Hypertext Mark-Up Language (HTML)

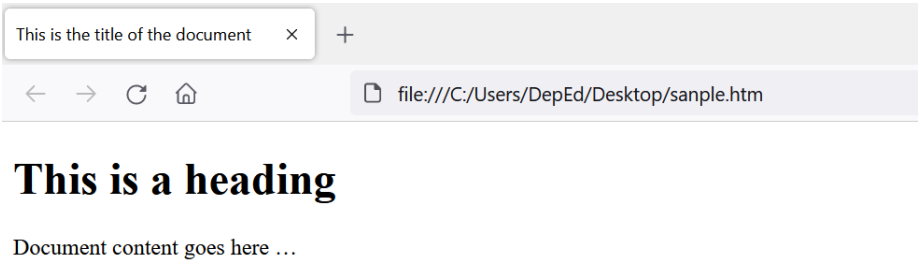
- the most widely used language to write Web Pages
- **Hypertext** refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called **Hypertext**
- HTML is a **Markup Language** which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display
- originally, it was developed with the intent of defining the structure of documents like headings, paragraphs, lists, ad so forth to facilitate the sharing of scientific information between researchers
- created by Berner's-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995
- the fifth and final major HTML version is html 5.0

Sample HTML Document

```
<!DOCTYPE HTML>

<html>
<head>
<title>This is the title of the document </title>
</head>

<body>
<h1>This is a heading </h1>

<p>Document content goes here … </p>
</body>

</html>
```

Type the code above using any text editor and save it as sample.htm. Open the file using any web browser.

Sample Output:



HTML Tags

- The tags are enclosed within angle braces <Tag Name>. Except few tags, most of the tags have their corresponding closing tags. For example, <html> has its closing tag with </html> and <body> tag has its closing tag </body> tag, etc.

The previous HTML document uses the following tags:

Tag	Description
<!DOCTYPE…>	This tag defines the document type and HTML version
<html>	This tag signifies the beginning and end of HTML code.
<head>	This tag contains the page title, which is the text that will appear in the title bar of the web browser
<title>	The <title> tag is used inside the <head> section to mention the document title.
<body>	This tag signifies the beginning and end of the visible portion of the web page when viewed in a browser. It keeps other HTML tags like <h1>, , <p>, etc.
<h1>	This tag represents the heading.
<p>	This tag represents a paragraph.

To learn HTML, you will need to study various tags and understand how they behave, while formatting a textual document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage.

World Wide Web consortium (W3C) recommends using lowercase tags starting from HTML 4.

The HTML Document Structure

A typical HTML document will have the following structure:

```
Document declaration tag

<html>
    <head>
        Document header related tags
    </head>

    <body>
        Document body related tags
    </body>
</html>
```

<!DOCTYPE HTML**>**

- the <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document
- DOCTYPE tag always begins with an exclamation point and is always placed at the beginning of the document before any other tag.
- DOCTYPE should always be uppercase
- optional tag but coder promises to conform to certain standards.
- if there is no DOCTYPE tag the web browser will assume something quirky and will process the page in quirks mode
- Web browser processes the page in standards mode
- current version of HTML is 5 and it makes use of the following declaration:

```
<!DOCTYPE html>
```

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used.

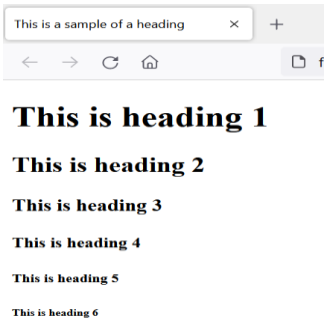
Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has 6 levels of headings, which use the elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6> While displaying any heading, browser **adds** one line before and one line after that heading.

Example:

```
<!DOCTYPE html>

<html>
    <head>
            <title>This is a sample of a heading </title>
     </head>

    <body>
            <h1>This is heading 1 </h1>
            <h2>This is heading 2 </h2>
            <h3>This is heading 3 </h3>
            <h4>This is heading 4 </h4>
            <h5>This is heading 5 </h5>
            <h6>This is heading 6 </h6>
   </body>

</html>
```

The output will be:



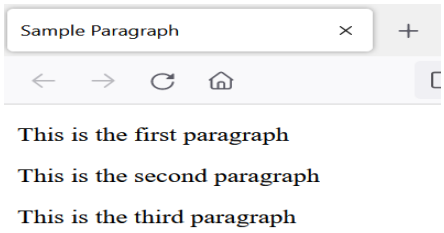
Paragraph Tag

The tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening and a closing tag.

Example:

```
<!DOCTYPE html>

<html>
        <head>
                <title>Sample Paragraph </title>
        </head>

        <body>
                <p>This is the first paragraph </p>
                <p>This is the second paragraph </p>
                <p>This is the third paragraph </p>
       </body>
</html>
```

Output:



Line Break Tag

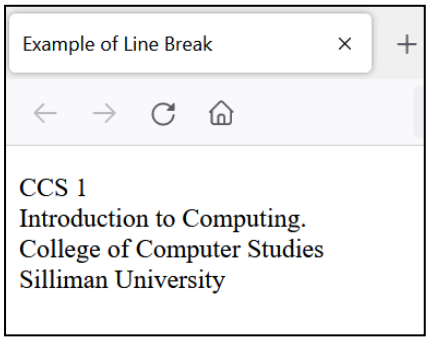
Whenever you use a **
** element, anything following it starts from the next line. This tag is an example of an **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The **
** tag **has a space** between the characters **br** and the **forward slash**. If you omit this space, older browser will have trouble rendering the line break, while if you miss the forward slash character and just use <br>, it is not valid in XHTML (Extensible HyperText Mark-up Language).

Example:

```
<!DOCTYPE html>
<html>
   <head>
          <title>Example of Line Break </title>
   </head>

   <body>
          <p>CCS 1 <br />
              Introduction to Computing. <br />
              College of Computer Studies <br />
              Silliman University
           </p>
  </body>
</html>
```

Output:



Centering Content

- you can use tag to put any content in the center of the page or any table cell

Example:

```
<center>
        <p>This text is center-aligned. </p>
```

Horizontal Lines

- used to visually break-up sections of a document
- the **<hr />** tag creates a like from the current position in the document to the right margin and breaks the line accordingly

Example:

```
<hr />
        <p>College of Computer Studies <p>
<hr />
```

Output:

College of Computer Studies

<hr> Style Attributes

- color, background-color, width, height

Example:

```
<hr color = “red” width = “100%” size = “10” />
```

16 Basic Color Names

- aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow

Preserve Formatting

Sometimes, you want your text to follow the exact format of how it is written in the HTML document. It these cases, you can use the preformatted tag **<pre>.**
Any text between the opening **<pre>** tag and the closing **</pre>** tag will preserve the formatting of the source document.

Example:

```
<!DOCTYPE html>
<html>
        <head>
                <title>Preformatting Example </title>
        </head>

        <body>
                <pre>    Hello everyone!        How are you?  </pre>
        </body>
</html>
```

Output:

```
    Hello everyone!        How are you?
```

Nonbreaking Spaces

A commonly used entity in HTML is the non-breaking space: ** **
A non-breaking space is a space that will not break into a new line.
Two words separated by a non-breaking space will stick together (not break into a new line). This is handy when breaking the words might be disruptive.

Examples:

- § 10
- 10 km/h
- 10 PM

Another common use of the non-breaking space is to prevent browsers from truncating spaces in HTML pages.
If you write 10 spaces in your text, the browser will remove 9 of them. To add real spaces to your text, you can use the ** ** character entity.

Useful HTML Character Entities

- codes beginning with an ampersand (&), followed by an entity name or entity number, and ending with a semicolon
- names and numbers represent the same thing

Symbol	Entity Name	Entity Number
& (ampersand)	&amp;	&#38;
< (less than)	&lt;	&#60;
> (greater than)	&gt;	&#62;
(non breaking space)		
(cent)	&cent;	&#162;
(pound)	&pound;	&#163;
(yen)	&yen;	&#165;
(copyright)	&copy;	&#169;
(registered trademark)	&reg;	&#174;
(degree)	&deg;	&#176;
(plus or minus)	&plusmn;	&#177;
(trademark)	&trade;	&#8482;

HTML Elements

An HTML element is defined by a starting tag. If the element contains other content, it ends with a closing tag, where the element is preceded by a forward slash as shown below with few tags:

Start Tag	Content	End Tag
	This is a paragraph content.	
<h1>	This is a heading content.	</h1>
	This is a division content.	

So here, … is an element, <h1>…</h1> is another HTML element. There are some HTML elements which do not need to be closed like <img…/>, <hr /> and <br /> elements. These are known as **void elements**.

HTML documents consists of a tree of these elements and they specify how HTML documents should be built, and what kind of content should be placed in what part of an HTML document.

HTML Tag vs. Element

An HTML element is defined by a starting tag. If the element contains other content, it ends with a closing tag.

For example, ** ** is starting tag of paragraph and ** ** is closing tag of the same paragraph but ** This is a paragraph ** is a paragraph element.

Nested HTML Elements

Keeping one HTML element inside another HTML element is allowed.

```
<!DOCTYPE html>
<html>
    <head>
                    <title>Example of Nested Elements </title>
        </head>

        <body>
                    <h1>This is <i>italic </i>heading </ih1>
                    <p>This is <u>underlined </u>paragraph</p>
        </body>

</html>
```

The output will be:

This is *italic* heading

This is underlined paragraph

HTML Comment Tag

- **<!--…-->**
- used to insert comments in the source code
- comments are not displayed in the browsers.
- you can use comments to explain your code, which can help you when you edit the source code at a later date

Example:

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

<p>This is a paragraph.</p>
```

Activity:

Create a web page containing your biodata (see sample output to follow). Use appropriate tags discussed in this chapter. Do not include other tags, properties, etc. which are not yet discussed. This is to validate the learnings you have acquired from the discussions.

The following must be present:

- Title of page
- Horizontal Lines
- Paragraph
- Line break
- Heading
- DOCTYPE
- HTML character entities
- Lists
- Background and font color
- Image

File Name: index.html

SAMPLE OUTPUT:

BIODATA

PERSONAL DETAILS

Name	: (type your complete name)	Father's Name: (type complete name)	
Mobile No.	:	Email Address:	
Gender	:	Date of Birth:	
Marital Status	:	Religion:	
Nationality	:	Language:	
Address	:		

EDUCATION

Elementary	: (provide data)	Year Graduated: (provide the data)
Secondary	:	Year Graduated:
College	:	Year Graduated: (leave blank if still studying)

Fill the entire page with other relevant data like data of your parents, hobbies, talents, skills, etc. At the bottom part of the page, display your name like the one shown below.

© Your complete name here

HTML Formatting

If you use a word processor, you must be familiar with the ability to make text bold, italicized, or underlined; these are just three of the ten options available to indicate how text can appear in HTML and XHTML.

Italic Text
- **** and ****
- anything that appears within the… element is displayed in italicized
- as substitute

Bold Text
- **** and ****
- <**strong**> as substitute

Superscript Text
- **^{** and **}**
- the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters

Subscript Text
- **_{** and **}**
- the font size used is the same as the characters surrounding it, but is displayed half character's height beneath the other characters

Underlined Text
- anything that appears within **<u>** and **</u>** element is displayed with underline

Strike Text
- **<strike>** and **</strike>**
- anything that appears within the <strike> and </strike> element is displayed with strikethrough, which is a thin line through the text

Monospaced Font
- the content of the **<tt>…</tt>** is written in *monospaced* font
- most of the fonts are known as variable-width fonts because different letters are of different widths (for example, the letter 'm' is wider than the letter "l")

Larger Text
- the content of the **<big>…</big>** element is displayed one font size larger than the rest of the text surrounding it

Smaller Text
- the content of the **<small>…</small>** element is displayed one font size smaller than the rest of the text surrounding it

HTML Phrase Tags

These tags are displayed in a similar with as other basic tags like **<b>, <i>,** and **<tt>.**

Emphasized Text
- anything that appears within **…** element is displayed as emphasized text

Marked Text
- anything that appears within **<mark>…</mark>** element is displayed as marked with yellow ink

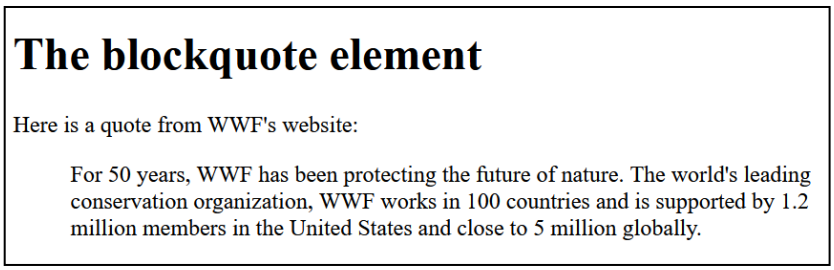
Strong Text
- anything that appears within **…** element is displayed as important text

Quoting Text
- tag for quoting blocks from other sources
- text inside the <blockquote> element is usually indented from the left and right edges of the surrounding text, and sometimes uses an italicized font
- **<blockquote>** and **</blockquote>**

Example:

```
…other codes

<blockquote cite=" http://www.worldwildlife.org/who/index.html">
For 50 years, WWF has been protecting the future of nature. The world's leading conservation organization, WWF works in 100 countries and is supported by 1.2 million members in the United States and close to 5 million globally.
</blockquote>
```

Output:



Creating a List

An ordered list starts with the **<ol>** tag. Each list item starts with the **<li>** tag

Example:

```
<ol>
        <li>BSIT</li>
        <li>BSCS</li>
        <li>BSIS</li>
</ol>
```

Ordered HTML List – The Type Attribute

The **type** attribute of the **<ol>** tag defines the type of the list item marker:

Type	Description
type = "1"	The list items will be numbered with numbers (default)
type = "A"	The list items will be numbered with uppercase letters
type = "a"	The list items will be numbered with lowercase letters
type = "I"	The list items will be numbered with uppercase roman numbers
type = "i"	The list items will be numbered with lowercase roman numbers

Example:

```
<ol type ="1">
        <li>BSIT</li>
        <li>BSCS</li>
        <li>BSIS</li>
</ol>
```

Output:

1. BSIT
2. BSCS
3. BSIS

Unordered HTML List

An unordered list starts with the **<ul>** tag. Each list item starts with the **<li>** tag.
The list items will be marked with bullets (small black circles) by default:

Example of bulleted list:

```
<ul>
      <li>Banana </li>
      <li>Grapes </li>
      <li>Apple </li>
</ul>
```

Unordered HTML List – Choose List Item Market

Value	Description
disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked

Example

```
<ul style = “list-style-type: disc”>
      <li>Banana </li>
      <li>Grapes </li>
      </liApple </li>
</ul>
```

Output:

- Banana
- Grapes
- Apple

HTML <dl>, <dt> and <dd> Tags

- **<dl>…</dl>** defines a *definition list*
  - **<dt>…</dt>** defines definition term
  - **<dd>…</dd>** defines definition description

Example:

```
<!DOCTYPE html>
<html>
   <body>
        <h1>The dl, dd, and dt elements</h1>
         <p>These three elements are used to create a description list:</p>

        <dl>
             <dt>Fruits</dt>
                          <dd>Orange, Apple, Mango</dd>
              <dt>Vegetables</dt>
                          <dd>Carrots, Pechay, Squash</dd>
        </dl>
</body>
</html>
```

Output:

The dl, dd, and dt elements

These three elements are used to create a description list:

Fruits
    Orange, Apple, Mango
Vegetables
    Carrots, Pechay, Squash

HTML Attributes

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

The **name** is the property you want to set. For example, the paragraph **<p>** element in the example carries an attribute whose **name** is **align**, which you can use to indicate the alignment of paragraph on the page.

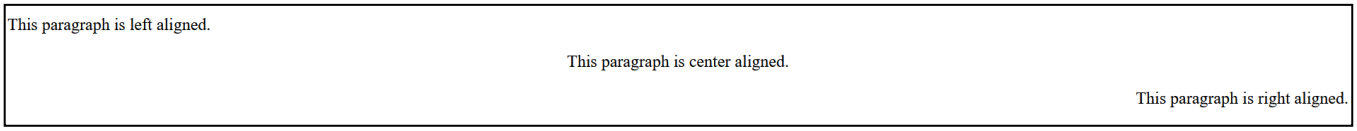
The **value** is what you want the value of the property to be set and always put within quotations. The example below shows three possible values of align attribute: **left**, **center** and **right**.

Attribute names and attribute values are case-sensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

Example:

```
<!DOCTYPE html>
<html>
        <head>
                <title>Example of Align Attribute </title>
        </head>

        <body>
                <p align="left">This paragraph is left aligned. </p>
                <p align="center">This paragraph is center aligned. </p>
                <p align="right">This paragraph is right aligned. </p>
        </body>

</html>
```

Output:



Background color

- style = "background-color: color" attribute to the opening **<body>** tag

Example:

```
<body style = "background-color: yellow">
```

or

```
<body style = "background-color: #FF0000">
```

or

```
<body style = "background-color: rgb(255,255,0)">
```

Foreground color

- style = “color: color” attribute to the opening **<body>** tag

Example:

```
<body style = “color: navy”>
```

- when combining two attributes in a single style = statement, separate them with a semicolon.

Example:

```
<body style = “background-color: black; color: yellow”>
```

Adding background image

- by default, image is tiled
- use **style = “background-image: url(‘image’)”** attribute in the opening body tag

Example:

```
<body style = “background-image: url(‘sample.jpg’)”>
```

No repeat

```
style = “background-repeat: no-repeat”
```

Fixed image

```
style = “background-attachment: fixed”
```

Inserting an Image

You can insert any image in your web page by using **<img>** tag

```
<img src = “Image URL”… attributes-list/>
```

The **<img>** tag is an **empty** tag, which means that, it can contain only list of attributes and it has no closing tag.

Example 1: Image file is placed together with other files in one folder

```
<!DOCTYPE html>
<html>

    <head>
        <title>Inserting an Image</title>
    </head>

    <body>
        <p>Please see picture below:</p>
        <img src = “car.jpg” alt = "Car picture"/>
    </body>

</html>
```

You can use PNG, JPEG or GIF image file based on your comfort but make sure you specify correct image file name in **src** attribute. **Image name is always case sensitive.**

The **alt** attribute is a *mandatory* attribute which specifies an alternate text for an image, if the image cannot be displayed.

Example 2: Image file is placed in a specific folder named folder1

```
<!DOCTYPE html>
<html>

    <head>
          <title>Inserting an Image</title>
    </head>

    <body>
           <p>Please see picture below:</p>
           <img src= folder1/car.jpg" alt = "Car picture" />
    </body>

</html>
```

Set Image Width/Height

You can set image width and height based on your requirement using **width** and **height** attributes. You can specify **width** and **height** of the image in terms of either *pixels* or *percentage* of its actual size.

```
<!DOCTYPE html>
<html>

  <head>
          <title>Set Image Width and Height</title>
  </head>

  <body>
          <p>Setting image width and height</p>
          <img src = “pic.png” alt = "Test Image" width = "150" height = "100"/>
  </body>

</html>
```

Set Image Border

By default, image will have a border around it, you can specify border thickness in terms of pixels using border attribute. A thickness of 0 means, no border around the picture.

Example:

```
<!DOCTYPE html>
<html>

  <head>
          <title>Set Image Border</title>
  </head>

  <body>
          <p>Setting image Border</p>
           <img src = pic.png" alt = "Test Image" border = "3"/>
  </body>

</html>
```

Set Image Alignment

By default, image will align at the left side of the page, but you can use **align** attribute to set it in the center or right.

Example:

```
<!DOCTYPE html>
<html>

  <head>
          <title>Set Image Alignment</title>
  </head>

  <body>
          <p>Setting image Alignment</p>
          <img src = "pic.png" alt = "Test Image" border = "3" align = "right"/>
  </body>

</html>
```

Hyperlink

- activated by clicking a text or graphic
- depending on the link, opens a different location on the page, opens a different Web page, starts and email message, downloads a file, opens video or audio player, starts a Web-based program, etc.

<a> tag

- href= attribute for a path to a different destination
- If link is not for a specific page, default page for the site is returned (pages named either index or default)

Example:

```
Visit <a href= "https://su.edu.ph"> Silliman University</a> for more details.
```

Absolute path

- paths that contain a complete address

Relative path

- shortened name/address for path
- link to files in the same Web site as the link itself
- destination file is relative to current file's location

Example:

```
<a href="newpage.htm">New Page</a>
```

Link to a file in a subfolder of current one (child folder)

Example:

```
<a href="articles/artpage.htm">Article page</a>
```

Link to a file up one level of current folder (parent folder)

Example:

```
<a href="../da_home.htm">Home</a>
```

Opening in link in a New Browser or Tab

```
<a href="https://www.su.edu.ph" target="_blank">Silliman University </a>
```

Link to send a message to an email address

Example:

```
<a href="mailto: kimlfaburada@su.edu.ph">Email</a>
```

Adding a subject to the message

Example:

```
<a href="mailto: kimlfaburada@su.edu.ph? subject=Comments">Contact us</a>
```

Adding title to hyperlink

Example:

```
<a href=”mailto: kimlfaburada@su.edu.ph” title=”Email us”>Send us a message</a>
```

Anchor

- marker within an HTML document
- mark the anchor location then create a hyperlink to it

Example (for anchor point):

```
<a name=”anchor1”>Sample anchor</a>
```

Example (to refer to anchor point):

```
<a href=”#anchor1”>Go to sample anchor</a>
```

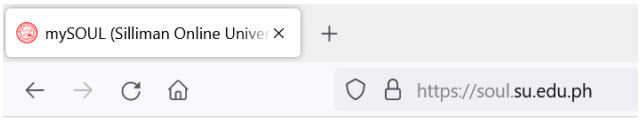
Example (if in a different file):

```
<a href=”differentfile.htm#anchor1”>Different file anchor</a>
```

How to Add Favicon in HTML

Favicon

- a small image displayed next to the page title in the browser tab
- t should be a simple image with high contrast
- you can use any image you like as your favicon. You can also create your own favicon on sites like https://iconarchive.com/tag/picture
-



To add a favicon to your website, either save your favicon image to the root directory of your webserver, or create a folder in the root directory called images, and save your favicon image in this folder. A common name for a favicon image is "**favicon.ico**".

Next, add a <link> element to your "index.html" file, after the <title> element, like this:

Use the HTML **<link>** element to insert a favicon

```
<head>
     <title>My Page Title</title>
     <link rel="icon" type="image/x-icon" href="/images/favicon.ico">
</head>
```

DIVs and ID's

Div Tag

The **div** tag is a block level HTML element. It is used to *divide* or *section* other HTML tags into meaningful groups.

Id Attribute

The **id** attribute is used to *label* sections or parts of your HTML document. You may only use the same id once per page, so save it for important major sections of your page. Additionally, the id selector in CSS has a high specificity level and therefore overrules other things (like the class selector).

A perfect example of the use of a **div** tag and **id** attribute is to designate a navigation list:

```
<div id="navigation">
  <ul>
      <li><Facebook </li>
      <li>Instagram</li>
      <li>Youtube</li>
  </ul>
</div>
```

References:

www.w3schools.com
https://www.tutorialspoint.com/
www.tutorialspoint.com
http://web.simmons.edu/~grabiner/comm244/weekthree/html-hooks.html

HTML Tables

- HTML tables allow web developers to arrange data into rows and columns.

HTML Table Tags

<table>	Defines a table
<th>	Defines a header cell in a table
<tr>	Defines a row in a table
<td>	Defines a cell in a table
<caption>	Defines a table caption

Table Cells

- each table is defined by a **<td>** and a **</td>** tag
- **td** stands for table data
- everything between <td> and </td> are the content of the table

Example:

```
<table>
    <tr>
        <td>Colors</td>
        <td>Fruits</td>
        <td>Numbers</td>
    </tr>
</table>
```

Table Rows

- each table row starts with a <tr> and ends with a </tr> tag
- **tr** stands for table row

Example:

```
<table>
    <tr>
        <td>Colors</td>
        <td>Fruits</td>
        <td>Numbers</td>
    </tr>
    <tr>
        <td>Blue</td>
        <td>Mango</td>
        <td>1</td>
    </tr>
</table>
```

Table Headers

- if you want to use a cell as a header, use the **<th>** tag instead of the <td> tag

Example:

```
<table>
    <tr>
        <th>Colors</th>
        <th>Fruits</th>
        <th>Numbers</th>
    </tr>
    <tr>
        <th>Blue</th>
        <th>Mango</th>
        <th>1</th>
    </tr>
</table>
```

Spanning Multiple Rows and Columns

Spanning allows you to extend table rows and columns across multiple other rows and columns. Normally, a table cell cannot pass over into the space below or above another table cell. But you can use the **rowspan** or **colspan** attributes to span multiple rows or columns in a table.

Before Applying Colspan:

```
<table>
    <tr>
        <th>Grade</th>
        <th>Teacher</th>
    </tr>
    <tr>
        <td>1</td>
        <td>2</td>
        <td>Mrs. Tan</td>
    </tr>
</table>
```

After Applying Colspan:

```
<table>
    <tr>
        <th>Grade</th>
        <th colspan="2">Teacher</th>
    </tr>
    <tr>
        <td>1</td>
        <td>2</td>
        <td>Mrs. Tan</td>
    </tr>
</table>
```

Before Applying Rowspan

You can use the **rowspan** attribute to create a cell that spans more than one row.

Example:

```
<table>
    <tr>
        <th>Name:</th>
        <td>Mrs. Tan</td>
    </tr>
    <tr>
        <th rowspan="2">Phone</th>
        <td>0915-123-4567</td>
    </tr>
    <tr>
        <td>09121-123-4567</td>
    </tr>
</table>
```

Adding Captions to Tables

You can specify a caption (or title) for your tables using the <**caption**> element.
The <caption> element must be *placed directly after the opening <table>* tag. By default, caption appears at the top of the table.

Example:

```
<table>
    <caption>Masterlist </caption>
    <tr>
        <td>Name </td>
        <td>Address </td>
    </tr>
</table>
```

How to Add Border to HTML Table

Example:

```
<table border="1">
    <tr>
        <td>Name </td>
        <td>Address </td>
    </tr>
</table>
```

This creates a table with double borders

How to Add Border to HTML Table (Collapsed Border)

Example:

```
<table border=”1” cellpadding = “0” cellspacing=”0”>
    <tr>
        <td>Name </td>
        <td>Address </td>
    </tr>
</table>
```

- **Cell padding** is the space between the cell edges and the cell content.
- **Cell spacing** is the space between each cell.
- By default the space is set to 2 pixels.

Adding Background Color and Setting a Size to the EntireTable

Example:

```
<table style="width:100%; text-align:left;background-color:yellow">
    <tr>
        <td>Name </td>
        <td>Address </td>
    </tr>
</table>
```

Adding Background Color to a Specific Row in a Table

Example:

```
<table style="width:100%; text-align:left;background-color:yellow">
    <tr>
        <td>Name </td>
        <td>Address </td>
    </tr>
    <tr style="background-color:green; color:white">
        <td>Juan</td>
        <td>Dumaguete City</td>
    <tr>
</table>
```

References:

https://www.w3schools.com/html/html_tables.asp
https://www.tutorialrepublic.com/html-tutorial/html-tables.php

Multimedia

Multimedia comes in many different formats. It can be almost anything you can hear or see, like images, music, sound, videos, records, films, animations, and more.

Web pages often contain multimedia elements of different types and formats.

Browser Support

The first web browsers had support for text only, limited to a single font in a single color.
Later came browsers with support for colors, fonts, images, and multimedia!

Types of multimedia: *animation, sound and video*

Animation

- refers to the ability to see continuous movement of something that otherwise would be stationary.
- 2 types of animation: animated text and animated pictures

Animated text

- <marquee>…</marquee> allows you to introduce streaming text
- the text can "stream" in various ways: it can scroll slide, or alternate
- many other attributes of this element allow you to modify characteristics and achieve the desired effect

Syntax:

<marquee
behavior=alternate || **scroll** || slide
direction = down | up | **left** | right
loop = blank or integer number
scrollamount = number of pixels
scrolldelay = number of milliseconds
hspace = *integer pixels*
vspace = *integer pixels*
height = *integer pixels*
width = *integer pixels*

…streaming text…
</marquee>

Example: <marquee bgcolor="red" behavior="scroll" direction="left" scrollamount="5"
scrolldelay="1" title="Marquee Example" width="80%">

Animated .GIF's

- the simplest form of animation
- first supported by Netscape 2 but are now supported by most current browsers
- consist of several files packaged into one

Example:

<img src="mario.gif" width="50" height="50" border="0" alt="Smiley Face">

Multimedia Formats

Multimedia elements (like audio or video) are stored in media files. The most common way to discover the type of a file, is to look at the file extension. Multimedia files have formats and different extensions like: .wav, .mp3, .mp4, .mpg, .wmv, and .avi.

Common Video Formats

There are many video formats out there.

The MP4, WebM, and Ogg formats are supported by HTML.

The MP4 format is recommended by YouTube.

The HTML Video

The HTML `<video>` element is used to show a video on a web page.

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

   <video width="320" height="240" controls>
        <source src="movie.mp4" type="video/mp4">
        <source src="movie.ogg" type="video/ogg">
         Your browser does not support the video tag.
   </video>
</html>
```

HTML Audio

Common Audio Formats

MP3 is the best format for compressed recorded music. The term MP3 has become synonymous with digital music. If your website is about recorded music, MP3 is the choice.

Note: Only MP3, WAV, and Ogg audio are supported by the HTML standard.

Embedding a sound file within the page

Example:

```
<audio>
    <source src="music1.wav" type="audio/wav”>
    <source src="music1.mp3" type="audio/mp3">

    Your browser does not support the audio element.
</audio>
```

The controls attribute adds audio controls, like play, pause, and volume.
The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.

The text between the <audio> and </audio> tags will only be displayed in browsers that do not support the <audio> element.

HTML Audio Autoplay

Example:

```
<audio controls autoplay>
    <source src="music1.wav" type="audio/wav”>
    <source src="music1.mp3" type="audio/mp3">

    Your browser does not support the audio element.
</audio>
```

Note: Chromium browsers do not allow autoplay in most cases. However, muted autoplay is always allowed.

Add muted after autoplay to let your audio file start playing automatically (but muted):

Browser Support

The numbers in the table specify the first browser version that fully supports the <audio> element.

Element					
<audio>	4.0	9.0	3.5	4.0	10.5

HTML Audio Formats

There are three supported audio formats: MP3, WAV, and OGG. The browser support for the different formats is:

Browser	MP3	WAV	OGG
Edge/IE	YES	YES*	YES*
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	YES	NO
Opera	YES	YES	YES

HTML Audio - Media Types

File Format	Media Type
MP3	audio/mpeg
OGG	audio/ogg
WAV	audio/wav

HTML Audio - Methods, Properties, and Events

The HTML Data Object Model (DOM) defines methods, properties, and events for the <audio> element. This allows you to load, play, and pause audios, as well as set duration and volume. There are also DOM events that can notify you when an audio begins to play, is paused, etc.

References:

https://www.w3schools.com/html/html_media.asp

The HTML Style Attribute

Setting the style of an HTML element, can be done with the **style** attribute.

The HTML style attribute has the following syntax:

```
<tagname style="property:value;">
```

The **property** is a CSS property. The **value** is a CSS value.

Background Color:

```
<body style="background-color:powderblue;">
<h1 style="background-color:powderblue;">This is a heading</h1>
<p style="background-color:tomato;">This is a paragraph.</p>
```

Text Color

```
<h1 style="color:blue;">This is a heading</h1>
<p style="color:red;">This is a paragraph.</p>
```

Fonts

```
<h1 style="font-family:verdana;">This is a heading</h1>
<p style="font-family:courier;">This is a paragraph.</p>
```

Text Size

```
<h1 style="font-size:300%;">This is a heading</h1>
<p style="font-size:160%;">This is a paragraph.</p>
<p style="font-size:30px">This is a paragraph.</p>
<p style="font-size:11px">This is another paragraph.</p>
```

Text Alignment

```
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>
```

Reference:

https://www.w3schools.com