- HTML INTRODUCTION:
  The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph
- HTML tags normally come **in pairs** like and

*<h1> defines the most important heading. <h6> defines the least important heading:
*HTML links are defined with the **<a>** tag:
*<a href ="www.w3school.com">this is a link </a>
*Attributes are used to provide additional information about HTML elements.
*The source file (src), alternative text (alt), width, and height are provided as attributes:

HTML EDITORS:
*<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">
this command is for image adding.
*An HTML element usually consists of a **start** tag and **end** tag, with the content inserted in between:

HTML ELEMENTS:
<tagname>Content goes here...</tagname>

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

* the closing tag is considered optional.
*HTML elements with no content are called empty elements.

 is an empty element without a closing tag (the <br> tag defines a line break).

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

The HTML5 standard does not require lowercase tags, but
W3C **recommends** lowercase in HTML, and **demands**lowercase for stricter

document types like XHTML.

HTML ATTRIBUTES:

- Attributes provide **additional information** about an element
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

*The **alt** attribute specifies an alternative text to be used, when an image cannot be displayed.
*The HTML5 standard does not require quotes around attribute values.
*In some situations, when the attribute value itself contains double quotes, it is necessary to use single quotes:

*All HTML Attributes:
Check the link https://www.w3schools.com/tags/ref_attributes.asp
*The <hr> element is used to separate content (or define a change) in an HTML page:
*With HTML, you cannot change the output by adding extra spaces or extra lines in your HTML code.

HTML PARAGRAPHS:

The browser will remove any extra spaces and extra lines when the page is displayed:
*The HTML <pre> element defines preformatted text.
*

<p>	Defines a paragraph
 	Inserts a single line break
<pre>	Defines pre-formatted text

HTML STYLES:

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

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

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

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

Example

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

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

Example

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

*Formatting elements were designed to display special types of text:

HTML FORMATTING:

- <b> - Bold text
- - Important text
- - Italic text
- - Emphasized text
- <mark> - Marked text
- <small> - Small text
- <del> - Deleted text
- <ins> - Inserted text
- <sub> - Subscript text
- <sup> - Superscript text

*This is a strong

This text is italic

This text is emphasized

<h5>HTML <small>Small</small> Formatting</h2>

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

My favorite <ins>color</ins> is red.

This is _{subscripted} text.

This is ^{superscripted} text.

*__This is a bold__

__This is a strong__ *This text is italic This text is emphasized*

HTML Small Formatting

My favorite color is ~~blue~~ red.

My favorite color is red.

This is subscripted text.

This is superscripted text

HTML QUOTATIONS:

*.The HTML **<q>** element defines a short quotation.

Browsers usually insert quotation marks around the <q> element.

Example

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

*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 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>
```

*The HTML **<abbr>** element defines an abbreviation or an acronym.

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

*The HTML **<address>** element defines contact information (author/owner) of a document or an article.

The <address> element is usually displayed in italic. Most browsers will add a line break before and after the element.

Example

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

Written by John Doe,
Visit us at:
Example.com
Box 564,Disneyland
USA

*HTML <cite> for Work Title

The HTML **<cite>** element defines the title of a work.

Browsers usually display <cite> elements in italic.

Example

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

Try it Yourself »

*

<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 COMMENTS:

*Notice that there is an exclamation point (!) in the opening tag, but not in the closing tag.

Note: Comments are not displayed by the browser, but they can help document your HTML source code.

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

Example

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

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

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

Try it Yourself »

HTML COLOURS:

*

Color	Name
	Red
	Orange
	Yellow
	Cyan
	Blue

*

In HTML, a color can also 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 between 0 and 255.

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

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

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

*Example

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

Try it Yourself »

Shades of gray are often defined using equal values for all the 3 light sources:

Example

Color	RGB
	rgb(0,0,0)
	rgb(90,90,90)
	rgb(128,128,128)
	rgb(200,200,200)
	rgb(255,255,255)

HTML CSS:

* **CSS** stands for **C**ascading **S**tyle **S**heets.

CSS describes **how HTML elements are to be displayed on screen, paper, or in other media**.

CSS **saves a lot of work**. It can control the layout of multiple web pages all at once.

CSS can be added to HTML elements in 3 ways:

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

HTML LINKS:

*In HTML, links are defined with the **<a>** tag:

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

- *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 default colors, by using styles:

Example

```
<style>
a:link {
    color: green;
    background-color: transparent;
    text-decoration: none;
}

a:visited {
    color: pink;
    background-color: transparent;
    text-decoration: none;
}

a:hover {
    color: red;
    background-color: transparent;
    text-decoration: underline;
}

a:active {
    color: yellow;
    background-color: transparent;
    text-decoration: underline;
}
</style>
```

*The **target** attribute specifies where to open the linked document.

The target attribute can have one of the following values:

- _blank - Opens the linked document in a new window or tab
- _self - Opens the linked document in the same window/tab as it was clicked (this is default)
- _parent - Opens the linked document in the parent frame
- _top - Opens the linked document in the full body of the window
- framename - Opens the linked document in a named frame

This example will open the linked document in a new browser window/tab:

Example

```
<a href="https://www.w3schools.com/"target="_blank">Visit W3Schools!</a>
```

Try it Yourself »

* If your webpage is locked in a frame, you can use target="_top" to break out of the frame:

Example

```
<ahref="https://www.w3schools.com/html/"target="_top">HTML5
tutorial!</a>
```

*First, create a bookmark with the id attribute:

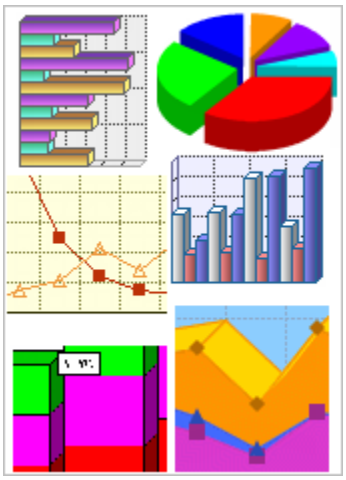
```
<h2 id="C4">Chapter 4</h2>
```

HTML IMAGES:

*JPG Images

GIF Images

PNG Images



*In HTML, images are defined with the **<img>** tag.

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

The src attribute specifies the URL (web address) of the image:

```
<img src="url" alt="some_text"style="width:width;height:height;">
```

*The 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).

If a browser cannot find an image, it will display the value of the alt attribute: The alt attribute is required. A web page will not validate correctly without it.

*HTML Screen Readers

A screen reader is a software program that reads the HTML code, converts the text, and allows the user to "listen" to the content. Screen readers are useful for people who are blind, visually impaired, or learning disabled.

***Note:** Always specify the width and height of an image. If width and height are not specified, the page will flicker while the image loads.

*Both the width, height, and style attributes are valid in HTML5.

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

*Images in Another Folder

If not specified, the browser expects to find the image in the same folder as the web page.

However, it is common to store images in a sub-folder. You must then include the folder name in the src attribute:

Example

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

Try it Yourself »

*Animated Images

The GIF standard allows animated images:

Example

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

Try it Yourself »

Note that the syntax of inserting animated images is no different from non-animated images.

Using an Image as a Link

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

Example

```
<a href="default.asp">
  <img src="smiley.gif" alt="HTML
tutorial"style="width:42px;height:42px;border:0;">
</a>
```

*Image Floating

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

Example

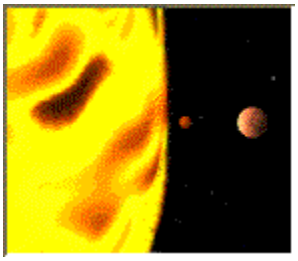
```
<p><img src="smiley.gif" alt="Smiley
face"style="float:right;width:42px;height:42px;">
The image will float to the right of the text.</p>

<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>
```

Try it Yourself »

* Click on the sun or on one of the planets to watch it closer:

```
<img src="planets.gif" alt="Planets" usemap="#planetmap"
style="width:145px;height:126px;">


<map name="planetmap">

  <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">

  <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm">

  <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">

</map>


</body>

</html>
```

Click on the sun or on one of the planets to watch it closer:



HTML TABLES:

*Defining an HTML Table

An HTML table is defined with the **<table>** tag.

Each table row is defined with the **<tr>** tag. A table header is defined with the **<th>** tag. By default, table headings are bold and centered. A table data/cell is defined with the **<td>** tag.

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>
```

Try it Yourself »

Note: The <td> elements are the data containers of the table.
They can contain all sorts of HTML elements; text, images, lists, other tables, etc.

*A border is set using the CSS **border** property:

Example

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

*Padding

Cell padding specifies the space between the cell content and its borders.

If you do not specify a padding, the table cells will be displayed without padding.

To set the padding, use the CSS **padding** property:

Example

```
th, td {
    padding: 15px;
}
```

*HTML Table - Left-align Headings

By default, table headings are bold and centered.

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

Example

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

*HTML Table - Adding Border Spacing

Border spacing specifies the space between the cells.

To set the border spacing for a table, use the CSS **border-spacing** property:

Example

```
table {
    border-spacing: 5px;
}
```

*To make a cell span more than one column, use the **colspan** attribute:

Example

```
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gates</td>
```

```
    <td>55577854</td>
    <td>55577855</td>
  </tr>
</table>
```

*HTML Table - Cells that Span Many Rows

To make a cell span more than one row, use the **rowspan** attribute:

Example

```
<table style="width:100%">
  <tr>
    <th>Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>55577854</td>
  </tr>
  <tr>
    <td>55577855</td>
  </tr>
</table>
```

Try it Yourself »

*HTML Table - Adding a Caption

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>
```

Try it Yourself »

Note: The <caption> tag must be inserted immediately after the <table> tag.

*A Special Style for One Table

To define a special style for a special table, add an **id** attribute to the table:

```
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}

th, td {

    padding: 5px;

    text-align: left;

}

table#t01 {

    width: 100%;
```

```
    background-color: #f1f1c1;
}
</style>
</head>
<body>

<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>
```

```
  <tr>
    <td>John</td>
    <td>Doe</td>
    <td>80</td>
  </tr>
</table>
<br>

<table id="t01">
  <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>
  <tr>
    <td>John</td>
    <td>Doe</td>
    <td>80</td>
  </tr>
</table>


</body>
</html>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

HTML LISTS:

*An Unordered List:

- Item
- Item
- Item
- Item

An Ordered List:

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

*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>
```

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">[***Always give – after  list]
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

*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

Numbers:

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

Try it Yourself »

*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:

Example

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

*Nested HTML Lists

List can be nested (lists inside lists):

Example

```
<ul>
  <li>Coffee</li>
```

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

Try it Yourself »

***The last element of the table was out of mind.So we will see it leter.**

HTML BLOCKAND INLINE ELEMENTS:

*Block-level Elements

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

The element is a block-level element.

Examples of block-level elements:

-
- <h1> - <h6>
- <p>
- <form>

*The Element

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

The element has no required attributes, but both **style** and **class** are common.

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

Example

```
<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>
```

Try it Yourself »

The Element

The element is often used as a container for some text.

The element has no required attributes, but both **style** and **class** are common.

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

Example

```
<h1>My <spanstyle="color:red">Important</span>Heading</h1>
```

HTML THE CLASS ATTRIBUTES:

*Using The class Attribute

The HTML class attribute makes it possible to define equal styles for elements with the same class name.

Here we have three elements that point to the same class name:

Example

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

```
<style>
div.cities {
    background-color: black;
    color: white;
    margin: 20px 0 20px 0;
    padding: 20px;
}
</style>
</head>
<body>

<div class="cities">
<h2>London</h2>
<p>London is the capital of England. It is the most populous city in the
United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
</div>

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

<div class="cities">
<h2>Tokyo</h2>
<p>Tokyo is the capital of Japan, the center of the Greater Tokyo Area,
and the most populous metropolitan area in the world.</p>
</div>

</body>
</html>
```

Try it Yourself »

Paris

Paris is the capital and most populous city of France.

Situated on the Seine River, it is at the heart of the Île-de-France region, also known as the région parisienne.

Within its metropolitan area is one of the largest population centers in Europe, with over 12 million inhabitants.

Tokyo

Tokyo is the capital of Japan, the center of the Greater Tokyo Area, and the most populous metropolitan area in the world.

It is the seat of the Japanese government and the Imperial Palace, and the home of the Japanese Imperial Family.

The Tokyo prefecture is part of the world's most populous metropolitan area with 38 million people and the world's largest urban economy.

HTML IFRAMES:

*Iframe Syntax

An HTML iframe is defined with the **<iframe>** tag:

```
<iframe src="URL"></iframe>
```

The **src** attribute specifies the URL (web address) of the inline frame page.

Iframe - Set Height and Width

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

The attribute values are specified in pixels by default, but they can also be in percent (like "80%").

Example

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

Try it Yourself »

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:

Example

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

Try it Yourself »

With CSS, you can also change the size, style and color of the iframe's border:

Example

```
<iframe src="demo_iframe.htm"style="border:2px solid grey;"></iframe>
```

Try it Yourself »

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:

Example

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

<p><a href="https://www.w3schools.com"target="iframe_a">W3Schools.com</a></p>
```

Try it Yourself »

HTML JAVASCRIPT:

***JavaScript makes HTML pages more dynamic and interactive.**

***The** <script> **tag is used to define a client-side script (JavaScript). Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.**

HTML File Paths:

*

Path	Description
<img src="picture.jpg">	picture.jpg is located in the same folder as the current page
<img src="images/picture.jpg">	picture.jpg is located in the images folder located in the current folder

<img src="/images/picture.jpg">	picture.jpg is located in the images folder located at the root of the current web
<img src="../picture.jpg">	picture.jpg is located in the folder one level up from the current folder

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

- Web pages
- Images
- Style sheets
- JavaScripts

*It is a best practice to use relative file paths (if possible).

When using relative file paths, your web pages will not be bound to your current base URL. All links will work on your own computer (localhost) as well as on your current public domain and your future public domains.

HTML Head:

*The HTML <head> Element

The **<head>** element is a container for metadata (data about data) and is placed between the <html> tag and the <body> tag.

HTML metadata is data about the HTML document. Metadata is not displayed.

Metadata typically define the document title, character set, styles, links, scripts, and other meta information.

The following tags describe metadata: <title>, <style>, <meta>, <link>, <script>, and <base>.

*The <title> element:

- defines a title in the browser tab
- provides a title for the page when it is added to favorites
- displays a title for the page in search engine results

*The **<meta>** element is used to specify which character set is used, page description, keywords, author, and other metadata.

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

Define the character set used:

```
<meta charset="UTF-8">
```

Define a description of your web page:

```
<meta name="description" content="Free Web tutorials">
```

Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, XML, JavaScript">
```

Define the author of a page:

```
<meta name="author" content="John Doe">
```

Refresh document every 30 seconds:

```
<meta http-equiv="refresh"content="30">
```

*HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

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

You should include the following <meta> viewport element in all your web pages:

```
<meta name="viewport"content="width=device-width, initial-scale=1.0">
```

*The <script> element is used to define client-side JavaScripts.

This JavaScript writes "Hello JavaScript!" into an HTML element with id="demo":

Example

```
<script>
function myFunction {
   document.getElementById("demo").innerHTML = "Hello JavaScript!";
}
</script>
```

*The <base> element specifies the base URL and base target for all relative URLs in a page:

Example

```
<basehref="https://www.w3schools.com/images/" target="_blank">
```

*Omitting <html>, <head> and <body>?

According to the HTML5 standard; the <html>, the <body>, and the <head> tag can be omitted.

The following code will validate as HTML5:

*W3Schools does not recommend omitting the <html> and <body> tags. Omitting these tags can crash DOM or XML software and produce errors in older browsers (IE9).

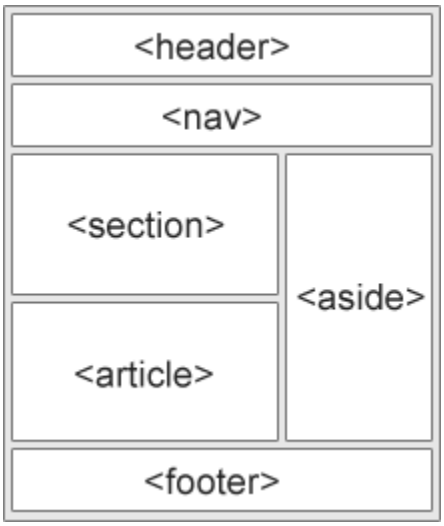
However, omitting the <head> tag has been a common practice for quite some time now.

HTML Layouts:

* HTML Layout Elements

Websites often display content in multiple columns (like a magazine or newspaper).

HTML5 offers new semantic elements that define the different parts of a web page:



- <header> - Defines a header for a document or a section
- <nav> - Defines a container for navigation links
- <section> - Defines a section in a document
- <article> - Defines an independent self-contained article
- <aside> - Defines content aside from the content (like a

sidebar)
- <footer> - Defines a footer for a document or a section
- <details> - Defines additional details
- <summary> - Defines a heading for the <details> element

*HTML Layout Techniques

There are four different ways to create multicolumn layouts. Each way has its pros and cons:

- HTML tables
- CSS float property
- CSS framework
- CSS flexbox

*Which One to Choose?

HTML Tables

The <table> element was not designed to be a layout tool! The purpose of the <table> element is to display tabular data. So, do not use tables for your page layout! They will bring a mess into your code. And imagine how hard it will be to redesign your site after a couple of months.

Tip: Do NOT use tables for your page layout!

CSS Frameworks

If you want to create your layout fast, you can use a framework, like W3.CSS or Bootstrap.

CSS Floats

It is common to do entire web layouts using the CSS float property. Float is easy to learn - you just need to remember how the float and clear properties work. Disadvantages: Floating elements are tied to the document flow, which may harm the flexibility. Learn more about float in our CSS Float and Clear chapter.

Float Example

City Gallery

- London
- Paris
- Tokyo

London

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.

Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.

Copyright © W3Schools.com

Try it Yourself »

CSS Flexbox

Flexbox is a new layout mode in CSS3.

Use of flexbox ensures that elements behave predictably when the page layout must accommodate different screen sizes and different display devices.
Disadvantages: Does not work in IE10 and earlier.

Learn more about flexbox in our CSS Flexbox chapter.

Flexbox Example

City Gallery

- London
- Paris
- Tokyo

London

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.

Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.

*CSS Frameworks

If you want to create your layout fast, you can use a framework, like W3.CSS or Bootstrap.

HTML Responsive Web Design:

*Responsive Web Design is about using CSS and HTML to resize, hide, shrink, enlarge, or move the content to make it look good on any screen:

*Using W3.CSS

Another way to create a responsive design, is to use a responsive style sheet, like W3.CSS

W3.CSS makes it easy to develop sites that look nice at any size; desktop, laptop, tablet, or phone:

HTML Computer Code Elements:

*HTML <kbd> For Keyboard Input

The HTML **<kbd>** element represents user input, like keyboard input or voice commands.

Text surrounded by <kbd> tags is typically displayed in the browser's default monospace font:

Example

```
<p>Save the document by pressing <kbd>Ctrl + S</kbd></p>
```

Result:

Save the document by pressing `Ctrl + S`

Try it Yourself »

HTML <samp> For Program Output

The HTML **<samp>** element represents output from a program or computing system.

Text surrounded by <samp> tags is typically displayed in the browser's default monospace font:

Example

```
<p>If you input wrong value, the program will return <samp>Error!</samp></p>
```

Result:

If you input wrong value, the program will return `Error!`

Try it Yourself »

*Example

```
<code>
x = 5;
y = 6;
z = x + y;
</code>
```

Result:

```
x = 5; y = 6; z = x + y;
```

Try it Yourself »

Notice that the <code> element does not preserve extra whitespace and line-breaks.

To fix this, you can put the <code> element inside a **<pre>** element:

Example

```
<pre>
<code>
x = 5;
```

```
y = 6;
z = x + y;
</code>
</pre>
```

Result:

```
x = 5;
y = 6;
z = x + y;
```

Try it Yourself »

HTML Entities:

*HTML Entities

Some characters are reserved in HTML.

If you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags.

Character entities are used to display reserved characters in HTML.

A character entity looks like this:

&*entity_name*;

OR

&#*entity_number*;

To display a less than sign (<) we must write: **<** or **<**

Advantage of using an entity name: An entity name is easy to remember.
Disadvantage of using an entity name: Browsers may not support all entity names, but the support for numbers is good.

*Some Other Useful HTML Character Entities

Result	Description	Entity Name	Entity Number
	non-breaking space		
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
"	double quotation mark	"	"
'	single quotation mark (apostrophe)	'	'
¢	cent	¢	¢
£	pound	£	£

¥	yen	¥	¥
€	euro	€	€
©	copyright	©	©
®	registered trademark	®	®

*Combining Diacritical Marks

Mark	Character	Construct	Result
`	a	à	à
´	a	á	á
^	a	â	â
~	a	ã	ã
`	O	Ò	Ò

´	O	Ó	Ó
ˆ	O	Ô	Ô
˜	O	Õ	Õ

HTML Symbols:

*Some Mathematical Symbols Supported by HTML

Char	Number	Entity	Description
∀	∀	∀	FOR ALL
∂	∂	∂	PARTIAL DIFFERENTIAL
∃	∃	∃	THERE EXISTS

∅	∅	∅	EMPTY SETS
∇	∇	∇	NABLA
∈	∈	∈	ELEMENT OF
∉	∉	∉	NOT AN ELEMENT OF
∋	∋	∋	CONTAINS AS MEMBER
∏	∏	∏	N-ARY PRODUCT
∑	∑	∑	N-ARY SUMMATION

Some Greek Letters Supported by HTML

Char	Number	Entity	Description
Α	Α	Α	GREEK CAPITAL LETTER ALPHA
Β	Β	Β	GREEK CAPITAL

			LETTER BETA
Γ	Γ	Γ	GREEK CAPITAL LETTER GAMMA
Δ	Δ	Δ	GREEK CAPITAL LETTER DELTA
E	Ε	Ε	GREEK CAPITAL LETTER EPSILON
Z	Ζ	Ζ	GREEK CAPITAL LETTER ZETA

Full Greek Reference

Some Other Entities Supported by HTML

Char	Number	Entity	Description
©	©	©	COPYRIGHT SIGN
®	®	®	REGISTERED SIGN

€	€	€	EURO SIGN
™	™	™	TRADEMARK
←	←	←	LEFTWARDS ARROW
↑	↑	↑	UPWARDS ARROW
→	→	→	RIGHTWARDS ARROW
↓	↓	↓	DOWNWARDS ARROW
♠	♠	♠	BLACK SPADE SUIT
♣	♣	♣	BLACK CLUB SUIT
♥	♥	♥	BLACK HEART SUIT
♦	♦	♦	BLACK DIAMOND SUIT

For more check the internet.

HTML Encoding (Character Sets):

*To display an HTML page correctly, a web browser must know which character set (character encoding) to use.

***UTF-8 (Unicode) covers almost all of the characters and symbols in the world.**

***ASCII was the first** character encoding standard **(also called character set). ASCII defined 128 different alphanumeric characters that could be used on the internet:**

*The HTML charset Attribute

To display an HTML page correctly, a web browser must know the character set used in the page.

This is specified in the <meta> tag:

For HTML4:

```
<meta http-equiv="Content-Type"content="text/html;charset=ISO-8859-1">
```

For HTML5:

```
<meta charset="UTF-8">
```

***If a browser detects ISO-8859-1 in a web page, it defaults to ANSI, because ANSI is identical to ISO-8859-1 except that ANSI has 32 extra characters.**

*Differences Between Character Sets

The following table displays the differences between the character sets described above:

Numb	ASCII	ANSI	8859	UTF-8	Description
32					space
33	!	!	!	!	exclamation mark
34	"	"	"	"	quotation mark
35	#	#	#	#	number sign
36	$	$	$	$	dollar sign
37	%	%	%	%	percent sign
38	&	&	&	&	ampersand
39	'	'	'	'	apostrophe
40	(	(	(	(	left parenthesis

41	)	)	)	)	right parenthesis
42	*	*	*	*	asterisk
43	+	+	+	+	plus sign
44	,	,	,	,	comma
45	-	-	-	-	hyphen-minus
46	.	.	.	.	full stop
47	/	/	/	/	solidus
48	0	0	0	0	digit zero
49	1	1	1	1	digit one
50	2	2	2	2	digit two
51	3	3	3	3	digit three

52	4	4	4	4	digit four
53	5	5	5	5	digit five
54	6	6	6	6	digit six
55	7	7	7	7	digit seven
56	8	8	8	8	digit eight
57	9	9	9	9	digit nine
58	:	:	:	:	colon
59	;	;	;	;	semicolon
60	<	<	<	<	less-than sign
61	=	=	=	=	equals sign
62	>	>	>	>	greater-than sign

63	?	?	?	?	question mark
64	@	@	@	@	commercial at
65	A	A	A	A	Latin capital letter A
66	B	B	B	B	Latin capital letter B
67	C	C	C	C	Latin capital letter C
68	D	D	D	D	Latin capital letter D
69	E	E	E	E	Latin capital letter E
70	F	F	F	F	Latin capital letter F
71	G	G	G	G	Latin capital letter G

72	H	H	H	H	Latin capital letter H
73	I	I	I	I	Latin capital letter I
74	J	J	J	J	Latin capital letter J
75	K	K	K	K	Latin capital letter K
76	L	L	L	L	Latin capital letter L
77	M	M	M	M	Latin capital letter M
78	N	N	N	N	Latin capital letter N
79	O	O	O	O	Latin capital letter O
80	P	P	P	P	Latin capital

					letter P
81	Q	Q	Q	Q	Latin capital letter Q
82	R	R	R	R	Latin capital letter R
83	S	S	S	S	Latin capital letter S
84	T	T	T	T	Latin capital letter T
85	U	U	U	U	Latin capital letter U
86	V	V	V	V	Latin capital letter V
87	W	W	W	W	Latin capital letter W
88	X	X	X	X	Latin capital letter X

89	Y	Y	Y	Y	Latin capital letter Y
90	Z	Z	Z	Z	Latin capital letter Z
91	[	[	[	[	left square bracket
92	\	\	\	\	reverse solidus
93	]	]	]	]	right square bracket
94	^	^	^	^	circumflex accent
95	_	_	_	_	low line
96	`	`	`	`	grave accent
97	a	a	a	a	Latin small letter a

98	b	b	b	b	Latin small letter b
99	c	c	c	c	Latin small letter c
100	d	d	d	d	Latin small letter d
101	e	e	e	e	Latin small letter e
102	f	f	f	f	Latin small letter f
103	g	g	g	g	Latin small letter g
104	h	h	h	h	Latin small letter h
105	i	i	i	i	Latin small letter i
106	j	j	j	j	Latin small letter j

107	k	k	k	k	Latin small letter k
108	l	l	l	l	Latin small letter l
109	m	m	m	m	Latin small letter m
110	n	n	n	n	Latin small letter n
111	o	o	o	o	Latin small letter o
112	p	p	p	p	Latin small letter p
113	q	q	q	q	Latin small letter q
114	r	r	r	r	Latin small letter r
115	s	s	s	s	Latin small letter s

116	t	t	t	t	Latin small letter t
117	u	u	u	u	Latin small letter u
118	v	v	v	v	Latin small letter v
119	w	w	w	w	Latin small letter w
120	x	x	x	x	Latin small letter x
121	y	y	y	y	Latin small letter y
122	z	z	z	z	Latin small letter z
123	{	{	{	{	left curly bracket
124	|	|	|	|	vertical line

125	}	}	}	}	right curly bracket
126	~	~	~	~	tilde
127	DEL				
128		€			euro sign
129		•	•	•	NOT USED
130		‚			single low-9 quotation mark
131		ƒ			Latin small letter f with hook
132		„			double low-9 quotation mark
133		…			horizontal ellipsis
134		†			dagger

135	‡			double dagger
136	ˆ			modifier letter circumflex accent
137	‰			per mille sign
138	Š			Latin capital letter S with caron
139	‹			single left-pointing angle quotation mark
140	Œ			Latin capital ligature OE
141	•	•	•	NOT USED
142	Ž			Latin capital letter Z with caron
143	•	•	•	NOT USED

144	•	•	•	NOT USED
145	‘			left single quotation mark
146	’			right single quotation mark
147	“			left double quotation mark
148	”			right double quotation mark
149	•			bullet
150	–			en dash
151	—			em dash
152	˜			small tilde
153	™			trade mark sign

154	š			Latin small letter s with caron
155	›			single right-pointing angle quotation mark
156	œ			Latin small ligature oe
157	•	•	•	NOT USED
158	ž			Latin small letter z with caron
159	Ÿ			Latin capital letter Y with diaeresis
160				no-break space
161	¡	¡	¡	inverted exclamation mark
162	¢	¢	¢	cent sign

163		£	£	£	pound sign
164		¤	¤	¤	currency sign
165		¥	¥	¥	yen sign
166		¦	¦	¦	broken bar
167		§	§	§	section sign
168		¨	¨	¨	diaeresis
169		©	©	©	copyright sign
170		ª	ª	ª	feminine ordinal indicator
171		«	«	«	left-pointing double angle quotation mark
172		¬	¬	¬	not sign

173				soft hyphen
174	®	®	®	registered sign
175	¯	¯	¯	macron
176	°	°	°	degree sign
177	±	±	±	plus-minus sign
178	²	²	²	superscript two
179	³	³	³	superscript three
180	´	´	´	acute accent
181	µ	µ	µ	micro sign
182	¶	¶	¶	pilcrow sign
183	·	·	·	middle dot

184	¸	¸	¸	cedilla
185	¹	¹	¹	superscript one
186	º	º	º	masculine ordinal indicator
187	»	»	»	right-pointing double angle quotation mark
188	¼	¼	¼	vulgar fraction one quarter
189	½	½	½	vulgar fraction one half
190	¾	¾	¾	vulgar fraction three quarters
191	¿	¿	¿	inverted question mark
192	À	À	À	Latin capital letter A with

				grave
193		Á	Á	Á	Latin capital letter A with acute
194		Â	Â	Â	Latin capital letter A with circumflex
195		Ã	Ã	Ã	Latin capital letter A with tilde
196		Ä	Ä	Ä	Latin capital letter A with diaeresis
197		Å	Å	Å	Latin capital letter A with ring above
198		Æ	Æ	Æ	Latin capital letter AE
199		Ç	Ç	Ç	Latin capital letter C with cedilla

200	È	È	È	Latin capital letter E with grave
201	É	É	É	Latin capital letter E with acute
202	Ê	Ê	Ê	Latin capital letter E with circumflex
203	Ë	Ë	Ë	Latin capital letter E with diaeresis
204	Ì	Ì	Ì	Latin capital letter I with grave
205	Í	Í	Í	Latin capital letter I with acute
206	Î	Î	Î	Latin capital letter I with circumflex
207	Ï	Ï	Ï	Latin capital letter I with

				diaeresis
208		Ð	Ð	Ð	Latin capital letter Eth
209		Ñ	Ñ	Ñ	Latin capital letter N with tilde
210		Ò	Ò	Ò	Latin capital letter O with grave
211		Ó	Ó	Ó	Latin capital letter O with acute
212		Ô	Ô	Ô	Latin capital letter O with circumflex
213		Õ	Õ	Õ	Latin capital letter O with tilde
214		Ö	Ö	Ö	Latin capital letter O with diaeresis

215	×	×	×	multiplication sign
216	Ø	Ø	Ø	Latin capital letter O with stroke
217	Ù	Ù	Ù	Latin capital letter U with grave
218	Ú	Ú	Ú	Latin capital letter U with acute
219	Û	Û	Û	Latin capital letter U with circumflex
220	Ü	Ü	Ü	Latin capital letter U with diaeresis
221	Ý	Ý	Ý	Latin capital letter Y with acute

222	Þ	Þ	Þ	Latin capital letter Thorn
223	ß	ß	ß	Latin small letter sharp s
224	à	à	à	Latin small letter a with grave
225	á	á	á	Latin small letter a with acute
226	â	â	â	Latin small letter a with circumflex
227	ã	ã	ã	Latin small letter a with tilde
228	ä	ä	ä	Latin small letter a with diaeresis
229	å	å	å	Latin small letter a with ring above
230	æ	æ	æ	Latin small letter

				ae
231		ç	ç	ç	Latin small letter c with cedilla
232		è	è	è	Latin small letter e with grave
233		é	é	é	Latin small letter e with acute
234		ê	ê	ê	Latin small letter e with circumflex
235		ë	ë	ë	Latin small letter e with diaeresis
236		ì	ì	ì	Latin small letter i with grave
237		í	í	í	Latin small letter i with acute
238		î	î	î	Latin small letter i with circumflex

239	ï	ï	ï	Latin small letter i with diaeresis
240	ð	ð	ð	Latin small letter eth
241	ñ	ñ	ñ	Latin small letter n with tilde
242	ò	ò	ò	Latin small letter o with grave
243	ó	ó	ó	Latin small letter o with acute
244	ô	ô	ô	Latin small letter o with circumflex
245	õ	õ	õ	Latin small letter o with tilde
246	ö	ö	ö	Latin small letter o with diaeresis
247	÷	÷	÷	division sign

248	ø	ø	ø	Latin small letter o with stroke
249	ù	ù	ù	Latin small letter u with grave
250	ú	ú	ú	Latin small letter u with acute
251	û	û	û	Latin small letter with circumflex
252	ü	ü	ü	Latin small letter u with diaeresis
253	ý	ý	ý	Latin small letter y with acute
254	þ	þ	þ	Latin small letter thorn
255	ÿ	ÿ	ÿ	Latin small letter y with diaeresis

HTML Uniform Resource Locators:

*URL - Uniform Resource Locator

*****

```
scheme://prefix.domain:port/path/filename
```

Explanation:

- **scheme** - defines the **type** of Internet service (most common is **http or https**)
- **prefix** - defines a domain **prefix** (default for http is **www**)
- **domain** - defines the Internet **domain name** (like w3schools.com)
- **port** - defines the **port number** at the host (default for http is **80**)
- **path** - defines a **path** at the server (If omitted: the root directory of the site)
- **filename** - defines the name of a document or resource

*Common URL Schemes

The table below lists some common schemes:

Scheme	Short for	Used for
http	HyperText Transfer Protocol	Common web pages. Not encrypted
https	Secure HyperText Transfer Protocol	Secure web pages.

		Encrypted
ftp	File Transfer Protocol	Downloading or uploading files
file		A file on your computer

*ASCII Encoding Examples

Your browser will encode input, according to the character-set used in your page.

The default character-set in HTML5 is UTF-8.

Character	**From Windows-1252**	**From UTF-8**
€	%80	%E2%82%AC
£	%A3	%C2%A3
©	%A9	%C2%A9

®	%AE	%C2%AE
À	%C0	%C3%80
Á	%C1	%C3%81
Â	%C2	%C3%82
Ã	%C3	%C3%83
Ä	%C4	%C3%84
Å	%C5	%C3%85

- # HTML and XHTML:

*XHTML stands for E**X**tensible **H**yper**T**ext **M**arkup **L**anguage

- XHTML is almost identical to HTML
- XHTML is stricter than HTML
- XHTML is HTML defined as an XML application

- XHTML is supported by all major browsers

*Why XHTML?

Many pages on the internet contain "bad" HTML.

This HTML code works fine in most browsers (even if it does not follow the HTML rules):

* Smaller devices often lack the resources or power to interpret "bad" markup.

*XML is a markup language where documents must be marked up correctly (be "well-formed").

*By combining the strengths of HTML and XML, XHTML was developed.

XHTML is HTML redesigned as XML.

*The Most Important Differences from HTML:

Document Structure

- XHTML DOCTYPE is **mandatory**
- The xmlns attribute in <html> is **mandatory**
- <html>, <head>, <title>, and <body> are **mandatory**

XHTML Elements

- XHTML elements must be **properly nested**
- XHTML elements must always be **closed**
- XHTML elements must be in **lowercase**
- XHTML documents must have **one root element**

XHTML Attributes

- Attribute names must be in **lower case**
- Attribute values must be **quoted**
- Attribute minimization is **forbidden**

*In HTML, some elements can be improperly nested within each other, like this:

```
<b><i>This text is bold and italic</b></i>
```

In XHTML, all elements must be properly nested within each other, like this:

```
<b><i>This text is bold and italic</i></b>
```

*XHTML Elements Must Always Be Closed

This is wrong:

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

This is correct:

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

Empty Elements Must Also Be Closed

This is wrong:

```
A break: <br>
A horizontal rule: <hr>
An image: <img src="happy.gif" alt="Happy face">
```

This is correct:

```
A break: <br />
A horizontal rule: <hr />
An image: <img src="happy.gif" alt="Happy face" />
```

*

XHTML Elements Must Be In Lower Case

*Attribute Minimization Is Forbidden

```
<input type="checkbox" name="vehicle"value="car
" checked="checked" />
```

*

How to Convert from HTML to XHTML

1. Add an XHTML <!DOCTYPE> to the first line of every page
2. Add an xmlns attribute to the html element of every page
3. Change all element names to lowercase
4. Close all empty elements
5. Change all attribute names to lowercase
6. Quote all attribute values

Reference:

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