ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇਵ
ਪੰਜਾਬ ਸਟੇਟ ਓਪਨ ਯੂਨੀਵਰਸਿਟੀ
ਪਟਿਆਲਾ

The Motto of Our University
(SEWA)
SKILL ENHANCEMENT
EMPLOYABILITY
WISDOM
ACCESSIBILITY

JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

Bachelor of Arts

BAB32408T
WEB DESIGNING AND DEVELOPMENT

Head Quarter: C/28, The Lower Mall, Patiala-147001
WEBSITE: www.psou.ac.in

JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY,PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

WEB DESIGNING AND DEVELOPMENT

Max. Marks: 100
External: 70
Internal: 30
Pass: 40%
Credits: 4

SECTION A

UNIT NO.	UNIT NAME
UNIT 1	INTRODUCTION TO HTML
UNIT 2	OVERVIEW OF HTML
UNIT 3	TAGS OF HTML
UNIT 4	LAYOUT
UNIT 5	LIST AND FRAMES

SECTION B

UNIT NO.	UNIT NAME
UNIT 6	STYLE SHEETS
UNIT 7	ATTRIBUTES OF STYLE SHEETS
UNIT 8	HTML FORMS
UNIT 9	SCRIPTING LANGUAGE
UNIT 10	DOCUMENT OBJECT MODEL

UNIT 1 INTRODUCTION TO HTML

Structure

1.0 INTRODUCTION

Internet is the most common now these days and it is used in all the fields of life. Now we can access a wide variety of data or information over websites using the internet. This can be possible only if we have an internet connection and a browser on our system. The most popular browsers which are used for accessing the websites are Internet Explorer, Mozilla Firefox, Google Chrome, Netscape Navigator, Opera. While connecting, a browser presents the information received from the server on the system. Once the file is displayed on the browser its contents are stored in an HTML format i.e., Hypertext Markup Language. HTML format can be interpreted by the browser that is displayed on the system.

The process to create a web design begins with visual concepts that can be represented either by hand or with any software. After the designing process, designers can use HTML or CSS to build a website. HTML manages the basic structure of the code while CSS manages the styles applied to the text on a website.

HTML is generalized from SGML i.e., Standard Generalized Markup Language. SGML is complex as compared to HTML. The basic purpose behind the invention of HTML is to create websites in a simple format and is easily accessed by browsers. HTML pages are created by using tags and data. The tags assist the browser with the kind of data and how the data appears on the browser. Different tags are available in HTML. These files have the extension of ".html"

or ".htm". For creating the web pages a tool is required to write the code.

HTML can also include different tags that helps in creating list, fonts, background images, images, hyperlinks, frames , forms, etc. that are discussed in later sections.

1.1 OBJECTIVES

The objective of this unit is to illustrate the foundation of HTML. After completing this unit, learners will be able to understand the fundamental concepts of HTML and they will be able to recognize the following features of web design:

- Introduction of HTML
- Overview and Basic concepts of HTML
- Web Design
- Process of Web Designing and Publishing
- Website Development Phases

This unit deals with the basics of web design and how the website will be created including the steps involved in designing and development.

1.2 What is HTML?

HTML stands for **Hyper-Text-Markup Language**. HTML isn't a programming language like C, C++, or another type of language. It is a web-based language that can run only on a browser; therefore, it is referred to as a scripting language. HTML is used for designing web pages. It acts as the basic building block to create any website that defines the appearance and contents of a webpage. It consists of a series of elements that informs the browser, how to display the content.

In **HTML**, each letter has a specific meaning:

- **Hyper** means a link between the two pages.
- **Text** means a normal text which user wants to show on the web page.
- **Markup** means symbols and sequence of characters added with normal text and tells the browser how to display the text.
- **Language** means **HTML** is a scripting language, not a programming language.

HTML is a Human-readable language because it can be modified with the help of text, images, and other content in form of the required format. The web page can be created by using **Notepad, Notepad++, Browser**, and **Internet**. The source of the web page can be viewed by right click on the selected HTML page and click on the view source option. It will open a window containing the HTML source code. The basic structure to create a web page is:

```
<html>
    <head>
        <title> page title </title>
    </head>
    <body>
        Includes the text or images of web page.
    </body>
</html>
```

This is the basic structure to create a web page on a notepad or any other tool. It begins with the <html> tag and ends with the </html> tag. And after the beginning tag includes the <head> and </head>, <body> and </body> tags that include the title and actual content of the web page.

HTML includes a list of tags that can be merged in a single document and it can have text, graphics, audio, video, etc. The tags are usually given in simple words and abbreviations such as for paragraph (opening) (closing) is used. All the HTML tags start with an open bracket (<) and end with the closing bracket (>).
There are two types of tags in HTML.

1. **Singular tags:** These are standalone tags that do not have a closing tag. such as
 i.e. break row tag.
2. **Paired tags:** These are the tags that include the opening and closing tags. such as opening and closing tag.

1.3 Introduction of Web design:

Web design is the visual appearance and layout of a website. It is a way to design websites that are displayed on the browser with the help of the internet. A website is designed according to the user requirements and Web designers fulfill the user requirements related to the colors, font, and images, etc. The structure of the web page depends on the information, which user wants to display on their website. The designer designs the website that focuses on simplicity. Thus, the user operates the website without any trouble. Nowadays, there are two common methods for designing websites that work efficiently on mobile and desktop applications such as reactive and adjustable design.

In **Reactive design**, the layout moves dynamically and adjusted according to screen size;
In **Adjustable design**, the website layout is fixed according to the common screen sizes.

In the modern world, if we want to grow our business throughout the world, websites play a major role in advertisement. It can operate on both the computer system as well as mobile. The design of a website should follow these principles such as contrast, balance, emphasis, unity, and the elements such as shapes, images, color, texture, etc.

1.3.1 Key Features while designing the web page:

1. **Navigation:** The design of a website should be straightforward. It should be navigated easily

and the menu should be accessible from any web page. The user or viewer can easily access the page and is aware of the exact location on the website.

2. **Accessibility:** During web designing, web developers make sure that it will be accessible throughout the world. Most of the user who visits our website is not professional, thus they need a straightforward design that should be simple to access the website. Accessibility also includes the scope of information accessible. The website contains information that is easily accessible by all users.

3. **Speedy:** While designing a website, the web developer makes sure that website speed is much better. If the website does not load within five seconds, the user will skip and continue to the other site that will affect the business.

Figure 1: Features of Web Design

4. **Related Information:** The users may visit the website and acquire the related information according to the need. Thus, it is the responsibility of the designer to create a website that includes useful information that should be easily available, concise, and informative. Content writing plays a major role in this feature.

5. **Search Engine Optimization (SEO): A properly designed website can be accessed by many users and the other strategy for attracting users is search engine optimization. It includes the search keywords, profile links, and other factors. This is created to improve the ranking.**

6. **Maintenance: A website can be accessed over 24 hours; thus, it requires proper maintenance to support the users at any time.**

7. **Mobile-Friendly:** In the modern era, if we need to grow our business then the website is the best way to spread all over the world. Therefore, the mobile phone will help to reduce the burden in the form of mobile apps. The website developed by the designer must be mobile-friendly that it can be accessed on any mobile device.

Check Your Progress 1

1. State Whether True or false:
   i. The markup language tells the browser how to manage and display content on the web page.
   ii. There exist two types of tags in HTML one is singular and the other is paired.
   iii. Once the designing of a website can be done it can be edited.
   iv. In an adjustable design, the website layout is not fixed according to the screen.
2. Explain the features of designing the website.
3. Define HTML and the types of tags in HTML.

1.4 Web Browser:

Web Browser is software that can be used to access the information available on the Internet. Whenever a user wants to access information, the browser fetches the data from the server and displays it on the screen. The browser makes surfing easy and multiple pages can be accessed by the browser at once. There exist different types of browsers:

1. WWW (world wide web)
2. Mosaic
3. Netscape Navigator
4. Internet Explorer
5. Firefox
6. Google Chrome

1.5 Introduction to Website:

A website is a collection of web pages. It is connected with the client through a single URL. All the web pages are stored in a central location on the server that is accessed by visiting the homepage of a website using a browser. A website contains information in the form of images, content, and videos. The first website was designed with small text and a few pictures. With the advancement of technology and the expansion of bandwidth by internet providers, websites have become more and more complicated. In the modern world, more than 1.3 billion websites are available over the internet. Most of the websites are available for every age group of people, therefore the websites arise in an endless variety, including educational sites, news sites, forums, social media sites, e-commerce sites, and many more.

1.5.1 Types of Websites:

1. **E-Commerce Website**: E-commerce stands for electronic commerce. An e-commerce website refers to an online shopping place where users can buy and sell products or other services over the internet. Most common websites such as a flip cart, Amazon, and Shopify.

2. **Blogs**: The blog contains discussion or information, that is represented in reverse order and the recent post appears at the top of the web page. A blogger regularly updates the articles, videos, and photos. In the modern world, every business and the major brand has its blogs, even celebrities and politicians write their blogs for updating their views.

3. **Social Media Websites**: It is an online medium where people create a social network with other people sharing similar career or personal interests, backgrounds, real-life connections, activities, etc. Examples of social networks are Facebook, Twitter, Instagram, Snapchat, and Linkedln.

4. **Magazines Websites**: An online magazine website provides articles of current awareness, photos, and videos that are informational and educational also. A few years ago, the magazine industry has changed from a printed platform to a digital platform. Some magazines are related to young children are Men's Health, linked, and The New Yorker.

5. **Portfolio Website**: A portfolio website created by creative professionals such as artists, writers, designers, filmmakers, and builders for showing their best work. A portfolio website is a bit more creative by nature, thus the layout and features are unique.

1.6 Introduction to Webpage:

Web pages are documents that are written in HTML language and are translated by a web browser. Web pages contain the content, information, images, and videos related to that particular page. The web pages are created in two ways, static and dynamic web pages. These are the pages written in scripting languages such as PHP, ASP, etc. A website is a collection of web pages and a web page is an individual document written in HTML.

1.6.1 Types of Webpage

A web page is a document available over the internet. It includes graphics, images, contents, links, audio, video, etc. It can be created in two ways:

1. Static Page
2. Dynamic Page

1. **Static Page**: Static pages are those pages that contain static content (text or images). Thus, the viewer can view the web page content without being able to change it. The content of the webpage is fixed and never be changed by the viewer. It is also known as a stationary or flat web page. These pages are loaded on the client's browser in the same way as stored in the server. Users can just read the information available on the static web pages. These are created with the help of HTML.

2. **Dynamic Page**: Dynamic pages are those where the content of the webpage depends on the viewer's input. Different information is provided to the user whenever the user accesses the web page. These are the temporary pages that are available according to the user's request. In dynamic pages, a portion of the content is restricted, it should be visible only when the login credentials are given by the user. Dynamic pages can decide what should be displayed on the browser with the help of scripts. Scripts have their way to issue commands to the browser.

Advantages of Static pages

1. Static pages are quick and easy to implement and development of static pages are so easy.
2. The cost of static pages are cheap with less amount of hosting.

Disadvantages of Static pages

1. In Static pages are updation of the content takes more time, so it is time consuming process.

Advantages of Dynamic pages

1. Updation of dynamic pages are easily implemented.
2. Dynamic pages shows more functionality than static pages.
3. Dynamic content help in search engines.

Disadvantages of Dynamic pages

1. Dynamic pages are more expensive at hosting time.
2. Dynamic pages are so slow than static pages because of database and images.

Difference between static pages and a new pages.

Sr no	Static Pages	Dynamic Pages
1	Content is predefined in static pages.	Content is generated quickly and changes regularly.
2	HTML is only one used for designing the websites.	PHP, Asp.net ,JSP and number of languages to develop dynamic websites.
3	It sends same response for every request.	It may generate different pages after each request.
4	Content is only changed when updation is needed.	Contact is changed during runtime.
5	Static pages are more flexible.	Dynamic pages are less flexible.

Table 1: Comparison of Static and Dynamic pages

1.7 What is the Process of Web Designing?

The process to design a website is different from communication processes. Web designers usually think about the design of a webpage with the center of attention on technical difficulties such as code, the blueprint of a web page, and code management. However, good design is based on how much users access the website. Well-designed websites attract visitors and assist people to understand the products, companies, and services, which are given by the website. The process to design a website requires the following steps:

1. **Goal identification**: Initially, identification of the goals lets the web designer focus on what will provide the website and what will be the impact of the website on the viewers. It should be done in close collaboration with the client. This phase includes the following questions that need to explore in designing a website.

- Who will use the site?
- What will they suppose to find?
- Is this website a static site or a dynamic site?
- How this site is different from its competitor website?

These are important considerations while designing a website. Initially, if all these questions are not clearly stated, it may lead to the wrong direction for the project.

Tools used for goal setting of a website at the identification stage:

- Audience
- Brief introduction
- Analyses of a Competitor site

- Brand features

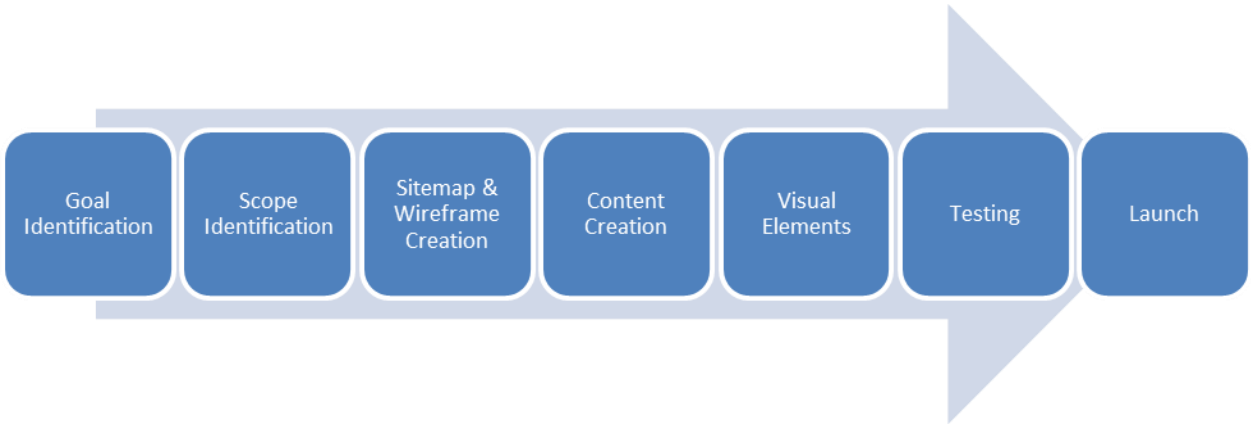


Figure 2: Web Design Process

2. **Scope identification**: Once the goal of the website is identified, the next step is to define the scope of the project. The scope includes the work that needs to be performed to achieve the objective. In other words, scope means the process to identify, achieving the project goal, cost of the project, and timeline to achieve the objective. The client sets one goal in mind, but it may change during the design process. This is not an issue for the designers but may lead to more effort for the project. If the rise in expectations is not according to the increased timeline or budget, the project will become completely unrealistic.

   Tools used for scope identification are a Gantt chart and a contract. The Gantt chart includes the details of the timeline of the project and the designer follows this Gantt chart for web design.

3. **Sitemap and wireframe creation:** Once the scope of the website is determined, the designer can set up the roadmap under the client's guidance and a detailed view of the content. Thus, the designer can design the website according to user requirements. The sitemap provides the basis for designing an effective and well-designed website. The designer gets a clear view of website architecture and shows the relationship between different pages and content. After the sitemap, a wireframe is built that provides a framework for storing the design of a site that helps in the identifications of gaps and challenges with a sitemap. A wireframe guides a designer that shows the final look of the site.

   Tools used for sitemap and wireframe

- Pen/pencil and paper
- Sketch
- Writemaps
- Mindnode, etc.

4. **Content Creation**: Once the framework is created, the next step is to write the content for the site. The content of the site has two purposes. First, the content of the website

engages the readers and encourages them to take appropriate action for fulfilling the goals. If the content of a website is not appropriate like dull or not formatted, the readers may not visit the website for long. Short and useful contents grab the attention of readers. Secondly, the content also promotes site visibility by search engines. This can be done by Search Engine Optimization.

Tools used for Content creation and SEO

- Google Docs
- Dropbox Paper
- Content Management System (CMS)
- Google Keyword Planner
- Google Trends

5. **Visual Elements**: The next step is to create a visual presentation of the content on the site. This can be possible by the design elements such as logos, color, images, etc. Images are playing a significant role by giving a professional look and feel to the website. It delivers a message to the readers without even reading the content and is mobile-friendly.

Tools used for Visual Elements

- Photoshop
- Illustrator
- Coolare
- PicMonkey
- Venngage
- Google Drawings

6. **Testing:** After the visual content and the site is ready, the designer first tests the working of all the web pages. Web designers follow manual and automatic testing, thus after launching the website the client does not face any problems. Testing thoroughly every page and all the links to make sure the website is working properly. Some errors in the website may result in any fault to the working of the site, which must be fixed earlier before presenting the site publicly.

Tools used for Testing are

- SEO Spider
- W3C line checker

7. **Launch:** Once every page of the website has been tested, and the client and designer are satisfied with the working of a website, the next step is to launch the website. Web designing is an ongoing process that needs regular maintenance.

8. **Site maintenance**: Websites need constant attention and maintenance, which requires updating the information, change the requirements, and fixing the errors. These are the major considerations in the maintenance of the website.

All these steps for designing the website are equally important, and the strategy should be followed for its implementation. This is a sequential process that works step by step.

Check Your Progress 2

1. State whether True or False:
    i. The designer is not concerned about the look and feel of a website but is also about the interface.
    ii. Is a Static page a temporary page?
    iii. An E-commerce website can display information about the current awareness.
2. Define a Website and the types of a website.
3. Explain the process of web designing.
4. Define Web pages and their types.
5. Difference between website and webpage.

1.8 How to Publish the Website?

Web publishing is a process to publish your content on the internet by creating, updating, and uploading. The content on a website includes images, texts, videos, etc. Once the testing phase overs, the designer needs to set it up online. Therefore people can access the website. Publishing a website is a multiform method and it requires three things: Software for developing a website, Internet Connection, and the webserver. It works through the following steps:

1. Firstly buy the web hosting and the domain name from hosting companies.
2. The web hosting companies provided the rented space and put our website files on their web server.
3. Select the domain name that is a unique address where people search our website such as https://www.facebook.com.

Check Your Progress 3

1. HTML stands for

    a. High Text Machine Language
    b. HyperText and links Markup Language
    c. HyperText Markup Language
    d. None of these

2. The correct order of HTML tags are

    a. Head, Title, HTML, body
    b. HTML, Body, Title, Head
    c. HTML, Head, Title, Body
    d. HTML, Head, Title, Body

3. HTML uses

a. User defined tags
b. Pre-specified tags
c. Fixed tags defined by the language
d. Tags only for linking

4. Which tag is used for the declaration for an HTML document?

a. <html>
b. <! DOCTYPE Html>
c. <body>
d. <p>

5. HTML fundamental blocks also known as_____.

a. HTML Body
b. HTML Tag
c. HTML Attribute
d. HTML Element

6. The first tag of HTML document is

a. <head>
b. <title>
c. <html>
d. <document>

7. Web pages are interpreted by
a. Compiler
b. Server
c. Web Browser
d. Interpreter

8. Which is not a browser?

a. Microsoft's Bing
b. Netscape Navigator
c. Mozilla Firefox
d. Opera

9. The brackets used for HTML tags are
a. Curly
b. Round
c. Square
d. Angle

10. Content which is not displayed on the web page are covered in_____tag.
a. <head>
b. <title>
c. <body>
d. <html>

11. Which of these is also known as nexus browser?
    a. Opera
    b. Chrome
    c. Firefox
    d. WWW

12. Which is the world's popular browser?
    a. Opera
    b. Chrome
    c. Firefox
    d. Mosaic

13. Which browser first introduces GUI?
    a. Opera
    b. Netscape
    c. Firefox
    d. Mosaic

14. How can a website be published?
15. What are the different versions of HTML?
16. What do you mean be site map and wireframe creation?

1.9 SUMMARY

This unit describes the basic concepts of the design and development of a website. It also defines the method to create HTML pages and the role of HTML pages. The HTML page is divided into two parts. One is the head section and the other is the body section. The content of the web page is given in the body part of the webpage. The head section includes the title of the web page. It also covers the introduction to web design and the features for designing efficient web design. The unit also describes the website and types of the websites such as E-commerce sites, blogs, social media sites, etc. A website consists of many web pages. There exist different types of web pages such as static and dynamic pages. The unit covers the different phases of web designing and the way to publish a website. The phases to develop the website are also explained in this unit. This unit includes the basic information of browser and its types. And the steps to be included for publishing a website.

UNIT 2 OVERVIEW OF HTML

Structure **Page No**

2.0 INTRODUCTION

HTML is a Hypertext Markup Language which is generalized from SGML i.e., Standard Generalized Markup Language. SGML is difficult to understand and implement for beginners. Thus, HTML plays an important role in the development of web pages simply and easily that can be accessed on the browser. HTML includes different types of tags. The two most important categories of tags are Singular and Paired tags. Singular can be represented only with an opening tag and Paired includes opening and closing tags. The tag represents the browser the action that should be taken on the given data between them. Formatting tags can be used to give headings or paragraphs a proper format that should be visible on the browser.

2.1 OBJECTIVES

The learners will be able to understand the concepts of HTML and they will be able to design and format the webpages using different formatting tags given in this unit. This unit includes the details of the following topics:

- Implementation of web
- Phases of website development
- Role of HTML in the web
- Paired and Singular Tags
- Formatting Tags

2.2 Implementation of Web

Implementation of web includes the following steps that should be followed in actual implementation. These are:

1. **Product Design**: The web designer first creates a blueprint of the website and the packages, pages, and connection diagram. These are the primary guidelines for a web designer to start working on the web. The design plays a major role in the implementation.

2. **Specifications of the web**: The second step is the specification of the web that sets the parameters, limitations, or other specifications for designing. It helps in designing the web and making decisions for the designer.

3. **Specifications for update and maintenance:** Once the analysis has been done on the web, the designer may get the possible corrections that need to be implemented in HTML or any information updates that need to be performed to get the dynamic website.

4. **Specification for technology**: This specification includes the objective of the project and the strategy that should be followed in designing a website. It covers the constraints such as deadline, budget, or technical restrictions and the detail of the team involved in the project.

5. **Inputs from users**: The designers are in touch with the users through the mail for feedback. The user sends the feedback for the overall working of the web. This feedback should be considered and followed for implementation issues. The goal of a designer is to create the best working website. The client may send the queries and updated information to the designer to update the website.

6. **Integrating**: The design, strategy listed in the above phases must be combined for implementing a web.

7. **Designing**: It includes the designing of an actual web by following the blueprint that meets the need of a client.

8. **Creating**: By using templates, web components, and software, a designer can create or implement the web easily and quickly.

9. **Writing**: The files of the web are created either by hand or through the editors or with the help of an integrated development environment.

10. **Maintaining**: This phase of implementation checks the files that they are technically correct, up-to-date, and usable that meets the client's need.

2.3 What are the phases of Website development?

Web Development includes the different processes for developing a website. It consists of several phases that are information gathering, planning, designing, development, testing, maintenance. It requires a team that is responsible for the different tasks of website development.

A website represents a company and people will search the website during the research. It should be unique. A web designer uses HTML, CSS to create a website. Website development is the process that carries multiple steps during development and various phases are given below:

1. **Information gathering**: This is the first and the most important phase of website development. This is the primary phase of website development that is information gathering. This phase analysis the requirements and needs of clients. It is also known as the discovery phase. At this stage, the designer gets a perception to create a website and

identify the goal of the website and the target audience. These are the crucial factors at this phase of development.

2. **Planning**: "A good **website is that the results of smart** planning" is what **we tend to believe**. Once the information is gathered, the next step is to do the planning. It involves prioritizing the task for the development phase. Here, the sitemap is created, and then the menus and content are decided for the site. The basic purpose of a sitemap is to create a user-friendly site and it also decides the functions that we want to add to the site.

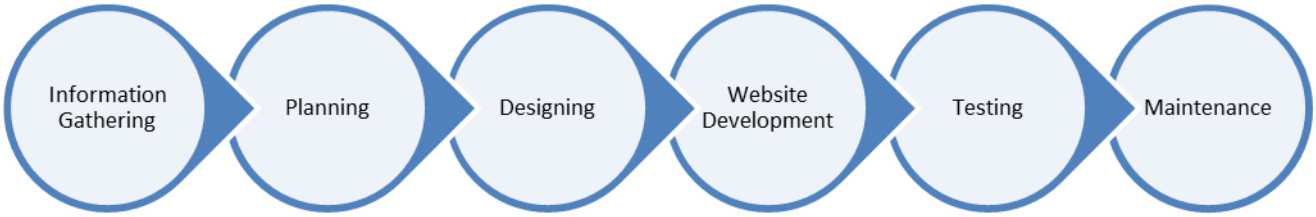


Figure 1: Website Development Phases

3. Designing: The next phase in the development is designing which is the artistic phase. The best design is possible when there is no communication gap between client and designer i.e., communication plays a significant role in this phase. Web designers understand every part of the client's assumptions and try to draw on the web page, from overview to final layout. This phase also finalizes the logos and the templates used for designing the website. Other things that should be kept in mind while designing a website are color contrast, text area, images, font, size, video, etc. This phase discovers the shape of a website.

4.Website development: This phase implements the code for the website. This phase is also known as the implementation phase. The designer makes the working website by writing code for smooth functioning. At this phase, the actual implementation of the website starts. Similar to the sitemaps, initially the home pages are implemented then the other pages start implementing. This phase merges the information gathered in the earlier phases. Here, the database is created, and programming is done.

5. Testing and Delivery: After the implementation phase, the next phase is testing and delivery. Testing will fix the errors through manual and automated testing. The software tester is responsible for preparing the test cases and these test cases are solved by the web developer with the help of test cases and again this process will continue until all the errors have been fixed. Following are the types of Testings:

- Design Testing
- Functional Testing
- Content Testing
- Speed

- Browser Compatibility

After these tests are necessary before the official launch of the website. Once the testing is done, the website is delivered.

6. Maintenance: The last phase of website development is maintenance. This stage performs the maintenance of the website for a limited period. It involves updations and modification of contents or design of the website. Extra charges need to be paid for the extension of the limited period of maintenance. Most of the companies provide source code, documentation, and other support to the customers. This is also a very important phase of website development because once the website is live, it may need maintenance.

2.4 Role of HTML

HTML stands for Hyper-Text-Markup Language. The role of HTML is to design the layout and it also defines the content of the website. The document includes tags, that specify the structural layer of a web page. Web Designer has maintained the relationship between the entire sites with the help of HTML and represent the connection of documents with one another. HTML is very useful to represent the content according to the user's need where tags and attributes assist the designer to balance the creativity and artistic sides. HTML plays a vital role in designing the website or without HTML the future of designing the websites will not be possible. The layouts, images, display, and font of a website make it attractive for the viewers. This is a unique language interpreted by the browser like chrome, Mozilla, etc.

2.5 Overview of HTML

HTML was developed by Tim Berners-Lee in 1991 and the first version of HTML was 1.0, after that number of versions were introduced with advanced features for modern browsers.

HTML versions	Features	Year
HTML 1.0	Text and Image control only	1991
HTML 2.0	Improved markup tags, Forms and Tables	1995
HTML 3.2	Better Support for new Form elements, CSS and Frames	1997
HTML 4.0.1	External CSS with extended support	1999
HTML 5	Audio tags, Semantic tags, Email, and Password from the user.	2014

Table 1: Versions of HTML

At present, HTML has great improvement and web designers have maintained the standard of W3C (World Wide Web Consortium). W3C is an international authority that develops open standards to ensure the long-term growth of the web.

2.6 Basic Concepts of HTML

1. **HTML Editor:** HTML editor is used for writing the text, format the text and it can be used to translate to other languages, for instance, CSS, XML, or JavaScript. Some editors might be easier to use while others perform more functionality with better performance and some of them are given below:

1.1.Notepad++: It is free available on the internet and that was developed for Windows as well as Linux operating system. Notepad++ is simple to use for beginners as well as professionals both.

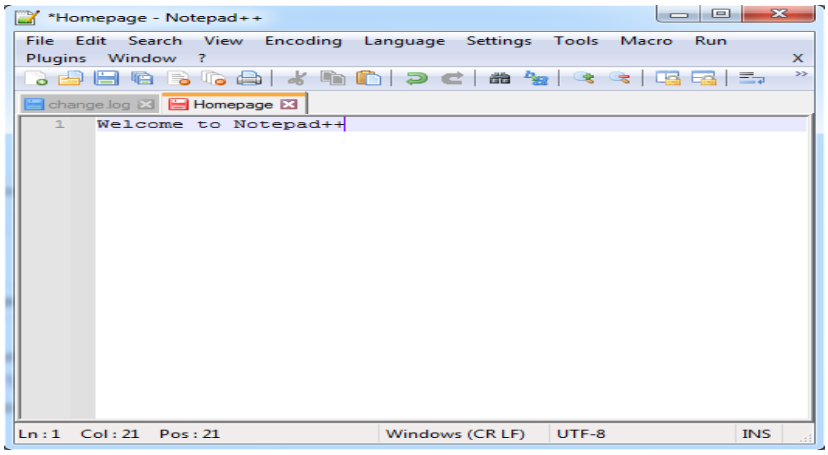


Figure 2: Notepad++

Key Features:

- Notepad++ platform is easy to use, lightweight, and fast.
- It handles multi-language on a single platform.
- It is easily compatible with Windows and other operating system.
- It is free available on the internet.

1.2. Adobe Dreamweaver: It is one of the most popular editors that handle both textual and WYSIWYG methods. It is a most powerful and flexible tool. Dreamweaver is used for both front-end programming as well as back-end programming languages.

Key Features:

- Dreamweaver allows the web designer to write code in any major programming languages such as C, C++, XML, PHP, and many more.
- It supports textual and WYSIWYG editor modes.
- It is easy to format the text and quick access on the browser.
- The performance of Adobe Dreamweaver is magnificent and support by

Adobe Inc.

Figure 3: Adobe Dreamweaver

2. **Browser**: A web browser is application software for accessing the information from World Wide Web. When the user searches the data from a specific website, the browser help to get the related content from a web server and then shows it on the browser window. It just connects the server and the user with the help of the internet. Today's Google chrome is the fastest and most popular browser and other browsers such as Firefox, Internet Explorer and Safari contribute their services throughout the world.

Figure 4: Browsers

3. **HTML Structure:** Webpage can contain number of tags with attributes; however, every HTML webpage contain five basic tags such as <!doctype>, <html>, < head >, <title>, < body >. Each tag has open tag < > and closing tag</ >except <!doctype>.To illustrate with,<html> has its closing tag</html> and <title> tag has its closing tag </title> etc.

The basic structure of HTML tags:

```
<HTML>
< HEAD >
        <TITLE> First document</TITLE>
< /HEAD >
< BODY >
    <H1> WEB PAGE INTRODUCTION</H1>
</BODY>
</HTML>
```

Key Features:

- The closing tag is compulsory with opening a tag.
- Forward Slash (/) is used while closing the tag.

2.7 Structure of HTML Documents

The basic structure of HTML is given as:

```
<!DOCTYPE html>
<HTML>
< HEAD >
        <TITLE> </TITLE>
< /HEAD >
< BODY >
…….
</BODY>
</HTML>
```

These tags give instructions to the browser and tags are given in the opening angular brackets such as (<) and (>).

1. **<!DOCTYPE html> :** All the documents must start with this tag. This is not a declaration tag, rather it provides information to the browser about the type of document. This tag is optional in an HTML document.

2. **<HTML> Tag**: This tag specifies the beginning of HTML document with <HTML> tag and ending the document with the </HTML>. It also includes the < head > and < /head > tags and <BODY > and </BODY> tags.

3. **< HEAD > Tag**: This tag identifies the document header and the < HEAD > tag contains following tags such as <TITLE>…</TITLE>, <SCRIPT>…</SCRIPT>, <LINK>…</LINK>, <STYLE>…</STYLE> and also identifies the metadata. This information is not displayed in the browser window. This is the optional tag in the HTML document.

4. **<TITLE> Tag:** The <TITLE> tag is used under the< head > <title> **Text** </title> < /head > to represent the title of webpage document. The title should be displayed on the title bar of browser's window.

5. **< BODY > Tag:** This tag represents the document's body which contains other HTML tags according to client requirements. The < BODY > tag contains the actual part that is visible on the browser and it is closed with the </BODY> tag.

Steps to Creating HTML Simple Document: To Create the HTML page then follow the following steps:

1. Use any text editor like Notepad++.
2. Write the <html> tag on the top of the page.
3. After that adding the < head > tag on the next line.
4. < head > tag contain the <title></title> tag and closing header tag:< /head >.
5. After that type, the < body >tag and write the text, which we want to display and closing the </body>tag.
6. Finally, type the closing tag of </html>.
7. In the file menu of Notepad++, choose Save As.
8. Type the name of the file with HTML extension .html i.e., **filename.html**.
9. Click Save.

View of HTML Page: To display our page then follow the following steps:

1. First Open the Web browser and click File--→Open File from the top menu.
2. Navigate the location of the file.
3. Click on the web page file and then click open.
4. The web page is open on the browser window.

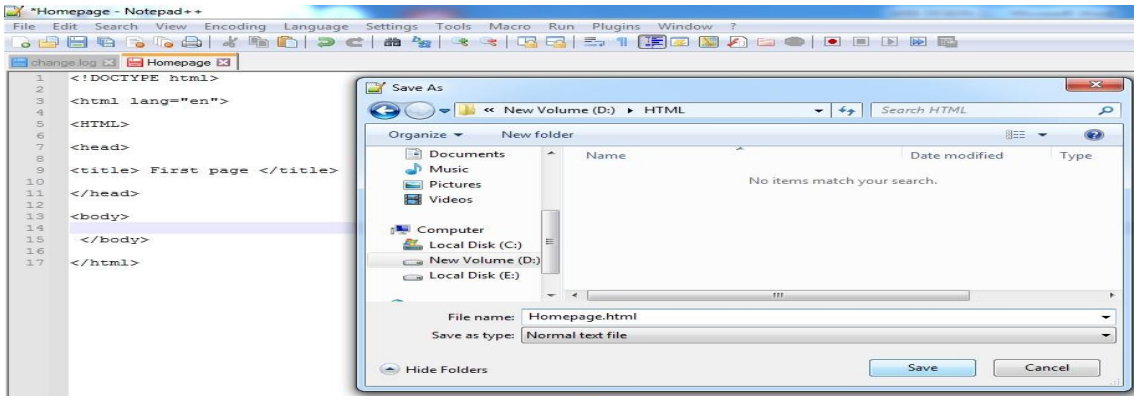


Figure 5: Web page code with output

2.8 Various Elements of HTML

Elements can be defined as commands which are being directly attached to the text in an HTML document. It represents the instructions to the web browser to do some tasks and display information on the browser. In an HTML document, various tag names reflect different outputs on the web browser. In HTML all tags are different from each other and perform a different task. The tag names are enclosed in between angular brackets for a 'less than' or 'greater than sign (<) and (>) and the ending tag is represented with a forward slash (/) before the tag name.

Types of elements in HTML:

Web page is created by using various type of tags. HTML tags are also known as elements. Thus, it can be divided and defined on a different basis. Let us discuss the tags based on the following categories.

2.8.1. **Paired tags**: In HTML most of the tags are defined in a paired format such as HTML, head, title, body, and so on, all are paired tags. An HTML pair tag represented with opening <> and closing tag </ >both.

Tag	Description	Syntax	Example
Heading tags	The heading tags used to represent the main heading/subject related to whole page and these tags levels are starting from <h1> to <h6> where <h1> defines the most important heading on the other side <h6> defines the least important.	<h1> Text </h1> <h2> Text </h2> <h3> Text <h3> <h4> Text </h4> <h5> Text </h5> <h6> Text </h6>	<h1> Heading 1</h1> <h2> Heading 2</h2> <h3> Heading 3 <h3> <h4> Heading 4</h4> <h5> Heading 5</h5> <h6> Heading 6</h6>
Anchor Tag	The <a> tag is used to create a link between two web pages and it can be possible through a most important attribute is href.	<a href = "URL ">	 Facebook
Pre-initialization	The <pre> display the text exactly as written in the HTML source code.	<pre> Text </pre>	<pre> The <pre> display the text exactly as written in the HTML source code. </pre>
Paragraph	The < p > tag defines the paragraph. If designers break the line then they will use tag.	< p > Content < /p >	< p > This tag defines the paragraph. < /p >
Address	The <address> tag defines the residence information	<address> Address </address>	<address> Sandeep Kumar B-1/2770, Gobind Nagar,

	for the owner / owner of a related document.		 Sector-22, Punjab. </address>
Blockquote	The <blockquote> tag specifies a limited section that is quoted from another source.	<blockquote> Content </blockquote>	<blockquote cite = "https://blog.kerridgecs.com/10-inspirational techquotes"> “Technology is best when it brings people together.” </blockquote>
Marquee	The <marquee> tag is used for scrolling the text either horizontally or vertically. Its direction can be up, left, right, down.	<marquee> Text </marquee>	<marquee direction = "right"> Text </marquee>

Table 2: Paired tags example

2.8.2. **Singular tags**: In HTML some of the tags do not need a closing tag or companion tag. Singular tags are also known as an empty tags.The <hr> tag defines a thematic break in an HTML page (e.g., a shift of topic). The <hr> element is most often displayed as a horizontal rule that is used to separate content (or define a change) in an HTML page.

Tag	Description	Syntax	Example
Break	The tag is used for break the single line and there are no need to insert a closing tag.	 	< p > Technology is best when it brings people together. < /p >
Image	The <img> tag is used for showing visual effects that make web pages are more attractive.	<img src = "url" alt = "alternatetext”>	<img src = "rose.jpg" alt = "Flower">
Horizontal Line	The <hr> tag is mostly used for displaying the horizontal rule.	<hr>	HTML < p > HTML is a language for describing web pages. < /p >

Table 3: Singular Tags Examples

i. **Self-closing tags:** In HTML, these HTML tags do not have a companion tag (partner tag), where the opening tag is only compulsory in an HTML document. It contained important information in the form of attributes. An image tag is the best example of self-closing tags.

Tag	Description	Syntax	Example
Image	The <img> tag is used for showing visual effects that make web pages are more attractive.	<img src = "url" alt = "alternate text">	<img src = "rose.jpg" alt = "Flower">

Horizontal Line	The <hr> tag is mostly used for displaying the horizontal rule.		HTML Definition < p > HTML is a language for describing web pages.< /p >

Table 4: Self- closing Tags Examples

ii. **Formatting tags:** In HTML formatting tags are used to change the text format according to user requirements. Due to these tags, the text looks better, and feel such tags are bold italicized or underlined.In HTML the formatting tags are divided into categories:

Tag	Description	Syntax	Example
Bold and Strong	The tag is used for displaying the text in bold letter and The tag used to display the text with strong importance.	Content Content	Technology is best when it brings people together. Technology is best when it brings people together.
Italic and Emphasized	The tag is used for displaying the text in italic format and The tag is used to display the text in emphasizes way. The HTML element defines emphasized text. The content inside is typically displayed in italic.	Content	Technology is best when it brings people together.
Underline	The <u> tag is used for displaying the text in normal way but it can include underline below the text.	<u> Content </u>	<u> Technology is best when it brings people together. </u>
Align	The alignment specifies how the text appear in the document	<p align = "left | center | Right | justify "> Content< /p >	<p align = "center "> This paragraph is aligned center. < /p >

Table 5: Formatting tags

Check Your Progress 1

1. State Weather True or false:
   i. HTML tags are given in singular tags only.
   ii. HTML heading can have 6 different levels.
   iii. An HTML document can be divided into three parts: < head >, < body >, <STYLE>.
   iv. < p > tag is used for the paragraph that skips one line.
2. Write the HTML code that makes use of the following tags:

- Italics
- Break
- Paragraph
- Headings

3. Explain the phases of web development.
4. What is the role of HTML?
5. Explain in detail the list of formatting tags in HTML.

Hands on Practice

Example 1:

```
<!DOCTYPE html>
<html>

  < head >
    <title> Smaller Text Example </title>
  < /head >

  < body >
    < p > The following word uses a < small > small < /small > typeface. < /p >
   < p > The following word uses a < big > big < /big > typeface. < /p >
   < p > The following word uses a <sup> superscript </sup>  typeface.< /p >
  </body>

   </html>
```

Example 2:

```
<!DOCTYPE html>
<html lang = "en">
< head >
   <title> Formatting Text in HTML </title>
< /head >
< body >
   < p >This is <b> bold text </b>. < /p >
   < p >This is <strong> strongly important text </strong>.< /p >
   < p >This is <i> italic text </i>.< /p >
   < p >This is <em> emphasized text </em>.< /p >
   < p >This is <mark> highlighted text </mark> . < /p >
   < p >This is <code> computer code </code> . < /p >
   < p >This is <small> smaller text </small>. < /p >
   < p >This is <sub> subscript </sub> and <sup> superscript </sup> text. < /p >
   < p >This is <del> deleted text </del>. < /p >
   < p >This is <ins> inserted text </ins>. < /p > < p > This is <b> bold text </b>. < /p >
</body>
</html>
```

Example 3:

```
 <!DOCTYPE html>
<html lang = "en">
< head >
   <title>HTML strong Vs b Tag</title>
< /head >
< body >
   < p > <strong> WARNING! </strong> Please proceed with caution. < /p >
   < p > The concert will be held at <b> Hyde Park </b> in London. < /p >
</body>
</html>
```

Example 4:

```
<html>
< head >
<title> Example of marquee tag in HTML </title>
< /head >
< body >
  < marquee behavior = "scroll" direction = "left" scrollamount = "3"> Slow speed scroll speed
</ marquee >
  < marquee behavior = "scroll" direction = "left" scrollamount = "10"> Medium speed scroll
speed < /marquee >
  < marquee behavior = "scroll" direction = "left" scrollamount = "17"> Fast speed scroll speed
< /marquee >
</body>
</html>
```

2.9 Summary

This unit helps beginners to understand the steps to create simple web pages. The implementation of the website also follows the sequence of steps that need to be performed to get a better website. To create a web page, a designer may require a tool to write the code. It can be a simple text editor or any other Integrated software. After designing a simple web page, the next step is to run the code with the help of a browser. The browser does not display the tags but the content, images given in the Html file. The < head > tag includes the title of the document and the < body > tag contains the content of the web page. The important is to display the text in proper format by using formatting tags. A large number of formatting tags exist in the HTML. Headings are represented using <H1> to <H6> tags for sections and subsections. < p > tag is used to create a paragraph and <B>, <I>, <U> tags are used for bold, italic, and underline the particular text.

UNIT 3 TAGS OF HTML

Structure

3.0 INTRODUCTION

Along with basic tags, there exist some tags that makes an interactive webpage. In which hyperlinks, images plays an important role. Hyperlinks are also playing an important role in designing a large number of web pages that help in linking one document with another. Links can be of two types. One is internal link and the other is external link. A web page can be made more attractive by placing the image on it. As we know "A picture is worth a thousand words", thus the designer is not required to make content, rather it creates an interactive website by placing images. There exist different types of images that can be applied on a web page to make the web site attractive. Semantic links and meta tags can also be used on the website that creates an efficient web page.

3.1 OBJECTIVES

The learners will be able to understand the concepts of HTML and they will be able to design the webpage using images and hyperlinks given in this unit. This unit includes the details of the following topics:

- Hyperlinks
- Working of Images
- Image Map
- Meta Tag

This helps in understanding the link of one document with another document. In this way a large website can be created that link one web page with the another in sequence.

3.2 Hyperlinks

The website consists of many web pages that are linked with one another through Hyperlinks. HTML provides linking of one document with another document as well as with images. It has two ends one contains the source address and another contains the destination. Links are used to connect the various resources such as images, videos, documents, and a program also.

Syntax:
Links and specified in HTML by using the <a> anchor tag.

 Text

Where the href contains the specific destination address and Text represents the visible part of the link. Links are of two types:

3.2.1 Internal links: An internal link is the type of a link whose target is to open an image or document on the same website. By using these links, the color of text changes in different formats such as:

1. The default color of an unvisited link is blue and underlined.
2. The default color of the visited link is purple and is underlined.
3. The default color of an active link is red and underlined.
4. When the mouse pointer moves over the hyperlink, the shape of the pointer changes to a hand shape.

Syntax:
Links are specified in HTML by using the <a> anchor tag.
For Location :<a name = " location name ">
For Link: Text

3.2.2. External links: External link is a type of link that connects one website with another website when a user navigates to other pages by clicking on a link of another website and reached

that another location which is given in the <a> tag.

Syntax:
Links and specified in HTML by using the <a> anchor tag.

 Text

Sr No.	Internal Links	External Links
1	These are fast and easy to control.	These are difficult to control.
2	It links the pages with the same website.	It links the pages with another website.
3	These are the links to the web pages on the same page or different page locations.	These are the links to the web pages on a different page or different website.

Table 1: Difference between Internal and External Links

The default colors of the links can be changed by using the following attributes in the <BODY> tag. These are given as:

Sr No.	Attributes	Description
1	LINK	It changes the color of the default hyperlink with the color specified in this attribute.
2	ALINK	It changes the color of the activated hyperlink with the color specified in this attribute.
3	VLINK	It changes the color of the already visited hyperlink with the color specified in this attribute.

Table 2: Color specified with a hyperlink

URL (Uniform Resource Locator) is defined as the address of the web page. It can be specified in two forms:

1. **Absolute URL:** It refers to the complete address of the document on the world wide web. It includes the protocol to be used, the name of the server, complete pathname, then files name with extension.
   For Example:
   URL = " http://www.university.ac.in/syllabus/bca.html "

   The first part specifies the HTTP i.e., the Hypertext transfer protocol that defines the set of rules. The next part specifies the server's name or domain name that specifies the location where the files are stored. And then followed by the directory which contains the actual HTML documents.

2. **Relative URL:** It refers to the address of the document only. It does not contain the complete address compared to the absolute address. Protocol name and the server or domain name are not included in the relative URL.
   It can be specified according to the location of the web page.
   - If both the pages are placed in the same folder.

Example:

 Syllabus

- If the second page is in subfolder.

Example:

 Syllabus

- If the webpage is in higher folder than the source file.

Example:

 Syllabus

Sr No.	Absolute URL	Relative URL
1	It is used for linking of different web pages.	It is used for linking the web pages within the website.
2	Exact location is to be given in this URL.	It is easy to maintain, even if the server or protocol changes.
3	Slower to access these links.	Faster to access these links.

Table 3: Comparison of absolute and relative URL

Attributes of <a> Tag:

Attributes	Description	Syntax	Example
Href	This is used to give the name of the web page to create a link. If the user wants to link to the particular location on the web page. This is given by the #(hash) sign.	 Text 	<a href = "first.html">
Name	This is used to give name to the location for hyperlink.	<a name = "location name">	 Text
Title	This is used to specify the description of hyperlink. Whenever user places cursor on the link, it displays the title on screen.	 Text 	 Text
Target	The default page is opened in the current browser but if the designer needs to open it on another page, then the target is specified in the tag. It can have the following values. **_blank** will open the linked page on new window. **_self** by default it opens the window in same document. **_parent** opens the document in the parent frame. **_top** will open the document in full	<a href = "locàtion_name" target = "_self | _ blank | _parent | _top">	 First

	body.		

Table 4: Attributes of <a> anchor tag

E-mail Links: These are used to send email messages easily and quickly. Whenever the user clicks on the link, it will open up the outlook express or other mail options that enable the user to write e-mail messages to the address given in the links.

For example:

 E-mail Us

Check Your Progress 1

1. State Wheather True or false:
   i. By default, the hyperlink color is red.
   ii. The target attribute of anchor always links to the beginning of the document.
   iii. The color of the visited link or activated link can be changed in the <body> tag.
   iv. The name of the location in target always begins with the # symbol.
2. Create an HTML code that contains different links of section 1, section 2, and section 3. By clicking on these links, it will jump to the exact section on the same web page. This web page contains long information about the sections.
3. Explain in detail the attributes of anchor tag in HTML.

3.3 Images

Images are the visual elements that make web pages more attractive and give distinctive look to the page, thus web designers must understand how to use the images properly. The images can be added to the web page by using the <IMG> tag and use its attribute for its proper visibility.

<IMG> tag: This tag is used for inserting images on the web page for showing some visual effects in the form of graphics, images, and pictures. HTML image tag is a singular tag that contain attributes only in <img /> tag.

Syntax:

<IMG SRC = "URL" ALT = "alternate text">

Example:

```
<!DOCTYPE html>
<html>
<body>
   <img src = "image_name.extension" alt = ”alternate_text” />
</body>
</html>
```

3.3.1. Attributes of HTML image tag:

Image tag can contain attributes to represent the layout more beautifully, thus the following attributes are used in <img> tag.

ATTRIBUTE	Description	Syntax	Example
SRC	It is the necessary attribute that describes the source or location of the image. In the image, the src attribute is essential to specify the source that tells the web browser where the image is located, so that image can be displayed correctly, if the path of the image doesn't correct then it will not be displayed on the web page.		<body> It means the image is present in the same folder as the HTML document. It means the image is present in the one folder down from the HTML document. It means that the image is in one folder up from HTML document. It means that the image is two folders up from the HTML document. </body>
ALT	It is alternative text for the image. If, for some reason browser doesn't display the image then the value of the alt attribute shows in the place of an image and it can be used for SEO (search engine optimization) for intended purpose.		<body> <img src = "rose.jpg" alt = "Flower"> </body>
WIDTH AND HEIGHT	These attributes are setting the dimension of the image in percentage or pixel. The width and height attribute indicate the exact size of the image. Both are an optional attribute and in HTML5, CSS will be used in place of height and width attribute.		<body> </body>
ALIGN	It can be used to align the image horizontally to the right or the left side. By default, image is aligned to the left side on the web page.		<body> </body>
BORDER	It can be used for showing the border of the image in pixel.	<img src = "url" border = "number in pixels"	<body>

		/>	</body>
HSPACE	It is used to set the space to the left and right of the image.		
VSPACE	It is used to set the space to the top and bottom of the image.		

Table 5: Attributes of <IMG> tag

3.3.2. Common Image Formats

The most common image formats are

Image File Format	Description	Abbreviation	File Extension
Gif	.gif images are used for buttons, clip art, and banners. The main advantage of .gif files can have a transparent background but it has limited colors only 256.	Graphics Interchange Format	.gif
Jpeg	JPEG has unlimited colors and a high compression rate; thus, it takes less memory on the hard disk. Although .jpeg files don't allow transparent background.	Joint Photographic Expert Group image	.jpg, .jpeg, .jfif,. pjpeg, .pjp
Png	png have large number of color combinations, low file size and don't allow transparent background same as .jpg. With high compression rate it can be best use in graphics.	Portable Network Graphics	.png

Table 6: Types of image formats

i. **Image as a link:** To use the image as a link in HTML, use the anchor tag<a> with the href attribute. In which <img> tag is placed inside an anchor tag. Sometimes, the larger images may take a long time to display on the browser. In that case, the links can be used to display the full image on another page that reduces the time to load the image on the browser.

Syntax:
<html>

```
<head></head>
<body>
<a href = "*.html"> <img src = "*.jpg"> </a>
</body>
</html>
```

3.3.4. Image as Button: Image buttons have the same effect as a submit button, the only difference is instead of a normal submit button we can place an image of our choice. An Image button can be created as:

Syntax: <input type = "image">

Example: <input type = "image" src = "img_submit.gif" alt = "Submit">

Attribute	Description	Syntax	Example
TYPE	This attributes specifies the type of button and it contain the various values such as submit, reset, button, and menu.	<input type = "value">	<input type = "image">
NAME	This attributes is used to assign the name of the control.	<input name = " name of the control ">	<input name = "submit">
SRC	This attribute specifies the address of the image where it is located.	<input src = " url ">	<input src = "img_submit.gif">
ALT	This attribute specifies the alternate text while cursor on the button then the text will be displayed on it.	<input alt = "alternate text ">	<input alt = "alternate text">

Table 7: Attributes for the image as Button

3.3.5. Image mapping: In image mapping, the image is specified with a different set of coordinates which act as hyperlink areas to connect with different locations.

An example of an image map is: Animal's map is an incomplete picture on a webpage that provides different links and these links are connected with other web pages image maps are the same as a map of currently we are different links are connected with different locations.

Types of image maps: There are two kinds of image maps.

1. **Server-Side Image Map:** This is an older method to create image maps. In this case, whenever the user clicks on the image, the browser sends the coordinates of the clicked region to the server, and then, the CGI program interprets the coordinates that these are valid or not. If it is a valid region, the server returns the URL of the region and sends it to the browser.

2. **Client-Side Image Map**: The image map is also known as client-side because the

browser does not need a CGI program and interprets the image map based on HTML tags client-side image maps are more reliable and faster than server-side image maps. The client-side image map does not need the presence of a server. Instead, it links the coordinates with the same HTML document.

To set up a client-side image map different tags are required and these are given as below:

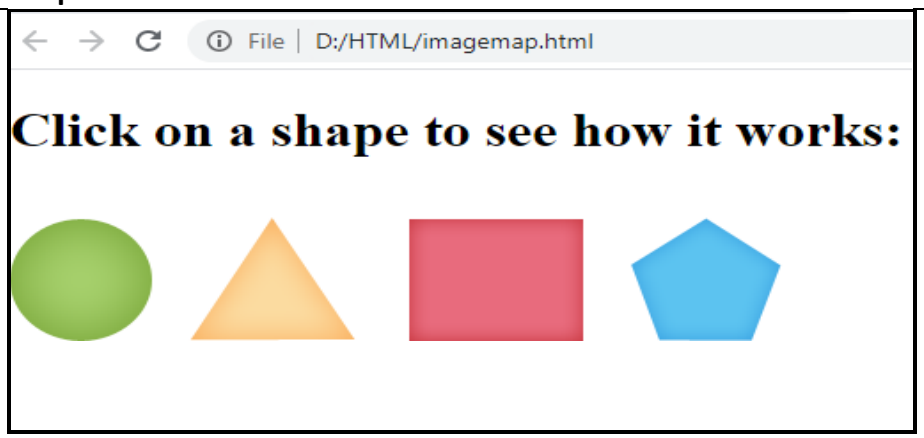
1. **Map:** It is used to create a map of the image with different clickable areas.
2. **SRC:** It is used for the image source.
3. **Area:** It is used with the URL and the coordinates of an image.
4. **Coords:** It defines the exact coordinates in pixels of hotspot regions. The number and order depend on the shape attribute. The shape can be given in

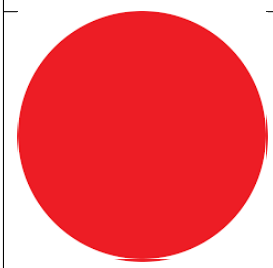
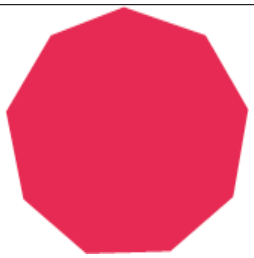
    a. **RECT:** x1, y1, x2, y2 (x1 and y1 represents the top-left coordinates and x2 and y2 represents the bottom-right coordinates).
    b. **CIRC:** x, y, r (x and y are the centers of the circle, and r specify the radius).
    c. **POLY:** x1, y1, x2, y2, …xn, yn (each value of x and y represents the vertex of polygon and the value of n represents the number of vertices).

Example

```
<html>
<head>
<title> Example of HTML image map </title>
</head>
<body>
   <h1> Click on a shape to see how it works: </h1>
   <img src = "D: \ HTML \ images \ finalimagemap.png " alt = "Geometrical Shapes" usemap = "#shapes">
   <map name = "shapes">
      <area shape = "circle" coords = "40 , 40 , 40" href = "D: \ HTML \ images \ img1 . png" alt = "Circle">
         <area shape = "rect" coords = " 226 , 2 , 323 , 80 " href = "D: \ HTML \ images \ img2 .png" alt = "Rectangle">
   <area shape = "poly" coords = "392 , 2 , 352 , 32 , 366 , 80 , 418 ,
```

Output



After clicking the circle	After clicking the rectangle	After clicking the polygon
		

80 , 432 , 32 " href = "D : \ HTML \ images \ img3 .png" alt = "Pentagon"> </map> </body> </html>	

Table 8: Example of image Map

3.4 Semantic Link Tag

The HTML <link>tag is used to represent the relationship between the current document and an external resource, such as a CSS file. This tag is empty means it has no need to close and it is always enclosed under the<head> <link> </head> tags.

Attributes	Description	Syntax	Exmaple
Href	This attribute is used to represent the URL of the external resource.	<link href = "URL">	<link href = "http:// www .xyz.com/style.css">
rel	This attribute is used to represent the relationship between current and linked document and it takes various kind of value, such as author, next, prev, style sheet etc.	<link rel = "value">	<link rel = "stylesheet" href = "http://www.xyz.com/style.css">
Type	This attribute is used to represent the media type of the linked document.	<link type = "media_type">	<link rel = "stylesheet" type = "text/css" href = "http://www.xyz.com/style.css">
Sizes	This attribute is used to repersent the sizes of icons for visual media.	<link sizes = "HeightxWidth| any">	<link rel = "icon" href = "http://www.xyz.com/style.css" type = "image/gif" sizes = "10x10">

Table 9: Attributes of Link tag

3.5 META Tag

The META tag is used for the easy accessibility of the web page. The search engines can easily locate the web page if the META tag is specified in the HTML code. Once a user enters a search

string, the search engine displays the pages that contain the string, but the only condition is that the META tag contains the information on the web page. The search engine finds the META Tag of HTML to display the required page.

The Meta tags contain two attributes:

1. NAME: This attribute includes the type of META tag.
2. CONTENT: This attribute is containing the keywords that are listed by the search engine.

Attributes of meta tag :

Attributes	Properties	Discription
charset	character-set	It specify the character encoding for the HTML document.
content	text	It specifies the value associated with the http- equiv or name attributes.
http-equiv	content, security, policy, content-type, default-style, refresh	It provides an HTTP header for the info|value of the content attributes.
name	application,name,author, description, generater, keywords	It Specify a name for the metadata.

Table 10: Attributes of Meta Tag

Check Your Progress 2

1. State Weather True or false:
   i. HTML web pages can accept .JPEG and .GIF formats.
   ii. HSPACE is used to describe the amount of space to the top and bottom of an image.
   iii. ALT attribute always displays the text of an image, even if the image is displayed on the browser.
   iv. SRC can also be specified with the absolute or relative pathname.
2. Explain the properties or attributes of an Image tag.
3. Explain the different types of images supported by the browser.
4. What do you mean by image map and how it is implemented on the browser?
5. What is the purpose of the META tag?

Objective Type questions:

1. What is the correct sequence of HTML?

a) HTML, BODY, HEAD, TITLE
b) HTML, HEAD, TITLE, BODY
c) HTML, TITLE, HEAD, BODY
d) HEAD, BODY, HTML, TITLE

2. Which formatting tag is used to make the text bold?

a) <pre>
b) <b>
c)
d) <i>

3. Which is the second largest heading in HTML?

a) <h3>
b) <h1>
c) <h6>
d) <h2>

4. The tag used to insert a line break in HTML is:

a)
b) <a>
c) <hr>
d) <b>

5. How to create a list with bulleted items in HTML?

a) <OL>
b) <LI>
c) <UL>
d) <I>

6. Which syntax is correct for the hyperlink?

a) Text
b) Text
c) Text
d) <a href = "abc.html" Text /a>

7. How to insert an image in the web document?

a)
b)
c)
d)

8. What is the correct way to send mail using the HTML element?

a) Text
b) Text

c) <mail ab@a> </mail>
d) None of the above

9. Which tag is used to create a paragraph in HTML?

a) <a>
b) <p>
c) <b>
d) <pre>

10. The extension used to save HTML document is

a) .ht
b) .html
c) .hml
d) None of the above

11. The program created in HTML can be read and interpreted by

a) Server
b) Compiler
c) Browser
d) Interpreter

12. The tags in HTML can be

a) Case - sensitive
b) Upper case
c) Lower case
d) Case insensitive

Hands on Practice

Example 1:

```
<!DOCTYPE html>
<html>
  <head>
    <title> Hyperlink Example </title>
  </head>
  <body>
    <p> Click any of the following links </p>
    <a href  =  "index.html" target =  " _blank "> Opens in New </a> |
    <a href  =  "index.html" target =  " _self "> Opens in Self </a> |
    <a href =  "index.html " target  =  " _parent " > Opens in Parent </a> |
    <a href =  "index.html" target  =  " _top "> Opens in Body </a>
  </body>
```

```
</html>
```

Example 2:

```
<!DOCTYPE html>
<html>
<head>
<style>
a:link {
   color: green;
   background-color: transparent;
   text-decoration: none;
}
a:visited {
   color: pink;
   background-color: transparent;
   text-decoration: none;
}
a:hover {
   color: red;
   background-color: transparent;
   text-decoration: underline;
}
a:active {
   color: yellow;
   background-color: transparent;
   text-decoration: underline;
}
</style>
</head>
<body>
<p> You can change the default colors of links </p>
<a href = "html_images.html" target = "_blank"> HTML Images </a>
</body>
</html>
```

3.6 Summary

This unit helps to create an interactive web page using the hyperlinks and images. Hyperlinks are used to create a link between one or more documents of a website. Internal links helps in linking a part of document within particular location and the external link helps in creating a link outside the document or a new document. <IMG> tag is used to insert images in the HTML document and also used to create image maps or image links in HTML. Different formats of images are there that helps in creating an attractive website. It can also be possible to create image maps. An example of image map is google map. Semantic links can be used to create link with external sources. The META tag is another useful tag that can be used for redirecting the

search engine to the appropriate web page. It includes the keywords that are accessible by the search engine and displays the web page on the browser.

UNIT 4 LAYOUT

Structure

4.0 INTRODUCTION

The previous unit contains information about the basics of HTML and after studying the implementation of the static web page. The next step is to learn the implementation of an interactive website. An interactive website is easy to use and design. Interactive websites are the websites that take input from the user and based on the inputs, the output is displayed. This can be made possible with the advanced HTML features. Some advanced features are font, background iamges, Tables, etc. With the help of these tags, one can easily divide the browser's window into multiple parts and represent the list in proper order. There are some other attributes of advanced tags that are listed in this unit.

4.1 OBJECTIVES

After completing this unit, learners will be able to understand the concepts of HTML and they will be able to design the web pages.

- Webpage Background with color and images
- Control text attributes using different font
- Working with Background Images
- Data represented in the form of Tables

4.2 HTML Background

In HTML, the <body> tag contains the information or the content. All the content of the web page is to be given in the body section. Thus, the text can be any headings, textbox, checkbox, images, links, etc. anything that needs to be displayed on the browser must be given in the body section. To give the background image or color the designer must specify the setting in the body tag. HTML provides two ways to decorate the web page.

1. Background with colors
2. Background with images

4.2.1. Background with Colors:

The BGCOLOR attribute of the body tag is used to give the background color of the web page. The default color is white. The value of color can be given in two formats such as RGB format or Hexadecimal format. The value of this format is given as "#000000" to "#FFFFFF". The # indicates to the browser that it is a hexadecimal value. Total six digits are available in hexadecimal number in which the first two represents red color, second two represents the green

color and the last two represents the blue color. Thus, the combination of primary colors Red, Green, Blue is also known as RGB colors. It is also given as for red "#FF0000", for green " #00FF00", and for blue "#0000FF" and the other combination gives the different color.

Syntax:

<Body BGCOLOR = " #FF0000 ">

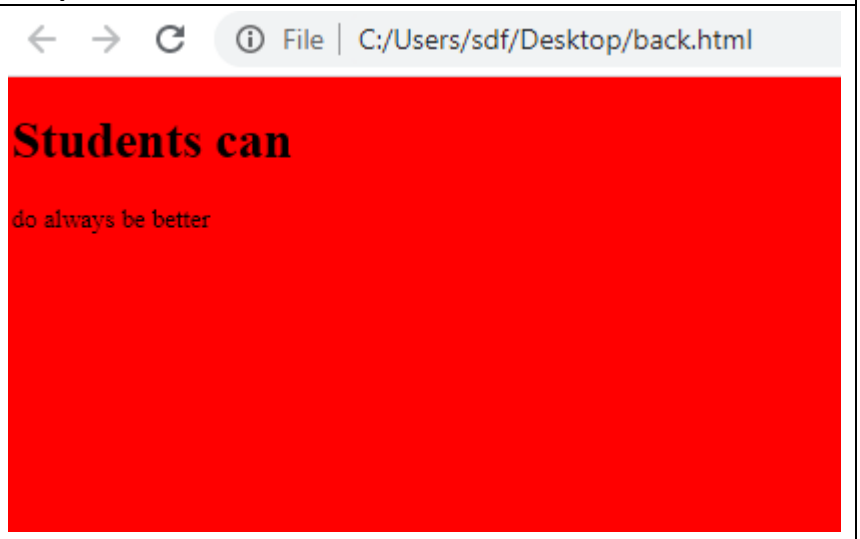
4.2.2. HTML background with images:

The background attribute can also represent a background image for an HTML webpage. The value of the attribute is the address (URL) of the image, which is used by the designer. The size of the image is given according to a page in pixels(px) or percentage (%) format. If the size of the image is smaller than the browser window, then it will repeat itself until it fills the whole window.

Note: The background attribute was deprecated in HTML5. The W3C (World Wide Web consortium) has eliminated these attributes from the latest version of HTML5, but at present, it will be used in CSS for setting the background.

Syntax: <Body Background = " URL of Image " >

Handy Tips of background color and image

Example of background-color	**Output**
<html> <head> <title> HTML Backgorund Color </title> </head> <body bgcolor = red> Students can do always be better </body> </html>	 **Figure 1: Output of background-color**
Example of background Image	**Output**

```
<Html>
<Head>
<Title>
 Background image
</Title>
</Head>
<Body background  =  "C:\ Users \Public
\Pictures \Sample Pictures \Lighthouse.jpg">
HTML <br>
DHTML  <br>
This code will help the bigners to understand
the background - image
</Body>
</Html>
```

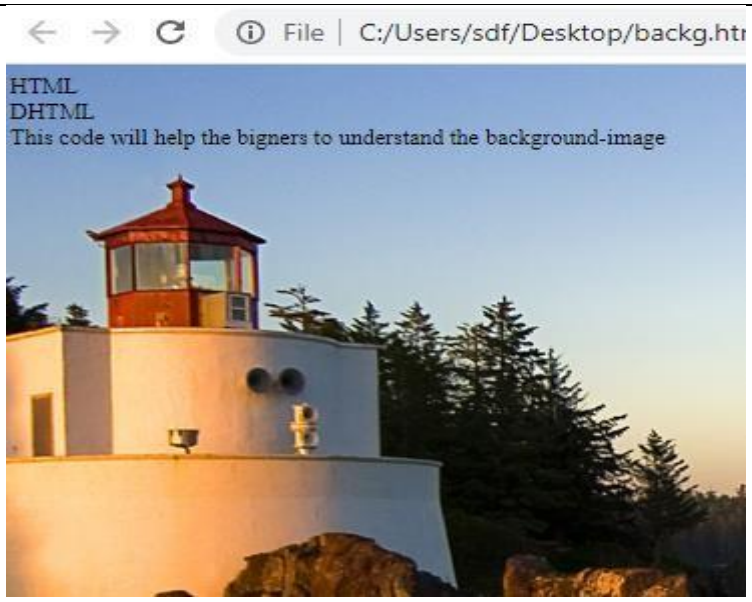


Figure 2: Output of background image

Table 1: Example of Background-color and Background-image

4.3 Colors in HTML

Colors play an important role in creating attractive websites. By default, in HTML there is no separate tag to represent the colors, it makes use of the style attribute of the color's property. Colors give a stylish look to the web page, background, tables, paragraph, etc.

There are following three different color methods and these are explained as:

4.3.1. Color names:A web designer can design a web page with a color name such as yellow, red, and so on. W3C has listed 16 numbers of basic color names but major browsers supported 200 different color names.

4.3.2. Hexadecimal code or Hex code: A web designer can design the web page with another method i.e., hexadecimal code. It is a six-digit code representing the two amounts of red, two for green, and the last two digits for blue color that make up the color. Each hexadecimal code will be preceded by a hash sign (#).

	White	#FFFFFF
	Yellow	#FFFF00
	Fuchsia	#FF00FF
	Red	#FF0000
	Silver	#C0C0C0
	Gray	#808080
	Olive	#808000
	Purple	#800080
	Maroon	#800000
	Aqua	#00FFFF
	Lime	#00FF00
	Teal	#008080
	Green	#008000
	Blue	#0000FF
	Navy	#000080
	Black	#000000

Figure 3: Hexadecimal codes

4.3.3. RGB values: The web designer can design the web page with RGB () property. The property takes three values one for red, one for green, and one for blue, and it ranges from 0 to 255 or a percentage value.

Color	Color RGB
	rgb(0,0,0)
	rgb(255,0,0)
	rgb(0,255,0)
	rgb(0,0,255)
	rgb(255,255,0)
	rgb(0,255,255)
	rgb(255,0,255)
	rgb(192,192,192)
	rgb(255,255,255)

Figure 4: RGB color combinations

Note: Most of the computer systems supported 216 colors out of 256 different colors because all computer systems use 216 cross-platform colors, so these 216 colors are known as "web-safe" colors. RGB combinations can produce more than 16 million different colors to play with (256 * 256 * 256) patterns.

4.4 Text

The HTML <body> text Attribute is used to define a color for the text in the webpage.

Attribute	Description	Syntax	Example

color_name	It specifies the name of the color for the text in the Document.	<body text = "color_name">	<body text = "red">
hex_number	It specifies the hex code of the color of the Text in the Document.	<body text = " hexadecimal_code " >	<body text = " #ff0000 ">
rgb_number	It specifies the rgb value of the Text in the Document.	<body text = rgb_number (RR, GG, BB) >	<body text = rgb (255,0,0) >

Table 2: Text attributes for color

4.5 Design Requirement

Website Requirements only tell the designer what he will do for their clients. It also tells about features, functions, characteristics related to the website, and the content of the website. Requirements can begin as a step-by-step process, thus the client and the designer have no issue after the completion of the website or we can say it is an agreement between the two parties because it can be a complex task, therefore it cannot be possible without understanding. Several types of requirements may be defined during the process that comes together to focus and prioritize the project plan. The following requirements are given below:

Types of Requirements:

- **Business Requirements**: It defines the objectives and the problems solved related to the owner.
- **User Requirements** It describes the user expectations and interaction with the product by the user. In which user gives their requirement to the designer, so that designer fulfills all the tasks according to user needs.
- **Functional Requirements**: It provides detailed information about the product behavior, functional behavior, and needs related to the development of the project.
- **Quality-of-Service Requirements:** It provides a detailed overview of the characteristics of the product, to maintain its effectiveness and constraints under the development process.
- **Implementation Requirements**: It shows the team roles, detailed changes in process and migration from one system to another system, etc.

- **Some other Requirements:** In which some information related to logo designing, color combination, domain hosting, image optimization, user friendly, compatibility on mobile and computer system and maintenance.

4.6 Attributes for <BODY>

The <body> tag has the following attributes which can be used to set different colors.

Attributes	Description	Syntax	Example
BGCOLOR	It will be used to set a color of the background of the page.	<body bgcolor = " colorname ">	<body bgcolor = "red">
Text	It will be used to set a color for the individual text.	<body text = "color name">	<body text = "red">
Alink	It will be used to set a color for the active link on a web page.	<body alink = "color name | Hex number | rgb_number " >	<body alink = "green">
Vlink	It will be used to set a color for the visited link when a client clicks on the text.	<body vlink = "color name | hex number | rgb number">	<body vlink = "green">

Table 3: Attributes of Body tag

Handy Tips of Body attributes:

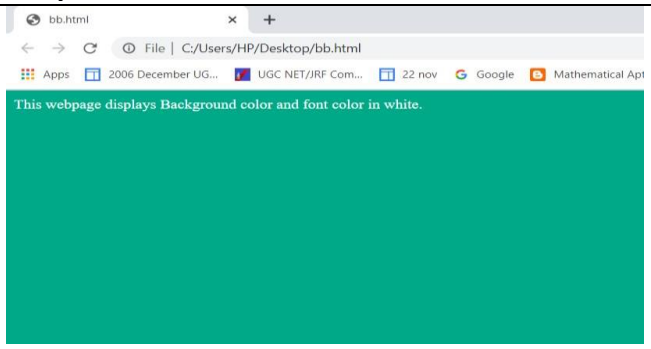
Example	Output
<html> <head> </head> <body bgcolor = " 00AA88 " text = white> This webpage displays Background color and font color in white. </body> </html>	 **Figure 5: Output of body attribute example**

Table 4: Examples and output of Body attributes

4.7 HTML approach to Web design

Web designing is not only a process for creating a website. There are four different ways to create a website and all these techniques do not require the knowledge of HTML and CSS, though it is for the designer convenience that they have some basic idea of HTML and CSS. The four ways to creating a website are:

1. Program it from scratch:

If a new Web designer or beginner has already good HTML and CSS skills, then designing a website from the initial point is the most cost-effective otherwise HTML and CSS are a time-consuming option. Thus, there are several latest code editors available in the market such as Dreamweaver, notepad-plus-plus, etc. These editors help to create a website a little easier than HTML and CSS.

2. Start with a template:

A web designer creates the website by using a website template. In this case, the programmer has to use their programming skills for editing the content according to their needs.

3. Use a website builder:

Website builders advertise themselves as the easiest and quickest way to create a website. Companies are the best alternative for helping to create the website, they offer a full package for design a website with free hosting and a free subdomain name. Sometimes, companies charge monthly or yearly from their customers for the full domain name and they also charged for customer advertisements, extra storage space, and bandwidth restrictions.

4. Utilize content management system:

CMS is the short-term content management system if a person has skills related to HTML and CSS. Then, with the help of CMS, the designer can create the website according to their needs. Some CMS are available which helps to create HTML websites such as word press, Joomla, and blogger, all these are available online for free of cost. These CMS are built with a database and drag and drop facilities are available in these CMS.

4.8 Introduction to Font

Font tag is one of the most important tags used in an HTML webpage. It is used in place of the text with different attributes font size, color, and style. Font tag is used within the inline element to change some features of block text written in HTML document. The text enclosed within tag.

1. Size: The size attribute is used to define the text with a specific size from 1 to 7. The default size of the font is 3.

Very Small	Small	Regular	Medium	Large	Extra Large	Largest
1	2	3	4	5	6	7

Table 5: Size of font

2. Face or Type: This attribute is used to define the font type such as Times New Roman, Arial, etc. Its value can be the name of the available font or desired font that must exist in the system. If the name not already available in the system, the browser will take the default name.

3. **Color:** This attribute is used to change the color of the text with RGB or hexadecimal format as already described. The hexadecimal code is given in the earlier part. The default color of the font is black. If the text color is given in this tag, then the following text after this tag appears in the color given in that tag.

Handy Tips of Font tag:

Example	Output

<html> <head> <title> Font Tag </title> </head> <body> Font tag example with attributes “Successful and unsuccessful people do not vary greatly in their abilities. They vary in their desires to reach their potential.” “Strive for progress, not perfection.” “There are no shortcuts to any place worth going.” </body> </html>	 **Figure 6: Output of Font attribute example**

Table 6: Examples and output of Font attributes

4.9 Introduction to TABLES

A two-dimensional matrix that consists of rows and columns is known as a Table. It is used to display data in the form of rows and columns on a web page. These are formulated by using top-down and left-right strategies. Following are the tags used to create a table, row, column.

<TABLE> </TABLE> In HTML, the table is created by adding all the tags in these tags.
<TR> </TR> Each row of the table starts with the beginning tag and ends with the ending tag. The number of columns can be calculated automatically by adding several cells in the row.

Tables rows in HTML can be given using two types:

- **<TH> … </TH>** It is referred to as the Table Header. The content given in between the opening <TH> and closing </TH> tag is automatically aligned as a center and in

boldface.

- **<TD> … </TD>** It is defined as Table data. It is used to represent the data cells given in horizontal rows. A table includes single data (single column) and multiple data (multiple columns). The text given in the table is by default left-aligned or justified.
- **<caption>** It is used to defines the caption of table.
- **<callgroup>** It is used to specifies a group of one or more columns in a table.
- **<tbody>** It is used to represent the group of the body content in a table.
- **<thead>** It is used to specify the group of header content in a table.
- **<tfooter>** It is used to group the footer content in a table.
- **<Label> tag** The label tag is used for screen reader users because our readers will read out the labels when users focus on the input tag. It should be same as ID attribute of the input element to bind them together.

4.9.1. Attributes of Tables: The attributes of the table can be used for setting the width, border, cell-spacing, etc. The most commonly adopted attributes are:

Attributes	Values	Syntax	Example
ALIGN	This attribute is used to set the horizontal alignment of the table concerning the browser window. The value for ALIGN attribute is left, center, right.	<Table align = " left | center | right ">	<Table align = " center ">
VALIGN	This attribute is used to set the vertical alignment of the content given in the cell. The value for VALIGN attributes is the top, middle, bottom.	< Table valign = "top | middle | baseline | bottom" >	<Table valign = " middle">
BORDER	The border attribute displays the width of the table border. The table does not contain a border by default. by default its value is 1. While applying the border attribute, it appears in all the tables as well as in each cell of the table.	< Table Border > or < Table Border = "1" >	<Table Border> or <Table Border = "1">
WIDTH	The Width attribute sets the table width in the form number of pixels or according to the horizontal width of the browser in percentages such as (50%, 150%, etc.). If the width attribute is not specified then it will be adjusted according to the cell's data.	< Table width = "value" border = "value" >	<Table width = "50%" border = "2">

CELLPADDING	Cellpadding attributes set the free space between content and the boundary of the cell. The default value of cell padding is 1 pixel. Its value can be given either in pixels or in percentage.	< Table cellpadding = "value" border = "value" >	<Table cellpadding = "10%" border = "1">
CELLSPACING	The cellspacing attribute is used to set the spacing between the cells. The default value of Cellspacing is 2 pixels.	< table cellspacing = "value" border = "value" >	<table cellspacing = "5" border = "3">
ROWSPAN	The rowspan attribute allows a single cell to spread in more than one row or how many rows can be occupied by one cell. It can be represented with the <TD> or <TH> tag. the default value of the rowspan attribute is 1.	< TD rowspan = "value" >	<TD rowspan = "2">
COLSPAN	The colspan attribute allows a single cell to spread in more than one column or how many columns can be occupied by one cell. It can be represented with the <TD> or <TH> tag.	< TD colspan = "value" >	<TD colspan = "2">
RULES	This attribute is used to control the inner borders in a table between the cells. Its value can be given as none, cols, rows, all, groups. Syntax:	< table rules = " " >	<table rules = "">

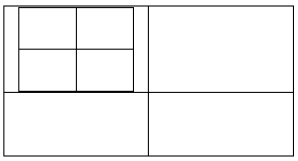
Table 7: Attributes of <Table> tag

4.9.2. The Caption Tag: The caption represents the heading that gives readers a piece of information about the related content in the table. It can be given using the <CAPTION> ...</CAPTION> tag. This tag is given in between the <TABLE>... </TABLE> tags. It can appear above or below the table using the align attribute.

Attributes	**Values**
ALIGN	This tag places the caption of the table at the top or bottom. ALIGN = TOP caption appears above the table. ALIGN = BOTTOM caption appears below the table. Syntax:<CAPTION ALIGN = " TOP "> …. </CAPTION>

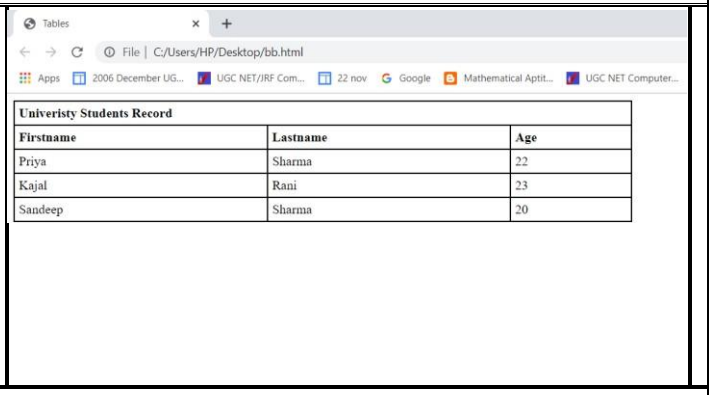
Table 8: Attributes of <Caption> tag

4.9.3. Nested Tables: These kinds of tables are created by placing one table inside another table. In other words, an entire table can be placed in a cell of another table.
Example of Nested Table is:



Handy Tips of Tables:

Example	Output

<html>
<head>
<style>
table, th, td {
border: 2px solid black ;
border-collapse: collapse ;
background-color: green ;
align: center ;
}
th, td {
padding: 5px ;
text-align: left ;
align: center ;
}
 	
table#t01 {
width: 100% ;
}
</style>
<title> Tables </Title>
</head>
<body>
 	
<table style = "width:50%">
<tr> <td colspan = "3"> Univeristy Students Record </td> </td>
<tr>
<th> Firstname </th>
<th> Lastname </th>
<th> Age </th>
	
	Tables
	File | C:/Users/HP/Desktop/bb.html
	Apps 2006 December UG... UGC NET/JRF Com... 22 nov Google Mathematical Aptit... UGC NET Computer...
	Univeristy Students Record
	Firstname | **Lastname** | **Age**
	Priya | Sharma | 22
	Kajal | Rani | 23
	Sandeep | Sharma | 20
	Figure 7: Output of table

```
    </tr>
    <tr>
      <td> Priya </td>
      <td> Sharma </td>
      <td> 22 </td>
    </tr>
    <tr>
      <td> Kajal </td>
      <td> Rani </td>
      <td> 23 </td>
    </tr>
    <tr>
      <td> Sandeep </td>
      <td> Sharma </td>
      <td> 20 </td>
    </tr>
  </table>
  <br />
  <br />
</body>
</html>
```

Table 9: Example of Tables

Check your progress 1

6. State Weather True or false:
   v. The vertical and horizontal alignment can be controlled by the VALIGN and ALIGN attributes of the row using the <TR> tag.
   vi. Table header is defined using the <TD> … </TD> tag.
   vii. RULES attribute controls the spacing between cells.
   viii. cellspacing is used to set the space between data in a cell.
7. Design a web page using HTML that includes the detail of academic achievement in the table.
8. Design a web page that contains the detail of the following trains.
   - Name of Train
   - Starting location
   - Stopping location
   - Check-in and check-out
   - Price

A table includes the border and cell padding and alignment should be center. The output of the table is given in the following format.

Name of Train	Starting Location	Stopping location	Timings		Price
			Check-in	Check-out	
A	Delhi	Mumbai	9:00	9:10	200
B	Mumbai	Delhi	10:00	10:10	300

C	Rajasthan	Delhi	11:00	11:10	400

Table 10: Example of Table

Hands on Practice

Example 1:

```
<!doctype html>
<html>
<head>
<meta charset = "utf-8">
<title> NestedTables </title>
</head>
<body>
<caption title = "Container Table"> Container Table </caption>
<table border = "5px" bordercolor =  "red">
<tr>
<td >
<table>
<tr> <th colspan = "2"> Nested Table 2 </th> </tr>
<tr> <th> Column 1 </th> <th> Column 2 </th> </tr>
<tr> <td> Our First Table </td>
<td> Nested Within </td> </tr>
</table>
</td>
<td>
<table >
<tr> <th> Nested Table 2 </th> </tr>
<tr>
<td>
<ul>
<li> List Object 1 </li>
<li> List Object 2 </li>
<li> List Object 3 </li>
</ul>
</td>
</tr>
</table> </td> </tr>
<tr>
<td>
<table>
<tr> <th colspan = "2" align = "center"> Nested Table 3 </th> </tr>
<tr>
<td> <a href = ""> Nested Table </a> </td>
<td> Demo Continued </td> </tr>
```

```
</table> </td>
<td>
<table>
<tr> <th> Nested Table 4 </th> </tr>
<tr>
<td> <img src = "images.png" height = "120px" width = "120px" alt = "Sorry Image could not be displayed"> </td> </tr> </table>
</td>
</tr>
</table>
</body>
</html>
```

Example 2:

```
<!DOCTYPE html>
<html>
<head> <title> Special symbols in HTML </title> </head>
<body>
<h3> Mathematical Operations Symbols </h3>
<p> Plus Minus Sign  <b> &#177 </b> </p>
<p> Multiplication sign <b> &#215 </b>  </p>
<p> Division sign <b>  &#247 <b> </p>
<p> One fourth part showing sign <b> &#188 </b> </p>
<p> Half part showing sign <b> &#189 </b> </p>
<p> Three Fourth part showing sign <b> &#190 </b>  </p>
<p> There Exist Sign <b> &#8707 </b> </p>
<p> Empty Set Sign <b> &#8709 </b> </p>
<p> CONTAINS AS MEMBER sign <b> &#8715 </b>  </p>
<p> N - ARY PRODUCT sign <b> &#8719 </b> </p>
<p> N - ARY SUMMATION <b> &#8721 </b>  </p>
<p> Not a element sign <b> &#8713 </b> </p>
</body>
</html>
```

Example 3:

```
<!DOCTYPE html>
<html>
<head>
<title> Setting Basefont Color </title>
</head>
<body>
<basefont face = "arial, verdana, sans-serif" size = "2" color = "#ff0000">
<p> This is the page's default font . </p>
<h2> Example of the &lt;basefont&gt; Element </h2>
</body>
```

```
</html>
```

Example 4:

```
<HTML>
<HEAD>
<TITLE>
 HTML Table Design
</TITLE>
<style>
table, th, td {
 border: 1px solid black;
 }
</style>
</HEAD>
<BODY>
 <TABLE style= "width: 80%" align = "center">
 <caption> CSC503 timetable < / caption>
 <tr >
 <td width=50%> </td>
 <th width = 150> Monday < /th>
 <th width = 150> Tuesday </th>
 <th width = 150> Wednesday </th>
 <th width = 150> Thursday </th>
 <th width = 150> Friday </th>
 </tr>
 <tr >
 <td > 6-7pm </td>
 <td > Look at website</td>
 <td > free </td>
 <td > Implementation </td>
 <td > free </td>
 <td > free </td>
 </tr>
 <tr >
 <td > 7 - 8 pm </td>
 <td > Take some notes</td>
 <td > free </td>
 <td > Implementation </td>
 <td > free </td>
 <td > free </td>
 </tr>
<tr >
 <td > 8 - 9 pm </td>
 <td > Take some notes</td>
 <td > free </td>
```

```
<td > Implementation </td>
<td > free </td>
<td > free </td>
</tr>

</TABLE>
</BODY >
</HTML >
```

4.10 SUMMARY

This unit covers the important and advanced features of HTML. This helps in designing an attractive website and the data is represented in the form of tables. It display the screen with the color or background images that help in designing an attractive website. The table can be designed by following the instructions given in this unit that includes the details in the form of biodata or any reports in a very simple way to better understand the information. The tables can also be represented in nested form. One can easily understand the tabular data instead of a paragraph. There exist some properties of table that can be represented as cellspacing, cellpadding, border, margin, color, bgcolor, etc. that help in giving an attractive look to the web page.

UNIT 5 LIST AND FRAMES

Structure **Page No**

5.0 INTRODUCTION

The previous unit contains information about the basic tags and advanced features of creating tables and nested tables. Some other advanced features are List, Frameset, etc. With the help of these tags, one can easily divide the browser's window into multiple parts and represent the list in proper order. Lists can be categorized into Ordered List, Unordered List and Definition List. There are some other attributes of advanced tags that are listed in this unit.

5.1 OBJECTIVES

This unit helps the learners to understand the concepts of HTML Lists, frames and they will be able to design the web pages and website that contain multiple frames.

- Lists
- Frames for dividing the web page into different parts

5.2 List

The list represents the data in points mode such as items list, etc. The list breaks the long paragraphs into points for directing the reader's attention to important information. The list can be classified into three forms:

1. Ordered List
2. Unordered List
3. Definition List
4. Nested List

5.2.1. Ordered List:

The ordered List includes the numbered listing of items. The numbering is in the form of A, B, C or a, b, c or 1, 2, 3 or I, II, III, etc. The list is incremented after each successive order. An ordered list can be created by the following tags:

<OL></OL> <OL> represents an ordered list with a beginning and ending tag.
<LI></LI> <LI> represents list item.

Attributes of an ordered list: It includes the attributes of the ordered list:

Attributes	Values	Syntax	Example
TYPE	Following are the feasible options of the numbered list. <OL TYPE = "1" > default case that includes numbers such as 1,2,3... <OL TYPE = "a"> Lowercase letters such as a, b, c..... <OL TYPE = "A"> Uppercase letters such as A, B, C..... <OL TYPE = "i" > Lowercase letters that include numerals such as i, ii, iii.... <OL TYPE = "I"> Uppercase letters that include numerals such as I, II, III....	<OL TYPE = "1" >	<OL TYPE = " 1" >
START	<OL TYPE = "A" START = 5> It starts the numbered list beginning from E. It can be set to any number.	<OL TYPE = "A" START = value>	<OL TYPE = "A" START = 5>

Table 1: Attributes of Ordered List

5.2.2. Unordered List

An unordered list includes the collection of items without having any sequence and order. The items in the list are represented in the form of bullets. An Unordered list can be created by the following tags:

<UL></UL> <UL> represents an Unordered list with a beginning and ending tag.
<LI></LI> <LI> represents list item.

Attributes of Unordered List:

Attributes	Values	Syntax	Example
TYPE	Following are the feasible options of the bulleted list. <UL TYPE = "square">It shows the solid square bullet in black. <UL TYPE = "circle"> It shows the hollow circle. <UL TYPE = "disc"> By default it is used and it will give a solid filled circle.	<UL TYPE = "value">	<UL TYPE = "square">

Table 2: Attributes of Unordered List

5.2.3. Definition List

The definition list includes the definitions that are represented in the form of a dictionary or encyclopedia. It represents the dictionary, list of definitions, etc. in the order of the list. It includes the heading and the text appeared below the heading. This list is created using the following tags:

<DL></DL> Definition list creates the list of definitions.
<DT></DT> Definition term includes the heading.
<DD></DD> Definition description includes the definition.

5.2.4. Nested List

Lists can be represented in nested form. A nested list can be defined as the list within another list. In other words, a Nested list can be represented as an element of another list or a sub-list. It works for creating a menu-like entity where the hierarchy is required.

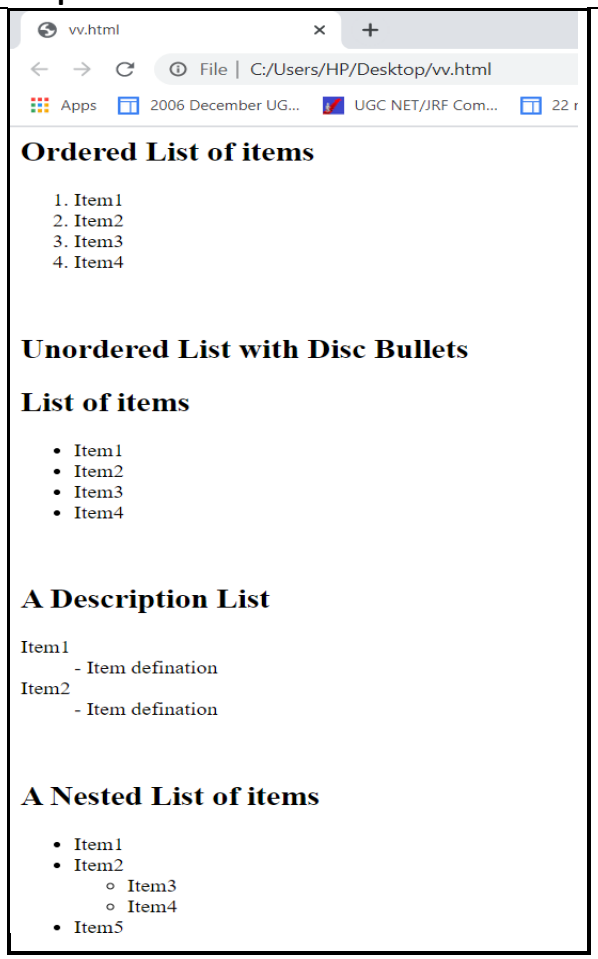
Example:

1. Tea
    - Black Tea
    - Green Tea
2. Milk
3. Shakes
    a. Banana Shake
    b. Mango Shake
    c. Strawberry Shake

Advantages of HTML List:

1. **Flexibility**: If a designer has to change the order of the list items, then he/she just move the items according to requirement.
2. **Styling**: List allows the designer to style the list properly by using CSS, so it can be format according to need.
3. **Semantics**: HTML lists give the content the proper semantic rules. It helps the readers while reading as well as they does not confuse in text and numbers.

Handy Tips of Lists:

Example	Output
html> <head> <title> Types of List </title> </head> <body> Ordered List of items - Item1 - Item2 - Item3 - Item4 Unordered List with Disc Bullets List of items <ul style = "list-style-type:disc"> - Item1 - Item2 - Item3 - Item4 A Description List <dl> <dt> Item1 </dt> <dd> Item definition </dd> <dt> Item2 </dt> <dd> Item definition </dd> </dl> A Nested List of items - Item1 - Item2 - Item3 - Item4 - Item5 </body>	 **Figure 1: Output of Lists**

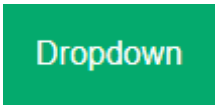
</html>

Dropdown List:

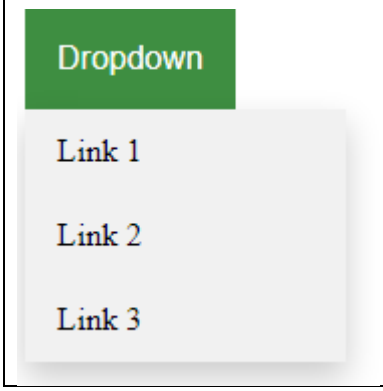
```
<!DOCTYPE html>
<html>
<head>
<meta name = "viewport" content = "width = device-width, initial-scale = 1">
<style>
.dropbtn {
  background-color: #04AA6D;
  color: white;
  padding: 16px;
  font-size: 16px;
  border: none;
}

.dropdown {
  position: relative;
  display: inline-block;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f1f1f1;
  min-width: 160px;
  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

.dropdown-content a {
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
}

.dropdown-content a:hover {background-color: #ddd;}

.dropdown:hover .dropdown-content {display: block;}

.dropdown:hover .dropbtn {background-color: #3e8e41;}
```

Output

Before Click the Dropdown Button



After Click the Dropdown Button



```
</style>
</head>
<body>
<h2>Hoverable Dropdown</h2>
<p>Move the mouse over the button to open the
dropdown menu.</p>

<div class = "dropdown">
  <button class = "dropbtn">Dropdown</button>
  <div class = "dropdown-content">
    <a href = "#">Link 1</a>
    <a href = "#">Link 2</a>
    <a href = "#">Link 3</a>
  </div>
</div>

</body>
</html>
```

Table 3: Example of Ordered and Unordered List

Check Your Progress 1

1. State Weather True or false:
   i. List are categorized into three types: <OL>, <UL>, <DL>
   ii. Starts attribute maintains the ordering sequence in an unordered list.
   iii. An Unordered List type attribute's value can be disc, circle, square.
   iv. Definition list consists of < DL>, <DT>, <DD> tags.
2. Design the following given web page using List.

// Example of Unordered List

- Time
- Space
- Business

// Example of Ordered List

v. Sports
vi. Dance
vii. Music

// Example of Definition List

Computer
: Common operating machines are particularly used in trade, education, and research.

HTML
: Hypertext Markup Language.

Table 4: List example

5.3 Frames

Frames are one of the most important concepts in HTML. The chief objective of frames is to divide the browser's window into multiple regions where each region contains a different HTML page. Each region works independently and is known as Frame. The frame helps us to display multiple web pages on a single browser window. With the help of frames, one can load and reload several parts of the window instead of the complete browser window.

Each frame is separated by a border on a browser. Each frame may contain a different HTML document which is represented by its URL. The scroll bar is available in each frame to change the size of the frame and one frame can be linked with the other frame.

A collection of frames on the browser's window is known as a frameset. It does not contain any content, though it tells the browser to divide the window into frames. The frameset can be created using <FRAMESET> ... </FRAMESET> tags.

The two important frame tags are <FRAMESET> and <FRAME>.

5.3.1. <FRAMESET> Tag: It divides the browser's window into frames by embedding the <FRAMESET> … </FRAMESET> tag in the HTML document. It requires one of the two attributes to divide the screen into rows and columns.

Attribute	Description
Rows	This attribute divides the screen into multiple rows on the browser window. The number of frames specified by the number of values and is separated by a comma. The value can be in the form of • In Pixel • In Percentage • Symbol '*' represents remaining space.
Cols	This attribute divides the screen into multiple columns on the browser window. The number of frames specified by the number of values and is separated by a comma. The value can be in the form of • In Pixel • In Percentage • Symbol '*' represents remaining space.

Table 5: <Frameset> attributes

5.3.2. <Frame> Tag: Once the <Frameset> divides the browser window into horizontal or vertical sections, each defined section can be loaded with a separate HTML document. It is a singular tag. and it contains the following attributes.

Attribute	Description
SRC = "URL"	This attribute indicates the address or URL of the document that is loaded in the frame.
Name	This attribute gives the name to the frame to be referenced. This can be used when the links of one frame need to be loaded in another frame. Syntax: <frame name = "value" src = "value">
MarginHeight = "n"	This attribute is used to set the white space above and below the frame. The value is to be specified in pixels. Syntax: <frame marginheight = "10">
Marginwidth = "n"	This attribute is used to set the white space left and right of the frame. The value is to be specified in pixels. Syntax: <frame marginwidth = "10">
Noresize	This attribute will not allow the users to resize the frame. It does not contain any value. The default setting of this attribute is disabled. Syntax: <frame noresize src = " ">
Scrolling	This attribute controls the appearance of the horizontal and vertical scroll bars in a frame. The values can be given as yes/no/auto. **AUTO** feature automatically adds the scrollbar as the content in the frame increases. **YES** indicates the scrollbars are included in the frame.

	NO indicates that no frame includes scroll bars.
Frameborders and Bordercolor	This attribute makes the border on or off and also adds the color of borders. frameborder takes 2 values: 1 (ON) or 0 (OFF). The default value will be 1. Syntax: <Frame Frameborder = "1">
Longdesc	This attribute is used to provide the link to another page with a longer description.

Table 6: <Frame> attributes

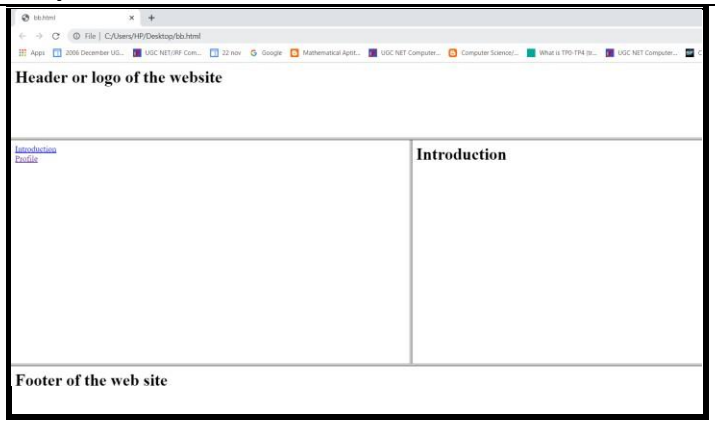
Advantage of Frames

1. Without loading the complete window it can load and reload multiple regions.
2. By using a frame, multiple web pages can be displayed on a single browser instead of going back and forth.
3. Navigation system can be updated by making changes only on a single web page for the complete website.

Disadvantage of Frames

1. The loading time of frames is increased as Frame is loaded separately.
2. Some devices smaller in size may not work with the frames.
3. Some browsers do not support frame technology.
4. Bookmarking is difficult on frames.
5. Mobile users may not work with frames.

Handy Tips of Frames:

Example	Output
<frameset rows = "20%,60%,*"> <frame src = "f1.html"> <frameset cols = "50%,*"> <frame src = "f2.html"> <frame src = "f3.html" name = "f3"> </frameset> <frame src = "f4.html"> </frameset>	 **Figure 2: Output of frames**
F1.html	

<table>
<tr><td>

```
<html>
<head>
</head>
<body>
<h1>Header or logo of the website</h1>
</body
</html
```

</td><td> 

Figure 3: Output of f1.html</td></tr>
<tr><td>F2.html</td><td></td></tr>
<tr><td>

```
<html>
<head>
</head>
<body>
<a href = "f3.html" target = "f3">Introduction </a>
<br>
<a href = "f5.html" target = "f3">Profile </a>
</body
</html>
```

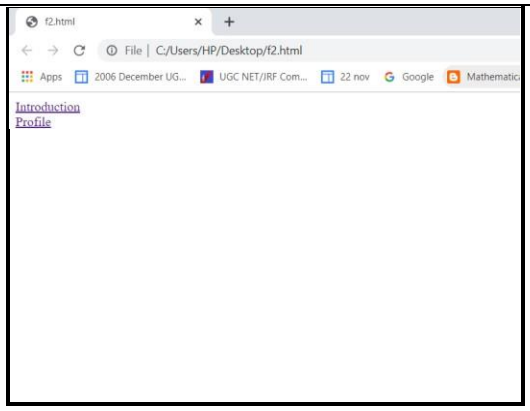
</td><td> 

Figure 4: Output of f2.html</td></tr>
<tr><td>F3.html</td><td></td></tr>
<tr><td>

```
<html>
<head>
</head>
<body>
<h1>Introduction </h1>
</body
</html
```

</td><td> 

Figure 5: Output of f3.html</td></tr>
<tr><td>F4.html</td><td></td></tr>
</table>

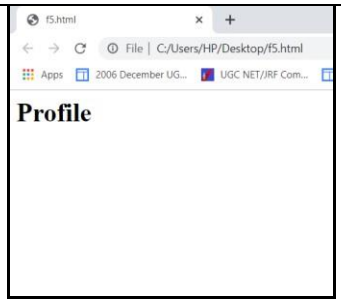
<html> <head> </head> <body> Footer of the web site </body> </html>	 **Figure 6: Output of f4.html**
F5.html	
<html> <head> </head> <body> Profile </body> </html>	 **Figure 7: Output of f5.html**

Table 7: Example of Frameset

Check Your Progress 2

1. State Weather True or false:
   i. The frame cannot be resized by default.
   ii. <FRAME> enables the web page to be divided into different regions.
   iii. The scrolling attribute enables the visibility of only the horizontal scrollbar.
   iv. URL contains the address of the document to be loaded in the frame.
2. Create two vertical frames with 30%, *. The first frame contains the links of the company, profile, contact information, etc. And the second frame will display the document whose link is clicked
3. Write the code to design the frame of the following figure:

Figure 7: Design Frames

Objective Type questions:

13. What is the correct way to create a list by using the lowercase letters?

e) <ol type = "a">
f) <ul type = "a">
g) <ol alpha = "a">
h) None of the above.

14. Which is the correct method to start an ordered list that starts with the number 4?

e) <ol type = "1" start = 1>
f) <ol type = "1" begin = 1>
g) <ol type = "1" start = 4>
h) <ol type = "1" begin = 4>

15. The tag that can be used to add power in the following example $(a+b)^2$.

e) <sub>
f) <sup>
g) <p>
h) None of the above

16. The correct way to apply the font tag in HTML is:

a) Text
b) Text
c) Text
d) None of the above

17. Gif and jpeg are the formats of which kind of files?

e) Images
f) Audio
g) Video
h) None of the above

18. The tags used to create checkbox in the HTML are:

e) <input name = "checkbox">
f) <checkbox> Text </checkbox>
g) <input type = "checkbox">
h) None of the above

19. DD stands for

e) Data Description
f) Data Definition
g) Detailed Data

h) None of the above

20. Which tag is used to align the text center in the table?

e) <tdalign>
f) <td valign = "center">
g) <td align = "center">
h) <tr valign = "center">

21. The attributes of the <frameset> tag include

e) Rows
f) Cols
g) Both a and b
h) None of the above

22. The attribute used to create two vertical frames are

e) cols
f) rows
g) both a and b
h) None of the above

23. The attribute used to force all the web links to open in the target frame is
e) Frame and iframe
f) Frame and target
g) Base and target
h) Base and frame

24. What is the use of iframe in HTML?

e) It displays a web page within another web page.
f) It displays a web page with animation effects.
g) It displays a web page without a browser.
h) All of the above.

Hands on Practice

Example 1:

```
<html>
<head>
<title> Example </title>
</head>
<body>
<h2>Horizontal List</h2>
```

```
<ul id = "menu">
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
  <li>PHP</li>
</ul>
</body>
</html>
```

Example 2:

```
<html>
<head>
<title> Example </title>
</head>
<body>

<dl>
  <dt>Aries</dt>
  <dd>-One of the 12 horoscope sign.</dd>
  <dt>Bingo</dt>
 <dd>-One of my evening snacks</dd>
 <dt>Leo</dt>
 <dd>-It is also an one of the 12 horoscope sign.</dd>
  <dt>Oracle</dt>
  <dd> It is a multinational technology corporation.</dd>
</dl>
</body>
</html>
```

Example 3:

```
<html>
<head>
<title> Example </title>
</head>

<body>
   <p> List of Indian States with their capital </p>
<ol>
  <li> Delhi
      <ul>
        <li> NewDelhi </li>
      </ul>
   </li>
   <li> Haryana
      <ul>
          <li> Chandigarh </li>
      </ul>
   </li>
```

```
   <li> Gujarat
      <ul>
         <li> Gandhinagar </li>
      </ul>
   </li>
   <li> Rajasthan
      <ul>
         <li> Jaipur </li>
      </ul>
   </li>
   <li> Maharashtra
      <ul>
         <li> Mumbai </li>
      </ul>
   </li>
   <li> Uttarpradesh
     <ul>
         <li> Lucknow </li></ul>
   </li>
</ol>
</body> </html>
```

Example 4 :

```
<!DOCTYPE html>
<html>
  <head>
    <title> HTML Frames </title>
  </head>
  <frameset rows = " 10% , 80% , 10% " >
    <frame name = "top" src = "top_frame.html" />
    <frame name = "main" src = “main_frame.html" />
    <frame name = "bottom" src = “bottom_frame.html" />
   <noframes>
       <body> Your browser does not support frames. </body>
    </noframes>
</frameset>
</html>
```

Example 5:

```
<!DOCTYPE html>
<html>
  <head>
    <title> HTML Frames </title>
  </head>

  <frameset cols = " 25% , 50% , 25% " >
    <frame name = "left" src = “ top_frame.htm" />
    <frame name = "center" src = "main_frame.htm" />
```

```
    <frame name = "right" src = " bottom_frame.htm" />
    <noframes>
      <body> Your browser does not support frames. </body>
    </noframes>
  </frameset>
 </html>
```

Example 6:

```
<!DOCTYPE html>
<html>
  <head>
    <title> HTML Target Frames </title>
  </head>
  <frameset cols = " 200 ,  * " >
    <frame src = " / html / menu . html " name = "menu_page" />
    <frame src = " /html / main . html " name = "main_page" />
     <noframes>
      <body> Your browser does not support frames. </body>
    </noframes>
  </frameset>
</html>
```

5.4 SUMMARY

This unit helps in designing an attractive website and the data is represented in the form of lists, frames are used etc. List tag helps the designer to design web pages with primary data. The list can be categorized into three types: Ordered, Unordered, and Definition. Ordered list helps in creating an order between different elements in the sequence and in Unordered list, there is no sequence in the number of elements. Frames can also be created for dividing the browser window, to display more than one document on a screen.

UNIT 6 STYLE SHEETS

Structure

6.0 INTRODUCTION

Styles sheets are the most powerful mechanism in providing the styles to the web pages. It helps in making a standard or uniformity throughout the website. The styles sheets provide a method where properties, text, image formatting can be applied on the web page. The purpose of the style sheet is to give look and feel to the website using different properties. The tags given in the HTML document are pinned to the styles. The benefits are to create a global change in the website i.e., on all the pages from a single location. If any changes need to be performed, this can be done in one place only. This helps in generating the rules on the HTML document that specify the content appearing on the browser. For example, all the pages of the website contain a background in cream color, all the paragraphs are given using Times New Roman font face, and all the headings follow the same pattern on all the pages. In this way, CSS helps in creating

uniformity throughout the website.

6.1 OBJECTIVES

Once we know how to write the HTML code, the next step is the use of CSS which helps in giving styles to the content or images on a web page. This topic will cover the following in detail:

- Introduction to CSS
- The rules to write CSS code
- The working of CSS code in HTML document.
- The properties of different elements of CSS.

The remaining topics in this section will cover the various CSS properties that can be used for designing attractive pages.

6.2 Introduction to CSS

Cascading style sheet (CSS) describes how web pages are presented on the screen. A cascading style sheet is a code that helps to change the content of an individual page or style represented on the entire screen. W3C has actively announced the use of style sheets on the web page in 1994.

CSS controls the styles of a web page. Cascading style sheet (CSS) provides the easiest and effective way to represent various properties of the HTML tags. In CSS, each property has a name and value separated by a colon (:) and if a designer uses more than one property in the same line then the code is separated by (,) and each property declaration is separated by a semicolon (;) at the end of code.

This is a simple language used for designing web pages and makes them more attractive and presentable. This is to be used by the designers and the following are the features of CSS that must be required to create a website:

- **Designing Skills:** The web designer must have the skills of creating a website using HTML and CSS. With the knowledge of these languages, anyone can design a website.

- **Creating Astonishing sites**: CSS gives the websites a look by using different colors of text, styles of fonts, layouts, background colors, images, etc. for the different display devices and screen sizes.

- **Easy to Understand:** CSS is easy to understand, it provides control over the content in HTML documents. It is combined with HTML or XHTML i.e.; It works with the markup languages.

- **Language Understanding:** With the help of HTML and CSS, anyone can understand the related languages which become easy to learn and understand such as PHP, ASP.NET, JavaScript, etc.

  HTML 4.0 provides the support for the following features:

- **Flexible Placement of style information:** By placing the style sheets in a separate files, makes it easy to reuse. Sometimes it is useful to include the rendering instructions in the document to which they apply, either added in the attributes of the element or at the start of the document.

- **Independence from specific language:** This specification does not tie HTML document to any particular style sheet language.

- **Cascading:** This is the capability provided by the style sheet like CSS that allows styles from several sources to be blended together. In this the styles can be given to the group of documents or it can be given to the any specific document. By storing thesestyles sheets separately, makes it reusable. Not all the style sheet languages support cascading feature.

- **Alternate Styles:** The document can be viewed by different ways. For example, a style sheet for rendering documents with small font size or the one that specifies larger font size for increased legibility. This specification allows the authors to specify a preferred style sheet as well as alternate styles that target specific users or media.

6.3 CSS Advantages

CSS is one of the most widely adopted languages used for giving styles on web pages. The advantages of CSS are as follows:

- **Time-saving:** CSS code can be written once and can be used multiple times in HTML documents. For example, the styles created for the paragraph can be used in multiple paragraphs.

- **Maintenance is easy:** If the designer needs to change the style, then it can be changed in one place and in the output, its effect applies to all tags that use a particular style.

- **Global standards for Web:** HTML tags are obsolete and the designers are focusing on CSS. Thus, using CSS within the HTML pages makes the pages compatible with the forthcoming browsers.

- **Compatibility:** CSS code can be compatible with all the devices i.e., one document of HTML can be accessed by a different kind of device like PDA, Mobiles, or Systems.

- **Fast Loading of Pages:** It contains less coding, as the code for CSS is written once and can be used multiple times on the HTML pages. This makes for faster loading of web pages.

- **Style sheets can be reusable:** Once the styles can be created by the designer, the style sheet can be used on any HTML document. This ensures that the web pages must have uniform appearance when they are displayed on the browser. For example, if we have more number of web pages and we have to define same style in all the pages, then this can be possible only with the help of CSS.

- **One Time Effort:** The style sheet can be created once and this sheet can be linked several times in the web pages. The most important thing in style sheets if any modification needs to be performed, it can be done at only one place and the changes are reflected on all the web pages.

- **Relative measurement:** By using the relative measurement in style sheet, the web page looks effective and interactive on the any screen with any resolution.

6.4 CSS Syntax

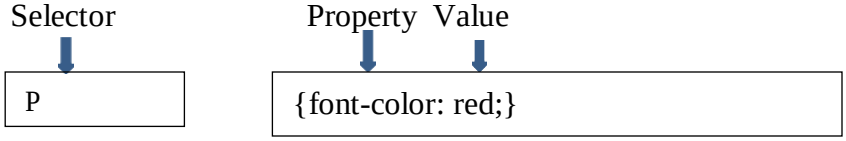
The rules must be followed for creating the styles which are interpreted by the browser. It can be applied to the specific element in the document. Syntax to create CSS styles consists of three parts:

1. **Selector**: It is an HTML tag on which the style can be applied. Each rule set must be wrapped in curly brackets {}. The tags can be like <h1> or < p> etc.

2. **Property:** A property is a rule of the attribute of an HTML tag. Put all the attributes in the curly brackets with the name and the value of the attribute and end with the semicolon (;). They could be color, border, etc.

3. **Value:** After the colon (:) the value of the attribute is defined and semicolon (;) used to separate each declaration from the next one. For example, the color property can be set to the value either Red or #FF0000, etc.

For Example:



Here, the P selector is used for the paragraph. And the property of font color can be set to the value red.

Another example of CSS:

p

{
font-family: Arial;
font-size:12px;
}

Explanation of the Syntax:
1. The selector is the HTML element which designer can change according to the style.
2. The declaration block contains a property and values separated by a colon (:).
3. The declaration block contains one or more declarations in a single line and it can be separated by a semicolon (;). Additionally, declaration blocks are contained in between the curly brackets ({}).

6.5 Types of Selectors

In CSS, there are many different types of selectors. Here, some common types of selectors are given in Table 1.

Selector	Description	Syntax	Example

Elements Selector	All HTML element of the specified type.	P {
		property: value;
		}
			P {
			text-align: center;
			color: green;
			}

ID	This selector is specified with a particular element and represent with a hash sign (#).	*#id_value {*
		style properties;
		}
			#id_name
			{
			text-align: center;
			color: green;
			}

Class	This selector is specified with the whole web page and it will be used on multiple instances on the same page. The class selector is represented with a dot(.) sign.	*.class {*
		style properties;
		}
			.class_name
			{
			text-align: right;
			color: green;
			}

Attribute Selector	The elements on the page with the specified attribute.	a[target] {
		background-color: yellow;
		}
			<head><style>
			input[type = text] {
			width: 150px;
			display: block;
			margin-bottom: 10px;
			background-color: yellow;
			}
			</style></head>
			<body>
			<form name = "input" action =

			"" method = "get"> Firstname:<input type = "text" name = "Name" value = "Peter" size = "20"> </form></body>

Pseudo Class	This selector is used with specified elements but only when the specified state such as when a user moves the mouse over the text, visited and unvisited link differently, and when a user gets focus on the text.	selector: pseudo-class {
		property: value;
		}
			p::first-line {
			color: blue;
			text-transform: uppercase;
			}

Universal Selector	The universal selector (*) selects all the HTML elements on the page.	*{style properties}	*{
			text-align: right;
			color: red;
			}

Grouping Selector	The grouping selector selects the entire HTML element with the same style in a single declaration. CSS provides the best way to represent the same style in a single group selector but it can be separated by comma (,).	element, element
		{
		style properties;
		}
			h1 {
			text-align: align;
			color: green;
			}
			
			h2{
			text-align: left;
			color: green;
			}
			h1, h2
			{
			text-align: left;
			color: green;
			}

Table 1: Types of Selectors

Handy Tips for the selectors:

Example of the ID selector	Output

```
<html>
<head>
<title> selector </title>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>
<p>Web Designing</p>
<p id = "a">This is use of ID
selector.</p>
<p>CSS (Cascading Style Sheet)</p>
</body>
</html>
```

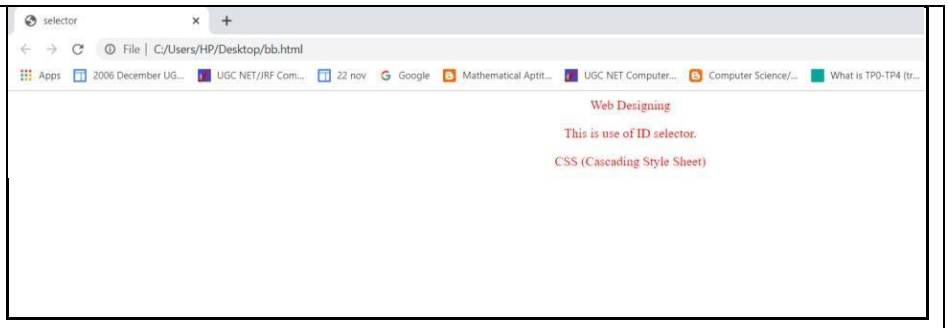


Figure 1: Output of CSS document using ID

Another Example of ID Attribute

```
<html>
<head>
<style>
#mystyle{
  background-color: lightgreen;
  color: black;
  padding: 30px;
  text-align: center;
}
</style>
</head>
<body>

<h2>The id Attribute</h2>
<p>Use CSS to style an element by
using the id</p>

<h1 id = "mystyle">Style Used In
CSS</h1>

</body>
</html>
```

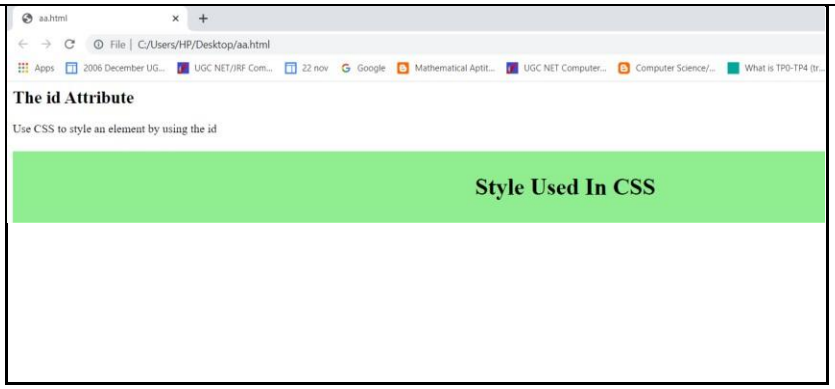


Figure 2: Output of CSS document using ID

Example of Class Selector | **Output**

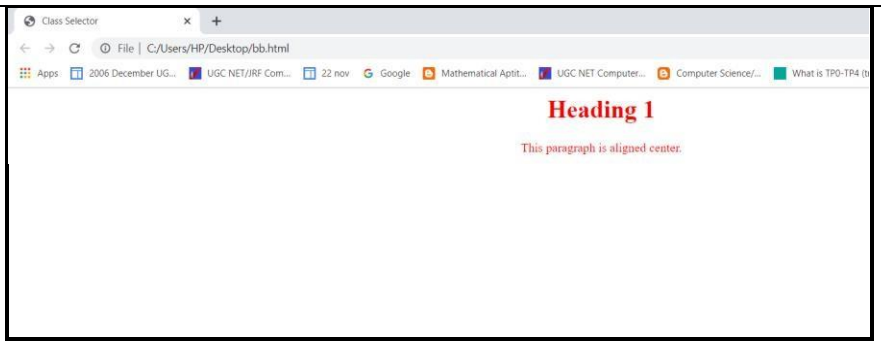
<html>
<head>
<title> Class Selector </title>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>
 	
<h1 class = "center">Heading 1</h1>
 This paragraph is aligned center.
 	
</body>
</html>
	
	
	Figure 3: Output of CSS document using class

Table 2: Example of Selectors

6.6 CSS Comments

CSS comments are useful for describing the code and they can be placed inside the style element start with /* and ends with */. Comments are usually ignored by the browser. A web designer can add comments where ever they want in the code. Comments can also be useful for single lines as well as multi lines. Due to these above elements commented lines will not be displayed on the browser window.

The syntax for single line and multiple lines:
/* This is a single line as well as used for multiple lines comment */

6.7 CSS Inclusion

Styles can be represented in four ways in HTML documents. The most common ways of using CSS are Embedded, Inline, and External CSS.

6.7.1. Embedded CSS: In this, the CSS rules can be added in the <style> tag. This tag is given in between the <HEAD> and </HEAD> tags. These rules are applied according to the tags of HTML in the body section. It should be applied only within the document. The style tag contains two attributes.

Attributes	Description	Syntax	Examples
TYPE	It specifies the language to be used i.e., CSS.	<style type = "text/css">	<style type = "text/css">
MEDIA	It specifies the medium on which the document will be displayed. By default, its value is all.	<style type = "text/css" media = "all">	<style type = "text/css" media = "all"> Or media value can be a screen, tv, projection, print, etc.

Table 3: Attributes of <style> tag

Advantages of Embedded Style Sheets:

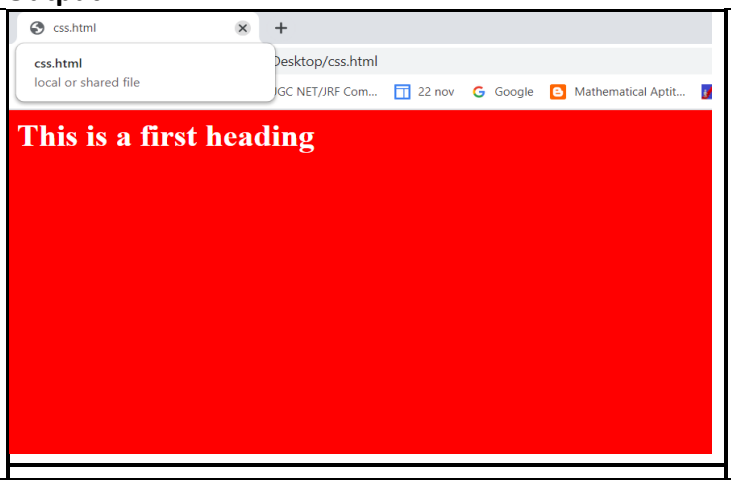
1. It controls the styles of a document in one place.
2. No extra file needs to be created or downloaded for the style information.

Disadvantages of Embedded Style Sheets:

1. If the same style is to be applied to another element, then it needs to be written again.

Handy Tips for the embedded CSS

Example of Embedded CSS	Output

<html>
<head>
<style type = "text/css">
body {
background-color: RED;
}
h1 {
color: WHITE;
}
</style>
</head>
<body>
	

This is a first heading </body> </html>	**Figure 4: output**

Table 4: Example of embedded CSS

6.7.2. Inline CSS: This style can be applied to the tag itself. Syntax to create Inline style is as follows:

<element style = "style rules">
For example:
<h1 style = "font-color: red;">

Advantages of Inline Style Sheets:

1. It controls the style of only a single element.
2. It overrides external or document styles.

Disadvantages of Inline Style Sheets:

1. The style needs to be applied again, throughout the document.
2. It is bound to HTML elements; thus, it is difficult to update the styles.

Handy Tips for the Inline CSS

Example of Inline CSS	Output

```
<html>
   <head>
   <title>Inline CSS</title>
   </head>
<body>
<h1 style = "color:blue; text-
   align:left;"> Cascading style sheet
   (CSS) </h1>
<p style = "color:red;"> This is used to
   give styles to the web pages.
</p>
</body>
</html>
```

Figure 5: Inline CSS

Table 5: Example of inline CSS

6.7.3. External CSS: An external style sheet can be created as a separate file by saving the file with a .css extension. All the rules can be defined in this file and after that, this file will be included in the HTML document by using the <link> tag.

Syntax of <link> tag:

```
<html>
<head>
<link type = "text/css" href = "URL">
</head>
</html>
```

Let us consider an example to create a style sheet that is saved with the name sample.css.

```
h1, h2, h3 {
  color: #0000FF;
font-family: serif;
  }
```

Now, this style sheet can be added to the HTML document using the <link> tag.

```
<head>
  <link type = "text/css" href = "sample.css" />
</head>
```

Handy Tips for the External CSS

Example of External CSS	Output

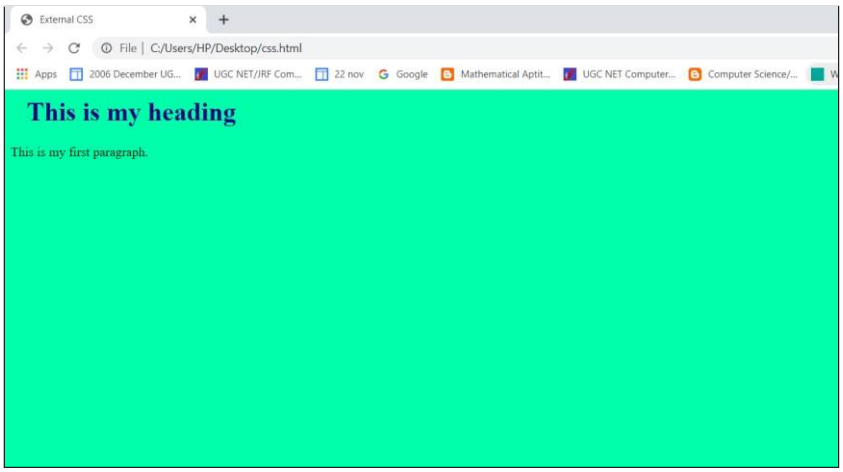
This is an external style sheet that is saved with mystyle.css.
 	
body {
background-color: #00ffaa;
}
 	
h1 {
color: navy;
margin-left: 20px;
}
Code of HTML file:
<html>
<head>
<link rel = "stylesheet" href = "mystyle.css">
<title>External CSS</title>
</head>
<body>
 This is my heading
 This is my first paragraph.
</body>
</html>
	
	Figure 6: External CSS

Table 6: Example of External CSS

Advantages of External Style Sheets:

1. It controls the style of multiple documents by creating one external style sheet.

Disadvantages of External Style Sheets:

1. It needs extra time to download the style sheets.

6.7.4. Imported CSS: To import the CSS file similar to the external style sheet with the help of @import.

Syntax of @import is given as follows:

```
<head>
  @import "URL"; | @import url("URL");
</head>
```

6.8 CSS Colors

The colors can be specified in CSS using different methods such as RGB, Hexadecimal codes, or Keyword. These are used to give background color or text color. It can also be used for giving border color and other effects on the web page.

Format	Syntax	Example
Hexadecimal code	#RRGGBB	p{color:#0000FF;}
RGB values	rgb (rrr, ggg,bbb)	p{color:rgb(0,0,255);}
Keywords	Red, black, blue, etc.	P{color:blue;}

Table 7: Colors in CSS

1. **Hexadecimal codes:** It is a 6-digit code that represents the color in CSS or HTML. The first two digits represent the red color, the middle two digits represent the green color and the last two digits represent the blue color.
   This color can also be generated using different software. One example is to create hexadecimal code is using paint and a calculator. The steps to generate the color are:
   1. Open the color palette in paint.
   2. Here, the RGB value is given. Select the color that is required for the webpage as shown in Figure 6.

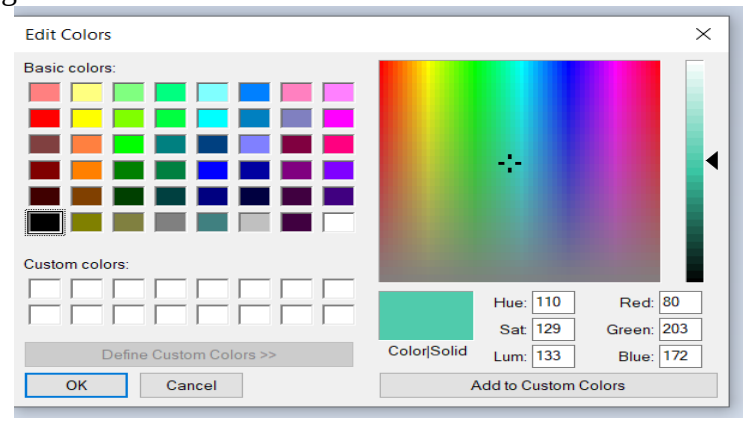


Figure 7: Color palette in paint

   3. Open programmer calculator, that can convert this RGB decimal format into hexadecimal format.
4. Enter the decimal value of RGB given in paint to the calculator one by one and write the hexadecimal code in sequence,
   For example: For RED decimal value is 80 and the hexadecimal value is 50.
   For GREEN decimal value is 203 and the hexadecimal value is CB.
   For BLUE decimal value is 172 and the hexadecimal value is AC.
5. After getting the hexadecimal codes of RGB, put these values in the sequence such as #50CBAC. This code is similar to RGB value (80,203,172).

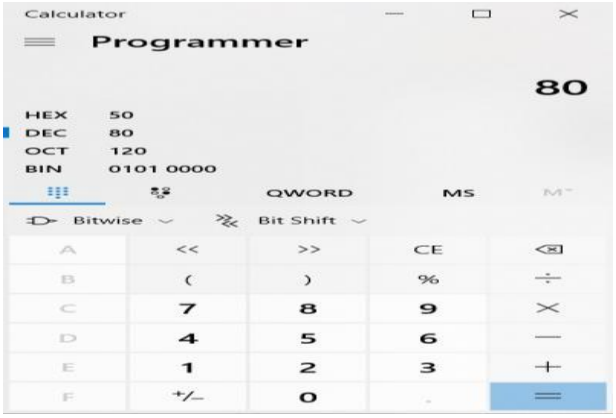


Figure 8: Calculator to convert Decimal to Hexadecimal

2. **RGB Values:** The color property can also be specified using the RGB values. It can take three values, one for red, the second for green, and the third for blue color. The value can be given in the range from 0 to 255 i.e., a total of 256 colors are there.
   For Example: RGB (0, 0, 0) is used for black color.

3. **Keywords:** These are used to give simple names of the color without any code like Red, green, blue, aqua, etc.

Handy Tips for the CSS Colors

Example of Colors	Output

```
<html>
<head>
<title> CSS colors </title>
</head>
<body>
<h1 style = "border: 2px red; background-color:rgb(255, 99, 71);">Cascading Style Sheet</h1>
<p style = "border: 2px solid blue;">This is my first paragraph. </p>
<p style = "border: 2px solid Violet;">This is my second paragraph. </p>
</body></html>
```

Figure 9: CSS Colors

Example of CSS

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML CSS</title>
  </head>

  <body>
    <p> <font color = "green" size = "5"> Example of CSS  </font> </p>
  </body>

</html>
```

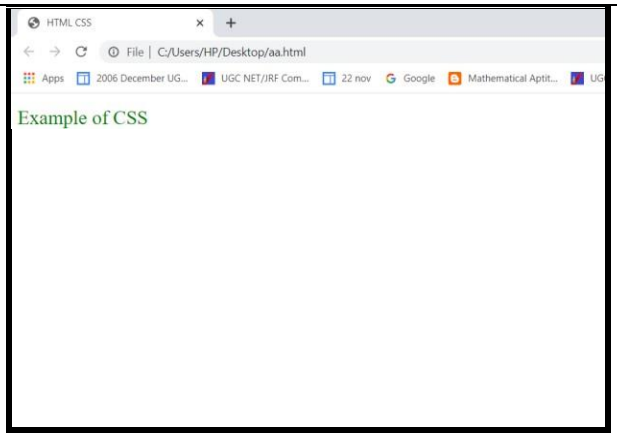


Figure 10: CSS font Colors

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Inline CSS</title>
  </head>

  <body>
    <p style  =  "color:red;">This is red</p>
    <p style  =  "font-size:20px;">This is thick</p>
    <p style  =  "color:green;">This is green</p>
    <p style = "color:green;font-size:20px;">This is thick and green</p>
  </body>

</html>
```

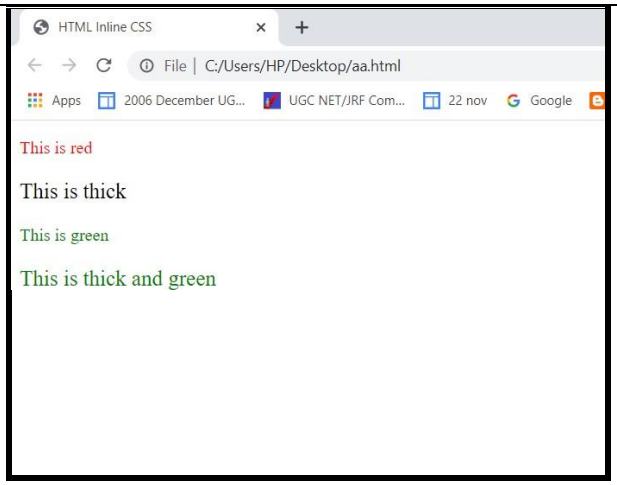


Figure 11: CSS Inline Example

Table 8: Example of CSS Colors

Check Your Progress

1. State whether True or False:
   i. An external style sheet can be saved using a .html file.
   ii. Text-color is used to set the font color.
   iii. The style tag can also be used within <BODY> and </BODY> tags.
2. What are the different ways of including CSS in a web page?
3. What do you mean by CSS and explain the advantages of CSS compared to HTML?
4. Explain the types of selectors in CSS.
5. Explain in detail the properties of CSS.
6. Why is external style sheet benefitial?
7. What is the use of embedded style sheet?

Objective Type questions:

25. Full form of CSS.

i) Cascading Style sheet
j) Color style sheet
k) Configuration style sheet
l) None of the above.

26. Which is the correct syntax to implement external CSS?

i) <style src = a.css>
j) <style src = "a.css">
k) <link rel = "stylesheet" type = "text/css" href = "mystyle.css">
l) <stylesheet> mystyle.css </stylesheet>

27. To change the background color of a document which property is used in the CSS.

i) bgcolor
j) background-color
k) color
l) all of the above

28. To change the text color which property is used in the CSS.

e) bgcolor
f) background-color
g) color
h) all of the above

29. The CSS property used to make the text bold is

a) Font-weight : bold
b) Weight : bold
c) Font : bold
d) Style : bold

30. The CSS property used to specify the transparency of an element is

a) Opacity property
b) Filter property
c) Visibility property
d) Overlay property

31. The CSS property used to specify whether the text is written in the horizontal or vertical direction?

a) Writing-mode
b) Word-break
c) text-indent
d) None of the above

Hands on Practice:

Example 1:

```
<html>
<head> <title> CSS example </title>
</head>
<body>
<div class = "section">
  <label for = "guestname" class = "field-label">Please Enter Your Name</label>
  <label for = "guestname" class = "field prepend-icon">
    <input type = "text" name = "guestname" id = "guestname" class = "gui-input" required = ""
placeholder = "John Doe/Jane Doe">
    <span class = "field-icon"><i class = "fa fa-user"></i></span>
  </label>
</div>

<div class = "frm-row">
  <div class = "section colm colm6">
    <label for = "guestemail" class = "field-label">Email Address</label>
    <label for = "guestemail" class = "field prepend-icon">
      <input type = "email" name="guestemail" id="guestemail" class="gui-input" required=""
placeholder="john@something.com">
      <span class="field-icon"><i class="fa fa-envelope"></i></span>
    </label>
  </div>

  <div class="section colm colm6">
    <label for="guestelephone" class="field-label">Telephone / Mobile</label>
    <label for="guestelephone" class="field prepend-icon">
      <input type="text" name="guestelephone" id="guestelephone" class="gui-input" required=""
placeholder="Telephone / Moble Number">
```

```
      <span class="field-icon"><i class="fa fa-phone-square"></i></span>
    </label>
  </div>
</div>

<div class="frm-row">
  <div class="section colm colm6">
    <label for="adults" class="field-label">Number of Adults</label>
    <label for="adults" class="field prepend-icon">
      <input type="number" id="adults" name="adults" class="gui-input" required=""
placeholder="Number of adults">
      <span class="field-icon"><i class="fa fa-users"></i></span>
    </label>
  </div>

  <div class="section colm colm6">
    <label for="children" class="field-label">Number of Children</label>
    <label for="children" class="field prepend-icon">
      <input type="number" id="children" name="children" class="gui-input" required=""
placeholder="Number of children">
      <span class="field-icon"><i class="fa fa-users"></i></span>
    </label>
  </div>
</div>

<div class="frm-row">
  <div class="section colm colm6">
    <label for="checkin" class="field-label">Check-in Date</label>
    <label for="checkin" class="field prepend-icon">
      <input type="text" id="checkin" name="checkin" class="gui-input" required=""
placeholder="mm/dd/yyyy">
      <span class="field-icon"><i class="fa fa-calendar"></i></span>
    </label>
  </div>

  <div class="section colm colm6">
    <label for="checkout" class="field-label">Check-out Date</label>
    <label for="checkout" class="field prepend-icon">
      <input type="text" id="checkout" name="checkout" class="gui-input" required=""
placeholder="mm/dd/yyyy">
      <span class="field-icon"><i class="fa fa-calendar"></i></span>
    </label>
  </div>
</div>

<div class="spacer-t20 spacer-b30">
```

```
  <div class="tagline"><span>Please answer these questions for a pleasant stay</span></div>
</div>

<div class="frm-row">
  <div class="option-group field">

    <div class="section colm colm6">
      <label class="switch">
        <input type="checkbox" name="switch1" id="switch1" value="switch1">
        <span class="switch-label" data-on="YES" data-off="NO"></span>
        <span>Will you be bringing a pet?</span>
      </label>
    </div>

    <div class="section colm colm6">
      <label class="switch">
        <input type="checkbox" name="switch2" id="switch2" value="switch2">
        <span class="switch-label" data-on="YES" data-off="NO"></span>
        <span>Do you need us to pick you up?</span>
      </label>
    </div>

  </div>
</div>

<div class="section">
  <label for="comment" class="field-label">Anything else we should know about?</label>
  <label for="comment" class="field prepend-icon">
    <textarea class="gui-textarea" id="comment" name="comment" placeholder="Let us know
about any special accommodation needs"></textarea>
    <span class="field-icon"><i class="fa fa-comments"></i></span>
    <span class="input-hint">
      <strong>Please:</strong> Be as descriptive as possible
    </span>
  </label>
</div>
</body>
</html>
```

Example 2:

```
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial, Helvetica, sans-serif;}
```

```
* {box-sizing: border-box;}

input[type=text], select, textarea {
  width: 100%;
  padding: 12px;
  border: 1px solid #ccc;
  border-radius: 4px;
  box-sizing: border-box;
  margin-top: 6px;
  margin-bottom: 16px;
  resize: vertical;
}

input[type=submit] {
  background-color: #04AA6D;
  color: white;
  padding: 12px 20px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

input[type=submit]:hover { background-color: #45a049;}

.container {
  border-radius: 5px;
  background-color: #f2f2f2;
  padding: 20px;
}
</style>
</head>
<body>

<h3>Contact Form</h3>

<div class="container">
  <form action="/action_page.php">
    <label for="fname">First Name</label>
    <input type="text" id="fname" name="firstname" placeholder="Your name..">

    <label for="lname">Last Name</label>
    <input type="text" id="lname" name="lastname" placeholder="Your last name..">

    <label for="country">Country</label>
    <select id="country" name="country">
      <option value="australia">Australia</option>
```

```
    <option value="canada">Canada</option>
    <option value="usa">USA</option>
  </select>

  <label for="subject">Subject</label>
  <textarea id="subject" name="subject" placeholder="Write something.." style="height:200px"></textarea>
  <input type="submit" value="Submit">
 </form>
</div>
</body>
</html>
```

6.9 SUMMARY

This unit covers the basics of CSS that provides a look and feel to the website and makes it attractive. CSS considers each element of HTML and creates the rules for its appearance. Rules are created with the help of selectors that indicates the HTML tag on which the settings can be applied and the declarations indicate what property should be applied. This unit includes the different selectors that can be used to apply styles to different elements in HTML. The styles can be given in four ways i.e., Embedded, Inline, Externally, Export. The benefit of CSS is to create the styles once and use them on multiple tags and multiple pages.

UNIT 7 ATTRIBUTES OF STYLE SHEETS

Structure **Page No**

7.9 Summary

7.0 INTRODUCTION

The previous unit describes the basic properties of CSS that helps in creating an efficient website. The basic purpose of this unit is to cover the advanced attributes of CSS such as background, font, text, margins, positioning, etc. These properties help in presenting a look and feel to the website. The benefits of CSS are to make a global change in the website i.e., on all the pages from a single location. CSS allows to create the rules that specifies how the content should appear on the screen.

CSS treats each element of HTML as it appears inside a box that uses different rules that indicates how the element looks on the browser. Rules can be created using different methods such as selectors and declarations. There exist different types of selectors and declarations consists of two parts: where the first part specifies the properties and the second part represents the value of that property. There exist three ways to include colors in CSS that is RGB values , hexadecimal codes, color names. The designer must ensure that there must exist enough contrast between the background color and the font color. CSS3 introduced extra color value for RGB that indicates the opacity. It is also known as RGBA.

7.1 OBJECTIVES

In the previous Unit, the beginner can understand how to embbed CSS in HTML document. And this Unit properties of different HTML tags are given.

- The properties of different elements of CSS for font and text.
- Padding
- Margins
- Color, background color
- List
- Positioning in CSS

7.2 CSS Background

CSS background helps the Web designer to set the background colors and background image. There exist different attributes of background that can be applied on a web page.

Property	Description	Syntax	Example

background - color	It defines an element background colors.	background - color : color ;	body {
			background - color : coral ;
			}

background - position	It defines the origin of a background image.	background - position : value ;	background - position : center top ; background - position : 50px 150px ; background - position : 50% 50% ;
background - image	It defines an element s element background image.	background - image : url ;	background - image : url ("gallery.gif") ;
background - repeat	It specifies how the background image is tiled and there are three different ways of defining position.	background - repeat : repeat | repeat - x | repeat - y | no - repeat ;	background - repeat : repeat - y ; background - repeat : no - repeat ;
background - size	It specifies is a percentage, length and keyword size of the background image.	background - size : auto | length ;	background - size : auto ; background - size : 100% 100% ;

Table 1 : CSS Background Properties

Handy Tips for the CSS Colors

Example of Background Color	Output
<html> <head> <title> CSS background color </title> <style> h1 { background - color : pink ;	**Figure 1 : CSS Background Colors**

}
 	
p
{
background - color : RED ;
}
</style>
</head>
<body>
  HTML and CSS
 	
This paragraph contains background color.
 	
</body>
</html>	

Example of background - image, and repeat	**Output**

<html>
<head>
<style>
body {
background – image : url ("logo.png") ;
background – repeat : repeat - x ;
}
</style>
</head>
<body>
</body>
</html>
	Figure 2 : CSS Background Image in x - direction

Table 2 : Example of CSS Background - Color and background - image with the repeat property

7.3 CSS Font

CSS provides the designer a great control of their text, which is displayed on the browser window. CSS changes the text size, color, style, and much more.

Property	Description	Syntax	Example
Color	CSS font color changes the color of the	color : color | initial | inherit	Color : red ;

	text by using color - name, RGB, hexadecimal color code combinations.	;	
font - family	CSS font families can be divided into two categories serif and scan serif. A serif font does include small lines on the other hand scan - serif font does not include the small lines on the end of the characters and it can much easier to read on a computer monitor than serif style.	font – family : family – name | *generic – family* | initial | inherit ;	font - family : serif ;
font - size	CSS font size can change the size of the font by using values, percentages, and key terms.	font - size : medium | xx – small | x – small | small | large | x – large | xx – large | smaller | larger | length | initial | inherit ;	font - size : 15px ; font - size : large ; font - size : 150% ;
font - style	CSS font style can change the style of font and it will be bold, italic, or not.	font - style : normal | italic | oblique | initial | inherit ;	font - style : normal ; font - style : italic ;
font - weight	CSS font - weight can specify the weight of the text and it ranges from 100(thin) to - 900(thick) size.	font - weight : normal | bold | bolder | lighter | number | initial | inherit ;	font - weight : bold ; font - weight : 900 ;
font - variant	CSS font - variant allows a designer to convert their font to all small caps.	font - variant : normal | small – caps | initial | inherit ;	font - variant : small - caps ;

Table 3 : CSS Font Properties

Handy Tips for the CSS Font

Example of Font	Output

<html>
<head>
<style>
.p1 {
font - family : "Times New Roman" ;
font - size : xx - large ;
font - weight : bold ;
}
	Figure 3 : CSS Font

.p2 {
font - family : Arial ;
font - size : 20 px ;
font - style : italic ;
}
</style>
</head>
<body>
  CSS font – family
 This is the first paragraph given in the Times New Roman font.
 This is the second paragraph given in the Arial font .
 This is a third paragraph that uses the styles of first paragraph.
</body>
</html>	

Table 4 : Example of CSS Font

7.4 CSS Text

Text property can be used to format the text given on the web page by using text - align, text - transformation, color properties, text - indent, etc. It helps to format paragraphs by text - indent, text - align, word - spacing, etc.

Property	Description	Syntax	Example
text - decoration	This property is used to underline, overline or strikethrough the text in the browser.	text - decoration : *text - decoration - line text - decoration - color text - decoration – style* | initial | inherit ;	text - decoration : underline ; text - decoration : underline overline ;
text - align	Web designers can set the text in left, right, center, or justified format. Justified is used to set the equal width and set the left and right margins in a straightway. It can especially, be used to align the text in magazines.	text - align : left | center | right ;	text - align : left ; text - align : center ;
vertical - align	Vertical alignment of an element.	vertical - align : baseline | length | sub | super | top | text – top | middle | bottom | text – bottom | initial | inherit ;	vertical - align : text - top ;

text - transform	It converts the text in lower - or upper - case letters.	text - transform : none | capitalize | uppercase | lowercase | initial | inherit ;	text - transform : uppercase ; text - transform : capitalize ;

word - spacing	It is used to set the spacing between words and it should be defined with exact values.	word - spacing : normal | length | initial | inherit ;	p {
			word - spacing : 30px ;
			}

letter - spacing	It is used to set the spacing between the characters.	letter - spacing : normal | length | initial | inherit ;	h1 {
			letter - spacing : 3px ;
			}

text - shadow	This property adds a shadow to the text. It is a very simple process just specifying the horizontal shadow (2px) and the vertical shadow (2px).	text - shadow : h - shadow v - shadow blur - radius color | none | initial | inherit ;	h1 {
			text - shadow : 2px 2px #ff0000 ;
			}

Table 5 : CSS Text Properties

Handy Tips for the CSS Text

Example of Text	Output

<html>
<head>
<title> CSS Text properties </title>
<style>
h1 {
text - decoration : overline ;
text - transform : uppercase ;
letter - spacing : 3px ;
text - shadow : 2px 2px red ;
}
 	
h2 {
text - decoration : line - through ;
}
h3 {
text - decoration : underline ;
text - transform : capitalize ;
	Figure 4 : CSS Text

} </style> </head> <body> This is first heading This is second heading This is third heading </body> </html>	

Table 6 : Example of CSS Text

7.5 CSS Margins

This property is used to create space all over the elements or it can set the margin on all four sides of an element top, right, bottom, and left. following are the properties given below :

Property	Description	Syntax	Example

Margin	This property sets the margin in all directions. If two values are given then the first value is for top and bottom and the second value is for left and right.	Margin : Length	p {
			margin : 30px ;
			}
			or
			p{margin : 10px 2% ; }

margin - top	The property sets the top margin of an element and the value can be in length, percentage, or auto.	margin - top : length | auto | initial | inherit ;	p {
			margin - top : 30px ;
			}

margin – right	The property sets the right margin of an element and the value can be in length, percentage, or auto.	margin - right : length | auto | initial | inherit ;	p {
			margin - right : 30px ;
			}

margin - bottom	The property sets the bottom margin of an element and the value can be in length, percentage, or auto.	margin - bottom : length | auto | initial | inherit ;	p {
			margin - bottom : 30px ;
			}

margin - left	The property sets the left margin of an element and the value can be in length, percentage, or auto.	margin - left : length | auto | initial | inherit ;	p {
			margin - left : 30px ;
			}

Table 7 : CSS Margin Properties

All the margin properties can have the given values :
Auto : It sets the browser margin by itself.
Length : It sets the margin value in px, pt, cm, etc.
Percentage : It sets the margin in percentage (%).
Inherit : specifies that the margin should be inherited from the parents of the elements.

Handy Tips for the CSS Margin

Example of Margin	Output
<html> <head> <title> Margin </title> <style> div { border : 3px solid green ; margin - bottom : 150px ;	**Figure 5 : CSS Margin**

margin - right : 100px ; margin - top : 150px ; margin - left : 100px ; background - color : yellow ; } </style> </head> <body> Use of Margin Properties This div element contains a right margin of 100px, top margin of 150px, a bottom margin of 150px, and a left margin of 100px. </body> </html>	

Table 8 : Example of CSS Margin

7.6 CSS Padding

This property is used to create space around the content of an element inside of any defined borders. Following are the CSS properties for defining the padding for each direction of an element.

Property	Description	Syntax	Example

padding - top	This property is used to set the top space or padding of an element and it can take values in form of a length of %.	Padding - top : length | Initial | inherit ;	P
			{
			Padding - top : 2% ;
			}

padding - right	This property is used to set the right space	Padding - right : length |	P

	or padding of an element and it can take values in form of a length of %.	Initial | inherit ;	{
			Padding - right : 2% ;
			}

padding – bottom	This property is used to set the bottom space or padding of an element and it can take values in form of a length of %.	Padding - bottom : length | Initial | inherit ;	P
			{
			Padding - bottom : 2% ;
			}

padding - left	This property is used to set the left space or padding of an element and it can take values in form of a length of %.	Padding - left : length | Initial | inherit ;	P
			{
			Padding - left : 2% ;
			}

Table 9 : CSS Padding Properties

All padding properties can have the following values :
Length : It specifies padding in pixel, pt, etc.
Percentage : It specifies the padding in percentage (%) of the box containing content.
Inherit : it specifies that the padding should be inherited from the parents of elements

Handy Tips for the CSS Padding

Example of Padding	Output
<html> <head> <style> p.a1 { width : 400px ; background - color : pink ; padding - top : 50px ; margin - left : 100px ;	**Figure 6 : CSS Padding Example**

}
p.a2 {
width : 300px ;
padding - left : 50px ;
box - sizing : border - box ;
background - color : lightblue ;
}
</style> </head> <body>
 	
  Padding and element width and color is applied on paragraphs
 This paragraph is 400px wide. with top padding of 50px.
 	
 The width of this pargraph remains at 300px, border - box property, with padding from left at 50 px.
 </body> </html>	

Table 10 : Example of CSS Padding

7.7 CSS List

Lists are used to display or present the information in the form of numbers or bullets. This topic covers the list types, positions, styles, etc. with the help of CSS.

Property	Description	Syntax	Example
List - style – type	This property is used to set the style of the bullets in an unordered list and for an ordered list, it styles the numbering. The value for <UL> is none, disc, circle, square.	List - style - type : value ;	<ul style = "list - type - style : square" ; } Or <ol style = "list - type

	And for <OL> the values are decimal, decimal - leading - zero, lower - alpha, upper - alpha, lower - roman, upper - roman, etc.		- style : upper - alpha" ; }
List - style – position	This property is used to indicate that the list marker will be displayed inside or outside the box. It contains one of the two values i.e., inside or outside.	List - style - position : none | inside | outside	<ol style = "list - type - style : upper - alpha ; list - style - position : inside ; "}
List - style – image	This property is used to set the image in place of the bullet. Its syntax is similar to background - image that contains the URL of an image and begins with the URL keyword.	List - style - image : url (URL) ;	<li style = "list - style - image(a.jpg) ; "}
List – style	This property allows setting all the list properties in one expression.	List - style : value1 value2 ;	List - style : inside upperalpha ;

Table 11 : CSS List Properties

Handy Tips for the List

Example of List	Output
<html> <head> <title> List </title> <style> ul { background - color : lightblue ; padding : 20px ;	**Figure 7 : CSS List Example**

```
}ol {
  background - color : pink ;
  padding :  15px ;
}
ol li {
  background - color : white ;
  padding :  15px ;
  margin - top :  20px ;
}
ul li {
  background :  #cce5ff ;
  margin - top :  15px ;
}
</style>
</head>
<body>
<h1> LIST of Courses </h1>
<ol>
  <li> BCA </li>
  <li> B.Com </li>
  <li> BBA </li>
</ol>
<h1> LIST of Subjects </h1>
<ul>
  <li> C </li>
  <li>  C++ </li>
  <li>  Web Designing </li>
</ul>
</body>
</html>
```

Table 12 : Example of CSS List

7.8 CSS Positioning

This property is used to set the positions of elements on a web page. The default position of elements is given in normal flow i.e., block - level elements move from top to bottom for example <p>, headings, and the inline elements move from left to right for example <B>, <I>, <U>, etc. With the help of these properties, elements can be placed at any location on the browser screen.

Property	Description	Syntax	Example

relative	It changes the position of elements relative to the exact location where it appears. Only top and left can be used with position property to move the element.	position : relative ; left : value ; top : value ;	<p style = "position : relative ; left : 80px ; top : 2px ; </p>
absolute	It positions the element at a specific location relative to top left positions.	position : absolute ;	<p style = "position : absolute ; left : 80px ; top : 2px ; </p>
Fixed	This property fix the element position at a particular location on the screen	position : fixed ;	<p style = "position : fixed ; left : 80px ; top : 2px ; </p>

Table 13 : CSS Positioning Properties

Handy Tips for the CSS Positioning

Example of Positioning	Output
<html> <head> <style> p.a1	**Figure 8 : CSS Positioning Example**

```
{
  position :  relative ;
  left :  30px ;
  border :  5px solid #734535 ;
}
p.a2
{
  position :  absolute ;
  left :  30px ;
  border :  3px solid #ABAD21 ;
}
</style>
</head>
<body>
<h2>  Examples  of  absolute  and  relative positioning </h2>
<p class = "a1"> An element with position the relative and the border of the pargraph is specified with 5 px using solid line. </p>
<p class = "a2"> An element with position the absolute and the border of the pargraph is 3 px with left spacing </p>
</body>
</html>
```

Table 14 : Example of CSS Positioning

Check Your Progress

8. State whether True or False :
   i. Background - repeat : repeat - y will repeat the image horizontally.
   ii. CSS positioning can be categorized into two types : absolute and relative.
   iii. CSS list can be used for both ordered and unordered list.
9. Explain the CSS Positioning properties.

Objective Type questions :

32. The property is used to add shadow to the text in CSS is.

i) Text - stroke
c) Text - shadow
j) Text - overflow
d) Text - decoration

33. To implement inline CSS which attribute is used with the HTML tags.

i) Style
c) class

j) styles d) none of the above

34. Internal sheets are creating using which attribute in CSS.

i) <style> c) <styles>
j) <link> d) <script>

35. The CSS property is used to set the background image of any element.

i) Background - attachment c) Background - image
j) Background - color d) None of the above

36. The property used for the shorthand of padding is :

i) Padding c) padding - right
j) padding - left d) all of the above

37. Is there any negative value to be used with padding?

i) Yes c) May be
j) No d) Can't say

38. Which property is used for setting margin from the left?

i) Margin c) Margin - up
j) Margin - left d) Margin - right

39. Which property is used to set the spacing before and after every letter?

i) Character - spacing c) Alpha - spacing
j) Letter - spacing d) None of the above

Hands on Practice

Example 1 :

```
< html >
  < head >
        < title > Example of CSS < /title >
    < style >
```

```
        #borderimg1 {
          border : 10px solid transparent ;
          padding : 15px ;
          border - image - source : url(/css/images/border.png) ;
          border - image - repeat : round ;
          border - image - slice : 30 ;
          border - image - width :  10px ;
        }
        #borderimg2 {
          border : 10px solid transparent ;
          padding : 15px ;
          border - image - source : url(/css/images/border.png) ;
          border - image - repeat : round ;
          border - image - slice : 30 ;
          border - image - width :  20px ;
        }
        #borderimg3 {
          border : 10px solid transparent ;
          padding : 15px ;
          border - image - source : url(/css/images/border.png) ;
          border - image - repeat : round ;
          border - image - slice : 30 ;
          border - image - width :  30px ;
        }
      </style >
    </head>
    <body>
      <p id  =  "borderimg1" > This is image boarder example. </p>
      <p id  =  "borderimg2" > This is image boarder example. </p>
      <p id  =  "borderimg3" > This is image boarder example. </p>
    < /body >
< /html >
```

Example 2 :

```
< html >
  < head >
       < style type = "text/css" >
       table.one { border - collapse : collapse ; }
       table.two { border - collapse : separate ; }

       td . a {
```

```
        border - style : dotted ;
        border - width : 3px ;
        border - color : #000000 ;
        padding : 10px ;
      }
      td . b {
        border - style : solid ;
        border - width : 3px ;
        border - color : #333333 ;
        padding : 10px ;
      }
    < /style >
  < /head >
  < body >
      <table class = "one">
      <caption> Collapse Border Example </caption>
      <tr> <td class = "a">  Cell A Collapse Example </td> </tr>
      <tr> <td class = "b">  Cell B Collapse Example </td> </tr>
    < /table >
    < br />
    < table class = "two">
      < caption > Separate Border Example < /caption >
      < tr > < td class = "a" > Cell A Separate Example < /td > < /tr >
      < tr > < td class = "b" > Cell B Separate Example < /td > < /tr >
    < /table >
< /body >
< /html >
```

7.9 SUMMARY

This unit includes the CSS properties that helps in creating an interactive website. CSS rules consists of selector and declarations. Different selectors exist in CSS that specifies the rules on individual element of HTML. CSS rules can exist in a separate file as well as appears in the same HTML document. These attributes can be applied in four different ways as discussed in the previous unit such as Embedded, Inline, Externally, Export. Selectors are used to apply these

attributes on the HTML elements. These attributes include background, text, font, positioning, padding, margin, list etc. All the CSS properties contain different styles for example text property can have different styles such as decoration, stroke, overflow, etc. In this way, different CSS attributes can be respresented by different styles.

UNIT 8 HTML FORMS

Structure

8.0 INTRODUCTION

Forms are used for creating the website more attractive and they can also be used for collecting the data from different users. With the help of forms, surveys can be conducted for the registration or online surveys that help in getting the information about the interest in a particular field. It helps in creating an interactive medium between the server and the client. A well-designed website is easy to use and understand. Initially, static websites are created that may not ask for user input. Then, dynamic websites are created that can create interactive websites. It helps in getting the inputs from the user and displays the output based on given inputs.

8.1 OBJECTIVES

The objective of this unit is to create an interactive web page using the concept of Forms. After completing this unit, learners will be able to understand the following concepts.

- HTML Forms for accepting user input
- Form attributes

This unit deals with designing the interactive web page that helps in accepting user input and displays the output according to input.

8.2 HTML Forms

HTML forms are used to collect the data from the visitors. These forms are similar to ordinary forms that we fill in our everyday life like application forms, reservation forms, etc. The forms can be used to create an interactive website that captures necessary input from the user and displays the processed result on the client's browser.

An HTML form is a container of other elements that can be used for data gathering. It provides the complete GUI elements. The data filled by the user can be submitted to the back-end server by the CGI, PHP scripts, Server-side scripts, etc. There are basic form elements such as text fields, checkbox, radio buttons, drop-down list, etc. Form can be created using <FORM> tag.

An HTML form is a main part of the website or web document. Users can use the form to enter their information and this information is passed back to the webserver and recorded in the database in the form of tables for future use. Forms are used for two-way communication between users and the server.
The common use of forms includes in HTML:

1. **Online form:** These forms are quite common on the web to provide a visual interface for the user to order products from online platforms such as Flip-kart, Amazon, and Alibaba. Order entry forms need user-related information like address, credit card number, delivery date, and some other information that is necessary to facilitate online commerce.

2. **Subscription forms:** Subscription forms are acquired by many websites especially, those which generate revenue through direct subscriptions or by selling advertising space.

3. **Registration forms:** These forms are used for collecting information about viewers and set an authentication system for accessing the website online by the registered participants.

4. **Feedback form:** It is generally used by companies as a way to collect feedback from users for their services and give their response in the favour of companies' improvements.

8.3 <FORM> Tag and Controls

HTML forms are used as a container that includes different controls such as text boxes, buttons, radio buttons, checkboxes, text areas, etc. The form is created using the <FORM> tag in the HTML document. It facilitates the user to enter the data that should be stored on the server for

further processing such as id, name, password, contact number, etc. An opening form is represented using <FORM> and the closing tag is represented using </FORM> tag. and in between these tags, all other tags will be given.

Controls used in forms: There exist different controls in HTML that can be added in the form to collect data from a user. The most common element is the < INPUT> tag. It is a singular tag. TYPE is the attribute used with this tag that determines the type of control which can be used in the form. These are given as follows:

8.3.1. Text Controls: This control helps in gathering information from the user. Text controls can be represented in different ways such as:

- **Single line text control**: <INPUT TYPE = " TEXT"> control is used to enter a single line of text such as user name, city, etc.
  Syntax: <INPUT TYPE = "TEXT" NAME = "USERNAME">
  This tag specifies a single text box. And the NAME attribute is used to give the name to the input control. There may exist other attributes of this <INPUT> tag.

Attributes	**Description**	**Syntax**	**Example**
TYPE	It indicates the type of control that can be added in the form.	<INPUT TYPE = "text" >	Enter username <INPUT TYPE = "text">
NAME	It specifies the name of the input control. It is not visible on the form. And it must be unique within the form.	<INPUT TYPE = " TEXT" NAME="control name">	<INPUT TYPE = "TEXT" NAME = " USERNAME">
SIZE	It specifies the number of characters that should be entered in the control. The default value is 20 characters.	<INPUT TYPE = "TEXT" NAME = "control name" SIZE = "value">	<INPUT TYPE = "TEXT" NAME = "USERNAME" SIZE = "15">
MAXLENGTH	It specifies the maximum number of characters that should be allowed to enter in the control. The default length of this attribute is unlimited.	<INPUT TYPE = "TEXT" NAME = "CONTROL NAME" MAXLENGTH = value>	<INPUT TYPE="TEXT" NAME = "USERNAME" MAXLENGTH = 15 >

VALUE	It represents the default text that should be available in the control.	<INPUT TYPE = "TEXT" NAME = "control name" VALUE = "text">	<INPUT TYPE = "TEXT" NAME = "USERNAME" VALUE = "PLEASE ENTER" >

Table 1: Attributes of Text Control

Let us consider an Example:	**Output:**
<HTML> <HEAD> </HEAD> <BODY> <FORM> <H1> LOGIN FORM </H1> PLEASE ENTER YOUR NAME: <INPUT TYPE = "TEXT" VALUE = "PLEASE ENTER" > </FORM> </BODY> </HTML>	**Figure 1: Output of text field**

Table 2: Example of Text Control

- **Password text control:** This control is a single line text control with only one difference the text is represented in ciphertext i.e., in the form of a password. The text is displayed either in asterisk '*' or dots '.'
  For example:

  <INPUT TYPE = "PASSWORD" NAME = "Password" VALUE = "Enter Password">

Example of Password Field:	**Output:**
<HTML> <HEAD> </HEAD> <BODY> <FORM> <H1> LOGIN FORM </H1>	**Figure 2: Output of Password field**

Username: <INPUT TYPE = "TEXT" VALUE = "PLEASE ENTER"> Password <INPUT TYPE = "PASSWORD" NAME = "Password" VALUE = "Enter Password"></FORM></BODY></HTML>	

Table 3: Example of Password Control

- **Multiline text control:** This control can be used to enter multiple lines of text using the <TEXTAREA> tag. This tag begins with the <TEXTAREA> and ends with the </TEXTAREA> tag.

Attributes	Description	Syntax	Example
NAME	It is used to give name to the control.	<INPUT NAME = "TEXT">	<INPUT TYPE = "TEXT" ID = "LNAME" NAME = "LNAME">
ROWS	It represents the number of rows in the textbox i.e., the height of the textbox.	<TEXTAREA ROWS = "VALUE">TEXT CONTENT...</TEXTAREA>	<TEXTAREA ROWS = "5" COLS = "23"> The textarea represents 5 rows. </TEXTAREA>
COLS	It represents the number of columns in the textbox i.e., the width of the textbox.	<TEXTAREA COLS = "VALUE">TEXT CONTENT...</TEXTAREA>	<TEXTAREA ROWS = "5" COLS = "5"> This paragraph has to represent an equal number of rows and columns. </TEXTAREA>
WRAP	If the wrap is off then it sets the text in one long sequence without having any line break. It can be set as Virtual or Physical.	< TEXTAREA COLS = "VALUE" WRAP = PHYSICAL> TEXT CONTENT... </TEXTAREA>	<TEXTAREA ROWS = "5" COLS = "5" WRAP = PHYSICAL > This paragraph has to represent an equal number of rows and columns. </TEXTAREA>

Table 4: Attributes of Multiline Text Control

Example: <TEXTAREA ROWS = "10" COLS = "10" NAME = "text">
…… Text to be entered
</TEXTAREA>

8.3.2. Checkbox Control: Checkbox control is used when there are more than one options and the user can select multiple options. These controls are also known as toggle control, which means that when the control is selected it is set to 1 or ON otherwise 0 or OFF. This control

can be created using the <INPUT> tag with the type value as “CHECKBOX”.

Attributes	Description	Syntax	Example
NAME	It is used to give the name to the control.	<INPUT NAME = "TEXT">	<INPUT TYPE = "CHECKBOX" NAME = "course">
TYPE	For the checkbox control, it will be set to CHECKBOX.	<INPUT TYPE = "CONTROL_NAME">	<INPUT TYPE = "CHECKBOX" NAME = "course">
VALUE	By default, its value is ON. Once the control checked property is TRUE, its value is stored on the server for processing.	<INPUT VALUE = "TEXT">	<INPUT TYPE = "CHECKBOX" VALUE = "course">
CHECKED	This property by default selects the checkbox to true.	<INPUT TYPE = "CONTROL_NAME" NAME = "TEXT" VALUE = "TEXT" CHECKED>	<INPUT TYPE = "CHECKBOX" NAME = "course" VALUE = " COMPUTERS" CHECKED>

Table 5: Attributes of Checkbox Control

Example:
<INPUT TYPE = "CHECKBOX" NAME = "Course" VALUE = "C" CHECKED>

Output:

- ☑ C
- ☑ C++
- ☑ Java
- ☑ Web Design

8.3.3. Radio Buttons: This control is used to select one option from multiple options. This is used where only one selection is allowed such as for gender selection, objective questions, etc. Radio buttons are represented in groups i.e., all the radio buttons share a common name. In this case, if one option is selected, the other will remain unselected. This control can be created using the <INPUT> tag with the type value as “RADIO”.

Attributes	Description	Syntax	Example
NAME	It is used to give the name to the control.	<INPUT NAME = "TEXT">	<INPUT TYPE = "RADIO" NAME = "GENDER">
TYPE	For the radio buttons in the form, it will be set to the radio.	<INPUT TYPE = "CONTROL_NAME">	<INPUT TYPE = "RADIO" NAME = "GENDER">

VALUE	The value of the radio button is used if it is selected.	<INPUT VALUE = "TEXT">	<INPUT TYPE = "RADIO" VALUE = "COURSE">
CHECKED	This property by default selects the radio button to true.	<INPUT TYPE = "CONTROL_NAME" NAME = "TEXT" VALUE = "TEXT" CHECKED>	<INPUT TYPE = "RADIO" NAME = "COURSE" VALUE = " COMPUTERS" CHECKED>

Table 6: Attributes of Radio button Control

Example:
<INPUT TYPE = "RADIO" NAME = " Gender" VALUE = "Male" >
<INPUT TYPE = "RADIO" NAME = "Gender" VALUE = "Female" CHECKED >

Output:

Male
Female

8.3.4. Label: Label control is used to display labels for different controls in the form. This control is useful for the readers that indicates the types of values to be entered in the textbox. It makes the website user-friendly by adding labels in the form.

Syntax: <LABEL FOR = "Username" > User Name </LABEL>
< INPUT TYPE = "TEXT" ID = "Username" >

In this syntax, the attribute of the label includes the FOR, which represents it should be used for which textbox. and the ID of the textbox and FOR contains the same value.

8.3.5. Drop Down List: This control is used for selecting a single option or multiple options from the list of choices. It displays only one choice, when the user clicks on the drop-down list the complete list will be displayed on the screen. The <SELECT> tag represents the menu that should be displayed and the <OPTION> tag is used to add the choices in the list. Its length can be represented in the longest choice present in the list. Following are the attributes of the drop-down list.

Attributes of <SELECT>	**Description**	**Syntax**	**Example**
NAME	This attribute assigns the name to the drop-down list. When there exist multiple lists, each list name must be unique.	<SELECT NAME = "TEXT">	<SELECT NAME = "COURSE">

SIZE	This can be used to set the number of choices that will be displayed on the screen at once.	<SELECT NAME = "TEXT" SIZE = "NUMBER">	<SELECT NAME = " COURSE" SIZE = "3"> <OPTION>BCA </OPTION> <OPTION>B.Com</OPTION> <OPTION>MCA </OPTION> <OPTION>PGDCA</OPTION> <OPTION>B.SC. </OPTION> </SELECT>
MULTIPLE	This attribute helps the user to select multiple choices from the drop-down list. To select multiple options press CTRL+ click on the choice.	<SELECT NAME = "TEXT" SIZE = "NUMBER" MULTIPLE>	<SELECT NAME = "Course" SIZE = "3" Multiple>

Table 7: Attributes of <SELECT> Tag

Attributes of <OPTION>	**Description**	**Syntax**	**Example**
VALUE	This attribute is used to send the value to the server for further processing.	<OPTION VALUE = "VALUE">	<OPTION VALUE = "DELL">DELL</OPTION>
SELECTED	This attribute will select the option automatically.	<OPTION SELECTED>	<OPTION VALUE = "DELL" SELECTED>DELL</OPTION>
OPTGROUP	This option is used to select groups of multiple options in logical order. This works when there are a large group of options. More than one <OPTGROUP> tag is available.	<OPTGROUP LABEL = " ">...</OPTGROUP>	<SELECT> <OPTGROUP LABEL = "LAPTOP"> <OPTION VALUE = "DELL">Dell</OPTION> <OPTION VALUE = "HP">HP</OPTION> </OPTGROUP> <OPTGROUP LABEL = "MOBILE"> <OPTION VALUE = "APPLE">Apple</OPTION> <OPTION VALUE = "NOKIA">Nokia</OPTION>

			</OPTGROUP> </SELECT> OUTPUT: **FIGURE 3: OUTPUT OF DROP-DOWN LIST**

Table 8: Attributes of <OPTION> Tag

8.3.6. File Upload: This control helps the user to upload the files to the server at the time of submission of a form. <INPUT> tag is used to create this control that includes TYPE attribute with value FILE.

Example:
Select the file you want to upload: <INPUT TYPE = "FILE" NAME = "Fileupload">

Output:

Figure 4: Output of file upload

The browser displays this screen after creating file upload control in the form. To upload the file on the server, the user is required to click on the choose file button, the dialog box appears. Select the file from the dialog box and then click ok. In the end, on submitting the form to the server the file is appended to the data and the record is sent to the server.

8.3.7. Buttons (Submit and Reset): The button control can be used to send the data to the server. There are three types of buttons in HTML.

- **Submit button**: This button is used to send the form data to the server that is entered by the user. <INPUT> tag is used to create the submit button and attribute TYPE is set to Submit.

Syntax:

<INPUT TYPE = "SUBMIT" VALUE = "SUBMIT">

The default value of this button is submitted. If a user wants to change the value, then it can specify in the VALUE attribute of the INPUT tag. Once the user clicks on this button the values filled by the user in the form are sent to the server.

- **Reset button:** This button clears all the entries filled by the user in the form and resets all the controls to their default or initial value. <INPUT> tag is used to create the reset button and attribute TYPE is set to Reset.

Syntax:

<INPUT TYPE = "RESET" VALUE = "RESET">

The value can be changed and the default value of the RESET button is always set to RESET.

- **Push-button:** Similar to the Submit or reset button, general buttons can also be created to generate events on the browser. These buttons are created by setting the TYPE attribute to BUTTON and can also be edited with the image, text, or markups.

  Syntax:

  <INPUT TYPE = "BUTTON" VALUE = "SEND TO SERVER">

  This button sends the value to the server. It includes the label SEND TO SERVER.

- **Image button:** This button is similar to the push button. It can include the image at the place of value.

  Syntax:

  <INPUT TYPE = "IMAGE" SRC = "*.jpg" ALT = "submit button">

The TYPE attribute of the INPUT tag contains IMAGE and the SRC is represented as a source of the image. ALT attribute specifies the alternate text of the image, if by default the image is not displayed, the alternate text appears.

8.3.8. Hidden Controls: The hidden control is used to embed the data in a page that is not visible and is not updated by the user. The <INPUT> tag is used to represent the hidden field. The attribute of this control is NAME, TYPE, and VALUE. The TYPE attribute contains the HIDDEN value.

Syntax:

<INPUT TYPE = "HIDDEN" NAME = "aa" VALUE = "field">

There exists a large number of controls used in Forms in HTML. To group these controls, two more tags are used for grouping and that makes a user-friendly interface.

- **<FIELDSET>:** This tag creates a border around the related information in the form such as personal details are created separately, professional qualification is separate.
- **<LEGEND>:** This tag gives the title to the block in such a way as to identify the information required in the block. For example, personal details, qualifications, etc.

8.4 Special <Input> Types:

1. **Files:** HTML also supports the special input fields, a file field that helps the visitors to upload the files on the server. For example, if any one wants to submit the information

such as picture, word document, excel file or any scanned document. This field can be used to simply upload the file using FTP or email.

Syntax: <input type = "file" name = "id" size = "n" accept = "file_type">

Attributes	Description	Syntax	Example
NAME	This is used to specify the name of field.	<INPUT NAME = "TEXT">	<INPUT TYPE = "FILE" NAME = "ATTACHMENT">
TYPE	It specifies the type of file.	<INPUT TYPE = "CONTROL_NAME">	<INPUT TYPE = "FILE" NAME = "ATTACHMENT">
SIZE	It specifies the size of input field.	<INPUT SIZE = "VALUE">	<INPUT SIZE = "30" TYPE = "FILE">
ACCEPT	This attribute is used to restrict the types of files that is allowed to be uploaded.	<INPUT TYPE = "CONTROL_NAME" SIZE = "VALUE" ACCEPT = "IMAGE/JPG" >	<INPUT TYPE = "FILE" NAME = "ATTACHMENT" SIZE = "30" ACCEPT = "IMAGE/jpg">

Table 9: Special Attributes of <INPUT> Tag

2. **Access keys:** The form elements can be accessed either by the mouse click or by using the tab key. The visitors can access the elements directly by using the access key attribute. This can be possible by assigning the characters to the form element as an access key attribute. To access the control the particular key is pressed with the alt key. While assigning the access key, always be careful that the user is not override the browser's own access keys. And also, the function keys are not used as the access keys because they are already assigned a function.

Syntax: <input type = "file" name = "id" size = "n" accept = "file_type" accesskey = "U">

Handy Tips of Forms:

Example of Form elements	Output
<HTML> <HEAD> <TITLE> Basic controls of form </TITLE> </HEAD> <BODY> <FORM METHOD = POST> Enter Username	**Figure 5: Output of Forms**

<INPUT TYPE = TEXT NAME = "ADDRESS" SIZE = 10> Enter Password <INPUT TYPE = PASSWORD MAXLENGTH = 10> Course <INPUT TYPE = RADIO NAME = "course" CHECKED> BCA <INPUT TYPE = RADIO NAME = "course"> BBA <INPUT TYPE = RADIO NAME = "course"> B.Com<P> Subjects <INPUT TYPE = CHECKBOX> C <INPUT TYPE = CHECKBOX> C++ <INPUT TYPE = CHECKBOX> Web Designing<P> City <SELECT> <OPTION>Patiala <OPTION>Chandigarh <OPTION>Rajpura <OPTION>Mohali <OPTION>Delhi <OPTION>Ludhiana </SELECT> </FORM> </HTML>	

Table 10: Example of Form controls

Handy Tips of Forms: University Registration Form

```
<html>
<head>
<title>
University Registration Page
</title>
</head>
<body bgcolor = " 2999cc ">
<br>
<br>
```

```
 <form>
  <h1>University Registration Form</h1>
<label> <b>Enter Your First name</b> </label>
<input type = "text" name = "firstname" size = "15"/> <br> <br>
< label> <b> Enter Your Middle name:</b> </label>
<input type = "text" name = "middlename" size = "15"/>  <br> <br>
<label> <b>Enter Your Last name:</b> </label>
<input type = "text" name = "lastname" size = "15"/>  <br>  <br>
<label>
<b> Course:</b>
</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>
<b>Gender:</b>
</label><br>
<input type = "radio" name = "male"/> Male<input type = "radio" name = "female"/> Female
<input type = "radio" name = "other"/> Other
<br>
<br>
<label>
<b> Enter Your Phone: </b>
</label>
<input type = "text" name = "country code" value = "+91" size = "2"/>
<input type = "text" name = "phone" size = "10"/> <br> <br>
<b> Enter Your Address: </b>
<br>
<textarea cols = "68" rows = "5" value = "address">
</textarea>
<br> <br>
<b> Enter Your Email: </b>
<input type = "email" id = "email" name = "email"/> @mail.com <br>
<br> <br>
<b> Enter Your Password: </b>
<input type = "Password" id = "password" name = "password" maxlength = "8"> up to 8
characters <br>
<br> <br>
```

Figure 6: Registration Forms

```
<b> Enter Your Re-type password : </b>
<input type = "Password" id = "repassword" name = "repassword"> up to 8 characters <br> <br>
<input type = "button" value = "Submit"/>
</form>  </body></html>
```

Check Your Progress 1

1. State whether True or False:
   i. There exist two attributes of <form> tag i.e., Action and Method.
   ii. The text field can be used to enter multiline text.
   iii. The INPUT tag can be used for dropdown lists.
2. Design the registration form of the website.
3. Design the following given format:

Figure 7: Design Forms

Hands On Practice

Example 1:

```
<html>
<head>
<title> signup form </title>
</head>
<body>
<form action = "/signup" method = "POST">
  <p>
    <label>Title</label>
    <label>
      <input type = "radio" name = "title" value = "mr">
      Mr
    </label>
    <label>
      <input type = "radio" name = "title" value = "mrs">
      Mrs
    </label>
    <label>
      <input type = "radio" name = "title" value = "miss">
      Miss
    </label>
```

```
  </p>
  <p>
    <label>First name</label>
    <input type = "text" value = "first_name">
  </p>
  <p>
    <label>Last name</label>
    <input type = "text" value = "last_name">
  </p>
  <p>
    <label>Email</label>
    <input type = "email" value = "email">
  </p>
-  <p>
    <label>Phone number</label>
    <input type = "tel" value = "phone">
  </p>
  <p>
    <label>Password</label>
    <input type = "password" value = "password">
  </p>
  <p>
    <label>Confirm your password</label>
    <input type = "password" value = "password_confirm">
  </p>
  <p>
    <label>Country</label>
    <select>
      <option>Canada</option>
      <option>France</option>
      <option>Germany</option>
      <option>Italy</option>
      <option>Japan</option>
      <option>Russia</option>
      <option>United Kingdom</option>
      <option>United States</option>
    </select>
  </p>
  <p>
    <label>
      <input type = "checkbox" value = "terms">
      I agree to the <a href = "/terms">terms and conditions</a>
    </label>
  </p>
  <p>
    <button>
```

```
    Sign up
   </button>
  </p>
</form>
</body>
</html>
```

Example 2:

```
<html>
<head>
<title> booking form </title>
</head>
<body>
<form method = "post">
  <div class = "elem-group">
    <label for = "name">Your Name</label>
    <input type = "text" id = "name" name = "visitor_name" placeholder = "John Doe" pattern =
[A-Z\sa-z]{3,20} required>
  </div>
  <div class = "elem-group">
    <label for = "email">Your E-mail</label>
    <input type = "email" id = "email" name = "visitor_email" placeholder =
"john.doe@email.com" required>
  </div>
  <div class = "elem-group">
    <label for = "phone">Your Phone</label>
    <input type = "tel" id = "phone" name = "visitor_phone" placeholder = "498-348-3872"
pattern=(\d{3})-?\s?(\d{3})-?\s?(\d{4}) required>
  </div>
  <hr>
  <div class="elem-group inlined">
    <label for="adult">Adults</label>
    <input type="number" id="adult" name="total_adults" placeholder="2" min="1" required>
  </div>
  <div class="elem-group inlined">
    <label for="child">Children</label>
    <input type="number" id="child" name="total_children" placeholder="2" min="0" required>
  </div>
  <div class="elem-group inlined">
    <label for="checkin-date">Check-in Date</label>
    <input type="date" id="checkin-date" name="checkin" required>
  </div>
  <div class="elem-group inlined">
    <label for="checkout-date">Check-out Date</label>
```

```
    <input type="date" id="checkout-date" name="checkout" required>
  </div>
  <div class="elem-group">
    <label for="room-selection">Select Room Preference</label>
    <select id="room-selection" name="room_preference" required>
      <option value="">Choose a Room from the List</option>
      <option value="connecting">Connecting</option>
      <option value="adjoining">Adjoining</option>
      <option value="adjacent">Adjacent</option>
    </select>
  </div>
  <hr>
  <div class="elem-group">
    <label for="message">Anything Else?</label>
    <textarea id="message" name="visitor_message" placeholder="Tell us anything else that
might be important." required></textarea>
  </div>
  <button type="submit">Book The Rooms</button>
</form>
</body>
</html>
```

8.5 SUMMARY

This unit covers the important and advanced topics that enable users to create interactive websites. The most important concept of creating a dynamic website is the use of Forms, Client-side, or Server programming, and the use of CGI scripts makes the interactive websites. Forms use different kinds of controls such as Textbox, Radio buttons, Checkbox, Dropdown list, etc. These controls help the users to enter the details in the form and submit the details to the server.

UNIT 9 SCRIPTING LANGUAGE

Structure

9.0 INTRODUCTION

There exist two types of webistes i.e., static and dynamic. Dynamic sites are created by using the concept of forms and client/server programming. Client/Server Programming is used for the communication of the client and server where the client sends data or information received through a browser on the server. CGI or Common Gateway Interface standards are used to create interactive websites that help in submitting data to the server and after submitting the client may get the acknowledgment of the submission. This unit covers the concepts of DHTML i.e., Dynamic Hypertext Markup Language. It is not HTML but an extension of HTML.

9.1 OBJECTIVES

The objective of this unit is to create an interactive web page using the concept of Forms and helps in creating dynamic website. After completing this unit, learners will be able to understand the following concepts.

- Action and Method attributes
- Server-side Programming
- Client-side Programming
- Common Gateway Interface
- DHTML

9.2 ACTION and METHOD Attributes

The form contains two attributes i.e., ACTION and METHOD.

1. **ACTION:** This attribute contains the address of the page that will receive the form's data after submission by the user. This attribute can be set to either email address using mailto: address or by using processing scripts such as registration. asp, login.asp, etc.

   For example:

<FORM ACTION=mailto:abc@gmail.com>
<FORM ACTION="registration.asp">

2. **METHOD:** This attribute represents how to send the data to the server once received from the user. There exist two methods of sending the data. These are:

   - **GET Method:** This method gets the data from the form elements entered by the user, encodes the data, and appended the data to the URL.

     <FORM METHOD="GET" ACTION="registration.asp">

     Once the user clicks on the submit button, the data filled in the registration form will be represented in the URL. The submission will encode the data in encrypted form.

   - **POST Method:** This method gets the data from the form elements and sends the data through HTTP without the need to append it to the URL.

     <FORM METHOD="POST" ACTION="registration.asp">

     This method is used for sending a large amount of data, even the data files.

   The POST method is better as compared to the GET method because while sending sensitive data in the POST method data is not visible in the URL. This data cannot be altered or read during transmission.

Example of methods:

```
<html>
<head>
        <title>HTML form action Attribute Example</title>
</head>
<body>
        <form action="action.asp" method="get" enctype="text/plain">
                First Name: <input type="text" name="first_name" /><br />
                Last Name:  <input type="text" name="last_name" /><br />
                <input type="submit" value="Submit" />
        </form>
</body>
</html>
```

Difference between get and post method:

Sr no	Get Method	Post Method
1	Get method is requested from a specific resource.	Post method is submitted to to be processed to a specific resource.
2	In get method parameters passed in url.	Post method parameters passed in body.
3	Get method is used for fetching the documents.	Post method used for updating the data.

4	Get method is used to request the data from a particular resource.	Post method used to send information to a server to create/update a resource.
5	Get method can be used for cache the data.	Get method cannot be used for cache the data.
6	In which limited data is sent.	In which large amount of data is sent.
7	It sends data as part of URL (Uniform resource identifier).	It sends data as http content.
8	It is no not secure.	It is more secure.

Table 1: Comparison between Get Method and Post Method

9.3 Scripting Languages

Web development is all about the communication between two ends client and the server via various kinds of protocols such as HTTP, HTTPS and, FTP, etc. The client and server are working as follows.

Client: The client requests the webpage from the server and displays the requested pages to the end-user browser's window.

Server: The server fulfills all the requirements of the client/end-user. It can be either static or dynamic.

The browsers request the servers using the protocol i.e., HTTP (Hypertext Transfer Protocol). Once the client clicks the link on the web page for submitting the form, then through the HTTP protocol the browser sends the request to the server.

While sending the request, it includes the URL and the other information of the client using the GET or POST method. The GET method may append the sensitive information in the URL or the POST method sends the large data files. Whenever the server receives the client request, the server responds to the requests through bowsers. Scripting languages are the example for creating client-side or side programming.

The scripting language is a programming language i.e., interpreted at the run time. Scripting languages are often used for short scripts or series of commands that can be executed without compiling. Although, all the scripting languages such as Python, Ruby, and Java scripts are programming languages. All the programming languages are not scripting languages. These languages are of two types:

9.3.1 Client-Side Programming: In Client-side programming, an entire program that runs on the client-side, or we can say that client-side programming mostly deals with the user interface with which the user interacts on the web. It allows for a more interactive way to perform several actions without going to the user and without interaction with the database directly. It can also be used to create “cookies” at the client end and store the temporary file (cookies) on the user’s computer.

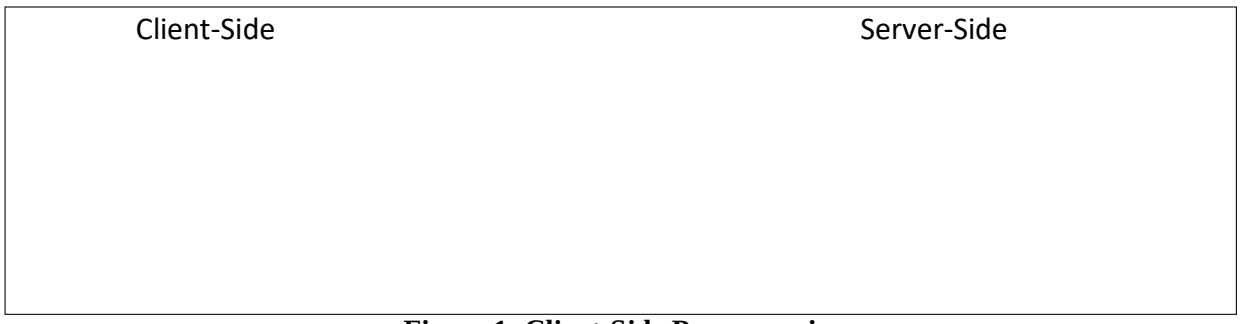


Figure 1: Client-Side Programming

The above diagram shows the basic structure of a web server for the client-side. In which a client sends a request through the browser to the server by using the HTTP "GET" method. On the other hand, the server receives a request and returns an HTTP response to the client browser. If the requested file cannot be retrieved for some reason, an error status is returned. Client-side programming provides various benefits:

1. It helps to make interactive web pages.
2. It has to create dynamic websites.
3. It interacts with the temporary storage.
4. It creates the interface between the user and the server.
5. It helps for transforming the data from the client to the server.
6. It retrieves the data from the server.
7. It provides remote access for the client-server program

The client-side scripting language are JavaScript, Html, VBScript, CSS, Ajax, jQuery, etc.

9.3.2. Server-Side Programming: Server-side programming means every program that runs directly on the server and deals with dynamic content that interacts with the dynamic pages. Server-side programming helps the developer for sending and retrieves the information from the database as per the requirement. Web server is used to execute server-side scripting for creating dynamic pages. Server-side scripting is faster than client-side scripting because it interacts directly with a database. It can also be used to maintain the "session" at the server-side, which creates a file in a temporary file on the server where registered session variables and the information about the client are stored. A session creates a file in a temporary directory on the server where registered session variables and the value of clients are stored.

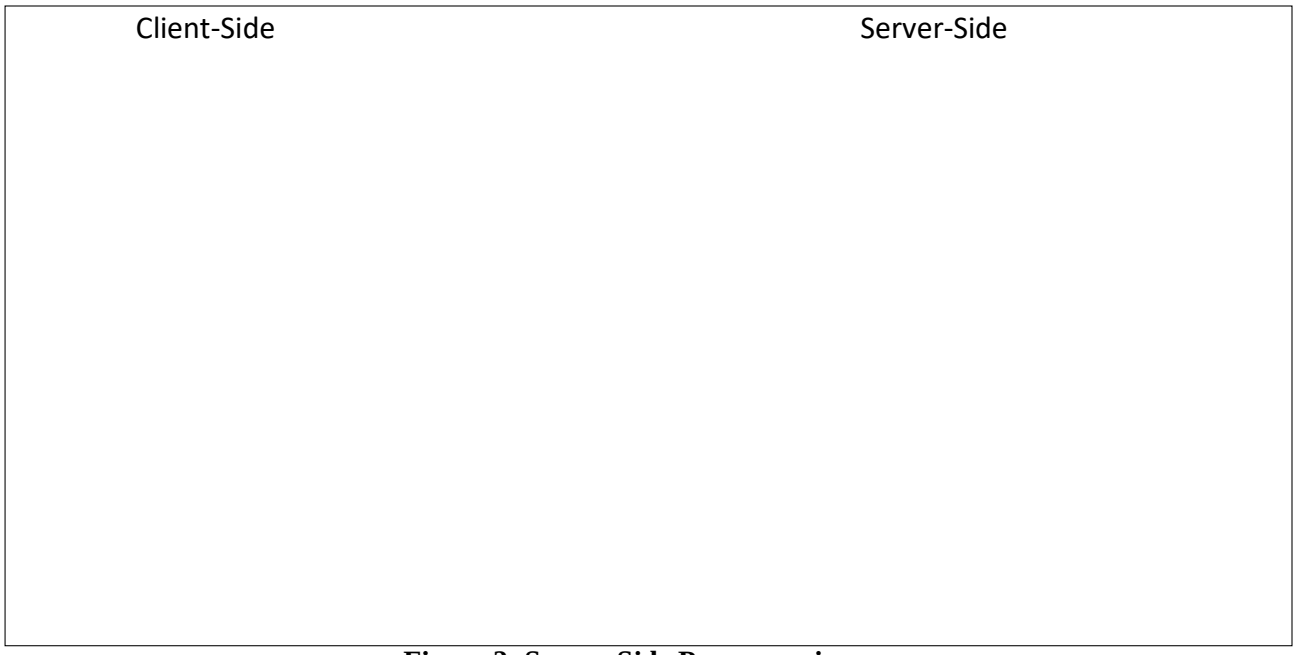


Figure 2: Server-Side Programming

The above diagram shows the basic structure of a web server for a server-side. As in the previous diagram, the browser sends HTTP requests to the server, and then the server fulfills the requests and returns the HTTP response to the browser window. On the server-side, the server interprets the request, reads required information from the database, combines the retrieved data with HTML layout, and sends back a response to the browser end. Server-side programming provides the following benefits:

1. It interacts with the user input.
2. It helps to create the structure of web applications.
3. It interacts with server or database storage.
4. It fulfills querying the database.
5. It helps to encode the data into HTML format.
6. It fulfills the operations like deletion, updating, and insertion.

Server-side scripting languages are Java and JSP, PHP, C++, ASP.NET, etc.

9.3.3. Difference between Client-Side Programming and Server-Side Programming:

Parameters	**Client-Side Programming**	**Server-Side Programming**
Visibility of code	Source Code invisible on the client-side.	Source code is not visible to a client because its code is stored on the server end.
Stored Procedure	It usually depends on the browser because the values are stored on the browser in the	In which any server-side technology such as PHP, ASP.NET can be used and it does not

	form of “cookies”.	depend on the client.
Executed On	It can run on the user's computer.	It can run on the webserver.
Speed	It is slow a process.	It is faster than the client-side.
Security	It does not provide any kind of security.	It provides security.
Examples	HTML, CSS, and JavaScript are the example of client-side programming.	PHP, ASP.NET, JAVA, and Ruby are the versions of server-side programming.

Table 2: Comparison between client-side and Server-side programming

9.4 Common Gateway Interface (CGI)

The Common Gateway Interface (CGI) standard or set of rules is defined by W3C (World Wide Web Consortium) that defines the interaction of the program with the HTTP server. It can be used to run the server programs and implements as a middleware between the servers and other sources of information. The programs run using these features of CGI are called CGI scripts. The server processes the data and passes the information to the program and may receive a confirmation acknowledgment. The process of sending data between server and application is referred to as Common Gateway Interface (CGI).

These concepts are used to create interactive pages, that help to store the client data on the server and may receive the response depending on the input. An example of an interactive web page is a search engine. The client requests by entering keywords and the server responds by sending a list of related web pages based on the keyword. This list is dynamic i.e., it is based on the client's requested keywords.

HTML form elements are used to create interactive web pages that accept the client input and pass the control to the server. and the server then responds to the client. These programs are referred to as gateways. Gateways are generally the medium between server and data sources that exchanges the information. This is the reason for creating CGI programming that handles client requests.

Features of CGI:

- Well-defined standard.
- Scripts are written in C, Perl, or a simple shell script.
- Gather the information using the HTML interface.
- Compatible with the latest browsers.
- The results will be displayed in HTML format after database access.

9.4.1 Working Of CGI

The working CGI script includes a sequence of steps. This work is under observation by the server and needs to be configured to process the script. It is written in C, Perl, Shell script, etc. The CGI script can be created for the dynamic implementation of HTML. Following are the steps that can be considered while clicking a link on a web page as shown in the Figure.

- The browser sends the request to the client for the information from the server.
- The server interprets the URL and activates the CGI script given in the URL.
- The script processes the parameters by searching the information and if the information is found, it will be returned to the client's browser by adding the MIME header, otherwise, an error message will be displayed.
- In the end, the browser displays the information received from the server or the error message.

Figure 3: CGI Communication

In this, if the server is configured in between then the information that can be accessed by the server is available in a certain drive. It is not sent to the browser rather a program is executed and the output is sent back to the user's browser. In this way, Common Gateway Interface plays an important role.

9.4.2 GET and POST Method

The form tag uses two methods to transfer the data from the browser to the CGI script.

1. **GET**: The browser submits the form's data entered by the user is encoded with the URL

and then sent to the server.

2. **POST**: The browser sends a large volume of data or even files to the server using this method.

9.4.3 Applications of CGI:

CGI can be used for simple interactive applications. some examples, where CGI can be used, are given as:

- Email forms, Registration forms, Feedback forms
- Maintaining the list of emails
- any user can leave a message for any other user through the guest book.
- Blogs
- Allow users user to color images using a coloring book.

9.4.4 Advantages of CGI:

- It is easy to perform the task in CGI as compared to Java.
- Easy to use existing code than creating your code.
- It specifies the program can be written on any platform or in any language.
- A large existing library of CGI is available that can be reused.
- CGI scripts run on the server; thus, they are secure.

9.4.5 Disadvantages of CGI:

- There is a lot of overhead while loading programs in memory.
- It takes a lot of processing time.
- Less efficient, as data cannot be cached between page loads.

Check Your Progress 1

10. State whether True or False:
    i. A gateway acts as a mediator between the server and the external sources of information.
    ii. CGI returns the output to the client browser.
    iii. Client-side programming does not provide any security as compared to server-side programming.
    iv. CGI scripts are not secure.
11. Explain the working of CGI scripts.
12. What are the advantages and disadvantages of CGI scripts?
13. Compare the client-side programming and server-side programming.
14. Explain in detail client-side and server-side programming.

9.5 DHTML

DHTML refers to Dynamic Hypertext Markup Language that is used to move the content on the web page. It is a new and growing technology that helps in creating an eye-catching and attractive website. It is not a unique language rather a combination of multiple languages such as Hypertext Markup Languages (HTML), Cascading Style Sheets (CSS), Scripting languages such as JavaScript, and DOM (Document Object Model) that helps in the creation of dynamic web pages. To support dynamic HTML some browsers are available such as Internet Explorer and Netscape Navigator with higher versions compared to 4.0. DHTML includes the properties of HTML, CSS, JavaScript, DOM that make dynamic content of the web pages.

With the help of HTML, elements are created and then formatted with CSS. After creation and formatting, it is combined with JavaScript to add the animations and effects on the web page. This is done with the help of DOM i.e., Document Object Model that uncovers the HTML tags and CSS properties to JavaScript.

DHTML helps in designing a dynamic web page that helps in enhancing the effects and it involves games, animations, show and hides the text or images alternatively, banners are created that display the latest articles or news, rollover buttons, and menus. With the help of DHTML, one can easily add effects in the document that was difficult to achieve with the help of HTML. DHTML can be used for making productive applications for business. Dynamic forms can be created that helps to take the user input such as tracking sales and analysis, registering students for any activity, order details, etc. and it helps in getting additional details in the background. This can be done dynamically and in this dynamic HTML plays an important role in creating client-server business applications. For example,

- It hides the text and images of the document and remains hide until the user interacts with the page.
- It helps in animating text and images by moving the objects on the browser.
- By creating forms, and reading, processing, and responding to the users with the given input.

Thus, DHTML technology is composed of HTML, CSS, JavaScript, VBScript standards. In some browsers, JavaScript is supported and some follow VBScript. Netscape Navigator does not support VBScript and is only supported by Internet Explorer.

DHTML=HTML+CSS+JavaScript/VBScript

To create real-time and interactive web pages, DHTML is used. To access the dynamic web pages, the code is analyzed by the server, and then the processed HTML page is displayed on the browser.
The components of the DHTML are as follows:

1. **HTML**: It stands for Hypertext Markup Language that works on the client's browser. It is used to create static web pages.
2. **CSS:** It stands for Cascading Style Sheet. Once the static web page is created, CSS is used to give styles that help in making an attractive web page.

3. **JavaScript**: This is the most widely used scripting language that is supported by all browsers. It can also store the cookies for the user's identification at the browser.
4. **DOM (Document Object Model)**: This model can be used to create dynamic and moving content across the web page.

9.5.1. Comparison of HTML and DHTML

Sr No.	HTML	DHTML
1	It is a markup language.	It is a combination of technologies for web development.
2	It is used to create static web pages.	It is used to create dynamic and interactive web pages.
3	Server-side scripting code is not required.	Server-side scripting code is required.
4	The document of HTML is stored with the .html or .htm extension.	The document of DHTML is stored with the .dhtm extension.
5	It does not need any database connectivity.	This language needs database connectivity because user responses need to be stored.
6	It creates simple images and menus.	It is used to create dynamic menus and animations
7	It does not require processing from the browser.	Processing is required by the browser.

Table 3: Comparison of HTML and DHTML

9.6 Features of DHTML

Following are the features of DHTML that make it the most widely used language for creating interactive pages:

1. **Dynamic styles in DHTML:** This helps in creating dynamic styles that can change the color, font, text size when the user requests for the change. Style includes positioning, indentation, visibility, etc.

2. **Changing the content dynamically**: This feature enables the designer to change the content of the elements after it loads onto the server. This is possible only with the help of the Document Object Model (DOM) that can access all the elements of the web page such as (<p>, <table>, <image>, etc.). The appearance of these elements can be changed easily without the need to insert or delete the element. For example, if a user wants to change the font size, color any of the elements on the web page by moving the cursor on that element. Then, this can be done with the help of JavaScript and CSS.

3. **Real-time Positioning**: The positioning of a single element or group of elements can be done with the help of DHTML on a web page. The real-time positioning can be performed in DHTML. All the elements have their transparent layer that allows the

elements to be made hidden, visible, or moved on the web page without any disturbance. This technique also helps in creating animations on the website.

4. **Downloading Fonts**: This feature allows the downloading of fonts from the server dynamically. If the font given in CSS by using font-family is not available on the system, then it will use the default font of the system. But with the help of this feature, the font that can be used by the designer is automatically downloaded on the system.

5. **Dynamic pages**: DHTML helps in creating dynamic web pages that can be used to create animations, navigation of elements on the web page, games, etc. By combining these languages make the website dynamic.

6. **Transitions and Filters:** Transitions are used to adjust the appearance of the elements. The images can switch from one to another either horizontally or vertically. Filters are the visual effects applied on the web pages like inner glow, outer glow, shadow, etc. These features are supported on internet explorer using CSS.

7. **Less server load:** Dynamic HTML helps the designer to create dynamic content on the client or the server. Dynamic content can be modified based on the responses entered by the user. It can alter the content dynamically on the web page. In this way, it will reduce the server load.

9.7 Working of DHTML

Dynamic HTML allows the designer to edit any page element, style, content, positioning at any time after loading the web page. It can modify the content without interrupting the browser services. It presents a formatted page and the user interacts with the page without downloading additional software from the server. For Example, the user can get an immediate response from the web page, just by click on any option with the mouse.

Three main components of DHTML are:

1. **Positioning:** By creating style blocks of content on a web page, and placing that block at any location on the browser.
2. **Style Update**: This can be made possible by updating the style and display on the browser.
3. **Event Handling**: It is used to handle the events that can be performed dynamically.

Here, JavaScript accesses an HTML document's property via DOM. It represents the attributes of HTML and CSS to JavaScript. In DHTML, different browsers support different versions of DOM.

9.8 Uses of DHTML

Following are the points that describe the uses of DHTML:

1. It helps in designing an interactive and animated web page that works in real time environment.

2. DHTML allows the user to add effects on the web pages.

3. DHTML helps in adding rollover buttons and drop down menus in the web pages.

4. DHTML is also used to create the browser related games.

5. DHTML allows the users ton animate the text and images in the web documents.

6. It is also used to add the ticker on various websites, which needs to refresh their content automatically.

Objective Type questions:

40. Which element is used as a form attribute in HTML?

m) Textbox
n) Radio button
o) checkbox
p) All of the above.

41. Which tag is used for creating the dropdown list?

m) <select>
n) <text>
o) <textarea>
p) <dropdown>

42. How can you select more than one option in the drop-down list?

m) By using multiple attribute in <select> tag.
n) By using multiple attribute in <option> tag.
o) By using multiple attribute in <text> tag.
p) None of the above.

43. CGI stands for

i) Common Gateway Interface
j) Common Gateway Internet
k) Central Gateway Internet
l) Central Gateway Interface

44. The program that can be used to open a web page is

k) Protocol
l) Web Browser
m) Search Engine
n) Web Server

45. The interactive documents can be created using:

k) Forms
l) Tables
m) Frames
n) none of the above

46. Client-side programming can be done on which computer

k) Server
l) Main-frame
m) Client
n) None of the above

47. Which attribute of the checkbox or radio button is used for default checking.

k) Check
l) Checked
m) Checked= True
n) None of the above

48. The tag used to identify the author's name:

k) Anchor tag
l) Title tag
m) Font tag
n) all of the above

Hands on Practice

Example 1:

```
<html>
<head>
<title> POST Method </title>
</head>
```

```
<body>

<h1>The form method="post" attribute</h1>

<form action="/action_page.php" method="post" target="_blank">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Last name:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>

<p>Click on the submit button, and the form will be submittied using the POST method.</p>

</body>
    </html>
```

Example 2:

```
<html>
<head>
<title> Registration Form </title>
</head>
<body>
<h3>Registration form</h3>
            <form>
               <label>Enter User name</label><br>
                 <input type="text" name="uname" value="USER"  disabled><br><br>
                 <label>Enter your Email address</label><br>
             <input type="email" name="email" placeholder="example@gmail.com"><br><br>
                 <label>Enter your password</label><br>
                 <input type="password" name="pass" placeholder="your password"><br><br>
                 <input type="submit" value="login">
        </form>

</body>
</html>
```

9.8 SUMMARY

This unit covers the important and advanced topics of client/server programming that shows the communication of the client's browser with the server. and the last part covers the Common Gateway Interface i.e., CGI. It acts as a middleware between the server and other sources of information that uses the GET and POST method to send data to the server. This unit includes the applications, advantages, and disadvantages of CGI. And also covers the Dynamic HTML, its features and working of DHTML.

UNIT 10 DOCUMENT OBJECT MODEL

Structure

10.0 INTRODUCTION

HTML was designed to create static pages whereas DHTML was designed to create dynamic pages. DHTML is a combination of HTML, CSS, JavaScript languages. This technique helps in the creation of dynamic pages that move the text across the page. This unit also covers the Document Object Model (DOM) which is a programming interface that allows the dynamic modification of content, styles, script, and structure without the need to close the document. And also includes structure, properties, and the levels of DOM. This unit also covers the events supported by DHTML, the rollover buttons, and the advantages and disadvantages of DHTML. The basic purpose of this unit is to implement animations on the web page and to create dynamic web pages that helps in creating a web site with moving effects or objects.

10.1 OBJECTIVES

The objective of this unit is to create an interactive and dynamic web page using the concept of DHTML. After completing this unit, learners will be able to understand the following concepts.

- Document Object Model (DOM)
- Scripting language
- Rollover buttons
- Events
- Moving objects with DHTML

This unit deals with designing the dynamic web page that helps in changing the content, layout, style dynamically on the browser without to need to exit the window.

10.2 DOCUMENT OBJECT MODEL (DOM)

The DOM i.e., Document Object Model is an interface that enables the dynamic modification of content, style using the scripting languages. By using DOM, the designer creates the document, adds, modify, deletes the contents. The DOM can represent the logical structure of documents and the way how these documents can be accessed and created.

DOM represents the structure into a hierarchical format that helps programmers and users easily accessing and updating the document. One can easily access and update tags, classes, elements, attributes using methods or commands provided by DOM. It is designed for any programming language such as JavaScript, VBScript, Python, etc.

10.2.1. Structure of Document Object Model (DOM):

Once the web page is loaded, it is visible on the browser as a visual construct. However, the internal structure is created by the browser in the form of a tree that is hidden in the browser.

DOM can be represented as a Tree or Forest i.e., more than one tree. Structural isomorphism is one of the most important properties of DOM. It means if the two implementations are managed for representing a document, it will create the same structure, with the same objects and shows the relationships between them. DOM can be represented as shown in Figure 1.

Figure 1: DOM Structure

DOM treats all the objects as scriptable objects on the web page. Scriptable objects can be associated with properties, events, methods. Here, the properties represent the characteristics of an object, the method can act on the particular object and the events are the actions on which the application responds.

10.2.2. Properties of DOM: The properties of DOM that can be accessed and modified are given below:

1. **Window:** It is always represented at the top of the hierarchy.
2. **Document**: Once the HTML page is loaded on the browser, it will be a document object.
3. **Form**: HTML web page contains form tags.
4. **Link:** Web pages contain link tags.
5. **Ancho**r: Web page contains <a href> tags.
6. **Form elements:** Form in HTML contains different controls such as textbox, radio buttons, checkbox, options button, submit, reset, etc.

These elements are represented in a tree-like structure as given below:

Example	DOM Model

<TABLE> <ROW> <TR> <TD> BCA </TD> <TD> BBA </TD> </TR> <TR> <TD> B.COM </TD> <TD> B.Sc. </TD> </TR> </ROWS> </TABLE>	**Figure 2: Example of the table in DOM**

Table 1: Example and output of DOM

10.2.3. Programming Interface:

- The Document Object Model can be accessed with JavaScript and with any other programming languages.
- All HTML elements in DOM are defined as objects.
- The programming interface is the properties and methods of each object.
- A property is a value that can be used to get or set like changing the content of an HTML element.
- A method is an action that helps in adding or removing controls from the web page.

10.2.4. Levels of DOM:

There are four different levels of DOM:

1. **DOM Level 0:** It was the first object-level specification that was supported by the earlier versions of browsers such as Netscape Navigator 3.0 and IE 3.0. This level was supporting only the limited number of objects such as windows, documents, history, location, etc. Level 0 was initially designed for form controls, validations, image rollovers, etc. It provides low-level interfaces for objects.

2. **DOM Level 1:** It is a World Wide Web Consortium representation that offers access to not only the basic elements instead every element of the web page. It can be described in two parts; one is CORE and the other is HTML. Here, CORE represents low-level interfaces that represent any structural document. and the HTML provides high-level interfaces that can represent any element of the HTML document. This level 1 can be supported by all the browsers such as IE 5+, Opera 7+, Netscape Navigator 6+, etc.

3. **DOM Level 2:** Level 2 is the more advanced version than level 1. It consists of 6 specifications: CORE2, VIEWS, EVENTS, STYLE, TRAVERSAL, and RANGE.

- **CORE2:** It extends the functionality of CORE that was specified by level1.
- **VIEWS:** It dynamically accesses and handles the document's content.
- **EVENTS:** These are the scripts that can be executed when the user reacts to the web page on the browser.
- **STYLE:** It allows to dynamically access and manipulates the style information given on the web page.
- **TRAVERSAL:** It traverses the document dynamically.
- **RANGE:** It dynamically identifies the range of content on the web page.

In this way, level 2 can manipulate the style sheets also and other advanced specifications as compared to level 1.

4. **DOM Level 3:** Level 3 of DOM is still in the development phase that includes 5 specifications: CORE3, LOAD & SAVE, VALIDATION, EVENTS, and XPATH.

- **CORE3:** It extends the functionality of CORE defined by the DOM level 2.
- **LOAD & SAVE:** It allows to dynamically load the XML document content in DOM and save the DOM document in XML document in serialization.
- **VALIDATION:** It validates the document during dynamic updations of content and structure.
- **EVENTS:** It extends the functionality specified in level 2.
- **XPATH:** It is a path language that accesses the DOM tree.

Let us consider an example to understand the internal working of DOM level 1 with the help of a hierarchical diagram as shown in Figure 3.

A node in DOM is represented by every HTML element. In Figure 3, the document, HTML, title, body, h1, p, document content, headings, paragraphs, all are nodes in DOM. In this HTML, body, head, h1, p are the element nodes and document content, headings, paragraphs, all are text nodes. The root should be the document that is a container for all the other nodes. With these nodes, there are some other nodes such as Comment nodes, attribute nodes, etc.

Example	DOM Model

<HTML> <HEAD> <TITLE> DOM Level 1 </TITLE> </HEAD> <BODY> </BODY> </HTML>	**Figure 3: Example of the Tree structure of HTML page**

Table 2: Example and output of DOM level 1

10.3 HTML and Scripting Access

The elements can be located in the tree either by using ID or by using the name attribute. The elements can be accessed and modify the content, style, structure of the web page with the help of the following techniques:

1. **getElementById () Method**: It works with the elements of the script to access the specific element of the web page. The element should be uniquely identified by the ID attribute:
   For example:
   <IMG ID="first" SRC="a.jpg">

This element can be referenced as document.getElementById("first").If no such ID exists, it returns NULL.
Let us consider another example, where getElementById () is used to calculate the square of a number given in the textbox. It returns the result in the alert box.

Example	Output

<html>
<head>
 	
<script type="text/javascript">
function getsquare(){
var number=document.getElementById
("num").value;
alert(number*number);
}
</script>
</head>
<body>
<form>
Enter No:<input type="text" id="num"
name="number"/>
 	
<input type="button" value="square"
onclick="getsquare()"/>
</form>
</body>
</html>
	Figure 4: Output of document.getElementById()

Table 3: Example and output of document.getElementById()

2. **getElementByName () Method**: This method returns the array of the elements referenced in the document with the matching attributes of ID and Name. ID should be unique, but the name of different elements can be the same on a web page.
Once the browser opens up, it shows the checkboxes that contain different values. This code is designed to click on the button that checked the checkboxes that contain the value "animals".The rest of the checkbox contains different values; thus, these are remained unchecked.These two methods i.e., getElementById() and getElementByName() are supported by all the web browsers.

Example	Output

```
<html>
<head>
<script>
function Function() {
 var x =
      document.getElementsByName("animal")
      ;
 var i;
 for (i = 0; i < x.length; i++) {
  if (x[i].type == "checkbox") {
   x[i].checked = true;
  } }}
</script>
</head>
<body>
Cats: <input name="animal" type="checkbox"
      value="Cats">
Dogs:  <input name="animal" type="checkbox"
      value="Dogs">
Cars: <input name="vehical" type="checkbox"
      value="Cars">
Bike: <input name="vehical" type="checkbox"
      value="Bike">
Activa: <input name="vehical" type="checkbox"
      value="Activa">
<br>
<br>
Click the on the button to check all checkboxes
      that contains the name attribute with the
      value "animal"
<button onclick="Function()">Try it</button>
</body>
</html>
```

Figure 5: Output of document.getElementByName()

Table 4: Example and output of document.getElementsByName()

10.4 Dynamic Content Changing

DHTML changes the content dynamically without the need to communicate with the server. By writing the script, one can modify the contents of a web page, places one element at the place of the other, For example, by clicking on an image, it changes the image and places another image, etc. It helps to create dynamic tables, controlling the content of tables, etc. There exist different methods in DHTML and these are given in Table 4.

Method	Description
document.write("content")	By using this method, one can dynamically change the content using the write() of the document object on the web page.
innerText()	It changes the content given within starting and ending elements.
outerText()	It changes all the content including the element.
innerHTML()	It allows changing the text and elements given in starting and ending tags.
outerHTML()	It allows changing all the text and elements given in starting and ending tags including the element.

Table 5: Basic Methods of Content changing

Handy Tips of Dynamic Content Changing:

- Using document.write method and document.getElementById

Example	Output
<HTML> <head> <title> DHTML </title> </head> <body> <script type="text/javascript"> document.write("DHTML-Dynamic Hypertext Markup Language"); </script> </body> </html>	**Figure 6: Output of document.write()**
<html> <head> <title> Example of getElementById </title> </head> <body> This text changes color when click on the following different buttons. <button onclick="change_Color('green');">	**Figure 7: Output of document.getElementById()**

Green </button>
<button onclick="change_Color('blue');">
Blue </button>
<script type="text/javascript">
 	
function change_Color(newColor) {
var element =
document.getElementById('demo').style.color
= newColor;
}
</script>
</body>
</html>	

Table 6: Example and Output of document.write()

10.5 Dynamic Styles

Dynamic styles are an important feature in DHTML. By using the style sheets, the designer can change the color, size, style, appearance, and formatting of elements without the need to add or remove the elements. It helps in creating a small and manipulate the document dynamically. It takes less time to implement dynamic styles.

With the help of CSS, one can easily apply the styles in different ways such as Inline, External, or on Embedded. Inline styles are the styles that can be applied to the individual elements in the document. For example, a user wants to change the style of heading by increasing the size whenever the user moves the cursor on the heading. This can be done with the help of Inline or External CSS in DHTML. External CSS can also work much like Inline, but the only difference is that it can be created separately and can be linked with the HTML document.

Handy Tips of Dynamic Styles:

Example	Output

<head>
<title>Dynamic Styles</title>
<script type = "text/javascript">
function givebackcolor()
{
var inputColor = prompt("Enter the name for the color " +
"for the background of this page", "");
 	
document.body.style.backgroundColor = inputColor;
}
</script>
</head>
<body onload = "givebackcolor()">
 Welcome to the website
</body> </html>
	Figure 8: Output of dynamic styles

Table 7: Example of dynamic styles

10.6 Roll Over buttons

This is the most common feature of DHTML is to create a dynamic image by using the rollovers. Rollovers mean the collection of images that changes when the user moves the mouse pointer over them. These are used for images that are linked. By using this feature, the current image or link can be highlighted with a different color, the border or the image can be changed completely.

An image can be turned to roll over by adding onMouseOver and onMouseOut events. The onMouseOver event can replace the image with highlighted image and the onMouseOut event returns the original image on the web page.

Handy Tips of Roll Over Buttons:

Example	Output
<html> <head> <title>JavaScript Image Rollovers</title></head> <body> <img src="banner.gif" name="image1"> </body> </html>	**Figure 9: Output of onMouseOver Event**
Figure 10: Output of onMouseOut Event	

Table 8: Example of Rollover buttons

In the above-given example, the property of src of image project is changed during onMouseOver and onMouseOut events. This is the simple and effective implementation to create a dynamic image rollover. This is done by having two images, that may be the same or different. One image is displayed after loading the page and the other one is displayed only when the user moves the mouse over the first image.

10.7 DHTML Events

An event defines the change in the occurrence of an object. To create the dynamic pages, it is necessary to add the events. By clicking on the particular element, an event fires, and the action is triggered on the browser. For example, a user clicks on the button, then the JavaScript code linked with that element is executed.

Some examples of events are:

1. Clicking on a button.
2. Form submission.
3. Loading an image or a web page

The events contain the information about:

- The event type i.e., it may be onmousedown, onkeypress, etc.
- Coordinates of the screen, which specifies where the mouse button is pressed.

The life cycle of an event can be defined as the action that starts the event and the response that is generated after the event by the handler. The life cycle consists of the following phases whenever the user clicks on the button. The event starts from the document, moves to the HTML, then onto the body section, and finally reaches the button as shown in Figure 10.

Figure 11: Life cycle of clicking on a button

In this, the conceptual DOM is divided into three phases,

1. **Capture Phase:** This phase includes all the Documents, HTML, and BODY tags. All the elements from the document to the target are covered in this phase.
2. **Target Phase:** This phase occurs after the event is fired. An example of this phase is clicking on a button.
3. **Bubbling Phase:** It consists of all the DOM components from the beginning to the ending of the target.

A list of events that are used in DHTML to make interactive web pages.

Event	Description	Examples
onblur	On leaving the HTML object, this event occurs.	<INPUT TYPE="text" name=" year " onBlur=" window.status=('The value entered for the year is correct.'); return true; ">
onchange	On changing or updating the value.	<INPUT TYPE=" text " name=" year " onChange="window. status=('The value of the year is changed.'); return true; ">
Onerror	Alerts is given if any error found.	<img src="a.jpg" onerror="alert('The image is not loaded')">
onclick	On clicking any object.	<button onclick="alert('you may leave the site')">Click me</button>
ondblclick	On double clicking any object.	<button ondbclick="alert('you may leave the site')">Click me</button>
onFocus	On focusing on any object. It is just opposite to onblur.	function callFunction(a) { a.style.background = "red";} <input type="text" onfocus="callFunction(this)">
onLoad	On loading the form or object.	<body onload="alert('Welcome to the webiste'); ">
onMouseDown	On pressing the button of mouse.	 click here
onMouseMove	On moving the cursor of mouse on object.	 click here

onMouseOver	On moving the cursor of mouse over object.	 click here
onMouseOut	On moving the cursor of the mouse out of an object.	 click here
onMouseUp	On releasing the mouse button over an object.	 click here
onSubmit	On submitting the form's data.	Function f1() {alert("The form data is submitted"); } <form onsubmit="f1()">
onunload	On closing the browser window.	<body onunload="alert('Document unloaded'); ">

Table 9: Events in DHTML

Handy Tips of Events:

Example of the onclick event	**Output**
<html> <head> <script type = "text/javascript"> function wel() { alert("Welcome") } </script> </head> <body> Click on button to display welcome <form> <input type = "button" onclick = "wel()" value =" welcome" /> </form> </body> </html>	**Figure 12: Output of onclick event**
Example of the onsubmit event	**Output**
<html> <head>	

<title>
Example of onsubmit event
</title>
</head>
<body>
<form onsubmit="Submit_Form()">
<label> Enter your name: </label>
<input type="text">
<label> Enter your Roll no: </label>
<input type="Number">
<input type="submit" value="submit">
</form>
<script type="text/javascript">
function Submit_Form()
{
alert(" Your form is submitted");
}
</script>
</body>
</html>
	Figure 13: Output of onsubmit event

Table 10: examples of Events

10.8 Moving Objects with DTML

An object is said to be moving when it will change its position, thus it will be characterized by its position, its direction, and its movement from one location to another location.

Motion can be found in so many real examples such as:
1. A ball is thrown upward.
2. A flying airplane.
3. Running animals.
4. Pendulum moving from one direction to another direction.

10.8.1 Examples of moving objects:

Motion can be represented in various ways

1. **One dimensional**: Motion of a body along the straight line.

Figure 14: Example of moving object

2. **Periodic Motion**: The object will move after a fixed period.

Figure 15: Example of moving object in periodic motion

3. **Circular Motion**: The object will move its body uniformly in a circular way.

Figure 16: Example of moving object in a circular motion

10.8.2. Moving Object Style:

Moving object positions can be represented in the following ways:
1." position: relative".
2." position: absolute".
Example:

Example	Output

<html>
<style>
#container1 {
width: 400px;
height: 400px;
position: relative;
background: pink;
}
#animate {
width: 100px;
height: 50px;
position: absolute;
background-color: blue;
}
</style>
<body> <button onclick="Move()">Click Me</button>
 	
<script>
var id = null;
	Figure 17: Output of moving object in the middle
	
	Figure 18: Output of moving object at the end

function Move() {
  var elem = document.getElementById("animate");
  var pos = 0;
  clearInterval(id);
  id = setInterval(frame, 5);
  function frame() {
  if (pos == 380) {
  clearInterval(id);
  } else {
  pos++;
  elem.style.top = pos + "px";
  elem.style.left = pos + "px";
  } }}
</script></body></html>	

Table 11: Example and output

10.9 Advantages of DHTML

- The size should be less of these files as compared to other files that are created using software like flash and downloading is fast.
- It is supported by all the major browsers.
- It should be viewed and changed easily; no extra plugins are required for browsing the DHTML pages on the browser.
- Due to its dynamic nature, it is possible to change the content or any element on the browser after being loaded. No separate pages are created for the modification of elements or styles.
- On static web pages, the content cannot be changed and it contains only limited data. But on dynamic HTML, it can handle more content on the page simultaneously.

10.10 Ramification of DHTML

- Not all browsers can handle DHTML. Most recent browsers such as Netscape Navigator 6, Internet Explorer 5.5, Opera 5 support these documents.
- To start designing web pages using DHTML, one should know the languages like HTML, CSS, JavaScript.
- If the code is running on one browser, it may not be necessary that it works on other browsers in the same way.
- DHTML needs some tools and software that are expensive for its functionality. for example, Dreamweaver, the latest text editor, along with the cost of transferring from HTML to DHTML.

Check Your Progress 1

4. State whether True or False:
    i. DHTML is a combination of HTML, CSS and JavaScript.

ii. DHTML helps to make the web pages attractive and dynamic.
iii. DOM stands for Document Object Manipulation.
iv. document. write is used to access the element in DHTML.
v. Changing the value of an object can be fired by an onchange event.
vi. onreset event fires when a user submits the form.

5. Define DHTML.
6. What do you mean by DOM and explain levels of DOM.?
7. Explain rollover buttons with example.
8. What are the different ways to access the HTML elements?
9. Define events. Types of event handlers in detail with example.
10. What is the current state of Document Object Model (DOM) implementation?
11. Which programming language is used for Document Object Model (DOM) ?
12. Explain the difference between DHTML and DOM?
13. What is the relationship between style sheet and DOM?

Objective Type questions:

49. Why the attribute property is avoided in HTML DOM?

q) Attributes do not have attributes.
r) Found Unnecessary.
s) Attributes contain attributes.
t) It is considered Irrelevant.

50. In HTML DOM, how everything is to be treated?

q) Attribute
c) Elements
r) Node
d) Arrays

51. The objects in HTML DOM are organized in the form of:

q) Queue
b) Hierarchy
r) Stack
d) Class wise

52. DOM stands for

m) Dynamic Object Model
n) Document Object Model
o) Distributed Object Model
p) None of the above

53. The objects used to pass the arguments for handling the keyup, keydown, and key press events.

o) Keyboard Events
b) Key Events
p) Mouse Events
d) Alphabet Events

54. A simple text file that tells the browser what to cache is known as

o) Output Files
p) Input Files
c) A Manifest File
d) HTML File

55. The HTML tag that can be used to define a client side script is given as
a)<script>
b)Both a and c
c) <unscript>
d) None of the above

56. Which technology is used by the DHTML to create dynamic content on a web page?

o) AJAX
p) JavaScript
c) CSS
d) CSS and JavaScript

57. What does the document.getElementById (" Intro") function returns if intro does not exist?

o) Return false
p) Return True
q) Return 0
r) Create an element with id= intro and return that element to the function.

10.11 Summary

This unit covers the advanced topics to create dynamic web pages. The combination of HTML, CSS, JavaScript, Document Object Model (DOM) can be used in DHTML. It helps in creating animations and moving the content around the browser several times. HTML can be used to design web pages, CSS is used for giving unique styles to the website, JavaScript allows the user to add events or animations to create dynamic and attractive websites. This unit also covers the Document Object Model (DOM) and the levels of DOM with the events associated with DHTML. From this unit, learners will able to understand how the objects and content move across the browser. This topic also covers the events, the Life cycle of events, and the events associated with the particular object that handles the events.