

## Learning Module

|                     |            |                                                         |
|---------------------|------------|---------------------------------------------------------|
| <b>Subject Code</b> | CS3        | Client-side Web Development                             |
| <b>Module Code</b>  | 2.0        | Basic HTML                                              |
| <b>Lesson Code</b>  | 2.3<br>2.4 | img, a, br, p<br>Tag-specific attributes (href and src) |
| <b>Time Frame</b>   | 30 mins.   |                                                         |



### TARGET

**Time allocation:** 1 minute

**After completing this module, you are expected to:**

- Proficiently use basic HTML tags
- Create your own webpages



### HOOK

**Time allocation:** 1 minute

In the previous module, you have learned about basic HTML tags used to contain text and to contain other tags.

HTML tags to be discussed in this section is outlined as follows:

- `<img>`
- `<a>`
- `<p>`
- `<br>` and `<hr>`

**NOTE:** All of these tags will be placed inside the `<body>` tag.



### IGNITE

**Time allocation:** 6 minutes.

#### The `<p>` Element

- Used to place paragraphs of text in the HTML code, thus the “p”
- It is a block element.
- Enclose all contents of the `<p>` element with `</p>`

## The <br> and <hr> Elements

- <br> is used to add a blank line after HTML elements, much like the “Enter” in your keyboard when using MS Word. It is a block element.
- <hr> is used to add a full-width horizontal line to the document. It is a block element.
- Both tags do not need any text inside it, therefore a closing tag is not needed.

## The <img> tag

- This tag is used to add images to the webpage.
- It is an inline element.

```

```

- Inside **src**, type in the file name of the image. Take note that file name is case sensitive. Files can be .jpg, .png, .gif, .svg, among others.
- Inside **alt**, type in an alternative text that will be displayed instead when the browser was not able to load the image.
- You can also set the height and width of an image. Remember from the last lesson that the width of inline elements cannot be modified.

This is an exception of an inline element wherein its width could be modified.

```

```

- You can also add a title for an image. It will not be displayed on the webpage but will show as a tooltip when a user hovers over the image. You can also use the title attribute to cite the source of the image.

```

```

## The <a> tag

- This tag is used to add links to webpages.
- It is an inline element.
- Links are used to direct a user to other web pages or files when clicked.
- Links can either be text, images, or any other HTML element, depending on the design of the webpage.

```
<a href="url">Click here!</a>
```

- Inside **href**, type in the filename of the webpage or file that you want to link to. It can be another HTML file, an image file, or even a pdf file. Use the complete URL format when you want to link to a website.

- You can replace the text “**Click here!**” with any other text that you want to use to prompt a user to click. You can also use an image instead of text for a user to click on. Below is a sample code:

```
<a href="url">
    
</a>
```

- Aside from directing a user to a page outside of it, links can also be used to direct a user to another part of the page. These are called **anchor** tags. You may have already encountered websites with “Back to Top” links at the bottom of the page. Refer to the code below, when a user clicks on Back to top, the browser will go to the element of a web page having an id called “top”.

```
<a href="#top"> Back to top </a>
```

### Tag-specific attributes: src and href

You may have noticed that for the code of `<img>` and `<a>`, there are words typed inside the `< >` symbols aside from the tag name.

```
<a href="url">Click here!</a>
```

Content

HTML Element

HTML tag attribute value

These are called **attributes**. Some tags have attributes that are applicable only on them.

Example: `href` is an attribute specific to `<a>`  
`src` is an attribute specific to `<img>`

There are attributes applicable for a lot of HTML tags, example is **name**, **id**, and **class**. These will be discussed in the next modules.

Each attribute has their own set of **values**.

- The possible values for **href** is the file name or URL of the document where you want to direct a user to when the link is clicked.
- The possible value for **src** is the file name of the image that you want to display with the `<img>` tag. You can also use the URL of an image instead of a file name.

### Additional Notes: Using File Paths

When typing only the filename of the image or the document in `<img>` and `<a>`, make sure that the HTML document and the file/image are in the same folder.

But when there are dozens of images or links for a single HTML file, the folder where your files are located will become messy. In order to be more organized, it is a good practice to place all images in a folder dedicated only for images and another folder dedicated only for HTML files.

In order to access files or images located in other folders, we should use file paths. **The concept of file paths is applicable both for images and files for links.** Later in our lessons, you will learn that it is also applicable for stylesheets (CSS) and scripts (JavaScript).

Below are examples of file paths. Consider an image file named “picture.jpg”

```

```

The image is located in the same folder as the HTML file where it is coded. This is what we have demonstrated in our previous examples of links and images.

```

```

The image is located inside folderName. The HTML file where this is coded is saved in the same folder as folderName.

```

```

The image is located in folderName at the root of the current webpage

```

```

The image is located in the folder one level up from the current folder. Current folder means the folder where the HTML file for this code is saved in.



## NAVIGATE

**Time allocation:** 8 minutes

**Exercise:**(non-graded)

Execute the following instructions:

1. Create **two (2)** HTML files.
  - a. The first file will contain a header (use h1) and a link. The link will lead a user to the second HTML file. Refer to figure 1 for a sample.  
  
Save the file as **mod2-3\_surname\_1.html**.
  - b. The second file will contain a header, an image, and a simple description of the image. Feel free to choose any image files available to you. Use the <p> tag for the image description. Refer to figure 2 for a sample.  
  
Save the file as **mod2-3\_surname\_2.html**.
2. Remember to type in the complete basic structure of an HTML file. Place all HTML tags that are expected to be visible on the webpage inside the <body> tag.
3. Place the two files in a folder named **mod2-3\_surname**. When submitting to your teacher, remember to send multiple files as zipped files.

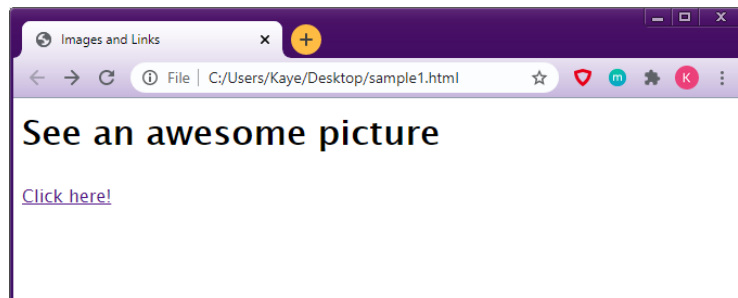


Figure 1. The first HTML file.

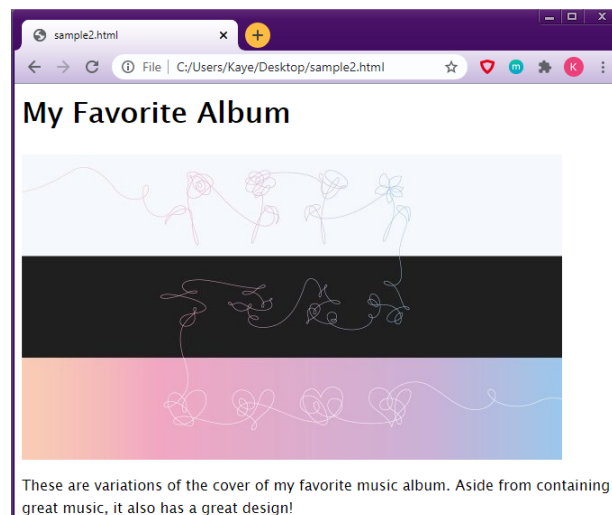


Figure 2. The second HTML file.



KNOT

**Time allocation:** 15 minutes

**Exercise** (graded)

This exercise will serve as a summary of everything that you have learned about Basic HTML. Take note that this is a graded exercise and you are required to submit your final output to your teacher.

1. Create **four (4)** HTML files.
  - a. All four files should contain the meta tags discussed in the previous lessons to document your work.
  - b. The first file will contain a header (use h1), a horizontal rule, and three links. Add a paragraph after the three links. The links will lead a user to three other HTML files. Name this file as **home.html**. Refer to figure 4 for a sample homepage.
  - c. Use 100px by 100px images for the links. Remember to add alternate texts to your images. Feel free to use any image files available to you.
  - d. The three other files will contain a header (use h1), a horizontal rule, and an unordered list. Each file will have the same format as the others but have different

content. Name the files as **page1.html**, **page2.html**, and **page3.html**. These will be subpages. Refer to figure 5 for a sample subpage.

- e. The unordered list in the subpages should contain at least three items.
2. All images used should be placed in a folder named *images*. All subpages should be placed in a folder named *files*. Use the concept of file paths in your code. Refer to figure 3 for a visualization of the expected file structure.
3. Save all your files in a folder named **exercise\_surname**. Remember to submit multiple files as zipped files.

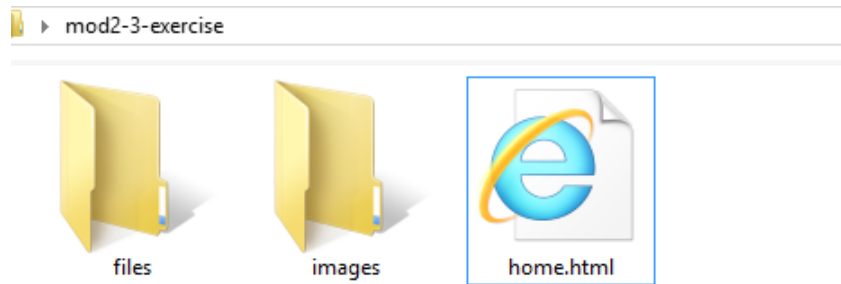


Figure 3. Sample folder contents

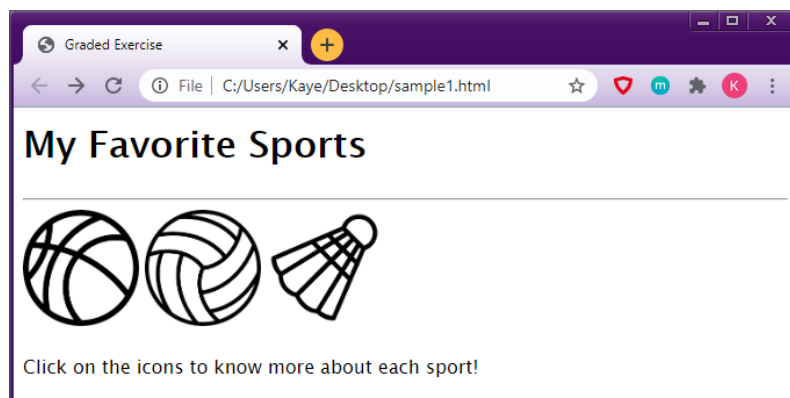


Figure 4. Sample homepage

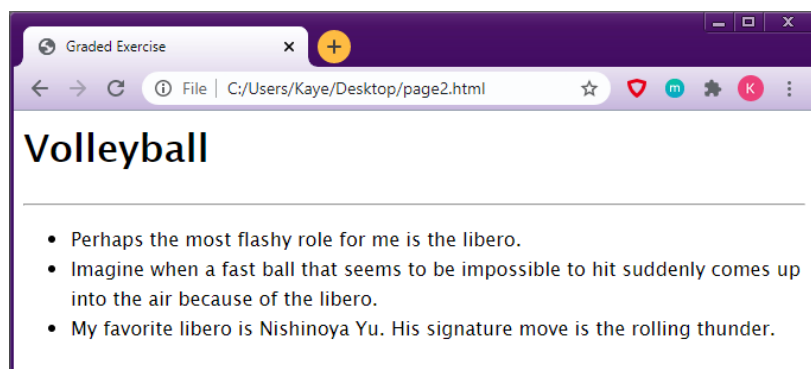


Figure 5. Sample subpage

## Grading Rubric

Perfect score: 30/30

|                              |                                                                                                                                                                                                     |                                                                                                                                                                                    |                                                                                                                                                                      |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Content and structure</b> | <b>11-15</b><br>All HTML tags listed in the specs are present in the final output<br><br>Placement of HTML tags is according to specs.<br><br>Necessary meta tags are also present in the HTML file | <b>6-10</b><br>1 or 2 HTML tags required in the specs are missing/wrongly implemented in the code<br><br>1 or 2 HTML tags are wrongly placed<br><br>Only two meta tags are present | <b>1-5</b><br>Three or more HTML tags in the specs are missing/wrongly implemented<br><br>Three or more HTML tags are wrongly placed.<br><br>1 or no meta tags used. |
|                              | <b>8-10</b><br>Code observes proper indentation.<br><br>All basic HTML tags are present (i.e. head, title, etc)<br><br>All HTML tags have their closing counterpart where applicable                | <b>5-7</b><br>1 or 2 lines of code are improperly indented<br><br>1 or 2 HTML tags with no closing pair (where applicable)                                                         | <b>1-4</b><br>Three or more lines of code are improperly indented<br><br>Three or more HTML tags with no closing pair (where applicable)                             |
| <b>File management</b>       | <b>3-5</b><br>Proper file naming and folder naming is observed.<br><br>All files are placed in the specified folders given in the specs                                                             | <b>1-2</b><br>1 or more files are not named properly / according to specs                                                                                                          |                                                                                                                                                                      |

## REFERENCES

*HTML File Paths*. (n.d.). Retrieved from [https://www.w3schools.com/html/html\\_filepaths.asp](https://www.w3schools.com/html/html_filepaths.asp): [https://www.w3schools.com/html/html\\_filepaths.asp](https://www.w3schools.com/html/html_filepaths.asp)

*HTML Images*. (n.d.). Retrieved July 13, 2020, from W3Schools: [https://www.w3schools.com/html/html\\_images.asp](https://www.w3schools.com/html/html_images.asp)

*HTML Links*. (n.d.). Retrieved July 13, 2020, from W3Schools: [https://www.w3schools.com/html/html\\_links.asp](https://www.w3schools.com/html/html_links.asp)

*HTML Paragraphs*. (n.d.). Retrieved July 13, 2020, from W3Schools: [https://www.w3schools.com/html/html\\_paragraphs.asp](https://www.w3schools.com/html/html_paragraphs.asp)

Prepared by:

KAYE B. ALAMAG  
Special Science Teacher II  
PSHS-CARC

Reviewed by:

ALINE TERESA MENDOZA  
SST V  
PSHS- Main