# What is HTML5?

- HTML5 is the newest version of HTML, only recently gaining partial support by the makers of web browsers.

- It incorporates all features from earlier versions of HTML, including the stricter XHTML.

- It adds a diverse set of new tools for the web developer to use.

- It is still a work in progress. No browsers have full HTML5 support. It will be many years – perhaps not until 2018 or later - before being fully defined and supported.

# Goals of HTML5

- Support all existing web pages. With HTML5, there is no requirement to go back and revise older websites.

- Reduce the need for external plugins and scripts to show website content.

- Improve the semantic definition (i.e. meaning and purpose) of page elements.

- Make the rendering of web content universal and independent of the device being used.

- Handle web documents errors in a better and more consistent fashion.

# Browsers support-HTML5

- Firefox, Safari, Chrome, Opera, and mobile browsers already support canvas, video, geolocation, local storage, and more. Google already supports microdata annotations.

- Even Microsoft — rarely known for blazing the trail of standards support — will be supporting most HTML5 features in the upcoming Internet Explorer 9.

# Other New Features in HTML5

- Canvas element for drawing
-  Video/audio elements for media playback
-  Better support for local offline storage
- New content specific elements, like article footer, header, nav, section
- New form controls, like calendar, date, time, email, url, search

# First Look at HTML5

Remember the DOCTYPE declaration from XHTML?

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

In HTML5, there is just one possible DOCTYPE declaration and it is simpler:

```
<!DOCTYPE html>
```

Just 15 characters!

The DOCTYPE tells the browser which type and version of document to expect. This should be the last time the DOCTYPE is ever changed. From now on, all future versions of HTML will use this same simplified declaration.

# The <html> Element

This is what the <html> element looked like in XHTML:

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
    lang="en">
```

Again, HTML5 simplifies this line:

```
<html lang="en">
```

The **lang** attribute in the <html> element declares which language the page content is in. Though not strictly required, it should always be specified, as it can assist search engines and screen readers.

Each of the world's major languages has a two-character code, e.g. Spanish = "es", French = "fr", German = "de", Chinese = "zh", Arabic = "ar".

# The <head> Section

Here is a typical XHTML <head> section:

```
<head>
  <meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
  <title>My First XHTML Page</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
```

And the HTML5 version:

```
<head>
  <meta charset="utf-8">
  <title>My First HTML5 Page</title>
  <link rel="stylesheet" href="style.css">
</head>
```

Notice the simplified character set declaration, the shorter CSS stylesheet link text, and the removal of the trailing slashes for these two lines.
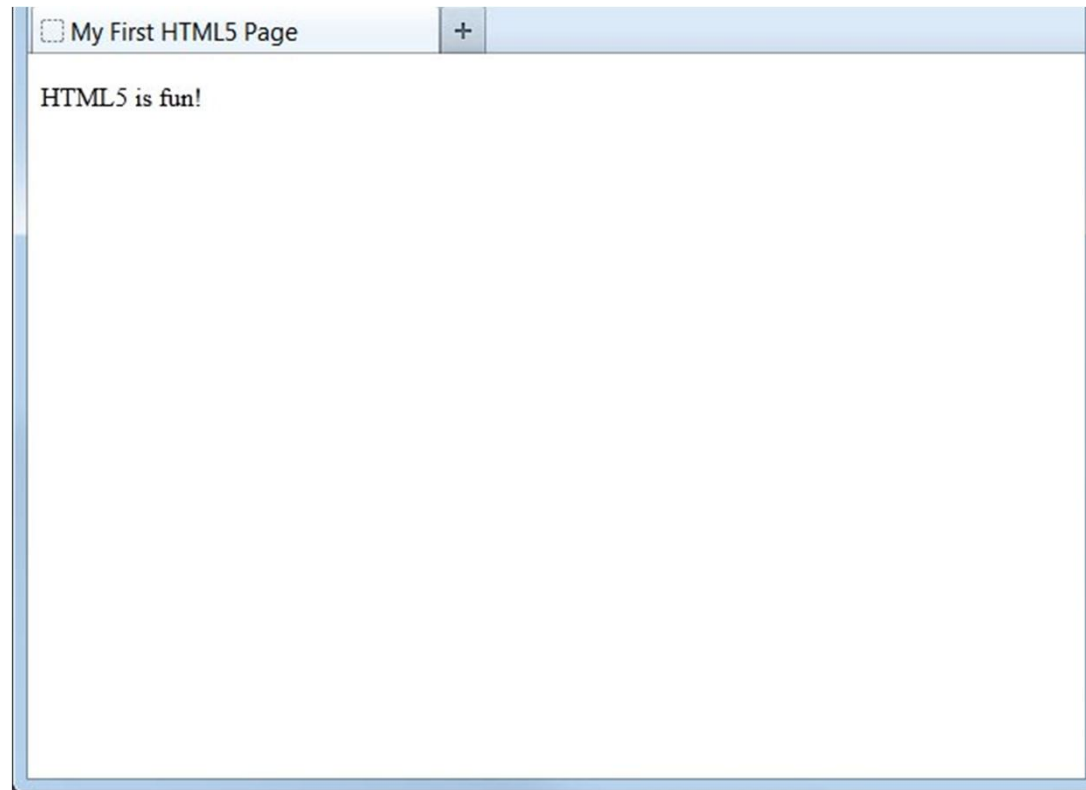
# Basic HTML5 Web Page

Putting the prior sections together, and now adding the <body> section and closing tags, we have our first complete web page in HTML5:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="utf-8">
   <title>My First HTML5 Page</title>
   <link rel="stylesheet" href="style.css">
</head>
<body>
   <p>HTML5 is fun!</p>
</body>
</html>
```

Let's open this page in a web browser to see how it looks…

# Viewing the HTML5 Web Page

My First HTML5 Page     +

HTML5 is fun!

Even though we used HTML5, the page looks exactly the same in a web browser as it would in XHTML. Without looking at the source code, web visitors will not know which version of HTML the page was created with.
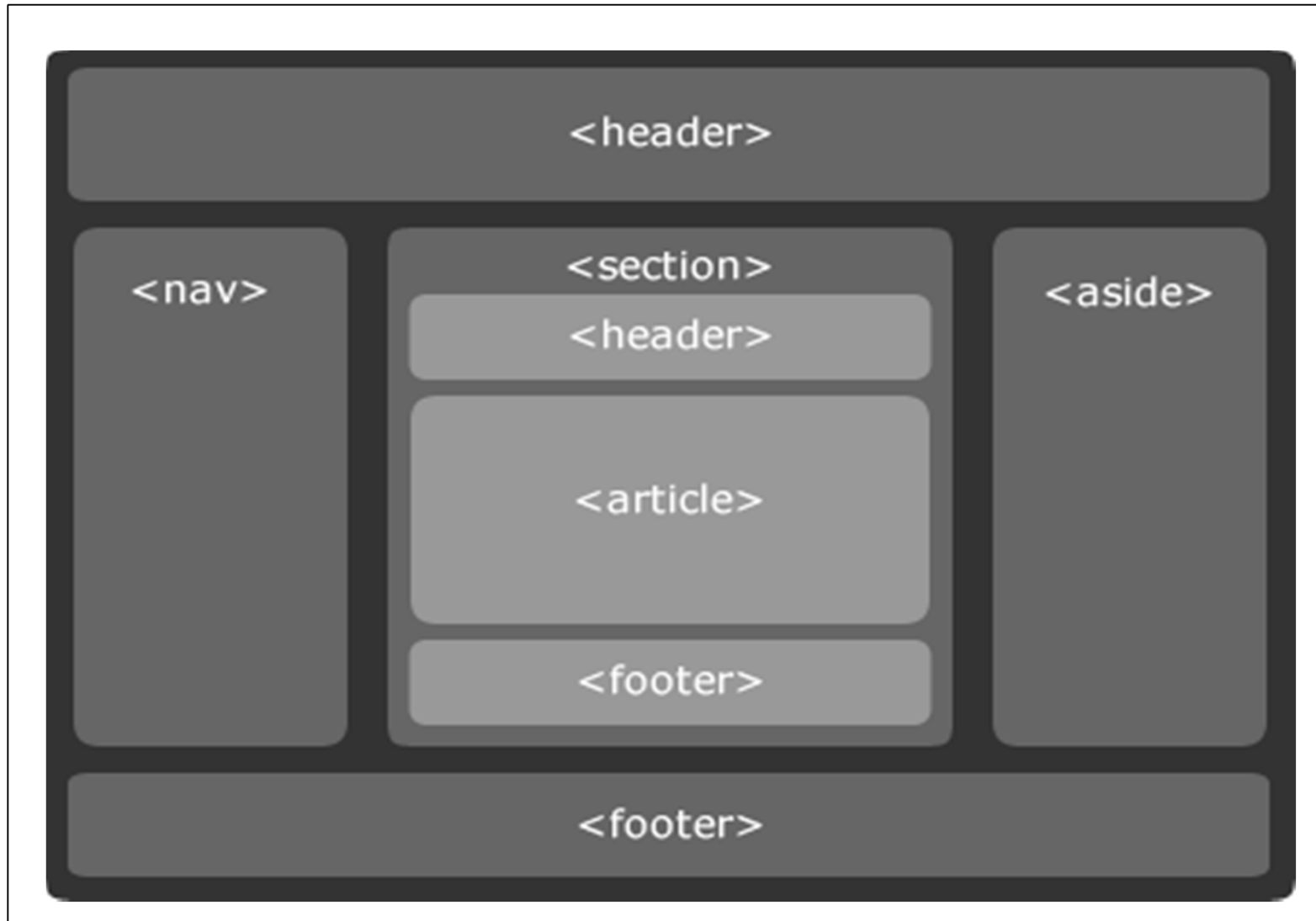
# Html4 Structure

# Html5 Structure

| div id = "header" |
|---|

| div id = "menu" | div class = "post" |
| | div class = "post" |

| div id = "footer" |
|---|

**VS**
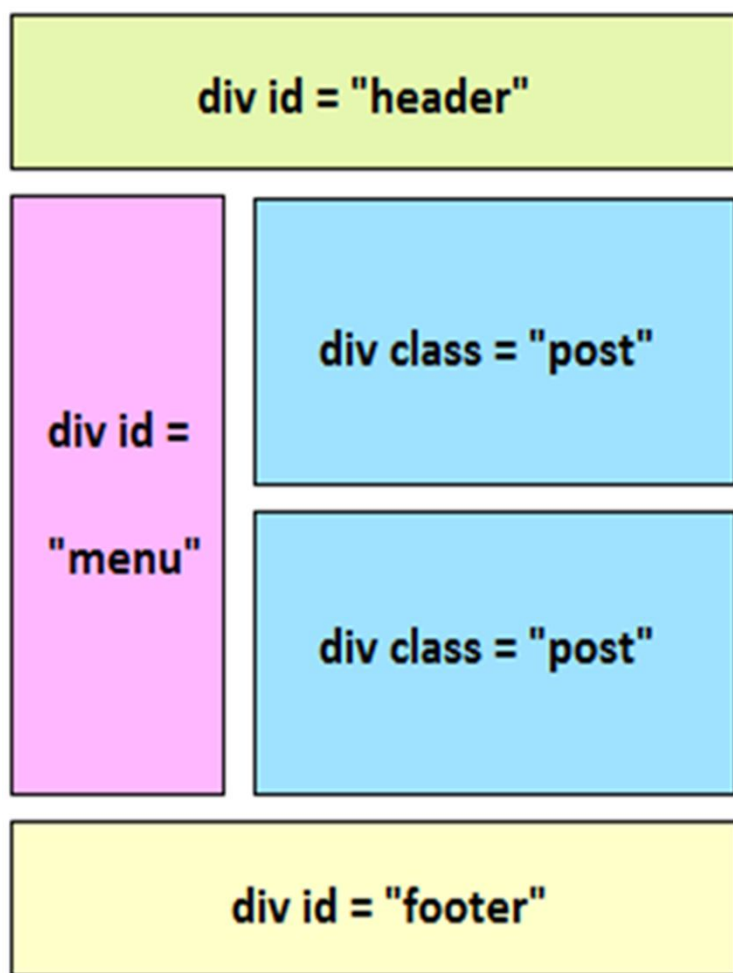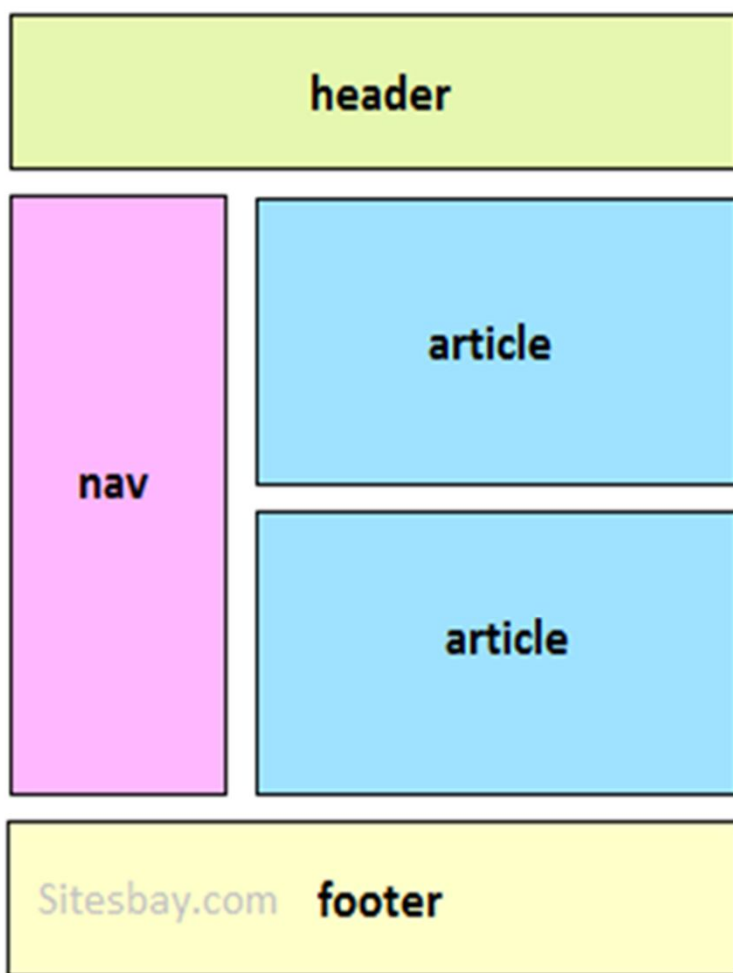
| header |
|---|

| nav | article |
| | article |

| footer |
|---|

| Html | Html5 |
|---|---|
| Doctype declaration in Html is too longer <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"> | DOCTYPE declaration in Html5 is very simple "<!DOCTYPE html> |
| character encoding in Html is also longer <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"> | character encoding (charset) declaration is also very simple <meta charset="UTF-8"> |
| Audio and Video are not part of HTML4 | Audio and Videos are integral part of HTML5 e.g. <audio> and <video> tags. |
| Vector Graphics is possible with the help of technologies such as VML, Silverlight, Flash etc | Vector graphics is integral part of HTML5 e.g. SVG and canvas |
| It is almost impossible to get true GeoLocation of user browsing any website especially if it comes to mobile devices. | JS GeoLocation API in HTML5 helps identify location of user browsing any website (provided user allows it) |
| Html4 use cookies. | It provides local storage in place of cookies. |
| Not possible to draw shapes like circle, rectangle, triangle. | Using Html5 you can draw shapes like circle, rectangle, triangle. |
| Does not allow JavaScript to run in browser. JS runs in same thread as browser interface. | Allows JavaScript to run in background. This is possible due to JS Web worker API in HTML5 |
| Works with all old browsers | Supported by all new browser. |

| Html5 Tags | Description |
| --- | --- |
| <header> | Defines a header for a document |
| <footer> | Defines a footer for a document |
| <article> | Defines an article in your document |
| <aside> | Defines content aside from the page content |
| <audio> | Allow to play audio on browser. |
| <canvas> | Used to draw Graphics on web. |
| <figure> | Specifies self-contained content |
| <meter> | Defines a scalar measurement within a known range |
| <nav> | Defines navigation links |
| <section> | Defines a section in a document |
| <source> | Defines multiple media resources for media elements (audio and video) |
| <video> | Defines a video or movie |
| <progress> | Represents the progress of a task |
| <mark> | Defines marked/highlighted text |
| <details> | Used to create an interactive widget that the user can open and close |
| <bdi> | Full form of BDI is Bi-Directional Isolation. This element is useful when embedding user-generated content with an unknown directionality. |
| <summary> | It specifies a visible heading for <detailed> element. |
| <wbr> | Used for possible line bread |
| <time> | Used for define date and time |
| <track> | Used for defines text tracks for media elements (<video> and <audio>) |
| <source> | Used for defines multiple media resources for media elements (<video> and <audio>) |
| <ruby> | Used for defines a ruby annotation |
| <output> | Used for represents the result of a calculation |

# HTML5: What's New

- <header>
- <nav>
- <article>
- <section>
- <aside>
- <footer>

**Other Tags**

- <canvas>
- <hgroup>
- <address>
- <figure>
- <meter>
- <progress>
- <time>

# New semantic Elements in HTML5

<article>       <figcaption>      <progress>
<aside>         <footer>          <section>
<audio>         <header>          <source>
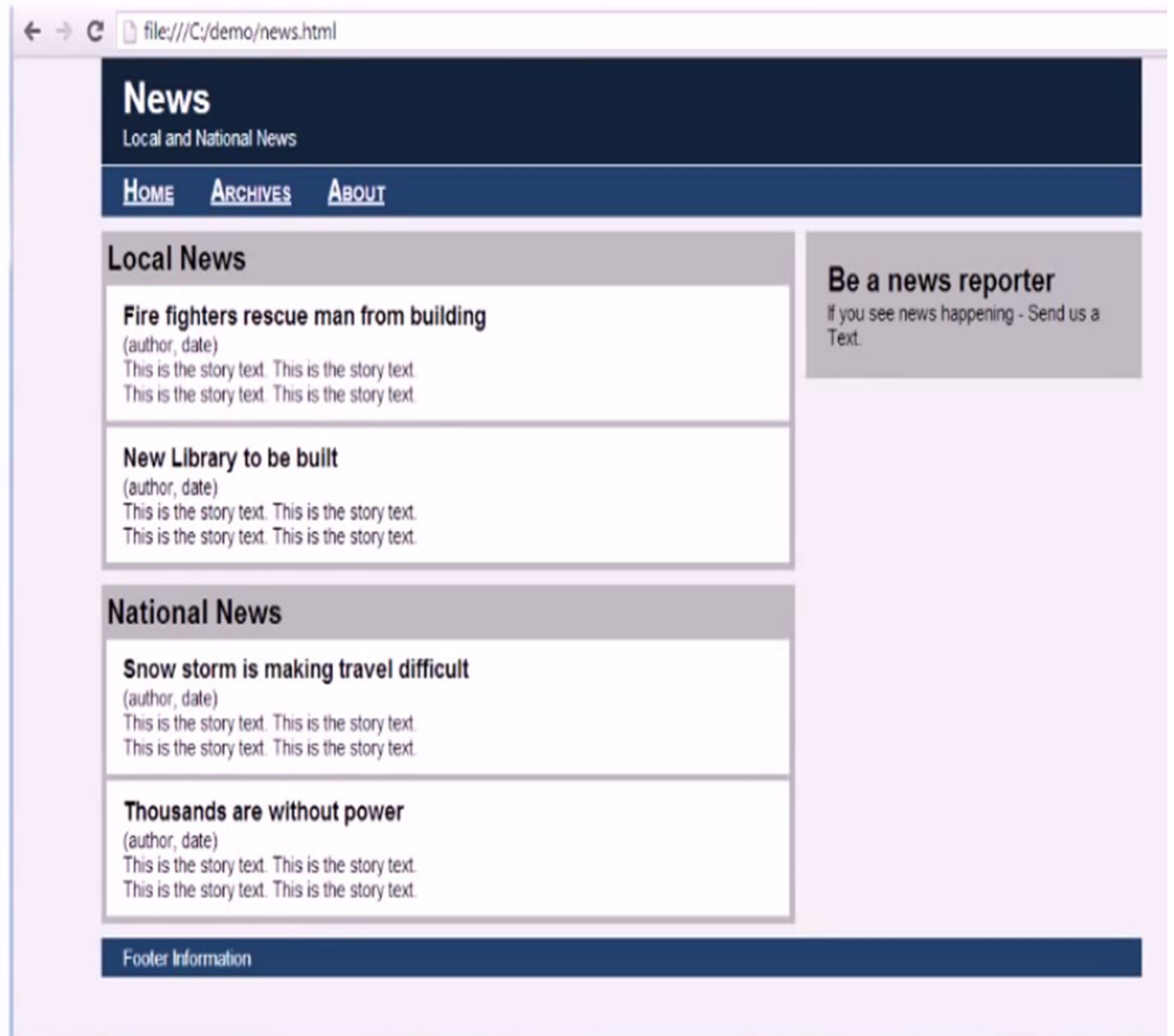<canvas>        <hgroup>          <svg>
<datalist>      <mark>            <time>
<figure>        <nav>             <video>

- **section** − This tag represents a generic document or application section. It can be used together with h1-h6 to indicate the document structure.

- **article** − This tag represents an independent piece of content of a document, such as a blog entry or newspaper article.

- **aside** − This tag represents a piece of content that is only slightly related to the rest of the page.

- **header** − This tag represents the header of a section.

- **footer** − This tag represents a footer for a section and can contain information about the author, copyright information, etc.

- **nav** − This tag represents a section of the document intended for navigation.

- **figure** − This tag can be used to associate a caption together with some embedded content, such as a graphic or video.

# Web Page Without CSS

file:///C:/demo/news.html

# News

Local and National News

- Home
- Archives
- About

## Local News

### Fire fighters rescue man from building

(author, date)

This is the story text. This is the story text.

This is the story text. This is the story text.

### New Library to be built

(author, date)

This is the story text. This is the story text.

This is the story text. This is the story text.

## National News

### Snow storm is making travel difficult

(author, date)

# Web Page With CSS

file:///C:/demo/news.html

# News
Local and National News

**HOME    ARCHIVES    ABOUT**

## Local News

**Fire fighters rescue man from building**
(author, date)
This is the story text. This is the story text.
This is the story text. This is the story text.

**New Library to be built**
(author, date)
This is the story text. This is the story text.
This is the story text. This is the story text.

## National News

**Snow storm is making travel difficult**
(author, date)
This is the story text. This is the story text.
This is the story text. This is the story text.

**Thousands are without power**
(author, date)
This is the story text. This is the story text.
This is the story text. This is the story text.

**Be a news reporter**
If you see news happening - Send us a Text.

Footer Information

# News

Local and National News

- Home
- Archives
- About

## Local News

### Fire fighters rescue man from building

(author, date)

This is the story text. This is the story text.

This is the story text. This is the story text.

### New Library to be built

(author, date)

This is the story text. This is the story text.

This is the story text. This is the story text.

## National News

### Snow storm is making travel difficult

(author, date)

# <header>

- The HTML <header> tag represents a group of introductory or navigational aids.
- Headers can contain headings, subheadings, version information, navigational controls, etc.
- No attributes

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <title>News</title>
 </head>
 <body>
  <header>
   <h1>News</h1>
   <p>Local and National News</p>
  </header>
```

# \<nav\>

- The \<nav\> element defines a set of navigation links.

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8">
  <title>News</title>
 </head>
 <body>
  <header>
   <h1>News</h1>
   <p>Local and National News</p>
  </header>
<nav>
   <ul>
     <li><a href="index.html">Home</a></li>
     <li><a href="archive.html">Archives</a></li>
     <li><a href="about.html">About</a></li>
   </ul>
  </nav>
```

# \<main>

- \<main> tag is used to represent the main content area within an HTML document.
- \<main> tag surrounds the main content of the page - content that is unique to that document and is obviously the "main" content for that page. This excludes any content that is repeated across multiple pages (such as navigation bars, headers, footers, etc).
- An HTML document can have a maximum of one \<main> element. It must not appear within the [\<article>](), [\<aside>](), [\<footer>](), [\<header>]() or [\<nav>]() tags.

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>News</title>
  </head>
  <body>
    <header>
      <h1>News</h1>
      <p>Local and National News</p>
    </header>
<nav>
    <ul>
      <li><a href="index.html">Home</a></li>
      <li><a href="archive.html">Archives</a></li>
      <li><a href="about.html">About</a></li>
    </ul>
   </nav>
<main>
<p>This is the main content of the web page</p>
</main>
```

# &lt;section&gt;

- &lt;section&gt; tag is used to represent a section within an article.
- Any given web page or article could have many sections.
- For example, a homepage could have a section for introducing the company, another section for news items, and another section for contact information.

```
<main>
        <section>
                <h2>Local News</h2>
        </section>
         <section>
                <h2>National News</h2>

        </section>
    </main>
```

# <article>

- <article> tag is used to represent an article.
- The content within the <article> tag is independent from the other content on the site (even though it could be related).
- By "independent" means that its contents could stand alone, for example in syndication.

```
<section>
  <h2>Local News</h2>
    <article>
      <header>
        <h3>Fire fighters rescue man from
building</h3>
        <p>(author, date)</p>
      </header>
      <p>This is the story text. This is the story
text.</p>
      <p>This is the story text. This is the story
text.</p>
    </article>
```

```
<article>
    <header>
      <h3>New Library to be built</h3>
      <p>(author, date)</p>
    </header>
    <p>This is the story text. This is the
story text.</p>
    <p>This is the story text. This is the
story text.</p>
  </article>
</section>
```

```html
<section>
    <h2>National News</h2>
    <article>
      <header>
        <h3>Snow storm is making travel difficult</h3>
        <p>(author, date)</p>
      </header>
      <p>This is the story text. This is the story text.</p>
      <p>This is the story text. This is the story text.</p>
    </article>

    <article>
      <header>
        <h3>Thousands are without power</h3>
        <p>(author, date)</p>
      </header>
      <p>This is the story text. This is the story text.</p>
      <p>This is the story text. This is the story text.</p>
    </article>
  </section>
```

# <aside>

- The HTML <aside> tag is used to represent content that is related to the surrounding content within an article or web page, but could still stand alone in its own right.
- This type of content is often represented in sidebars.

```
<aside>
    <h2>Be a news reporter</h2>
    <p>If you see news happening - Send us a
Text.</p>
    </aside>
```

# <footer>

- <footer> tag is used for defining the footer of an HTML document or section.
- Footers usually contain information such as the author of the document, copyright information, links to terms of use, privacy policy, etc.
- Contact information within a <footer> tag should be marked up using the <address> tag
- Although footers are typically located at the bottom of a document, this is not required (although it cannot be placed within a <header> or another <footer> element, and it cannot contain a <header> element).
- A document/section can have more than one footer.

```html
<!DOCTYPE html>
<html lang="en">
 <head>
   <meta charset="UTF-8">
   <title>News</title>
 </head> <body>
   <header>
     <h1>News</h1>
     <p>Local and National News</p>
   </header>
<nav>
     <ul> <li><a href="index.html">Home</a></li>
      <li><a href="archive.html">Archives</a></li>
      <li><a href="about.html">About</a></li>
     </ul>
   </nav>
<main>
<p>This is the main content of the web page</p>
</main>
<footer>
     <p>Footer Information</p>
   </footer></body></html>
```

```html
<html>
<head></head>
<body>
<header>
<h1>Talking Dogs</h1>
<b><p>Humans aren't the only talkers!</p></b>
</header>
<nav></nav>
<main>
<article>
<p>Ever encountered a talking dog? I have.</p>
<p>It all happened one day as I was walking down the street...</p>
</article>
</main>
<footer>
© 2009 Woofer Dog Corporation
</footer>
</body>
</html>
```

# HTML5: Video

Until now, there hasn't been a standard for showing video on a web page.

Today, most videos are shown through a plugin (like Flash). However, not all browsers have the same plugins.

HTML5 specifies a standard way to include video with the video element.

# HTML5: Video

Currently, there are 3 supported video formats for the video element:

| Browser | MP4 | WebM | Ogg |
|---|---|---|---|
| Internet Explorer | YES | NO | NO |
| Chrome | YES | YES | YES |
| Firefox | YES<br>from Firefox 21<br>from Firefox 30 for Linux | YES | YES |
| Safari | YES | NO | NO |
| Opera | YES<br>From Opera 25 | YES | YES |

# HTML5: Video

The last example uses an Ogg file, and will work in Firefox, Opera and Chrome.

To make the video work in Safari and future versions of Chrome, we must add an MPEG4 and WebM file.

The video element allows multiple source elements. Source elements can link to different video files. The browser will use the first recognized format:

# HTML5: Video

| Attribute | Value | Description |
|-----------|-------|-------------|
| audio | muted | Defining the default state of the the audio. Currently, only "muted" is allowed |
| autoplay | autoplay | If present, then the video will start playing as soon as it is ready |
| controls | controls | If present, controls will be displayed, such as a play button |
| height | *pixels* | Sets the height of the video player |
| loop | loop | If present, the video will start over again, every time it is finished |
| poster | *url* | Specifies the URL of an image representing the video |
| preload | preload | If present, the video will be loaded at page load, and ready to run. Ignored if "autoplay" is present |
| src | *url* | The URL of the video to play |
| width | *pixels* | Sets the width of the video player |

# HTML5: Video

```
<!DOCTYPE HTML>
<html>
<body>

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

</body>
</html>
```

# &lt;source&gt; tag

- &lt;source&gt; tag is used to specify multiple media resources for media elements, such as &lt;video&gt; and &lt;audio&gt;.
- unpaired tag

# <source> tag attributes

- src->Specifies the URL of the media file

- media->Specifies the type of media resource

- type->Specifies the media type of the media resource

# examples

```
<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>
```

# HTML5: Audio

Until now, there has never been a standard for playing audio on a web page.

Today, most audio is played through a plugin (like Flash). However, not all browsers have the same plugins.

HTML5 specifies a standard way to include audio, with the audio element. The audio element can play sound files, or an audio stream.

# HTML5: Audio

Currently, there are 3 supported formats for the audio element:

| 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 |

# HTML5: Audio

| Attribute | Value | Description |
| --- | --- | --- |
| autoplay | autoplay | Specifies that the audio will start playing as soon as it is ready. |
| controls | controls | Specifies that controls will be displayed, such as a play button. |
| loop | loop | Specifies that the audio will start playing again (looping) when it reaches the end |
| preload | preload | Specifies that the audio will be loaded at page load, and ready to run. Ignored if autoplay is present. |
| src | *url* | Specifies the URL of the audio to play |

# HTML5: Audio

```
<!DOCTYPE HTML>
<html>
<body>

<audio src="song.ogg" controls="controls">
Your browser does not support the audio element.
</audio>

</body>
</html>
```

# HTML5: Audio

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

# \<figure\> & \<figcaption\>

- \<figure\> tag is used for annotating illustrations, diagrams, photos, code listings, etc.
- \<figure\> element to associate a caption together with some embedded content, such as a graphic or video.
- use the \<figure\> element in conjunction with the [\<figcaption\>]() element to provide a caption for the contents of your \<figure\> element.

```
<p><a href="#1">Figure 1</a> provides the JavaScript code for
creating an alert box:</p>
<figure id="1">
<figcaption>Figure 1. JavaScript Alert Box.</figcaption>
<pre><code>alert('Hello!');</code></pre>
</figure>
```

# <meter>

- <meter> tag is used for indicating a scalar measurement within a known range, or a fractional value.
- Also known as a *gauge*, usage could include displaying disk usage, the amount raised during fundraising activities, or the relevance of a search query result.
- It is important to note that the <meter> element is used to represent a *range*.

| Attributes | value | Description |
|---|---|---|
| form | form_id | Specifies one or more forms the <meter> element belongs to |
| low | number | Specifies the range that is considered to be a low value |
| high | number | Specifies the range that is considered to be a high 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 |

```html
<ol>
<li><meter min="0" max="100" value="25">25%</meter></li>
<li><meter min="100" max="200"
value="150">50%</meter></li>
<li><meter min="0" max="100" value="75">75%</meter></li>
<li><meter min="0" max="800" value="400">50%</meter></li>
</ol>
<dl>
<dt>Width:</dt>
<dd><meter min="0" max="200" value="123"
title="millimeters">123mm</meter></dd>
<dt>Height:</dt>
<dd><meter min="0" max="100" value="25" optimum="30"
title="millimeters">25mm</meter></dd>
</dl>
```

# <progress>

- <progress> tag is used for representing the progress of a task.
- This element could be used in conjunction with JavaScript to display the progress of a task or process as it is underway.

| 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 |

```
<!DOCTYPE html>
<html>
<body>

Downloading progress:
<progress value="22" max="100">
</progress>

<p><strong>Note:</strong> The progress tag is not
supported in Internet Explorer 9 and earlier
versions.</p>

</body>
</html>
```

# \<dialog>

- 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.

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

```html
<div>
<dialog id="myFirstDialog" style="width:50%;background-color:#F4FFEF;border:1px
dotted black;">
<p><q>The world is full of magical things patiently waiting for our wits to grow
sharper.</q> - <cite>Bertrand Russell</cite></p>
<button id="hide">Close</button>
</dialog>
<!-- "Show" button -->
<button id="show">Show Dialog</button>
</div>

<!-- JavaScript to provide the "Show/Close" functionality -->
<script type="text/JavaScript">
(function() {
  var dialog = document.getElementById('myFirstDialog');
  document.getElementById('show').onclick = function() {
    dialog.show();
  };
  document.getElementById('hide').onclick = function() {
    dialog.close();
  };
})();
</script>
```

# <mark>

- <mark> tag is used for indicating text as marked or highlighted for reference purposes, due to its relevance in another context.

```
<!DOCTYPE html>
<title>My Example</title>

<p>Despite the stockmarket crash in 2008, the
value of my share portfolio <mark>increased by
100 percent</mark>. I must be doing something
right.</p>
<p>I should point out that the value only increased
because I kept pumping more money in!</p>
```

# &lt;details&gt; &amp; &lt;summary&gt;

- The HTML &lt;details&gt; tag specifies additional details that the user can view or hide on demand.
- used in conjunction with the HTML5 &lt;summary&gt; tag to provide a heading that can be clicked on to expand/collapse the details as required.

```
<!DOCTYPE html>
<html>
<body>

<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>

<p><b>Note:</b> The summary element is not supported in
Edge/Internet Explorer.</p>

</body>
</html>
```

# &lt;wbr&gt;

- The &lt;wbr&gt; (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 &lt;wbr&gt; element to add word break opportunities.

```
<!DOCTYPE html>
<html><body>
<p>Try to shrink the browser window, to view how the very long word in  the
paragraph below will break:</p>
<p>This is a
veryveryveryveryveryveryveryveryveryveryveryveryveryveryveryveryveryver
y<wbr>longwordthatwillbreakatspecific<wbr>placeswhenthebrowserwindo
wisresized.</p>
<p><b>Note:</b> The wbr element is not supported in Internet Explorer 11
and
earlier versions.</p>
</body>
</html>
```

# HTML5: Canvas

The HTML5 canvas element uses JavaScript to draw graphics on a web page.

A canvas is a rectangular area, and you control every pixel of it.

The canvas element has several methods for drawing paths, boxes, circles, characters, and adding images.

# HTML5: Canvas

Adding a canvas element to the HTML5 page.

Specify the id, width, height of the element:

```
<canvas id="myCanvas" width="200" height="100"></canvas>
```

# HTML5: Canvas

The canvas element has no drawing abilities of its own.
All drawing must be done inside a JavaScript:

```
<script type="text/javascript">
var c=document.getElementById("myCanvas");
var cxt=c.getContext("2d");
cxt.fillStyle="#FF0000";
cxt.fillRect(0,0,150,75);
</script>
```