

## VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS

Grade **A++** Accredited Institution by NAAC

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;  
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

### SCHOOL OF ENGINEERING & TECHNOLOGY

#### B. Tech Programme: B. Tech AI-ML (A)

#### Course Title: Web Programming Lab

#### Course Code: AIML - 259

**Submitted To:**

**Ms. Anshula Gupta**  
**Assistant Professor**

**Submitted By:**

**Name: Kunsh Sabharwal**  
**Enrolment No: 01117711623**



## **VISION OF INSTITUTE**

To be an educational institute that empowers the field of engineering to build a sustainable future by providing quality education with innovative practices that supports people, planet and profit.

## **MISSION OF INSTITUTE**

To groom the future engineers by providing value-based education and awakening students' curiosity, nurturing creativity and building capabilities to enable them to make significant contributions to the world.



**VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS**

**Grade A++ Accredited Institution by NAAC**

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

**SCHOOL OF ENGINEERING & TECHNOLOGY**



**VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS**

**Grade A++ Accredited Institution by NAAC**

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

**SCHOOL OF ENGINEERING & TECHNOLOGY**



**VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS**

**Grade A++ Accredited Institution by NAAC**

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;  
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

**SCHOOL OF ENGINEERING & TECHNOLOGY**



VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS

**Grade A++ Accredited Institution by NAAC**

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

SCHOOL OF ENGINEERING & TECHNOLOGY

## INDEX



VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS

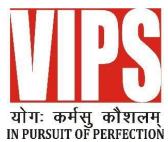
Grade A++ Accredited Institution by NAAC

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

## SCHOOL OF ENGINEERING & TECHNOLOGY



VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS

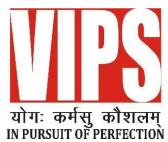
Grade A++ Accredited Institution by NAAC

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

## SCHOOL OF ENGINEERING & TECHNOLOGY



VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS

**Grade A++ Accredited Institution by NAAC**

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

SCHOOL OF ENGINEERING & TECHNOLOGY



IN PURSUIT OF PERFECTION

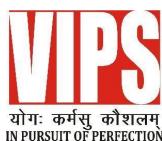
Grade A++ Accredited Institution by NAAC

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;

Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

SCHOOL OF ENGINEERING & TECHNOLOGY



IN PURSUIT OF PERFECTION

Grade A++ Accredited Institution by NAAC

NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;  
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE

An ISO 9001:2015 Certified Institution

## SCHOOL OF ENGINEERING & TECHNOLOGY

## **EXPERIMENT 1**

**Problem statement:** Create a web page that covers your CV using various HTML tags (UL, OL, Table etc.)

**Theory:**

## Source Code:

```

④ Resume_Kunsh_Sabharwal.html ✎ ...
```

```

1 <html>
2   <head>
3     <title>Kunsh Sabharwal Resume</title>
4   </head>
5   <body>
6     <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
7     <h1><div style="background-color: #rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="center">RESUME</div></p></h1>
8     <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
9     
11  <br>
12  <h2><p align="left">Kunsh Sabharwal</p></h2>
13  <h2><p align="left">Student @ VIPS-TC</p></h2>
14  <h2><p align="left">Pursuing B.Tech AI-ML</p></h2>
15  <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
16  <h1><div style="background-color: #rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="left">Career Objective</div></p></h1>
17  <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
18  <p align="left">First-year Bachelor of Technology student specializing in Artificial Intelligence and Machine Learning at
19  Vivekananda Institute of Professional Studies (Technical Campus), New Delhi, India under the aegis of Guru Gobind Singh Indraprastha University (GGSIPU).<br/>
20  Passionate about the newest innovations and trends in technologies with interests in Artificial Intelligence, Machine Learning and Data Science.<br/>
21  Equipped with knowledge of C, C++, and Python along with numerous skills including Leadership, Teamwork and Communication.<br/>
22  A strong academics background with a firm and confident attitude instilled with an integral moral system.<br/>
23  Enthusiastic and dedicated to master machine learning algorithms, neural networks, and data analysis with other cutting-edge technologies.<br/>
24  Passionate and eager to leverage technology to solve real-world problems and innovate new solutions.<br/>
25  Driven by curiosity and committed to continuous learning.<br/>
26  Always ready to learn something new.<br/></p>
27  <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
28  <h1><div style="background-color: #rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="left">Academic Qualifications</div></p></h1>
29  <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
30  <table border="5px" width="1000" align="left">
31    <tr>
32      <th>S.No.</th>
33      <th>Institution Name</th>
34      <th>Board/University</th>
35      <th>Class/Semester</th>
```

```
<tr><td>GGSIPU</td>
      <th>Class/Semester</th>
      <th>Year of Passing</th>
      <th>Percentage/CGPA</th>
    </tr>
    <tr>
      <td>1.</td>
      <td>New Era Public School</td>
      <td>CBSE</td>
      <td>10th</td>
      <td>2021</td>
      <td>94.6%</td>
    </tr>
    <tr>
      <td>2.</td>
      <td>New Era Public School</td>
      <td>CBSE</td>
      <td>12th</td>
      <td>2023</td>
      <td>94.8%</td>
    </tr>
    <tr>
      <td>3.</td>
      <td>VIPS-TC</td>
      <td>GGSIPU</td>
      <td>1st</td>
      <td>2024</td>
      <td>8.9/10</td>
    </tr>
    <tr>
      <td>4.</td>
      <td>VIPS-TC</td>
      <td>GGSIPU</td>
      <td>2nd</td>
      <td>2024</td>
      <td>8.9/10</td>
```

```

① Resume_Kunsh_Sabharwal.html X
68     <td>2024</td>
69     <td>8.9/10</td>
70   </tr>
71 </table>
72 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
73 <h1><div style="background-color:#rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="left">Technical Skills</div></p></h1>
74 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
75 <ul>
76   <li><p align="left"><b>Python</b></p></li>
77   <li><p align="left"><b>C</b></p></li>
78   <li><p align="left"><b>MySQL</b></p></li>
79   <li><p align="left"><b>HTML & CSS</b></p></li>
80 </ul>
81 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
82 <h1><div style="background-color:#rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="left">Achievements</div></p></h1>
83 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
84 <ul>
85   <li><p align="left">Participated in International Conference on Advanced Materials for Sustainable Innovation (IC-AMSI) 2024 in VIPS-TC,
86   | publishing a Research Paper on "Recent Solutions : Combating Climate Change and its Impacts".</p></li>
87   <li><p align="left">Created Python Project, connecting MySQL & Python, titled "Sneaker Dream".</p></li>
88   <li><p align="left">Created Python Project to display various popular codes and concepts in the form of an Olympiad Code Revision Booklet.</p></li>
89 </ul>
90 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
91 <h1><div style="background-color:#rgba(2, 64, 127, 0.859);color:#white; padding :5px"><p align="left">Contact Information</div></p></h1>
92 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
93 <ul>
94   <li><p align="left"><b>Mobile Number:- </b>99900159801</p></li>
95   <li><p align="left"><b>Gmail:-</b>sabharwalkunsh05@gmail.com</p></li>
96   <li><p align="left"><b>LinkedIn:</b> https://in.linkedin.com/in/kunsh-sabharwal-b133ab277?trk=people-guest\_people\_search-card</a></p></li>
97 </ul>
98 <hr style="width:100%; height:3px;background-color: #rgb(0, 183, 255);">
99 </body>
100 </html>

```

## Output:

The screenshot shows a web browser window with the title "Kunsh Sabharwal Resume". The address bar displays the file path: file:///C:/Users/DeLL/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/Resume\_Kunsh\_Sabharwal.html. The page content is as follows:

**RESUME**

**Kunsh Sabharwal**

**Student @ VIPS-TC**

**Pursuing B.Tech AI-ML**

**Career Objective**

First-year Bachelor of Technology student specializing in Artificial Intelligence and Machine Learning at Vivekananda Institute of Professional Studies (Technical Campus), New Delhi, India under the aegis of Guru Gobind Singh Indraprastha University (GGGIPU). Passionate about the newest innovations and trends in technologies with interests in Artificial Intelligence, Machine Learning and Data Science.

# KUNSH SABHARWAL

Kunsh Sabharwal Resume +

File file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/Resume\_Kunsh\_Sabharwal.html

For quick access, place your bookmarks here on the bookmarks bar. [Import bookmarks now...](#)

First-year Bachelor of Technology student specializing in Artificial Intelligence and Machine Learning at Vivekananda Institute of Professional Studies (Technical Campus), New Delhi, India under the aegis of Guru Gobind Singh Indraprastha University (GGSPU).  
Passionate about the newest innovations and trends in technologies with interests in Artificial Intelligence, Machine Learning and Data Science.  
Equipped with knowledge of C, C++, and Python along with numerous skills including Leadership, Teamwork and Communication.  
A strong academic background with a firm and confident attitude instilled with an integral moral system.  
Enthusiastic and dedicated to master machine learning algorithms, neural networks, and data analysis with other cutting-edge technologies.  
Passionate and eager to leverage technology to solve real-world problems and innovate new solutions.  
Driven by curiosity and committed to continuous learning.  
Always ready to learn something new.

## Academic Qualifications

S.No.	Institution Name	Board/University	Class/Semester	Year of Passing	Percentage/CGPA
1.	New Era Public School	CBSE	10th	2021	94.6%
2.	New Era Public School	CBSE	12th	2023	94.8%
3.	VIPS-TC	GGSPU	1st	2024	8.9/10
4.	VIPS-TC	GGSPU	2nd	2024	8.9/10

## Technical Skills

Kunsh Sabharwal Resume +

File file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/Resume\_Kunsh\_Sabharwal.html

For quick access, place your bookmarks here on the bookmarks bar. [Import bookmarks now...](#)

## Technical Skills

- Python
- C
- MySQL
- HTML & CSS

## Achievements

- Participated in International Conference on Advanced Materials for Sustainable Innovation (IC-AMSI) 2024 in VIPS-TC, publishing a Research Paper on "Recent Solutions : Combating Climate Change and its Impacts".
- Created Python Project, connecting MySQL & Python, titled "Sneaker Dream".
- Created Python Project to display various popular codes and concepts in the form of an Olympiad Code Revision Booklet.

The screenshot shows a web browser window with a blue header bar containing the title "KUNSH SABHARWAL". Below the header is a dark blue navigation bar with icons for file operations like back, forward, and search, along with a "File" menu item. The main content area has a white background with a blue header section. The first section is titled "Achievements" in white text. Below it is a horizontal line, followed by a bulleted list of accomplishments:

- Participated in International Conference on Advanced Materials for Sustainable Innovation (IC-AMSI) 2024 in VIPS-TC, publishing a Research Paper on "Recent Solutions : Combating Climate Change and its Impacts".
- Created Python Project, connecting MySQL & Python, titled "Sneaker Dream".
- Created Python Project to display various popular codes and concepts in the form of an Olympiad Code Revision Booklet.

Below this is another blue header section titled "Contact Information" in white text. A horizontal line follows, and then a bulleted list of contact details:

- Mobile Number:-** 9990159801
- Gmail:-** sabharwalkunsh05@gmail.com
- LinkedIn:-** [https://in.linkedin.com/in/kunsh-sabharwal-b133ab277?trk=people-guest\\_people\\_search-card](https://in.linkedin.com/in/kunsh-sabharwal-b133ab277?trk=people-guest_people_search-card)

## Learning Outcome:

## **EXPERIMENT 2**

**Problem statement:** Create a webpage that displays brief details of various programming language using various types of CSS.

**Theory:**

## Source Code:

The screenshot shows a code editor window with the following details:

- Title Bar:** Various Programming Languages\_Kunsh Sabharwal.html
- Code Content:**

```
1 <html>
2   <head>
3     <title>Various Programming Languages</title>
4   </head>
5   <style>
6     #hd1{
7       background-color: #ffb6c6;
8       width: 100%;
9       height: 80px;
10      color: black;
11    }
12    .hd2{
13      background-color: #f0889d;
14      width: 100%;
15      height: 80px;
16      color: black;
17    }
18    * {
19      margin: 0;
20      padding: 0;
21    }
22    .navbar {
23      display: flex;
24      align-items: center;
25      justify-content: center;
26      position: sticky;
27      top: 0;
28      padding: 15px;
29      cursor: pointer;
30    }
31    .background {
32      background: black;
33      background-blend-mode: darken;
34      background-size: cover;
35    }
```
- Status Bar:** Run and Debug (Ctrl+Shift+D)

## &lt;&gt; Various Programming Languages\_Kunsh Sabharwal.html X

```
31     .background {
32         background: black;
33         background-blend-mode: darken;
34         background-size: cover;
35     }
36     .nav-list {
37         width: 70%;
38         display: flex;
39         align-items: center;
40         gap: 20px;
41         list-style: none;
42     }
43     .logo {
44         display: flex;
45         justify-content: center;
46         align-items: center;
47     }
48     .logo img {
49         width: 180px;
50         border-radius: 50px;
51     }
52     .nav-list li {
53         list-style: none;
54         padding: 26px 30px;
55         padding: 10px;
56     }
57     .nav-list li a {
58         text-decoration: none;
59         color: white;
60     }
61     .nav-list li a:hover {
62         color: grey;
63     }
64     .rightnav {
65         width: 30%;
```

```

 66 |         text-align: right;
67 |
68 |     </style>
69 |     <body style="background-color: #EBE9E1;">
70 |         <nav class="navbar background">
71 |             <div class="logo">
72 |                 <img src=
73 | "https://media.geeksforgeeks.org/gfg-gg-logo.svg"
74 |                     style="height: 30px;" alt="Logo">
75 |             </div>
76 |             <ul class="nav-list">
77 |                 <li><a href="https://www.geeksforgeeks.org/c-programming-language/">C Programming</a></li>
78 |                 <li><a href="https://www.geeksforgeeks.org/c-plus-plus/">C++ Programming</a></li>
79 |                 <li><a href="https://www.geeksforgeeks.org/java/">Java Programming</a></li>
80 |                 <li><a href="https://www.geeksforgeeks.org/python-programming-tutorial/">Python Programming</a></li>
81 |                 <li><a href="https://www.geeksforgeeks.org/web-development/">Web Technology</a></li>
82 |                 <li><a href="https://www.geeksforgeeks.org/javascript/">JavaScript Programming</a></li>
83 |             </ul>
84 |             <div class="rightnav">
85 |                 <input type="text" name="search" id="search">
86 |                 <button class="btn btn-sm">Search</button>
87 |             </div>
88 |         </nav>
89 |         <h1 style="background-color: #fad788; width: 100%; height: 115px; text-align: center; color: black; border: none;"><br/>
90 |             <u><b>Various Programming Languages</b></u></h1>
91 |         <p><br/>In today's era where computers are the basic necessity of every company and industry to grow. The need for programmers and software developers is also increasing. As you know that different programming languages have their own specialty and features. So, In order to become a skilled programmer, you should have a basic understanding of the different types of Programming Languages available. Some major and the most popular programming languages include the following:</p>
92 |         <br/>
93 |         <hr style="width:100%; height:5px;background-color: #AFD275;">

```

```

 101 |         <br/>
102 |
103 |         <br/>
104 |         <center><h2 id="hd1">
105 |             
106 |             
107 |             <br><br/>1. C </b></h2></center>
108 |         <p><br/>The C programming language is a procedural and general-purpose language that provides low-level access to system memory. A program written in C must be run through a C compiler to convert it into an executable that a computer can run. Many versions of Unix-based operating systems (OSes) are written in C and it has been standardized as part of the Portable Operating System Interface (POSIX). Today, the C programming language runs on many different hardware platforms and OSes such as Microsoft and Linux. Structured. It offers a structured programming approach for breaking down problems into smaller modules or functions that are easy to understand and modify. Portable - C is machine-independent and C programs can be executed on different machines.<br/></p>
109 |         <br/>
110 |         <ul>
111 |             <li><b>Mid-level programming language -</b> It's a mid-level language that supports the features of both a low-level and a high-level language. </li>
112 |             <li><b>Rich library - </b>It offers numerous built-in library functions that expedite the development process.</li>
113 |             <li><b>Dynamic memory allocation -</b> C supports the dynamic memory allocation feature, which can be used to free the allocated memory at any time by calling the free() function.</li>
114 |             <li><b>Speed - </b>It's a compiler-based language, which makes the compilation and execution of code faster. Since only essential and required features are included in C, it saves processing power and improves speed.</li>
115 |             <li><b>Pointers - </b>C uses pointers, which improve performance by enabling direct interaction with the system memory.</li>
116 |             <li><b>Recursion -</b> C enables developers to backtrack by providing code reusability for every function.</li>
117 |             <li><b>Extensible - </b>A C program can be easily extended. If code is already written, new features and functionalities can be added to it with minor alterations.</li>
118 |         </ul>
119 |         <br/>
120 |         <hr style="width:100%; height:5px;background-color: #AFD275;">
121 |         <br/>
122 |         <center><h2 id="hd1">
123 |             
124 |             
125 |             <br><br/>2. C++ </b></h2></center>
126 |         <p><br/>C++ is a general-purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm.

```

```

<div style="background-color: #e0f2e0; padding: 10px; border: 1px solid black; border-radius: 5px; margin-bottom: 10px;">
    <h2>C++</h2>
    <p>C++ is an imperative and a compiled language.<br/></p>
    <p>C++ is a high-level, general-purpose programming language designed for system and application programming.<br/>
    It was developed by Bjarne Stroustrup at Bell Labs in 1983 as an extension of the C programming language. <br/>
    C++ is an object-oriented, multi-paradigm language that supports procedural, functional, and generic programming styles.<br/>
    One of the key features of C++ is its ability to support low-level, system-level programming, making it suitable
    for developing operating systems, device drivers, and other system software.
    At the same time, C++ also provides a rich set of libraries and features for high-level application programming,
    making it a popular choice for developing desktop applications, video games, and other complex applications.<br/>
    C++ has a large, active community of developers and users, and a wealth of resources and tools available for
    learning and using the language.
    <br/>
    <br/>Some of the key features of C++ include:
    <br/>
    <ul></ul>
        <li><b>Object-Oriented Programming:</b> C++ supports object-oriented programming, allowing developers to create classes and
            objects and to define methods and properties for these objects.<br/></li>
        <li><b>Templates:</b> C++ templates allow developers to write generic code that can work with any data type,
            making it easier to write reusable and flexible code.<br/></li>
        <li><b>Standard Template Library (STL):</b> The STL provides a wide range of containers and algorithms for
            working with data, making it easier to write efficient and effective code.<br/></li>
        <li><b>Exception Handling:</b> C++ provides robust exception handling capabilities, making it easier to
            write code that can handle errors and unexpected situations.<br/></li>
    </ul>
    </p>
    <br/>
    <hr style="width:100%; height:5px;background-color: #AFD275;">
    <br/>
    
    
    <center><h2 class="hd2"><b>3. Java</b></h2></center>
    <p><br/>Java is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible.
        It is intended to let application developers write once, and run anywhere (WORA),
        meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.
        Java was first released in 1995 and is widely used for developing applications for desktop, web, and mobile devices.
        Java is known for its simplicity, robustness, and security features, making it a popular choice for enterprise-level applications.
    </p>

```

```

<div style="background-color: #e0f2e0; padding: 10px; border: 1px solid black; border-radius: 5px; margin-bottom: 10px;">
    <h2>Java</h2>
    <p>Java was developed by James Gosling at Sun Microsystems Inc in May 1995 and later acquired by Oracle Corporation.
        It is a simple programming language.
        Java makes writing, compiling, and debugging programming easy.
        It helps to create reusable code and modular programs.
        Java is a class-based, object-oriented programming language and is designed to have as few implementation dependencies as possible.
        A general-purpose programming language made for developers to write once run anywhere that is compiled Java code can run on all platforms that support Java.
        Java applications are compiled to byte code that can run on any Java Virtual Machine.
        The syntax of Java is similar to C/C++.<br/>
    <br/>
    Java has various features and advantages:-</p>
    <ul></ul>
        <li><b>Platform Independent:</b></li>
        <li><b>Object-Oriented Programming:</b></li>
        <li><b>Simplicity:</b></li>
        <li><b>Robustness:</b></li>
        <li><b>Security:</b></li>
        <li><b>MultiThreading:</b></li>
        <li><b>Portability:</b></li>
    </ul>
    <br/>
    <hr style="width:100%; height:5px;background-color: #AFD275;">
    <br/>
    
    
    <center><h2 id="hd1"><b>4. Python</b></h2></center>
    <p><br/>Python is a programming language that is interpreted, object-oriented, and considered to be high-level too.
        Python is one of the easiest yet most useful programming languages which is widely used in the software industry.
        People use Python for Competitive Programming, Web Development, and creating software.
        Due to its easiest syntax, it is recommended for beginners who are new to the software engineering field.
        Its demand is growing at a very rapid pace due to its vast use cases in Modern Technological fields like Data Science, Machine learning, and Automation Tasks.
        For many years now, it has been ranked among the top Programming languages.<br/>
    <br/>
    Python has plenty of features that make it the most demanding and popular. Let's read about a few of the best features that Python has:</p>
    <ul></ul>
        <li><b>Easy to read and understand:</b></li>

```

```

<li><b>Interpreted language</b></li>
<li><b>Object-oriented programming language</b></li>
<li><b>Free and open-source</b></li>
<li><b>Versatile and Extensible</b></li>
<li><b>Multi-platform</b></li>
<li><b>Hundreds of libraries and frameworks</b></li>
<li><b>Flexible, supports GUI</b></li>
<li><b>Dynamically typed</b></li>
<li><b>Huge and active community</b></li>
</ul>
<br/>
<hr style="width:100%; height:5px; background-color: #AFD275;">
<br/>


<center><h2 class="hd2"><b><br/>5. HTML</b></h2></center>
<p><br/>HTML stands for HyperText Markup Language and it is used to create webpages.
It uses HTML tags and attributes to describe the structure and formatting of a web page.
HTML consists of various elements, that are responsible for telling search engines how to display page content.
For example, headings, lists, images, links, and more.<br/>
<br/>
Some Features of HTML include:-</p>
<ul></ul>
<li><b>It is easy to learn and easy to use.</b></li>
<li><b>It is platform-independent.</b></li>
<li><b>Images, videos, and audio can be added to a web page.</b></li>
<li><b>Hypertext can be added to the text.</b></li>
<li><b>It is a markup language.</b></li>
</ul>
<br/>
<hr style="width:100%; height:5px; background-color: #AFD275;">
<br/>


<center><h2 id="hd1"><b><br/>6. CSS</b></h2></center>

```

```

<p><br/>CSS stands for Cascading style sheets. It describes to the user how to display HTML elements on the screen in a proper format.
CSS is the language that is used to style HTML documents.
In simple words, cascading style sheets are a language used to simplify the process of making a webpage.
CSS is used to handle some parts of the webpage.
With the help of CSS, we can control the colour of text and style of fonts, and we can control the spacing between the paragraph and many more things.
CSS is easy to understand but provides strong control on the Html documents.
CSS is combined with HTML.<br/>
Some advantages of using CSS include:-</p>
<br/>
<ul></ul>
<li><b>Faster page speed:</b>It has a faster page speed than other code's page speeds.
With the help of the CSS rule, we can apply it to all occurrences of certain tags in HTML documents.</li>
<li><b>Better user experience:</b>CSS makes a webpage very attractive to the eyes.
Also, CSS makes it user-friendly.
When the button or text is in a proper format, it improves the user experience.</li>
<li><b>Quicker Development time:</b>With the help of CSS, we can specify the format and style the multiple pages into one code string.
In cascading style sheet, we can make a duplicate copy of several website pages.
If we make a webpage, it has the same formatting, looks, and feel,
so with the help of the CSS rule for one page, and it is sufficient for all the pages.</li>
<li><b>Easy Formatting changes:</b>In CSS, if we need to make changes in the format, it is very easy;
we only need to change the one-page format it will automatically apply to the other pages of CSS.
There is no need to correct individual pages in a CSS style sheet.
If we fix a CSS style sheet, it will automatically update the other CSS style sheet.</li>
<li><b>Compatibility:</b>Compatibility is very important in today's age.
If we create any webpage, it should be very responsive and user-friendly.
CSS is used with Html to make webpage design responsive.</li>
</ul>
<br/>
<hr style="width:100%; height:5px; background-color: #AFD275;">
<br/>


<center><h2 class="hd2"><b><br/>7. JavaScript</b></h2></center>
<p><br/>JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language.
It is also known as the scripting language for webpages.

```

```

< Various Programming Languages_Kunsh Sabharwal.html >
265      If we create any webpage, it should be very responsive and user-friendly.
266      CSS is used with Html to make webpage design responsive.</li>
267    </ul>
268    <br/>
269    <hr style="width:100%; height:5px;background-color: #AFD275;">
270    <br/>
271    
272    
273    <center><h2 class="hd2"><b>7. JavaScript</b></h2></center>
274    <p><br/>JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language.
275    It is also known as the scripting language for webpages.
276    It is well-known for the development of web pages, and many non-browser environments also use it.
277    JavaScript is a weakly typed language (dynamically typed).
278    JavaScript can be used for Client-side developments as well as Server-side developments.
279    JavaScript is both an imperative and declarative type of language.
280    JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of
281    language elements like operators, control structures, and statements.
282    <br/>
283    <br/>
284    Some features of JavaScript include:-</p>
285    <ul></ul>
286    <li>JavaScript was created in the first place for DOM manipulation.
287    |   Earlier websites were mostly static, after JS was created dynamic Web sites were made.</li>
288    <li>Functions in JS are objects.
289    |   They may have properties and methods just like other objects.
290    |   They can be passed as arguments in other functions.</li>
291    <li>Can handle date and time.</li>
292    <li>Performs Form Validation although the forms are created using HTML.</li>
293    <li>No compiler is needed.</li>
294    </ul>
295    <br/>
296    <hr style="width:100%; height:5px;background-color: #AFD275;">
297    <br/>
298  </body>
299 </html>

```

## Output:

Various Programming Languages

In today's era where computers are the basic necessity of every company and industry to grow. The need for programmers and software developers is also increasing. As you know that different programming languages have their own specialty and features. So, In order to become a skilled programmer, you should have a basic understanding of the different types of Programming Languages available. Some major and the most popular programming languages include the following:

**1. C**

The C programming language is a procedural and general-purpose language that provides low-level access to system memory. A program written in C must be run through a C compiler to convert it into an executable that a computer can run. Many versions of Unix-based operating systems (OSes) are written in C and it has been standardized as part of the Portable Operating System Interface (POSIX). Today, the C programming language runs on many different hardware platforms and OSes such as Microsoft and Linux. Structured. It offers a structured programming approach for breaking down problems into smaller modules or functions that are easy to understand and modify. Portable - C is machine-independent and C programs can be executed on different machines.

- Mid-level programming language - It's a mid-level language that supports the features of both a low-level and a high-level language.
- Rich library - It offers numerous built-in library functions that expedite the development process.
- Dynamic memory allocation - C supports the dynamic memory allocation feature, which can be used to free the allocated memory at any time by calling the free() function.
- Speed - It's a compiler-based language, which makes the compilation and execution of code faster. Since only essential and required features are included in C, it saves processing power and improves speed.
- Pointers - C uses pointers, which improve performance by enabling direct interaction with the system memory.
- Recursion - C enables developers to backtrack by providing code reusability for every function.

Various Programming Languages

C Programming C++ Programming Java Programming Python Programming Web Technology JavaScript Programming

Search

- Recursion - C enables developers to backtrack by providing code reusability for every function.
- Extensible - A C program can be easily extended. If code is already written, new features and functionalities can be added to it with minor alterations.

## 2. C++

C++ is a general-purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm. It is an imperative and a compiled language. C++ is a high-level, general-purpose programming language designed for system and application programming. It was developed by Bjarne Stroustrup at Bell Labs in 1983 as an extension of the C programming language. C++ is an object-oriented, multi-paradigm language that supports procedural, functional, and generic programming styles. One of the key features of C++ is its ability to support low-level, system-level programming, making it suitable for developing operating systems, device drivers, and other system software. At the same time, C++ also provides a rich set of libraries and features for high-level application programming, making it a popular choice for developing desktop applications, video games, and other complex applications. C++ has a large, active community of developers and users, and a wealth of resources and tools available for learning and using the language.

Some of the key features of C++ include:

- Object-Oriented Programming:** C++ supports object-oriented programming, allowing developers to create classes and objects and to define methods and properties for these objects.
- Templates:** C++ templates allow developers to write generic code that can work with any data type, making it easier to write reusable and flexible code.
- Standard Template Library (STL):** The STL provides a wide range of containers and algorithms for working with data, making it easier to write efficient and effective code.
- Exception Handling:** C++ provides robust exception handling capabilities, making it easier to write code that can handle errors and unexpected situations.

## 3. Java

Java is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is intended to let application developers write once, and run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java was first released in 1995 and is widely used for developing applications for desktop, web, and mobile devices. Java is known for its simplicity, robustness, and security features, making it a popular choice for enterprise-level applications. Java was developed by James Gosling at Sun Microsystems Inc in May 1995 and later acquired by Oracle Corporation. It is a simple programming language. Java makes writing, compiling, and debugging programming easy. It helps to create reusable code and modular programs. Java is a class-based, object-oriented programming language and is designed to have as few implementation dependencies as possible. A general-purpose programming language made for developers to write once run anywhere that is compiled Java code can run on all platforms that support Java. Java applications are compiled to byte code that can run on any Java Virtual Machine. The syntax of Java is similar to C/C++.

Java has various features and advantages:-

- Platform Independent
- Object-Oriented Programming
- Simplicity
- Robustness
- Security
- MultiThreading
- Portability

Various Programming Languages

C Programming C++ Programming Java Programming Python Programming Web Technology JavaScript Programming

Search

## 4. Python

Python is a programming language that is interpreted, object-oriented, and considered to be high-level too. Python is one of the easiest yet most useful programming languages which is widely used in the software industry. People use Python for Competitive Programming, Web Development, and creating software. Due to its easiest syntax, it is recommended for beginners who are new to the software engineering field. Its demand is growing at a very rapid pace due to its vast use cases in Modern Technological fields like Data Science, Machine learning, and Automation Tasks. For many years now, it has been ranked among the top Programming languages.

Python has plenty of features that make it the most demanding and popular. Let's read about a few of the best features that Python has:

The screenshot shows a web browser window with a blue header bar containing the title "KUNSH SABHARWAL". Below the header is a navigation bar with links for "C Programming", "C++ Programming", "Java Programming", "Python Programming", "Web Technology", and "JavaScript Programming". A search bar is also present. The main content area displays a list of features for Python, followed by a section titled "5. HTML" with a pink header and icons for HTML and CSS.

Python has plenty of features that make it the most demanding and popular. Let's read about a few of the best features that Python has:

- Easy to read and understand
- Interpreted language
- Object-oriented programming language
- Free and open-source
- Versatile and Extensible
- Multi-platform
- Hundreds of libraries and frameworks
- Flexible, supports GUI
- Dynamically typed
- Huge and active community

## 5. HTML

HTML stands for HyperText Markup Language and it is used to create webpages. It uses HTML tags and attributes to describe the structure and formatting of a web page. HTML consists of various elements, that are responsible for telling search engines how to display page content. For example, headings, lists, images, links, and more.

Some Features of HTML include:-

- It is easy to learn and easy to use.
- It is platform-independent.
- Images, videos, and audio can be added to a web page.
- Hypertext can be added to the text.
- It is a markup language.

The screenshot shows a web browser window with a blue header bar containing the title "KUNSH SABHARWAL". Below the header is a navigation bar with links for "C Programming", "C++ Programming", "Java Programming", "Python Programming", "Web Technology", and "JavaScript Programming". A search bar is also present. The main content area displays a list of advantages of using CSS, followed by a section titled "6. CSS" with a pink header and icons for CSS and HTML.

CSS stands for Cascading style sheets. It describes to the user how to display HTML elements on the screen in a proper format. CSS is the language that is used to style HTML documents. In simple words, cascading style sheets are a language used to simplify the process of making a webpage. CSS is used to handle some parts of the webpage. With the help of CSS, we can control the colour of text and style fonts, and we can control the spacing between the paragraph and many more things. CSS is easy to understand but provides strong control on the Html documents. CSS is combined with HTML.

Some advantages of using CSS include:-

- **Faster page speed:**It has a faster page speed than other code's page speeds. With the help of the CSS rule, we can apply it to all occurrences of certain tags in HTML documents.
- **Better user experience:**CSS makes a webpage very attractive to the eyes. Also, CSS makes it user-friendly. When the button or text is in a proper format, it improves the user experience.
- **Quicker Development time:**With the help of CSS, we can specify the format and style the multiple pages into one code string. In cascading style sheet, we can make a duplicate copy of several website pages. If we make a webpage, it has the same formatting, looks, and feel, so with the help of the CSS rule for one page, and it is sufficient for all the pages.
- **Easy Formatting changes:**In CSS, if we need to make changes in the format, it is very easy; we only need to change the one-page format it will automatically apply to the other pages of CSS. There is no need to correct individual pages in a CSS style sheet. If we fix a CSS style sheet, it will automatically update the other CSS style sheet.
- **Compatibility:**Compatibility is very important in today's age. If we create any webpage, it should be very responsive and user-friendly. CSS is used with Html to make webpage design responsive.

The screenshot shows a web browser window with a blue header bar containing the title 'KUNSH SABHARWAL'. Below the header is a navigation bar with links for 'C Programming', 'C++ Programming', 'Java Programming', 'Python Programming', 'Web Technology', and 'JavaScript Programming'. A search bar is also present. The main content area displays text about the advantages of CSS, followed by a section titled '7. JavaScript' with a yellow and pink background. This section contains text about JavaScript's features and history.

Some advantages of using CSS include:-

- **Faster page speed:** It has a faster page speed than other code's page speeds. With the help of the CSS rule, we can apply it to all occurrences of certain tags in HTML documents.
- **Better user experience:** CSS makes a webpage very attractive to the eyes. Also, CSS makes it user-friendly. When the button or text is in a proper format, it improves the user experience.
- **Quicker Development time:** With the help of CSS, we can specify the format and style the multiple pages into one code string. In cascading style sheet, we can make a duplicate copy of several website pages. If we make a webpage, it has the same formatting, looks, and feel, so with the help of the CSS rule for one page, and it is sufficient for all the pages.
- **Easy Formatting changes:** In CSS, if we need to make changes in the format, it is very easy; we only need to change the one-page format it will automatically apply to the other pages of CSS. There is no need to correct individual pages in a CSS style sheet. If we fix a CSS style sheet, it will automatically update the other CSS style sheet.
- **Compatibility:** Compatibility is very important in today's age. If we create any webpage, it should be very responsive and user-friendly. CSS is used with Html to make webpage design responsive.

## 7. JavaScript

JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language. It is also known as the scripting language for webpages. It is well-known for the development of web pages, and many non-browser environments also use it. JavaScript is a weakly typed language (dynamically typed). JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript is both an imperative and declarative type of language. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.

Some features of JavaScript include:-

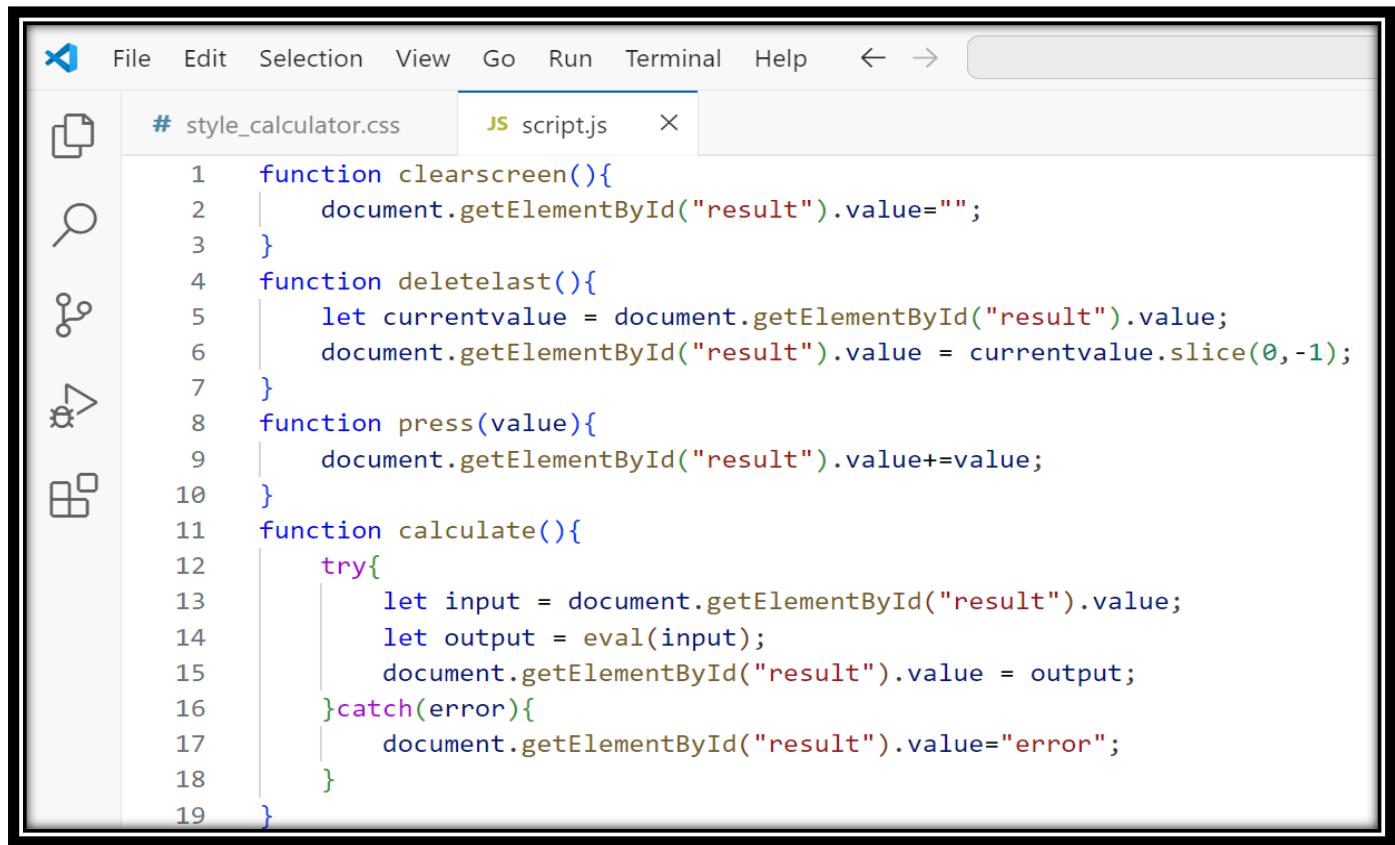
- JavaScript was created in the first place for DOM manipulation. Earlier websites were mostly static, after JS was created dynamic Web sites were made.
- Functions in JS are objects. They may have properties and methods just like other objects. They can be passed as arguments in other functions.
- Can handle date and time.
- Performs Form Validation although the forms are created using HTML.
- No compiler is needed.

## Learning Outcome:

## EXPERIMENT 3

**Problem statement:** Create a webpage using JavaScript and HTML to demonstrate Simple Calculator Application.

**Theory:**

**Source Code:****(a) JavaScript Screenshot:**

The screenshot shows a code editor interface with the following details:

- File Explorer:** On the left, there are icons for files, search, and other development tools.
- File Bar:** At the top, there are tabs for "style\_calculator.css" (highlighted in blue) and "script.js".
- Menu Bar:** Includes File, Edit, Selection, View, Go, Run, Terminal, Help, and navigation arrows.
- Code Area:** The main area contains the following JavaScript code:

```
1 function clearscreen(){
2     document.getElementById("result").value="";
3 }
4 function deletelast(){
5     let currentvalue = document.getElementById("result").value;
6     document.getElementById("result").value = currentvalue.slice(0,-1);
7 }
8 function press(value){
9     document.getElementById("result").value+=value;
10 }
11 function calculate(){
12     try{
13         let input = document.getElementById("result").value;
14         let output = eval(input);
15         document.getElementById("result").value = output;
16     }catch(error){
17         document.getElementById("result").value="error";
18     }
19 }
```

**(b) HTML screenshot:**

The screenshot shows a code editor interface with the following details:

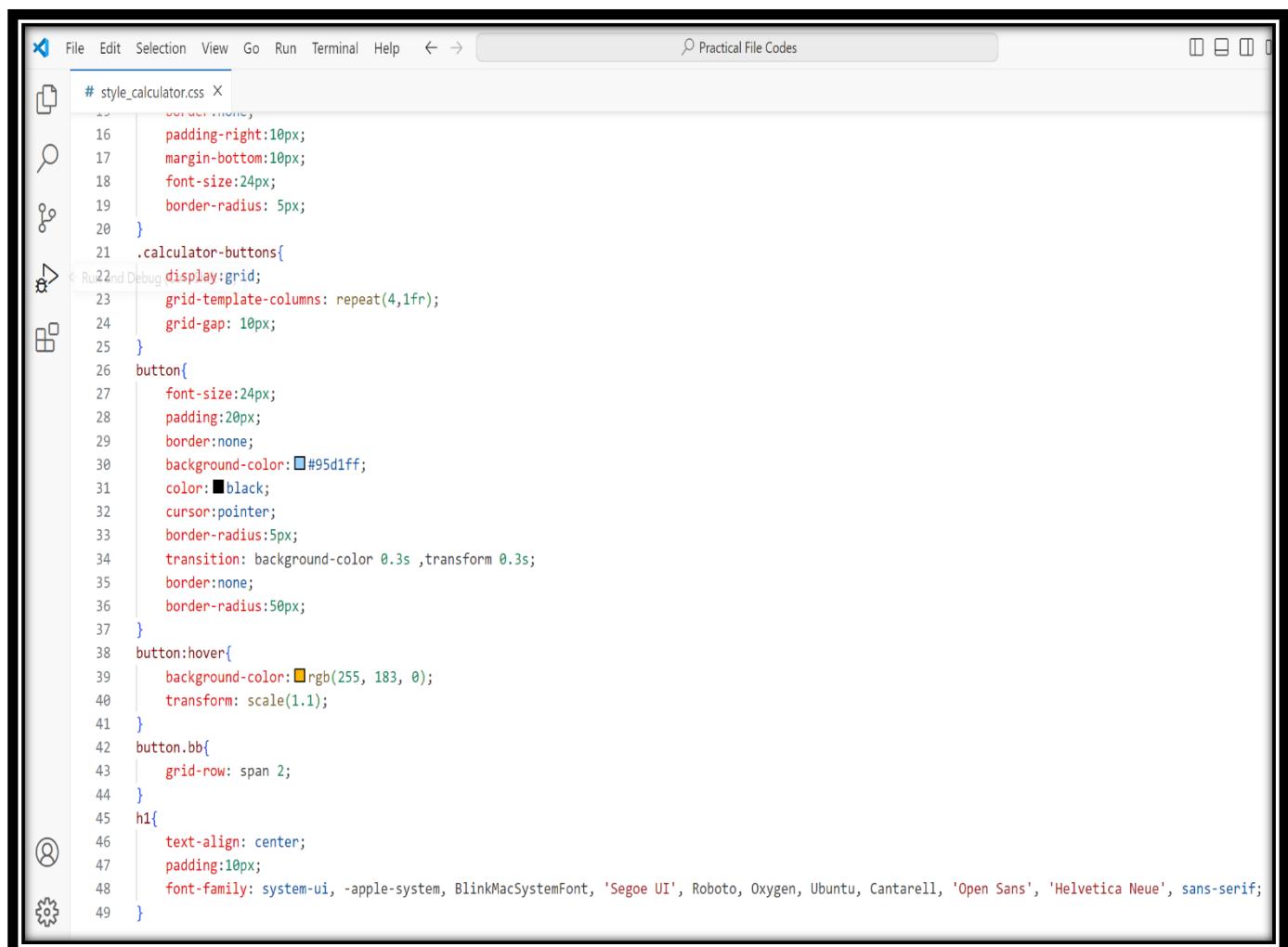
- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Practical File Co.
- Left Sidebar:** Includes icons for file operations (New, Open, Save, Find, Replace, Copy, Paste, Delete, Undo, Redo), a search icon, a refresh icon, and a settings gear icon.
- Tab Bar:** # style\_calculator.css, JS script.js, and the active tab, Calculator using JavaScript.html.
- Code Area:** The content of the Calculator using JavaScript.html file is displayed, consisting of 35 numbered lines of HTML and JavaScript code.
- Bottom Status Bar:** Shows navigation icons (Back, Forward, Home, Stop, Reload) and status indicators (0 0 0 0).

```
1 <html>
2 <head>
3     <meta charset="UTF-8">
4     <meta name="viewport" content="width=`, initial-scale=1.0">
5     <title>Calculator</title> <link rel="stylesheet" href="style_calculator.css">
6 </head>
7 <body>
8     <div class="calculator">
9         <h1>Calculator using JavaScript</h1>
10        <input type="text" id="result" class = "calculator_screen" value="" readonly/>
11        <div class="calculator-buttons">
12            <button onClick="clearscreen()">C</button>
13            <button onClick="deletelast()">DEL</button>
14            <button onClick="press('/')">/</button>
15            <button onClick="press('*')">x</button>
16            <button onClick="press('7')">7</button>
17            <button onClick="press('8')">8</button>
18            <button onClick="press('9')">9</button>
19            <button onClick="press('-')">-</button>
20            <button onClick="press('4')">4</button>
21            <button onClick="press('5')">5</button>
22            <button onClick="press('6')">6</button>
23            <button onClick="press('+')">+</button>
24            <button onClick="press('1')">1</button>
25            <button onClick="press('2')">2</button>
26            <button onClick="press('3')">3</button>
27            <button onClick="calculate()" class="bb">=</button>
28            <button onClick="press('0')">0</button>
29            <button onClick="press('.')">. </button>
30        </div>
31    </div>
32 </div>
33 <script src="script.js"></script>
34 </body>
35 </html>
```

## (c) CSS Screenshot:

The screenshot shows a code editor interface with a dark theme. On the left is a sidebar with various icons: a file folder, a magnifying glass, a gear, a person, and a settings gear. The main area displays a CSS file named "style\_calculator.css". The code defines styles for a body element, a .calculator class, a .calculator\_screen class, a .calculator-buttons class, and a button class.

```
# style_calculator.css X
1  body{
2      font-family:Arial, Helvetica, sans-serif;
3      display:flex;
4      align-items: center;
5      justify-content:center;
6  }
7  .calculator{
8      padding:20px;
9      background-color: #e0e0e0;
10 }
11 .calculator_screen{
12     height: 100px;
13     width: 400px;
14     text-align:right;
15     border:none;
16     padding-right:10px;
17     margin-bottom:10px;
18     font-size:24px;
19     border-radius: 5px;
20 }
21 .calculator-buttons{
22     display:grid;
23     grid-template-columns: repeat(4,1fr);
24     grid-gap: 10px;
25 }
26 button{
27     font-size:24px;
28     padding:20px;
29     border:none;
30     background-color: #95d1ff;
31     color: black;
32     cursor:pointer;
33     border-radius:5px;
34     transition: background-color 0.3s ,transform 0.3s;
35     border:none;
}
```

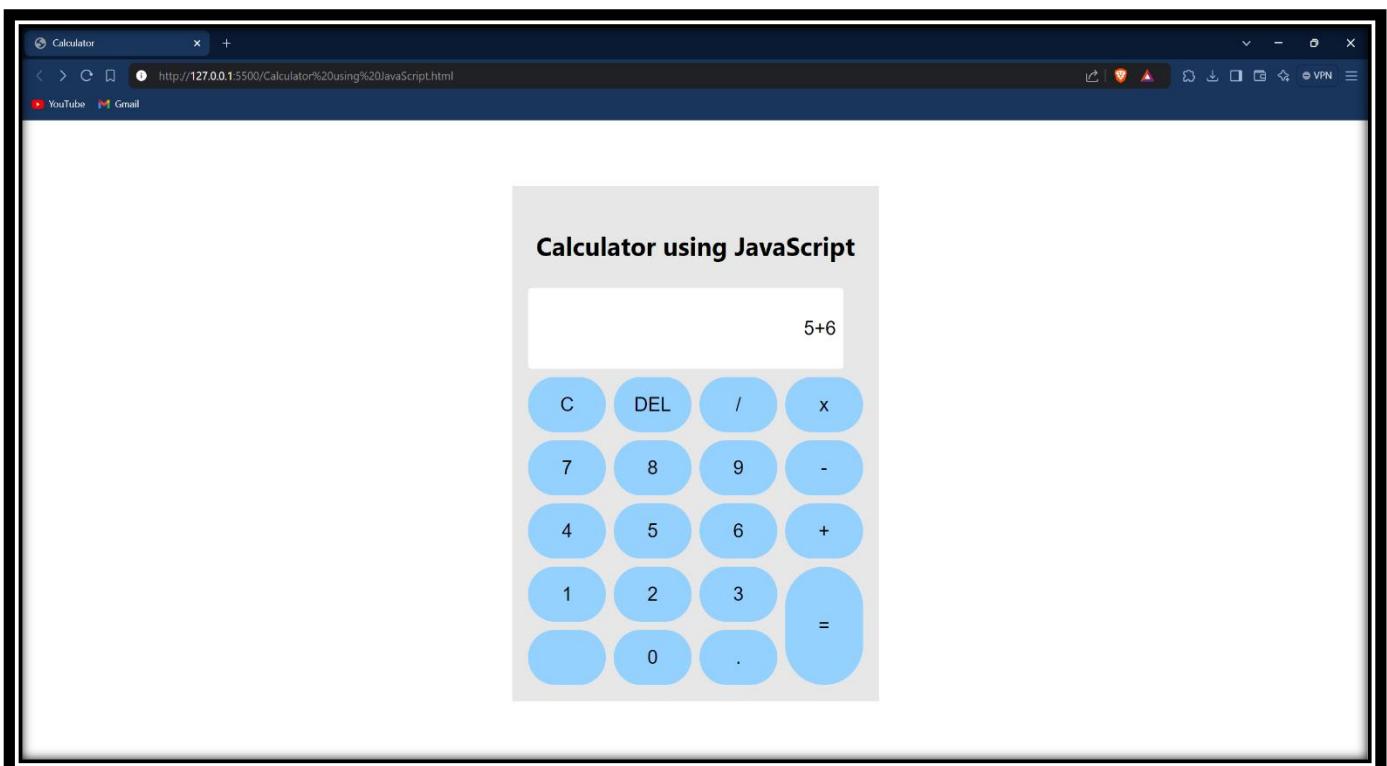


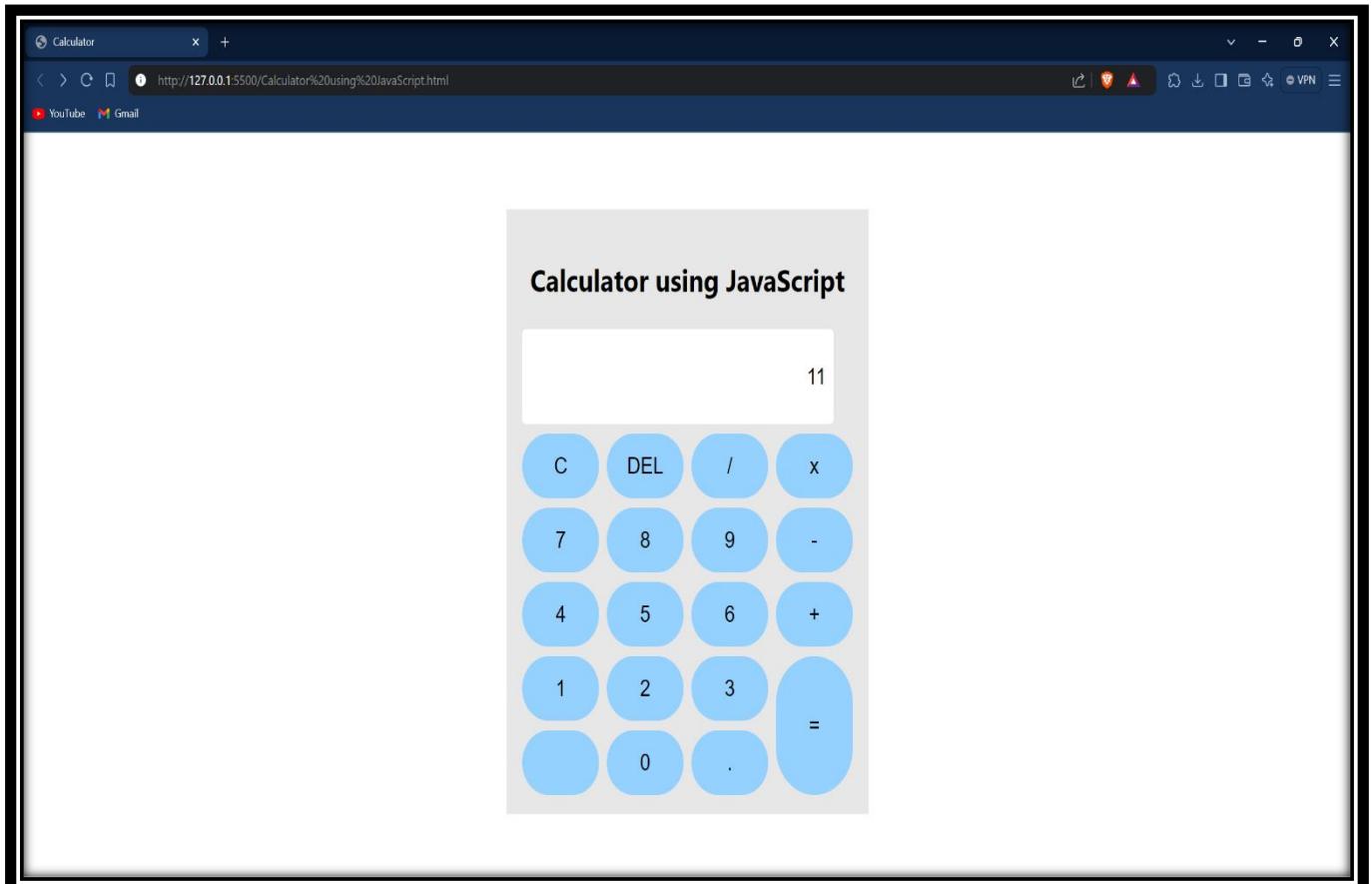
```

# style_calculator.css X
15    border:none;
16    padding-right:10px;
17    margin-bottom:10px;
18    font-size:24px;
19    border-radius: 5px;
20  }
21 .calculator-buttons{
22  display:grid;
23  grid-template-columns: repeat(4,1fr);
24  grid-gap: 10px;
25  }
26 button{
27  font-size:24px;
28  padding:20px;
29  border:none;
30  background-color: #95dfff;
31  color: black;
32  cursor:pointer;
33  border-radius:5px;
34  transition: background-color 0.3s ,transform 0.3s;
35  border:none;
36  border-radius:50px;
37  }
38 button:hover{
39  background-color: #rgb(255, 183, 0);
40  transform: scale(1.1);
41  }
42 button.bb{
43  grid-row: span 2;
44  }
45 h1{
46  text-align: center;
47  padding:10px;
48  font-family: system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;
49  }

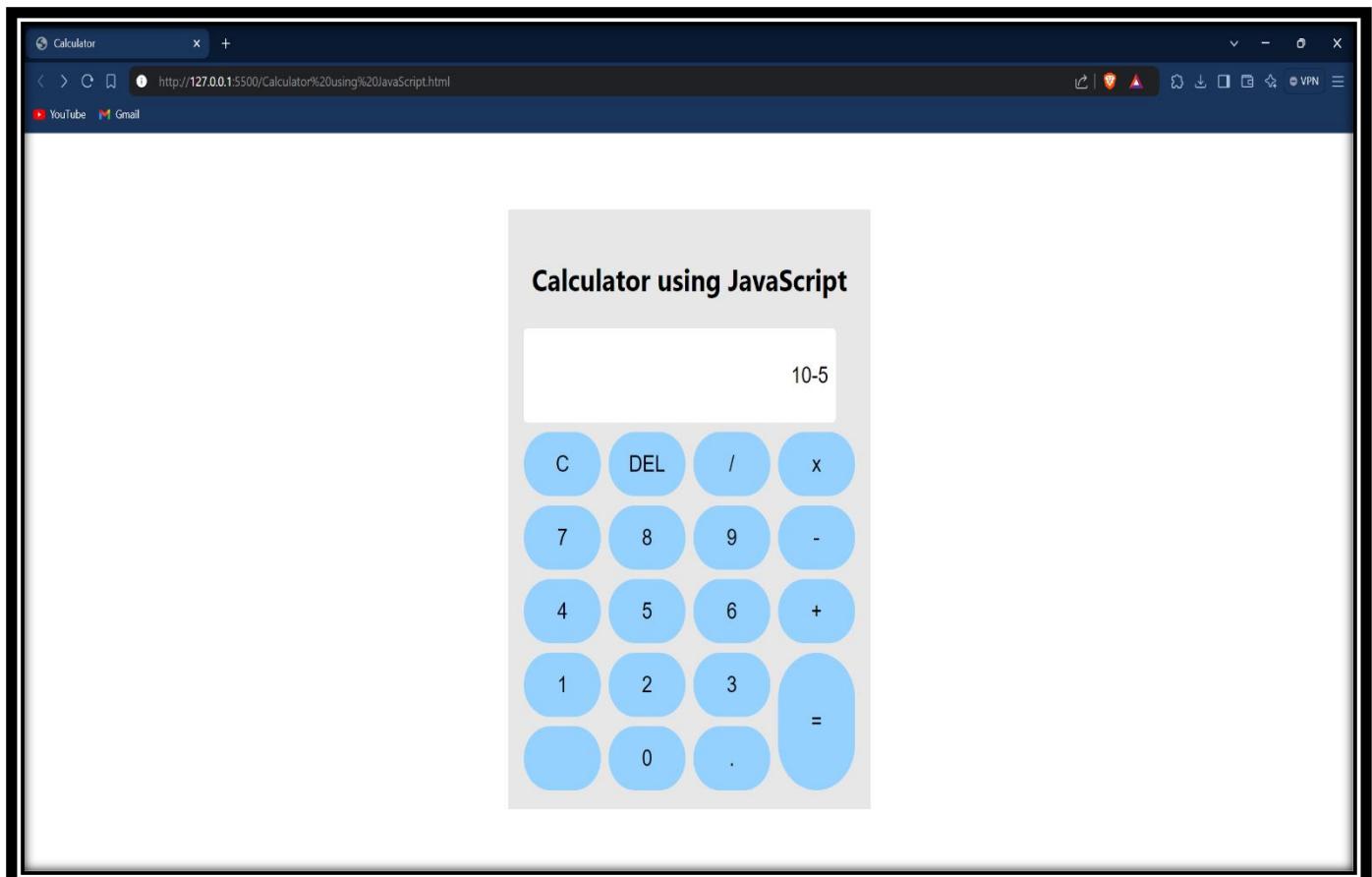
```

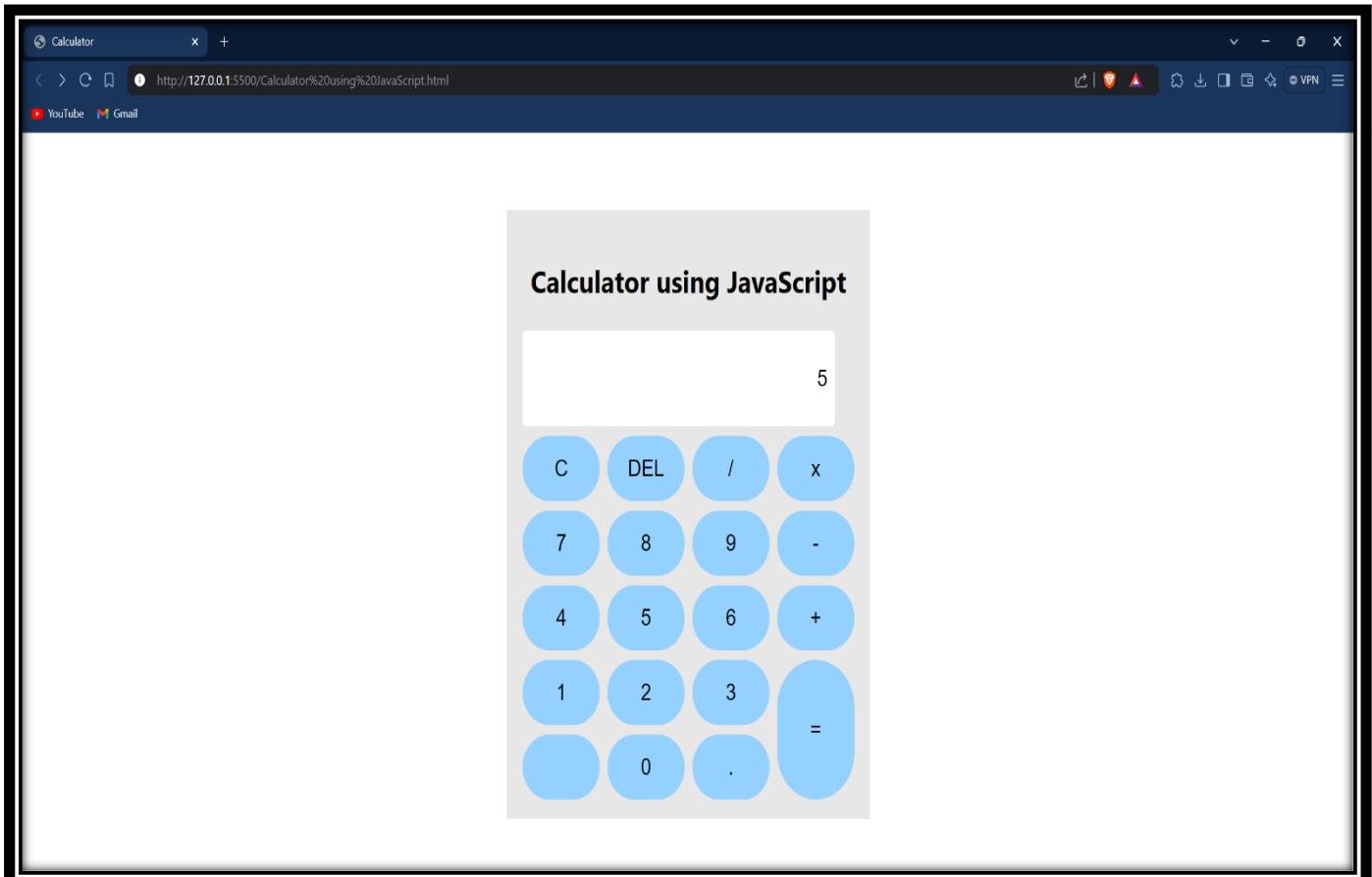
## Outputs: (a) Addition



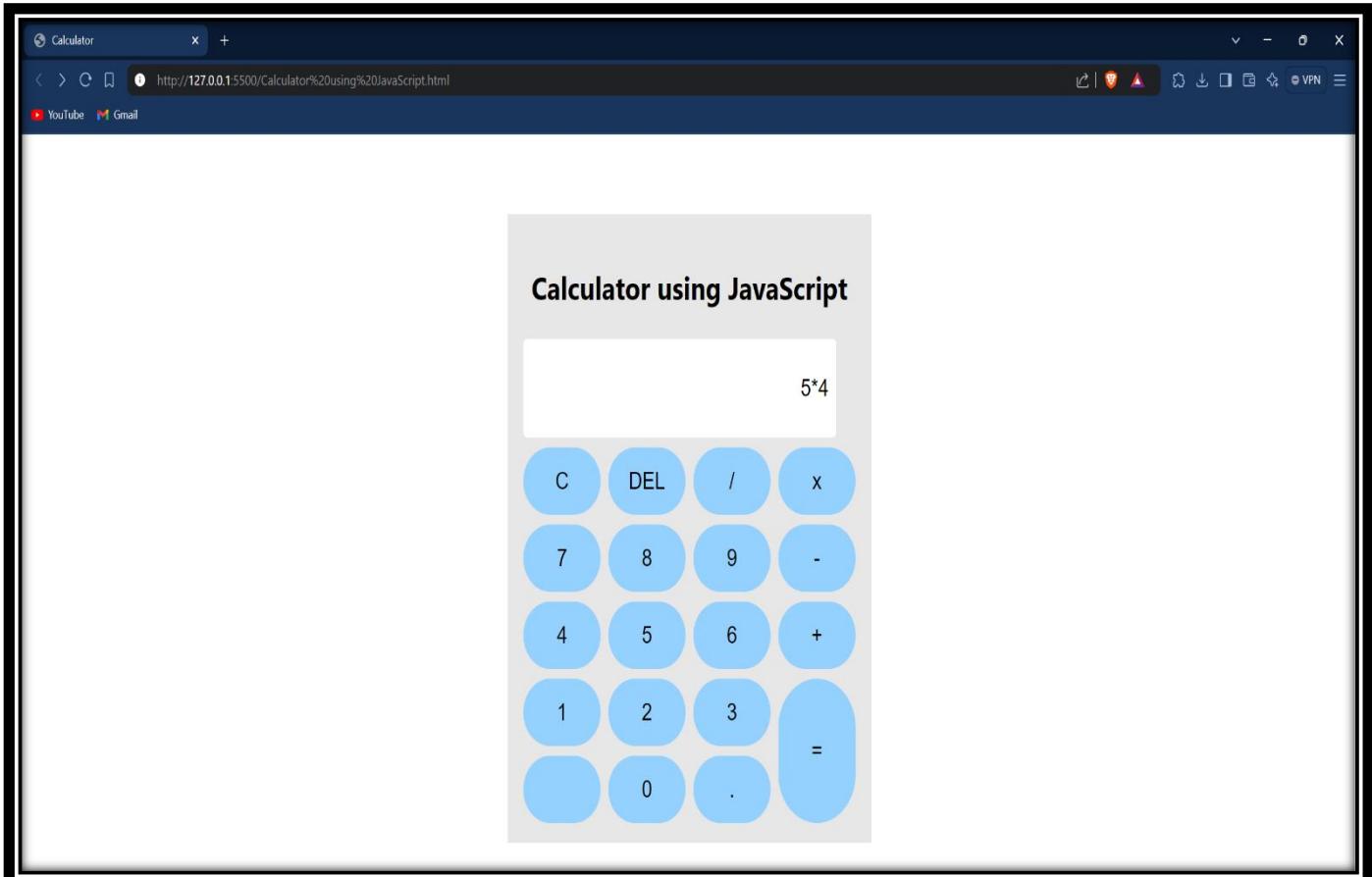


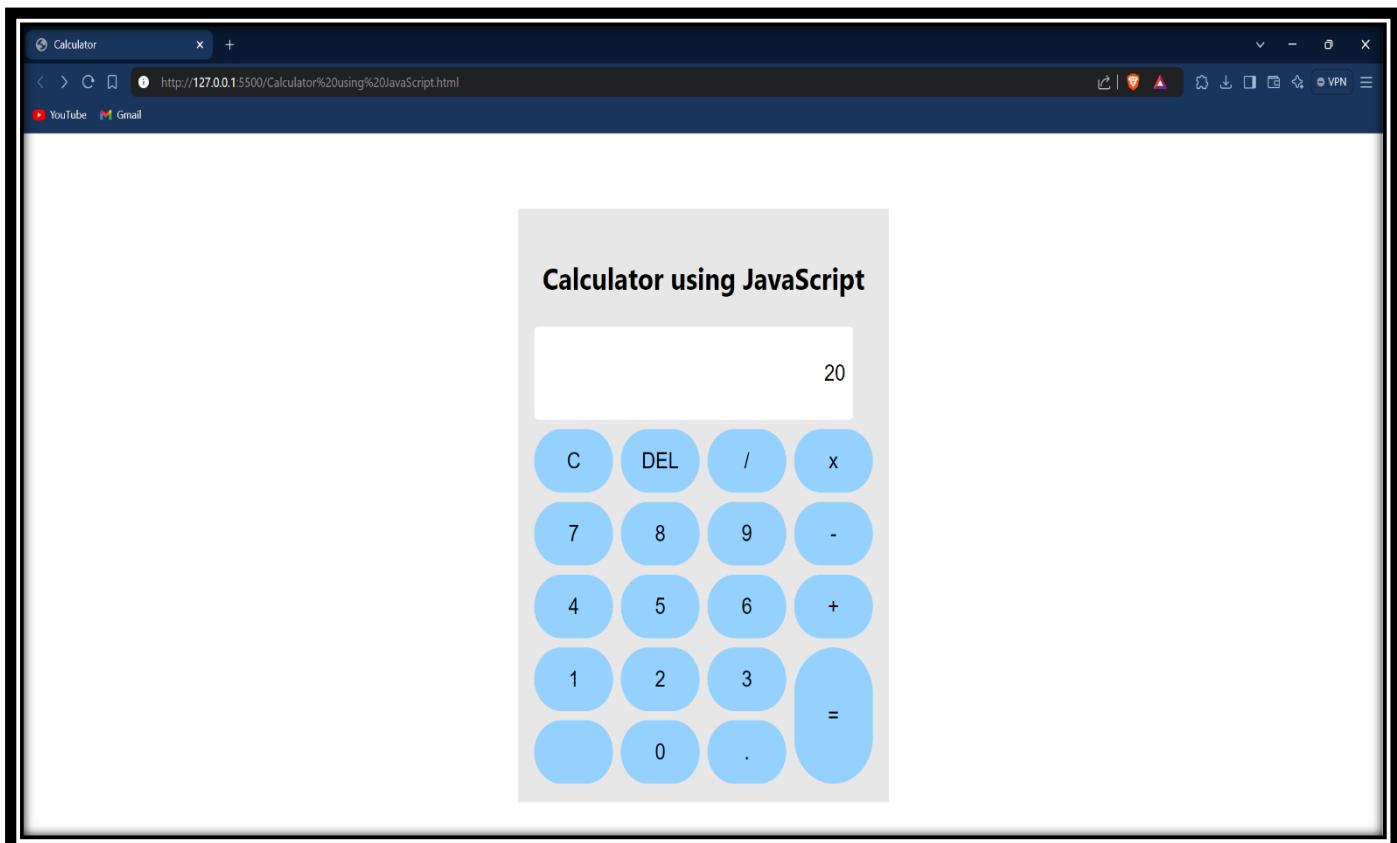
**(b) Subtraction:**



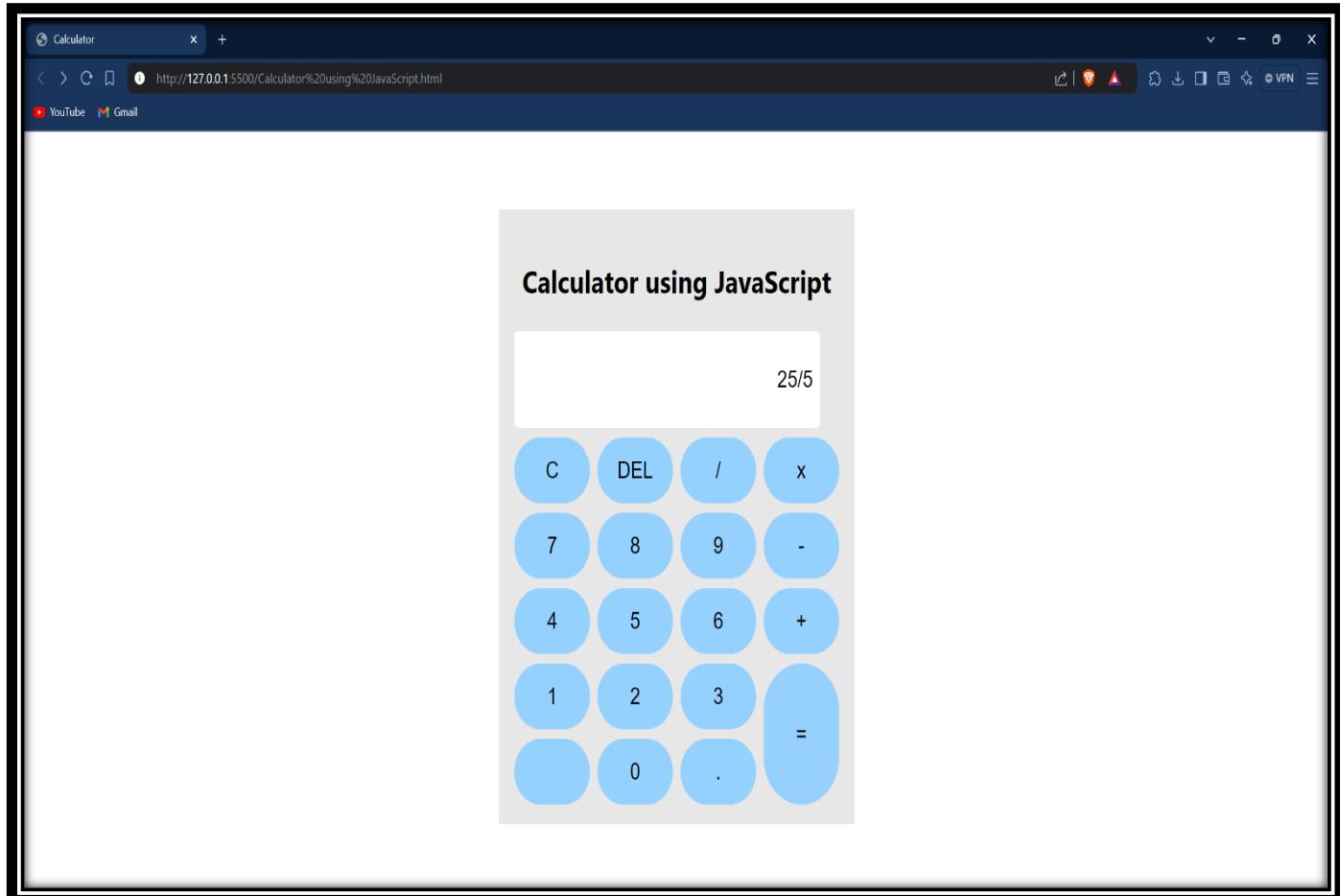


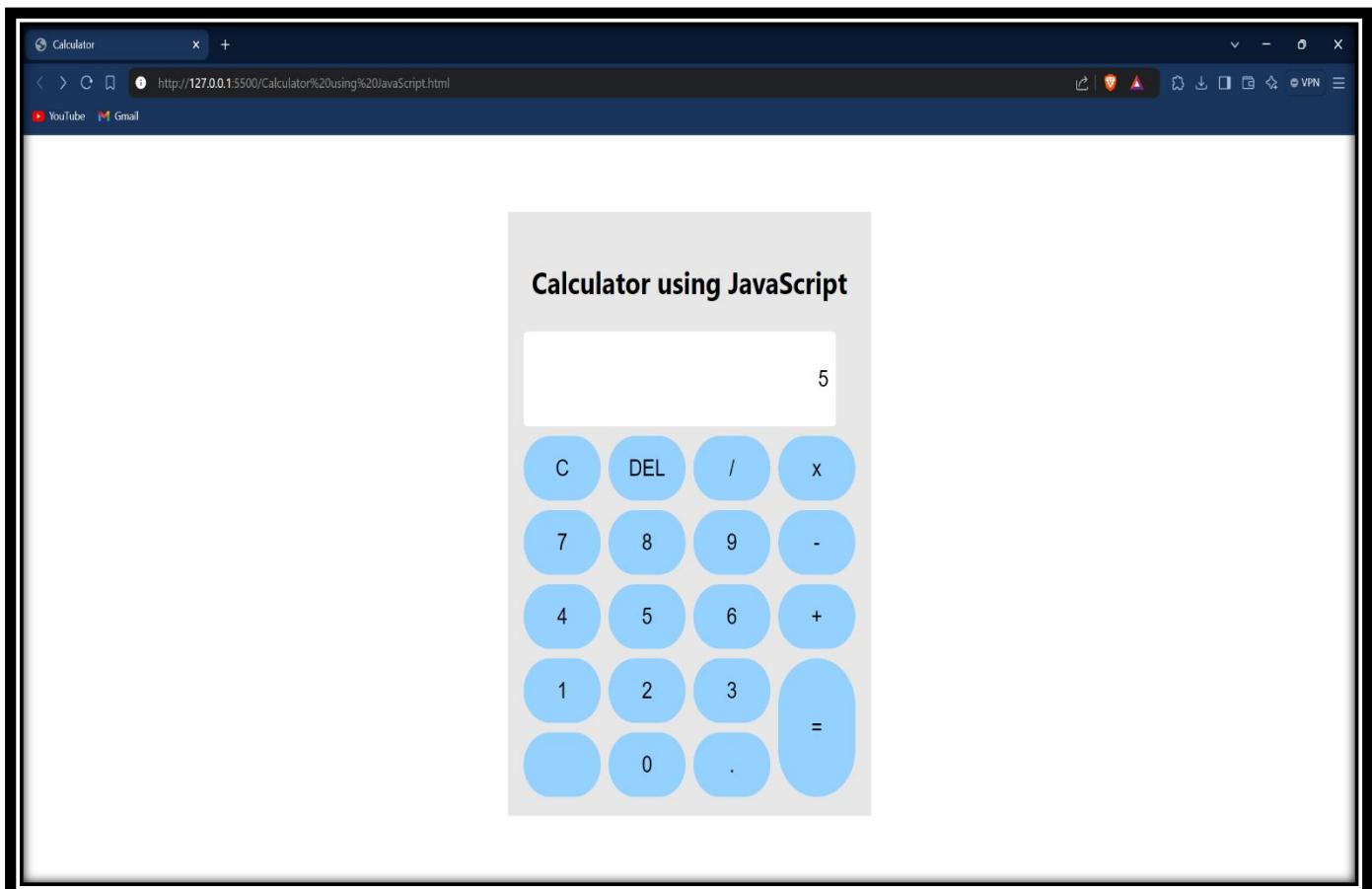
### (C) Multiplication:



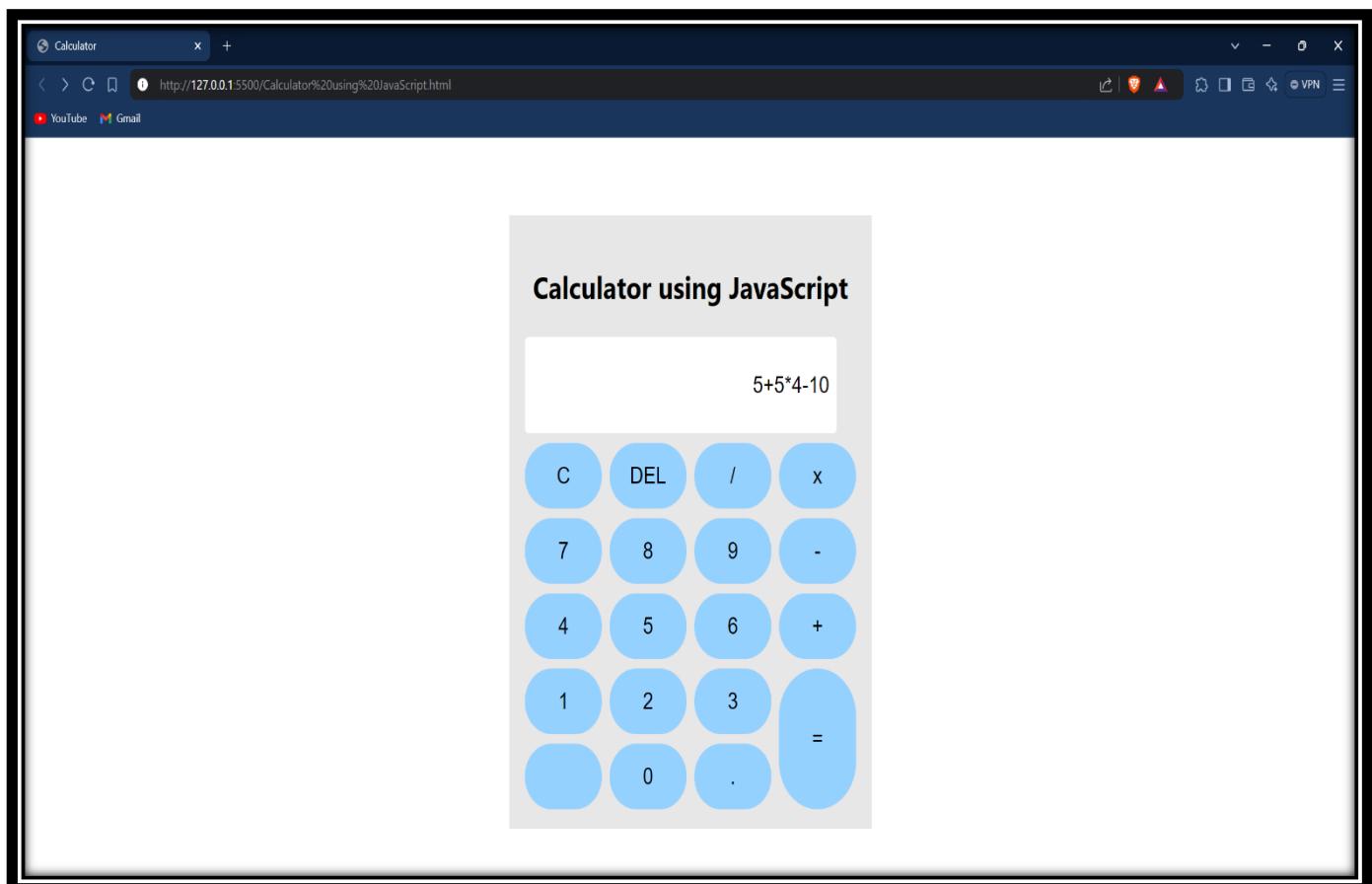


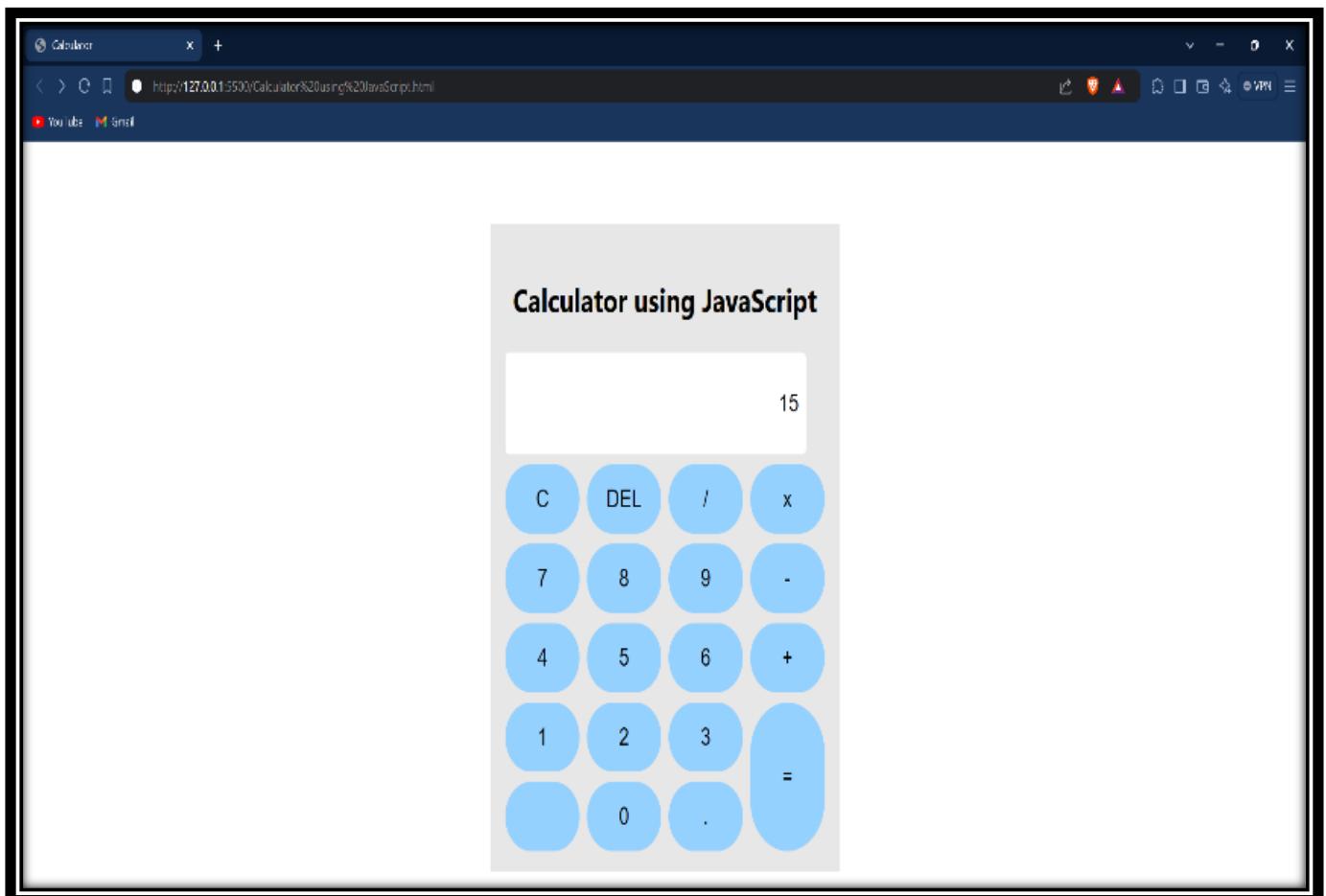
**(d) Division:**





(e) Multiple Operations at once:





### Learning Outcome:

## EXPERIMENT 4

**Problem statement:** Create a web page covering the basic CRUD operations (Create, Read, Update, Delete) that implements To-do/Grocery lists using JavaScript and HTML

**Theory:**

## Source Code:

The screenshot shows a code editor window with the following details:

- Title Bar:** File Edit Selection View Go Run Terminal Help
- Search Bar:** < Search (Ctrl+Shift+F)
- Code Area:** The code is for a file named "new.html". It contains HTML and CSS for a To-Do list application.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>To-Do/Grocery List</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }
        input[type="text"], input[type="number"] {
            width: 100px;
            padding: 10px;
            margin-right: 10px;
        }
        button {
            padding: 10px 15px;
        }
        table {
            width: 100%;
            border-collapse: collapse;
            margin-top: 20px;
        }
        th, td {
            border: 1px solid #4000ff;
            padding: 8px;
            text-align: left;
        }
        th {
            background-color: #89c5f0;
        }
        td button {
            margin-left: 5px;
        }
    </style>

```

- Sidebar:** A sidebar on the left side of the editor window contains various icons for file operations, search, and other developer tools.
- Bottom Bar:** Includes buttons for file operations like New, Open, Save, and Print, along with Add Logs and Share Code Link.

The screenshot shows a code editor interface with a blue header bar at the top containing the name "KUNSH SABHARWAL". Below the header is a toolbar with various icons: a file icon, a search icon, a refresh icon, and a help icon. To the right of the toolbar is a search bar with the placeholder text "Practical File Codes". The main workspace displays an HTML file titled "to-do list(aiml a) new.html". The code includes an HTML structure for a grocery list with a table for items and a script for adding items to an array.

```
</style>
</head>
<body>
<style>body {background-image: url('backgroundimagefortodolist.jpg');}</style>
<h1>To-Do/Grocery List</h1>
<input type="text" id="itemInput" placeholder="Product Name..." >
<input type="number" id="priceInput" placeholder="Price..." step="1">
<input type="number" id="quantityInput" placeholder="Quantity..." min="1">
<button onclick="addItem()">Submit</button>
<table id="itemList">
  <thead>
    <tr>
      <th>Product Name</th>
      <th>Product Price</th>
      <th>Product Quantity</th>
      <th>Edit/Delete</th>
    </tr>
  </thead>
  <tbody></tbody>
</table>
<script>
let items = [];
function addItem() {
  const itemName = document.getElementById('itemInput').value.trim();
  const itemPrice = parseFloat(document.getElementById('priceInput').value);
  const itemQuantity = parseInt(document.getElementById('quantityInput').value);
  if (itemName === '' || isNaN(itemPrice) || isNaN(itemQuantity) || itemQuantity <= 0) {
    alert('Please enter valid item details.');
    return;
  }
  items.push({ name: itemName, price: itemPrice, quantity: itemQuantity });
  clearInputs();
  renderItems();
}
</script>
```

The screenshot shows a code editor interface with a dark theme. On the left, there is a sidebar with various icons: a file icon, a magnifying glass, a gear, a person icon, and a refresh icon. The main area displays a file titled "to-do list(aiml a) new.html". The code is written in JavaScript and HTML, handling the creation and rendering of a todo list. It includes functions for clearing inputs, rendering items, editing items, and deleting items. The code uses document.getElementById and getElementsByTagName to interact with the DOM, and prompt to get user input for editing.

```
File Edit Selection View Go Run Terminal Help ← → 🔍 Practical File Codes

to-do list(aiml a) new.html X

71
72     function clearInputs() {
73         document.getElementById('itemInput').value = '';
74         document.getElementById('priceInput').value = '';
75         document.getElementById('quantityInput').value = '';
76     }
77
78     function renderItems() {
79         const itemList = document.getElementById('itemList').getElementsByTagName('tbody')[0];
80         itemList.innerHTML = '';
81         items.forEach((item, index) => {
82             const row = itemList.insertRow();
83             row.innerHTML =
84                 <td>${item.name}</td>
85                 <td>${item.price.toFixed(2)}</td>
86                 <td>${item.quantity}</td>
87                 <td>
88                     <button onclick="editItem(${index})">Edit</button>
89                     <button onclick="deleteItem(${index})">Delete</button>
90                 </td>
91             ;
92         });
93     }
94
95     function editItem(index) {
96         const item = items[index];
97         const newName = prompt('Edit product name:', item.name);
98         const newPrice = prompt('Edit price:', item.price);
99         const newQuantity = prompt('Edit quantity:', item.quantity);
100
101         if (newName !== null && newName.trim() !== '' &&
102             !isNaN(newPrice) && !isNaN(newQuantity) &&
103             parseInt(newQuantity) > 0) {
104             items[index] = {
105                 name: newName.trim(),
```

The screenshot shows a code editor interface with a dark theme. On the left, there is a vertical toolbar with icons for file operations, search, refresh, and other development tools. The main area displays a file named "to-do list(aiml a) new.html". The code is written in HTML and JavaScript. It includes a table structure with edit and delete buttons, and two functions for editing and deleting items from an array of objects.

```
87     <td>
88         <button onclick="editItem(${index})>Edit</button>
89         <button onclick="deleteItem(${index})>Delete</button>
90     </td>
91     `;
92   );
93 }
94
95 function editItem(index) {
96   const item = items[index];
97   const newName = prompt('Edit product name:', item.name);
98   const newPrice = prompt('Edit price:', item.price);
99   const newQuantity = prompt('Edit quantity:', item.quantity);
100
101  if (newName !== null && newName.trim() !== '' &&
102      !isNaN(newPrice) && !isNaN(newQuantity) &&
103      parseInt(newQuantity) > 0) {
104    items[index] = {
105      name: newName.trim(),
106      price: parseFloat(newPrice),
107      quantity: parseInt(newQuantity)
108    };
109    renderItems();
110  }
111}
112
113 function deleteItem(index) {
114  if (confirm('Are you sure you want to delete this item?')) {
115    items.splice(index, 1);
116    renderItems();
117  }
118}
119 </script>
120 </body>
121 </html>
```

**Output:**

To-Do/Grocery List

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50	2	

**(a) Adding elements:**

To-Do/Grocery List

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50	2	

To-Do/Grocery List

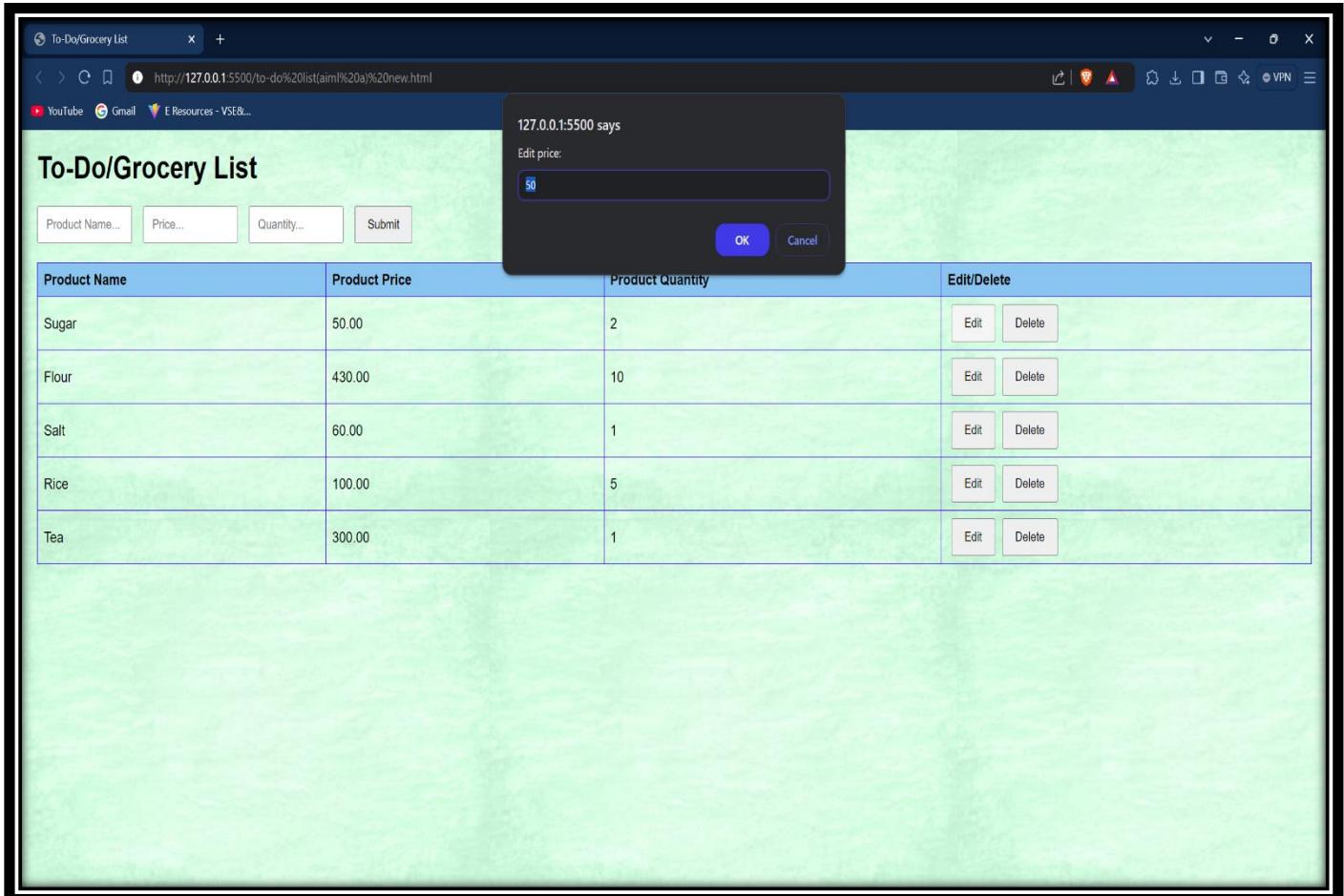
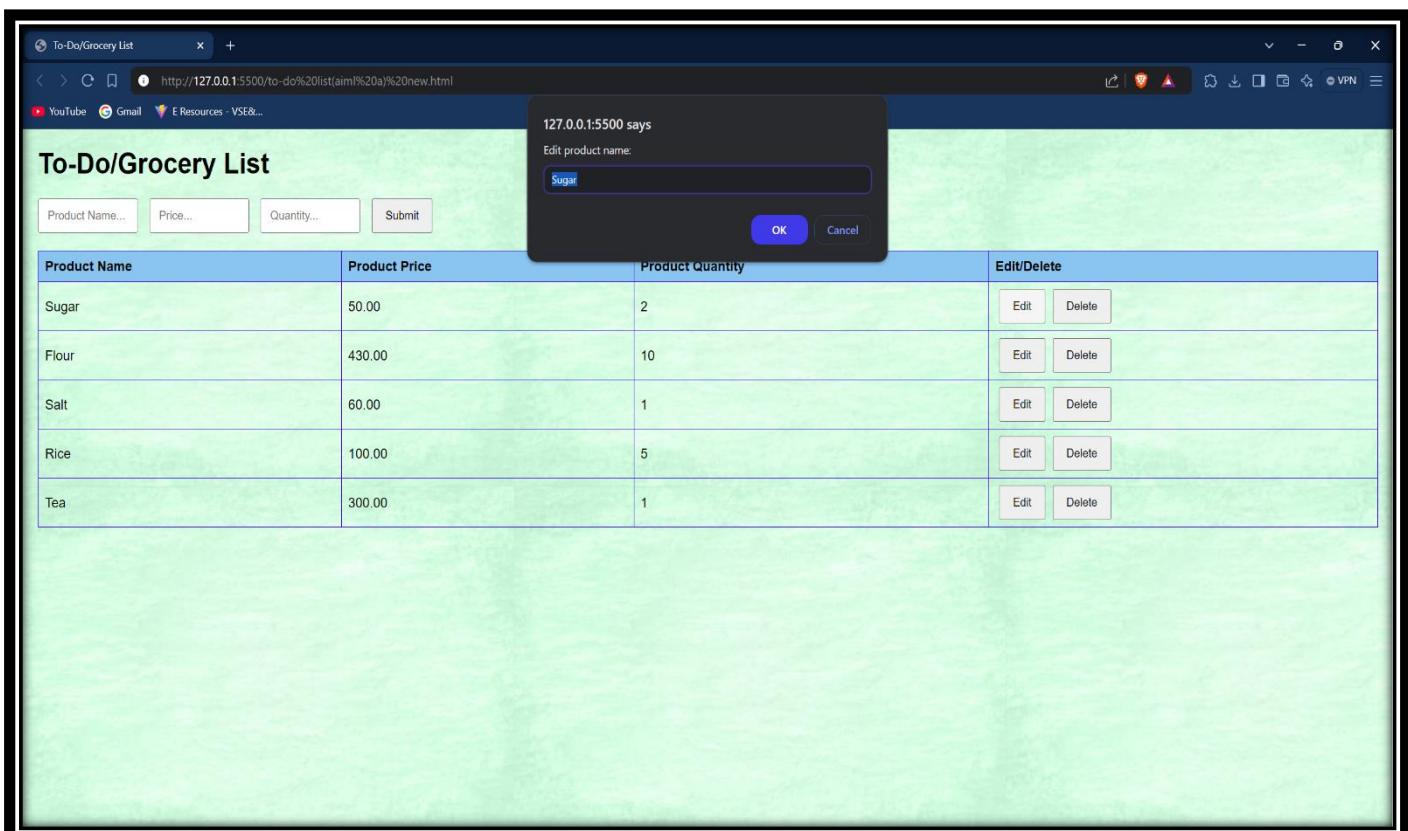
Product Name... Price... Quantity... Submit

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50.00	2	<button>Edit</button> <button>Delete</button>

To-Do/Grocery List

Product Name... Price... Quantity... Submit

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50.00	2	<button>Edit</button> <button>Delete</button>
Flour	430.00	10	<button>Edit</button> <button>Delete</button>
Salt	60.00	1	<button>Edit</button> <button>Delete</button>
Rice	100.00	5	<button>Edit</button> <button>Delete</button>
Tea	300.00	1	<button>Edit</button> <button>Delete</button>

**(b) Updating an element:**

# KUNSH SABHARWAL

To-Do/Grocery List

Product Name... Price... Quantity... Submit

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50.00	2	<button>Edit</button> <button>Delete</button>
Flour	430.00	10	<button>Edit</button> <button>Delete</button>
Salt	60.00	1	<button>Edit</button> <button>Delete</button>
Rice	100.00	5	<button>Edit</button> <button>Delete</button>
Tea	300.00	1	<button>Edit</button> <button>Delete</button>

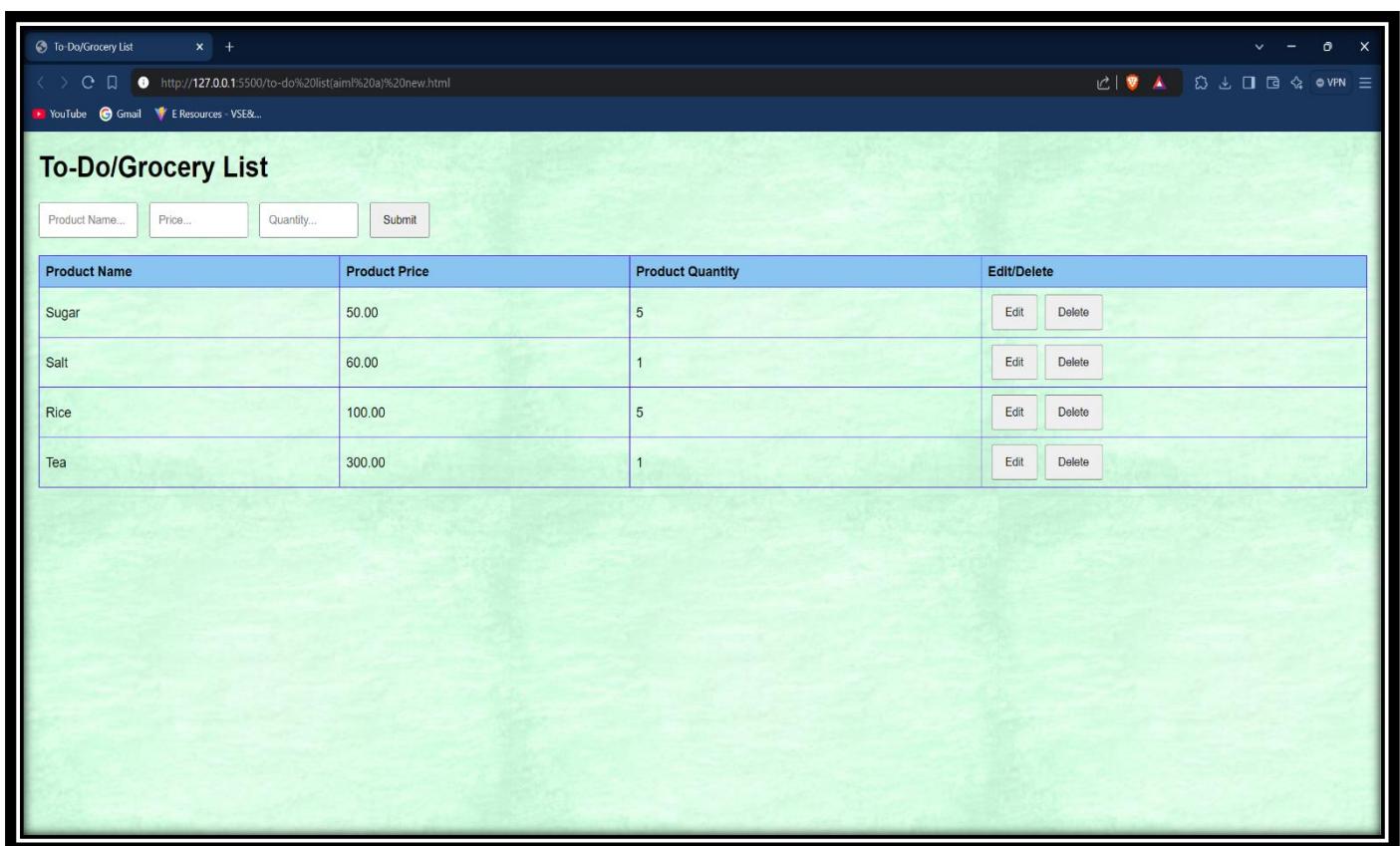
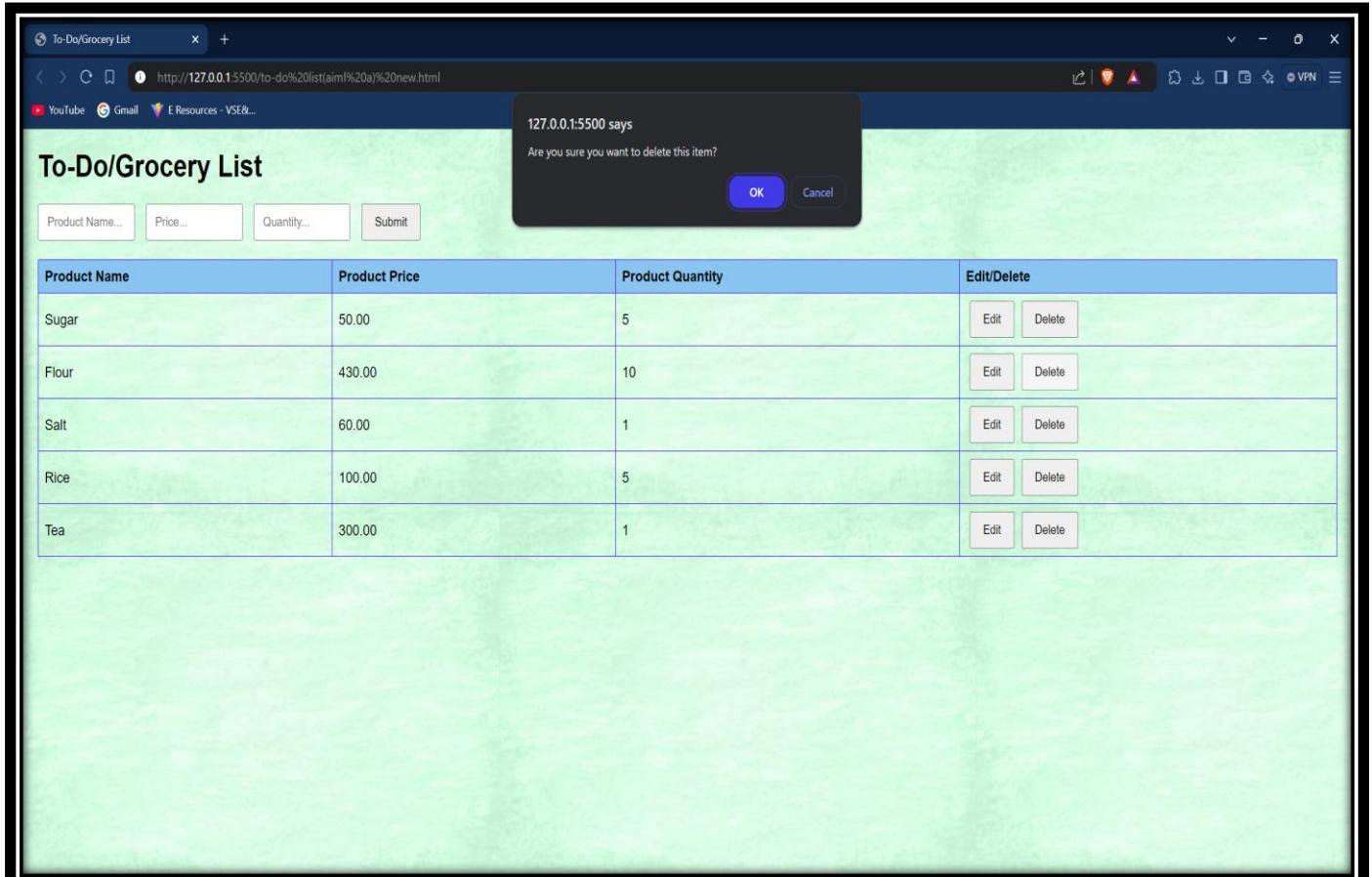
127.0.0.1:5500 says  
Edit quantity:  
5|

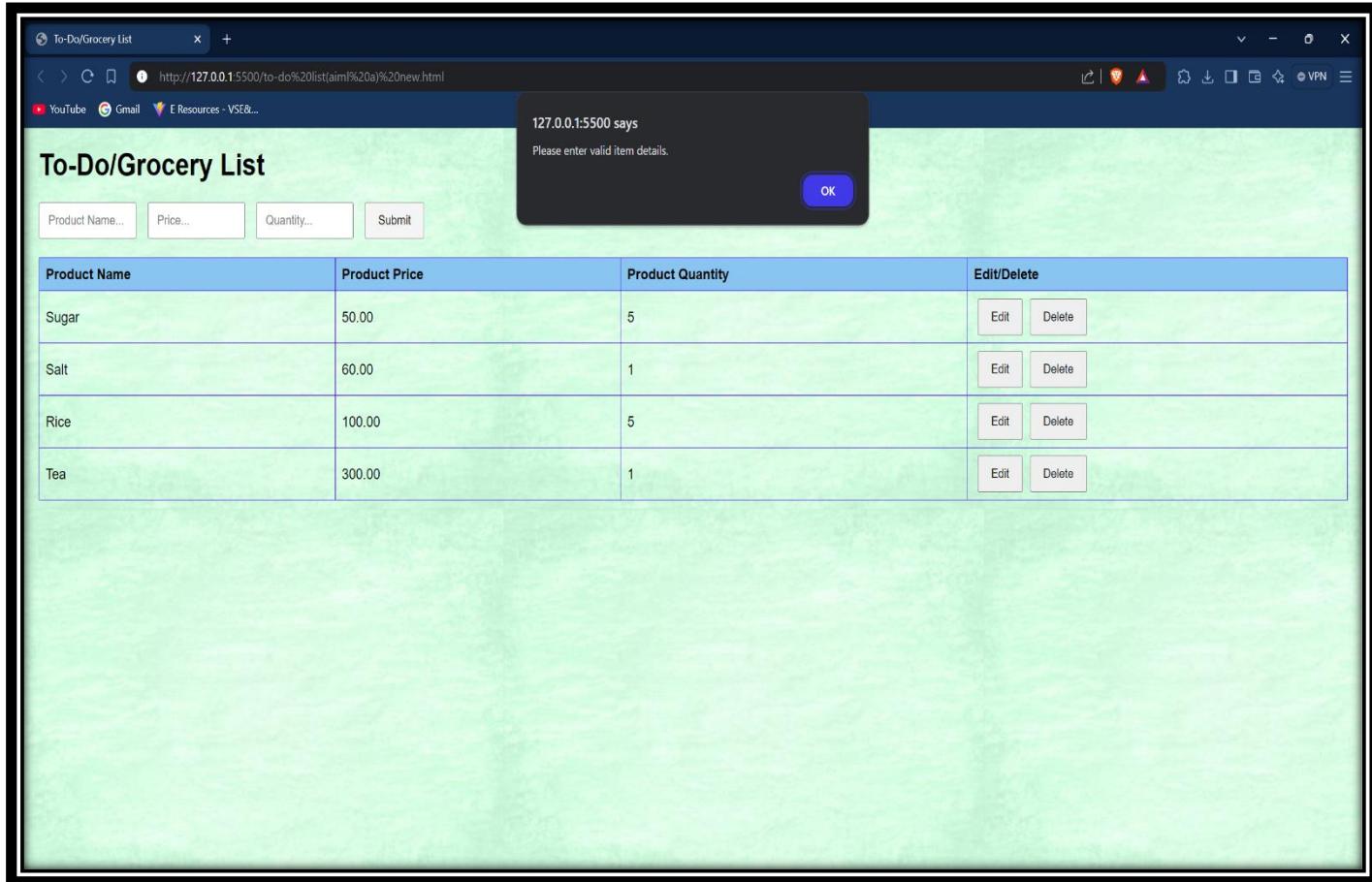
OK Cancel

To-Do/Grocery List

Product Name... Price... Quantity... Submit

Product Name	Product Price	Product Quantity	Edit/Delete
Sugar	50.00	5	<button>Edit</button> <button>Delete</button>
Flour	430.00	10	<button>Edit</button> <button>Delete</button>
Salt	60.00	1	<button>Edit</button> <button>Delete</button>
Rice	100.00	5	<button>Edit</button> <button>Delete</button>
Tea	300.00	1	<button>Edit</button> <button>Delete</button>

**(c) Deleting an Element:**

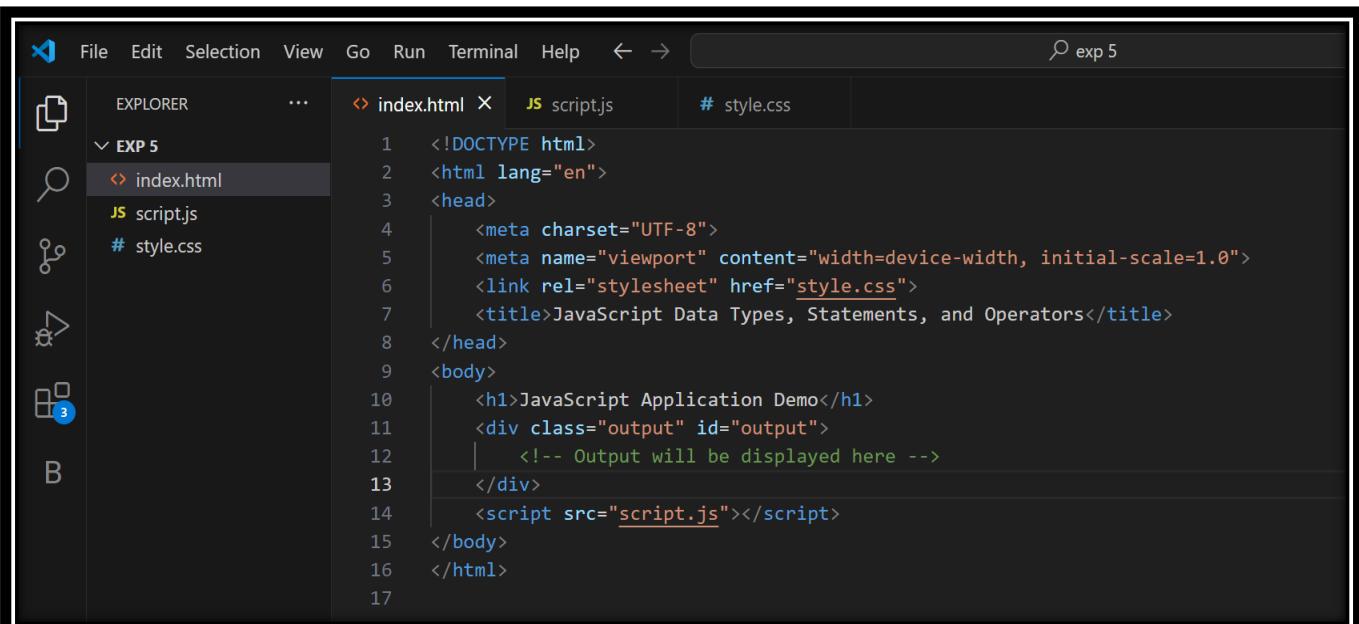
**(d) Exception Handling:****Learning Outcome:**

## **EXPERIMENT 5**

**Problem statement:** Create a JavaScript application based on various Data Types, Statements, Keywords and Operators.

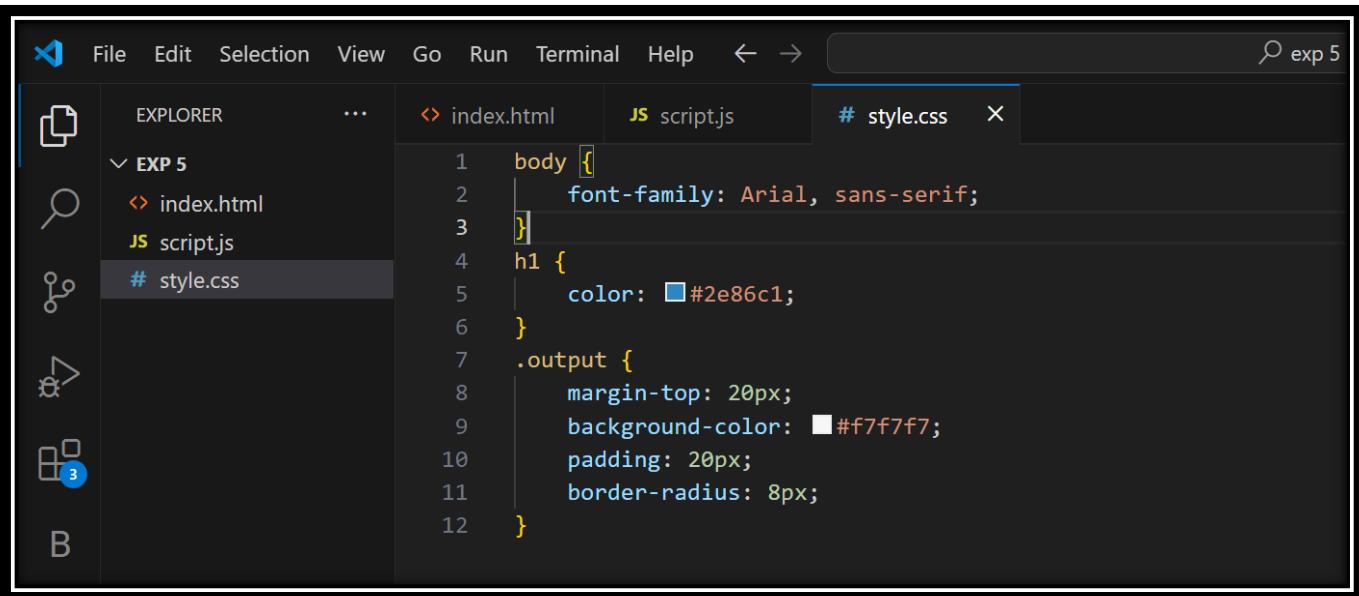
**Theory:**

## Source Code:



VS Code interface showing the Explorer, Editor, and Terminal panes. The Explorer pane shows files: index.html, script.js, and style.css. The Editor pane displays the content of index.html.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="style.css">
    <title>JavaScript Data Types, Statements, and Operators</title>
</head>
<body>
    <h1>JavaScript Application Demo</h1>
    <div class="output" id="output">
        <!-- Output will be displayed here -->
    </div>
    <script src="script.js"></script>
</body>
</html>
```



VS Code interface showing the Explorer, Editor, and Terminal panes. The Explorer pane shows files: index.html, script.js, and style.css. The Editor pane displays the content of style.css.

```
body {
    font-family: Arial, sans-serif;
}
h1 {
    color: #2e86c1;
}
.output {
    margin-top: 20px;
    background-color: #f7f7f7;
    padding: 20px;
    border-radius: 8px;
}
```

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows a project named "EXP 5" containing files: "index.html", "script.js" (selected), and "# style.css".
- Code Editor:** Displays the contents of "script.js".

```
function printToWindow(message) {
    let outputDiv = document.getElementById("output");
    outputDiv.innerHTML += message + "<br>";
}

let number = 42;
let string = "Hello, World!";
let isBoolean = true;
let undefinedVariable;
let nullValue = null;

let person = {
    name: "Yash",
    age: 20,
    isStudent: true
};

let fruits = ["Apple", "Banana", "Mango"];
printToWindow("Number: " + number);
printToWindow("String: " + string);
printToWindow("Boolean: " + isBoolean);
printToWindow("Undefined: " + undefinedVariable);
printToWindow("Null: " + nullValue);
printToWindow("Object: " + JSON.stringify(person));
printToWindow("Array: " + fruits.join(", "));

if (number > 20) {
    printToWindow("Number is greater than 20");
} else {
    printToWindow("Number is less than or equal to 20");
}

for (let i = 0; i < fruits.length; i++) {
    printToWindow("Fruit " + (i + 1) + ": " + fruits[i]);
}

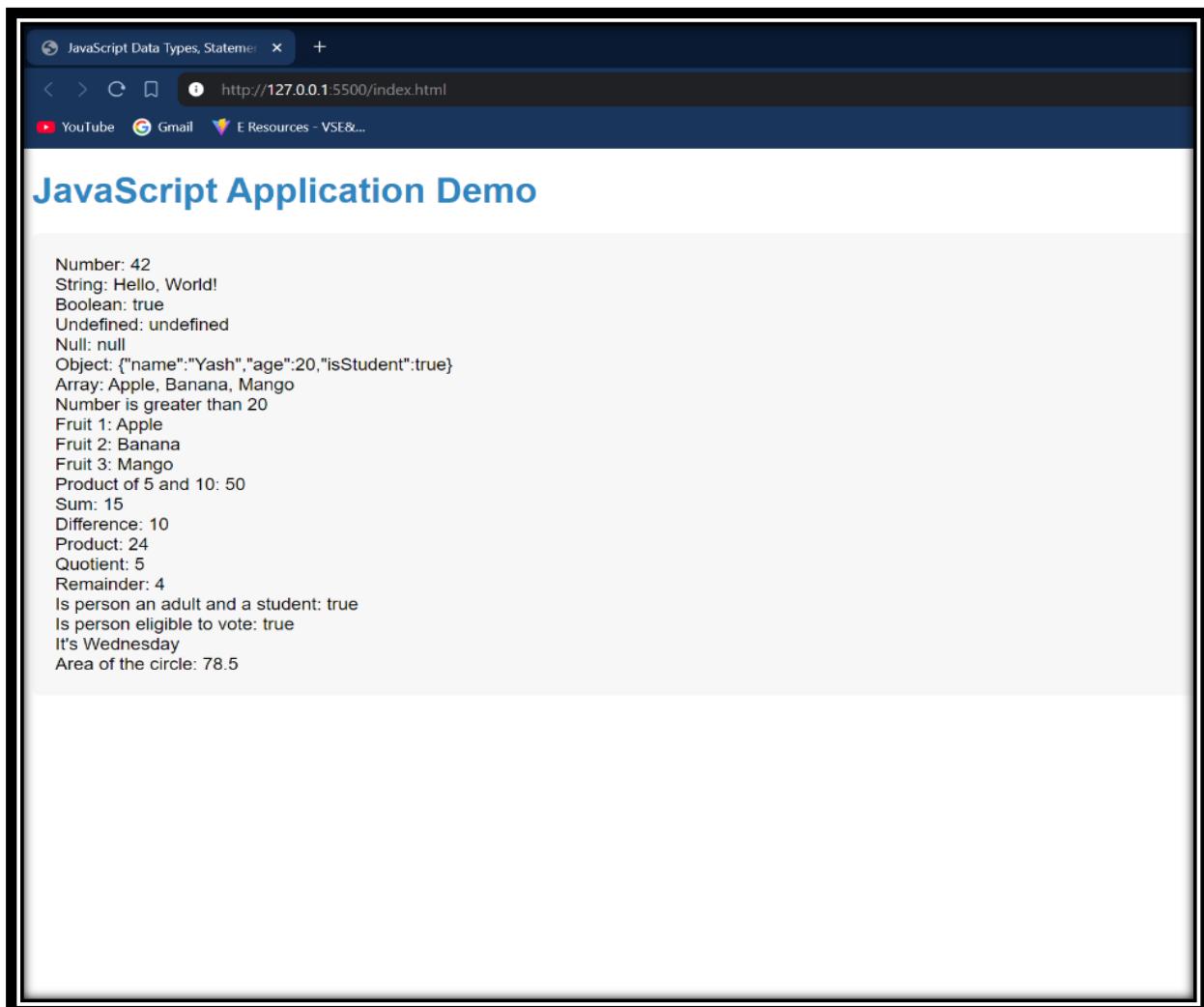
function multiply(a, b) {
    return a * b;
}

let product = multiply(5, 10);
printToWindow("Product of 5 and 10: " + product);
```
- Status Bar:** Shows icons for file status (0△0), logs (0), add logs, share code link, line 66, column 35, spaces: 4, UTF-8, LF, JavaScript, and port 5500.

The screenshot shows a code editor interface with the following details:

- File Explorer:** On the left, there is a sidebar with icons for file operations. Below it, the project structure is shown: **EXP 5** contains **index.html**, **script.js** (which is currently selected), and **# style.css**.
- Code Editor:** The main area displays the contents of **script.js**. The code includes:
  - Line 34: `let product = multiply(5, 10);`
  - Line 35: `printToWindow("Product of 5 and 10: " + product);`
  - Line 36: `let sum = 5 + 10;`
  - Line 37: `let difference = 15 - 5;`
  - Line 38: `let product2 = 4 * 6;`
  - Line 39: `let quotient = 20 / 4;`
  - Line 40: `let remainder = 19 % 5;`
  - Line 41: `printToWindow("Sum: " + sum);`
  - Line 42: `printToWindow("Difference: " + difference);`
  - Line 43: `printToWindow("Product: " + product2);`
  - Line 44: `printToWindow("Quotient: " + quotient);`
  - Line 45: `printToWindow("Remainder: " + remainder);`
  - Line 46: `let isAdult = (person.age >= 18 && person.isStudent);`
  - Line 47: `let canVote = (person.age >= 18 || person.isStudent);`
  - Line 48: `printToWindow("Is person an adult and a student: " + isAdult);`
  - Line 49: `printToWindow("Is person eligible to vote: " + canVote);`
  - Line 50: `let day = 3;`
  - Line 51: `switch (day) {`
  - Line 52:  `case 1:`
  - Line 53:  `printToWindow("It's Monday");`
  - Line 54:  `break;`
  - Line 55:  `case 2:`
  - Line 56:  `printToWindow("It's Tuesday");`
  - Line 57:  `break;`
  - Line 58:  `case 3:`
  - Line 59:  `printToWindow("It's Wednesday");`
  - Line 60:  `break;`
  - Line 61:  `default:`
  - Line 62:  `printToWindow("It's another day");`
  - Line 63: `}`
  - Line 64: `const PI = 3.14;`
  - Line 65: `let radius = 5;`
  - Line 66: `let area = PI * radius * radius;`
  - Line 67: `printToWindow("Area of the circle: " + area);`
  - Line 68: (empty line)
- Bottom Bar:** Includes icons for file operations (New, Open, Save, etc.), a search bar, and status information: Line 66, Col 35, Spaces: 4, UTF-8, LF, JavaScript, Port: 5500.

## Outputs:



The screenshot shows a web browser window with the title "JavaScript Data Types, Statement" and the URL "http://127.0.0.1:5500/index.html". The browser's toolbar includes icons for YouTube, Gmail, and E Resources - VSE&... The main content area displays the heading "JavaScript Application Demo" in blue. Below it, there is a block of text representing the output of a JavaScript program, which includes various data types and statements.

```
Number: 42
String: Hello, World!
Boolean: true
Undefined: undefined
Null: null
Object: {"name":"Yash","age":20,"isStudent":true}
Array: Apple, Banana, Mango
Number is greater than 20
Fruit 1: Apple
Fruit 2: Banana
Fruit 3: Mango
Product of 5 and 10: 50
Sum: 15
Difference: 10
Product: 24
Quotient: 5
Remainder: 4
Is person an adult and a student: true
Is person eligible to vote: true
It's Wednesday
Area of the circle: 78.5
```

## Learning Outcome:

## **EXPERIMENT 6**

**Problem statement:** Create a JavaScript application with Window Objects and Document Object.

**Theory:**

## Source Code:

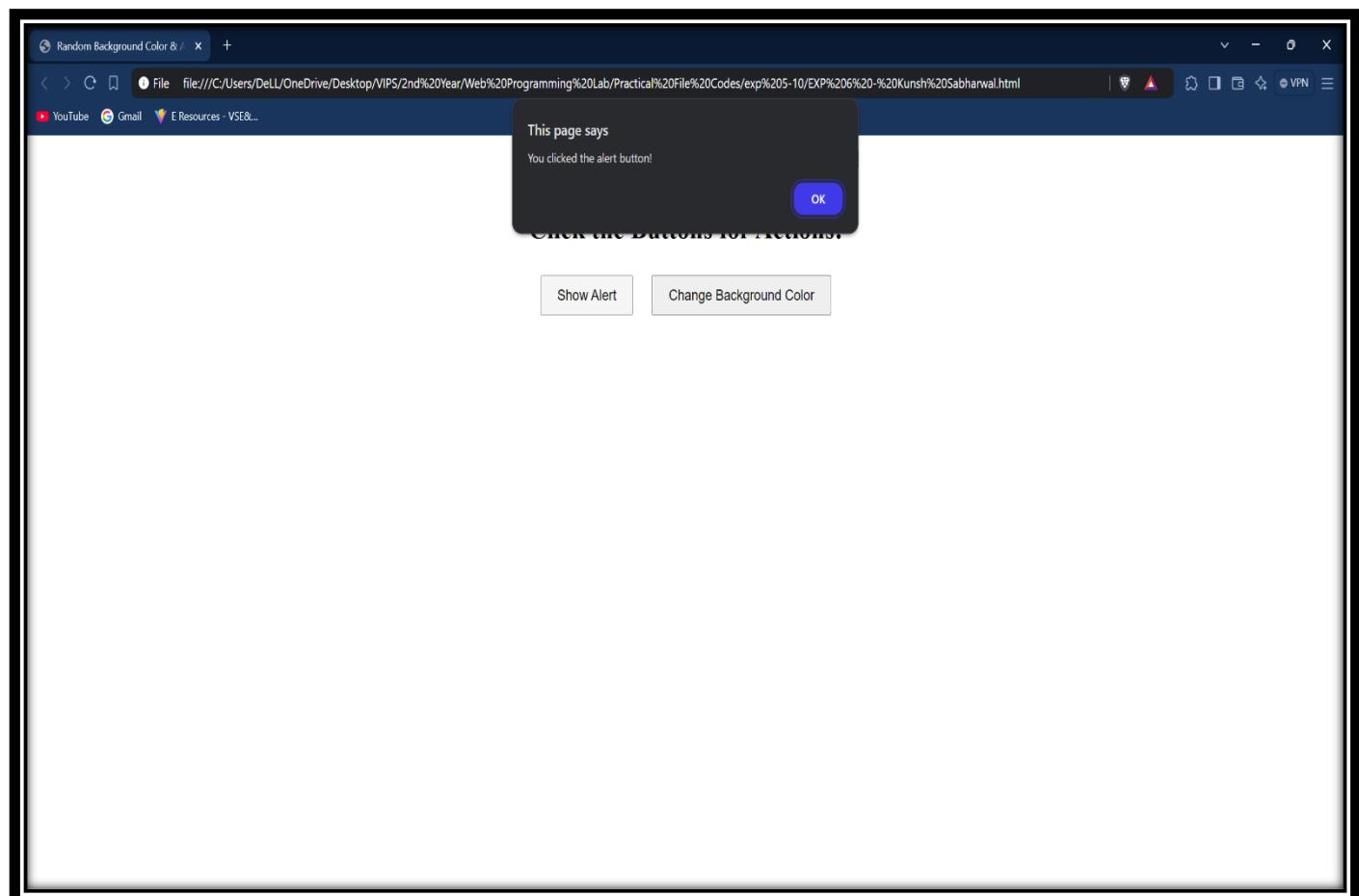
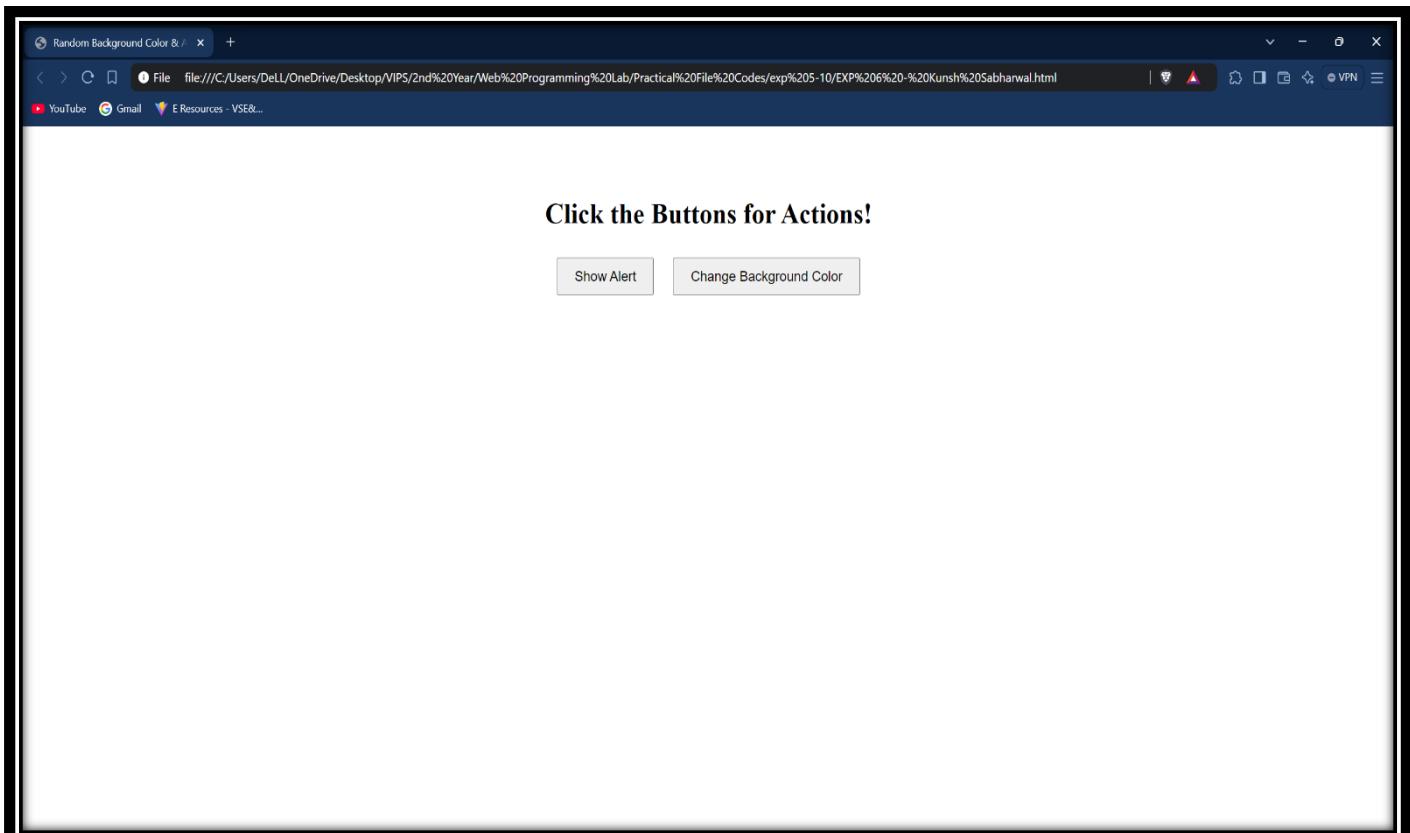
The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Includes icons for file operations like Open, Save, Find, Copy, Paste, and a refresh button.
- Code Area:** Displays the following code:

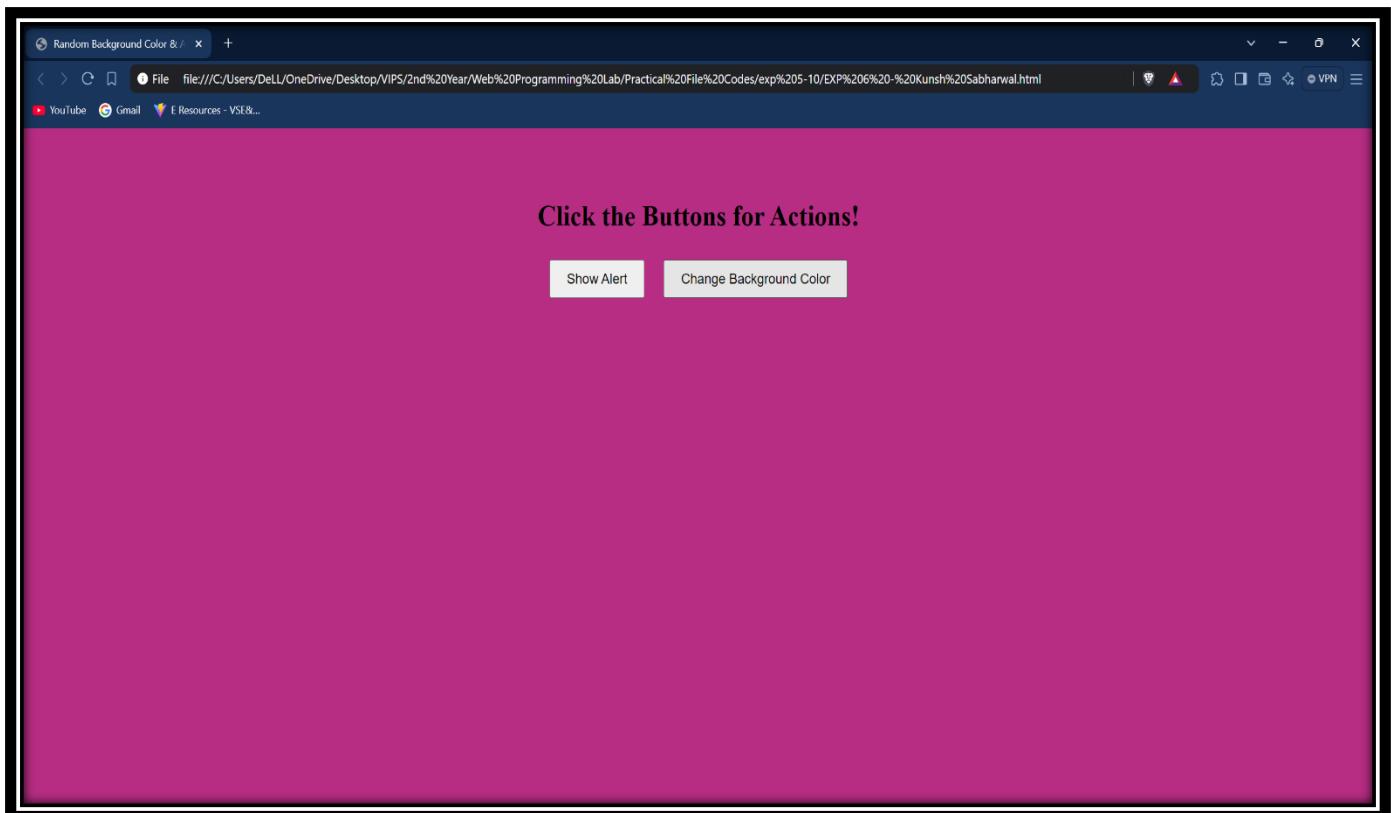
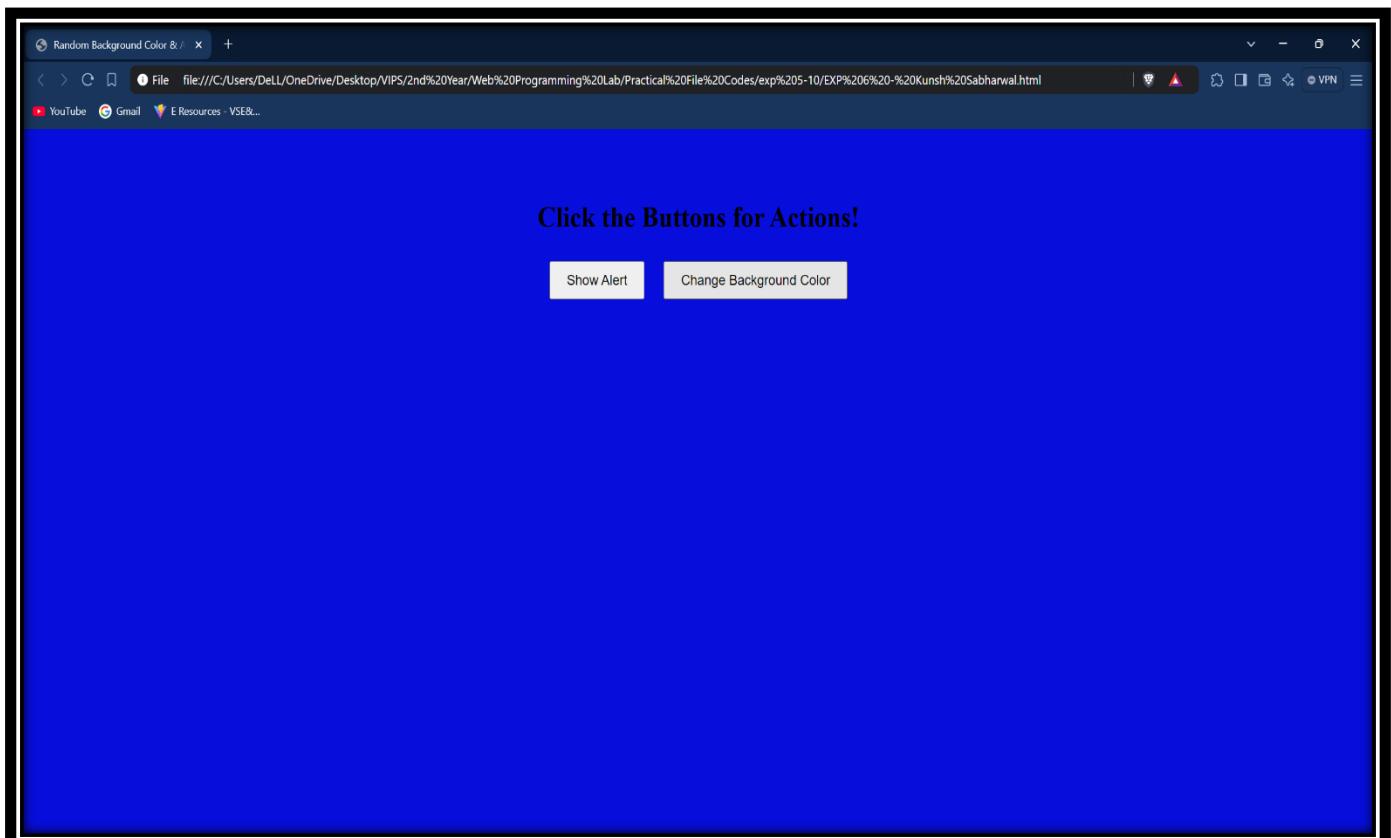
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Random Background Color & Alert</title>
    <style>
        body {
            text-align: center;
            padding: 50px;
        }
        button {
            padding: 10px 20px;
            font-size: 16px;
            margin: 10px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <h1>Click the Buttons for Actions!</h1>
    <button id="alertButton">Show Alert</button>
    <button id="colorButton">Change Background Color</button>
    <script>
        function getRandomColor() {
            const letters = '0123456789ABCDEF';
            let color = '#';
            for (let i = 0; i < 6; i++) {
                color += letters[Math.floor(Math.random() * 16)];
            }
            return color;
        }
        document.getElementById('alertButton').addEventListener('click', function() {
            window.alert("You clicked the alert button!");
        });
        document.getElementById('colorButton').addEventListener('click', function() {
            document.body.style.backgroundColor = getRandomColor();
        });
    </script>
</body>
</html>
```

**Status Bar:** Shows navigation icons (back, forward, search), log counts (0 errors, 0 warnings), and links for Add Logs, Improve Code, Version Control, and Share Code Link.

## Outputs:



## KUNSH SABHARWAL



**Learning Outcome:**

## EXPERIMENT 7

**Problem statement:** Create a JavaScript application with object Creation and by adding methods of Objects.

**Theory:**

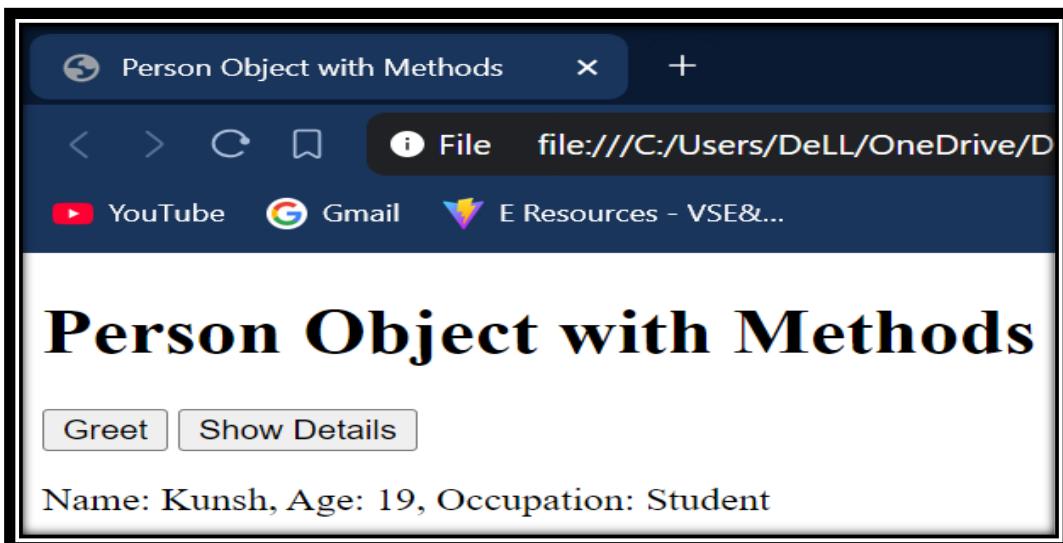
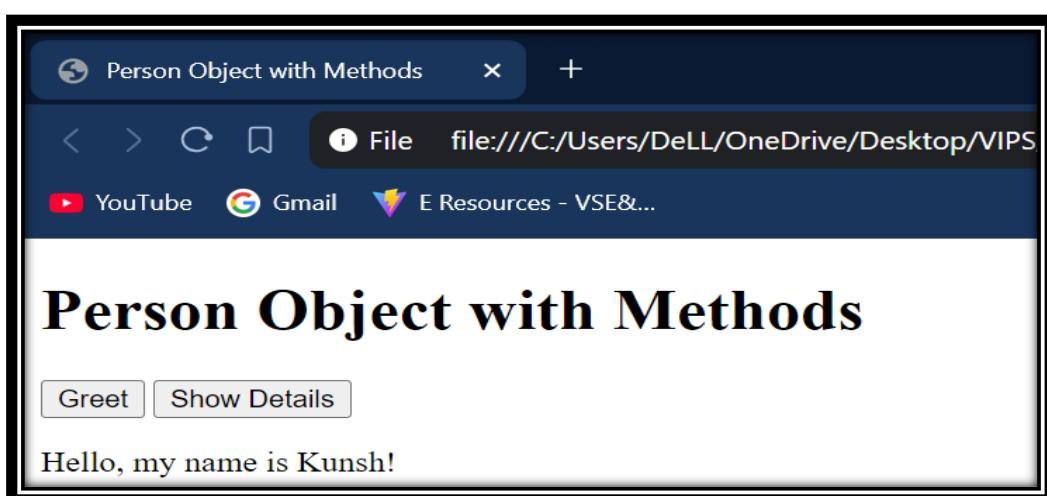
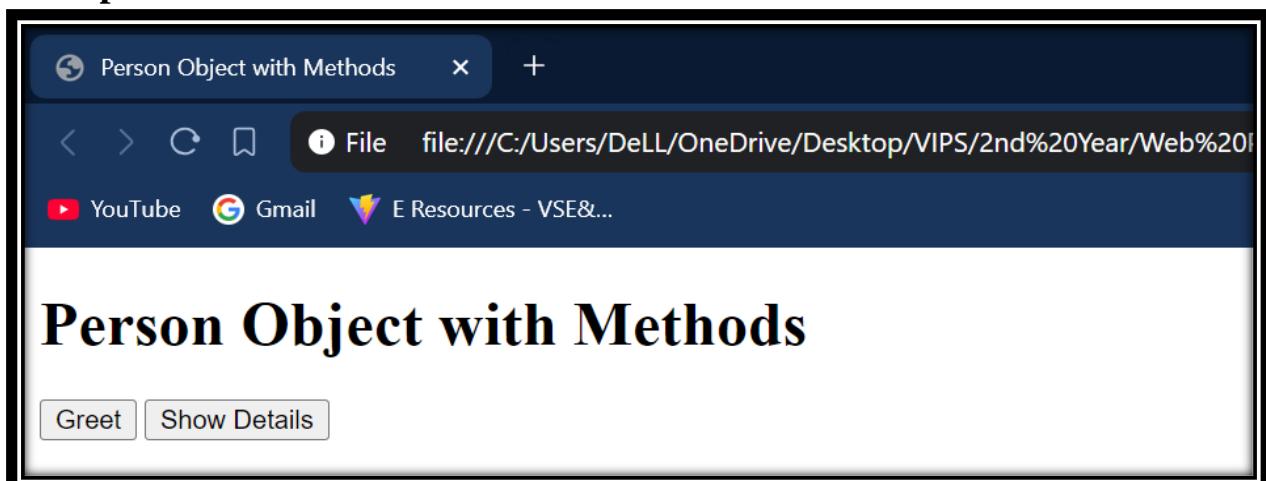
## Source Code:

The screenshot shows a code editor interface with a dark theme. On the left, there is a vertical toolbar with various icons: a file icon, a search icon, a copy/paste icon, a refresh icon, a file list icon, a bold/italic icon, a comment icon, a question mark icon, and a gear icon. The main area displays a file titled "JS Person Object with Methods\_Kunsh Sabharwal.html". The code is as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Person Object with Methods</title>
</head>
<body>
    <h1>Person Object with Methods</h1>
    <button id="greetButton">Greet</button>
    <button id="showDetailsButton">Show Details</button>
    <p id="output"></p>
    <script>
        const person = {
            name: 'Kunsh',
            age: 19,
            occupation: 'Student',
            greet: function() {
                return `Hello, my name is ${this.name}!`;
            },
            showDetails: function() {
                return `Name: ${this.name}, Age: ${this.age}, Occupation: ${this.occupation}`;
            }
        };
        document.getElementById('greetButton').addEventListener('click', function() {
            document.getElementById('output').textContent = person.greet();
        });
        document.getElementById('showDetailsButton').addEventListener('click', function() {
            document.getElementById('output').textContent = person.showDetails();
        });
    </script>
</body>
</html>
```

The code defines a JavaScript object named "person" with properties "name", "age", and "occupation", and methods "greet" and "showDetails". It also adds event listeners to two buttons to update the content of a paragraph element with the results of these methods.

**Outputs:**



**Learning Outcome:**

## EXPERIMENT 8

**Problem statement:** Create a JavaScript application with Loops to incorporate the concept of Iteration.

**Theory:**

## Source Code:

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help, back arrow, forward arrow.
- Title Bar:** JS Loops for Iteration\_Kunsh Sabharwal.html
- Sidebar Icons:** Document (1), Magnifying glass, Network, Refresh, Folders, Bold (B), User, Gear.
- Code Area:**

```
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>Loop Iteration Example</title>
7      <style>
8          body {
9              text-align: center;
10             margin: 50px;
11             font-family: Arial, sans-serif;
12             background-color: #f0f8ff;
13             color: #333;
14         }
15         h1 {
16             color: #70d7f0;
17             font-size: 2.5em;
18             margin-bottom: 20px;
19         }
20         button {
21             padding: 10px 20px;
22             font-size: 1.2em;
23             color: white;
24             background-color: #70d7f0;
25             border: none;
26             cursor: pointer;
27             border-radius: 5px;
28             transition: background-color 0.3s ease;
29         }
30         button:hover {
31             background-color: #70d7f0;
32         }
33         ul {
34             list-style-type: none;
35             padding: 0;
```
- Bottom Bar:** X, 0, ▲, 0, Add Logs, Improve Code, Version Control, Share Code Link.

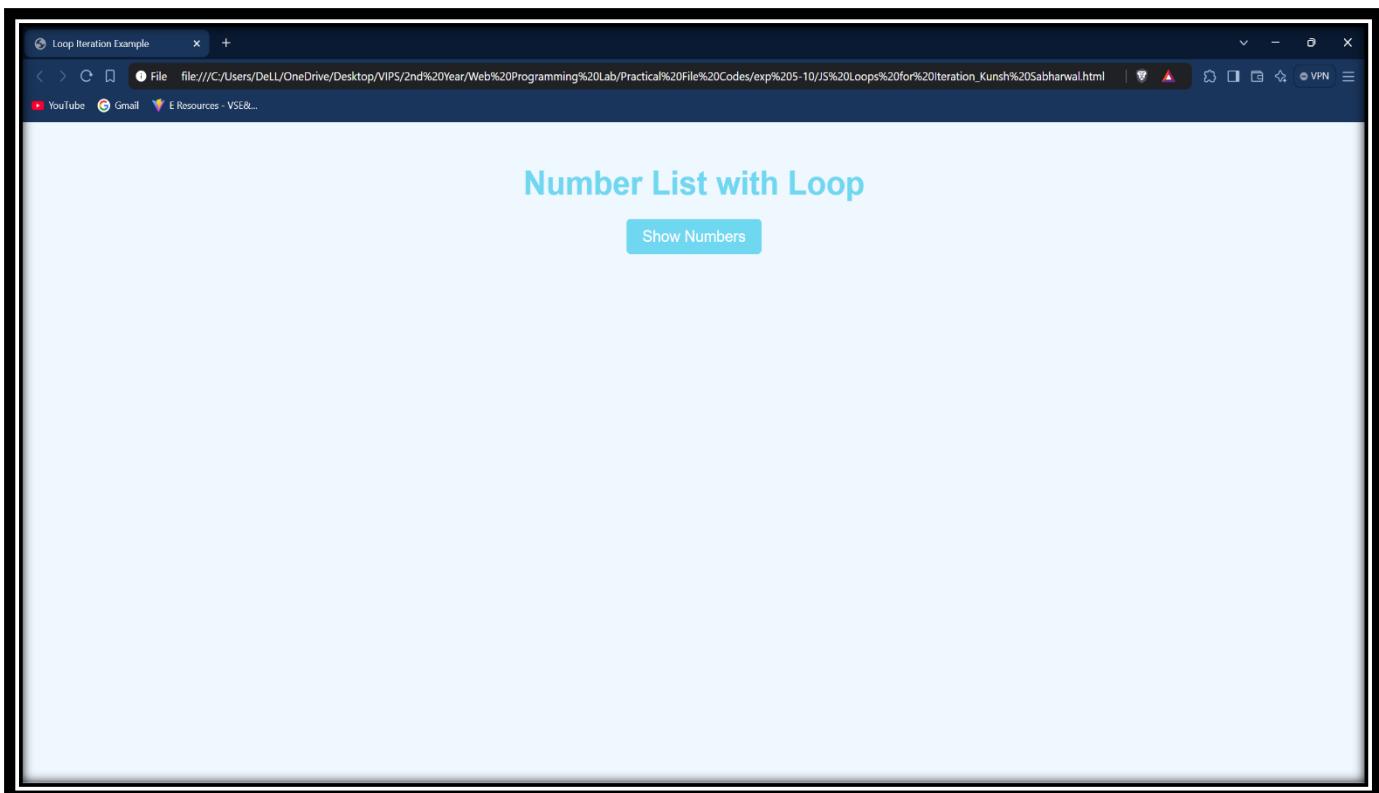
The screenshot shows a code editor interface with a dark theme. On the left, there is a vertical toolbar with various icons: a file icon, a search icon, a refresh icon, a copy/paste icon, a zoom icon, a file list icon, a bold/italic icon, a user profile icon, and a gear settings icon.

The main area displays an HTML file titled "JS Loops for Iteration\_Kunsh Sabharwal.html". The code includes CSS styles for an ul and li elements, and a JavaScript script that creates a numbered list from 1 to 10 when the "Show Numbers" button is clicked.

```
32 }  
33     ul {  
34         list-style-type: none;  
35         padding: 0;  
36     }  
37     li {  
38         padding: 10px;  
39         margin: 5px 0;  
40         font-size: 1.2em;  
41         background-color: #f9f9f9;  
42         border-radius: 5px;  
43         border: 1px solid #ddd;  
44     }  
45     li:nth-child(odd) {  
46         background-color: #e0f1fa;  
47     }  
48 </style>  
49 </head>  
50 <body>  
51     <h1>Number List with Loop</h1>  
52     <button id="iterateButton">Show Numbers</button>  
53     <ul id="numberList"></ul>  
54     <script>  
55         document.getElementById('iterateButton').addEventListener('click', function() {  
56             document.getElementById('numberList').innerHTML = '';  
57             for (let i = 1; i <= 10; i++) {  
58                 let listItem = document.createElement('li');  
59                 listItem.textContent = `Number: ${i}`;  
60                 document.getElementById('numberList').appendChild(listItem);  
61             }  
62         });  
63     </script>  
64 </body>  
65 </html>
```

At the bottom of the editor, there are several status indicators: a file icon, a search icon, a refresh icon, a file list icon, a bold/italic icon, an "Add Logs" button, an "Improve Code" button, a "Version Control" button, and a "Share Code Link" button.

## Outputs:



## Learning Outcome:

## EXPERIMENT 9

**Problem statement:** Create a JavaScript application for random-number generation.

**Theory:**

## Source Code:

The screenshot shows a code editor interface with a dark theme. On the left, there is a vertical toolbar with various icons: a file icon, a search icon, a copy/paste icon, a refresh icon, a file list icon, a bold/italic icon, a comment icon, a user icon with a '1' notification, and a gear icon.

The main area displays an HTML file named "JS Random Number Generation\_Kunsh Sabharwal.html". The code contains HTML structure and CSS styles for a random number generator.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Random Number Generator</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
            background-color: #dcdcdc;
            margin: 50px;
            color: #333;
        }
        h1 {
            color: #0051ff;
        }
        input {
            padding: 10px;
            margin: 10px;
            font-size: 1em;
            border-radius: 5px;
            border: 1px solid #ccc;
        }
        button {
            padding: 10px 20px;
            font-size: 1.2em;
            background-color: #00ffff;
            color: rgb(51, 0, 255);
            border: none;
            cursor: pointer;
            border-radius: 5px;
            transition: background-color 0.3s ease;
        }
        button:hover {
    
```

At the bottom of the editor, there are several status indicators: a file icon with '0' changes, a warning icon with '0' errors, an 'Add Logs' button, an 'Improve Code' button, a 'Version Control' button, and a 'Share Code Link' button.

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help, a back arrow, a forward arrow, and a search bar containing "exp 5-10".
- Title Bar:** JS Random Number Generation\_Kunsh Sabharwal.html X
- Code Area:** The code is written in HTML and JavaScript. It includes a CSS style block for a button hover effect, an HTML structure with input fields for minimum and maximum values, and a script block containing a function to generate random numbers and an event listener for the generate button.

```
button:hover {
    background-color: #00ffff;
}
#result {
    margin-top: 20px;
    font-size: 1.5em;
    font-weight: bold;
}
</style>
</head>
<body>
    <h1>Random Number Generator</h1>
    <label for="minValue">Enter Minimum Value:</label>
    <input type="number" id="minValue" placeholder="Min Value">
    <label for="maxValue">Enter Maximum Value:</label>
    <input type="number" id="maxValue" placeholder="Max Value">
    <button id="generateButton">Generate Random Number</button>
    <div id="result"></div>
    <script>
        function getRandomNumber(min, max) {
            return Math.floor(Math.random() * (max - min + 1)) + min;
        }
        document.getElementById('generateButton').addEventListener('click', function() {
            const min = parseInt(document.getElementById('minValue').value);
            const max = parseInt(document.getElementById('maxValue').value);
            if (isNaN(min) || isNaN(max) || min > max) {
                document.getElementById('result').textContent = 'Please enter valid numbers with Min <= Max.';
            } else {
                const randomNumber = getRandomNumber(min, max);
                document.getElementById('result').textContent = `Random Number: ${randomNumber}`;
            }
        });
    </script>
</body>
</html>
```

- Sidebar Icons:** A vertical sidebar on the left contains icons for file operations like copy, paste, search, refresh, and settings.
- Bottom Bar:** Icons for file operations (New, Open, Save, etc.), a gear icon, and buttons for Add Logs, Improve Code, Version Control, Share Code Link, Ln 29, Col 20, and Spaces.

## Outputs:

A screenshot of a web browser window titled "Random Number Generator". The address bar shows the URL: file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/exp%205-10/J5%20Random%20Number%20Generation\_Kunsh%20Sabharwal.html. The page content includes a title "Random Number Generator" and two input fields: "Enter Minimum Value:" with "Min Value" and "Enter Maximum Value:" with "Max Value". A blue button labeled "Generate Random Number" is positioned to the right of the maximum value field.

A screenshot of a web browser window titled "Random Number Generator". The address bar shows the URL: file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/exp%205-10/J5%20Random%20Number%20Generation\_Kunsh%20Sabharwal.html. The page content includes a title "Random Number Generator" and two input fields: "Enter Minimum Value:" with "0" and "Enter Maximum Value:" with "100". A blue button labeled "Generate Random Number" is positioned to the right of the maximum value field.

A screenshot of a web browser window titled "Random Number Generator". The address bar shows the URL: file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/exp%205-10/J5%20Random%20Number%20Generation\_Kunsh%20Sabharwal.html. The page content includes a title "Random Number Generator" and two input fields: "Enter Minimum Value:" with "0" and "Enter Maximum Value:" with "100". Below the input fields, the text "Random Number: 63" is displayed in bold black font. A blue button labeled "Generate Random Number" is positioned to the right of the maximum value field.

A screenshot of a web browser window titled "Random Number Generator". The address bar shows the URL: file:///C/Users/Dell/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/exp%205-10/J5%20Random%20Number%20Generation\_Kunsh%20Sabharwal.html. The page content includes a title "Random Number Generator" and two input fields: "Enter Minimum Value:" with "0" and "Enter Maximum Value:" with "100". Below the input fields, the text "Random Number: 26" is displayed in bold black font. A blue button labeled "Generate Random Number" is positioned to the right of the maximum value field.

## Learning Outcome:

## **EXPERIMENT 10**

**Problem statement:** Build a unit convertor application using HTML & JavaScript.

**Theory:**

## Source Code:

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Includes icons for Close, Save, Undo, Redo, Find, Replace, Copy, Paste, Select All, Cut, Copy, Paste, Find, Replace, and Share.
- Left Sidebar:** Contains icons for File, Search, Project, Refresh, and a gear icon with a '1' notification.
- Code Area:** Displays the HTML and CSS code for a unit converter. The code includes DOCTYPE, meta tags for charset and viewport, a title, and styles for body, h1, select/input, and button elements.
- Bottom Bar:** Includes buttons for X, Save (with version 0), Add Logs, Improve Code, Version Control, and Share Code Link.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Unit Converter</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
            background-color: #e6e6e6;
            margin: 50px;
        }
        h1 {
            color: #006aff;
        }
        select, input {
            padding: 10px;
            margin: 10px;
            font-size: 1em;
            border-radius: 5px;
            border: 1px solid #ccc;
        }
        button {
            padding: 10px 20px;
            font-size: 1.2em;
            background-color: #00fbff;
            color: black;
            border: none;
            cursor: pointer;
            border-radius: 5px;
            transition: background-color 0.3s ease;
        }
        button:hover {
            background-color: #009be3;
        }
    </style>
</head>
<body>
    <h1>Unit Converter</h1>
    <form>
        <label>Length:</label>
        <input type="text" value="100" />
        <select>
            <option>Centimeters</option>
            <option>Inches</option>
            <option>Meters</option>
            <option>Feet</option>
        </select>
        <label>To:</label>
        <select>
            <option>Inches</option>
            <option>Centimeters</option>
            <option>Meters</option>
            <option>Feet</option>
        </select>
        <button>Convert</button>
    </form>
</body>
</html>
```

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Includes icons for file operations (New, Open, Save, Find, Copy, Paste, Select All, Undo, Redo), a search icon, a source control icon (Source Control (Ctrl+Shift+G)), a refresh icon, and a help icon.
- Code Area:** The main area displays the content of a file named "JS Unit Converter\_Kunsh Sabharwal.html".

```
36      }
37      #result {
38          margin-top: 20px;
39          font-size: 1.5em;
40          font-weight: bold;
41      }
42  </style>
43 </head>
44 <body>
45     <h1>Unit Converter</h1>
46     <label for="inputValue">Enter Value:</label>
47     <input type="number" id="inputValue" placeholder="Enter a number">
48     <label for="fromUnit">From:</label>
49     <select id="fromUnit">
50         <option value="meters">Meters</option>
51         <option value="kilometers">Kilometers</option>
52         <option value="feet">Feet</option>
53         <option value="miles">Miles</option>
54     </select>
55     <label for="toUnit">To:</label>
56     <select id="toUnit">
57         <option value="meters">Meters</option>
58         <option value="kilometers">Kilometers</option>
59         <option value="feet">Feet</option>
60         <option value="miles">Miles</option>
61     </select>
62     <button id="convertButton">Convert</button>
63     <div id="result"></div>
64     <script>
65         const conversionFactors = {
66             meters: {
67                 kilometers: 0.001,
68                 feet: 3.28084,
69                 miles: 0.000621371
70             },
71         }
72     </script>
```
- Status Bar:** Shows file statistics (0△0, 0□0), Add Logs, Improve Code, Version Control, Share Code Link, and a gear icon with a '1' notification.

The screenshot shows a code editor interface with a dark theme. On the left, there's a vertical toolbar with icons for file operations, search, and other functions. The main area displays a JavaScript file named 'JS Unit Converter\_Kunsh Sabharwal.html'. The code is a unit converter script that includes conversion factors for kilometers, meters, feet, and miles, and a function to handle unit conversion events.

```
File Edit Selection View Go Run Terminal Help ← → ⌂ exp 5-10  
1 JS Unit Converter_Kunsh Sabharwal.html  
kilometers: {  
 71   meters: 1000,  
 72   feet: 3280.84,  
 73   miles: 0.621371  
 74 },  
 75 feet: {  
 76   meters: 0.3048,  
 77   kilometers: 0.0003048,  
 78   miles: 0.000189394  
 79 },  
 80 miles: {  
 81   meters: 1609.34,  
 82   kilometers: 1.60934,  
 83   feet: 5280  
 84 }  
 85 };  
 86 function convertUnits(value, fromUnit, toUnit) {  
 87   if (fromUnit === toUnit) {  
 88     return value; // No conversion needed  
 89   }  
 90   return value * conversionFactors[fromUnit][toUnit];  
 91 }  
 92 document.getElementById('convertButton').addEventListener('click', function() {  
 93   const inputValue = parseFloat(document.getElementById('inputValue').value);  
 94   const fromUnit = document.getElementById('fromUnit').value;  
 95   const toUnit = document.getElementById('toUnit').value;  
 96   if (isNaN(inputValue)) {  
 97     document.getElementById('result').textContent = 'Please enter a valid number.';  
 98     return;  
 99   }  
100   const convertedValue = convertUnits(inputValue, fromUnit, toUnit);  
101   document.getElementById('result').textContent = `${inputValue} ${fromUnit} = ${convertedValue.toFixed(4)} ${toUnit}`;  
102 });  
103 </script>  
104 </body>  
105 </html>
```

Ln 103, Col 14 Spaces:4 UTF-8 LF {} HTML

## Outputs:

The screenshot shows a series of five identical browser windows, each displaying a "Unit Converter" application. The application has a light gray background with a dark blue header bar. The header bar contains the title "Unit Converter" and the URL "file:///C:/Users/DeLL/OneDrive/Desktop/VIPS/2nd%20Year/Web%20Programming%20Lab/Practical%20File%20Codes/exp%205-10/J5%20Unit%20Converter\_Kunsh%20Sabharwal.html". Below the header are standard browser controls (back, forward, search, etc.). The main content area is titled "Unit Converter" in blue. It features a text input field labeled "Enter Value:" with the placeholder "Enter a number", a dropdown menu labeled "From:" set to "Meters", a dropdown menu labeled "To:" set to "Meters", and a teal-colored "Convert" button. In the first window, all fields are empty. In the second window, the value "1000" is entered into the "Enter Value:" field, and the "From:" and "To:" dropdowns are still set to "Meters". In the third window, the "From:" dropdown is changed to "Kilometers" and the "To:" dropdown is changed to "Feet". In the fourth window, the "From:" dropdown is changed to "Miles" and the "To:" dropdown is changed to "Kilometers". In the fifth window, the "From:" dropdown is changed to "Feet" and the "To:" dropdown is changed to "Meters". Each window also displays a conversion result below the input fields.

Unit Converter

Enter Value: Enter a number From: Meters To: Meters Convert

1000 meters = 1.0000 kilometers

Unit Converter

Enter Value: 1 From: Kilometers To: Feet Convert

1 kilometers = 3280.8400 feet

Unit Converter

Enter Value: 1 From: Miles To: Kilometers Convert

1 miles = 1.6093 kilometers

Unit Converter

Enter Value: 1000 From: Feet To: Meters Convert

1000 feet = 304.8000 meters

## Learning Outcome: