



# ADITYA

## COLLEGE OF ENGINEERING

Aditya Nagar, ADB Road, Surampalem - 533437

Department of

Name :

Roll No. :

--	--	--	--	--	--	--	--	--	--	--	--

**Certified that this is the bonafide record of  
practical work done by**

Mr. /Ms. ....

a student of .....with PIN No. ....

in the ..... Laboratory during the year .....

No. of Practicals Conducted :

No. of Practicals Attended :

Signature - Faculty Incharge

Signature - Head of the Department

Submitted for the Practical examination held on .....

EXAMINER - 1

EXAMINER - 2

## **VISION & MISSION OF THE INSTITUTE**

### **VISION**

To induce higher planes of learning by imparting technical education with

- International standards
- Applied research
- Creative Ability
- Value based instruction and to emerge as a premiere institute.

### **MISSION**

Achieving academic excellence by providing globally acceptable technical education by forecasting technology through

- Innovative Research and development
- Industry Institute Interaction
- Empowered Manpower

## **VISION & MISSION OF THE DEPARTMENT**

### **VISION**

To be a recognized Computer Science and Engineering hub striving to meet the growing needs of the Industry and Society.

### **MISSION**

M1: Imparting Quality Education through state-of-the-art infrastructure with industry Collaboration

M2: Enhance Teaching Learning Process to disseminate knowledge.

M3: Organize Skill based, Industrial and Societal Events for overall Development.

# Pointer

# Pointer

## WEB APPLICATION DEVELOPMENT USING FULL STACK

### INTRODUCTION TO HTML

**Web site:** A set of interconnected web pages, usually including a homepage, generally located on the same server, and prepared and maintained as a collection of information by a person, group, or organization.

**Web Page:** A web page is a document that's created in html that shows up on the internet when you type in or go to the web page's address.

#### Types of Web Pages:

- **Static web page:** is delivered exactly as stored, as web content in the web server's file system. Contents cannot be changed.
- **Dynamic web page:** is generated by a web application that is driven by server-side software or client-side scripting. Dynamic web pages help the browser (the client) to enhance the web page through user input to the server. Contents can be changed as evolution over time.

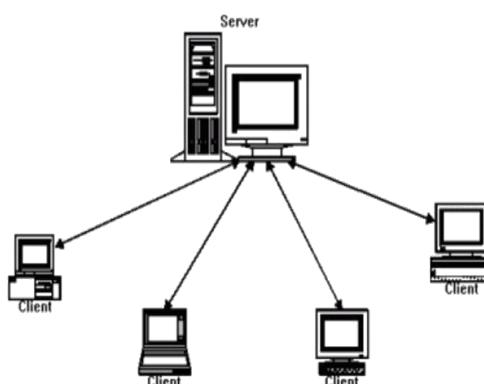
#### Browsers & their types

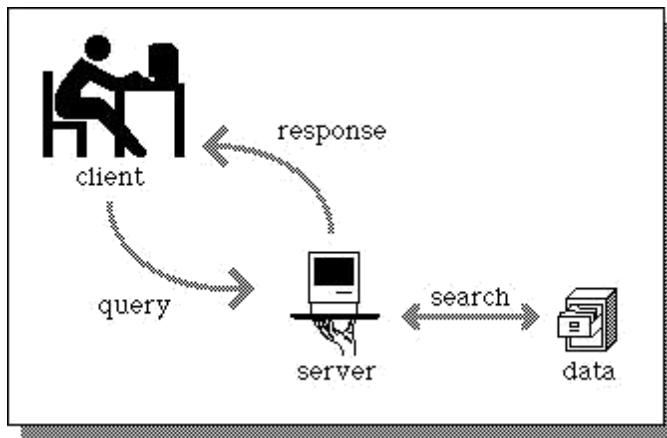
A web browser (commonly referred to as a browser) is a software application for retrieving, presenting and traversing information resources on the World Wide Web.

The major web browsers are Google Chrome, Firefox, Internet Explorer, Opera, and Safari.

#### Client –Server Model

The client–server model is a distributed application structure in computing that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. Often clients and servers communicate over a computer network. A server is a host that is running one or more server programs which share their resources with clients. A client requests a server's content or service function.





## Web –Server

Web server refers to either the hardware (the computer) or the software (the computer application) that helps to deliver web content that can be accessed through the Internet.

The most common use of web servers is to host websites, but there are other uses such as gaming, data storage or running enterprise application.

## Working of different types of web pages

The different types of web pages are :

**Advocacy:** An advocacy web page is one sponsored by an organization to influence opinion. URL ends with .org

**Business and marketing:** It is one sponsored by a commercial enterprise to sell or market their services. URL ends with .com

**News:** It provides timely information about current events and issues.

**Informational:** This includes reports, research findings, schools and college information.

URL ends with .edu or .gov.

**Personal:** It is created by an individual for his /her own personal need. URL has tiddle(~).

### General structure of a Web Page

A basic HTML page contains a Head section and a Body section. The contents of the head section are normally invisible in a web browser and mainly consists of some Metatags. The Body consist of those HTML elements that you want to have displayed in your browser.

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

**Scripting language:** A scripting language or script language is a programming language that supports the writing of scripts, programs written for a special runtime environment that can interpret and automate the execution of tasks which could alternatively be executed one-by-one by a human operator.

**URL:** A uniform resource locator (URL), also known as web address, is a specific character string that constitutes a reference to a resource. In most web browsers, the URL of a web page is displayed on top inside an address bar. An example of a typical URL would be "http://en.example.org/wiki/Main\_Page".

### Popular Search Engines

- Yahoo Search
- Google Search
- Bing
- Info.com
- Search.com
- Infospace

**WWW:** The World Wide Web (WWW) is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia, and navigate between them via hyperlinks.

## ILLUSTRATING HTML TAGS AND THEIR ATTRIBUTES

**HTML:**Hyper Text Markup Language is the most widely used language to write web pages.it is a markup language.

**Hypertext :**Refers to the way in which web pages are linked together.

**Markup Language:** The user simply markups a text document with tags that tell a web browser how to structure it to display.

**Creating HTML document :**To begin coding HTML user needs only two things:

1. A simple text editor (notepad).
2. A web browser.

### Simple steps to create a basic HTML document:

1. Open notepad or another text editor.
2. At the top of the page type <html>
3. Add the opening header tag <head>
4. On the next line type <title> give title for page </title>
5. Go to next line and type closing header tag </head>
6. Go to next line and type opening body tag<body>
7. Go to next line and type closing body tag</body>
8. Finally, go to next line and type</html>
9. In the file menu,choose save as.
10. In the save as type option box,choose all files.
11. Name the file filename.html
12. Click save.

**HTML document structure:** An HTML document starts and ends with <html> and </html> tags.These tags tell the browser that the entire document is composed in HTML.Inside these two tags,the document is split into 2 sections:

1. The `<head>.....</head>` elements contains information about the document such as title of the document etc.

2. The `<body>.... </body>` elements contains the real content of the document that you see on your screen.

#### ATTRIBUTES:

An attribute is used to define the characteristics of an element and is placed inside the element's opening tag. All attributes are made up of 2 parts: a name and a value.

-The name is the property you want to set.

-The value is what you want the value of the property to be.

Example: `<font face="arial" color="red">`

## GRADED EXPERIMENTS

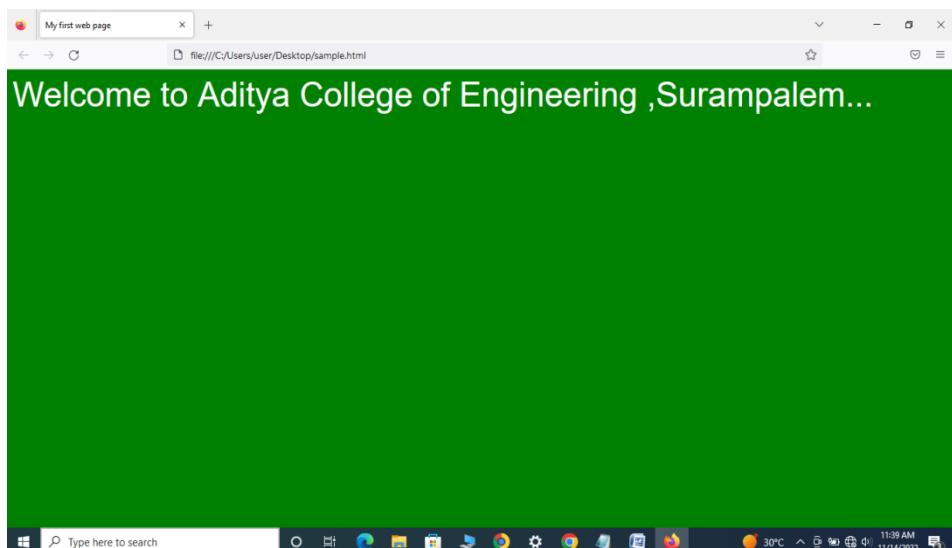
### EXPERIMENT: 1

1. Design a page having suitable background colour and text colour with title “My First Web Page” using all the attributes of the Font tag.

**Procedure:**

1. Go to start ->all program – >accessories-> notepad.
2. Type the html code.
3. Include title ”my first web page” in title tag.
4. Set the background for the webpage by using bgcolor attributes of the body tag.
5. Use font tag to display text.
6. Go to file->save->save the file with html extension.
7. Run the html code using browser.

```
<html>
<head>
<title>My first web page</title>
</head>
<body bgcolor="green">
<font size="16" color="white" face="Arial"> Welcome to Aditya College of Engineering
,Surampalem...
</font>
</body>
</html>
```

**OUTPUT:**

## EXPERIMENT: 2

Create a HTML document giving details of your [Name, Age], [Address, Phone] and [Register Number, Class] aligned in proper order using alignment attributes of Paragraph tag.

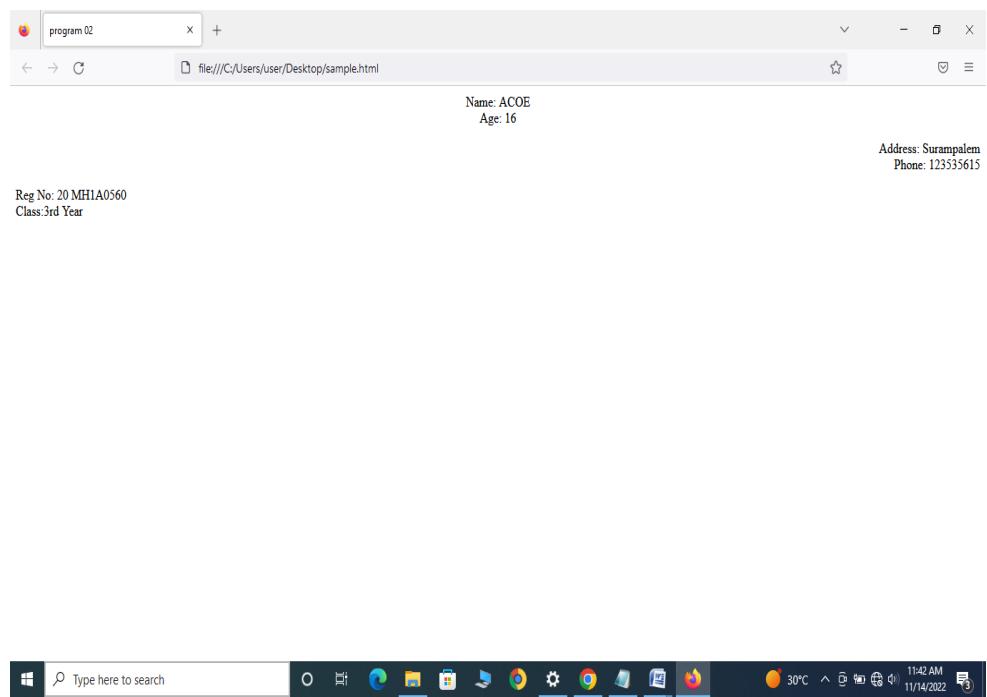
**Procedure:**

1. Go to start ->all program – >accessories-> notepad
2. Type the html code
3. Align [name,age]to center by using align =”left ” of <p> tag
4. Align [register,number,class] to right using align=”right” of <p> tag
5. Go to file-> save->save the file with html extension
6. Run the html code using browser

```
<html>
<head>
<title>program 02</title>
</head>
<body>
<p align="center">
Name: ACOE<br>
Age: 16
</p>
<p align="right">
Address: Surampalem<br>
Phone: 123535615
</p>
<p align="left">
Reg No: 20 MH1A0560<br>
Class:3rd Year

```

```
</p>  
  
</body>  
  
</htmL>
```

**OUTPUT:**

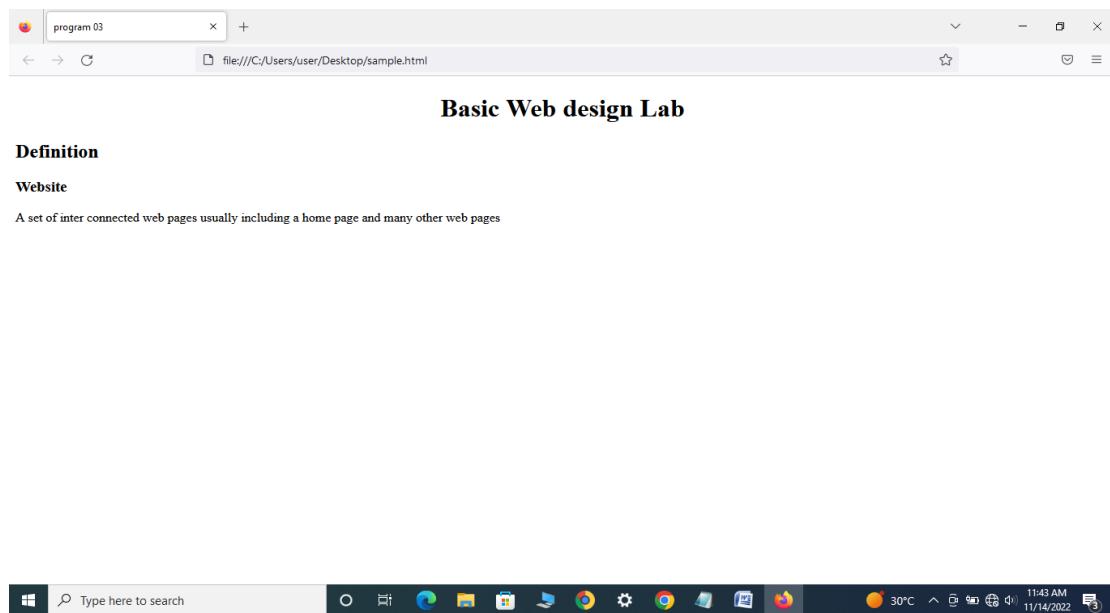
### EXPERIMENT: 3

**Write HTML code to design a page containing some text in a paragraph by giving suitable heading style.**

**Procedure:**

1. Go to start ->all program → accessories-> notepad
2. Type the html code
3. Use<h1> to<h6>tag to display heading
4. Use <p> tag for the paragraph
5. Go to file- >save->save the file with html extension
6. Run the html code html using browser

```
<html>
<head>
<title>program 03</title>
</head>
<body>
<center>
<h1>Basic Web design Lab</h1>
</center>
<h2 align="left"> Definition </h2>
<h3 align="left">Website</h3>
<p>
A set of inter connected web pages usually including a home page and many other web pages
</p>
</body>
</html>
```

**OUTPUT:**

## EXPERIMENT: 4

Create a page to show different character formatting (B, I, U, SUB, SUP) tags.

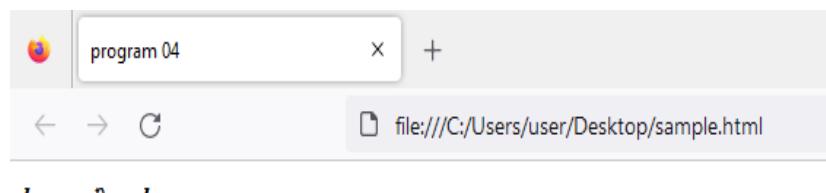
viz :  $\log_b m^p = p \log_b m$

**Procedure:**

1. Go to start ->all program ->accessories-> notepad
2. Type the html code
3. Include <b>,<u>,<sup>,<sub>,tags
4. Go to file->save-> save the file with html extension
5. Run the html code using browsers

```
<html>
<head>
<title>program 04</title>
</head>
<body>
<p> <b> <i> log</i></b> <sub>b </sub>m<sup> p</sup> =p<b> <i> log </i></b><sub>b</sub>
m</p>
</body>
</html>
```

**OUTPUT:**



## EXPERIMENT: 5

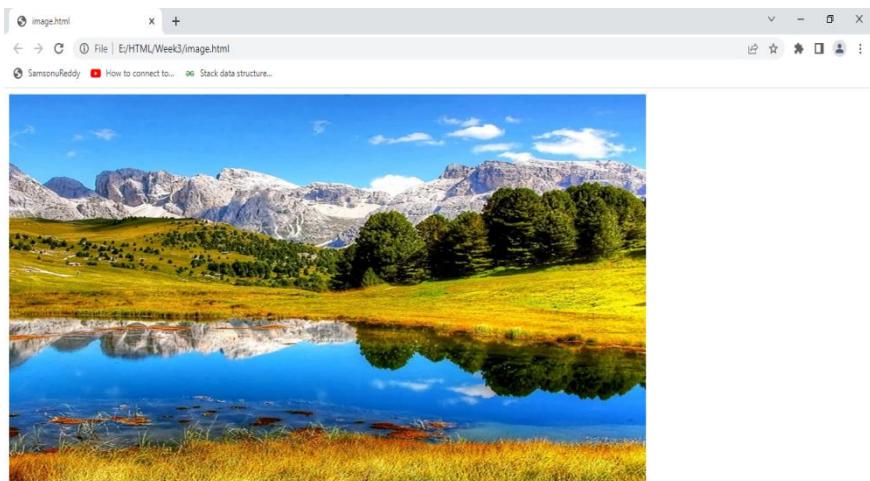
**Write HTML code to create a Web Page that contains an Image at its centre.**

**Procedure:**

1. Go to start ->all program –>accessories-> notepad
2. Type the html code
3. Include <img> tag inside <center> tag and set align attribute of img tag to middle
4. Go to file->save->save the file with html extension
5. Run the html code using browsres

```
<html>
<head>
</head>
<body>
</img>
</body>
</html>
```

**OUTPUT:**



## EXPERIMENT: 6

Create a web page with an appropriate image towards the left hand side of the page, when user clicks on the image another web page should open.

**Procedure:**

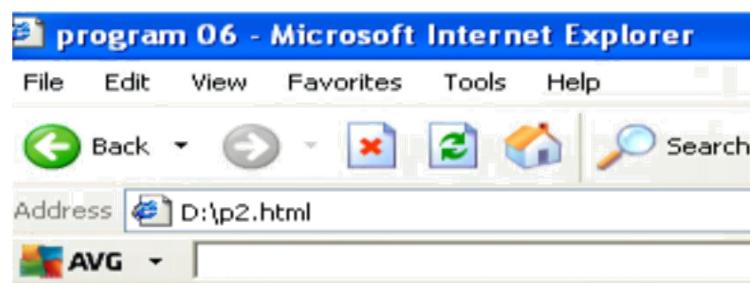
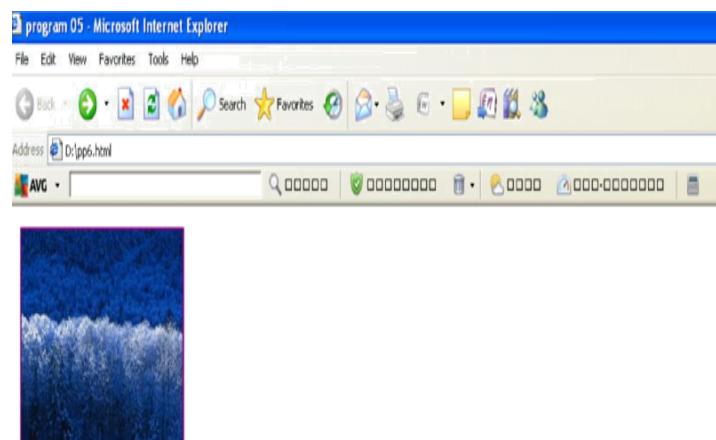
1. Go to start ->all programs-> accessories->notepad
2. Include<img> tag inside<a> tag
3. Save the file as p1.html
4. Create another file
5. Type the html code
6. Save the file as p2.html
7. In <a> tag assign p2.html to href attribute
8. Go to file-> save->save the file with html extension
9. Run the html code using browsers

```
<html>
<head>
<title>program 05</title>
</head>
<body>
<a href="p2.html">

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

**P2.html**

```
<html>
<head>
<title>program 06</title>
</head>
<body> Hi Welcome...</body>
</html>
```

**OUTPUT:**

## EXPERIMENT: 7

Create web Pages using Anchor tag with its attributes for external links.

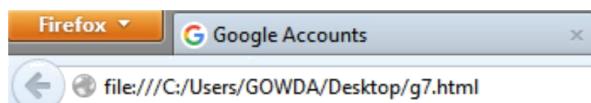
**Procedure:**

1. Go to start->all programs->accessories->notepad.
2. Create file called as p1.html
3. To provide external link between p1.html and gmail website use <a> in p1.html and mention url of gmail website in href attribute.
4. Save the file.
5. Run the p1.html file using browsers.
6. If we click on the link it should link to gmail website.

**P1.html**

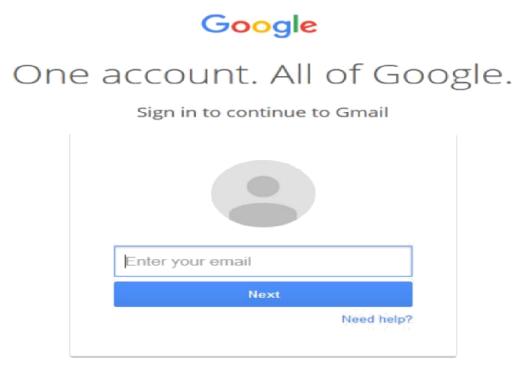
```
<html>
<head><title>page 1</title>
</head>
<body>
<p>Welcome to gpt</p>
<a href="http://www.gmail.com">click here to link gmail website</a>
</body>
</html>
```

**OUTPUT:**



Welcome to gpt

[click here to link gmail website](http://www.gmail.com)



## **EXPERIMENT: 8**

**Create a web page for internal links; when the user clicks on different links on the web page it should go to the appropriate locations/sections in the same page.**

**Procedure:**

1. Go to start->all programs->accessories->notepad.
2. Type the html code.
3. First give name for the each section of the page by using name attribute of the <a>tag (i.e <a name="home"> </a>)
4. To provide link for the sections use href="#name of section" in<a> tag.
5. Go to file->save->save the file with .html extension.
6. Run the html code using browsers.

**Note: Include paragraph means, include some text paragraph in <p> tag**

```
<html>

<head><title>program 8</title></head>

<body>

<a name="home"></a>

<p>include paragraph

</p>

<h1>contents</h1>

<a href="#link1">History</a><br>

<a href="#link2">Html version</a><br>

<a name="link1"></a>

<p>include paragraph

</p><br>

<a name="link2"></a>

    html versions

<p>include paragraph

</p><br>

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

</body>

</html>
```

## OUTPUT:



hyperText Markup Language, commonly referred to as HTML, is the standard markup language used to create web pages. Along with CSS, and JavaScript, HTML is a cornerstone technology, used by most websites to create visually engaging web pages, user interfaces for web applications, and user interfaces for many mobile applications [1]. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. The language is written in the form of HTML elements consisting of tags enclosed in angle brackets (like ). Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page. HTML can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages. Web browsers can also refer to Cascading Style Sheets (CSS) to define the look and layout of text and other material. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997 [2]. HyperText Markup Language, commonly referred to as HTML, is the standard markup language used to create web pages. Along with CSS, and JavaScript, HTML is a cornerstone technology, used by most websites to create visually engaging web pages, user interfaces for web applications, and user interfaces for many mobile applications [1]. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. The language is written in the form of HTML elements consisting of tags enclosed in angle brackets (like ). Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page. HTML can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages. Web browsers can also refer to Cascading Style Sheets (CSS) to define the look and layout of text and other material. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997 [2].

### contents

[History](#)  
[Final version](#)

In 1980, physicist Tim Berners-Lee, then a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system [3]. Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes [4] from 1990 he listed [5] "some of the many areas in which hypertext is used" and put an encyclopedia first. The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991 [6][7]. It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGML, a de facto Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exists in HTML 4 [8]. HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formating languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system; these formating commands were derived from the commands used by typewriters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and markup; HTML has been progressively moved in this direction with CSS. Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the Internet Engineering Task Force (IETF) with the mid-1993 publication of the first proposal for an HTML specification: "Hypertext Markup Language (HTML)" Internet-Draft by Berners-Lee and Dan Connolly, which included a SGML Document Type Definition to define the grammar [9]. The draft expired after six months, but was notable for its acknowledgement of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes [10]. Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms [11]. After the HTML and HTML+ drafts expired in early 1994, IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based [12]. Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C) [13]. However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). HTML 4.01 was published in late 1999, with further errata published through 2001. In 2004, development began on HTML5 in the Web Hypertext Application Technology Working Group (WHATWG), which became a joint deliverable with the W3C in 2008, and completed and standardized on 28 October 2014 [14].

[History](#)  
[Final version](#)

In 1980, physicist Tim Berners-Lee, then a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system [3]. Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes [4] from 1990 he listed [5] "some of the many areas in which hypertext is used" and put an encyclopedia first. The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991 [6][7]. It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGML, a de facto Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exists in HTML 4 [8]. HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formating languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system; these formating commands were derived from the commands used by typewriters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and markup; HTML has been progressively moved in this direction with CSS. Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the Internet Engineering Task Force (IETF) with the mid-1993 publication of the first proposal for an HTML specification: "Hypertext Markup Language (HTML)" Internet-Draft by Berners-Lee and Dan Connolly, which included a SGML Document Type Definition to define the grammar [9]. The draft expired after six months, but was notable for its acknowledgement of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes [10]. Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms [11]. After the HTML and HTML+ drafts expired in early 1994, IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based [12]. Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C) [13]. However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). HTML 4.01 was published in late 1999, with further errata published through 2001. In 2004, development began on HTML5 in the Web Hypertext Application Technology Working Group (WHATWG), which became a joint deliverable with the W3C in 2008, and completed and standardized on 28 October 2014 [14].

[HTML versions](#)

HTML versions timeline November 24, 1995: HTML 2.0 was published as IETF RFC 1866. Supplemental RFCs added capabilities: November 25, 1995: RFC 1867 (form-based file upload) May 1996: RFC 1942 (tables) August 1996: RFC 1960 (client-side image maps) January 1997: RFC 2070 (internationalization) January 14, 1997: HTML 3.2 [15] was published as a W3C Recommendation. It was the first version developed and standardized exclusively by the W3C, as the IETF had closed its HTML Working Group in September 12, 1996 [16]. Initially code-named "Walrus" [17], HTML 3.2 dropped math formulas entirely, reconciled overlap among various proprietary extensions, and adopted most of Netscape's visual markup tags. Netscape's `blink` element and Microsoft's `marquee` element were omitted due to a mutual agreement between the two companies [18]. A markup for mathematical formulas similar to  $\text{t}$  in HTML was not standardized until 14 months later in MathML. December 18, 1997: HTML 4.0 [18] was published as a W3C Recommendation. It offers three variations: [HTML](#)

## EXPERIMENT: 9

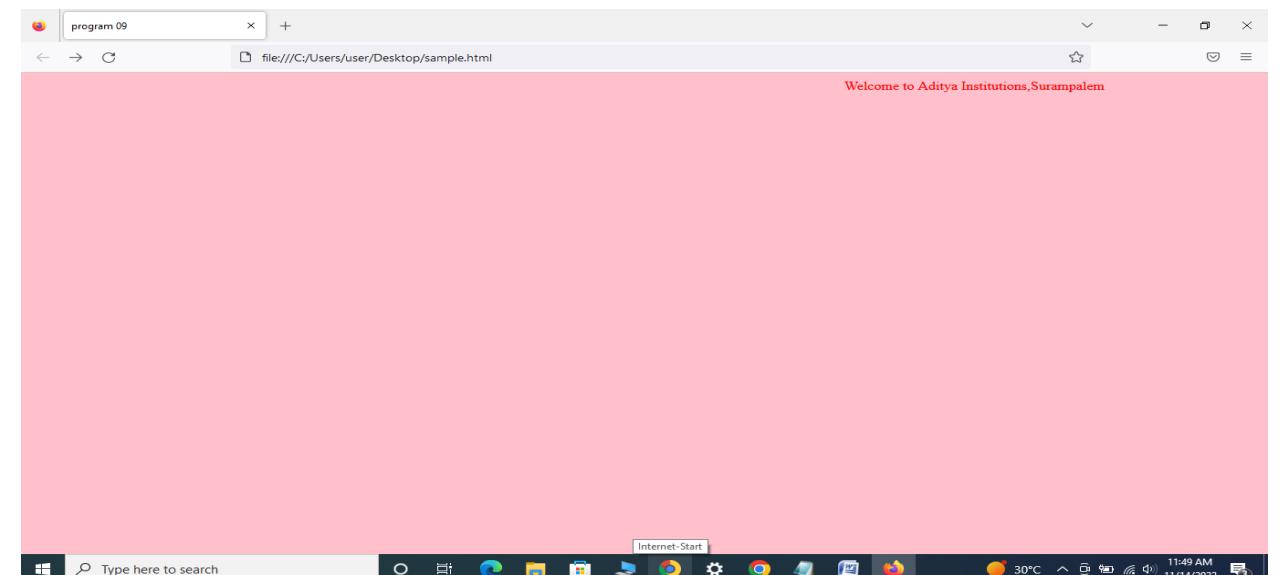
**Write a HTML code to create a web page with pink color background and display moving message in red color.**

**Procedure:**

1. Go to start->all programs->accessories->notepad
2. Type the html code
3. Include bgcolor="pink" attribute in the body tag
4. Use font tag to display moving text in red color by setting color="red" attribute
5. Include marquee tag inside font tag to display moving text
6. Go to file ->save->save the file with html extension
7. Run the html code using browser

```
<html>
<head>
<title>program 09</title>
</head>
<body bgcolor="pink">
<font color="red"><marquee direction="left">
Welcome to Aditya Institutes Surampalem</marquee></font>
</body>
</html>
```

**OUTPUT:**



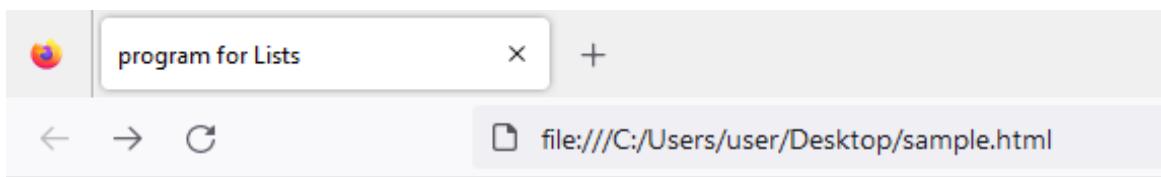
## EXPERIMENT: 10

Create a web page, showing an ordered list of all second semester courses (Subjects)

**Procedure:**

1. Go to start->all programs->accessories->notepad
2. Type the html code
3. Use <ol>-----</ol> tag to display ordered list of all subject
4. Inside <ol>tag use <li>---</li> tag to list each subject
5. Go to file ->save->save the file with html extension
6. Run the html code using browser

```
<html>
<head>
<title>program 09</title>
</head>
<body>
<ol>
<li> Maths-3 </li>
<li> Operating Systems</li>
<li> Data Structures </li>
<li> Java Programming</li>
<li> Java Programming Lab</li>
<li> Web Application Development Lab</li>
</ol>
</body>
</html>
```

**OUTPUT:**

## EXPERIMENT: 11

**Create a web page, showing an unordered list of names of all the Engineering Programmes (Branches) in your institution.**

**Procedure:**

1. Go to start- >all programs- >accessories->notepad
2. Type the html code
3. Use <ul> .....</ul> tag to display unordered list of names of all the branches
4. Inside <ul>.....</ul> tag use <li>.....</li> tag to list each branch
5. Go to file – >save –> save the file with html extension
6. Run the html code using browser

```
<html>
<head>
<title>program 11</title>
</head>
<body>
<ul>
<li> Computer science and Engg </li>
<li> Electronics and communication </li>
<li> Civil Engg </li>
<li> Mechanical Engg</li>
</ul>
</body>
</html>
```

**OUTPUT:**



- Computer science and Engg
- Electronics and communication
- Civil Engg
- Mechanical Engg

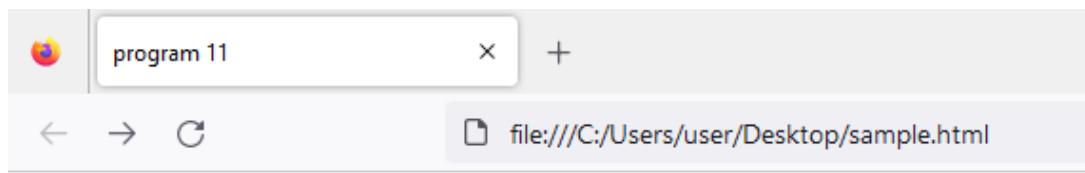
## EXPERIMENT: 12

Create a HTML document containing a nested list showing a content page of any book.

**Procedure:**

1. Go to start->all programs->accessories->notepad.
2. Type the html code.
3. Include <ol>tag for naming the chapter of the contents page.
4. Within the previous <ol>tag includes another <ol> tag to mention the different section of the particular chapter.
5. Use <li> tag for adding the list items.
6. Go to file->save->save the file with html extension.
7. Run the html code using browsers.

```
<html>
<head><title>program 12</title>
</head>
<body>
<ol type=1>
<li>chapter 1</li>
<ol type=1>
<li>section one</li>
<li>section two</li>
</ol type=1>
<li>chapter 2</li>
<ol type=1>
<li>section one</li>
<li>section two</li>
</ol>
</ol>
</body>
</html>
```

**OUTPUT:**

1. chapter 1  
    1. section one  
    2. section two  
2. chapter 2  
    1. section one  
    2. section two

**EXPERIMENT: 13**

Create the following table in HTML with Dummy Data:

Reg. Number	Student Name	Year/Semester	Date of Admission

**Procedure:**

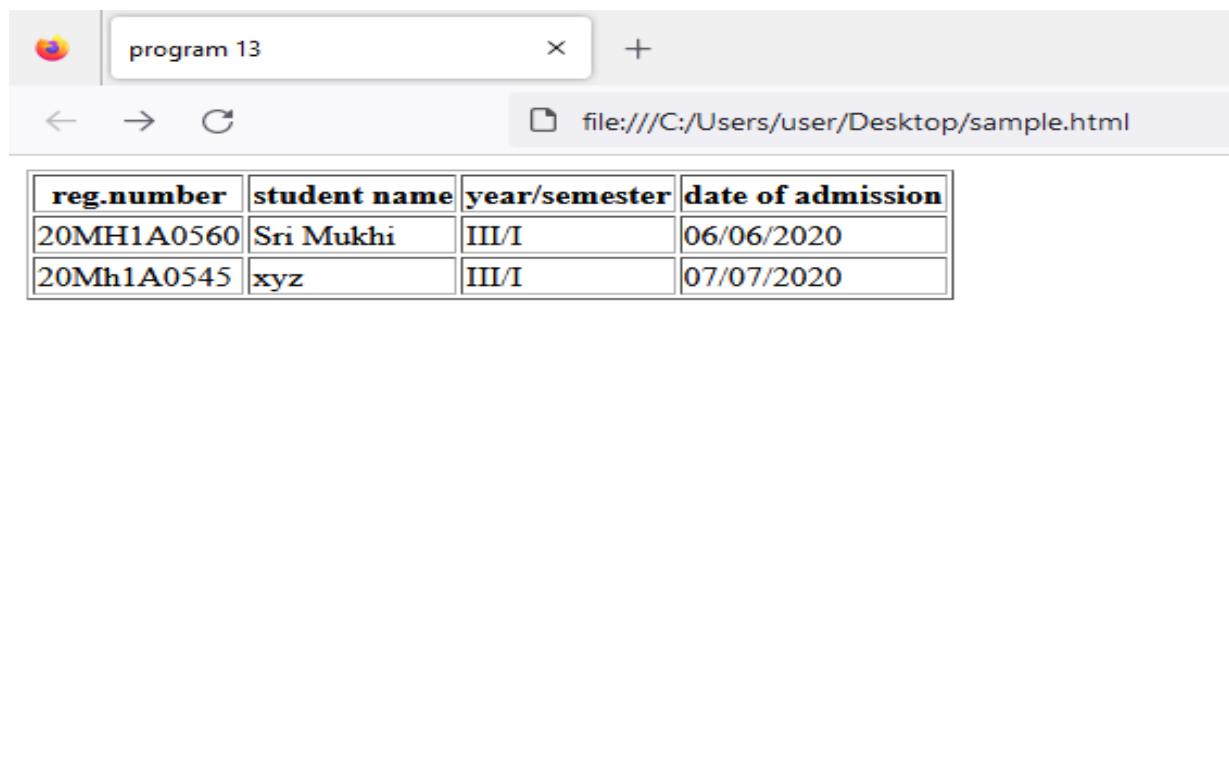
1. Go to start- all programs- accessories-notepad
2. Type the html code
3. Use `<table>...</table>` tag to draw table
4. Create 3 rows in the table by using `<tr>...</tr>` tag
5. In the first row create table heading by using `<th>.....</th>` tag
6. In the next 2 rows add the data by using `<th>.....</th>` tag
7. Go to file → save → save the with.html extension]
8. Run the html file using firefox browser

```

<html>
<head>
<title>program 13</title>
</head>
<body>
<table border= "1">
<tr>
<th> reg.number</th>
<th> student name</th>
<th> year/semester </th>
<th>date of admission</th>
</tr>

```

```
<tr>
<td> 20MH1A0560</td>
<td> Sri Mukhi</td>
<td> III/II</td>
<td> 06/06/2020 </td>
</tr>
<tr>
<td> 20MH1A0560</td>
<td> xyz</td>
<td> III/II </td>
<td> 07/07/2020 </td>
</tr>
</table>
</body>
</html>
```

**OUTPUT:**

**EXPERIMENT: 14****Create a HTML Page for Time Table**

```

<!DOCTYPE html>
<html>
<head>
    <title>II YEAR I SEM TIME TABLE</title>
</head>
<body>
    <h1 ALIGN="CENTER">TIME TABLE</h1>
    <table border="3" bordercolor="Pink" cellpadding="13" align="center" >
        <tr>
            <td bgcolor="Yellow">Day/Period</td>
            <td bgcolor="Lightgreen"><center>I<br>9:30-10:20</center></td>
            <td bgcolor="Lightgreen"><center>II<br>10:20-11:10</center></td>
            <td bgcolor="Lightgreen"><center>III<br>11:10-12:00</center></td>
            <td bgcolor="Lightgreen"><center>12:00-12:40</center></td>
            <td bgcolor="Lightgreen"><center>IV<br>12:40-1:30</center></td>
            <td bgcolor="Lightgreen"><center>V<br>1:30-2:20</center></td>
            <td bgcolor="Lightgreen"><center>VI<br>2:20-3:10</center></td>
            <td bgcolor="Lightgreen"><center>VII<br>3:10-4:00</center></td>
        </tr>
        <tr bgcolor="pink">
            <td bgcolor="skyblue"><center>Monday</center></td>
            <td ><center>Mat</center></td>
            <td ><center>SE</center></td>
            <td ><center>OS</center></td>
            <td rowspan="6"
bgcolor="white"><center>L<br>U<br>N<br>C<br>H</center></td>
            <td colspan="3"><center>SOC LAB</center></td>
            <td ><center>C++</center></td>
        </tr>
        <tr bgcolor="pink">
            <td bgcolor="skyblue"><center>Tuesday</center></td>
            <td colspan="3"><center>C++ LAB</center></td>
        </tr>
    </table>
</body>

```

```
<td><center>SE</center></td>
<td><center>OS</center></td>
<td><center>C++</center></td>
<td><center>Sports</center></td>
</tr>
<tr bgcolor="pink">
    <td bgcolor="skyblue"><center> Wednesday </center></td>
    <td><center>Mat</center></td>
    <td><center>SE</center></td>
    <td><center>OS</center></td>
    <td><center>C++</center></td>
    <td colspan="3"><center>LIBRARY</center></td>
</tr>
<tr bgcolor="pink">
    <td bgcolor="skyblue"><center>Thursday</center></td>
    <td><center>OS</center></td>
    <td><center>SE</center></td>
    <td><center>C++</center></td>
    <td colspan="3"><center>OS Lab</center></td>
    <td><center>Mat</center></td>
</tr>
<tr bgcolor="pink">
    <td bgcolor="skyblue"><center>Frinday</center></td>
    <td colspan="3"><center>OS LAB</center></td>
    <td><center>Mat</center></td>
    <td><center>OS</center></td>
    <td><center>SE</center></td>
    <td><center>C++</center></td>
</tr>
<tr bgcolor="pink">
    <td bgcolor="skyblue"><center>Saturday</center></td>
    <td><center>SE</center></td>
    <td><center>OS</center></td>
    <td><center>C++</center></td>
    <td colspan="3"><center>SEMINAR</center></td>
```

```

<td><center>Sports</center></td>
</tr>
</table>
</body>
</html>

```

## OUTPUT:

II YEAR I SEM TIME TABLE

file:///E:/HTML/Week 7/timetable.html

**TIME TABLE**

Day/Period	I 9:30-10:20	II 10:20-11:10	III 11:10-12:00	12:00-12:40	IV 12:40-1:30	V 1:30-2:20	VI 2:20-3:10	VII 3:10-4:00	
Monday	Mat	SE	OS	L U N C H	SOC LAB		C++		
Tuesday	C++ LAB				SE	OS	C++	Sports	
Wednesday	Mat	SE	OS		LIBRARY				
Thursday	OS	SE	C++		OS Lab		Mat		
Frinday	OS LAB				Mat	OS	SE	C++	
Saturday	SE	OS	C++		SEMINAR			Sports	



## Introduction to CSS:

- CSS stands for Cascading Style Sheet.
- CSS is used to design HTML tags.
- CSS is a widely used language on the web.
- Inline CSS
- We can apply CSS in a single element by inline CSS technique.
- The inline CSS is also a method to insert style sheets in HTML document. This method mitigates some advantages of style sheets so it is advised to use this method sparingly.
- If you want to use inline CSS, you should use the style attribute to the relevant tag.

## Disadvantages of Inline CSS

- You cannot use quotations within inline CSS. If you use quotations the browser will interpret this as an end of your style value.
- These styles cannot be reused anywhere else.
- These styles are tough to be edited because they are not stored at a single place.
- It is not possible to style pseudo-codes and pseudo-classes with inline CSS.
- Inline CSS does not provide browser cache advantages.

1. Syntax: <htmltag style="cssproperty1:value; cssproperty2:value;"> </htmltag>

```
<html>
<body>
<h1 style="color:red;margin-left:40px;">Inline CSS is applied on this heading.</h1>
<p>This paragraph is not affected.</p>
</body>
</html>
```

## Internal CSS:

The internal style sheet is used to add a unique style for a single document. It is defined in <head> section of the HTML page inside the <style> tag.

```
<html>
<head>
<style>
```

```

body {
    background-color: linen;
}
h1 {
    color: red;
    margin-left: 80px;
}
</style>
</head>
<body>
<h1>The internal style sheet is applied on this heading.</h1>
<p>This paragraph will not be affected.</p>
</body>
</html>

```

### **External CSS**

The external style sheet is generally used when you want to make changes on multiple pages. It is ideal for this condition because it facilitates you to change the look of the entire web site by changing just one file.

It uses the `<link>` tag on every pages and the `<link>` tag should be put inside the head section.

The external style sheet may be written in any text editor but must be saved with a .css extension. This file should not contain HTML elements.

```

<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>

```

File: mystyle.css

```

body {
    background-color: lightblue;
}
h1 {
    color: navy;
    margin-left: 20px;
}

```

## EXPERIMENT: 15

**Create a web page using Embedded CSS and multimedia**

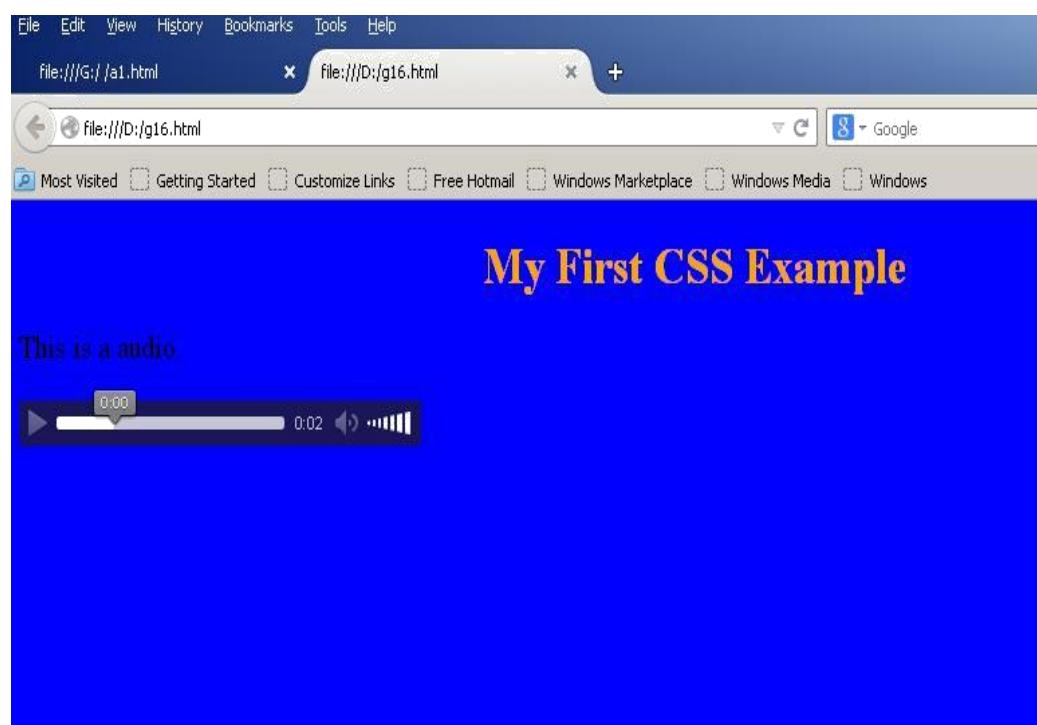
**Procedure:**

1. Go to start->all programs->accessories->notepad
2. Type the html code
3. Use <style> tag for css to apply styles for body, h1, p tags
4. Include audio file by using <audio> tag
5. Go to file ->save -> save the file with html extension
6. Run the html file using firefox browser

```
<html>
<body>
<style>
body {
    background-color:blue;
}
h1 {
    color:orange;
    text-align:center;
}
p {
    font-family: "Times New Roman";
    font-size: 20px;
}
</style>
<h1>My First CSS Example</h1>
<p>This is a audio.</p>
<audio controls>
<source src="Horse-neighing.mp3">
</audio>
```

```
</body>
```

```
</html>
```

**OUTPUT:**

## EXPERIMENT: 16

Create a HTML web page using CSS with background properties

**CSS Background:**

CSS background property is used to define the background effects on element. There are 5 CSS background properties that affects the HTML elements:

background-color

background-image

background-repeat

background-attachment

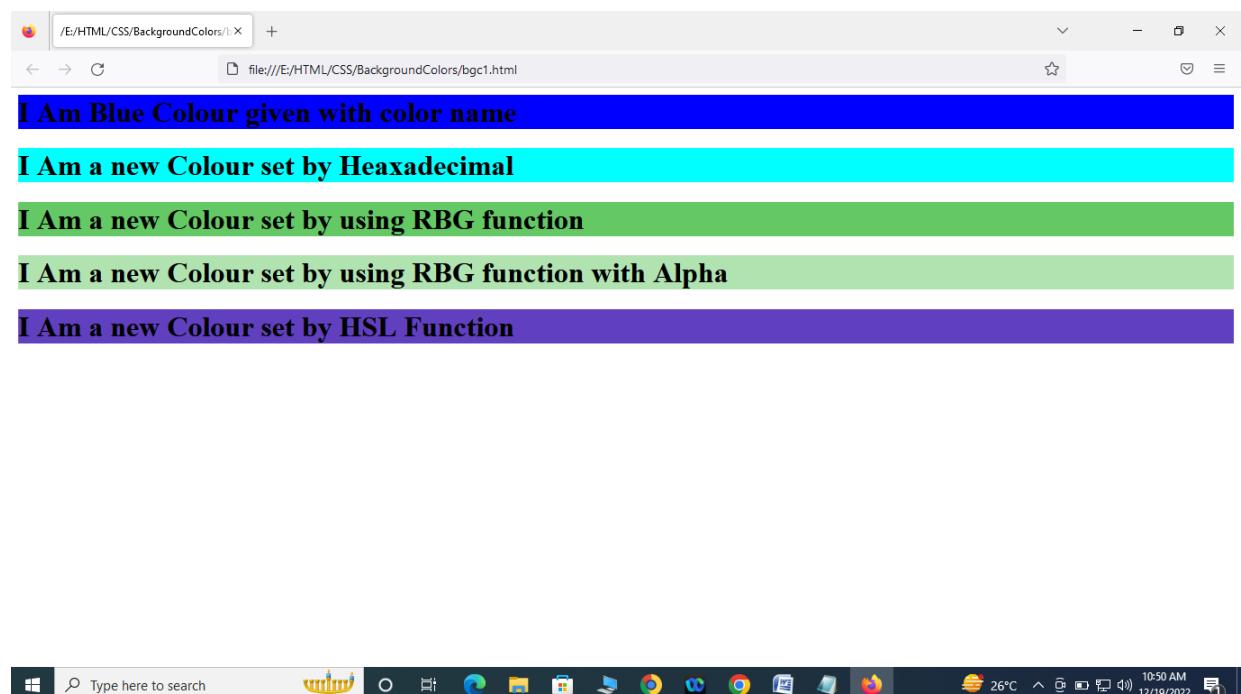
background-position

1) CSS background-color

The background-color property is used to specify the background color of the element.

You can set the background color like this:

```
<html>
<body>
<h1 style=background-color:blue>I Am Blue Colour given with color name</h1>
<h1 style=background-color:#00FFFF>I Am a new Colour set by Hexadecimal</h1>
<h1 style=background-color:RGB(100,200,100)>I Am a new Colour set by using RGB
function</h1>
<h1 style=background-color:RGB(100,200,100,0.5)>I Am a new Colour set by using RBG
function with Alpha</h1>
<h1 style=background-color:HSL(255,50%,50%)>I Am a new Colour set by HSL Function</h1>
</body>
</html>
```

**OUTPUT:****2) CSS Borders**

The CSS border is a shorthand property used to set the border on an element.

The CSS border properties are used to specify the style, color and size of the border of an element. The CSS border properties are given below

- border-style
- border-color
- border-width
- border-radius

**1) CSS border-style.**

1.1K

Hello Java Program for Beginners

Value	Description
None	It doesn't define any border.
Dotted	It is used to define a dotted border.
Dashed	It is used to define a dashed border.

Solid	It is used to define a solid border.
Double	It defines two borders with the same border-width value.
Groove	It defines a 3d grooved border. effect is generated according to border-color value.
Ridge	It defines a 3d ridged border. effect is generated according to border-color value.
Inset	It defines a 3d inset border. effect is generated according to border-color value.
Outset	It defines a 3d outset border. effect is generated according to border-color value.

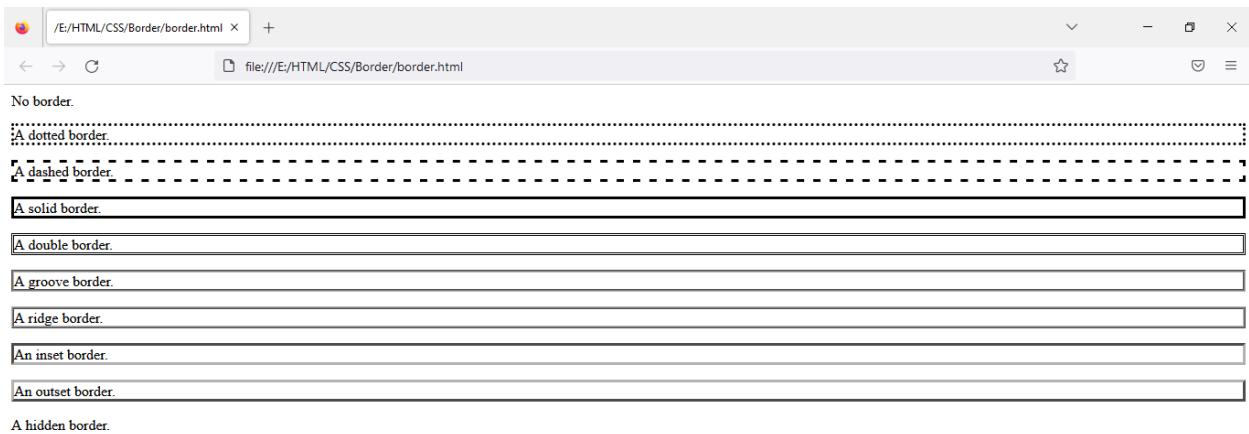
**Example:**

```

<html>
<head>
<style>
p.none {border-style: none;}
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.hidden {border-style: hidden;}
</style>
</head>
<body>
<p class="none">No border.</p>

```

## OUTPUT:



### 3) Background Image

The **background-image** property in CSS is used to set an image as the background of an element. Using this CSS property, we can set one or more than one background image for an element.

By default, the image is positioned at the top-left corner of an element and repeated both horizontally as well as vertically. The background image should be chosen according to the text color. The bad combination of text and background image may be a cause of poorly designed and not readable webpage.

The **url()** value of this property allows us to include a file path to any image. It will show the element's background. We can use multiple images or a mixture of gradients and images for the background. If the `background-image` is failed to load or if we are using the gradients, but they are not supported on the corresponding browser then, we can use the fallback value (the value used as the substitution) as the background color of the element.

#### Syntax

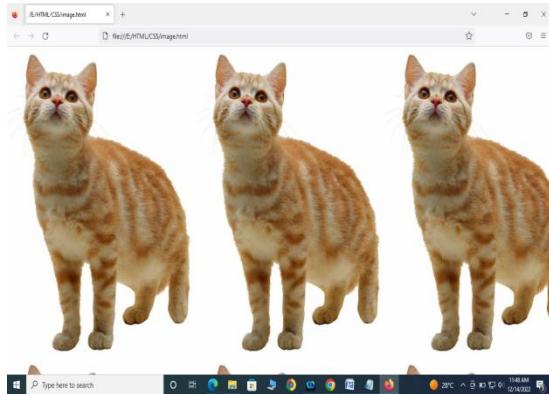
1. `background-image: url();`

#### Values

**url():** It is the URL to the image. We can separate the URLs by a comma if we want to specify more than one image.

```
<html>
<head>
<style>
body {
    background-image: url("img1.jpeg");
    background-color: lightgray;
}
```

```
</style>
</head>
<body>
</body>
</html>
```



```
<html>
<head>
<style>
body{background-color:green;
background-image: url("img1.jpg");
background-repeat:no-repeat;
background-position:center;
background-attachment:fixed;
border-style:dashed double solid dotted
}
</style>
</head>
<body>
<h1>Hi</h1>
<h1>Hi</h1>
<h1>Hi</h1>
<h1>Hi</h1>
```

## **OUTPUT:**



**EXPERIMENT: 17**

**Write HTML web page to set margins for content using CSS Margins**

**CSS Margins:** CSS Margin property is used to define the space around elements. It is completely transparent and doesn't have any background color. It clears an area around the element.

CSS Margin Properties

Property	Description
Margin	This property is used to set all the properties in one declaration.
margin-left	It is used to set left margin of an element.
margin-right	It is used to set right margin of an element.
margin-top	It is used to set top margin of an element.
margin-bottom	It is used to set bottom margin of an element.

CSS Margin Values

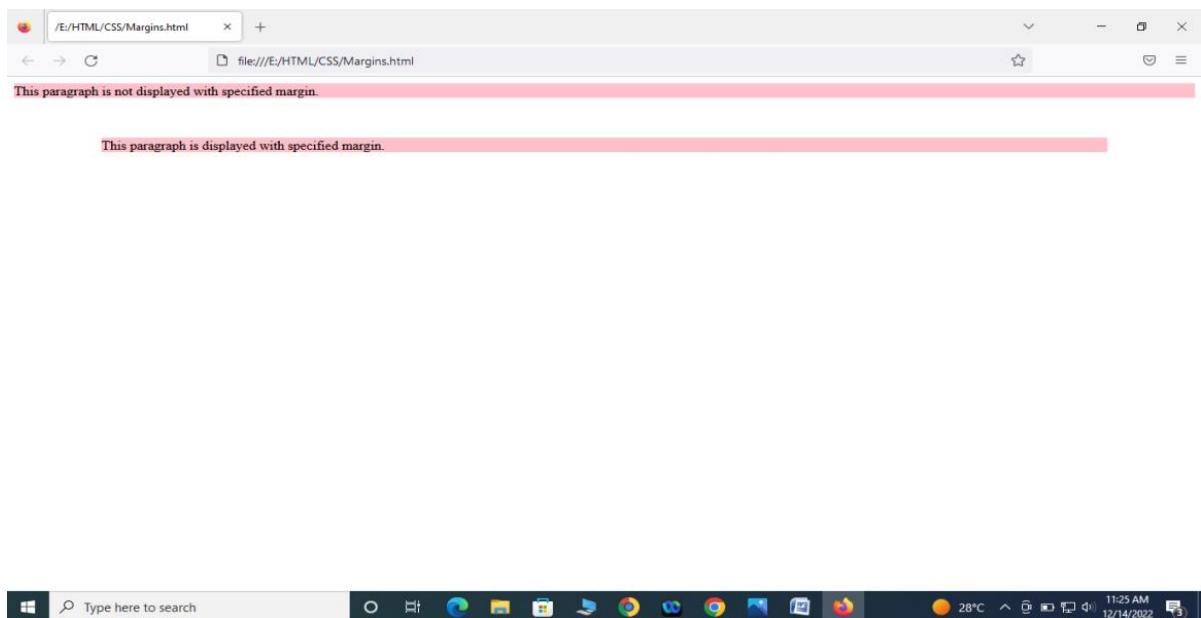
These are some possible values for margin property.

Value	Description
Auto	This is used to let the browser calculate a margin.
Length	It is used to specify a margin pt, px, cm, etc. its default value is 0px.
%	It is used to define a margin in percent of the width of containing element.
Inherit	It is used to inherit margin from parent element.

```
<html>
<head>
<style>
```

```
p {  
    background-color: pink;  
}  
  
p.ex {  
  
    margin-top: 50px;  
    margin-bottom: 50px;  
    margin-right: 100px;  
    margin-left: 100px;  
}  
  
</style>  
  
</head>  
  
<body>  
  
<p>This paragraph is not displayed with specified margin. </p>  
  
<p class="ex">This paragraph is displayed with specified margin.</p>  
  
</body>  
  
</html>
```

## OUTPUT:



## EXPERIMENT: 18

**Write HTML web page to set padding for content using CSS Puddings**

**CSS Padding property** is used to define the space between the element content and the element border.

It is different from CSS margin in the way that CSS margin defines the space around elements. CSS padding is affected by the background colors. It clears an area around the content.

CSS Padding Properties

Property	Description
Padding	It is used to set all the padding properties in one declaration.
padding-left	It is used to set left padding of an element.
padding-right	It is used to set right padding of an element.
padding-top	It is used to set top padding of an element.
padding-bottom	It is used to set bottom padding of an element.

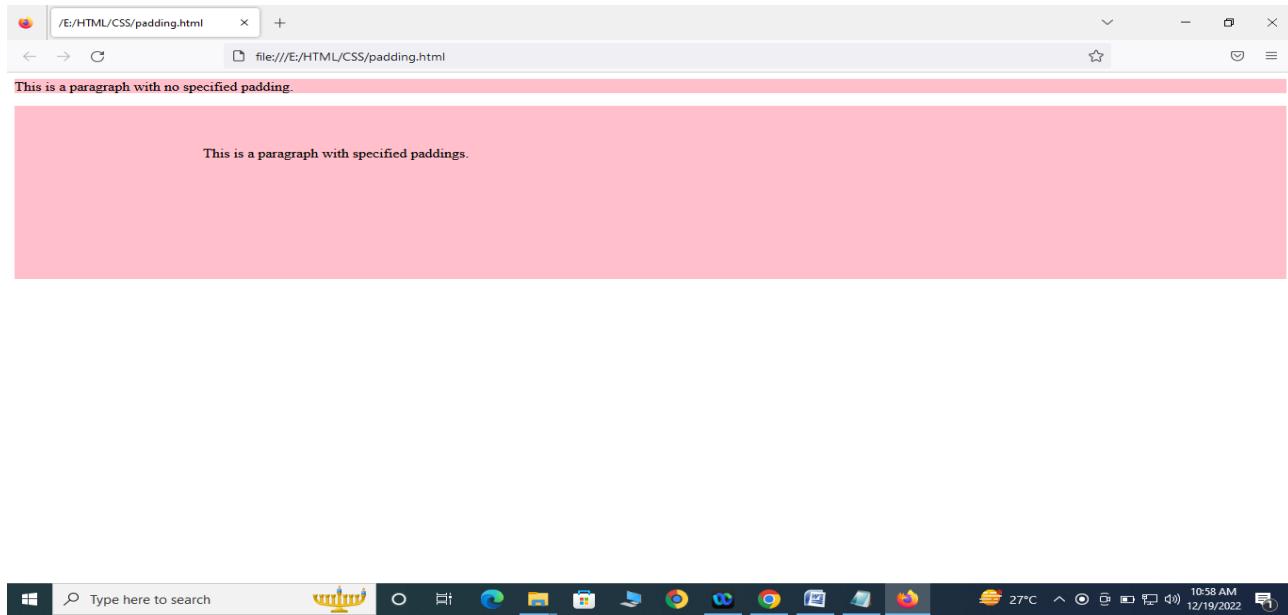
CSS Padding Values

Value	Description
Length	It is used to define fixed padding in pt, px, em etc.
%	It defines padding in % of containing element.

```
<html>
<head>
<style>
p {
    background-color: pink;
}
```

```
p.padding {  
    padding-top: 50px;  
    padding-right: 100px;  
    padding-bottom: 150px;  
    padding-left: 200px;  
}  
</style>  
</head>  
<body>  
<p>This is a paragraph with no specified padding.</p>  
<p class="padding">This is a paragraph with specified paddings.</p>  
</body>  
</html>
```

## OUTPUT:



## EXPERIMENT: 19

### **Write HTML web page to show Text Effects using CSS**

#### CSS Text Effects

We can apply different effects on the text used within an HTML document. Some properties can be used for adding the effects on text.

Using CSS, we can style the web documents and affects the text. The properties of the text effect help us to make the text attractive and clear. There are some text effect properties in CSS that are listed below:

- word-break
- text-overflow
- word-wrap
- writing-mode

#### word-break

It specifies how words should break at the end of the line. It defines the line break rules.

#### Syntax

1. word-break: normal |keep-all | break-all | inherit ;

The default value of this property is normal. So, this value is automatically used when we don't specify any value.

#### Values

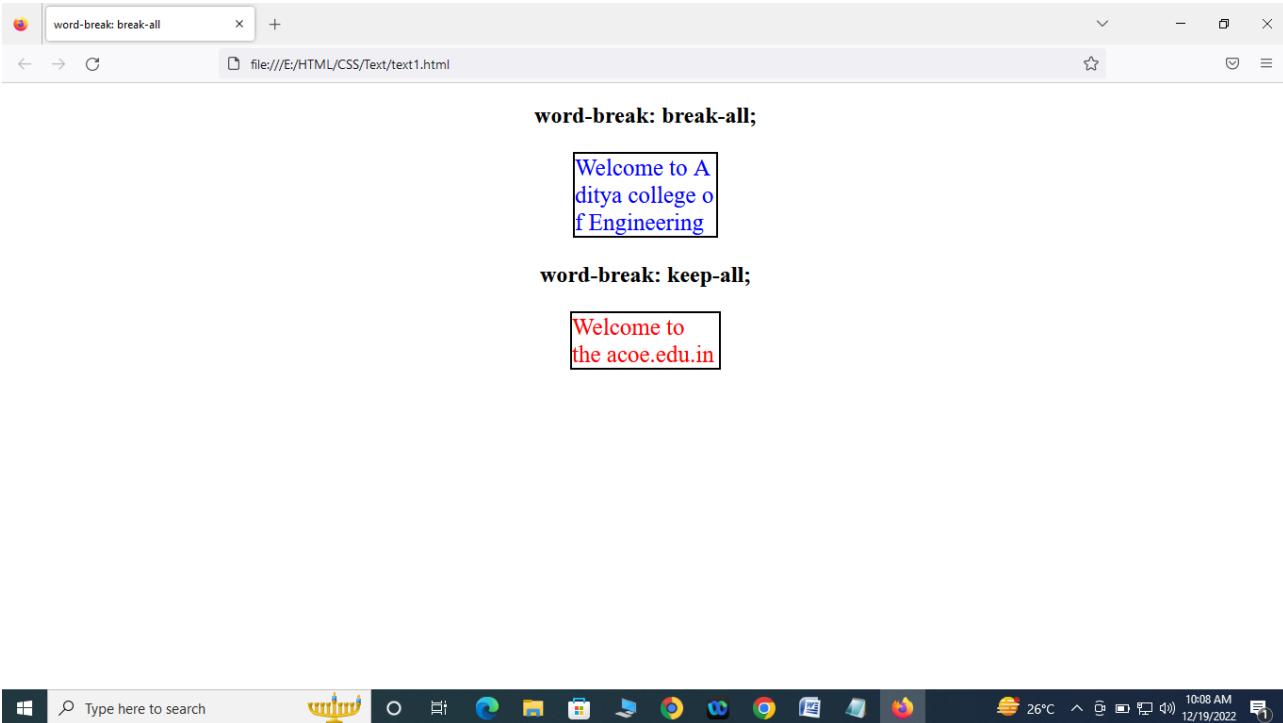
**keep-all:** It breaks the word in the default style.

**break-all:** It inserts the word break between the characters in order to prevent the word overflow.

```
<html>
  <head>
    <title>word-break: break-all</title>
  <style>
    .one{
      width: 150px;
      border: 2px solid black;
      word-break: break-all;
      text-align: left;
    }
  </style>
</head>
<body>
  <div class="one">A very long word that will be broken into multiple lines</div>
</body>
</html>
```

```
        font-size: 25px;  
  
        color: blue;  
  
    }  
  
.two{  
  
    width: 156px;  
  
    border: 2px solid black;  
  
    word-break: keep-all;  
  
    text-align: left;  
  
    font-size: 25px;  
  
    color: red;  
  
}  
  
</style>  
  
</head>  
  
<center>  
  
<body>  
  
<h2>word-break: break-all;</h2>  
  
<p class="one">  
  
    Welcome to Aditya college of Engineering  
  
</p>  
  
<h2>word-break: keep-all;</h2>  
  
<p class="two">  
  
    Welcome to the acoe.edu.in  
  
</p>  
  
</center>  
  
</body>  
  
</html>
```

## OUTPUT:



### text-overflow

It specifies the representation of overflowed text, which is not visible to the user. It signals the user about the content that is not visible. This property helps us to decide whether the text should be clipped or show some dots (ellipsis).

This property does not work on its own. We have to use **white-space: nowrap;** and **overflow: hidden;** with this property.

#### Syntax

1. text-overflow: clip | ellipsis;

#### Property Values

**clip:** It is the default value that clips the overflowed text.

**ellipsis:** This value displays an ellipsis (...) or three dots to show the clipped text. It is displayed within the area, decreasing the amount of text.

```
<html>
  <head>
    <style>
      .jtp{
        white-space: nowrap;
        height: 30px;
      }
    </style>
  </head>
  <body>
    <div class="jtp">
      Welcome to Aditya College of Engineering
    </div>
  </body>
</html>
```

```
width: 250px;  
  
overflow: hidden;  
  
border: 2px solid black;  
  
font-size: 25px;  
  
text-overflow: clip;  
  
}  
  
.jtp1 {  
  
white-space: nowrap;  
  
height: 30px;  
  
width: 250px;  
  
overflow: hidden;  
  
border: 2px solid black;  
  
font-size: 25px;  
  
text-overflow: ellipsis;  
  
}  
  
h2{  
  
color: blue;  
  
}  
  
div:hover {  
  
overflow: visible;  
  
}  
  
p{  
  
font-size: 25px;  
  
font-weight: bold;  
  
color: red;  
  
}  
  
</style>  
  
</head>
```

```
<center>
<body>

<p> Hover over the bordered text to see the full content. </p>

<h2>
    text-overflow: clip;
</h2>

<div class="jtp">
    Welcome to Aditya College of Engineering
</div>

<h2>
    text-overflow: ellipsis;
</h2>

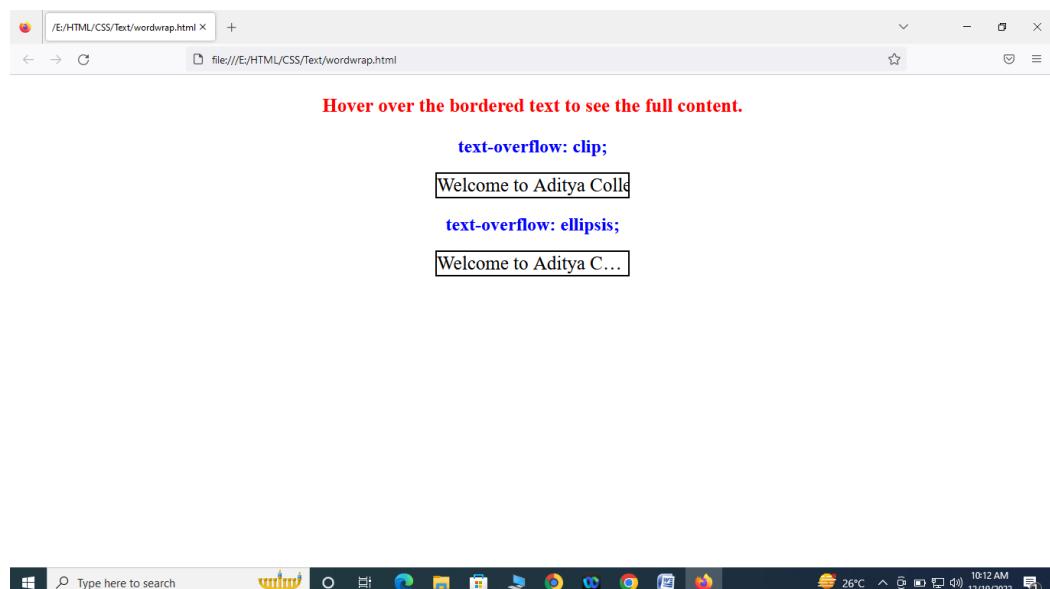
<div class="jtp1">
    Welcome to Aditya College of Engineering
</div>

</center>

</body>

</html>
```

## OUTPUT:



## EXPERIMENT: 20

**Write HTML web page to show all CSS font properties**

### CSS Font

CSS Font property is used to control the look of texts. By the use of CSS font property you can change the text size, color, style and more. You have already studied how to make text bold or underlined. Here, you will also know how to resize your font using percentage.

These are some important font attributes:

1. **CSS Font color:** This property is used to change the color of the text. (standalone attribute)
2. **CSS Font family:** This property is used to change the face of the font.
3. **CSS Font size:** This property is used to increase or decrease the size of the font.
4. **CSS Font style:** This property is used to make the font bold, italic or oblique.
5. **CSS Font variant:** This property creates a small-caps effect.
6. **CSS Font weight:** This property is used to increase or decrease the boldness and lightness of the font.

```
<html>
<head>
<style>
.p1 {
    font-family: "Times New Roman", Times, serif;
}
.p2 {
    font-family: Arial, Helvetica, sans-serif;
}
.p3 {
    font-family: "Lucida Console", "Courier New", monospace;
}
</style>
</head>
```

```
<body>  
  
<h1>CSS font-family</h1>  
  
<p class="p1">This is a paragraph, shown in the Times New Roman font.</p>  
  
<p class="p2">This is a paragraph, shown in the Arial font.</p>  
  
<p class="p3">This is a paragraph, shown in the Lucida Console font.</p>  
  
</body>  
  
</html>
```

## OUTPUT:



### CSS font-family

This is a paragraph, shown in the Times New Roman font.  
This is a paragraph, shown in the Arial font.  
This is a paragraph, shown in the Lucida Console font.



## EXPERIMENT: 21

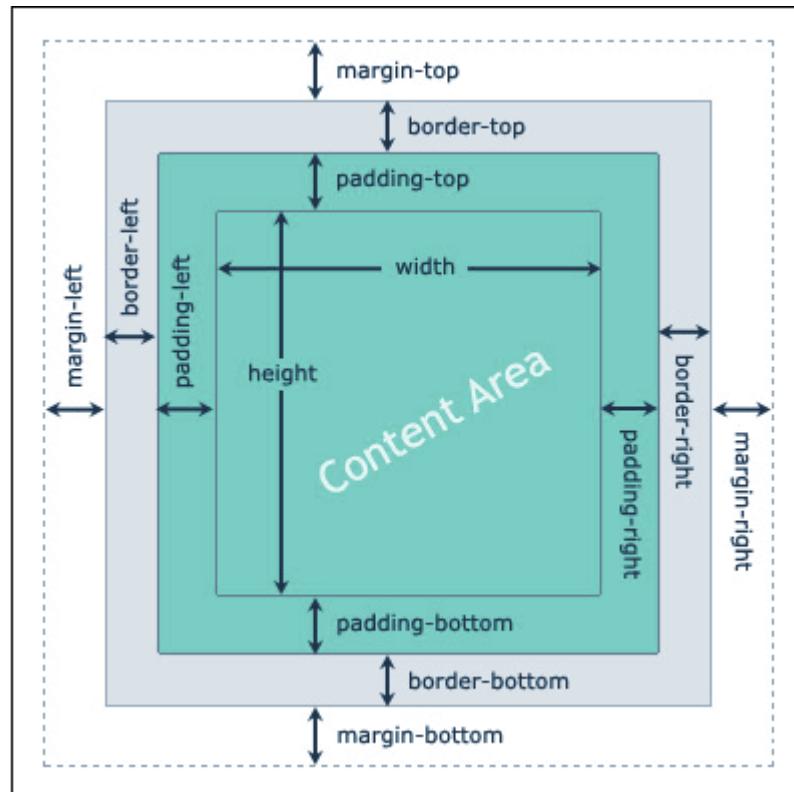
**Write HTML web page to Demonstrate CSS Box Properties**

### CSS Box Model

The components that can be depicted on the web page consist of one or more than one rectangular box.

A CSS box model is a compartment that includes numerous assets, such as edge, border, padding and margin. It is used to develop the design and structure of a web page. It can be used as a set of tools to personalize the layout of different components. According to the CSS box model, the web browser supplies each element as a square prism.

The following diagram illustrates how the CSS properties of width, height, padding, border and margin dictate that how much space an attribute will occupy on a web page.



The CSS box model contains the different properties in CSS. These are listed below.

The CSS box model contains the different properties in CSS. These are listed below.

- **Border**
- **Margin**
- **Padding**

- **Content**
- **Border Field**
- It is a region between the padding-box and the margin. Its proportions are determined by the width and height of the boundary.
- **Margin Field**
- This segment consists of the area between the boundary and the edge of the border.
- The proportion of the margin region is equal to the margin-box width and height. It is better to separate the product from its neighbor nodes.
- **Padding Field**
- This field requires the padding of the component. In essence, this area is the space around the subject area and inside the border-box. The height and the width of the padding box decide its proportions.
- **Content Field**
- Material such as text, photographs, or other digital media is included in this area.
- It is constrained by the information edge, and its proportions are dictated by the width and height of the content enclosure.

```

<head>
<title>CSS Box Model</title>
<style>
    .main
    {
        font-size:30px;
        font-weight:bold;
        text-align:center;
    }
    .gfg
    {
        margin-left:50px;
        border:50px solid Purple;
        width:300px;
        height:200px;
        text-align:center;
        padding:50px;
    }

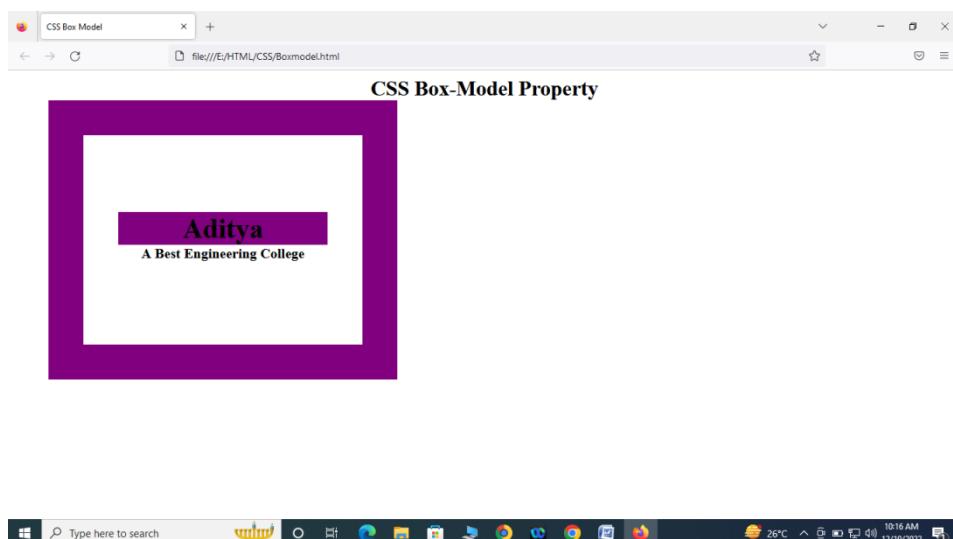
```

```
.gfg1
{
    font-size:40px;
    font-weight:bold;
    color:black;
    margin-top:60px;
    background-color:purple;
}

.gfg2
{
    font-size:20px;
    font-weight:bold;
    background-color:white;
}

</style>
</head>
<body>
<div class = "main">CSS Box-Model Property</div>
<div class = "gfg">
<div class = "gfg1">Aditya</div>
<div class = "gfg2">A Best Engineering College</div>
</div>
</body>
</html>
```

## OUTPUT:



**EXPERIMENT: 22**

**Write HTML web pages to print any two sample forms**

### **HTML Form**

An **HTML form** is a section of a document which contains controls such as text fields, password fields, checkboxes, radio buttons, submit button, menus etc.

An HTML form facilitates the user to enter data that is to be sent to the server for processing such as name, email address, password, phone number, etc. .

---

### **Why use HTML Form**

HTML forms are required if you want to collect some data from the site visitor.

For example: If a user want to purchase some items on internet, he/she must fill the form such as shipping address and credit/debit card details so that item can be sent to the given address.

### **HTML Form Syntax**

```
<form action="server url" method="get|post">
  //input controls e.g. textfield, textarea, radiobutton, button
</form>
```

### **HTML Form Tags**

Let's see the list of HTML 5 form tags.

<b>Tag</b>	<b>Description</b>
<form>	It defines an HTML form to enter inputs by the user side.
<input>	It defines an input control.
<textarea>	It defines a multi-line input control.
<label>	It defines a label for an input element.
<fieldset>	It groups the related elements in a form.
<legend>	It defines a caption for a <fieldset> element.

<select>	It defines a drop-down list.
<optgroup>	It defines a group of related options in a drop-down list.
<option>	It defines an option in a drop-down list.
<button>	It defines a clickable button.

### HTML <form> element

The HTML <form> element provide a document section to take input from user. It provides various interactive controls for submitting information to web server such as text field, text area, password field, etc.

*Note: The <form> element does not itself create a form but it is container to contain all required form elements, such as <input>, <label>, etc.*

#### Syntax:

```
<form>
```

//Form elements

```
</form>
```

### HTML <input> element

The HTML <input> element is fundamental form element. It is used to create form fields, to take input from user. We can apply different input filed to gather different information form user. Following is the example to show the simple text input.

#### HTML TextField Control

The type="text" attribute of input tag creates textfield control also known as single line textfield control. The name attribute is optional, but it is required for the server side component such as JSP, ASP, PHP etc.

#### HTML <textarea> tag in form

The <textarea> tag in HTML is used to insert multiple-line text in a form. The size of <textarea> can be specify either using "rows" or "cols" attribute or by CSS.

#### Label Tag in Form

It is considered better to have label in form. As it makes the code parser/browser/user friendly.

If you click on the label tag, it will focus on the text control. To do so, you need to have for attribute in label tag that must be same as id attribute of input tag.

### HTML Password Field Control

The password is not visible to the user in password field control.

```
<input type="password" id="password" name="password"/>
```

### Radio Button Control

The radio button is used to select one option from multiple options. It is used for selection of gender, quiz questions etc.

If you use one name for all the radio buttons, only one radio button can be selected at a time.

Using radio buttons for multiple options, you can only choose a single option at a time.

### Checkbox Control

The checkbox control is used to check multiple options from given checkboxes.

### Submit button control

HTML **<input type="submit">** are used to add a submit button on web page. When user clicks on submit button, then form get submit to the server.

Syntax:

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

HTML **<fieldset>** element:

The **<fieldset>** element in HTML is used to group the related information of a form. This element is used with **<legend>** element which provide caption for the grouped elements.

### HTML Form Example

```
<html>
<head>
<title>Form in HTML</title>
</head>
<body>
<h2>Registration form</h2>
<form>
<fieldset>
<legend>User personal information</legend>
<label>Enter your full name</label><br>
<input type="text" name="name"><br>
```

```

<label>Enter your email</label><br>
<input type="email" name="email"><br>
<label>Enter your password</label><br>
<input type="password" name="pass"><br>
<label>confirm your password</label><br>
<input type="password" name="pass"><br>
<br><label>Enter your gender</label><br>
<input type="radio" id="gender" name="gender" value="male"/>Male <br>
<input type="radio" id="gender" name="gender" value="female"/>Female <br/>
<input type="radio" id="gender" name="gender" value="others"/>others <br/>
<br>Enter your Address:<br>
<textarea></textarea><br>
<input type="submit" value="sign-up">
</fieldset>
</form>
</body>
</html>

```

Test it Now

### **Output:**

HTML Form Example

HTML Form Example

Let's see a simple example of creating HTML form.

```

<form action="#">
<table>
<tr>
  <td class="tdLabel"><label for="register_name" class="label">Enter name:</label></td>
  <td><input type="text" name="name" value="" id="register_name" style="width:160px"/></td>
</tr>
<tr>
  <td class="tdLabel"><label for="register_password" class="label">Enter
password:</label></td>
  <td><input type="password" name="password" id="register_password"
style="width:160px"/></td>
</tr>

```

```
<tr>
<td class="tdLabel"><label for="register_email" class="label">Enter Email:</label></td>
<td>
<input type="email" name="email" value="" id="register_email" style="width:160px"/></td>
</tr>
<tr>
<td class="tdLabel"><label for="register_gender" class="label">Enter Gender:</label></td>
<td>
<input type="radio" name="gender" id="register_gendermale" value="male"/>
<label for="register_gendermale">male</label>
<input type="radio" name="gender" id="register_genderfemale" value="female"/>
<label for="register_genderfemale">female</label>
</td>
</tr>
<tr>
<td class="tdLabel"><label for="register_country" class="label">Select Country:</label></td>
<td><select name="country" id="register_country" style="width:160px">
<option value="india">india</option>
<option value="pakistan">pakistan</option>
<option value="africa">africa</option>
<option value="china">china</option>
<option value="other">other</option>
</select>
</td>
</tr>
<tr>
<td colspan="2"><div align="right"><input type="submit" id="register_0" value="register"/>
</div></td>
</tr>
</table>
</form>
```

**OUTPUT:**

**Registration form**

User personal information

Enter your full name

Enter your email

Enter your password

confirm your password

Enter your gender

Male  
 Female  
 others

Enter your Address:

**sign-up**

Test it Now Output: HTML Form Example HTML Form Example Let's see a simple example of creating HTML form.

Enter name:

Enter password:

Enter Email:

Enter Gender:  male  female

Select Country:

**register**

```

<html>
<head>
<title>
    Registration Page
</title>
</head>
<body bgcolor="Lightskyblue">
<br>
<br>
<form>
    <label> Firstname </label>
    <input type="text" name="firstname" size="15"/> <br> <br>
    <label> Middlename: </label>
    <input type="text" name="middlename" size="15"/> <br> <br>
    <label> Lastname: </label>
    <input type="text" name="lastname" size="15"/> <br> <br>
    <label>

```

**Course :**

```
</label>
<select>
<option value="Course">Course</option>
<option value="BCA">BCA</option>
<option value="BBA">BBA</option>
<option value="B.Tech">B.Tech</option>
<option value="MBA">MBA</option>
<option value="MCA">MCA</option>
<option value="M.Tech">M.Tech</option>
</select>
```

```
<br>
```

```
<br>
```

```
<label>
```

Gender :

```
</label><br>
<input type="radio" name="male"/> Male <br>
<input type="radio" name="female"/> Female <br>
<input type="radio" name="other"/> Other
```

```
<br>
```

```
<br>
```

```
<label>
```

Phone :

```
</label>
<input type="text" name="country code" value="+91" size="2"/>
<input type="text" name="phone" size="10"/> <br> <br>
```

Address

```
<br>
<textarea cols="80" rows="5" value="address">
</textarea>
<br> <br>
```

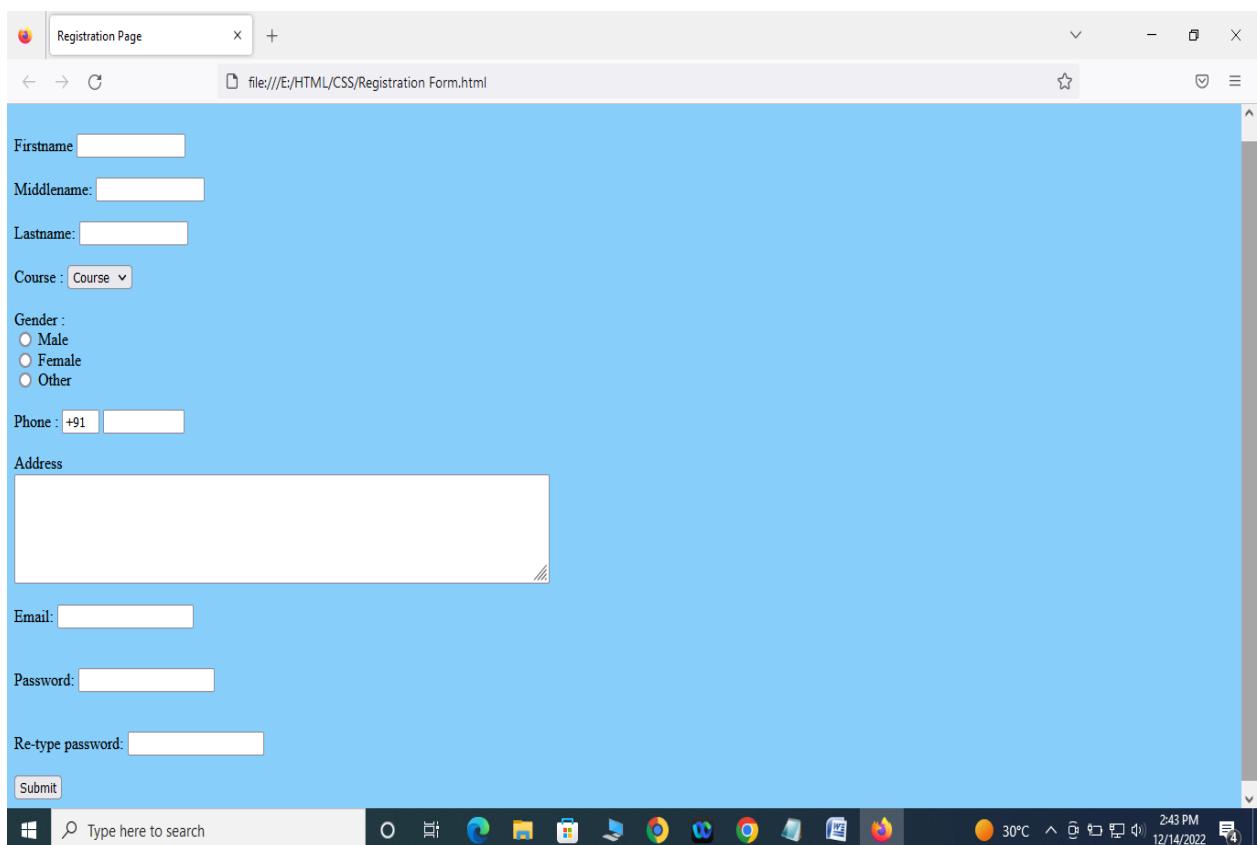
Email:

```
<input type="email" id="email" name="email"/> <br>
<br> <br>
```

Password:

```
<input type="Password" id="pass" name="pass"> <br>
<br> <br>
Re-type password:
<input type="Password" id="repass" name="repass"> <br> <br>
<input type="button" value="Submit"/>
</form>
</body>
</html>
```

## OUTPUT:



## EXPERIMENT: 23

**Write HTML web page to show HTML simple Animation**

```
<html>
<head>
<style>
div {
    width: 100px;
    height: 100px;
    background: red;
    -webkit-animation: myfirst 6s; /* Chrome, Safari, Opera */
    animation: myfirst 5s;
}
/* Chrome, Safari, Opera */
@-webkit-keyframes myfirst {
    from {background: red;}
    to {background: black;}
}
/* Standard syntax */
@keyframes myfirst {
    from {background: red;}
    to {background: black;}
}
</style>
</head>
<body>
<p><b>Note:</b> The IE 9 and earlier versions don't support CSS3 animation property. </p>
```

```
<p>See the rectangle background from RED to BLACK.</p>  
  
<div></div>  
  
</body>  
  
</html>
```

**OUTPUT:**