

## Lesson Objectives



After completing this lesson, you will be able to:

- Understand the structure of an HTML page.
- Learn to apply physical/logical character effects.
- Learn to manage document spacing.
- New Semantic Elements in HTML 5

**1.1: Understand the structure of an HTML page**

## What is HTML?

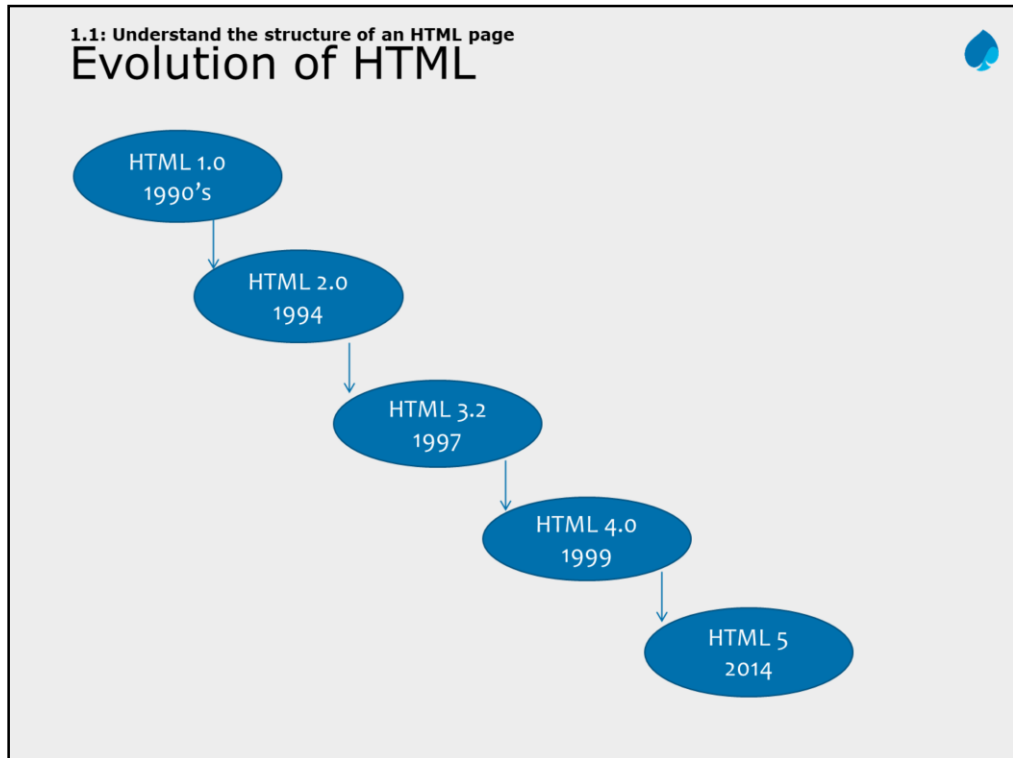


HTML is a language for describing web pages.

- It stands for Hyper Text Markup Language
- HTML is a markup language and not a programming language
- HTML uses markup tags to describe web pages.

HTML is a markup language and these documents describe web pages. Any HTML document contains HTML tags and plain text. HTML documents are also called web pages.

A web browser (like Internet Explorer or Firefox) reads HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:



HTML is a markup language and these documents describe web pages. Any HTML document contains HTML tags and plain text. HTML documents are also called web pages.

A web browser (like Internet Explorer or Firefox) reads HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:

1.1: Understand the structure of an HTML page

## HTML5 – The Good News !!!



Good support on modern mobile devices (iOS, Android)  
Simpler, more intuitive syntax  
Video and Audio can be included without requiring a plug-in  
Incremental improvements to previous HTML challenges  
Much needed next step in HTML evolution  
Creative enhancements: Rounded corners, gradients, text layout  
Promising support of Mobile JS Frameworks (Sencha, jQTouch)

HTML is a markup language and these documents describe web pages. Any HTML document contains HTML tags and plain text. HTML documents are also called web pages.

A web browser (like Internet Explorer or Firefox) reads HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:

1.1: Understand the structure of an HTML page

## HTML5 - Browser Support



**Android 2.2**



**Chrome 6**



**Firefox 4.0**



**Opera  
10.6**



**IE 9**



**Safari 5**



**Safari  
Mobile**

Capgemini Internal

#### Browser Support:

HTML5 is still a work in progress. However, most modern browsers have some HTML5 support. The latest versions of Safari, Chrome, Firefox, and Opera support some HTML5 features. Internet Explorer 9 will support some HTML5 features.

#### Some Statistics:

Chrome 10.0 is the big winner coming in. They have already implemented most of the functionality to be a compatible HTML5 browser. In addition, many of the features

that are not added are partially added. So Google Chrome is definitely way out in front in the race for HTML compatible browsers.

Firefox 4.0 is next in line. They are still missing a lot of key elements but got bonus points for the audio and video implementation as well as their parsing rules.

Internet Explorer 8 is pretty far behind the curve. Pretty dismal showing for what used to be the top internet browser in the world. Internet Explorer has been playing catch up with rendering design since the implementation of CSS and their poor showing here tells us it does not seem like much will change in the future.

So those are the statistics. At the moment the only HTML5 browser that is going to get you very far is Google Chrome and until browsers catch up with the newer language it is probably a good idea to use it sparingly in your designs until it actually is a true and tested standard.

## 1.1: Understand the structure of an HTML page

# HTML Elements



Most Web documents are created using HTML.

Documents are saved with extension .html or .htm.

Tags are strings in the language surrounded by a less-than (<) and a greater-than (>) sign.

- Opening tag: <html> Ending tag: </html>

Can have Attributes

- Attributes are Name-Value pairs added to HTML start tags.

HTML documents are text files made up of HTML elements. These files are saved with an extension of .htm or .html. A file with this extension indicates to the browser that the file is an HTML document.

You can check for the document structure and look for the presence of HTML element or tag. HTML elements are defined using HTML tags. Following are features of HTML tags:

Used to mark-up HTML elements.

Surrounded by the two characters "<" and ">".

Normally come in pairs.

HTML element starts with a start tag <b>. Then, the content of the HTML element is placed. This ends with an end tag </b>

Not case sensitive.

For HTML, the usual filename extension is .html  
(.htm for PC-based servers).

\*\*\*\*\*

Q: Do we need to write the end tag for all the elements?

A: In HTML, there are some tags which do not have a closing tag. For example, <br> and <hr> tags. So, it is not necessary to write the closing tags for such tags. But it is a good idea to write an empty tag by providing a "/" in the end tag.

\*\*\*\*\*



## 1.1: Understand the structure of an HTML page

## Rules applicable for HTML5 Tags



Rules for HTML5 Tags are:

- The document must include with an HTML5 DOCTYPE.
- Tags and attributes are case-insensitive.
- Attributes do not need to be quoted.
- End tags are not required for every element.
- Some attributes may be empty such as checked and disabled .
  - For example, `<input type=checkbox checked>`
- Only void elements such as `br`, `img` and `link` may be "self-closed" with `/>`.

We stated that HTML tags are not case sensitive, but let's always use lowercase tags. To prepare yourself for the next generations of HTML, start using lowercase tags.

### Tag Attributes

Tags can have attributes. Attributes provide additional information about HTML elements on your page.

For an Example:

This tag defines the `<body>` element of your HTML page. With an added `bgcolor` attribute, you can tell the browser that the background color of your page should be red, like this:

```
<body bgcolor="red">
```

Attributes always come in name/value pairs like this: `name="value"`. Attributes are always added to the start tag of an HTML element.

### Quote Styles

Always enclose values in quotes. Double quotes are more common, but single quotes are also allowed. In some rare situations, like when the attribute value itself contains quotes, it is necessary to use single quotes:

```
name='John "ShotGun" Nelson'
```

## 1.1: Understand the structure of an HTML page

## HTML Elements (Code)



## Code Snippet

An HTML document appears as follows:

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Title of page</title>
  </head>
  <body>
    This is my first homepage. <b>This text is
    bold</b>
  </body>
</html>
```

Save the file as "mypage.html".

Click "Browse" (or "Choose File") and locate the HTML file you just created - "mypage.html". Select it and click "Open". Now you should see an address in the dialog box, for example "C:\MyDocuments\mypage.htm". Click OK, and the browser will display the page.

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">

It tells type, version, language of particular document. The first tag in your HTML document is <html>. This tag tells your browser that this is the start of an HTML document. The last tag in your document is </html>. This tag tells your browser that this is the end of the HTML document.

Text between the <head> tag and the </head> tag is the header information. This information is not displayed in the browser window. Text between the <title> tags is the title of your document. Title is displayed in your browser's title bar. Text between the <body> tags is the text that is displayed in your browser. Text between the <b> and </b> tags is displayed in a bold font.

## 1.1: Understand the structure of an HTML page

## HTML Document Sections



### HTML Head Section:

- `<head>...</head>`
  - Page Title, Base URL, Meta Information

### HTML Body Section:

- `<body>...</body>`
  - Text, Images, Tables Colors, etc.

Head element can contain information about the document. The browser does not display this information to the user. Following tags can be in the head section: `<base>`, `<link>`, `<meta>`, `<script>`, `<style>`, and `<title>`.

Body element defines the document's body. It contains all the contents of the document (like text, images, colors, graphics, etc.).

1.1: Understand the structure of an HTML page

## Head Section: Page Title (Code)



### Code Snippet

Document Title is displayed using <title>.....</title> tag.

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <head>
```

```
    < title>
```

```
      My First Page
```

```
    </ title>
```

```
  </head>
```

```
</html>
```

<TITLE> element does not have any visible effect within a browser's client area; however, the enclosed title appears in the title bar of the browser window.

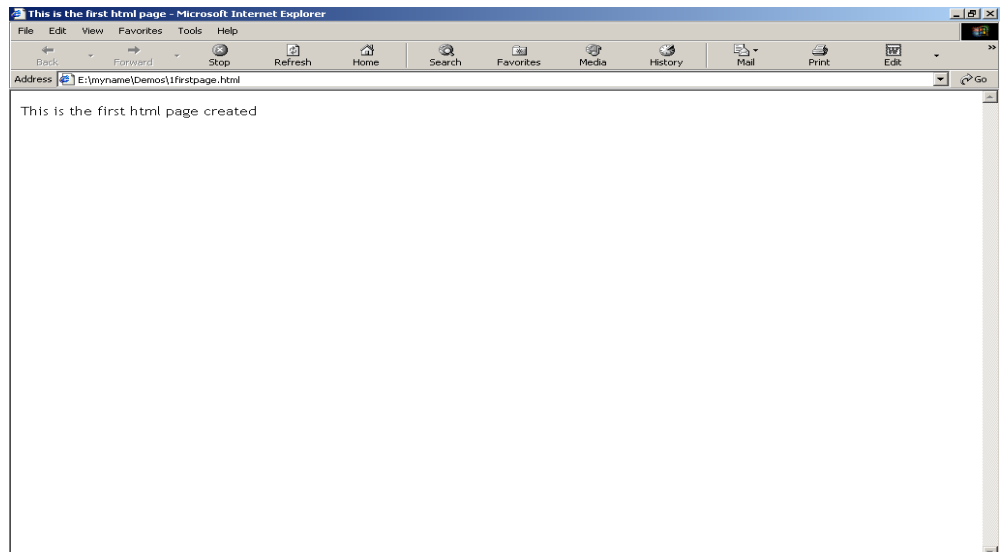
## 1.1: Understand the structure of an HTML page

## Demo



Firstpage.html

```
<html>  
<head><title>This is the first html page</title>  
<body>This is the first html page created</body>  
</head>  
</html>
```



## 1.1: Understand the structure of an HTML page

## Head Section: Document Base URL



## Code Snippet

```
<!DOCTYPE HTML>
<html>
  <head>
    <title> Document Base URL Manipulation </title>
    <base href="URL/">
  </head>
</html>
```

**More Examples:**

```
<base href="http://www.state.edu/images/">
<base href="ftp://ftp.state.edu/images/">
```

Base element specifies a base URL for all the links in a page.

Note: The `<base>` tag must go inside the head element.

You can set the `<base>` element only once in a document, in the header. `<base>` element does not work outside the header.

Assume that the absolute address for an image is:

```

```

Now, we insert the `<base>` tag, which specifies a base URL for all of the links on a page, in it's head section:

```
<head>
<base href="http://www.state.edu/images/" />
</head>
```

When we insert images on the page, in the example above, we just specify the relative address, `` and the browser looks for that file using the full URL, "http://www.state.edu/images/smile.gif".

## 1.1: Understand the structure of an HTML page

## Head Section: Meta Information



The <meta> tag provides metadata about the HTML document such as descriptions and keywords for search engine. Metadata will not be displayed on the page, but will be machine parsable. Examples:

```
<meta .....>
<meta name="keywords" content="HTML, CSS" />
<meta name="author" content="Username">
<meta http-equiv=refresh content=60 />
<meta http-equiv=refresh content="20;url=c:/html/htm3.htm" />
<meta charset="UTF-8">
```

META: This element is used to specify additional meta information and appears in the <head> tag. You can specify multiple <meta> tags for the document. It provides meta-information about your page, such as descriptions and keywords for search engines and refresh rates. Following are some uses of meta:

Define keywords for search engines:

```
<meta name="keywords" content="HTML, DHTML, CSS, XML, XHTML,
JavaScript, VBScript" />
```

Provide a description of your web page:

```
<meta name="description" content="Free Web tutorials on HTML, CSS,
XML, and XHTML" />
```

Define the last revision of your page:

```
<meta name="revised" content="Hege Refsnes, 6/10/99" />
```

Refresh page every 5 seconds:

```
<meta http-equiv="refresh" content="5" />
```

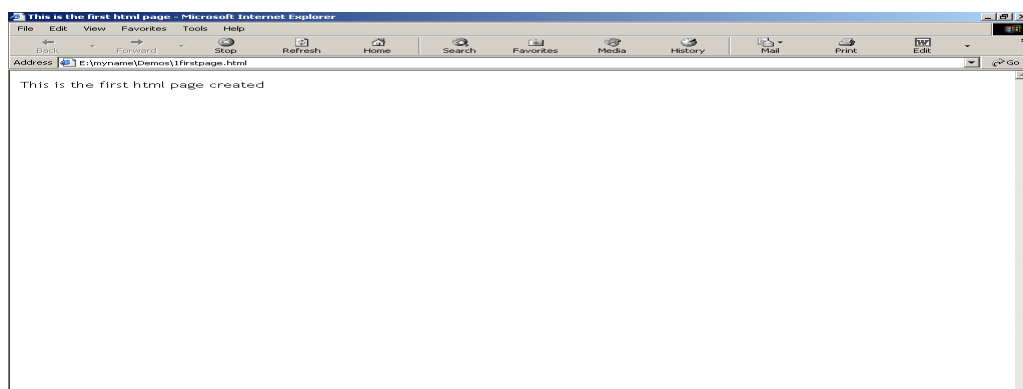
## 1.1: Understand the structure of an HTML page

## Demo



Meta.html

```
<!DOCTYPE HTML>
<html>
<head>
<!-- in meta tag in name attribute user can specify the value like keywords, description -->
<meta name="generator" content="microsoft visual studio 6.0">
<title>meta information</title>
<!-- this meta information help you to refresh the page after 2 sec and load image -->
<meta http-equiv=refresh content="2;url=c:\image\birthday.bmp"/>
</head>
<body>
Hi <p>&nbsp;</p>
</body>
</html>
```





## 2.1: Understand the structure of an HTML page

## HTML Body Section



### <body> Element:

- Represents information content.
- Each document can have at most one <body> element.
- Body element is placed between </head> and </html> elements.
- Attributes supported in <body> element are:
  - Event Handler attributes like ononline, onoffline, onunload, onpagehide, onpageshow, etc..
  - Global attributes like id, style, class, hidden, lang, etc..

<body> tag defines the HTML element containing the body of the HTML document.

## 2.1: Understand the structure of an HTML page

## Document (Body) Contents



### Body Text

- HTML truncates spaces in your text.
- Use `<br>` to insert new lines.
- Use `<p>` tag to create paragraphs.

### Other Elements of Body Section:

- `<table>` tags are used to create tables.
- `<img>` tags are used to insert images.

When you write HTML text, you are never sure how text is displayed in another browser. Some computers have large displays whereas some have small ones. Text is reformatted every time the user resizes the window.

Avoid reformatting text in your editor by adding empty lines and spaces. HTML truncates spaces. Any number of spaces as well as a new line counts as one space. Use `<br>` to insert blank lines. You might have noticed that `<br>` tags can be written without the closing tag `</br>`.

HTML automatically adds an extra blank line before and after some elements, like before and after paragraphs and headings.

Use a `<p>` tag to start a new paragraph, to which you can assign new attributes. Most browsers also place an extra space after a `<p>`. A `<br>` causes the browser to maintain the current paragraph attributes but to start placing text on a new line.

`</table>` is used to close the table tag.

To insert an image, the tag `` is used. Image should be saved with extension `.gif`. The `<img>` tag is empty, which means that it contains attributes only and it has no closing tag.

## 2.1: Understand the structure of an HTML page

## Comments in HTML Document



Increase code readability.

Ignored by the browser.

Example of HTML comment:

- `<!-- This is a Sample HTML Comment -->`

Comments are used for better code readability. Browser will ignore a comment. You can use comments to explain your code, or to store program-specific information, which can help you when you edit the source code at a later date.

Comments are not visible to the user, but they are still available to the program. A good practice is to comment text inside the script and style elements to prevent older browsers, that do not support scripting or styles, from showing it as plain text.

To include comments in the document, use the comment tag. Comment is a special tag starting with a “<” sign, followed by “!” and two hyphens. Then, type the commented text. It ends with two hyphens and a “>” sign.

## 2.1: Understand the structure of an HTML page

## Demo



Body.html

```
<html>
<head>
<title>body tag</title>
</head>
<body bgcolor="pink" text="red" alink="green" link="yellow">
<a href="body.html">background</a> color of the page is pink.
and text color is red
</body>
</html>
```

**2.1: Understand the structure of an HTML page**

## Home Page Concepts



Most popular home pages reflects the personality of the sponsoring organization or corporation.

Keep the initial home page short and to the point.

First element visitor sees is a collection of navigation buttons to navigate to other pages.

When you publish a URL in print or any other marketing material, it points to the location of your home page.

## 2.2: physical/logical character effects

## Physical Character Effects



Tags applies physical character effects by formatting the characters are listed below:

Tag Name	Description	Example
<code>&lt;b&gt;...&lt;/b&gt;</code>	To specify bold text	<b>Language</b>
<code>&lt;i&gt;...&lt;/i&gt;</code>	Displays content in italic to define a part of text in an alternate voice or mood	<i>Language</i>
<code>&lt;u&gt;...&lt;/u&gt;</code>	Tags make text between them underlined	<u>Language</u>
<code>&lt;s&gt;...&lt;/s&gt;</code>	Tags make text between them appear striked out	<del>Language</del>
<code>&lt;sub&gt;...&lt;/sub&gt;</code>	To specify subscript text	H <sub>2</sub> O
<code>&lt;sup&gt;...&lt;/sup&gt;</code>	To define superscript text	27 <sup>th</sup> May

## BOLD

`<b>... </b>`: Tags make text between them bold e.g. `<b>capgemini</b>` is displayed as “capgemini”.

## ITALIC

`<i>...</i>`: Tags make text between them italicized e.g. `<i>icapgemini </i>` is displayed as “icapgemini”.

UNDERLINE `<u>...</u>`: Tags make text between them underlined e.g. `<u>capgemini </u>` is displayed as “capgemini”.

## STRIKETHROUGH

`<strike>` or `<s>` ..... `<strike>` or `<s>`: Tags make text between them appear striked out e.g. `<s>capgemini </s>` is displayed as “capgemini” (with a strike across it.)

## Subscripts and Superscripts

`<sub>text</sub>` subscripts text e.g. Chemical formula of water is H`<sub>2</sub>`O is displayed as “H<sub>2</sub>O”:

Formula for a parabola is `y=x<sup>2</sup>` is displayed as “Y=X<sup>2</sup>”.

2.2: physical/logical character effects

## Demo



PhysicalCharacterEffect.html

2.2: physical/logical character effects

## Logical Character Effects



Heading Styles:

- `<hn>.....</hn>`

Value of n can range from 1 to 6

Syl

```
<h1>This is level 1 heading</h1>
```

Heading Style

`<hn>.....</hn>`: Tag identifies headings and subheadings in a document. Here, n is the size of the heading ranging from 1 to 6 where 1 is largest and 6 is smallest.

For example:

```
<h1>This is an example of level 1 heading</h1>
```

```
<h6>This is an example of level 6 heading</h6>
```

```
<html>
<body>This is the first html page created
<h1>This is level 1 heading</h1>
<h2>This is level 2 heading</h2>
<h3>This is level 3 heading</h3>
</body>
</html>
```



## 2.2: physical/logical character effects

## Logical Character Effects (Contd...)



<b>&lt;code&gt;</b>	Displays any code part in the web page.
<b>&lt;var&gt;</b>	Displays any variable on the web page.
<b>&lt;kbd&gt;</b>	Displays computer commands and arguments.
<b>&lt;em&gt;</b>	Highlights specific areas of text enclosed within it.
<b>&lt;strong&gt;</b> <b>&gt;</b>	Used for defining important text.
<b>&lt;dfn&gt;</b>	Designed specifically for words and phrases defined in text.
<b>&lt;address&gt;</b> <b>&gt;</b>	Specifies information about the creator e.g. Address, mailing address.

Code : `<code>...</code>`: Tag displays any code part on the web page.

e.g. `<code>perform 2000-modify-para</code>` displays  
“perform 2000-modify-para

Var : `<var>....</var>`: Tag displays any variable on the web page

e.g. `<var>count</var>` displays count.

Keyboard

`<kbd>...</kbd>`: Tag to display computer commands and arguments, especially those to be entered by the user. The text tagged by `<kbd>` is usually shown in a fixed-width font. Indicating text as keyboard input.

E.g. `<kbd>copy *.exe c:\dir1</kbd>` displays

copy \*.exe c:\dir1.

## 2.2: physical/logical character effects

## Special Characters in HTML



### Character Entities

- Comprise following three parts:
  - Ampersand (&),
  - Entity name or a #
  - Character code
  - Semicolon (;)
- Included in HTML page using:
  - Character code/Entity number: Include any character using its ISO Latin 1 character code.
  - To display ">" symbol, character code is 62 i.e. &#62;

Some characters like the "<" character, have a special meaning in HTML, and therefore cannot be used in text. To display a less than sign (<) in HTML, we need use a character entity.

### Character Entities

Some characters have a special meaning in HTML, like the less than sign (<) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in the HTML source.

A character entity has three parts: an ampersand (&), an entity name or a # and an entity number, and finally a semicolon (;). To display a greater than sign in an HTML document we write: &gt; or &#62;

The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good.

Note: Entities are case sensitive.

### Non-Breaking Space

The most common character entity in HTML is the non-breaking space. Normally HTML will truncate spaces in your text. If you write 10 spaces, HTML will remove 9 of them. To add spaces, use the &nbsp; character entity.

### 2.3: Managing Document Spacing

## Managing Document Spacing



#### Horizontal Spacing

- Use `<hr>` tag for including horizontal rule in an HTML document

#### Vertical Spacing

- `<p>`: Paragraph Break
- `<br>`: Line Break

`<p> ...</p>`: Tag to start and end the new paragraph.

`<br>...</br>` Tag maintains current paragraph but enters text in the new line.

## 2.3: Managing Document Spacing

## Demo



LCharacterEffects.htm  
Spacing.html

```
<html>
<head><title>Horizontal and Vertical spacing</title></head>
<body>
<p>
This paragraph contains a lot of lines
in the source code,
but the browser ignores it.
</p>
<hr size="2" width="50%" color="blue">
<p>
Notice the horizontal rule occupying 50 % of the window width.
</p>
This paragraph contains <br> line breaks in the
source code <br> so this
is the third line displayed within the paragraph.
</body>
</html>
```

## 2.3: Managing Document Spacing

## Preformatted Text (Code)



`<pre>...</pre>` Tags display preformatted blocks of text with a fixed-space font.

`<pre>` tag displays text with white space, line breaks, and tabs.

Examples:

```
<!DOCTYPE html>
<html>
  <head><title> Preformatted Text</title>
  <body>
    <pre>Browser would display this paragraph
      as you are viewing here.
      No need to provide line or paragraph breaks.
    </pre>
  </body>
</html>
```

`<pre>...</pre>` Tags display preformatted blocks of text with a fixed-space font. When they appear inside `<PRE>` tag, white space, line breaks, and tabs are also displayed. It is mainly useful for simulating program listings.

Note: The `<pre>` tag takes care of carriage returns, tabs, spaces and even the bold tag.

E.g.:

```
<html>
<head><title>PREFORMATTED TEXT EXAMPLE</title></head>
<body>
<h3>EMPLOYEE DETAILS PREFORMATTING</h3>
<hr>
<pre>
  <b>EMPLOYEE DETAILS</b>
  EMPID      EMPNAME      DEPT
  1001      JOHN      ACCOUNTS<BR>
  1002      TOM      PURCHASE<BR>
  1003      TAMMY      SALES<BR>
</pre>
</body></html>
```

## 2.3: Managing Document Spacing

## Divisions in an HTML Document (Code)



The <div> tag defines a division or a section in an HTML document. The <div> tag is used to group block-elements to format them with CSS. Syntax:

Example:

```
<div>.....</div>
```

```
<!DOCTYPE html>
<html>
<body>
<div style="text-align: center">Text is center aligned.</div>
<div style="text-align: left">Text is left aligned.</div>
</body>
</html>
```

<div>...</div>: Tag creates divisions in Web pages. These can be used to set the alignment for an entire section of the page.

<div align=center>This text is at the center of the browser window.</div>:  
Tag to align the text to the center of web page.

<div align=left>This text is at the left side of the browser window.</div>:  
Tag to align the text to the left of the web page.

<div align=right>This text is at the right side of the browser window.</div>: Tag to align the text to the right of the web page.

Difference Between pre and div Tags:

<pre> is used for formatting the entire text.

<div> is used to divide the webpage and use center, left or right attribute to align the text of the divided web page.

Difference Between div and span Tags:

div is used as a paragraph break as it creates a logical division of the document. In contrast, SPAN simply applies style and alignment as specified.

div has align attribute in it which is not present in case of span.

div is generally used for a block of text. span is used for words or sentences.

Primary difference between them is that <span> does not format by itself. The <div> tag includes a paragraph break, as it defines a logical division in the document.

## 2.3: Managing Document Spacing

## <span> in an HTML Document



The HTML <span> tag is used for grouping and applying styles to inline elements.

The span tag is used with inline elements whilst the div tag is used with block-level content.

Example:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
HyperText Markup Language is the standard markup language  
used to create <span style="color:blue;font-weight:bold">
```

```
static </span> web pages </body>
```

```
</html>
```

SPAN is used for words or sentences.

Primary difference between them is that <span> does not format by itself. The <div> tag includes a paragraph break, as it defines a logical division in the document.

## Block level element



Block elements separate content into blocks

Each block element is displayed on its own with a line break before and after it

Example for block elements are `<p>`, `<div>`, `<ol>`, `<ul>`, `<li>` etc..

- `<div>` element is a block level element that can be used as a container for other HTML elements

SPAN is used for words or sentences.

Primary difference between them is that `<span>` does not format by itself. The `<div>` tag includes a paragraph break, as it defines a logical division in the document.



## Inline element



Inline elements won't separate content into blocks

Inline elements are normally displayed without line breaks.

Inline element is just displayed in the flow of the paragraph

Example for Inline elements are `<b>`, `<i>`, `<u>`, `<em>`, `<a>`, `<td>`, `<img>`, etc..

- `<span>` element is an inline element that can be used as a container for text.

SPAN is used for words or sentences.

Primary difference between them is that `<span>` does not format by itself. The `<div>` tag includes a paragraph break, as it defines a logical division in the document.

## Rules



All inline elements and text need to be nested inside another block element before they can go in the `<body>` element

Block elements are not allowed inside an inline element

Keep block elements out of your `<p>` element

Put text and inline elements inside block elements before adding them to a `<blockquote>`

For an example:

- `<ol>` and `<ul>` can have only `<li>`
- We can put text, inline elements or block elements inside `<li>`

SPAN is used for words or sentences.

Primary difference between them is that `<span>` does not format by itself. The `<div>` tag includes a paragraph break, as it defines a logical division in the document.

## Example of Inline and Block elements

```
<!DOCTYPE html>
<html>
<body>
<p>HyperText Markup Language is the standard markup
language used to create <span style="color:blue;font-
weight:bold"> static </span> web pages. JavaScript is a
scripting language used to make web page content as
<span style="color:orange;font-
weight:bold">dynamic</span>.</p>
</body>
</html>
```

SPAN is used for words or sentences.

Primary difference between them is that <span> does not format by itself. The <div> tag includes a paragraph break, as it defines a logical division in the document.

## 2.4 New Semantic Elements in HTML 5

## Laying out a page with HTML5

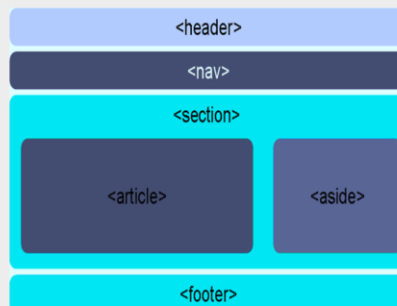
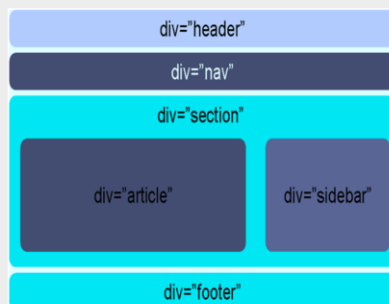


Most HTML 4 pages include a variety of common structures, such as headers, footers and columns

It's common to mark them up using div elements, giving each a descriptive id or class

HTML 5 addresses this issue by introducing new elements for representing each of these different sections

Elements that make it much easier to structure pages



SPAN is used for words or sentences.

Primary difference between them is that `<span>` does not format by itself. The `<div>` tag includes a paragraph break, as it defines a logical division in the document.

## 2.4 New Semantic Elements in HTML 5

## New Semantic Elements in HTML 5



| HTML Tag  | Description   |
|-----------|---|
| <article> | Defines a complete, self-contained block of related elements.   |
| <aside>   | Defines some content aside from the content it is placed in (like a sidebar)  |
| <footer>  | Defines a footer for a document or section which contains the author of the document, copyright information, links to terms of use, contact information |
| <header>  | Defines a header for a document or section. It can be used as a container for introductory content.   |
| <nav>     | Defines a set of navigation links   |
| <section> | Defines a section in a document   |

SPAN is used for words or sentences.

Primary difference between them is that <span> does not format by itself. The <div> tag includes a paragraph break, as it defines a logical division in the document.

## 2.4: New Semantic Elements in HTML5

## Demo

[LayoutDemo.html](#)

```
<html>
<head><title>Horizontal and Vertical spacing</title></head>
<body>
<p>
This paragraph contains a lot of lines
in the source code,
but the browser ignores it.
</p>
<hr size="2" width="50%" color="blue">
<p>
Notice the horizontal rule occupying 50 % of the window width.
</p>
This paragraph contains <br> line breaks in the
source code <br> so this
is the third line displayed within the paragraph.
</body>
</html>
```

## Lab Session



Lab 1

## Summary



In this lesson, you have learnt about

- The structure of an HTML page.
- Physical/logical character effects
- Managing document spacing.
- New Semantic elements in HTML5





## Review Question



Question 1: Which of the following are newly added content tags in HTML5?

- Option1: article
- Option2: div
- Option3: section
- Option4: p



Question 2: HTML document is saved with an extension .xml.

- True/False

Question 3: A Var tag is used to display the \_\_\_\_\_ in the web page.

## Review Question: Match the Following



|               |   |
|---------------|---|
| 1. Code       | a) Tag displays user-entered computer commands and arguments                          |
| 2. Var        | b) Tag is used for defining important text.   |
| 3. Kbd        | c) Tag displays any code part on the web page.  |
| 4. Emphasis   | d) Tag displays any variable on the web page.   |
| 5. Strong     | e) This tag designed specifically for words and phrases that are defined in the text. |
| 6. Definition | f) Tag for generic emphasis means this tag highlight specific areas of text.          |

