HTML 5 New Features

By Ata Ebrahimi
www.AtaEbrahimi.com

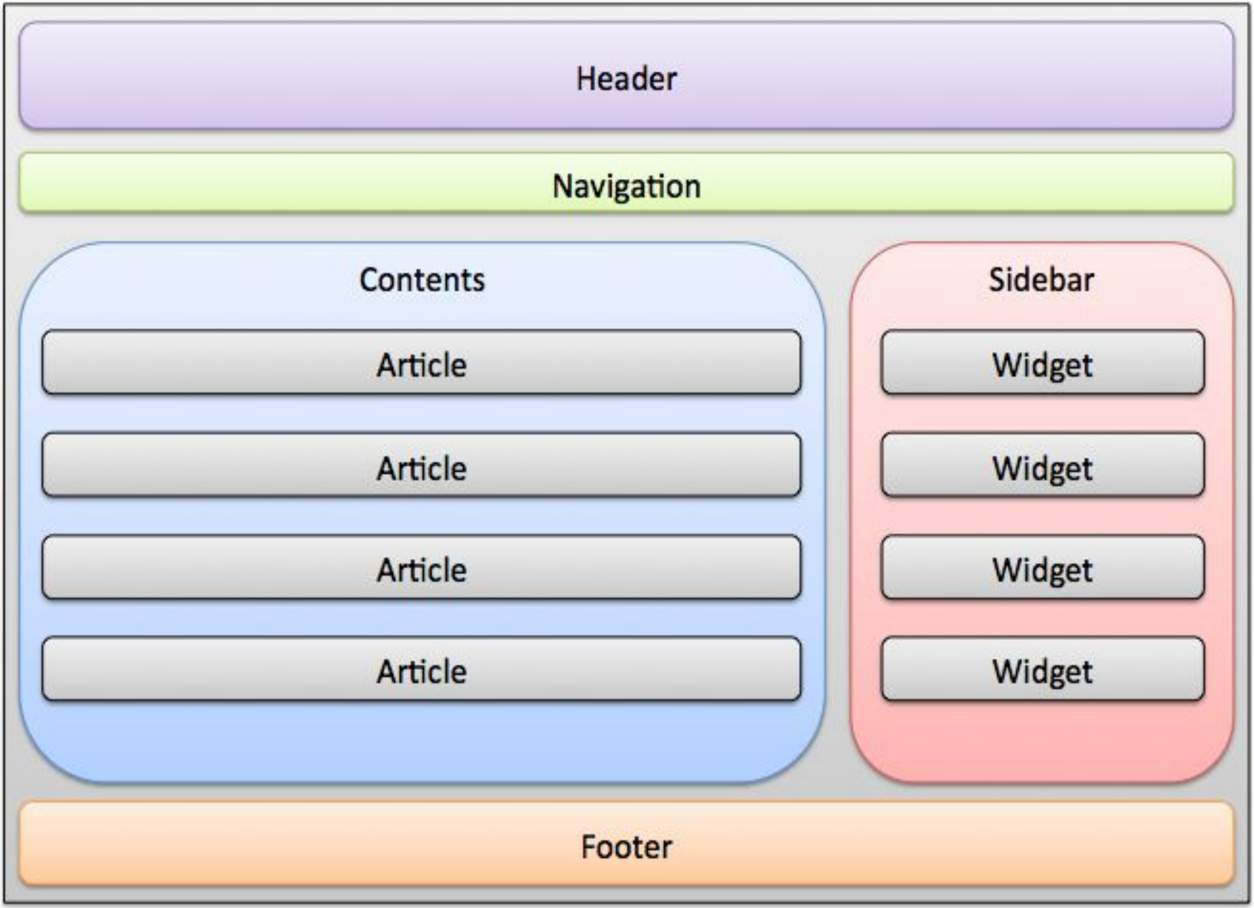
New Features

HTML5 introduces a number of new elements and attributes that helps in building a modern websites. Following are great features introduced in HTML5.

- **New Semantic Elements:** These are like <header>, <footer>, and <section>.

- **Forms 2.0:** Improvements to HTML web forms where new attributes have been introduced for <input> tag.

- **Persistent Local Storage:** To achieve without resorting to third-party plugins.



Header

Navigation

Contents

Article

Article

Article

Article

Sidebar

Widget

Widget

Widget

Widget

Footer

New Features

- **WebSocket** : A next-generation bidirectional communication technology for web applications.
- **Server-Sent Events:** HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).
- **Canvas:** This supports a two-dimensional drawing surface that you can program with JavaScript.
- **Audio & Video:** You can embed audio or video on your web pages without resorting to third-party plugins.

New Features

- **Geolocation:** Now visitors can choose to share their physical location with your web application.
- **Microdata:** This lets you create your own vocabularies beyond HTML5 and extend your web pages with custom semantics.
- **Drag and drop:** Drag and drop the items from one location to another location on a the same webpage.

New Tags

- <article>
- <aside>
- <audio>
- <bdi>
- <canvas>
- <datalist>
- <details>
- <diaog>
- <embed>
- <figcaption>
- <figure>
- <footer>

New Tags

- <header>
- <keygen>
- <main>
- <mark>
- <menuitem>
- <meter>
- <nav>
- <output>
- <progress>
- <rp>
- <rt>
- <ruby>

New Tags

- <section>
- <source>
- <summary>
- <time>
- <track>
- <video>
- <wbr>

Removed Tags

- <acronym>
- <applet>
- <basefont>
- <big>
- <center>
- <dir>
- <font>
- <frame>
- <frameset>
- <noframes>
- <strike>
- <tt>

<article> Tag
Definition and Usage

The <article> tag 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 site.

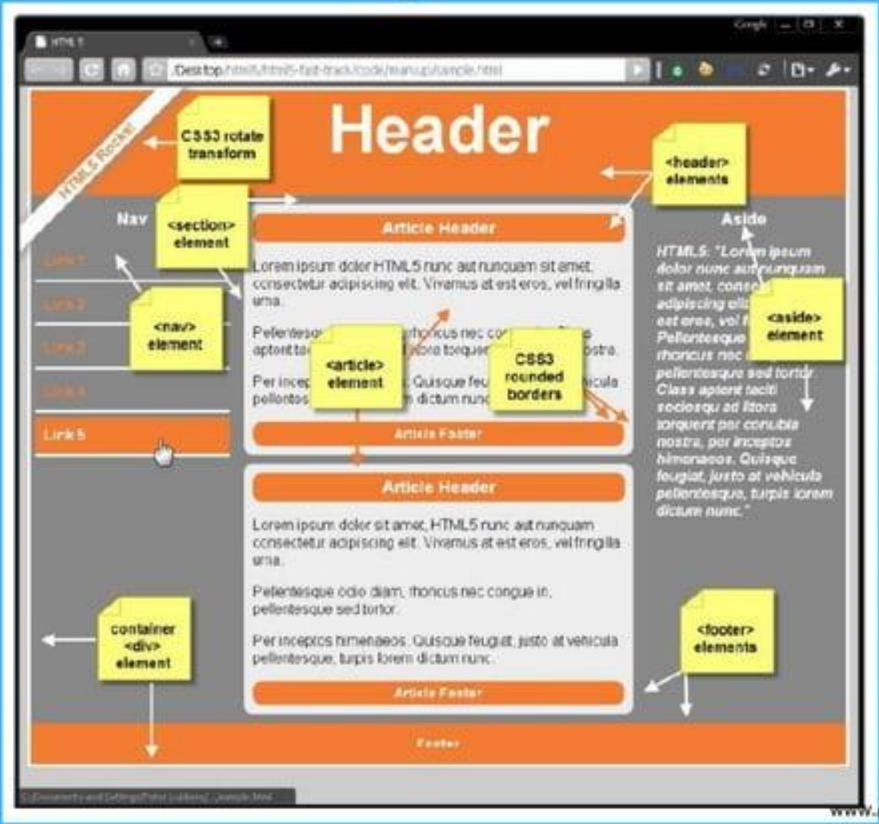
Potential sources for the <article> element:

- Forum post
- Blog post
- News story
- Comment

<article> Tag Example

```
<article>
  <h1>HTML 5 New Features Presentation</h1>
  <p>HTML5 is a standard for structuring and presenting web content</p>
</article>
```

<article> Tag



Lorem ipsum dolor HTML5 nunc aut nunquam sit amet, consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesq rhoncus nec co aptent ta litora torque nostra.

Per incep Quisque feu vehicula pellentes dictum nunc

Lorem ipsum dolor sit amet, HTML5 nunc aut nunquam consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesque odio diam, rhoncus nec congue in, pellentesque sed tortor.

Per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

HTML5: "Lorem ipsum dolor nunc aut nunquam sit amet, conse adipiscing eli est eros, vel f Pellentesque rhoncus nec pellentesque sed tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

<aside> Tag
Definition and Usage

The <aside> tag defines some content aside from the content it is placed in.

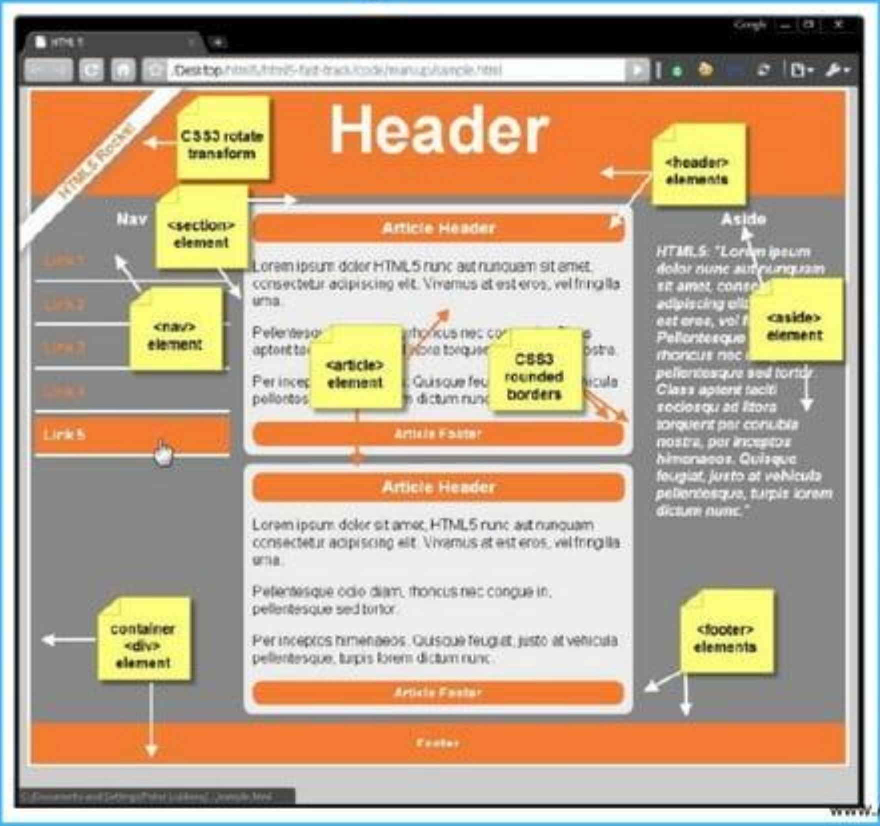
The aside content should be related to the surrounding content.

<aside> Tag Example

```
<p>My family and I visited The Hyper ME center this summer.</p>

<aside>
  <h4>Hyper ME Center</h4>
  <p>The Hyper ME Center is located in the west of Tehran.</p>
</aside>
```

<aside> Tag



Lorem ipsum dolor HTML5 nunc aut nunquam sit amet, consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesq… rhoncus nec co… aptent ta… litora torque… nostra.

Per incep… Quisque feu… vehicula pellentes… dictum nunc.

Lorem ipsum dolor sit amet, HTML5 nunc aut nunquam consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesque odio diam, rhoncus nec congue in, pellentesque sed tortor.

Per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

HTML5: "Lor… ipsum dolor nunc aut nunquam sit amet, conse… adipiscing eli… est eros, vel f… Pellentesque … rhoncus nec … pellentesque sed tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

<audio> Tag
Definition and Usage

The <audio> tag defines sound, such as music or other audio streams.

Currently, there are 3 supported file formats for the <audio> element: MP3, Wav, and Ogg:

Browser	MP3	Wav	Ogg
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	YES	NO
Opera	YES	YES	YES

<audio> Tag
MIME Types for Audio Formats

Format	MIME-type
MP3	audio/mpeg
Ogg	audio/ogg
Wav	audio/wav

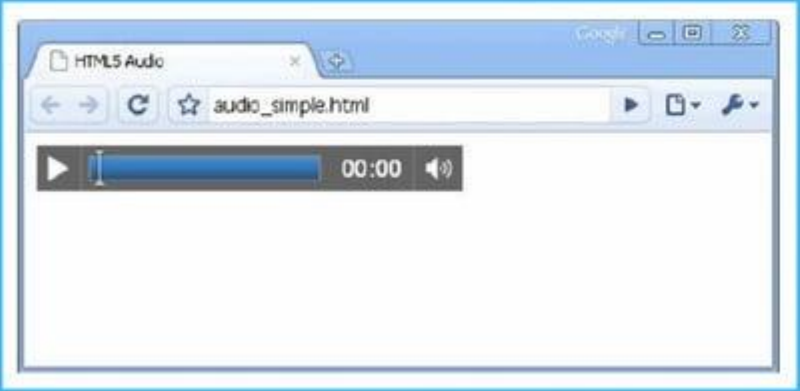
<audio> Tag Attributes

Attribute	Value	Description
autoplay	autoplay	Specifies that the audio will start playing as soon as it is ready
controls	controls	Specifies that audio controls should be displayed (such as a play/pause button etc)
loop	loop	Specifies that the audio will start over again, every time it is finished
muted	muted	Specifies that the audio output should be muted
preload	auto metadata none	Specifies if and how the author thinks the audio should be loaded when the page loads
src	*URL*	Specifies the URL of the audio file

<audio> Tag Example

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio tag.
</audio>
```

<audio> Tag



<bdi> Tag
Definition and Usage

bdi stands for Bi-directional Isolation.

The <bdi> tag isolates a part of text that might be formatted in a different direction from other text outside it.

This element is useful when embedding user-generated content with an unknown directionality.

<bdi> Tag Example

```
<ul>
<li>User <bdi>hrefs</bdi>: 60 points</li>
<li>User <bdi>jdoe</bdi>: 80 points</li>
<li>User <bdi>/>إيانbdi>: 90 points</li>
</ul>
```

<canvas> Tag
Definition and Usage

The <canvas> tag is used to draw graphics, on the fly, via scripting (usually JavaScript).

The <canvas> tag is only a container for graphics, you must use a script to actually draw the graphics.

<canvas> Tag Example

```
<canvas id="myCanvas"></canvas>

<script>
var canvas = document.getElementById("myCanvas");
var ctx = canvas.getContext("2d");
ctx.fillStyle = "#FF0000";
ctx.fillRect(0, 0, 80, 80);
</script>
```

<canvas> Tag Attributes

Attribute	Value	Description
Height	*pixels*	Specifies the height of the canvas
width	*pixels*	Specifies the width of the canvas

<datalist> Tag Definition and Usage

The <datalist> tag specifies a list of pre-defined options for an <input> element.

The <datalist> tag is used to provide an "autocomplete" feature on <input> elements. Users will see a drop-down list of pre-defined options as they input data.

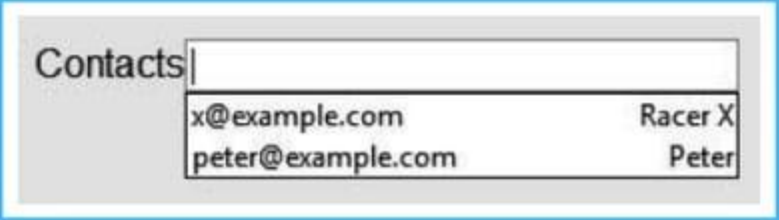
Use the <input> element's list attribute to bind it together with a <datalist> element.

<datalist> Tag Example

```
<input list="browsers">

<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

<datalist> Tag



Contacts

x@example.com	Racer X
peter@example.com	Peter

<details> Tag
Definition and Usage

The <details> tag specifies additional details that the user can view or hide on demand.

The <details> tag can be used to create an interactive widget that the user can open and close. Any sort of content can be put inside the <details> tag.

The content of a <details> element should not be visible unless the open attribute is set.

<details> Tag Example

```
<details>
  <summary>Copyright 1999-2015.</summary>
  <p> - by Ata Ebrahimi. All Rights Reserved.</p>
  <p>All content and graphics on this presentation are the property of the
Ata Ebrahimi.</p>
</details>
```

<details> Tag Attributes

Attribute	Value	Description
open	open	Specifies that the details should be visible (open) to the user

<dialog> Tag
Definition and Usage

The <dialog> tag defines a dialog box or window.

The <dialog> element makes it easy to create popup dialogs and modals on a web page.

<dialog> Tag Example

```
<table>
<tr>
  <th>January <dialog open>This is an open dialog window</dialog></th>
  <th>February</th>
  <th>March</th>
</tr>
<tr>
  <td>31</td>
  <td>28</td>
  <td>31</td>
</tr>
</table>
```

<dialog> Tag Attributes

Attribute	Value	Description
open	open	Specifies that the dialog element is active and that the user can interact with it

<embed> Tag
Definition and Usage

The <embed> tag defines a container for an external application or interactive content (a plug-in).

<embed> Tag Example

```
<embed src="helloworld.swf">
```

<embed> Tag Attributes

Attribute	Value	Description
height	*pixels*	Specifies the height of the embedded content
src	*URL*	Specifies the address of the external file to embed
type	*media_type*	Specifies the media type of the embedded content
width	*pixels*	Specifies the width of the embedded content

<figcaption> Tag
Definition and Usage

The <figcaption> tag defines a caption for a <figure> element.

The <figcaption> element can be placed as the first or last child of the <figure> element.

<figcaption> Tag Example

```
<figure>
  <img src="img_pulpit.jpg" alt="The Pulpit Rock" width="304"
height="228">
  <figcaption>Fig1. - A view of the pulpit rock in Norway.</figcaption>
</figure>
```

<figure> Tag
Definition and Usage

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

While the content of the <figure> element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

<figure> Tag Example

```
<figure>
  <img src="img_pulpit.jpg" alt="The Pulpit Rock" width="304"
height="228">
</figure>
```

<footer> Tag
Definition and Usage

The <footer> tag defines a footer for a document or section.

A <footer> element should contain information about its containing element.

A <footer> element typically contains:

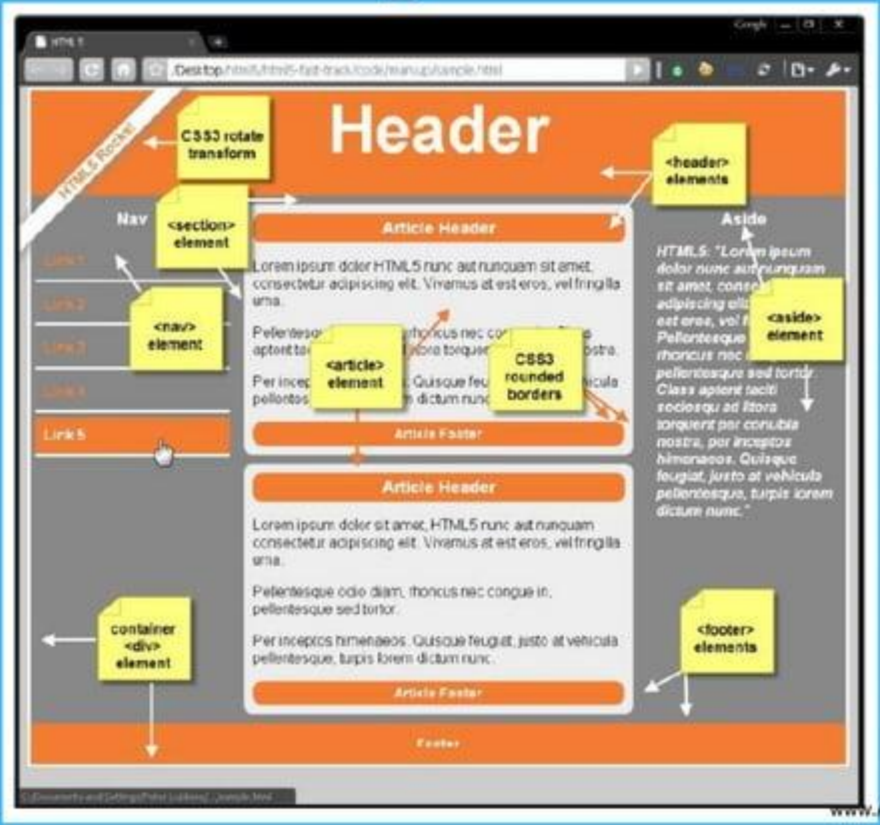
- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

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

<footer> Tag Example

```
<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
  someone@example.com</a>.</p>
</footer>
```

<footer> Tag



Lorem ipsum dolor HTML5 nunc aut nunquam sit amet, consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesq… rhoncus nec co… aptent ta… litora torque… nostra.

Per incep… Quisque feu… vehicula pellentes… dictum nunc.

Lorem ipsum dolor sit amet, HTML5 nunc aut nunquam consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesque odio diam, rhoncus nec congue in, pellentesque sed tortor.

Per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

HTML5: "Lorem ipsum dolor nunc aut nunquam sit amet, conse… adipiscing eli… est eros, vel f… Pellentesque… rhoncus nec… pellentesque sed tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

<header> Tag
Definition and Usage

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

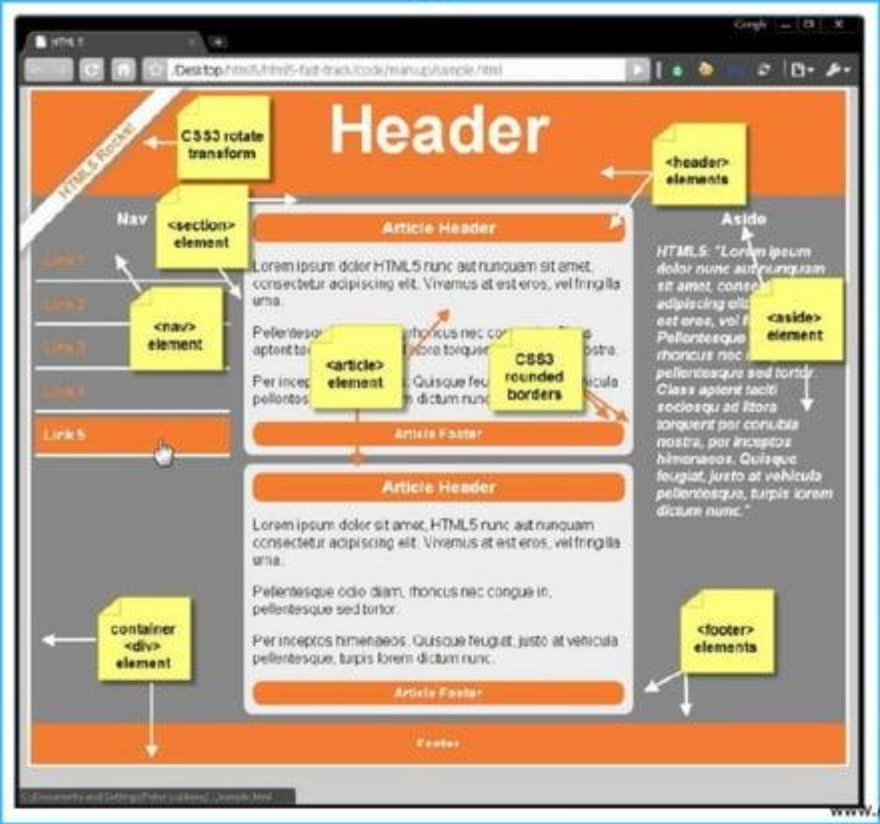
You can have several <header> elements in one document.

Note: A <header> tag cannot be placed within a <footer>, <address> or another <header> element.

<header> Tag Example

```
<article>
  <header>
    <h1>Most important heading here</h1>
    <h3>Less important heading here</h3>
    <p>Some additional information here</p>
  </header>
  <p>This presentation designed by Ata Ebrahimi</p>
</article>
```

<header> Tag



Lorem ipsum dolor HTML5 nunc aut nunquam sit amet, consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesq... rhoncus nec co... aptent ta... litora torque... nostra.

Per incep... Quisque feu... vehicula pellentes... dictum nunc

Lorem ipsum dolor sit amet, HTML5 nunc aut nunquam consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesque odio diam, rhoncus nec congue in, pellentesque sed tortor.

Per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

HTML5: "Lorem ipsum dolor nunc aut nunquam sit amet, conse... adipiscing eli... est eros, vel f... Pellentesque ... rhoncus nec ... pellentesque sed tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

<keygen> Tag
Definition and Usage

The <keygen> tag specifies a key-pair generator field used for forms.

When the form is submitted, the private key is stored locally, and the public key is sent to the server.

<keygen> Tag Example

```
<meter value="2" min="0" max="10">2 out of 10</meter><br>
<meter value="0.6">60%</meter>
```

<keygen> Tag Attributes

Attribute	Value	Description
autofocus	autofocus	Specifies that a <keygen> element should automatically get focus when the page loads
challenge	challenge	Specifies that the value of the <keygen> element should be challenged when submitted
disabled	disabled	Specifies that a <keygen> element should be disabled
form	*form_id*	Specifies one or more forms the <keygen> element belongs to
keytype	rsa dsa ec	Specifies the security algorithm of the key
name	*name*	Defines a name for the <keygen> element

<main> Tag
Definition and Usage

The <main> tag specifies the main content of a document.

The content inside the <main> element should be unique to the document. It should not contain any content that is repeated across documents such as sidebars, navigation links, copyright information, site logos, and search forms.

Note: There must not be more than one <main> element in a document. The <main> element must NOT be a descendant of an <article>, <aside>, <footer>, <header>, or <nav> element.

<main> Tag Example

```
<main>
 <h1>Web Browsers</h1>
 <p>Google Chrome, Firefox are the most used browsers today.</p>
 <article>
  <h1>Google Chrome</h1>
  <p>Google Chrome is a free web browser developed by Google,
  released in 2008.</p>
 </article>
 <article>
  <h1>Internet Explorer</h1>
  <p>Internet Explorer is a free web browser from Microsoft</p>
 </article>
</main>
```

<mark> Tag
Definition and Usage

The <mark> tag defines marked text.

Use the <mark> tag if you want to highlight parts of your text.

<mark> Tag Example

```
<p>Do not forget to buy <mark>milk</mark> today.</p>
```

<menuitem> Tag Definition and Usage

The <menuitem> tag defines a command/menu item that the user can invoke from a popup menu.

<menuitem> Tag Example

```
<menu type="context" id="mymenu">
  <menuitem label="Refresh" onclick="window.location.reload();"
icon="ico_reload.png">
  </menuitem>
  <menu label="Share on...">
    <menuitem label="Twitter" icon="ico_twitter.png"
    onclick="window.open('//twitter.com/intent/tweet?text='+window.location.href);">
    </menuitem>
    <menuitem label="Facebook" icon="ico_facebook.png"
    onclick="window.open('//facebook.com/sharer/sharer.php?u='+window.location.
href);">
    </menuitem>
  </menu>
  <menuitem label="Email This Page"
  onclick="window.location='mailto:?body='+window.location.href;"></menuitem>
</menu>
```

<menuitem> Tag Attributes

Attribute	Value	Description
checked	Checked	Specifies that the command/menu item should be checked when the page loads. Only for type="radio" or type="checkbox"
command		
default	default	Marks the command/menu item as being a default command
disabled	disabled	Specifies that the command/menu item should be disabled
icon	*URL*	Specifies an icon for the command/menu item
label	*text*	Required. Specifies the name of the command/menu item, as shown to the user
radiogroup	*groupname*	Specifies the name of the group of commands that will be toggled when the command/menu item itself is toggled. Only for type="radio"
type	checkbox command radio	Specifies the type of command/menu item. Default is "command"

<meter> Tag
Definition and Usage

The <meter> tag defines a scalar measurement within a known range, or a fractional value. This is also known as a gauge.

Examples: Disk usage, the relevance of a query result, etc.

Note: The <meter> tag should not be used to indicate progress (as in a progress bar). For progress bars, use the <progress> tag.

<meter> Tag Example

```
<meter value="2" min="0" max="10">2 out of 10</meter><br>
<meter value="0.6">60%</meter>
```

<meter> Tag Attributes

Attribute	Value	Description
form	*form_id*	Specifies one or more forms the <meter> element belongs to
high	*number*	Specifies the range that is considered to be a high value
low	*number*	Specifies the range that is considered to be a low value
max	*number*	Specifies the maximum value of the range
min	*number*	Specifies the minimum value of the range
optimum	*number*	Specifies what value is the optimal value for the gauge
value	*number*	Required. Specifies the current value of the gauge

<nav> Tag
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 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.

<nav> Tag Example

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

<output> Tag
Definition and Usage

The <output> tag represents the result of a calculation (like one performed by a script).

<output> Tag Example

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">0
 <input type="range" id="a" value="50">100
 +<input type="number" id="b" value="50">
 =<output name="x" for="a b"></output>
</form>
```

<output> Tag Attributes

Attribute	Value	Description
for	*element_id*	Specifies the relationship between the result of the calculation, and the elements used in the calculation
form	*form_id*	Specifies one or more forms the output element belongs to
name	*name*	Specifies a name for the output element

<progress> Tag
Definition and Usage

The <progress> tag represents the progress of a task.

Tip: Use the <progress> tag in conjunction with JavaScript to display the progress of a task.

Note: The <progress> tag is not suitable for representing a gauge (e.g. disk space usage or relevance of a query result). To represent a gauge, use the <meter> tag instead.

<progress> Tag Example

```
<progress value="22" max="100"></progress>
```

<progress> Tag Attributes

Attribute	Value	Description
max	*number*	Specifies how much work the task requires in total
value	*number*	Specifies how much of the task has been completed

<rp> Tag
Definition and Usage

The <rp> tag defines what to show if a browser does NOT support ruby annotations.

Ruby annotations are used for East Asian typography, to show the pronunciation of East Asian characters.

Use the <rp> tag together with the <ruby> and the <rt> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that not support ruby annotations.

<rp> Tag Example

```
<ruby>
漢 <rt><rp>(</rp>ㄏㄢˋ<rp>)</rp></rt>
</ruby>
```

<rt> Tag
Definition and Usage

The <rt> tag defines an explanation or pronunciation of characters (for East Asian typography) in a ruby annotation.

Use the <rt> tag together with the <ruby> and the <rp> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that not support ruby annotations.

<rt> Tag Example

```
<ruby>
漢 <rt> ㄏㄢˋ </rt>
</ruby>
```

<ruby> Tag
Definition and Usage

The <ruby> tag specifies a ruby annotation.

Ruby annotations are used for East Asian typography, to show the pronunciation of East Asian characters.

Use the <ruby> tag together with the <rt> and/or the <rp> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that not support ruby annotations.

<ruby> Tag Example

```
<ruby>
漢 <rt> ㄏㄢˋ </rt>
</ruby>
```

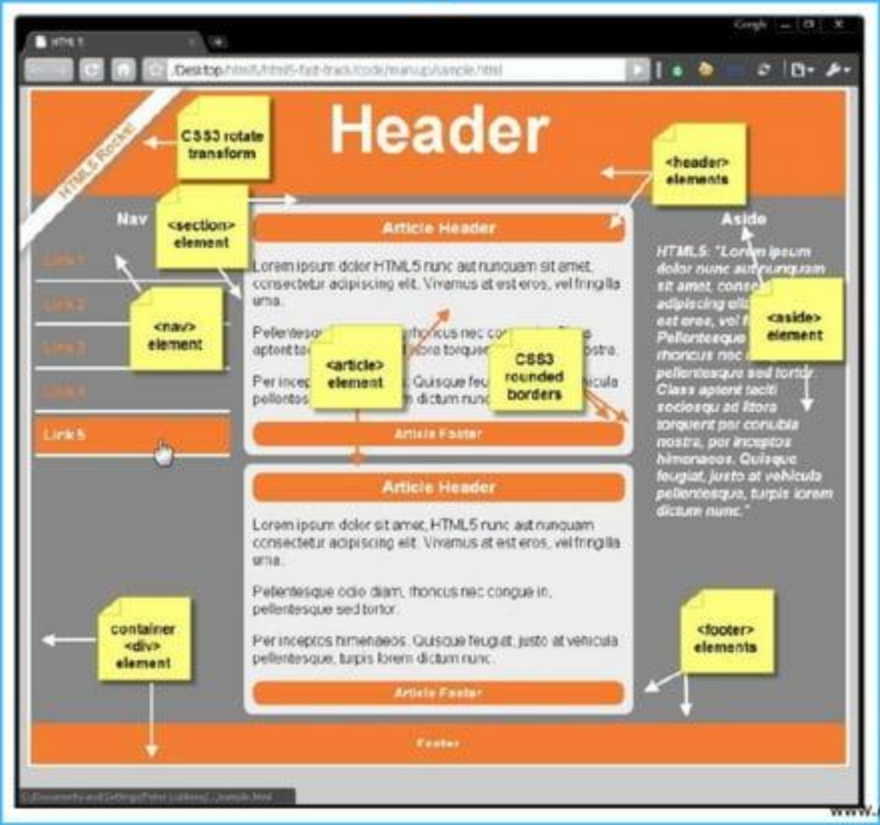
<section> Tag
Definition and Usage

The <section> tag defines sections in a document, such as chapters, headers, footers, or any other sections of the document.

<section> Tag Example

```
<section>
 <h1>WWF</h1>
 <p>The World Wide Fund for Nature (WWF) is....</p>
</section>
```

<section> Tag



Lorem ipsum dolor HTML5 nunc aut nunquam sit amet, consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesq [illegible] rhoncus nec co [illegible] aptent ta [illegible] litora torque [illegible] ostra.

Per incep [illegible] Quisque feu [illegible] vehicula pellentes [illegible] dictum nunc.

Lorem ipsum dolor sit amet, HTML5 nunc aut nunquam consectetur adipiscing elit. Vivamus at est eros, vel fringilla urna.

Pellentesque odio diam, rhoncus nec congue in, pellentesque sed tortor.

Per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

HTML5: "Lorem ipsum dolor nunc aut nunquam sit amet, conse [illegible] adipiscing eli [illegible] est eros, vel f [illegible] Pellentesque [illegible] rhoncus nec [illegible] pellentesque sed tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

<source> Tag
Definition and Usage

The <source> tag is used to specify multiple media resources for media elements, such as <video> and <audio>.

The <source> tag allows you to specify alternative video/audio files which the browser may choose from, based on its media type or codec support.

<source> Tag Example

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

<source> Tag Attributes

Attribute	Value	Description
media	*media_query*	Specifies the type of media resource
src	*URL*	Specifies the URL of the media file
type	*media_type*	Specifies the media type of the media resource

<summary> Tag
Definition and Usage

The <summary> tag defines a visible heading for the <details> element. The heading can be clicked to view/hide the details.

Note: The <summary> element should be the first child element of the <details> element.

<summary> Tag Example

```
<details>
  <summary>Copyright 1999-2014.</summary>
  <p> - by Refsnes Data. All Rights Reserved.</p>
  <p>All content and graphics on this web site are the property of the
company Refsnes Data.</p>
</details>
```

<time> Tag
Definition and Usage

The <time> tag defines a human-readable date/time.

This element can also be used to encode dates and times in a machine-readable way so that user agents can offer to add birthday reminders or scheduled events to the user's calendar, and search engines can produce smarter search results.

<time> Tag Example

```
<p>We open at <time>10:00</time> every morning.</p>

<p>I have a date on <time datetime="2008-02-14 20:00">Valentines
day</time>.</p>
```

<time> Tag Attributes

Attribute	Value	Description
datetime	*datetime*	Represent a machine-readable date/time of the <time> element

<track> Tag
Definition and Usage

The <track> tag specifies text tracks for media elements (<audio> and <video>).

This element is used to specify subtitles, caption files or other files containing text, that should be visible when the media is playing.

<track> Tag Example

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
label="Norwegian">
</video>
```

<track> Tag Attributes

Attribute	Value	Description
default	default	Specifies that the track is to be enabled if the user's preferences do not indicate that another track would be more appropriate
kind	captions chapters descriptions metadata subtitles	Specifies the kind of text track
label	*text*	Specifies the title of the text track
src	*URL*	Required. Specifies the URL of the track file
srclang	*language_code*	Specifies the language of the track text data (required if kind="subtitles")

<video> Tag
Definition and Usage

The <video> tag specifies video, such as a movie clip or other video streams.

Currently, there are 3 supported video formats for the <video> element: MP4, WebM, and Ogg:

Browser	MP4	WebM	Ogg
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES

Firefox	YES
	from Firefox 21
	from Firefox 30 for Linux
		YES	YES

Safari	YES	NO	NO
Opera	YES From Opera 25	YES	YES

<video> Tag
MIME Types for Video Formats

Format	MIME-type
MP4	video/mp4
WebM	video/webm
Ogg	video/ogg

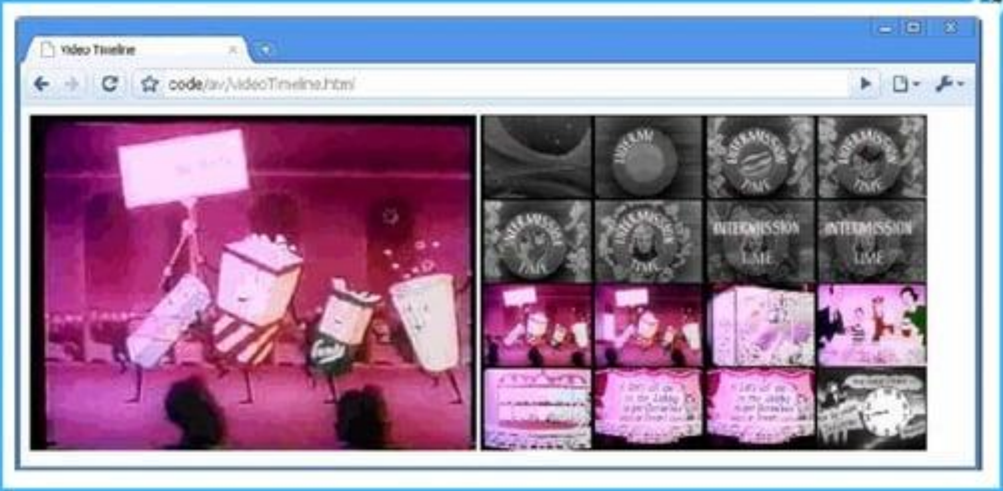
<video> Tag Example

```
<video width="320" height="240" controls>
 <source src="movie.mp4" type="video/mp4">
 <source src="movie.ogg" type="video/ogg">
 Your browser does not support the video tag.
</video>
```

<video> Tag Attributes

Attribute	Value	Description
autoplay	Autoplay	Specifies that the video will start playing as soon as it is ready
controls	controls	Specifies that video controls should be displayed (such as a play/pause button etc).
height	*pixels*	Sets the height of the video player
loop	loop	Specifies that the video will start over again, every time it is finished
muted	muted	Specifies that the audio output of the video should be muted
poster	*URL*	Specifies an image to be shown while the video is downloading, or until the user hits the play button
preload	auto metadata none	Specifies if and how the author thinks the video should be loaded when the page loads
src	*URL*	Specifies the URL of the video file
width	*pixels*	Sets the width of the video player

<video> Tag


Video Timeline
code/av/videoTimeline.html
INTERMISSION TIME

<wbr> Tag
Definition and Usage

The <wbr> (Word Break Opportunity) tag specifies where in a text it would be ok to add a line-break.

Tip: When a word is too long, or you are afraid that the browser will break your lines at the wrong place, you can use the <wbr> element to add word break opportunities.

<wbr> Tag Example

```
<p>To learn AJAX, you must be familiar with the
XML<wbr>Http<wbr>Request Object.
</p>
```

<input> Tag in HTML 5

HTML5 input tag introduced several new values for the **type** attribute. These are listed below:

- datetime
- datetime-local
- date
- month
- week
- time
- number
- range
- email
- url

Placeholder attribute in HTML 5

HTML5 introduced a new attribute called **placeholder**.

This attribute on <input> and <textarea> elements provides a hint to the user of what can be entered in the field. The placeholder text must not contain carriage returns or line-feeds.

Placeholder attribute Example

```
<input type="text" name="search" placeholder="search the web"/>
```

Autofocus attribute in HTML 5

This is a simple one-step pattern, easily programmed in JavaScript at the time of document load, automatically focus one particular form field.

Autofocus attribute Example

```
<input type="text" name="search" autofocus/>
```

Required attribute in HTML 5

Now you do not need to have javascript for client side validations like empty text box would never be submitted because HTML5 introduced a new attribute called .

Required attribute Example

```
<input type="text" name="search" required/>
```

SVG in HTML 5

SVG stands for **S**calable **V**ector **G**raphics and it is a language for describing 2D-graphics and graphical applications in XML and the XML is then rendered by an SVG viewer.

SVG is mostly useful for vector type diagrams like Pie charts, Two-dimensional graphs in an X,Y coordinate system etc.

HTML5 allows embeding SVG directly using **<svg>...</svg>** tag which has following simple syntax:

```
<svg xmlns="http://www.w3.org/2000/svg">
...
</svg>
```

WebSocket in HTML 5

Web Sockets is a next-generation bidirectional communication technology for web applications which operates over a single socket and is exposed via a JavaScript interface in HTML 5 compliant browsers.

Once you get a Web Socket connection with the web server, you can send data from browser to server by calling a **send()** method, and receive data from server to browser by an **onmessage** event handler.

Following is the API which creates a new WebSocket object.

```
var Socket = new WebSocket(url, [protocal] );
```

Here first argument, url, specifies the URL to which to connect. The second attribute, protocol is optional, and if present, specifies a sub-protocol that the server must support for the connection to be successful.