Brief Guide to GIT on GITHUB

(Mostly from the command line)

References:

https://github.com/
https://guides.github.com/activities/hello-world/
https://en.wikipedia.org/wiki/Git
https://en.wikipedia.org/wiki/GitHub

Introduction:

GIT is a version control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows.

GitHub (originally known as **Logical Awesome LLC**) is a web-based hosting service for version control using git. It is mostly used for computer code. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

Getting Started:

Create an Account:

Go to: https://github.com/ and create a new account

Create a Repository:

The easiest way to create a new repository is to go to your GITHUB account and then:

1. In the upper right corner, next to your avatar or identicon, click and then select **New repository**.
2. Name your repository hello-world.
3. Write a short description.
4. Select **Initialize this repository with a README**.
5. Click **Create repository**.

Cloning Your Repository:

Start by creating a clone of your repository on your local machine. This assume that GIT is installed and you have an Internet account.

1. Click the Clone or Download button
2. Copy the link
3. Then at the command line enter:

```
git clone GITHUB_URL
```

for example:

```
git clone https://github.com/taktse/An-Najah-HTC.git
```

Add Code to Your Repository:

Using the above example:

```
cd An-Najah-HTC
```

Create the file Another-README.md

```
vi Another-README.md
```

Now to "push" the file to GITHUB, use:

```
git add . --all
git commit -m "Added the Another-README.md"
git push origin master
```

Modifying An Exiting File:

Edit README.md and then push it the repository:

```
vi README.md
git add . --all
git commit -m "Modified README.md"
git push origin master
```

Some Notes:

- git pull - pulls updates from the repository
- git add . --all - The --add means to remove files from the repository that have been removed from the local clone.
- Empty directories are not stored in GTHUB
- Use .gitgnore to exclude files from the repository:
  For example:

```
pbsgold-laptop> cat .gitignore
kickstarts
*~
.\#*
\#*\#
```

- More information can be found with:

```
man git
```

- **GIT tags** - Please https://git-scm.com/book/en/v2/Git-Basics-Tagging
  Git has the ability to tag specific points in history as being important. Typically people use this functionality to mark release points (v1.0, and so on). Some commands:

  ```
  ○ git tag -a 0.0.2 -m "Tagged version 0.0.2"
  ○ git show 0.0.2
  ○ git push origin 0.0.2
  ```

- git push origin --tags
- git checkout 0.0.2