HTML!

Q1: Week 4

Written by Randolf Dela Cruz

what is HTML?

what is HTML?

HTML stands for **Hypertext Markup Language.**

Let's break it down!

what is HTML?

Hypertext[1] Markup Language[2]

1. Hypertext - A text that links to other texts. It's what enables websites!

2. Markup Language - The text-encoding system of HTML itself.

what is HTML?

Hypertext[1] Markup Language[2]

1. Hypertext - A text that links to other texts. It's what enables websites!

2. Markup Language - The text-encoding system of HTML itself.

Okay... but *why* should I care?

why care about HTML?

For one, you're in Reboot's SoftDev division! It's a large part of what we will be learning throughout the school year.
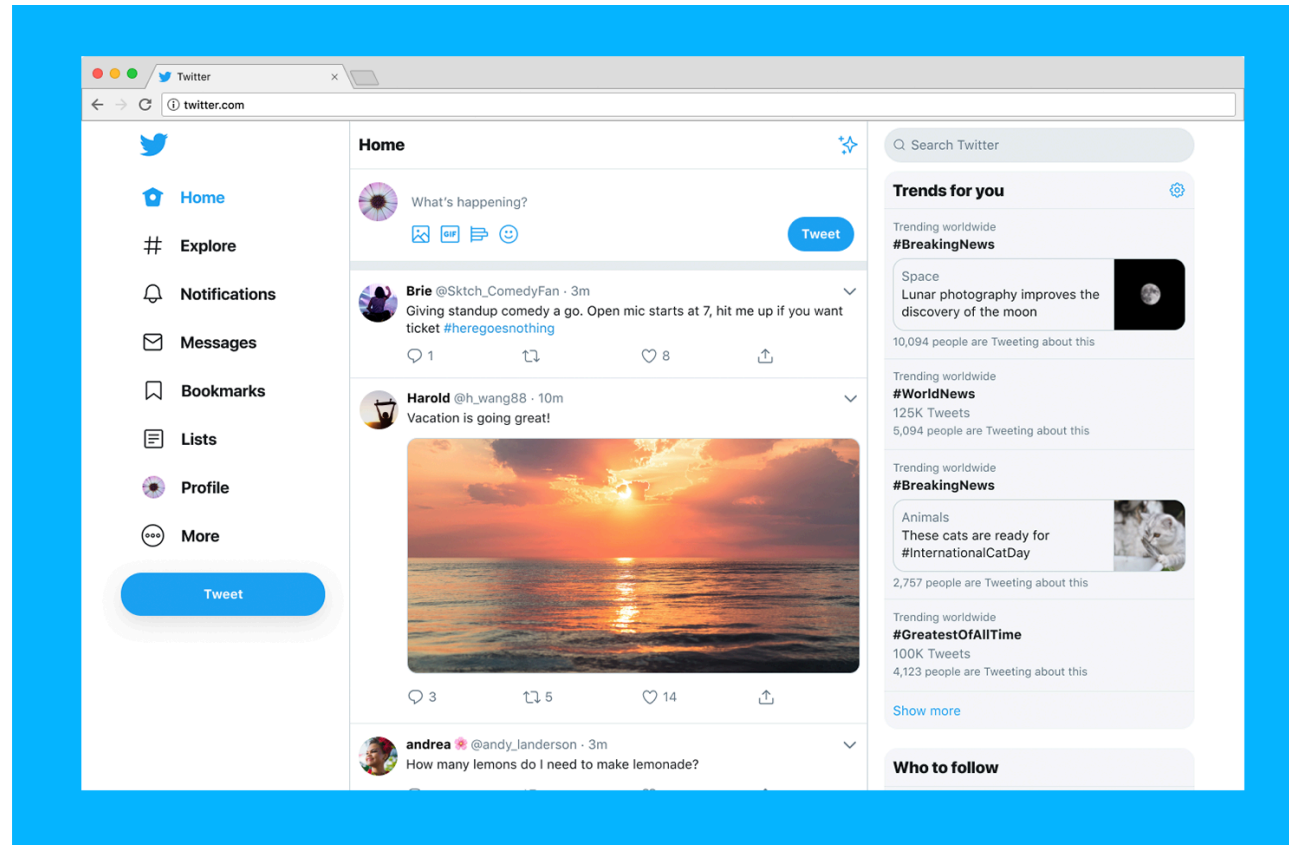
why care about HTML?

For one, you're in Reboot's SoftDev division! It's a large part of what we will be learning throughout the school year.

And two: HTML IS **EVERYWHERE!!!!!!!!!!** Because

HTML is the language of the web.

HTML is everywhere in the web!

For example, Twitter is built on HTML!

The **goal**, then, is to equip you with the skills to create web apps.
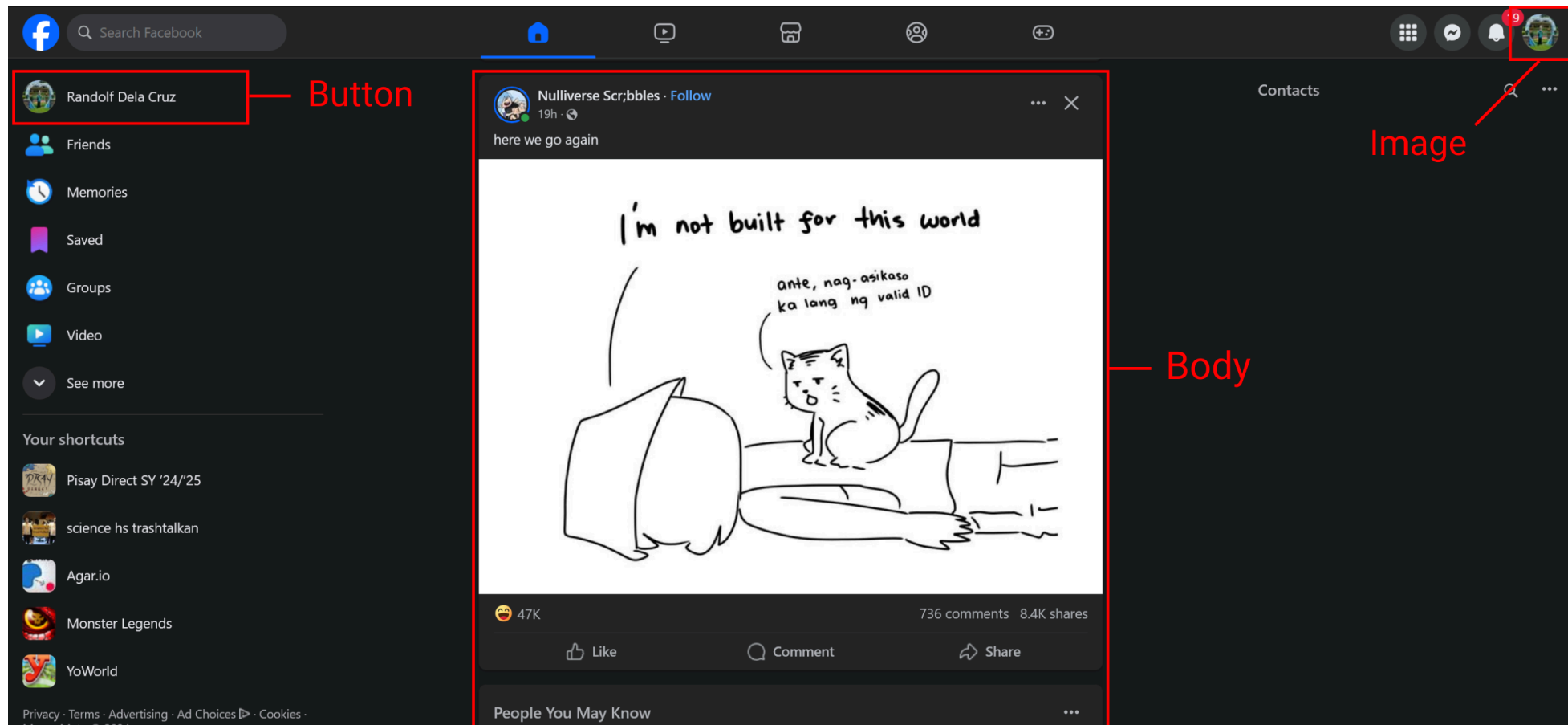
Think you can make the next Twitter?

(...please don't)

ok... how DOES HTML work?

let's get into details.

HTML Elements

HTML is essentially made up of individual *elements*.

HTML Elements

In HTML code, most elements are manually enclosed. These elements will have this general structure:

```
<tag> content </tag>
```

Below is an example of a **paragraph** <p> element, which adds a paragraph to the page!

```
<p> Hello darkness, my old friend </p>
```

HTML Elements

Some elements may be self-enclosing:

```
<img/> (an image)

<br/> (a new line)
```

The page's document type is often indicated by

```
<!DOCTYPE html>
```
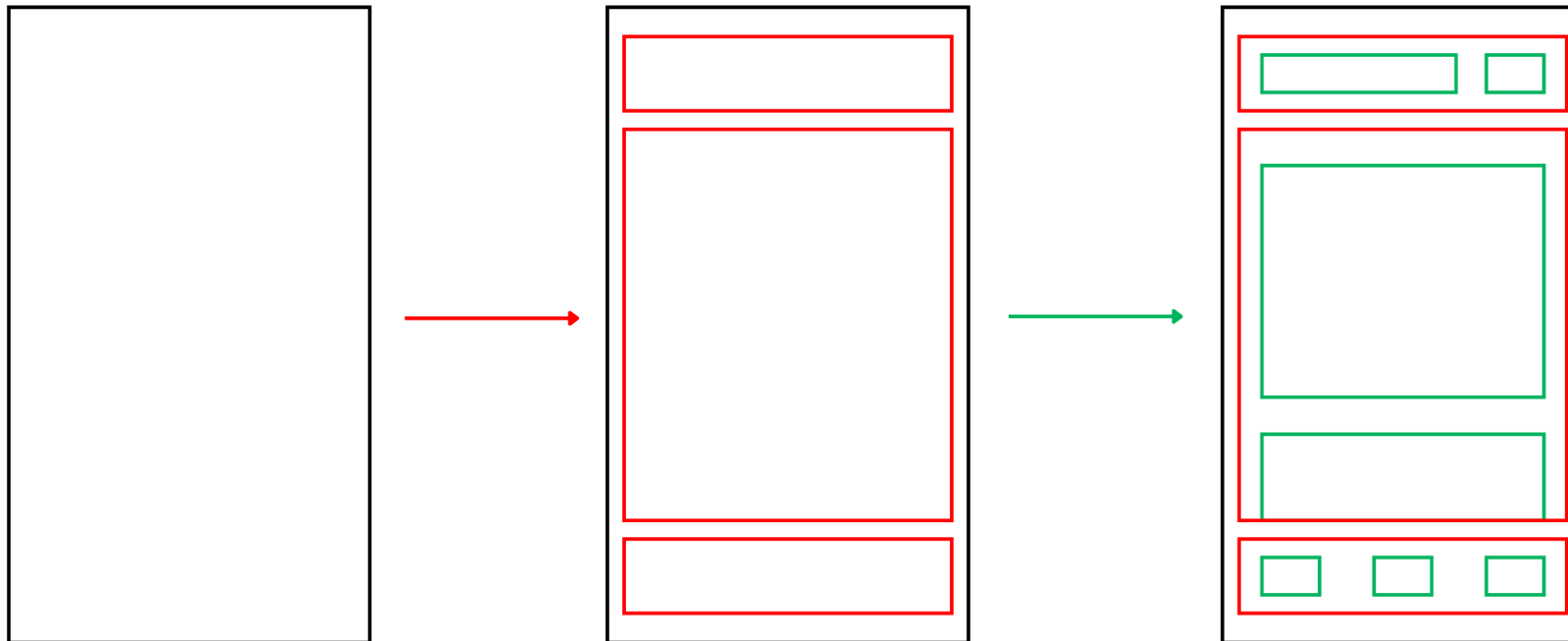
HTML Elements

Most importantly, HTML elements <u>can enclose other elements.</u>

This gives HTML its most important power: **element hierarchy**.

```
<body>
    <p> Element inside an element! </p>
</body>
```

HTML Element Hierarchy

Notice how element hierarchy is used below to arrange elements inside elements, creating a cohesive web layout.

HTML Element Hierarchy

When you're designing the layout of your webpage, it's <u>crucial</u> to think of the subdivision of the HTML elements into an intuitive hierarchy.

There are several semantic HTML elements that can guide you with this task.

HTML Element Hierarchy

The <**body**> element encloses the main body of content inside your webpage.

```
<body>
    <p> BREAKING NEWS: Pineapple doesn't belong on pizza </p>
</body>
```

HTML Element Hierarchy

The **<header>** element encloses content found at the top of your page, like introductory content or titles.

<h1> to **<h6>** are heading elements of decreasing importance.

```
<header>
    <h1> Totally Legitimate Unbiased News Platform </h1>
    <h2> Written by: Pineapple Hater </h2>
</header>

<body>
    <p> BREAKING NEWS: Pineapple doesn't belong on pizza </p>
</body>
```

HTML Element Hierarchy

The **\<footer>** element encloses content found at the bottom of your page, typically reserved for extra information.

```html
<header>
    <h1> Totally Legitimate Unbiased News Platform </h1>
    <h2> Written by: Pineapple Hater </h2>
</header>

<body>
    <p> BREAKING NEWS: Pineapple doesn't belong on pizza </p>
</body>

<footer>
    <p>The views shown here are not indicative of the views of Reboot as a whole.</p>
</footer>
```

HTML Element Hierarchy

The **\<nav\>** element encloses content which serve to navigate the user throughout the page or into another page.

```
<nav>
    <a href="my_home.html"> My Home </a>
    <a href="account_settings.html"> Account Settings </a>
</nav>
```

HTML Element Hierarchy

The **\<section>** element encloses *some part* your page. It could be the reviews section, the comments section, the posts section, or really... anything.

```
<body>
    <section>
        <p>Just went to Bohol last week!!</p>
        <img src="bohol_image.png"/>
    </section>
    <section>
        <p>When are they going to release the grades?????</p>
        <img src="annoyed_image.png"/>
    </section>
</body>
```
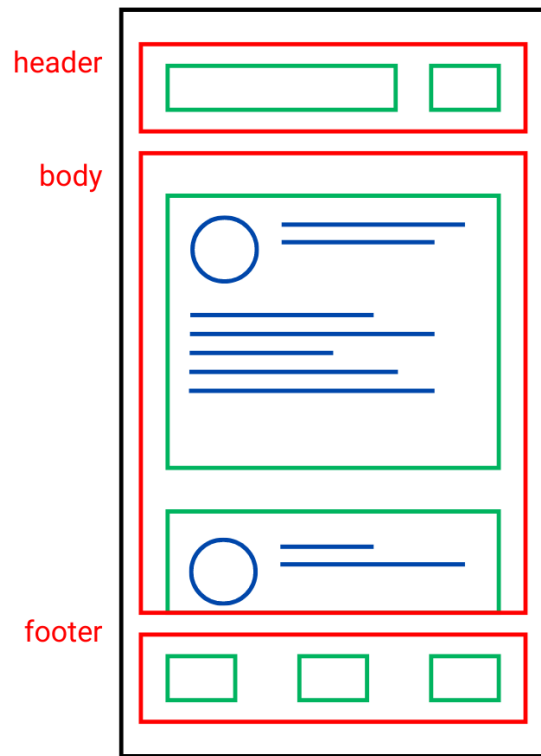
HTML Element Hierarchy

The **\<div>** element is a general-purpose container.

```
<div>
    <div>
        <div> Hi! </div>
    </div>
    <div>
        Hello!
    </div>
</div>
```
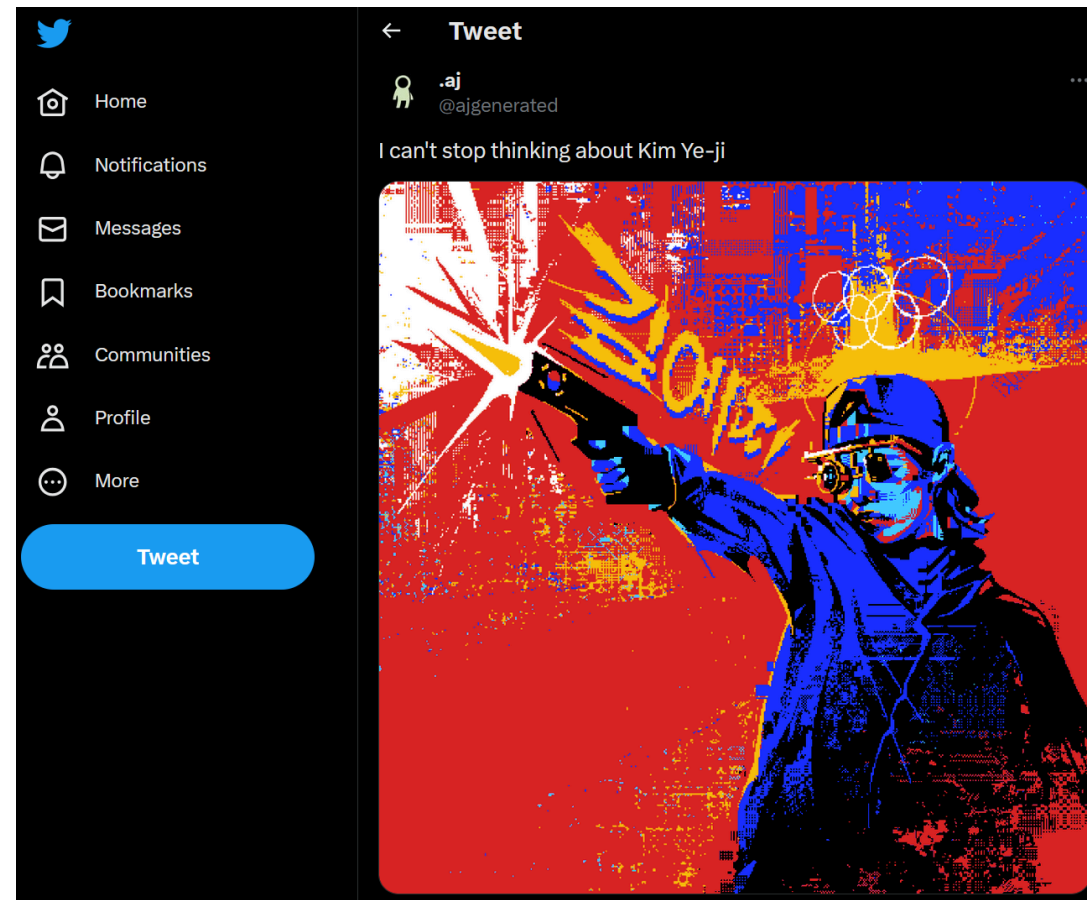
Example of HTML Hierarchy at Work

page
| header
| | section
| | section

page
| body
| | section
| | | image
| | | text
| | section
| | | image
| | | text

page
| footer
| | button
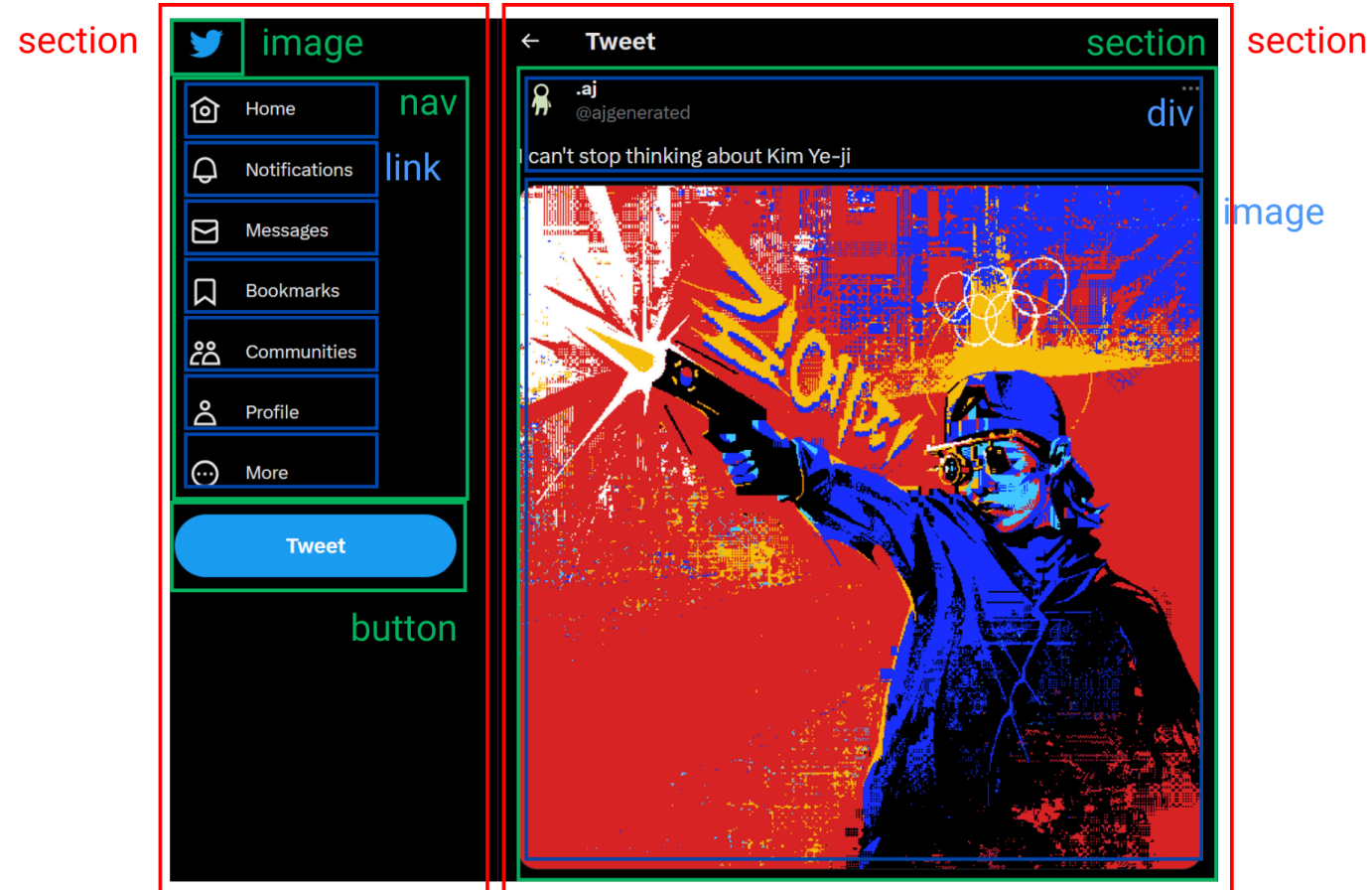| | button
| | button

Quick Exercise! (1-2 mins.)

Look at this picture of a Twitter page, and think about how the elements here could be subdivided into a hierarchy!

Quick Exercise! (1-2 mins.)

Here's one possible answer:

you are almost ready to make your own HTML page!

but we need to go over a few more details first.

HTML Attributes

Attributes indicate certain properties of an element. In code, their general structure looks like this:

```
<tag attribute="value" ...> </tag>
```

HTML Attributes

The **\<img\>** element encodes an image. It uses the `src` attribute for the image link.

```
<img src="my_image.png"/>
```

HTML Attributes

The `<a>` tag uses the `href` (hypertext reference) attribute to make a link.

```
<a href="https://www.youtube.com"> YouTube </a>
```

HTML Identifiers

Elements in your page may require an *identifier* so they can be identified by other elements.

The **id** attribute gives an ID to an element. IDs **cannot be shared**, and are unique to each element.

```
<p id="special_paragraph"> This paragraph has an identifier. </p>
```

HTML Identifiers Example

Here, the `label` element links itself to the `input` element using the input element's ID.

```
<input type="checkbox" id="UNIQUE_ID">

<label for="UNIQUE_ID"> this label is linked to the checkbox </label>
```

☐ this label is linked to the checkbox

HTML Identifiers

Likewise, the **class** attribute indicates the class of an element. Classes **are shared**, and can be used to identify a group of similar elements.

```
<p class="stupid_paragraphs"> This is a stupid paragraph. </p>
<p class="stupid_paragraphs"> This is also a stupid paragraph. </p>
```
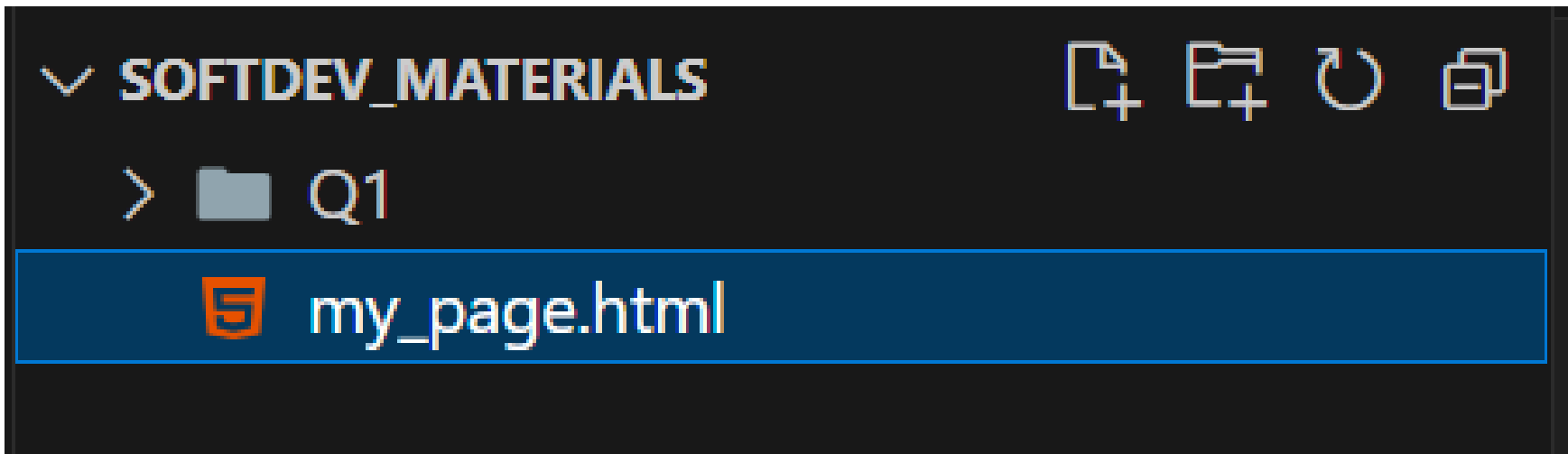
HTML Identifiers

By now, you should be able to see how **IDs** and **classes** can be used to identify HTML elements, allowing them to be called or referred to.

Later on, we'll see how these identifiers can be used to refer to elements even in external code, like CSS and JavaScript.

now let's try MAKING an HTML page!

HTML Setup

1. Open **VScode.**

2. Create a new file and name it **"my_page.html"**

Make an HTML Page! (~5-10 mins.)

Make a page with THESE characteristics:

- A **header** containing the title of your page.

- A **div** containing a **paragraph**. Write any message you want!

- A **section** inside the body, below the paragraph, containing a link to any website.

- A **section** inside the body, below the first section, containing any image.

- A **footer** with the ID: `my_footer`.

Feel free to ask for help from your seatmates or instructor/s! Try to get 5/5! <u>Be creative!</u>