Git

1 What is Git?

Git is a Version Control System (VCS). Git allows you to save the state of your project, tracking changes to the previous snapshot (*commit*). With Git, we initialize a working directory, commit changes, and then push these changes to our repository.

Using Git is a requirement for your CST 205 final project.

2 Install Git

- To install, refer to Getting Started - Installing Git.
  - For Windows, you can use PowerShell (instead of Git Bash). The Git Windows installer allows further customization of PowerShell to *enhance* your Git experience: more info

3 Set up Git

Issue the following commands at the command line prompt (use the email address you have already used on GitHub or the one which you intend to use on GitHub):

```
git config --global user.name "Your Name"
git config --global user.email your_email@csumb.edu
```

4 Start tracking

- It is easiest to start with a new, empty directory
  - e.g., from the command line navigate to the location for your new directory and then execute `mkdir 205git`
- It is helpful to inform Git from the get-go of certain files to ignore. We can use a special file called `.gitignore` for this.
  - Create a new file called .gitignore and add the following to it:

    ```
    *.pyc
    .DS_Store
    ```
- Make sure you are in the new directory (e.g., `cd 205git`) and then execute `git init`.
  - If successful, you will get a message back saying something like, “Initialized empty Git repository...”.

5 Create a repository on GitHub

- Follow these steps: Create A Repo
  - Make the repository public and do not yet add a `README`.
  - You are next given information on how to access the newly-created repository.

6 Back to your local machine

- Add a file to the directory you just created, e.g., **git_test.py**.
- Add some code to it, e.g., print a “hello git”.
- Make sure you are in the correct directory and execute the following from the command line: `git add -A`
  - This stages ALL of your files.
- Execute `git status`
  - Your new file will be listed here.
- Execute `git commit -m` with a brief message about your changes, e.g., `git commit -m "First commit"`.
  - If successful, you will see a message about files changed and insertions.
- Next we want to push our changes to our GitHub repository. We need to provide where the repository is. This information is given to us by GitHub after the repository is created:

  ```
  git remote add origin https://github.com/avner-csumb/tarantula-equinox.git
  ```

  - Your command will be different depending on the name you gave your GitHub repository and your GitHub username.
  - Lastly, we push our files over:

    ```
    git push -u origin master
    ```

    - You might be asked to authenticate.
    - View your repository on the GitHub website to see your code.

7 Changes

- Let’s add a comment to your code.
  - In Python, single-line comments are preceded by a `#`.
- `git status` now tells us that we have some changes not yet staged for commit.
- As before, we need to first add the file. There are different ways to do this, but we again do:

  ```
  git add -A
  ```

- Then, we commit with:
  - `git commit -m "Added a comment"`

- Lastly, push to master:

  ```
  git push -u origin master
  ```

- If we view our code on GitHub, we see our changes.