Chapter 33: 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. This topic covers the usage and purpose of these tags.

Section 33.1: 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.

Section 33.2: 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.

Version = 5

```
<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">
```

See also the Encoding Standard, to view all available character encoding labels that browsers recognize.

Section 33.3: 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.

Section 33.4: 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="">
<!-- Facebook: https://developers.facebook.com/docs/sharing/webmasters#markup -->
<!-- Open Graph: http://ogp.me/ -->
```

- Facebook Open Graph Markup
- Open Graph protocol

Facebook / Instant Articles

```
<meta charset="utf-8">
<meta property="op:markup_version" content="v1.0">

<!-- The URL of the web version of your article -->
<link rel="canonical" href="http://example.com/article.html">

<!-- The style to be used for this article -->
<meta property="fb:article_style" content="myarticlestyle">
```

- Facebook Instant Articles: Creating Articles
- Instant Articles: Format Reference

Twitter uses its own markup for metadata. This metadata is used as information to control how tweets are displayed when they contain a link to the site.

Twitter

```
<meta name="twitter:card" content="summary">
<meta name="twitter:site" content="@site_account">
<meta name="twitter:creator" content="@individual_account">
<meta name="twitter:url" content="https://example.com/page.html">
<meta name="twitter:title" content="Content Title">
<meta name="twitter:description" content="Content description less than 200 characters">
<meta name="twitter:image" content="https://example.com/image.jpg">
```

- Twitter Cards: Getting Started Guide
- Twitter Card Validator

Google+ / Schema.org

```
<link href="https://plus.google.com/+YourPage" rel="publisher">
<meta itemprop="name" content="Content Title">
<meta itemprop="description" content="Content description less than 200 characters">
<meta itemprop="image" content="https://example.com/image.jpg">
```

Section 33.5: 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`. `initial-scale=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:

1 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.

2 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.

Section 33.6: 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.

Section 33.7: 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">
```

Section 33.8: 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 is 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.

Section 33.9: 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">
```