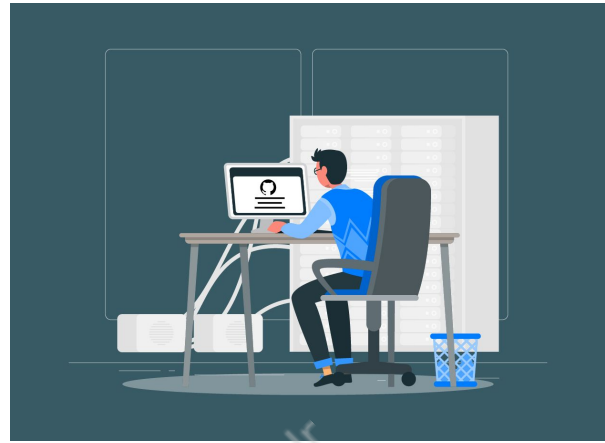


Git and Github



What is our GOAL for this MODULE?

The goal of this module is to learn to use git from the command line.

What did we ACHIEVE in the class TODAY?

- Installed command line git and git bash
- Cloned the activity repository
- Made some changes to the code and committed these changes to a remote repository

Which CONCEPTS/ CODING BLOCKS did we cover today?

- git and git bash
- Committing changes to github from command line.

How did we DO the activities?

1. We used GitHub website to host our code online. GitHub used a tool called Git to keep track of all the changes in the code.

Installing Git

For Mac:

<https://sourceforge.net/projects/git-osx-installer/files/>

For Windows:

<https://gitforwindows.org/>

- Open your git bash shell (for Windows) or a normal terminal (for Mac). Enter the password on the bash shell and it shows the output.

```
$ pwd
/home/rajeev
```

- We used commands like "ls"(to list all the files in the directory), "cd" (change to a different directory), "mkdir", etc.

```
$ cd ..
$ cd /home/rajeev/ProjectAlgorithms/
$ ls
index.html  p5.dom.min.js  p5.js  p5.play.js  p5.sound.min.js  sketch.js  style.css
```

2. We created a new directory/folder in any location using 'mkdir' commands. 'mkdir' stands for make directory.

```
$ mkdir Projects
$ cd Projects/
$ git clone https://github.com/whitehatjr/angryBirdsStage2
Cloning into 'angryBirdsStage2'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 38 (delta 10), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (38/38), done.
$ ls
angryBirdsStage2/
$ cd angryBirdsStage2/
$ ls
BaseClass.js  Box.js  index.html  matter.js  p5.min.js  p5.js  sketch.js  style.css
Bird.js  Ground.js  Log.js  p5.dom.min.js  p5.sound.min.js  README.md  sprites/
```

- The three areas in Git:
 - A working directory: Files are first modified in the working directory.
 - A staging area: Hosts all the files which are changed before they are committed.
 - A repository (local and remote): Hosts all the different committed versions of the files.

Note: Every modified file travels from the working directory to the staging area to a repository.

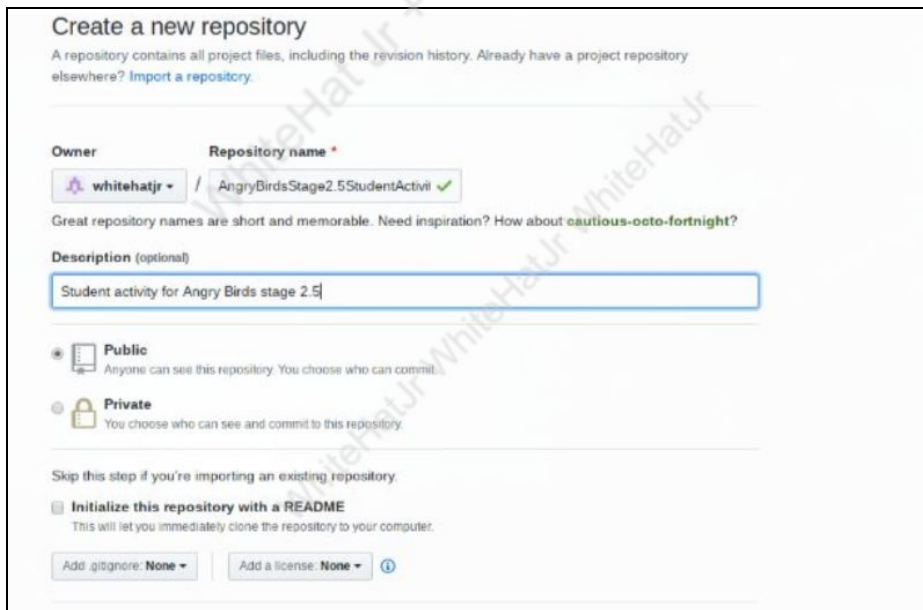
3. To make a git add and git commit the files with a message use:

- `git add <file_name>`
- `git commit -m "<message>"` (-m stands for message)



```
git add sketch.js
git commit -m "Add Platform"
[master 02d825a] Add Platform
1 file changed, 5 insertions(+), 3 deletions(-)
```


4. We created an empty remote Github Repository. DO NOT INCLUDE A README.md file.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner **Repository name ***

 whitehatjr / AngryBirdsStage2.5StudentActivi ✓

Great repository names are short and memorable. Need inspiration? How about [cautious-octo-fortnight](#)?

Description (optional)

Student activity for Angry Birds stage 2.5

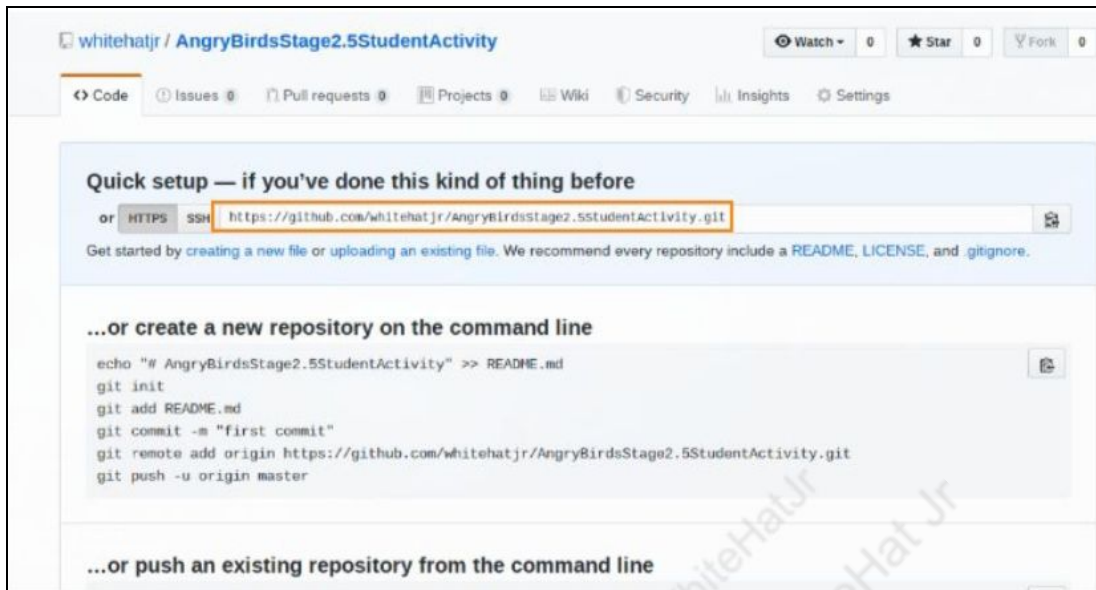
☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** Add a license: **None** ⓘ



- To associate the working directory on our git bash shell to this remote repository use: `git remote add <remote_name> <remote_github_url>`
 - We used any name for the <remote_name> and the url of the github repository you created.

```
$ git remote add test https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
```

- To push files to a remote repository use: `git push -u <remote_name>` (-u stands for username which is promoted by this command)

```
$ git push -u angryBirds2.5
Username for 'https://github.com': whitehatjr
Password for 'https://whitehatjr@github.com':
Enumerating objects: 41, done.
Counting objects: 100% (41/41), done.
Delta compression using up to 4 threads
Compressing objects: 100% (37/37), done.
Writing objects: 100% (41/41), 2.54 MiB | 410.00 KiB/s, done.
Total 41 (delta 12), reused 0 (delta 0)
remote: Resolving deltas: 100% (12/12), done.
To https://github.com/whitehatjr/AngryBirdsStage2.5StudentActivity
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'angryBirds2.5'.
$
```

7. You could see all the commits ever made on the repository by typing the command "git log".

```
$ git log
commit a2e7acbe0692ceed25264239ce2c19fa98139067 (HEAD -> master, angryBirds2.5/master)
Author: rajeev.artha <rajeev.artha@gmail.com>
Date: Sun Jul 14 03:47:18 2019 +0530

    Changed Readme

commit 02d825a57be6d990130f85683763c50118baa1e9
Author: rajeev.artha <rajeev.artha@gmail.com>
Date: Sun Jul 14 03:18:29 2019 +0530

    Add Platform

commit 7510aae70f02d82653bbf1a3b36ba33acf1e705e (origin/master, origin/HEAD)
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Fri Jul 12 00:15:23 2019 +0530

    Update sketch.js

commit 8e6a9e2765d781619dead3a302de654065884858
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Fri Jul 12 00:10:49 2019 +0530

    Delete test

commit ebfc4fd674a96f3786287aea18a857752a02350b
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Thu Jul 11 23:59:54 2019 +0530

    Add files via upload

commit e3e87fc8e423b55a535ee5932023eccecd1dd6cc4
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Thu Jul 11 23:59:04 2019 +0530

    Create test module

commit dad39f2e944eb5e923b203587f0a414461d18243
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Thu Jul 11 23:58:32 2019 +0530

    Add files via upload

commit ab53b027c3934524c580ea3d704ce652e9b8d7d5
Author: whitehatjr <52084703@whitehatjrusers.noreply.github.com>
Date: Thu Jul 11 23:57:11 2019 +0530

    Initial commit
```

What's NEXT?

In the next class, you will be learning about constrained bodies.

EXTEND YOUR KNOWLEDGE:

To understand git and command line watch the following video.

<https://youtu.be/HVsySz-h9r4>