W00 🏃🏽‍♂️ Activity: Enabling Git in VS Code

**To-Do Date: Sep 15 at 11:59pm**

The purpose of this activity is to **install git** and to use the built-in git support features within VS Code to connect with your remote wdd230 repository on your GitHub account. As you make updates to your primary, local work, you will then be able to push/publish/upload your additions, deletions, and changes to your remote, public repository.

Instructions

In order to publish your web work to your remote repository, you will need to install GIT services.  Git commands can be used directly within VS Code. Our goal is to establish a local repository folder which will be linked to your remote GitHub pages repository. Changes that you make to this local repository will be tracked and can be uploaded to the remote copy.

In this activity you will download git, run a couple of one time git configuration commands, clone your remote GitHub Pages repository to a local folder on your computer, and test that you can publish.

Step One - Install Git

1. **Navigate** to [https://git-scm.com/downloadsLinks to an external site.](https://git-scm.com/downloads) in your browser and download the appropriate version for your platform.
2. **Install** git on your computer using the default options.  
   *Note that Mac users will need to use a third-party installer for git.*

Step Two - Configure Git

🎦 Step Two Video: [Demonstration using VS Code Terminal Window to set global values for git.Links to an external site.](https://video.byui.edu/media/t/1_psnt10l6)

1. **Open** up a terminal interface to the command line in Visual Studio Code editor or with the command window (Windows) or with Terminal application (Mac).

**Sample Terminal Options**

**Visual Studio Code**: Press the **ctrl** +**~** keys to open up a terminal window.

**Windows OS**: Press the **WinKey** + **r** keys on your keyboard and then type *cmd* in the dialog box and then click OK.

**Mac Terminal**: The *Terminal* application is found within the *utilities* folder in *Applications*.

1. In the terminal on a new command line, run git configuration commands to set the *user.name* and *user.email*. Type out the following commands in the command line exactly as shown except use your github username and your email as indicated.  
     
   You will need to **press enter** after each of these command lines in order to run the command. No error message means that you were successful.

git config --global user.name "Use your GitHub username"

git config --global user.email "Enter your email address"

**Explanation**  
  **git** : Since you installed git on your local system, git allows you access to the git library of commands.  
  **config** : This is a command function that sets configuration variables.  
  **--global** : This configuration argument applies the setting that follows to the current operating system user.   
  **user.name and user.email** : These are configuration properties that can be given values.  
 **"*username*" :** The value to be stored in the property. In these cases, string data type values which are contained within double quotes.

1. **Close** the terminal application.

Step Three - Clone GitHub Repository

🎦 Steps Three and Four Video: [DemonstrationLinks to an external site.](https://video.byui.edu/media/t/1_c6i8j2zl" \t "_blank)

1. Go to your GitHub **wdd230**named repository on github.com and **copy** the given URL.  Figure: Screenshot of location of copy button.

**!** Do not use the **Set up in Desktop** application.

💡 **If** you do not see this screen shot on your repo's home screen, use the green **Code**button to obtain the URL.

💡 The format of this URL address that you are to clone from GitHub is similar to this example:

https://github.com/*username*/wdd230.git

If you are having a tough time finding this URL to clone, then just copy this URL example and substitute in your own GitHub username.

1. **Launch** or relaunch Visual Studio Code (VS Code).
2. In VS Code, open up the **Command Palette** from the *View* menu.
3. Type ***git clone*** in the command line provided by the Command Palette.

💡 If this command does not work, then you may need to restart VS Code or revisit Step One in these instructions and make sure git was installed.

1. **Paste** the copied URL from the previous step into the text input area provided and press *Enter* to confirm.
2. A dialog box will appear to allow you to **Select** a folder or create a new folder on your computer that will house the local repository and that will be linked to your GitHub Pages repository. This will be your working folder for the class web assignments and all changes will be tracked for publication to your remote GitHub Pages.

💡  Note that **a new folder will be created** in your selected folder location with the default name of the GitHub Pages repository ending in .github.io.  Hidden git confirmation files are added to this new folder so be careful not to manually move it on your system.  
  
💡  You can repeat these steps to establish a different folder as your working folder if you need to restart.

1. A notification dialog box will popup, typically in the lower right portion of the screen, to allow you to **Open** the repository. Click Open. This will direct your VS Code Explorer to open that folder location. It will be empty.

Step Four - Create Test File and Upload to GitHub

* 1. **Test** your configuration by creating a new HTML file named*index.html* in that folder, your local repository. Make it simple with simple content for testing.

💡 You can create a new file by using File > New File or by right mouse clicking in blank explorer space to open up a menu create a new file or use the small icon for creating a new file by your directory name in the explorer window by the name of your new folder.   
 🎦 Video: [Demonstration of Creating a New HTML File in VS CodeLinks to an external site.](https://video.byui.edu/media/t/0_txuodie2) (visual only)

* 1. **Save** the index.html file and a number will appear next to the source control icon indicating changes have been made to the repository that need to be committed and pushed to the remote repository for publication. You will need to:
     + **Write** a comment description for the commit. There are no requirements except that it cannot be empty.
     + **Stage**and **Commit**the changes to your local repo
     + **Push**to the committed files to your remote GitHub repo site.

🎦 Video: [Demonstration of Publish Process from VS CodeLinks to an external site.](https://video.byui.edu/media/VS+Code+Commit+and+Push+Process/0_a4x8gjts) (visual only)

* 1. Verify the page exists on your GitHub repo by **navigating** in your browser to your GitHub Repo site. e.g., *https://github.com/susanbsmith/wdd230*

💡 Please note that the process may take a bit for the files to be uploaded, mapped, and actually being made available on the web server.