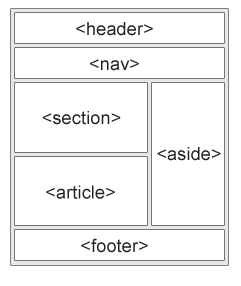
**HTML Basics**

* **HTML Layout Elements**
  + - HTML5 offers new semantic elements that define the different parts of a webpage:
      * <header> -- Defines a header for a document or a section
      * <nav> -- Defines a container for navigation links
      * <section> -- Defines a section in a document
      * <article> -- Defines an independent self-contained article
      * <aside> -- Defines content aside from the content
      * <footer> -- Defines a footer for a document or a section
      * <details> -- Defines additional details
      * <summary> -- Defines a heading for the <details> element



* **HTML Layout Techniques (Will have detailed discussion in CSS topics)**
  + - There are five different ways to create multicolumn layouts. Each way has its pros and cons
      * HTML tables (not recommended)
      * CSS float property
      * CSS flexbox
      * CSS framework
      * CSS grid
* **HTML Computer Code Elements**
  + **HTML <kbd> for Keyboard Input**
    - The HTML **<kbd>** element represents user input, like keyboard input or voice commands.
    - Text surrounded by **<kbd>** tags is typically displayed in the browser’s default monospace font:

|  |
| --- |
| <p>Save the document by pressing <kbd>Ctrl + S</kbd></p> |

Result

|  |
| --- |
| Save the document by pressing Ctrl + S |

* + **HTML <samp> for Keyboard Input**
    - The HTML **<samp>** element represents output from a program or computing system.
    - Text surrounded by **<samp>** tags is typically displayed in the browser’s default monospace font:

|  |
| --- |
| <p>If you input wrong value, the program will return <samp>Error!</samp></p> |

Result

|  |
| --- |
| If you input wrong value, the program will return Error! |

* + **HTML <code> for Keyboard Input**
    - The HTML **<code>** element defines a fragment of computer code.
    - Text surrounded by **<code>** tag is typically displayed in the browser’s default monospace font:

|  |
| --- |
| <code> x = 5; y = 6; z = x + y; </code> |

Result

|  |
| --- |
| x = 5; y = 6; z = x + y; |

* + - The HTML **<code>** element defines a fragment of computer code.
    - Notice that the **<code>** element does not preserve extra whitespace and line brakes.
    - To fix this we can put **<code>** element inside **<pre>** element

|  |
| --- |
| <pre> <code> x = 5; y = 6; z = x + y; </code> </pre> |

Result

|  |
| --- |
| x = 5;  y = 6;  z = x + y; |

* + **HTML <var> for Keyboard Input**
    - The HTML <var> element defines a variable
    - The variable could be a variable in mathematical expression or a variable in programming context:

|  |
| --- |
| Einstein wrote: <var>E</var> = <var>mc</var><sup>2</sup>. |

Result:

|  |
| --- |
| Einstein wrote: E = mc2. |

* **HTML Entities**
  + - Reserved characters in HTML must be replaced with character entities.
    - Characters that are not present on our keyboard can also be replaced by entities.
    - Some characters are reserved in HTML.
    - If we use the (<) or (>) sign in our text, the browser might mix them tags.
    - Character entities are used to display reserved characters in HTML.
    - A character entity looks like this:

|  |
| --- |
| &*entity\_name*;  OR  &#*entity\_number*; |

* + - To display a less than sign (<) we must write: **&lt;** or **&#60;**

|  |  |
| --- | --- |
| Advantages of entity name | An entity name is easy to remember |
| Disadvantages of entity name | Browsers may not support all entity names, but the support for number is good. |

* + **Non-breaking Space**
    - A common character entity used in HTML is the non-breaking space: **&nbsp;**
    - A non-breaking space is space that will not break into a new line.
    - Two words separated by a non-breaking space will stick together (not break into a new line). This is handy when breaking the words might be disruptive.

**Example:**

|  |
| --- |
| * § 10 * 10 km/h * 10 PM |

* + - Another common use of the non-breaking space is to prevent browsers from truncating spaces in HTML pages.
    - If we write 10 spaces in our text, the browser will remove 9 of them. To add real space to our text, we can use the **&nbsp;** character entity.
  + **Some of the useful HTML Character Entities**

|  |  |  |  |
| --- | --- | --- | --- |
| Result | Description | Entity Name | Entity Number |
|  | Non-breaking space | &nbsp; | &#160; |
| < | Less than | &lt; | &#60; |
| > | Greater than | &gt; | &#60; |
| & | ampersand | &amp; | &#38; |
| “ | Double quotation mark | &quot; | &#34; |
| ‘ | Single quotation mark(apostrophe) | &apos; | &#39; |
| ¢ | Cent | &cent; | &#162; |
| £ | Pound | &pound; | &#163; |
| ¥ | Yen | &yen; | &#165; |
| € | Euro | &euro; | &#8364; |
| © | Copyright | &copy; | &#169; |
| ® | Registered trademark | &reg; | &#174; |

* + **Combining Diacritical Marks**
    - A diacritical mark is a “glyph” added to a letter.
    - Some diacritical marks, like grave (  ̀) and acute (  ́) are called accents.
    - Diacritical marks can appear both and below a letter, inside a letter, and between two letters.
    - Diacritical mark can be used in combination with alphanumeric characters to produce a character that is not present in the character set (encoding) using in the page.

|  |  |  |  |
| --- | --- | --- | --- |
| Mark | Character | Construct | Result |
| ̀ | a | a&#768; | à |
| ́ | a | a&#769; | á |
| ̂ | a | a&#770; | â |
| ̃ | a | a&#771; | ã |
| ̀ | O | O&#768; | Ò |
| ́ | O | O&#769; | Ó |
| ̂ | O | O&#770; | Ô |
| ̃ | O | O&#771; | Õ |

* **HTML Encoding (Character Sets)**
  + **What is character encoding?**
    - To display an HTML page correctly, a browser must know which character set (Character Encoding) to use.
      * ASCII was the first **character encoding standard** (also called character set), and ASCII defined 128 alphanumeric characters that could be used on the internet.
      * ISO-8859-1 was the default character set for HTML4. This character set also supported 256 different character codes.
      * ANSI (Windows-1252). In addition to ISO-8859-1, ANSI has 32 extra characters.
      * UTF-8 (Unicode) covers almost all of the characters and symbols in the world.
      * The default character encoding for HTML5 is UTF-8
  + **The HTML charset Attribute**
    - To display an HTML page correctly, a browser must know character set used in the page.
    - This is specified in the **<meta>** tag.

**For HTML 4:**

|  |
| --- |
| <meta http-equiv="Content-Type" content="text/html;charset=ISO-8859-1"> |

**For HTML 5:**

|  |
| --- |
| <meta charset="UTF-8"> |

* + **The @charset CSS Rule**
    - You can use the CSS **@charset** rule to specify the character encoding used in a style sheet.

**Example: CSS style sheet encoding by UTF-8**

|  |
| --- |
| @charset "UTF-8"; |

* **HTML URL Encoding**
  + - A URL is another word for a web address
    - A URL can be composed of words (w3school.com), or an Internet Protocol (IP) address (192.168.20.50).
    - Most people entering the name when surfing, because names are easier to remember than numbers.
  + **URL – Uniform Resource Locator**
    - Web browsers request pages from web servers by using a URL.
    - A Uniform Resource Locator is used to address a document (or other data) on the web.

**Explanation:**

* + - * **scheme –** defines the type of Internet service (most common is **http** or **https)**.
      * **prefix –** defines a domain **prefix** (default for http is **www)**.
      * **domain –** defines the Internet **domain name** (like w3schools.com**)**.
      * **port –** defines the port number at the host (default for **http** is **80)**.
      * **path –** defines a **path** at the server (If omitted: the root directory of the site**)**.
      * **filename –** defines the name of a document or resource.
  + **Common URL Schemes**
    - http (Hyper Text Transfer Protocol) – Common webpages not encrypted.
    - https (Secure Hyper Text Transfer Protocol) – Secure webpages encrypted.
    - ftp (File Transfer Protocol) – Downloading or Uploading files.
    - File – a file on our computer
  + **URL Encoding**
    - URLs can be sent over the Internet using the ASCII character-set
    - If a URL contains characters outside the ASCII set, the URL has to be converted.
    - URL encoding converts non-ASCII characters into a format that can be transmitted over the Internet.
    - URL encoding replaces non-ASCII characters with a “%” followed by Hexadecimal digits.
    - URLs cannot contain spaces. URL encoding normally replaces a space with a plus (+) sign, or %20.

**Example:**

**Input Text Transmitted over Internet: Hello Günter**

**Input Received as:** text=Hello+G%C3%BCnter