Outline

- Introduction to HTML5
- Components of HTML5 document
- Create Web Pages
- Create and use hyperlinks to help users navigate web pages
- Add different tags in a web page
- Create tables
- Create and use forms to get user input

Introduction to HTML5

- HTML stands for Hyper Text Markup Language.
- HTML is the standard markup language for creating Web.
- 1994: World Wide Web Consortium created "W3C": Goal to maintain and develop standards about how the web should work.
- HTML versions Timeline:

Version	First draft	Candidate recommendation	Recommendation	Retired
HTML5	May 2007	December 2012	October 2014	March 2018
HTML 5.1	December 2012	June 2016	November 2016	January 2021
HTML 5.1 2nd Ed	—	June 2017	October 2017	
HTML 5.2	August 2016	August 2017	December 2017	January 2021
HTML 5.3	December 2017	—	—	January 2021

Introduction to HTML5

- Design of HTML follow the concept of "living standard" is that it is never complete and is always being updated and improved. New features can be added but functionality will not be removed.
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

Introduction to HTML5

- **Weird about browsers:**
  - Browsers don't fail when given invalid HTML
  - Browsers not only don't fail, but they render invalid HTML seemingly "correctly"
- **Q: Why browser doesn't reject poorly written HTML?**
  - There was a (failed) attempt to enforce this, but it was too late the Internet grew too big.

Introduction to HTML5

Anatomy of HTML Page

```
<html>
    <head>
        <title> page title </title>
    </head>
    <body>
        <h1> This is heading</h1>
        <p> This is a paragraph </p>
        <p> This is another paragraph </p>
    </body>
</html>
```

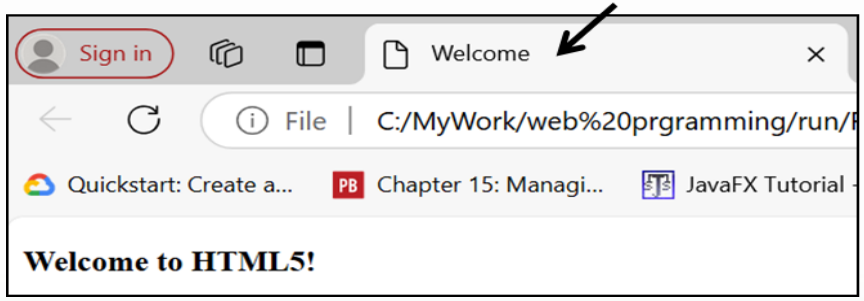
First HTML5 Example

❑ Start Tags and End Tags

- ❖ **HTML5** documents **delimit** most **elements** with a **start tag and end tag**.
- ❖ **Start tag:** consists of the element name in angle brackets, **(for example: <html>).**
- ❖ **End tag:** consists of the element name preceded by a forward slash (/) in angle brackets **(for example: </html>).**
- ❖ There are **tags** called **"void elements"** that **do not have an end tags, (for example:
).**
- ❖ Many start tags have attributes that provide additional information about an element, which browsers use to determine how to process the element.
- ❖ Each **attribute** has a **name** and a **value** separated by an equals sign (=), **(for example: charset = "utf-8")**

```
<!DOCTYPE html>
<!-- FirstHtml.html  -->
<html>
    <head>
        <meta charset = "utf-8">
        <title>Welcome</title>
    </head>

    <body>
        <p>
            <strong>
                Welcome to HTML5!
            </strong>
            <br>
        </p>
    </body>
</html>
```

Tab shows content of the title element



Introduction to HTML5

HTML Element Reference - By Category

Basic elements:

Element	Description
<!DOCTYPE>	Defines the document type
<html>	Defines an HTML document
<head>	Contains metadata/information for the document
<title>	Defines a title for the document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings
<p>	Defines a paragraph
 	Inserts a single line break
<hr>	Defines a thematic change in the content
<!--...-->	Defines a comment

Introduction to HTML5

HTML Element Reference - By Category

Formatting:

Element	Description
<abbr>	Defines an abbreviation or an acronym
<address>	Defines contact information for the author/owner of a document/article
<b>	Defines bold text
<code>	Defines a piece of computer code
<del>	Defines text that has been deleted from a document
	Defines emphasized text
<ins>	Defines a text that has been inserted into a document
<kbd>	Defines keyboard input
<mark>	Defines marked/highlighted text
<pre>	Defines preformatted text
<small>	Defines smaller text
	Defines important text
<sub>	Defines subscripted text
<sup>	Defines superscripted text
<time>	Defines a specific time (or datetime)
<u>	Defines some text that is unarticulated and styled differently from normal text
<var>	Defines a variable
<wbr>	Defines a possible line-break

Introduction to HTML5

HTML Element Reference - By Category

Audio / Video:

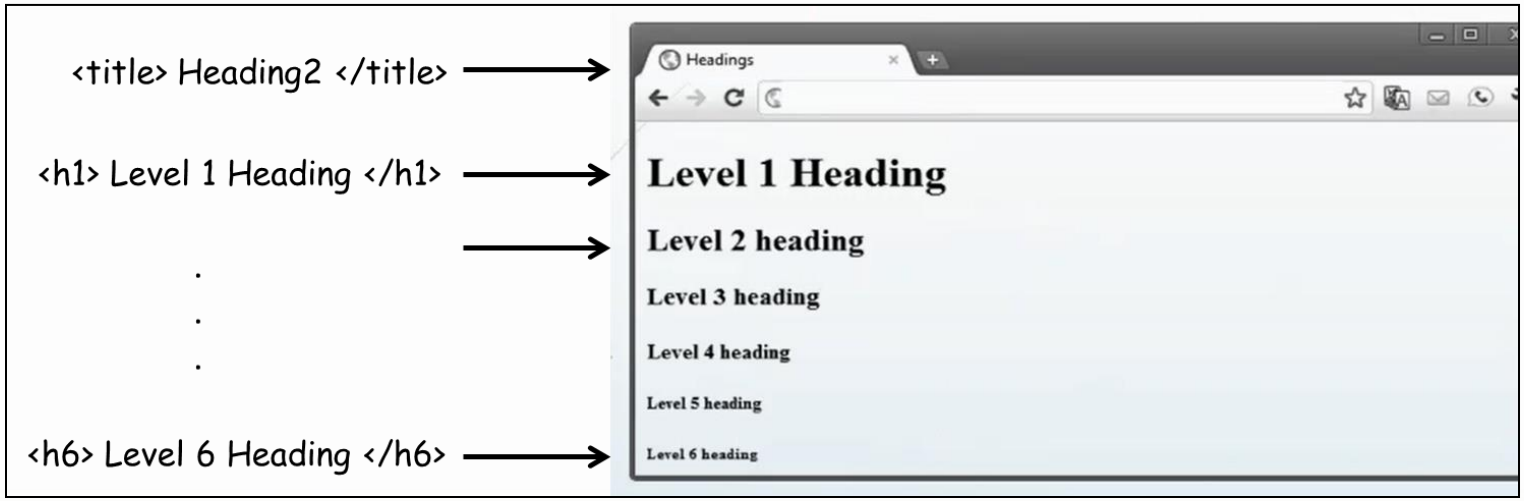
Element	Description
<audio>	Defines sound content
<source>	Defines multiple media resources for media elements (<video>, <audio> and <picture>)
<track>	Defines text tracks for media elements (<video> and <audio>)
<video>	Defines a video or movie

Link:

Element	Description
<a>	Defines a hyperlink
<nav>	Defines navigation links

First HTML5 Example

```
<!DOCTYPE html>
<!-- HTML01.html  -->
<html>
      <head>
            <title>Headings</title>
      </head>
      <body>
            <h1>Level1 Heading</h1>      <!-- biggest size -->
            <h2>Level1 Heading</h2>
            <h3>Level1 Heading</h3>
            <h4>Level1 Heading</h4>
            <h5>Level1 Heading</h5>
            <h6>Level1 Heading</h6>      <!-- smallest size -->
      </body>
</html>
```



First HTML5 Example

```
<!DOCTYPE html>
<!-- HTML01.html  -->
<html>
      <head>
            <title>Headings</title>
      </head>
      <body>
            <h1>Level1 Heading</h1>
            <h2>Level1 Heading</h2>
            <h3>Level1 Heading</h3>
            <h4>Level1 Heading</h4>
            <h5>Level1 Heading</h5>
            <h6>Level1 Heading</h6>
      </body>
</html>
```

- The **<!DOCTYPE html>** declaration defines that this document is an HTML5 document
- The **<html>** element is the **root element** of an HTML page
- The **<head>** element contains **meta information** about the HTML page
- The **<title>** element specifies a **title** for the HTML **page** (which is shown in the browser's title bar or in the page's tab)
- The **<body>** element defines the document's body, and is a **container for all the visible contents**, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- The **<h1>** element defines a **large heading**

HTML Elements

- Types of HTML elements

HTML element is categorized by the HTML spec into one of 3 categories:

- **Block** (large blocks of content, has height and width):

  <p>, <h1>, <blockquote>, <ol>, <ul>, <table>

- **Inline** (small amount of content, no height or width):

  **<a>, , ,
**

inline block (inline content with height and width):

<img>

- **Metadata:(information about the page, usually not visible):**

  <title>, <meta>

<a> … Defines a hyperlink (link to other pages)
 … Emphasized text in a document
 … order list
 … unordered list

HTML Elements

- An HTML element is defined by a **start tag**, some content, and an **end tag**:

  <tagname> Content goes here... </tagname>

- The HTML element is everything from the start tag to the end tag:

```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```

- HTML is Not Case Sensitive

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
	My first paragraph.	
 	none	none

Note:

- **Some HTML elements have no content (like the
 element).**
- **These elements are called empty elements.**
- **Empty elements (void elements) do not have an end tag!**

Nested HTML Elements

- HTML elements can be nested (this means that elements can contain other elements).
- All HTML documents consist of nested HTML elements.
- The following example contains four HTML elements:

(<html>, <body>, <h1> and <p>):

```
<!DOCTYPE html>
<html>
  <body>
    <h1>
        My First Heading
    </h1>
    <p>
        My first paragraph.
    </p>
  </body>
</html>
```

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"**
- **The title Attributes:**
  - The title attribute defines some extra information about an element.
  - The value of the title attribute will be displayed as a tooltip when you mouse over the element:
- **The lang Attributes:**
  - You should always include the lang attribute inside the <html> tag, to declare the language of the Web page.
  - This is meant to assist search engines and browsers.
  - The following example specifies English as the language:

```
<!DOCTYPE html>
<!-- attribute.html -->
<html lang="en">
    <body>
        <p title="I'm a tooltip">This is a paragraph.</p>
    </body>
</html>
```

HTML Attributes

The href Attribute:

- The **<a>** tag defines a hyperlink.
- The **href (Hypertext REFerence)** attribute specifies the URL of the linked page.
- **Example:** `<a href="https://www.sha.edu.ed.com">Visit SHK academy</a>`

The src Attribute:

- The **<img>** tag is used to embed an image in an HTML page.
- The **src** attribute specifies the path to the image to be displayed.
- **Example:** `<img src="img_girl.jpg">`

The width and height Attributes:

- The **<img>** tag should also contain the width and height attributes, which specify the width and height of the image (in pixels).
- **Example:** `<img src="img_girl.jpg" width="500" height="600">`

The alt Attribute:

- The required **alt** attribute for the **<img>** tag specifies an alternate text for an image, if the image for some reason cannot be displayed.
- This can be due to a slow connection, or an error in the **src** attribute.
- **Example:** `<img src="img_girl.jpg" alt="Girl with a jacket">`

HTML Attributes

- The HTML standard does not require quotes around attribute values.

Good: `<a href="https://www.sha.edu.eg/html/">Visit our HTML tutorial</a>`

Bad: `<a href=https://www.sha.edu.eg/html/>Visit our HTML tutorial</a>`

Must: `<p title="About Shorouk Academy">`

error: `<p title=About Shorouk Academy>`

- Single or Double Quotes?
  - Double quotes around attribute values are the most common in HTML, but single quotes can also be used.
  - In some situations, when the attribute value itself contains double quotes, it is necessary to use single quotes:

```
<p title='John "ShotGun" Nelson'>

<p title="John 'ShotGun' Nelson">
```

HTML Attributes

The Style Attribute

- The HTML **style** attribute is used to add styles to an element, such as color, font, size, and more. The HTML style attribute has the following syntax:
  - `<tagname style="property:value;">`

- Example:

```
<p style="color:red;">This is a red paragraph.</p>
```

- Style summary:

Use the style attribute for styling HTML elements:	
Use background-color	for background color
Use color	for text colors
Use font-family	for text fonts
Use font-size	for text sizes
Use text-align	for text alignment
Use border	for text border

HTML Attributes

❑ The Style Attribute

❖ Fonts:

```
<!-- Set the font for an HTML element -->
<h1 style="font-family:verdana;">This is a heading</h1>
<p style="font-family:courier;">This is a paragraph.</p>
```

❖ Background Color:

```
<!-- Set the background color for a page to → blue: -->
<body style="background-color:blue;">
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
</body>
```

```
<!-- Set background color for two different elements:
<body>
    <h1 style="background-color:blue;">This is a heading</h1>
    <p style="background-color:tomato;">This is a paragraph.</p>
</body>
```

❖ Text Size:

```
<!-- Set the text size for an HTML element
<h1 style="font-size:300%;">This is a heading</h1>
<p style="font-size:160%;">This is a paragraph.</p>>
```

❖ Text Alignment

```
<!-- Set horizontal text alignment for an HTML element
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>>>
```

HTML Attributes

The Style Attribute

- **Color Names :** In HTML, a color can be specified by using a color name:

- **Background Color:**

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>
<p style="background-color:Tomato;">Lorem ipsum...</p>
```

- **Text Color:**

```
<h1 style="color:Tomato;">Hello World</h1>
<p style="color:DodgerBlue;">Lorem ipsum...</p>
<p style="color:MediumSeaGreen;">Ut wisi enim...</p>
```

- **Border Color:**

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

HTML Attributes

The Style Attribute

Color Values:

```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>
<h1 style="background-color:#ff6347;">...</h1>  <!- RGB in hexadecimal value
<h1 style="background-color:hsl(9, 100%, 64%);">...</h1>
<h1 style="background-color:rgba(255, 99, 71, 0.5);">...</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">...</h1>
```

Note:

rgb: stands for red, green, blue

hsl: stands for Hue, Saturation, and Lightness.

rgba: stands for red, green, blue, transparency

hsla: stands for Hue, Saturation, and Lightness, transparency.

HTML Comments

- HTML comments are not displayed in the browser, but they can help document your HTML source code.
- We can add comments to your HTML source by using the following syntax:

```
<!-- Write your comments here -->
```

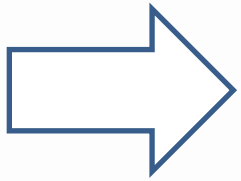
- Example:

```
<!-- This is a comment -->
<p>This is a paragraph.</p>
<!-- Remember to add more information here -->
```

HTML Headings

- HTML headings are titles or subtitles that you want to display on a webpage.
- HTML Headings example:

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



Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

<h1> defines the most important heading.

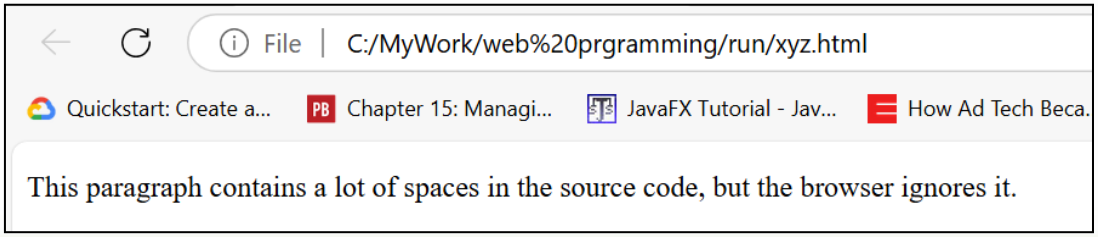
<h6> defines the least important heading.

HTML Paragraphs

- **A paragraph always starts on a new line, and is usually a block of text, and browsers automatically add some white space (a margin) before and after a paragraph.**
- **The browser will automatically remove any extra spaces and lines when the page is displayed**
- The HTML **<p>** element defines a **paragraph**.
- HTML Paragraphs examples:

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

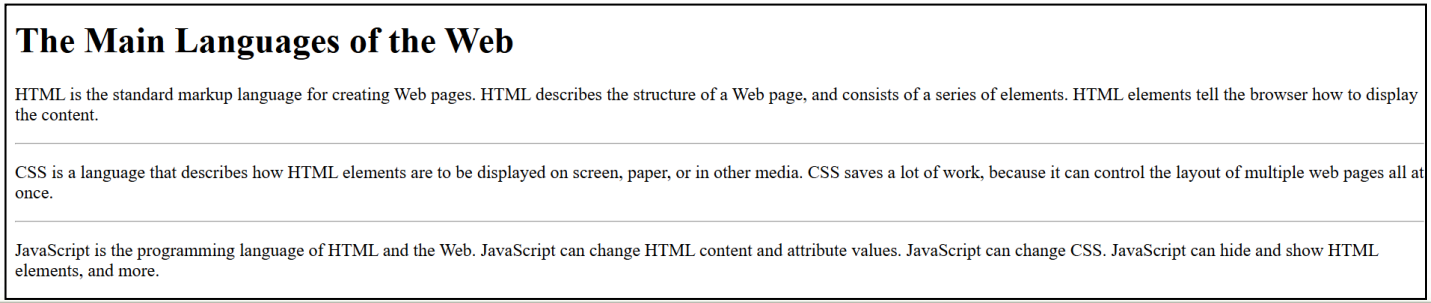
```
<p>
This paragraph
contains a lot of lines
in the source code,
but the browser
ignores it.
</p>
```



HTML Paragraphs

- The HTML **
** element defines a **line break.**
- The **<hr>** (**horizontal rule**) element is used to **separate content.** The **<hr>** tag is an **empty tag,** which means that it has no end tag.

```
<!DOCTYPE html>
<!-- hr.html  -->
<html>
    <body>
        <h1>The Main Languages of the Web</h1>
        <p>
            HTML is the standard markup language for creating Web pages.
            HTML describes the structure of a Web page,
            and consists of a series of elements.
            HTML elements tell the browser how to display the content.
        </p>
        <hr>
        <p>
            CSS is a language that describes how HTML elements are
            to be displayed on screen, paper, or in other media.
            CSS saves a lot of work, because it can control
            the layout of multiple web pages all at once.
        </p>
        <hr>
        <p>
            JavaScript is the programming language of HTML and the Web.
            JavaScript can change HTML content and attribute values.
            JavaScript can change CSS.
            JavaScript can hide and show HTML elements, and more.
        </p>
    </body>
</html>
```



The Main Languages of the Web

HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page, and consists of a series of elements. HTML elements tell the browser how to display the content.

CSS is a language that describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work, because it can control the layout of multiple web pages all at once.

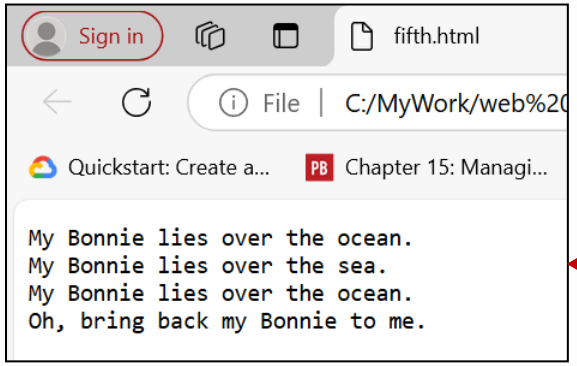
JavaScript is the programming language of HTML and the Web. JavaScript can change HTML content and attribute values. JavaScript can change CSS. JavaScript can hide and show HTML elements, and more.

HTML Paragraphs

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

```
<!DOCTYPE html>
<!-- fifth.html -->
<html>
    <body>
        <pre>
            My Bonnie lies over the ocean.
            My Bonnie lies over the sea.
            My Bonnie lies over the ocean.
            Oh, bring back my Bonnie to me.
        </pre>
    </body>
</html>
```

Without <pre> tag



With <pre> tag

HTML Link

- Links are found in nearly all web pages. Links allow users to click their way from page to page.
- A hyperlink references or links to other resources, such as HTML5 documents and images.
- Web browsers typically underline text hyperlinks and color them blue by default.

1. Link to another webpage or website
2. Link in the same document
3. Link to an email address

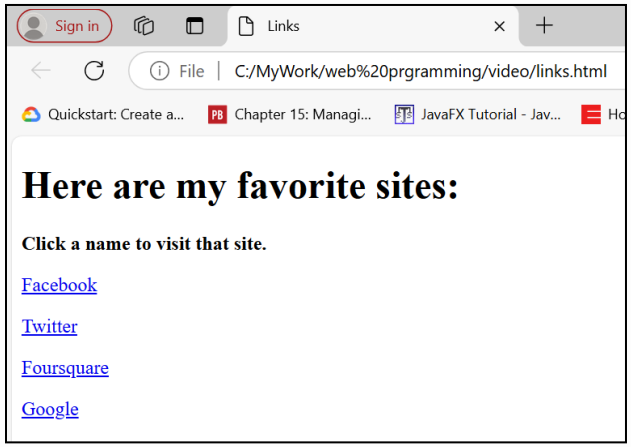
HTML Link

Links (anchors): <a>

- links, or "anchors", to other pages (inline)
- HTML Links – Syntax: **link text**
- The most important attribute of the **<a>** element is the **href** attribute, which indicates the link's destination.
- The **link text** is the part that will be visible to the reader.
- **Anchors are inline elements; must be placed in a block element such as <p> or <h1>**
- Clicking on the link text, will send the reader to the specified URL address
- Example: **Visit SHK academy!**
- 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

HTML Link

- Link to another webpage or website




```
<!DOCTYPE html>
<!-- links.html -->
<!-- Linking to other web pages. -->
<html>
    <head>
        <meta charset = "utf-8">
        <title>Links</title>
    </head>
    <body>
        <h1>Here are my favorite sites:</h1>
        <p><strong>Click a name to visit that site.</strong></p>
        <!-- create four text hyperlinks -->
        <p><a href = "http://www.facebook.com">Facebook</a></p>
        <p><a href = "http://www.twitter.com">Twitter</a></p>
        <p><a href = "http://www.foursquare.com">Foursquare</a></p>
        <p><a href = "http://www.google.com">Google</a></p>
    </body>
</html>
```

HTML Link

❑ Link in same document

- HTML links can be used to create bookmarks, so that readers can jump to specific parts of a web page.
- Links can help users navigate a single web page. Locations within web pages can be marked for direct access by links on the same page. Intra-document hyperlinks include such familiar features as
  - Back to Top links.
  - Tables of contents.

Note: In this case, id attribute of the anchor tag is used to navigate the web page: first.html

HTML Link

❑ Link in same document

Hash:

- ❖ A hash - `#` within a hyperlink specifies an **HTML element id** to which the window should be scrolled.
- ❖ href="#some-id" would scroll to an element on the current page such as .
- ❖ Example:

```
<h1 id="sec">Class 10-Feb-25</h1>
<a href="#sec">GOTo Class 10-Feb-25</a>
```

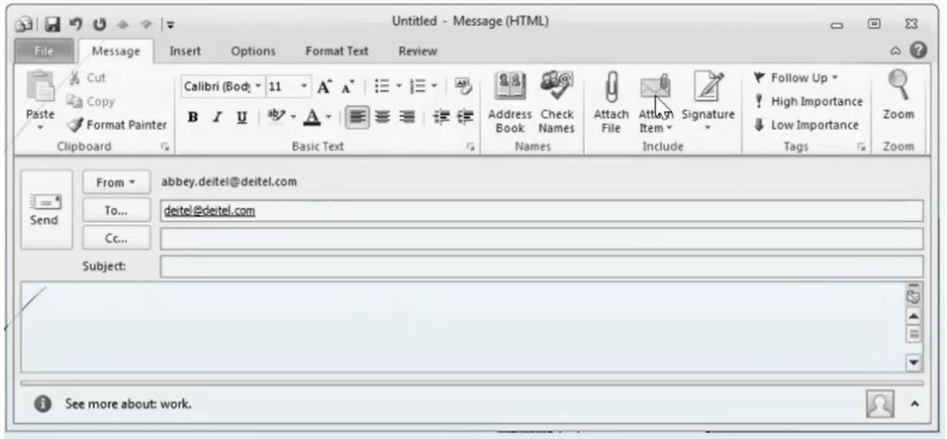
```
<!DOCTYPE html>
<!-- first.html -->
<!-- Linking to other web pages. -->
<html>
    <head>
        <title>second</title>
    </head>
    <body>
        <h1 id="top">At the top of the Page!!</h1>
        <h1>Class 26-Jan-25</h1>
        <h2>First Lecture</h2>
        <h3>First Lecture</h3>
        <h4>First Lecture</h4>
        <h5>First Lecture</h5>
        <h6>First Lecture</h6>

        <h1 id="sec">Class 10-Feb-25</h1>
        <h2>Third Lecture</h2>
        <h3>Third Lecture</h3>
        <h4>Third Lecture</h4>
        <h5>Third Lecture</h5>
        <h6>Third Lecture</h6>

        <h1>Class 17-Feb-25</h1>
        <h2>Fourth Lecture</h2>
        <h3>Fourth Lecture</h3>
        <h4>Fourth Lecture</h4>
        <h5>Fourth Lecture</h5>
        <h6>Fourth Lecture</h6>

        <h1>Class 24-Feb-25</h1>
        <h2>Fifth Lecture</h2>
        <h3>Fifth Lecture</h3>
        <h4>Fifth Lecture</h4>
        <h5>Fifth Lecture</h5>
        <h6>Fifth Lecture</h6>

        <div id="div2">
            <a href="https://www.sha.edu.eg/">ClickMe</a>
            <br>
            <br>
            <a href="#top">GOToTop</a>
            <br><br>
            <a href="#sec">GOTo Class 10-Feb-25</a>

        </div>
    </body>
</html>
```

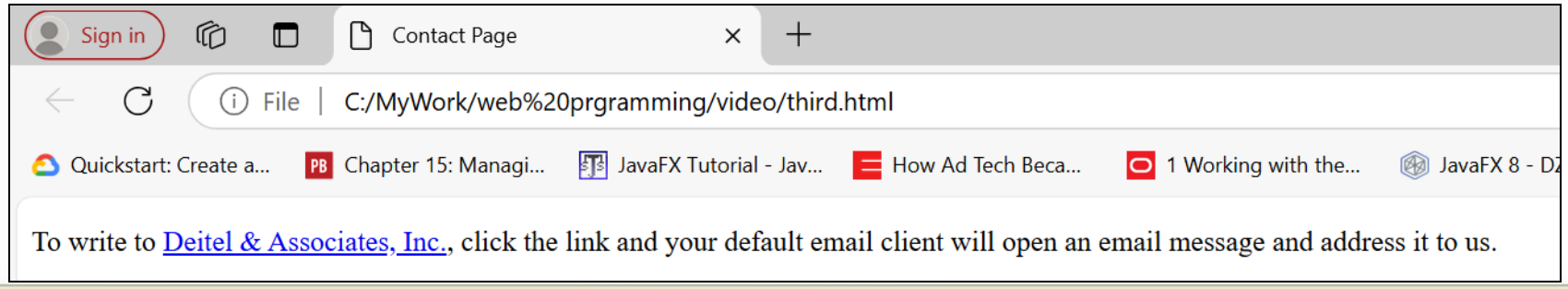
HTML Link

❑ Hyperlinking to an E-Mail Address

- ❖ Anchors can link to an e-mail address using a mailto: URL
- ❖ When a user clicks this type of anchored link, most browsers launch the default e-mail program (e.g., Microsoft Outlook or Apple Mail) to enable the user to write an e-mail message to the linked address.

```
<!DOCTYPE html>
<!-- third.html -->
<!-- Linking in the page pages. -->
<html>
   <head>
      <meta charset = "utf-8">
      <title>Contact Page</title>
   </head>
   <body>
      <p>
         To write to <a href = "mailto:deitel@deitel.com">
         Deitel & Associates, Inc.</a>, click the link and your default email
         client will open an email message and address it to us.
      </p>
   </body>
</html>
```





Images

Images Element Reference:

Element	Description
<img>	Defines an image
<map>	Defines a client-side image map
<area>	Defines an area inside an image map
<figcaption>	Defines a caption for a <figure> element
<figure>	Specifies self-contained content
<picture>	Defines a container for multiple image resources
<svg>	Defines a container for SVG graphics

Images

- HTML images are defined with the <img> tag.
  - The source file (**src**), alternative text (**alt**), **width**, and **height** are provided as attributes:
  - **Width** and **height** are optional attributes
    - If omitted, the browser uses the image's actual width and height
    - Images are measured in pixels
  - Example:

```
<img src="smiley.jpg" alt="smiley face" width="104" height="142">
```

- Image can used as a link if <img> placed in an <a> anchor tag:

```
<a href="https://courses.cs.washington.edu/courses/cse154/20sp/">
  <img src="img/cse154logo.png" alt="CSE154 Course Logo" title="Logo">
</a>
```

Images

- Every **img** element **must have** an **alt** attribute, which contains text that is displayed if the client cannot render the image
  - The **alt** attribute makes web pages more accessible to users with disabilities.
  - How **alt** works:
    - **Screen readers:** These devices or software programs read aloud the content of a webpage, including text, headings, and images. However, they can't "see" images. So, the alt attribute provides a description of the image, which the screen reader can read out loud to the user.
    - **Visual impairments:** For users with low vision or who are blind, the alt attribute ensures that they can understand the content of the image. Without it, they might miss out on important context or information that the image conveys.
    - **Search engines:** Useful for search engines to better understand the content of an image and can improve SEO (Search Engine Optimization).

Images

```
<!-- nav.html -->
<html>
   <head>
      <meta charset = "utf-8">
      <title>Navigation Bar</title>
   </head>
   <body>
      <p>
         <a href="links.html">
           <img src = "buttons/links.jpg" width = "123" height = "83" alt = "Links">
         </a>
         <a href="list.html">
           <img src = "buttons/list.jpg" width = "123" height = "83" alt = "List of Features">
         </a>
         <a href="contact.html">
           <img src = "buttons/contact.jpg" width = "123" height = "83" alt = "Contact Me">
         </a>
         <a href="table1.html">
           <img src = "buttons/table.jpg" width = "123" height = "83" alt = "Tables Page">
         </a>
         <a href="form.html">
           <img src = "buttons/form.jpg" width =  "123" height = "83" alt = "Feedback Form">
         </a>
      </p>
   </body>
</html>
```

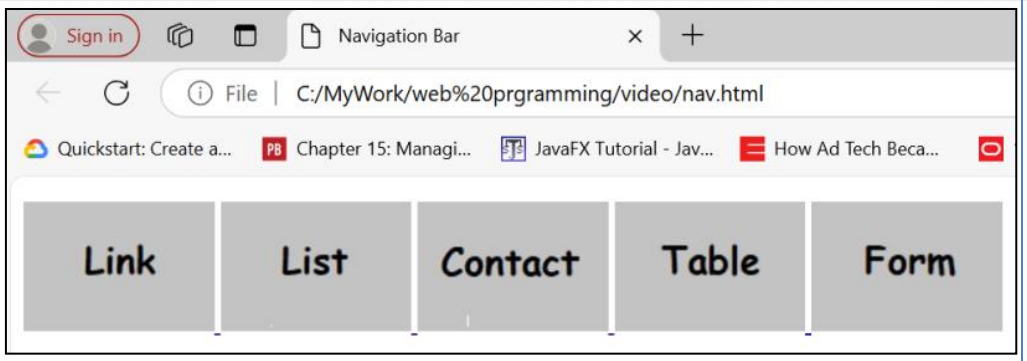


Image used as a link

HTML Favicon

- A favicon is a small image displayed next to the page title in the browser tab.
- We can use any image as a favicon.
- We can also create our own favicon on sites like https://www.favicon.cc.
- Example:

```
<link rel="icon" href="demo_icon.gif" type="image/gif" sizes="16x16">
```

```
<!DOCTYPE html>
<!-- Favicon.html -->

<html>
<head>
  <title>My Page Title</title>
  <link rel="icon" type="image/x-icon" href="app-expertise-tracker.ico">
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Lists

Lists Element Reference:

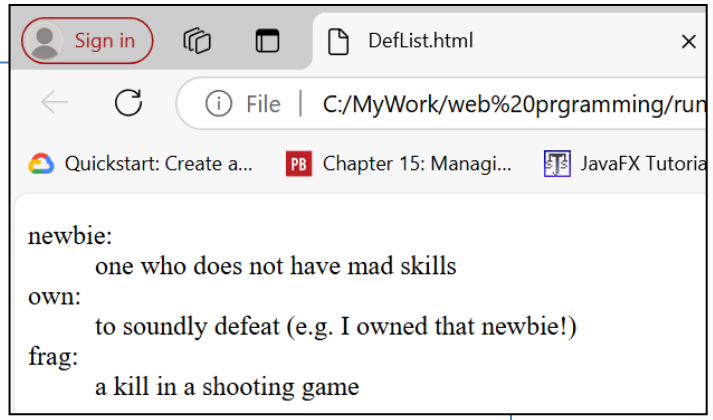
Element	Description
	Defines an unordered list
	Defines an ordered list
	Defines a list item
<dl>	Defines a description list
<dt>	Defines a term/name in a description list
<dd>	Defines a description of a term/name in a description list

Lists

List: <dl>,<dt>,<dd>

- <dl> : represents definition of the list
- <dt> : represents term definition
- <dd> : represents data for the definition of the term

```
<!-- DefList.html -->
<html>
  <body>
    <dl>
      <dt>newbie:</dt>
      <dd>one who does not have mad skills</dd>
      <dt>own:</dt>
      <dd>to soundly defeat (e.g. I owned that newbie!)</dd>
      <dt>frag:</dt>
      <dd>a kill in a shooting game</dd>
    </dl>
  </body>
</html>
```



Lists

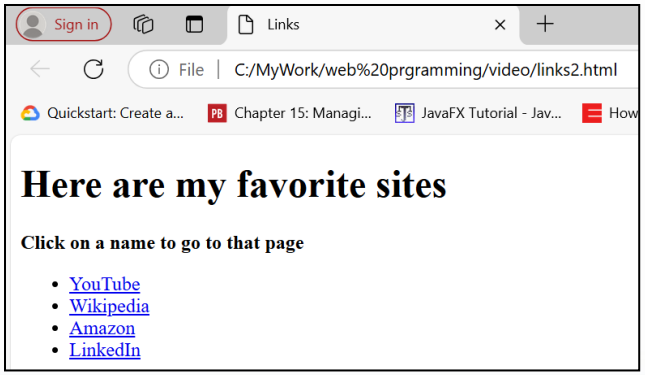
Unordered list element ul

- Creates a list in which each item in the list begins with a bullet symbol (typically a disc)
- Each entry is an li (list item) element.
- Most web browsers render these elements with a line break and a bullet symbol at the beginning of the line.

Lists

Unordered list : ul

```
<!DOCTYPE html>
<!-- links2.html -->
<!-- Unordered list containing hyperlinks. -->
<html>
    <head>
        <meta charset = "utf-8">
        <title>Links</title>
    </head>
    <body>
        <h1>Here are my favorite sites</h1>
        <p><strong>Click on a name to go to that page</strong></p>
        <!-- create an unordered list -->
        <ul>
            <!-- the list contains four list items -->
            <li><a href = "http://www.youtube.com">YouTube</a></li>
            <li><a href = "http://www.wikipedia.org">Wikipedia</a></li>
            <li><a href = "http://www.amazon.com">Amazon</a></li>
            <li><a href = ttp://www.linkedin.com">LinkedIn</a></li>
        </ul>
    </body>
</html>
```



Lists

Unordered list : ul

List Item Marker:

Value	Description
disc	Sets the list item marker to a bullet (**default**)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked

```
<html>
  <body>
    <h2>Unordered List with Disc Bullets</h2>
      <ul style="list-style-type:disc;">
        <li>Coffee</li>
        <li>Tea</li>
        <li>Milk</li>
      </ul>
  </body>
</html>
```

Unordered List with Disc Bullets

- Coffee
- Tea
- Milk

```
<html>
  <body>
    <h2>Unordered List with Disc Bullets</h2>
      <ul style="list-style-type:circle;">
        <li>Coffee</li>
        <li>Tea</li>
        <li>Milk</li>
      </ul>
  </body>
</html>
```

Unordered List with Circle Bullets

- Coffee
- Tea
- Milk

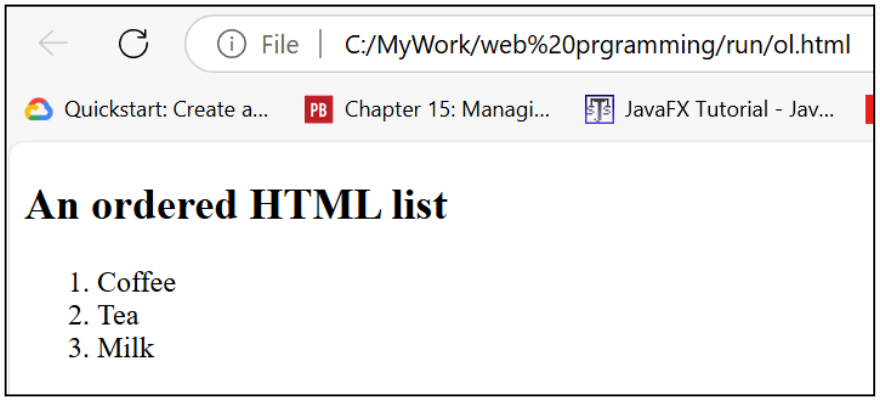
Lists

Ordered list : ol

List Item Marker:

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers
Type="square"	The list items will be square box

```
<!DOCTYPE html>
<!-- ol.html -->
<!-- Images as link anchors. -->

<html>
    <body>
        <h2>An ordered HTML list</h2>
            <ol type="1">
            <li>Coffee</li>
            <li>Tea</li>
            <li>Milk</li>
        </ol>
    </body>
</html>
```



Lists

Ordered list : ol

The li value attribute:

```
<!DOCTYPE html>
<html>
   <body>
      <h1>The li value attribute</h1>
      <ol type="1">
            <li value="100">Coffee</li>
            <li>Tea</li>
            <li>Milk</li>
            <li>Water</li>
            <li>Juice</li>
            <li>Beer</li>
      </ol>
   </body>
</html>
```

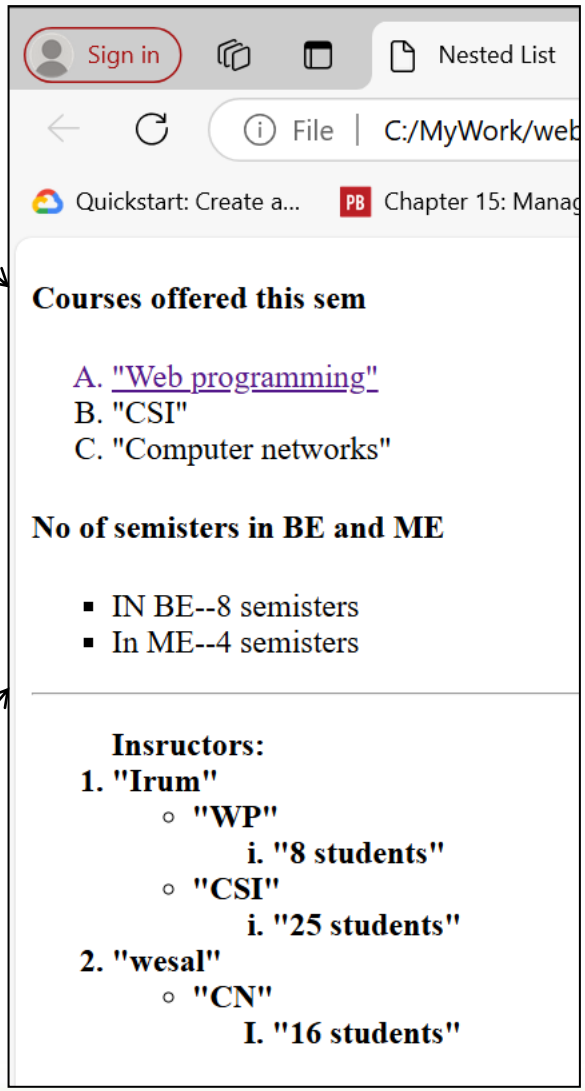
The li value attribute

100. Coffee
101. Tea
102. Milk
103. Water
104. Juice
105. Beer

Lists

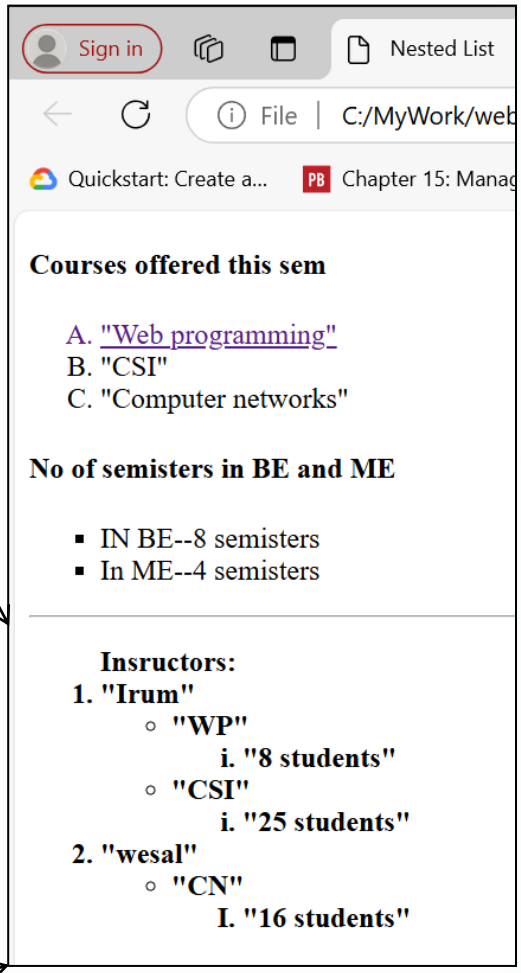
Nested List

```
<!DOCTYPE html>
<!-- NesttedList.html  -->
<html>
    <head>
       <title>Nested List</title>
    </head>
    <body>
       <h4>Courses offered this sem</h4>
       <ol type="A">          <!-- try "I" romandlist -->
          <a href>
             <li>"Web programming"</li>
          </a>
          <li>"CSI"</li>
          <li>"Computer networks"</li>
       </ol>
       </hr>
       <h4>No of semisters in BE and ME</h4>
       <ul type="square">   <!-- try circle -->
          <li>IN BE--8 semisters</li>
          <li>In ME--4 semisters</li>
       </ul>
       <hr>
       <h4>No of semisters in BE and ME</h4>
       <ul type="square">          <!-- try circle -->
             <li>IN BE--8 semisters</li>
             <li>In ME--4 semisters</li>
       </ul>
       <hr>
```



Nested List

```
<ol type="1">
    <b>Insructors:
        <li> "Irum"
            <ul>
                <li> "WP"
                    <ol type="i">
                        <li>"8 students" </li>
                    </ol>
                <li>
                    "CSI"
                    <ol type="i">
                        <li>"25 students"</li>
                    </ol>
                </li>
            </ul>
        </li>
        <li>
            "wesal"
            <ul>
                <li>
                    "CN"
                    <ol type = "I"> <li>"16 students"</li></ol>
                <li>
            </ul>
        </li>
    </b>
</ol>
</body>
<html>
```



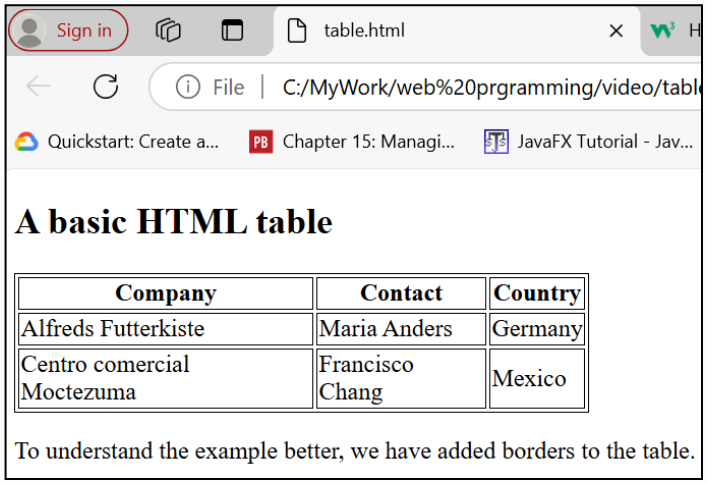
Tables

Tables Element Reference:

Element	Description
<table>	Defines a table
<caption>	Defines a table caption
<th>	Defines a header cell in a table
<tr>	Defines a row in a table
<td>	Defines a cell in a table
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table
<col>	Specifies column properties for each column within a <colgroup>
<colgroup>	Specifies a group of one or more columns in a table for formatting

Tables

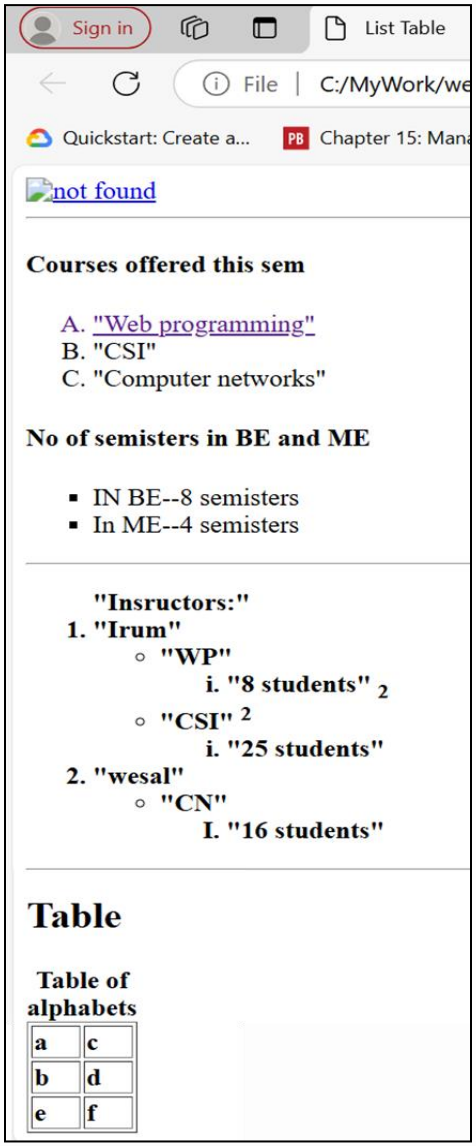
- Tables are frequently used to organize data into rows and columns.
- A table cell can contain all sorts of HTML elements: text, images, lists, links, other tables, etc.




```
<!DOCTYPE html>
<!-- table.html -->
<html>
   <body>
      <h2>A basic HTML table</h2>
      <table border="1", style="width:50%">
        <tr>
          <th>Company</th>
          <th>Contact</th>
          <th>Country</th>
        </tr>
        <tr>
          <td>Alfreds Futterkiste</td>
          <td>Maria Anders</td>
          <td>Germany</td>
        </tr>
        <tr>
          <td>Centro comercial Moctezuma</td>
          <td>Francisco Chang</td>
          <td>Mexico</td>
        </tr>
      </table>
   </body>
</html>
```

- **<tr> (table row).** Each table row starts with a <tr> and ends with a </tr> tag.
- **<th> (table header).** Contains header information
- **<td> (table data).** Everything between **<td>** and **</td>** are the content of the table cell.

Tables



Run me

Tables

Complex table using row span and col span

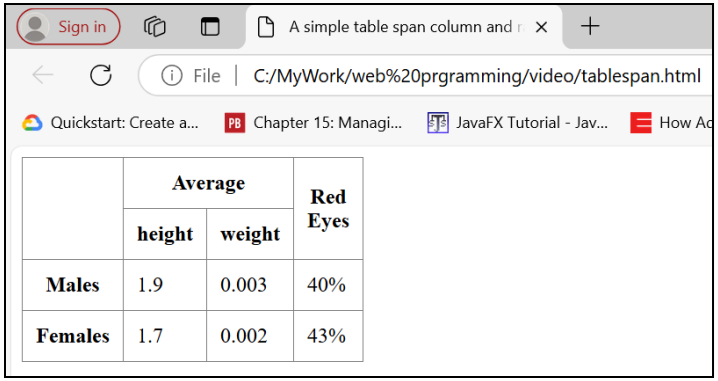
NAME		

APRIL		

2022		
FIESTA		

Tables

Complex table using row span and col span



	Average		Red Eyes
	height	weight	
Males	1.9	0.003	40%
Females	1.7	0.002	43%

- **colspan attribute:** used To make a cell span over multiple columns.
- **rowspan attribute:** used To make a cell span over multiple rows. The value of the **rowspan** attribute represents the number of rows to span.

```
<!DOCTYPE html>
<!-- tablespan.html -->
<!-- Creating a table span -->
<html>
    <head>
        <meta charset "utf-8">
        <title>A simple table span column and raws</title>
    </head>
    <body>
        <table border="1px" cellpadding="10" style="border-collapse:collapse">
            <!-- try cellpadding = 15 -->
            <!-- try remove style -->
            <thead>
                <tr>
                    <th rowspan="2"></th>
                    <th colspan="2">Average</th>
                    <th rowspan="2">
                        Red
                        <br>
                        Eyes
                    </th>
                </tr>
                <tr>
                    <th>height</th>
                    <th>weight</th>
                </tr>
                <tr>
                    <th>Males</th>
                    <td>1.9</td>
                    <td>0.003</td>
                    <td>40%</td>
                </tr>
                <tr>
                    <th>Females</th>
                    <td>1.7</td>
                    <td>0.002</td>
                    <td>43%</td>
                </tr>
            </thead>
        </table>
    </body>
</html>
```

Tables

Table Styling

Zebra Striped Table

For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:

First Name	Last Name	Points
Peter	Griffin	$100
Lois	Griffin	$150
Joe	Swanson	$300
Cleveland	Brown	$250

```
<!DOCTYPE html>
<!-- TableStyling -->
<html>
    <head>
        <style>
            table {border-collapse: collapse;width: 100%;}
            th, td {text-align: left;padding: 8px;}
            tr:nth-child(even) {background-color: #D6EEEE;}  <!-- try odd instead of even -->
        </style>
    </head>
    <body>
        <h2>Zebra Striped Table</h2>
        <p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>
        <table>
            <tr>
                <th>First Name</th> <th>Last Name</th> <th>Points</th></tr>
            <tr>
                <td>Peter</td> <td>Griffin</td> <td>$100</td></tr>
            <tr>
                <td>Lois</td> <td>Griffin</td> <td>$150</td></tr>
            <tr>
                <td>Joe</td> <td>Swanson</td> <td>$300</td></tr>
            <tr>
                <td>Cleveland</td> <td>Brown</td> <td>$250</td></tr>
        </table>
    </body>
</html>
```

Moved to CSS

Tables

Table Styling

Zebra Striped Table

For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:

First Name	Last Name	Points
Peter	Griffin	$100
Lois	Griffin	$150
Joe	Swanson	$300
Cleveland	Brown	$250

```
<!DOCTYPE html>
<!-- TableStyling1 -->
<html>
    <head>
        <style>
            table {border-collapse: collapse;width: 30%;}
            th, td {text-align: left;padding: 8px;}
            td:nth-child(even), th:nth-child(even) {background-color: #D6EEEE;}  <!-- try odd instead of even -->
        </style>
    </head>
    <body>
        <h2>Zebra Striped Table</h2>
        <p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>
        <table>
            <tr>
                <th>First Name</th> <th>Last Name</th> <th>Points</th></tr>
            <tr>
                <td>Peter</td> <td>Griffin</td> <td>$100</td></tr>
            <tr>
                <td>Lois</td> <td>Griffin</td> <td>$150</td></tr>
            <tr>
                <td>Joe</td> <td>Swanson</td> <td>$300</td></tr>
            <tr>
                <td>Cleveland</td> <td>Brown</td> <td>$250</td></tr>
        </table>
    </body>
</html>
```

Moved to CSS

Tables

Table Styling

Zebra Striped Table

For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:

First Name	Last Name	Points
Peter	Griffin	$100
Lois	Griffin	$150
Joe	Swanson	$300
Cleveland	Brown	$250

```
<!DOCTYPE html>
<!-- TableStyling2 -->
<html>
    <head>
        <style>
            table {border-collapse: collapse;width: 30%;}
            th, td {text-align: left;padding: 8px;}
            tr:nth-child(even) { background-color: rgba(150, 212, 212, 0.4);}
            th:nth-child(even),td:nth-child(even) {background-color: rgba(150, 212, 212, 0.4);}
        </style>
    </head>
    <body>
        <h2>Zebra Striped Table</h2>
        <p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>
        <table>
            <tr>
                <th>First Name</th> <th>Last Name</th> <th>Points</th></tr>
            <tr>
                <td>Peter</td> <td>Griffin</td> <td>$100</td></tr>
            <tr>
                <td>Lois</td> <td>Griffin</td> <td>$150</td></tr>
            <tr>
                <td>Joe</td> <td>Swanson</td> <td>$300</td></tr>
            <tr>
                <td>Cleveland</td> <td>Brown</td> <td>$250</td></tr>
        </table>
    </body>
</html>
```

Moved to CSS

Tables

HTML Table Colgroup

Add the a colgroup with a col element that spans over two columns to define a style for the two columns:

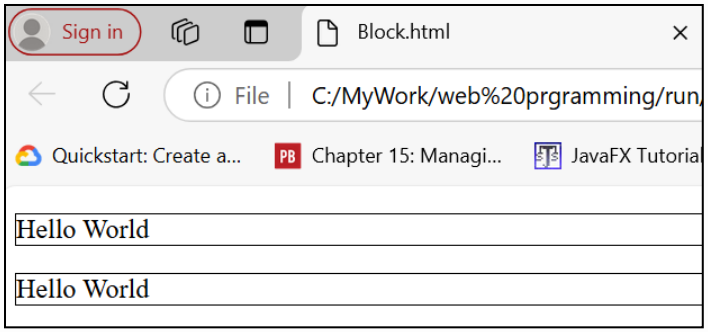
MON	TUE	WED	THU	FRI	SAT	SUN
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

```
<!DOCTYPE html>
<!-- TableStyling3 -->
<html>
    <head>
        <style> table, th, td {border: 1px solid black; border-collapse: collapse;}
    </style>
    </head>
    <body>
        <h2>Colgroup</h2>
        <p>Add the a colgroup with a col element that spans over two columns to define a style for the two columns:</p>
        <table style="width: 30%;">
            <colgroup>
                <col span="2" style="background-color: #D6EEEE">
            </colgroup>
            <tr> <th>MON</th> <th>TUE</th> <th>WED</th> <th>THU</th> <th>FRI</th> <th>SAT</th> <th>SUN</th> </tr>
            <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> <t
            <tr> <td>8</td> <td>9</td> <td>10</td> <td>11</td> <td>12</td>
            <tr> <td>15</td> <td>16</td> <td>17</td> <td>18</td> <td>19</t
            <tr> <td>22</td> <td>23</td> <td>24</td> <td>25</td> <td>26</t
        </table>
    </body>
</html>
```

Moved to CSS

HTML Block Level Elements

- A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.
- A block-level element always takes up the full width available (stretches out to the left and right as far as it can).
- Two commonly used block elements are: **<p> and .**
- The **<p>** element defines a paragraph in an HTML document.
- The ** ** element defines a division or a section in an HTML document.

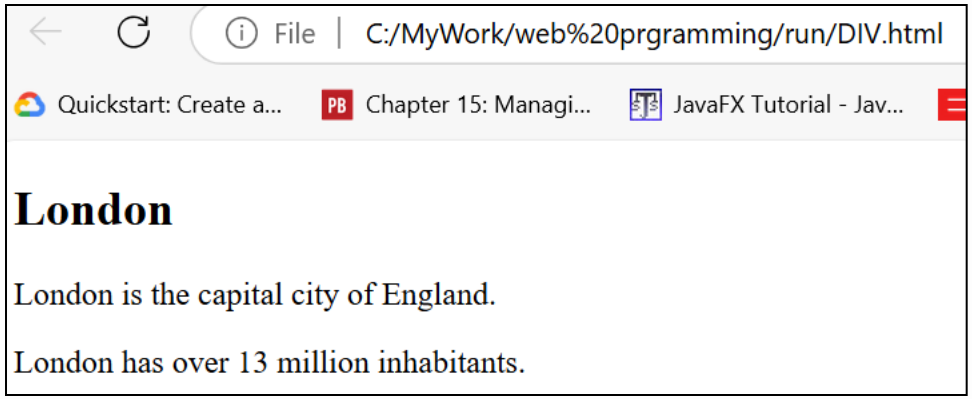



```
<!DOCTYPE html>
<!-- Block.html -->
<html>
    <body>
        <p style="border: 1px solid black">Hello World</p>
        <div style="border: 1px solid black">Hello World</div>
    </body>
</html>
```

HTML Block Level Elements

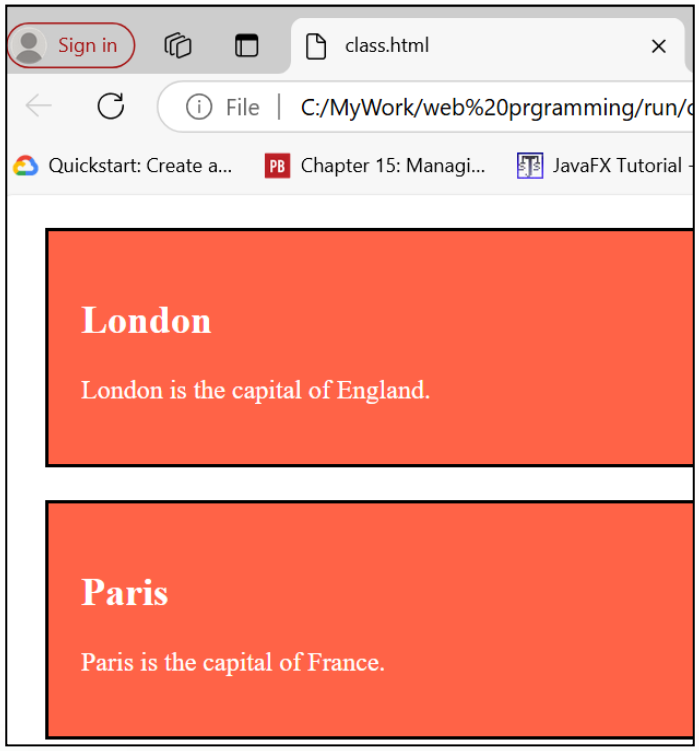
- The ** ** element is often used to group sections of a web page together.
- Example:

```
<div>
  <h2>London</h2>
  <p>London is the capital city of England.</p>
  <p>London has over 13 million inhabitants.</p>
</div>
```



HTML class Attribute

- The **class** attribute is often used to point to a class name in a style sheet. It is used to access and manipulate elements with the specific class name.
- In the following example we have three ** ** elements with a class attribute with the value of "city". All of the three ** ** elements will be styled equally according to the **.city style** definition in the head section.
- Multiple HTML elements can share the same class:




```
<!DOCTYPE html>
<!-- class.html -->
<html>
    <head>
        <style>
            .city {
              background-color: tomato;
              color: white;
              border: 2px solid black;
              margin: 20px;
              padding: 20px;
            }
        </style>
    </head>
    <body>
        <div class="city">
            <h2>London</h2>
            <p>London is the capital of England.</p>
            <p>Paris is the capital of France.</p>
        </div>
    </body>
</html>
```

Moved to CSS

HTML class Attribute

Multiple Classes

- HTML elements can belong to more than one class.
- To define multiple classes, separate the class names with a space, e.g. . The element will be styled according to all the classes specified.
- In the following example, the first <h2> element belongs to both the city class and also to the main class, and will get the CSS styles from both of the classes:

```
<!DOCTYPE html>
<!-- multipleClass.html -->
<html>
    <head>
        <style>
            .city {
              background-color: tomato;
              color: white;
              padding: 10px;
            }
            .main {
              text-align: center;
            }
        </style>
    </head>
    <body>
        <h2>Multiple Classes</h2>
        <p>Here, all three h2 elements belongs to the "city" class.
           In addition, London  belongs to the "main" class,
           which center-aligns the text.</p>
        <h2 class="city main">London</h2>
        <h2 class="city">Paris</h2>
        <h2 class="city">Tokyo</h2>

</body>
</html>
```

HTML class Attribute

- In the following example we have two elements with a class attribute with the value of "note".
- Both elements will be styled equally according to the .note style definition in the head section:

```
<!DOCTYPE html>
<!-- span.html -->
<html>
    <head>
        <style>
            .note {
              font-size: 120%;
                color: red;
            }
        </style>
    </head>
    <body>
        <h1>My <span class="note">Important</span> Heading</h1>
        <p>This is some <span class="note">important</span> text.</p>
    </body>
</html>
```

Moved to CSS

HTML id Attribute

- The **id** attribute specifies a unique id for an HTML element.
- The value of the **id** attribute must be unique within the HTML document.
- The **id** attribute is used to point to a specific style declaration in a style sheet. It is also used by JavaScript to access and manipulate the element with the specific id.
- The syntax for **id** is: write a hash character (**#**), followed by an **id name**. Then, define the CSS properties within curly braces {}.
- In the following example we have an <h1> element that points to the id name "myHeader". This <h1> element will be styled according to the #myHeader style definition in the head section:


```
<!DOCTYPE html>
<!-- id.html -->
<html>
    <head>
        <style>
            #myHeader {
              background-color: lightblue;
              color: black;
              padding: 40px;
              text-align: left;
            }
```

Moved to CSS

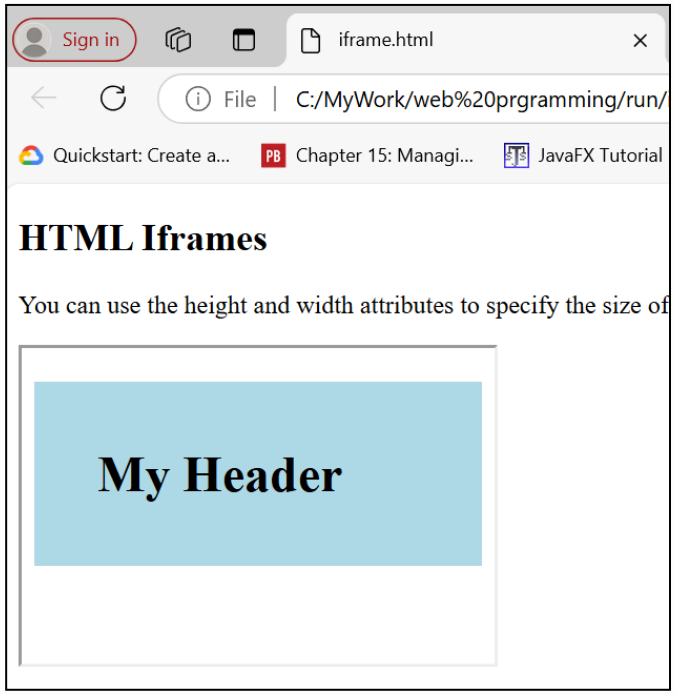
```
</h1>
</body>
</html>
```

Difference Between Class and ID:

A class name can be used by multiple HTML elements, while an id name must only be used by one HTML element within the page:

HTML iframe element

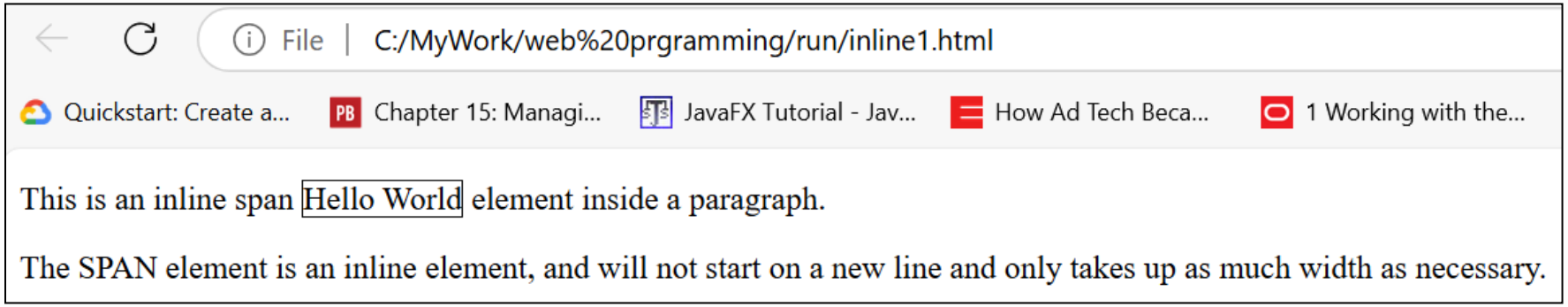
- An HTML iframe is used to display a web page within a web page.
- Syntax: `<iframe src="url" title="description"></iframe>`




```
<!DOCTYPE html>
<!-- iframe.html -->

<html>
    <body>
        <h2>HTML Iframes</h2>
        <p>You can use the height and width attributes to specify the size of the iframe:</p>
        <iframe src="id.html" height="200" width="300" title="Iframe Example"></iframe>
    </body>
</html>
```

HTML Inline Elements

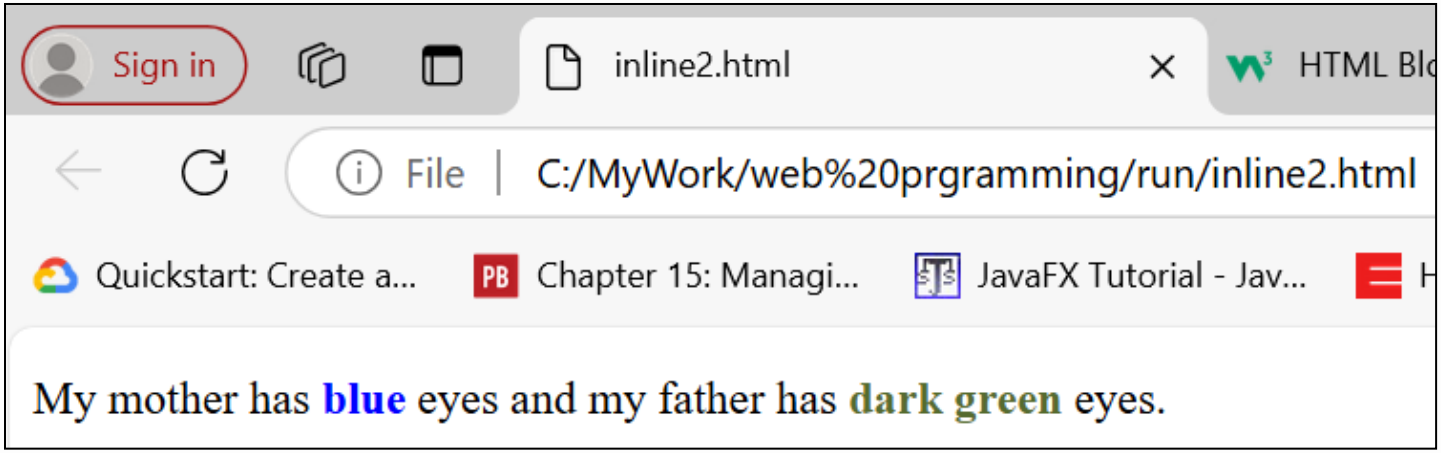
- An inline element does not start on a new line.
- An inline element only takes up as much width as necessary
- The **** element is an inline container used to mark up a part of a text, or a part of a document.
- The **** element has no required attributes, but style, class and id are common.
- Example:




```
<!DOCTYPE html>
<!-- inline1.html -->
<html>
    <body>
        <p>This is an inline span <span style="border: 1px solid black">Hello World</span> element inside a paragraph.</p>
        <p>The SPAN element is an inline element, and will not start on a new line and only takes up as much width as necessary.</p>
    </body>
</html>
```

HTML Inline Elements

- Example:




```
<!DOCTYPE html>
<!-- inline2.html -->
<html>
    <body>
        <p>My mother has <span style="color:blue;font-weight:bold;">blue</span>
           eyes and my father has <span style="color:darkolivegreen;font-weight:bold;">dark green</span> eyes.
        </p>
    </body>
</html>
```

<script> Tag

- The HTML <script> tag is used to define a client-side script (JavaScript).
- The <script> element either contains script 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.
- To select an HTML element, JavaScript most often uses the document.getElementById() method.

<script> Tag

- This JavaScript example writes "Hello JavaScript!" into an HTML element with id="demo":

```
<!DOCTYPE html>
<html>
   <body>
      <h2>Use JavaScript to Change Text</h2>
      <p>This example writes "Hello JavaScript!" into an HTML element with  id="demo":</p>
      <p id="demo"></p>
      <script>
         document.getElementById("demo").innerHTML = "Hello JavaScript!";
      </script>
   </body>
</html>
```

<script> Tag

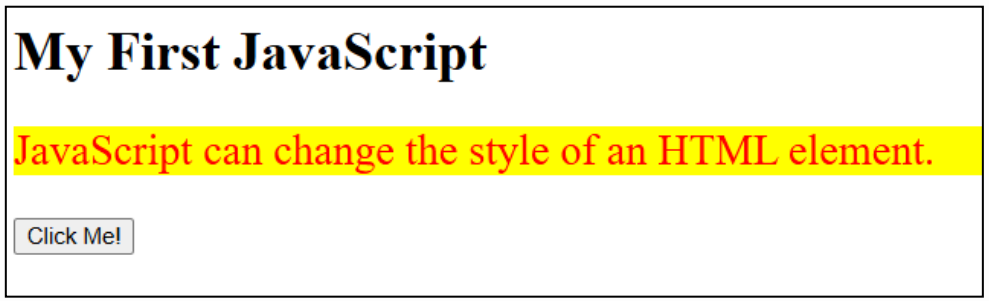
- Example - what JavaScript can do:

```
<!DOCTYPE html>
<html>
   <body>
      <h1>My First JavaScript</h1>
      <p>JavaScript can change the content of an HTML element:</p>
      <button type="button" onclick="myFunction()">Click Me!</button>
      <p id="demo">This is a demonstration.</p>
      <script>
          function myFunction() {
            document.getElementById("demo").innerHTML = "Hello JavaScript!";
          }
      </script>
</body>
</html>
```

<script> Tag

- Example - JavaScript can change styles:

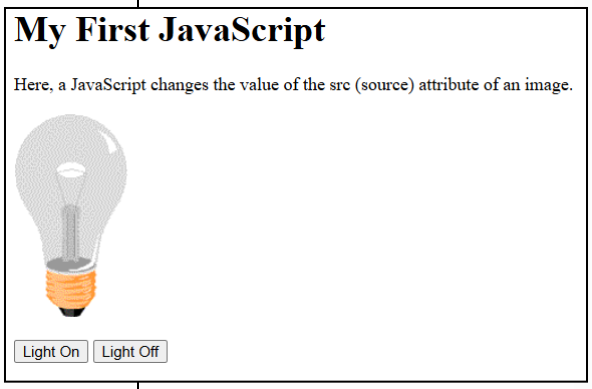
```
<!DOCTYPE html>
<html>
   <body>
      <h1>My First JavaScript</h1>
      <p id="demo">JavaScript can change the style of an HTML element.</p>
      <script>
          function myFunction() {
            document.getElementById("demo").style.fontSize = "25px";
            document.getElementById("demo").style.color = "red";
            document.getElementById("demo").style.backgroundColor = "yellow";
          }
      </script>
      <button type="button" onclick="myFunction()">Click Me!</button>
   </body>
</html>
```

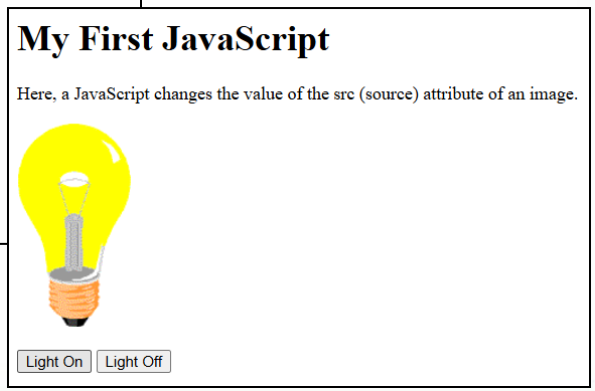


<script> Tag

- Example - JavaScript can change attributes:

```
<!DOCTYPE html>
<html>
    <body>
        <h1>My First JavaScript</h1>
        <p>Here, a JavaScript changes the value of the src (source) attribute of an image.</p>
        <script>
            function light(sw) {
                var pic;
                if (sw == 0) {
                    pic = "pic_bulboff.gif"
                } else {
                    pic = "pic_bulbon.gif"
                }
                document.getElementById('myImage').src = pic;
            }
        </script>
        <img id="myImage" src="pic_bulboff.gif" width="100" height="180">
        <p>
            <button type="button" onclick="light(1)">Light On</button>
            <button type="button" onclick="light(0)">Light Off</button>
        </p>
    </body>
</html>
```

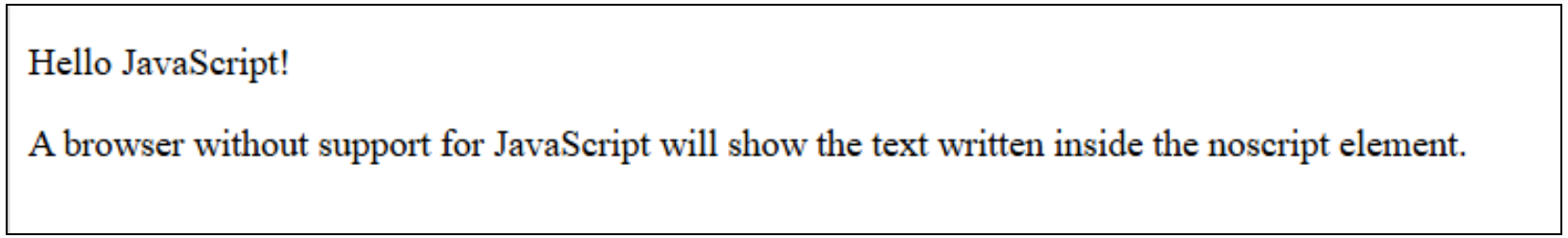




<noscript> Tag

- The HTML <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 scripts:

```
<!DOCTYPE html>
<html>
   <body>
      <p id="demo"></p>
      <script>
         document.getElementById("demo").innerHTML = "Hello JavaScript!";
      </script>
      <noscript>Sorry, your browser does not support JavaScript!</noscript>
      <p>A browser without support for JavaScript will show the text written  inside the noscript
         element.</p>
   </body>
</html>
```



Insert markup characters in content

- < Displays <
- > Displays >
- & Displays &
- " Displays "
-   **Nonbreaking space** (won't insert a line break at this space)

Forms and Input

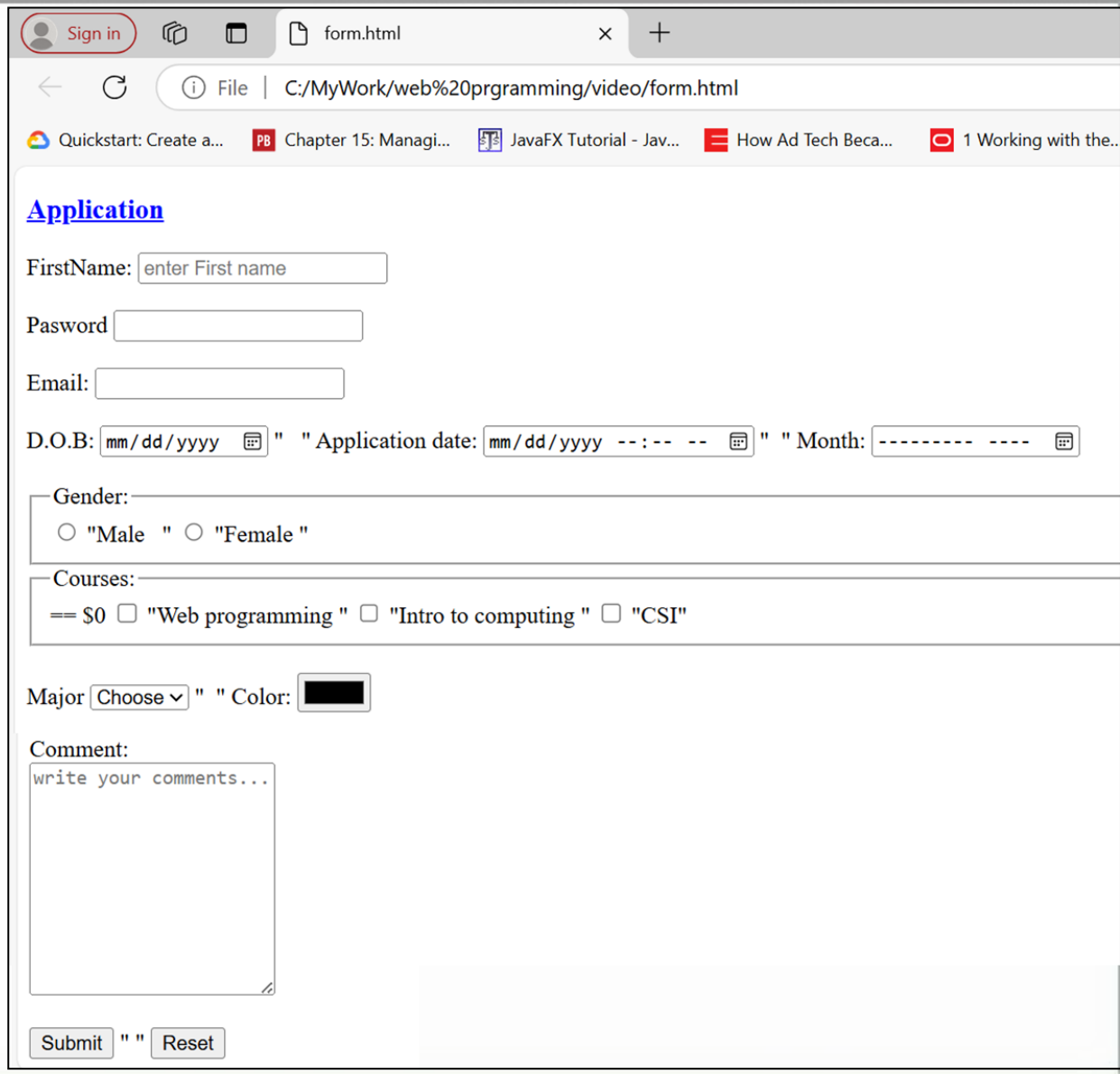
Forms and Input Element Reference:

Element	Description
<form>	Defines an HTML form for user input
<input>	Defines an input control
<textarea>	Defines a multiline input control (text area)
<button>	Defines a clickable button
<select>	Defines a drop-down list
<optgroup>	Defines a group of related options in a drop-down list
<option>	Defines an option in a drop-down list
<label>	Defines a label for an <input> element
<fieldset>	Groups related elements in a form
<legend>	Defines a caption for a <fieldset> element
<datalist>	Specifies a list of pre-defined options for input controls
<output>	Defines the result of a calculation

Form

- HTML5 provides forms for collecting information from users.
- Figure in the next slide is a simple form that sends data to the web server for processing.
- **A form is defined by a <form> element**
  - Attribute method specifies how the form's data is sent to the web server.
    - Using method = "**post**" **appends form data to the browser request**, which contains the protocol (HTTP) and the requested resource's URL.
    - The other possible value, method = "**get**", **appends the form data directly to the end of the URL** of the script, where it's visible in the browser's Address field.
  - The **action** attribute of the form element **specifies the script** to which the form data will be sent

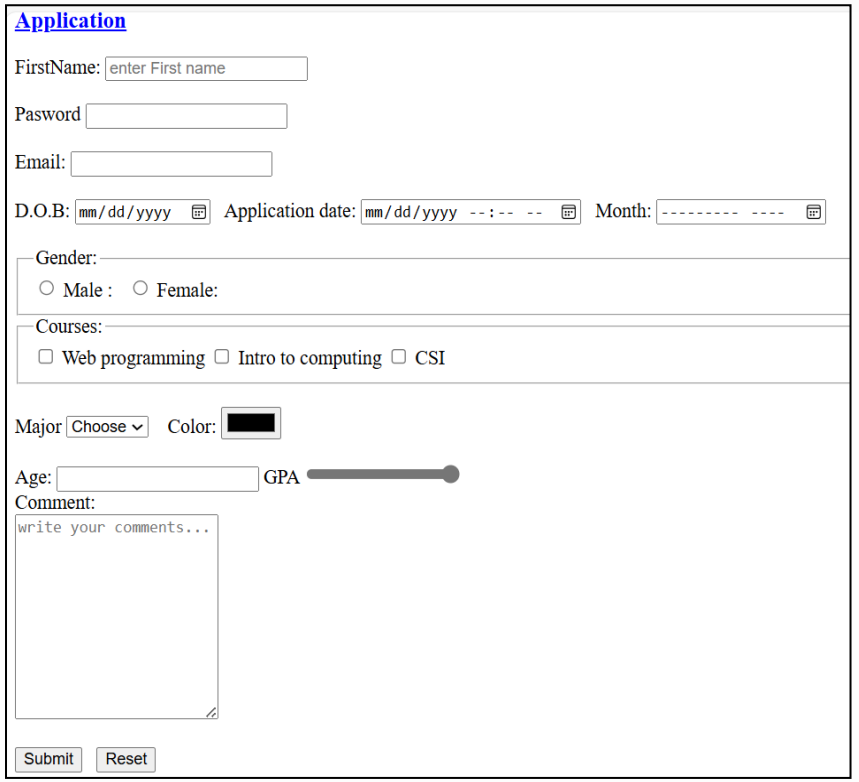
Form



Click here

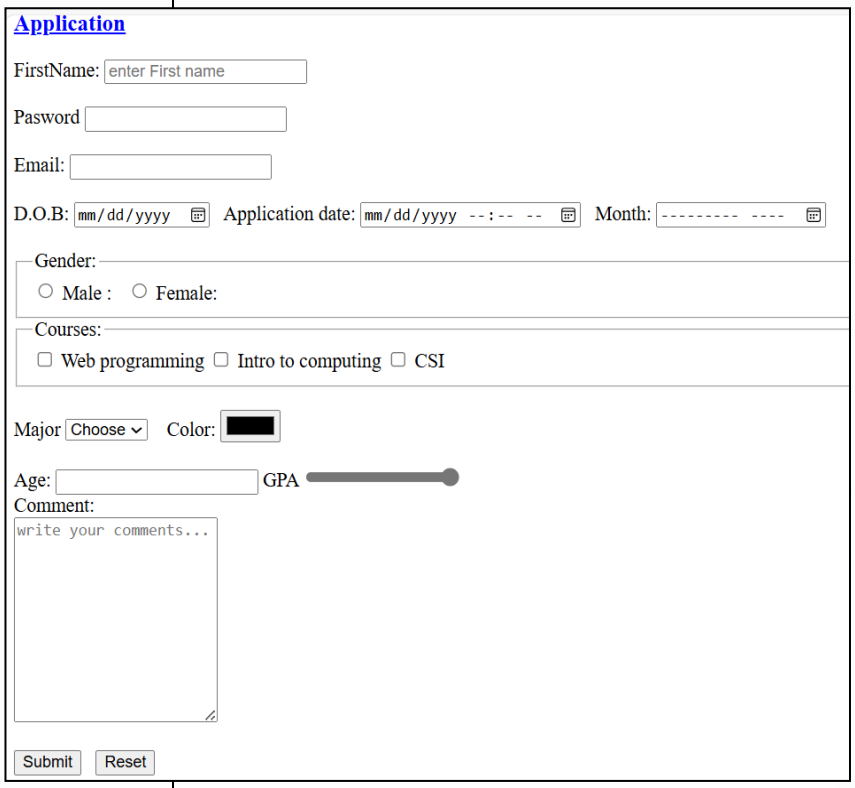
Form

```
<!DOCTYPE html>
<!-- form.html -->
<!-- createing form. -->
<html>
    <head>
    </head>
    <body>
        <form method="get/post" action="#">
            <h3  style="text-decoration:underline;color:blue">Application  </h3>
            <label> FirstName: </label>
            <input type="text" placeholder="enter First name" required>
            <br>
            <br>
            <label>Pasword</label>
            <input type="password">
            <br>
            <br>
            <label>Email:</label>
            <input type="email">
            <br>
            <br>
            <label>D.O.B:</label>
            <input type="date">
            "  
            "
            <label>Application date:</label>
            <input type="datetime-local">
            " 
            "
            <label>Month: </label>
            <input type="month">
            <br>
            <br>
            <fieldset>
                <legend>Gender: </legend>
                <input type="radio" name="gender">
                "Male  
                "
                <input type="radio" name="gender">
                "Female
                "
            </fieldset>
            <fieldset>
                <legend>Courses: </legend> == $0
                <input type="checkbox">
                "Web programming
                "
                <input type="checkbox">
                "Intro to computing
                "
                <input type="checkbox">
                "CSI"
```



Form

```
            <fieldset>
                <legend>Gender: </legend>
                <input type="radio" name="gender">
                "Male  
                "
                <input type="radio" name="gender">
                "Female
                "
            </fieldset>
            <fieldset>
                <legend>Courses: </legend> == $0
                <input type="checkbox">
                "Web programming
                "
                <input type="checkbox">
                "Intro to computing
                "
                <input type="checkbox">
                "CSI"
                <br>
            </fieldset>
            <br>
            <label>Major</label>
            <select>
                <option selected>Choose</option>
                <option>CS</option>
                <option>CE</option>
                <option>IT</option>
            </select>
            "  "
            <label>Color: </label>
            <input type="color">
            <br>
            <br>
            <label>Comment: </label>
            <br>
            <textarea col="18" rows="10" placeholder="write your comments..."></textarea>
            <br>
            <br>
            <input type="submit">
            " "
            <input type="reset">
        </form>
    </body>
</html>
```



Form

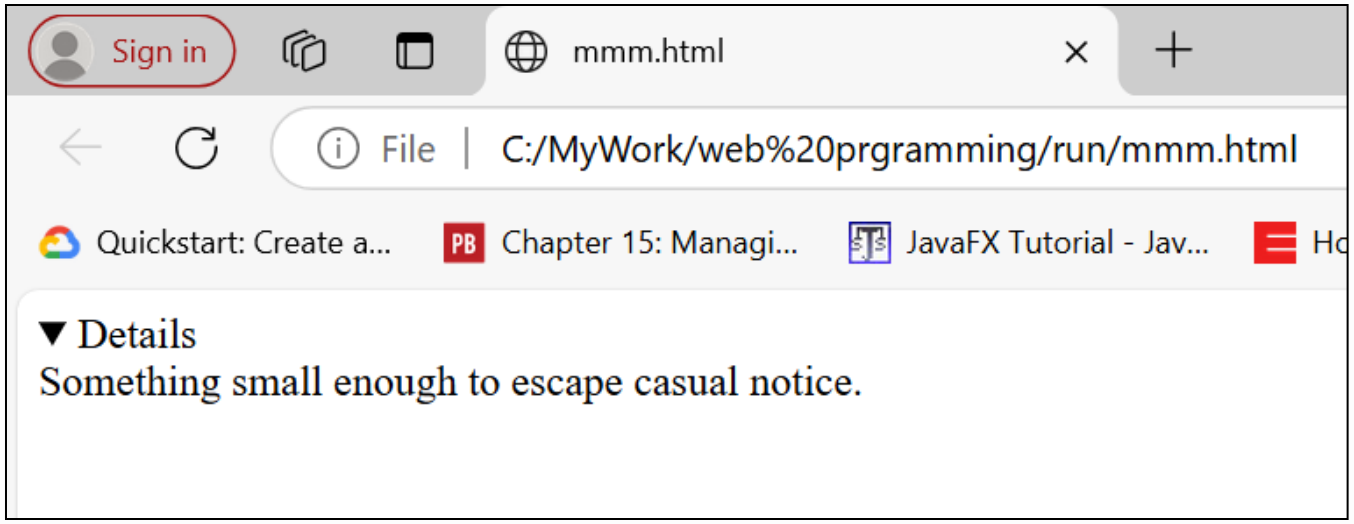
HTML5 Form input Types

- input Type color
- input Type date
- input Type datetime
- input Type datetime-local
- input Type email
- input Type month
- input Type number
- input Type range
- input Type search
- input Type tel
- input Type time
- input Type url
- input Type week

Summary Tag

- <summary> tag used with delete tag.
- Used to create a drop down summary paragraph for topics.

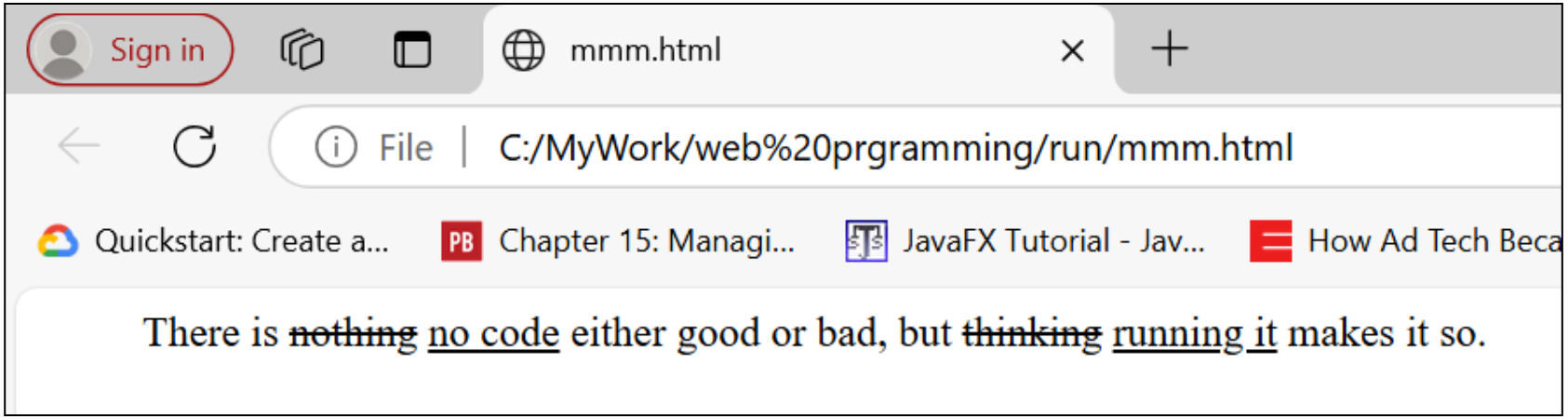
```
<details>
      <summary>Details</summary>
      Something small enough to escape casual notice.
</details>
```



Delete Tag

- <del> tag presents a range of text that has been deleted form the document

```
<blockquote>
    There is <del>nothing</del> <ins>no code</ins> either good or bad, but
    <del>thinking</del> <ins>running it</ins> makes it so.
</blockquote>
```



Insert Tag

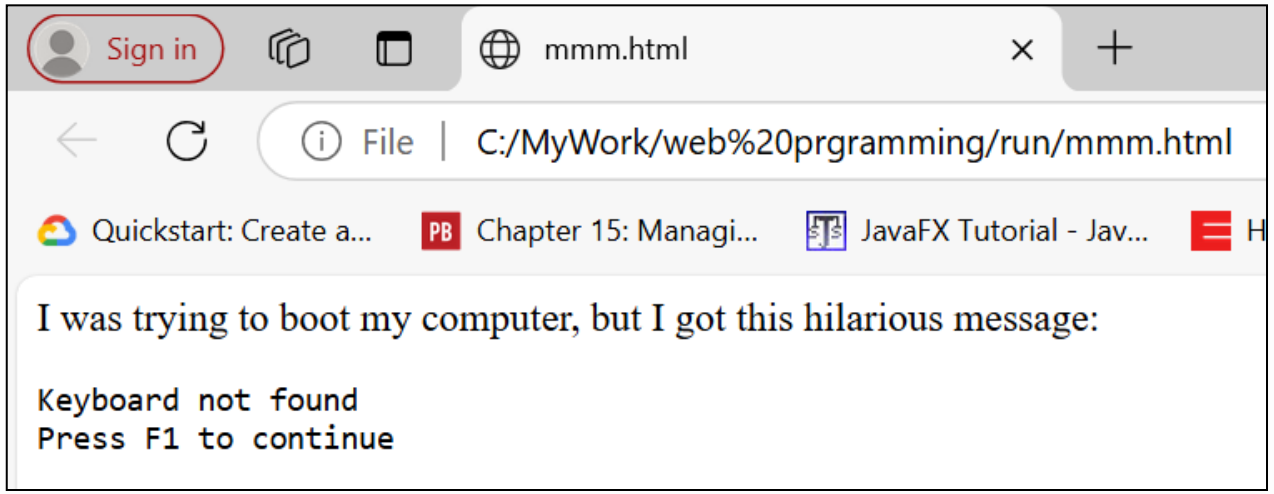
- <ins> tag or insert tag which is usually used along with the delete tag.
- Used when you want to show that some new text has been added in place of some deleted text.

```
<p>"You're late!"</p>
<del>
   <p>"I apologize for the delay."</p>
</del>
<ins cite="../howtobeawizard.html" date time="2018-05">
   <p>"A wizard is never late "</p>
</ins>
```

Sample Tag

- <samp> tag or sample tag which most programmers would love.
- It is used to show inline text which represents sample output from a computer program

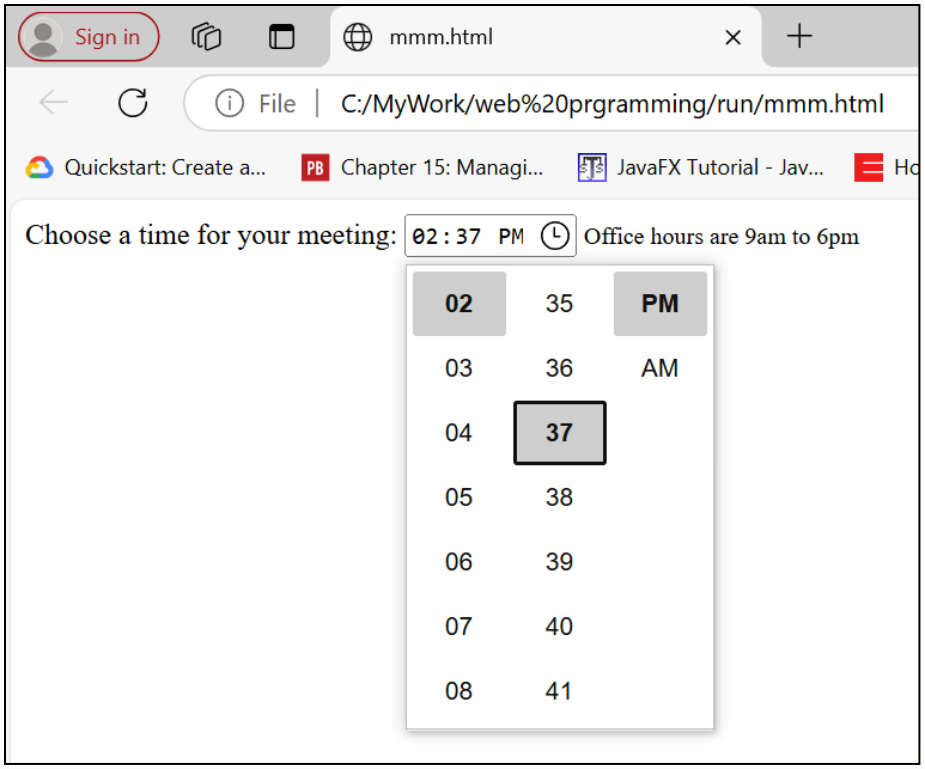
```
<p>I was trying to boot my computer, but I got this hilarious message:</p>
<p>
   <samp>Keyboard not found <br />Press F1 to continue</samp>
</p>
```



Time Tag

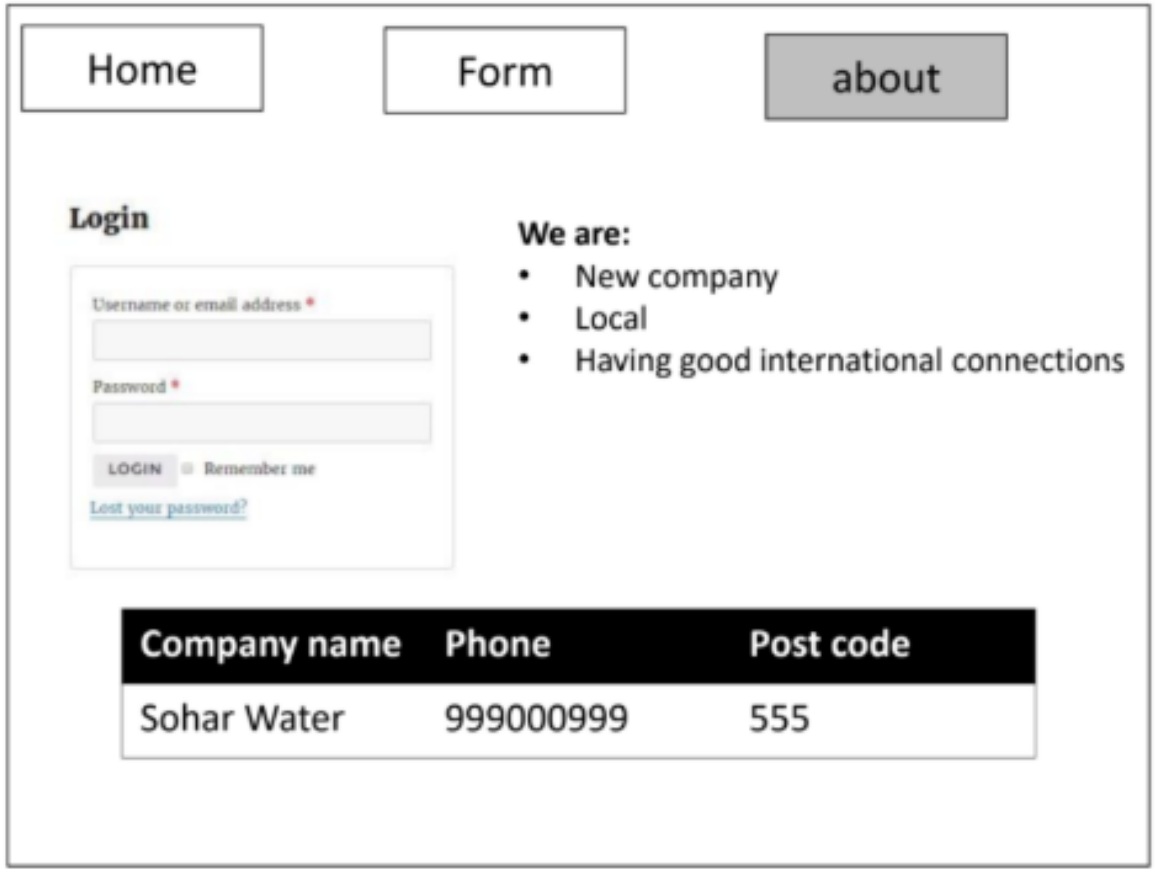
- <time> tag creates input field designed to let the user enter time.

```
<label for="appt">Choose a time for your meeting: </label>
<input type="time" id="appt" name="appt" min="09:00" max="18:00" required />
<small>Office hours are 9am to 6pm</small>
```



Assignment

Write the necessary tags and input elements to create the web page shown below?



Assignment

Create the web page shown below using HTML file name it "index.html". Include below image in the your web page.

Assignment

Fill in the blank in the CSS part to create the webpage shown . Assume that the CSS style sheet is linked to the html file.

CSS

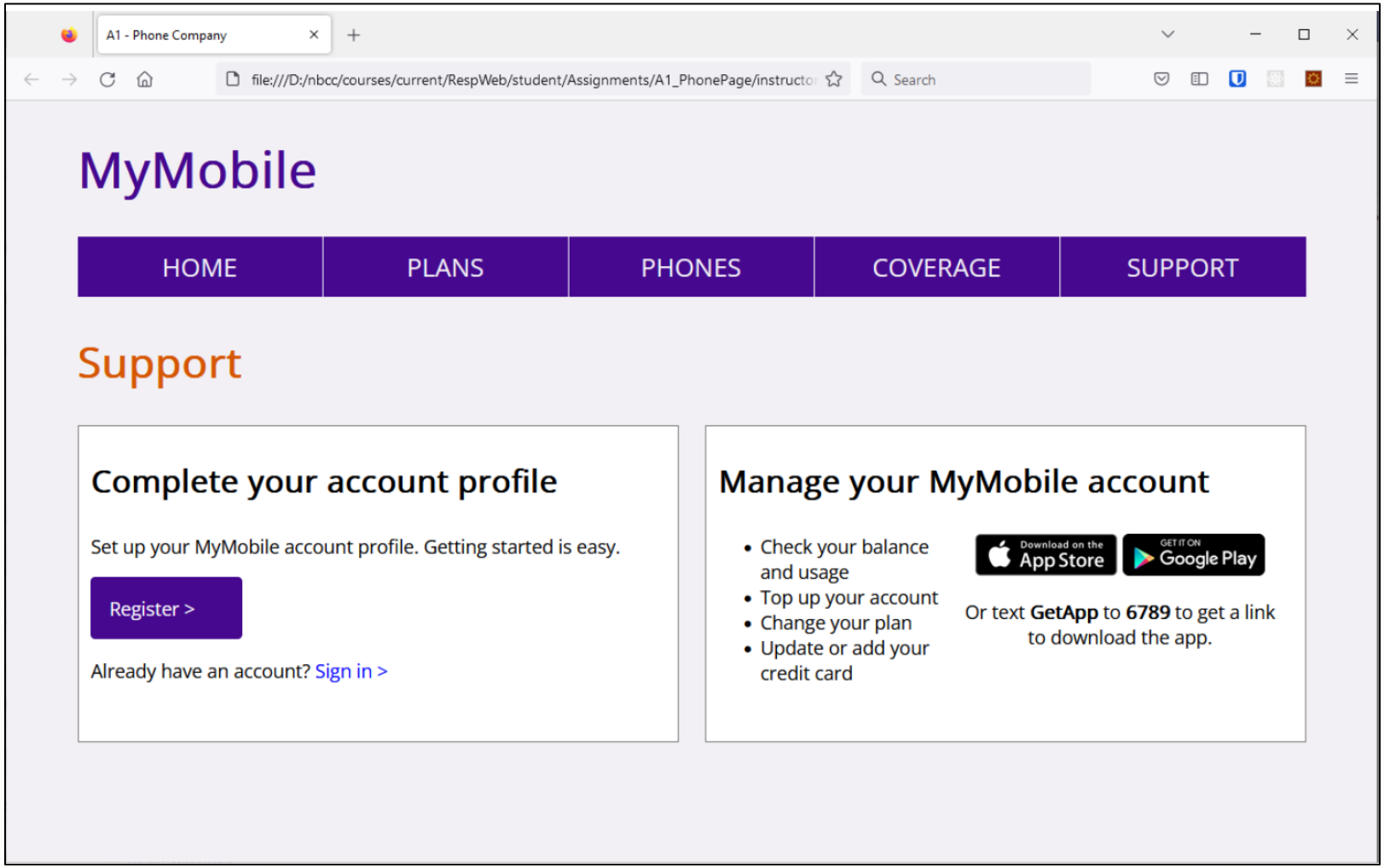
```
body {
  color: _____;
  background-color: _____;
}
.important {
  color: _____;
  background-color: _____;
}
.highlight {
  color: _____;
}
ul li, ol, p {
  color: _____;
  background-color: _____;
}

#highlight {
color: _____;
}
```

HTML

```
<html>
  <body>
    <p> Paragraph 1</p>
    <ul>
       <li class="important"> Item 1</li>
    </ul>
    <p id= "important"> Paragraph 2
    </p>
    <p class="important" id= "highlight">
     Paragraph 3
    </p>
    <ol>
        <li class="highlight">
        Item 3 </li>
    </ol>
   </body>
</html>
```

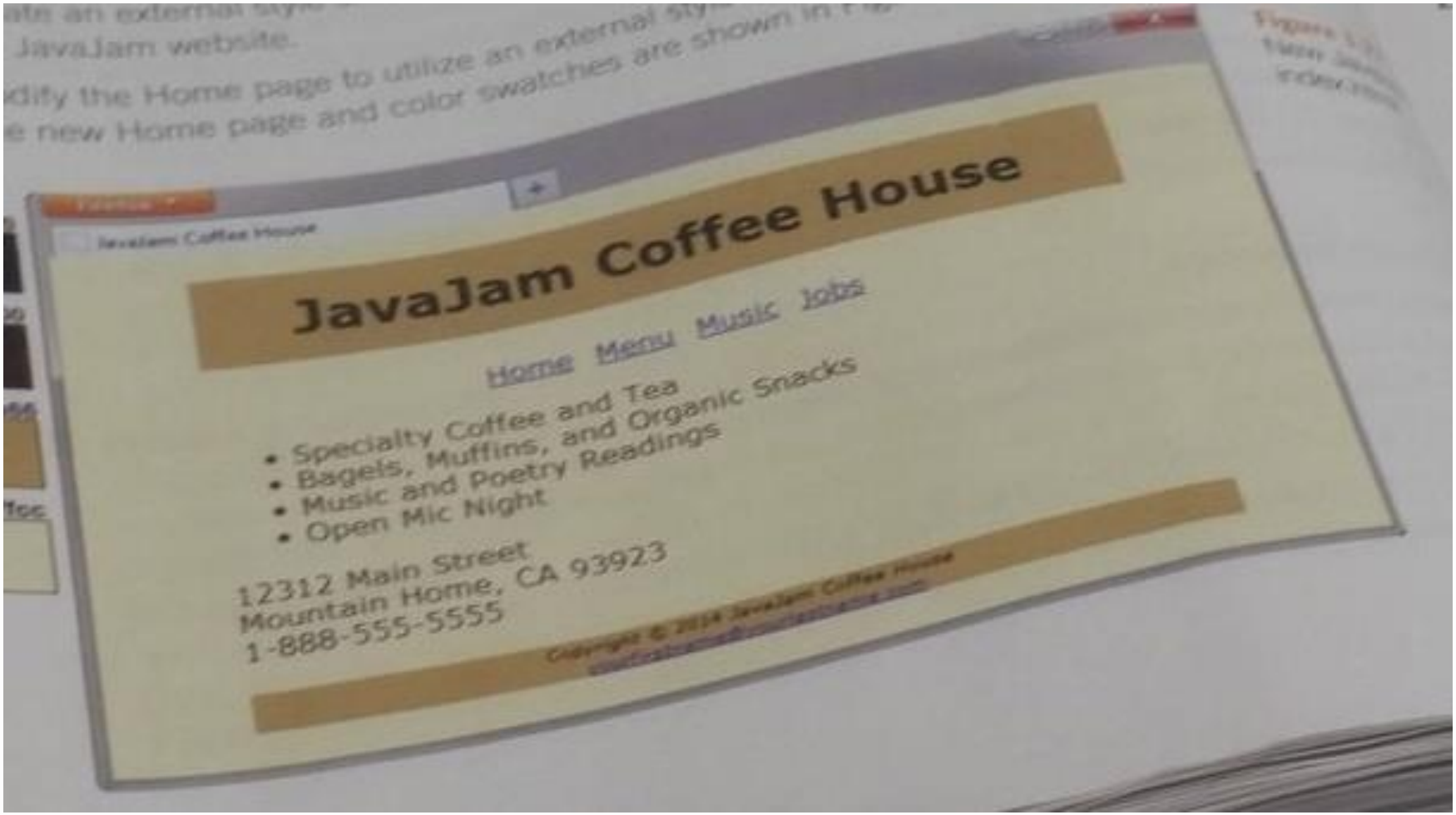
Assignment

Create the web page shown below using HTML file name it "MyMobile.html".



Assignment

Create the web page shown below using HTML file name it "JavaJam.html".



Assignment

Create the web page shown below using HTML file name it "CSS.html".

CSS Levels

W3C | W3Schools | CSS Working Group | CSS Current Work

CSS Level 1

The original CSS Level 1 recommendation was released by the W3C in December 1996. There have been many revisions since its original release. You can find the original recommendation and the latest revision at w3.org

CSS Level 2

The CSS Level 2 recommendation was originally released by the W3C in May 1998. There have been many revisions since its original release. CSS Level 2 has had two revisions, CSS 2.1 and CSS 2.2.

CSS Level 3 and Beyond

The CSS Level 3 recommendation was released in May 2001. Since its release, there have been several changes, including the method for making changes. For the latest updates, visit the CSS Current Work webpage.

Student Name:

HTML Tags Reference:

Tag	Description
<p>	Defines a paragraph
<hr>	Defines a thematic change in the content
 	Inserts a single line break
<pre>	Defines pre-formatted text
<!--...-->	Defines a comment
<a>	Defines a hyperlink (link to other pages)
<address>	Defines contact information for the author/owner of a document
<b>	Defines bold text
<base>	Specifies the base URL/target for all relative URLs in a document
<body>	Defines the document's body
<button>	Defines a clickable button
<caption>	Defines a table caption
	Defines a section in a document
<form>	Defines an HTML form for user input
<h1> to <h6>	Defines HTML headings
<head>	Contains metadata/information for the document
<header>	Defines a header for a document or section
<html>	Defines the root of an HTML document

HTML Tags Reference:

Tag	Description
<img>	Defines an image
<input>	Defines an input control
<kbd>	Defines keyboard input
<label>	Defines a label for an <input> element
<meta>	Defines metadata about an HTML document
	Defines an ordered list
<output>	Defines the result of a calculation
<p>	Defines a paragraph
<small>	Defines smaller text
	Defines important text
<style>	Defines style information for a document
<table>	Defines a table
<td>	Defines a cell in a table
<th>	Defines a header cell in a table
<thead>	Groups the header content in a table
<title>	Defines a title for the document
<tr>	Defines a row in a table
	Defines an unordered list
<var>	Defines a variable