What is HTML?

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

-
  ```
  <!DOCTYPE html>
  <html>
  <head>
  <title>Page Title</title>
  </head>
  <body>

  <h1>My First Heading</h1>
  <p>My first paragraph.</p>

  </body>
  </html>
  ```

What is an HTML Element?

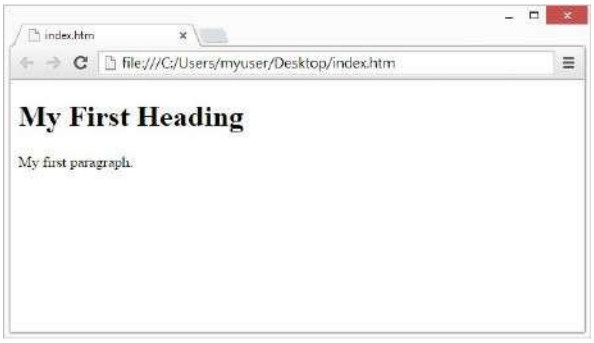
- An HTML element is defined by a start tag, some content, and an end tag:
- <tagname> Content goes here... </tagname>
- The HTML element is everything from the start tag to the end tag:
- <h1>My First Heading</h1> □ <p>My first paragraph.</p>

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
<p>	My first paragraph.	</p>
<br>	none	none

Web Browsers

The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them correctly.

A browser does not display the HTML tags, but uses them to determine how to display the document:



My First Heading

My first paragraph.

HTML Page Structure

Below is a visualization of an HTML page structure:

```
<html>
<head>
<title>Page title</title>

</head>

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

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

<p>This is another paragraph.</p>

</body>

</html>
```

Note: The content inside the <body> section (the white area above) will be displayed in a browser. The content inside the <title> element will be shown in the browser's title bar or in the page's tab.

HTML History

Since the early days of the World Wide Web, there have been many versions of HTML:

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2

1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1
2017	W3C Recommendation: HTML5.1 2nd Edition
2017	W3C Recommendation: HTML5.2

Step 1: Open Notepad (PC)

Windows 8 or later:

Open the Start Screen (the window symbol at the bottom left on your screen). Type Notepad.

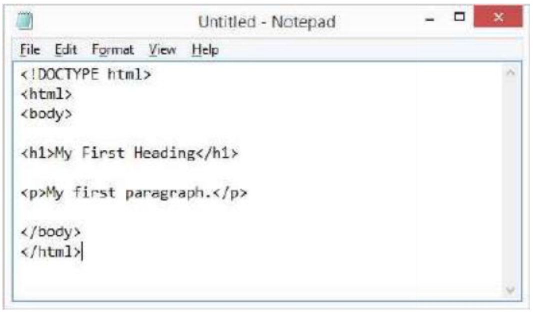
Then under "Open and Save", check the box that says "Display HTML files as HTML code instead of formatted text".

Then open a new document to place the code.

Step 2: Write Some HTML

Write or copy the following HTML code into Notepad:

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

<h1>My First Heading</h1>

<p>My first paragraph.</p>


</body>
</html>
```

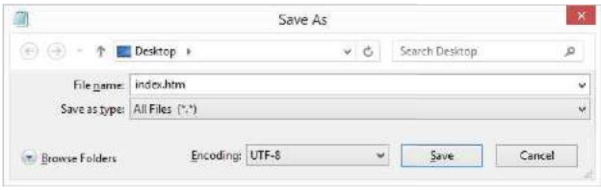


```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

Step 3: Save the HTML Page

Save the file on your computer. Select File > Save as in the Notepad menu.

Name the file "index.htm" and set the encoding to UTF-8 (which is the preferred encoding for HTML files).



File name: index.htm

Save as type: All Files (*.*)

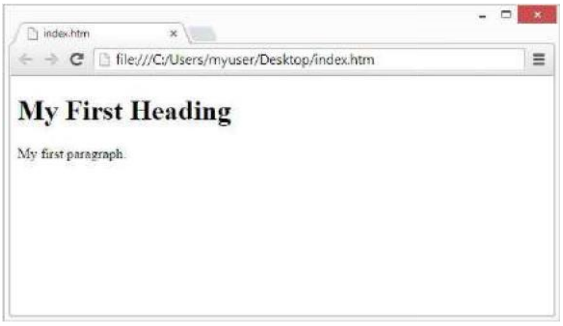
Encoding: UTF-8

You can either use .htm or .html as file extension. There is no difference.

Step 4: View the HTML Page in Your Browser

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").

The result will look much like this:



My First Heading

My first paragraph.

It is the perfect tool when you want to test code fast. It also has color coding and the ability to save and share code with others:

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
```

```
<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

HTML Documents

All HTML documents must start with a document type declaration: <!DOCTYPE html>.

The HTML document itself begins with <html> and ends with </html>.

The visible part of the HTML document is between <body> and </body>.

Example

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

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

The <!DOCTYPE> Declaration

The <!DOCTYPE> declaration represents the document type, and helps browsers to display web pages correctly.

It must only appear once, at the top of the page (before any HTML tags).

The <!DOCTYPE> declaration is not case sensitive.

The <!DOCTYPE> declaration for HTML5 is:

```
<!DOCTYPE html>
```

HTML Headings

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading:

Example

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
```

HTML Paragraphs

HTML paragraphs are defined with the <p> tag:

Example

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

HTML Links

HTML links are defined with the <a> tag:

Example:

<a href="https://www.tryschools.com">This is a

link</a> The link's destination is specified in the

href attribute.

Attributes are used to provide additional information about HTML elements.

You will learn more about attributes in a later chapter.

HTML Images

HTML images are defined with the <img> tag.

The source file (src), alternative text (alt), width, and height are provided as attributes:

Example:

```
<img src="tryschools.jpg" alt="TRYSchools.com" width="104" height="142">
```

How to View HTML
Source View HTML
Source Code:

Right-click in an HTML page and select "View Page Source" (in Chrome) or "View Source" (in Edge), or similar in other browsers. This will open a window containing the HTML source code of the page.

Inspect an HTML Element:

Right-click on an element (or a blank area), and choose "Inspect" or "Inspect Element" to see what elements are made up of (you will see both the HTML and the CSS). You can also edit the HTML or CSS on-the-fly in the Elements or Styles panel that opens.

HTML Elements

The HTML element is everything from the start tag to the end tag:

```
<tagname>Content goes here...</tagname>
```

Examples of some HTML elements:

```
<h1>My First Heading</h1>
```

```
<p>My first paragraph.</p>
```

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
<p>	My first paragraph.	</p>
<br>	none	none

Note: Empty elements have no end tags

Nested HTML Elements

HTML elements can be nested (this means that elements can contain other elements).

All HTML documents consist of nested HTML elements.

The following example contains four HTML elements (<html>, <body>, <h1> and <p>):

Example:

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

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

The <html> element is the root element and it defines the whole HTML document.

It has a start tag <html> and an end tag </html>.

Then, inside the <html> element there is a <body> element:

```
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
```

The <body> element defines the document's body.

It has a start tag <body> and an end tag </body>.

Then, inside the <body> element there are two other elements: <h1> and <p>:

```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```

The <h1> element defines a heading.

It has a start tag <h1> and an end tag </h1>:

```
<h1>My First Heading</h1>
```

The <p> element defines a paragraph.

It has a start tag <p> and an end tag </p>:

```
<p>My first paragraph.</p>
```

Never Skip the End Tag

Some HTML elements will display correctly, even if you forget the end tag:

Example:

```
<html>
<body>

<p>This is a paragraph
<p>This is a paragraph

</body
>
</html
>
```

However, never rely on this! Unexpected results and errors may occur if you forget the end tag!

Empty HTML Elements

HTML elements with no content are called empty elements.

The <br> tag defines a line break, and is an empty element without a closing tag:

Example:

```
<p>This is a <br> paragraph with a line break.</p>
```

HTML is Case Sensitive

HTML tags are not case sensitive: <P> means the same as <p>.

The HTML standard does not require lowercase tags, but W3C recommends lowercase in HTML, and demands lowercase for stricter document types like XHTML.

HTML Tag Reference

Tag	Description
<html>	Defines the root of an HTML document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings

HTML Attributes

- All HTML elements can have attributes
- Attributes provide additional information about elements
- Attributes are always specified in the start tag
- Attributes usually come in name/value pairs like: name="value"

The href Attribute

The <a> tag defines a hyperlink. The href attribute specifies the URL of the page the link goes to:

Example:

```
<a href="https://try.com">visit our school</a>
```

The src Attribute

The <img> tag is used to embed an image in an HTML page. The src attribute specifies the path to the image to be displayed:

Example:

```
<img src="img_girl.jpg">
```

There are two ways to specify the URL in the src attribute:

1. Absolute URL - Links to an external image that is hosted on another website. Example: src="https://www.tryschools.com/images/img_girl.jpg".

Notes: External images might be under copyright. If you do not get permission to use it, you may be in violation of copyright laws. In addition, you cannot control external images; it can suddenly be removed or changed.

2. Relative URL - Links to an image that is hosted within the website. Here, the URL does not include the domain name. If the URL begins without a slash, it will be relative to the current page. Example: src="img_girl.jpg". If the URL begins with a slash, it will be relative to the domain. Example: src="/images/img_girl.jpg".

Tip: It is almost always best to use relative URLs. They will not break if you change domain.

The width and height Attributes

The <img> tag should also contain the width and height attributes, which specify the width and height of the image (in pixels):

Example:

```
<img src="img_girl.jpg" width="500" height="600">
```

The alt Attribute

The required alt attribute for the <img> tag specifies an alternate text for an image, if the image for some reason cannot be displayed. This can be due to a slow connection, or an error in the src attribute, or if the user uses a screen reader.

Example:

```
<img src="img_girl.jpg" alt="Girl with a jacket">

<img src="img_typo.jpg" alt="Girl with a jacket">
```

The style Attribute

The style attribute is used to add styles to an element, such as color, font, size, and more.

Example:

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

The lang Attribute

You should always include the lang attribute inside the <html> tag, to declare the language of the Web page. This is meant to assist search engines and browsers.

The following example specifies English as the language:

```
<!DOCTYPE html>
<html lang="en">
<body>
...
</body>
</html>
```

Country codes can also be added to the language code in the lang attribute. So, the first two characters define the language of the HTML page, and the last two characters define the country.

The following example specifies English as the language and United States as the country:

```
<!DOCTYPE html>
<html lang="en-US">
<body>
...
</body>
</html>
```

The title Attribute

The title attribute defines some extra information about an element.

The value of the title attribute will be displayed as a tooltip when you mouse over the element:

Example:

```
<p title="I'm a tooltip">This is a paragraph.</p>
```

The HTML standard does not require lowercase attribute names.

The title attribute (and all other attributes) can be written with uppercase or lowercase like title or TITLE.

The HTML standard does not require quotes around attribute values.

Good:

```
<a href="https://www.try.com/html/">Visit our HTML tutorial</a>
```

Bad:

```
<a href=https://www.try.com/html/>Visit our HTML tutorial</a>
```

Sometimes you have to use quotes. This example will not display the title attribute correctly, because it contains a space:

Example:

```
<p title=About My

school>
```

Single or

Double Quotes?

Double quotes around attribute values are the most common in HTML, but single quotes can also be used.

In some situations, when the attribute value itself contains double quotes, it is necessary to use single

quotes: `<p title='John "ShotGun" Nelson'>`

Or vice versa:

```
<p title="John 'ShotGun' Nelson">
```

HTML Headings

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading.

Example:

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

Headings Are Important

Search engines use the headings to index the structure and content of your web pages.

Users often skim a page by its headings. It is important to use headings to show the document structure.

<h1> headings should be used for main headings, followed by <h2> headings, then the less important <h3>, and so on.

HTML Paragraphs

The HTML <p> element defines a paragraph.

A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph.

Example:

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

HTML Display

You cannot be sure how HTML will be displayed.

Large or small screens, and resized windows will create different results.

With HTML, you cannot change the display by adding extra spaces or extra lines in your HTML code.

The browser will automatically remove any extra spaces and lines when the page is displayed:

Example:

```
<p>
This
paragraph
contains a
lot of lines in
the source
code, but the
browser
ignores it.
</p>
```

```
<p>
This paragraph
contains         a lot
of spaces in the
source         code,
but the
browser ignores it.
</p>
```

HTML Horizontal Rules

The <hr> tag defines a thematic break in an HTML page, and is most often displayed as a horizontal rule.

The <hr> element is used to separate content (or define a change) in an HTML page:

Example:

```
<h1>This is heading 1</h1>
<p>This is some text.</p>
<hr>
<h2>This is heading 2</h2>
<p>This is some other text.</p>
<hr>
```

HTML Line Breaks

The HTML <br> element defines a line break.

Use <br> if you want a line break (a new line) without starting a new paragraph:

Example:

```
<p>This is<br>a paragraph<br>with line breaks.</p>
```

The Poem Problem

This poem will display on a single line:

Example:

```
<p>
  My Bonnie lies over the ocean.

  My Bonnie lies over the sea.

  My Bonnie lies over the ocean.

  Oh, bring back my Bonnie to me.
</p>
```

Solution - The HTML <pre> Element

The HTML <pre> element defines preformatted text.

The text inside a <pre> element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks:

Example:

```
<pre>
  My Bonnie lies over the ocean.

  My Bonnie lies over the sea.

  My Bonnie lies over the ocean.

  Oh, bring back my Bonnie to me.
</pre>
```

Tag	Description
<p>	Defines a paragraph
<hr>	Defines a thematic change in the content
<br>	Inserts a single line break
<pre>	Defines pre-formatted text

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;">
```

Background Color

The CSS background-color property defines the background color for an HTML element.

Example:

```
<body style="background-color:powderblue;">

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
```

Example:

Set background color for two different elements:

```
<body>
```

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

</body>
```

Text Color

The CSS color property defines the text color for an HTML element:

Example:

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

Fonts

The CSS font-family property defines the font to be used for an HTML element:

Example:

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

Text Size

The CSS font-size property defines the text size for an HTML element:

Example:

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

Text Alignment

The CSS text-align property defines the horizontal text alignment for an HTML element:

Example:

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

HTML Formatting Elements

Formatting elements were designed to display special types of text:

- <b> - Bold text
- <strong> - Important text
- <i> - Italic text
- <em> - Emphasized text
- <mark> - Marked text
- <small> - Smaller text
- <del> - Deleted text
- <ins> - Inserted text
- <sub> - Subscript text

- <sup> - Superscript text

HTML <b> and <strong> Elements

The HTML <b> element defines bold text, without any extra importance.

Example:

```
<b>This text is bold</b>
```

The HTML <strong> element defines text with strong importance. The content inside is typically displayed in bold.

Example:

```
<strong>This text is important!</strong>
```

HTML <i> and <em> Elements

The HTML <i> element defines a part of text in an alternate voice or mood. The content inside is typically displayed in italic.

Tip: The <i> tag is often used to indicate a technical term, a phrase from another language, a thought, a ship name, etc.

Example:

```
<i>This text is italic</i>
```

The HTML <em> element defines emphasized text. The content inside is typically displayed in italic.

Tip: A screen reader will pronounce the words in <em> with an emphasis, using verbal stress.

Example:

```
<em>This text is emphasized</em>
```

HTML <small> Element

The HTML <small> element defines smaller text:

Example:

```
<small>This is some smaller text.</small>
```

HTML <mark> Element

The HTML <mark> element defines text that should be marked or highlighted:

Example:

```
<p>Do not forget to buy <mark>milk</mark> today.</p>
```

HTML <del> Element

The HTML <del> element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text:

Example:

```
<p>My favorite color is <del>blue</del> red.</p>
```

HTML <ins> Element

The HTML <ins> element defines a text that has been inserted into a document. Browsers will usually underline inserted text:

Example:

```
<p>My favorite color is <del>blue</del> <ins>red</ins>.</p>
```

HTML <sub> Element

The HTML <sub> element defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas, like H_2O:

Example:

```
<p>This is <sub>subscripted</sub> text.</p>
```

HTML <sup> Element

The HTML <sup> element defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font. Superscript text can be used for footnotes, like WWW[1]:

Example:

```
<p>This is <sup>superscripted</sup> text.</p>
```

HTML Text Formatting Elements

Tag	Description
<b>	Defines bold text
<em>	Defines emphasized text
<i>	Defines a part of text in an alternate voice or mood
<small>	Defines smaller text
<strong>	Defines important text

<sub>	Defines subscripted text
<sup>	Defines superscripted text
<ins>	Defines inserted text
<del>	Defines deleted text
<mark>	Defines marked/highlighted text

HTML <blockquote> for Quotations

The HTML <blockquote> element defines a section that is quoted from another source.

Browsers usually indent <blockquote> elements.

Example:

```
<p>Here is a quote from WWF's website:</p>
<blockquote cite="http://www.worldwildlife.org/who/index.html">
For 60 years, WWF has worked to help people and nature thrive. As the world's leading conservation
organization, WWF works in nearly 100 countries. At every level, we collaborate with people around the
world to develop and deliver innovative solutions that protect communities, wildlife, and the places in
which they live.
</blockquote>
```

HTML <q> for Short Quotations

The HTML <q> tag defines a short quotation.

Browsers normally insert quotation marks around the quotation.

Example:

```
<p>WWF's goal is to: <q>Build a future where people live in harmony with nature.</q></p>
```

HTML <abbr> for Abbreviations

The HTML <abbr> tag defines an abbreviation or an acronym, like "HTML", "CSS", "Mr.", "Dr.", "ASAP", "ATM".

Marking abbreviations can give useful information to browsers, translation systems and search-engines.

Tip: Use the global title attribute to show the description for the abbreviation/acronym when you mouse over the element.

Example:

```
<p>The <abbr title="World Health Organization">WHO</abbr> was founded in 1948.</p>
```

HTML <address> for Contact Information

The HTML <address> tag defines the contact information for the author/owner of a document or an article.

The contact information can be an email address, URL, physical address, phone number, social media handle, etc.

The text in the <address> element usually renders in italic, and browsers will always add a line break before and after the <address> element.

Example:

```
<address>
Written by John Doe.<br>
Visit us at:<br>
Example.com<br>
Box 564, Disneyland<br>
USA
</address>
```

HTML <cite> for Work Title

The HTML <cite> tag defines the title of a creative work (e.g. a book, a poem, a song, a movie, a painting, a sculpture, etc.).

Note: A person's name is not the title of a work.

The text in the <cite> element usually renders in italic.

Example:

```
<p><cite>The Scream</cite> by Edvard Munch. Painted in 1893.</p>
```

HTML <bdo> for Bi-Directional

Override BDO stands for Bi-

Directional Override.

The HTML <bdo> tag is used to override the current text direction:

Example:

```
<bdo dir="rtl">This text will be written from right to left</bdo>
```

HTML Quotation and Citation Elements

Tag	Description
<abbr>	Defines an abbreviation or acronym
<address>	Defines contact information for the author/owner of a document

<bdo>	Defines the text direction
<blockquote>	Defines a section that is quoted from another source
<cite>	Defines the title of a work
<q>	Defines a short inline quotation

HTML Comment Tag

You can add comments to your HTML source by using the following syntax:

```
<!-- Write your comments here -->
```

Notice that there is an exclamation point (!) in the start tag, but not in the end tag.

Add Comments

With comments you can place notifications and reminders in your HTML code:

Example:

```
<!-- This is a comment -->

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

<!-- Remember to add more information here -->
```

Hide Content

Comments can be used to hide content.

This can be helpful if you hide content temporarily:

Example

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

<!-- <p>This is another paragraph </p> -->

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

Example

Hide a section of HTML code:

```
<p>This is a paragraph.</p>
<!--
<p>Look at this cool image:</p>
<img border="0" src="pic_trulli.jpg" alt="Trulli">
-->
```

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

Hide Inline Content

Comments can be used to hide parts in the middle of the HTML code.

Example:

```
<p>This <!-- great text --> is a paragraph.</p>
```

Background Color

You can set the background color for HTML elements:

Example:

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>
<p style="background-color:Tomato;">Lorem ipsum...</p>
```

Text Color

You can set the color of text:

Example

```
<h1 style="color:Tomato;">Hello World</h1>
<p style="color:DodgerBlue;">Lorem
ipsum...</p> <p
style="color:MediumSeaGreen;">Ut wisi
enim...</p> Example:
```

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

Color Values

In HTML, colors can also be specified using RGB values, HEX values, HSL values, RGBA values, and HSLA values.

The following three <div> elements have their background color set with RGB, HEX, and HSL values:

```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>
<h1 style="background-color:#ff6347;">...</h1>
<h1 style="background-color:hsl(9, 100%, 64%);">...</h1>

<h1 style="background-color:rgba(255, 99, 71, 0.5);">...</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">...</h1>
```

RGB Color Values

In HTML, a color can be specified as an RGB value, using this formula:

```
rgb(red, green, blue)
```

Each parameter (red, green, and blue) defines the intensity of the color with a value between 0 and 255.

This means that there are 256 x 256 x 256 = 16777216 possible colors!

For example, rgb (255, 0, 0) is displayed as red, because red is set to its highest value (255), and the other two (green and blue) are set to 0.

Another example, rgb(0, 255, 0) is displayed as green, because green is set to its highest value (255), and the other two (red and blue) are set to 0.

To display black, set all color parameters to 0, like this: rgb(0, 0, 0).

To display white, set all color parameters to 255, like this: rgb(255, 255, 255).

What is CSS?

Cascading Style Sheets (CSS) is used to format the layout of a webpage.

With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more!

Using CSS

CSS can be added to HTML documents in 3 ways:

- Inline - by using the style attribute inside HTML elements
- Internal - by using a <style> element in the <head> section
- External - by using a <link> element to link to an external CSS file

The most common way to add CSS, is to keep the styles in external CSS files. However, in this tutorial we will use inline and internal styles, because this is easier to demonstrate, and easier for you to try it yourself.

Inline CSS

An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the style attribute of an HTML element.

The following example sets the text color of the <h1> element to blue, and the text color of the <p> element to red:

Example:

```
<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>
```

Internal CSS

An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

The following example sets the text color of ALL the <h1> elements (on that page) to blue, and the text color of ALL the <p> elements to red. In addition, the page will be displayed with a "powderblue" background color:

Example:

```
<!DOCTYPE html>
<html>
<head
>
<style
>
body {background-color:
powderblue;} h1   {color: blue;}
```

```
p    {color: red;}
</style
>
</head
>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page:

Example:

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

The external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.

Here is what the "styles.css" file looks like:

“styles.css”:

```
body {
  background-color: powderblue;
}
h1
{
col
or:
blu
e;
}
p
{
```

```
co
lo
r:
re
d;
}
```

CSS Colors, Fonts and Sizes

Here, we will demonstrate some commonly used CSS properties. You will learn more about them later.

The CSS color property defines the text color to be used.

The CSS font-family property defines the font to be used.

The CSS font-size property defines the text size to be used.

Example:

```
<!DOCTYPE html>
<html>
<head>
<style> h1 {
color: blue;
font-family:
verdana;
font-size:
300%;
} p {   color:
red;   font-
family:
courier;
font-size:
160%;
}
</style
>
</head
>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

CSS Border

The CSS border property defines a border around an HTML element.

Tip: You can define a border for nearly all HTML elements.

Example:

```
p {   border: 2px solid
powderblue; }
```

CSS Padding

The CSS padding property defines a padding (space) between the text and the border.

Example:

```
p {   border: 2px solid
powderblue;
padding: 30px; }
```

CSS Margin

The CSS margin property defines a margin (space) outside the border.

Example:

```
p {   border: 2px solid
powderblue;   margin:
50px; }
```

Link to External CSS

External style sheets can be referenced with a full URL or with a path relative to the current web page.

Example:

```
<link rel="stylesheet" href="https://www.tryschools.com/html/styles.css">
```

Example

This example links to a style sheet located in the html folder on the current web site:

```
<link rel="stylesheet" href="/html/styles.css">
```

Example

This example links to a style sheet located in the same folder as the current page:

```
<link rel="stylesheet" href="styles.css">
```

HTML Links - Hyperlinks

HTML links are hyperlinks.

You can click on a link and jump to another document.

When you move the mouse over a link, the mouse arrow will turn into a little hand.

Note: A link does not have to be a text. A link can be an image or any other HTML element.

HTML Links - Syntax

The HTML <a> tag defines a hyperlink. It has the following syntax:

```
<a href="url">link text</a>
```

The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

The link text is the part that will be visible to the reader.

Clicking on the link text, will send the reader to the specified URL address.

Example:

<a href="https://www.try.com/">Visit my

school.com!</a> By default, links will appear as

follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

HTML Links - The target Attribute

By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.

The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

- _self - Default. Opens the document in the same window/tab as it was clicked
- _blank - Opens the document in a new window or tab
- _parent - Opens the document in the parent frame □ _top - Opens the document in the full body

  of the window Example:

```
<a href="https://www.try.com/" target="_blank">Visit my schools!</a>
```

Absolute URLs vs. Relative URLs

Both examples above are using an absolute URL (a full web address) in the href attribute.

A local link (a link to a page within the same website) is specified with a relative URL (without the "https://www" part):

Example:

```
<h2>Absolute URLs</h2>
<p><a href="https://www.try.org/">TRYC</a></p>
<p><a href="https://www.google.com/">Google</a></p>

<h2>Relative URLs</h2>
```

```
<p><a href="html_images.asp">HTML Images</a></p>
<p><a href="/css/default.asp">CSS Tutorial</a></p>
```

HTML Links - Use an Image as a Link

To use an image as a link, just put the <img> tag inside the <a> tag:

Example:

```
<a href="default.asp">
<img src="smiley.gif" alt="HTML tutorial"
style="width:42px;height:42px;"> </a>
```

Link to an Email Address

Use mailto: inside the href attribute to create a link that opens the user's email program (to let them send a new email):

Example:

```
<a href="mailto:someone@example.com">Send email</a>
```

Button as a Link

To use an HTML button as a link, you have to add some JavaScript code.

JavaScript allows you to specify what happens at certain events, such as a click of a button:

Example:

```
<button onclick="document.location='default.asp'">HTML Tutorial</button>
```

Link Titles

The title attribute specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.

Example:

```
<a href="https://www.tryschools.com/html/" title="Go to TRYSchools HTML section">Visit our HTML
Tutorial</a>
```

HTML Link Colors

By default, a link will appear like this (in all browsers):

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

You can change the link state colors, by using CSS:

Example: Here, an unvisited link will be green with no underline. A visited link will be pink with no underline. An active link will be yellow and underlined. In addition, when mouse is over a link (a:hover) it will become red and underlined:

```
<sty
le>
a:lin
k {
colo
r:
gree
n;
  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>
```

Link Buttons

A link can also be styled as a button, by using CSS:

Example:

```
<style>          a:link,
a:visited                {
background-color:
#f44336;
  color: white;
padding:
15px 25px;
text-align:
center;   text-
decoration:
none;
display:
inline-block;
}

a:hover, a:active {
background-color: red;
}
</style>
```

Create a Bookmark in HTML

Bookmarks can be useful if a web page is very long.

To create a bookmark - first create the bookmark, then add a link to it.

When the link is clicked, the page will scroll down or up to the location with the bookmark.

Example

First, use the id attribute to create a bookmark:

```
<h2 id="C4">Chapter 4</h2>
```

Example

```
<a href="#C4">Jump to Chapter 4</a>
```

You can also add a link to a bookmark on another page:

```
<a href="html_demo.html#C4">Jump to Chapter 4</a>
```

HTML Images

Example:

```
<img src="pic_trulli.jpg" alt="Italian Trulli">

<img src="img_girl.jpg" alt="Girl in a jacket">
```

HTML Images Syntax

The HTML <img> tag is used to embed an image in a web page.

Images are not technically inserted into a web page; images are linked to web pages. The <img> tag creates a holding space for the referenced image.

The <img> tag is empty, it contains attributes only, and does not have a closing tag.

The <img> tag has two required attributes:

- src - Specifies the path to the image
- alt - Specifies an alternate text for the image

Syntax:

```
<img src="url" alt="alternatetext">
```

The src Attribute

The required src attribute specifies the path (URL) to the image.

Note: When a web page loads, it is the browser, at that moment, that gets the image from a web server and inserts it into the page. Therefore, make sure that the image actually stays in the same spot in relation to the web page, otherwise your visitors will get a broken link icon. The broken link icon and the alt text are shown if the browser cannot find the image.

Example:

```
<img src="img_chania.jpg" alt="Flowers in Chania">
```

The alt Attribute

The required alt attribute provides an alternate text for an image, if the user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).

The value of the alt attribute should describe the image:

Example:

```
<img src="img_chania.jpg" alt="Flowers in Chania">
```

Image Size - Width and Height

You can use the style attribute to specify the width and height of an image.

Example:

```
<img src="img_girl.jpg" alt="Girl in a jacket"

style="width:500px;height:600px;">
```

Width and Height, or Style?

The width, height, and style attributes are all valid in HTML.

However, we suggest using the style attribute. It prevents styles sheets from changing the size of images:

Example:

```
<!DOCTYPE html>
<html>
<hea
d>
<styl
e>
img
{
widt
h:
100
%;
}
</style
>
```

```
</head
>
<body>

<img src="html5.gif" alt="HTML5 Icon" width="128" height="128">

<img src="html5.gif" alt="HTML5 Icon" style="width:128px;height:128px;">

</body>
</html>
```

Images in another folder

If you have your images in a sub-folder, you must include the folder name in the src attribute:

Example:

```
<img src="/images/html5.gif" alt="HTML5 Icon" style="width:128px;height:128px;">
```

Images on another Server/Website

Some web sites point to an image on another server.

To point to an image on another server, you must specify an absolute (full) URL in the src attribute:

Example:

```
<img src="https://www.try.com/images/my_green.jpg" alt="try.com">
```

Animated Images

HTML allows

animated GIFs:

Example:

```
<img src="programming.gif" alt="Computer Man" style="width:48px;height:48px;">
```

Image as a Link

To use an image as a link, put the <img> tag inside the <a> tag:

Example:

```
<a href="default.asp">
  <img src="smiley.gif" alt="HTML tutorial"
style="width:42px;height:42px;"> </a>
```

Image Floating

Use the CSS float property to let the image float to the right or to the left of a text:

Example:

```
<p><img src="smiley.gif" alt="Smiley face" style="float:right;width:42px;height:42px;">
The image will float to the right of the text.</p>
```

```
<p><img src="smiley.gif" alt="Smiley face" style="float:left;width:42px;height:42px;">
The image will float to the left of the text.</p>
```

Common Image Formats

HTML Image Tags

Tag	Description
<img>	Defines an image
<map>	Defines an image map
<area>	Defines a clickable area inside an image map
<picture>	Defines a container for multiple image resources

Image Maps

The HTML <map> tag defines an image map. An image map is an image with clickable areas. The areas are defined with one or more <area> tags.

Try to click on the computer, phone, or the cup of coffee in the image below:

Example

Here is the HTML source code for the image map above:

```
<img src="workplace.jpg" alt="Workplace" usemap="#workmap">

<map name="workmap">
  <area shape="rect" coords="34,44,270,350" alt="Computer" href="computer.htm">
  <area shape="rect" coords="290,172,333,250" alt="Phone" href="phone.htm">
  <area shape="circle" coords="337,300,44" alt="Coffee"
href="coffee.htm"> </map>
```

How does it Work?

The idea behind an image map is that you should be able to perform different actions depending on where in the image you click.

To create an image map you need an image, and some HTML code that describes the clickable areas.

The Image

The image is inserted using the <img> tag. The only difference from other images is that you must add a usemap attribute:

```
<img src="workplace.jpg" alt="Workplace" usemap="#workmap">
```

The usemap value starts with a hash tag # followed by the name of the image map, and is used to create a relationship between the image and the image map.

Create Image Map

Then, add a <map> element.

The <map> element is used to create an image map, and is linked to the image by using the required name

attribute: <map name="workmap">

The name attribute must have the same value as the <img>'s usemap attribute .

The Areas

Then, add the clickable areas.

A clickable area is defined using an <area> element.

Shape

You must define the shape of the clickable area, and you can choose one of these values:

- rect - defines a rectangular region
- circle - defines a circular region
- poly - defines a polygonal region
- default - defines the entire region

You must also define some coordinates to be able to place the clickable area onto the image.

Shape="rect"

The coordinates for shape="rect" come in pairs, one for the x-axis and one for the y-axis.

So, the coordinates 34,44 is located 34 pixels from the left margin and 44 pixels from the top:

Example:

```
<area shape="rect" coords="34, 44, 270, 350" href="computer.htm">
```

Shape="circle"

To add a circle area, first locate the coordinates of the center of the circle: 337,300

Example:

```
<area shape="circle" coords="337, 300, 44" href="coffee.htm">
```

Shape="poly"

The shape="poly" contains several coordinate points, which creates a shape formed with straight lines (a polygon).

This can be used to create any shape.

Example

```
<area shape="poly"
coords="140,121,181,116,204,160,204,222,191,270,140,329,85,355,58,352,37,322,40,259,103,
161,128,147" href="croissant.htm">
```

Image Map and JavaScript

A clickable area can also trigger a JavaScript function.

Add a click event to the <area> element to execute a JavaScript function:

Example:

```
<map name="workmap">
  <area shape="circle" coords="337,300,44" href="coffee.htm"
onclick="myFunction()"> </map>

<script>
function myFunction() {
alert("You clicked the
coffee cup!");
}
</script>
```

Background Image on a HTML element

To add a background image on an HTML element, use the HTML style attribute and the CSS backgroundimage property:

Example:

```
<p style="background-image: url('img_girl.jpg');">
```

You can also specify the background image in the <style> element, in the <head> section:

Example:

```
<
s
t
y
l
e
>
p
{
  background-image: url('img_girl.jpg');
}
</style>
```

Background Image on a Page

If you want the entire page to have a background image, you must specify the background image on the <body> element:

Example:

```
<st
yle
>
bod
y {
  background-image: url('img_girl.jpg');
}
</style>
```

Background Repeat

If the background image is smaller than the element, the image will repeat itself, horizontally and vertically, until it reaches the end of the element:

Example

```
<style>  body  {       background-image:
url('example_img_girl.jpg');
}
</style>
```

To avoid the background image from repeating itself, set the background-repeat property to no-repeat.

Example:

```
<st
yle
>
bod
y {
  background-image: url('example_img_girl.jpg');
background-repeat: no-repeat;
}
</style>
```

Background Cover

If you want the background image to cover the entire element, you can set the background-size property to

cover. Also, to make sure the entire element is always covered, set the background-attachment property to

fixed:

This way, the background image will cover the entire element, with no stretching (the image will keep its original proportions):

Example:

```
<st
yle
```

```
>
bod
y {
  background-image:
url('img_girl.jpg');
background-repeat: no-repeat;
background-attachment: fixed;
background-size: cover;
}
</style>
```

Background Stretch

If you want the background image to stretch to fit the entire element, you can set the background-size property to 100% 100%:

Example

```
<style> body {   background-
image: url('img_girl.jpg');
background-repeat: no-repeat;
background-attachment:
fixed;   background-size:
100% 100%;
}
</style>
```

The HTML <picture> Element

The HTML <picture> element gives web developers more flexibility in specifying image resources.

The <picture> element contains one or more <source> elements, each referring to different images through the srcset attribute. This way the browser can choose the image that best fits the current view and/or device.

Each <source> element has a media attribute that defines when the image is the most suitable.

Example:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_food.jpg">
  <source media="(min-width: 465px)" srcset="img_car.jpg">
  <img src="img_girl.jpg">
</picture>
```

When to use the Picture Element

There are two main purposes for the <picture> element:

1. Bandwidth

If you have a small screen or device, it is not necessary to load a large image file. The browser will use the first <source> element with matching attribute values, and ignore any of the following elements.

2. Format Support

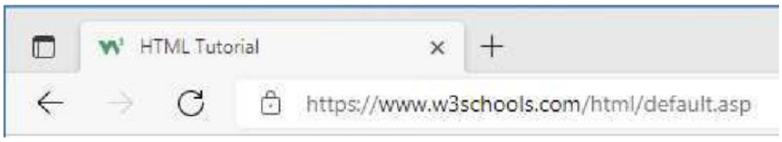
Some browsers or devices may not support all image formats. By using the <picture> element, you can add images of all formats, and the browser will use the first format it recognizes, and ignore any of the following elements. <picture>

```
  <source srcset="img_avatar.png">
  <source srcset="img_girl.jpg">
  <img src="img_beatles.gif" alt="Beatles" style="width:auto;">
</picture>
```

How To Add a Favicon in HTML

You can use any image you like as your favicon. You can also create your own favicon on sites like https://www.favicon.cc.

A favicon image is displayed to the left of the page title in the browser tab, like this:



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:

Example

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

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Favicon File Format Support

Define an HTML Table

A table in HTML consists of table cells inside rows and columns.

Example:

```
<table>
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
```

Table Cells

Each table cell 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 cell.

Example:

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</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>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>
```

Table Headers

Sometimes you want your cells to be table header cells. In those cases use the <th> tag instead of the <td> tag: th stands for table header.

Example:

```
<table>
  <tr>
    <th>Person 1</th>
    <th>Person 2</th>
    <th>Person 3</th>
  </tr>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>
```

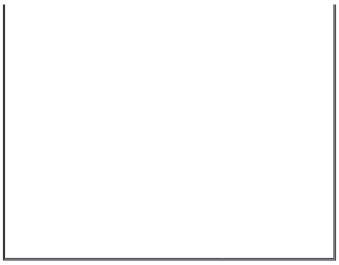
HTML Table Tags

Tag	Description
<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
<colgroup>	Specifies a group of one or more columns in a table for formatting

<col>	Specifies column properties for each column within a <colgroup> element
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table

How To Add a Border

When you add a border to a table, you also add borders around each table cell:



To add a border, use the CSS border property on table, th, and td elements:

Example:

```
table, th, td {
border: 1px
solid black; }
```

Collapsed Table Borders

To avoid having double borders like in the example above, set the CSS border-collapse property to collapse.

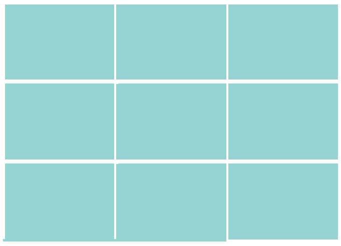
This will make the borders collapse into a single border:

Example:

```
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
```

Style Table Borders

If you set a background color of each cell, and give the border a white color (the same as the document background), you get the impression of an invisible border:

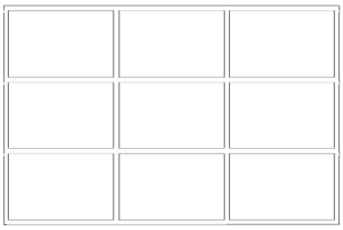


Example:

```
table, th, td {
border: 1px solid
white;   border-
collapse:
collapse;
}
t
h
,
t
d
{
  background-color:
#96D4D4; }
```

Round Table Borders

With the border-radius property, the borders get rounded corners:

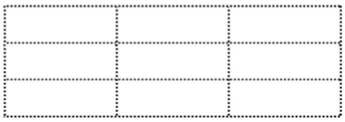


Example:

```
table, th, td {
border: 1px
solid black;
border-radius:
10px; }
```

Dotted Table Borders

With the border-style property, you can set the appearance of the border.



The following values are allowed:

Example

```
 th,    td    {
border-
style:
dotted; }
```

Border Color

With the border-color property, you can set the color of the border.

Example:

```
 th, td {
  border-color:
#96D4D4; }
```

HTML Table Width

To set the width of a table, add the style attribute to the <table> element:

Example:

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

To set the size of a specific column, add the style attribute on a <th> or <td> element:

Example:

```
<table style="width:100%">
  <tr>
    <th style="width:70%">Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
```

```
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

HTML Table Row Height

To set the height of a specific row, add the style attribute on a table row element:

Example:

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr style="height:200px">
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

HTML Table Headers

Table headers are defined with th elements. Each th element represents a table cell.

Example:

```
<table>
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Vertical Table Headers

To use the first column as table headers, define the first cell in each row as a <th> element:

Example:

```
<table>
  <tr>
    <th>Firstname</th>
    <td>Jill</td>
    <td>Eve</td>
  </tr>
  <tr>
    <th>Lastname</th>
    <td>Smith</td>
    <td>Jackson</td>
  </tr>
  <tr>
    <th>Age</th>
    <td>94</td>
    <td>50</td>
  </tr>
</table>
```

Align Table Headers

By default, table headers are bold and centered:

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

To left-align the table headers, use the CSS text-align property:

Example:

```
th {
text-
align:
left; }
```

Header for Multiple Columns

You can have a header that spans over two or more columns.

Name		Age
Jill	Smith	50
Eve	Jackson	94

To do this, use the colspan attribute on the <th> element:

Example:

```
<table>
  <tr>
    <th colspan="2">Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Table Caption

You can add a caption that serves as a heading for the entire table.

Monthly savings

Month	Savings
January	$100
February	$50

To add a caption to a table, use the <caption> tag:

Example:

```
<table style="width:100%">
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$50</td>
  </tr>
</table>
```

HTML Table Padding & Spacing

HTML tables can adjust the padding inside the cells, and also the space between the cells.

With Padding

hello	hello	hello
hello	hello	hello
hello	hello	hello

With Spacing

hello	hello	hello
hello	hello	hello
hello	hello	hello

HTML Table - Cell Padding

Cell padding is the space between the cell edges and the cell content.

By default the padding is set to 0.

To add padding on table cells, use the CSS padding property:

Example:

```
th, td {
paddin
g:
15px;
}
```

To add padding only above the content, use the padding-top property.

And the others sides with the padding-bottom, padding-left, and padding-right properties:

Example:

```
th, td {
padding-top:
10px;
padding-
bottom: 20px;
padding-left:
30px;
padding-right:
40px; }
```

HTML Table - Cell Spacing

Cell spacing is the space between each cell.

By default the space is set to 2 pixels.

To change the space between table cells, use the CSS border-spacing property on the table element:

Example:

```
table {
border-
spacing:
30px; }
```

HTML Table Colspan & Rowspan

HTML tables can have cells that span over multiple rows and/or columns.

NAME		

	APRIL		

2022		
FIESTA		

HTML Table - Colspan

To make a cell span over multiple columns, use the colspan attribute:

Example:

```
<table>
  <tr>
    <th colspan="2">Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>43</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>57</td>
  </tr>
</table>
```

HTML Table - Rowspan

To make a cell span over multiple rows, use the rowspan attribute:

Example:

```
<table>
  <tr>
    <th>Name</th>
    <td>Jill</td>
  </tr>
  <tr>
    <th rowspan="2">Phone</th>
    <td>555-1234</td>
  </tr>
  <tr>
    <td>555-8745</td>
</tr>
</table>
```

HTML Table - Zebra Stripes

If you add a background color on every other table row, you will get a nice zebra stripes effect.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20

To style every other table row element, use the :nth-child(even) selector like this:

Example:

```
tr:nth-child(even) {
background-color:
#D6EEEE; }
```

HTML Table - Vertical Zebra Stripes

To make vertical zebra stripes, style every other column, instead of every other row.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20

Set the :nth-child(even) for table data elements like this:

Example:

```
td:nth-child(even),       th:nth-
child(even)  {     background-
color: #D6EEEE; }
```

Combine Vertical and Horizontal Zebra Stripes

You can combine the styling from the two examples above and you will have stripes on every other row and every other column.

If you use a transparent color you will get an overlapping effect.

Use an rgba() color to specify the transparency of the color:

Example:

```
tr:nth-child(even) {
  background-color: rgba(150, 212, 212, 0.4);
}

th:nth-child(even),td:nth-
child(even)  {      background-color:
rgba(150, 212, 212, 0.4); }
```

Horizontal Dividers

First Name	Last Name	Savings
Peter	Griffin	$100

If you specify borders only at the bottom of each table row, you will have a table with horizontal dividers.

Add the border-bottom property to all tr elements to get horizontal dividers:

```
tr {
  border-bottom: 1px
solid #ddd; }
```

Hoverable Table

Use the :hover selector on tr to highlight table rows on mouse over:

First Name	Last Name	Savings
Peter	Griffin	$100

```
tr:hover {background-color: #D6EEEE;}
```

HTML Table Colgroup

If you want to style the two first columns of a table, use the <colgroup> and <col> elements.

MON	TUE	WED	THU	FRI	SAT	SUN
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

The <colgroup> element should be used as a container for the column specifications.

Each group is specified with a <col> element.

The span attribute specifies how many columns that get the style.

The style attribute specifies the style to give the columns.

Example

```
<table>
  <colgroup>
    <col span="2" style="background-color: #D6EEEE">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
...
```

Multiple Col Elements

If you want to style more columns with different styles, use more <col> elements inside the <colgroup>:

```
<table>
  <colgroup>
    <col span="2" style="background-color: #D6EEEE">
    <col span="3" style="background-color: pink">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
...
```

Empty Colgroups

If you want to style columns in the middle of a table, insert a "empty" <col> element (with no styles) for the columns before:

Example:

```
<table>
  <colgroup>
    <col span="3">
    <col span="2" style="background-color: pink">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
...
```

Hide Columns

You can hide columns with the visibility: collapse property:

Example:

```
<table>
  <colgroup>
    <col span="2">
    <col span="3" style="visibility: collapse">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
...
```

HTML Lists

HTML lists allow web developers to group a set of related items in lists.

Example : an unordered HTML list:

- Item
- Item
- Item

An ordered HTML list:

1. First item
2. Second item
3. Third item
4. Fourth item

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:

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Ordered HTML List

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

The list items will be marked with numbers by default:

Example:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

HTML Description Lists

HTML also supports description lists.

A description list is a list of terms, with a description of each term.

The <dl> tag defines the description list, the <dt> tag defines the term (name), and the <dd> tag describes each term:

```
<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>
```

HTML List Tags

Tag	Description
<ul>	Defines an unordered list
<ol>	Defines an ordered list
<li>	Defines a list item
<dl>	Defines a description list
<dt>	Defines a term in a description list
<dd>	Describes the term in a description list

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:

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Unordered HTML List - Choose List Item Marker

The CSS list-style-type property is used to define the style of the list item marker. It can have one of the following values:

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: Disc

```
<ul style="list-style-type:disc;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Example - Circle

```
<ul style="list-style-type:circle;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Example - Square

```
<ul style="list-style-type:square;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Example - None

```
<ul style="list-style-type:none;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Nested HTML Lists

Lists can be nested (list inside list):

```
<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  </li>
  <li>Milk</li>
</ul>
```

Horizontal List with CSS

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

One popular way is to style a list horizontally, to create a navigation menu:

```
<!DOCTYPE html>
<html>
<head> <style> ul {
list-style-type:
none;   margin: 0;
padding: 0;
overflow: hidden;
background-color:
#333333;
}
 li
{
fl
o
at
:
le
ft
;
}

li a {
display:
block;   color:
white;   text-
align: center;
padding:
16px;   text-
decoration:
none;
}

li a:hover {
background-color:
#111111;
}
</style
>
</head
>
<body>
```

```
<ul>
  <li><a href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>

</body>
</html>
```

Ordered HTML List

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

The list items will be marked with numbers by default:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</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

```
<ol type="1">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Uppercase Letters:

```
<ol type="A">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Lowercase Letters:

```
<ol type="a">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Uppercase Roman Numbers:

```
<ol type="I">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Lowercase Roman Numbers:

```
<ol type="i">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Control List Counting

By default, an ordered list will start counting from 1. If you want to start counting from a specified number, you can use the start attribute:

```
<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

HTML Description Lists

A description list is a list of terms, with a description of each term.

The <dl> tag defines the description list, the <dt> tag defines the term (name), and the <dd> tag describes each term:

```
<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>
```

Block-level Elements

A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.

A block-level element always takes up the full width available (stretches out to the left and right as far as it can).

Two commonly used block elements are: <p> and <div>.

The <p> element defines a paragraph in an HTML document.

The <div> element defines a division or a section in an HTML document.

The <p> element is a block-level element.

The <div> element is a block-level element.

Example:

```
<p>Hello World</p>
<div>Hello World</div>
```

Here are the block-level elements in HTML:

```
<address><article><aside><blockquote><canvas>

<dd><div><dl><dt><fieldset><figcaption>

<figure><footer><form><h1>-<h6><header>

<hr><li><main><nav><noscript><ol>

<p><pre><section><table><tfoot><ul><video>
```

Inline Elements

An inline element does not start on a new line.

An inline element only takes up as much width as necessary.

This is a <span> element inside a paragraph.

```
<span>Hello World</span>
```

Here are the inline elements in HTML:

```
<a><abbr><acronym><b><bdo><big><br><button>
<cite><code><dfn><em><i><img><input><kbd>
<label><map><object><output><q><samp>
<script><select><small><span><strong>
<sub><sup><textarea><time><tt><var>
```

The <div> Element

The <div> element is often used as a container for other HTML elements.

The <div> element has no required attributes, but style, class and id are common.

When used together with CSS, the <div> element can be used to style blocks of content:

```
<div style="background-color:black;color:white;padding:20px;">
  <h2>London</h2>
  <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a
metropolitan area of over 13 million inhabitants.</p> </div>
```

The <span> Element

The <span> element is an inline container used to mark up a part of a text, or a part of a document.

The <span> element has no required attributes, but style, class and id are common.

When used together with CSS, the <span> element can be used to style parts of the text:

```
<p>My mother has <span style="color:blue;font-weight:bold;">blue</span> eyes and my
father has <span style="color:darkolivegreen;font-weight:bold;">dark green</span>
eyes.</p>
```

Using The class Attribute

The class attribute is often used to point to a class name in a style sheet. It can also be used by a JavaScript to access and manipulate elements with the specific class name.

In the following example we have three <div> elements with a class attribute with the value of "city". All of the three <div> elements will be styled equally according to the .city style definition in the head section:

```
<!DOCTYPE html>
<html>
<head>
<st
yle
>
.cit
y {
  background-
color: tomato;
color: white;
border: 2px solid
black;   margin:
20px;   padding:
20px;
}
</style
>
</head
>
<body>

<div class="city">
```

```
  <h2>London</h2>
  <p>London is the capital of England.</p>
</div>

<div class="city">
  <h2>Paris</h2>
  <p>Paris is the capital of France.</p>
</div>

<div class="city">
  <h2>Tokyo</h2>
  <p>Tokyo is the capital of Japan.</p>
</div>

</body
>
</html
>
```

In the following example we have two <span> elements with a class attribute with the value of "note". Both <span> elements will be styled equally according to the .note style definition in the head section:

```
<!DOCTYPE html>
<html>
<head>
<style>
.note {
font-
size:
120%;
color:
red; }
</style
>
</head
>
<body>

<h1>My <span class="note">Important</span> Heading</h1>
<p>This is some <span class="note">important</span> text.</p>

</body>
</html>
```

The Syntax For Class

To create a class; write a period (.) character, followed by a class name. Then, define the CSS properties within curly braces {}:

Example: Create a Class named ‘City’

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

```
<st
yle
>
.cit
y {
  background-
color: tomato;
color: white;
padding: 10px;
}
</style
>
</head
>
<body>

<h2 class="city">London</h2>
<p>London is the capital of England.</p>

<h2 class="city">Paris</h2>
<p>Paris is the capital of France.</p>

<h2 class="city">Tokyo</h2>
<p>Tokyo is the capital of Japan.</p>

</body>
</html>
```

Multiple Classes

HTML elements can belong to more than one class.

To define multiple classes, separate the class names with a space, e.g. <div class="city main">. The element will be styled according to all the classes specified.

In the following example, the first <h2> element belongs to both the city class and also to the main class, and will get the CSS styles from both of the classes:

```
<h2 class="city main">London</h2>
<h2 class="city">Paris</h2>
<h2 class="city">Tokyo</h2>
```

Different Elements Can Share Same Class

Different HTML elements can point to the same class name.

In the following example, both <h2> and <p> point to the "city" class and will share the same style:

```
<h2 class="city">Paris</h2>
<p class="city">Paris is the capital of France</p>
```

Using The id Attribute

The id attribute specifies a unique id for an HTML element. The value of the id attribute must be unique within the HTML document.

The id attribute is used to point to a specific style declaration in a style sheet. It is also used by JavaScript to access and manipulate the element with the specific id.

The syntax for id is: write a hash character (#), followed by an id name. Then, define the CSS properties within curly braces {}.

In the following example we have an <h1> element that points to the id name "myHeader". This <h1> element will be styled according to the #myHeader style definition in the head section:

```
<!DOCTYPE html>
<html>
<head>
<style>
#myHeader {
background-color:
lightblue;   color:
black;   padding:
40px;   text-align:
center;
}
</style
>
</head
>
<body>

<h1 id="myHeader">My Header</h1>

</body>
</html>
```

Difference Between Class and ID

A class name can be used by multiple HTML elements, while an id name must only be used by one HTML element within the page:

```
<style>
/* Style the element with the id "myHeader" */
#myHeader {
 background-color:
lightblue;   color:
black;   padding:
40px;   text-align:
center;
}

/* Style all elements with the class name "city" */
.city {
background-color:
tomato;   color:
```

```
white;   padding:
10px;
}
</style>

<!-- An element with a unique id -->
<h1 id="myHeader">My Cities</h1>

<!-- Multiple elements with same class -->
<h2 class="city">London</h2>
<p>London is the capital of England.</p>

<h2 class="city">Paris</h2>
<p>Paris is the capital of France.</p>

<h2 class="city">Tokyo</h2>
<p>Tokyo is the capital of Japan.</p>
```

HTML Bookmarks with ID and Links

HTML bookmarks are used to allow readers to jump to specific parts of a webpage.

Bookmarks can be useful if your page is very long.

To use a bookmark, you must first create it, and then add a link to it.

Then, when the link is clicked, the page will scroll to the location with the bookmark.

Example

First, create a bookmark with the id attribute:

```
<h2 id="C4">Chapter 4</h2>
```

Then, add a link to the bookmark ("Jump to Chapter 4"), from within the same page:

```
<a href="#C4">Jump to Chapter 4</a>
```

HTML Iframe Syntax

The HTML <iframe> tag specifies an inline frame.

An inline frame is used to embed another document within the current HTML document.

```
<iframe src="url" title="description"></iframe>
```

Tip: It is a good practice to always include a title attribute for the <iframe>. This is used by screen readers to read out what the content of the iframe is.

Iframe - Set Height and Width

Use the height and width attributes to specify the size of the iframe.

The height and width are specified in pixels by default:

```
<iframe src="demo_iframe.htm" height="200" width="300" title="Iframe Example"></iframe>
```

```
<iframe src="demo_iframe.htm" style="height:200px;width:300px;" title="Iframe Example"></iframe>
```

Iframe - Remove the Border

By default, an iframe has a border around it.

To remove the border, add the style attribute and use the CSS border property:

```
<iframe src="demo_iframe.htm" style="border:none;" title="Iframe Example"></iframe>
```

Iframe - Target for a Link

An iframe can be used as the target frame for a link.

The target attribute of the link must refer to the name attribute of the iframe:

```
<iframe src="demo_iframe.htm" name="iframe_a" title="Iframe Example"></iframe>
```

```
<p><a href="https://www.try.com" target="iframe_a">try.com</a></p>
```