Hypertext Markup Language (HTML)

2. Working with text in HTML

2.1. Web Technology

Structure

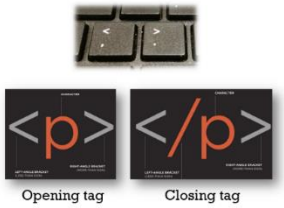
Many webpages act like electronic versions of documents such as Newspapers.

Think about the stories you read in a newspaper: for each story, there will be a headline, some text, and possibly some images. If the article is a long piece, there may be subheadings that split the story into separate sections or quotes from those involved. Structure helps readers understand the stories in the newspaper.

Structure is a way to build a webpage to make it easier for the reader to understand what he is reading. HTML code describes the structure of webpages.

The HTML code

The HTML code is made up of characters that live inside angled brackets which are called HTML **elements**. Elements are usually made up of two **tags:** an opening tag and a closing tag.



Each HTML element tells the browser something about the information that sits between its opening and closing tags.

Basic HTML elements

An HTML document has three required parts:

- a line containing HTML version information. When the webpage is loaded, the declaration line is the first line that the browser reads. It tells the browser that the following file is an HTML file.
- a head element. The head element contains metadata, which is data about the HTML document, that is not displayed on the browser when the webpage is loaded, except for the title.
- a body element. The body element contains the document's actual content. Elements in the body build the visual part of a webpage.

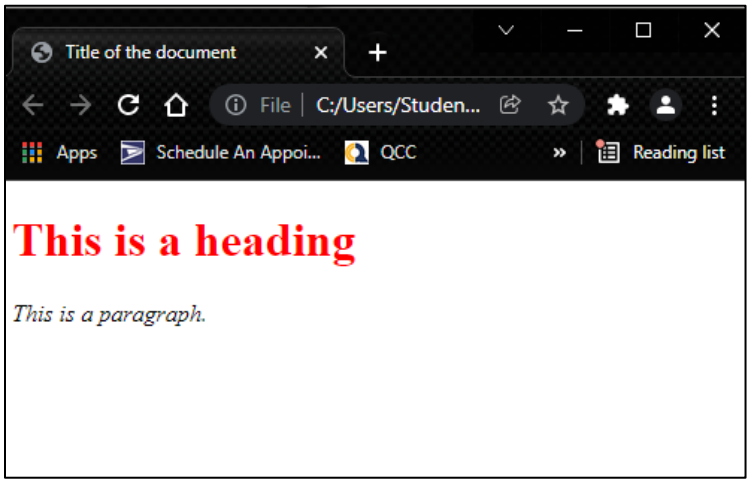
HTML version information

Head element

Body element

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Title of the document</title>
  <style media="screen">
    .title{font-size: 1.5; color: red;}
    p{font-style: italic;}
  </style>
</head>
<body>
  <h1 class="title">This is a heading</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

Browser display



The basics HTML elements are:

1. The opening **<html>** tag element indicates that anything between it and a closing **</html>** tag is HTML code
2. A **<head>** element contains information about the page such as title
3. The contents of the **<title>** element are either shown in the top of the browser, above where you usually type in the URL of the page you want to visit, or on the tab for that page (if your browser uses tabs to allow you to view multiple pages at the same time).
4. The **<body>** tag indicates that anything between it and the closing tag </body> should be inside the main browser window.
5. A paragraph of text appears between these ** ** and ** ** tags.

6. A heading element is uses to display titles or subtitles on a webpage **<h1>** and **</h1>.** The largest heading is h1, which is twice the normal size, and the smallest heading is h6, which is 0.75% of the normal size.

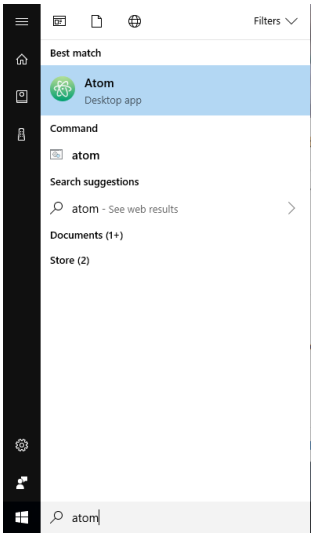
   Usually **<h1>** is used for main headings or titles, **<h2>** is used for subheadings, and if there are further sections under the subheadings, then the **<h3>** element is used, and so on.

Exercises) Creating your first webpage

Activity 1 we are going to create our first app view using HTML and check our code through google Chrome browser.

Steps:

1. We start by writing the HTML code using Atom.



2. In Atom, open a new file and save it with your last name and the file extension .html. Make sure that you save the file in your local drive or computer Desktop.

3. The first line that we write in an html file is <html> tag to declare the html file. One of the great thing about Atom is that when you type a tag name, it will automatically build a basic code structure of the code. Write html and click enter and Atom will create the following code:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>

  </body>
</html>
```

The **<!DOCTYPE>** declaration must be the very first thing in your HTML document, before the <html> tag and it specifies the rules for the markup language, so that the browsers render the content correctly.

The attribute **lang** indicates the language and the value **en** means English. This line specifies that the html code is based in English.

The attribute **dir** indicates the direction of the text and the value **ltr** indicates that the text will be display from left-to-right.

Metadata is data (information) about data. The **<meta>** tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.

Meta elements are typically used to specify page description, keywords, author of the document, last modified and other metadata

UTF-8 is the Unicode character sets from 1 to 4 bytes long. UTF-8 can represent any character in the Unicode standard. UTF-8 is backwards compatible with ASCII and the encoding for e-mail and web pages.

For app view development, it is an important to add the following line:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

The **viewport** is the user's visible area of a web page. It varies with the device, and will be smaller on a mobile phone than on a computer screen. viewport element gives the browser instructions on how to control the page's dimensions and scaling.

The **width=device-width** part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The **initial-scale=1.0** part sets the initial zoom level when the page is first loaded by the browser.

Without the viewport meta tag

With the viewport meta tag

Once we have the **<head>** element sets, we can use some tags in the **<body>.** Most of the tags that we use in **<body>** are visible in the web app. We are try to add some heading the body.

HTML has six levels of headings:

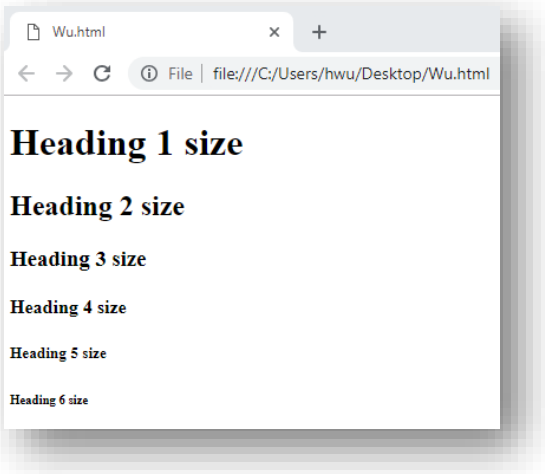
- <h1>is used for main headings
- <h2> is used for subheadings.
- If there are further sections under the subheadings then the <h3>element is used, and so on...

HTML

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

To see the output of the html code, we can go to the location of our html file, and double click on html icon

The html file will open in internet browser



More elements

Paragraph

The <**p**> **tag** specifies a paragraph of text. It is a block-level element and it automatically have margin before and after the tag.

Bold

By enclosing words in the tags **** and **** we can make characters appear bold.

Italic

By enclosing words in the tags **** and **** we can make characters appear italic.

Line Breaks

The browser will automatically show each new paragraph or heading on a new line. But if you wanted to add a line break inside the middle of a paragraph you can use the line break tag **
**

Horizontal Rules

To create a break between themes — such as a change of topic in a book or a new scene in a play — you can add a horizontal rule between sections using the **<hr/>**

Strikethrough

Strikethought means to cross something out by drawing a line through it.

The **<s>** element indicates something that is no longer accurate or relevant (but that should not be deleted).

Visually the content of an **<s>** element will usually be displayed with a line through the center.

Abbreviations and Acronyms

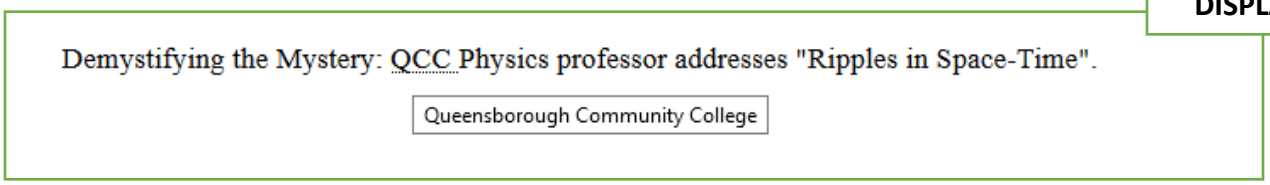
If you use an abbreviation or an acronym, then the **<abbr>** element can be used. A title attribute on the opening tag is used to specify the full term.

Example

HTML

```
<p>Demystifying the Mystery: <abbr title="Queensborough Community College">QCC
</abbr>Physics professor addresses "Ripples in Space-Time".</p>
```

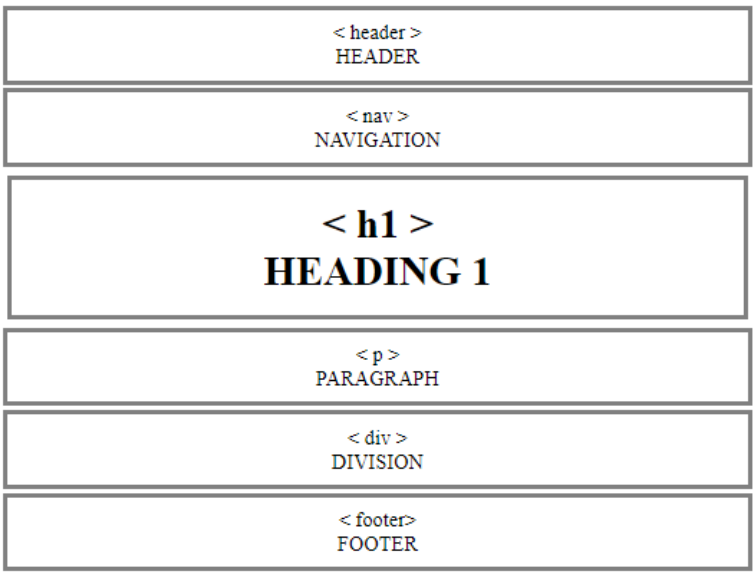
DISPLAY



Block and inline elements

There are two display elements in HTML: block and inline.

Block-level elements always starts on a new line and takes a full width of the screen. It also has a top and bottom margin.



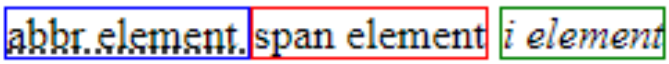
Some of the block-level elements in HTML are:

<address>	<figcaption>	<noscript>
<article>	<figure>	
<aside>	<footer>	<p>
<blockquote>	<form>	<pre>
<canvas>	<h1>-<h6>	<section>
<dd>	<header>	<table>
	<hr>	<tfoot>
<dl>		
<dt>	<main>	<video>
<fieldset>	<nav>	

Inline elements do not start on a new line; instead, it only takes up to the space as needed within the element content. They can set width and height values.

```
<abbr title="Abbreviation Template" style = "border:1px solid blue;"> abbr element </abbr>
<span style = "border:1px solid red;"> span element</span>
<i style = "border:1px solid green;">i element</i>
```

HTML code



Some of the inline-level elements in HTML are:

<a>	<i>	<small>
<abbr>	<img>	
<acronym>	<input>	
<b>	<kbd>	<sub>
<bdo>	<label>	<sup>
<big>	<map>	<textarea>
 	<object>	<time>
<button>	<output>	<tt>
<cite>	<q>	<var>
<code>	<samp>	
<dfn>	<script>	
	<select>	

2.2. Working with *lists* in HTML

There are many occasions when we need to use lists. HTML provides us with three different types: Ordered lists, unordered lists, definition lists.

Ordered listed

Ordered lists are lists where each item in the list is numbered. For example, the list might be a set of steps for a recipe that must be performed in order, or a legal contract where each point needs to be identified by a section number.

The ordered list is created with the **<ol>** element.

Each item in the list is placed between an opening **
** tag and a closing **
** tag. (The **li** stands for list item.)

DISPLAY

How to pass ET570 with a good grade:

1. Never miss classes
2. Always participate in class activities and finish them on time
3. Dedicate at least 2 hours of my free time to practice the codes I learned in my class
4. Seek help from my professor, classmates, or tutors in case I do not understand the class material
5. Never give up!

HTML

```
<p><b>How to pass ET570 with a good grade:</b>
    <ol>
      <li>Never miss classes</li>
      <li>Always participate in class activities and finish them on time</li>
      <li>Dedicate at least 2 hours of my free time to practice the codes I learned in
      my class</li>
      <li>Seek help from my professor, classmates, or tutors in case I do not
      understand the class material</li>
     <li> Never give up!</li>
    </ol>
</p>
```

Ordered List – "type" attribute

Attributes provide additional information about HTML elements.

The **type** attribute of the <ol> tag, defines the type of the list item marker. Attributes provide additional information about HTML elements.

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

DISPLAY

Ordered List with Letters

A. Coffee
B. Tea
C. Milk

HTML

```
<h2>Ordered List with Letters</h2>
<ol type="A">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Bullet or unordered listed

- Unordered lists are lists that begin with a bullet point (rather than characters that indicate order).
- The unordered list is created with the **<ul>** element.
- Each item in the list is placed between an opening **
** tag and a closing **
** tag.

DISPLAY

```
  <ul><b>Lists of fruits</b>
      <li>Apples</li>
      <li>Oranges</li>
      <li>Grapes</li>
  </ul>
```

Definition listed

Definition lists are made up of a set of terms along with the definitions for each of those terms.

The definition list is created with the **<dl>** element.

Inside the <dl> element you will usually see pairs of **<dt>** and **<dd>** elements.

<dt> is used to contain the term being defined (the definition term).

<dd> is used to contain the definition.

HTML

```
<dl>
      <dt><b>ET-570 Creating Smartphone Applications</b></dt>
            <dd>This course introduces the use and features of smartphones in modern
            life and how to create working applications. </dd>
      <br/>
      <dt><b>ET-710 Web Technology: Building and Maintaining Web Sites</b></dt>
            <dd>ET 710 focuses on the skills needed to build, maintain, and
            administrate a website.  </dd
</dl>
```

DISPLAY

ET-570 Creating Smartphone Applications
This course introduces the use and features of smartphones in modern life and how to create working applications.

ET-710 Web Technology: Building and Maintaining Web Sites
ET 710 focuses on the skills needed to build, maintain, and administrate a website.

2.3. Tables

There are several types of information that need to be displayed in a grid or table. For example: sports results, stock reports, and train timetables.

Basic tables structure

The **<table>** element is used to create a table. The contents of the table are written out row by row.

You indicate the start of each row using the opening **<tr>** tag. (The tr stands for table row). At the end of the row you use a closing **</tr>** tag.

Each cell of a table is represented using a **<td>** element. (The td stands for table data.)

Example) Create a table with the following information

DISPLAY

Column 1	Column 2	Column 3
45	60	45
60	90	90

HTML

```
<table>
  <tr> <td>Column 1</td> <td> Column 2</td> <td>Column 3</td> </tr>
  <tr align="center"> <td> 45</td> <td >60</td> <td>45</td></tr>
  <tr align="center"><td>60</td> <td>90</td> <td>90</td> </tr>
</table>
```

The **<th>** element is used just like the **<td>** element but its purpose is to represent the heading for either a column or a row. (The **th** stands for table heading.)

You can use the **scope** attribute on the **<th>** element to indicate whether it is a heading for a **column** or a **row**. It can take the values row to indicate a heading for a row or col to indicate a heading for a column.

Example) Create the following table

DISPLAY

	Saturday	Sunday
Tickets sold:	120	135
Total sales:	$600	$675

HTML

```
<table>
    <tr> <th></th> <th>Saturday</th> <th>Sunday</th></tr>
    <tr><th>Tickets sold:</th> <td align='center'>120</td> <td align='center'>135</td></tr>
    <tr><th>Total sales:</th> <td align='center'>$600</td> <td align='center'>$675</td></tr>
</table>
```

Border and background

The **`border`** attribute was used on both the <table> and <td> elements to indicate the width of the border in pixels.

The **`bgcolor`** attribute was used to indicate background colors of either the entire table or individual table cells.

Color can code in three ways:

- By the color name such as red, blue, green, black, lightblue, etc
- By the hex color palette code. You can get the hex color palette in: http://www.color-hex.com/color-palettes
- By the combination of RGB code.

Example) Create a following table with table border = 2px, table background ***yellow*** , 2nd row to ***pink*** , and one cell to ***lightgreen***

HTML

```
<table border="2" bgcolor="yellow">
    <tr>
        <th width="100"></th><th width="100">Withdrawn</th><th width="100">Credit</th>
        <th width="100">Balance</th></tr>
    <tr align="center" bgcolor="pink">
        <th>January</th><td>$250.00</td><td>$660.50</td><td>$410.50</td></tr>
    <tr align="center">
        <th>February</th><td>$135.55</td><td>$895.20</td><td bgcolor="lightblue">$1170.15</td></tr>
</table>
```

DISPLAY

	Withdrawn	Credit	Balance
January	$250.00	$660.50	$410.50
February	$135.55	$895.20	$1170.15

2.4. Managing images in HTML

There are many reasons why you might want to add an image to a web page: you might want to include a logo, photograph, illustration, diagram, or chart.

Images should…

- Be relevant
- Convey information
- Convey the right mood
- Be instantly recognizable
- Fit the color palette

Stock photos

If you do not have photographs to use on your website, there are companies who sell stock images or icons:

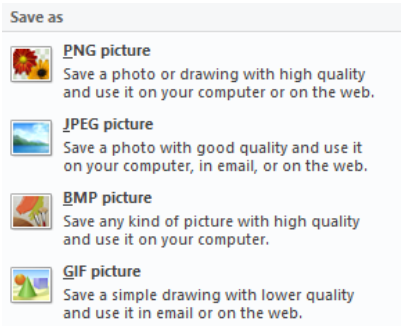
- www.pexels.com
- www.iconfinders.com

What is the best screen size to design for?

It is important to understand the size of the computer screen in order to coordinate the size of the image. Based on the worldwide screen resolution stats (nov 2016 – nov 2017) chart above and the charts below, these are the resolutions you need to be aware of in 2018.

- 360×640 – 21.54%
- 1366×768 – 12.85%
- 1920×1080 – 7.76%
- 375×667 – 4.94%
- 1440×900 – 3.32%
- 1280×800 – 2.67%

Image formats



PNG Portable Network Graphics

JPG or JPEG Joint Photographic Experts Group

BMP BitMaP

GIF Graphics Interchange format

The images on your computer screen are made up of lots of tiny squares known as pixels. The resolution of the screen is the number of pixels represented on it, and on most computers you can increase and decrease this number.

JPEG offers good quality when the image has many different colors.

GIF or PNG images are low quality images which are good for images with few colors or large areas of the same color (flat color). Example of them are logos, illustrations, and diagrams.

Exercise) which format would you save the following images?

Image dimensions

The images you use on your website should be saved at the same width and height that you want them to appear on the page.

Cropping image

When cropping images it is important not to lose valuable information. It is best to source images that are the correct shape if possible.

Fill to screen view - Portrait

Cropped to landscape view

Fill to screen view - landscape

SVG - scalable vector graphics

The other image format is SVG - scalable vector graphics.

The **<image>** SVG element includes images inside SVG documents. It can display raster image files or other SVG files. A ***raster image*** is an image file defined as a grid of pixels. They're also referred to as *bitmaps*. Common raster image formats on the Web are JPEG, PNG, GIF, and ICO.

The only image formats SVG software must support are JPEG, PNG, and other SVG files.
Animated GIF behavior is undefined.

Using SVGs is an easy choice once you consider the advantages they offer. For a client, you get superb quality on any device. For us as developers, there are even more reasons to use SVG.

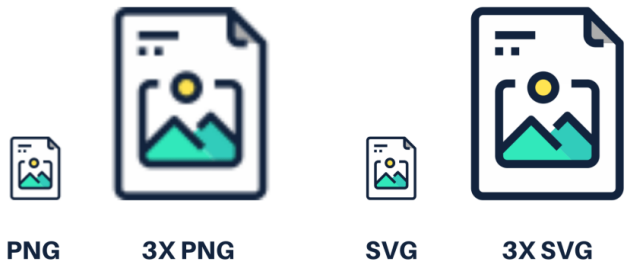
Let's discuss some of the benefits of SVG now.

1. Text-based format

SVG elements contain text, which greatly improves the accessibility of a website. But the main advantage is that this text is indexed by search engines. And a user can find an SVG file via Google.

2. Scalability

The quality of SVG images does not depend on the resolution. Unlike images of other formats or icon fonts, SVGs look perfectly sharp on any device with any screen size. Scalability also means that if you use the same image throughout the website but in different sizes, you use a single SVG. You do not have to create multiple copies of it as in the case of PNG. Instead, you embed the same image and define the size of it directly in SVG code.



3. High performance

If you prioritize performance, you should use SVG. With SVG, there is no need for an HTTP request to load in an image file. The page loads faster as it has no files to download. Faster loading time translates into better webpage performance and higher search engine ranking. In turn, it improves user experience.

4. Small file size

The size of simple SVG files is defined by the colors, layers, gradients, effects, and masks that it contains. The size of a PNG or any other raster graphics file is defined by the number of pixels that it consists of. The larger a PNG image is, the heavier it gets in size. This is not the case for SVG icons, though. Also, SVGs can be optimized, and I will tell how later in this article.

5. Numerous editing and animating opportunities

Unlike raster images, vector images can be edited both in special vector drawing programs and directly in a text editor. You can also edit colors or sizes of SVG icons directly via CSS. As for animating SVGs, it can be done with the help of SMIL, Web Animations API, WebGL, or CSS animation. Scroll down to learn more about CSS animation of SVG images.

6. Integration with HTML, XHTML, and CSS

SVG was designed "to integrate with and extend other prominent open Web platform technologies, such as X/HTML, CSS, and Javascript", according to W3C. So, unlike different image formats, this format can be easily integrated with other documents and technologies.

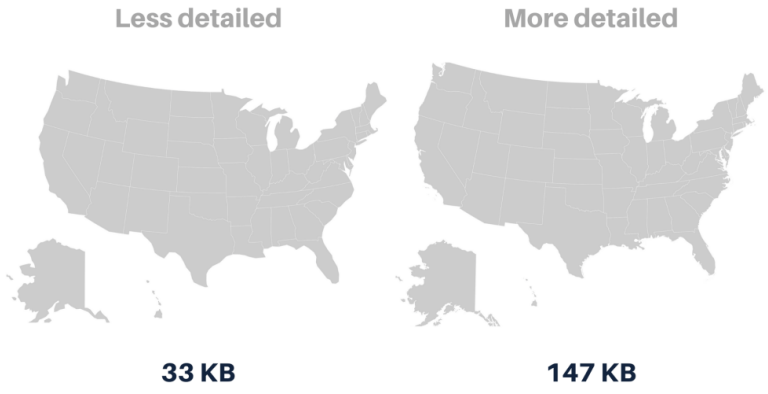
7. W3C Document Object Model support

There is growing community support for SVG. The World Wide Web Consortium (W3C) has always claimed that the Internet cannot do without vector images. This organization basically created the SVG format, and they actively support it nowadays.

What Are the Inconveniences of SVG?

The large number of small parts makes the use of the SVG format irrational. The more parts an image consists of, the heavier it grows in size.

For example, here are two SVG maps of the United States. The second one is slightly more detailed than the first one. But the higher level of detail cost almost a fivefold increase in file size – 33 KB compared to 147 KB. If this map was not monochromatic, the increase would be much greater.

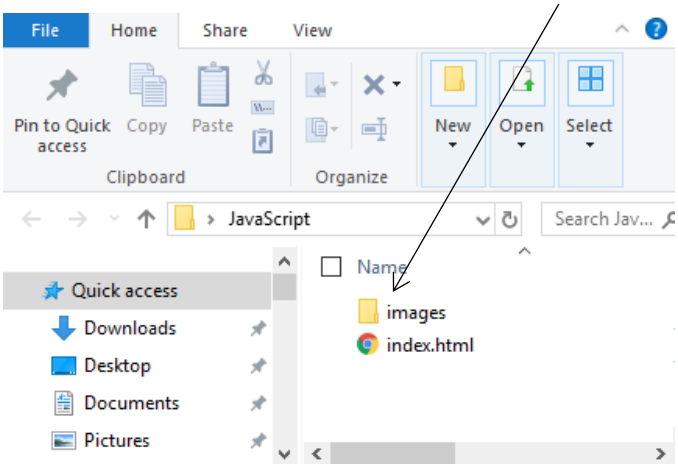


If the picture is linear and contains a few colors – SVG is a solution. However, if the details matter and there are a lot of them, PNG or JPEG may be more suitable. Also note that SVG cannot be used for

photographs. If you use a photograph on your website, SVG is not the best option. You definitely should go with a raster image format.

Store Images

As a website grows, keeping images in a separate folder helps you understand how the site is organized. Usually they are stored in sub-folder called “images”.



On a big site you might like to add subfolders inside the images folder. For example, images such as logos and buttons might sit in a folder called interface, product photographs might sit in a page called products, and images related to news might live in a folder called news.

Adding images to a webpage

To add an image into the page you need to use an **`<img>`** element. This is an empty element (which means there is no closing tag). It must carry the attribute src.

<img> attributes

`src` tells the browser where it can find the image file. This will usually be a relative URL pointing to an image on your own site.

`alt` provides a text description of the image which describes the image if you cannot see it.

`title` provides additional information about the image. Most browsers will display the content of this attribute in a tooltip when the user hovers over the image.

`height` specifies the height of the image in pixels.

`width` specifies the width of the image in pixels.

Alignment of an image in a webpage

The align attribute was commonly used to indicate how the other parts of a page should flow around an image. It has been removed from HTML5 and new websites should use CSS to control the alignment of images.

The align attribute can take these values:

left aligns the image to the left (allowing text to flow around its right-hand side).

right aligns the image to the right (allowing text to flow around its left-hand side).

top aligns the first line of the surrounding text with the top of the image.

middle aligns the first line of the surrounding text with the middle of the image.

bottom aligns the first line of the surrounding text with the bottom of the image.

HTML

```
<h3 align="center">Images and attributes</h3>
<img src="QCC_Tiger.jpg" alt="Logo of QCC tiger" title="This is tile of logo"
height="200px" width="200px" border="2px" align="right"/>
```

DISPLAY

HTML5: figure and figure caption

HTML5 has introduced a new **<figure>** element to contain images and their caption so that the two are associated.

You can have more than one image inside the **<figure>** element as long as they all share the same caption.

The **<figcaption>** element has been added to HTML5 in order to allow web page authors to add a caption to an image.

HTML

```
<h3 align="center">Images using figure and attributes</h3>
<figure>
  <img src="QCC_Tiger.jpg" alt="Logo 1" height="90px" width="90px" border="1px">
  <img src="QCC_Tiger.jpg" alt="Logo 2" height="90px" width="90px" border="1px">
  <img src="QCC_Tiger.jpg" alt="Logo 3" height="90px" width="90px" border="1px">
  <figcaption>This is a caption using tab figcaption <br/><i>QCC Logos </i>
  </figcaption>
</figure>
```

DISPLAY

Images using figure and attributes

This is a caption using tab figcaption
QCC Logos

2.5. Links

Links are the defining feature of the web because they allow you to move from one web page to another — enabling the very idea of browsing or surfing

Writing Links

Links are created using the **<a>** element. Users can click on anything between the opening **<a>** tag and the closing **** tag.

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

Some of **<a>** attributes are **href** and target:

- **href:** Specifies the URL of the page the link goes to.
- **target:** Specifies where to open the linked document. Some of the value of attribute target are: _blank, _parent, _self, _top

Linking to other sites

Links are created using the **<a>** element which has an attribute called **href**. The value of the href attribute is the page that you want people to go to when they click on the link.

DISPLAY

Linking to other sites

For more information about the college, click in the following link: Queensborough Community College

HTML

```
<h3>Linking to other sites </h3>
<p>For more information about the college, click in the following link:
<a href="http://www.qcc.cuny.edu">Queensborough Community College</a></p>
```

Opening Links in a New Window

If you want a link to open in a new window, you can use the **target** attribute on the opening <a> tag. The value of this attribute should be **"blank".**

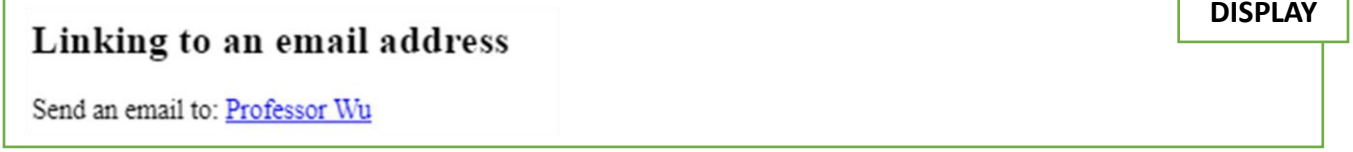
HTML

```
<h3>Linking to other sites </h3>
<p>For more information about the college, click in the following link:
<a href=http://www.qcc.cuny.edu target=_blank>Queensborough Community
College</a></p>
```

Email Links

To create a link that email to a specified email address, you use the **<a>** element, follow by the attribute **href** starts with **mailto**: and is followed by the email address you want the email to be sent to.

DISPLAY



HTML

```
<h3>Linking to an email address </h3>
<p>Send an email to: <a href="mailto:hwu@qcc.cuny.edu">Professor Wu</a></p>
```

Linking to a specific part of the same page

Before you can link to a specific part of a page, you need to identify the points in the page that the link will go to. You do this using the **id** attribute (which can be used on every HTML element).

The value of the **id** attribute should start with a letter or an underscore (not a number or any other character) and, on a single page, no two id attributes should have the same value.

To link to an element that uses an **id** attribute you use the <a> element again, but the value of the href attribute starts with the **#** symbol, followed by the value of the **id** attribute of the element you want to link to.

Example)

identify the part of your page where the **id bottom** will return when is clicked:

```
<img src="images/ny.jpg" id="bottom" alt="New York City">
```

When the word ***Go to Image*** is clicked, it will link to an element with id **bottom:**

```
<p><a href="#bottom">Go to Image</a></p>
```

This is possible if the image is at the same webpage. If the link is in the other webpage, the code has to link to the webpage that it has followed by the id name:

```
<p><a href="index.com#bottom">Top</a></p>
```

Navigation Bars

Having easy-to-use navigation is important for any web site.

Later in the class, you will use CSS to transform the HTML menus into good-looking navigation bars. For now, you can create navigation bar using unordered list.

Example) Create navigation bars using <ul> element.

DISPLAY

Navigation Bars Using "ul"

- Different type of lists
- Tables and Attributes
- Images and Figure
- Different type of links

HTML

```
<h3>Navigation Bars Using "ul"</h3>
<ul>
  <li><a href="#list1">Different type of lists</a></li>
  <li><a href="#tables2">Tables and Attributes</a></li>
  <li><a href="#images">Images and Figure</a></li>
  <li><a href="#links">Different type of links</a></li>
</ul>
```

In this case, we are using internal links that will link to a specific part of the same page.For that, we have to mark the element that the link has as reference using the `id` attribute. For example, to make an internal link to the Different type of lists, we can id el `<ol>` element with `id="list1"`

HTML

```
<ol id="list1">
      <li>Never miss classes</li>
      <li>Always participate in class activities and finish them on time</li>
       <li>Dedicate at least 2 hours of my free time to practice the codes I learned in
       my class</li>
      <li> Seek help from my professor, classmates, or tutors in case I do not
      understand the class material</li>
      <li> Never give up!</li>
</ol>
```

The **<nav>** tag defines a set of navigation links in an inline form. The **nav** tag is not supported in Internet Explorer 8 and earlier versions.

DISPLAY

Navigation Liks using

QCC | CUNY Center |

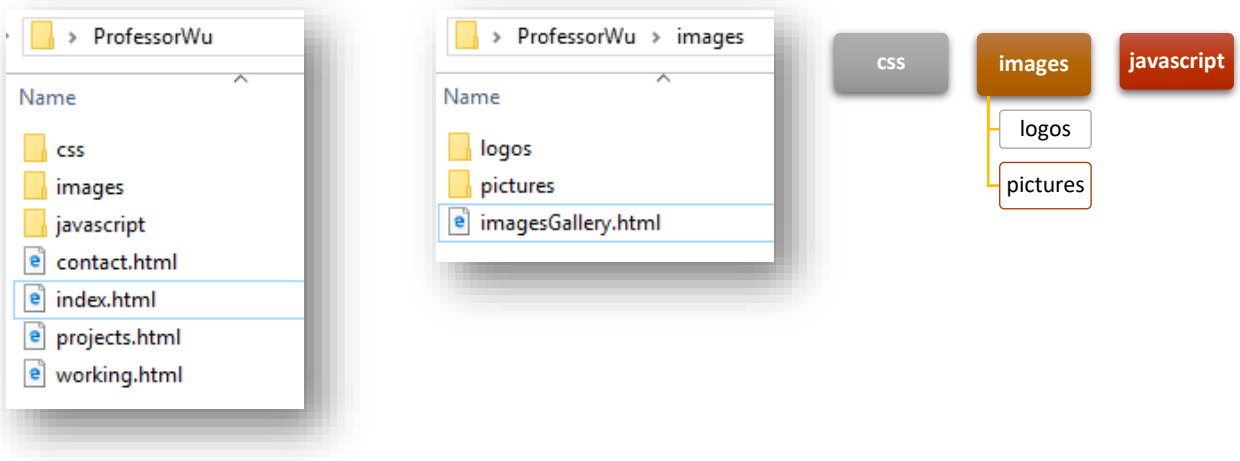
HTML

```
<h3> Navigation Liks using </h3>
<nav>
    <a href="http://www.qcc.cuny.edu" target="blank">QCC |</a>
    <a href="http://www2.cuny.edu/" target="blank">CUNY Center |</a>
</nav>
```

Linking to other pages on the same site

To link the pages within the same, you will need to create and organize the html files in their respective folder. You will need to create parents folder, child folders, grandchildren folders (if it is necessary), etc.

The main homepage of a site written in HTML (and the homepages of each section in a child folder *recommended) is called **index.html**



How to link a webpage within the same site?

- To link to a file in the same folder, just use the file name

```
<p><a href="projects.html">Professional Projects</a></p>
```

- To link a file in a child folder, use the name of the child folder, followed by a forward slash, then the file name

```
<p><a href="images/imagesGallery.html">Project Gallery Images</a></p>
```

- To link to the homepage from a child folder, or to one folder before it, use **../**

For example, if you are writing a code in webpage imagesGallery.html, which is located in a child folder “images”, and you want to make a link to go back to the index page, which is located in a parent folder, you will write the code as the following:

```
<p><a href="../index.html">Go back to homepage</a></p>
```

How to link an image to other side

To use an image as a link, put the <img> tag inside the <a> tag:

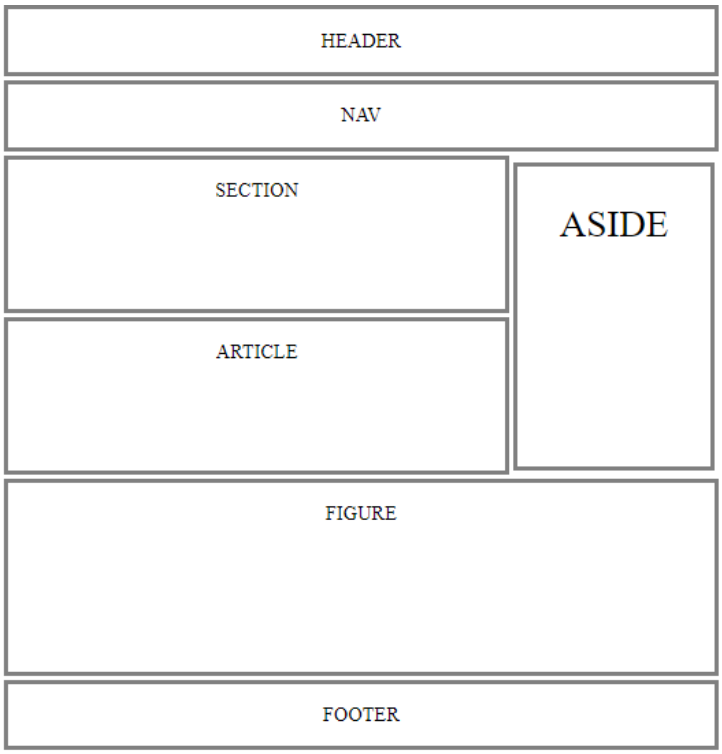
HTML

```
<h3>How to link an image to other side</h3>
<a href="http://www.qcc.cuny.edu/index.html" target="_blank">
<img src="QCC_Tiger.jpg" alt="Logo of QCC tiger" title="This is tile of logo"
height="200px" width="200px" /></a>
```

DISPLAY

How to link an image to other side

2.6. Semantic elements



Semantic elements are elements with a meaning and clearly describe its meaning to both, the browser and the developer.

There are some semantic elements in HTML that can use to create the layout of a webpage as shown in the image on the left:

Some of the semantic elements are:

Tag	Description
<article>	Defines independent, self-contained content
<aside>	Defines content aside from the page content
<details>	Defines additional details that the user can view or hide
<figcaption>	Defines a caption for a <figure> element
<figure>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<footer>	Defines a footer for a document or section
<header>	Specifies a header for a document or section
<main>	Specifies the main content of a document
<mark>	Defines marked/highlighted text
<nav>	Defines navigation links
<section>	Defines a section in a document
<summary>	Defines a visible heading for a <details> element
<time>	Defines a date/time

Bibliography

Duckett, J. (2016). *HTML and CSS Design and build websites.* Indianapolis: John Wiley and Sons Inc.

Duckett, J. (2016). *JavaScript and JQuery: interactive front-end developer.* Indianapolis: John Wiley and Sons Inc.

HTML, CSS, JavaScript, JQuery, and Bootstrap. (2017, June). Retrieved from w3schools: www.w3schools.com

Some material compiled from www.lynda.com (May 2018), www.w3schools.com (2020), www.coursera.org (2020)