Chapter 1: Introduction to HTML

1.1 Introduction to HTML

To publish information for global distribution i.e. Internet, one need a universally understood language. The publishing language used by WWW is HTML.

Hypertext Markup Language (**HTML**) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

It is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995.

HTML Stands for **Hypertext markup language**.

A markup language is simply a way of annotating a document in such a way as to make the annotations distinct from the text being annotated. Markup languages such as **HTML, Tex, XML,** and **XHTML** allow users to control how text and visual elements will be laid out and displayed.

Hypertext is text which is not constrained to be linear. It is a text which contains links to other texts. Hypermedia is a term used for hypertext which is not constrained to be text. It can include graphics, video and sound.

Web browsers evolved into quite permissive and lenient programs. They could handle sloppy HTML, missing or malformed tags, and other syntax errors. However, it was somewhat unpredictable how each browser would handle such errors.

XHTML

XHTML stands for Extensible Hyper Text Markup Language. It is identical to HTML that uses stricter XML (extensible markup language) syntax rules. The goal of XHTML with its strict rules was to make page rendering more predictable by forcing web authors to create web pages without syntax errors.
Two versions of XHTML were created:

XHTML 1.0 Strict

XHTML 1.0 Transitional.

HTML validators provide the means of verifying that web pages follow the rules for XHTML Transitional or Strict. The strict version was meant to be rendered

XML

XML is a textual markup language. Also like HTML, the formal rules for XML were set by the W3C. XML is a more general markup language than HTML. It is (and has been) used to mark up any type of data. XML-based data formats (called schemas in XML) are almost everywhere. For instance, Microsoft Office products.

HTML5

At around the same time the XHTML 2.0 specification was being developed, a group of developers at Opera and Mozilla formed the **WHATWG** (Web Hypertext Application Technology Working Group) group within the W3C.

The work at WHATWG progressed quickly, and eventually, by 2009, the W3C stopped work on XHTML 2.0 and instead adopted the work done by WHATWG and named it HTML5.

There are three main aims to HTML5:

1. Specify unambiguously how browsers should deal with invalid markup.
2. Provide an open, nonproprietary programming framework (via JavaScript) for creating rich web applications.
3. Be backwards compatible with the existing web.

HTML vs HTML5

HTML5 was developed by a group known as WHATWG and was designed to improve upon the previous HTML versions and solve some cross-browser compatibility issues. The key differences between HTML and HTML5:

- HTML5 offers better support for various forms of media, such as audio and video, by providing tags for them. HTML didn't have these tags and relied on third-party plug-in.
- HTML doesn't allow JavaScript to run in the browser. HTML5 solved this problem by introducing the JS Web Worker API. Now, because of the native support of JavaScript, web pages can be better designed by using front-end scripts to enhance the user experience.
- HTML5 includes new input attributes, such as email, URLs, date and time, search, etc.
- HTML5 is device independent and has better browser compatibility than previous HTML versions.
- HTML5 has better parsing (parsing is the processing of text) rules and allows for greater flexibility in parsing than HTML.
- HTML5 also makes it easy to find the location without any third-party plugins.
- HTML5 also has native support for vector graphics, so, there is no need of third-party software such as Adobe Flash.

In short, HTML5 is a better version of HTML with added features and functionalities.

HTML vs XHTML

XHTML was developed as an extension to HTML. There aren't many differences between HTML4 and XHTML, and XHTML is basically a stricter version of HTML4. The main differences between HTML and XHTML are:

- HTML4 allows some elements to omit the end tag. End tags are added when closing a certain part of the text, such as a paragraph. They are usually symbolized with a backslash (for example, opening tag of a paragraph is while end tag is). XHTML requires that all elements include the end tag.
- HTML4 allows overlapping of certain elements. XHTML doesn't allow any elements to overlap.
- Attribute values (such as font size) have to be quoted in XHTML, even if they are numeric. HTML doesn't have include quoted values for attributes.
- Attributes cannot be minimized in HTML.
- There is a small difference in the way empty elements are handled.

While the above are the more important differences, there are also some very subtle differences, but they are really rare to run into. You can always check out the documentation of XHTML for more information. The takeaway is that XHTML was designed to solve some problems in HTML, by incorporating some features of XML.

HTML5 vs XHTML

Since XHTML and HTML are largely the same, the differences between XHTML and HTML5 are the same as the ones between HTML4 and HTML5. However, the following are some of the subtler differences between HTML5 and XHTML:

- While XHTML is case sensitive, HTML5 is not (HTML is also case insensitive).
- HTML5 has a much simpler doctype than XHTML and HTML (Doctypes tell the browser how to interpret the data).
- HTML5 is compatible with all browsers while XHTML isn't.
- HTML5 is more lenient, following the footsteps of HTML4, than XHTML.
- HTML5 is better suited for mobile devices such as tablets and phones, while XHTML is better suited for computer screens.

HTML Syntax

Elements and Attributes

HTML documents are composed of textual content and HTML elements. HTML tags are element names surrounded by angle brackets:

```
<tagname>content goes here...</tagname>
```

- HTML tags normally come **in pairs** like and
- The first tag in a pair is the **opening tag,** the second tag is **closing tag**
- The end tag is written like the start tag, but with a **forward slash** inserted before the tag name.

HTML elements can also contain **attributes**. An HTML attribute is a name=value pair that provides more information about the HTML element. In XHTML, attribute values had to be enclosed in quotes; in HTML5, the quotes are optional.

- Open tag - on switch eg: <a>
- Close tag- off switch eg:

An **empty HTML element** does not contain any text content, instead it is an instruction to the browser to do something. Perhaps the most common empty element is <img>, the image element. In XHTML, empty elements had to be terminated by a trailing slash (as shown in Figure 2.4). In HTML5, the trailing slash in empty elements is optional.

FIGURE 2.4 The parts of an HTML element

- **Semantic Markup:**

Semantic markup is used to reinforce the semantics (meaning) of information in WebPages/web application rather than to define presentation or look.

Semantics is the study of meanings of words and phrases in a language.

Semantic elements= Elements with a meaning

Ex: Non semantic elements: , etc... – **tells nothing about the content**
Semantic elements : <form>, <table>, <article> etc… **-clearly defines the content**

Eliminating presentation-oriented markup and writing semantic HTML markup has a variety of important advantages:

- **Maintainability**- Semantic markup is easier to update and change than web pages that contain a great deal of presentation markup.
- **Faster-** Semantic web pages are typically quicker to author and faster to download.
- **Accessibility**-Not all web users are able to view the content on web pages. Users with sight disabilities experience the web using voice reading software. Visiting a web page using voice reading software can be a very frustrating experience if the site does not use semantic markup. As well, many governments insist that sites for organizations that receive federal government funding must adhere to certain accessibility guidelines.
- **Search engine optimization**- For many site owners, the most important users of a website are the various search engine crawlers. These crawlers are automated programs that cross the web scanning sites for their content, which is then used for users' search queries. Semantic markup provides better instructions for these crawlers: it tells them what things are important content on the site.

Structure of HTML Documents

Basic structure of any HTML document consists of following section or elements:

- DTD (Document Type Definition)
- Main Container (<html>)
- Head Section
- Body Section

HTML Page Structure

FIGURE 2.9 Structure elements of an HTML5 document

- The <!DOCTYPE html> declaration defines this document to be HTML5
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the document
- The <title> element specifies a title for the document
- The <body> element contains the visible page content
- The <h1> element defines a large heading
- The <p> element defines a paragraph

DOCTYPE

The DOCTYPE (**Document Type Definition**) element, which tells the browser (or any other client software that is reading this HTML document) what type of document it is about to process (it does not indicate what version of HTML) It must only appear once, at the top of the page (before any HTML tags). The <!DOCTYPE> declaration is not case sensitive.

The <!DOCTYPE> declaration for HTML5 is:

<!DOCTYPE html>

The standard doctype specifications for XHTML:

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

Head and Body

The **<html> element** is sometimes called the **root element** as it contains all the other HTML elements in the document.

HTML pages are divided into two sections: the **head** and the **body**, which correspond to the **<head>** and **<body>** elements.

The head contains descriptive elements *about* the document, such as its title, any style sheets or JavaScript files it uses, and other types of Meta information used by search engines and other programs. Body contains the visible page content.

Tags inside the head section

There are several tags that go inside the head section of an HTML document.

1. **The <base> tag:** The **<base>** tag acts as a reference point for all the links on a page. This reference point is specified with the **<base>** tag's **href** attribute.

   Example:<base href="http://www.landofcode.com />

   If absolute URL is http://www.somewebsite.com/page1.php. On this page, if there are seven images displayed and their absolute location is http://www.somewebsite.com/images/.

   When you try to display these images on your page, you can do so by specifying their location as in , , and so on, OR you can use the <base> tag to specify a reference point for all the images to simplify displaying them on your page. The <base> tag to be used in this situation would look like this:

   <base href="http://www.somewebsite.com/images" />

Now whenever you need to display an image on your page, you can do so by specifying the name of the image by its name alone, as in , , and so on, and the browser will automatically look for it in http://www.somewebsite.com/images/.

2. **The <link> tag :**The **<link>** tag defines a relationship between two documents. It is used to define the relationship between the document where it is located and another document. A common use of the **<link>** tag is to call an external style sheet:

   <link rel="stylesheet" type="text/css" href="/css/style.css" />

3. **The <meta> tag:**The **<meta>** tag is one of the most important tags in HTML. The **<meta>** tag is used to describe the page in some way as well as other things such as refreshing a page automatically after a certain amount of time, and preventing WebPages from being displayed in another websites frames page. Some of the things that can be described using the **<meta>** tag include the pages author, the software used to create the page, and a description of the content on the page.

   Example: <meta name="keywords" content="computers, electronics, cameras" />

4. **The <script> tag:** The **<script>** tag is used to place scripts on a webpage. When using the **<script>** tag, you have to use its **type** attribute to specify the language the script is written in.

Example: <script type="text/javascript"> document.write("This is a Javascript script."); </script>

5. **The <style> tag :**The **<style>** tag is used to declare an internal stylesheet. The **<style>** tag specifies the content type of a stylesheet with its *type* attribute which should be set to "text/css".

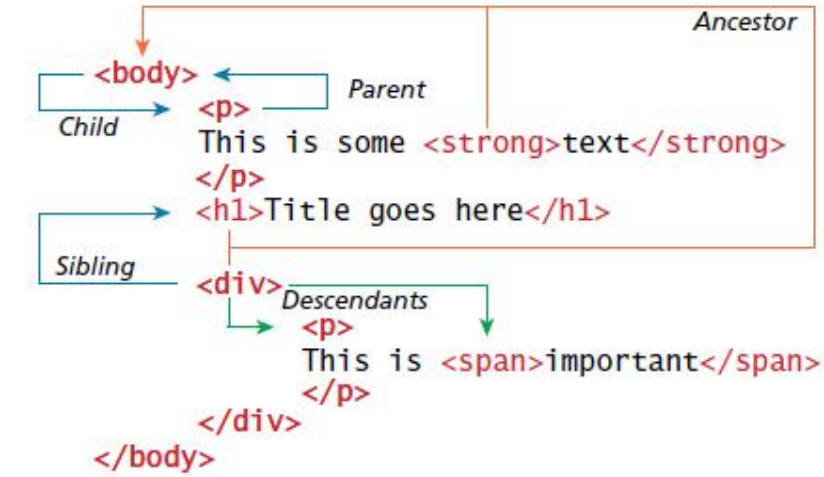
Example: <style type="text/css"> h1 { color: #000080; } </style>

6. **The <title> tag :** The **<title>** tag declares the title for a webpage. The title of a webpage can be seen in the top left corner of the web browser. If you look at the top left corner of your web browser window right now, you will see the title **HTML head section**, as that is the title of this page.

Example: <title>Introduction to HTML</title>

Nesting HTML Elements

Often an HTML element will contain other HTML elements. In such a case, the container element is said to be a **parent** of the contained, or **child**, element. Any elements contained within the child are said to be **descendants** of the parent element; likewise, any given child element, may have a variety o f **ancestors**.



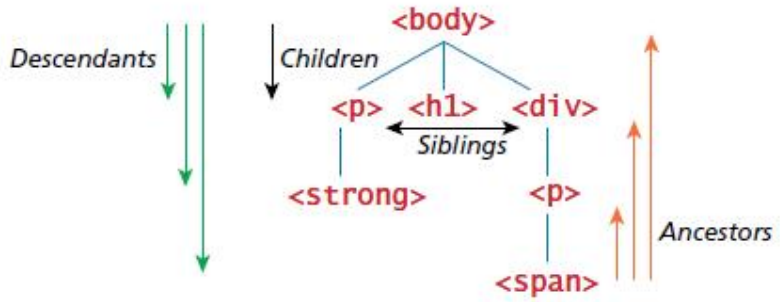


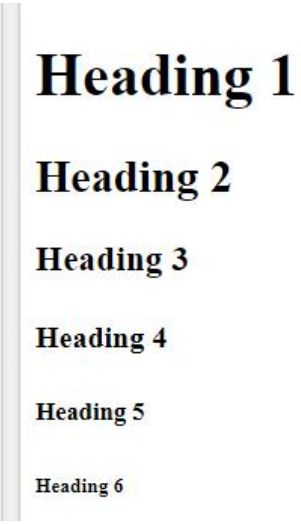
FIGURE 2.5 HTML document outline

- ## Quick Tour of HTML Elements

1. HTML Headings

HTML headings are defined with the <h1> to <h6> tags. <h1> defines the most important heading. <h6> defines the least important heading. Search engines use the headings to index the structure and content of your web pages. Users often skim a page by its headings. It is important to use headings to show the document structure.
Ex:

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

<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>

</body>
</html>
```



2. Paragraphs and Divisions

Paragraphs, the most basic unit of text in an HTML document. <p> tag is a container and can contain HTML and other inline HTML elements like and <a> elements.

```
<p>This is a paragraph.</p>
<p>This is another with break <br> paragraph.</p>
```

 element is also a container element and is used to create a logical grouping of content (text and other HTML elements, including containers such as <p> and other elements).

```
<p>By Ricardo on <time>September 15, 2015</time></p>
<p>Easy on the HDR buddy.</p>
</div>
```

Ex:

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

<p>
This paragraph
contains a lot of lines
</p>

<p>
This paragraph
contains      a lot of spaces
</p>

<p>
The number of lines in a paragraph depends on the size
of the browser window.
</p>

</body>
</html>
```

This paragraph contains a lot of lines

This paragraph contains a lot of spaces

The number of lines in a paragraph depends on the size of the browser window.

EX2:

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

<p>This is<br>a paragraph<br>with line breaks</p>

</body>
</html>
```

This is
a paragraph
with line breaks

The HTML <pre> Element

The HTML <pre> element defines preformatted text. The text inside a <pre> element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks:

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

<p>The pre tag preserves both spaces and line breaks:
</p>

<pre>
   My Bonnie lies over the ocean.

   My Bonnie lies over the sea.

   My Bonnie lies over the ocean.

   Oh, bring back my Bonnie to me.
</pre>

</body>
</html>
```

The pre tag preserves both spaces and line breaks:

```
   My Bonnie lies over the ocean.

   My Bonnie lies over the sea.

   My Bonnie lies over the ocean.

   Oh, bring back my Bonnie to me.
```

3. Links

HTML links are hyperlinks. You can click on a link and jump to another document. Links are created using the <a> element (the "a" stands for anchor). A link has two main parts: the destination and the label. As can be seen in Figure 2.15, the label of a link can be text or another HTML element such as an image. You can use the anchor element to create a wide range of links. These include:

- Links to external sites (or to individual resources such as images or movies on an external site).
- Links to other pages or resources within the current site.
- Links to other places within the current page.
- Links to particular locations on another page (whether on the same site or on an external site).
- Links that are instructions to the browser to start the users email program.
- Links that are instructions to the browser to execute a JavaScript function.
- Links that are instructions to the mobile browser to make a phone call.

The href Attribute : HTML links are defined with the <a> tag. The link address is specified in the href attribute:

```
<a href="https://www.w3schools.com">This is a link</a>
```

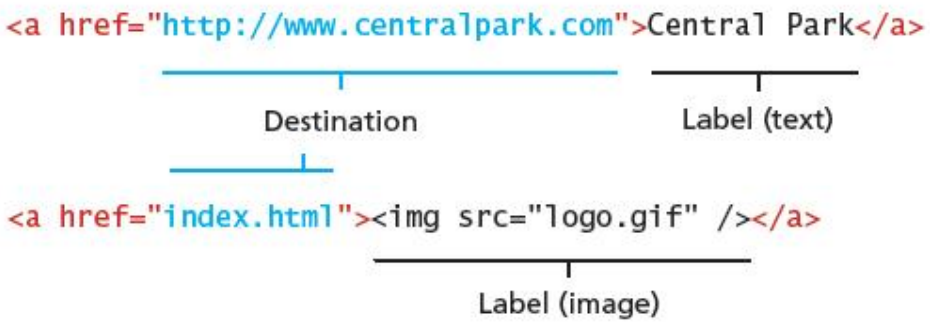


FIGURE 2.15 Two parts of a link

Example:

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

<h2>HTML Links</h2>
<p><a href="http://www.google.com">Google</a></p>

</body>
</html>
```

HTML Links

Google

URL Relative Referencing:

A file path describes the location of a file in a websites folder structure. They are used when linking to external files.

Absolute file path: It is the full URL to an internet file i.e Domain Name, Paths and finally the file name of the desired resource.

Ex: http://www.centralpark.com/images/picture.jpg"

Relative filepath: It points to file relative to current page. It refers to same server of the HTML document. If URL does not include "https:// " , then the browser will request the current server for file. Table below gives sample relative referencing.

Relative Link Type	Example
1 Same Directory To link to a file within the same folder, simply use the file name.	To link to example.html from about.html (in Figure 2.17), use: `<a href="example.html">`
2 Child Directory To link to a file within a subdirectory, use the name of the subdirectory and a slash before the file name.	To link to logo.gif from about.html, use: `<a href="images/logo.gif">`
3 Grandchild/Descendant Directory To link to a file that is multiple subdirectories *below* the current one, construct the full path by including each subdirectory name (separated by slashes) before the file name.	To link to background.gif from about.html, use: `<a href="css/images/background.gif">`

4 Parent/Ancestor Directory
Use "../" to reference a folder *above* the current one. If trying to reference a file several levels above the current one, simply string together multiple "../".
	To link to about.html from index.html in members, use:
	`<a href="../about.html">`
	To link to about.html from bio.html, use:
	`<a href="../../about.html">`

5 Sibling Directory
Use "../" to move up to the appropriate level, and then use the same technique as for child or grandchild directories.
	To link to about.html from index.html in members, use:
	`<a href="../images/about.html">`
	To link to background.gif from bio.html, use:
	`<a href="../../css/images/background.gif">`

6 Root Reference
An alternative approach for ancestor and sibling references is to use the so-called root reference approach. In this approach, begin the reference with the root reference (the "/") and then use the same technique as for child or grandchild directories. **Note that these will only work on the server! That is, they will not work when you test it out on your local machine.**
	To link to about.html from bio.html, use:
	`<a href="/about.html">`
	To link to background.gif from bio.html, use:
	`<a href="/images/background.gif">`

7 Default Document
Web servers allow references to directory names without file names. In such a case, the web server will serve the default document, which is usually a file called index.html (Apache) or default.html (IIS). **Again, this will only generally work on the web server.**
	To link to index.html in members from about.html, use either:
	`<a href="members">`
	Or
	`<a href="/members">`

TABLE 2.1 Sample Relative Referencing

4. Inline Text Elements

Inline elements are those that do not disrupt the flow of text (i.e., cause a line break). HTML defines over 30 of these elements. Table 2.2 lists some of the most commonly used of these elements.

Element	Description
<a>	Anchor used for hyperlinks.
<abbr>	An abbreviation
 	Line break
<cite>	Citation (i.e., a reference to another work).
<code>	Used for displaying code, such as markup or programming code.
	Emphasis
<mark>	For displaying highlighted text
<small>	For displaying the fine-print, i.e., "non-vital" text, such as copyright or legal notices.
	The inline equivalent of the element. It is generally used to mark text that will receive special formatting using CSS.
	For content that is strongly important.
<time>	For displaying time and date data

TABLE 2.2 Common Text-Level Semantic Elements

Example program to demonstrate tags of html:

```
<!DOCTYPE html>
<html>
<head>
        <title>My First HTML</title>
        <meta charset="ANSI">
</head>

<body>
        <!-- This is a comment -->
        <p>This is a paragraph.</p>

        <em>Emphasized text</em><br>
```

```
<p><strong>Note:</strong> The mark tag is not supported in Internet Explorer 8 and earlier versions.</p>

<code>A piece of computer code</code><br>

<p>Do not forget to buy <mark>milk</mark> today.</p>
<p>My mother has <span style="color:blue;font-weight:bold">blue</span> eyes and my father has <span style="color:darkolivegreen;font-weight:bold">dark green</span> eyes.</p>
<p>We open at <time>10:00</time> every morning.</p>
<hr>

<p>This is a <u>parragraph</u>.</p>

<p>He named his car <i>The lightning</i>, because it was very fast.</p>

<p>This is normal text - <b>and this is bold text</b>.</p>

<p>This text is normal.</p>
<p><big>This text is bigger.</big></p>

</body>
</html>
```

Output:

This is a paragraph.

Emphasized text

Note: The mark tag is not supported in Internet Explorer 8 and earlier versions.

```
A piece of computer code
```

Do not forget to buy milk today.

My mother has **blue** eyes and my father has **dark green** eyes.

We open at 10:00 every morning.

This is a parragraph.

He named his car *The lightning*, because it was very fast.

This is normal text - **and this is bold text**.

This text is normal.

This text is bigger.

5. Images

HTML images are defined with the <img> tag. The source file (src), alternative text (alt), width, and height are provided as attributes

The src Attribute: HTML images are defined with the <img> tag. The filename of the image source is specified in the src attribute:

```
<img src="img_girl.jpg">
```

The width and height Attributes : HTML images also have width and height attributes, which specifies the width and height of the image:

```
<img src="img_girl.jpg" width="500" height="600">
```

The alt Attribute : The alt attribute specifies an alternative text to be used, if an image cannot be displayed. The value of the alt attribute can be read by screen readers. This way, someone "listening" to the webpage, e.g. a vision impaired person, can "hear" the element.

```
<img src="img_girl.jpg" alt="Girl with a jacket">
```

Example:

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

<h2>Alternative text</h2>

<p>The alt attribute should reflect the image content,
so users who cannot see the image gets an
understanding of what the image contains:</p>

<img src="img_chania.jpg" alt="Flowers in Chania"
width="460" height="345">

</body>
</html>
```

Alternative text

The alt attribute should reflect the image content, so users who cannot see the image gets an understanding of what the image contains:

6. Character Entities

Character entity are special characters for symbols for which there is either no easy way to type them via a keyboard (such as the copyright symbol or accented characters) or which have a reserved meaning in HTML (for instance the “<” or “>” symbols). There are many HTML character entities. They can be used in an HTML document by using the entity name or the entity number. Some of the most common are listed in Table 2.3.

Entity Name	Entity Number	Description
		Nonbreakable space. The browser ignores multiple spaces in the source HTML file. If you need to display multiple spaces, you can do so using the nonbreakable space entity.
&lt;	&#60;	Less than symbol ("<").
&gt;	&#62;	Greater than symbol (">").
&copy;	&#169;	The © copyright symbol.
&euro;	&#8364;	The € euro symbol.
&trade;	&#8482;	The ™ trademark symbol.
&uuml;	&#252;	The ü—i.e., small u with umlaut mark.

TABLE 2.3 Common Character Entities

7. Lists

The most common block-level elements in HTML are list. HTML provides three types of lists:

■ **Unordered lists**. Collections of items in no particular order; these are by default rendered by the browser as a bulleted list. However, it is common in CSS to style unordered lists without the bullets. Unordered lists have become the conventional way to mark-up navigational menus.

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

■ **Ordered lists**. Collections of items that have a set order; these are by default rendered by the browser as a numbered list. An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.

■ **Definition lists**. Collection of name and definition pairs. These tend to be used infrequently

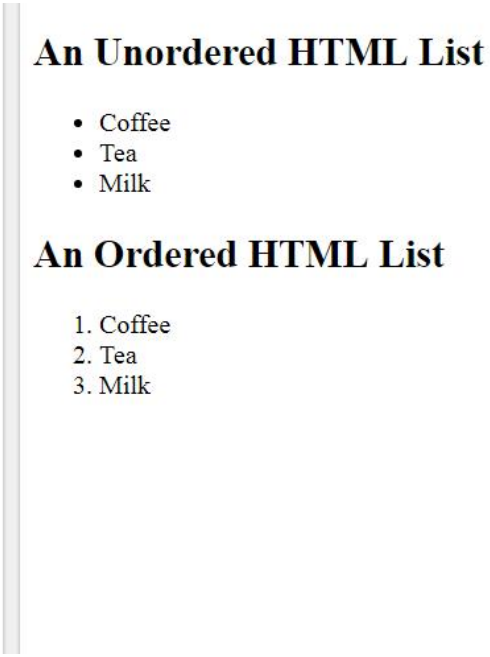
The <dd> tag is used to describe a term/name in a description list.

The <dd> tag is used in conjunction with <dl> (defines a description list) and <dt> (defines terms/names). Inside a <dd> tag you can put paragraphs, line breaks, images, links, lists, etc.

.

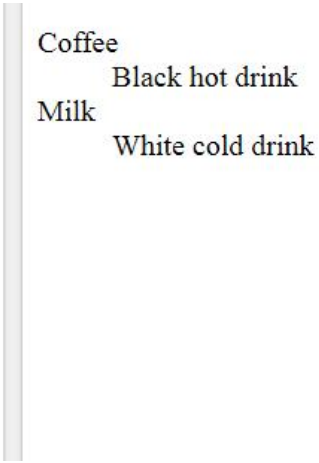
Example: ordered and unordered:

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

<h2>An Unordered HTML List</h2>

<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>

<h2>An Ordered HTML List</h2>

<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

</body>
</html>
```

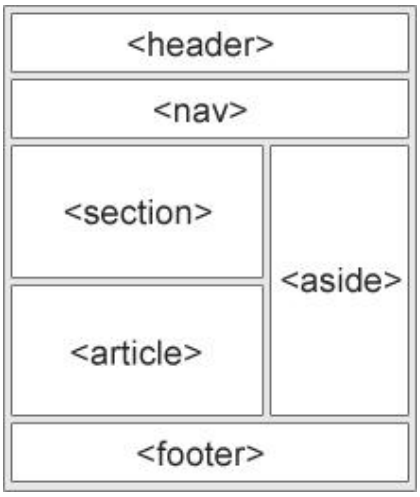


Example of Definition List:

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

<dl>
  <dt>Coffee</dt>
  <dd>Black hot drink</dd>
  <dt>Milk</dt>
  <dd>White cold drink</dd>
</dl>

</body>
</html>
```



- **HTML5 Semantic Structure Elements**



Html5- Parts of web Page

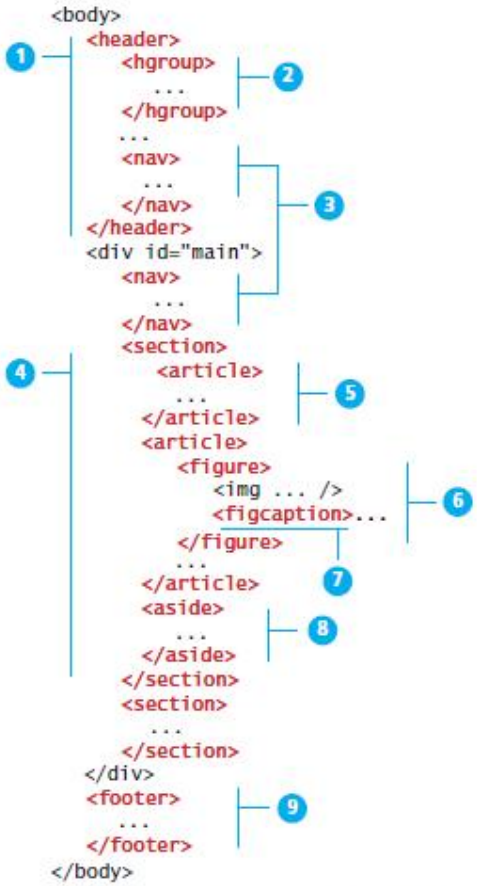


FIGURE 2.21 Sample layout using new HTML5 semantic structure elements

Header and Footer

Most website pages have a recognizable header and footer section. The header element contains the site logo and title (and perhaps additional subtitles or taglines), horizontal navigation links, and perhaps one or two horizontal banners.

Header can contain multiple headings in close proximity. The <hgroup> element can be used in such a circumstance to group them together within one container. The <hgroup> element can be used in contexts other than a header. For instance, one could also use an <hgroup> within an <article> or a <section> element as well. The <hgroup> element can only contain <h1>, <h2>, etc., elements

The typical footer contains information such as smaller text versions of the navigation, copyright notices, information about the site's privacy policy, and perhaps twitter feeds or links to other social sites.

1. **Example: Footer**

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

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

</body>
</html>
```

Posted by: Hege Refsnes

Contact information: someone@example.com.

2. **Example: Header**

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

<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>

</body>
</html>
```

What Does WWF Do?

WWF's mission:

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.

Navigation

The <nav> element represents a section of a page that contains links to other pages or to other parts within the same page. Like the other new HTML5 semantic elements, the browser does not apply any special presentation to the <nav> element. Not all links of document should be inside a <nav> element. The <nav> element is intended only for major block of navigation links.

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

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

</body>
</html>
```

HTML | CSS | JavaScript | jQuery

Articles and Sections

The article element represents a section of content that forms an independent part of a document or site; for example, a magazine or newspaper article, or a blog entry. The section element represents a section of a document, typically with a title or heading.

The **<article>** element specifies independent, self-contained content.
The **<section> element** defines section in a document.

HTML5 <article> Element

The <article> element specifies independent, self-contained content.
An article should make sense on its own, and it should be possible to read it independently from the rest of the web site. Examples of where an <article> element can be used:

- Forum post
- Blog post

- Newspaper article

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

<article>
  <h1>What Does WWF Do?</h1>
  <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>

</body>
</html>
```

What Does WWF Do?

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.

HTML5 <section> Element

The <section> element defines a section in a document. According to W3C's HTML5 documentation: "A section is a thematic grouping of content, typically with a heading." A home page could normally be split into sections for introduction, content, and contact information.

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

<section>
  <h1>WWF</h1>
  <p>The World Wide Fund for Nature (WWF) is an
international organization working on issues .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
</p>
</section>

</body>
</html>
```

WWF

The World Wide Fund for Nature (WWF) is an international organization working on issues .WWF was founded in 1961.

WWF's Panda symbol

The Panda has become the symbol of WWF. The well-known panda logo of WWF originated from a panda named

Figure and Figure Captions:

Screen captures or diagrams or photographs that are separate from the text (but related to it), which are described by a caption, and which are given the generic name of *Figure*.

<figure> element can be used to annotate illustrations, diagrams, photos, code, listing etc. ,that are moved away from the primary content.

<figcaption> used to give generic name for the figure.

- <img> element is used to display an **image** such as a photograph or illustration. It includes a link to the **image** file, which is typically a .jpg, .png or .gif. ...
- <**figure**> element is used as a container to hold an **image**, graph or other illustration. ..

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

<h2>Places to Visit</h2>

<p>Puglia's most famous sight is the unique conical
houses (Trulli) found in the area around Alberobello,
a declared UNESCO World Heritage Site.</p>

<figure>
  <img src="pic_trulli.jpg" alt="Trulli"
style="width:100%">
  <figcaption>Fig.1 - Trulli, Puglia, Italy.
</figcaption>
</figure>

</body>
</html>
```

Places to Visit

Puglia's most famous sight is the unique conical houses (Trulli) found in the area around Alberobello, a declared UNESCO World Heritage Site.

Fig.1 - Trulli, Puglia, Italy.

Aside

The <aside> element is similar to the <figure> element in that it is used for marking up content that is separate from the main content on the page.

The <aside> element is used for sidebars, pull quotes, groups of advertising images, or any other grouping of non-essential elements.