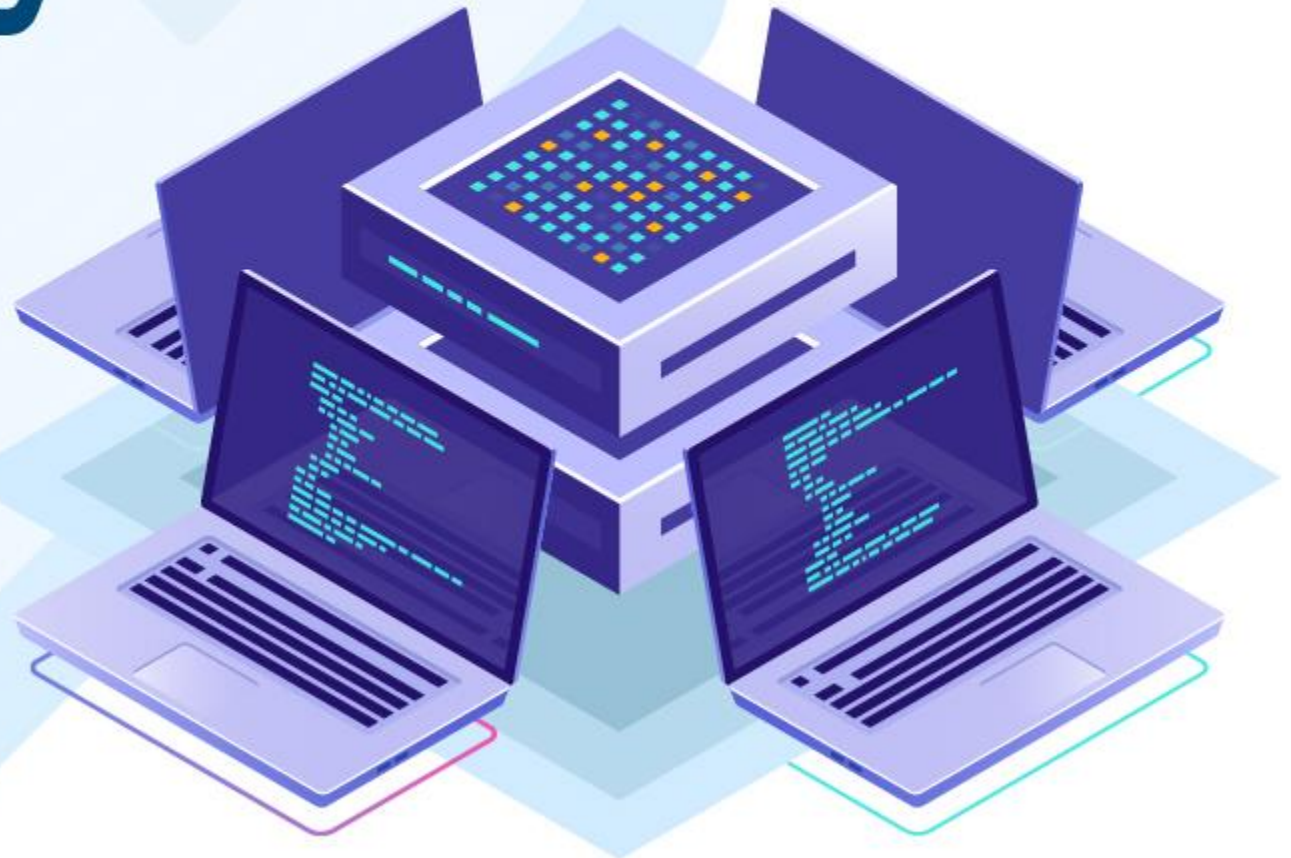


Information Technology

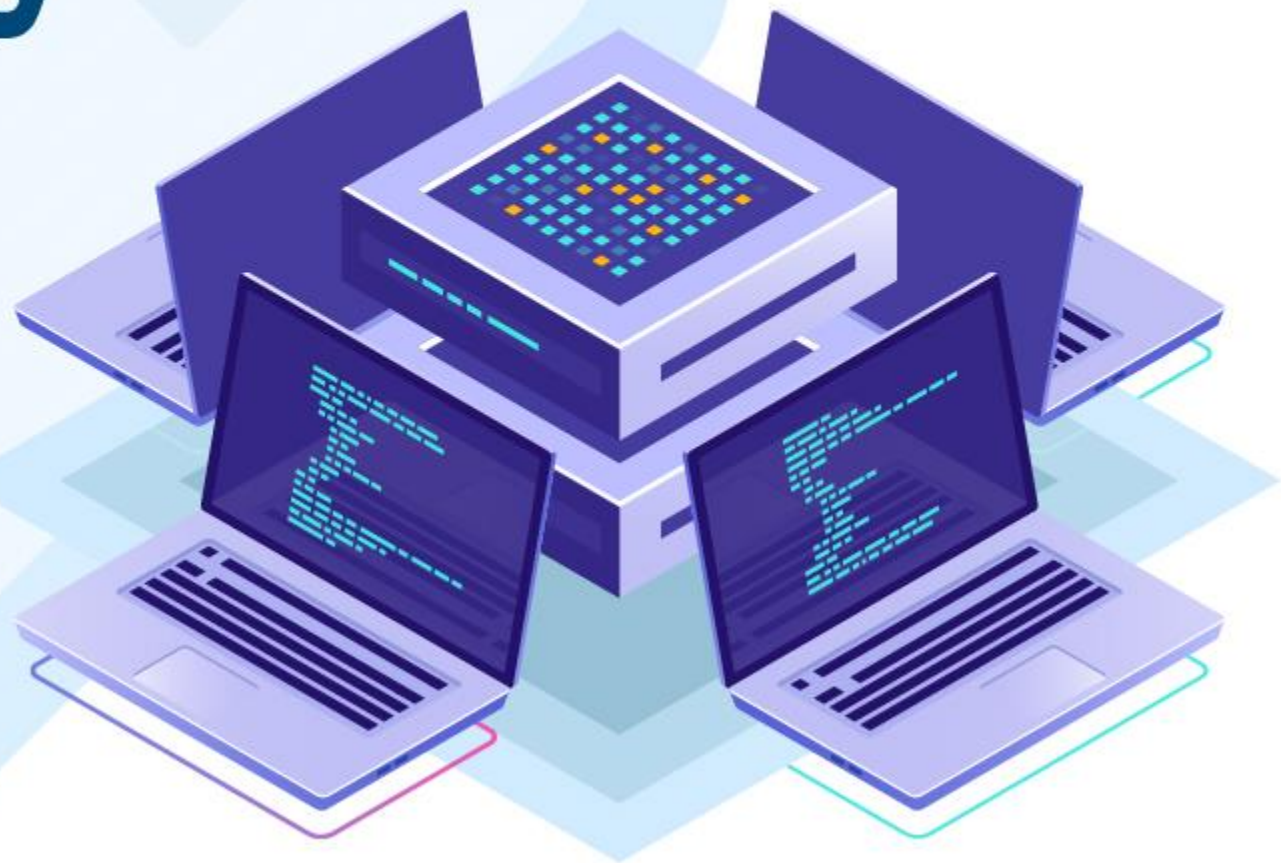
Spring 23-24
Semester 4

- » Web Technology
- » Dr. Rodaina Abdelsalam
rabdelsalam@eelu.edu.eg



Information Technology

- » Introduction to
- » internet concepts



EELU

الجامعة المصرية للتعليم الإلكتروني
THE EGYPTIAN E-LEARNING UNIVERSITY

Outline

- **Hypertext Markup Languages (HTML)**

Hypertext Markup Languages (HTML)

- Hypertext Markup Language (HTML) is the language for specifying the static content of Web pages.
- It is a simple markup whose primary purpose is to enable cross-referencing of documents through hyperlinks.
- Hypertext refers to the fact that Web pages are more than just text, it can contain multimedia, provide links for jumping within a page or out to other pages.
- Markup refers to the fact that it works by augmenting text with special symbols (tags <>) that identify structure and content type.
- HTML pages are presented using different web browsers.
- There are many versions of HTML and different browsers have their own add-ons.

Hypertext Markup Languages (HTML)

- HTML stands for **Hyper Text Markup Language**
- HTML is the standard markup language **for creating Web pages**
- HTML describes **the structure of a Web page**
- HTML consists of a **series of elements**
- HTML **elements label pieces of content** such as "this is a heading", "this is a paragraph", "this is a link", etc.
- An HTML file is a text file containing small **markup tags**
- An HTML file must have an **html file extension**
- An HTML file can be created using a **simple text editor**

Hypertext Markup Languages (HTML)

The <!DOCTYPE> Declaration

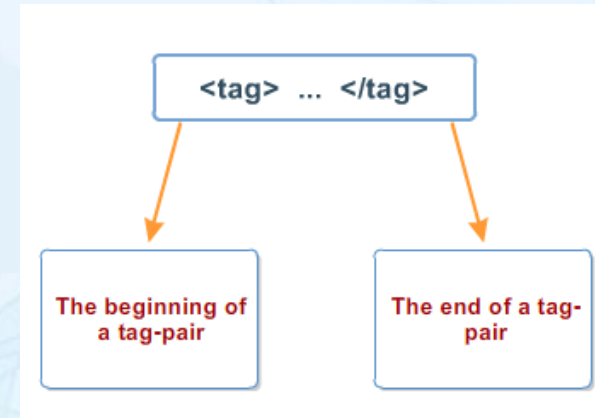
- The <!DOCTYPE> declaration represents the document type, and helps browsers to display web pages correctly.
- It must only appear once, at the top of the page (before any HTML tags).
- The <!DOCTYPE> declaration is not case sensitive.
- The <!DOCTYPE> declaration for HTML5 is:

<!DOCTYPE html>

HTML evolution

- HTML is an evolving language. It doesn't stay the same for long before a revised set of standards and specifications are brought in to allow easier creation of prettier and more efficient sites.
 - **HTML 1.0** was released in 1993 with the intention of sharing information that can be readable and accessible via web browsers.
 - Then comes the **HTML 2.0**, published in 1995, which contains all the features of HTML 1.0 along with that few additional features, which remained as the standard markup language for designing and creating websites until January 1997 and refined various core features of HTML.
 - Then comes the **HTML 3.0**, it included improved new features of HTML, giving more powerful characteristics for webmasters in designing web pages. But these powerful features of new HTML slowed down the browser in applying further improvements.
 - Then comes **HTML 4.01**, which is widely used and was a successful version of HTML before HTML 5.0, which is currently released and used worldwide.
 - **HTML 5** can be said for an extended version of HTML 4.01, which was published in the year 2012.

Hypertext Markup Languages (HTML)



- Tags typically occur in begin-end pairs.
- These pairs are in the form of `<tag> ... </tag>` that you can insert the tag content between them.
- Where the `<tag>` indicates the beginning of a tag-pair, and the `</tag>` indicates the end. (The three dots indicate an arbitrary amount of content between the tags.)
- `<tag>` and `</tag>` pairs define containers, any content within a container has the rules of that container applied to it.
 - For example, the text within a "boldface container" would be boldfaced. Similarly, paragraphs are defined by using a "paragraph container."

Hypertext Markup Languages (HTML)

HTML structural Elements:

- An HTML document has two main structural elements
 - HEAD that contains setup information for the browser and the Web page
 - e.g., the title for the browser window, style definitions, JavaScript code, description of the page, background information, and some other commands we will discuss later.
 - BODY contains the actual content to be displayed in the Web page
 - The visible part of the HTML document is between <body> and </body>.
- HEAD section enclosed between <head> and </head>
- BODY section enclosed between <body> and </body>

Hypertext Markup Languages (HTML)

Major HTML File Syntax Components

- HTML elements are defined using **HTML tags**.
- HTML tags are surrounded by the two characters `<` and `>`
- HTML tags normally come in pairs like `` and ``
- The first tag in a pair is the **start tag**, the second tag is the **end tag**
- HTML tags are **NOT case sensitive**, `` means the same as ``
- HTML tags may have many parameters or no parameters at all.

HTML tags

- **HTML `<html>..... </html>`:**
 - The first and last tags in a document should always be the HTML tags. These are tags that tell a Web browser where the HTML in your document begins and ends.
- The most absolute basic of all possible Web documents is:
 - **HEAD `<head>..... </head>`:**
 - The HEAD tags contain all of the document's header information. "header," doesn't mean what appears at the top of the browser window, but things like the document title and so on.
 - **TITLE `<title>.....</title>`:**
 - This container is placed within the HEAD structure, it contains the page title. [This will appear at the top of the browser's title bar, and also appears in the history list.](#) Finally, the contents of the TITLE container go into your bookmark file, if you create a bookmark to a page.

HTML tags

- **BODY tags in HTML:**

- **BODY `<body>..... </body>`:**

- BODY comes after the HEAD structure. Between the BODY tags, you find all of the stuff that gets displayed in the browser window. All of the text, the graphics, and links, and so on – these things occur between the BODY tags.

- **Comment Tags:**

- We write a comment to make a note in the code, but this comment doesn't appear in the browser like the following comment.

`<!-- Computer Application -->`

- You can write the comment [in any place in the code](#).

Hypertext Markup Languages (HTML)

```
1 <html>
2 <!--This is a comment. Computer Application-->
3
4 <head>
5   <title>Text Layout</title>
6 </head>
7
8 <body>
9   <p>
10     This is a paragraph of text<br />
11     made up of two lines.
12   </p>
13   <p>
14     This is another paragraph with a &nbsp; GAP &nbsp; between some of the words.
15   </p>
16   <p>
17     &nbsp;&nbsp;&nbsp; This paragraph is<br />
18     indented on the first line<br />
19     but not on subsequent lines.
20   </p>
21 </body>
22
23 </html>
24
```

This is a paragraph of text
made up of two lines.

This is another paragraph with a GAP between some of the words.

This paragraph is
indented on the first line
but not on subsequent lines.

HTML tags

- **Headings:**

- The heading structures are most commonly used to set apart document or section titles.
- There are six levels of headings, from Heading 1 to Heading 6.
- Heading 1 (H1) is "most important" and Heading 6 (H6) is "least important."
- By default, browsers will display the six heading levels in the same font, with the point size decreasing as the importance of the heading decreases.

Hypertext Markup Languages (HTML)

HTML Headings

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

<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>

</body>
</html>
```

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

HTML tags

- **Paragraph:**

- The beginning of a paragraph is marked by `<P>`, and the end by `</P>`.
- A new paragraph starts on a **new line, preceded by a blank line**.
- HTML automatically adds an extra blank line (white space , a margin) before and after a paragraph.

```
<body>  
    <p> This is a paragraph</p>  
    <p> This is another paragraph</p>  
</body>
```


HTML tags

- **Paragraph:**

- HTML Display

- You cannot be sure how HTML will be displayed. Large or small screens, and resized windows will create different results.
 - With HTML, you cannot change the display by adding extra spaces or extra lines in your HTML code. The browser will automatically remove any extra spaces and lines when the page is displayed.

This paragraph contains a lot of lines in the source code, but the browser ignores it.

This paragraph contains a lot of spaces in the source code, but the browser ignores it.

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

<p>
This paragraph
contains a lot of lines
in the source code,
but the browser
ignores it.
</p>

<p>
This paragraph
contains      a lot of spaces
in the source      code,
but the      browser
ignores it.
</p>

</body>
</html>
```

HTML tags

- **Line Break and Horizontal Ruler Tags**

- If you want to end a line after a certain word, but don't want to start a new paragraph so, what you need is a **line break**, which is invoked by using the `
` tag. This forces a line break wherever you place it in the content.
- You can insert a **horizontal ruled line** that delineates and separates sections of your document using the `<hr>`

This paragraph contains a lot of lines in the source code,
but the browser ignores it.

This paragraph contains a lot of spaces in the source code,

but the browser ignores it.

```
1  <!DOCTYPE html>
2  <html>
3
4  <body>
5
6      <p>
7          This paragraph
8          contains a lot of lines
9          in the source code,<br>
10         but the browser
11         ignores it.
12     </p>
13
14     <p>
15         This paragraph
16         contains a lot of spaces
17         in the source code,
18         <hr>
19         but the browser
20         ignores it.
21     </p>
22
23 </body>
24
25 </html>
```

Hypertext Markup Languages (HTML)

HTML structural Elements:

- Example:

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

This is my first homepage. **This text is bold**

Hypertext Markup Languages (HTML)

HTML structural Elements:

- Example:

```
<!DOCTYPE html>
```

```
<html>
```

```
  <body>
```

```
    <h1>My First Heading</h1>
```

```
    <p>My first paragraph.</p>
```

```
  </body>
```

```
</html>
```

- The `<html>` element is the root element and it defines the whole HTML document.
 - It has a start tag `<html>` and an end tag `</html>`.
- Then, inside the `<html>` element there is a `<body>` element. The `<body>` element defines the document's body.
 - It has a start tag `<body>` and an end tag `</body>`.
- Then, inside the `<body>` element there are two other elements: `<h1>` and `<p>`
- The `<h1>` element defines a heading.
 - It has a start tag `<h1>` and an end tag `</h1>`
- The `<p>` element defines a paragraph.
 - It has a start tag `<p>` and an end tag `</p>`

My First Heading
My first paragraph.

Hypertext Markup Languages (HTML)

Preformatted Text

- The `<pre>` `</pre>` tag identifies preformatted text.
- Text within this element preserves spaces and line breaks.

Hypertext Markup Languages (HTML)

HTML structural Elements:

- Every sequence of white space is interpreted as a single space (browsers ignore multiple spaces between words within paragraphs)
 - If you try to type two words separated by five spaces, only **one** space will be displayed in the browser.
- Browser automatically wraps the text to fit the window size.
- Extra line spacing can be done inserting NBSP (Non-Breaking Space), which is an invisible character that takes up one space, by typing ** **

Hypertext Markup Languages (HTML)

```
1 <html>
2 <!--This is a comment. Computer Application-->
3
4 <head>
5   <title>Text Layout</title>
6 </head>
7
8 <body>
9   <p>
10     This is a paragraph of text<br />
11     made up of two lines.
12   </p>
13   <p>
14     This is another paragraph with a &nbsp; GAP &nbsp; between some of the words.
15   </p>
16   <p>
17     &nbsp;&nbsp;&nbsp; This paragraph is<br />
18     indented on the first line<br />
19     but not on subsequent lines.
20   </p>
21 </body>
22
23 </html>
24
```

This is a paragraph of text
made up of two lines.

This is another paragraph with a GAP between some of the words.

This paragraph is
indented on the first line
but not on subsequent lines.

Hypertext Markup Languages (HTML)

Line Breaks

- Note that Browsers neglect spaces and newline in the html source.
- The `
` tag is used when you want to break a line, it forces a line break wherever you place it.

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


Hypertext Markup Languages (HTML)

```
1 <html>
2 <!--This is a comment. Computer Application-->
3
4 <head>
5   <title>Text Layout</title>
6 </head>
7
8 <body>
9   <p>
10    This is a paragraph of text<br />
11    made up of two lines.
12  </p>
13  <p>
14    This is another paragraph with a &nbsp; GAP &nbsp; between some of the words.
15  </p>
16  <p>
17    &nbsp;&nbsp;&nbsp; This paragraph is<br />
18    indented on the first line<br />
19    but not on subsequent lines.
20  </p>
21 </body>
22
23 </html>
24
```

This is a paragraph of text
made up of two lines.

This is another paragraph with a GAP between some of the words.

This paragraph is
indented on the first line
but not on subsequent lines.

Example of (HTML)

```
<!DOCTYPE html>
<html >
<!-- Computer Application -->
<head>
  <title> Blocks of titles </title>
</head>

<body>
  <h1>Major heading 1</h1>
  <p>
    Here is some text.
  </p>
  <h2>Subheading</h2>
  <p>
    Here is some subtext.
  </p>
  <hr/>
  <h1>Major heading 2</h1>
  <p>
    Here is some more text.
  </p>
</body>
</html>
```

Major heading 1

Here is some text.

Subheading

Here is some subtext.

Major heading 2

Here is some more text.

HTML Attributes

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

Hypertext Markup Languages (HTML)

HTML Links

- The `<a>` tag defines a hyperlink, which is used to link from one page to another.
- `<a>` must have a closing tag ``
- The most important attribute of the `<a>` element is the **href** attribute, which indicates the link's destination.
- By default, links will appear as follows in all browsers:
 - An unvisited link is underlined and **blue**
 - A visited link is underlined and **purple**
 - An active link is underlined and **red**
- The href attribute specifies the URL of the page the link goes to, i.e., the link's destination. Attributes are used to provide additional information about HTML elements.

HTML Attributes

Example of href Attribute

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

<h2>The href Attribute</h2>

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

<a href="https://www.w3schools.com">Visit W3Schools</a>

</body>
</html>
```

The href Attribute

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

[Visit W3Schools](https://www.w3schools.com)

The src and alt Attributes

The IMG link tag

- Images are placed in Web documents using the `` tag . This tag has no closing tag like `<a>`.
- Images are not technically inserted into a web page; images are linked to web pages. So in order to make the `` tag work, you need to use an `src` attribute.
- `src` stands for "the source of this graph." The value of `src` is the path (URL) to the image that you want to have displayed on your Web page.
- **Note:** When a web page loads; it is the browser, at that moment, that gets the image from a web server and inserts it into the page. Therefore, make sure that the image actually stays in the same spot in relation to the web page, otherwise your visitors will get a broken link icon.

The src and alt Attributes

- Thus, a typical image tag will take the form:

```

```

- The tag is used to embed an image in an HTML page.
- The tag has two required attributes:
 - **src** - Specifies the path to the image
 - **alt** - Specifies an alternate text for the image, if the image for some reason cannot be displayed
- **Note:** Also, always specify the width and height of an image. If width and height are not specified, the page might flicker while the image loads.

The src and alt Attributes

- Images can be placed almost anywhere within the body of the document. They can be between paragraphs, within paragraphs, in list items or definition-list definitions, and even within headings.
- How to insert images from another folder or from another web site:
``

Outline

1. HTML Page
2. LINKS
3. Frames and List
4. TABLES
5. Advanced HTML
6. DHTML

LINKS

Relative URL Versus Absolute URL:

- There are two ways to specify the URL in the src attribute:
 1. **Absolute URL** - Links to an external image that is hosted on another website.
Example: `src="https://www.w3schools.com/images/img_girl.jpg"`.
 - **Notes:** External images might be under copyright. If you do not get permission to use it, you may be in violation of copyright laws. In addition, you cannot control external images; it can suddenly be removed or changed.
 2. **Relative URL** - Links to an image that is hosted within the website. Here, the URL does not include the domain name.
Example: `src="img_girl.jpg"`.
If the URL begins with a slash, it will be relative to the domain.
Example: `src="/images/img_girl.jpg"`.
- **Tip:** It is almost always best to use relative URLs. They will not break if you change domain.

LINKS

Relative URL Versus Absolute URL:

- You can link to documents in other directories by specifying the relative path from the current document to the linked document.
- For example, a link to a file MyfirstwebPage.html located in the subdirectory Pages would be this line of code :

```
<a href="pages/My first webPage.html">WebPage</a>
```

- You can also use the complete URL. For example, to include a link to this primer in your document, enter this line of code:

```
< a href ="http://www.eelu.edu.eg>Egyptian ELearning University</a>
```

LINKS

- **Links to Specific Sections:**

- Anchors can also be used to move a reader to a particular section in a document (either the same or a different document) rather than to the top, which is the default.
- This type of an anchor is commonly called a **named anchor** because it creates the links, you insert HTML names within the document.

LINKS

Suppose you want to set a link from document A (documentA.html) to a specific section in another document (MainDocument.html).

- Enter the HTML coding for a link to a named anchor:

documentA.html:

```
<a href="MainDocument.html#ANP">Acadia National Park</a>.
```

- Next, create the named anchor (in this example "ANP") in MainDocument.html:

```
<H2><A NAME="ANP">Acadia National Park</a></H2>
```

- Instead of writing the name of another web page file you can go to a specific title in the same document web page, for example "Useful Tips Section" by:
 - Putting a named anchor inside an HTML document, then,
 - Creating a link to the "Useful Tips Section" inside the same document.

LINKS

The MAILTO link:

- Mailto links are used to redirect to an email address instead of a web page URL.
- When a user clicks on the Mailto link, the default email client on the visitor's computer opens and suggests sending a message to the email address mentioned in the Mailto link.
- To create a Mailto link, you need to use the HTML `<a>` tag with its `href` attribute, and insert a "mailto:" parameter after it, like the following:

```
<a href="mailto:email@example.com">Send Email</a>
```

- For example, enter:

```
<a href="mailto:rabdelsalam@eelu.edu.eg">Web Engineering 1 mail</a>
```

LINKS

The MAILTO link:

- If you want to send the email to more than one address, separate your email addresses with a comma:

```
<a href="mailto:email@example.com, secondemail@example.com">Send Email</a>
```


LINKS

The MAILTO link:

- The following fields can be filled out beforehand:
 - **subject** - for the subject line,
 - **cc** - for sending a carbon copy,
 - **bcc** - for sending a blind carbon copy,
 - **body** - for the message's body text.

LINKS

The MAILTO link:

- If you want to have a **subject field**, which is already filled out, add the “subject” parameter to the href attribute:

```
<a href="mailto:email@example.com?subject=Mail from our Website">Send Email</a>
```

- To add **CC** and **BCC** to your email, use the "cc" or "bcc" parameter on the href attribute:

```
<a href="mailto:email@example.com?cc=secondemail@example.com,anotheremail@example.com, &bcc=lastemail@example.com &subject=Mail from our Website">Send Email</a>
```

- To add a **body** text, use the "body" parameter with other parameters:

```
<a href="mailto:email@example.com?cc=secondemail@example.com,anotheremail@example.com, &bcc=lastemail@example.com &subject=Mail from our Website &body=Some body text here">Send Email</a>
```


HTML Formatting Elements

- Formatting elements were designed to display special types of text:
 - `` - Bold text
 - `` - Important text
 - `<i>` - Italic text
 - `` - Emphasized text
 - `<sub>` - Subscript text
 - `<sup>` - Superscript text
 - `<mark>` - Marked text
 - `<small>` - Smaller text
 - `` - Deleted text
 - `<ins>` - Inserted text
 - `<u>` - underlined text

<code>text</code>	Writes text as bold
<code><i>text</i></code>	Writes text in italics
<code><u>text</u></code>	Writes underlined text
<code><sub>text</sub></code>	Lowers text and makes it smaller
<code><sup>text</sup></code>	Lifts text and makes it smaller
<code><strike>text</strike></code>	Strikes a line through the text
<code><tt>text</tt></code>	Writes text as on a classic typewriter
<code><pre>text</pre></code>	Writes text exactly as it is, including spaces
<code>text</code>	Usually makes text emphasized
<code>text</code>	Usually makes text bold

HTML Formatting Elements

- Example using:
 - `` - Bold text
 - `<i>` - Italic text
 - `<sub>` - Subscript text
 - `<sup>` - Superscript text

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

<p><b>This text is bold</b></p>
<p><i>This text is italic</i></p>
<p>This is<sub> subscript</sub> and <sup>superscript</sup></p>

</body>
</html>
```

This text is bold

This text is italic

This is _{subscript} and ^{superscript}

HTML Formatting Elements

HTML `` and `` Elements

- The HTML `` element defines bold text, without any extra importance.
- Example

`This text is bold`

This text is normal.

This text is bold.

- The HTML `` element defines text with strong importance. The content inside is typically displayed in bold.
- Example

`This text is important!`

This text is normal.

This text is important!

HTML Formatting Elements

HTML <i> and Elements

- The HTML <i> element defines a part of text in an alternate voice or mood. The content inside is typically displayed in italic.
 - **Tip:** The <i> tag is often used to indicate a technical term, a phrase from another language, a thought, etc.

- Example

<i>This text is italic</i>

This text is normal.

This text is italic.

- The HTML element defines emphasized text. The content inside is typically displayed in italic.
 - **Tip:** A screen reader will pronounce the words in with an emphasis, using verbal stress.

- Example

This text is emphasized

This text is normal.

This text is emphasized.

HTML Formatting Elements

- **HTML <sup> Element**

- The HTML <sup> element defines superscript text.
- Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font.
- Superscript text can be used for footnotes, like WWW^[1]:

- **HTML <sub> Element**

- The HTML <sub> element defines subscript text.
- subscript text appears half a character below the normal line.

- **Example**

```
<p>This is<sub> subscript</sub> and <sup>superscript</sup></p>
```

This is_{subscript} and^{superscript}

HTML Formatting Elements

HTML <mark> Element

- The HTML <mark> element defines text that should be marked or highlighted:

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

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

</body>
</html>
```

Do not forget to buy **milk** today.

HTML Formatting Elements

HTML <small> Element

- The HTML <small> element defines smaller text:

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

<p>This is some normal text.</p>
<p><small>This is some smaller text.</small></p>

</body>
</html>
```

This is some normal text.

This is some smaller text.

HTML Formatting Elements

HTML Element

- The HTML element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text:

```
<p>My favorite color is <del>blue</del> red.</p>
```

My favorite color is ~~blue~~ red.

HTML <ins> Element

- The HTML <ins> element defines a text that has been inserted into a document. Browsers will usually underline inserted text:

```
<p>My favorite color is <ins>red</ins>.</p>
```

My favorite color is red.

HTML Formatting Elements

HTML Element

- The HTML element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text:

```
<p>My favorite color is <del>blue</del> red.</p>
```

My favorite color is ~~blue~~ red.

HTML <ins> Element

- The HTML <ins> element defines a text that has been inserted into a document. Browsers will usually underline inserted text:

```
<p>My favorite color is <ins>red</ins>.</p>
```

My favorite color is red.

HTML Formatting Elements

Text Appearance

- **We can specify more styles for fonts by using the following tags:**
 - `< tt >... < /tt >` to specify typewriter-like (fixed-width)font
 - `& amp;` to specify **&**
 - `& lt;` to specify **<**
 - `& gt;` to specify **>**
 - `& quot;` to specify **"**
 - `& copy;` to specify **©**

HTML Formatting Elements

Text Variations

We can use **simple** tags to *change* the appearance of **text** within web pages. Even ^{super}script and _{scripts} are supported.

Text Escape Sequences

& < > " ' ©

Preformatted text

Egyptian E-Learning University
Center of Alexandria

What's wrong here????

```
1  <!DOCTYPE html>
2  <html>
3  <!-- Computer Application -->
4
5      <head>
6          <title>
7              Text Variations and EscapeSequences
8          </title>
9      </head>
10
11     <body>
12         <h1>Text Variations</h1>
13         <p> We can use <b>simple</b> tags to
14             <i>change</i> the appearance of
15             <strong>text< /strong> within
16                 <tt>Web pages</tt>.
17                 Even super<sup>script</sup>
18                 and sub<sub>scripts</sub> are
19                 <em>supported</em>.
20         </p>
21         <h1>Text Escape Sequences</h1>
22         <p>
23             & < > &quot; &copy;
24         </p>
25         <h1>Preformatted text</h1>
26         <pre>
27             Egyptian E-Learning University
28             Center of Alexandria
29         </pre>
30     </body>
31
32 </html>
```

Text Size

- The following table displays tags used for controlling fonts sizes and its meaning:

<code><big>text</big></code>	increase the size by one
<code><small>text</small></code>	decrease the size by one
<code><h1>text</h1></code>	writes text in biggest heading
<code><h6>text</h6></code>	writes text in smallest heading
<code>text</code>	writes text in smallest font size. (8 pt)
<code>text</code>	writes text in biggest font size (36 pt)

Outline

1. HTML Page
2. LINKS
- 3. Frames and List**
4. TABLES
5. Advanced HTML
6. DHTML

Frames: iframe

- In HTML5 the [`<iframe>`](#) tag is used to embed another document within the current HTML document:

```
<iframe src="https://www.w3schools.com"></iframe>
```

- The **HTML `<iframe>` tag** specifies an inline **frame**.

```
<iframe src="https://www.w3schools.com" title="W3Schools Free Online Web Tutorials"></iframe>
```

- The `<iframe>` tag comes in pairs. The content is written between the opening (`<iframe>`) and closing (`</iframe>`) tags.
- It is a good practice to always include a title attribute for the `<iframe>`. This is used by screen readers to read out what the content of the `<iframe>` is.

Frames: iframe

- Example

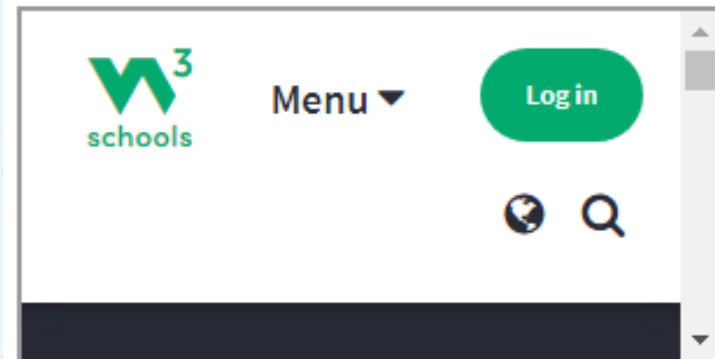
```
<!DOCTYPE html>
<html>
<body>

<h1>The iframe element</h1>

<iframe src="https://www.w3schools.com" title="W3Schools Free Online Web
Tutorials">
</iframe>

</body>
</html>
```

The iframe element



Frames: iframe

Some of iframe attributes:

- To set the size of iframe, use the height and width attributes, or use CSS. The attribute values are set in pixels by default, but they can also be in percent.
 - Height; its value is *pixels*, it specifies the height of an <iframe>. Default height is 150 pixels
 - Width; its value is *pixels*, it specifies the width of an <iframe>. Default width is 300 pixels.
- Example of height and width attributes in percentage:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <iframe src="https://www.w3docs.com" width="80%" height="300"></iframe>
  </body>
</html>
```

Frames: iframe

Some of iframe attributes:

- name; its value is *text*, it specifies the name of an <iframe>
- src; its value is *URLS*, it specifies the address of the document to embed in the <iframe>
- By default, an iframe is surrounded by a border. To remove the border, you can use CSS border property.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Title of the document</title>
```

```
</head>
```

```
<body>
```

```
<iframe src="https://www.w3docs.com" width="80%" height="300" style="border: none"></iframe>
```

```
</body>
```

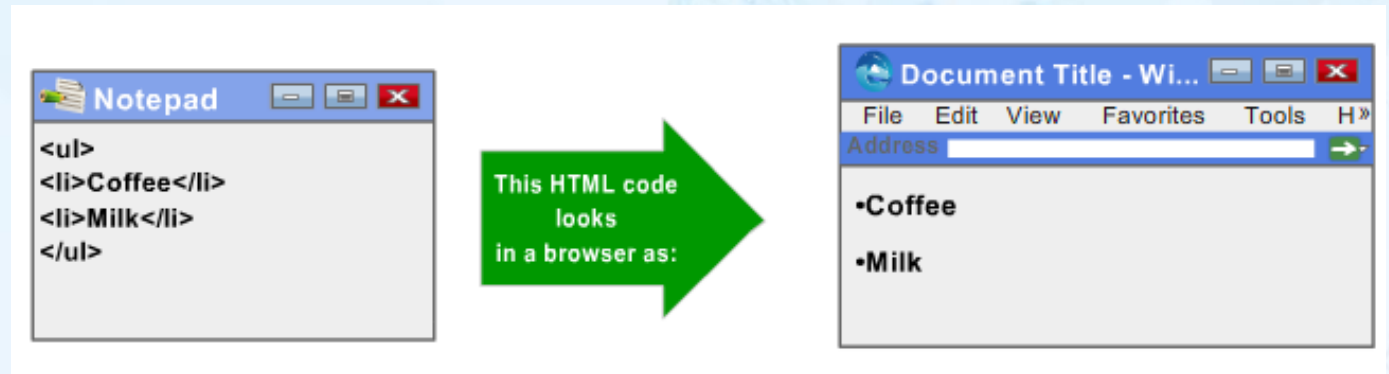
```
</html>
```

HTML Unordered Lists

- **An unordered list** starts with the `` tag. Each list item starts with the `` tag.
- The list items are marked with bullets (typically small black circles)

- **Example 1:**

```
<ul>  
  <li>Coffee</li>  
  <li>Milk</li>  
</ul>
```



HTML Unordered Lists

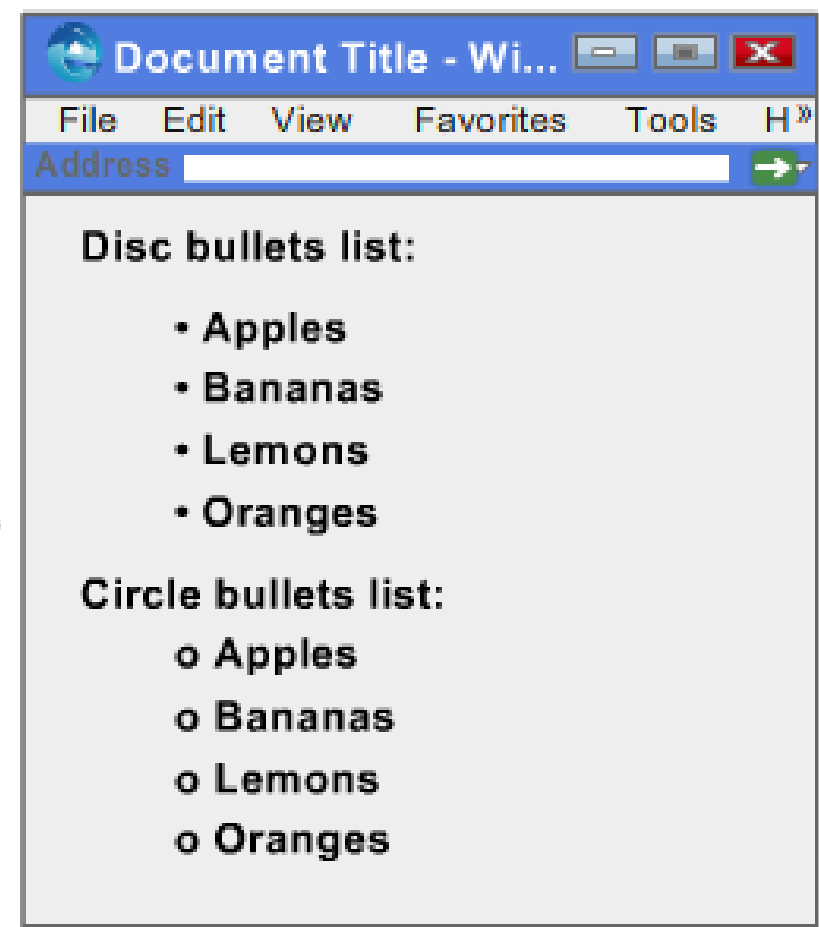
- **Unordered lists** have extension that produce differently shaped bullets.
- Web browsers support three types of bullets—a **solid disc**, a **circle**, and a **square**
- You can choose which bullet to use for your unordered list by specifying a type attribute.
- For example, you can change the bullet shape for that specific list item by specifying `type="shape"`.
 - For example, `<ul type="square">` means that list items is bulleted with a square.

HTML Unordered Lists

• Example 2

```
Notepad
<html>
<body>
<h4>Disc bullets list:</h4>
<ul type="disc">
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ul>
<h4>Circle bullets list:</h4>
<ul type="circle">
  <li>Apples</li>
  <li>Bananas</li>
  <li>Lemons</li>
  <li>Oranges</li>
</ul> </body> </html>
```

This HTML code
looks
in a browser as:



HTML Ordered Lists

- An ordered list starts with the `` tag. Each list item starts with the `` tag
- The list items are marked with numbers.

Example 1:

```
<ol>  
  <li>Coffee</li>  
  <li>Milk</li>  
</ol>
```

This HTML code looks in a browser as:

1. Coffee
2. Milk

HTML Ordered Lists

Example: 2

```
<html>
  <body>
    <h4>Numbered list:</h4>
    <ol>
      <li>Apples</li>
      <li>Bananas</li>
      <li>Lemons</li>
      <li>Oranges</li>
    </ol>
    <h4>Letters list:</h4>
    <ol type="A">
      <li>Apples</li>
      <li>Bananas</li>
      <li>Lemons</li>
      <li>Oranges</li>
    </ol>
  </body>
</html>
```

This HTML code looks in a browser as:

Numbered list:

1. Apples
2. Bananas
3. Lemons
4. Oranges

Letters list:

- A. Apples
- B. Bananas
- C. Lemons
- D. Oranges

HTML Definition Lists:

- A definition list is a list of items, with a description of each item.
- The `<dl>` tag defines a definition list.
- The `<dl>` tag is used in conjunction with `<dt>` (**defines** the item in the list) and `<dd>` (**describes** the item in the list).

HTML Definition Lists:

Example

```
<html>
  <body>
    <h4>A Definition List:</h4>
    <dl>
      <dt>Coffee</dt>
      <dd>Black hot drink</dd>
      <dt>Milk</dt>
      <dd>White cold drink</dd>
    </dl>
  </body>
</html>
```

This HTML code looks in a browser as:

A Definition List:
Coffee
Black hot drink
Milk
White cold drink

Nested list:

Example:

```
<html>
  <body>
    <h4>A nested List :</h4>
    <ul>
      <li>Coffee</li>
      <li>Tea
        <ul>
          <li>Black tea</li>
          <li>Green tea</li>
        </ul>
      </li>
      <li>Milk</li>
    </ul>
  </body>
</html>
```

A nested List :

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

Outline

1. HTML Page
2. LINKS
3. Frames and List
4. **TABLES**
5. Advanced HTML
6. DHTML

Tables

- HTML tables allow web developers to arrange data into rows and columns.
- The `<table>` tag defines an HTML table.
- Each table row is defined with a `<tr>` tag.
- In each row of the table, you can create the (Table header) tags. Each table header is defined with a `<th>` tag.
- In each row of the table, you can create the (Table Data) tags. Each table data/cell is defined with a `<td>` tag.
- **Note:** The `<td>` elements are the data containers of the table. They can contain all sorts of HTML elements; text, images, lists, other tables, etc.

Tables

A simple HTML table

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

By default, the text in `<th>` elements are bold and centered.

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

By default, the text in `<td>` elements are regular and left-aligned.

Tables

To make a cell span more than one column, use the **colspan** attribute:
In this example colspan="2" means will accept data in two columns

```
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gates</td>
    <td>55577720 </td>
    <td>55588830 </td>
  </tr>
</table>
```

Name	Telephone	
Bill Gates	55577720	55588830

Tables

To make a cell span more than one row, use the **rowspan** attribute:
In this example **rowspan="2"** means will accept data in two rows

```
<table style="width:100%">
  <tr>
    <th>Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td> 55577720 </td>
  </tr>
  <tr>
    <td> 55588830 </td>
  </tr>
</table>
```

Name:	Bill Gates
Telephone:	55577720 55588830

Tables

To add a caption to a table, use the `<caption>` tag.

Note: The `<caption>` tag must be inserted immediately after the `<table>` tag.

```
<table style="width:100%">
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$50</td>
  </tr>
</table>
```

Monthly Savings	
Month	Savings
January	\$100
February	\$50

Tables

Tables for No tabular Information:

- Some HTML authors use tables to present no tabular information.
- For example, because links can be included in table cells, some authors use a table with no borders to create "one" image from separate images.
- However using table borders with images can create an impressive display as well.

Tables

Tables Formatting

- The following parameters (attributes) are used for tables formatting:

Parameter	Meaning
border="n"	Width of enclosed area surrounding table; if border="0", this also eliminates the grid lines within the table itself.
cellspacing="n"	Spacing between individual cells.
cellpadding="n"	Space between border and contents of cell.
width="n"	Desired width; overrides automatic width calculations (value or percentage).

Tables

Add a Border

- To add a border to a table, use the CSS border property:
- Example

```
table, th, td {  
  border: 1px solid black;  
}
```

Collapsed Borders

- To let the borders collapse into one border, add the CSS border-collapse property:
- Example

```
table, th, td {  
  border: 1px solid black;  
  border-collapse: collapse;  
}
```

Tables

Add Cell Padding

- Cell padding specifies the space between the cell content and its borders.
- If you do not specify a padding, the table cells will be displayed without padding.
- To set the padding, use the CSS padding property:
- Example

```
th, td {  
    padding: 15px;  
}
```


Tables

Tables Formatting

- You can change the table background color or the table border color using the parameters, bgcolor and bordercolor:

Tables

- **Nested tables:**

- The nested tables or 'tables within table' is a concept used while creating bigger and complex tables.
- A table can be created within another table by simply using the table tags like <table>, <tr>, <td>, etc., to create our nested table.

```
<body>
<table border="5px" bordercolor="#8707B0">
<tr>
<td>Left side of the main table</td>
<td>
<table border="5px" bordercolor="#F35557">
<h4 align="center">Nested Table</h4>
<tr>
<td>nested table C1</td>
<td>nested table C2</td>
</tr>
<tr>
<td>nested table</td>
<td>nested table</td>
</tr>
</table>
</td>
</tr>
</table>
```

Left side of the main table	Nested Table	
	nested table C1	nested table C2
	nested table	nested table

HTML <div> Tag

- The <div> tag defines a division or a section in an HTML document.
- The <div> tag is used as a container for HTML elements - which is then styled with CSS or manipulated with JavaScript.
- The <div> tag is easily styled by using the class or id attribute.
- Any sort of content can be put inside the <div> tag!
- **Note:** By default, browsers always place a line break before and after the <div> element.

The div element

This is a heading in a div element

This is some text in a div element.

This is some text outside the div element.

```
<!DOCTYPE html>
<html>
<head>
<style>
.myDiv {
  border: 5px outset red;
  background-color: lightblue;
  text-align: center;
}
</style>
</head>
<body>

<h1>The div element</h1>

<div class="myDiv">
  <h2>This is a heading in a div element</h2>
  <p>This is some text in a div element.</p>
</div>

<p>This is some text outside the div element.</p>

</body>
</html>
```

Color Values

- HTML colors are specified with:
 - Predefined color names
 - RGB values
 - HEX values
 - HSL values
 - RGBA values
 - HSLA values

Color Values

Color Names

- In HTML, a color can be specified by using a color name:
- HTML supports [140 standard color names](#).

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

<h1 style="background-color:Tomato;">Tomato</h1>
<h1 style="background-color:Orange;">Orange</h1>
<h1 style="background-color:DodgerBlue;">DodgerBlue</h1>
<h1 style="background-color:MediumSeaGreen;">MediumSeaGreen</h1>
<h1 style="background-color:Gray;">Gray</h1>
<h1 style="background-color:SlateBlue;">SlateBlue</h1>
<h1 style="background-color:Violet;">Violet</h1>
<h1 style="background-color:LightGray;">LightGray</h1>

</body>
</html>
```

Tomato

Orange

DodgerBlue

MediumSeaGreen

Gray

SlateBlue

Violet

LightGray

Color Values

Color Names

Background Color

- You can set the background color for HTML elements:

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <h1 style="background-color: ■DodgerBlue;">Good Morning</h1>
6
7  <p style="background-color: ■Tomato;">
8  Always start your day with a simle :).
9  </p>
10
11 </body>
12 </html>
13
...
```

Good Morning

Always start your day with a simle :).

Color Values

Color Names

Text Color

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <h3 style="color: ■ Tomato;">Good Morning</h3>
6
7  <p style="color: ■ DodgerBlue;">Always start your day with a smile:) </p>
8
9  <p style="color: ■ MediumSeaGreen;">Try to make someone happy everyday.</p>
10
11 </body>
12 </html>
```

Good Morning

Always start your day with a smile:)

Try to make someone happy everyday.

Color Values

Color Names

Border Color

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

<h1 style="border: 2px solid Tomato;">Good Morning</h1>

<h1 style="border: 2px solid DodgerBlue;">Good Morning</h1>

<h1 style="border: 2px solid Violet;">Good Morning</h1>

</body>
</html>
```










Good Morning

Good Morning

Good Morning

Color Values

- HTML colors defined using a **hexadecimal notation (HEX)** for the combination of Red, Green, and Blue color values (RGB), the lowest value that can be given to one of the light sources is 0 (in HEX: 00). The highest value is 255 (in HEX: FF). HEX values are specified as 3 pairs of two-digit numbers, starting with a # sign

Color	Color HEX	Color RGB
	# 000000	rgb(0,0,0)
	# FF0000	rgb(255,0,0)
	# 00FF00	rgb(0,255,0)
	# 0000FF	rgb(0,0,255)
	# FFFF00	rgb(255,255,0)
	# 00FFFF	rgb(0,255,255)
	# FF00FF	rgb(255,0,255)
	# CoCoCo	rgb(192,192,192)
	# FFFFFFFF	rgb(255,255,255)

16 Million Different Colors

- The combination of Red, Green, and Blue values from 0 to 255, gives more than 16 million different colors (256 x 256 x 256).

Color Values

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

<p>Same as color name "Tomato":</p>

<h1 style="background-color:rgb(255, 99, 71);">rgb(255, 99, 71)</h1>
<h1 style="background-color:#ff6347;">#ff6347</h1>
<h1 style="background-color:hsl(9, 100%, 64%);">hsl(9, 100%, 64%)</h1>

<p>Same as color name "Tomato", but 50% transparent:</p>
<h1 style="background-color:rgba(255, 99, 71, 0.5);">rgba(255, 99, 71, 0.5)
</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">hsla(9, 100%, 64%, 0.5)
</h1>

<p>In addition to the predefined color names, colors can be specified using
RGB, HEX, HSL, or even transparent colors using RGBA or HSLA color values.</p>

</body>
</html>
```

Same as color name "Tomato":

rgb(255, 99, 71)

#ff6347

hsl(9, 100%, 64%)

Same as color name "Tomato", but 50% transparent:

rgba(255, 99, 71, 0.5)

hsla(9, 100%, 64%, 0.5)

In addition to the predefined color names, colors can be specified using RGB, HEX, HSL, or even transparent colors using RGBA or HSLA color values.

Outline

1. HTML Page
2. LINKS
3. Frames and List
4. TABLES
5. **Advanced HTML**
6. DHTML

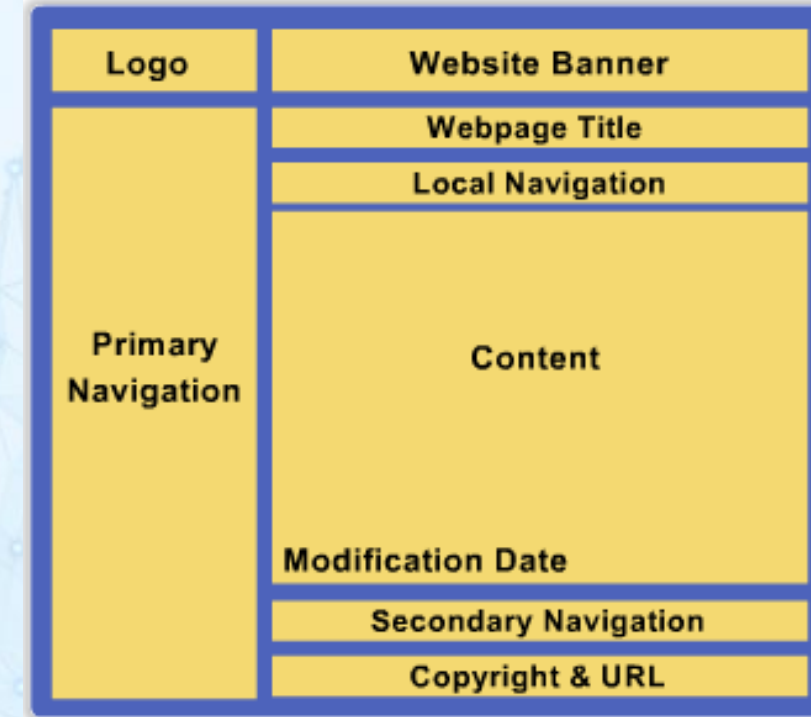
Advanced HTML

HTML Layouts

- Web page layout is very important to make your website look good
- Design your webpage layout very carefully

Website Layouts

- Most websites have put their content in multiple columns (formatted like a magazine or newspaper).
- Multiple columns are created by using <table> or <div> tags.
- Some CSS are normally also added to position elements, or to create backgrounds or colorful look for the pages.



1-HTML Layouts - Using Tables

- The simplest way of creating layouts is by using the HTML `<table>` tag.
- **Note:** Even though it is possible to create nice layouts with HTML tables, tables were designed for presenting tabular data - NOT as a layout tool!

1-HTML Layouts - Using Tables

Example:

This example uses a table with 3 rows and 2 columns -the first and last row spans both columns using the colspan attribute:

```
<!DOCTYPE html>
<html>
<body>
<table width="500" border="0">
<tr>
<td colspan="2" style="background-color:#FFA500;">
<h1>Main Title of Web Page</h1>
</td>
</tr>
<tr>
<td colspan="2" style="background-color:#FFD700;width:100px;text-align:top;">
<b>Menu</b><br />
HTML<br />
CSS<br />
JavaScript
</td>
<td style="background-color:#EEEEEE;height:200px;width:400px;text-align:top;">
Content goes here</td>
</tr>
<tr>
<td colspan="2" style="background-color:#FFA500;text-align:center;">
Copyright © 2011 W3Schools.com</td>
</tr>
</table>
</body>
</html>
```

Main Title of Web Page

Menu
HTML
CSS
JavaScript

Content goes here

Copyright © 2011 W3Schools.com

2- HTML Layouts - Using Div Elements

- The div element is a block level element used for grouping HTML elements.

Example:

- Using five div elements to create a multiple column layout, creating the same result as in the previous example

```
<!DOCTYPE html>
<html>
<body>
<div id="container" style="width:500px">
<div id="header" style="background-color:#FFA500;">
  <h1 style="margin-bottom:0;">Main Title of Web Page</h1></div>
<div id="menu" style="background-
color:#FFD700;height:200px;width:100px;float:left;">
<b>Menu</b><br />
HTML<br />
CSS<br />
JavaScript</div>
<div id="content" style="background-
color:#EEEEEE;height:200px;width:400px;float:left;">
Content goes here</div>
<div id="footer" style="background-color:#FFA500;clear:both;text-
align:center;">
Copyright © 2011 W3Schools.com</div>
</div>
</body>
</html>
```

Main Title of Web Page

Menu
HTML
CSS
JavaScript

Content goes here

Copyright © 2011 W3Schools.com

Metadata

Metadata is information about data.

- The `<meta>` tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsing.
- Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.
- The `<meta>` tag always goes inside the head element.
- The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.



Keywords for Search Engines

- Some search engines will use the name and content attributes of the meta element to index your pages.
- The following meta element defines a description of a page:
`<meta name="description" content="Free Web tutorials on HTML, CSS, XML" />`
- The following meta element defines keywords for a page:
`<meta name="keywords" content="HTML, CSS, XML" />`
- The intention of the **name** and **content** attributes is to describe the content of a page

The HTML script Element

- The `<script>` tag is used to define a client-side script, such as a JavaScript.
- The script element either contains scripting statements or it points to an external script file through the `src` attribute.
- Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.

The HTML script Element

Example :

- The script writes Hello World! to the HTML output:

```
<html>
  <body>
    <script type="text/javascript">
      document.write("Hello World!")
    </script>
  </body>
</html>
```

- **The result is:**

Hello World!

- **Note:** If the "src" attribute is present, the <script> element must be empty.

The HTML script Element

Example :

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

<h1>The script element</h1>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>

</body>
</html>
```

The script element

Hello JavaScript!

The HTML Noscript Element

- The `<noscript>` tag defines an alternate content to be displayed to users that have disabled scripts in their browser or have a browser that doesn't support client-side scripting.
- The `<noscript>` element can be used in both `<head>` and `<body>`. When used inside `<head>`, the `<noscript>` element could only contain `<link>`, `<style>`, and `<meta>` elements.
- The `noscript` element can contain all the elements that you can find inside the `body` element of a normal HTML page.

The HTML Noscript Element

- Example :

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

<h1>The noscript element</h1>

<p>A browser with JavaScript disabled will show the text inside the noscript
element ("Hello World!" will not be displayed).</p>

<script>
document.write("Hello World!")
</script>
<noscript>Sorry, your browser does not support JavaScript!</noscript>

</body>
</html>
```

The noscript element

A browser with JavaScript disabled will show the text inside the noscript element ("Hello World!" will not be displayed).

Hello World!

- A browser without support for JavaScript will show the text in the noscript element

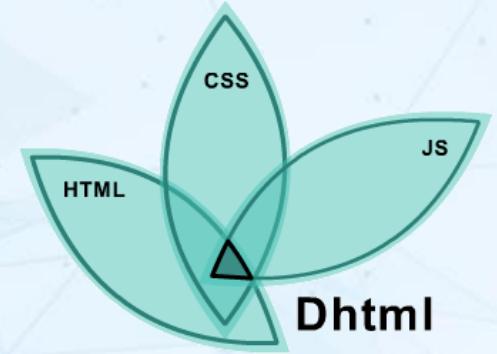
Sorry, your browser does not support JavaScript!

Outline

1. HTML Page
2. LINKS
3. TABLES
4. Frames and List
5. Advanced HTML
6. DHTML

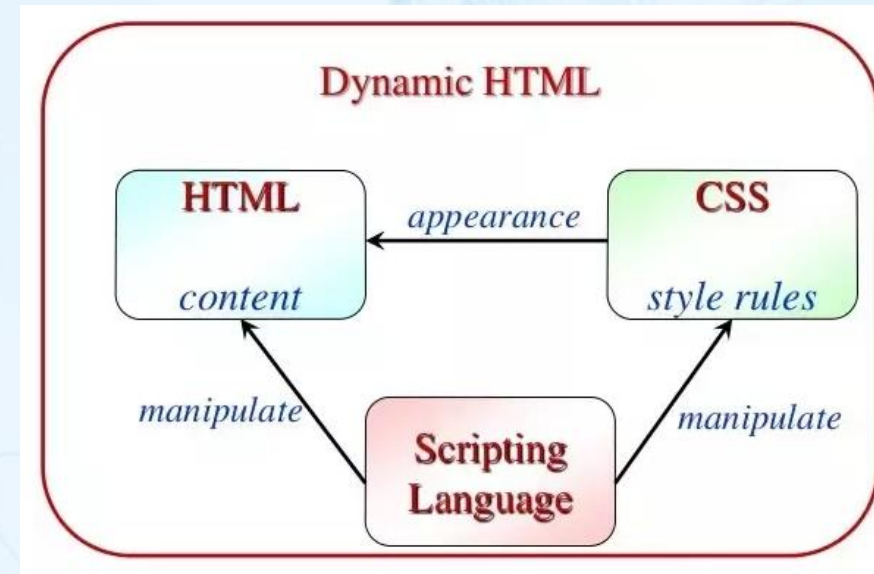
DHTML

- DHTML stands for Dynamic Hypertext Markup Language.
 - DHTML is NOT a language or a web standard.
 - It is NOT a markup language
 - It is NOT a scripting language like JavaScript.
 - DHTML is not one particular, technology or set of features. It includes several technologies and describes how these technologies interact.
 - Web pages built with DHTML are richer and more interactive, react faster, and don't use much bandwidth.



DHTML

- **DHTML** refers to the use of three languages together in Web design: HTML, CSS, and JavaScript.
 - **HTML** is used for the basic structure of the document
 - **Cascading Style Sheets (CSS)** to define the presentation and style of the document.
 - **JavaScript** to manipulate the Document Object Model (DOM)
- As a result, the new tags supported in HTML trigger additional JavaScript events enabling more control over the content being rendered.



DHTML

HTML

- HTML 4 standards have rich support for dynamic content:
 - HTML supports JavaScript
 - HTML supports the Document Object Model (DOM)
 - HTML supports HTML Events
 - HTML supports Cascading Style Sheets (CSS)
- DHTML is about using these features, to create dynamic and interactive web pages.

DOM

- The HTML DOM describes the Document Object Model for HTML.
- The HTML DOM defines a standard way for accessing and manipulating HTML documents.
- DHTML is about using the DOM to access and manipulate HTML elements

DHTML

CSS

- CSS defines how to display HTML elements.
- DHTML is about using JavaScript and the HTML DOM to change the style and positioning of HTML elements.

JavaScript

- JavaScript is the most popular scripting language on the internet, and it works in all major browsers.
- DHTML is about using JavaScript to control, access and manipulate HTML elements

DHTML

Advantages of DHTML:

1. DHTML makes documents dynamic; that:
 - Allow the designer to control how the HTML displays Web pages' content.
 - React and change with the actions of the visitor.
 - Can exactly position any element in the window, and change that position after the document has loaded.
 - Can hide and show content as needed.
2. DHTML allows any HTML element (any object on the screen that can be controlled independently using JavaScript) in Internet Explorer to be manipulated at any time, turning plain HTML into dynamic HTML.
3. With DHTML, changes occur entirely on the client-side (on the user's browser)
4. Using DHTML gives the author more control over how the page is formatted and how content is positioned on the page.



EELU

الجامعة المصرية للتعليم الإلكتروني الأهلية
THE EGYPTIAN E-LEARNING UNIVERSITY

THANK YOU FOR WATCHING

QUESTIONS?

