

HTML Semantic Elements

Semantic elements = elements with a meaning.

What are Semantic Elements?

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic** elements: `<div>` and `` - Tells nothing about its content.

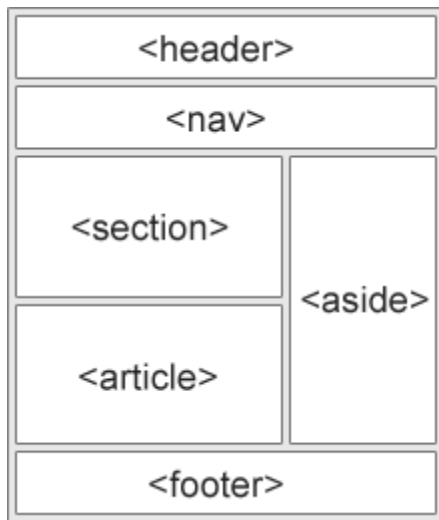
Examples of **semantic** elements: `<form>`, `<table>`, and `<article>` - Clearly defines its content.

Semantic Elements in HTML

Many web sites contain HTML code like: `<div id="nav">` `<div class="header">` `<div id="footer">` to indicate navigation, header, and footer.

In HTML there are some semantic elements that can be used to define different parts of a web page:

- `<article>`
- `<aside>`
- `<details>`
- `<figcaption>`
- `<figure>`
- `<footer>`
- `<header>`
- `<main>`
- `<mark>`
- `<nav>`
- `<section>`
- `<summary>`
- `<time>`



HTML `<section>` Element

The `<section>` element defines a section in a document.

According to W3C's HTML documentation: "A section is a thematic grouping of content, typically with a heading."

A web page could normally be split into sections for introduction, content, and contact information.

Example

Two sections in a document:

```
<section>
<h1>WWF</h1>
<p>The World Wide Fund for Nature (WWF) is an international organization working on
issues regarding the conservation, research and restoration of the environment,
formerly named the World Wildlife Fund. WWF was founded in 1961.</p>
</section>
```

```
<section>
<h1>WWF's Panda symbol</h1>
<p>The Panda has become the symbol of WWF. The well-known panda logo of WWF originated
from a panda named Chi Chi that was transferred from the Beijing Zoo to the London Zoo
in the same year of the establishment of WWF.</p>
</section>
```

HTML <article> Element

The `<article>` element specifies independent, self-contained content.

An article should make sense on its own, and it should be possible to distribute it independently from the rest of the web site.

Examples of where an `<article>` element can be used:

- Forum post
- Blog post
- Newspaper article

Example

Three articles with independent, self-contained content:

```
<article>
<h2>Google Chrome</h2>
<p>Google Chrome is a web browser developed by Google, released in 2008. Chrome is the
world's most popular web browser today!</p>
</article>
```

```
<article>
<h2>Mozilla Firefox</h2>
<p>Mozilla Firefox is an open-source web browser developed by Mozilla. Firefox has
been the second most popular web browser since January, 2018.</p>
</article>
```

```
<article>
<h2>Microsoft Edge</h2>
<p>Microsoft Edge is a web browser developed by Microsoft, released in 2015. Microsoft
Edge replaced Internet Explorer.</p>
</article>
```

Example 2

Use CSS to style the `<article>` element:

```
<html>
<head>
<style>
.all-browsers {
    margin: 0;
    padding: 5px;
}
```

```

    background-color: lightgray;
}

.all-browsers > h1, .browser {
    margin: 10px;
    padding: 5px;
}

.browser {
    background: white;
}

.browser > h2, p {
    margin: 4px;
    font-size: 90%;
}
</style>
</head>
<body>

<article class="all-browsers">
    <h1>Most Popular Browsers</h1>
    <article class="browser">
        <h2>Google Chrome</h2>
        <p>Google Chrome is a web browser developed by Google, released in 2008. Chrome is
the world's most popular web browser today!</p>
    </article>
    <article class="browser">
        <h2>Mozilla Firefox</h2>
        <p>Mozilla Firefox is an open-source web browser developed by Mozilla. Firefox has
been the second most popular web browser since January, 2018.</p>
    </article>
    <article class="browser">
        <h2>Microsoft Edge</h2>
        <p>Microsoft Edge is a web browser developed by Microsoft, released in 2015.
Microsoft Edge replaced Internet Explorer.</p>
    </article>
</article>

</body>
</html>

```

Nesting <article> in <section> or Vice Versa?

The `<article>` element specifies independent, self-contained content.

The `<section>` element defines section in a document.

Can we use the definitions to decide how to nest those elements? No, we cannot!

So, you will find HTML pages with `<section>` elements containing `<article>` elements, and `<article>` elements containing `<section>` elements.

HTML <header> Element

The `<header>` element represents a container for introductory content or a set of navigational links.

A `<header>` element typically contains:

- one or more heading elements (`<h1>` - `<h6>`)
- logo or icon
- authorship information

Note: You can have several `<header>` elements in one HTML document. However, `<header>` cannot be placed within a `<footer>`, `<address>` or another `<header>` element.

Example

A header for an `<article>`:

```
<article>
  <header>
    <h1>What Does WWF Do?</h1>
    <p>WWF's mission:</p>
  </header>
  <p>WWF's mission is to stop the degradation of our planet's natural environment,
  and build a future in which humans live in harmony with nature.</p>
</article>
```

HTML <footer> Element

The `<footer>` element defines a footer for a document or section.

A `<footer>` element typically contains:

- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

You can have several `<footer>` elements in one document.

Example

A footer section in a document:

```
<footer>
  <p>Author: Hege Refsnes</p>
  <p><a href="mailto:hege@example.com">hege@example.com</a></p>
</footer>
```

HTML <nav> Element

The `<nav>` element 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 block 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.

Example

A set of navigation links:

```
<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
```

```
<a href="/jquery/">jQuery</a>
</nav>
```

HTML <aside> Element

The `<aside>` element defines some content aside from the content it is placed in (like a sidebar).

The `<aside>` content should be indirectly related to the surrounding content.

Example

Display some content aside from the content it is placed in:

```
<p>My family and I visited The Epcot center this summer. The weather was nice, and
Epcot was amazing! I had a great summer together with my family!</p>
```

```
<aside>
<h4>Epcot Center</h4>
<p>Epcot is a theme park at Walt Disney World Resort featuring exciting attractions,
international pavilions, award-winning fireworks and seasonal special events.</p>
</aside>
```

Example 2

Use CSS to style the `<aside>` element:

```
<html>
<head>
<style>
aside {
  width: 30%;
  padding-left: 15px;
  margin-left: 15px;
  float: right;
  font-style: italic;
  background-color: lightgray;
}
</style>
</head>
<body>
```

```
<p>My family and I visited The Epcot center this summer. The weather was nice, and
```

Epcot was amazing! I had a great summer together with my family!</p>

<aside>

<p>The Epcot center is a theme park at Walt Disney World Resort featuring exciting attractions, international pavilions, award-winning fireworks and seasonal special events.</p>

</aside>

<p>My family and I visited The Epcot center this summer. The weather was nice, and Epcot was amazing! I had a great summer together with my family!</p>

<p>My family and I visited The Epcot center this summer. The weather was nice, and Epcot was amazing! I had a great summer together with my family!</p>

</body>

</html>

HTML <figure> and <figcaption> Elements

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

The <figcaption> tag defines a caption for a <figure> element. The <figcaption> element can be placed as the first or as the last child of a <figure> element.

The element defines the actual image/illustration.

Example

```
<figure>
  
  <figcaption>Fig1. - Trulli, Puglia, Italy.</figcaption>
</figure>
```

Why Semantic Elements?

According to the W3C: "A semantic Web allows data to be shared and reused across applications, enterprises, and communities."

Semantic Elements in HTML

Below is a list of some of the semantic elements in HTML.

Tag	Description
<code><article></code>	Defines independent, self-contained content
<code><aside></code>	Defines content aside from the page content
<code><details></code>	Defines additional details that the user can view or hide
<code><figcaption></code>	Defines a caption for a <code><figure></code> element
<code><figure></code>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<code><footer></code>	Defines a footer for a document or section
<code><header></code>	Specifies a header for a document or section
<code><main></code>	Specifies the main content of a document
<code><mark></code>	Defines marked/highlighted text

[<nav>](#)

Defines navigation links

[<section>](#)

Defines a section in a document

[<summary>](#)

Defines a visible heading for a <details> element

[<time>](#)

Defines a date/time

For a complete list of all available HTML tags, visit our [HTML Tag Reference](#).