Activity 1: Setting up a Simple Web Page (HTML)

Introduction

People use web pages to share information across the internet. You (along with an estimated 4.57 billion of your fellow human beings) have likely browsed the web to gather information, such as to conduct research for an essay or fact-check your science homework. The best web pages are easy to understand and navigate. But have you ever thought about what it takes to build a web page?

A typical web page uses a combination of three programming tools to create its look and feel.

- HTML sets up the landscape of a web page by positioning elements in different locations and creating the content of the page.
- CSS modifies the presentation of the content by controlling colors, fonts, and general formatting. This is essentially the style guide for a website.
- JavaScript enables the animation of web pages and creates the opportunity for user interaction.

Throughout this set of activities you will explore all three of these languages, one or two at a time. In this first activity you will explore how HTML is used to set the content of a web page.

Preparation

This activity, and the following activities, requires each of the following:

- Computing device with internet access
- Account on glitch.com (free to make and use)
- Vernier Go Direct Weather Sensor
- Other resources provided with this set of activities

Procedure

Understanding the HTML Document

1. Open Glitch.com, login, and create a new script. Choose the “hello-webpage” option in order to begin creating a website.
2. On the left side of your screen, locate the following menu:

a. Use the README.md section to provide an explanation of your program. If you want to edit this text, you need to click the Markdown button.
b. Use the index.html to create the basic objects or “elements” of your web page. This code either includes animation and style code, or links to other code in the following areas:
    - Script.js is a convenient location for your JavaScript code to animate your web page.
    - Style.css contains code to set the “look” for elements on your web page, much like a style guide.
    - Click through each of these elements and briefly review their contents.
    - Click on the index.html file to expose the code.

3. Click Show > In a New Window to execute the code and view the web page.

Figure 1 - The web page opens in a new window.

4. After you run the program, look closely at the elements on the web page.

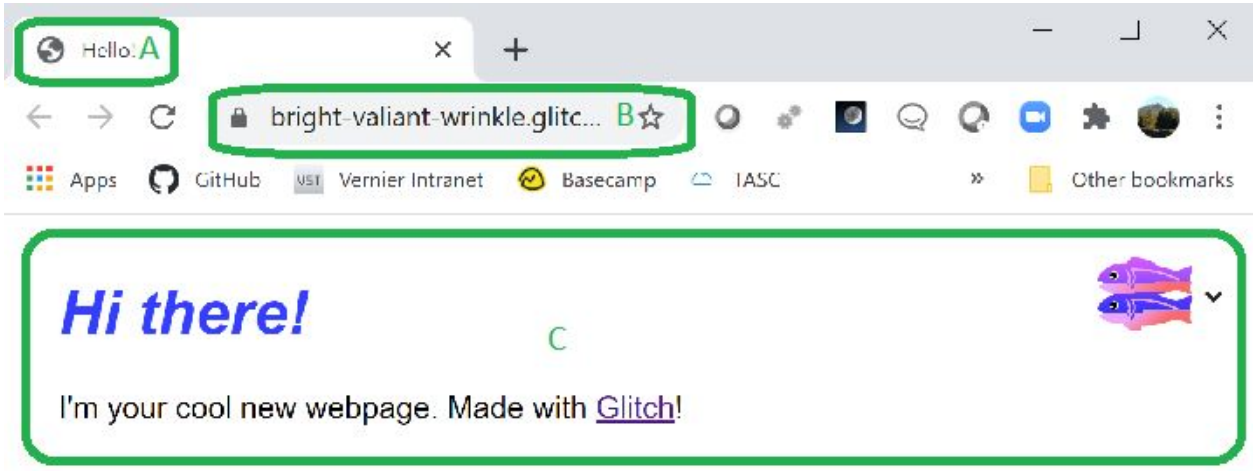

Figure 2. The anatomy of a web page

When looking at a web page, it is easy to overlook some of the design details. Pay close attention to the three areas highlighted in Figure 2. Can you identify the code that creates each of these elements?

5. Use the table below to understand how the code functions.

HTML and HTML Tags

HTML displays "elements" that are defined by start and end tags (< … > and </ … >). An element is used to refer to any text, images, etc., that are displayed on a web page. The tags define how those elements are displayed.

> For example, the <h1> *element* </h1> tags specify that the element between the tags should display as a heading (h1). The tag *element* would specify a paragraph format.

Here are a few of the elements that are used to get started. Take a moment to read through this list.

`<!DOCTYPE HTML>`	This is a declaration necessary for each HTML document and is found at the very beginning of the code.

`<html lang=en>`
`....`
`</html>`
	Everything in an html document needs to be written between the html tags. In the example to the left, "lang=en" is defined in this tag. This specifies the language for the

	html code. In this case it's English.

`<head>`
`...`
`</head>`
	Meta information about the HTML page is held between the head tags. This is where you can import libraries to be used in your code.
	
	Metadata is used by browsers (how to display content or reload page), search engines (keywords), and other web services.

`<title>...</title>`	The effect of a title depends on where it is used. When it is placed in the head tags it is used to specify the name of the web page that appears in the browser. In our code, this is "Hello!" Refer to A in Figure 2. When the title tag is within the body, the text appears on the web page itself. Find "Hi there!" in the code and on the web page.

`<link rel="stylesheet" href="/style.css" />`
 	
`<script src="/script.js" defer></script>`
	These tags reference and import the code found in the style.css and script.js tabs.
	
	CSS and JS can both be coded directly into an html document. However, it is often easier to navigate between the different languages if they are in separate linked sections.

`<body>...</body>`	This is where all the elements (headings, paragraphs, images, tables, etc.) of an HTML page are defined and contained.
`<h1>...</h1>`	These tags define an element as a heading.
`<p>...</p>`	These tags define an element as a paragraph.
`<!-- content -->`	This declares a comment. Comments are used to document the function of a line or section of code. Everything contained within <!-- ...--> is ignored when the code is executed.

 	This tag inserts a line break in the web page.

6. On your HTML document, try making the following edits:
    a. Edit the title of your web page to an appropriate name. For example, Figure 3 is titled “Vernier Weather Dashboard”. You can watch the change happen dynamically as you type.
    b. Edit the <h1> element to display a descriptive name. The example below displays, “Go Direct Weather DashBoard”.
    c. Change the <p> element and tags so that your name and class appear. Refresh your web page to view the changes, if necessary.
7. Click on the title of the program in Glitch and rename it so that you will recognize the purpose of the code. For example, ours is named wx-station-act-1.

Congratulations! You now have a basic understanding of HTML structure.

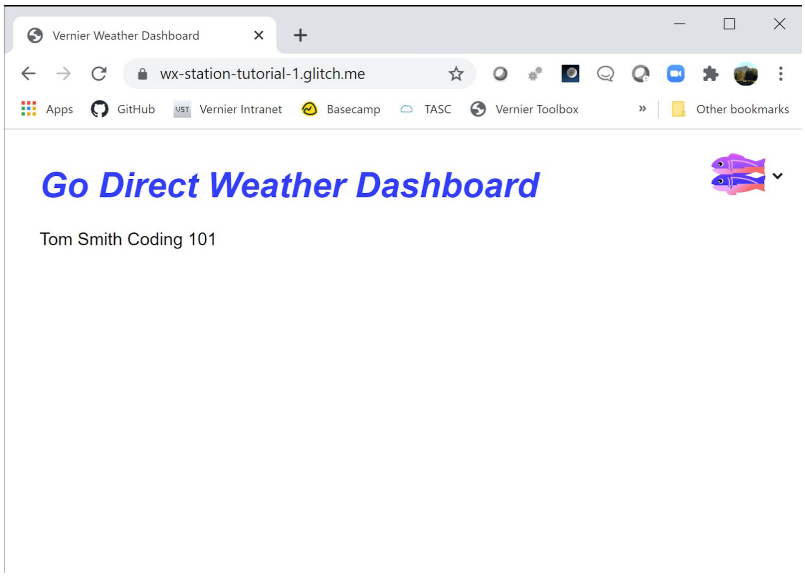

Figure 3 - example of a completed Activity 1