

HTML5

Presentation by Uplatz

Contact Us: <https://training.uplatz.com/>

Email: info@uplatz.com

Phone: +44 7836 212635

Table Of Contents:

- Iframes
- Content Languages
- SVG
- Canvas
- Meta Information
- Marking up computer code
- Marking-up Quotes

Iframes:

Attribute

Details

- name** Sets the element's name, to be used with an a tag to change the iframe's src.
- width** Sets the element's width in pixels.
- height** Sets the element's height in pixels.
- src** Specifies the page that will be displayed in the frame.
- srcdoc** Specifies the content that will be displayed in the frame, assuming the browser supports it. The content must be valid HTML.
- Sandbox** When set, the contents of the iframe is treated as being from a unique origin and features including scripts, plugins, forms and popups will be disabled.

Restrictions can be selectively relaxed by adding a space separated list of values.

allowfullscreen Whether to allow the iframe's contents to use requestFullscreen()

Basics of an Inline Frame:

- The term "IFrame" means Inline Frame.
- It can be used to include another page in your page.
- This will yield a small frame which shows the exact contents of the base.html.

```
<iframe src="base.html"></iframe>
```

Sandboxing

The following embeds an untrusted web page with all restrictions enabled

```
<iframe sandbox src="http://example.com/"></iframe>
```

- To allow the page to run scripts and submit forms, add allow-scripts and allow-forms to the sandbox attribute.

```
<iframe sandbox="allow-scripts allow-forms"  
src="http://example.com/"></iframe>
```

- If there is untrusted content (such as user comments) on the same domain as the parent web page, an iframe can be used to disable scripts while still allowing the parent document to interact with it's content using JavaScript.

```
<iframe sandbox="allow-same-origin allow-top-  
navigation"  
src="http://example.com/untrusted/comments/pa  
ge2">
```

- The parent document can add event listeners and resize the IFrame to fit its contents.

- This, along with allow-topnavigation, can make the sandboxed iframe appear to be part of parent document

Setting the Frame Size:

- The IFrame can be resized using the width and height attributes, where the values are represented in pixels (HTML 4.01 allowed percentage values, but HTML 5 only allows values in CSS pixels).

```
<iframe src="base.html" width="800"  
height="600"></iframe>
```

Using the "srcdoc" Attribute:

- The srcdoc attribute can be used (instead of the src attribute) to specify the exact contents of the iframe as a whole HTML document.
- This will yield an IFrame with the text "IFrames are cool!"

<iframe srcdoc="<p>IFrames are cool!</p>"></iframe>

- If the srcdoc attribute isn't supported by the browser, the IFrame will instead fall back to using the src attribute, but if both the src and srcdoc attributes are present and supported by the browser, srcdoc takes precedence.

<iframe srcdoc="<p>Iframes are cool!</p>" src="base.html"></iframe>

Using Anchors with Iframes:

- Normally a change of webpage within an IFrame is initiated from within the IFrame, for example, clicking a link inside the IFrame.
- However, it is possible to change an IFrame's content from outside the IFrame.

- You can use an anchor tag whose href attribute is set to the desired URL and whose target attribute is set to the iframe's name attribute.

Content Languages:

Base Document Language:

- It's a good practice to declare the primary language of the document in the html element:

<html lang="en">

- If no other lang attribute is specified in the document, it means that everything (i.e., element content and attribute text values) is in that language.
- If the document contains parts in other languages, these parts should get their own lang attributes to "overwrite" the language declaration

Element Language:

- The lang attribute is used to specify the language of element content and attribute text values:

<p lang="en">The content of this element is in English.</p>

<p lang="en" title="The value of this attribute is also in English.">The content of this element is in English.</p>

- The language declaration gets inherited:

<div lang="en">

<p>This element contains English content.</p>

<p title="This attribute, too.">Same with this element.</p>

</div>

Elements with Multiple Languages:

You can "overwrite" a language declaration:

<p lang="en">This English sentence contains the German word Hallo.</p>

Handling Attributes with Different Languages:

- You can "overwrite" a parent element's language declaration by introducing any element apart from applet, base, basefont, br, frame, frameset, hr, iframe, meta, param, script (of HTML 4.0) with an own lang attribute:

<p lang="en" title="An English paragraph">

Hallo Welt!

</p>

SVG:

- SVG stands for Scalable Vector Graphics.
- SVG is used to define graphics for the Web The HTML `<svg>` element is a container for SVG graphics.
- SVG has several methods for drawing paths, boxes, circles, text, and graphic images.

Inline SVG:

- SVG can be written directly into a HTML document.
- Inline SVG can be styled and manipulated using CSS and JavaScript.

`<body>`

```
<svg class="attention"
xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink"
viewBox="0 0 1000 1000" >
```

```
<path id="attention"
d="m571,767l0,-106q0,-8,-5,-13t-12,-5l-108,0q-7,0,-
12,5t-5,13l0,106q0,8,5,13t12,6l108,0q7,0,12,-6t5
,-13Zm-1,-208l10,-257q0,-6,-5,-10q-7,-6,-14,-6l-122,0q-
7,0,-14,6q-5,4,-5,12l9,255q0,5,6,9t13,3l103,
0q8,0,13,-3t6,-9Zm-7,-522l428,786q20,35,-1,70q-10,17,-
26,26t-35,10l-858,0q-18,0,-35,-10t-26,-26q-21
,-35,-1,-70l429,-786q9,-17,26,-27t36,-10t36,10t27,27Z" />
</svg>
</body>
```

The above inline SVG can then be styled using the corresponding CSS class:

```
.attention {  
  fill: red;  
  width: 50px;  
  height: 50px;  
}
```

Embedding external SVG files in HTML:

- You can use the `` or `<object>` elements to embed external SVG elements.
- Setting the height and width is optional but is highly recommended.

Using the image element

```

```

- Using `` does not allow you to style the SVG using CSS or manipulate it using JavaScript.

Using the object element

<object type="image/svg+xml" data="attention.svg" width="50" height="50">

- Unlike , <object> directly imports the SVG into the document and therefore it can be manipulated using Javascript and CSS.

Embedding SVG using CSS:

- You can add external SVG files using the background-image property, just as you would do with any other image.

HTML:

<div class="attention"></div>

CSS:

.attention {

background-image: url(attention.svg);

background-size: 100% 100%;

```
width: 50px;  
height: 50px;  
}
```

- You can also embed the image directly into a css file using a data url:

background-image:

Canvas:

Attribute	Description
------------------	--------------------

height	Specifies the canvas height
---------------	-----------------------------

width	Specifies the canvas width
--------------	----------------------------

- Basic Example The canvas element was introduced in HTML5 for drawing graphics.
- Cannot display graphic.
- Canvas is not supported by your browser (IE The above will create a transparent HTML element of 300×150 px in size.

- You can use the canvas element to draw amazing stuff like shapes, graphs, manipulate images, create engaging games etc. with JavaScript.
- The canvas's 2D drawable layer surface Object is referred to as CanvasRenderingContext2D; or from a HTMLCanvasElement using the .getContext("2d") method:

```
var ctx =  
document.getElementById("myCanvas").getContext("2d")  
);
```

// now we can refer to the canvas's 2D layer context using

```
`ctx` ctx.fillStyle = "#f00";  
ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height); //  
x, y, width, height
```

```
ctx.fillStyle = "#000"; ctx.fillText("My red canvas with  
some black text", 84, 32); // text, x, y
```


Drawing two rectangles on a <canvas>:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <title>Draw two rectangles on the canvas</title>
  <style>
    canvas{
      border:1px solid gray;
    }
  </style>
  <script async>
    window.onload = init; // call init() once the window is
    completely loaded
```

```
function init(){
```

```
// #1 - get reference to <canvas> element
```

```
var canvas = document.querySelector('canvas');
```

```
// #2 - get reference to the drawing context and  
drawing API
```

```
var ctx = canvas.getContext('2d');
```

```
// #3 - all fill operations are now in red
```

```
ctx.fillStyle = 'red';
```

```
// #4 - fill a 100x100 rectangle at x=0,y=0
```

```
ctx.fillRect(0,0,100,100);
```

```
// #5 - all fill operations are now in green
```

```
ctx.fillStyle = 'green';
```

```
// #6 - fill a 50x50 rectangle at x=25,y=25
```

```
ctx.fillRect(25,25,50,50);
```

```
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<canvas width=300 height=200>Your browser does  
not support canvas.</canvas>
```

```
</body>
```

```
</html>
```

Meta Information:

- Meta tags in HTML documents provide useful information about the document including a description, keywords, author, dates of modifications and around 90 other fields.

Page Information:

application-name:

- Giving the name of the Web application that the page represents.

**<meta name="application-name"
content="OpenStreetMap">**

- If it's not a Web application, the application-name meta tag must not be used.

author

Set the author of the page:

<meta name="author" content="Your Name">

Only one name can be given.

description

Set the description of the page:

**<meta name="description" content="Page
Description">**

- The description meta tag can be used by various search engines while indexing your web page for searching purpose.
- Usually, the description contained within the meta tag is the short summary that shows up under the page/website's main title in the search engine results.
- Google usually uses only the first 20-25 words of your description.

Generator:

<meta name="generator" content="HTML Generator 1.42">

- Identifies one of the software packages used to generate the document.
- Only to be used for pages where the markup is automatically generated.

keywords

Set keywords for search engines (comma-separated):

<meta name="keywords" content="Keyword1, Keyword2">

- The keywords meta tag is sometimes used by search engines to know the search query which is relevant to your web page.
- As a rule of thumb, it is probably a good idea to not add too many words, as most search engines that use this meta tag for indexing will only index the first ~20 words.
- Make sure that you put the most important keywords first.

Character Encoding:

- The charset attribute specifies the character encoding for the HTML document and needs to be a valid character encoding (examples include windows-1252, ISO-8859-2, Shift_JIS, and UTF-8).

- UTF-8 (Unicode) is the most widely used and should be used for any new project.

<meta charset="UTF-8">

<meta charset="ISO-8859-1">

- All browsers have always recognized the <meta charset> form, but if you for some reason need your page to be valid HTML 4.01, you can use the following instead:

<meta http-equiv="content-type" content="text/html; charset=UTF-8">

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

Robots:

- The robots attribute, supported by several major search engines, controls whether search engine spiders are allowed to index a page or not and whether they should follow links from a page or not.

<meta name="robots" content="noindex">

This example instructs all search engines to not show the page in search results.

Other allowed values are:

Value/Directive

Meaning

all Default.

Equivalent to index, follow. See note below.

noindex

Do not index the page at all.

nofollow

Do not follow the links on this page

follow The links on the page can be followed. See note below.

none Equivalent to noindex, nofollow.

noarchive Do not make a cached version of this page available in search results.

nocache Synonym of noarchive used by some bots such as Bing.

nosnippet Do not show a snippet of this page in search results.

noodp Do not use metadata of this page from the Open Directory project for titles or snippets in search results.

notranslate Do not offer translations of this page in search results.

noimageindex Do not index images on this page.

unavailable_after [RFC-850

date/time] Do not show this page in search results after the specified date/time.

- The date/time must be specified in the RFC 850 format.

Note:

- Explicitly defining index and/or follow, while valid values, is not necessary as pretty much all search engines will assume they are allowed to do so if not explicitly prevented from doing so.
- Similar to how the robots.txt file operates, search engines generally only look for things they are not allowed to do.
- Only stating things a search engine isn't allowed to do also prevents accidentally stating opposites (such as index, ..., noindex) which not all search engines will treat in the same way.

- Social Media Open Graph is a standard for metadata that extends the normal information contained within a site's head markup.
- This enables websites such as Facebook to display deeper and richer information about a website in a structured format.
- This information is then automatically displayed when users share links to websites containing OG metadata on Facebook.

- **Facebook / Open Graph**

```
<meta property="fb:app_id" content="123456789">
```

```
<meta property="og:url"  
content="https://example.com/page.html">
```

```
<meta property="og:type" content="website">
```

```
<meta property="og:title" content="Content Title">
```

```
<meta property="og:image"
content="https://example.com/image.jpg">
<meta property="og:description" content="Description
Here">
<meta property="og:site_name" content="Site Name">
<meta property="og:locale" content="en_US">
<meta property="article:author" content="">
```

Mobile Layout Control:

Common mobile-optimized sites use the

<meta name="viewport"> tag like this:

<meta name="viewport" content="width=device-width, initial-scale=1">

- The viewport element gives the browser instructions on how to control the page's dimensions and scaling based on the device you are using.

- In the above example, content="width=device-width" means that the browser will render the width of the page at the width of its own screen.
- So if that screen is 480px wide, the browser window will be 480px wide.
- initialscale=1 depicts that the initial zoom (which is 1 in this case, means it does not zoom).
- Below are the attributes this tag supports

Attribute

Description

width The width of the virtual viewport of the device. Values1: device-width or the actual width in pixels, like 480

height The height of the virtual viewport of the device. Values2: device-height or the actual width in pixels, like 600

initial-scale The initial zoom when the page is loaded. 1.0 does not zoom.

minimum-scale The minimum amount the visitor can zoom on the page. 1.0 does not zoom.

maximum-scale The maximum amount the visitor can zoom on the page. 1.0 does not zoom.

user-scalable Allows the device to zoom in and out. Values are yes or no. If set to no, the user is not able to zoom in the webpage. The default is yes. Browser settings can ignore this rule.

Notes:

- The width property can be either specified in pixels (width=600) or by device-width (width=device-width) which represents the physical width of the device's screen.

- Similarly, the height property can be either specified in pixels (height=600) or by device-height (height=device-height) which represents the physical height of the device's screen.

Automatic Refresh:

- To refresh the page every five seconds, add this meta element in the head element:

<meta http-equiv="refresh" content="5">

- CAUTION! While this is a valid command, it is recommended that you do not use it because of its negative effects on user experience.
- Refreshing the page too often can cause it to become unresponsive, and often scrolls to the top of the page.
- If some information on the page needs to be updated continuously, there are much better ways to do that by only refreshing a portion of a page.

Phone Number Recognition:

- Mobile platforms like iOS automatically recognize phone numbers and turn them into tel: links.
- While the feature is very practical, the system sometimes detects ISBN codes and other numbers as telephone numbers.
- For mobile Safari and some other WebKit-based mobile browsers to turn off automatic phone number recognition and formatting, you need this meta tag:

**<meta name="format-detection"
content="telephone=no">**

Automatic redirect

- Sometimes your webpage needs a automatic redirect.
- For example, to redirect to example.com after 5 seconds:

**<meta http-equiv="refresh"
content="5;url=https://www.example.com/" />**

- This line will send you to the designated website (in this case example.com after 5 seconds).
- If you need to change the time delay before a redirect, simply changing the number right before your ;url= will alter the time delay.

Web App:

- You can set up your web app or website to have an application shortcut icon added to a device's homescreen, and have the app launch in full-screen "app mode" using Chrome for Android's "Add to homescreen" menu item.
- Below meta tag(s) will open web app in full-screen mode (without address bar).

Android Chrome

```
<meta name="mobile-web-app-capable"  
content="yes">
```

IOS

```
<meta name="apple-mobile-web-app-capable"  
content="yes">
```

- You can also set color for status bar and address bar in meta tag.

Android Chrome

```
<meta name="theme-color" content="black">
```

IOS

```
<meta name="apple-mobile-web-app-status-bar-style"  
content="black">
```

Marking up computer code:

Block with <pre> and <code>:

- If the formatting (white space, new lines, indentation) of the code matters, use the pre element in combination with the code element:

<pre>

<code>

x = 42

if x == 42:

print "x is 42"

</code>

</pre>

- You still have to escape characters with special meaning in HTML (like < with <), so for displaying a block of HTML code (<p>This is a paragraph.</p>), it could look like this:

```
<pre>
  <code>
    &lt;p>This is a paragraph.&lt;/p>
  </code>
</pre>
```

Inline with `<code>`:

- If a sentence contains computer code (for example, the name of an HTML element), use the code element to mark it up:

`<p>`The `<code>a</code>` element creates a hyperlink.**`</p>`**

Marking-up Quotes

Inline with `<q>`:

The q element can be used for a quote that is part of a sentence:

<p>She wrote <q>The answer is 42.</q> and everyone agreed.</p>

Quotation marks:

- Quotation marks should not be added. User agents should (in HTML 4.01) resp. must (in HTML 4.0) render them automatically.
- Quotation marks must not be added. User agents will render them automatically.

Source URL (cite attribute)

- The cite attribute can be used to reference the URL of the quoted source:

**<p>She wrote <q
cite="http://example.com/blog/hello-world">The
answer is 42.</q> and everyone
agreed.</p>**

- Note that browsers typically don't show this URL, so if the source is relevant, you should add a hyperlink (a element) in addition.

Block with <blockquote>

- The blockquote element can be used for a (block-level) quote:

<blockquote>

<p>The answer is 42.</p>

</blockquote>

Source URL (cite attribute)

- The cite attribute can be used to reference the URL of the quoted source:

<blockquote cite="http://example.com/blog/hello-world">

<p>The answer is 42.</p>

</blockquote>

- Note that browsers typically don't show this URL, so if the source is relevant, you should add a hyperlink (a element) in addition .
- The citation/attribution should not be part of the blockquote element:

```
<blockquote cite="http://example.com/blog/hello-world">
```

```
<p>The answer is 42.</p>
```

```
</blockquote>
```

```
<p>Source: <cite><a  
href="http://example.com/blog/hello-world"  
rel="external">Hello  
World</a></cite></p>
```

- You can add a div element to group the quote and the citation, but it exists no way to associate them semantically.
- The cite element can be used for the reference of the quoted source (but not for the author name).
- The citation/attribution (e.g., the hyperlink giving the source URL) can be inside the blockquote, but in that case it must be within a cite element (for in-text attributions) or a footer element:

<blockquote cite="http://example.com/blog/hello-world">

<p>The answer is 42.</p>

<footer>

<p>Source: <cite><a


```
href="http://example.com/blog/hello-world"  
rel="external">Hello  
World</a></cite></p>  
</footer>  
</blockquote>
```

- The cite element can be used for the reference of the quoted source, or for the name of the quote's author.

Summary:



Iframes



Content Languages



SVG



Canvas



Meta Information



Marking up computer code



Marking-up Quotes

Thank You.....

If you have any queries please write to info@uplatz.com".