Setting up Git on Windows PC

**Purpose:**

This will only cover the initial setup and configuration of your git environment, and get your machine configured to make SSH pull requests to remote repositories (github,gitlab, and hosted servers). It will not cover basic commands and actual utilization of the tools. Note there are extensions that exist that make use of custom UI’s to navigate commands these will be left to the user. These documents will be using windows powershell exclusively.

**Process For Git Install/Configure:**

1. Download the correct version for your machine from the following website <https://git-scm.com/download/win>.
   1. If you are unsure of your version of windows you are running in the search bar in the bottom left corner, type system information and hit enter your version will be under system type.

A screenshot of a computer

Description automatically generated

1. While installing keep all the options as default. We will be manually configuring the setting we want.
2. Next open windows PowerShell or command prompt (I suggest PowerShell but if for some reason your system doesn’t have it command prompt works too).
   1. PowerShell: From the search bar at the bottom left, type PowerShell and hit enter.
   2. Command Prompt: From the search bar type cmd and hit enter.

**TIP:** When in a file explorer if you go to the file path bar and type PowerShell or cmd a console gets made and has its path already set.

1. Once you have git installed you can open windows power shell or cmd and run the following to verify your installation worked correctly.

git -v

If everything was set up correctly you will see the following.

A blue screen with white text

Description automatically generated with low confidence

1. We can now do some configuration setup for our profile. It is suggested that this is done before trying to do any commits so we can properly track changes. I would run the following commands to attach an email and user name to your machine.

git config --global user.name “Your name”

git config –global user.email youremail



**NOTE:** I have my personal email address attached this is because I use this PC for more then school work, I will show later how to change your email for individual projects(repositories).

Also I would suggest running the following command to change the default text editor that git will try to use. I am going to suggest using [Visual Studio Code](https://code.visualstudio.com/download) the below commands should work if you want to use something else refer to that editor’s documentation, the default code editor is VIM and that is difficult to use. This editor will only really popup when we want to add a message to our code so you will not be interacting with it much.

git config --global core editor “code –wait”

**Setting up SSH**

1. To make pulling from the repository easy we are going to set up SSH keys. Each computer will have a key created that will get attached to your github/gitlab account.
2. To generate the key from the PowerShell console run the following command.

ssh-keygen -t ed25519 -C “youremail”

A blue screen with white text

Description automatically generated with low confidence

When it says enter file in which to save the key just use the default location. **NOTE the .ssh folder will be hidden.**

1. You will then be prompted to enter a password and then re-enter it. You will know it is successful when you get the following.

A screenshot of a computer program

Description automatically generated with low confidence

1. At this point we are set up to go, I will show you an example of how to move your public key onto github so your accounts are linked. **NOTE:** This may not be the route we go with this project, but the process should be about the same no matter what we use.
2. Go to github.com and go to your settings on the left hand side there is an option called SSH and GPG keys.
3. At the top of the page there will be a new SSH key button, click it.
4. For a title set it as whatever you like, remember you can have multiple keys at once so I typically name it after the machine I am on.
5. For the key itself navigate to the folder you saved your keys under. If left default it will be located in.

C:\Users\name\.ssh

The file you want is the one with the .pub extension. **NOTE** Do not share the other file that is your private key.

1. Right click the .pub file and open with notepad and copy the entire message.
2. Paste that onto github and you are setup.