

**Dr. SNS RAJALAKSHMI COLLEGE OF ARTS AND SCIENCE**

**(Autonomous)**

**Coimbatore -641049**

**Accredited by NAAC (Cycle- III) with 'A+' Grade.**

**COMPUTER APPLICATIONS**

**WEB DESIGNING**

**(16UCA301)**

**UNIT-I**

**Class: III BCA**

**Regulation: 2020-2023**

**Academic year: 2022-2023**

**Subject In charge: Ms.S.Lalitha**

## UNIT-I:

### WEB DESIGN:

Web design is the creation of websites and pages to reflect a company's brand and information and ensure a user-friendly experience. Appearance and design are incorporated as vital elements whether you're designing a website, mobile app or maintaining content on a web page.

#### A Simple HTML Document

##### Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page                               Title</title>
</head>
<body>
<h1>My                                   First           Heading</h1>
<p>My                                   first           paragraph.</p>
</body>
</html>
```

- The `<!DOCTYPE html>` declaration defines that this document is an HTML5 document
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the HTML page
- The `<title>` element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- The `<body>` element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

#### HTML core Element and Attribute:

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>
 	<i>none</i>	<i>none</i>

## 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>
 	<i>none</i>	<i>none</i>

**Note:** Some HTML elements have no content (like the `<br>` element). These elements are called empty elements. Empty elements do not have an end tag!

## 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>`

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://www.w3schools.com">Visit W3Schools</a>
```

You will learn more about links in our [HTML Links chapter](#).

---

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

```

```

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.w3schools.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

```

```

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

```

```

```

```

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<sub>2</sub>O:

### 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<sup>[1]</sup>:

### Example

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

## Presentational Elements

There are a handful of (X)HTML elements that are explicitly presentation oriented. Sometimes called “physical” styles, they provide instructions for the size, weight, or style of the font used to display the element.

If you’ve been paying attention, you already know that Cascading Style Sheets are now the preferred way to specify presentation instructions. [Table 10-2](#) lists the presentational inline elements, along with the preferred alternative for achieving the same visual effect.

*Table 10-2. Presentational inline elements and style sheet alternatives*



Element	Description	Alternative
<b>B</b>	Bold	Use the strong element instead if appropriate, or use the font-weight: bold
<b>Big</b>	Big	Use a relative font-size keyword to make text display slightly font-size: bigger
<b>I</b>	Italic	Use the em element instead if appropriate, or use the font-style: italic
<del>s</del> ( <i>deprecated</i> )	Strike-through	Use the text-decoration property to make text display with a line through text-decoration: line-through
<b>Small</b>	Small	Use a relative font-size keyword to make text display slightly font-size: smaller
<del>strike</del> ( <i>deprecated</i> )	Strike-through	Use the text-decoration property to make text display with a line through text-decoration: line-through
<b>Tt</b>	Teletype	Use the font-family property to select a specific or generic fixed-width font font-family: "Andale Mono", monospace

#### HTML phrase tags Elements:

HTML phrase tags are unique purpose tags designed to use indefinite cases, even though they're implemented in the same manner as other tags are used, which you might have seen in previous chapters. So now, let's dig deep into each of its different types of phrase tags. You might have briefly read about these in the earlier chapters, but not as a subcategory of phrase tags.

#### HTML phrase tags list:

- [Emphasis Tag](#)
- [Marked Tag](#)
- [Strong Tag](#)
- [Abbreviation Tag](#)
- [Acronym Tag](#)
- [Special Terminology / Definition Tag](#)
- [Quotation Tag](#)
- [Short Quote Tag](#)
- [Code Tag](#)
- [Keyboard Tag](#)
- [Address Tag](#)

Let's see what each of these does:

### Emphasis Tag

Emphasis Tag emphasizes or stresses the texts within its opening and closing tags.

Example:

```
This is a <em>beautiful</em> language.
```

Output:

This is a *beautiful* language.

### Marked Tag

The mark tag defines text with a highlighted yellow background that should be marked or highlighted.

Example:

```
Mark these <mark>words</mark>.
```

Output:

Mark these words.

### Strong Tag

The mark tag defines the appearance of the text as solid or bold, which should be more significant.

Example:

```
These examples are <strong>very important</strong>.
```

Output:

These examples are **very important**.

### Abbreviation Tag

Abbreviation Tag is used to give an abbreviation to any word written within the opening and closing tag.

Example:

```
Do you know the nickname of <abbr title = "Rowan Atkinson">Mr. Bean</abbr>?
```

Output:

Do you know the nickname of Mr. Bean?

### Acronym Tag

Acronym Tag is used to show that the text within the opening and closing tag is an acronym. This tag is Not Supported in HTML5.

Example:

```
You are studying <acronym>HTML</acronym>.
```

Output:

You are studying HTML.

### Special Terminology / Definition Tag

Special Terminology/Definition Tag is used when HTML developers introduce a special term and display it differently.

Example:

```
This can be defined as <dfn>scripting</dfn>.
```

Output:

This can be defined as *scripting*.

### Quotation Tag

Quoting Tag is used when HTML developers want to add a quote to a passage from a different source.

Example:

```
<blockquote>This is all about India, which is a developing country having the second largest population.</blockquote>
```

Output:

This is all about India, which is a developing country having the second largest population.

You can visit the [HTML Blockquote Tag tutorial](#) for more information on the Quotation tag.

### Short Quote Tag

Short Quote Tag is used for adding a double quote to your statement on your web page.

Example:

```
This can be defined as <q>scripting</q>.
```

Output:

This can be defined as scripting.

### Code Tag

Code Tag is used for making text visualization as mono-spaced font, just like most of the programming books display code text.

Example:

```
This is an example of code <code><?php echo "Hello World!" ?></code>
```

Output:

This is an example of code <?php echo "Hello World!" ?>

---

### Keyboard Tag

The HTML <kbd> tag defines text as input from a keyboard such as the Ctrl key, Enter, etc. Browsers render text within <kbd> tags in monospace font.

Example:

```
<p>This is an example of <kbd>scripting</kbd>.</p>  
<p>Press <kbd>ctrl+c</kbd> to copy.</p>
```

Output:

This is an example of scripting. Press ctrl+c to copy.

---

### Address Tag

The address tag is used to specify and emphasize that a set of text is an address.

Example:

```
He is Mr. Bean.  
  
He lives in:  
<address>ADDRESS LINE 1<br>  
ADDRESS LINE 2, CITY, STATE CODE, PIN</address>
```

---

### HTML Lists

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

Example

An unordered HTML list:

- Item
- 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>
```

---

## ADVERTISEMENT

---

### 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>
```

#### Output:

Coffee

- black hot drink

Milk

- white cold drink

### HTML text Editors

- An HTML file is a text file, so to create an HTML file we can use any text editors.
- Text editors are the programs which allow editing in a written text, hence to create a web page we need to write our code in some text editor.
- There are various types of text editors available which you can directly download, but for a beginner, the best text editor is Notepad (Windows) or TextEdit (Mac).
- After learning the basics, you can easily use other professional text editors which are, **Notepad++**, **Sublime Text**, **Vim**, etc.
- In our tutorial, we will use Notepad and sublime text editor. Following are some easy ways to create your first web page with Notepad, and sublime text.

#### A. HTML code with Notepad. (Recommended for Beginners)

Notepad is a simple text editor and suitable for beginners to learn HTML. It is available in all versions of Windows, from where you easily access it.

## character entities in HTML

HTML character entities are basically a set of characters (entity) used to represent few characters reserved by the HTML, especially invisible characters or characters difficult to type out using a regular keyboard. HTML provides some entity names and entity numbers to use these symbols.

**Example:** ‘<’ is already reserved in HTML language. Sometimes this character needs to display on the web page that creates ambiguity in code. Along with these are the character which is normally not present in basic keyboard ( £, ¥, €, © ), etc. We will explore the HTML entities in detail as we grow in the article.

### Syntax:

&entity\_name; or &#entity\_number;

Some useful symbols with their entity name and entity numbers are discussed below:

<i>Symbol</i>	<i>Entity Name</i>	<i>Entity Number</i>	<i>Details</i>
&	&nbsp;	&#160;	non-breaking space
<	&lt;	&#60;	less than
>	&gt;	&#62;	greater than
®	&reg;	&#174;	registered trademark
©	&copy;	&#169;	Copyright
₹		&#8377;	India rupee sign
€	&euro;	&#8364;	Euro
¥	&yen;	&#165;	Yen..
\$	&dollar;	&#36;	Dollar
£		&#163;	Pound
¢	&cent;	&#162;	Cent
‘	&apos;	&#39;	single quotation mark (apostrophe)
“	&quot;	&#34;	double quotation mark

*Symbol   Entity Name   Entity Number   Details*

&   &amp;   &#38;   Ampersand

**Reserved Characters:** Reserved characters are those characters that are either reserved for HTML or those which are not present in the basic keyboard.

**Example:** As Entity numbers are easy to learn. Here, a few of the reserved characters are discussed:

Reserved Character   Entity Name

&   &amp;

<   &lt;

>   &gt;

“   &quot;

**Other Characters:** Some characters that aren't available directly on our keyboard, therefore such an entity can be used to represent it by using their entity number.

**Example:** These keywords are directly not available to type, we need to specify their entity number. Here #x before the numbers represents the hexadecimal representation.

Character   Entities

≥   &#8805; &#x2265; & GreaterEqual;

©   &copy; &#169;

€   &euro; &#8364;

∃   &#8707; &#x2203; &exist;

**Example:**

- HTML

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Character entities</title>
</head>
<body>
```



```
<h3>Indian rupee sign = ₹</h3>
<h3>Euro sign = €</h3>
<h3>Dollar sign = $</h3>
<h3>Pound sign = £</h3>
</body>
</html>
```

**Output:**

**Indian rupee sign = ₹**

**Euro sign = €**

**Dollar sign = \$**

**Pound sign = £**

An HTML comment:

Definition and Usage

The comment tag is used to insert comments in the source code. Comments are not displayed in the browsers.

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

An HTML comment:

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

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

Output: This is a paragraph

**HTML <font> Tag**

The **<font> tag** in HTML plays an important role in the web page to create an attractive and readable web page. The font tag is used to change the color, size, and style of a text. The base font tag is used to set all the text to the same size, color and face.

**Syntax:**

```
<font attribute = "value"> Content </font>
```

**Example:** In this example, we have used the <font> tag with a font size as 5.

- HTML

```
<!DOCTYPE html>
<html>

<body>
<h2>GeeksforGeeks</h2>

<!--Normal paragraph tag-->

<p>Hello Geeks!.</p>

<!--font tag-->
<font size="5"> Welcome to GeeksforGeeks </font>
</body>

</html>
```

**Output:**

The font tag has basically three attributes which are given below:

- [Font Size attribute](#)
- [Face/Type attribute](#)
- [Color attribute](#)

**Note:** Font tag is not supported in HTML5.

We will discuss all these attributes & understand them through the examples.

**font Size:** This attribute is used to adjust the size of the text in the HTML document using a font tag with the size attribute. The range of size of the font in HTML is from 1 to 7 and the default size is 3.

**Syntax:**

```
<font size="number">
```

**Example:** This example uses the <font> tag where different font sizes are specified.

- HTML

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

```
<body>
<!--HTML font size tag starts here-->
<font size="1">GeeksforGeeks!</font><br />
<font size="2">GeeksforGeeks!</font><br />
<font size="3">GeeksforGeeks!</font><br />
<font size="4">GeeksforGeeks!</font><br />
<font size="5">GeeksforGeeks!</font><br />
<font size="6">GeeksforGeeks!</font><br />
<font size="7">GeeksforGeeks!</font>
<!--HTML font size tag ends here-->
</body>

</html>
```

### Output:

GeeksforGeeks!  
GeeksforGeeks!  
GeeksforGeeks!  
GeeksforGeeks!  
GeeksforGeeks!  
GeeksforGeeks!  
GeeksforGeeks!

### *font size attribute*

**Font Type:** Font type can be set by using face attribute with font tag in HTML document. But the fonts used by the user need to be installed in the system first.

### **Syntax:**

```
<font face="font_family">
```

**Example:** This example describes the <font> tag with different font type & font size.

- HTML

```
<!DOCTYPE html>
<html>

<body>
<!--HTML font face tag starts here-->
<font face="Times New Roman" size="6">
GeeksforGeeks!!
</font> <br />
```

```
<font face="Verdana" size="6">
GeeksforGeeks!!
</font><br />
<font face="Comic sans MS" size=" 6">
GeeksforGeeks!!
</font><br />
<font face="WildWest" size="6">
GeeksforGeeks!!
</font><br />
<font face="Bedrock" size="6">
GeeksforGeeks!!
</font><br />
<!--HTML font face tag ends here-->
</body>
</html>
```

**Output:**

GeeksforGeeks!!  
GeeksforGeeks!!  
*GeeksforGeeks!!*  
GeeksforGeeks!!  
GeeksforGeeks!!

*font type attribute*

**Font Color:** Font color is used to set the text color using a font tag with the color attribute in an HTML document. Color can be specified either with its name or with its hex code.

**Syntax:**

```
<font color="color_name|hex_number|rgb_number">
```

**Example:** This example describes the <font> tag with different font colors.

- HTML

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>

<!--HTML font color tag starts here-->

<font color="#009900">GeeksforGeeks</font><br />

<font color="green">GeeksforGeeks</font>

<!--HTML font color tag ends here-->

</body>

</html>
```

## Output:

GeeksforGeeks  
GeeksforGeeks

## HTML Block and Inline Elements



HTML element has a default display value, depending on what type of element it is.

There are two display values: block and inline.

### 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

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p style="border: 1px solid black">Hello World</p>
```

```
<div style="border: 1px solid black">Hello World</div>
```

<p>The P and the DIV elements are both block elements, and they will always start on a new line and take up the full width available (stretches out to the left and right as far as it can).</p>

```
</body>
```

```
</html>
```

<b>Output:</b> Hello World
----------------------------

Hello World

The P and the DIV elements are both block elements, and they will always start on a new line and take up the full width available (stretches out to the left and right as far as it can).

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.

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:

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>
```

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:

Example

```
<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>
```

- There are two display values: block and inline
- A block-level element always starts on a new line and takes up the full width available
- An inline element does not start on a new line and it only takes up as much width as necessary
- The <div> element is a block-level and is often used as a container for other HTML elements
- The <span> element is an inline container used to mark up a part of a text, or a part of a document

## HTML Tags

| Tag                          | Description                                   |
|------------------------------|-----------------------------------------------|
| <a href="#">&lt;div&gt;</a>  | Defines a section in a document (block-level) |
| <a href="#">&lt;span&gt;</a> | Defines a section in a document (inline)      |

### HTML <link> Tag

#### Example

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="styles.css">
</head>
<body>
<h1>Hello World!</h1>
<h2>I am formatted with a linked style sheet.</h2>
<p>Me too!</p>
</body>
</html>
```

#### Definition and Usage

The `<link>` tag defines the relationship between the current document and an external resource.

The `<link>` tag is most often used to link to external style sheets or to add a [favicon](#) to your website.

The `<link>` element is an empty element, it contains attributes only.

### HTML <nav>TAG

## Example

A set of navigation links:

```
<!DOCTYPE html>

<html>

<body>

<h1>The nav element</h1>

<p>The nav element defines a set of navigation links:</p>

<nav>

<a href="/html/">HTML</a> |

<a href="/css/">CSS</a> |

<a href="/js/">JavaScript</a> |

<a href="/python/">Python</a>

</nav>

</body>

</html>
```

## The nav element

The nav element defines a set of navigation links:

[HTML](#) | [CSS](#) | [JavaScript](#) | [Python](#)

### Definition and Usage

The `<nav>` tag defines a set of navigation links.

Notice that NOT all links of a document should be inside a `<nav>` element. The `<nav>` element is intended only for major blocks of navigation links.

Browsers, such as screen readers for disabled users, can use this element to determine whether to omit the initial rendering of this content.

## 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

```
<!DOCTYPE html>

<html>

<body>

<h2>Image as a Link</h2>

<p>The image below is a link. Try to click on it.</p>

<a href="default.asp"></a>

</body>

</html>
```

### OUTPUT: Image as a Link

The image below is a link. Try to click on it.



---

## 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):

### 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:

### 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

```
<!DOCTYPE html>
```

```
<html lang="en-US">
```

```
<body>
```

```
<h2>Link Titles</h2>
```

<p>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.</p>

```
<a href="https://www.w3schools.com/html/" title="Go to W3Schools HTML section">Visit our HTML Tutorial</a>
```

```
</body>
```

```
</html>
```

### **OUTPUT: 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.

## Directory Structure

### **Organizing Files and Folder Structure for Web Pages**

It is a good idea to organize your code by placing the pages for each different section of the site into a new folder. Folders are referred to as directories on websites.

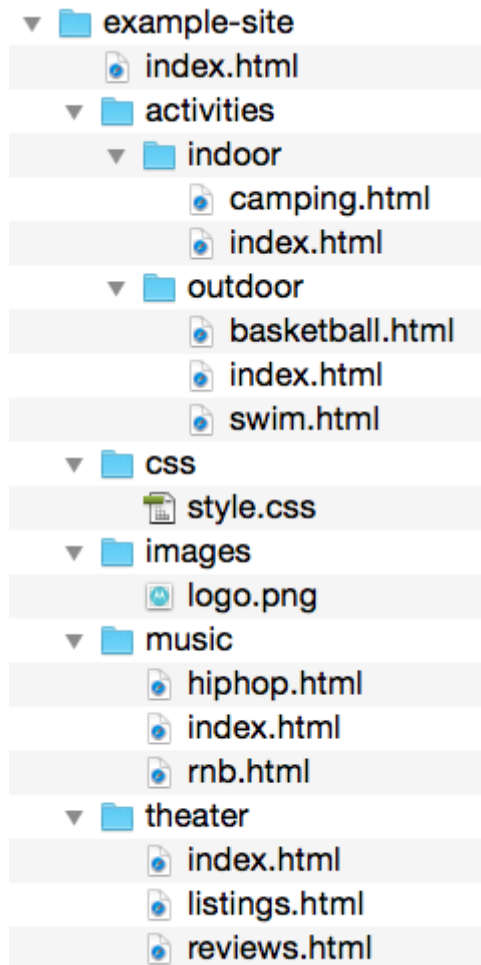


Diagram of the directory structure for a website called Example-Site.

## Structure

The top-level is known as the **root** folder. (In this example is know as the **root** folder called **examples-site**.) The root folder is contains all the other files and folders for a website.

## Relationships

The relationship between files and folders on a website is described using the same terminology as a family tree.

The **example-site** folder is a parent of the **activities**, **music** and **theater** folders. The **activities**, **music** and **theater** folders are children of the example-site folder.

## Homepages

The main homages of a site written in HTML is called *index.html* ( and the homepage of each section in a child folder).

Web servers are usually set up to return to the *index.html* file if no file name is specified. Therefore, if you enter example-site.com it will return example-site.com/index.html, and example-site.com/music will return example-site.com/music/index.html.

The root folder contains:

- A file called index.html which is the homepage for the entire site
- Individual folders for the activities, music and theater section of the site

Each sub-directory contains:

- A file called *index.html* which is the homepage for that section
- A review page called *reviews.html*
- A listing page called *listing.html6*

Share this:

- [Twitter](#)
- [Facebook](#)

## HTML Email Tag

HTML `<a>` tag provides you option to specify an email address to send an email. While using `<a>` tag as an email tag, you will use **mailto: email address** along with *href* attribute. Following is the syntax of using **mailto** instead of using http.

```
<a href = "mailto: abc@example.com">Send Email</a>
```

This code will generate the following link which you can use to send email.

[Send Email](#)

Now, if a user clicks this link, it launches one Email Client (like Lotus Notes, Outlook Express etc. ) installed on your user's computer. There is another risk to use this option to send email because if user do not have email client installed on their computer then it would not be possible to send email.

## Default Settings

You can specify a default *email subject* and *email body* along with your email address. Following is the example to use default subject and body.

```
<a href = "mailto:abc@example.com?subject = Feedback&body = Message">  
Send Feedback  
</a>
```

## Definition and Usage

The `<object>` tag defines a container for an external resource.

The external resource can be a web page, a picture, a media player, or a plug-in application.

To embed a picture, it is better to use the `<img>` tag.

To embed HTML, it is better to use the `<iframe>` tag.

To embed video or audio, it is better to use the `<video>` and `<audio>` tags.

---

```
<!DOCTYPE html>
<html>
<body>
,
<h1>The object element</h1>

<object data="pic_trulli.jpg" width="300" height="200"></object>

</body>
</html>
```

Output:

## The object element

### <object>: The External Object element

The <object> [HTML](#) element represents an external resource, which can be treated as an image, a nested browsing context, or a resource to be handled by a plugin.

#### [Try it](#)

#### [Attributes](#)

This element includes the [global attributes](#).

**archive** Deprecated

A space-separated list of URIs for archives of resources for the object.

**border** Deprecated

The width of a border around the control, in pixels.

**classid** Deprecated

The URI of the object's implementation. It can be used together with, or in place of, the **data** attribute.

**codebase** Deprecated



The base path used to resolve relative URIs specified by **classid**, **data**, or **archive**. If not specified, the default is the base URI of the current document.

### **codetype** Deprecated

The content type of the data specified by **classid**.

### **data**

The address of the resource as a valid URL. At least one of **data** and **type** must be defined.

### **declare** Deprecated

The presence of this Boolean attribute makes this element a declaration only. The object must be instantiated by a subsequent <object> element. Repeat the <object> element completely each time the resource is reused.

### **form**

The form element, if any, that the object element is associated with (its *form owner*). The value of the attribute must be an ID of a <form> element in the same document.

### **height**

The height of the displayed resource, in [CSS pixels](#). — (Absolute values only. [NO percentages](#))

### **name**

The name of valid browsing context (HTML5), or the name of the control (HTML 4).

### **standby** Deprecated

A message that the browser can show while loading the object's implementation and data.

### **type**

The [content type](#) of the resource specified by **data**. At least one of **data** and **type** must be defined.

### **usemap**

A hash-name reference to a <map> element; that is a '#' followed by the value of a [name](#) of a map element.

## width

The width of the display resource, in [CSS pixels](#). — (Absolute values only. [NO percentages](#))

## Examples

### Embed a YouTube Video

```
<object  
type="video/mp4"  
data="https://www.youtube.com/watch?v=Sp9ZfSvpf7A"  
width="1280"  
height="720"></object>
```

Copy to Clipboard

## 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:

```

```

```
<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:

```

```

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:



The coordinates `270,350` is located 270 pixels from the left margin and 350 pixels from the top:



Now we have enough data to create a clickable rectangular area:

Example

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

This is the area that becomes clickable and will send the user to the page "computer.htm":



---

Shape="circle"

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

337,300



Then specify the radius of the circle:

44 pixels



Now you have enough data to create a clickable circular area:

#### 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.

Like maybe a croissant shape!

How can we make the croissant in the image below become a clickable link?





We have to find the x and y coordinates for all edges of the croissant:



The coordinates come in pairs, one for the x-axis and one for the y-axis:

#### 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">
```

[Try it Yourself »](#)

This is the area that becomes clickable and will send the user to the page "croissant.htm":

