MIT Vishwaprayag University

School of Computing

FULL STACK DEVELOPMENT ASSIGNMENTS

(HTML, CSS, JavaScript, React)

Submitted By:

Name: Zaid M Shaikh
Roll No.: SCFP125065
Course: MCA (Semester–1)

Submitted To:
Instructor: Prof. Anand Shimpi

Academic Year: 2025–2026

TABLE OF CONTENTS

Sr. No.	Assignment / Program Title	Page No.
Assignment 1		
1	Basic HTML Tags	
2	Portfolio Program	
3	Student Survey Form	
Assignment 2		
4	Student Registration Form	
5	HTML Selector Elements (Select, Input Types)	
Assignment 3		
6	CSS Selectors Demonstration	
7	Menu Driver Application (HTML + CSS + JS)	
Assignment 4		
8	Student Database Using Integrated CSS	
Assignment 5		
9	CSS Overflow and Positioning	
Assignment 6		
10	CSS Box Model Demonstration	
Assignment 7		
11	Pseudo Class Hover + Bootstrap Buttons	
Assignment 8		
12	JavaScript Events (Click, Mouse, Keyboard)	
Assignment 9		
13	JavaScript Strings Demonstration	
Assignment 10		
14	JavaScript Conditional Statements	
Assignment 11		
15	JavaScript Arrays (push, pop, map, filter)	
Assignment 12		
16	JavaScript Functions	
Assignment 13		
17	JavaScript Objects Demonstration	
Assignment 14		
18	JavaScript DOM Selectors Program	
Assignment 15		
19	Advanced DOM Events + querySelector All()	
Assignment 16		
20	ReactJS Basic Components + Props	

Assignment 01 – Basic Programs

Aim: Basic HTML Tags.

Source Code

```
<html>
<head>
<title>WELCOME TO MY WEBSITE</title>
<body>
    <h1>Welcome Zaid</h1>
    <h1>This is heading tag<h1><br>
    <b>Bold tag</b><br>
    <strong>strong tag</strong><br>
    <i>Italic text</i><br>
    <em>Emphasized tag</em><br>
    <u>Underlined tag</u><br>
    <img src="image.jpg" alt="My Image"><br>
    <P>This is paragraph</P><br>
</body>
</head>
</html>
```

Output

Welcome Zaid

This is heading tag

Bold tag
strong tag
Italic text
Emphasized tag
Underlined tag
My Image

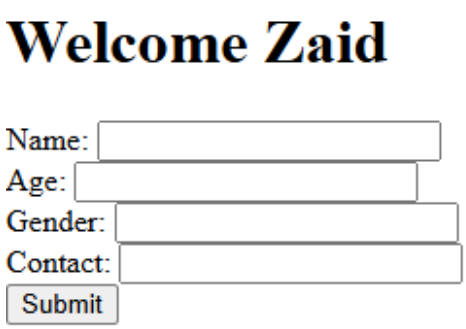
This is paragraph

Aim: Portfolio.

Source Code

```
<!DOCTYPE html>
<html>
<head>
<title>Portfolio for Students </title>
</head>
<body>
    <h1>Welcome Zaid</h1>
    Name: <input type="text" name="name"><br>
    Age: <input type="text" name="Age"><br>
    Gender: <input type="text" name="Gender"><br>
    Contact: <input type="text" name="contact"><br>
    <input type="submit" value-submit>
</body>
</html>
```

Output



Aim

Design an HTML page to create a simple **Student Survey Form**.

Source Code

```
<!DOCTYPE html>
<html>
    <head>
        <img src="download.jpg"width="400"height="400">
        <body>
            <h1>Welcome Zaid</h1>
            <h1>Student Survey</h1>
            Name:<input type="text"name="name"><br>
            Age:<input type="text"name="age"><br>
            Gender:<input type="text"name="Gender"><br>
            Contact:<input type="text"name="contact"><br>

            Favourite Subject:
            <select name="subjects"><br>
            <option value="node js">node js</option>
            <option value="react js">react js</option>
            <option value="C++">C++</option><br>
            </select><br>

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

Output

Welcome Zaid

Student Survey

Name:
Age:
Gender:
Contact:
Favourite Subject: node js
submit

Assignment 02 – Forms with all Different Attribute.

Aim: Create an HTML page for a **Student Registration Form**

Source Code

```
<!DOCTYPE html>
<html>

<head>
    <title>Student Registration</title>

</head>

<body>

    <h1>Student Registration</h1>

    <h1>Welcome Zaid</h1>
    <div class="container">
        <form>
            <label>name</label>
            <input type="text" id="name"><br>
            <label>email id</label>
            <input type="email" id="email"><br>
            <label>mobile</label>
            <input type="text" id="mobile"><br><br>
            <b>contact </b><br>
            <input type="radio" value="email">
            <label>email</label><br>
            <input type="radio" value="phone">
            <label>phone</label><br><br>
            <b>Do you want to subscribe newsletter?</b><br>
            <input type="checkbox" value="yes">
            <label>yes</label>
            <input type="checkbox" value="no">
            <label>no</label><br><br>
            <input type="submit" value="Submit">

        </form>
    </div>

</body>

</html>
```

Output


Student Registration
Welcome Zaid
name
email id
mobile
contact
email
phone
Do you want to subscribe newsletter?
yes
no
Submit

Assignment 03 – HTML Select Element

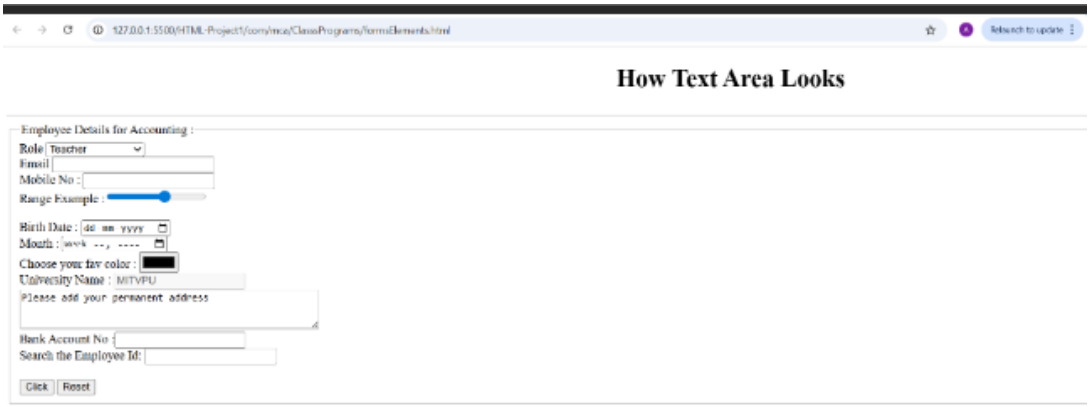
Aim: Write a program for HTML Selector Elements.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>This is example of textarea looks like</h1>
    <form>
    <fieldset>
         <h1>Welcome Zaid</h1>
        <input type="search" value="search" id="search1"><input type="button"
           name="button" value="Search" ><br><br>
        Role
        <select name="roles" >
            <option value="Deisnger">Designer</option>
            <option value="Tester">Tester</option>
            <option value="DevOps">DevOps</option>
            <option value="Engineer">Engineer</option>
        </select><br>
        <legend>Employee Details for accounting</legend>

        <textarea rows="3" cols="50">Please add your permnanat address</textarea
           ><br><br>
        Bank Account No : <input type="password" name="baccount"><br><br>
        Email
        <input type="email" name="email"><br><br>
        Mobile No
        <input type="number" name="number"><br><br>
        Date
        <input type="date" name="date"><br><br>
        Month
        <input type="month" name="month"><br><br>
        WeekDays
        <input type="week" name="week"><br><br>
        Range
        <input type="range" min="5" max="10"><br><br>
        Color
        <input type="color" name="color"><br><br>
        Class
        <input type="text" name="en" value="MCA-I 2025" disabled><br><br>
        <input type="button" onclick="alert('your Details is submitted
           successfully')" value="click">
        <input type="reset" >
```

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

Output



Assignment 04 – CSS Selectors & Menu Driver Application

Aim : Demonstrate different **CSS selectors**.

source code

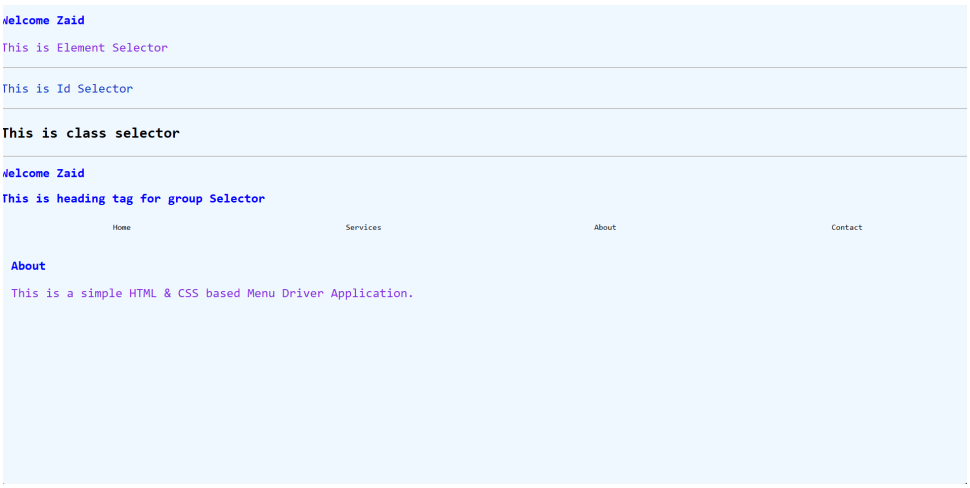
```
Assignment 04
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Selector Example</title>
<style>
    p {
        font-size: 25px;
        color: blueviolet;
    }
    #heading {
        font-family: Consolas;
        color: rgb(19, 61, 201);
        font-size: 25px;
    }
    .section {
        font-family: "Times New Roman", Times, serif;
        font-size: 25px;
    }
    * {
        background-color: aliceblue;
        font-family: Consolas;
    }
    h1, h2 {
        color: blue;
        font-size: 25px;
    }
</style>
</head>
<body>
<p>This is Element Selector</p><hr>
<p id="heading">This is Id Selector</p><hr>
<div class="section">
    <h3>This is class selector</h3><hr>
</div>
<h1>Welcome Zaid </h1>
<h2>This is heading tag for group Selector</h2>
</body>
</html>
```

Menu Driver Application (HTML + CSS + JS)

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Menu Driver Application</title>
<style>
    body {
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
        margin: 0;
        padding: 0;
    }
    nav {
        background-color: #333;
    }
    nav ul {
        list-style-type: none;
        margin: 0;
        padding: 0;
        display: flex;
    }
    nav ul li {
        flex: 1;
    }
    nav ul li a {
        display: block;
        text-align: center;
        padding: 12px;
        color: white;
        text-decoration: none;
    }
    nav ul li a:hover {
        background: #ff9800;
    }
    section {
        display: none;
        padding: 20px;
    }
    .active {
        display: block;
    }
</style>
</head>
<body>
<nav>
    <ul>
        <li><a href="#" onclick="Showsection('home')">Home</a></li>
        <li><a href="#" onclick="Showsection('services')">Services</a></li>
        <li><a href="#" onclick="Showsection('about')">About</a></li>
```

```
        <li><a href="#" onclick="Showsection('contact')">Contact</a></li>
    </ul>
</nav>
<section id="home" class="active">
    <h2>Welcome Zaid!</h2>
    <p>Select any option from the menu above.</p>
</section>

<section id="services">
    <h2>Services</h2>
    <p>Perform simple operations here.</p>
</section>
<section id="about">
    <h2>About</h2>
    <p>This is a simple HTML & CSS based Menu Driver Application.</p>
</section>
<section id="contact">
    <h2>Contact</h2>
    <p>Email: support@example.com</p>
</section>
<script>
    function Showsection(id) {
        let section = document.querySelectorAll("section");
        section.forEach(sec => sec.classList.remove("active"));
        document.getElementById(id).classList.add("active");
    }
</script>
</body>
</html>
```

Output



Assignment 05 – Integrated CSS Application for Student Database

Aim: Create a **student database web page** using HTML tables and integrated CSS .

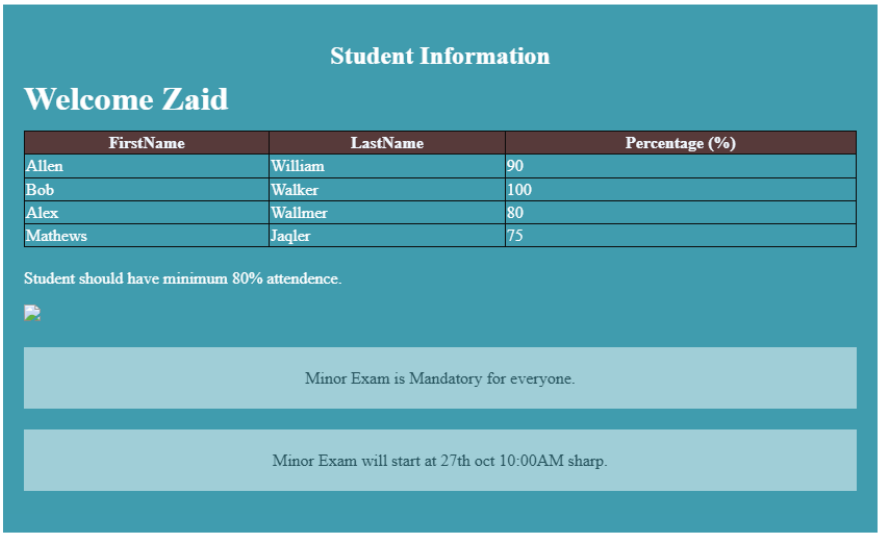
Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Integrated CSS Application for Student Database</title>
<style>
    body{
        font-family: 'Times New Roman', Times, serif;
        line-height: 120%;
    }
    .container{
        width: 90%;
        max-width: 800px;
        min-width: 400px;
        margin: 20px auto;
        padding: 20px;
        background: rgb(54, 151, 170);
        color: white;
        opacity: 0.95;
    }
    table{
        border-collapse: collapse;
        width: 100%;
        margin-bottom: 20px;
    }
    th, td{
        border: solid black 1px;
    }
    th{
        background-color: rgb(78, 47, 47);
        color: aliceblue;
    }
    tr:hover{
        background-color: #efdd88;
        opacity: 0.7;
    }
    .demo-img:hover{
        opacity: 0.5;
    }
    .custom-textSm{
        background-color: rgb(255,255,255);
        margin: 20px auto;
```

```
        padding: 20px;
        opacity: 0.5;
        color: rgb(0,0,0);
        font-style: bold;
        text-align: center;
    }
    .custom-textms{
        background-color: rgb(255,255,255);
        margin: 20px auto;
        padding: 20px;
        opacity: 0.5;
        color: rgb(0,0,0);
        font-style: bold;
        text-align: center;
    }
</style>
</head>
<body>
<div class="container">
    <h2 align="center">Student Information</h2>
    <table>
        <tr>
            <th>FirstName</th>
            <th>LastName</th>
            <th>Percentage (%)</th>
        </tr>
        <tr>
            <td>Allen</td>
            <td>William</td>
            <td>90</td>
        </tr>
        <tr>
            <td>Bob</td>
            <td>Walker</td>
            <td>100</td>
        </tr>
        <tr>
            <td>Alex</td>
            <td>Wallmer</td>
            <td>80</td>
        </tr>
        <tr>
            <td>Mathews</td>
            <td>Jaqler</td>
            <td>75</td>
        </tr>
    </table>
    <div class="image-box">
        <p>Student should have minimum 80% attendence.</p>
        <img src="./photo1.jpeg">
    </div>
```

```
    <div class="custom-textSm">
        Minor Exam is Mandatory for everyone.
    </div>
    <div class="custom-textms">
        Minor Exam will start at 27th oct 10:00AM sharp.
    </div>
</div>
</body>
</html>
```

Output



Assignment 06 – CSS Overflow and Positioning

Aim: Demonstrate different **overflow** properties and CSS **positioning** types (static, relative, absolute, sticky, fixed).

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>CSS Overflow property</title>
<style>
.box1{
    width: 200px;
    height: 100px;
    border: 2px dashed red;
    margin: 20px;
    padding: 10px;
    background-color: darkseagreen;
}
/* Overflow Types */
.overflow-visible{
    overflow: visible;
}
.overflow-hidden{
    overflow: hidden;
}
.overflow-scroll{
    overflow: scroll;
}
.overflow-auto{
    overflow: auto;
}
/* Position Container */
.container{
    position: relative;
    width: 400px;
    height: 300px;
    border: 2px dotted green;
    margin: 20px;
    background-color: #e5afe5;
}
/* Inner Boxes */
.box{
    width: 100px;
    height: 50px;
    padding: 10px;
    color: blue;
    font-weight: bold;
```

```
    text-align: center;
}
/* Positions */
.static-box{
    position: static;
    background-color: aqua;
}
.relative-box{
    position: relative;
    top: 0px;
    left: 0px;
    background: green;
}
.absolute-box{
    position: absolute;
    top: 50px;
    right: 20px;
    background: red;
}
.sticky-box{
    position: sticky;
    top: 0;
    background-color: chartreuse;
}
.fixed-box{
    position: fixed;
    bottom: 10px;
    right: 10px;
    background: purple;
}
</style>
</head>
<body>
<h2>Overflow Property</h2>
<div class="box1 overflow-visible">
    <b>Overflow: visible</b><br>
    ITIL (Information Technology Infrastructure Library) is a framework of best
        practices for managing IT services.
</div>
<div class="box1 overflow-hidden">
    <b>Overflow: hidden</b><br>
    ITIL (Information Technology Infrastructure Library) is a framework of best
        practices for managing IT services.
</div>
<div class="box1 overflow-scroll">
    <b>Overflow: scroll</b><br>
    ITIL (Information Technology Infrastructure Library) is a framework of best
        practices for managing IT services.
</div>
<div class="box1 overflow-auto">
    <b>Overflow: auto</b><br>
```

```
    ITIL (Information Technology Infrastructure Library) is a framework of best
        practices for managing IT services.
</div>
<br><br>
<div class="container">
    <div class="box static-box">Static</div>
    <div class="box relative-box">Relative</div>
    <div class="box absolute-box">Absolute</div>
    <div class="box sticky-box">Sticky</div>
</div>
<p style="height: 800px;">
    This is the property of positioning in CSS that affected each and every
        property we used
</p>
<div class="box fixed-box">
    <b>Overflow: fixed</b><br>
    ITIL (Information Technology Infrastructure Library).
</div>
</body>
</html>
```

Output

Assignment 07 – CSS Box Model Demonstration

Aim: Demonstrate the **CSS box model** using content area, padding, border, and margin with a student information example.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<style>
.box{
    width: 250px;
    height: 100px;
    padding: 20px;
    border: 5px solid black;
    margin: 20px;
    background-color: rgb(58, 227, 211);
    color: rgb(159, 8, 59);
    font-weight: bold;
}
.info{
    font-family: Arial, Helvetica, sans-serif;
    background-color: brown;
    padding: 20px;
    margin: 20px 0;
    border: 2px dashed gray;
}
</style>
</head>
<body>
<h2>Student Information CSS Box Model Demonstration</h2>
<div class="info">
    <p><b>Student Database:</b></p>
    <ul>
        <li>Contact - Student Name : Mr. John William</li>
        <li>Percentage - 90%</li>
        <li>Branch : MCA</li>
        <li>University - MITWPU</li>
    </ul>
</div>
<div class="box">
    This is the Student CONTENT AREA.<br>
    (250px  100px)<br>
</div>
</body>
</html>
```

Output


Welcome Zaid
Student Information CSS Box Model Demonstration
Student Database:
Contact - Student Name : Mr. John William
Percentage - 90%
Branch : MCA
University - MITWPU
This is the Student CONTENT AREA.
(250px × 100px)

Assignment 08 – Pseudo Class Hover and Bootstrap Buttons

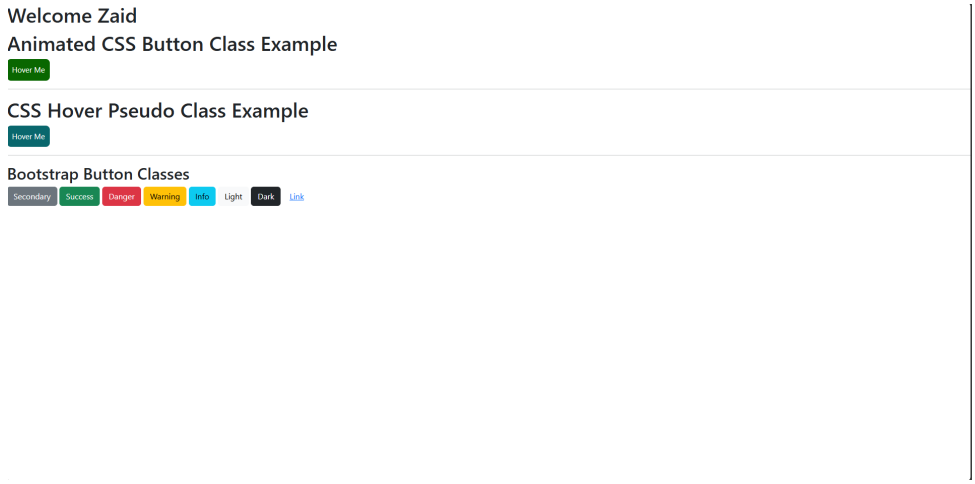
Aim: Create a web page demonstrating **CSS pseudo-class :hover** effects and use **Bootstrap button classes** for styled buttons.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>This is pseudo class CSS hover effect and bootstrap buttons</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.
    css" rel="stylesheet">

<style>
/* Basic Button Class */
.btn-class{
    background-color: rgb(9, 103, 0);
    color: white;
    padding: 10px;
    border: 0;
    border-radius: 8px;
    font-size: 15px;
}
/* Hover Button */
.btn-hover{
    background-color: rgb(9, 103, 109);
    color: white;
    padding: 10px;
    border: 0;
    border-radius: 8px;
    font-size: 15px;
}
/* Hover Animation */
.btn-hover:hover{
    background-color: rgb(127, 196, 121);
    transform: scale(1.1);
    transition: 0.4s;
}
/* General Button Styling */
.btnf{
    display: inline-flex;
    padding: 10px;
    transition: all 0.5s ease;
    cursor: pointer;
}
```

```
/* Hover Translation Effect */
.btnf:hover{
    transform: translateX(5px);
    box-shadow: 0px 8px 15px rgba(0,0,0,0.2);
}
</style>
</head>
<body>
<h1>Animated CSS Button Class Example</h1>
<button class="btn btn-class">Hover Me</button>
<hr>
<h1>CSS Hover Pseudo Class Example</h1>
<button class="btn-hover">Hover Me</button>
<hr>
<h2>Bootstrap Button Classes</h2>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>
<button type="button" class="btn btn-link">Link</button>
</body>
</html>
```

Output



Assignment 09 – CSS 2D Transform & Attribute Selectors

Aim: Create a web page demonstrating **CSS 2D transform** effects (rotate, translate, scale, skew, combination) and another page demonstrating different **CSS attribute selectors**.

Source Code – CSS 2D Transform Demo

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>CSS 2D Transform Demo</title>

<style>
body{
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-
        serif;
    background: #d0d6ce; /* light beige background */
    padding: 40px;
    margin: 0;
    color: #4a3e2e; /* dark brown text */
}

h2, p{
    text-align: center;
}

.container{
    display: flex;
    flex-wrap: wrap;
    gap: 15px;
    justify-content: center;
    align-items: center;
    margin-top: 40px;
}

.box{
    width: 140px;
    height: 140px;
    background: #8edbaa; /* soft terracotta */
    border: 2px solid #4b2e2e; /* dark brown border */
    display: flex;
    justify-content: center;
    align-items: center;
    font-weight: bold;
    font-size: 1.1em;
    color: #4b2e2e;
```

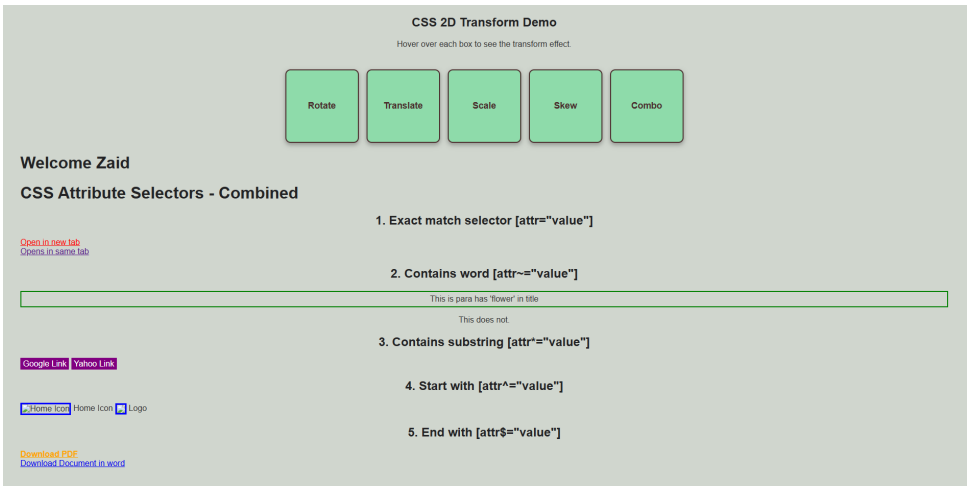
```
    border-radius: 10px;
    box-shadow: 0px 4px 9px rgba(0,0,0,0.3);
    transition: transform 0.5s ease-in-out;
}

/* Rotate */
.rotate:hover{
    transform: rotate(15deg);
}

/* Translate */
.translate:hover{
    transform: translate(15px, 25px);
}

/* Scale */
.scale:hover{
    transform: scale(1.3);
}

/* Skew */
.skew:hover{
    transform: skew(5deg, 3deg);
}

/* Combo  combination of multiple transforms */
.combo:hover{
    transform: rotate(10deg) scale(1.2) translate(10px, 20px);
}
</style>
</head>

<body>

<h2>CSS 2D Transform Demo</h2>
<p>Hover over each box to see the transform effect.</p>

<div class="container">
    <div class="box rotate">Rotate</div>
    <div class="box translate">Translate</div>
    <div class="box scale">Scale</div>
    <div class="box skew">Skew</div>
    <div class="box combo">Combo</div>
</div>

</body>
</html>
```

Source Code – CSS Attribute Selectors Demo

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>CSS Attribute Selectors Demo</title>

<style>
body{
    font-family: sans-serif;
    color: #333;
}

h1, h2{
    color: rgb(34,34,36);
}

/* 1. Exact match selector [attr="value"] */
a[target="_blank"]{
    color: red;
}

/* 2. Contains word selector [attr~="value"] */
p[title~="flower"]{
    border: 2px solid green;
    padding: 5px;
}

/* 3. Contains substring selector [attr*="value"] */
a[class*="link"]{
    background-color: purple;
    color: white;
    padding: 3px 5px;
    text-decoration: none;
}

/* 4. Starts with selector [attr^="value"] */
img[src^="pi"]{
    border: 3px solid blue;
    vertical-align: middle;
}

/* 5. Ends with selector [attr$=".pdf"] */
a[href$=".pdf"]{
    color: orange;
    font-weight: bold;
}
</style>
</head>

<body>
```

```
<h1>Welcome Zaid </h1>
<h1>CSS Attribute Selectors - Combined</h1>

<h2>1. Exact match selector [attr="value"]</h2>
<a href="#" target="_blank">Open in new tab</a>
<br>
<a href="#">Opens in same tab</a>

<h2>2. Contains word [attr~="value"]</h2>
<p title="This is a beautiful flower paragraph">This is para has 'flower' in
    title</p>
<p>This does not.</p>

<h2>3. Contains substring [attr*="value"]</h2>
<a href="#" class="google-link">Google Link</a>
<a href="#" class="yahoo-link">Yahoo Link</a>

<h2>4. Start with [attr^="value"]</h2>
<img src="pi.png" alt="Home Icon" width="20"> Home Icon
<img src="pione.png" width="20"> Logo

<h2>5. End with [attr$="value"]</h2>
<a href="files/document.pdf">Download PDF</a>
<br>
<a href="files/document.docx">Download Document in word</a>

</body>
</html>
```

Output



Assignment 10 – JavaScript Events Demo

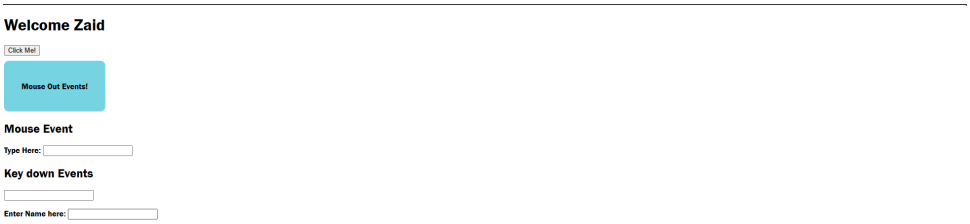
Aim: Create a page demonstrating different **JavaScript events** like click, mouseover, mouseout, keyup, keydown, and form events.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>JavaScript Events</title>

<style>
body {
    font-family: "Franklin Gothic Medium", "Arial Narrow", Arial, sans-serif;
}

.Mouse-Events{
    background-color: rgb(118, 211, 226);
    width: 200px;
    height: 100px;
    text-align: center;
    margin-top: 10px;
    border-radius: 10px;
    transition: 0.3s ease;
    line-height: 100px;
}
</style>
</head>

<body>

<button type="button" id="Event1" onclick="info()">Click Me!</button>

<div class="Mouse-Events"
     onmouseover="chngClr(this)"
     onmouseout="rstClr(this)">
    Mouse Out Events!
</div>

<h2>Mouse Event</h2>
Type Here:
<input type="text" id="textInput" onkeyup="keyPress(event)">
<p id="keyMsg"></p>

<h2>Key down Events</h2>
<input type="text" id="keydown" onkeydown="keydown(event)">
<p id="keyMsg1"></p>
```

```
<form onclick="show(event)">
    Enter Name here:
    <input type="text" id="txt1"><br>
    <label id="lb1"></label>
</form>

<script>
// CLICK EVENT
function info() {
    document.getElementById("Event1").innerHTML =
        "Hello!! Welcome to click events";
}

// MOUSEOVER EVENT
function chngClr(ele) {
    ele.style.backgroundColor = "bisque";
    ele.innerHTML = "Mouse Over Events!";
}

// MOUSEOUT EVENT
function rstClr(ele) {
    ele.style.backgroundColor = "teal";
    ele.innerHTML = "Mouse Out Events!";
}

// KEYPRESS EVENT
function keyPress(event) {
    let name = document.getElementById("textInput").value;
    document.getElementById("keyMsg").innerHTML = "Hello " + name + " !";
}

// KEYDOWN EVENT
function keydown(event) {
    let input = event.target;
    document.getElementById("keyMsg1").innerHTML =
        "Key Pressed : " + input.value.length;
}

// FORM CLICK EVENT
function show(event) {
    let name = document.getElementById("txt1").value;
    document.getElementById("lb1").innerHTML = "name is " + name;
}
</script>

</body>
</html>
```

Output


Welcome Zaid
Click Me!
Mouse Out Events!
Mouse Event
Type Here:
Key down Events
Enter Name here:

Assignment 11 – JavaScript Strings

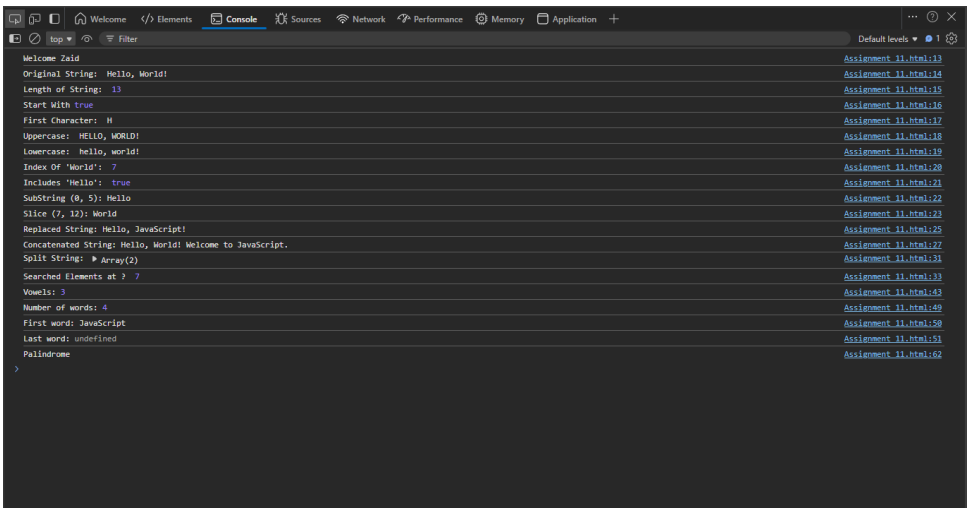
Aim: Write a JavaScript program to demonstrate various **string methods**, count vowels, count words in a sentence and check for **palindrome**.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>JavaScript Strings</title>
</head>
<body>

<script>
let str = "Hello, World!";
let string1 = "JavaScript is OOP language";
let cnt = 0;

console.log("Original String: ", str);
console.log("Length of String: ", str.length);
console.log("Start With", str.startsWith("Hello"));
console.log("First Character: ", str[0]);
console.log("Uppercase: ", str.toUpperCase());
console.log("Lowercase: ", str.toLowerCase());
console.log("Index Of 'World': ", str.indexOf("World"));
console.log("Includes 'Hello': ", str.includes("Hello"));
console.log("SubString (0, 5):", str.substring(0, 5));
console.log("Slice (7, 12):", str.slice(7, 12));

console.log("Replaced String:", str.replace("World", "JavaScript"));

console.log("Concatenated String:",
    str.concat(" Welcome to JavaScript.")
);

console.log("Split String:", str.split(", "));

console.log("Searched Elements at ? ",
    str.search("World")
);

// Count vowels
for (let char of str) {
    if ("aeiou".includes(char.toLowerCase())) {
        cnt += 1;
    }
}
console.log("Vowels:", cnt);
```

```
// Count words in second string
let wrd = string1.split(" ");
let wcnt = wrd.length;

console.log("Number of words:", wcnt);
console.log("First word:", wrd[0]);
console.log("Last word:", wrd[4]);

// Palindrome
let pstr = "wow";
let rev = "";

for (let i = pstr.length - 1; i >= 0; i--) {
    rev += pstr[i];
}

if (pstr === rev) {
    console.log("Palindrome");
} else {
    console.log("Not Palindrome");
}
</script>

</body>
</html>
```

Output



Assignment 12 – JavaScript Conditional Statements

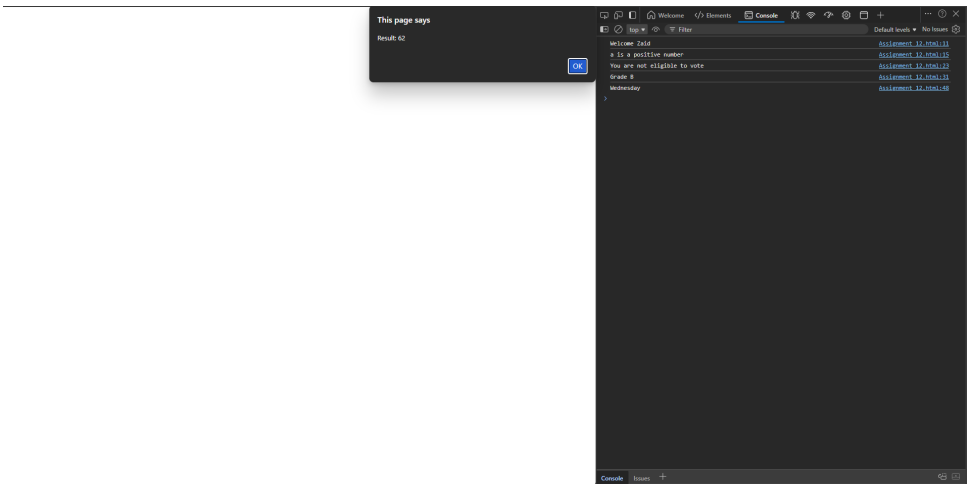
Aim: Demonstrate **if**, **if–else**, **if–else if–else**, **switch** statements and a simple calculator using **conditional statements**.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>JavaScript Conditional Statements</title>
</head>
<body>

<script>
// 1. If Statement
let a = 10;
if (a > 0) {
    console.log("a is a positive number");
}

// 2. IfElse Statement
let age = 15;
if (age >= 18) {
    console.log("You are eligible to vote");
} else {
    console.log("You are not eligible to vote");
}

// 3. IfElse IfElse Statement
let marks = 85;
if (marks >= 90) {
    console.log("Grade A");
} else if (marks >= 75) {
    console.log("Grade B");
} else if (marks >= 50) {
    console.log("Grade C");
} else {
    console.log("Grade F");
}

// 4. Switch Statement
let day = 3;
switch (day) {
    case 1:
        console.log("Monday");
        break;
    case 2:
        console.log("Tuesday");
```

```
        break;
    case 3:
        console.log("Wednesday");
        break;
    case 4:
        console.log("Thursday");
        break;
    case 5:
        console.log("Friday");
        break;
    case 6:
        console.log("Saturday");
        break;
    case 7:
        console.log("Sunday");
        break;
    default:
        console.log("Invalid day");
}

// 5. Simple Calculator using Conditional Statements
let num1 = parseFloat(prompt("Enter the first number:"));
let operator = prompt("Enter operator (+, -, *, /)");
let num2 = parseFloat(prompt("Enter the second number:"));
let result;

if (operator === "+") {
    result = num1 + num2;
} else if (operator === "-") {
    result = num1 - num2;
} else if (operator === "*") {
    result = num1 * num2;
} else if (operator === "/") {
    if (num2 === 0) {
        result = "Error: Division by zero is not allowed.";
    } else {
        result = num1 / num2;
    }
} else {
    result = "Invalid operator!";
}

alert("Result: " + result);
console.log("Result: " + result);
</script>

</body>
</html>
```

Output


This page says
Result: 62
OK
Welcome Zaid
a is a positive number
You are not eligible to vote
Grade B
Wednesday

Assignment 13 – JavaScript Arrays

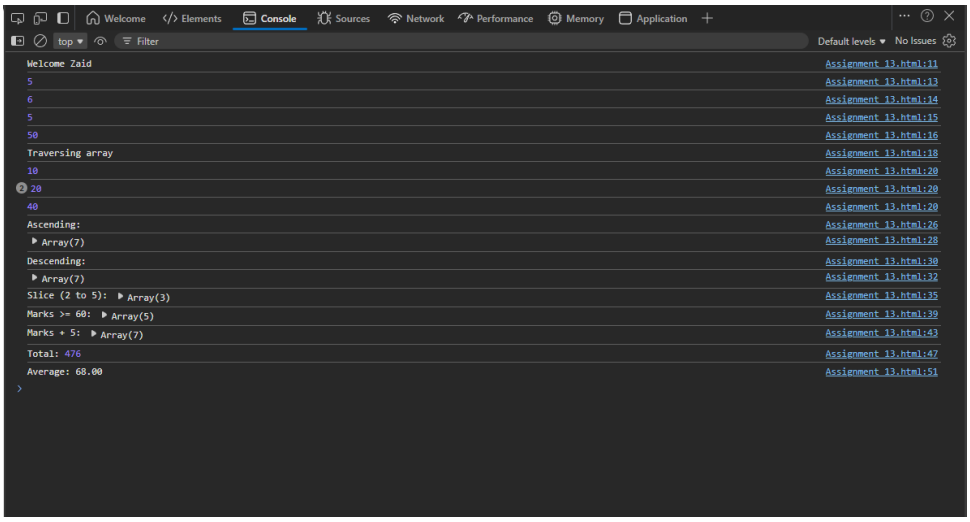
Aim: Write a JavaScript program to demonstrate **array operations** such as push, pop, shift, unshift, traversing, sorting, slice, filter, map, reduce and average.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>JavaScript Array</title>
</head>
<body>

<script>
let nums = [10, 20, 20, 40];
console.log(nums.push(50)); // push
console.log(nums.unshift(5)); // unshift
console.log(nums.shift()); // shift
console.log(nums.pop()); // pop

console.log("Traversing array");
for (let i = 0; i < nums.length; i++) {
    console.log(nums[i]);
}

// Sorting Example
let marks = [74, 70, 23, 59, 67, 88, 95];

console.log("Ascending:");
marks.sort((a, b) => a - b);
console.log(marks);

console.log("Descending:");
marks.sort((a, b) => b - a);
console.log(marks);

// Slice (2 to 5)
console.log("Slice (2 to 5):", marks.slice(2, 5));

// Filter marks >= 60
let passed = marks.filter(m => m >= 60);
console.log("Marks >= 60:", passed);

// Map  Add 5 to each marks
let newMarks = marks.map(m => m + 5);
console.log("Marks + 5:", newMarks);

// Reduce  sum
```

```
let total = marks.reduce((sum, m) => sum + m, 0);
console.log("Total:", total);

// Average
let avg = total / marks.length;
console.log("Average:", avg.toFixed(2));
</script>

</body>
</html>
```

Output



Assignment 14 – JavaScript Functions & Objects

Aim: (a) Demonstrate JavaScript **functions** for total, average, and grade calculation. (b) Demonstrate JavaScript **objects**, nested objects, array of objects, methods, and property manipulation.

Source Code – Part A: JavaScript Functions

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>JavaScript Functions</title>
</head>
<body>
<h1>Welcome Zaid </h1>
<script>
const marks = [85, 92, 78];


const Total = (arr) => arr.reduce((sum, m) => sum + m, 0);


const Avg = (arr) => Total(arr) / arr.length;


const Grade = (avg) => {
    if (avg >= 90)
        return "A";
    else if (avg >= 75)
        return "B";
    else if (avg >= 60)
        return "C";
    else
        return "D";
};

let total = Total(marks);
let avg = Avg(marks);
let grade = Grade(avg);

console.log("Marks:", marks);
console.log("Total:", total);
console.log("Average:", avg.toFixed(2));
console.log("Grade:", grade);
</script>

</body>
</html>
```

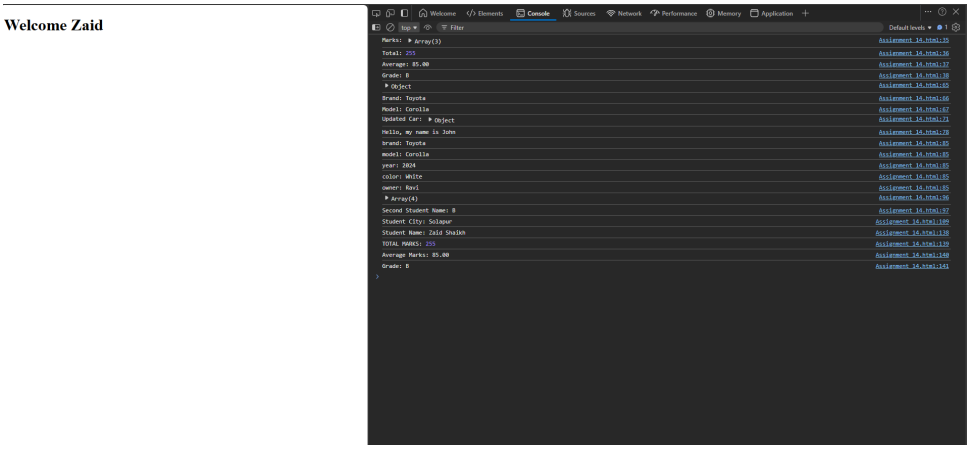
Source Code – Part B: JavaScript Objects and Methods

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>

<script>

let car = {
    brand: "Toyota",
    model: "Corolla",
    year: 2021,
    color: "White"
};

console.log(car);
console.log("Brand:", car.brand);
console.log("Model:", car["model"]);

car.owner = "Ravi";
car.year = 2024;
console.log("Updated Car:", car);


let person = {
    name: "John",
    age: 22,
    greet: function () {
        console.log("Hello, my name is " + this.name);
    }
};

person.greet();

for (let key in car) {
    console.log(key + ": " + car[key]);
}


let students = [
    { name: "A", age: 18, grade: "A" },
    { name: "B", age: 19, grade: "A" },
    { name: "C", age: 17, grade: "B" },
    { name: "D", age: 20, grade: "C" }
];
```

```
console.log(students);
console.log("Second Student Name:", students[1].name);


let employee = {
    name: "Zaid Vedi",
    address: {
        city: "Solapur",
        country: "India"
    },
    position: "Student"
};

console.log("Student City:", employee.address.city);

let student = {
    name: "Zaid Shaikh",
    rollNo: 12,
    marks: {
        english: 85,
        math: 90,
        science: 80
    },

    totalMarks: function () {
        return this.marks.english + this.marks.math + this.marks.science;
    },

    avgMarks: function () {
        return this.totalMarks() / 3;
    },

    calGrade: function () {
        let avg = this.avgMarks();
        if (avg >= 90) return "A";
        else if (avg >= 75) return "B";
        else if (avg >= 60) return "C";
        else return "D";
    },

    display: function () {
        console.log("Student Name:", this.name);
        console.log("TOTAL MARKS:", this.totalMarks());
        console.log("Average Marks:", this.avgMarks().toFixed(2));
        console.log("Grade:", this.calGrade());
    }
};

student.display();
</script>
```

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

Output



Assignment 15 – JavaScript DOM Selectors

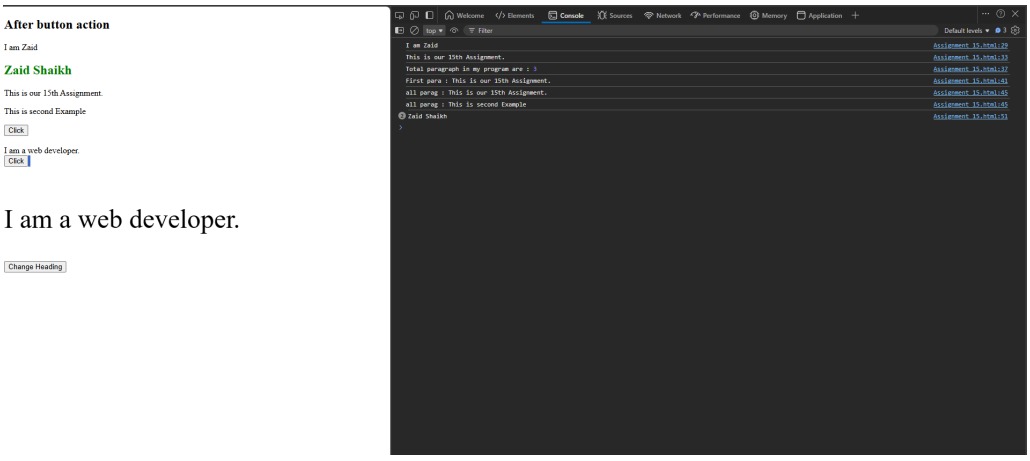
Aim: Demonstrate different **DOM selectors** in JavaScript.

Source Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>DOM Selector</title>
</head>

<body>

<h2 id="btn">Before button action</h2>
<h2 id="mainTitl">Dom selector Example</h2>
<h2 id="call">Zaid Shaikh</h2>
<p class="info">This is our 15th Assignment.</p>
<p class="info">This is second Example</p>

<button id="change">Click</button><br><br>

<label id="label">I am a web developer.</label><br>
<button id="click">Click</button><br><br>

<p id="demo"></p>

<button id="heading">Change Heading</button>

<script>

let a = document.getElementById("mainTitl");
console.log(a.outerText = "I am Zaid");

let par = document.getElementsByClassName("info");
console.log(par[0].innerText);

let tagName = document.getElementsByTagName("p");
console.log("Total paragraph in my program are :", tagName.length);

let firstPara = document.querySelector(".info");
console.log("First para :", firstPara.innerText);


let allInfo = document.querySelectorAll(".info");
allInfo.forEach(p => console.log("all parag :", p.innerText));


let b = document.getElementById("call");
```

```
document.getElementById("change").addEventListener("click", function () {
    console.log(b.innerText = "Zaid Shaikh");
    b.style.color = "green";
});

let label = document.getElementById("label");
let p = document.getElementById("demo");
document.getElementById("click").addEventListener("click", function () {
    p.innerText = label.innerText;
    p.style.fontSize = "50px";
});

let btn = document.getElementById("btn");

document.getElementById("heading").addEventListener("click", function () {
    btn.innerText = "After button action";
});
</script>

</body>
</html>
```

Output



Assignment 16 – React Application

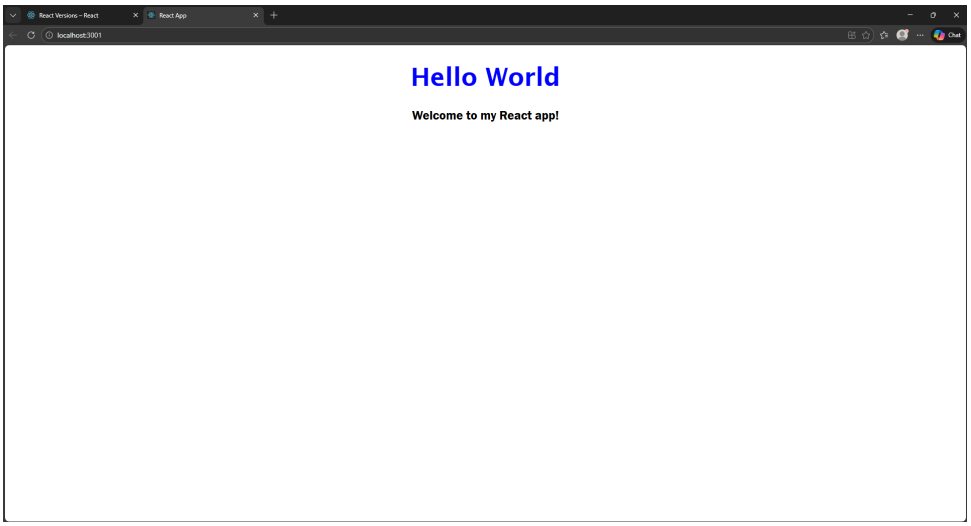
Aim: Create React Application

Program 1 – React Hello World Component

Source Code

```
import React from "react";

function App() {
  return (
    <div>
      <h1 style={{color:"blue"}}>Hello World</h1>
      <p>Welcome to my React app!</p>
    </div>
  );
}

export default App;
```

Output

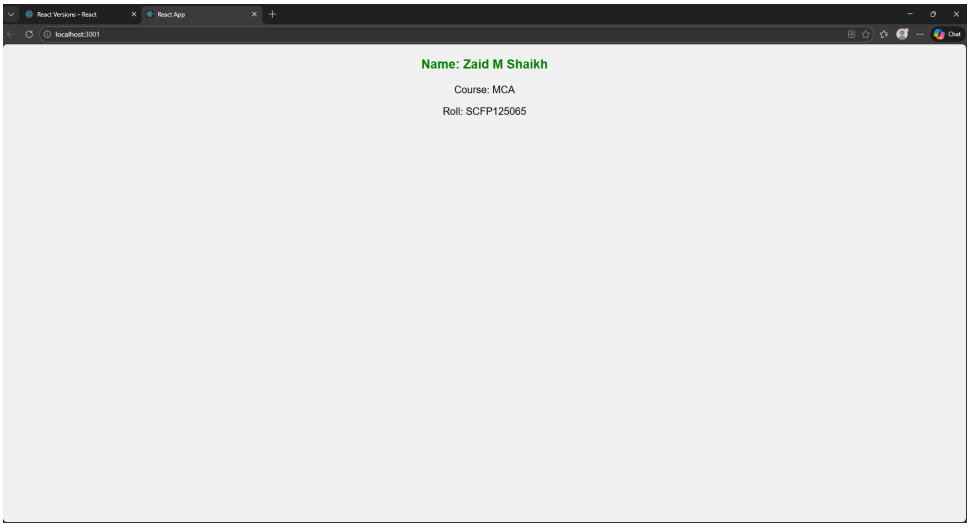


Program 2 – Display Student Information Using Props

Source Code

```
import React from "react";

function StudentInfo() {
  return (
    <div style={{textAlign:"center"}}>
      <h2 style={{color:"green"}}>Name: Zaid M Shaikh</h2>
      <p>Course: MCA</p>
      <p>Ref: SCFP125065</p>
    </div>
  );
}

export default function App() {
  return (
    <>
      <StudentInfo />
    </>
  );
}
```

Output

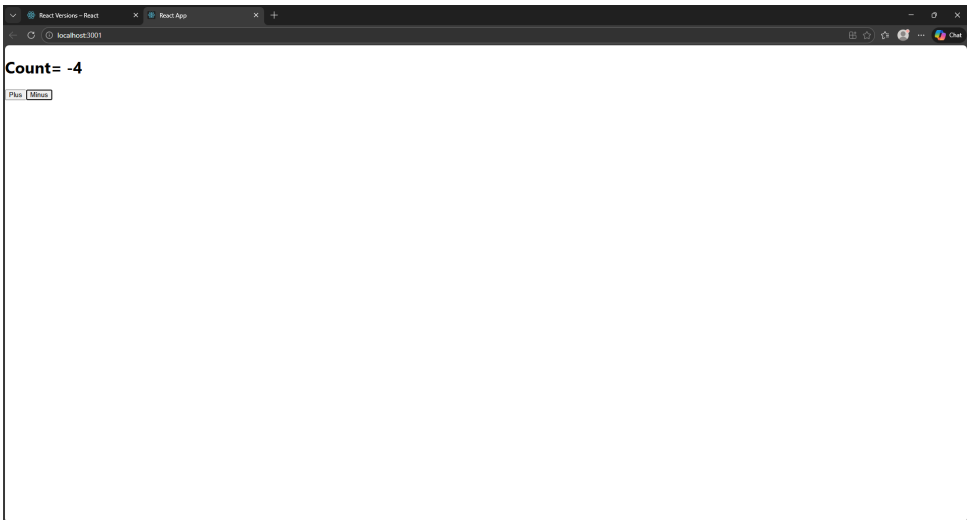


Program 3 – React Counter App

Source Code

```
import React, { useState } from "react";

export default function App() {
  const [count, setCount] = useState(0);

  return (
    <>
      <h1>Count = {count}</h1>

      <button onClick={() => setCount(count + 1)}>Inc</button>
      <button onClick={() => setCount(count - 1)}>Dec</button>
    </>
  );
}
```

Output

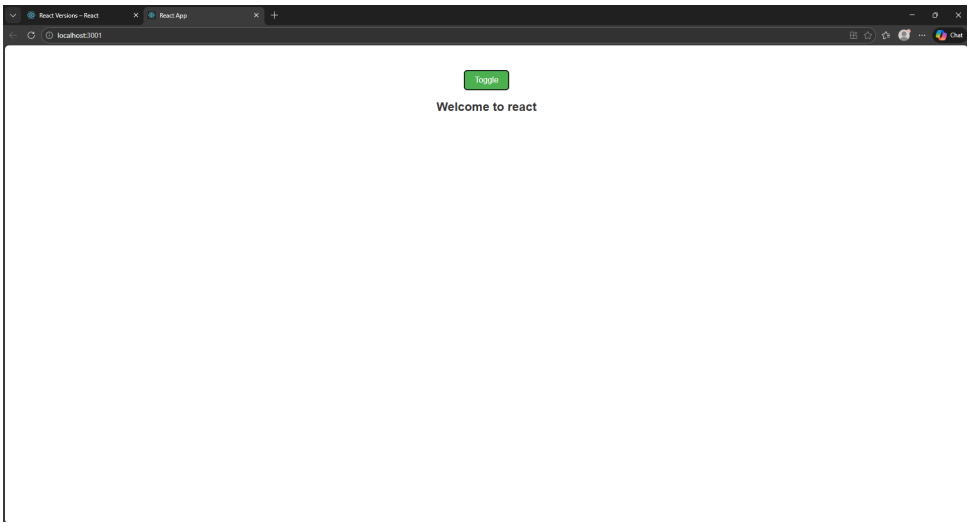


Program 4 – React Toggle Message

Source Code

```
import React, { useState } from "react";

export default function App() {
  const [show, setShow] = useState(false);

  return (
    <>
      <button style={{background:"green", color:"white"}}
              onClick={() => setShow(!show)}>
        Toggle
      </button>

      {show && <h2>Welcome to react</h2>}
    </>
  );
}
```

Output



Program 5 – React Array Map Example

Source Code

```
import React from "react";

export default function App() {
  let arr = [1, 2, 3, 4];

  return (
    <div>
      {arr.map((n) => (
        <h3>{n * 2}</h3>
      ))}
    </div>
  );
}
```

Output