



WEB APPLICATION DEVELOPMENT

20A05304



Prepared

By

Dr Nageswara Rao Putta

Dean – Admissions & Student Services

Associate Professor, Dept.of CSE

Ph: 9550195534

OCTOBER 12, 2022

VEMU INSTITUTE OF TECHNOLOGY

P KOTHAKOTA, NEAR PAKALA, ANDHRA PRADESH

Module - 1:

HTML: What is a browser? What is HTML? Elements and Tags, Basic HTML5 structure, Metadata, <title>, Adding favicon, Comments, headings.

Task: Create a Basic HTML document

Module - 2:

HTML (continued): Block-Level Elements & Inline Elements, Links (Understand Absolute vs Relative paths), Lists, Images, iframe (embed YouTube video)

Task: Create your Profile Page

Module - 3:

HTML (continued): Tables: <table>, <tr>, <th>, <td>, Attributes for each Table element

Task: Create a Class Timetable (to merge rows/columns, use rowspan/colspan)

Module - 4:

HTML (continued): Form Elements: <input>, <select>, <textarea>, <button>, Attributes for each Form element

Task: Create a Student Hostel Application Form

Module - 5:

Cascading Style Sheets (CSS): CSS Properties, Types of CSS, Selectors, box model, Pseudo-elements, z-index

Task: Make the Hostel Application Form designed in Module -4 beautiful using CSS (add colors, backgrounds, change font properties, borders, etc.)

Module - 6:

Bootstrap - CSS Framework: Layouts (Containers, Grid system), Forms, Other Components

Task: Style the Hostel Application Form designed in Module-5 still more beautiful using Bootstrap CSS (Re-size browser and check how the webpage displays in mobile resolution)

Module - 7:

HTTP & Browser Developer Tools: Understand HTTP Headers (Request & Response Headers), URL & its Anatomy, Developer Tools: Elements/Inspector, Console, Network, Sources, performance, Application Storage.

Task: Analyse various HTTP requests (initiators, timing diagrams, responses) and identify problems if any.

Module - 8:

Javascript: Variables, Data Types, Operators, Statements, Objects, Functions, Events & Event Listeners, DOM.

Task: Design a simple calculator using JavaScript to perform sum, product, difference, and quotient operations:

Module - 9:

Dynamic HTML with JavaScript: Manipulate DOM, Error Handling, Promises, async/await, Modules.

Task: Design & develop a Shopping Cart Application with features including Add Products, Update Quantity, Display Price(Sub-Total & Total), Remove items/products from the cart.

Module - 10:

JQuery - A Javascript Library: Interactions, Widgets, Effects, Utilities, Ajax using JQuery.

Task: Validate all Fields and Submit the Hostel Application Form designed in Module-6 using JQuery

Module - 11:

Google Charts: Understand the Usage of Pie chart, Bar Chart, Histogram, Area & Line Charts, Gantt Charts. Task: Develop an HTML document to illustrate each chart with real-time examples.

Module - 12:

Open Source CMS (Content Management System): What is a CMS?, Install CMS, Themes, Plugins.

Task: Develop an E-learning website using any CMS(for example WordPress)

Module - 1:

HTML: What is a browser? What is HTML? Elements and Tags, Basic HTML5 structure, Metadata, <title>, Adding favicon, Comments, headings.

Task: Create a Basic HTML document

What is a browser?

A browser is an application program that provides a way to look at and interact with all the information on the World Wide Web. This includes Web pages, videos and images. A Web browser is a client program that uses HTTP (Hypertext Transfer Protocol) to make requests of Web servers throughout the Internet on behalf of the browser user. Most browsers support e-mail and the File Transfer Protocol (FTP),

Web browsers work as part of a client/server model. The client is the browser which runs on the user's device and makes requests to the Web server, while the server-side is the Web server which sends information back to the browser. The browser then interprets and displays the information on the user's device.

					
Firefox which was developed by Mozilla.	Google Chrome Its was introduced by Google's foundation	Internet Explorer It was a product of Microsoft	Safari a browser for Apple computers and mobile devices.	Opera Telenor introduced this web browser	Microsoft Edge Used a placement for Internet Explorer, with Windows 10.

M.1.1: Popular Web Browsers Nowadays

What is HTML?

HTML is a markup language used to describing the simplest form of webpages. It is the main markup language for webpages. HTML is an abbreviation and stand for Hyper Text Markup Language. When a web browser such as Internet Explorer, Fire Fox or Chrome displays a webpage, what it is actually doing is reading and interpreting a HTML document. This document could be as short as 10 lines or as long as several hundred, the browser does not care – all it does is read your document.

The parent of all markup languages is SGML. SGML stands for Standard Generalized Markup Language.

HTML was first created by Tim Berners-Lee and others starting in 1989. It stands for Hyper Text Markup Language. Hypertext means that the document contains links that allow the reader to jump to other places in the document or to another document altogether. The latest version is known as HTML5.

A Markup Language is a way that computers speak to each other to control how text is processed and presented. To do this HTML uses two things: tags and attributes.

HTML is not a case sensitive markup language that mean treats both lower case and upper case as same. HTML is compatible with almost all popular web browsers available in market now. We can create static webpages with the help of HTML elements and tags.

We can incorporate Java Script, CSS, XML in HTML document.

HTML Elements and Tags

Tags are used to mark up the start of an HTML element and they are usually enclosed in angle brackets. An example of a tag is: `<h4>`. Most tags must be opened `<h4>` and closed `</h4>` in order to function.

Elements are the fundamentals of HyperText Markup Language (HTML). Each HTML document is made of elements that are specified using tags.

HTML elements and HTML tags are often confused. The tags are used to open and close the object, whereas the element includes both tags and its content. Let's consider an example with the `<h1>` tag: `<h1>` Title of the document `</h1>` - is an element, and `<h1>`, `</h1>` - are tags.

HTML elements can come in pairs or be empty. The paired elements have an opening (`<>`) and (`</>`) closing tags, and the content of the element is placed in between them.

The empty elements have no closing tags.

The empty elements in HTML are: `<area>`, `<base>`, `
`, `<col>`, `<embed>`, `<hr>`, ``, `<input>`, `<keygen>`, `<link>`, `<meta>`, `<param>`, `<source>`, `<track>` and `<wbr>`.

HTML Attributes

Attributes contain additional pieces of information. Attributes take the form of an opening tag and additional info is placed inside.

An *example of an attribute* is:

```

```

In this case, the image source (src) , width, height and the alternative text (alt) are attributes of the tag.

Rules To Remember

The majority of tags must be opened (<tag>) and closed (</tag>) with the element information such as a title or text resting between the tags.

When using multiple tags, the tags must be closed in the order in which they were opened.

For example:

```
<p><i> <strong><em>This is sample HTML document !</em></strong></i></p>
```

HTML Editors

There are many choices on the market. Here are a handful of the most popular:

Sublime Text 3

we will use the Sublime Text 3 as it is free and also offers cross-platform support for Windows, Mac, and Linux users.

Notepad ++

Another common choice for HTML and other language coders is Notepad ++. It is a tiny program to download and perform the functions you need for writing clean code.

Komodo Edit

Komodo Edit is one of two editors released by the same label. They offer a simple, open-source editor with a variety of extensions and language support.

Do not use Microsoft Word or any other word processor when writing HTML code, only an HTML editor or at the very least, your machine's built-in notepad, is suitable for the task.

Secondly, ensure that you've installed a number of different browsers such as Chrome and Firefox in order to preview your upcoming creation.

Basic Construction of an HTML Page

These tags should be placed underneath each other at the top of every HTML page that you create.

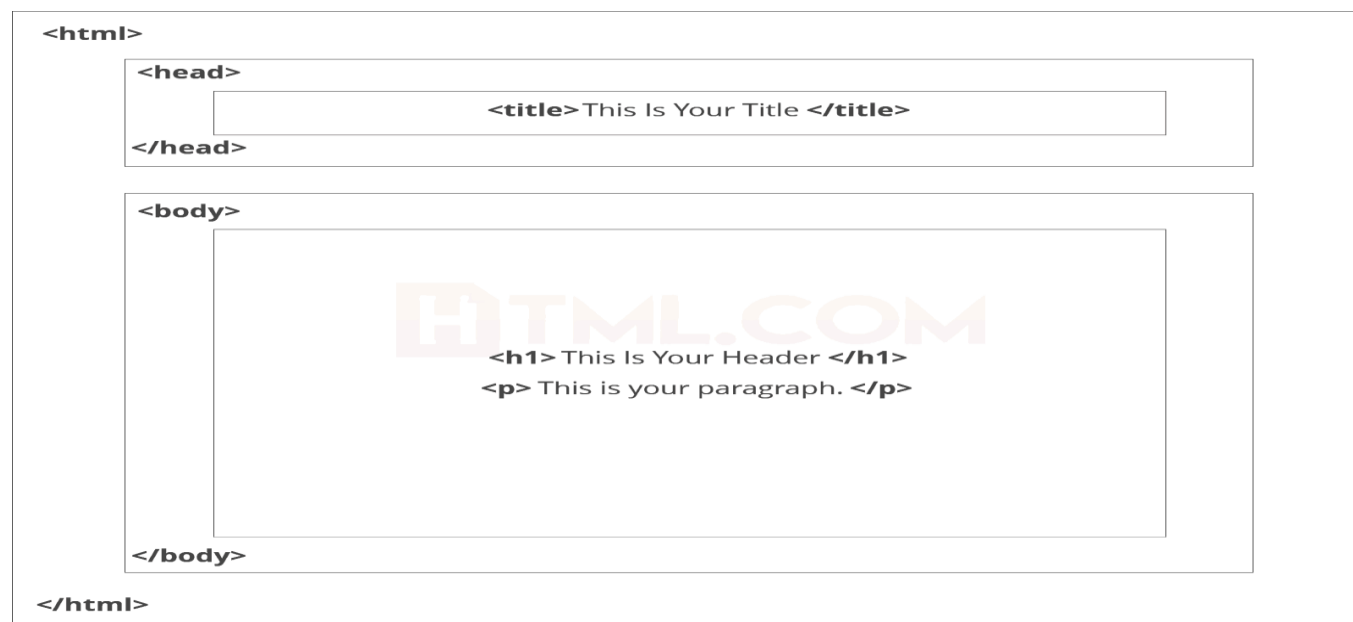
`<!DOCTYPE html>` — This tag specifies the language you will write on the page. In this case, the language is HTML 5.

`<html>` — This tag signals that from here on we are going to write in HTML code.

`<head>` — This is where all the metadata for the page goes — stuff mostly meant for search engines and other computer programs.

`<body>` — This is where the content of the page goes.

HTML Structure



Inside the <head> tag, there is one tag that is always included: <title>

<title>

This is where we insert the page name as it will appear at the top of the browser window or tab.

<meta>

This is where information about the document is stored: character encoding, name (page context), description.

Adding Content

The HTML <body> is where we add the content which is designed for viewing by human eyes.

This includes text, images, tables, forms and everything else that we see on the internet each day.

Add HTML Headings to Your Web Page

- <h1>
 - <h2>
 - <h3>
 - <h4>
 - <h5>
 - <h6>

<h1> is the largest heading and <h6> is the smallest heading

Add Text In HTML

Adding text to our HTML page is simple using an element opened with the tag <p> which creates a new paragraph.

Element	Meaning	Purpose
<code></code>	Bold	Highlight important information
<code></code>	Strong	Similarly to bold, to highlight key text
<code><i></code>	Italic	To denote text
<code></code>	Emphasised Text	Usually used as image captions
<code><mark></code>	Marked Text	Highlight the background of the text
<code><small></code>	Small Text	To shrink the text
<code><strike></code>	Striked Out Text	To place a horizontal line across the text
<code><u></code>	Underlined Text	Used for links or text highlights
<code><ins></code>	Inserted Text	Displayed with an underline to show an inserted text
<code><sub></code>	Subscript Text	Typographical stylistic choice
<code><sup></code>	Superscript Text	Another typographical presentation style

Add Links In HTML

The internet is made up of lots of links.

Almost everything you click on while surfing the web is a link **takes you to another page** within the website you are visiting or to an external site.

Links are included in an attribute opened by the `<a>` tag.

The Anchor Tag

The <a> (or anchor) opening tag is written in the format:

Click here to go to PNR BTech Blogger

The first part of the attribute points to the page that will open once the link is clicked.

Meanwhile, the second part of the attribute contains the text which will be displayed to a visitor in order to entice them to click on that link.

If you are building your own website then you will most likely host all of your pages on professional web hosting. In this case, internal links on your website will Click Here.

Metadata

Describe metadata within an HTML document

The <meta> tag defines metadata about an HTML document. Metadata is data (information) about data.

<meta> tags always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings.

Metadata will not be displayed on the page, but is machine parsable.

Metadata is used by browsers (how to display content or reload page), search engines (keywords), and other web services.

There is a method to let web designers take control over the viewport (the user's visible area of a web page), through the <meta> tag

Example:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="description" content="Free Web tutorials">
  <meta name="keywords" content="HTML,CSS,XML,JavaScript">
  <meta name="author" content="John Doe">
```

```
<meta http-equiv="refresh" content="30">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
<p>All meta information goes in the head section...</p>
</body>
</html>
```

Explanation:

Meta Tag Name	Description
<meta charset="UTF-8">	Specifies the character encoding for the HTML document
<meta name="description" content="Free Web Technologies Tutorials">	Define a description of your web page
<meta name="keywords" content="HTML, CSS, XML, JavaScript,AJAX,PHP">	Define keywords for search engines
<meta name="author" content="P J Nidhi">	Define the author of a page
<meta http-equiv="refresh" content="30">	Refresh document every 30 seconds
<meta name="viewport" content="width=device-width, initial-scale=1.0">	Setting the viewport to make your website look good on all devices

Setting the Viewport

The viewport is the user's visible area of a web page. It varies with the device - it will be smaller on a mobile phone than on a computer screen.

You should include the following <meta> element in all your web pages:
<meta name="viewport" content="width=device-width, initial-scale=1.0">

This gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

Adding Favicon

A favicon is a small file containing the one or more icons which are used to represent the website or a blog. It is also known as a tab icon, website icon, URL icon, or a bookmark icon.

This icon is actually displayed on the address bar, browser's tab, browser history, bookmark bar, etc. The image of a favicon is in .ico file format. There are various file formats, but .ico format is supported by all the browsers.

Create a Favicon

- Click on the following URL, to create the favicon: **<https://www.favicon.cc/>**
- Once the favicon is successfully created, we can download the favicon by clicking on the download favicon option.
- After the downloading, a favicon with the name favicon.ico is available in the file system drive.

Insert the Favicon in HTML file

Open the HTML file. Then use the following syntax to insert the favicon in the HTML file.

1. `<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">`
2. We have to use the above syntax in the tag of our html file. Then save the file.
3. Now. Open the HTML file in any browser. We can see the icon on the web page.

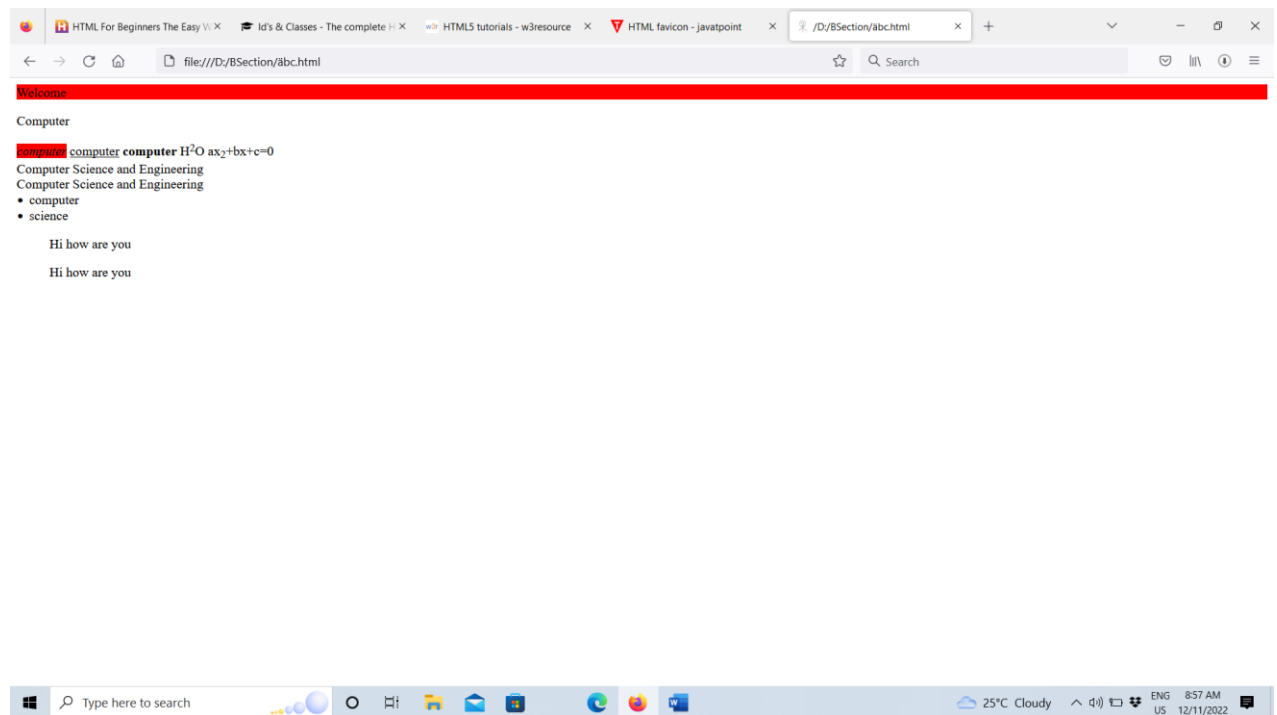
Example

```
<html>
<head>
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
</head>
<title> Example of favicon</title>
```

```
<body>
<br>
<br>
<p align="center">

</p>
</body>
</html>
```

Output



Comments

The comment tag is used to insert comments in the source code. Comments are not displayed in the browsers. The Web Browsers couldn't parse comments lines but they improve program understandability.

You can use comments to explain your code, which can help you when you edit the source code at a later date. This is especially useful if you have a lot of code.

The HTML comments are used to indicate sections of a document or insert notes explaining the code. They help understand the code and increase its readability. The comment tag can also be used for excluding temporary code blocks instead of deleting them.

For defining HTML comments we use the `<!-- ... -->` tag. Browsers ignore this tag and do not show its content to the user.

Only a few browsers support the comment tag for commenting a part of an HTML code.

Example

```
<!DOCTYPE html>
<html>
<body>

<!-- This is a comment -->
<p>This is a paragraph.</p>
<!-- Comments are not displayed in the browser -->

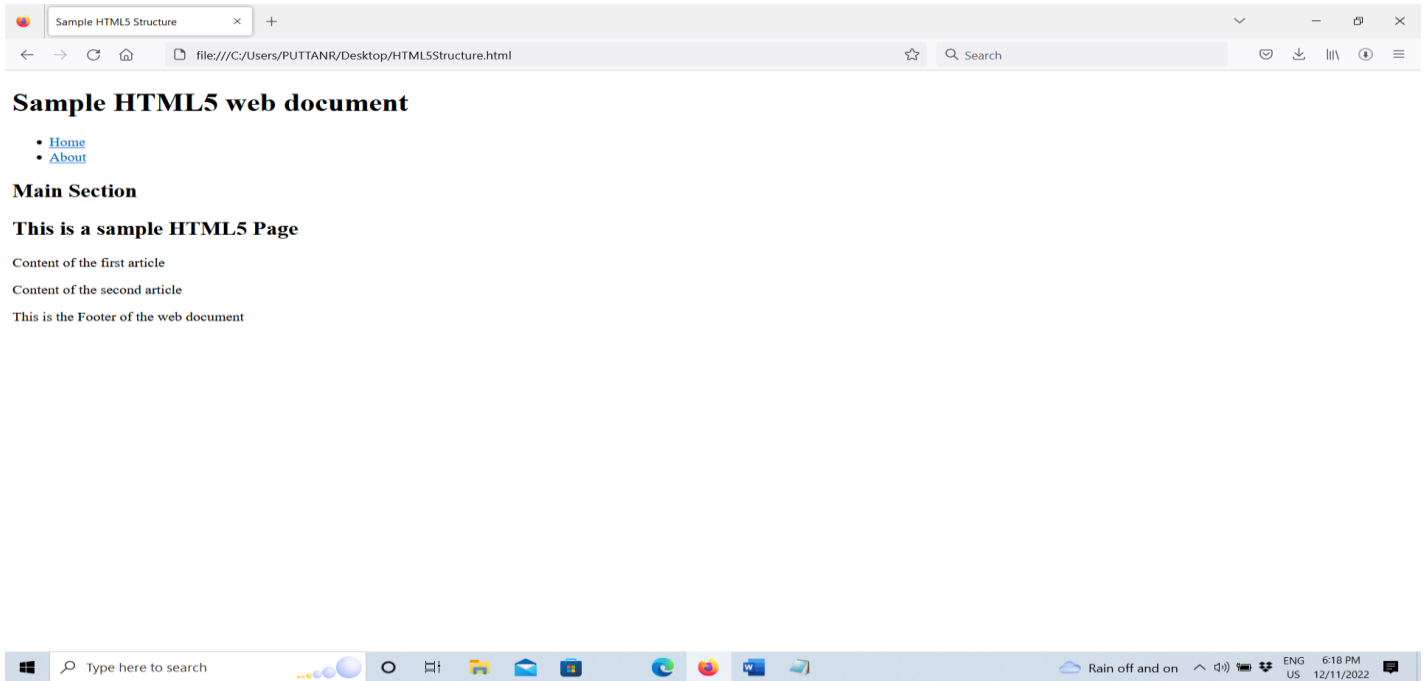
</body>
</html>
```

Basic HTML5 Structure

```
<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset=utf-8>
<title>Sample HTML5 Structure</title>
</head>
<body>
<div id="container">
<header>
<h1>Sample HTML5 web document</h1>
<nav>
```

```
<ul>
<li><a href="#">Home</a></li>
<li><a href="#">About</a></li>
</ul>
</nav>
</header>
<section>
<hgroup>
<h1>Main Section</h1>
<h2>This is a sample HTML5 Page</h2>
</hgroup>
<article>
<p>Content of the first article</p>
</article>
<article>
<p>Content of the second article</p>
</article>
</section>
<footer>
<p>This is the Footer of the web document</p>
</footer>
</div>
</body>
</html>
```


Output:



Task: Basic HTML5 Document

Source Code

```
<!DOCTYPE html>

<html>

  <head>

    <title>Title of the document</title>

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

  <style>

    html,

    body {

      height: 100%;

    }

    body {

      display: flex;

      flex-wrap: wrap;

      margin: 0;
```

```
}  
.header-menu,  
footer {  
    display: flex;  
    align-items: center;  
    width: 100%;  
}  
.header-menu {  
    justify-content: flex-end;  
    height: 60px;  
    background: #1c87c9;  
    color: #fff;  
}  
h2 {  
    margin: 0 0 8px;  
}  
ul li {  
    display: inline-block;  
    padding: 0 10px;  
    list-style: none;  
}  
aside {  
    flex: 0.4;  
    height: 165px;  
    padding-left: 15px;  
    border-left: 1px solid #666;  
}  
section {
```

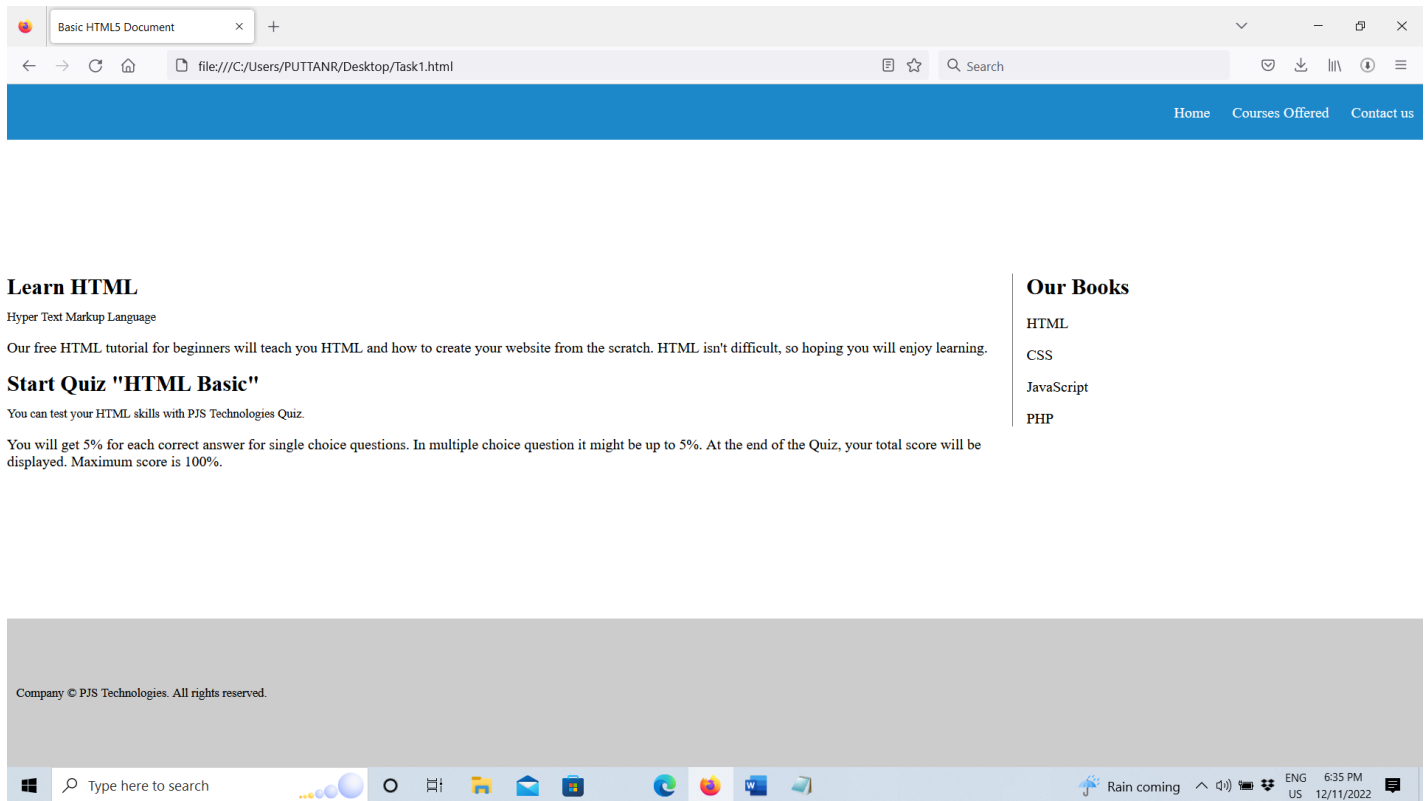
```
flex: 1;
padding-right: 15px;
}
footer {
padding: 0 10px;
background: #ccc;
}
</style>
</head>
<body>
<header class="header-menu">
<nav>
<ul>
<li>Home</li>
<li>Courses Offered</li>
<li>Contact us</li>
</ul>
</nav>
</header>
<section>
<article>
<header>
<h2>Learn HTML</h2>
<small>Hyper Text Markup Language</small>
</header>
<p>Our free HTML tutorial for beginners will teach you HTML and how to
create your website from the scratch. HTML isn't difficult, so hoping you will enjoy
learning. </p>
</article>
```

```
<article>
  <header>
    <h2>Start Quiz "HTML Basic"</h2>
    <small>You can test your HTML skills with PJS Technologies Quiz.</small>
  </header>

  <p>You will get 5% for each correct answer for single choice questions. In
multiple choice question it might be up to 5%. At the end of the Quiz, your total
score will be displayed. Maximum score is 100%.</p>

</article>
</section>
<aside>
  <h2>Our Books</h2>
  <p>HTML</p>
  <p>CSS</p>
  <p>JavaScript</p>
  <p>PHP</p>
</aside>
<footer>
  <small>Company © PJS Technologies. All rights reserved. </small>
</footer>
</body>
</html>
```

Output:



Explanation:

A basic HTML page starts with the Document Type Declaration or doctype. That is a way to inform the browser what type of document it is. The doctype is always the first item at the top of any HTML file. Then sections and subsections come, each possibly has its heading and subheading. These heading and sectioning elements help the reader to perceive the content meaning.

The <html> element

The <html> element follows the doctype information, which is used to inform the browser that this is an HTML document. You can use the lang attribute with the "en" value to specify that the document is in English.

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

</html>
```

The <head> section

The next part is the <head> section. The <head> element contains metadata (document title, character set, styles, links, scripts), specific information about the web page that is not displayed to the user.

The <meta> element is used to specify the metadata to provide browsers and search engines with technical information about the web page.

To define the character encoding for the document, you need to set the charset attribute with the "utf-8" value in nearly all cases. UTF-8 is the default character encoding for HTML5.

Use the <title> element to define the title of your document.

Next is the <link> element which sets the relationship between the current document and the external resource.

Required attributes for the <link> element are rel, href and type.

```
<head>
  <title>W3Docs - Learn Programming Languages Online.</title>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <meta name="Author" content="W3docs">
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

The <body> section

The <body> of a document contains the content of the document. The content may be presented by a user agent in different ways. E.g., the content can be text, images, links, colors, graphics, etc.

```
<body>
  ...
</body>
```

Between the body tags, there can be different elements, to which you can give style by using CSS properties. Just add an id or class selector to your HTML element and in the <style> section mention your preferred styling options (color, size, font, etc.).

```
<style>
  .header-style {
    color: blue;
  }
</style>

<body>
  <header class="header-style"></header>
</body>
```

The <script> element

In HTML5, the <script> tag is put to correspond the practices for embedding JavaScript. For example, it is related to the page loading speed.

```
<script src="js/scripts.js"></script>
```

Headings (h1-h6 elements)

The <h1>-<h6> heading elements are used to give a short description about the coming section. <h1> is considered to be the most important, and <h6> the least important one.

```
<body>
  <h1>First-level heading</h1>
  <h2>Second-level heading</h2>
</body>
```

The <header> element

Use the <header> element to define a header for the document or section. It usually contains a logo, search bar, navigation links, etc.

```
<header></header>
```


The <nav> element

The <nav> element defines a block of navigation links, either within the current document or to other documents. Note, that not all links in the HTML document can be placed inside the <nav> element; it can only include major navigation blocks.

```
<header>
  <nav>
    <ul>
      <li>Menu item</li>
      <li>Menu item</li>
      <li>Menu item</li>
    </ul>
  </nav>
</header>
```

The <article> element

The <article> element is used to define an independent, self-contained content (articles, blog posts, comments, etc.). The content of the element has its meaning, and it is easily differentiated from the rest of the webpage content.

```
<article>
  <p>Text of the article</p>
</article>
```

The <section> element

The <section> element is used to group standalone sections within a webpage containing logically connected content (news block, contact information, etc.).

```
<section>
  <h1>Header</h1>
  <p>Some paragraph for example</p>
</section>
<section>
  <h2>Header 2</h2>
  <p>Another paragraph for example.</p>
</section>
```

Module - 2:

HTML (continued): Block-Level Elements & Inline Elements, Links (Understand Absolute vs Relative paths), Lists, Images, iframe (embed YouTube video)

Task: Create your Profile Page

Block-level and Inline HTML Elements

All HTML elements are divided into two categories: block-level elements and inline elements.

Block-level elements are those that structure the main part of your web page, by dividing your content into coherent blocks. They always start on a new line and take up the full width of a page, from left to right; they also can take up one line or multiple lines and have a line break before and after the element. Block-level elements can include both block-level & inline elements.

Block-level elements are <address>, <article>, <aside>, <blockquote>, <canvas>, <dd>, <div>, <dl>, <dt>, <fieldset>, <figcaption>, <figure>, <footer>, <form>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <header>, <hr>, , <main>, <nav>, <noscript>, , <output>, <p>, <pre>, <section>, <table>, <tfoot>, and <video>.

All block-level elements have opening and closing tags.

Example:

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Block-Level Elements</title>
```

```
  </head>
```

```
  <body>
```

```
    <footer>
```

```
      <p>Author: PJS Technologies team</p>
```

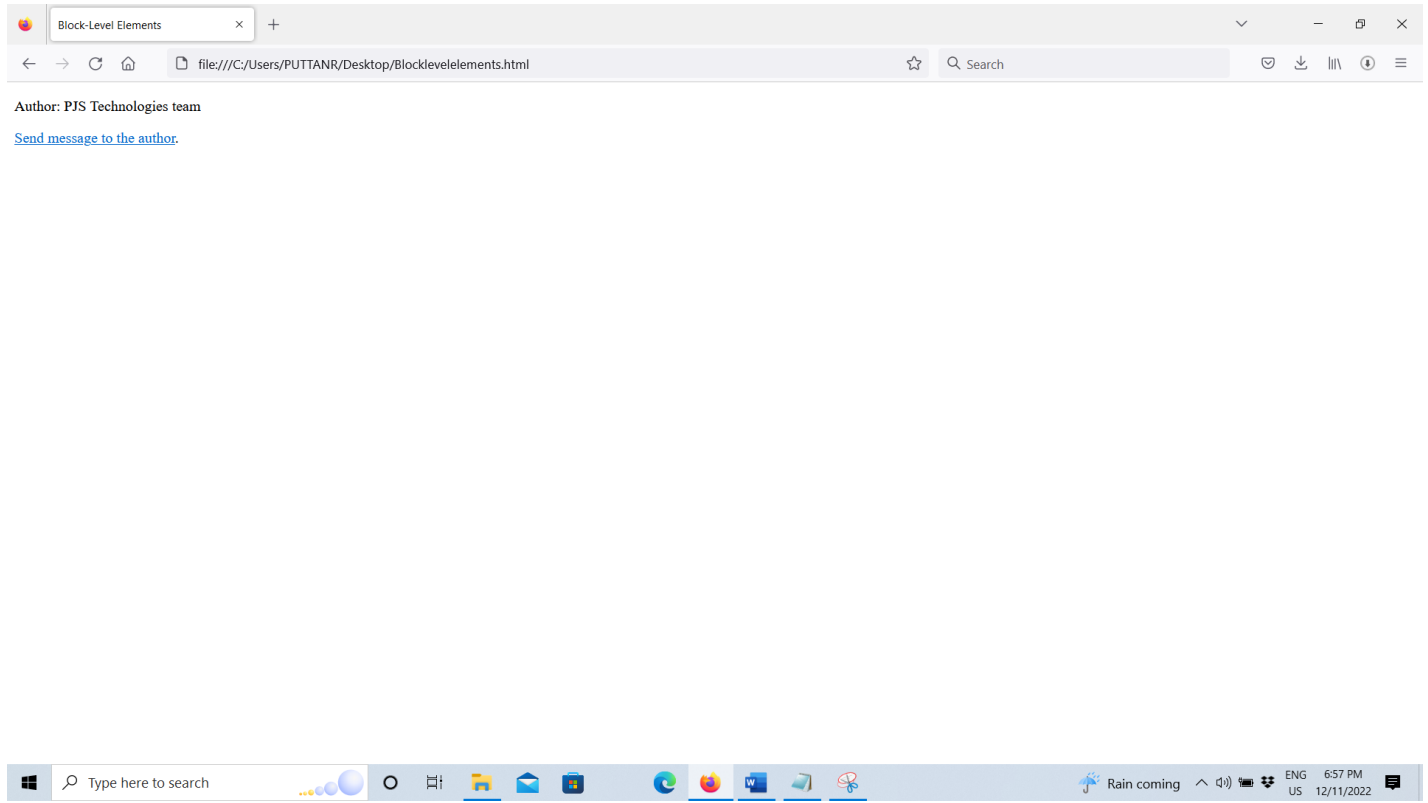
```
      <p><a href="mailto:info@pjstechnologies.com">Send message to the author</a>.</p>
```

```
    </footer>
```

```
  </body>
```

```
</html>
```

Output:



Inline Elements

Inline elements are used to distinguish part of a text, to give it a particular function or meaning. Inline elements usually include a single or a few words. They do not cause a line break and do not take up the full width of a page, only the space bounded by its opening and closing tag. The inline elements are usually used within other HTML elements.

Inline elements are: <a>, <abbr>, <acronym>, , <bdo>, <big>,
, <button>, <cite>, <code>, <dfn>, , <i>, , <input>, <kbd>, <label>, <map>, <object>, <q>, <samp>, <script>, <select>, <small>, , , <sub>, <sup>, <textarea>, <time>, <tt> and <var>.

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
<title>Inline Elements</title>
</head>
<body>
  <h1>Title of the document</h1>
  <p>The first paragraph</p>
  <p><strong>This part of the sentence</strong> has a strong emphasis and is
displayed as bold. <i> This is displayed in Italic. </i></p>
</body>
</html>
```

Output



Title of the document

The first paragraph

This part of the sentence has a strong emphasis and is displayed as bold.*This is displayed in Italic.*



Links (Understand Absolute vs Relative paths)

Websites contain different types of links that take you directly to other pages or allow to navigate to a particular part of the page. The links in HTML are called hyperlinks. They are defined using the <a> tag.

Hyperlinks are applied to a phrase, a word, an image or any HTML element.

The default color of links in HTML is:

- unvisited links: underlined and blue
- visited links: underlined and purple
- active links: underlined and red

This is default style of links, but you can remove underline or change the color of the links using CSS styles.

Syntax

The <a> tag comes in pairs, the opening <a> tells where the link should start and the closing indicates where the link ends.

To create a hyperlink, you should use the <a> tag and href attribute, the value of which is the URL, or location, where the link is pointing to.

```
<a href="url">your text</a>.
```

Example of the HTML <a> tag with the href attribute:

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Hyperlink</title>
```

```
  </head>
```

```
  <body>
```

```
    <h2>Hyperlink example</h2>
```

```
    <a href="https://www.amazon.com/">amazon.com</a>
```

```
  </body>
```

```
</html>
```

Output



Hyperlink example

[amazon.com](https://www.amazon.com)



In the example above, we used `<h2>` to define subheadings and the `<a>` tag to create links. Between `<a>` and `` tags we have `amazon.com`. Click on it and it will redirect you to the homepage of Amazon website.

Target Attribute

To open a link in a new page, you need to add `target="_blank"` to your code. The target attribute specifies where exactly to open the linked page. With `target="_blank"` the linked page will open in a new window or in a new tab.

```
<a href=https://www.amazon.com/ target="_blank">Amazon.com</a>
```

ID Attribute

This concept is also known as “INTERNAL LINKING” within a document.

To navigate to a specific part of the page, use the id attribute.

Here is how you should do it:

Use id attribute to give a name to the part of the page, where a user should be redirected after clicking on the link. The value of the attribute can be a word or a phrase that describe that part (if you use a phrase, there should be no spaces - use underscores instead.)

Ex. `<h2 id="jump"> Link example with id attribute. We used attribute id called "jump".`.

Create a hyperlink using the id of the link target, preceded by hash (#)

Ex. `When you click on this link, you will be redirected to the part of the page with "jump" id <a>`.

Example of the HTML <a> tag with the id attribute:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Internal linking</title>
```

```
</head>
```

```
<body>
```

```
<h2 id="jump">Link example with id attribute</h2>
```

```
<p>
```

Here can be any part of the page you want to the user to end up clicking on hyperlink. We use attribute id called “jump”.

```
</p>
```

```
<p>
```

Types of Computer

We can categorize computer in two ways: on the basis of data handling capabilities and size.

On the basis of data handling capabilities, the computer is of three types:

Analogue Computer

Digital Computer

Hybrid Computer

1) Analogue Computer

Analogue computers are designed to process analogue data. Analogue data is continuous data that changes continuously and cannot have discrete values. We can say that analogue computers are used where we don't need exact values always such as speed, temperature, pressure and current.

Analogue computers directly accept the data from the measuring device without first converting it into numbers and codes. They measure the continuous changes in physical quantity and generally render output as a reading on a dial or scale. Speedometer and mercury thermometer are examples of analogue computers.

</p>

<p>

Advantages of using analogue computers:

It allows real-time operations and computation at the same time and continuous representation of all data within the range of the analogue machine.

In some applications, it allows performing calculations without taking the help of transducers for converting the inputs or outputs to digital electronic form and vice versa.

The programmer can scale the problem for the dynamic range of the analogue computer. It provides insight into the problem and helps understand the errors and their effects.

Types of analogue computers:

Slide Rules: It is one of the simplest types of mechanical analogue computers. It was developed to perform basic mathematical calculations. It is made of two rods. To perform the calculation, the hashed rod is slid to line up with the markings on another rod.

Differential Analysers: It was developed to perform differential calculations. It performs integration using wheel-and-disc mechanisms to solve differential calculations.

Castle Clock: It was invented by Al-Jarazi. It was able to save programming instructions. Its height was around 11 feet and it was provided with the display of time, the zodiac, and the solar and lunar orbits. This device also could allow users to set the length of the day as per the current season.

Electronic Analogue Computer: In this type of analogue computer, electrical signals flow through capacitors and resistors to simulate physical phenomena. Here,

the mechanical interaction of components does not take place. The voltage of the electrical signal generates the appropriate displays.

2) Digital Computer

Digital computer is designed to perform calculations and logical operations at high speed. It accepts the raw data as input in the form of digits or binary numbers (0 and 1) and processes it with programs stored in its memory to produce the output. All modern computers like laptops, desktops including smartphones that we use at home or office are digital computers.

Advantages of digital computers:

It allows you to store a large amount of information and to retrieve it easily whenever you need it.

You can easily add new features to digital systems more easily.

Different applications can be used in digital systems just by changing the program without making any changes in hardware

The cost of hardware is less due to the advancement in the IC technology.

It offers high speed as the data is processed digitally.

It is highly reliable as it uses error correction codes.

Reproducibility of results is higher as the output is not affected by noise, temperature, humidity, and other properties of its components.

3) Hybrid Computer

Hybrid computer has features of both analogue and digital computer. It is fast like an analogue computer and has memory and accuracy like digital computers. It can process both continuous and discrete data. It accepts analogue signals and convert them into digital form before processing. So, it is widely used in specialized applications where both analogue and digital data is processed. For example, a processor is used in petrol pumps that converts the measurements of fuel flow into quantity and price. Similarly, they are used in airplanes, hospitals, and scientific applications.

Advantages of using hybrid computers:

Its computing speed is very high due to the all-parallel configuration of the analogue subsystem.

It produces precise and quick results that are more accurate and useful.

It has the ability to solve and manage big equation in real-time.

It helps in the on-line data processing.

</p>

<p>click here to go to bottom

On the basis of size, the computer can be of five types:

1) Supercomputer

Supercomputers are the biggest and fastest computers. They are designed to process huge amount of data. A supercomputer can process trillions of instructions in a second. It has thousands of interconnected processors.

Supercomputers are particularly used in scientific and engineering applications such as weather forecasting, scientific simulations and nuclear energy research. The first supercomputer was developed by Roger Cray in 1976.

Characteristics or applications of supercomputers:

It has the ability to decrypt your password to enhance protection for security reasons.

It produces excellent results in animations.

It is used for virtual testing of nuclear weapons and critical medical tests.

It can study and understand climate patterns and forecast weather conditions. It can run in NOAA's system (National Oceanic and Atmospheric Administration) that can execute any type of simple and logical data.

It helps in designing the flight simulators for pilots at the beginner level for their training.

It helps in extracting useful information from data storage centres or cloud system. For example, in insurance companies.

It has played a vital role in managing the online currency world such as stock market and bitcoin.

It helps in the diagnosis of various critical diseases and in producing accurate results in brain injuries, strokes, etc.

It helps in scientific research areas by accurately analysing data obtained from exploring the solar system, satellites, and movement of Earth.

It also used in a smog control system where it predicts the level of fog and other pollutants in the atmosphere.

2) Mainframe computer

Mainframe computers are designed to support hundreds or thousands of users simultaneously. They can support multiple programs at the same time. It means they can execute different processes simultaneously. These features of mainframe computers make them ideal for big organizations like banking and telecom sectors, which need to manage and process high volume of data.

Mainframe computers are designed to support hundreds or thousands of users simultaneously. They can support multiple programs at the same time. It means they can execute different processes simultaneously. These features of mainframe computers make them ideal for big organizations like banking and telecom sectors, which need to manage and process a high volume of data that requires integer operations such as indexing, comparisons, etc.

Characteristics of Mainframe Computers:

It can process huge amount of data, e.g. millions of transactions in a second in the banking sector.

It has a very long life. It can run smoothly for up to 50 years after proper installation.

It gives excellent performance with large scale memory management.

It has the ability to share or distribute its workload among other processors and input/output terminals.

There are fewer chances of error or bugs during processing in mainframe computers. If any error occurs it can fix it quickly without affecting the performance.

It has the ability to protect the stored data and other ongoing exchange of information and data.

Applications of mainframe computers:

In health care, it enabled hospitals to maintain a record of their millions of patients in order to contact them for treatment or related to their appointment, medicine updates or disease updates.

In the field of defence, it allows the defence departments to share a large amount of sensitive information with other branches of defence.

In the field of education, it helps big universities to store, manage and retrieve data related to their courses, admissions, students, teachers, employees and affiliated schools and colleges.

In the retail sector, the retail companies that have a huge customer base and branches use mainframe computers to handle and execute information related to their inventory management, customer management, and huge transactions in a short duration.

3) Miniframe or Minicomputer

It is a midsize multiprocessing computer. It consists of two or more processors and can support 4 to 200 users at one time. Miniframe computers are used in institutes and departments for tasks such as billing, accounting and inventory management. A minicomputer lies between the mainframe and microcomputer as it is smaller than mainframe but larger than a microcomputer.

Characteristics of miniframe or minicomputer:

- It is light weight that makes it easy to carry and fit anywhere.

- It is less expensive than mainframe computers.

- It is very fast compared to its size.

- It remains charged for a long time.

- It does not require a controlled operational environment.

Characteristics of a microcomputer:

- It is the smallest in size among all types of computers.

- A limited number of software can be used.

- It is designed for personal work and applications. Only one user can work at a time.

- It is less expensive and easy to use.

- It does not require the user to have special skills or training to use it.

Generally, comes with single semiconductor chip.

It is capable of multitasking such as printing, scanning, browsing, watching videos, etc.

</p>

<p>

Applications of minicomputers:

A minicomputer is mainly used to perform three primary functions, which are as follows:

Process control: It was used for process control in manufacturing. It mainly performs two primary functions that are collecting data and feedback. If any abnormality occurs in the process, it is detected by the minicomputer and necessary adjustments are made accordingly.

Data management: It is an excellent device for small organizations to collect, store and share data. Local hospitals and hotels can use it to maintain the records of their patients and customers respectively.

Communications Portal: It can also play the role of a communication device in larger systems by serving as a portal between a human operator and a central processor or computer.

4) Workstation

Workstation is a single user computer that is designed for technical or scientific applications. It has a faster microprocessor, a large amount of RAM and high speed graphic adapters. It generally performs a specific job with great expertise; accordingly, they are of different types such as graphics workstation, music workstation and engineering design workstation.

</p>

<p>

Characteristics of workstation computer:

It is a high-performance computer system designed for a single user for business or professional use.

It has larger storage capacity, better graphics, and more powerful CPU than a personal computer.

It can handle animation, data analysis, CAD, audio and video creation and editing.

Any computer that has the following five features, can be termed as a workstation or can be used as a workstation.

Multiple Processor Cores: It has more processor cores than simple laptops or computers.

ECC RAM: It is provided with Error-correcting code memory that can fix memory errors before they affect the system's performance.

</p>

<p>

RAID (Redundant Array of Independent Disks): It refers to multiple internal hard drives to store or process data. RAID can be of different types, for example, there can be multiple drives to process data or mirrored drives where if one drive does not work than other starts functioning.

SSD: It is better than conventional hard-disk drives. It does not have moving parts, so the chances of physical failure are very less.

</p>

<h3 id="jump1"> This is the last para</h3>

<p>

Optimized, Higher end GPU: It reduces the load on CPU. E.g., CPU has to do less work while processing the screen output.

5) Microcomputer

Microcomputer is also known as a personal computer. It is a general-purpose computer that is designed for individual use. It has a microprocessor as a central processing unit, memory, storage area, input unit and output unit. Laptops and desktop computers are examples of microcomputers. They are suitable for personal

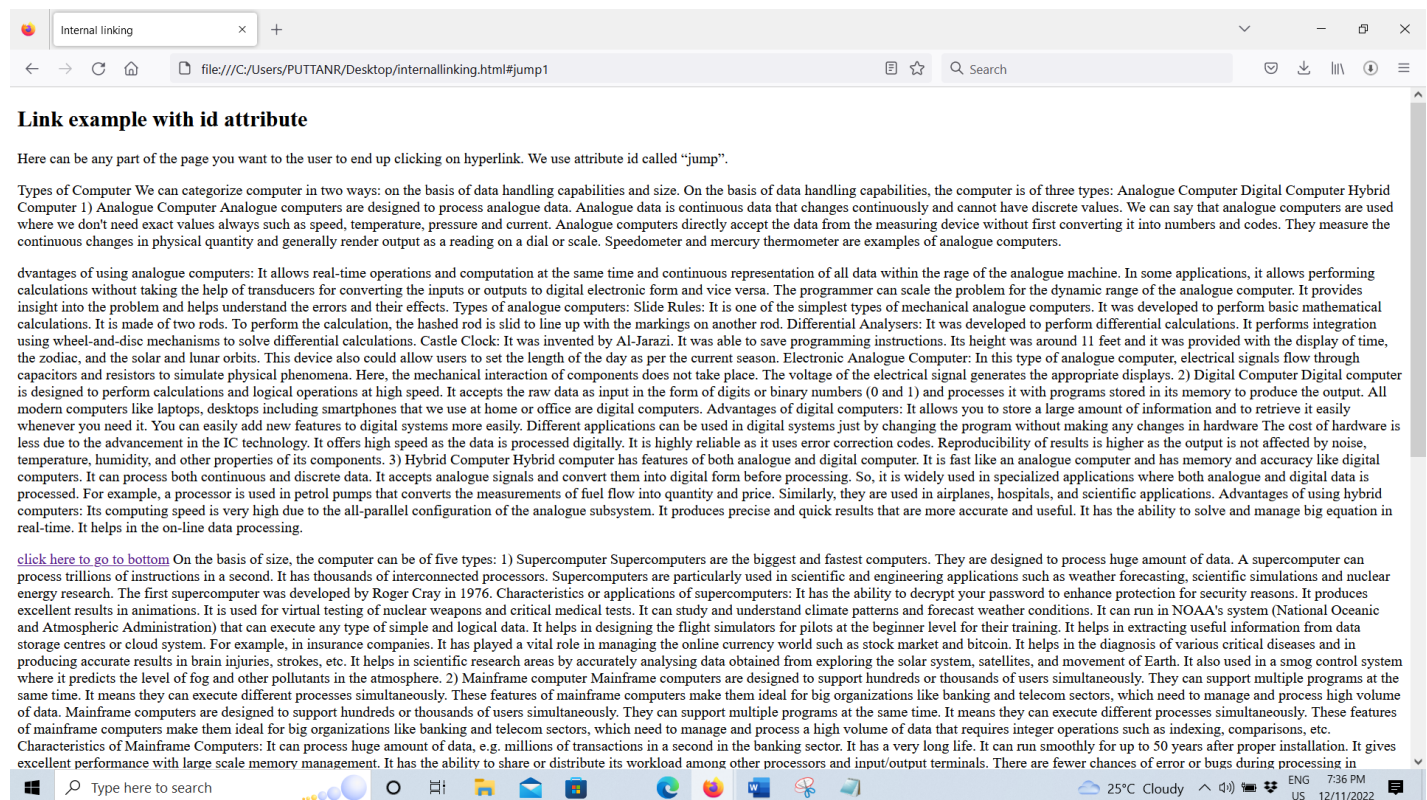
work that may be making an assignment, watching a movie, or at office for office work.

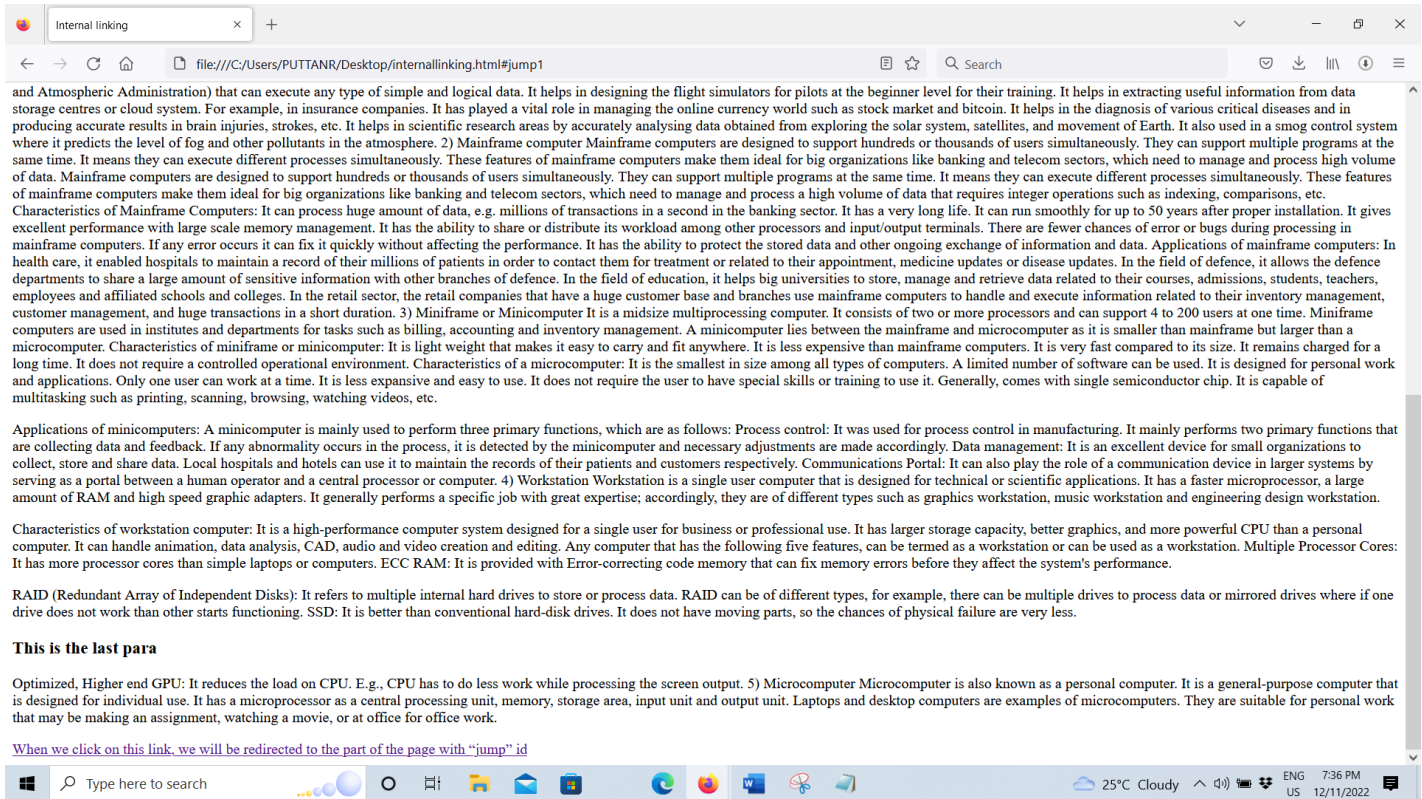
</p>

When we click on this link, we will be redirected to the part of the page with “jump” id

</body>

</html>





Apply a Hyperlink to an Image

To apply a hyperlink to an image, you just need to put the image in the <a> tag. This is done with the tag, which should have some required attributes:

1. src - the source of the image
2. alt - alternative text for the image
3. width - width of the image
4. height - height of the image

Example of the <a> and tags for applying a hyperlink to an image:

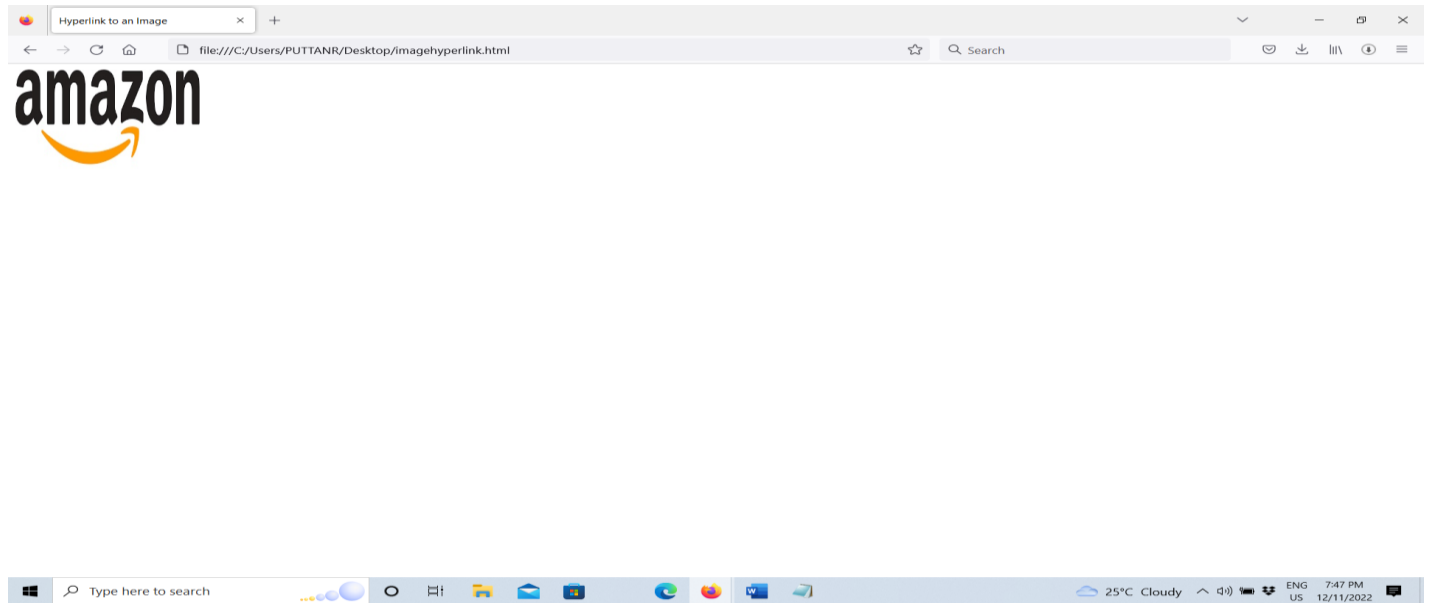
```
<!doctype html>
<html>
<head>
<title>Hyperlink to an Image</title>
</head>
<body>
```

```
<a href="https://www.amazon.in"></a>
```

```
</body>
```

```
</html>
```

Output:



HTML File Paths

A file path describes the location of a file in a web site's folder structure.

File paths are used when linking to external files, like:

- ✓ Web pages
- ✓ Images
- ✓ Style sheets
- ✓ JavaScripts

Absolute File Paths

An absolute file path is the full URL to a file:

Example

```

```

Relative File Paths

A relative file path points to a file relative to the current page.

In the following example, the file path points to a file in the images folder located at the root of the current web:

Example

```

```

In the following example, the file path points to a file in the images folder located in the folder one level up from the current folder:

Example

```

```

Lists

In HTML, there are three types of lists: unordered, ordered and description lists. Each of them is defined using different tags.

HTML Unordered Lists

We use unordered lists to group items having no numerical order. When changing the order of list items, the meaning will not change. To create an unordered list, we use the `` tag. This tag comes in pairs, the content is written between opening `` and closing `` tags.

Each element of an unordered list is declared inside the `` tag.

Example of the HTML `` tag for creating an unordered list:

```

<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h1>An unordered list:</h1>
    <ul>
      <li>This is a list item</li>
      <li>This is another list item</li>
      <li>This is one more list item</li>
    </ul>
  </body>
</html>

```

Output

- List item
- List item
- List item

- List item
- List item
- List item

The items in unordered lists are marked with bullets (small black circles) by default. However, the default bullet style for the list items can be changed using a type attribute.

The type attribute is used to change the default bullet style for the list items.

Example of the HTML tag for creating an unordered list, where the items are marked with bullets:

```

<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h2>Examples of unordered lists:</h2>

```

```
<ul style="list-style-type: square;">
  <li>Cold Drinks</li>
  <li>Hot Drinks</li>
  <li>Ice-Creams</li>
</ul>
<ul style="list-style-type: disc;">
  <li>Coca-Cola</li>
  <li>Fanta</li>
  <li>Ice Tea</li>
</ul>
<ul style="list-style-type: circle;">
  <li>Coca-Cola</li>
  <li>Fanta</li>
  <li>Ice Tea</li>
</ul>
</body>
</html>
```

Output

Examples of unordered lists:

- Cold Drinks
- Hot Drinks
- Ice-Creams

- Coca-Cola
- Fanta
- Ice Tea

- Coca-Cola
- Fanta
- Ice Tea

HTML Ordered Lists

HTML ordered list is used for listing items that are marked with numbers. It starts with the `` tag. This tag comes in pairs, the content is written between opening `` and closing `` tags.

Each item in the ordered list starts with opening `` tag and ends with `` closing tag.

Example of the HTML `` tag for creating an ordered list:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h1>An ordered list:</h1>
    <ol>
      <li>This is List item number 1</li>
      <li>This is List item number 2</li>
      <li>This is List item number 3</li>
    </ol>
  </body>
</html>
```

Output

An ordered list:

1. This is List item number 1
2. This is List item number 2
3. This is List item number 3

The items in the ordered list are marked with numbers by default. If you want to create ordered list with alphabet or Roman numbers, you just need to add type="a" or type="I" to the tag.

Example of the HTML tag for creating an ordered list with alphabet and Roman numbers:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h3>A numbered list:</h3>
    <ol>
      <li>Peach</li>
      <li>Apricot</li>
      <li>Banana</li>
      <li>Strawberry</li>
    </ol>
    <h3>An alphabetized list:</h3>
```

```

<ol type="A">
  <li>Peach</li>
  <li>Apricot</li>
  <li>Banana</li>
  <li>Strawberry</li>
</ol>
<h3>An alphabetized list (lowercase letters):</h3>
<ol type="a">
  <li>Peach</li>
  <li>Apricot</li>
  <li>Banana</li>
  <li>Strawberry</li>
</ol>
<h3>A numbered list (Roman numerals):</h3>
<ol type="I">
  <li>Peach</li>
  <li>Apricot</li>
  <li>Banana</li>
  <li>Strawberry</li>
</ol>
<h3>A numbered list (lowercase Roman numerals):</h3>
<ol type="i">
  <li>Peach</li>
  <li>Apricot</li>
  <li>Banana</li>
  <li>Strawberry</li>
</ol>
</body>
</html>

```

Output

A numbered list:

1. Peach
2. Apricot
3. Banana
4. Strawberry

An alphabetized list:

- A. Peach
- B. Apricot
- C. Banana
- D. Strawberry

An alphabetized list (lowercase letters):

- a. Peach
- b. Apricot
- c. Banana
- d. Strawberry

A numbered list (Roman numerals):

- I. Peach
- II. Apricot
- III. Banana
- IV. Strawberry

A numbered list (lowercase Roman numerals):

- i. Peach
- ii. Apricot
- iii. Banana
- iv. Strawberry

HTML Description Lists

HTML description list is used to arrange terms or names with a description the same way as they are arranged in a dictionary.

To create a description list, we use the `<dl>` tag. This tag comes in pairs.

In `<dl>`, we use `<dt>` tags for a term/name in a description list and `<dd>` for a description of a term/name in a description list.

Example of the HTML `<dl>` tag for creating a description list:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h1>Description Lists:</h1>
    <dl>
      <dt>Tea</dt>
      <dd>- hot drink</dd>
      <dt>Juice</dt>
      <dd>- cold drink</dd>
    </dl>
  </body>
</html>
```

Output

Description Lists:

Tea
- hot drink

Juice
- cold drink

HTML Nested Lists:

A nested list contains a list inside a list.

Example of an HTML nested list:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h2>A nested HTML list</h2>
    <p>A nested list contains a list inside a list.</p>
```

```

<ul>
  <li>Copybooks</li>
  <li>
    Books
    <ul>
      <li>Detective books</li>
      <li>Roman books</li>
      <li>Fairy tale books</li>
    </ul>
  </li>
</ul>
</body>
</html>

```

Output

A nested HTML list

A nested list contains a list inside a list.

- Copybooks
- Books
 - Detective books
 - Roman books
 - Fairy tale books

List Counting Control

By default, the enumeration in an ordered list starts from 1. Use the start attribute to start counting from a specified number.

Example of an HTML list for counting from a specified number:

```

<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <h2>List counting control</h2>
    <p>By default, the numeration in an ordered list starts from 1. Use the start attribute to start counting from a specified number.</p>
    <ol start="40">
      <li>Pen</li>
      <li>Pencil</li>
      <li>Copybook</li>
    </ol>
    <ol type="I" start="40">
      <li>Pen</li>
      <li>Pencil</li>
      <li>Copybook</li>
    </ol>
  </body>

```

```
</html>
```

Output:

List counting control

By default, the numeration in an ordered list starts from 1. Use the start attribute to start counting from a specified number.

40. Pen
41. Pencil
42. Copybook

- XL. Pen
- XLI. Pencil
- XLII. Copybook

Horizontal List with CSS

HTML lists can be styled in many different ways with CSS.

You can style HTML lists using different CSS properties.

For example, you can create a navigation menu styling the list horizontally.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Title of the document</title>
```

```
<style>
```

```
ul {
```

```
list-style-type: none;
```

```
margin: 0;
```

```
padding: 0;
```

```
overflow: hidden;
```

```
background-color: #F44336;
```

```
}
```

```
li {
```

```
float: left;
```

```
}
```

```
li a {
  display: block;
  color: white;
  text-align: center;
  padding: 16px;
  text-decoration: none;
}
li a:hover {
  background-color: #981816;
}
</style>
</head>
<body>
  <h2>Navigation Menu Example</h2>
  <p>
```

You can style HTML lists using different CSS properties. For example, you can create a navigation menu styling the list horizontally.

```
</p>
<ul>
  <li>
    <a href="#home">Home</a>
  </li>
  <li>
    <a href="https://www.pjs.com/tool/">Tools</a>
  </li>
  <li>
    <a href="https://www.pjs.com/snippets">Snippets</a>
  </li>
```

```
<li>
  <a href="https://www.pjs.com/quiz/">Quizzes</a>
</li>
<li>
  <a href="https://www.pjs.com/string-functions/">String Functions</a>
</li>
</ul>
</body>
</html>
```

Output

Navigation Menu Example

You can style HTML lists using different CSS properties. For example, you can create a navigation menu styling the list horizontally.



Images

The tag is used to insert an image into a document.

The attributes of this tag are:

1. the source file (src),
2. the alternative text (alt),
3. width,
4. height.

The tag is used for inserting images into HTML document

All tags must have a defined src attribute. This defines the image to be displayed.

Example:

```
<!DOCTYPE html>
<html>
  <head>
```

```
<title>Title of the document</title>
</head>
<body>
  <h1>This is an image example</h1>
  
</body>
</html>
```

Output



iframe (embed YouTube video)

Define an inline frame with HTML tag `<iframe>`. The `<iframe>` tag is not somehow related to `<frameset>` tag, instead, it can appear anywhere in your document. The `<iframe>` tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders. An inline frame is used to embed another document within the current HTML document.

The `src` attribute is used to specify the URL of the document that occupies the inline frame.

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>HTML Iframes</title>
```

```
  </head>
```

```
    <body>
```

```
      <p>Document content goes here...</p>
```

```
        <iframe src = "https://www.youtube.com/embed/watch?v= 33mOZlS10XU"
width = "555" height = "200">
```

```
          Sorry your browser does not support inline frames.
```

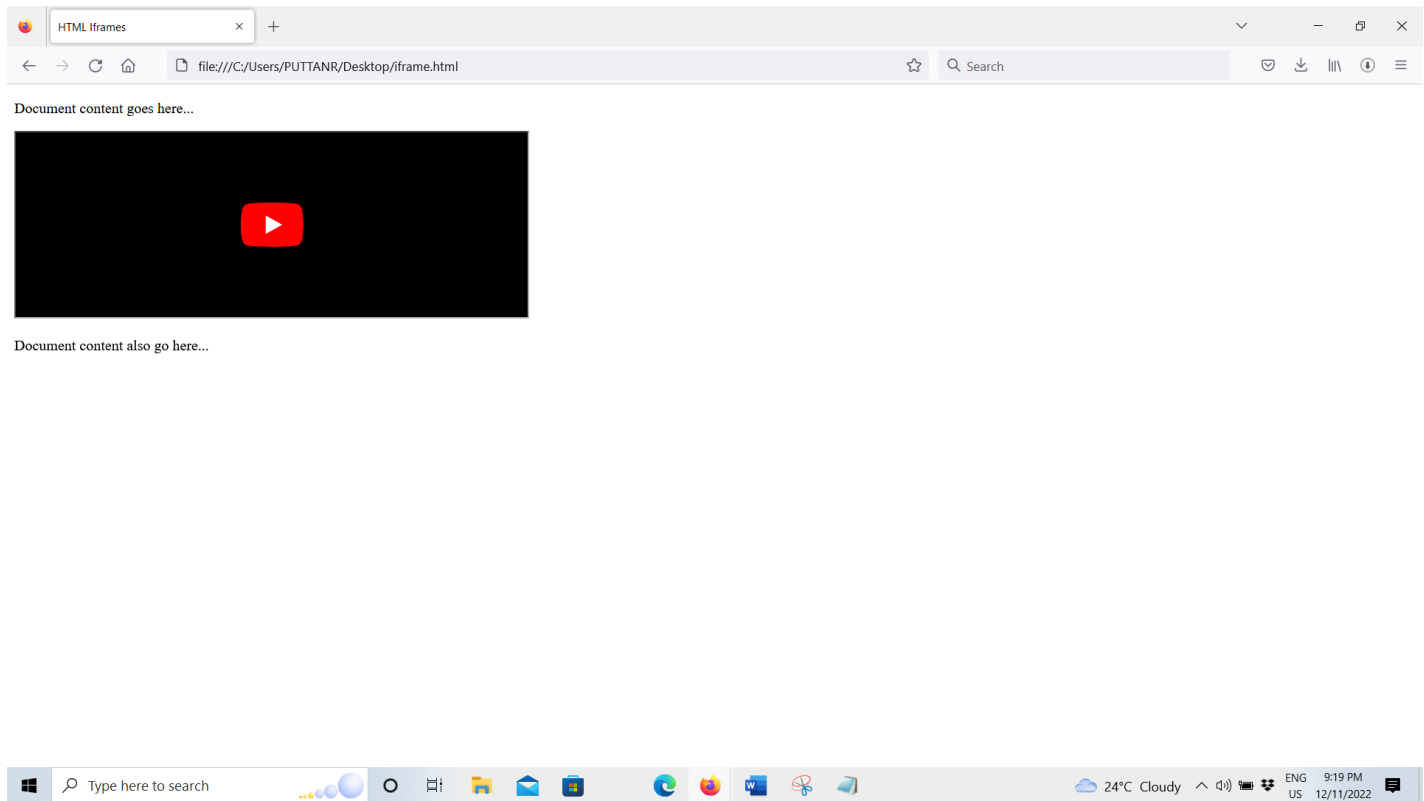
```
        </iframe>
```

```
      <p>Document content also go here...</p>
```

```
    </body>
```

```
</html>
```

Output



HTML Frames

HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset. The window is divided into frames in a similar way the tables are organized: into rows and columns.

Disadvantages of Frames

There are few drawbacks with using frames, so it's never recommended to use frames in your webpages –

- Some smaller devices cannot cope with frames often because their screen is not big enough to be divided up.
- Sometimes your page will be displayed differently on different computers due to different screen resolution.
- The browser's *back* button might not work as the user hopes.
- There are still few browsers that do not support frame technology.

Creating Frames

To use frames on a page we use `<frameset>` tag instead of `<body>` tag. The `<frameset>` tag defines, how to divide the window into frames. The **rows** attribute of

<frameset> tag defines horizontal frames and **cols** attribute defines vertical frames. Each frame is indicated by <frame> tag and it defines which HTML document shall open into the frame.

Example1

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Frames</title>
  </head>

  <frameset rows = "10%,80%,10%">
    <frame name = "top" src = "/html/top_frame.htm" />
    <frame name = "main" src = "/html/main_frame.htm" />
    <frame name = "bottom" src = "/html/bottom_frame.htm" />

    <noframes>
      <body>Your browser does not support frames.</body>
    </noframes>

  </frameset>

</html>
```

Example2

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Frames</title>
  </head>

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

    <noframes>
      <body>Your browser does not support frames.</body>
    </noframes>

  </frameset>

</html>
```