Week 01 - Setup: Tools

Overview

The purpose of this setup activity is to install Microsoft's **Visual Studio Code** (VS Code). It is free to install on Mac, Windows, and Linux operating systems. This is the software application that will be used to write and edit code.

Download and Install VS Code

1. In your browser, navigate to the Visual Studio Code download page.
2. You will be prompted to download VS Code for your operating system.
    - Click **download** and follow the directions to download and install VS Code.
    - The default installation settings should work well for this course.
3. Open VS Code to verify that it has been correctly installed on your computer.

Install the Live Server Extension

Normally when you visit a website in your browser it is being hosted by a server on the internet. In this class, you will create the web pages first on your own computer and then later publish it to a hosting server.

In order to test your web pages while you are working on them, you need to have a server program running on your computer that will host the pages for you. The program you will use in this course to host your pages is called, "Live Server (Five Server)" and it is installed as an extension to VS Code.

1. In your browser, navigate to the Five Server extension in Extension Marketplace.
2. Select **Install** and follow the prompts carefully to install the extension within VS Code.
3. If you would like, you can always see and manage the extensions that are currently enabled in VS Code by using the Extensions View. The Extensions View icon is found in the Activity Bar
4. Any file with an `.html` extension can be opened using Live Server (Five Server) from within VS Code. To view the page, right-click the html file and select "Open with Live/Five Server". This will launch your default browser show the page as if it were hosted on a web server.

After right mouse clicking on an HTML file in the VS Code explorer, this menu option appears.

Next - Create Your GitHub Repository

Overview

The purpose of this setup activity is to create a free **GitHub** account and set up the repository you will use throughout the course to store the web pages you will create.

GitHub is a popular and free online collaboration and version control service. You can think of GitHub repositories as a cloud-based storage system for your projects. GitHub can also be used as a web hosting service with a special web service named **GitHub Pages**.

Sign Up for a free GitHub Account

1. In your browser, visit the github.com site.
2. Select the **Sign Up** button.
3. Enter an email address and select **Continue**.
4. Create a **password** and select **Continue**.
5. Enter a **username** using only lowercase letters. Your username will be visible to others, so make sure to choose something professional. If you get a red '✕' to the left of your username, then someone else has already used that username and you need to choose a different one.

   Choose a professional name because you will use your GitHub account page in the future as a portfolio of your work.

6. Record your username and password in a secure location.
7. Complete the remaining steps of the sign up process which may ask if you want GitHub to send you emails and may have you verify that you are not a robot.
8. You can skip any personalization questions by selecting the Skip Personalization link near the bottom.

Create Your Course Repository

GitHub organizes work into a different repository (often abbreviated "repo") for each major project that you work on. In this step, you will create the repository that you will use for the entire course. Rather than creating a blank repository, you will start with a template that contains many files you will need for this course.

1. In your browser, visit the github.com site.
2. Make sure you are signed in.
3. Navigate to the starting template repository for this course.
4. Select the green button "Use this template" and select the option to "Create a new repository."
5. Do not check the box to include all branches.
6. Ensure that your username shows up as the owner, and type `wdd130` as the repository name.
7. Mark your repository as **public**.
8. Select the button to **Create Repository**.

Enable GitHub Pages

Now that you have a repository with the starting template files in it, you can set it up to host webpages by enabling GitHub pages.

1. Navigate to your course repository at GitHub.
    - The location, or URL, of your repository is: `https://github.com/your-username/wdd130` , but make sure to replace `your-username` with your actual username.
2. Select the **Settings** tab for the repository.
3. On the left hand menu, select **Pages**.
4. Configure the pages to build from the root of the main branch by doing the following:
    - Leave the Source as: **Deploy from a branch**
    - In the Branch section, select **main**.
    - After selecting **main**, leave the location as "`/ (root)`", which is the default.
    - Click the **Save** button
5. After clicking save, GitHub will begin building your website. It may take a few minutes for it to complete. When it has completed, you will be able to visit your new website in a browser at the location, or URL: `https://your-username.github.io/edutech100/` , but make sure to replace `your-username` with your actual username.
6. Verify that your page loads in a browser. It should look something like the following:

W01 Setup: Install Git and Clone Your Repository

Overview

You must complete the steps to install VS Code and set up your GitHub repository before starting this activity.

If you have not completed those activities successfully, you should stop and complete them before starting this activity.

Now that you have created your course repository at GitHub, you need to download a copy of that repository to your computer so that you can work on your files on your computer and then upload them to GitHub when you are ready to publish and submit them.

Git is a software product that manages the files you use in programs and webpages. It is very popular in industry and helps keep track of different versions of files and helps team members share code with each other. Using Git, you can **clone**, or download a repository to your computer. You can also **push**, or upload your code to GitHub when you are done.

This setup activity will help you install Git on your computer and use it to clone the repository you created at GitHub.

Instructions

Install Git on your computer

Select only **one** of the following instruction sets depending on your operating system.

Windows Users

1. Before downloading **Git**, check to make sure you don't already have Git installed on your computer.
    1. Open a **command prompt** by clicking the Start button and typing `cmd` in the search box. The command prompt terminal will open.

        Alternatively, in VS Code, you can use the terminal panel to complete any command line task. use the **`Ctrl+`** keyboard shortcut to toggle the terminal panel.

    2. Enter the following command and then press **Enter**:

        ```
        git --version
        ```

    3. If you see a version number, you already have Git installed. If you don't see a version number, you need to install Git.
2. To install Git for Windows. Go to git-scm.com/downloads and select **Download for Windows**.
   An .exe file will be downloaded.
3. Double click that .exe file to open and run the installation executable. The process of installing Git will start.
    1. Allow the installer to make changes to your computer.
    2. Select **Next** through all of the setup windows, leaving all the defaults. There will be many windows.
    3. The last window will let you select **Install** and then click **Finish**.
4. Open a **new** Command Prompt window by closing the first Command Prompt window and start a instance by clicking the Start button and typing `cmd` in the search box.
5. Enter the `git --version` command and then press **Enter** to see if Git is installed. You should see the version number.
6. While in the Command Prompt, type in two more commands to set up the username and email that are associated with your GitHub account.

    Make sure you to use your own username and email between the "" quotation marks. Use the username and email you used for the GitHub account. These will be different for each student.

    1. Type in this configuration command to set the global git username:

```
git config --global user.name "yourusername"
```

and select **Enter**. Nothing will happen if you did it right. If you get an error, you need to fix it.

2. Enter this configuration command to set the email:

```
git config --global user.email "youremail@byupathway.edu"
```

Nothing will happen if you did it right. If you get an error, you need to fix it.

3. Type in this configuration command in order to not ignore case changes in files and folder names:

```
git config --global core.ignorecase false
```

Nothing will happen if you did it right. If you get an error, you need to fix it.