

Github Guide

by Drew

You'll learn how to:

- Create and use a repository
- Make changes to a file and push them to Github as commits for assignments

What is Github?

Github is a code hosting platform for version control and collaboration. It lets you add other work together on projects from anywhere.

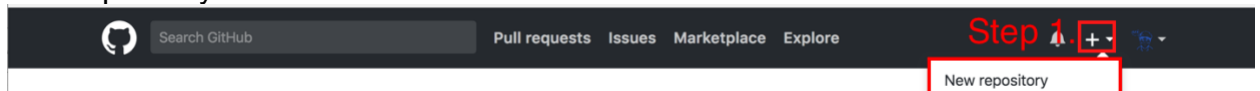
Step 1. Create a Repository

A **repository** is usually used to organize a single project. Repository (Repo) can contain folders and files, images, videos, spreadsheets, and data sets – anything your project needs.

Your repository can be a place where you store ideas, resources, or even share and discuss things with others.

To create a new repository

1. In the upper right corner, next to your avatar click on the plus + and then select new repository.



2. Name your repository the assignment name. Example: (kickstart-my-chart)
3. Write a short description
4. Select Initialize this repository with a README and add a license MIT License
5. Click on the Green button Create repository

Create a new repository

A repository contains all the files for your project, including the revision history.

Step 2.

Owner	Repository name
 codentell ▾	/ data-class-assignment-example ✓

Great repository names are short and memorable. Need inspiration? How about **supreme-enigma**.

Description (optional)

Github Data class assignment |

Step 3.

- ☒  **Public**
Anyone can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

Step 4.

☒ Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **MIT License** ▾

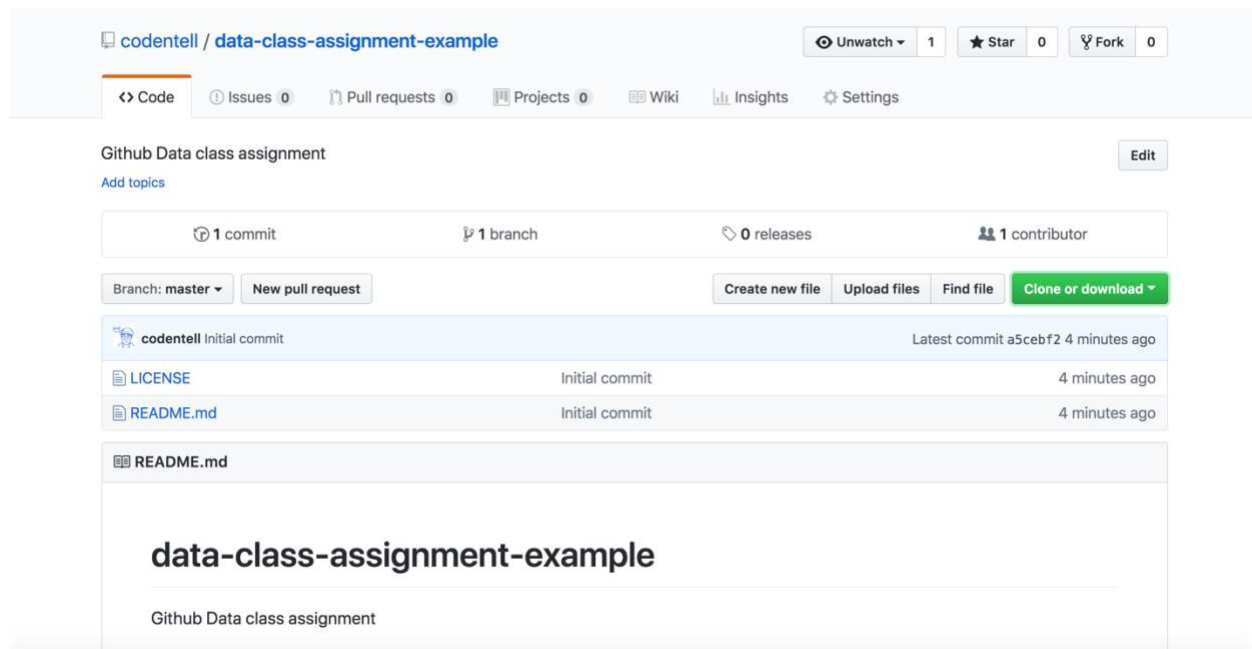


Create repository

Step 5.

Step 2. Clone your Repository

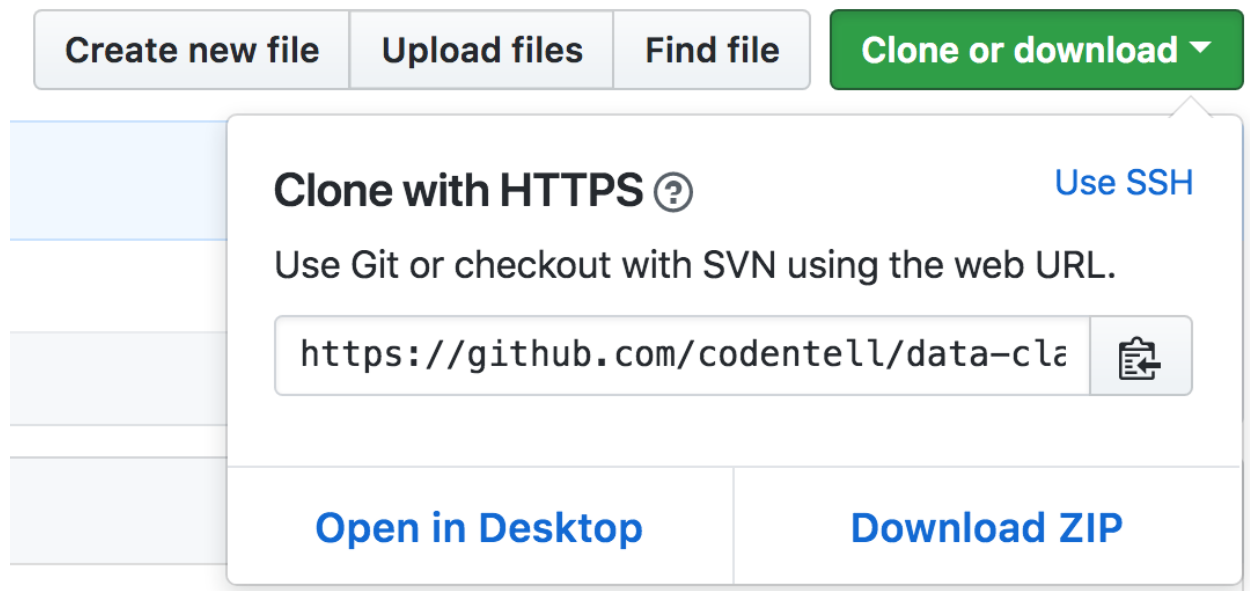
Your repository should look something similar to the image below.



Now we are going to Clone the repo to our local machine so in the future push all our work to github.

Clone or download ▾

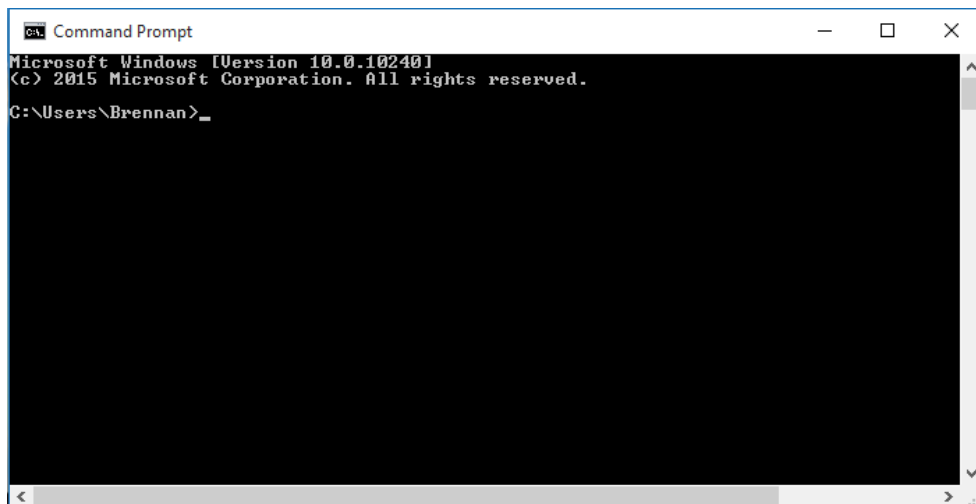
1. Click on the green button
2. A dropdown should show up and look like this



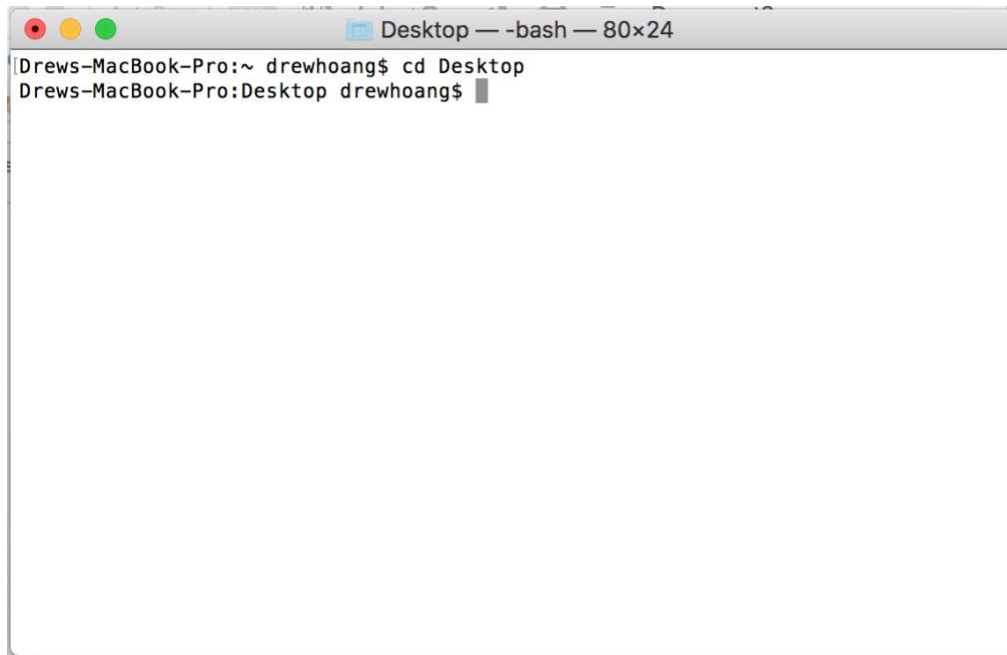
3. Click on the clipboard button



- Now we open the application Terminal (MAC) or Command Prompt (PC)

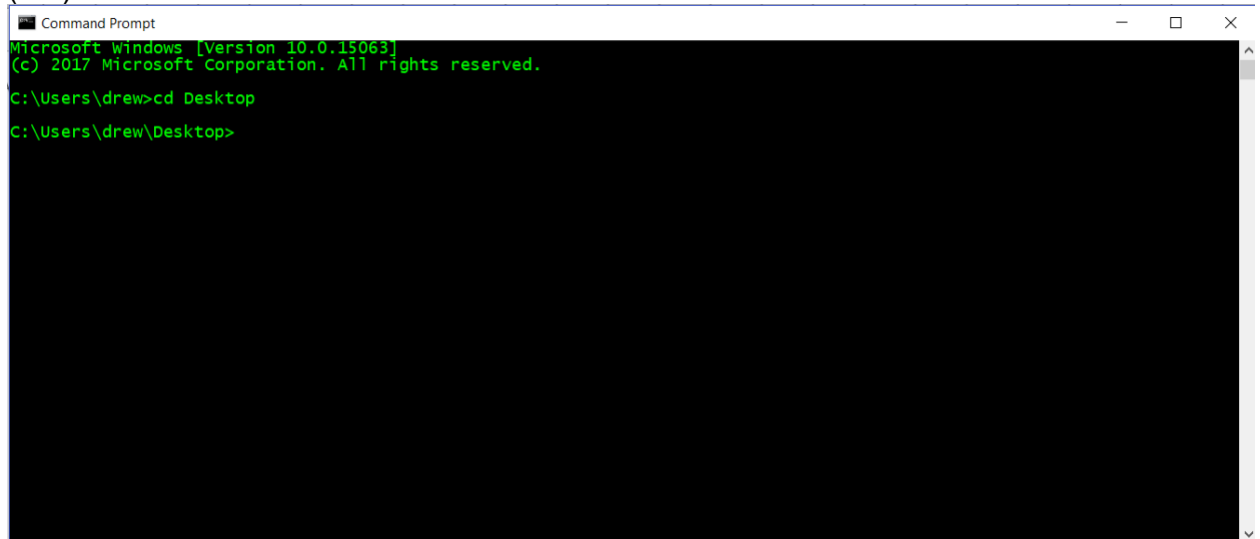


- Type this command: **cd Desktop**
(MAC)



```
Desktop — -bash — 80x24
Drews-MacBook-Pro:~ drewhoang$ cd Desktop
Drews-MacBook-Pro:Desktop drewhoang$
```

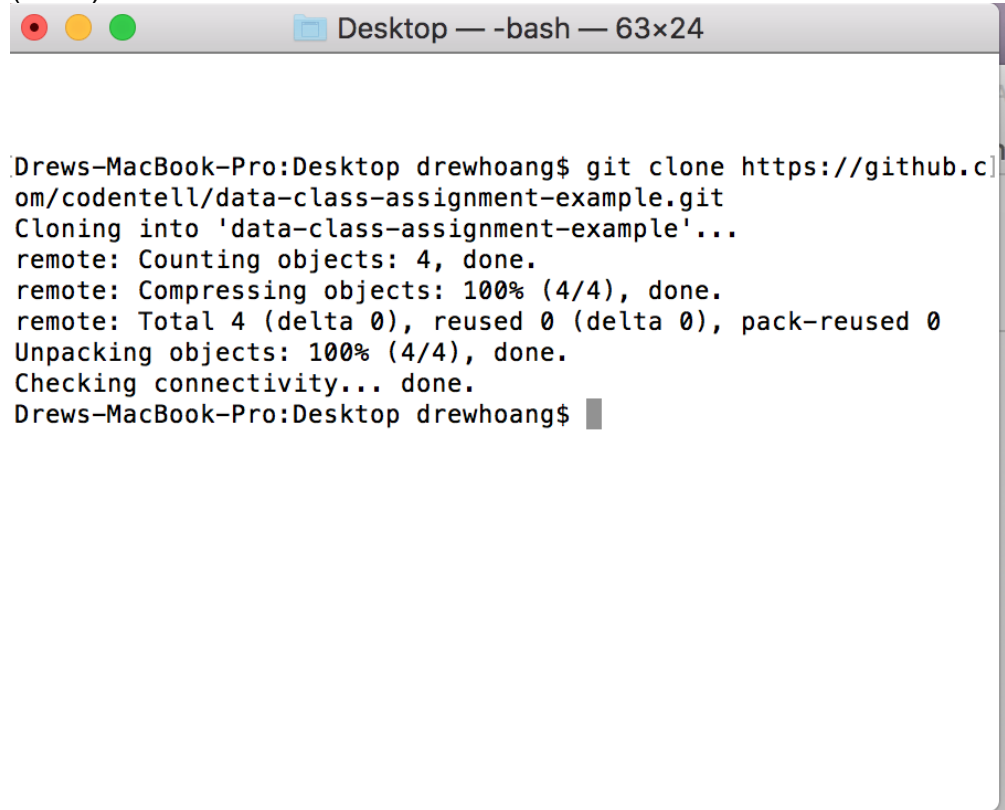
(PC)



```
Command Prompt
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.
C:\Users\drew>cd Desktop
C:\Users\drew\Desktop>
```

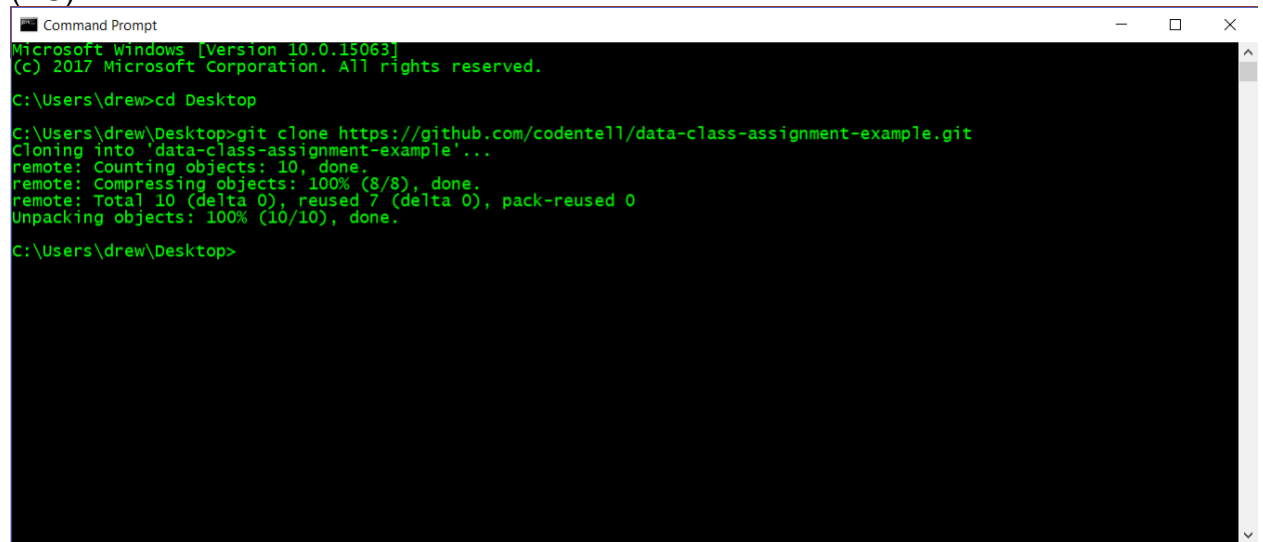
6. Once you are in Desktop you want to type: **git clone <repository-url>**
Note: the repository-url is the link we got from the clipboard button in step 2
So it will look like the image below:

(MAC)

A screenshot of a macOS terminal window. The title bar shows three colored window control buttons (red, yellow, green) and the text 'Desktop — -bash — 63x24'. The terminal text shows the execution of a git clone command, with progress updates from the remote repository and a final success message.

```
Drewhs-MacBook-Pro:Desktop drewhoang$ git clone https://github.com/codentell/data-class-assignment-example.git
Cloning into 'data-class-assignment-example'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
Checking connectivity... done.
Drewhs-MacBook-Pro:Desktop drewhoang$
```

(PC)

A screenshot of a Windows Command Prompt window. The title bar shows the text 'Command Prompt' and standard window control buttons. The terminal text shows the execution of a git clone command, with progress updates from the remote repository and a final success message.

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

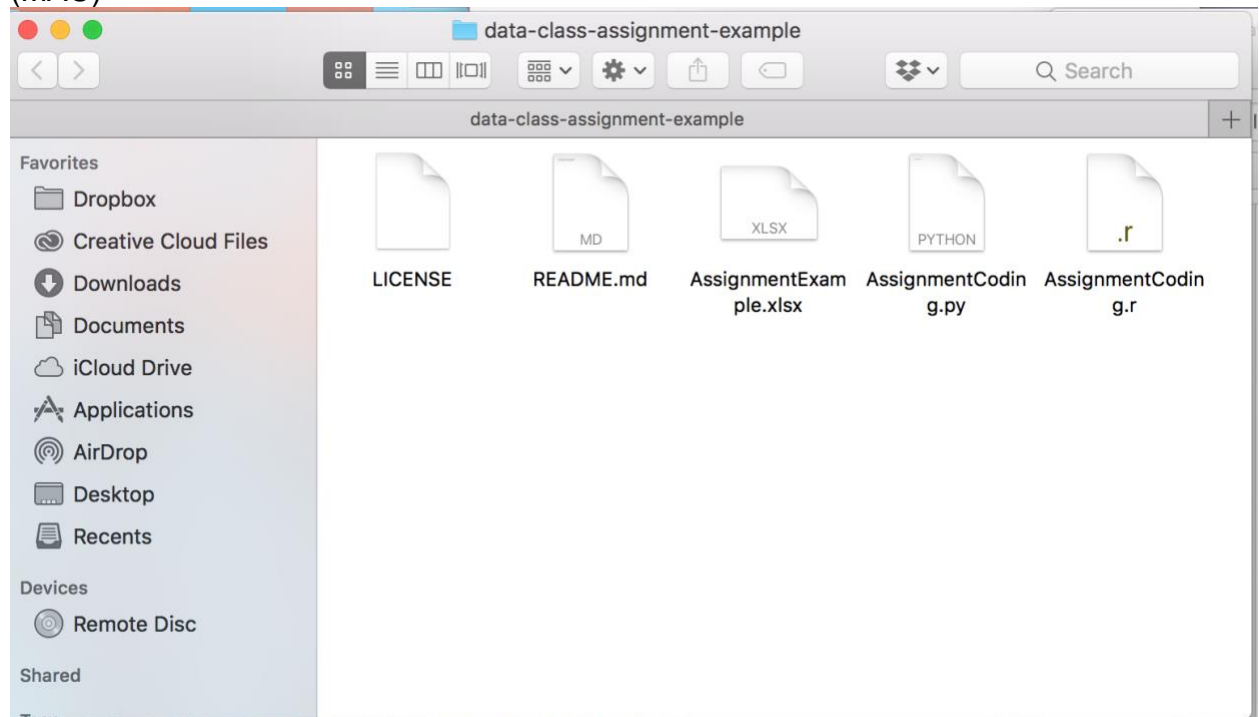
C:\Users\drew>cd Desktop

C:\Users\drew\Desktop>git clone https://github.com/codentell/data-class-assignment-example.git
Cloning into 'data-class-assignment-example'...
remote: Counting objects: 10, done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 10 (delta 0), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (10/10), done.

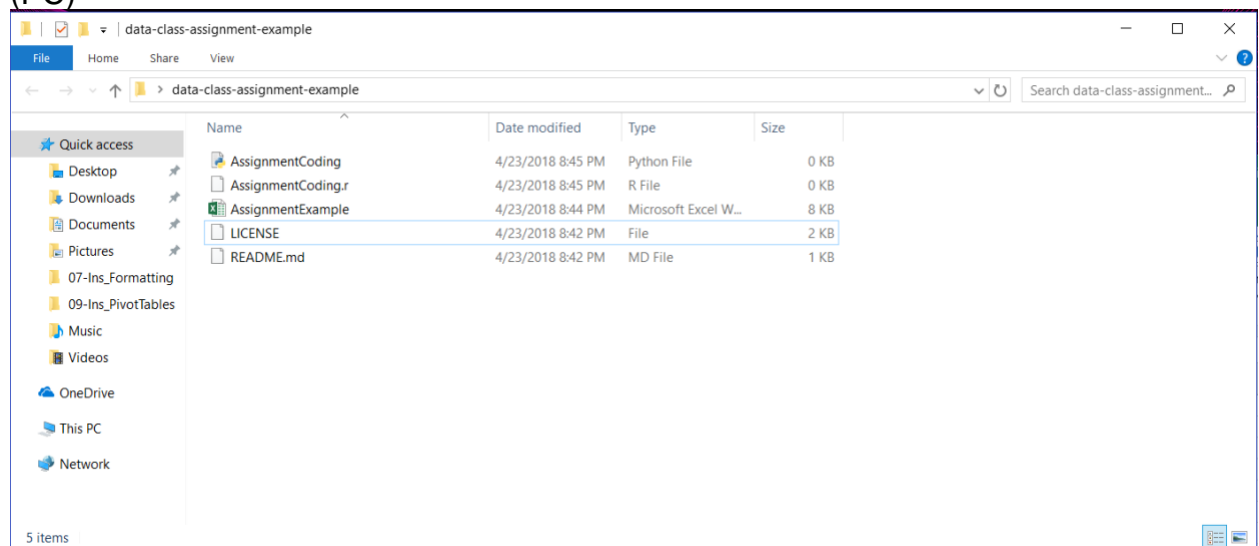
C:\Users\drew\Desktop>
```

7. Now check the Desktop and see if the folder is created.
8. Now let's add files to the our Repository. Example I added three files.

(MAC)



(PC)



9. In the terminal or command prompt we want to go to the repository root so type this command:

cd <folder name>

(MAC)

```
data-class-assignment-example — -bash — 83x24
[Drews-MacBook-Pro:Desktop drewhoang$ cd data-class-assignment-example/
Drews-MacBook-Pro:data-class-assignment-example drewhoang$
```

(PC)

```
Command Prompt
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\drew>cd Desktop

C:\Users\drew\Desktop>git clone https://github.com/codentell/data-class-assignment-example.git
Cloning into 'data-class-assignment-example'...
remote: Counting objects: 10, done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 10 (delta 0), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (10/10), done.

C:\Users\drew\Desktop>cd data-class-assignment-example
C:\Users\drew\Desktop\data-class-assignment-example>
```

10. In the terminal or command prompt we want to see the status. So type this command: **git status**
(MAC)


```
data-class-assignment-example — -bash — 83x24
Drews-MacBook-Pro:Desktop drewhoang$ cd data-class-assignment-example/
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
Drews-MacBook-Pro:data-class-assignment-example drewhoang$
```

(PC)

```
Command Prompt
C:\Users\drew\Desktop>cd data-class-assignment-example
C:\Users\drew\Desktop\data-class-assignment-example>git status
On branch master
Your branch is up to date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx
        ~$AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\drew\Desktop\data-class-assignment-example>
```

11. Then we want to add the changes in red so type the command: **git add .**
(MAC)

```
data-class-assignment-example — -bash — 83x24
Drews-MacBook-Pro:Desktop drewhoang$ cd data-class-assignment-example/
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git add .
Drews-MacBook-Pro:data-class-assignment-example drewhoang$
```

(PC)

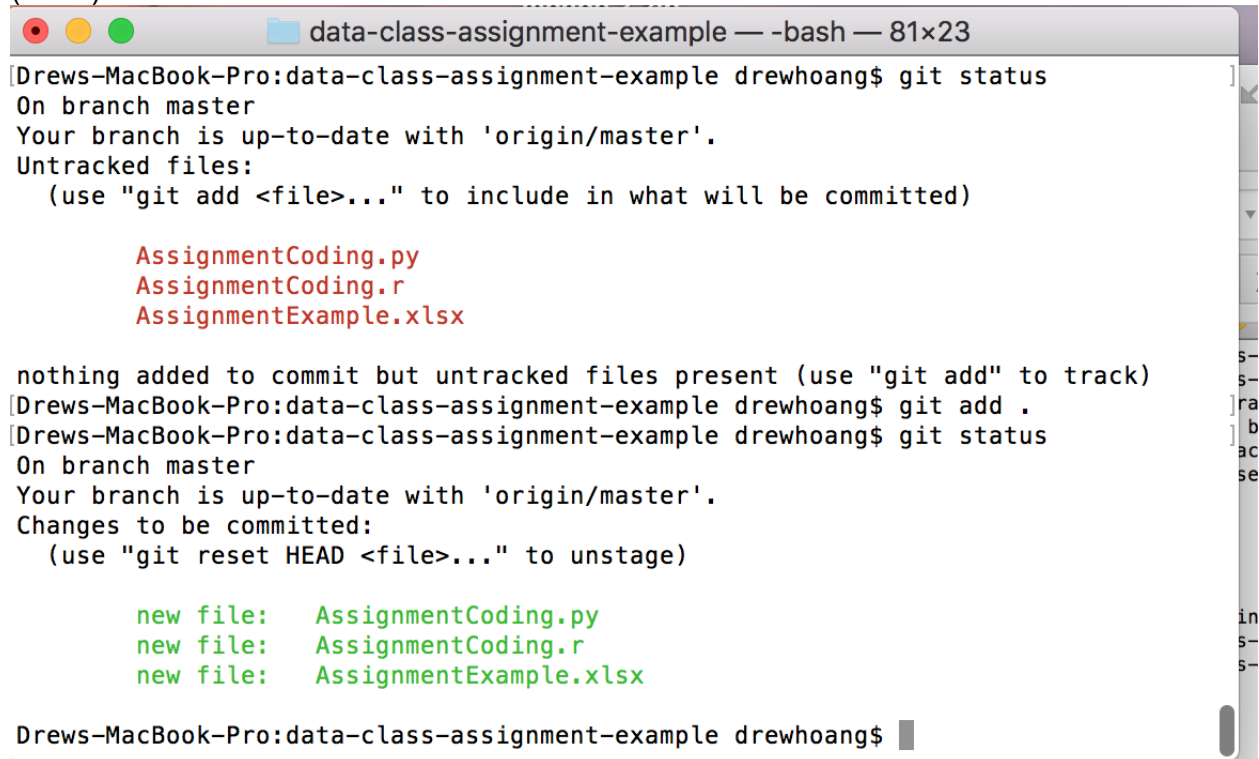
```
Command Prompt
C:\Users\drew\Desktop\data-class-assignment-example>git status
On branch master
Your branch is up to date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\drew\Desktop\data-class-assignment-example>git add .
C:\Users\drew\Desktop\data-class-assignment-example>
```

12. Then type the command **git status** to see your changes being approved in green

(MAC)



```
data-class-assignment-example — -bash — 81x23
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

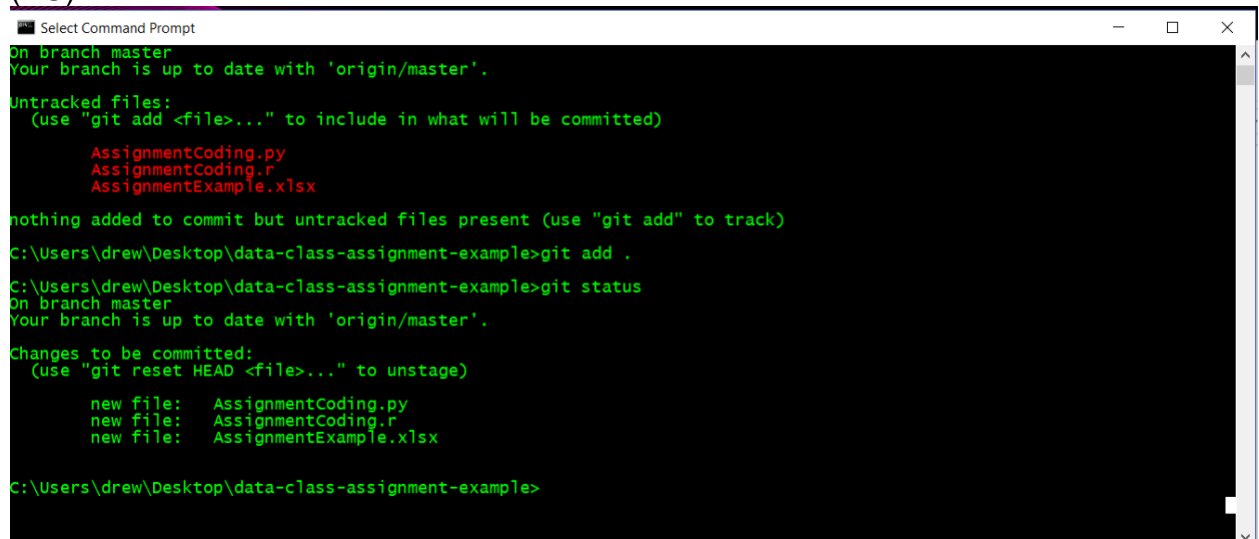
        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git add .
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   AssignmentCoding.py
        new file:   AssignmentCoding.r
        new file:   AssignmentExample.xlsx

Drews-MacBook-Pro:data-class-assignment-example drewhoang$
```

(PC)



```
Select Command Prompt
On branch master
Your branch is up to date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        AssignmentCoding.py
        AssignmentCoding.r
        AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\drew\Desktop\data-class-assignment-example>git add .
C:\Users\drew\Desktop\data-class-assignment-example>git status
On branch master
Your branch is up to date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   AssignmentCoding.py
        new file:   AssignmentCoding.r
        new file:   AssignmentExample.xlsx

C:\Users\drew\Desktop\data-class-assignment-example>
```

13. Then we must make our commit to indicate what we have done so far type the command: **git commit -m "<message of your commit>"**
(MAC)

```
data-class-assignment-example — -bash — 81x23

new file: AssignmentCoding.r
new file: AssignmentExample.xlsx

Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git commit -m "testing my commit"
[master 66325b9] testing my commit
Committer: codentell <drewhoang@Drews-MacBook-Pro.local>
Your name and email address were configured automatically based on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

3 files changed, 1 insertion(+)
create mode 100644 AssignmentCoding.py
create mode 100644 AssignmentCoding.r
create mode 100644 AssignmentExample.xlsx
Drews-MacBook-Pro:data-class-assignment-example drewhoang$
```

(PC)

```
Command Prompt

(use "git add <file>..." to include in what will be committed)

AssignmentCoding.py
AssignmentCoding.r
AssignmentExample.xlsx

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\drew\Desktop\data-class-assignment-example>git add .
C:\Users\drew\Desktop\data-class-assignment-example>git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   AssignmentCoding.py
    new file:   AssignmentCoding.r
    new file:   AssignmentExample.xlsx

C:\Users\drew\Desktop\data-class-assignment-example>git commit -m "testing my commit for github"
[master 73462a4] testing my commit for github
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 AssignmentCoding.py
create mode 100644 AssignmentCoding.r
create mode 100644 AssignmentExample.xlsx
C:\Users\drew\Desktop\data-class-assignment-example>
```

14. Then we push all our changes to github where it will be shown in your github account so type the command: **git push**

(MAC)

```
data-class-assignment-example — -bash — 81x23
Drews-MacBook-Pro:data-class-assignment-example drewhoang$ git push
warning: push.default is unset; its implicit value has changed in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the traditional behavior, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

When push.default is set to 'matching', git will push local branches
to the remote branches that already exist with the same name.

Since Git 2.0, Git defaults to the more conservative 'simple'
behavior, which only pushes the current branch to the corresponding
remote branch that 'git pull' uses to update the current branch.

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://github.com': codentell
Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 6.10 KiB | 0 bytes/s, done.
Total 5 (delta 0), reused 0 (delta 0)
To https://github.com/codentell/data-class-assignment-example.git
a5cebf2..66325b9 master -> master
```

(PC)

```
Command Prompt
C:\Users\drew\Desktop\data-class-assignment-example>git commit -m "testing my commit for github"
[master 73462a4] testing my commit for github
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 AssignmentCoding.py
create mode 100644 AssignmentCoding.r
create mode 100644 AssignmentExample.xlsx

C:\Users\drew\Desktop\data-class-assignment-example>git push
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 5.98 KiB | 2.99 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/codentell/data-class-assignment-example.git
e90ccb8..73462a4 master -> master

C:\Users\drew\Desktop\data-class-assignment-example>
```

15. Finally check your github account to see what you uploaded.

Github Data class assignment

[Edit](#)[Add topics](#)

4 commits

1 branch

0 releases

1 contributor

MIT

Branch: master

[New pull request](#)[Create new file](#)[Upload files](#)[Find file](#)[Clone or download](#)

codentell testing my commit for github

Latest commit 73462a4 2 minutes ago

AssignmentCoding.py	testing my commit for github	2 minutes ago
AssignmentCoding.r	testing my commit for github	2 minutes ago
AssignmentExample.xlsx	testing my commit for github	2 minutes ago
LICENSE	Initial commit	2 hours ago
README.md	Initial commit	2 hours ago

README.md

data-class-assignment-example