# Brief Windows Git(hub) Tutorial

## Install Git

I’d recommend downloading and installing git+BASH from the following link

<https://gitforwindows.org/>

This will install all command line utilities (via git BASH) and a GUI for interacting with git. I’d recommend getting familiar with the command line version. After installing, this can be accessed by typing `git` or `bash` into the windows search bar and then clicking the git BASH program that should appear. This will open up a command line.

## Set git username and email

You’ll want to keep this consistent with your github account information. If you don’t have a github account, just head to github.com and create one. Should be free. Creating the account with your gatech email will provide access to premium features like github copilot.

Open up a git BASH terminal (as described above) and enter the following commands to set your user name and email



Replacing “Mona Lisa” with your user name and



Filling in “YOUR\_EMAIL” with your github linked email.

NOTE: The `$` just indicates the start of a line and should not be included in any commands.

## Clone Group Repo

From the bash terminal, move to the folder where you’d like the group repo to live. On windows, the terminal should open inside your home folder, so this should be something simple like:

cd Documents

Followed by:

git clone <https://github.com/ltimmerman3/ProcSys_Project.git>

Voila, the repo should be on your computer

## Make changes, then commit to a new branch (after sending Lucas your github email so you have access 😊)

Make any changes you like to the document in the folder, or create your own. Once you’re satisfied with your changes, run the following commands to set up custom branch, then commit and push changes. The following commands should get the job done. NOTE: You will not be able to do this untile I have added your github email to the list of known collaborators.

git checkout –b custom\_branch\_name

git add –A

git commit –m “My first commit message”

git push -u origin custom\_branch\_name

These commands, sequentially

* Create and swtich to a new branch called “custom\_branch\_name”
* Add all files/changes so that git recognizes and begins tracking them
* Creates a commit on the new branch with the associated message
* Pushes all changes and establishes an upstream tracking branch – changes will now be visible on the github page under the custom\_branch\_name branch

## Check for changes and pull to local machine

If someone has made changes to a branch and you’d like to have them in your local repo, just run

git pull

This will pull all of the from the most recent commit to your local repo.