

HTML - Web Page Images and Rollovers

#img# puts an image on a web page. Image rollover effects and animated rollovers. #img# description, syntax, usage, attributes and examples. HTML-5.com is a great guide for web developers. [TV Series & Actors and Actresses](#). Follow [TV Series](#) and [HTML 5](#) on Google+.

[HTML-5.com](#) > [itemscopehttp://data-vocabulary.org/BreadcrumbHTML 5](#) > [itemscopehttp://data-vocabulary.org/BreadcrumbHTML Tags](#) > [itemscopehttp://data-vocabulary.org/BreadcrumbHTML - Web Page Images and Rollovers](#)

> [itemscopehttp://data-vocabulary.org/BreadcrumbExamples](#)

The Tag in HTML 5

The `` tag puts an image on a web page by reading the binary image data from a separate file or external resource and including it inline with other content on the page. It is one of the [embedded content tags in HTML](#). As `<dfn>embedded content</dfn>`, the image is read from the resource specified by the [src attribute](#) and dynamically inserted into the web page when it is rendered. For inline video content, see the [<iframe> tag](#), which works almost like an **img** element except the content in the external resource is HTML code rather than binary image data.

In the `` tag demo below, the image button on the left uses legacy image rollover techniques. The image button on the right uses more sophisticated animated rollover, but the animation may not be supported in all browsers yet.

```
</img> </img>
</img>
```

These are actual working examples of the [tag example code](#) below, since this entire page is [Valid HTML 5](#).

[back to top](#)

 Tag Syntax

Rules for coding HTML `img` elements

```
<a mode="pre" href="../body-tag/index.html#syntax"><body></a>
...
<a mode="pre" href="../html-tag-list.html#phrasing-content-model">... phrasing content
...
</body>
```

Rules for coding the HTML `img` element

Make sure you understand the [difference between a tag and element](#) and are familiar with the [definitions](#) of [namespace](#) and other [HTML terms](#).

1. Code the **img** element where [phrasing content](#) is expected.

Images and Rollovers

2. The `img` element consists of a standalone `` tag. The element name uses lower case letters and should be in the [HTML namespace](#), which it will pick up automatically from the [xmlns attribute](#) on the `<html>` tag.
3. Code a [src attribute](#) with the URL of the image resource.
4. Code an [alt attribute](#) with a description of the image for use by non-visual clients such as auditory browsers and as an alternative in case the image resource is unavailable for some reason.
5. Include any other [attributes of the tag](#) as appropriate.
6. Since the `img` element is a [void element](#), it should always be coded as a self-closing tag terminated with the delimiter string `>/>`.

 Content Model

Contents of the img element

Content: *Empty*. All properties are coded using [attributes](#).

Since the `` tag is a [void element](#), it is not allowed to have any content, even [HTML comments](#) and therefore should always be coded as a self-closing [standalone tag](#), ending with the delimiters `>/>` rather than just `>` (``).

[back to top](#)

 Tag Attributes

Attributes of the tag

global attributes	In addition to the local attributes of the <code></code> tag below, any of the common HTML attributes can also be coded.
<code>width</code> , <code>height</code>	
<code>src</code>	<p>a URI reference that resolves to the URL of a two-dimensional image resource</p> <p>Use percent escape codes as explained in the URL-encoding tutorial for any special characters in the URI reference.</p> <p>If the value of the <code>src</code> attribute resolves to an HTTP URL, it may contain any of the following components:</p> <ol style="list-style-type: none"> 1. protocol scheme, typically <code>http:</code> or else <code>https:</code> 2. username, followed by an "@" 3. host name or IP address 4. port number, which defaults to 80 for the <code>http</code> scheme and 443 for the <code>https</code> scheme 5. absolute or relative path 6. search query, indicated by "?"

Images and Rollovers

	If the protocol scheme, username, host name/IP address and port number are omitted the default is the current host - the same server as the base of the current document. If the path starts with a slash <code>/...</code> , it is an absolute path from the document root directory (AKA " <code><dfn>web root</dfn></code> ") on the server. A relative path will be resolved relative to the base of the current document.
<code>alt</code>	Provides alternative content when images are blocked or disabled or for users who are unable to view the image. See Difference between image alt text and image title below.
<code>ismap="ismap"</code>	Sets the value of the <code></code> boolean attribute to <code>true</code> . Omitting it sets to <code>false</code> .
<code>usemap</code>	a URI reference that resolves to the URL of an image map Use percent escape codes as explained in the URL-encoding tutorial for any special characters in the URI reference .

The image `<dfn>alt text</dfn>` in the [attribute](#) provides alternative content when images are blocked or disabled or for users who are unable to view the image.

The `<dfn>image title</dfn>` in the global [title attribute](#) provides a description of the image and may be displayed in a pop-up tool tip when the mouse is held over the image.

[back to top](#)

 Tag Examples

Examples of the tag in HTML 5

Simple `` code

```

```

The [src attribute](#) and [alt attribute](#) are both required.

Image link

```
<a href="http://www.ExampleOnly.com/todos/take-out-trash.html" title="Take
out the trash">
  
</a>
```

When the [a element](#) and `img` element represent the same resource or event, the [title attribute](#) should be put on the outermost element, which in this case is the [hypertext link](#)

Images and Rollovers

created by the [<a> tag](#). Since there is no [title attribute](#) on the [tag](#), the value of the title property of the [img](#) element will be inherited from the title property of the [a element](#).

Link with image and text

```
<a href="http://www.ExampleOnly.com/todos/take-out-trash.html" title="Take
out the trash">
 Take out
the trash</a>
```

In this case, the [a element](#) represents a different resource (a task in a "To Do" list) than the image element (the status of the task). Therefore the title property of the [img](#) element should *not* be inherited from the [<a> tag](#), which requires coding the [title attribute](#) on the [tag](#). The value of the title property is an empty string since the [attribute](#) provides the description of the image.

Clear after a floating image so headings do not flow around it

You can float an image with [style="float: left"](#) or [style="float: right"](#) then follow it with headings. If you try to position the headings below the image with [style="clear: both"](#), some browsers (Chrome, Safari) will move the headings below the image while other browsers (Firefox, Opera) will flow the headings around the image. For example:

```
style="float: left</b> ; border: 1px solid gray;
<a mode="pre" href=" ../p-tag/">p</a> >This is the text that flows to the right of the floating ima
<a mode="pre" href=" ../hgroup-tag/">hgroup</a> <b mode="pre">style="clear: both"</b> >
<a mode="pre" href=" ../heading-tags/">h3</a> >This Is A Heading That Should Appear After The In
<a mode="pre" href=" ../heading-tags/">h4</a> >It Is Followed By Another Heading That Is Just A
</hgroup>
```

View this in Firefox or Opera to see what happens:

```
</img>
```

This is the text that flows to the right of the floating image.

This Is A Heading That Should Appear After The Image

It Is Followed By Another Heading That Is Just A Little Longer

To make it look consistent in all browsers, code [display: block](#) in the [style attribute](#) to make sure the element with the [clear: both](#) style is rendered as a [block element](#). Now that the position of the headings is correct, code a [text-align: center](#) style to center the headings if desired.

```
style="float: left</b> ; border: 1px solid gray;
<a mode="pre" href=" ../p-tag/">p</a> >This is the text that flows to the right of the floating ima
<a mode="pre" href=" ../hgroup-tag/">hgroup</a> <b mode="pre">style="display: block; clear: both;
<a mode="pre" href=" ../heading-tags/">h3</a> >This Is A Heading That Should Appear After The In
<a mode="pre" href=" ../heading-tags/">h4</a> >It Is Followed By Another Heading That Is Just A
</hgroup>
```

```
</img>
```

This is the text that flows to the right of the floating image.

Images and Rollovers

~~This Is A Heading That Should Appear After The Image~~

It Is Followed By Another Heading That Is Just A Little Longer

Of course the styles can be put into a [CSS Style Sheet](#), possibly with a [class selector](#).

Simple image rollover

```

```

Animated image rollover

```
<div id="click-me-image-button">
<style scoped="scoped">
  div#click-me-image-button img {
    -webkit-transition: opacity 250ms ease-in-out;
    -moz-transition: opacity 250ms ease-in-out;
    -o-transition: opacity 250ms ease-in-out;
  }
  img#exlimg1, img#exlimg2:hover { opacity: 1.0; }
  img#exlimg2, img#exlimg1:hover { opacity: 0; }
</style>


</div>
```

Example of resizing image with link

```
<p style="text-align: center;">Click this image to view the &lt;video&gt; tag
demo<br/><br/>
  <a href="../video-tag/index.html#examples">
    
  </a>
</p>
```

This example puts an image in an <a> tag that links to the [HTML 5 <video> demo](#). Also shows how to resize an image; in this case the size of the image is changed from 320x240 to 160x120.

Click this image to view the <video> tag demo
[](#)

Image Fade-In Transition

```
</img>
</img>
```

Example of rotated image with drop shadow

```
<canvas id="ex3canvas" width="215" height="170">
  
</canvas>
```

The preferred size for the image is coded in both the tag attributes and the `drawImage` function and the size of the canvas is slightly larger to allow for the extra space used for rotation and the drop shadow. In HTML 5 browsers, the `width` and `height` attributes on the tag will be ignored. Browsers that do not support the [HTML 5 <canvas> tag](#), should fall back to the content of the <canvas> tag, which is simply a standard tag. If the `width` and `height` attributes were omitted, the image would only be resized in browsers drawing the image on the canvas with the `drawImage` function.

Here is an actual working example of the code above. If the edges are jagged or the drop shadow is missing then compare how it looks in other browsers. (Do *View Source* to verify that this page is using the [HTML 5 DOCTYPE](#). You can also verify it is [Valid HTML 5](#) using the [HTML Validator](#). Try using it to validate URLs with HTML examples from other places that claim to be HTML 5 web sites!)

```
<canvas id="ex3canvas" width="215" height="170"> </img> </canvas>
```

[back to top](#)

Changes in HTML 5 - Tag

What's new in HTML 5

Differences between HTML 5 and earlier versions of HTML

The following attributes should *not* be coded on the tag because they either have been deprecated or were never officially supported:

- `align`
- `border`
- `controls`
- `dynsrc`
- `hspace`
- `loop`
- `start`
- `vspace`

In [poyglot HTML documents](#) the tag is coded as a self-closing tag the same as in the [2000-2010 Recommendations from the W3C HTML Working Group](#). Those specifications changed the ending delimiter to `/>` from `>` in older recommendations. They also defined the [HTML namespace](#) for the `img` element type name and the names of other [HTML element types](#).

THE END