assignment01

Seung Yeop, Seon (20144753)

September 18, 2018

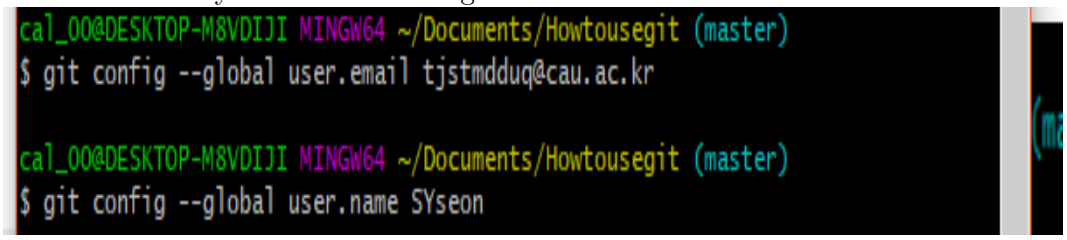
1 Introduction

This Document is dealing with how to use Git and benefits from utilizing it.

2 The explantions of Git's commands

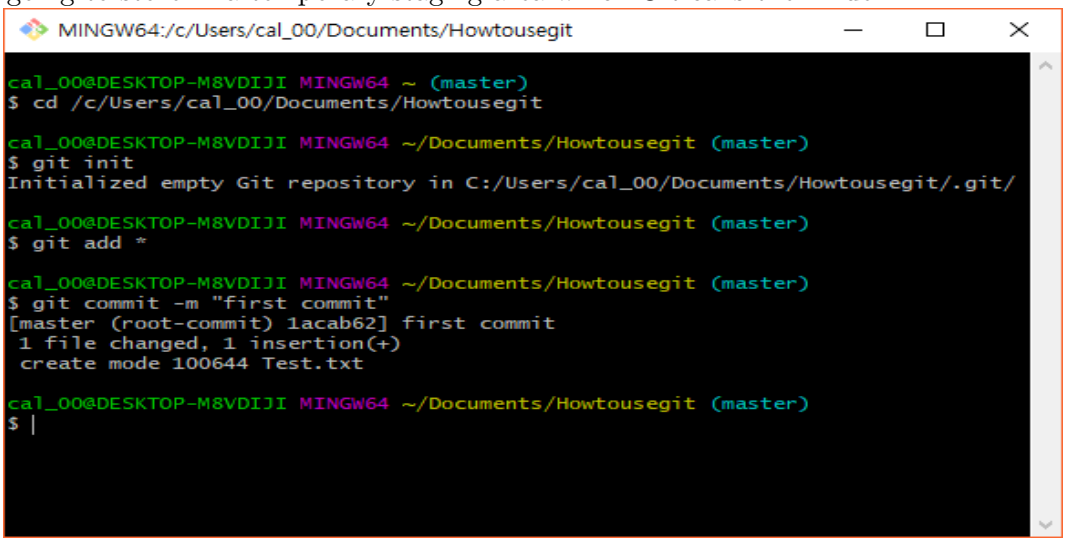
2.1 Put your personal informations

-Before doing any operation, put your email and name. When you collaborate with others, This informations are very useful for checking who wrote the code.



2.2 Importing a new project

-At first, enter the above code **cd address**. then, you can get in the folder named **Howtousegit**. Second, **git init** makes local repository in your folder. Third, **git add** makes your files in the folder are going to store in a temporary staging area which Git calls the "index".



2.3 Committing your files

You can permanently store the contents of the index in the repository by using git commit. Let's take a look at some useful code related to git commit.

-Through the following code, we can see if the file has been modified. If you use this code before you commit, it will be useful.

```
cal_OO@DESKTOP-M8VDIJI MINGW64 ~/Documents/Howtousegit (master)
$ git diff
diff --git a/Test.txt b/Test.txt
index d800886..ee64844 100644
--- a/Test.txt
+++ b/Test.txt
@@ -1 +1 @@
-123
\ No newline at end of file
+123asd
\ No newline at end of file
```

-Using **git status**, you can check the status about the commit.

```
cal_OO@DESKTOP-M8VDIJI MINGW64 ~/Documents/Howtousegit (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   Test.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

-Now, we are committing. **-m** means to leave messages. If you don't this method, error will be raised.

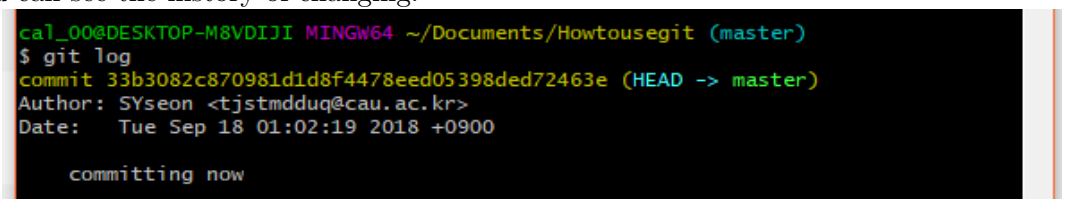
```
cal_OO@DESKTOP-M8VDIJI MINGW64 ~/Documents/Howtousegit (master)
$ git commit -m "committing now"
[master 33b3082] committing now
 1 file changed, 1 insertion(+), 1 deletion(-)
```

-Or you can use this method. If you use this method, you don't need to add your files before committing.

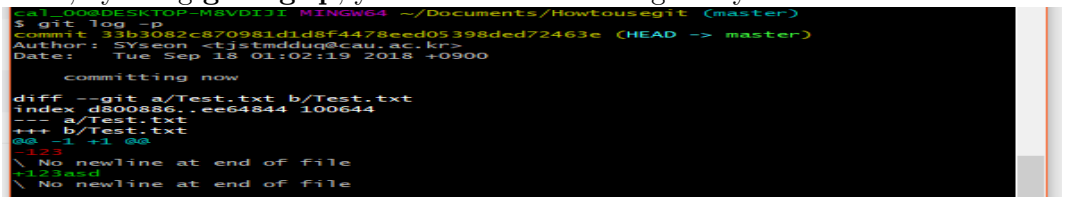
```
cal_OO@DESKTOP-M8VDIJI MINGW64 ~/Documents/Howtousegit (master)
$ git commit -a
On branch master
nothing to commit, working tree clean
```

2.4 Viewing project history

You can see the history of changing.



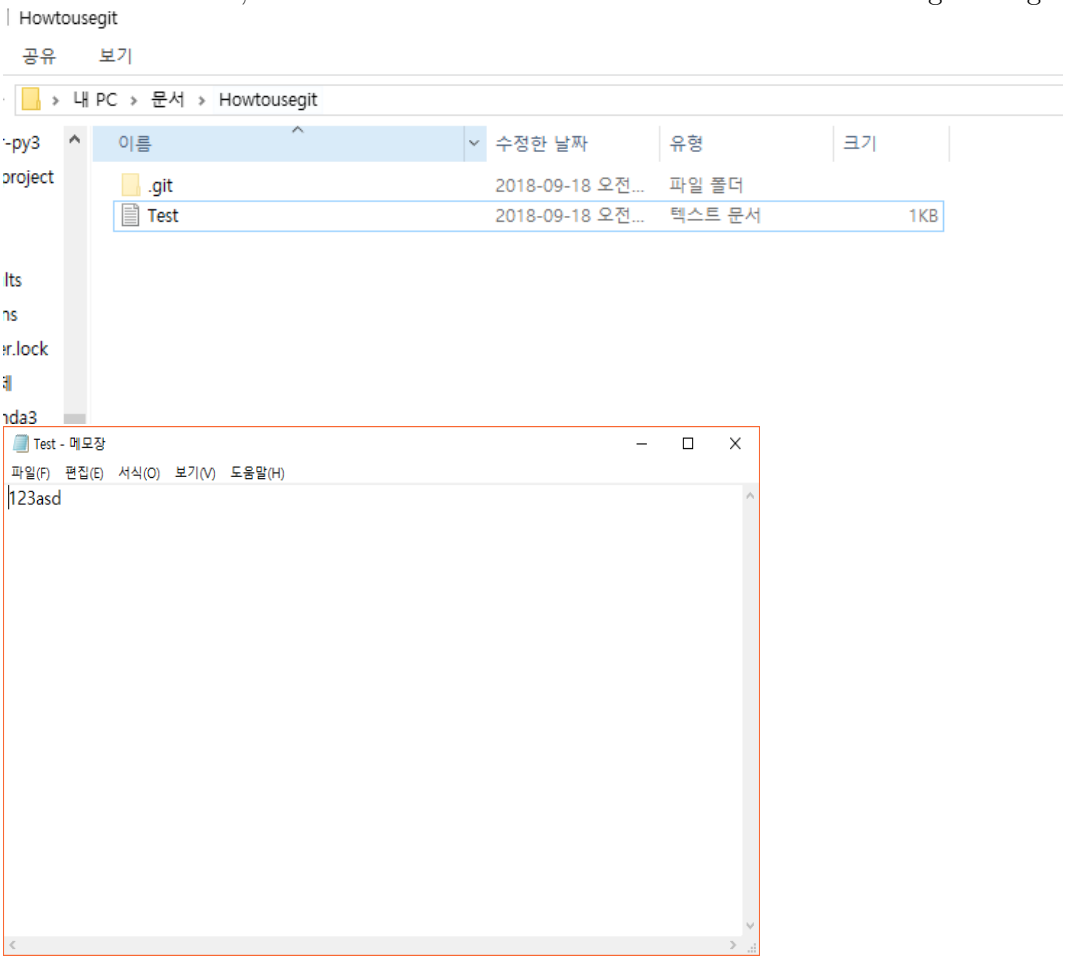
-Also, by using **git log -p**, you can see whole changes of your files.



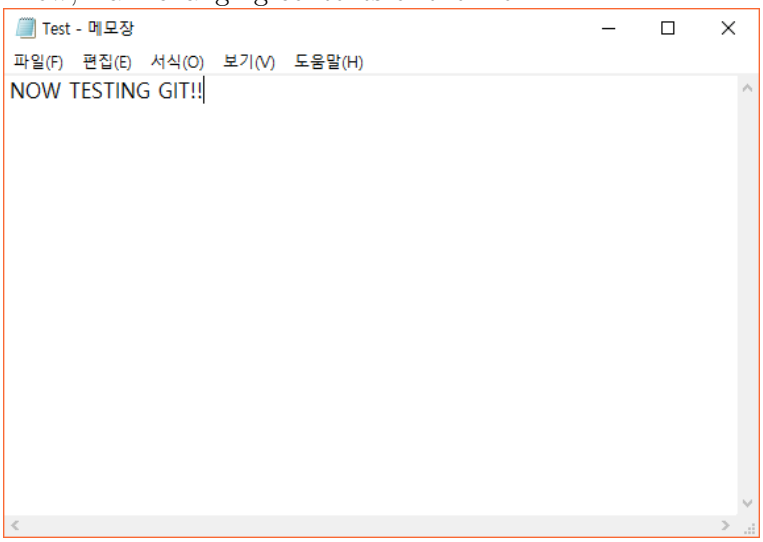
2.5 Branch

If you use branch, you can move to the point you specified. Let's look at how to use branch.

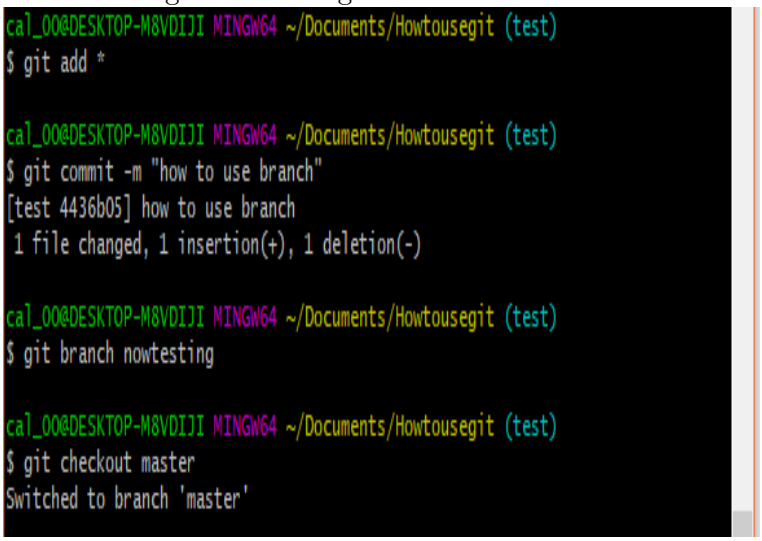
-Before committed, I created *Test.txt* and this file contained the following messages.



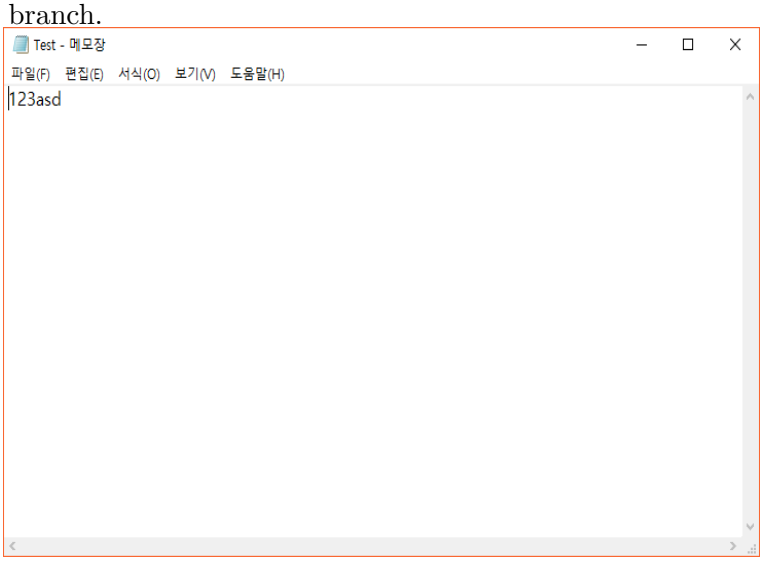
-Now, I am changing contents of the file.



-And entering the following commands.



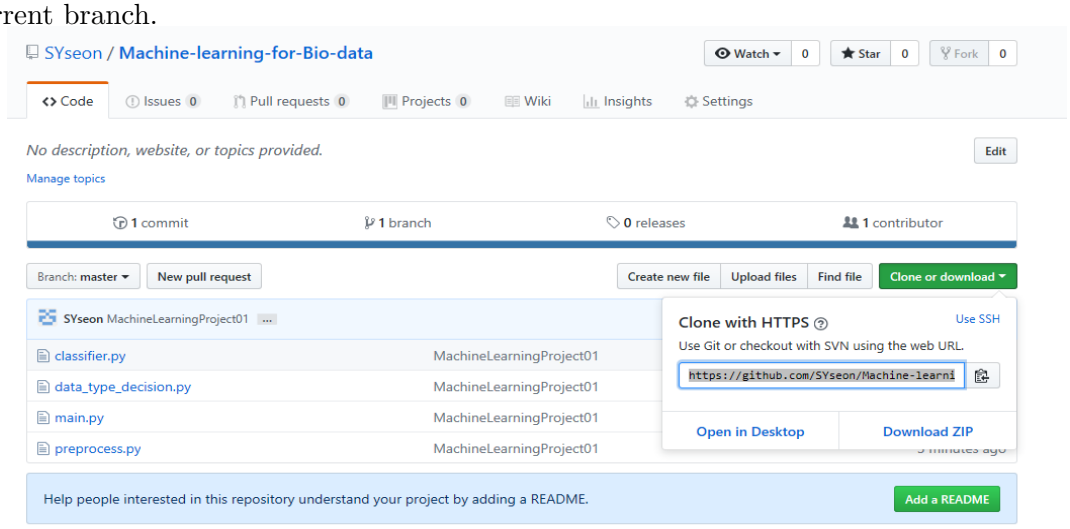
-We can see the file returns to first state. The master branch is the point at which the contents of the text file are saved before making changes. The nowtesting branch, on the other hand, is when the contents of the text file are saved after the change. We can move to the desired point through the branch.

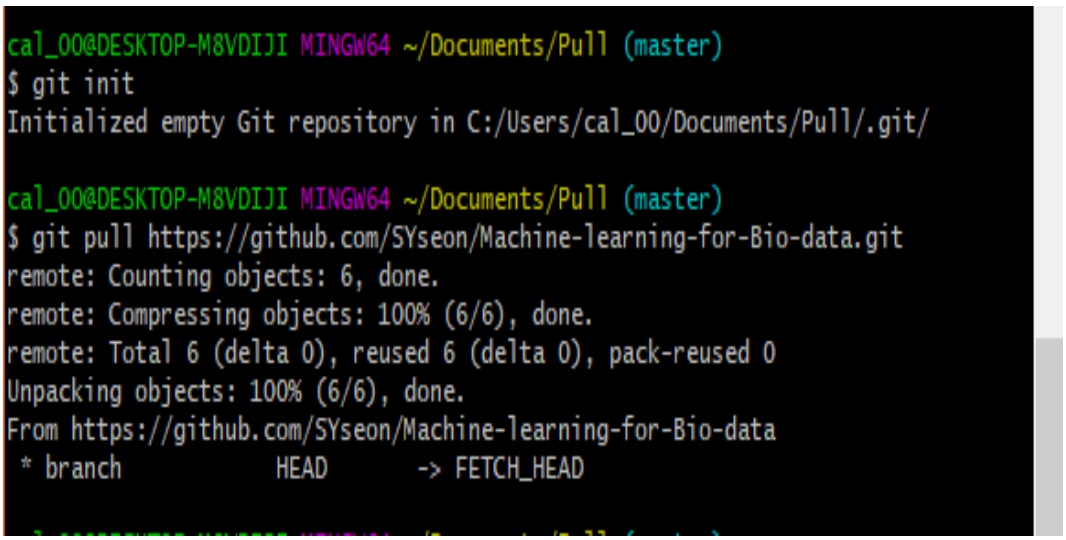


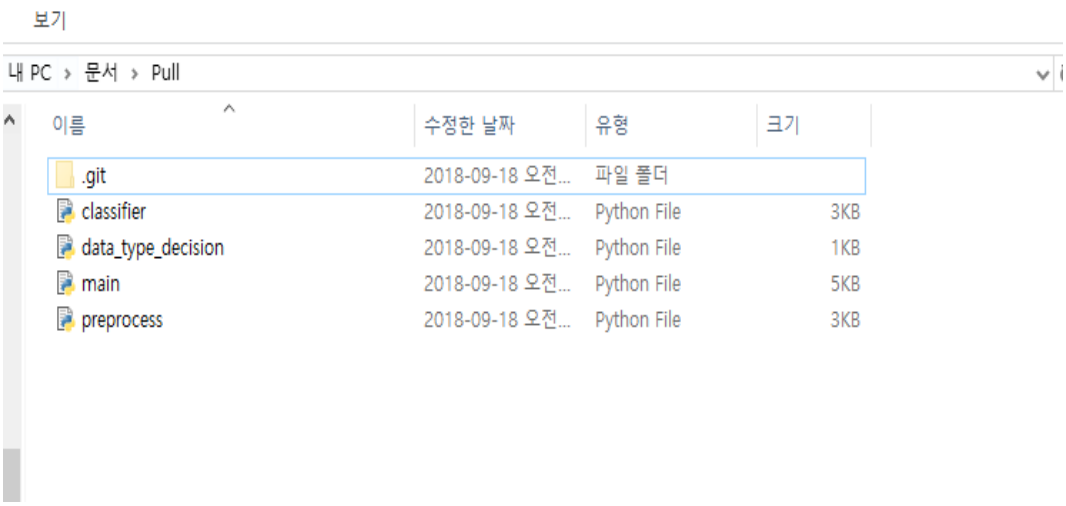
2.6 To access GitHub

2.6.1 git pull

Git pull merges the changes from Machine-learning-for-bio-data's "master" branch into Pull's current branch.

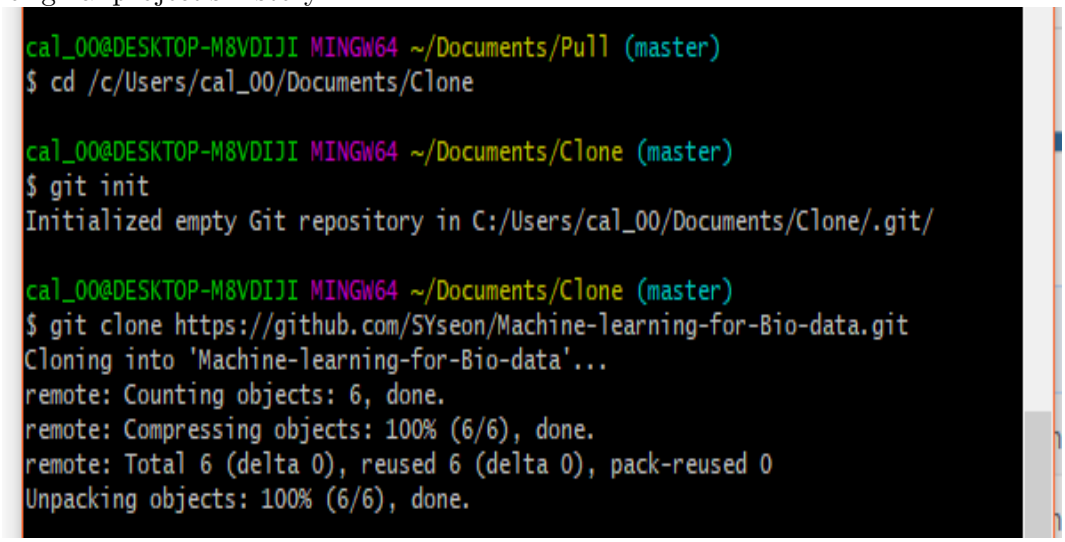


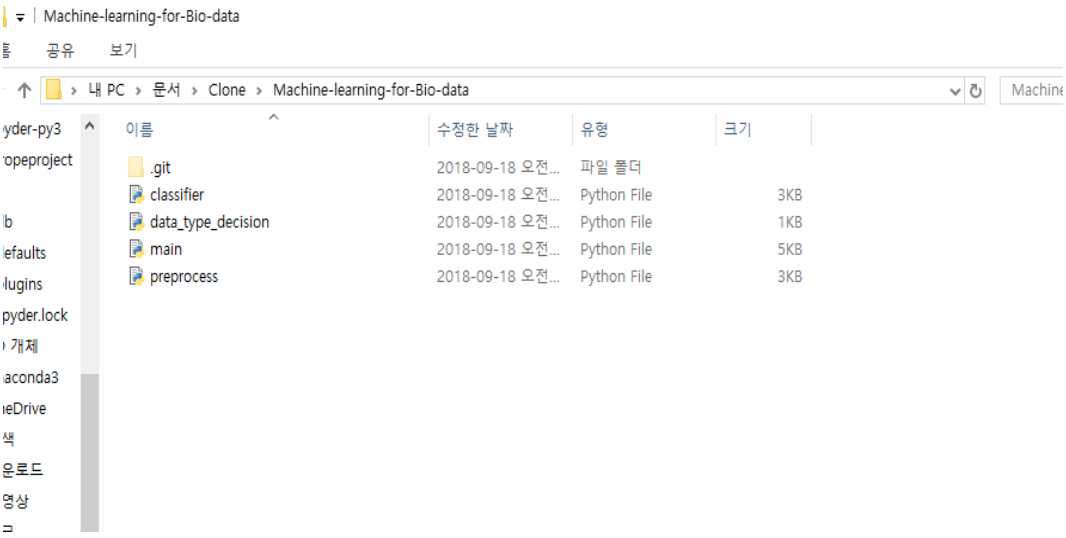




2.6.2 git clone

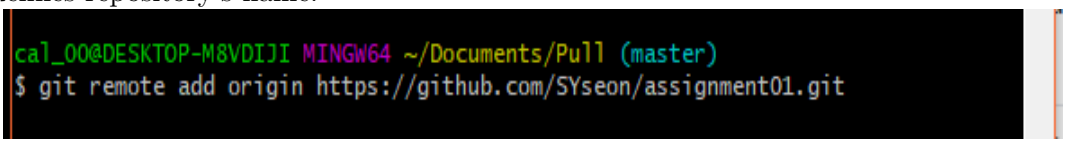
Git clone creates a new directory "Machine-learning-for-bio-data containing a clone of The Clone's repository. The clone is on an equal footing with the original project, possessing its own copy of the original project's history.





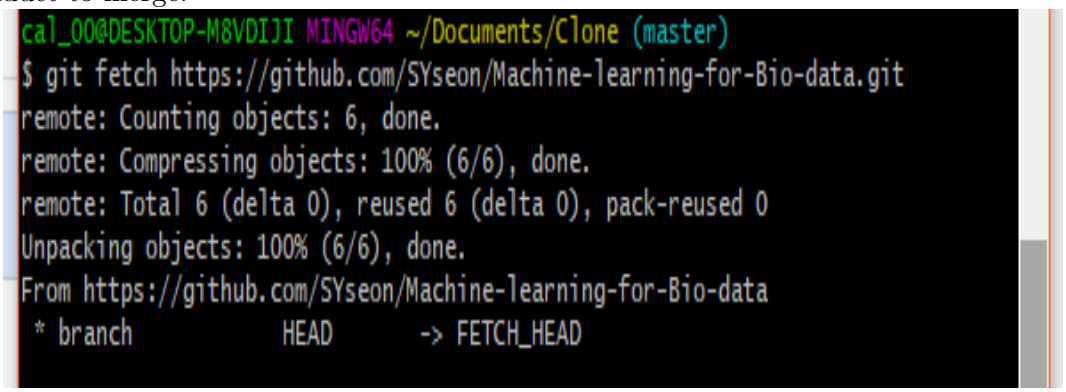
2.6.3 git remote

It defines repository's name.



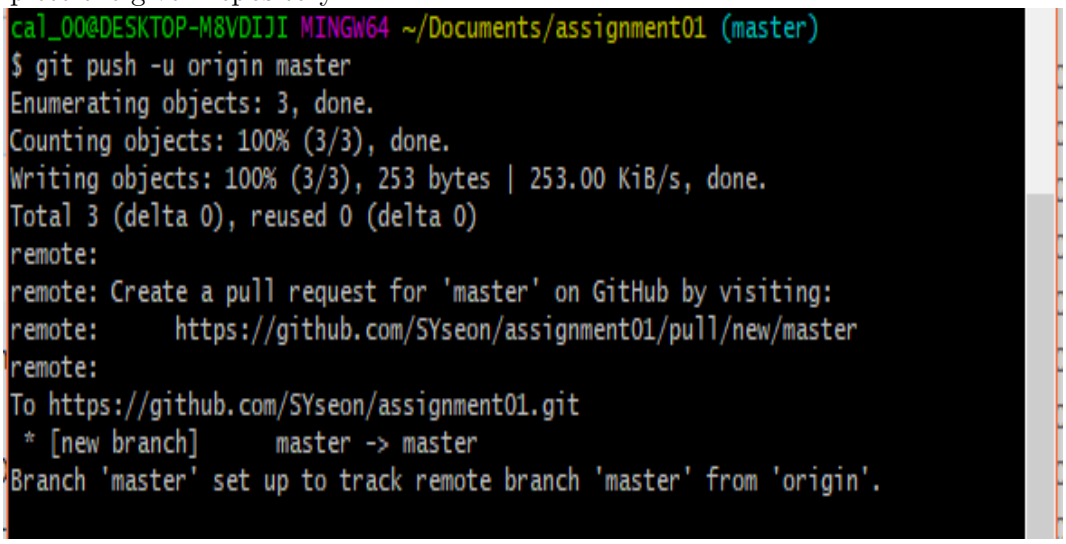
2.6.4 git fetch

Git fetch brings the source of the remote repository to the local repository. However it doesn't conduct to merge.



2.6.5 git push

Git push updates remote repository using local repository, while sending objects necessary to complete the given repository.



3 The benefits from using Git

Now, Let's talk about the advantages of Git using the features mentioned above.

3.1 Collaborating

In the above section, we can see that commit our branch in the local repository to the remote repository. Also, saw the reverse case. Through these processes, we can share our code with our colleagues instantly.

3.2 Version control

When you are programming, there are situations to code in a completely different way than the previous one. and sometimes you make mistakes. If you do not use git, this is pretty bothering. However, with git, you just have to go back to the previous branch.

3.3 The most popular code management tools

It's a big advantage that GitHub is the most popular code management tool. The vast majority of developers work with GitHub, we can only use GitHub to collaborate with other developers. Also, due to this trait, GitHub has a huge amount of open source. We can free access to open source.

4 GitHub link

https://github.com/SYseon/assignment01