C BYREGOWDA INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Affiliated to Visvesvaraya Technological University

"Jnana Sangama", Belgaum – 560 018.

LABORATORY MANUAL

"WEB TECHNOLOGY LAB - BCS358A"

Semester:V Scheme: CBCS

Prepared By

Asst.Professor ANITHA.P

CSE Department

Scrutinized by

VASUDEVA R

Associate Professor & HOD

NAME: ……

USN: ……

BATCH NO: ……..

C BYREGOWDA INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

An ISO 9001:2015 Certified Institute

Kolar – Srinivaspur Road,

Kolar – 563101

2024-2025

WEB TECHNOLOGY LAB SEMESTER – V		Semester	5
Course Code	**BCSL504**	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	100
Examination type (SEE)	Practical		

Course objectives:

- Learn HTML 5 elements and their use.
- Use of CSS for enhanced user interface presentation.
- Gain knowledge of JavaScript, AJAX and jQuery for dynamic presentation.
- Use of PHP to build Web applications.
- Design and develop Websites and Web applications.

Sl.NO	Experiments
1	Develop the HTML page named as "Myfirstwebpage.html". Add the following tags with relevant content. 1. Set the title of the page as "My First Web Page" 2. Within the body use the following tags: a) Moving text = "Basic HTML Tags" b) Different heading tags (h1 to h6) c) Paragraph d) Horizontal line e) Line Break f) Block Quote g) Pre tag h) Different Logical Style (<b>, <u>, <sub>, <sup> etc.)
2	Develop the HTML page named as "Table.html" to display your class time table. a) Provide the title as Time Table with table header and table footer, row-span and col-span etc. b) Provide various colour options to the cells (Highlight the lab hours and elective hours with different colours.) c) Provide colour options for rows.
3	Develop an external style sheet named as "style.css" and provide different styles for h2, h3, hr, p, div, span, time, img & a tags. Apply different CSS selectors for tags and demonstrate the significance of each.
4	Develop HTML page named as "registration.html" having variety of HTML input elements with background colors, table for alignment & provide font colors & size using CSS styles.
5	Develop HTML page named as "newpaper.html" having variety of HTML semantic elements with background colors, text-colors & size for figure, table, aside, section, article, header, footer… etc.
6	Apply HTML, CSS and JavaScript to design a simple calculator to perform the following operations: sum, product, difference, remainder, quotient, power, square-root and square.
7	Develop JavaScript program (with HTML/CSS) for: a) Converting JSON text to JavaScript Object b) Convert JSON results into a date c) Converting From JSON To CSV and CSV to JSON d) Create hash from string using crypto.createHash() method
8	a. Develop a PHP program (with HTML/CSS) to keep track of the number of visitors visiting the web page and to display this count of visitors, with relevant headings. b. Develop a PHP program (with HTML/CSS) to sort the student records which are stored in the database using selection sort.

9	Develop jQuery script (with HTML/CSS) for: a. Appends the content at the end of the existing paragraph and list. b. Change the state of the element with CSS style using animate() method c. Change the color of any div that is animated.
10	Develop a JavaScript program with Ajax (with HTML/CSS) for: a. Use ajax() method (without Jquery) to add the text content from the text file by sending ajax request. b. Use ajax() method (with Jquery) to add the text content from the text file by sending ajax request. c. Illustrate the use of getJSON() method in jQuery d. Illustrate the use of parseJSON() method to display JSON values.

Programming Assignment (5 marks):
Construct a Website (multiple Web pages) containing 'Resume' and Bio -data by using relevant HTML elements and appropriate styling for presentation with CSS/jQuery/JavaScript. Host the Website on a cloud platform.

Programming Assignment (5 marks): Build a Web application with HTML, CSS, JavaScript, jQuery and PHP for online application/registration form. Form should accept the information and print/display on a browser with formatting/styling upon submission (Button click) on success. Host the application on a cloud platform.

Course outcomes (Course Skill Set):
At the end of the course, the student will be able to:

- Design the experiment for the given problem using HTML, Javascript and CSS.
- Develop the solution for the given real-world problem using jQuery, Ajax and PHP.
- Analyze the results and produce substantial written documentation.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and recordwrite-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experimentwrite-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximummarks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experimentslisted in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and proceduralknowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learningability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is thetotal CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.

- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answerscript to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
Change of experiment is allowed only once and 15% of Marks allotted to the procedure partare to be made zero.
The minimum duration of SEE is 02 hours

Suggested Learning Resources:
Books:

1. Randy Connolly and Ricardo Hoar, Fundamentals of Web Development, 3rd edition, Pearson, 2021
2. Robert W Sebesta, Programming the World Wide Web, 8th Edition, Pearson Education, 2020.

Web Links:

- https://www.w3schools.com/html/default.asp
- https://www.w3schools.com/css/default.asp
- https://www.w3schools.com/js/js_examples.asp
- https://www.geeksforgeeks.org/javascript-examples/
- https://www.w3schools.com/php/default.asp
- https://www.w3schools.com/jquery/default.asp
- https://www.w3schools.com/js/js_ajax_intro.asp
- https://www.geeksforgeeks.org/jquery-tutorial/

INTRODUCTION

The **World Wide Web** (**WWW**), commonly known as **the Web**, is an information system where documents and other web resources are identified by Uniform Resource Locators (URLs, suchas https://www.example.com/), which may be interlinked by hypertext, and are accessible overthe Internet. The resources of the WWW may be accessed by users by a software application called a web browser.

English scientist Tim Berners-Lee invented the World Wide Web in 1989. He wrote the first web browser in 1990 while employed at CERN near Geneva, Switzerland. The browser was released outside CERN in 1991, first to other research institutions starting in January 1991 and then to the general public in August 1991. The World Wide Web has been central to the development of the Information Age and is theprimary tool billions of people use to interact on the Internet.

Web resources may be any type of downloaded media, but web pages are hypertext media that havebeen formatted in Hypertext Markup Language (HTML). Such formatting allows for embedded hyperlinks that contain URLs and permit users to navigate to other web resources. In addition to text, web pages may contain images, video, audio, and software components that are rendered inthe user's web browser as coherent pages of multimedia content.

HTML

The terms Internet and World Wide Web are often used without much distinction. However, the two terms do not mean the same thing. The Internet is a global system of interconnected computer networks. Incontrast, the World Wide Web is a global collection of documents and other resources, linked by hyperlinksand URIs. Web resources are accessed using HTTP or HTTPS, which are application-level Internet protocolsthat use the Internet's transport protocols.

Viewing a web page on the World Wide Web normally begins either by typing the URL of the pageinto a web browser, or by following a hyperlink to that page or resource.

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologiesfor the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and otherobjects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs,lists, links, quotes and other items. HTML elements are delineated by tags, writtenusinganglebrackets.

XHTML

- XHTML stands for Extensible Hyper Text Markup Language
- XHTML is almost identical to HTML 4.01
- XHTML is a stricter and cleaner version of HTML
- XHTML is HTML defined as an XML application
- XML is a markup language where everything must be marked up correctly, which results in "well-formed" documents.
- XML is designed to describe data, and HTML is designed to display data.
- The Most Important Differences:
    - XHTML elements must be properly nested.
    - XHTML elements must always be closed.
    - XHTML elements must be in lowercase.
    - XHTML documents must have one root element.

CSS

- CSS stands for Cascading Style Sheets.
- Styles define how to display HTML elements.
- Styles are normally stored in Style Sheets.
- Styles were added to HTML 4.0 to solve a problem.

- External Style Sheets can save a lot of work.
- External Style Sheets are stored in CSS files.
- Multiple style definitions will cascade into one.

JAVASCRIPT

- JavaScript was designed to add interactivity to HTML pages.
- JavaScript is a scripting language.
- A scripting language is a lightweight programming language.
- JavaScript is usually embedded directly into HTML pages.
- JavaScript is an interpreted language (means that scripts execute without preliminary compilation).
- Everyone can use JavaScript without purchasing a license.

PHP

- PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general- purpose scripting language that is especially suited for web development and can be embedded intoHTML.
- The PHP code is enclosed in special start and end processing instructions <?php and ?> that allowyou to jump into and out of "PHP mode."
- PHP is mainly focused on server-side scripting, so you can do anything any other CGI program cando, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more.

Tag	What it is	When to use it
<a>	Anchor (most commonly a link)	Vital. Use to create links in content. Use the title attribute whenever the contents of the **<a>…** pair do not accuratelydescribe what you'll get from selecting the link. Title attribute often displays as a tooltip in visual browsers, which may be a helpful usability aid.
<abbr>	Defines an abbreviation	Works in a similar way to **<dfn>** and**<acronym>**, using a **title** attribute (displays a tooltip in standard visual browsers). e.g. **<abbr title="Hypertext markup language">HTML</abbr>**
<BLOCKQUOTE>	Large quoted block of text	Use for any quoted text that constitutes one or more paragraphs (note: should contain <p> tags as well). Use **<q>** for quotations within a paragraph. Often used in conjunction with **<cite>** to citethe quotation's source.
<BODY>	Document body	Essential (unless you're using frames)
** **	Line break	This is arguably display information. Still in common use, but usewith restraint.
<B>	Bold text	Display info – never use it
<BUTTON>	Used for a standard clickable buttonwithin a form	Often better than **<input type="button" />** or **<input type="submit" />**, as it allows you to assign different styles basedon the HTML element alone, whereas differentiating style based on the type of input is less well supported.
<CAPTION>	Caption for a table: describesthe table's contents	The correct way to assign a title to a table
<CENTER>	Centred block	Display info – never use it. Use** ** or some other block-leveltag with the style text-align:center instead
<CITE>	Defines a citation	Defines the source of a quotation (in conjunction with contentin **<q>** or**<blockquote>** pairs).
<CODE>	Defines an extract of code	Not commonly used. Similar to**<pre>** tag, but collapsesconsecutive white spaces and line breaks in the source.

<COL>	Identifies a particular column in a table	Can be very useful. e.g. **<col class="namecol">** can be applied to each first column in a series of tables, then the width of each column may be set to be equal in the stylesheet, overriding the table's natural tendency to adjust its own column widths to fit its contents.
<DFN>	Definition of a term	Works in a similar way to **<abbr>**and **<acronym>**, using a **title**attribute (displays a tooltip in standard visual browsers).
<DIR>	Directory list	Now deprecated. Use a standard** ** or other list instead.
<DIV>	Division	Specifies a logical division within a document. Use it to separate or identify chunks of content that are not otherwise distinguished naturally using other tags. One of the most common HTML tags.
<DL>	Definition list	Contains one or more definition-term / definition-description pairs.
<DT>	Definition term	Used as part of a **<dt></dt>**
<DD>	Definition description	**<dd></dd>** pair within a definitionlist (**<dl></dl>**)
<EM>	Emphasis	Commonly used in place of the old**<i>** (italics) tag to indicate emphasis (but less than ****)
<FONT>	Font settings	Display info – never use it
<FORM>	Input form	Essential for data input
<H1>	Level 1 header	Aim to have one H1 on each page, containing a description of what the page is about.
<H2>	Level 2 header	Defines a section of the page
<H3>	Level 3 header	Defines a sub-section of the page (should always follow an H2 in the logical hierarchy)

<H4>	Level 4 header	Etc. Less commonly used
<H5>	Level 5 header	Less commonly used. Only complex academic documents will break down to this level of detail.
<H6>	Level 6 header	Less commonly used
<HEAD>	Document head	Essential. Contains information about a page that does not constitute content to be communicated as part of the page.
<HR>	Horizontal rule	Display info with no semantic value – never use it. “Horizontal”, by definition, is a visual attribute.
<HTML>		Core element of every web page.
<IMG >	Show an image	Vital. Always use the **alt** or **longdesc**attributes when the image has content value
<INPUT>	Input fields within forms	Vital. (I prefer to use **<button>** for buttons and submit buttons though)
<I>	Italicized text	Display info – never use it
<KBD>	Keyboard input	Display info – never use it
<LINK>	Defines a relationship to another document	Commonly used to reference external stylesheets, but has other minor uses
<LI>	List item	Specifies an item in an unordered or ordered list (** ** or ** **)
<MAP>	Client-side image map	May have occasional value, but only use when absolutely necessary
<META>	Meta-information	Useful way to insert relevant information into the <head> section of the page that does not need to be displayed.

<OL>	Ordered list	Type of list where the order of elements has some meaning. Generally rendered with item numbers (best managed with CSS).
<OPTION>	Selection list option	Vital for options within a drop-down control.
<PRE>	Preformatted text	Renders text in a pre-formatted style, preserving line breaks and all spaces present in the source. May be useful. *(This one's a paradox, as it is strictly display info that applies only to visual browsing, but it's still so commonly used and useful that I'm hesitant to advise against using it.)*
<P>	Paragraph	Only use to denote a paragraph of text. Never use for spacing alone.
<Q>	Short quotation	Use for inline quotations (whereas**<blockquote>** should be used for quotations of a paragraph or more). Often used in conjunction with**<cite>** to cite the quotation's source.
<SCRIPT>	Inline script (e.g. JavaScript)	It's better to have all scripts as separate files than to write inline orin the **<head>** section, however still has its uses.
<SELECT>	Selection list	A drop-down selector for a form.
<SMALL>	Smaller text	Display info – never use it
<SPAN>	An inline span within text	Use to apply meaning (and style) to a span of text that goes with the flow of content (whereas a ** ** tag is block-level and breaksthe flow)
<STRONG>	Strong emphasis	Use this instead of the old **<b>** tag.
<STYLE>	CSS style settings	Normally used in **<head>** section of a page. Try to use external style sheets, to enable you to apply different styles for different output media.
<SUB>	Subscript text	Arguably display info – recommend using alternative tags (e.g. **<cite>**). May be required in some academic uses, e.g. Chemical formulas.
<SUP>	Superscript text	
<TABLE>	Table	Use for repeated data that has a naturally tabular form. Never use for layout purposes.

<TD>	Table data cell	A cell containing actual data. If a cell actually contains a descriptor or identifier for a row or column, use a**<th>** (table header) tag, not a**<td>.** This usually applies to column headers (within a **<thead>**), column footers (within a **<tfoot>**), as well as row headers (usually the first cell in a row in the **<tbody>**).
<TEXTAREA>	Multi-line text input area in a form	Essential
<TH>	Table column or row header cell	May appear in a **<thead>** (to denote a column header cell), **<tbody>** (to denote a row header), and in**<tfoot>** (to denote a column foot cell, e.g. a total)
<TBODY>	Indicates the main body of a data table	It is always worth using this tag, as well as using **<thead>** and **<tfoot>**where appropriate. Note that it is permissible to have more than one **<tbody>**, **<thead>**, and **<tfoot>** in the same table.
<THEAD>	The head section of a table	The place to put column header cells (**<th>**)
<TFOOT>	The foot section of a table	Good place to put e.g. summary data, such as totals. Note that it goes before the **<tbody>** tag!
<TITLE>	Document title	Essential
<TR>	Table row	Essential with tables
<TT>	"Teletype" - simulates typewriter output	Similar to <pre>, except that it collapses white space like normal HTML (whereas <pre> leaves all consecutive white space intact). Avoid if possible
<UL>	Unordered list	Essential. Use for lists where the order or items has no particular importance.
<U>	Underline text	Display info – never use it
<VAR>	Variable in computer code	Obscure tag, may only be useful in academic documents. Avoid.

PROCEDURE TO EXCUTE THE PROGRAMS ON LINUX

Step1: Create on my computer → File System → var →www

- Create **.html** or **.php** or **.js** file under the directory html by specifying the file name with extension.

Step2: Save

- **.html** and **.php** files in html directory

Step3: For Database Programs

- Open command prompt
  - To establish connection with database
    - mysql --user= root --password=server
  - Create database

Step4: For php Programs

- Open Command prompt
  - cd /var/www/html
  - php filename.php

Step5: To Execute the program Open browser and give URL as

- For.html files
  ohttp://localhost/filename.html
- For .php files
  - http://localhost/filename.php

PROCEDURE TO EXCUTE THE PROGRAMS ON WINDOWS

1. Steps to Run HTML and Java Script

- open blank page in Notepad or Notepad++
- Save the HTML file with file extension as .html and select file type as all types or all files.
- Save the CSS file with file extension as .css and select file type as all types or all files.
- Open the file in the browser

2. Steps to run PHP Programs in XAMPP

- Double Click on XAMPP
- Click on Start Button for Apache Server
- Place your PHP file in the "htdocs" folder located under the XAMPP folder in your C: drive
- The file path is C:\xampp\htdocs for your web server.
- Make sure your PHP files are saved as such,they should have the .php file extension .
- Open up any web browser on your desktop and enter "localhost" into the address box
- http://localhost/filename.php

3. Steps to run Database Program

- Create Database in phpmyadmin. – localhost/phpmyadmin
- Create Tables
- Place your PHP file in the "htdocs" folder located under the XAMPP folder in your C: drive
- The file path is C:\xampp\htdocs for your web server.
- Make sure your PHP files are saved as such, they should have the .php file extension .
- Open up any web browser on your desktop and enter "localhost" into the address box
- http://localhost/filename.php

Program 1

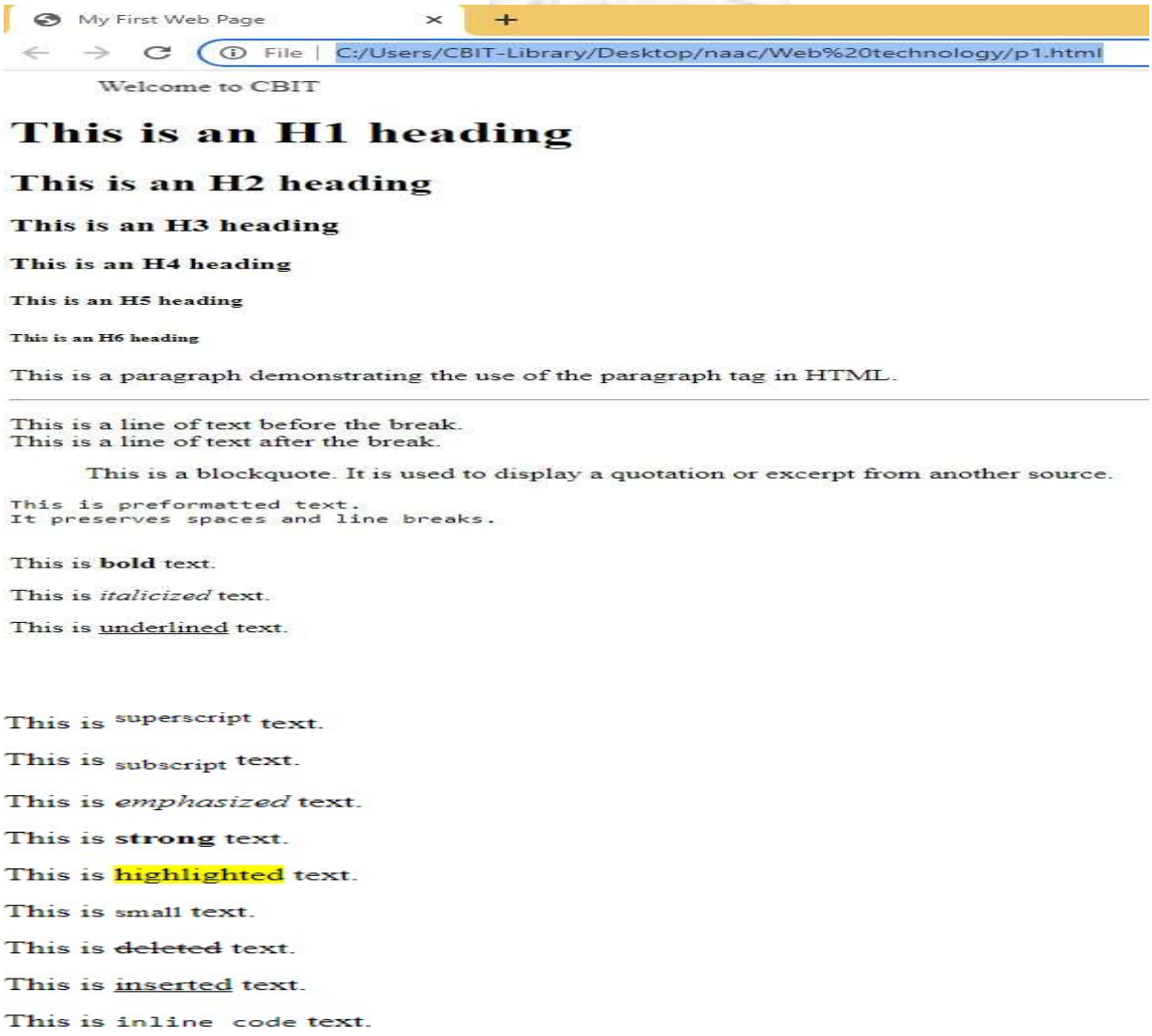
1. Develop the HTML page named as “Myfirstwebpage.html”. Add the following tags with relevant content.

1. Set the title of the page as “My First Web Page”
2. Within the body use the following tags:
a) Moving text = “Basic HTML Tags”
b) Different heading tags (h1 to h6)
c) Paragraph
d) Horizontal line
e) Line Break
f) Block Quote
g) Pre tag
h) Different Logical Style (< b >, < u >, < sub >, < sup > etc.)

```
<!DOCTYPE html>

<head>
   <title>My First Web Page </title>
</head>
<body>
   <!-- Moving text -->
   <marquee>Welcome to  CBIT</marquee>
   <!-- Different heading tags -->
   <h1>This is an H1 heading</h1>
   <h2>This is an H2 heading</h2>
   <h3>This is an H3 heading</h3>
   <h4>This is an H4 heading</h4>
   <h5>This is an H5 heading</h5>
   <h6>This is an H6 heading</h6>
   <!-- Paragraph -->
   <p>This is a paragraph demonstrating the use of the paragraph tag in HTML.</p>
   <!-- Horizontal line -->
   <hr>
   <!-- Line break -->
   <p>This is a line of text before the break.<br>This is a line of text after the break.</p>
   <!-- Block Quote -->
   <blockquote>
      This is a blockquote. It is used to display a quotation or excerpt from another source.
   </blockquote>
   <!-- Pre tag -->
   <pre>
This is preformatted text.
It preserves spaces and line breaks.
   </pre>
   <!-- Different Logical Style tags -->
   <p>This is <b>bold</b> text.</p>
```

```
    <p>This is <i>italicized</i> text.</p>
    <p>This is <u>underlined</u> text.</p>
    <p>This is <sup>superscript</sup> text.</p>
    <p>This is <sub>subscript</sub> text.</p>
    <p>This is <em>emphasized</em> text.</p>
    <p>This is <strong>strong</strong> text.</p>
    <p>This is <mark>highlighted</mark> text.</p>
    <p>This is <small>small</small> text.</p>
    <p>This is <del>deleted</del> text.</p>
    <p>This is <ins>inserted</ins> text.</p>
    <p>This is <code>inline code</code> text.</p>
</body>
</html>
```

Output:



Program.2

Develop the HTML page named as "Table.html" to display your class time table.

a) Provide the title as Time Table with table header and table footer, row-span and col-span etc.
b) Provide various colour options to the cells (Highlight the lab hours and elective hours with different colours.)
c) Provide colour options for rows.

```
<!DOCTYPE html>
<head>
   <title>Time Table </title>
   <style>
      body {
         font-family: Arial, sans-serif;
      }

      table {
         width: 80%;
         margin: 20px auto;
         border-collapse: collapse;
      }

      th,
      td {
         border: 1px solid #ddd;
         padding: 8px;
         text-align: center;
      }

      th {
         background-color: #f4f4f4;
         color: #333;
      }

      tr:nth-child(even) {
         background-color: #f9f9f9;
      }
      tr:nth-child(odd) {
         background-color: #e6f7ff;
      }
      .lab-hour {
         background-color: #ffcccc;
      }

      .elective-hour {
         background-color: #ccffcc;
```

```
        }
        .highlight {
            font-weight: bold;
            color: #d63384;
        }
        tfoot {
            background-color: #e0e0e0;
            font-weight: bold;
        }
    </style>
</head>
<body>
    <h1 style="text-align: center;">Time Table</h1>
    <table>
        <thead>
            <tr>
                <th>Day/Time</th>
                <th>9:00 - 10:00</th>
                <th>10:00 - 11:00</th>
                <th>11:00 - 12:00</th>
                <th>12:00 - 1:00</th>
                <th>Lunch Break</th>
                <th>2:00 - 3:00</th>
                <th>3:00 - 4:00</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>Monday</td>
                <td>Math</td>
                <td>English</td>
                <td class="lab-hour">Physics Lab</td>
                <td>Elective</td>
                <td rowspan="5" class="highlight">Break</td>
                <td class="elective-hour">Elective</td>
                <td>History</td>
            </tr>
            <tr>
                <td>Tuesday</td>
                <td class="elective-hour">Elective</td>
                <td>Biology</td>
                <td>Math</td>
                <td class="lab-hour">Chemistry Lab</td>
                <td>Geography</td>
                <td>PE</td>
            </tr>
            <tr>
                <td>Wednesday</td>
                <td>History</td>
```

```
            <td class="lab-hour">Computer Lab</td>
            <td>English</td>
            <td>Math</td>
            <td>Physics</td>
            <td class="elective-hour">Elective</td>
          </tr>
          <tr>
            <td>Thursday</td>
            <td>PE</td>
            <td>History</td>
            <td>Geography</td>
            <td class="elective-hour">Elective</td>
            <td>Biology</td>
            <td>Math</td>
          </tr>
          <tr>
            <td>Friday</td>
            <td class="lab-hour">Biology Lab</td>
            <td>Math</td>
            <td>English</td>
            <td>Physics</td>
            <td class="elective-hour">Elective</td>
            <td>Chemistry</td>
          </tr>
        </tbody>
        <tfoot>
          <tr>
            <td colspan="8">End of Timetable</td>
          </tr>
        </tfoot>
    </table>
</body>
</html>
```

Output:

Time Table

Day/Time	9:00 - 10:00	10:00 - 11:00	11:00 - 12:00	12:00 - 1:00	Lunch Break	2:00 - 3:00	3:00 - 4:00
Monday	Math	English	Physics Lab	Elective		Elective	History
Tuesday	Elective	Biology	Math	Chemistry Lab		Geography	PE
Wednesday	History	Computer Lab	English	Math	Break	Physics	Elective
Thursday	PE	History	Geography	Elective		Biology	Math
Friday	Biology Lab	Math	English	Physics		Elective	Chemistry
End of Timetable							

Program 3

Develop an external style sheet named as “style.css” and provide different styles for h2, h3, hr, p, div, span, time, img & a tags. Apply different CSS selectors for tags and demonstrate the significance of each.

```
<!DOCTYPE html>

<head>
   <title>Styled HTML Page</title>
   <link rel="stylesheet" href="style.css">
</head>
<body>

   <h2>Main Heading</h2>
   <h3>Subheading</h3>
   <hr>
   <p>This is a paragraph demonstrating the basic text styling applied by CSS.</p>

   <div>
      This is a styled <strong>div</strong> element with padding and a light border. Inside the div, we can also use
      <span>span elements</span> that have their own styles, like this bold and orange text.
   </div>

   <p>Current time: <time>10:30 AM</time></p>

   <img src="cbit image.jpg" alt="CBIT College Image">

   <p>Visit <a href="https://cbitkolar.edu.in ">cbitkolar.edu.in </a> to learn more about our services.</p>

   <p class="highlight">This paragraph is highlighted with a yellow background.</p>
   <p class="center">This text is centered using a class selector.</p>

   <p id="special-paragraph">This is a special paragraph with unique styles applied through an ID selector.</p>

</body>

</html>
```

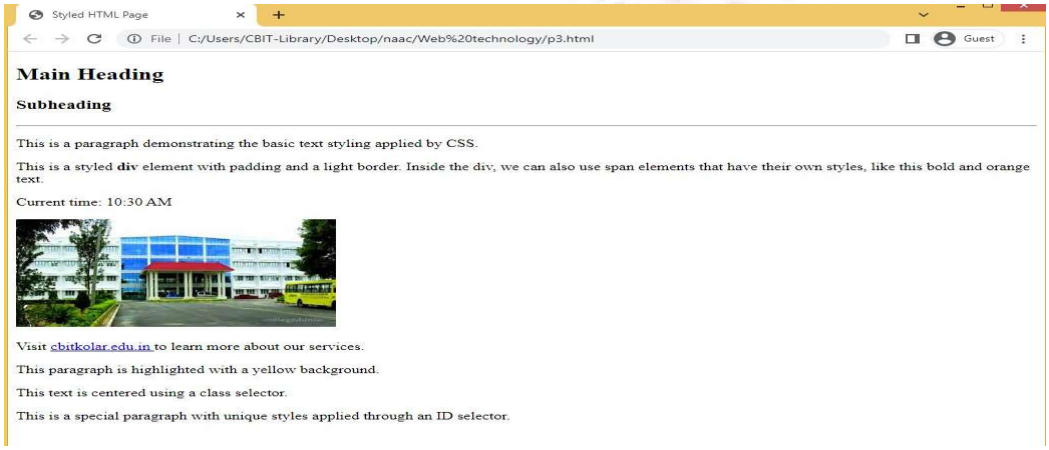
Style.css

```
{
   margin: 0;
   padding: 0;
   box-sizing: border-box;
}
```

```
h2 {
    color: #2c3e50;
    font-size: 2em;
    margin-bottom: 10px;
}

h3 {
    color: #34495e;
    font-size: 1.5em;
    margin-bottom: 8px;
}

hr {
    border: 0;
    height: 2px;
    background-color: #e74c3c;
    margin: 20px 0;
}

p {
    font-family: 'Arial', sans-serif;
    line-height: 1.6;
    margin: 10px 0;
}

div {
    padding: 15px;
    border: 1px solid #bdc3c7;
    background-color: #ecf0f1;
}

span {
    color: #e67e22;
    font-weight: bold;
}
time::before {
    content: '⏰ ';
    color: #16a085;
}
img {
    margin-left: 15px;
    height: 300px;
    width: 400px;
    border-radius: 8px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
    max-width: 100%;
}
a {
    text-decoration: none;
```

```
    color: #ea0e4c;
}
a:hover {
    color: #6200ee;
    text-decoration: underline;
}

.highlight {
    background-color: yellow;
    padding: 3px;
}

.center {
    text-align: center;
}

#special-paragraph {
    font-size: 1.2em;
    color: #8e44ad;
    background-color: #f5f5f5;
    padding: 10px;
    border-left: 5px solid #8e44ad;
}

h2,
h3,
p {
    margin-left: 20px;
}
```

Output :



Program 4

Develop HTML page named as "registration.html" having variety of HTML input elements with background colors, table for alignment & provide font colors & size using CSS styles.

```
<!DOCTYPE html>
<head>
   <title>Registration Form </title>
   <style>
      body {
         font-family: Arial, sans-serif;
         background-color: #f0f4f8;
         margin: 0;
         padding: 20px;
         display: flex;
         justify-content: center;
         align-items: center;
         min-height: 100vh;
      }

      .container {
         width: 100%;
         max-width: 600px;
         background-color: #fff;
         padding: 20px;
         border-radius: 8px;
         box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
         display: flex;
         flex-direction: column;
         gap: 20px;
      }

      h2 {
         text-align: center;
         color: #333;
         margin: 0;
      }

      .form-group {
         display: flex;
         flex-direction: column;
         gap: 5px;
         margin-bottom: 10px;
      }

      label {
```

```
    font-size: 14px;
    color: #555;
}

input[type="text"],
input[type="email"],
input[type="password"],
input[type="date"],
select,
textarea {
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
    font-size: 14px;
}

.gender-options {
    display: flex;
    gap: 10px;
    align-items: center;
}

input[type="submit"],
input[type="reset"] {
    padding: 10px 20px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    font-size: 16px;
    flex: 1;
}

.button-group {
    display: flex;
    gap: 10px;
    justify-content: center;
}

input[type="submit"] {
    background-color: #4CAF50;
    color: white;
}

input[type="reset"] {
    background-color: #f44336;
    color: white;
}

.form-group textarea {
```

```
            margin-bottom: 10px;
        }
    </style>
</head>

<body>
    <div class="container">
        <h2>Registration Form</h2>
        <form action="#" method="post">
            <div class="form-group">
                <label for="firstName">First Name:</label>
                <input type="text" id="firstName" name="firstName" required>
            </div>
            <div class="form-group">
                <label for="lastName">Last Name:</label>
                <input type="text" id="lastName" name="lastName" required>
            </div>
            <div class="form-group">
                <label for="email">Email:</label>
                <input type="email" id="email" name="email" required>
            </div>
            <div class="form-group">
                <label for="password">Password:</label>
                <input type="password" id="password" name="password" required>
            </div>
            <div class="form-group">
                <label for="dob">Date of Birth:</label>
                <input type="date" id="dob" name="dob">
            </div>
            <div class="form-group">
                <label>Gender:</label>
                <div class="gender-options">
                    <input type="radio" id="male" name="gender" value="male">
                    <label for="male">Male</label>
                    <input type="radio" id="female" name="gender" value="female">
                    <label for="female">Female</label>
                </div>
            </div>
            <div class="form-group">
                 <label for="country">Country:</label>
                 <select id="country" name="country">
                     <option value="usa">USA</option>
                    <option value="canada">Canada</option>
                    <option value="uk">UK</option>
                    <option value="india">India</option>
                </select>
            </div>
            <div class="form-group">
                <label for="bio">Bio:</label>
```

```
            <textarea id="bio" name="bio" rows="4"></textarea>
        </div>
        <div class="button-group">
            <input type="submit" value="Register">
            <input type="reset" value="Reset">
        </div>
    </form>
  </div>
</body>

</html>
```

Output :

Program 5

Develop HTML page named as “newpaper.html” having variety of HTML semantic elements with background colors, text-colors & size for figure, table, aside, section, article, header, footer… etc.

```
<!DOCTYPE html>

<head>
   <title>Newspaper Page </title>
   <style>
      * {
         margin: 0;
         padding: 0;
         box-sizing: border-box;
      }

      body {
         padding: 20px;
         font-family: 'Arial', sans-serif;
         color: #000000;
         display: flex;
         flex-direction: column;
         min-height: 100vh;
      }

      header {
         margin-bottom: 30px;
         border-radius: 10px;
         align-items: center;
         background-color: #7b38f7;
         color: #fff;
         padding: 20px;
         display: flex;
         justify-content: space-between;
         text-align: center;
         box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
      }

      header a {
         font-size: 25px;
         font-weight: 600;
         color: #fff;
         text-decoration: none;
      }


      nav {
```

```
    display: flex;
    gap: 20px;
    color: #fff;
    text-align: center;
}

nav a {
    font-size: 18px;
    color: #fff;
    text-decoration: none;
    font-weight: bold;
}

nav a:hover {
    text-decoration: underline;
}

.content {
    display: flex;
    justify-content: space-between;
    flex: 1;
    margin: auto;
    padding: 20px 0;
    gap: 20px;
    position: relative;
}

.main-content {
    cursor: pointer;
    flex: 1;
    display: grid;
    grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));
    gap: 20px;
    background-color: #fff;
    border-radius: 12px;
    padding: 25px;
    box-shadow: rgba(9, 30, 66, 0.25) 0px 1px 1px, rgba(9, 30, 66, 0.13) 0px 0px 1px 1px;
}

aside {
    border: 1px solid #ccc;
    padding: 20px;
    width: 350px;
    border-radius: 8px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    position: -webkit-sticky;
    position: sticky;
    top: 20px;
    color: #333;
```

```
    right: 0;
    margin-left: 20px;
}


.related-news h3 {
    text-align: center;
    border-radius: 7px;
    padding: 8px;
    background: #000;
    color: #ffffff;
    font-size: 1.4em;
    margin-bottom: 15px;
}

.related-news ul {
    list-style: outside;
    padding: 7px;
    margin: 0;
}

.related-news li {
    margin-bottom: 12px;
}

.related-news a {
    text-decoration: none;
    color: #7b38f7;
    font-weight: bold;
    transition: color 0.3s ease;
}

.related-news a:hover {
    text-decoration: underline;
}


footer {
    border-radius: 10px;
    background-color: #7b38f7;
    color: #fff;
    padding: 20px;
    font-weight: 500;
    text-align: center;
    margin-top: auto;
    font-size: 18px;
}

article {
```

```
    transition: all 0.3s ease;
    background-color: #fff;
    padding: 15px;
    box-shadow: rgba(9, 30, 66, 0.25) 0px 1px 1px, rgba(9, 30, 66, 0.13) 0px 0px 1px 1px;
    border-radius: 7px;
    color: #000000;
}

figure {
    background-color: #fafafa;
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 8px;
    text-align: center;
    margin: 0;
}

figcaption {
    font-size: 0.9em;
    color: #666;
}

img {
    max-width: 100%;
    height: auto;
    border-radius: 8px;
}

section {
    padding: 20px;
    width: 100%;
    background-color: #fff;
    border-radius: 8px;
    box-shadow: rgba(9, 30, 66, 0.25) 0px 1px 1px, rgba(9, 30, 66, 0.13) 0px 0px 1px 1px;
}

section h2 {
    color: #fff;
    background: #000;
    font-size: 24px;
    border-radius: 10px;
    text-align: center;
    padding: 10px;
    margin-bottom: 30px;
}

table {
    width: 100%;
    border-collapse: collapse;
```

```
}

caption {
    font-size: 18px;
    margin-bottom: 10px;
    color: #555;
}

thead {
    background-color: #007BFF;
    color: #fff;
}

th,
td {
    padding: 12px;
    text-align: left;
}

th {
    font-weight: bold;
}

tbody tr:nth-child(even) {
    background-color: #f9f9f9;
}

tbody tr:hover {
    background-color: #eaeaea;
}

@media (max-width: 600px) {

    th,
    td {
        padding: 8px;
        font-size: 14px;
    }
}
caption {
    background-color: #d9d9d9;
    padding: 10px;
    font-weight: bold;
    border-bottom: 2px solid #ddd;
    border-radius: 8px 8px 0 0;
    font-size: 1.1em;
    color: #333;
}
section {
```

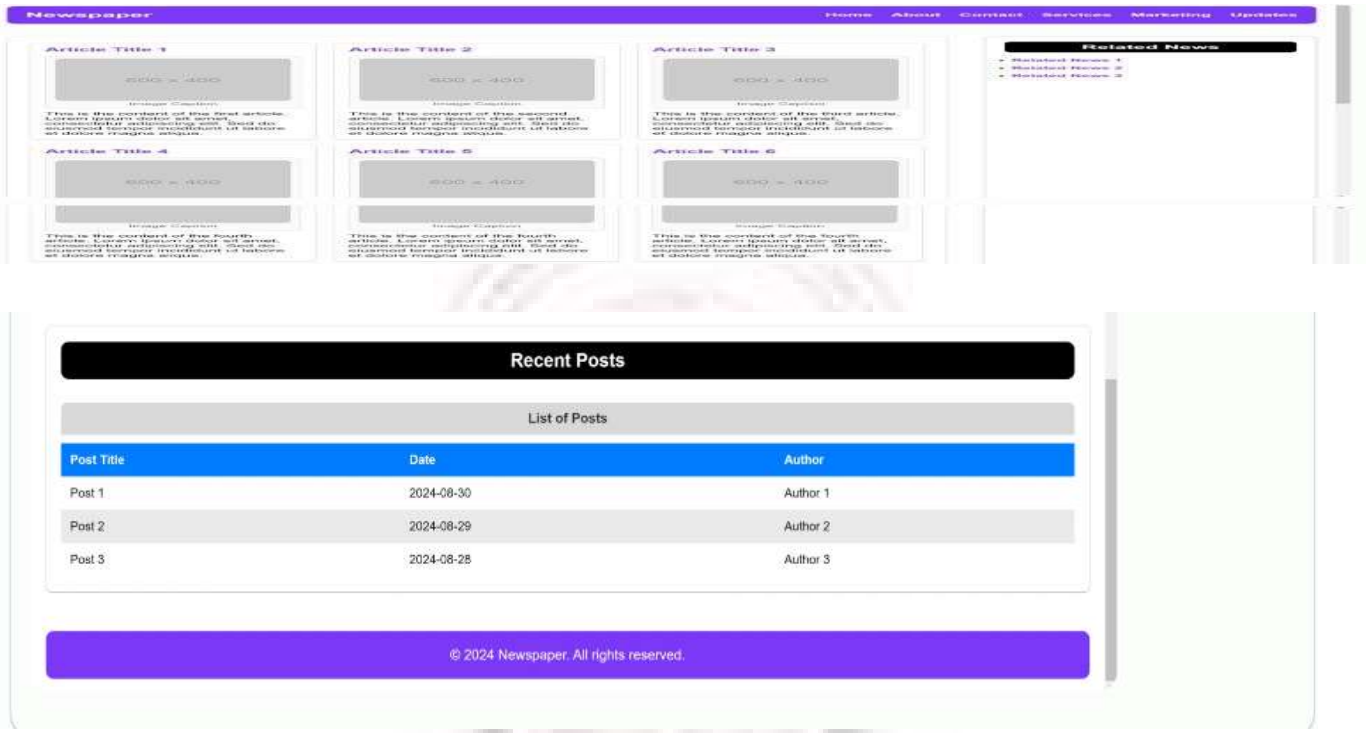
```
        margin-top: 40px;
        margin-bottom: 50px;
    }
    article h2 {
        color: #7b38f7;
        font-size: 1.3em;
        margin-bottom: 12px;
    }
    article p {
        text-align: left;
        line-height: 1.2;
        margin-top: 10px;
    }
    article:hover {
        background-color: #e7ddfb;
    }

    @media (max-width: 768px) {
        .content {
            flex-direction: column;
            padding: 10px;
        }
        aside {
            width: 100%;
            margin-top: 20px;
            position: static;
            margin-left: 0;
        }

        .main-content {
            grid-template-columns: 1fr;
        }
    }
  </style>
</head>
<body>
  <header>

    <a href="#">Newspaper</a>

    <nav>
        <a href="#">Home</a>
        <a href="#">About</a>
        <a href="#">Contact</a>
        <a href="#">Services</a>
        <a href="#">Marketing</a>
        <a href="#">Updates</a>
    </nav>
  </header>
```

```
  <div class="content">
    <main class="main-content">
      <article>
        <h2>Article Title 1</h2>
        <figure>
          <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
          <figcaption>Image Caption</figcaption>
        </figure>
        <p>This is the content of the first article. Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
      </article>

      <article>
        <h2>Article Title 2</h2>
        <figure>
          <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
          <figcaption>Image Caption</figcaption>
        </figure>
        <p>This is the content of the second article. Lorem ipsum dolor sit amet, consectetur adipiscing
elit.Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
      </article>

      <article>
        <h2>Article Title 3</h2>
        <figure>
          <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
          <figcaption>Image Caption</figcaption>
        </figure>
        <p>This is the content of the third article. Lorem ipsum dolor sit amet, consectetur adipiscing
elit.
          Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
      </article>
      <article>
        <h2>Article Title 4</h2>
        <figure>
          <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
          <figcaption>Image Caption</figcaption>
        </figure>
        <p>This is the content of the fourth article. Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
      </article>
      <article>
        <h2>Article Title 5</h2>
        <figure>
          <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
          <figcaption>Image Caption</figcaption>
        </figure>
        <p>This is the content of the fourth article. Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
```

```
        </article>
        <article>
          <h2>Article Title 6</h2>
          <figure>
            <img src="https://via.placeholder.com/600x400" alt="Placeholder Image">
            <figcaption>Image Caption</figcaption>
          </figure>
          <p>This is the content of the fourth article. Lorem ipsum dolor sit amet, consectetur adipiscing
eli Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
        </article>
      </main>
      <aside class="related-news">
        <h3>Related News</h3>
        <ul>
          <li><a href="#">Related News 1</a></li>
          <li><a href="#">Related News 2</a></li>
          <li><a href="#">Related News 3</a></li>
        </ul>
      </aside>
    </div>
    <section>
      <h2>Recent Posts</h2>
      <div>
        <table>
          <caption>List of Posts</caption>
          <thead>
            <tr>
              <th>Post Title</th>
              <th>Date</th>
              <th>Author</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td>Post 1</td>
              <td>2024-08-30</td>
              <td>Author 1</td>
            </tr>
            <tr>
              <td>Post 2</td>
              <td>2024-08-29</td>
              <td>Author 2</td>
            </tr>
            <tr>
              <td>Post 3</td>
              <td>2024-08-28</td>
              <td>Author 3</td>
            </tr>
          </tbody>
```

```
            </table>
        </div>
    </section>
    <footer>
        <p>© 2024 Newspaper. All rights reserved.</p>
    </footer>
</body>
</html>
```

Output :



Program 6

Apply HTML, CSS and JavaScript to design a simple calculator to perform the following operations: sum, product, difference, remainder, quotient, power, square-root and square.

```
<!DOCTYPE html>
<head>
   <title>Simple Calculator </title>
   <style>
      body {
         font-family: 'Arial', sans-serif;
         display: flex;
         justify-content: center;
         align-items: center;
         height: 100vh;
         margin: 0;
      }

      .calculator {
         background: #fff;
         padding: 20px;
         border-radius: 12px;
         box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
         width: 320px;
         text-align: center;
      }

      h1 {
         border-radius: 8px;
         background: #000;
         font-size: 24px;
         padding: 10px;
         color: #ffffff;
         margin-bottom: 30px;
      }

      input,
      select,
      button {
         width: 100%;
         margin: 10px 0;
         padding: 12px;
         border: 1px solid #0808081d;
         border-radius: 8px;
         font-size: 16px;
         box-sizing: border-box;
         transition: border-color 0.3s, box-shadow 0.3s;
```

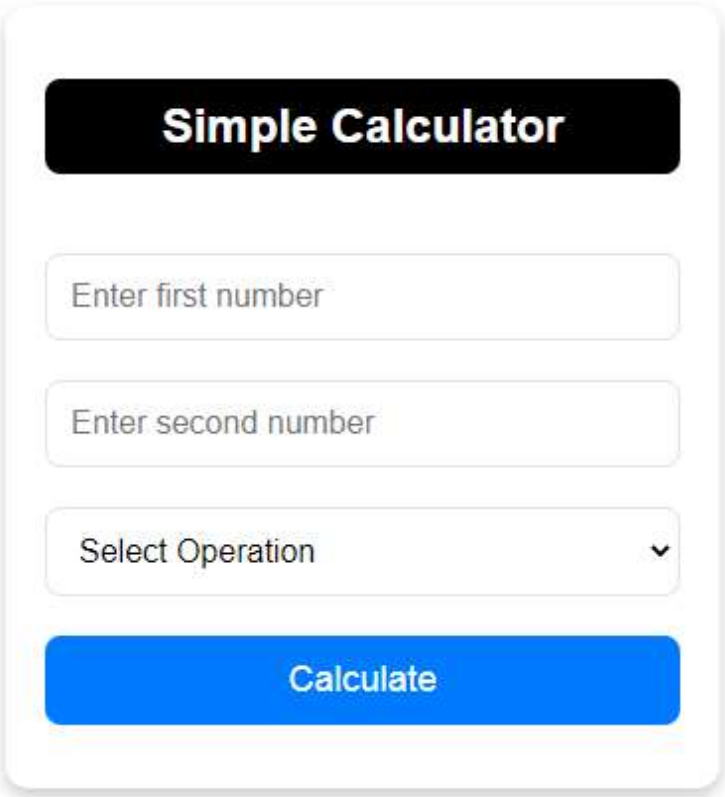
```
}

#operation {
    cursor: pointer;
}


input:focus,
select:focus,
button:focus {
    outline: none;
    border-color: #007bff;
    box-shadow: 0 0 0 3px rgba(38, 143, 255, 0.25);
}

option {
    background-color: #fff;
    color: #000;
    padding: 10px;
    border: none;
}

option:hover {
    background-color: #f1f1f1;
}

button {
    background-color: #007bff;
    color: white;
    border: none;
    cursor: pointer;
    font-size: 18px;
    transition: box-shadow 0.3s, transform 0.3s;
}

button:hover {
    box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
}

button:focus {
    box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
}

#result.error {
    background: #ffdddd;
    border-color: #ff0000;
}

#result.success {
```

```
            font-size: 17px;
            font-weight: 500;
            color: #000;
            background: #6ef08d38;
            border-color: #47e56d;
        }

        #result {
            font-size: 18px;
            color: #000000;
            border-radius: 8px;
            background: #afffe2;
            border: 1px solid #ccc;
            box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
            transition: opacity 0.5s, transform 0.5s;
            opacity: 0;
            transform: translateY(-20px);
        }


        #result.show {
            cursor: not-allowed;
            opacity: 1;
            margin-top: 20px;
            padding: 15px;
            transform: translateY(0);
        }

    </style>
</head>

<body>
    <div class="calculator">
        <h1>Simple Calculator</h1>
        <form id="calculator-form">
            <input type="number" id="num1" placeholder="Enter first number" required>
            <input type="number" id="num2" placeholder="Enter second number" required>
            <select id="operation" required>
                <option value="">Select Operation</option>
                <option value="sum">Sum</option>
                <option value="product">Product</option>
                <option value="difference">Difference</option>
                <option value="remainder">Remainder</option>
                <option value="quotient">Quotient</option>
                <option value="power">Power</option>
                <option value="sqrt">Square Root</option>
                <option value="square">Square</option>
            </select>
            <button type="button" onclick="calculate()">Calculate</button>
```

```
    </form>
    <div id="result"></div>
  </div>
  <script>
    function calculate() {
      const num1 = parseFloat(document.getElementById('num1').value);
      const num2 = parseFloat(document.getElementById('num2').value);
      const operation = document.getElementById('operation').value;
      let result = '';
      let resultClass = 'success';

      if (isNaN(num1) || isNaN(num2)) {
        result = 'Please enter valid numbers.';
        resultClass = 'error';
      } else {
        switch (operation) {
          case 'sum':
            result = `Sum: ${num1 + num2}`;
            break;
          case 'product':
            result = `Product: ${num1 * num2}`;
            break;
          case 'difference':
            result = `Difference: ${num1 - num2}`;
            break;
          case 'remainder':
            result = `Remainder: ${num1 % num2}`;
            break;
          case 'quotient':
            if (num2 === 0) {
              result = 'Cannot divide by zero';
              resultClass = 'error';
            } else {
              result = `Quotient: ${num1 / num2}`;
            }
            break;
          case 'power':
            result = `Power: ${Math.pow(num1, num2)}`;
            break;
          case 'sqrt':
            if (num1 < 0 || num2 < 0) {
              result = 'Square root is not defined for negative numbers';
              resultClass = 'error';
            } else {
              result = `Square Root of ${num1}: ${Math.sqrt(num1)} <br> Square Root of ${num2}:
${Math.sqrt(num2)}`;
            }
            break;
          case 'square':
```

```
                result = `Square of ${num1}: ${Math.pow(num1, 2)} <br> Square of ${num2}:
${Math.pow(num2, 2)}`;
                break;
            default:
                result = 'Please select an operation.';
                resultClass = 'error';
        }
    }
    const resultDiv = document.getElementById('result');
    resultDiv.innerHTML = result;
    resultDiv.classList.remove('show', 'error', 'success');
    resultDiv.classList.add(resultClass, 'show');
  }
 </script>
</body>
</html>
```

Output:



Program 7

Develop JavaScript program (with HTML/CSS) for:

a) Converting JSON text to JavaScript Object.

b) Convert JSON results into a date.

c) Converting From JSON To CSV and CSV to JSON.

d) Create hash from string using crypto.createHash() method.

```
<!DOCTYPE html>
<head>
   <script   src="https://cdnjs.cloudflare.com/ajax/libs/crypto-js/4.1.1/crypto-js.min.js"></script>
   <title>Simple Converter </title>
   <style>
      * {
         padding: 0;
         margin: 0;
         box-sizing: border-box;
      }

      body {
         font-family: Arial, sans-serif;
         color: #000000;
      }

      .container {
         width: 60%;
         margin: 0 auto;
         padding: 20px;
      }
      .head-title h1 {
         font-size: 28px;
         padding: 10px;
         color: #fff;
         margin-bottom: 50px;
      }

      .head-title {
         width: 100%;
         background: #000;
         text-align: center;
         border-radius: 10px;
      }
.section {
         margin-bottom: 40px;
         padding: 20px;
         border-radius: 8px;
         background: #fff;
         box-shadow: rgba(0, 0, 0, 0.1) 0px 1px 3px 0px, rgba(0, 0, 0, 0.06) 0px 1px 2px 0px;
```

```
    transition: all 0.3s;
    overflow: hidden;
}

.section h2 {
    color: #000000;
    font-size: 20px;
    margin-bottom: 15px;
}
textarea {
    font-size: 14px;
    width: 100%;
    height: 120px;
    margin-bottom: 15px;
    padding: 12px;
    border-radius: 8px;
    border: 1px solid #00000022;
    box-sizing: border-box;
    transition: border-color 0.3s ease, box-shadow 0.3s ease;
}

textarea:focus {
    background: transparent;
    border: 1px solid #00000022;
    border-color: #007BFF;
    box-shadow: 0 0 12px rgba(0, 123, 255, 0.5);
    outline: none;
}

input[type="text"] {
    width: calc(100% - 24px);
    padding: 12px;
    border-radius: 8px;
    border: 1px solid #ddd;
    box-sizing: border-box;
    transition: border-color 0.3s ease, box-shadow 0.3s ease;
    margin-bottom: 15px;
}

input[type="text"]:focus {
    border-color: #007BFF;
    box-shadow: 0 0 8px rgba(0, 123, 255, 0.5);
    outline: none;
}

button {
    display: inline-block;
    padding:  15px  15px;
    margin: 10px 0;
```

```
        font-weight: 600;
        border: none;
        border-radius: 7px;
        background-color: #007BFF;
        color: #fff;
        cursor: pointer;
        font-size: 16px;
        transition: box-shadow 0.3s ease, transform 0.3s ease;
    }

    button:hover {
        box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007BFF;
    }

    button:focus {
        box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007BFF;
    }

    pre {
        display: none;
        background: #f8f9fa;
        border: 1px solid #ddd;
        padding: 15px;
        border-radius: 8px;
        overflow: auto;
        transition: opacity 0.3s ease;
    }
    .error {
        margin-top: 10px;
        font-size: 14px;
        color: #000;
        background: #ffdddd;
        border-color: #ff0000;
        padding: 10px;
    }
    .success {
        margin-top: 10px;
        font-size: 14px;
        color: #000;
        background: #6ef08d38;
        border-color: #47e56d;
        padding: 10px;
    }

    .adjust-area {
        margin-top: 30px;
    }
  </style>
</head>
```

```
<body>
  <div class="container">
    <div class="head-title">
      <h1>Simple Converter</h1>
    </div>
    <div class="section">
      <h2>1. Convert JSON Text to JavaScript Object</h2>
      <textarea id="jsonInput" placeholder="Enter JSON here..."></textarea>
      <button onclick="convertJsonToObject()">Convert JSON</button>
      <pre id="jsonOutput" class="output"></pre>
    </div>
    <div class="section">
      <h2>2. Convert JSON Results into Date</h2>
      <textarea id="jsonDateInput" placeholder='Enter JSON with date in "yyyy-mm-dd"
format'></textarea>
      <button onclick="convertJsonToDate()">Convert to Date</button>
      <pre id="jsonDateOutput" class="output"></pre>
    </div>
    <div class="section">
      <h2>3. Convert JSON to CSV and CSV to JSON</h2>
      <textarea id="jsonCsvInput" placeholder="Enter JSON for CSV conversion..."></textarea>
      <button onclick="convertJsonToCsv()">JSON to CSV</button>
      <pre id="csvOutput" class="output"></pre>
      <textarea id="csvInput" placeholder="Enter CSV here..." class="adjust-area"></textarea>
      <button onclick="convertCsvToJson()">CSV to JSON</button>
      <pre id="jsonCsvOutput" class="output"></pre>
    </div>
    <div class="section">
      <h2>4. Create Hash from String</h2>
      <input type="text" id="hashInput" placeholder="Enter string to hash">
      <button onclick="createHash()">Create Hash</button>
      <pre id="hashOutput" class="output"></pre>
    </div>
  </div>
  <script>
    function showResult(id, text, isSuccess) {
      const element = document.getElementById(id);
      element.textContent = text;
      element.className = isSuccess ? 'success' : 'error';
      element.style.display = 'block';
      element.style.opacity = '1';
    }

    function convertJsonToObject() {
      const jsonInput = document.getElementById('jsonInput').value;
      try {
        const jsonObject = JSON.parse(jsonInput);
        showResult('jsonOutput', JSON.stringify(jsonObject, null, 2), true);
      } catch (error) {
```

```
            showResult('jsonOutput', 'Invalid JSON', false);
        }
    }

    function convertJsonToDate() {
        const jsonDateInput = document.getElementById('jsonDateInput').value;
        try {
            const data = JSON.parse(jsonDateInput);
            if (data.date && !isNaN(new Date(data.date).getTime())) {
                const date = new Date(data.date);
                showResult('jsonDateOutput', date.toString(), true);
            } else {
                showResult('jsonDateOutput', 'Invalid Date Format', false);
            }
        } catch (error) {
            showResult('jsonDateOutput', 'Invalid JSON', false);
        }
    }
    function convertJsonToCsv() {
        const jsonInput = document.getElementById('jsonCsvInput').value;
        try {
            const jsonArray = JSON.parse(jsonInput);
            if (Array.isArray(jsonArray) && jsonArray.length > 0) {
                const keys = Object.keys(jsonArray[0]);
                const csv = [
                    keys.join(','),
                    ...jsonArray.map(row => keys.map(key => JSON.stringify(row[key])).join(','))
                ].join('\n');
                showResult('csvOutput', csv, true);
            } else {
                showResult('csvOutput', 'Invalid JSON: Expected an array with objects.', false);
            }
        } catch (error) {
            showResult('csvOutput', 'Invalid JSON', false);
        }
    }

    function convertCsvToJson() {
        const csvInput = document.getElementById('csvInput').value;
        try {
            const lines = csvInput.trim().split('\n');
            if (lines.length > 1) {
                const keys = lines[0].split(',');
                if (keys.length > 0) {
                    const jsonArray = lines.slice(1).map(line => {
                        const values = line.split(',');
                        return keys.reduce((obj, key, index) => {
                            obj[key] = values[index];
                            return obj;
```

```
                    }, {});
                });
                showResult('jsonCsvOutput', JSON.stringify(jsonArray, null, 2), true);
            } else {
                showResult('jsonCsvOutput', 'Invalid CSV: No columns found.', false);
            }
        } else {
            showResult('jsonCsvOutput', 'Invalid CSV: No data found.', false);
        }
    } catch (error) {
        showResult('jsonCsvOutput', 'Invalid CSV', false);
    }
}

function createHash() {
    const hashInput = document.getElementById('hashInput').value.trim();
    if (hashInput.length > 0) {
        const hash = CryptoJS.SHA256(hashInput).toString();
        showResult('hashOutput', hash, true);
    } else {
        showResult('hashOutput', 'Input cannot be empty', false);
    }
}
</script>
</body>
</html>
```

Output:

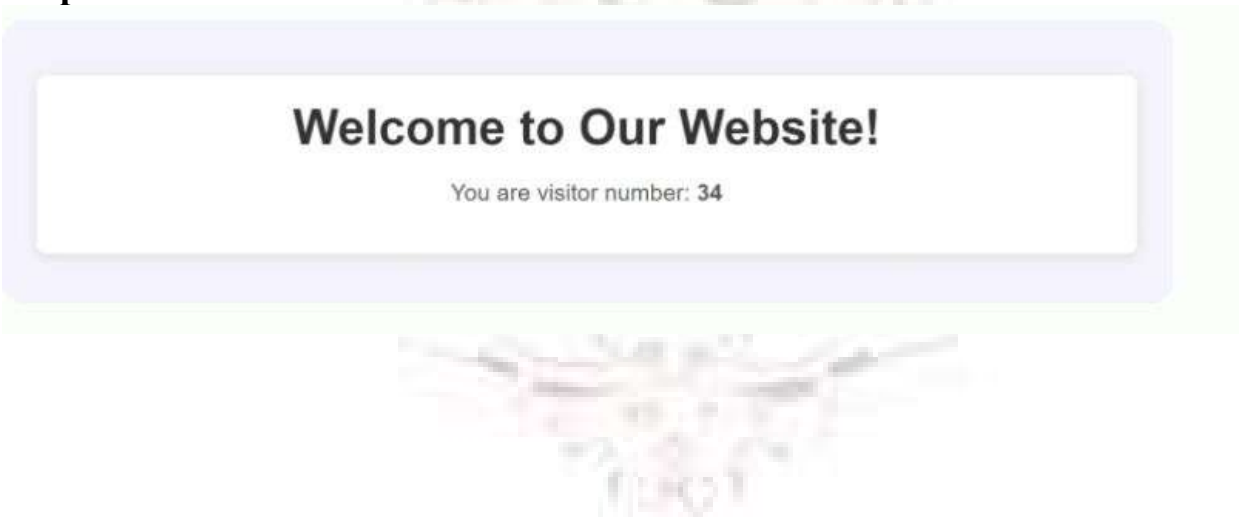
Program 8A

Develop a PHP program (with HTML/CSS) to keep track of the number of visitors visiting the web page and to display this count of visitors, with relevant headings.

```
<?php
$counterFile = "counter.txt";
    if (!file_exists($counterFile)) {
        file_put_contents($counterFile, "0");
    }

    $currentCount = file_get_contents($counterFile);

    $newCount = $currentCount + 1;

    file_put_contents($counterFile, $newCount);
    ?>

    <!DOCTYPE html>
    <html lang="en">

    <head>

        <title>Visitor Counter </title>
        <style>
            body {
                font-family: Arial, sans-serif;
                text-align: center;
                margin: 0;
                padding: 0;
                display: flex;
                flex-direction:  column;
                justify-content:  center;
                height: 100vh;
                background-color: #f4f4f9;
                color: #333;
            }

            .container {
                background: #fff;
                padding: 20px;
                box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
                border-radius: 8px;
                margin: 0 auto;
                width: 60%;
            }

            h1 {
```

```
            font-size: 2.5em;
            margin: 0;
        }

        p {
            font-size: 1.2em;
            color: #555;
        }
    </style>
</head>

<body>
    <div class="container">
        <h1>Welcome to Our Website!</h1>
        <p>You are visitor number: <strong><?php echo $newCount; ?></strong></p>
    </div>
</body>
</html>
```

Output:



Program 8B

Develop a PHP program (with HTML/CSS) to sort the student records which are stored in the database using selection sort.

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "students";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT * FROM students";
$result = $conn->query($sql);

$students = [];
if ($result->num_rows > 0) {
   while ($row = $result->fetch_assoc()) {
      $students[] = $row;
   }
}

function selectionSort(&$arr, $key)
{
   $n = count($arr);
   for ($i = 0; $i < $n - 1; $i++) {
      $minIndex = $i;
      for ($j = $i + 1; $j < $n; $j++) {
         if ($arr[$j][$key] < $arr[$minIndex][$key]) {
            $minIndex = $j;
         }
      }

      $temp = $arr[$i];
      $arr[$i] = $arr[$minIndex];
      $arr[$minIndex] = $temp;
   }
}

selectionSort($students, 'name');
?>
```

```
<!DOCTYPE html>

<head>
    <title>Sorted Student Records</title>
    <style>
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            background-color: #f0f2f5;
            color: #333;
            margin: 0;
            padding: 20px;
        }

        h2 {
            text-align: center;
            color: #4A90E2;
            margin-bottom: 20px;
        }

        table {
            width: 100%;
            border-collapse: collapse;
            background-color: #fff;
            border-radius: 10px;
            overflow: hidden;
            box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
            margin: 0 auto;
        }

        th,
        td {
            padding: 12px 15px;
            text-align: left;
            border-bottom: 1px solid #ddd;
        }

        th {
            background-color: #4A90E2;
            color: white;
            text-transform: uppercase;
            letter-spacing: 0.03em;
        }

        tr {
            transition: background-color 0.3s ease;
        }

        tr:hover {
            background-color: #f1f1f1;
```

```
}

td {
    font-size: 0.9em;
    color: #555;
}

@media (max-width: 768px) {

    table,
    th,
    td {
        display: block;
        width: 100%;
    }

    th,
    td {
        box-sizing: border-box;
    }

    tr {
        margin-bottom: 15px;
        display: block;
        box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
    }
    th {
        position: absolute;
        top: -9999px;
        left: -9999px;
    }

    td {
        border: none;
        position: relative;
        padding-left: 50%;
        text-align: right;
    }

    td:before {
        content: attr(data-label);
        position: absolute;
        left: 0;
        width: 50%;
        padding-left: 15px;
        font-weight: bold;
        text-align: left;
        text-transform: uppercase;
        color: #4A90E2;
```

```
            }
        }
    </style>
</head>
<body>
    <h2>Sorted Student Records by Name</h2>
    <table>
        <thead>
            <tr>
                <th>ID</th>
                <th>Name</th>
                <th>USN</th>
                <th>Branch</th>
                <th>Email</th>
                <th>Address</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($students as $student): ?>
                <tr>
                    <td data-label="ID"><?php echo htmlspecialchars($student['id']); ?></td>
                    <td data-label="Name"><?php echo htmlspecialchars($student['name']); ?></td>
                    <td data-label="USN"><?php echo htmlspecialchars($student['usn']); ?></td>
                    <td data-label="Branch"><?php echo htmlspecialchars($student['branch']); ?></td>
                    <td data-label="Email"><?php echo htmlspecialchars($student['email']); ?></td>
                    <td data-label="Address"><?php echo htmlspecialchars($student['address']); ?></td>
                </tr>
            <?php endforeach; ?>
        </tbody>
    </table>
</body>
</html>
```

Output :

Sorted Student Records by Name

ID	NAME	USN	BRANCH	EMAIL	ADDRESS
2	Aman Kumar	1ME21CS002	CSE	aman@gmail.com	Chennai
8	Arun Kumar	1ME21CS008	CSE	arun@gmail.com	Bhopal
5	Bikash Kumar Singh	1ME21CS005	CSE	bikash@gmail.com	Mumbai, India
1	Braham Kumar	1ME21CS001	DS	braham@gmail.com	Nepal
9	Dipesh Kumar Mandal	1ME21CS009	ISE	dipesh@gmail.com	Indore
6	Shaoib Akhtar	1ME21CS006	AI&ML	shoaib@gmail.com	Patna, India
7	Shiv Kumar Yadav	1ME21CS007	ECE	shiv@gmail.com	Lucknow
4	Shubham Kumar	1ME21CS004	CSE	shubham@gmail.com	Ghaziabad
10	Shyam Kumar Singh	1ME21CS010	ME	shyam@gmail.com	Pune
3	Sunil Kumar	1ME21CS003	CIVIL	sunil@gmail.com	Delhi, India

Program 9

Develop jQuery script (with HTML/CSS) for:

a. Appends the content at the end of the existing paragraph and list.

b. Change the state of the element with CSS style using animate() method.

c. Change the color of any div that is animated.

```
<!DOCTYPE html>
<head>
   <script  src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
   <title>jQuery Example </title>
   <style>
      body {
         font-family: 'Roboto', sans-serif;
         background-color: #f4f7f6;
         margin: 0;
         padding: 0;
         display: flex;
         justify-content: center;
         align-items: center;
         height: 100vh;
      }
      .container {
         text-align: center;
         background: #fff;
         padding: 30px;
         border-radius: 12px;
         box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
         transition: transform 0.3s ease-in-out;
      }
      .container:hover {
         transform: scale(1.02);
      }

      #paragraph {
         margin-bottom: 20px;
         color: #333;
         font-size: 18px;
         line-height: 1.5;
      }

      #list {
         margin-bottom: 20px;
         list-style: none;
         padding: 0;
      }
```

```
        #list li {
            background: #e8f0fe;
            margin: 5px 0;
            padding: 10px;
            border-radius: 8px;
            transition: background 0.3s;
        }
        #list li:hover {
            background: #d0e2fe;
        }
        .box {
            padding: 0 10px;
            width: 100px;
            height: 100px;
            background-color: #007bff;
            margin: 20px auto;
            line-height: 100px;
            color: white;
            text-align: center;
            border-radius: 8px;
            transition: all 0.3s ease;
        }
        button {
            padding: 12px 24px;
            margin: 10px;
            cursor: pointer;
            border: none;
            border-radius: 6px;
            font-size: 16px;
            background: #007bff;
            color: white;
            transition: box-shadow 0.3s, transform 0.2s;
            box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
        }
        button:hover {
            box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
        }
        button:focus {
            box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
        }
        button:active {
            background: #004494;
            transform: translateY(0);
        }
    </style>
</head>
<body>
    <div class="container">
        <p id="paragraph">This is an existing paragraph.</p>
```

```
    <ul id="list">
      <li>List item 1</li>
      <li>List item 2</li>
    </ul>
    <div class="box" id="box">Animate me!</div>
    <button id="appendButton">Append Content</button>
    <button id="animateButton">Animate Box</button>
  </div>
  <script>
    $(document).ready(function () {
      $("#appendButton").click(function () {
        $("#paragraph").append(" Appended text.");
        $("#list").append("<li>New appended list item</li>");
      });
      $("#animateButton").click(function () {

        $("#box").stop(true, true).css({
          width: "100px",
          height: "100px",
          opacity: 1,
          backgroundColor: "blue"
        }).animate({
          width: "200px",
          height: "200px",
          opacity: 0.5
        }, 1000, function () {

          $(this).css("background-color", "green");
        });
      });
    });
  </script>
</body>
</html>
```

Output :

Program 10

Develop a JavaScript program with Ajax (with HTML/CSS) for:

a. Use ajax() method (without Jquery) to add the text content from the text file by sending ajax request.

b. Use ajax() method (with Jquery) to add the text content from the text file by sending ajax request.

c. Illustrate the use of getJSON() method in jQuery.

d. Illustrate the use of parseJSON() method to display JSON values.

Note: Create two separate file within the same folder one is **textfile.txt** and other **data.json** then copy below text for the both separate file and paste it save it.

textfile.txt

hi this is example text...

data.json

{"name":"John Doe","age":30,"city":"New York","skills":["JavaScript","React","Node.js"],"address":{"street":"123 Elm Street","zipcode":"10001"},"projects":[{"name":"Website Redesign","year":2023,"technologies":["HTML","CSS","JavaScript"]},{"name":"Mobile App","year":2024,"technologies":["React Native","Expo"]}]}

PROGRAM:

```
<!DOCTYPE html>

<head>
    <title>AJAX Examples </title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f4f4f9;
        }

        h1 {
            text-align: center;
            color: #333;
            padding: 20px 0;
        }

        #content {
            flex-direction: column;
```

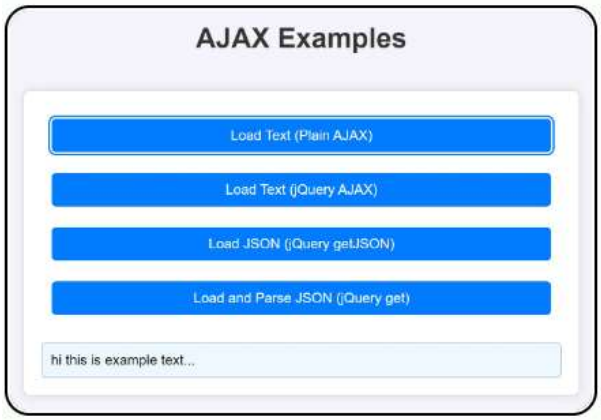
```
    display: flex;
    max-width: 600px;
    margin: 20px auto;
    padding: 20px;
    border: 1px solid #ddd;
    border-radius: 8px;
    background-color: #fff;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

button {
    display: inline-block;
    padding: 10px 15px;
    margin: 12px;
    border: none;
    border-radius: 5px;
    background-color: #007bff;
    color: #fff;
    font-size: 16px;
    cursor: pointer;
    transition: box-shadow 0.3s;
}

button:hover {
    box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
}

button:focus {
    box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;
}


#output {
    display: none;
    margin-top: 20px;
    padding: 10px;
    border-radius: 5px;
    white-space: pre-wrap;
    max-height: 300px;
    overflow-y: auto;
}

#output.plain-ajax {
    background-color: #f0f8ff;
    border: 1px solid #b0c4de;
}

#output.jquery-ajax {
    background-color: #f5fffa;
```

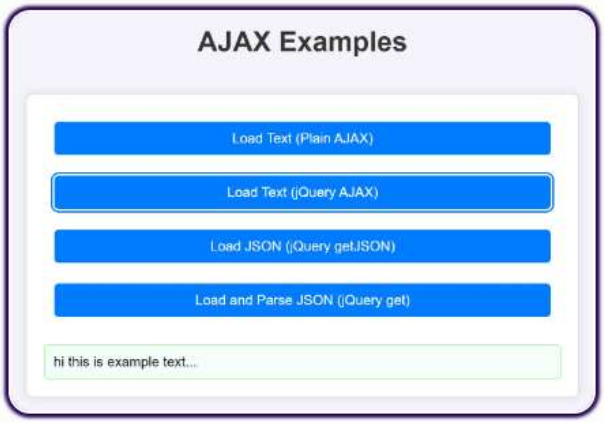
```
            border: 1px solid #98fb98;
        }

        #output.jquery-json {
            background-color: #fffaf0;
            border: 1px solid #ffd700;
        }

        #output.parse-json {
            background-color: #fff0f5;
            border: 1px solid #ff69b4;
        }
    </style>
</head>

<body>
    <h1>AJAX Examples</h1>
    <div id="content">
        <button id="plain-ajax-btn">Load Text (Plain AJAX)</button>
        <button id="jquery-ajax-btn">Load Text (jQuery AJAX)</button>
        <button id="jquery-json-btn">Load JSON (jQuery getJSON)</button>
        <button id="parse-json-btn">Load and Parse JSON (jQuery get)</button>
        <div id="output"></div>
    </div>

    <script>

        function showOutput(className) {
            const output = document.getElementById('output');
            output.className = className;
            output.style.display = 'block';
        }

        document.getElementById('plain-ajax-btn').addEventListener('click', function () {
            var xhr = new XMLHttpRequest();
            xhr.open('GET', 'textfile.txt', true);
            xhr.onload = function () {
                if (xhr.status === 200) {
                    document.getElementById('output').innerText = xhr.responseText;
                } else {
                    document.getElementById('output').innerText = 'Error loading file.';
                }
                showOutput('plain-ajax');
            };
            xhr.send();
        });

        $('#jquery-ajax-btn').on('click', function () {
            $.ajax({
```

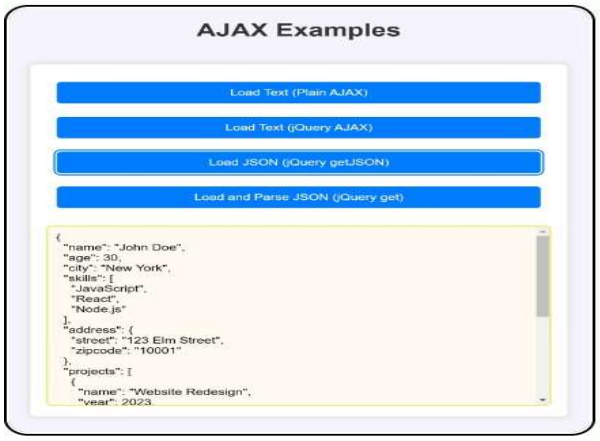
```
        url: 'textfile.txt',
        method: 'GET',
        success: function (data) {
            $('#output').text(data);
        },
        error: function () {
            $('#output').text('Error loading file.');
        }
    }).always(function () {
        showOutput('jquery-ajax');
    });
});


$('#jquery-json-btn').on('click', function () {
    $.getJSON('data.json')
        .done(function (data) {
            $('#output').text(JSON.stringify(data, null, 2));
        })
        .fail(function () {
            $('#output').text('Error loading JSON file.');
        })
        .always(function () {
            showOutput('jquery-json');
        });
});
$('#parse-json-btn').on('click', function () {
    $.get('data.json')
        .done(function (data) {
            try {
                let jsonData;

                if (typeof data === 'string') {
                    jsonData = JSON.parse(data);
                } else {
                    jsonData = data;
                }
                $('#output').text(JSON.stringify(jsonData, null, 2));
            } catch (e) {
                $('#output').text('Error parsing JSON: ' + e.message);
            }
        })
        .fail(function () {
            $('#output').text('Error loading JSON file.');
        })
        .always(function () {
            showOutput('parse-json');
        });
});
```

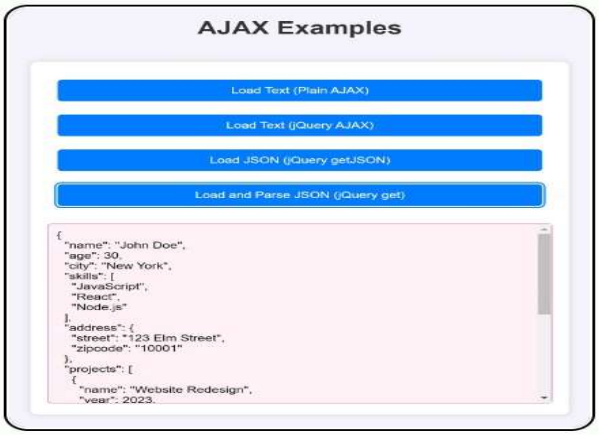
```
    </script>
</body>

</html>
```

Output:









Viva Questions

1. What is HTML?
2. What is tags?
3. What is the simplest HTML pages?
4. How can I include comments in HTML?
5. What is hypertext link?
6. What is CSS?
7. What are the most commonly used languages and platforms for website design?
8. Explain the term DOCTYPE.
9. Differentiate between HTML elements and HTML tags?
10. Explain the difference between a numbered list and bulleted list.