza_j33hml.jpg");
    min-height: 100%;
}

}

```

[HOME](#) [MENU](#) [ABOUT](#) [WORKING HOURS](#) [CONTACT](#)



OPEN FROM 10AM TO 12PM

[HOME](#) [MENU](#) [ABOUT](#) [WORKING HOURS](#) [CONTACT](#)

THE MENU

Pizza

SALADS

STARTER

MARGHERITA

\$12.50

FRESH TOMATOES, FRESH MOZZARELLA, FRESH BASIL

FORMAGGIO

\$15.50

FOUR CHEESES (MOZZARELLA, PARMESAN, PECORINO, JARLSBERG)

ABOUT

The Pizza Restaurant was founded in Blabla by Mr. Italiano in lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

THE CHEF? Mr. Italiano himself

We are proud of our interiors.



OPENING HOURS

MON & TUE CLOSED

FRIDAY 10:00 - 12:00

WEDNESDAY 10:00 - 24:00

SATURDAY 10:00 - 23:00

THURSDAY 10:00 - 24:00

SUNDAY CLOSED

CONTACT

FIND US AT SOME ADDRESS AT SOME PLACE OR CALL US AT 05050515-122330

EY! WE OFFER FULL-SERVICE CATERING FOR ANY EVENT, LARGE OR SMALL. WE UNDERSTAND YOUR NEEDS AND WE WILL CATER THE FOOD TO SATISFY THE BIGGEST CRITERIA OF THEM ALL, BOTH LOOK AND TASTE.

RESERVE A TABLE, ASK FOR TODAY'S SPECIAL OR JUST SEND US A MESSAGE:

NAME

HOW MANY PEOPLE

37 33 0000 00 00 00

2. Perform form-validation using javascript.

Form_Validation.html

```
<!DOCTYPE html>
<html>
  <title>Registration Form</title>
  <head>
    <link rel="stylesheet" href="style.css" />
  </head>

  <body>
    <h1>Student Registration Form</h1>

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

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

      <label for="rolltxt">Roll Number</label>
      <input type="number" id="rolltxt" name="roll" required /><br />

      <label for="age">Age:</label>
      <input
        type="number"
        id="age"
        name="age"
        min="18"
        max="100"
        required /><br />

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

      <label for="gender">Gender:</label>
      <input type="radio" id="male" name="gender" value="male" required />
      <label for="male">Male</label>
      <input type="radio" id="female" name="gender" value="female" required />
      <label for="female">Female</label>
      <input type="radio" id="other" name="gender" value="other" required />
      <label for="other">Other</label><br />

      <label for="course">Course:</label>
      <select id="course" name="course" required>
        <option value="">--Select Course--</option>
        <option value="MCA">MCA</option>
        <option value="BTech">BTech</option>
        <option value="MTech">MTech</option></select>
    ><br />

      <input type="submit" value="Submit" />
      <input type="reset" value="reset" />
```

```
</form>

<!-- script for form validation -->
<script type="text/javascript">
    function validate() {
        let nameElement = document.getElementById("nametxt");
        let rollElement = document.getElementById("rolltxt");
        let emailElement = document.getElementById("emaitxt");
        let mobileElement = document.getElementById("mobiletxt");

        <!-- Storing value of each element -->
        let name = nameElement.value;
        let roll = rollElement.value;
        let email = emailElement.value;
        let mobile = mobileElement.value;
        let error;

        <!-- regular expression if name contains only alphabets -->
        let regExp_name = new RegExp("^[a-zA-Z]+\$");

        <!-- regular expression if roll contains only numbers and is 8 digits long -->
        let regExp_roll = new RegExp("^[0-9]{8}\$");

        <!-- regular expression if email is valid -->
        let regExp_email = new RegExp(
            "^[a-zA-Z0-9_.\\-]+@[a-zA-Z_]+\\.[a-zA-Z]{2,3}\$"
        );

        <!-- regular expression if mobile no. contains only no. and is 10 digits long -->
        let regExp_mobile = new RegExp("^[0-9]{10}\$");

        <!-- Checking if the input details matches the required valid format -->
        if (!regExp_name.test(name)) {
            alert("Provided Name is not correct.");
            return false;
        }

        if (!regExp_roll.test(roll)) {
            alert(
                "Provided Roll Number is not correct. Roll Number should be of 8
digits."
            );
            return false;
        }

        if (!regExp_email.test(email)) {
            alert("Provided Email Address is not correct.");
            return false;
        }

        if (!regExp_mobile.test(mobile)) {
            alert(
                "Provided Mobile Number is not correct. Mobile Number should be of 10
digits."
            );
            return false;
        }
    }
}
```

```
        return true;
    }
</script>
</body>
</html>
```

style.css

```
form {
    width: 50%;
    margin: 0 auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

label {
    display: inline-block;
    width: 100px;
    text-align: left;
    margin: 10px;
}

h1, h2, h3 {
    margin-top: 50px;
    text-align:center;
}

input[type="text"],
input[type="email"],
input[type="password"],
select {
    padding: 5px;
    border-radius: 3px;
    border: 1px solid #ccc;
    width: 60%;
    margin-bottom: 10px;
}

input[type="radio"] {
    margin-right: 5px;
}

input[type="submit"] {
    background-color: #4CAF50;
    color: white;
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
    margin-top: 20px;
}
```

```
input[type="submit"]:hover {  
    background-color: #45a049;  
}
```

Student Registration Form

Name:

Email:

Roll Number

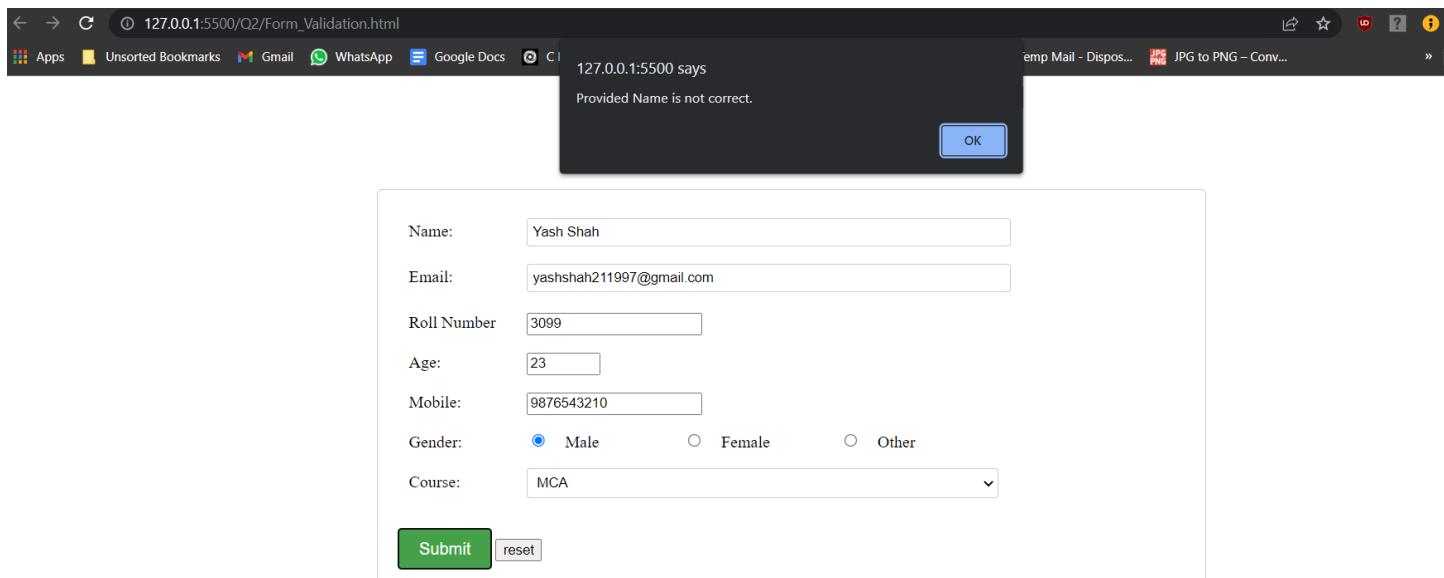
Age:

Mobile:

Gender: Male Female Other

Course: ▼

Submit reset



3. Create n (user-defined number) number of text inputs using javascript.

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>N Inputs</title>
    <link rel="stylesheet" href="style.css" />
    <script src="script.js"></script>
  </head>
  <body>
    </body>
    <h1>N-Input text fields generation using Javascript</h1>
    <form class="flex flex-column"></form>
  </body>
</html>
```

script.js

```
'use strict'
const runLoaded = () => {
  const promptMsg = "Enter how many input boxes you want";

  const promptAns = window.prompt(promptMsg);

  const submitAllInputs = (e) => {
    e.preventDefault();
    const allInputs = document.querySelectorAll(`input[type="text"]`);
    const allValues = [];
    allInputs.forEach((input) => {
```

```

        allValues.push(input.value);
    })
    alert(`All Values:\n${allValues.join('\n')}`);
    location.reload();
}

const createInputs = (n) => {
    let inputs = "";
    let inputSubmit = `<input type="submit" value="Alert">`;
    for(let i = 0; i < n; i++) {
        inputs += `<input type="text" placeholder="Enter Input - ${i+1}">`;
    }

    let form = document.querySelector("form");
    form.innerHTML = inputs + inputSubmit;

    form.addEventListener('submit', submitAllInputs);
}

try {
    const intPromptMsg = parseInt(promptAns);
    if(intPromptMsg > 0) {
        createInputs(intPromptMsg);
    } else {
        window.setTimeout(() => {
            location.reload();
        }, 5000);
    }
}
catch(err) {
    console.log(err);
}
}

document.addEventListener('DOMContentLoaded', runLoaded);

```

style.css

```

body {
    background: #f5f5f5;
    padding: 50px 20px;
}

h1{
    text-align: center;
}

form {
    width: 100%;
    max-width: 600px;
    margin: 0 auto;
    padding: 20px;
    background: #fff;
    border-radius: 5px;
    gap: 20px;
}

.container {

```

```

display: flex;
flex-direction: column;
justify-content: center;
}

.all-inputs {
  width: 100%;
  gap: 20px;
}

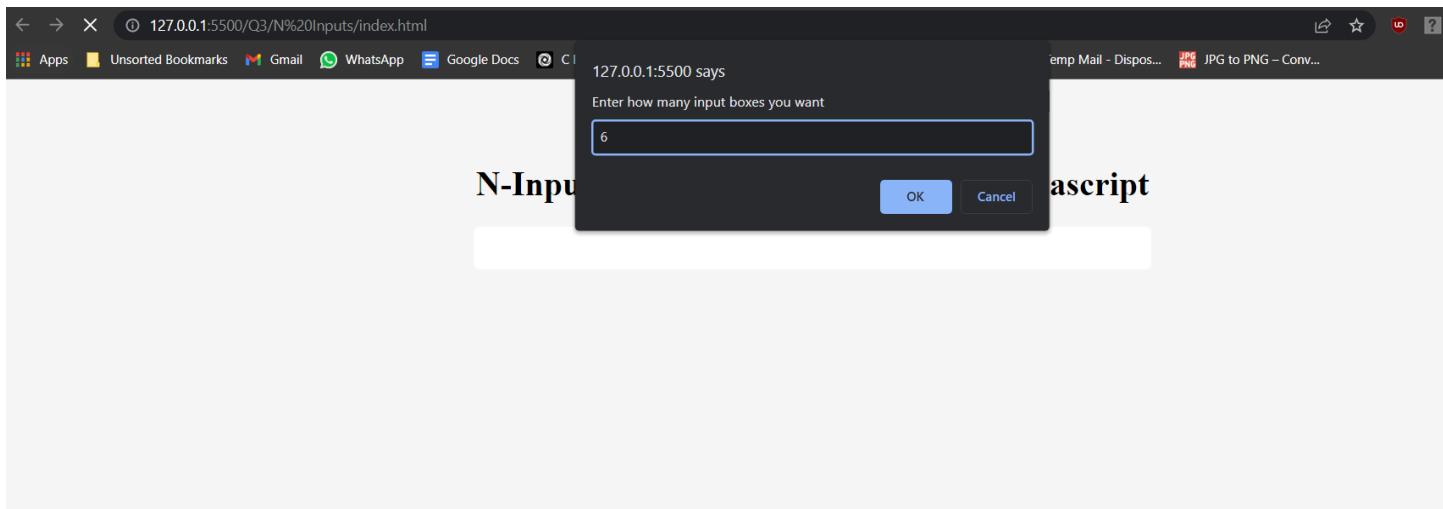
input {
  padding: 10px 15px;
  border: 1px solid #e0e0e0;
  border-radius: 5px;
  background: #f5f5f5;
  width: 100%;
  margin: 10px;
}

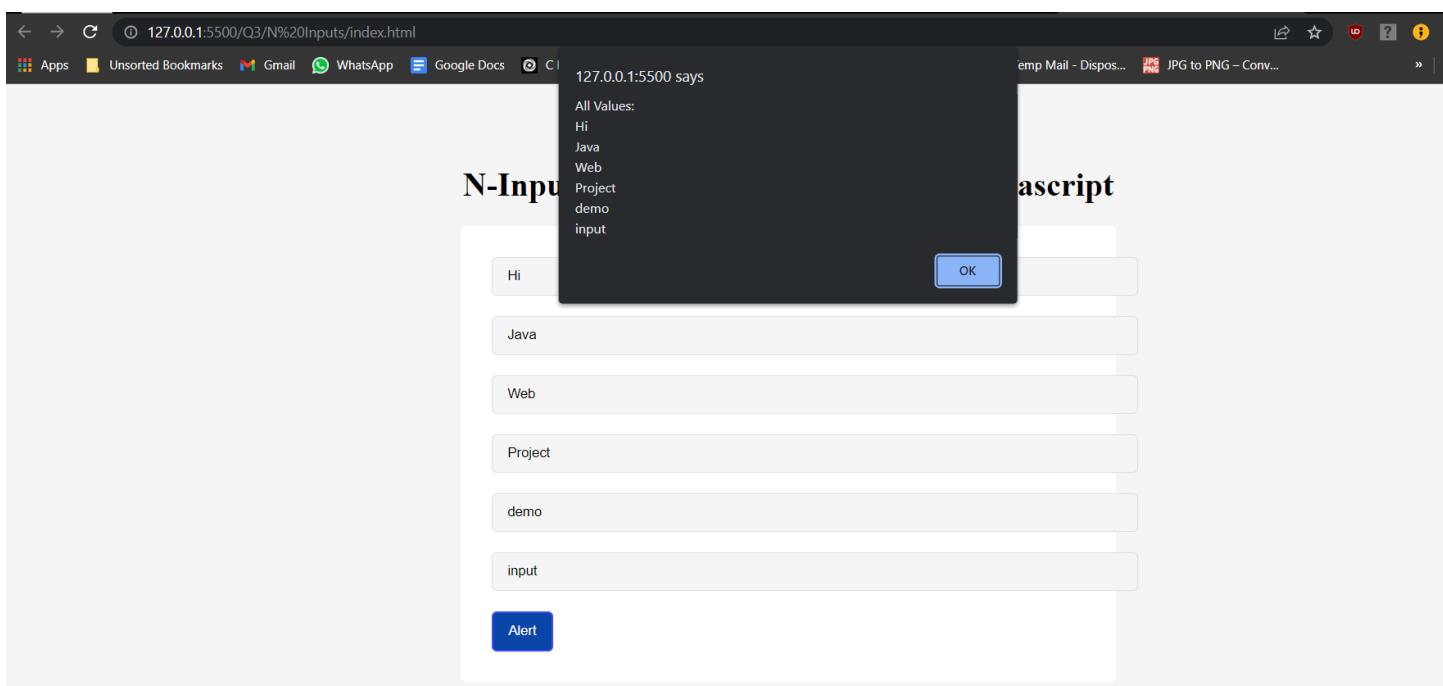
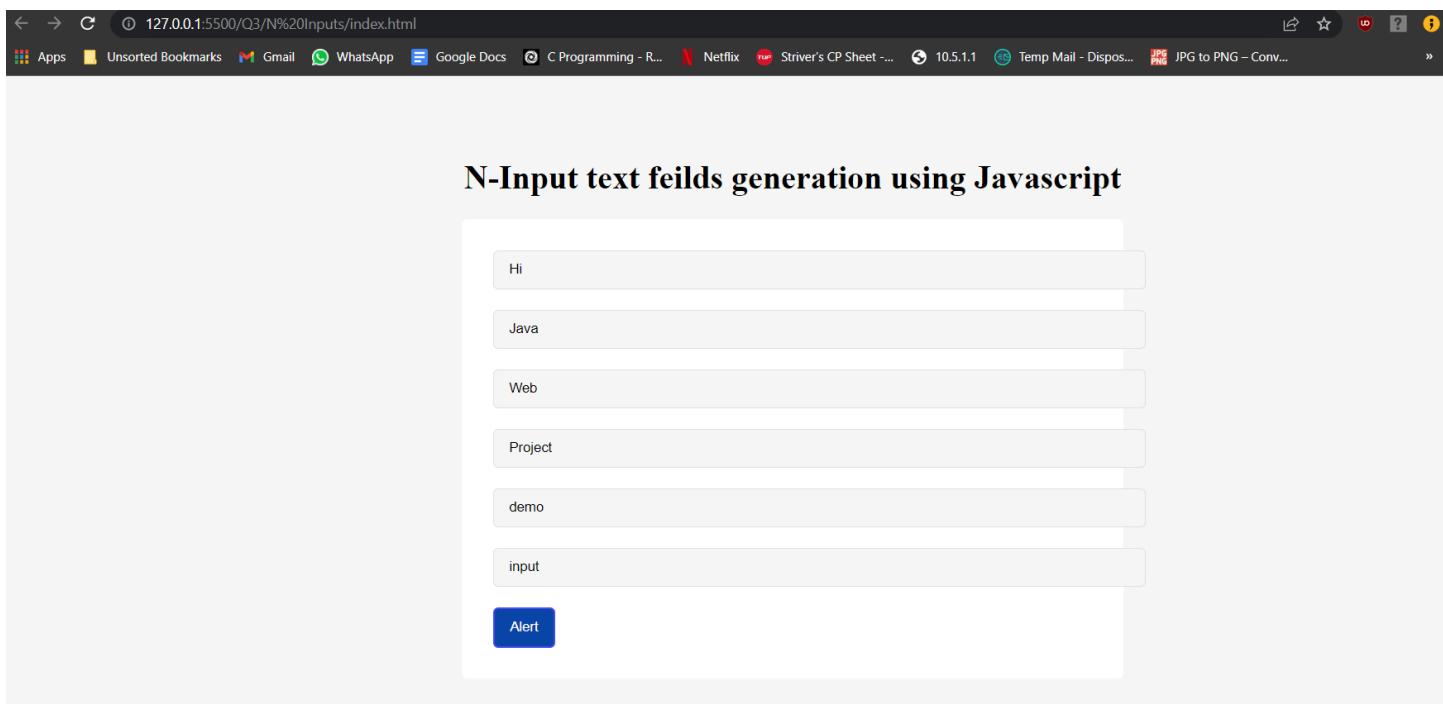
input:hover {
  border: 1px solid #c0c0c0;
}

/* input:focus,
select:focus,
textarea:focus,
button:focus {
  outline: 1px ridge #e0e0e0;
} */

input[type="submit"] {
  background-color: rgb(9, 69, 166);
  border: 2px solid rgb(69, 72, 244);
  width: fit-content;
  margin: 10px;
  color: #fff;
}

```





4. Create an XML document to store Book's records and access them using an HTML page.

index.html

```
<!DOCTYPE html>
<html>
  <head>
```

```

<title>My Library</title>
<style>
  table {
    border-collapse: collapse;
    width: 100%;
  }
  th,
  td {
    text-align: left;
    padding: 8px;
    border: 1px solid #ddd;
  }
  th {
    background-color: #4caf50;
    color: white;
  }
  tr:nth-child(even) {
    background-color: #f2f2f2;
  }
</style>

```

<!-- In this, the loadBooks() function uses an XMLHttpRequest to read the XML file, then extracts the book records and builds an HTML table to display them in the web page. The onload event of the <body> tag triggers the loadBooks() function when the page loads. -->

```

<script>
  function loadBooks() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function () {
      if (this.readyState == 4 && this.status == 200) {
        var xmlDoc = this.responseXML;
        var books = xmlDoc.getElementsByTagName("book");
        var table = "";
        for (var i = 0; i < books.length; i++) {
          var title =
            books[i].getElementsByTagName("title")[0].childNodes[0]
              .nodeValue;
          var author =
            books[i].getElementsByTagName("author")[0].childNodes[0]
              .nodeValue;
          var genre =
            books[i].getElementsByTagName("genre")[0].childNodes[0]
              .nodeValue;
          var year =
            books[i].getElementsByTagName("year")[0].childNodes[0]
              .nodeValue;
          var price =
            books[i].getElementsByTagName("price")[0].childNodes[0]
              .nodeValue;
          table += "<tr><td>" +
            title +
            "</td><td>" +
            author +
            "</td><td>" +
            genre +

```

```

        "</td><td>" +
        year +
        "</td><td>" +
        price +
        "</td></tr>" ;
    }
    document.getElementById("books").innerHTML = table;
}
};

xhttp.open("GET", "books.xml", true);
xhttp.send();
}

</script>
</head>
<body onload="loadBooks()">
<h1>My Library</h1>
<table>
<thead>
<tr>
<th>Title</th>
<th>Author</th>
<th>Genre</th>
<th>Year</th>
<th>Price</th>
</tr>
</thead>
<tbody id="books"></tbody>
</table>
</body>
</html>

```

books.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<library>
<book>
<title>The Lord of the Rings</title>
<author>J.R.R. Tolkien</author>
<genre>Fantasy</genre>
<year>1955</year>
<price>260 Rs</price>
</book>
<book>
<title>The Hunger Games</title>
<author>J.Suzanne Collins</author>
<genre>Dystopian</genre>
<year>2008</year>
<price>220 Rs</price>
</book>
<book>
<title>To Kill a Mockingbird</title>
<author>Harper Lee</author>
<genre>Classic</genre>
<year>1960</year>
<price>230 Rs</price>
</book>
<book>

```

```

<title>The Girl with the Dragon Tattoo</title>
<author>HStieg Larsson</author>
<genre>Crime Thriller</genre>
<year>2005</year>
<price>200 Rs</price>
</book>
<book>
    <title>The Hitchhiker's Guide to the Galaxy</title>
    <author>Douglas Adams</author>
    <genre>Comedic Fiction</genre>
    <year>1979</year>
    <price>250 Rs</price>
</book>
<book>
    <title>Pride and Prejudice</title>
    <author>Jane Austen</author>
    <genre>Classic</genre>
    <year>1813</year>
    <price>320 Rs</price>
</book>
</library>

```

My Library

Title	Author	Genre	Year	Price
The Lord of the Rings	J.R.R. Tolkien	Fantasy	1955	260 Rs
The Hunger Games	JSuzanne Collins	Dystopian	2008	220 Rs
To Kill a Mockingbird	Harper Lee	Classic	1960	230 Rs
The Girl with the Dragon Tattoo	HStieg Larsson	Crime Thriller	2005	200 Rs
The Hitchhiker's Guide to the Galaxy	Douglas Adams	Comedic Fiction	1979	250 Rs
Pride and Prejudice	Jane Austen	Classic	1813	320 Rs

5. Create a registration page and show the form-data on submission in another web page (using servlet/jsp).

index.html

```

<!DOCTYPE html>
<html>
    <head>
        <title>Input Form</title>
        <link rel="stylesheet" type="text/css" href="style.css">
    </head>
    <body>
        <h1> Student Registration Form </h1>

        <form action="RegisterServlet" method="POST">
            <label for="name">Name:</label>
            <input type="text" id="nametxt" name="name" required /><br />

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

```

```

        <label for="rolltxt">Roll Number</label>
        <input type="number" id="rolltxt" name="roll" required /><br />

        <label for="age">Age:</label>
        <input
            type="number"
            id="age"
            name="age"
            min="18"
            max="100"
            required /><br />

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

        <label for="gender">Gender:</label>
        <input type="radio" id="male" name="gender" value="male" required />
        <label for="male">Male</label>
        <input type="radio" id="female" name="gender" value="female" required
/>
        <label for="female">Female</label>
        <input type="radio" id="other" name="gender" value="other" required
/>
        <label for="other">Other</label><br />

        <label for="course">Course:</label>
        <select id="course" name="course" required>
            <option value="">--Select Course--</option>
            <option value="MCA">MCA</option>
            <option value="BTech">BTech</option>
            <option value="MTech">MTech</option>
        </select> <br />

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

```

style.css

```

form {
    width: 50%;
    margin: 0 auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

label {
    display: inline-block;
    width: 100px;
    text-align: left;
    margin: 10px;
}

```

```

h1, h2, h3 {
    margin-top: 50px;
    text-align:center;
}

input[type="text"],
input[type="email"],
input[type="password"],
select {
    padding: 5px;
    border-radius: 3px;
    border: 1px solid #ccc;
    width: 60%;
    margin-bottom: 10px;
}

input[type="radio"] {
    margin-right: 5px;
}

input[type="submit"] {
    background-color: #4CAF50;
    color: white;
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
    margin-top: 20px;
}

input[type="submit"]:hover {
    background-color: #45a049;
}

```

RegisterServlet.java

```

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 RegisterServlet extends HttpServlet {

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

            // fetch the form-data send as http request parameter

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

```

```

String email = request.getParameter("email");
String mobile = request.getParameter("mobile");
String age = request.getParameter("age");
String gender = request.getParameter("gender");
String course = request.getParameter("course");

response.setContentType("text/html");
PrintWriter pw = response.getWriter();

// Display the entered form-data on a new webpage

pw.println("<html>");
pw.println("<head>");
pw.println("<link rel=\"stylesheet\" type=\"text/css\" href=\"formServletStyle.css\" />");
pw.println("</head>");
pw.println("<body>");
pw.println("<div class=\"container\">");
pw.println("<h2>User Registered Successfully!</h2>");
pw.println("<h1><u>User Details:</u></h1>");
pw.println("<b> Name : </b>" + name + "<br>");
pw.println("<b> Age : </b>" + age + "<br>");
pw.println("<b> Gender : </b>" + gender + "<br>");
pw.println("<b> Roll Number : </b>" + roll + "<br>");
pw.println("<b> Course Selected : </b>" + course + "<br>");
pw.println("<b> Email Address : </b>" + email + "<br>");
pw.println("<b> Mobile Number : </b>" + mobile + "<br>");
pw.println("</div>");
pw.println("</body> </html>");

pw.close();
} catch (Exception e) {
    e.printStackTrace();
}
}
}
}

```

Student Registration Form

Name:	<input type="text" value="Yash Shah"/>
Email:	<input type="text" value="yashshah211997@gmail.com"/>
Roll Number	<input type="text" value="21223099"/>
Age:	<input type="text" value="23"/>
Mobile:	<input type="text" value="9876543210"/>
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Course:	<input type="text" value="MCA"/>
<input type="button" value="Submit"/> <input type="button" value="reset"/>	

User Registered Successfully!

User Details:

Name : Yash Shah
Age : 23
Gender : male
Roll Number : 21223099
Course Selected : MCA
Email Address : yashshah211997@gmail.com
Mobile Number : 9876543210

6. Create a web page that stores student record into database (oracle/MySQL). Also create a show student-information page that shows those records.

index.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Student Registration</title>
        <link rel="stylesheet" type="text/css" href="style.css">
    </head>
    <body>
        <h1> Student Registration Form </h1>

        <form action="RegisterServlet" method="POST">
            <label for="name">Name:</label>
            <input type="text" id="nametxt" name="name" required /><br />

            <label for="dob">Date of Birth:</label>
            <input type="date" id="dobtxt" name="dob" required /><br />

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

            <label for="rolltxt">Roll Number</label>
            <input type="number" id="rolltxt" name="roll" required /><br />

            <label for="age">Age:</label>
            <input
                type="number"
                id="age"
                name="age"
                min="18"
                max="100"
                required /><br />

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

            <label for="gender">Gender:</label>
            <input type="radio" id="male" name="gender" value="male" required />
            <label for="male">Male</label>
            <input type="radio" id="female" name="gender" value="female" required />
            <label for="female">Female</label>
            <input type="radio" id="other" name="gender" value="other" required />
            <label for="other">Other</label><br />

            <label for="course">Course:</label>
            <select id="course" name="course" required>
                <option value="">--Select Course--</option>
                <option value="MCA">MCA</option>
            </select>
        </form>
    </body>
</html>
```

```

        <option value="BTech">BTech</option>
        <option value="MTech">MTech</option>
    </select> <br />

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

```

style.css

```

form {
    width: 50%;
    margin: 0 auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

label {
    display: inline-block;
    width: 100px;
    text-align: left;
    margin: 10px;
}

h1, h2, h3 {
    margin-top: 50px;
    text-align:center;
}

input[type="text"],
input[type="email"],
input[type="password"],
select {
    padding: 5px;
    border-radius: 3px;
    border: 1px solid #ccc;
    width: 60%;
    margin-bottom: 10px;
}

input[type="radio"] {
    margin-right: 5px;
}

input[type="submit"] {
    background-color: #4CAF50;
    color: white;
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
    margin-top: 20px;
}

```

```
input[type="submit"]:hover {  
    background-color: #45a049;  
}
```

ConnectionProvider.java

```
// User-defined class to establish connection with the database using JDBC  
  
package Database;  
  
import java.sql.Connection;  
import java.sql.DriverManager;  
  
public class ConnectionProvider {  
  
    private static Connection con;  
  
    public static Connection getConnection() {  
  
        try {  
            if (con == null) {  
                Class.forName("com.mysql.cj.jdbc.Driver");  
                String url =  
"jdbc:mysql://localhost:3306/student?zeroDateTimeBehavior=CONVERT_TO_NULL";  
                String username = "root";  
                String password = "admin";  
  
                //create a connection  
                con = DriverManager.getConnection(url, username, password);  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
        return con;  
    }  
}
```

RegisterServlet.java

```
package Database;  
  
import java.io.IOException;  
import java.io.PrintWriter;  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.Statement;  
import java.util.Date;  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;
```

```
public class RegisterServlet extends HttpServlet {

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

        // Connect to the database
        Connection con = ConnectionProvider.getConnection();

        PreparedStatement st = con.prepareStatement("select * from
studentdetails where roll=?");
        st.setInt(1, Integer.parseInt(request.getParameter("roll")));

        //Run SQL query to fetch results from database

        ResultSet rs1 = st.executeQuery();

        out.println("<html><body style='display:flex; justify-content:center;
align-items:center; flex-direction:column; width:100vw; height:100vh;'>");

        //condition for checking student already exist

        if (!rs1.next()) {

            // Create a SQL query to insert data into studentdetails table

            st = con.prepareStatement("insert into studentdetails
values(?,?,?,?,?,?,?,?,?)");
            st.setInt(1, Integer.parseInt(request.getParameter("roll")));
            st.setString(2, request.getParameter("name"));
            st.setString(3, request.getParameter("email"));
            st.setString(4, request.getParameter("dob"));
            st.setInt(5, Integer.parseInt(request.getParameter("age")));
            st.setInt(6, Integer.parseInt(request.getParameter("mobile")));
            st.setString(7, request.getParameter("gender"));
            st.setString(8, request.getParameter("course"));

            st.executeUpdate();

            // now fetching student details with entered roll number -->

            st = con.prepareStatement("SELECT * FROM studentdetails where
roll = ?");
            st.setInt(1, Integer.parseInt(request.getParameter("roll")));
            ResultSet rs = st.executeQuery();

            if (rs.next()) {

                //Display values
                out.println("<h2 style=\"text-align:center\"> Student
Registered Successfully </h2>");
                out.println("<div style='border: 1px solid #000; width:40%;
height:40%; background-color:whiteSmoke'>");


```

```

        out.println("<table style='width:100%; height:100%'>");
        out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("name") + "</td></tr>");
            out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("roll") + "</td></tr>");
                out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("dob") + "</td></tr>");
                    out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("gender") + "</td></tr>");
                        out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("email") + "</td></tr>");
                            out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("mobileno") + "</td></tr>");
                                out.println("<tr><td style='font-size:1.5rem; text-align:center;'>" +
+ rs.getString("course") + "</td></tr>");
                                    out.println("</table>");
                                    out.println("</div>");

                }
            } else {
                out.println("<script> alert('User Already exist')</script>");
                out.println("<h2>Try Again</h2>");
            }
            out.println("<a style='margin-top:10px' href='./index.html'><button style='padding:10px'>Move to HomePage</button></a>");
            out.println("</body></html>");

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
}

```

Student Registration Form

Name:	Dante Adam
Date of Birth:	01 - 04 - 2023
Email:	danteberlin@gmail.com
Roll Number	123
Age:	40
Mobile:	1234567890
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Course:	MCA
<input type="button" value="Register"/> <input type="button" value="reset"/>	

Student Registered Successfully

Name:	Dante Adam
Roll:	123
DOB:	2023-04-01
Gender:	male
Email:	danteberlin@gmail.com
Mobile:	1234567890
Course:	MCA

[Move to HomePage](#)

7. Create a user registration page and login page. Also create a user homepage that should be redirected after successful login.

index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Login Form</title>
    </head>
    <body>
        <h2> User Login Form </h2>
        <form action="loginservlet" method="post">
            <table border="1" style ="width : 50%">
                <tr>
                    <td>Enter Email</td>
                    <td><input type="email" name="email" required/></td>
                </tr>
                <tr>
                    <td>Enter Password</td>
                    <td><input type="password" name="password" required/></td>
                </tr>
                <tr>
                    <td colspan="2" style="text-align: center;">
                        <input type="submit" value="Login"/>
                    </td>
                </tr>
            </table>
        </form>
        <a href="register.jsp">New User Click Here! </a>
    </body>
</html>
```

register.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>User Registration Form</title>
    </head>
    <body>
        <h2>User Registration Form </h2>
        <form action="registerservlet" method="post">
            <table border="1" style ="width : 50%">
                <tr>
                    <td>Enter Name</td>
                    <td><input type="text" name="name" required/></td>
                </tr>
                <tr>
                    <td>Enter Age</td>
                </tr>
            </table>
        </form>
    </body>
</html>
```

```

        <td><input type="number" name="age" required/></td>
    </tr>
    <tr>
        <td>Select Gender</td>
        <td>
            <input type="radio" name="gender" value="Male"
checked/>Male
            <input type="radio" name="gender" value="Female"
checked/>Female
        </td>
    </tr>
    <tr>
        <td>Enter City</td>
        <td><input type="text" name="city" required/></td>
    </tr>
    <tr>
        <td>Enter Email</td>
        <td><input type="email" name="email" required/></td>
    </tr>
    <tr>
        <td>Enter Password</td>
        <td><input type="password" name="password" required/></td>
    </tr>
    <tr>
        <td colspan="2">
            <input type="submit" value="Register"/>
        </td>
    </tr>
</table>
</form>
<a href="index.jsp">Click Here for Login! </a>
</body>
</html>
```

alldetails.jsp

```

<%@page import="java.sql.*"%>
<html>
    <head>
        <title> All User Details </title>
    </head>
    <body>
        <h2>All User Details</h2>
        <table border="1" style="width:100%;">
            <tr>
                <th>Sr. </th>
                <th>Name</th>
                <th>Age</th>
                <th>Gender</th>
                <th>City</th>
                <th>Email</th>
                <th>Options</th>
            </tr>
            <%
                try
                {
```

```

// Establishing connection with database and fetching all records

        Class.forName("com.mysql.cj.jdbc.Driver");
        String url =
"jdbc:mysql://localhost:3306/javaproject?zeroDateTimeBehavior=CONVERT_TO_NULL";
        String username = "root";
        String pass = "admin";
        Connection con = DriverManager.getConnection(url, username,
pass);
        Statement st = con.createStatement();
        ResultSet rs = st.executeQuery("select * from logindetails");
        int i =1;
        while(rs.next()){
            %>
            <tr>
            <td><%=i%></td>
            <td><%=rs.getString("name") %></td>
            <td><%=rs.getInt("age") %></td>
            <td><%=rs.getString("gender") %></td>
            <td><%=rs.getString("city") %></td>
            <td><%=rs.getString("email") %></td>
            <td> <a href="">Update</a> | <a href="">Delete</a>
            </td>
            <%
                i++;
            %}
        }
    }
    catch(Exception e){
        e.printStackTrace();
    }
    %>
</table>
</body>
</html>

```

loginServlet.java

```

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.*;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class loginServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            String email = request.getParameter("email");
            String password = request.getParameter("password");

```

```

        Class.forName("com.mysql.cj.jdbc.Driver");
        String url =
"jdbc:mysql://localhost:3306/javaproject?zeroDateTimeBehavior=CONVERT_TO_NULL";
        String id = "root";
        String pass = "admin";
        Connection con = DriverManager.getConnection(url, id, pass);

        //fetching details of user from database based on entered email and password

        PreparedStatement ps = con.prepareStatement("select * from
logindetails where email = ? and password = ?");
        ps.setString(1, email);
        ps.setString(2, password);
        ResultSet rs = ps.executeQuery();
        boolean status = rs.next();
        if (status) {
            String checkpass = rs.getString("password");

            // Checking if the password matches forward the request to welcome
page else redirect back to login page

            if (password.equals(checkpass)) {
                RequestDispatcher rd =
request.getRequestDispatcher("welcome");
                rd.forward(request, response);
            }
            } else {
                out.println("<html> <body> <script> alert('Wrong Credentials
Entered!')</script> <h2>Wrong Credentials Entered! <br> Try Again</h2> <a
style='margin-top:10px' href='index.jsp'><button style='padding:10px'>Go to Login
Page</button></a> </body> </html>");
                RequestDispatcher rd = request.getRequestDispatcher("index");
                rd.include(request, response);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

registerservlet.java

```
import java.sql.*;
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 registerservlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
```

welcome.java

```
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 welcome extends HttpServlet {

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

            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet welcome</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Login Successful!</h1><br>");
            out.println("<h2>Welcome " + request.getParameter("email") +
"</h1>");
            out.println("<h1>This is your Homepage</h1><br>");
            out.println("</body>");
            out.println("</html>");
        }
    }
}

```

User Login Form

Enter Email	yashshah@outlook.in
Enter Password	...
<input type="button" value="Login"/>	

[New User Click Here!](#)

Login Successful!

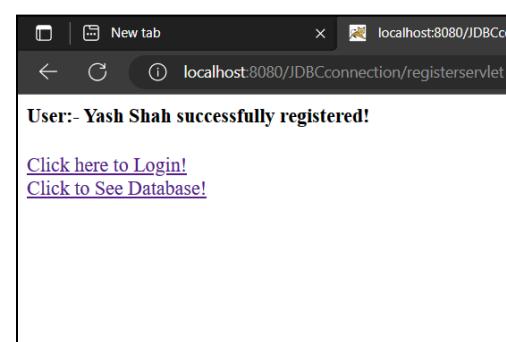
Welcome yashshah@outlook.in

This is your Homepage

User Registration Form

Enter Name	Yash Shah
Enter Age	23
Select Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Enter City	Patna
Enter Email	yashshah211997@gmail.co
Enter Password	...
<input type="button" value="Register"/>	<input type="radio"/>

[Click Here for Login!](#)



All User Details

Sr.	Name	Age	Gender	City	Email	Options
1	Aditya	30	Male	Delhi	aditya@gmail.com	Update Delete
2	Sunil	25	Male	RAIPUR	assa@gmail.com	Update Delete
3	Yash Shah	21	Male	Patna	yashshah@outlook.in	Update Delete
4	Yash Shah	23	Male	Patna	yashshah211997@gmail.com	Update Delete

8. Use sessions to track the user across the website.

index.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <title>Input Form</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>

<!-- Check if no session is previously stored --&gt;

&lt;%
    HttpSession s = request.getSession(false);
    String status = (String)s.getAttribute("status");
    if(status != null){
%&gt;
    &lt;script&gt;
        alert("&lt;%= status %&gt;");
    &lt;/script&gt;
&lt;% } %&gt;

&lt;h1&gt; User Registration From &lt;/h1&gt;

&lt;form action="RegisterServlet" method="POST"&gt;
    &lt;label for="name"&gt;Name:&lt;/label&gt;
    &lt;input type="text" id="name" name="name" required&gt;&lt;br&gt;

    &lt;label for="email"&gt;Email:&lt;/label&gt;
    &lt;input type="email" id="email" name="email" required&gt;&lt;br&gt;

    &lt;label for="password"&gt;Password:&lt;/label&gt;
    &lt;input type="password" id="password" name="password" required&gt;&lt;br&gt;

    &lt;label for="age"&gt;Age:&lt;/label&gt;
    &lt;input type="number" id="age" name="age" min="18" max="100"
required&gt;&lt;br&gt;

    &lt;label for="gender"&gt;Gender:&lt;/label&gt;
    &lt;input type="radio" id="male" name="gender" value="male" required&gt;
    &lt;label for="male"&gt;Male&lt;/label&gt;
    &lt;input type="radio" id="female" name="gender" value="female" required&gt;
    &lt;label for="female"&gt;Female&lt;/label&gt;
    &lt;input type="radio" id="other" name="gender" value="other" required&gt;
</pre>

```

```

<label for="other">Other</label><br>

<label for="course">Course:</label>
<select id="course" name="course" required>
    <option value="">--Select Course--</option>
    <option value="html">HTML</option>
    <option value="css">CSS</option>
    <option value="javascript">JavaScript</option>
</select><br>

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

```

UpdateProfile.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Update Profile</title>
        <link rel="stylesheet" type="text/css" href="style.css">
    </head>
    <body>
        <h1>Welcome <ins> <%= session.getAttribute("name") %></ins> to Profile
Update Page</h1>
        <h2> Profile Update Form </h2>
        <h3>Enter details to Update Profile </h3>

<form action="UpdateServlet" method="POST">

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

    <label for="age">Age:</label>
    <input type="number" id="age" name="age" min="18" max="100"
required><br>

    <label for="gender">Gender:</label>
    <input type="radio" id="male" name="gender" value="male" required>
    <label for="male">Male</label>
    <input type="radio" id="female" name="gender" value="female" required>
    <label for="female">Female</label>
    <input type="radio" id="other" name="gender" value="other" required>
    <label for="other">Other</label><br>

    <label for="course">Course:</label>
    <select id="course" name="course" required>
        <option value="">--Select Course--</option>
        <option value="html">HTML</option>
        <option value="css">CSS</option>
        <option value="javascript">JavaScript</option>
    </select><br>

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

```

```
    </form>
  </body>
</html>
```

homepage.jsp

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

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Profile Page</title>

    <style>
      body {
        display: flex;
        flex-direction: column;
        align-items: center;
        margin-top: 50px;
      }
      h1 {
        font-size: 24px;
        margin-bottom: 20px;
      }
      table {
        border-collapse: collapse;
        width: 50%;
        margin-bottom: 50px;
      }
      table td, table th {
        border: 1px solid black;
        padding: 10px;
        text-align: center;
      }
      table th {
        background-color: #4285F4;
        color: white;
      }
      input[type="submit"] {
        background-color: #4285F4;
        color: white;
        cursor: pointer;
        padding: 10px;
      }
      input[type="submit"]:hover {
        background-color: blue;
        color: white ;
        cursor: pointer;
      }
    </style>
  </head>
  <body>

    <!-- if session is stored then show page else redirect to login page -->
    <%
      String status = (String)session.getAttribute("status");
    %>
```

```

        if (status == null) {
            response.sendRedirect("index.jsp");
        }
        else {
    %>
        <script>
            alert("<%= status %>");
        </script>
    <% }
    %>
<h1>Welcome to your profile</h1>
<table>
    <tr>
        <th>Name</th>
        <th>Email</th>
        <th>Age</th>
        <th>Gender</th>
        <th>Selected Course</th>
    </tr>
    <tr>
        <td><%= session.getAttribute("name") %></td>
        <td><%= session.getAttribute("email") %></td>
        <td><%= session.getAttribute("age") %></td>
        <td><%= session.getAttribute("gender") %></td>
        <td><%= session.getAttribute("course") %></td>
    </tr>
</table>
<form action="UpdateProfile.jsp" method="POST">
    <input type="submit" value="Click to Update Profile">
</form> <br>
<form action="LogoutServlet" method="POST">
    <input type="submit" value="Click to Logout">
</form>
</body>
</html>

```

style.css

```

form {
    width: 50%;
    margin: 0 auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

label {
    display: inline-block;
    width: 100px;
    text-align: left;
    margin: 10px;
}

h1, h2, h3 {

```

```

margin-top: 50px;
text-align:center;
}

input[type="text"],
input[type="email"],
input[type="password"],
select {
    padding: 5px;
    border-radius: 3px;
    border: 1px solid #ccc;
    width: 60%;
    margin-bottom: 10px;
}

input[type="radio"] {
    margin-right: 5px;
}

input[type="submit"] {
    background-color: #4CAF50;
    color: white;
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
    margin-top: 20px;
}

input[type="submit"]:hover {
    background-color: #45a049;
}

```

RegisterServlet.java

```

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

public class RegisterServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try {
            String name = request.getParameter("name");
            String email = request.getParameter("email");
            String password = request.getParameter("password");
            int age = Integer.parseInt(request.getParameter("age"));
            String gender = request.getParameter("gender");
            String course = request.getParameter("course");

```

```
HttpSession session = request.getSession();

// save form-data as session attributes across pages of the website

session.setAttribute("name", name);
session.setAttribute("password", password);
session.setAttribute("email", email);
session.setAttribute("age", age);
session.setAttribute("gender", gender);
session.setAttribute("course", course);
session.setAttribute("status", "Login Successful");
response.sendRedirect("homepage.jsp");

} catch (Exception e) {
    e.printStackTrace();
}

}
```

UpdateServlet.java

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

public class UpdateServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try {
            String name = request.getParameter("name");
            int age = Integer.parseInt(request.getParameter("age"));
            String gender = request.getParameter("gender");
            String course = request.getParameter("course");

            HttpSession session = request.getSession();

            //Update user session details with the form-data

            session.setAttribute("name", name);
            session.setAttribute("age", age);
            session.setAttribute("gender", gender);
            session.setAttribute("course", course);

            //setting status attribute that the UPDATE is successful

            session.setAttribute("status", "Profile Update Successful");
            response.sendRedirect("homepage.jsp");
        } catch (Exception e) {
    }
}
```

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

LogoutServlet.java

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

public class LogoutServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try {
            HttpSession session = request.getSession(false);

            //delete all session attributes when the user logout

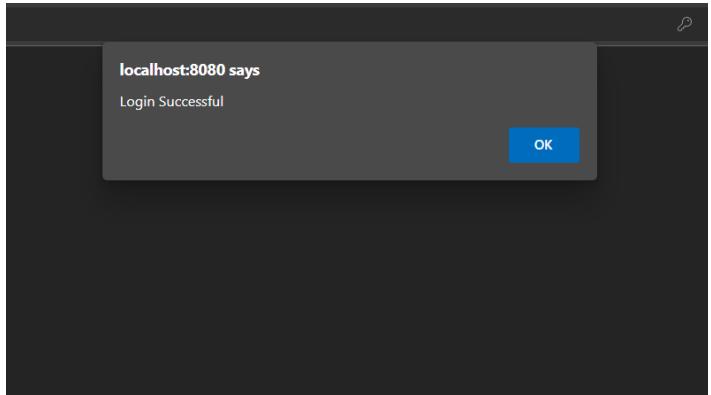
            if (session != null) {
                session.invalidate();
            }
            session = request.getSession();

            //setting status attribute that the LOGOUT is successful

            session.setAttribute("status", "Logout Successful");
            response.sendRedirect("index.jsp");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

User Registration Form

Name:	<input type="text" value="Yash Shah"/>
Email:	<input type="text" value="yashshah211997@gmail.com"/>
Password:	<input type="password" value="..."/>
Age:	<input type="text" value="23"/>
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Course:	<input type="text" value="CSS"/>
<input type="button" value="Submit"/>	



Welcome to your profile

Name	Email	Age	Gender	Selected Course
Yash Shah	yashshah211997@gmail.com	23	male	css

[Click to Update Profile](#)

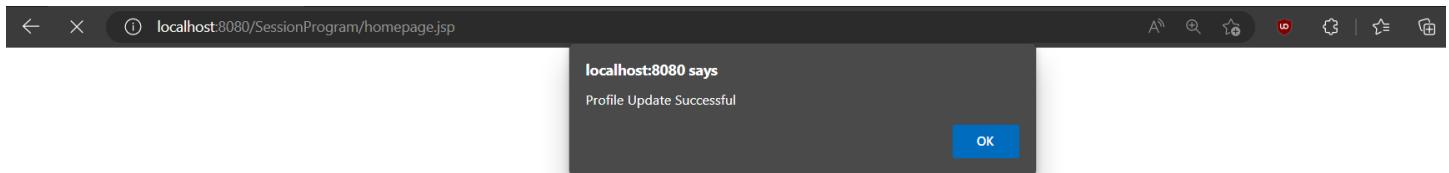
[Click to Logout](#)

Welcome Yash Shah to Profile Update Page

Profile Update Form

Enter details to Update Profile

Name:	<input type="text" value="Yash Shah"/>
Age:	<input type="text" value="25"/>
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Course:	<input type="text" value="JavaScript"/>
<input type="button" value="Update"/>	



Welcome to your profile

Name	Email	Age	Gender	Selected Course
Yash Shah	yashshah211997@gmail.com	25	male	javascript

[Click to Update Profile](#)

[Click to Logout](#)

localhost:8080 says

Logout Successful

[OK](#)

9. Store some cookie variables on user's computer and retrieve them in the next Http request.

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cookies Demo</title>
    <style>
      input[type=text], select {
        width: 100%;
        padding: 12px 20px;
        margin: 8px 0;
        display: inline-block;
        border: 1px solid #ccc;
        border-radius: 4px;
        box-sizing: border-box;
      }

      input[type=submit] {
        width: 100%;
        background-color: #4CAF50;
        color: white;
        padding: 14px 20px;
        margin: 8px 0;
        border: none;
        border-radius: 4px;
        cursor: pointer;
      }

      input[type=submit]:hover {
        background-color: #45a049;
      }

      .container {
        border-radius: 5px;
        background-color: #f2f2f2;
        padding: 20px;
        width: 70%;
        position: fixed;
        top: 50%;
        left: 50%;
        transform: translate(-50%, -50%);
      }
    </style>
  </head>
  <body>
    <h1 style="text-align:center;"> Cookie Demonstration Program</h1>
    <div class="container">
      <form action="servlet1" method="post">
        <label for="fname">First Name</label>
        <input type="text" id="fname" name="firstname" placeholder="Enter
your First Name">
```

```

        <label for="lname">Last Name</label>
        <input type="text" id="lname" name="lastname" placeholder="Your
Last Name..">

        <label for="country">Select your Country</label>
        <select id="country" name="country">
            <option value="India">India</option>
            <option value="Canada">Canada</option>
            <option value="USA">USA</option>
        </select>
        <input type="submit" value="Submit">

    </form>
</div>

</body>
</html>

```

servlet1.java

```

package cookies;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class servlet1 extends HttpServlet {

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

            // fetching form-data sent as request parameter

            String fname = request.getParameter("firstname");
            String lname = request.getParameter("lastname");
            String country = request.getParameter("country");

            out.println("<center>");
            out.println("<div style=\"color:blue;position: fixed;\n"
                    + "    top: 50%;\n"
                    + "    left: 50%;\n"
                    + "    transform: translate(-50%, -50%);\">");
            out.println("<h1>Welcome: " + fname + " " + lname + "</h1>");

            // Creating Cookies of User details

            Cookie ck1 = new Cookie("fname", fname);
            response.addCookie(ck1);
            Cookie ck2 = new Cookie("lname", lname);

```

```

        response.addCookie(ck2);
        Cookie ck3 = new Cookie("country", country);
        response.addCookie(ck3);

        out.print("<form action='servlet2'>");
        out.print("<input type='submit' value='Click to Fetch your Cookie
Values'>");
        out.print("</form>");
        out.println("</div>");
        out.println("</center>");
        out.close();
    } catch (Exception e) {
        System.out.println(e);
    }
}
}
}

```

servlet2.java

```

package cookies;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.Cookie;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class servlet2 extends HttpServlet {

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

            //Fetching already stored Cookie details and display it

            Cookie ck[] = request.getCookies();

            out.println("<center>");
            out.println("<div style=\"color:blue;position: fixed;\n"
                    + "    top: 50%;\n"
                    + "    left: 50%;\n"
                    + "    transform: translate(-50%, -50%);\">");
            out.print("<h1>Fetched values from saved Cookies</h1>");
            out.print("<h3>Hello User: " + ck[0].getValue() + " " +
ck[1].getValue() + "</h3>");
            out.print("<h3>Your country is: " + ck[2].getValue() + "</h3>");
            out.println("</div>");
            out.println("</center>");
            out.close();

        } catch (Exception e) {
            System.out.println(e);
        }
    }
}

```

```
        }  
    }  
}
```

Cookie Demonstration Program

First Name

Last Name

Select your Country

Welcome: Yash Shah

[Click to Fetch your Cookie Values](#)

Fetched values from saved Cookies

Hello User: Yash Shah

Your country is: India

10. Create a small single page application (SPA) using AngularJS.

CLI Commands to create an angular app and it's components

```
(c) Microsoft Corporation. All rights reserved.

C:\Users\mratu\OneDrive\Desktop>ng new angular-app
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE angular-app/angular.json (2725 bytes)
CREATE angular-app/package.json (1842 bytes)
CREATE angular-app/README.md (1864 bytes)
CREATE angular-app/tsconfig.json (981 bytes)
CREATE angular-app/editorconfig (274 bytes)
CREATE angular-app/.gitignore (548 bytes)
CREATE angular-app/tsconfig.app.json (263 bytes)
CREATE angular-app/tsconfig.spec.json (273 bytes)
CREATE angular-app/.vscode/extensions.json (138 bytes)
CREATE angular-app/.vscode/launch.json (479 bytes)
CREATE angular-app/.vscode/tasks.json (938 bytes)
CREATE angular-app/src/favicon.ico (94 bytes)
CREATE angular-app/src/index.html (296 bytes)
CREATE angular-app/src/main.ts (214 bytes)
CREATE angular-app/src/styles.css (88 bytes)
CREATE angular-app/src/assets/.gitkeep (0 bytes)
CREATE angular-app/src/app/app-routing.module.ts (245 bytes)
CREATE angular-app/src/app/app.module.ts (393 bytes)
CREATE angular-app/src/app/app.component.html (23115 bytes)
CREATE angular-app/src/app/app.component.spec.ts (1088 bytes)
CREATE angular-app/src/app/app.component.ts (215 bytes)
CREATE angular-app/src/app/app.component.css (0 bytes)

. Packages installed successfully.

warning: in the working copy of 'editorconfig', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'gitignore', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.vscode/extensions.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.vscode/launch.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.vscode/tasks.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'angular.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/app/app-routing.module.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/app/app.module.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/app/app.component.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/app/app.component.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/app/app.component.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/index.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/main.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/styles.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.app.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.spec.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.spec.json', LF will be replaced by CRLF the next time Git touches it

Successfully initialized git.

C:\Users\mratu\OneDrive\Desktop>
```



```
C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component first
? Would you like to share pseudonymous usage data about this project with the Angular Team at Google under Google's Privacy Policy at https://policies.google.com/privacy. For more details and how to change this setting, see https://angular.io/analytics. Yes
Thank you for sharing pseudonymous usage data. Should you change your mind, the following command will disable this feature entirely:
    ng analytics disable

Global setting: enabled
Local setting: enabled
Effective status: enabled
CREATE src/app/first/first.component.html (20 bytes)
CREATE src/app/first/first.component.spec.ts (592 bytes)
CREATE src/app/first/first.component.ts (198 bytes)
CREATE src/app/first/first.component.css (0 bytes)
UPDATE src/app/app.module.ts (471 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component second
CREATE src/app/second/second.component.html (21 bytes)
CREATE src/app/second/second.component.spec.ts (599 bytes)
CREATE src/app/second/second.component.ts (282 bytes)
CREATE src/app/second/second.component.css (0 bytes)
UPDATE src/app/app.module.ts (553 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component temp-component
CREATE src/app/temp-component/temp-component.component.html (20 bytes)
CREATE src/app/temp-component/temp-component.component.spec.ts (649 bytes)
CREATE src/app/temp-component/temp-component.component.ts (233 bytes)
CREATE src/app/temp-component/temp-component.component.css (0 bytes)
UPDATE src/app/app.module.ts (665 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component page-not-found
CREATE src/app/page-not-found/page-not-found.component.html (29 bytes)
CREATE src/app/page-not-found/page-not-found.component.spec.ts (643 bytes)
CREATE src/app/page-not-found/page-not-found.component.ts (232 bytes)
CREATE src/app/page-not-found/page-not-found.component.css (0 bytes)
UPDATE src/app/app.module.ts (775 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component page-content
CREATE src/app/page-content/page-content.component.html (27 bytes)
CREATE src/app/page-content/page-content.component.spec.ts (635 bytes)
CREATE src/app/page-content/page-content.component.ts (225 bytes)
CREATE src/app/page-content/page-content.component.css (0 bytes)
UPDATE src/app/app.module.ts (879 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>ng generate component page-content/sub-component
CREATE src/app/page-content/sub-component/sub-component.component.html (28 bytes)
CREATE src/app/page-content/sub-component/sub-component.component.spec.ts (642 bytes)
CREATE src/app/page-content/sub-component/sub-component.component.ts (229 bytes)
CREATE src/app/page-content/sub-component/sub-component.component.css (0 bytes)
UPDATE src/app/app.module.ts (1000 bytes)

C:\Users\mratu\OneDrive\Desktop\angular-app>
```

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Angular App</title>
    <base href="/" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="icon" type="image/x-icon" href="favicon.ico" />
  </head>
  <body>
    <app-root></app-root>
  </body>
</html>
```

app.component.html

```
<div class="main">
  <h1>Angular Router App</h1>
  <!-- This nav gives you links to click, which tells the router which route to use
  (defined in the routes constant in AppRoutingModule) -->
  <div class="flex col-2">
    <nav>
      <ul>
        <li>
          <a
            routerLink="/first-component"
            routerLinkActive="active">
```

```

        ariaCurrentWhenActive="page"
        >First Component</a>
    >
</li>
<li>
    <a
        routerLink="/second-component"
        routerLinkActive="active"
        ariaCurrentWhenActive="page"
        >Second Component</a>
    >
</li>
<li>
    <a
        routerLink="/page-content"
        routerLinkActive="active"
        ariaCurrentWhenActive="page"
        >Page Content</a>
    >
</li>
</ul>
</nav>
<!-- The routed views render in the &lt;router-outlet&gt;--&gt;
&lt;router-outlet&gt;&lt;/router-outlet&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;app-temp-component&gt;&lt;/app-temp-component&gt;</pre>

```

app-routing.module.ts

```

import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { FirstComponent } from './first/first.component';
import { SecondComponent } from './second/second.component';
import { PageContentComponent } from './page-content/page-content.component';
import { PageNotFoundComponent } from './page-not-found/page-not-found.component';
import { TempComponentComponent } from './temp-component/temp-component.component';

const routes: Routes = [
  { path: 'first-component', component: FirstComponent },
  { path: 'second-component', component: SecondComponent },
  { path: 'page-content', component: PageContentComponent },
  { path: 'temp-component-2', component: TempComponentComponent },
  { path: '', redirectTo: '/first-component', pathMatch: 'full' }, // redirect to `first-component`
  { path: '**', component: PageNotFoundComponent }, // Wildcard route for a 404 page
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule],
})
export class AppRoutingModule {}
```

first.component.html

```

<h1>{{ title }}</h1>
<!-- Write some dummy content to the page --&gt;</pre>

```

```
<p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed euismod, nisl
    euismod aliquam tincidunt, nisl nisl aliquam lorem, eget aliquam nisl nisl
    eget lorem. Sed euismod, nisl euismod aliquam tincidunt, nisl nisl aliquam
    lorem, eget aliquam nisl nisl eget lorem. Sed euismod, nisl euismod aliquam
    tincidunt, nisl nisl aliquam lorem, eget aliquam nisl nisl eget lorem. Sed
    euismod, nisl euismod aliquam tincidunt, nisl nisl aliquam lorem, eget aliquam
    nisl nisl eget lorem. Sed euismod, nisl euismod aliquam tincidunt, nisl nisl
    aliquam lorem, eget aliquam nisl nisl eget lorem. Sed euismod, nisl euismod
    aliquam tincidunt, nisl nisl aliquam lorem, eget aliquam nisl nisl eget lorem.
    Sed euismod, nisl euismod aliquam tincidunt, nisl nisl aliquam lorem, eget
    aliquam nisl nisl eget lorem. Sed euismod, nisl euismod aliquam tincidunt,
    nisl nisl aliquam lorem, eget aliquam nisl nisl eget lorem. Sed euismod, nisl
    euismod aliquam tincidunt, nisl nisl aliquam lorem, eget aliquam nisl nisl
    eget lorem. Sed euismod, nisl euismod aliquam tincidunt, nisl nisl aliquam
    lorem, eget aliquam nisl nisl eget lorem. Sed euismod, nisl euismod aliquam
    tincidunt, nisl nisl aliquam lorem, eget aliquam nisl nisl eget lorem. Sed
    euismod, nisl euismod aliquam tincidunt, nisl nisl aliquam lorem, eget aliquam
    nisl nisl eget lorem. Sed euismod, nisl euismod aliquam tincidunt, nisl nisl
    aliquam lorem, eget aliquam nisl nisl eget lorem. Sed euismod, nisl euismod
    aliquam t
</p>
```

```
<!-- Add a button to the page -->
<button (click)="goToPage2()">Alert btn</button>
```

first.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-first',
  templateUrl: './first.component.html',
  styleUrls: ['./first.component.css'],
})
export class FirstComponent {
  title: string = 'First Component Render';
  goToPage2 = function () {
    // Navigate to the /second-component route
    alert('"I am the first component"');
  };
}
```

page-content.component.html

```
<div class="flex flex-column app-comp">
  <h1>
    {{ title }}
  </h1>
  <app-sub-page-component></app-sub-page-component>
</div>
```

page-content.component.css

```
.app-comp {
  max-width: 75rem;
  width: 86.67%;
  margin: 0 auto 5rem;
  gap: 2rem;
```

```
}
```

```
app-page-content {  
  gap: 1rem;  
}
```

page-content.component.ts

```
import { Component } from '@angular/core';  
  
@Component({  
  selector: 'app-page-content',  
  templateUrl: './page-content.component.html',  
  styleUrls: ['./page-content.component.css'],  
})  
export class PageContentComponent {  
  title = 'Page Content';  
}
```

sub-page-component.html

```
<p>sub-page-component works!</p>  
<p>Angular is a framework for building client applications.</p>  
<p>This is a simple example of a component.</p>  
<p>The component is the most basic building block of an Angular application.</p>  
<p>  
  A component controls a portion of the screen – a view – through its associated  
  template.  
</p>  
<p>A component is made up of three things:</p>  
<ul>  
  <li>A component class that handles data and functionality.</li>  
  <li>An HTML template that determines the UI.</li>  
  <li>Component-specific styles that define the look and feel.</li>  
</ul>  
<p>Here's an example of a component's file structure:</p>  
<pre>  
  app  
    ├── app.component.css  
    ├── app.component.html  
    ├── app.component.spec.ts  
    └── app.component.ts  
</pre>  
<p>  
  The CLI generated three files for the AppComponent and a fourth file for the  
  application routing.  
</p>  
<p>The files are the following:</p>  
<ul>  
  <li>  
    <code>app.component.ts</code>— the component class code, written in  
    TypeScript.  
</li>  
  <li>  
    <code>app.component.html</code>— the component template, written in HTML.  
</li>  
  <li><code>app.component.css</code>— the component's private CSS styles.</li>  
<li>  
    <code>app.component.spec.ts</code>— the component test file, written in  
    TypeScript.  
</li>
```

```
Jasmine.
</li>
</ul>
<p>
  The CLI generated a <code>root</code> component that is named
  <code>AppComponent</code> and displays the application title.
</p>
<p>
  The component class is where you handle data and functionality for the view.
</p>
<p>
  The template is where you define how the view is presented to the user. The
  template and styles are optional if all you're doing is generating the
  application shell.
</p>
<p>
  The CLI generated the root <code>AppComponent</code> in the
  <code>src/app</code> folder. You can find the <code>app.component.ts</code>
  file there.
</p>
<p>
  The CLI also generated <code>app.module.ts</code>, which is the root module of
  the application. It tells Angular how to assemble the application. Initially
  <code>app.module.ts</code> declares only the <code>AppComponent</code>. As you
  add more components to the application, they must be declared here.
</p>
<p>
  The <code>app.component.ts</code> file contains the following import
  statements that import the <code>Component</code> symbol from the Angular core
  library and the <code>environment</code> object from the
  <code>environments</code> folder.
</p>
<pre style="color: red">
  import &lcub; Component &rcub; from '@angular/core';
  import &lcub; environment &rcub; from '../environments/environment';
</pre>
<p>
  The <code>@Component</code> decorator identifies <code>AppComponent</code> as
  an Angular component. The decorator provides metadata about the component,
  including its templates, styles, and a selector.
</p>
<p>
  The CLI generated the component's selector, <code>app-root</code>, and the
  <code>index.html</code> file that references that selector.
</p>
<p>
  The <code>templateUrl</code> and <code>styleUrls</code> properties tell
  Angular where to find the component's template and style files.
</p>
<p>
  The component definition also exports the class, <code>AppComponent</code>, so
  that you can use it in other components' templates.
</p>
<p>
  The <code>AppComponent</code> class contains a <code>title</code> property.
  The value of this property is displayed in the application by using the
  interpolation syntax in the <code>index.html</code> template.
</p>
```

page-not-found.html

```
<p>page-not-found works!</p>
```

second.component.html

```
<p>second works!</p>
```

temp-component.component.html

```
<H1>Hello temp Component</H1>
```

CLI to run Angular App

```
PS C:\Users\mratu\OneDrive\Desktop\angular-app-for-college-ppt> ng serve --open
✓ Browser application bundle generation complete.

Initial Chunk Files | Names      | Raw Size
vendor.js           | vendor     | 2.04 MB
polyfills.js        | polyfills  | 314.31 kB
styles.css, styles.js | styles    | 211.55 kB
main.js             | main       | 40.32 kB
runtime.js          | runtime    | 6.55 kB

| Initial Total | 2.60 MB

Build at: 2023-04-01T08:13:06.999Z - Hash: 25b2ed8755bba8b3 - Time: 5102ms
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.
✓ Browser application bundle generation complete.

5 unchanged chunks

Build at: 2023-04-01T08:14:18.304Z - Hash: 25b2ed8755bba8b3 - Time: 377ms

✓ Compiled successfully.
✓ Browser application bundle generation complete.

Initial Chunk Files | Names      | Raw Size
main.js             | main       | 40.32 kB
runtime.js          | runtime    | 6.55 kB

3 unchanged chunks

Build at: 2023-04-01T08:18:35.653Z - Hash: 0a3e90b1255649c6 - Time: 966ms

✓ Compiled successfully.
✓ Browser application bundle generation complete.

Initial Chunk Files | Names      | Raw Size
main.js             | main       | 40.32 kB
runtime.js          | runtime    | 6.55 kB

3 unchanged chunks

Build at: 2023-04-01T08:25:11.951Z - Hash: 11046be5631a7f75 - Time: 623ms

✓ Compiled successfully.
```

Output

Angular Router App

- First Component
- Second Component
- Page Content

Page Content

sub-page-component works!

Angular is a framework for building client applications.

This is a simple example of a component.

The component is the basic building block of an Angular application.

A component controls a portion of the screen – a view – through its associated template.

A component is made up of three things:

- A component class that handles data and functionality.
- An HTML template that determines the UI.
- Component-specific styles that define the look and feel.

Here's an example of a component's file structure:

```
app
  └── app.component.css
  └── app.component.html
  └── app.component.spec.ts
  └── app.component.ts
```

The CLI generated three files for the AppComponent, and a fourth file for the application routing.

The files are the following:

- `app.component.ts`—the component class code, written in TypeScript.
- `app.component.html`—the component template, written in HTML.
- `app.component.css`—the component's private CSS styles.

Angular Router App

- First Component
- Second Component
- Page Content

Hello temp Component

second works!

Angular Router App

- First Component
- Second Component
- Page Content

First Component Render

localhost:4200 says
"I am the first component"

OK

First Component Render

First Component Render

Alert box

Alert box

Hello temp Component

Alert box