MIT Vishwaprayag University

School of Computing

FSD ASSIGNMENTS
(Basic Programs)

Submitted By:

Name: **Atherv Deepak Telkar**

Roll No.: **SCFP125043**

Course: **MCA 1 Year (Sem1)**

Submitted To:

Instructor: **Prof. Anand Shimpi**

Academic Year 2024–2025

Index – Page 1

Web Development & JavaScript Assignment Index
Consolidated Coursework

No.	Assignment / Topic	Focus & Key Concepts
1	HTML Basics: Tags and Table Creation	**Focus:** Fundamental HTML structure and Data Tables. **Basic Tags:** <h1> to <h6>, <b>, <i>, <u>, <del>, <sub>, <sup>. **Table Attributes:** <table>, <tr>, <th>, <td>. Attributes: cellpadding, cellspacing, border, bgcolor. Merging with rowspan and colspan.
2	HTML Forms (Forms with Different At- tributes)	**Focus:** Building user input interfaces using the <form> element. **Inputs:** Text, Email, Number, Radio Buttons (Contact Preference, Yes/No). **Structure:** Usage of <fieldset> and <legend> for grouping.
3	Assignment No 3: HTML Select Ele- ment	**Focus:** Advanced Form Elements and Dropdowns. **Elements:** <select> and <option> for dropdowns. **Advanced Inputs:** Date, Month, Color, File, Range (sliders), Textarea.
4	CSS Selectors Demo	**Focus:** Targeting HTML elements for styling. **Types:** Tag, ID (#id), Class (.class), Group, Uni- versal (*). **Advanced Attributes:** Exact match [attr="val"], Contains word [~=], Substring [*=], Starts with [^=], Ends with [$=].
5	Assignment 6: CSS Positioning and Overflow	**Focus:** Layout control and handling excess content. **Positioning:** static, relative, absolute, fixed, sticky. **Overflow:** visible, hidden, scroll, auto.

Index – Page 2

Web Development & JavaScript Assignment Index
Consolidated Coursework

No.	Assignment / Topic	Focus & Key Concepts
1	**HTML Basics: Tags and Table Creation**	**Focus:** Fundamental HTML structure and Data Tables. **Basic Tags:** <h1> to <h6>, <b>, <i>, <u>, <del>, <sub>, <sup>. **Table Attributes:** <table>, <tr>, <th>, <td>. Attributes: cellpadding, cellspacing, border, bgcolor. Merging with rowspan and colspan.
2	**HTML Forms (Forms with Different At- trib utes)**	**Focus:** Building user input interfaces using the <form> element. **Inputs:** Text, Email, Number, Radio Buttons (Contact Preference, Yes/No). **Structure:** Usage of <fieldset> and <legend> for grouping.
3	**Assignment No 3 : HTML Select Ele- ment**	**Focus:** Advanced Form Elements and Dropdowns. **Elements:** <select> and <option> for dropdowns. **Advanced Inputs:** Date, Month, Color, File, Range (sliders), Textarea.
4	**CSS Selectors Demo**	**Focus:** Targeting HTML elements for styling. **Types:** Tag, ID (#id), Class (.class), Group, Uni- versal (*). **Advanced Attributes:** Exact match [attr="val"], Contains word [~=], Substring [*=], Starts with [^=], Ends with [$=].
5	**Assignment 6: CSS Positioning and Overfl ow**	**Focus:** Layout control and handling excess content. **Positioning:** static, relative, absolute, fixed, sticky. **Overflow:** visible, hidden, scroll, auto.

No.	Assignment / Topic	Focus & Key Concepts
14	**Assignment 14 (Part A): Grading System**	**Focus:** Logic and iteration. **Process:** Iterating through marks array, calculating Total/Average, and assigning grades using If/Else-If ladders.
15	**Assignment 14 (Part B): JavaScript Ob- jects**	**Focus:** Object-Oriented JavaScript basics. **Concepts:** Object syntax, Methods inside objects, Dot notation, Nested Objects, Arrays of Objects.
16	**Assignment 15: DOM Manipulation**	**Focus:** Dynamic HTML interaction. **Selectors:** getElementById, querySelector, etc. **Manipulation:** changing innerText, style.color, reading .value. **Listeners:** addEventListener.

Assignment 1: HTML Basic Tags and Table Creation

Code

```
1   <!DOCTYPE    html >
2   <html lang="en">
3   <head >
4       <meta charset="UTF-8">
5       <meta name="viewport" content="width=device-width, initial-scale
        =1.0">
6       <title >Table </title >
7   </ head >
8   <body >
9       <h1 align="center">This is table</h1>
10      <b align="center">Table 1</b><br>
11      <i align="center">Table 2</i><br>
12      <u align="center">Table 3</u><br>
13
14      <strong align="center">Table 4</strong><br>
15      <small align="center">Table 5</small><br>
16
17      <em align="center">Table 6</em><br>
18      <del align="center">Table 7</del><br>
19      <ins align="center">Table 8</ins><br>
20      <sub align="center">Table 9</sub><br>
21      <sup align="center">Table 10</sup><br>
22
23      <br><br>
24
25      <table bgcolor="orange" align="center" border="2" cellspacing="5">
26          <caption >Table  Records </caption >
27          <tr>
28              <th rowspan="2"><font color="white">Rollno </font></th >
29              <th colspan="1" rowspan="2"><font color="white">Name </font>
        </th >
30              <th colspan="3"><font color="white">Subjects </font></th >
31              <th rowspan="2"><font color="white">Total </font></th >
32          </tr>
33          <tr bgcolor="#00ffee">
34              <th font color="white">Machine Learning </th >
35              <th font color="white">Python </th >
36              <th font color="white">FSD </th >
37          </tr>
38
39          <tr>
```

```
40            <td >2 </td >
41            <td >Om </td >
42            <td > 50 </td >
43            <td > 50 </td >
44            <td > 50 </td >
45            <td > 150 </td >
46        </ tr>
47        <tr>
48            <td >3 </td >
49            <td >Atherv </td >
50            <td > 50 </td >
51            <td > 60 </td >
52            <td > 60 </td >
53            <td > 170 </td >
54        </ tr>
55        <tr>
56            <td >4 </td >
57            <td >Rushikesh </td >
58            <td > 90 </td >
59            <td > 89 </td >
60            <td > 96 </td >
61            <td > 300 </td >
62        </ tr>
63      </ table  >
64 </ body  >
65 </ html  >
```
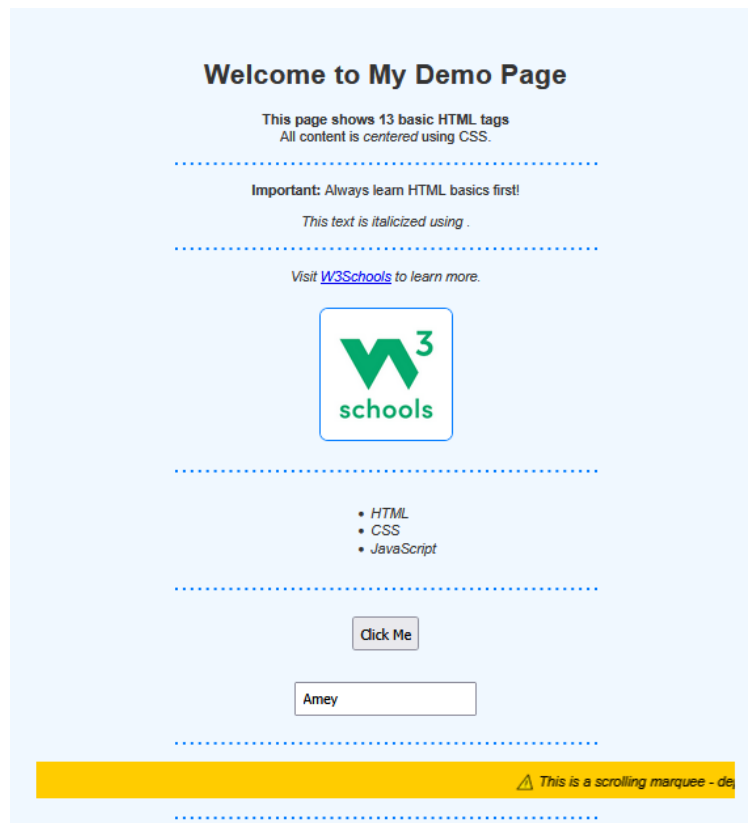
Listing 1.1: index.html

Output

[Insert Screenshot of Table Here]

Assignment 2: Forms with Different Attributes

Code

```
1  <!DOCTYPE    html >
2  <html  lang=" en">
3  <head >
4      <meta  charset=" UTF-8">
5      <title >Table </title >
6  </ head >
7  <body >
8      <form  action ="./ blank"  method ="post"  bgcolor="" >
9          <h1>Student  Registration  for  training  and  placement </h1>
10         <fieldset >
11             <legend >Student    Registration </legend >
12
13             Name : <input  type ="text"  name =" name "><br><br>
14             Email -ID: <input  type =" email"  name =" email" ><br><br>
15             Mobile  Number: <input  type =" number"  name =" number" ><br><br>
16
17             Contact  Preference :<br>
18             <input  type =" radio "  name =" Contact" >Email <br>
19             <input  type =" radio "  name =" Contact" >Phone  Number <br><br>
20
21             <label name =" news" >Do you want to Subscribe  newsletters </
   label ><br>
22             Yes: <input  type =" radio "  name =" yes" >
23             No: <input  type =" radio "  name =" yes" ><br><br>
24
25             <input  type =" submit"  value =" submit" >
26         </ fieldset >
27     </ form >
28  </ body >
29  </ html >
```

Listing 2.1: form.html

Output

[Insert Screenshot of Form Here]

Student Atherv Telkar are maintained in the application

Student Records are maintained in the application

Student Rushikesh chintanpalli are maintained in the application

Student Records are maintained in the application

Student Abhishek Kambale are maintained in the application

Student Records are maintained in the application

Student Records are maintained in the application

Student Records are maintained in the application

Student Records are maintained in the application

Student Records are maintained in the application

Student Records are maintained in the application

3

Assignment 3: HTML Select Element and Advanced Inputs

Code

```
1  <!DOCTYPE    html >
2  <html  lang=" en">
3  <body >
4      <h1>How  the  Textarea  looks </h1>
5      <fieldset >
6          <legend >Employees   Details   for  Accounting </legend >
7
8      <form   action ="./ blank"   method ="post" >
9          Upload : <input  type =" file "  name =" myfile "  /> <br><br>
10
11         Role :  <select   name ="mylab ">
12             <option   value ="teacher" >Teacher </option >
13             <option   value ="dev" >Developer </option >
14             <option  value ="devops" >DevOps  Engineer </option >
15             <option   value ="tester" >Tester </option >
16         </ select ><br><br>
17
18         Email:  <input   type =" email"    name =" email"   size =" 26 "><br>
19         Mobile  No: <input  type =" tel"  name =" phone "  pattern ="[0 -9]{10}" ><
    br>
20         Range  Example : <input  type =" range "  min =" 5 "  max=" 10 "><br>
21         Birth   Date : <input   type =" date  "  name =" birth "><br>
22         Month :   <input    type =" month "   name =" month "><br>
23         Choose  your  fav  color:  <input   type =" color"   name =" bgcol" ><br>
24
25         University  Name : <input  type =" text"  name =" us"  value =" MITVPU "
    disabled ><br>
26         <textarea  name ="address"  placeholder=" enter  your  address"  rows=
    " 3 "  cols=" 50 " ></ textarea ><br>
27
28         Bank  Account  No: <input  type =" password "  name =" Account" ><br>
29         Search   the  Employee   ID:  <input   type =" search " ><br>
30
31         <input  type =" button "  onClick ="alert(' Please  look  at  Board ')"
    value =" Click"/ >
32         <input   type =" reset"   value =" reset" >
33         <input    type =" submit"   value =" submit" >
34     </ form >
35     </ fieldset >
36 </ body >
```

```
37 </html >
```

Output

[Insert Screenshot of Select Elements Here]

Assignment 4: CSS Selectors Demo

Code

```html
1  <!DOCTYPE html>
2  <html lang=" en">
3  <head>
4      <meta charset=" UTF-8">
5      <title>CSS Selectors Demo</title>
6      <style>
7          /* Tag Selector - styles all <p> elements */
8          p {
9              color: darkblue;
10             font-family: Arial, sans-serif;
11         }
12         /* ID Selector */
13         #unique {
14             background-color: lightyellow;
15             padding: 10px;
16             border: 2px solid orange;
17         }
18         /* Class Selector */
19         .highlight {
20             color: crimson;
21             font-weight: bold;
22         }
23         /* Group Selector */
24         h1, h2 {
25             font-family: 'Trebuchet MS', sans-serif;
26             color: teal;
27         }
28         /* Universal Selector */
29         * {
30             box-sizing: border-box;
31         }
32     </style>
33 </head>
34 <body>
35     <h1>CSS Selectors Demo</h1>
36     <h2>Tag Selector</h2>
37     <p>This paragraph is styled by the tag selector.</p>
38
39     <h2>ID Selector</h2>
40     <p id=" unique ">This paragraph is styled by the ID selector.</p>
41
42     <h2>Class Selector</h2>
```

```
43      <p class=" highlight" >This  paragraph  is  styled  by  the  class  selector
     .</p>
44      <p class=" highlight" >Another  paragraph  with  the  same  class  selector
     .</p>
45  </ body >
46  </ html >
```

Listing 4.1: selectors.html

Output

[Insert Screenshot of CSS Selectors Here]

Assignment 4 – CSS Selectors

| Home | Students | Contact |

Student List
Amey Telkar
Atherv Telkar

This is highlighted using an ID selector.

Assignment 5: Menu Driver Application (HTML & CSS)

(Includes Assignment 4 Part 2 & Merge with Assignment 5)

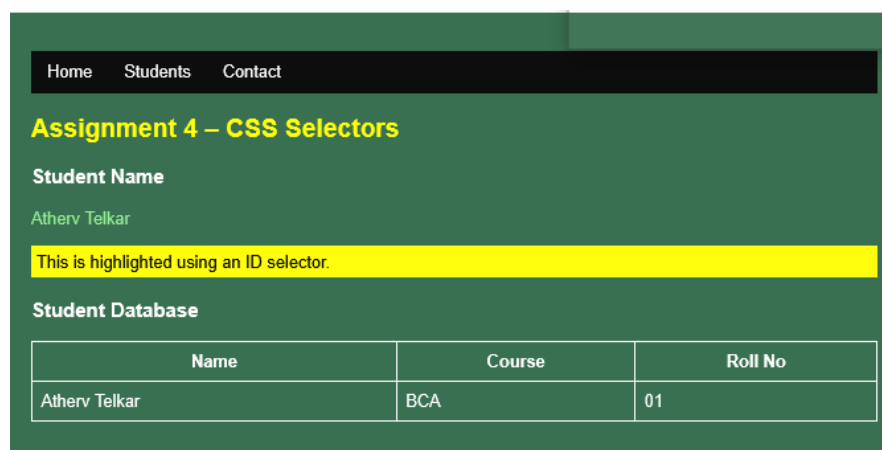## Code Structure

HTML (Login & Home Structure)

```html
1  <!-- Login Form -->
2  <div class="center-container" id="loginPage">
3      <div class="form-box">
4          <h2>MIT College Login</h2>
5          <input type="text" id="prn" placeholder="Enter PRN Number">
6          <input type="password" id="password" placeholder="Enter
   Password">
7          <button onclick="login()">Login</button>
8          <p>Don't have an account? <a href="#" onclick="showSignup()">
   Sign Up</a></p>
9      </div>
10 </div>
11
12 <!-- Menu Page (Hidden by default) -->
13 <div class="menu" id="menuPage" style="display:none;">
14     <header>
15         <nav>
16             <ul>
17                 <li><a href="#" onclick="showTab('home')">Home</a></li>
18                 <li><a href="#" onclick="showTab('services')">Services<
   /a></li>
19                 <li><a href="#" onclick="showTab('contact')">Contact</a
   ></li>
20                 <li><a href="#" onclick="logout()">Logout</a></li>
21             </ul>
22         </nav>
23     </header>
24
25     <section id="home">
26         <h2>Home</h2>
27         <p>Welcome to the MIT College Menu Driver Application!</p>
28         <img src="MIT_image.jpg" alt="MIT College Logo">
29     </section>
30 </div>
```

JavaScript (Logic for Navigation)

```
 1  function login () {
 2      let prn = document. getElementById ("prn"). value ;
 3      let password = document. getElementById ("password "). value ;
 4
 5      if     ( prn         &&          password )          {
 6          document. getElementById (" login Page "). style . display  =  " none ";
 7          document. getElementById (" menu Page "). style . display  =  " block ";
 8          show Tab (' home ');
 9      } else {
10          alert ("Please  enter  PRN  number  and  Password .");
11      }
12  }
13
14  function show Tab (tabId) {
15      let sections = document. querySelector All ("section ");
16      sections. forEach (sec => sec. style . display  = "none ");
17      document. getElementById (tabId). style . display  = "block ";
18  }
19
20  function                  logout    ()                       {
21      document. getElementById (" menu Page "). style . display  = "none ";
22      document. getElementById (" login Page "). style . display  = "flex ";
23      document. getElementById (" prn "). value      =    "";
24      document. getElementById ("password "). value  = "";
25      alert ("You  have  been  logged  out  successfully !");
26  }
```

Output

[Insert Screenshots of Login Page and Home Page Here]

Assignment 6: CSS Positioning and Overflow

Code

```
1  <!DOCTYPE html >
2  <html lang=" en">
3  <head >
4      <title >CSS  Position </title >
5      <style >
6          .container {
7              position : relative ;
8              width : 400 px;
9              height:  300 px;
10             border:  2 px solid  green ;
11         }
12         .box {
13             width : 120 px;
14             height:  50 px;
15             color: antiquewhite ;
16             text - align :  center;
17         }
18         .static - box {
19             position :  static;
20             background - color: rgb (6, 160, 160);
21         }
22         .relative -box {
23             position : relative ;
24             background - color: rgb (196 , 131 , 185);
25             top : 20 px; right: 10 px;
26         }
27         .absolute - box {
28             position : absolute ;
29             background - color: rgb (91, 72, 126);
30             top : 60 px; right: 10 px;
31         }
32         .sticky - box {
33             position :sticky;
34             top : 0;
35             background - color:  gray;
36         }
37     </style >
38  </head >
39  <body >
40      <h2 >CSS  Position  Demonstration </h2 >
41      <div  class=" container" >
```

```
42        <div class=" box static -box" >Static </div >
43        <div class=" box relative -box" >Relative </div >
44        <div class=" box sticky -box" >Sticky Scroll </div >
45        <div class=" box absolute -box" >Absolute </div >
46    </div >
47 </body >
48 </html >
```
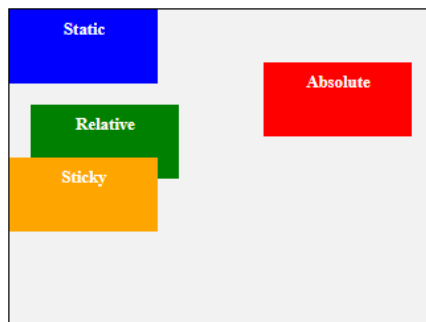
Listing 6.1: cssposition.html

Output

[Insert Screenshot of Positioning Demo Here]

Assignment 7: CSS Box Model

Code

```
1  body {
2      font - family: Cambria , serif;
3      background - color: # f4f4f9 ;
4      padding : 20 px;
5  }
6  .info {
7      padding : 20 px;
8      margin : 20 px auto ;
9      background : linear - gradient( to right , #0 e91e2 , # e4e5e6 );
10     width : 80%;
11     border: 2px solid #333;
12     border - radius: 10 px;
13     color: # fff;
14 }
15 .btn {
16     margin : 20 px auto ;
17     display : block;
18     padding : 12 px 24 px;
19     background -color: #088 f6f;
20     color: #000;
21     border - radius: 30 px;
22     cursor: pointer;
23 }
```

Listing 7.1: boxmodel.css

```
1  <div class="info">
2      <p><strong>Box Model in Use - Student  Layer:</strong></p>
3      <ul>
4          <li>Student Name : Actual Name and  Image </li>
5          <li>Padding : Creates space inside  the border</li>
6          <li>Margin : Creates space outside  the  border</li>
7          <li>Border: Encloses the content and padding</li>
8      </ul>
9      <button class="btn">Learn More</button >
10 </div >
```
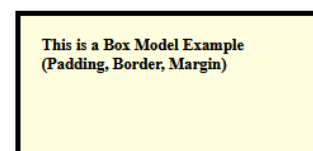
Listing 7.2: boxmodel.html

Output

[Insert Screenshot of Box Model Card Here]

Assignment 7 – CSS Box

Students: Amey Telkar & Atherv Telkar

This is a Box Model Example
(Padding, Border, Margin)

Student Names:
• Amey Telkar

12

Assignment 8: Pseudo Classes, Bootstrap & Animated Buttons

Code

```
1  .btn {
2      background-color: bisque;
3      color: brown;
4      padding: 12px 20px;
5      border: none;
6      cursor: pointer;
7      transition: 0.3s;
8  }
9  .btn:hover {
10     background-color: rgb(43, 160, 53);
11     transform: scale(1.1);
12 }
13 /* Keyframe Animation */
14 @keyframes pulse {
15     from { box-shadow: 0px 0px 10px #009978; }
16     to { box-shadow: 0px 0px 20px #00ffaa; }
17 }
18 .btn-animate :hover {
19     animation: pulse 1s infinite alternate;
20 }
```

Listing 8.1: animated*b*tn.css

Output

[Insert Screenshot of Buttons Here]

Assignment 8 – Pseudo Class, Bootstrap & Animation
Students: Amey Telkar & Atherv Telkar

Pseudo Class Example:
Hover over this link

Bootstrap Button:

Bootstrap Button

Animated Button:

Hover Me

Assignment 9: Combined Attribute Selectors & 2D Transforms

Code

```css
/* Attribute Selectors */
a[target ="_blank "]{color: rgb (0, 174, 255); font-weight: bold; }
a[href$ =".pdf"]    {   color:   brown; font-style:   italic;   }
img[src ^=" icon "] { border: 2px solid bisque; width: 50px; }

/* 2D Transforms */
.rotate :hover { transform : rotate (30deg); }
.scale :hover { transform : scale (1.2); }
.skew :hover { transform : skew (20deg, 10deg); }
.translate :hover { transform : translate (50px, 50px); }
```

Listing 9.1: transforms.css

Output

[Insert Screenshot of Transformations Here]

Assignment 9 – Attribute Selector & 2D Transform

Students: Amey Telkar & Atherv Telkar

Combined Attribute Selector

amey

2D Transformation Examples

Rotate   Scale   Translate

Assignment 10 – JavaScript Events

Students: Amey Telkar & Atherv Telkar

Click Event

Mouse Over / Out Event

Atherv   Type here

Form Events Example

atherv

Submit Form

Assignment 10: JS Datatypes, Operators & Event Handlers

Code (Operators)

```
1 let a = 10;
2 let b = 20;
3 document. writeln (" Arithmetic    Operators  <br >");
4 document. writeln ("a + b = " + (a + b) + "<br >");  // Addition
5 document. writeln ("b % a = " + (b % a) + "<br >");  // Modulus
6
7 let x = 5;
8 x += 5; // x = x + 5
9 document. writeln ("x += 5: " + x);
```

Code (Event Handlers)

```
1  <button   onclick =" show Message ()">Click   Button </button >
2  <div onmouseover=" change Color( this)" onmouseout=" resetColor( this)" >
3      Hover Over Me
4  </ div >
5  <input  type ="text"  oninput=" displayValue ()"  id=" name Input" >
6
7  <script >
8      function  show Message () {
9          alert(" You Clicked  the  button ");
10     }
11     function       change Color( elem )     {
12         elem .style .background Color  = "Orange ";
13     }
14     function        resetColor( elem )        {
15         elem .style .background Color  = "lightblue ";
16     }
17 </ script >
```

Output

[Insert Screenshot of JS Output Here]

Assignment 11:   JS Strings and Methods

Code

```
1  let str1 = " Hello ";
2  let str2 = "World ";
3
4  console .log( str1 + str2 );
5  console .log( str1 .length );
6  console . log( str1 . indexOf ("l"));
7  console .log( str1 .substring (0 , 4));
8
9  let sentence = "Welcome MCA - 1 student at FSD Tutorial ";
10 console . log( sentence    . includes    (" FSD   "));
11 console .log( sentence .replace (" FSD ", " Rushikesh "));
12 console . log( sentence . to Lower Case ());
```

Assignment 11 – JavaScript String Methods

Students: Amey Telkar & Atherv Telkar

| ☐ | ☐ Inspector | ▷ Console | ☐ Debugger | ↑↓ Network |

🗑 | ▽ Filter Output

Concatenation: Amey Telkar

Length of str1: 4

Character at index 2: e

Substring (0,3): Ame

Both strings equal? false

Index of 'Telkar': 7

Does sentence include 'JavaScript'? true

Before Replace: JavaScript is difficult

After Replace: JavaScript is easy

Uppercase: JAVASCRIPT IS DIFFICULT

Lowercase: javascript is difficult

Vowel Count: 11

Is 'madam' a palindrome? true

Total Words: 6

First Word: Amey

Last Word: students

Live reload enabled.

Assignment 12: Conditional Statements (Palindrome & Loops)

Code

```
1  // Palindrome Check
2  let str = " Java ";
3  let str2 = str. split (""). reverse (). join ("");
4
5  if (str == str2 ) {
6      console .log (" The string is a palindrome ");
7  } else {
8      console .log (" The string is not a palindrome ");
9  }
10
11 // Loop example
12 let sen = "Hi everybuddy welcome to javascript tutorial ";
13 let count = 0;
14 let words = sen .split (" ");
15
16 for( let w of words) {
17     count ++;
18 }
19 console .log (" Word count: " + count);
```

Assignment 12: JavaScript Conditional Statement

By: Atherv Telkar

Enter Your Marks: 299 ◇ [Check Result]

Grade A – Excellent Work, Atherv Telkar!

Assignment 13: JS Arrays

Code

```
1           =
2
3
4 let num = [10, 20, 30, 40, 50];
5 num.push(60);        // Add  at end
6 num.unshift(5);  // Add at  start
7 num.pop();           // Remove  last
8 num.shift();         // Remove  first
9
10
11
12                                m >=
13
14
15        and
16                     =        map((m) => m + 5);
17                                 =>       + m, O);
18
```

Assignment 13: JavaScript Array

Submitted by: Amey Telkar & Atherv Telkar

Array Operations

Student Marks Management System

```
    Inspector   Console   Debugger   Network   {} Style Editor
    Filter Output
Initial Array:  Array(4) [ "Amey", "Atherv", "Riya", "John" ]
After push:  Array(4) [ "Amey", "Atherv", "Riya", "John" ]
After unshift:  Array(4) [ "Amey", "Atherv", "Riya", "John" ]
After pop:  Array(4) [ "Amey", "Atherv", "Riya", "John" ]
After shift:  Array(4) [ "Amey", "Atherv", "Riya", "John" ]
Traversing Array:
Amey
Atherv
Riya
John
Original Marks:  Array(7) [ 95, 88, 76, 67, 50, 45, 32 ]
Sorted Marks:  Array(7) [ 95, 88, 76, 67, 50, 45, 32 ]
Reversed Marks:  Array(7) [ 95, 88, 76, 67, 50, 45, 32 ]
Top 3 Marks:  Array(3) [ 95, 88, 76 ]
Marks >= 60 (Passed):  Array(4) [ 95, 88, 76, 67 ]
Marks after +5 Bonus:  Array(7) [ 100, 93, 81, 72, 55, 50, 37 ]
Total Marks: 453
Average Marks: 64.71428571428571
```

Assignment 14 – JavaScript Objects

Submitted by: Amey Telkar & Atherv Telkar

Check browser console (Inspect → Console) to view results.

```
    Inspector   Console   Debugger   Network   {
    Filter Output
grade : Not Assigned
totalMarks : function(){
        return this.marks.reduce((a,b)=>a+b,0);
    }
Students Array:  Array [ {…}, {…} ]
Atherv's Computer Marks: 88
Name: Amey Telkar
Roll No: 1
Marks:  Array(3) [ 85, 90, 78 ]
Total: 253
Average: 84.33333333333333
Grade: A
------------------------
Name: Atherv Telkar
Roll No: 2
Marks:  Array(3) [ 80, 75, 88 ]
Total: 243
Average: 81
Grade: A
------------------------
Live reload enabled.
```

Assignment 14: JS Functions & Objects

Code (Grading System)

```
1  let marks = [90, 43, 56, 54];
2  let total = 0;
3  for( let i=0; i< marks. length ; i++) {
4      total += marks[ i];
5  }
6  let avg = total / marks. length ;
7
8  if( avg >= 90) console .log (" Grade : A+");
9  else if( avg >= 70) console .log (" Grade : A");
10 else if( avg >= 50) console .log (" Grade : B");
11 else console . log (" Fail ");
```

Code (Objects)

```
1  let student = {
2      id: 1234 ,
3      name : "Omkar",
4      city: "Pune ",
5      details: {
6          branch : "CSE",
7          year: 2024
8      }
9  };
10 console .log( student. name );
11
12 // Array of Objects
13 let students = [
14     { name : "Ram ", marks: 90 },
15     { name : "Shyam ", marks: 85 }
16 ];
17 students. forEach (s => console .log( s. name + " secured " + s. marks));
```

Assignment 15: DOM Manipulation

Code

```html
1  <h1 id="mainTitle">Welcome to DOM</h1>
2  <p class="info">This is assignment 15.</p>
3  <button id="changeBtn">Change Text</button>
4  <input type="text" id="userInput">
5  <label id="displayLabel"></label>
6
7  <script>
8      let title = document.getElementById("mainTitle");
9      console.log(title.innerText);
10
11     //                  Event                    Listener
12     document.getElementById("changeBtn").addEventListener("click",
       function () {
13         title.innerText = "DOM Manipulated Value";
14         title.style.color = "blue";
15     });
16
17     // Reading Input
18     let input = document.getElementById("userInput");
19     input.addEventListener("input", function () {
20         document.getElementById("displayLabel").innerText = input.value
       ;
21     });
22 </script>
```

Output

[Insert Screenshot of DOM Manipulation Here]

DOM Manipulated Successfully!

Submitted by: Amey Telkar & Atherv Telkar

Label changed successfully!

Change Title & Label

Paragraph 1

Paragraph 2

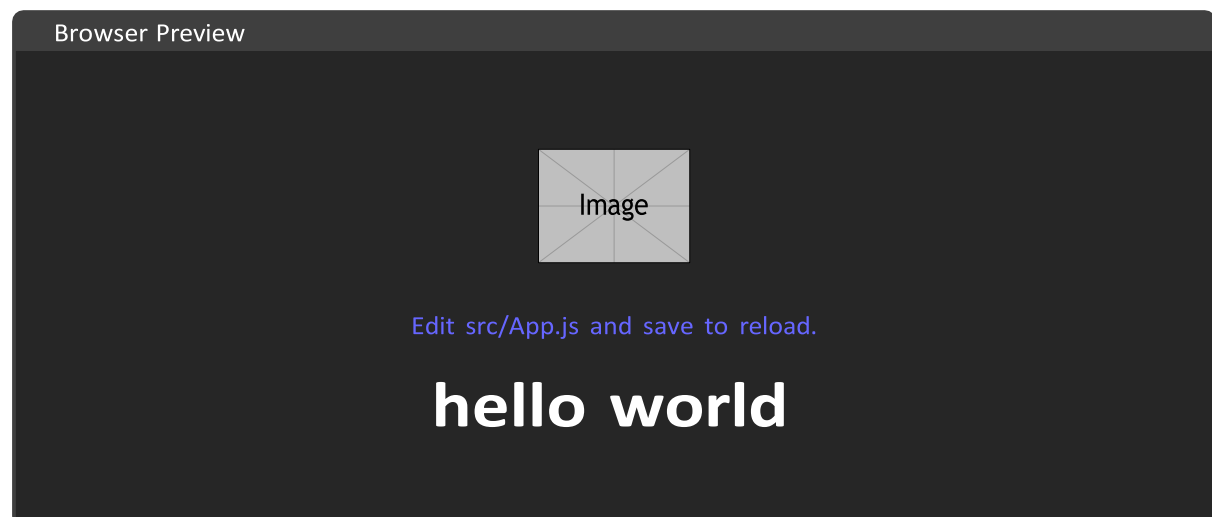- List Item 1
- List Item 2

React JS Components and Concepts

1 Hello World Component

```
1  import logo from './logo.svg';
2  import './App.css';
3
4  import { useState } from "react"
5
6  function App() {
7    return (
8      <div className="App">
9        <header className="App-header">
10          <img src={logo} className="App-logo" alt="logo" />
11        <p>
12            Edit <code>src/App.js</code> and save to reload.
13        </p>
14        <a>
15          <h1> hello world </h1>
16        </a>
17      </header>
18    </div>
19  );
20 }
```

Listing 1: App.js

Output Rendering

2 Student Details using Props

```
1  import './App.css';
2  function Student(props){
3     return (
4       <div>
5         <h3>Name: {props.name}</h3>
6         <p>Course: {props.course}</p>
7       </div>
8     );
9  }
10
11  export default function App(){
12     return(
13       <div>
14         <Student  name="Atherv"  course="MCA"/>
15         <Student  name="Amey  Telkar"  course="MCA"/>
16       </div>
17     );
18  }
```

Listing 2: App.js - Props Example

Output Rendering

Browser Preview

Name: Atherv
Course: MCA

Name: Amey Telkar
Course: MCA

3 List Rendering and State

3.1 Counter App (useState)

```
1  import { useState } from 'react';
2  import './App.css'
3
4  export default function App(){
5    const [count,setCount]=useState(0);
6    return (
7      <>
8        <h3>Count: {count}</h3>
9        <button onClick={()=>setCount(count+1)}> Plus </button>
10       <button onClick={()=>setCount(count-1)}> Minus </button>
11     </>
12   );
13 }
```

Listing 3: Counter Implementation

Output

3.2 List Rendering using map()

```
1  import {useState} from "react";
2  export default function App(){
3
4  const students=["amey","atherv","abhi"];
5
6  return(
7  <ul>
8    {students.map((name,index)=>(
9      <li key={index}>{name}</li>
10   ))}
11 </ul>
12 );
13 }
```

Listing 4: List Rendering

Output

4 Toggle Message (Conditional Rendering)

```
1  //Toggle Message using useState (Conditional Recording)
2
3  import { useState } from "react";
4
5  export default function App(){
6    const [show, setShow]=useState(true);
7    return (
8      <>
9        <button  onClick ={() =>setShow (!show )}>Toggle </button >
10       {show && <h3>Welcome to react !</h3>}
11     </>
12   );
13 }
```

Listing 5: Conditional Rendering with Operator

Output Rendering

Browser Preview

Toggle

Welcome to react !