HTML

Block 1: Introduction

A NOTE

One of the primary reasons why people fail and give up at programming — even at something as simple as HTML — is that one expects to grasp HTML and its concepts immediately.

You don’t just absorb everything and expect to be able to push it into fruition.

Programming is **using and understanding these concepts over and over until you can make even more complex creations.**

This is meant as an introduction for those who feel intimidated at grasping a programming language.

Not necessary for younger audiences, rather, for the older ones that are afraid of delving into these tools.

Encourage your learners to constantly **experiment, test, try,** and to not give up -- because this isn’t a skill you simply “learn” like a math lesson. Think of it as learning an instrument or language. (As it is the latter.)

Lesson 1

Objectives

- Build a basic webpage with the fundamentals of HTML content elements and tags
- Begin basic styling
- Understand and comprehend the syntax behind HTML and CSS
    - Properties and their identifiers

Topics

- Webpage Structure
- Tag Syntax
- Basic Elements
    - Formatting Elements
    - Breaks and Text Elements
- Styling
- Stylesheets

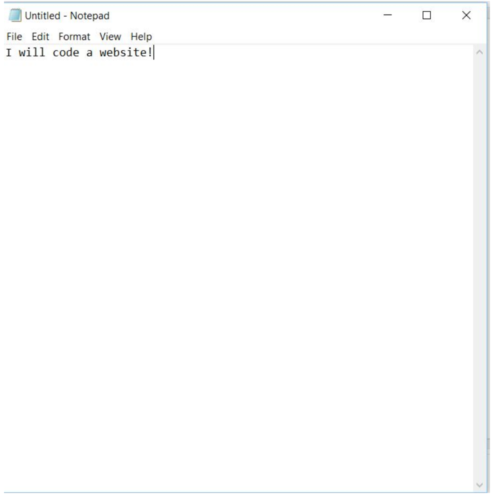
Set-up

Did you know that you can create webpages right from your computer?

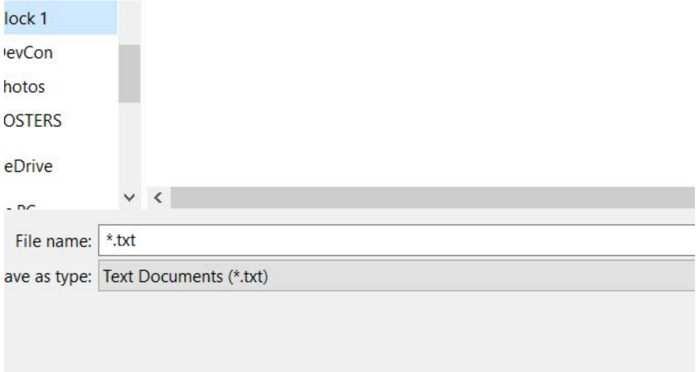
1. Open **Notepad** (or any other text editor)
2. Type anything!
3. Save your text file as **mypage**.html
4. Double-click the page to see your results

Guide students when performing these tasks. The next slides have image demonstrations of what these steps should consist of.

Make sure the file is saved as an html file (.html) and not a file.html.otherextension!

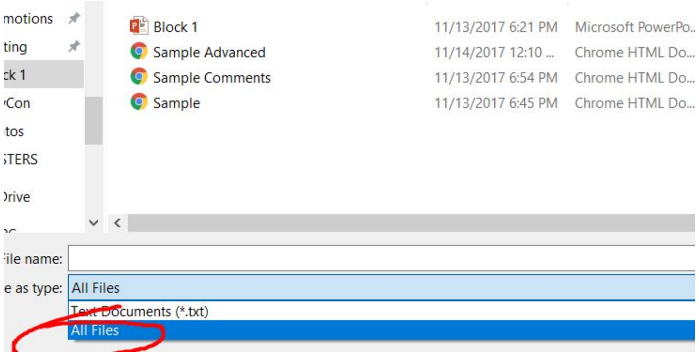


```
I will code a website!
```



File name: *.txt

ave as type: Text Documents (*.txt)



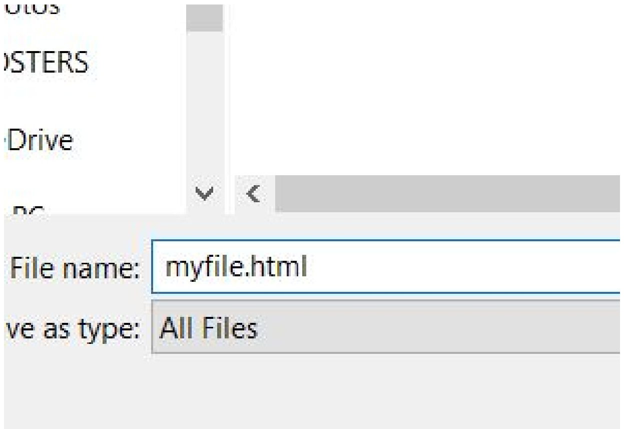
Block 1	11/13/2017 6:21 PM	Microsoft PowerPo...
Sample Advanced	11/14/2017 12:10 ...	Chrome HTML Do...
Sample Comments	11/13/2017 6:54 PM	Chrome HTML Do...
Sample	11/13/2017 6:45 PM	Chrome HTML Do...

ile name:

e as type: All Files

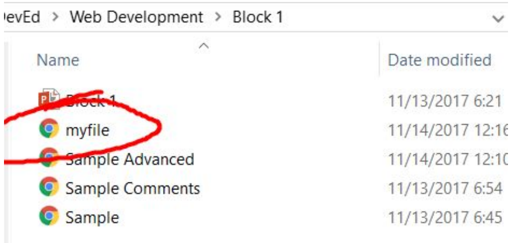
Text Documents (*.txt)

[✓ All Files]



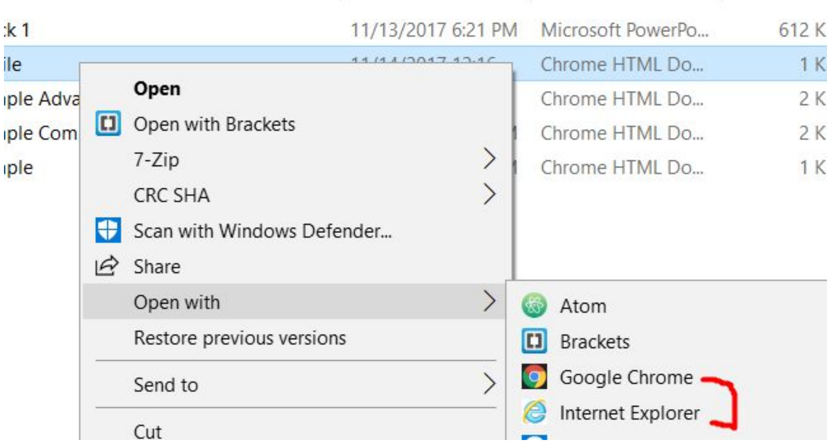
File name: myfile.html

ve as type: All Files



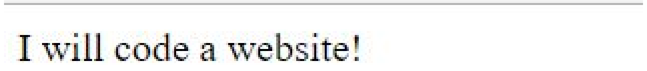
Name	Date modified
Block 1	11/13/2017 6:21
myfile	11/14/2017 12:16
Sample Advanced	11/14/2017 12:10
Sample Comments	11/13/2017 6:54
Sample	11/13/2017 6:45

If it isn't immediately set to open on the browser, right click > Open with > select a browser!



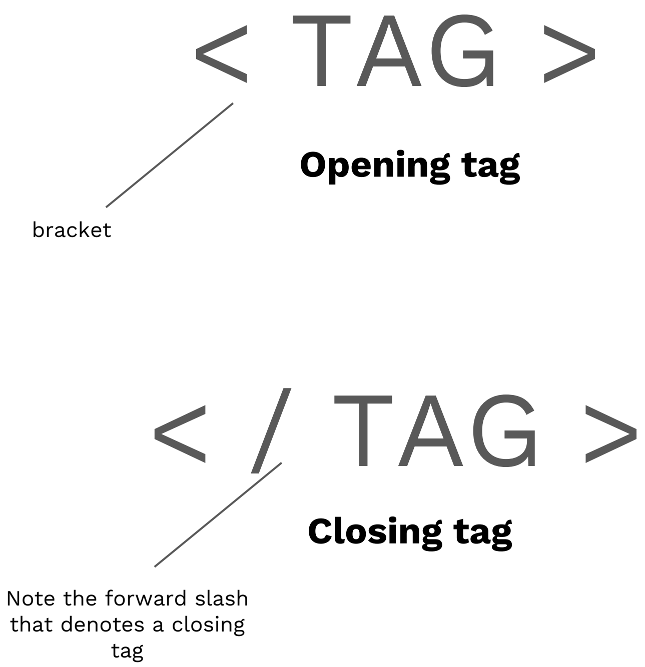
k 1	11/13/2017 6:21 PM	Microsoft PowerPo...	612 K
ile		Chrome HTML Do...	1 K
ple Adva		Chrome HTML Do...	2 K
ple Com		Chrome HTML Do...	2 K
ple		Chrome HTML Do...	1 K

This should open on the browser. Notice how the address bar isn't a normal website, but rather the link to where the file is hosted on your computer.



I will code a website!

Structure of an HTML tag



< TAG >

Opening tag

bracket

< / TAG >

Closing tag

Note the forward slash that denotes a closing tag

<strong>

I am bold

</strong>

<span style=”color:red;”>

I am red

</span>

<strong>aaa <em>ooo</em>

What happens if this tag isn’t closed?

Start by showing common formatting styles, such as bold and underline.

In other apps, you can highlight them and select “bold.” It’s as simple as that! In HTML, you select these words to bold by wrapping your text around with things called tags.

<b> I am bold. </b>

You open the tag, and close them by using a slash.

Show other common tags such as underline <u>, strikethrough <s>, and italicized <i>.

<strong>

I am bold

</strong>

<span style=”color:red;”>

I am red

</span>

<strong>aaa <em>ooo</em>

What happens if this tag isn’t closed?

The span tag is less straightforward, but a lot more versatile. It’s arbitrary alone -- it provides no change by itself. Just placing <span> and </span> around a word won’t do anything to it.

Instead, span is used for “grouping” words together to couple it for potential to be styled.

There’s no <red> tag, so placing the text “I am red” around a span lets us define changes around those words. In this case, we use another tag “style” (elaborated on later) to give this group of words the color red.

<strong>

I am bold

</strong>

<span style=”color:red;”>

I am red

</span>

<strong>aaa <em>ooo</em>

What happens if this tag isn’t closed?

If the tag in the box isn’t closed, then the text “ooo” has the effect from strong, too.

One might think that tags cut off automatically, but they can be chained together. It’s important to close tags that you open, otherwise the future items in your webpage will have unwanted effects!

```
<strong>

I am bold

</strong>
```

```
<span
style=”color:red;”>

I am red

</span>
```

<strong>aaa
<em>ooo</em>

What happens if this tag isn’t closed?

Webpage Structure

Tags to learn

1. <html>
2. <head> <body>
3. <title>

```
<html>

    <head>

    </head>

    <body>

Insert text here. :) Hi!

    </body>

</html>
```

1. Why do we indent?
2. What does this page structure mean?

Notice how code isn’t in a straight line? There are tabs (or spaces) before the tags in between the <html> tag, which is the “parent” tag and the most outer wrapper.

Code indentation is a habit that *must* be learned.

There are some programming languages where indentation is actually required, as it tells the interpreter of the code what should run and when.

```
<html>

    <head>

    </head>

    <body>

Insert text here. :) Hi!

    </body>

</html>
```

1. Why do we indent?
2. What does this page structure mean?

Indenting code keeps things readable and understandable.

The more lengthy or complex your code gets, the more unreadable it will be when placed on a straight line. Defining areas and sections by indenting new major tags helps other programmers and yourself know where and how everything is.

```
<html>
      <head>
      </head>
      <body>
Insert text here. :) Hi!
      </body>
</html>
```

1. Why do we indent?
2. What does this page structure mean?

Depending on the individual, people might not indent the <head> and <body> tags that comprise a webpage because they know it will always be there.

Further, though, when beginning to work with lots of different text styles, divs, links... it’s best to start indenting!

```
<html>

        <head>

        </head>

        <body>

Insert text here. :) Hi!

        </body>

</html>
```

1. Why do we indent?
2. What does this page structure mean?

This page structure is the **basic structure of an HTML webpage**.

You start with of course, the essential <html> and end with its closing tag.

What does it do? This informs your computer that this is an HTML document... an HTML webpage in the making!

```
<html>
    <head>
    </head>
    <body>
Insert text here. :) Hi!
    </body>
</html>
```

1. Why do we indent?
2. What does this page structure mean?

<head> comes right after html and before the body. It doesn’t have the content, but it does have things that style, organize, and classify the webpage before it runs.

For example, see the title of the webpage up on top of your screen? The one on the tab? That’s defined by a <title> tag placed in the head!

The content in <head> “runs” the webpage.

```
<html>
    <head>
    </head>
    <body>
Insert text here. :) Hi!
    </body>
</html>
```

1. Why do we indent?
2. What does this page structure mean?

<body> comes after head. (Don’t forget to close the head! A head merged with a body isn’t very… enticing, isn’t it?)

This is where the content goes! Your words and other regular elements go in this section. It contains almost all of the content that will be on the webpage. Anything not a tag (aka regular text) is that! Regular text.

```
<html>
    <head>
    </head>
    <body>
Insert text here. :) Hi!
    </body>
</html>
```

Input a <title> into your head and then place any text inside the <body>.

Open the .html file in your browser and view the result.

This should modify the text seen in the top of the screen, or the browser tab.

Playing with Basic HTML Tags

Tags to learn

1. Paragraphs and Headers

   <h1> <h2> <h3>…

   <p>

   (It goes down all the way to h6)

2. Formatting

   Italicized <i> Underline <u>

   Bold/Strong <strong> <b>

Don't forget to close your tags!

We experimented with formatting in the beginning.

Playing with Basic HTML Tags

Some more tag manipulation

1. Line Breaks

   <br> -- Try using this in the middle of your paragraphs!

2. Images

   <img src=” “ > -- What does this do?

3. Links

   <a href=”link”>text here</a> -- What’s href? How does this format work?

<br> starts a new line in your text. Writing stuff in a paragraph but want some space? Writing a poem? Use it!

<img> There’s no need to close this tag since there’s no reason to. It’s not “surrounding” anything, and can exist by itself. Place a link to a direct img in src=””. Refresh your page. Hi image! The tag grabs an image from that link placed in the source. (Make sure the link ends with an image file extension, if not, you may accidentally be grabbing a webpage!)

Playing with Basic HTML Tags

Some more tag manipulation

1. Line Breaks

   <br> -- Try using this in the middle of your paragraphs!

2. Images

   <img src=” “ > -- What does this do?

3. Links

   <a href=”link”>text here</a> -- What’s href? How does this format work?

If <img> has a source, the href in link is the link’s destination webpage. When you click on the link, which is what the words surrounded by the link tag are, you get brought to the link at href.

<a> defines a hyperlink. In the opening tag, the destination link is present. It surrounds the “content” that becomes the link.

Also! Notice how the colors of links change depending on how we interact with them. Purple means visited, etc.

Try to make a webpage with the things you’ve learned so far!

Don’t forget your page structure:	<html> <head> <body>
Headers and paragraphs:	<h1> <p>
Formatting:	 <b> <u> <i>
Images:	<img src=”link”>
Links:	<a href=”link”>text<a>

Styling

Let’s add some CSS!

CSS

Cascading **S**tyle **S**heets

Dictates how these basic HTML elements should look like on the webpage, allowing us to essentially specify their designs

<p style=”color:red;”>Your text here</a>

1. What did adding *style* to the paragraph tag do?
2. What’s the composition of the items in the style tags?

COLOR: RED;
PROPERTY: VALUE

1. The style tag added… some style! It let us specify “color” and set it to “red.”
2. The property here is “color,” we’re modifying the color. The value wet set it to it is “red,” which is why the color of the text it wrapped is, well, red!

You can check out more properties and their possible values here: https://www.w3schools.com/cssref/

<p style=”color:red;”>Your text here</a>

1. What did adding *style* to the paragraph tag do?
2. What’s the composition of the items in the style tags?

COLOR: RED;

PROPERTY: VALUE

Other style tag elements to consider playing around with

- Text-align - where will you place your text?
- Background-color - If there’s color, there’s a background, too.
- Font-size - Go big! Go small!
- Letter-spacing - Kind of weird, but yeah...
- Text-indent - If you’re writing a long paragraph, this makes everything a bit more tolerable.

<p style=”color:red;”>Your text here</a>

Aside from color:red, there are different properties you can try experimenting with to manipulate color: **background-color**

You can also try changing the size of the text: **font-size**

Or you can try placing a border around it: **border**
(Border is a bit more complicated! Different CSS properties take in different parameters. Border can take the stroke, color, and width of the border.)

<body style=”background-color:blue;”>

What if you try adding some style to the body tag?
What happens when you manipulate it this way?

<span style=”color:blue;”>text text</span>

What is the <span> tag?
How does it group inline elements?
Content inside a span tag can be designed and manipulated with CSS.

1. The property of background-color is given the value “blue” for the entire webpage! The entire HTML document should have a blue background.
2. As shown much earlier, the span tag alone doesn’t have any function alone. However, it can be used to group together content to apply modifiers to them, such as bringing text text together to give them some CSS styling.

<style>

1. Begin creating your stylesheet in the <head>

```
<head>
<style>
body{
    color:red;
}
</style>
```

**<style type=”text/css”>
Declare it as text/css**

Make sure this is being written in between the head tags. We’re not writing out content, we’re specifying the design of the materials in the webpage!

There are names for each of these parts.

Here, body is the **selector**. It *selects* what part of the document we’re manipulating.

The **property** color is being given the **value** of red. We use a colon to distinguish between the two, and when done, place a semicolon at the end of the property-value pair. You can add more below them.

The curly brackets are used to surround the area or block that contains the property-values of the selected element.

Classes

What if we want a certain portions of the body to have text with a **red background**?

Do we have to style it over and over again, specifying style=”background-color:red;” each time?

1. Declare the class in the stylesheet (see: .redstuff)
   The . blahblah makes it a class
2. Instead of repeating style=”background-color:red;” you can simply apply class=”redstuff” to the paragraphs where you want the text to be red!

```
<html>
<head>
        <style type=”text/css”>
                .redstuff{
                        background-color:red;
                }
        </style>
</head>
<body>
<p>This is regular text.</p>
<p>This is red text!</p>
<p>This is regular text.</p>
<p>This is red text!</p>
</body>
</html>
```

End of Block 1