GIT

https://www.liaoxuefeng.com/wiki/896043488029600

▼ Create repository (版本庫)

create a new repository

1. mkdir NAME
2. cd NAME
3. pwd (show the current path)
4. git init (turn the path into a repository that GIT can manage)

   If you don’t see “.git”, that means the path is hidden, use “ls -ah” and you’ll see it

add files to control version

1. git add (we can add a lot of files here) (put all the going to modify to temp store
2. git commit -m “XXX” (quote why we added the files above)

▼ Time machine

git status (see the status of the repoitory, modified or remain, but the file is still not pushed)

git diff NAME (show what is changed)

git diff HEAD —NAME (show the diff between the lastest version in the master and the temp store)

After changing a file, repeat the steps at “add files to control version”

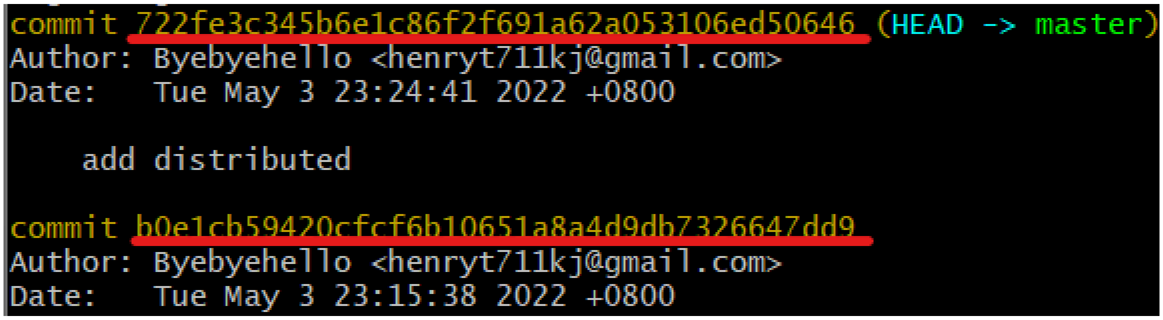
---

return to back versions

git log (we can see the commit we done before, from the latest to the oldest)

git reset —hard HEAD^ (return to the last one)

git reset —hard THE IDs behind commit (only needed to type in the head of the IDs) (return to the that has been reset, such as going into history then going back)

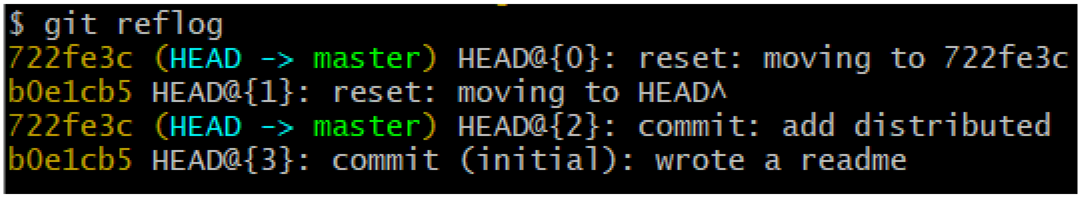


```
commit 722fe3c345b6e1c86f2f691a62a053106ed50646 (HEAD -> master)
Author: Byebyehello <henryt711kj@gmail.com>
Date:   Tue May 3 23:24:41 2022 +0800

    add distributed

commit b0e1cb59420cfcf6b10651a8a4d9db7326647dd9
Author: Byebyehello <henryt711kj@gmail.com>
Date:   Tue May 3 23:15:38 2022 +0800
```

git reflog (see every order you’ve done before)



```
$ git reflog
722fe3c (HEAD -> master) HEAD@{0}: reset: moving to 722fe3c
b0e1cb5 HEAD@{1}: reset: moving to HEAD^
722fe3c (HEAD -> master) HEAD@{2}: commit: add distributed
b0e1cb5 HEAD@{3}: commit (initial): wrote a readme
```

---

git restore — NAME (discard all the changes in NAME in the temp store, after using this command, NAME will return to the latest status in the master, if NAME is “git add”, this will do nothing)

---

git restore —staged NAME (return to the stage that it is not “git add”)

---

rm NAME (delete NAME)

▼ Remote repository

Add remote repository (Push)

1. Create a repo on github

2. $git remote add origin git@github.com:b086110466/(repoName).git
3. $git push -u origin master

   Using “$git push -u” will connect the local repo to remote repo. We will just have to “$git push origin master” from then on.

If you accidently typo the address of the remote repo

1. Check the address first

   ```
   $git remote -v
   ```

2. If it is wrong

   ```
   $git remote rm origin
   ```

3. Repeat the second step above.

Clone from remote repo

1. $git clone git@github.com:b08611046/(repoName).git
2. cd repoName

   Then the file will be there.

Branch management

git checkout -b dev (create a branch called dev)

= git branch dev + git checkout dev

git will * on the current branch

git checkout master (return to master branch)

git branch -d dev (delete branch dev)

---

git switch -c dev (new way to create a branch called dev)

git switch master (new way to switch between branches)

小結

Git鼓勵大量使用分支：

查看分支：`git branch`

創建分支：`git branch <name>`

切換分支：`git checkout <name>` 或者 `git switch <name>`

創建+切換分支：`git checkout -b <name>` 或者 `git switch -c <name>`

合併某分支到當前分支：`git merge <name>`

刪除分支：`git branch -d <name>`