

Lab 01

How to Survive & Introduction to Git

Software Studio
DataLab, CS, NTHU

Notice

- These slides will focus on how to submit you code by using Git command line
- You can also use other Git GUI tool or built-in Git tool in other IDE/editor

Outline

- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Outline

- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Teaching Assistants



Shao-Che Feng
馮邵哲



Chun-Ping Fang
方俊平



Tzu-Ling Hsieh
謝紫翎

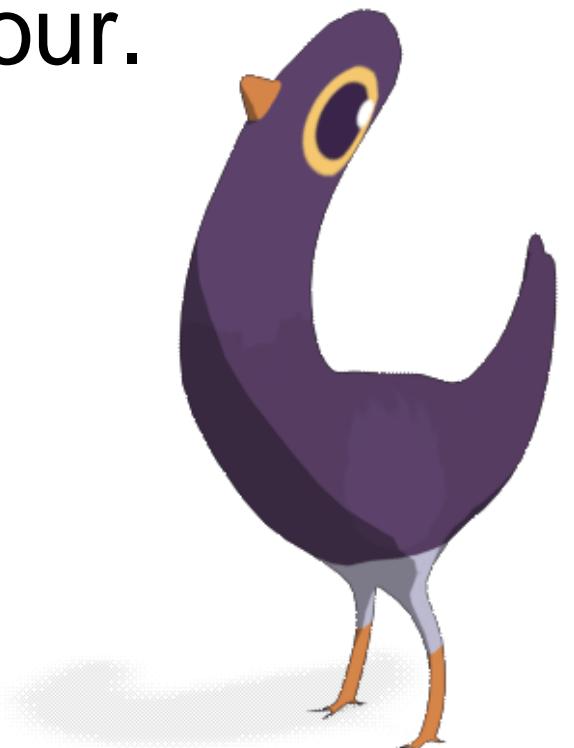
How to Find Us?

- Office Hour (TAs)
 - Mon. 10:10~12:00 at Delta 729
- Email (**Personal question only**)
 - ssta2025@datalab.cs.nthu.edu.tw
 - Shao-Che Feng: scfeng@datalab.cs.nthu.edu.tw
 - Chun-Ping Fang: cpfang@datalab.cs.nthu.edu.tw
 - Tzu-Ling Hsieh: tlhsieh@datalab.cs.nthu.edu.tw
- Online Forum
 - eeclass



If I have Question?

- Always Google first !
 - Learn how to google is important.
- If you try your best but still can't catch it.
 - Feel free to ask us on iLMS or office hour.



The Policy of Labs

- **All labs need to be submitted to GitLab.**
- Late submission will **not** be accepted.
- Plagiarism will not be tolerated.
 - If we find you copy someone's code, you will get **0 point** for that lab.

Team Up

- 2~4 people each team
 - 2 people is accepted if you can do as well as others.
- Please register your team here before **3/13 23:59**
 - Register form : <https://goo.gl/KVeKNE>
 - After that day we will match the rest student.

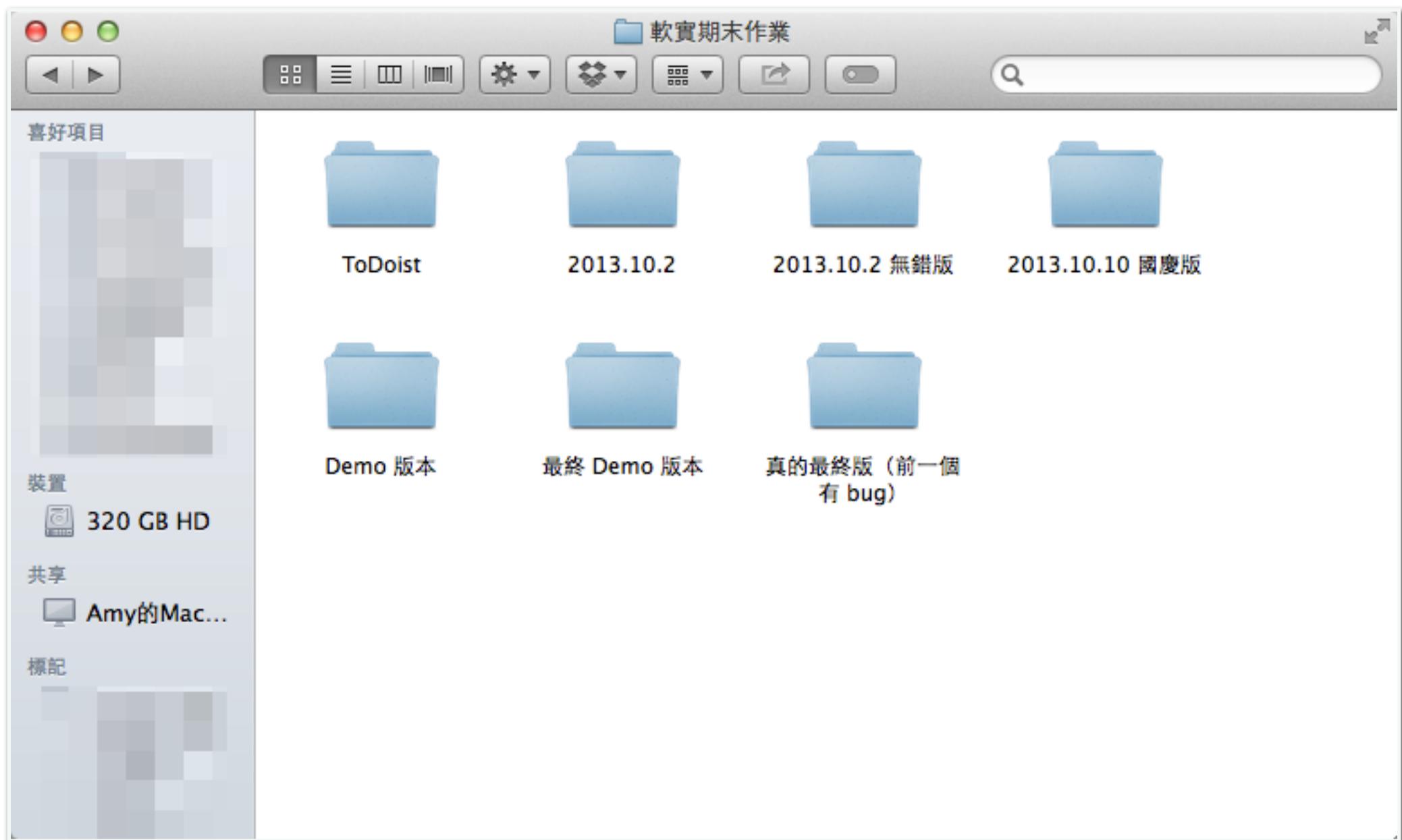
Outline

- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Why use version control?

We want to track what we did and when we did it.

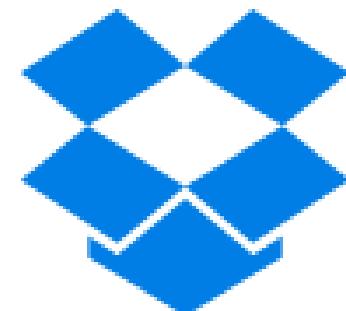
Students' VCS



How to work with others?



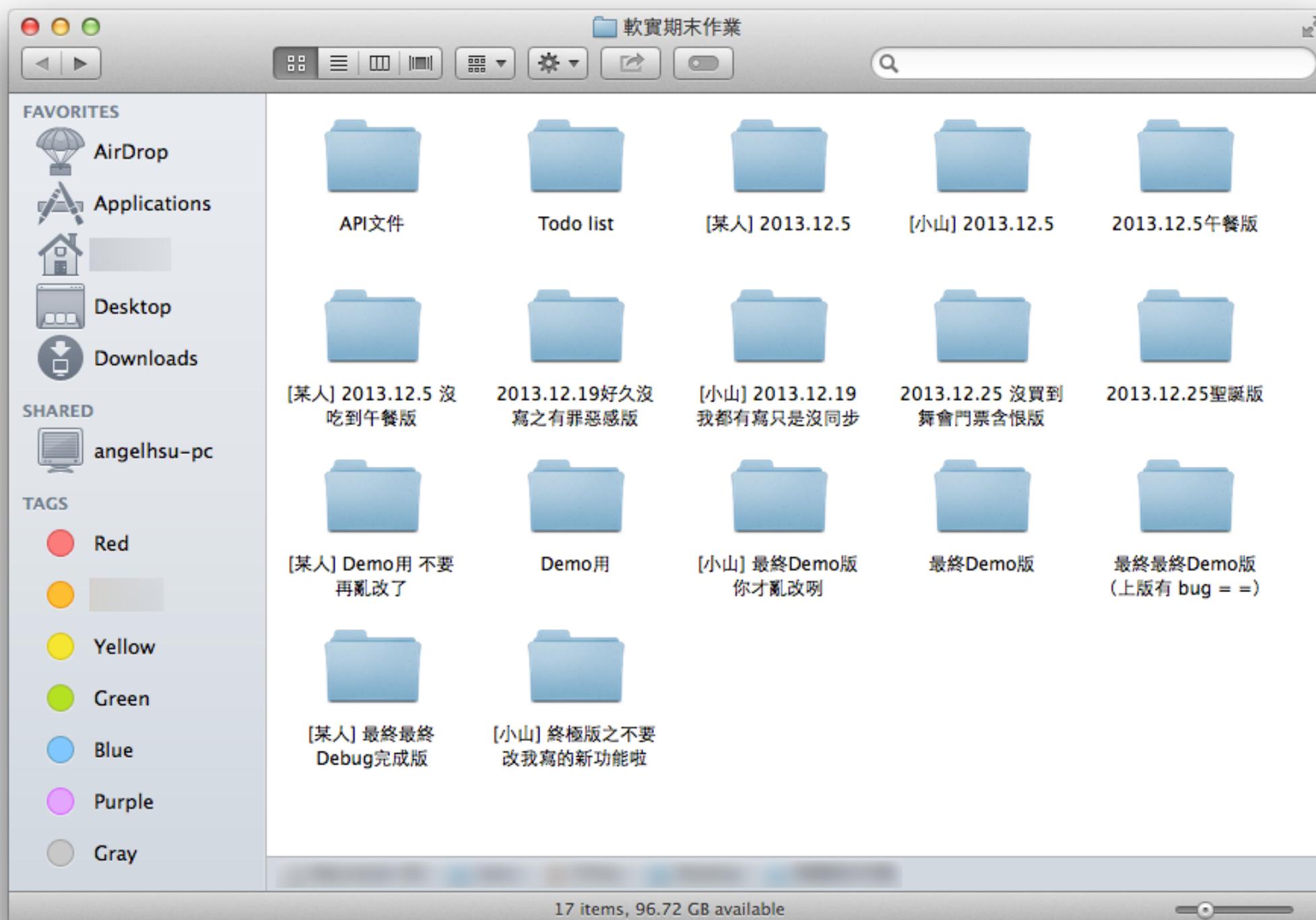
?



Dropbox

?

Dropbox VCS in Reality



Why use VCS?

- Managing your projects - tracking your files and modifications.
- Synchronization between modifications made by different developers.
- Revision history is still very helpful even if you work alone.

Outline

- General Rule
- Introduction to Git
 - Version control
 - **Git Basics**
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Git



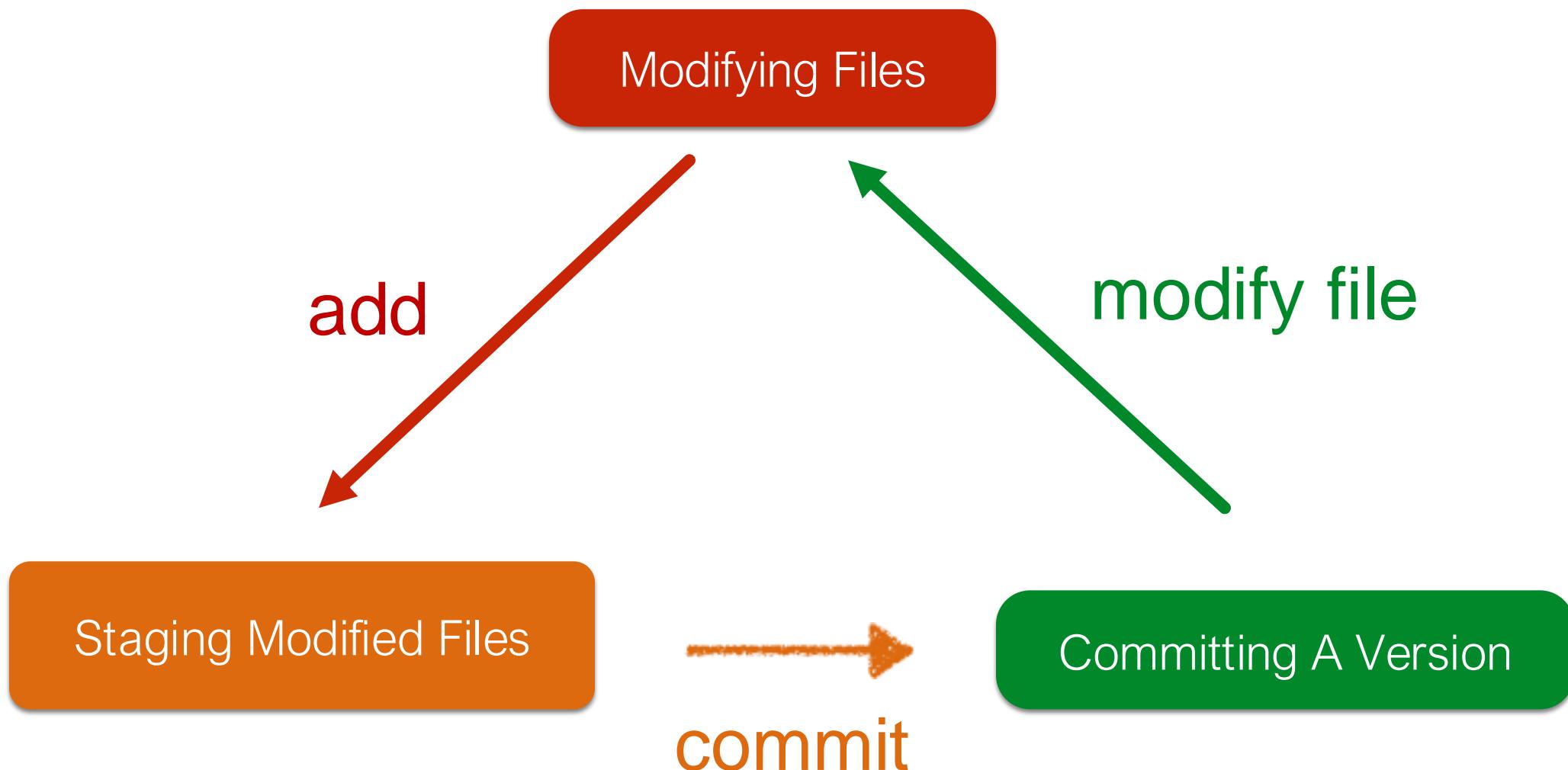
Git

- Git is a popular version control system which is
 - Fast
 - Easy to use
 - Distributed
- A git repository is a mini database that tracks your files.

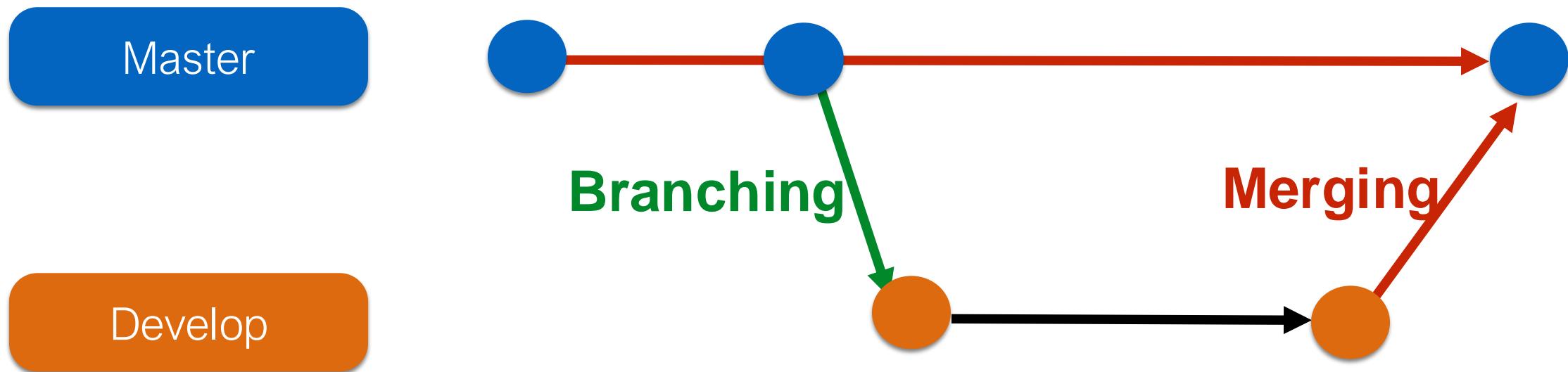
Git Workflow (1/2)

- With a local repository in your computer, you'll need following operations to make git track your work:
 - Create/modify files
 - Let git monitor the files by *adding* them to staging files.
 - Commit* your changes to and git will create snapshots (versions) of the files for you.

Git Workflow (2/2)



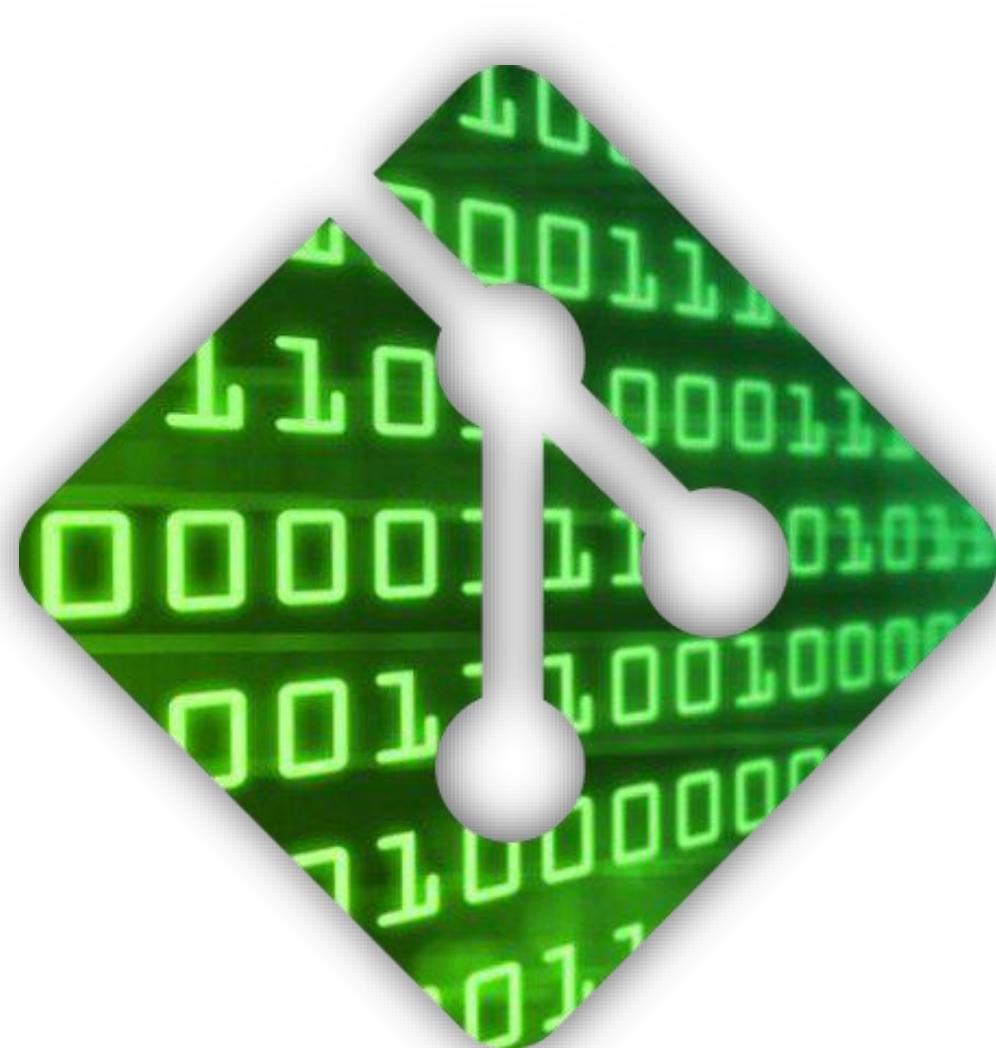
Git Branch



Outline

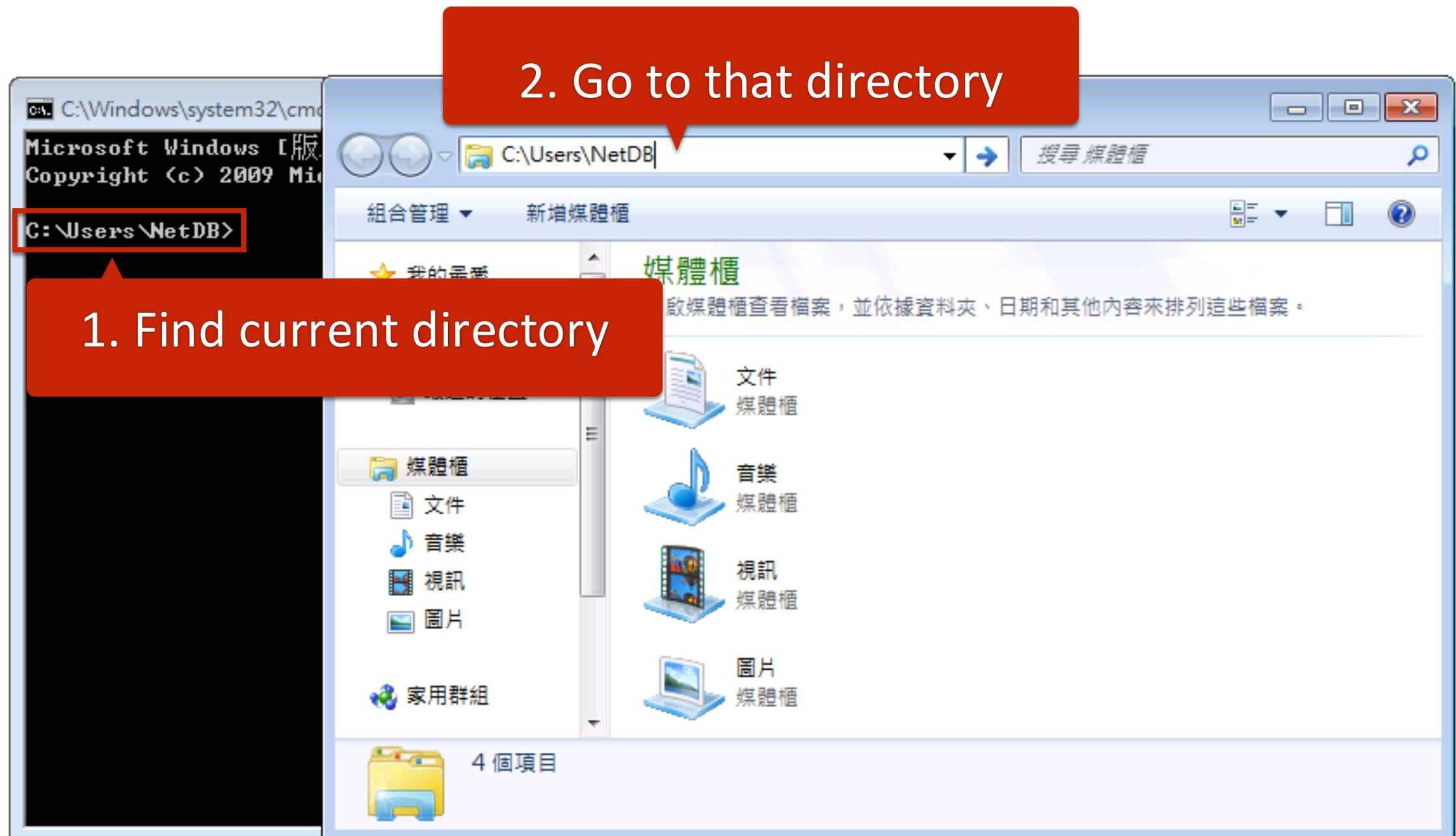
- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Be Professional

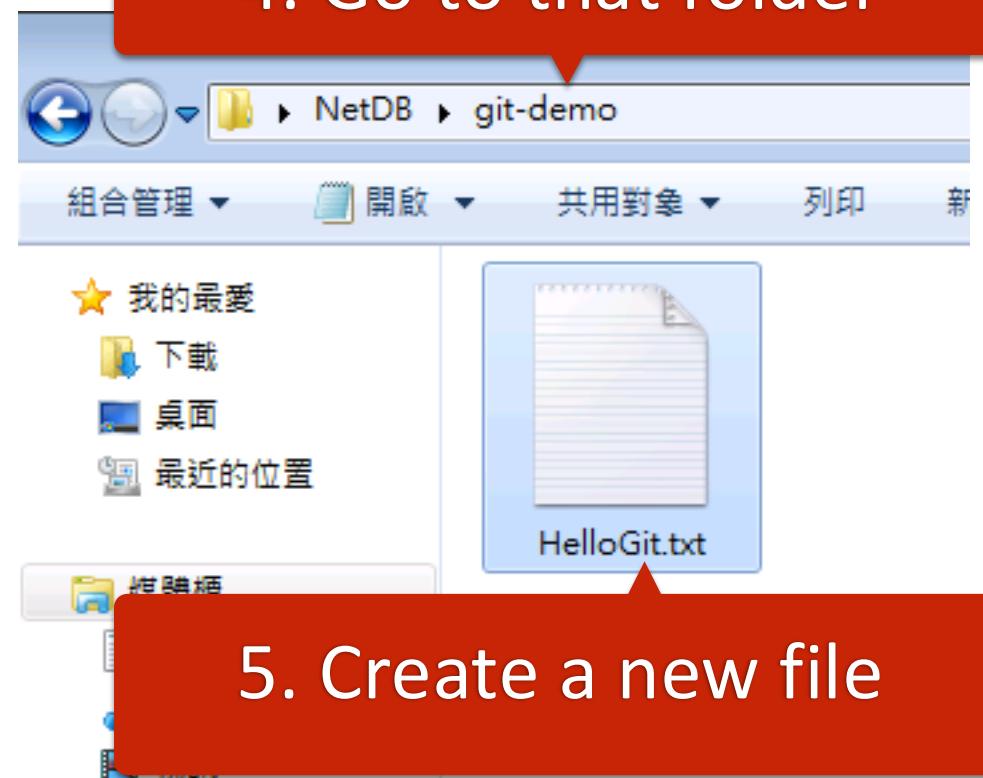


Basic Git Commands

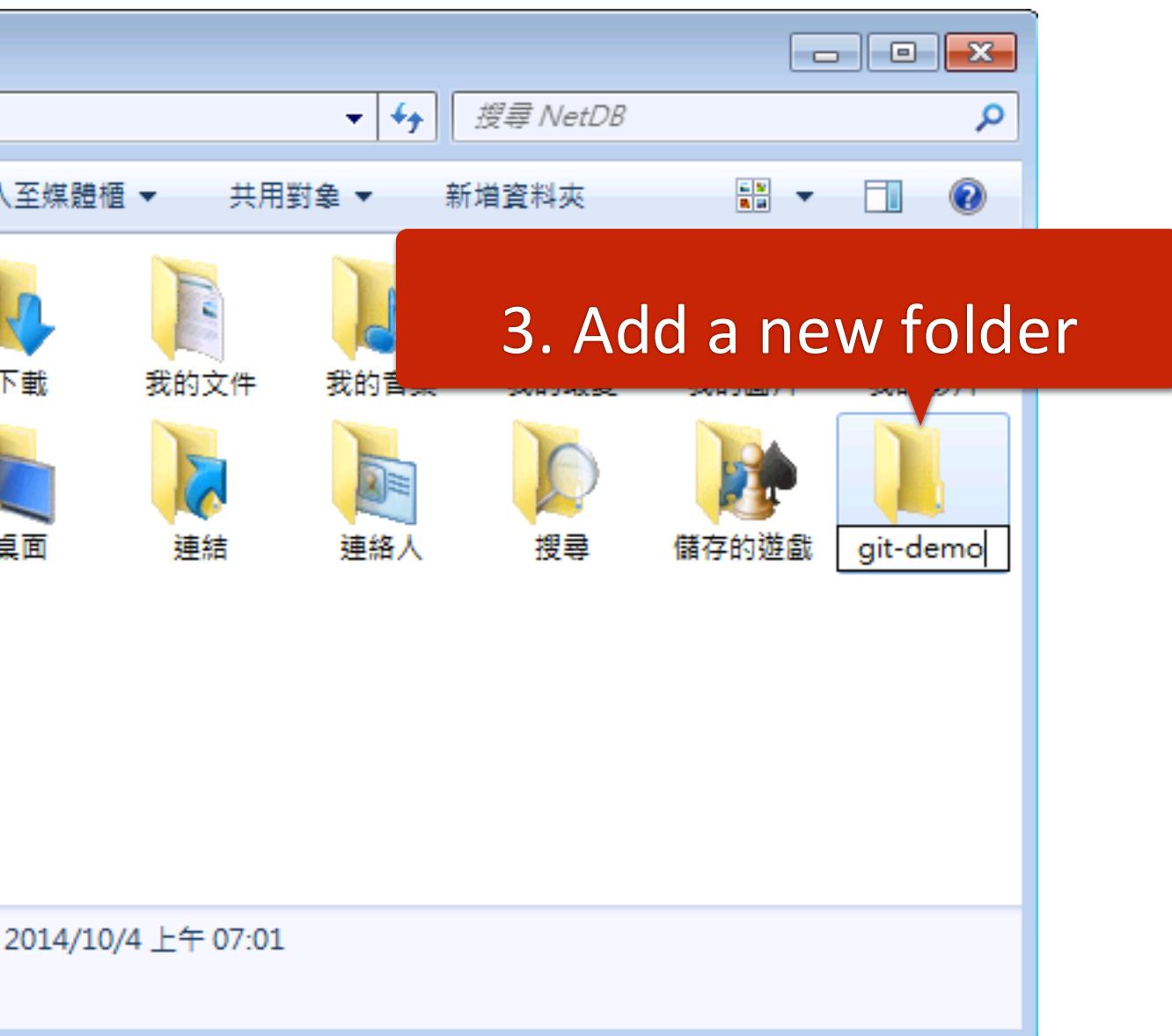
- **git init**
 - Initialize a repository at current directory.
- **git add [file_name]**
 - Add files to git repository and let git track them.
- **git commit -m "commit messages"**
 - Save the changes to the git repository and create snapshots of the files.
- **git checkout [version]**
 - Go to a specific version.



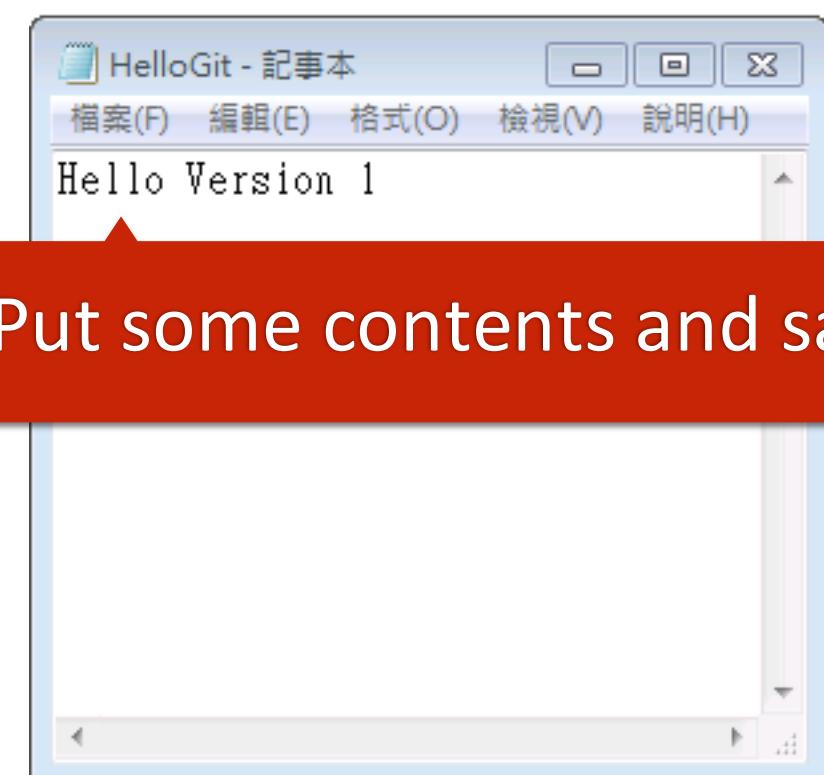
4. Go to that folder



3. Add a new folder



6. Put some contents and save



```
命令提示字元  
Microsoft Windows [版本 6.1.7601]  
Copyright (c) 2009 Microsoft Corporation. All rights reserved.  
  
C:\Users\NetDB>git config --global user.name "cyhsu"  
C:\Users\NetDB>git config --global user.email "cyhsu@netdb.cs.nthu.edu.tw"  
C:\Users\NetDB>
```

7. Setup user information

With --global: for all repositories in computer
Without --global: for current repository

```
$ git config --global user.name "name"  
$ git config --global user.email "email"
```

命令提示字元
Microsoft Windows [版本 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\> 'cyhsu'
C:\> "cyhsu@netdb.cs.nthu.edu.tw"

8. Go to “git-demo”

C:\> cd git-demo

C:\Users\NetDB\git-demo>dir
磁碟區 C 中的磁碟是 WIN7
磁碟區序號: 187B-C5C9

9. Show the files in “git-demo”

C:\Users\NetDB\git-demo 的目錄
2014/10/04 上午 02:12 <DIR>

..
15 HelloGit.txt
15 位元組
5,944 位元組可用

10. Initialize a Git repository

C:\Users\NetDB\git-demo>git init
Initialized empty Git repository in C:/Users/NetDB/git-demo/.git/
C:\Users\NetDB\git-demo>

```
$ cd git-demo # go to git-demo directory  
$ dir # list the files  
$ git init # initialize a repository
```

```
命令提示字元
C:\Users\NetDB>cd git-demo

C:\Users\NetDB\git-demo>dir
 磁碟區 C 中的磁碟是 WIN7
 磁碟區序號： 187B-C5C9

C:\Users\NetDB\git-demo 的目錄

2014/10/04 上午 07:17 <DIR> .
2014/10/04 上午 07:17 <DIR> ..
2014/10/04 上午 07:16 15 HelloGit.txt
                         1 個檔案          15 位元組

C:\Users\NetDB\git-demo>git add HelloGit.txt
C:\Users\NetDB\git-demo>git commit -m "version 1"
[master (root-commit) b302d9c] version 1
 1 file changed, 1 insertion(+)
  create mode 100644 HelloGit.txt
C:\Users\NetDB\git-demo>
```

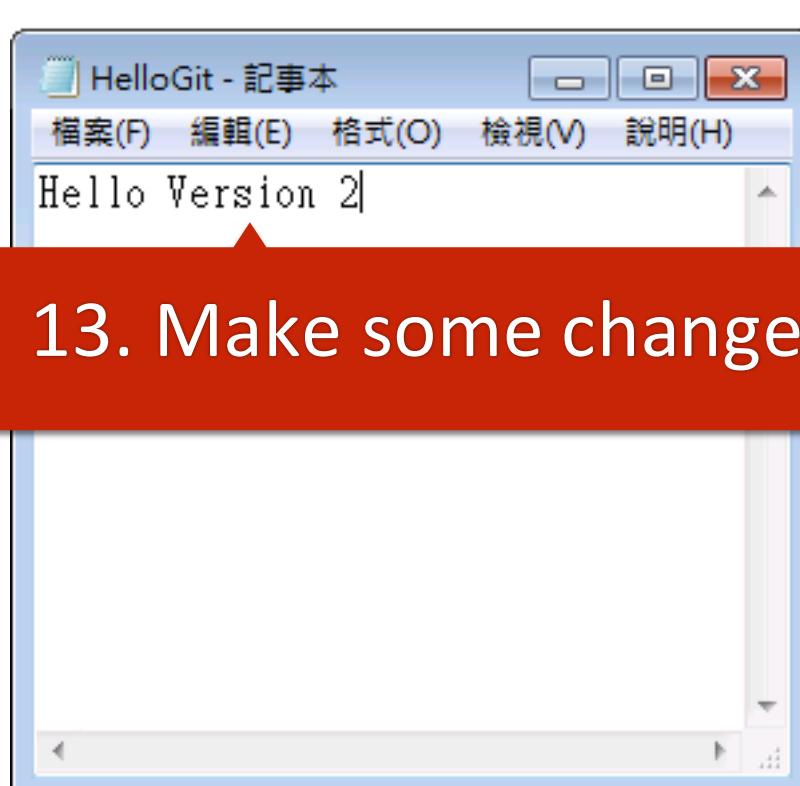
11. Add HelloGit.txt to staging files

12. Commit your changes

```
# Add HelloGit.txt to staging files
$ git add HelloGit.txt
```

```
# Commit the changes to the repository
# where "version 1" is the commit message
$ git commit -m "version 1"
```

14. Add it and commit again



13. Make some changes and save

```
C:\Users\NetDB\git-demo>git add HelloGit.txt  
C:\Users\NetDB\git-demo>git commit -m "version 2"  
[master e134c84] version 2  
 1 file changed, 1 insertion(+), 1 deletion(-)  
git-demo>
```

A screenshot of a terminal window. The command 'git add HelloGit.txt' is entered and highlighted with a red box. The command 'git commit -m "version 2"' is entered and highlighted with a red box. The output shows the commit was successful, adding one file and changing one line. The prompt 'git-demo>' is at the bottom.

15. View your versions

version ID

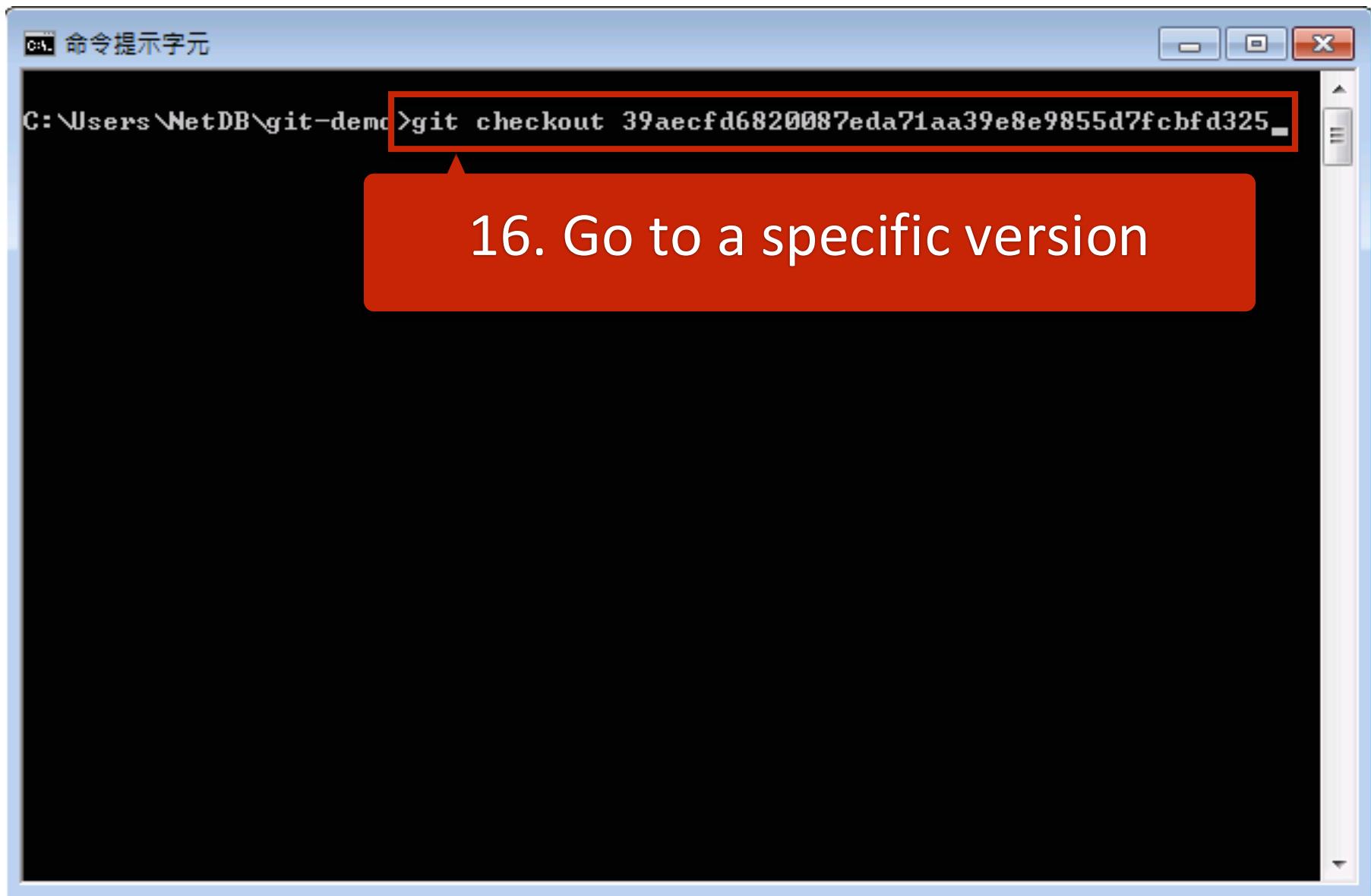
```
C:\Users\NetDB\git-demo>git log
commit e134c845df593f1451c4e9e6c874ddef6df42a76
Author: cyhsu <cyhsu@netdb.cs.nthu.edu.tw>
Date:   Sat Oct 4 08:09:55 2014 +0800

    version 2

commit 39aecfd6820087eda71aa39e8e9855d7fcfd325
Author: cyhsu <cyhsu@netdb.cs.nthu.edu.tw>
Date:   Sat Oct 4 08:09:16 2014 +0800

    version 1
```

```
# Show the versions you've created so far
$ git log
```



```
# Go to a specific version  
$ git checkout {version_id}
```

**LIFE IS
TOO SHORT
TO TYPE THAT
VERSION ID!**

which is 40 characters long...

version ID

A screenshot of a Windows Command Prompt window titled "命令提示字元". The command entered is "git log --oneline". The output shows two commits:

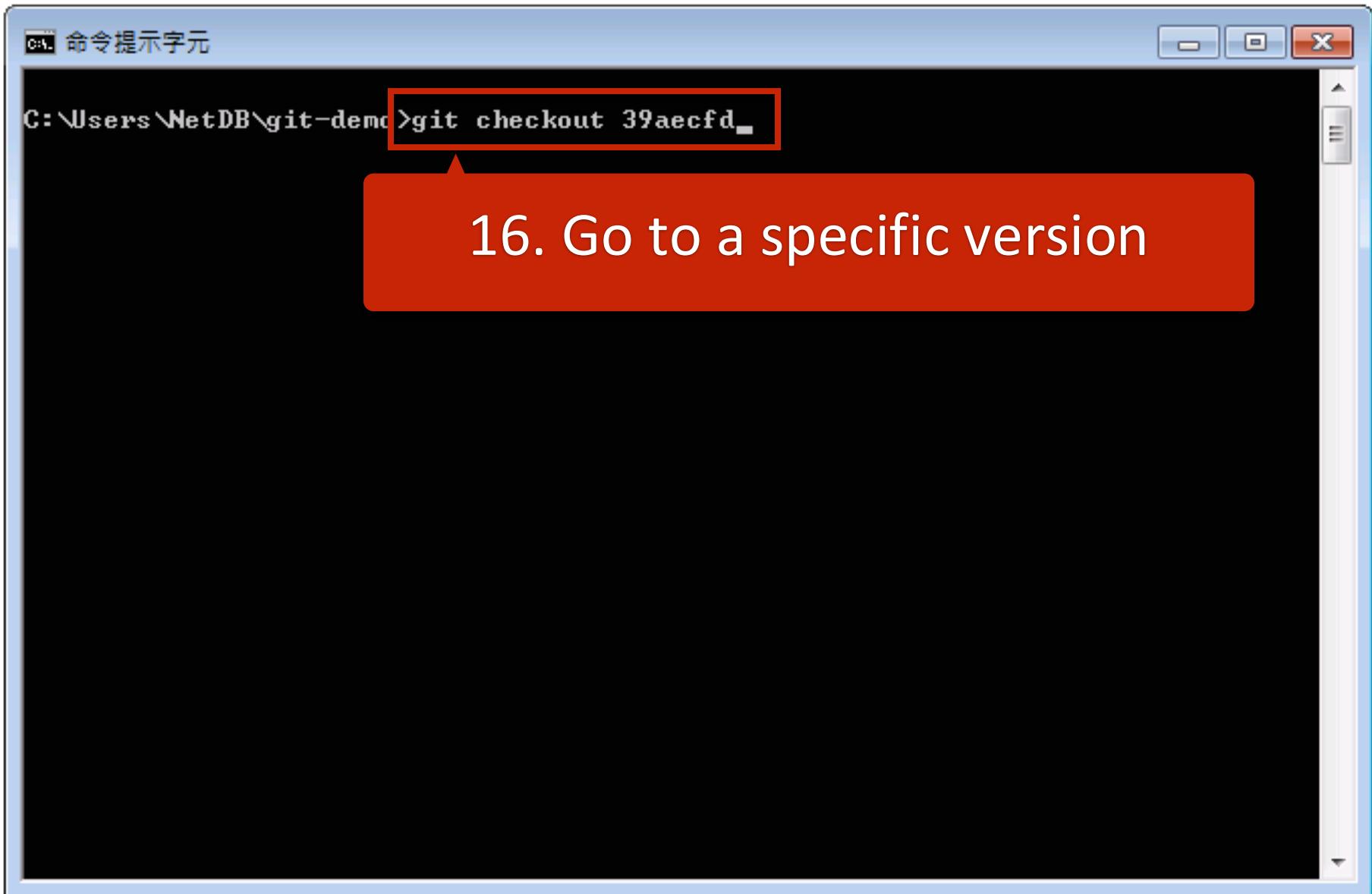
```
C:\Users\NetDB\git-demo>git log --oneline
e134c84 version 2
39aecfd version 1
```

The output is highlighted with a red box around the command line and the commit IDs. A blue arrow points from the text "version ID" to the commit IDs. A red callout box contains the text "15. Show versions with short version ID". Another red callout box contains the text "56% shorter!" with an arrow pointing to the commit IDs.

15. Show versions with short version ID

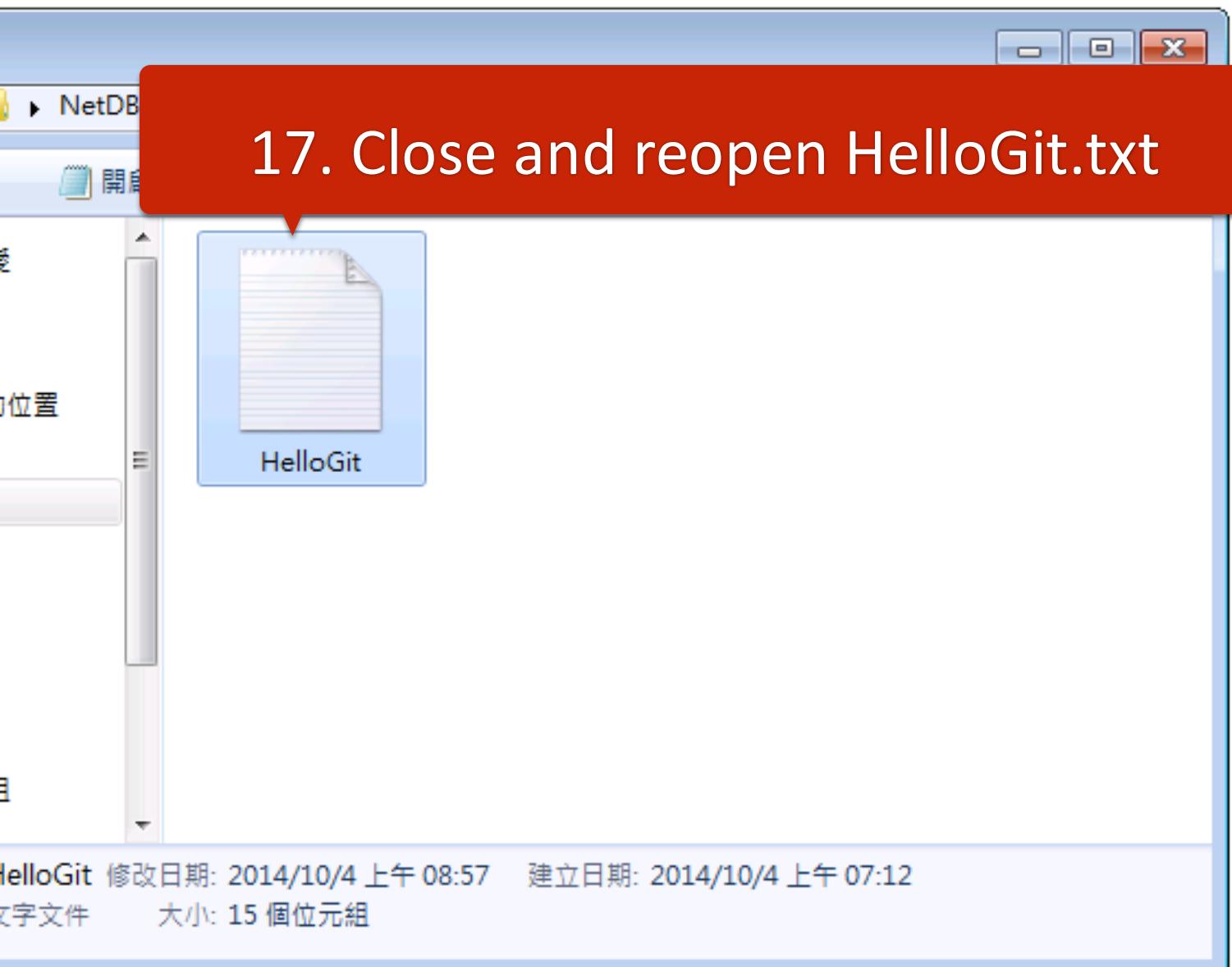
56% shorter!

```
# Show versions with short version id
$ git log --oneline
```

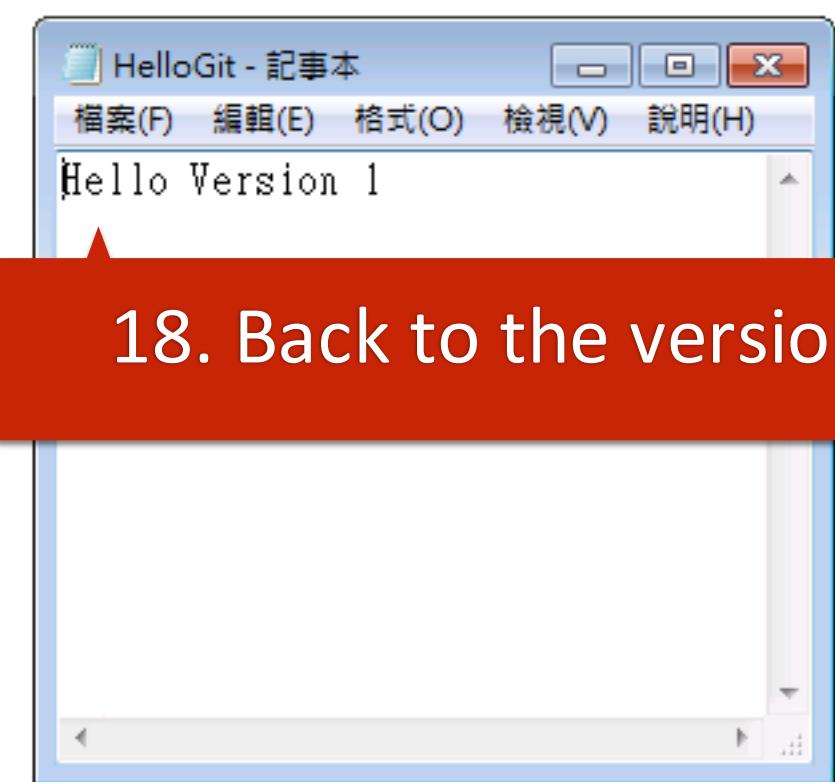


```
# Go to a specific version.  
# In fact, you only need to type  
# the first 5 characters.  
$ git checkout {short_version_id}
```

17. Close and reopen HelloGit.txt



18. Back to the version 1!



Try yourself (1/2)

- Branching steps
 - Creating a new branch

`git branch [branch name]`

- Checking out the branch

`git checkout [branch name]`

Try yourself (2/2)

- Merging steps
 - Checking out a branch to merge

`git checkout [branch 1 name]`

- Merging another branch

`git merge [branch 2 name]`

Outline

- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

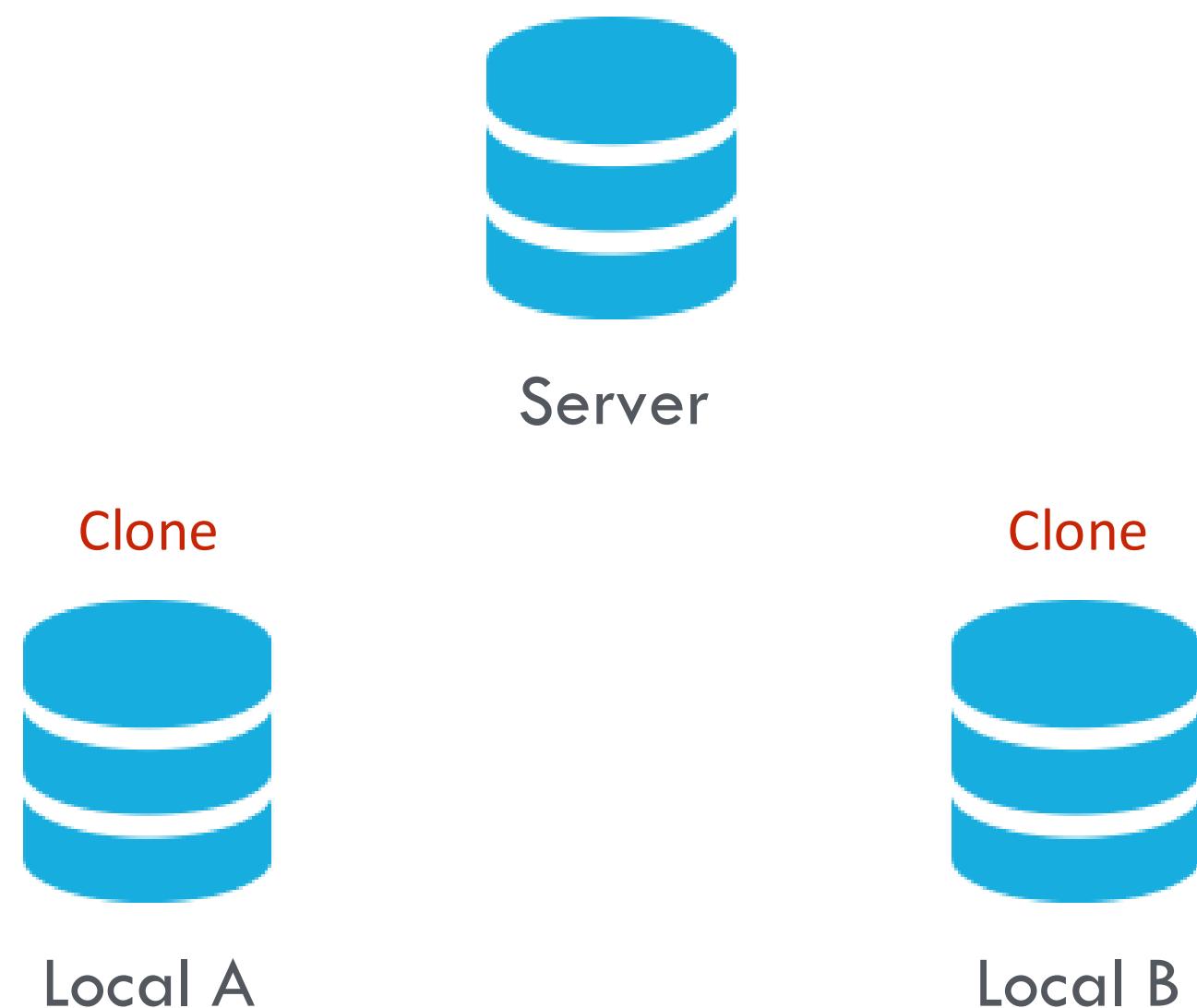
Collaboration

- To work with others using git, you'll need a server that store the repository.
- Git is distributed, which means
 - Everyone can store a copy of the repository downloaded from the server to their computer and do their jobs independently.

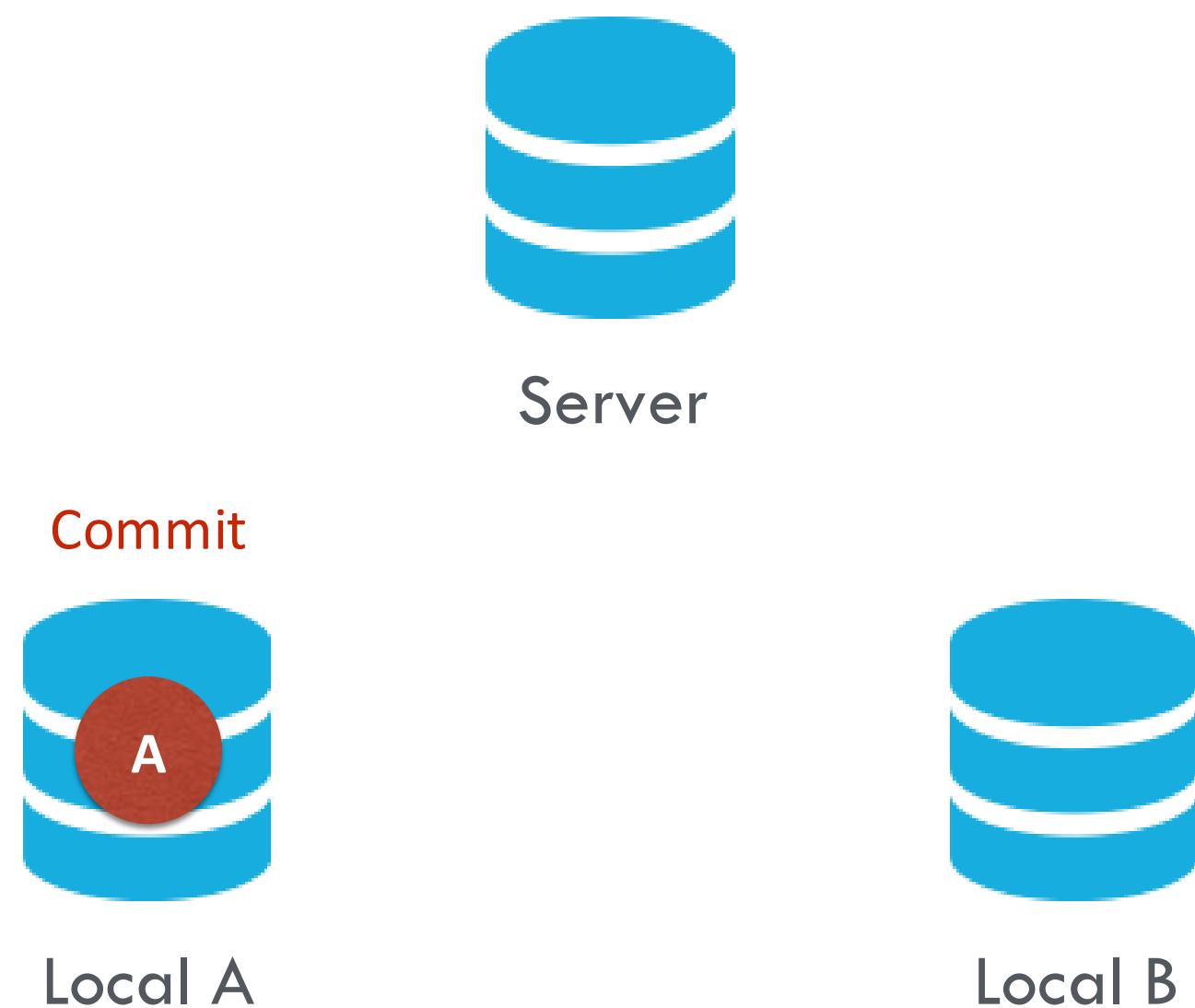
Collaboration Workflows (1/2)

1. If you don't have the project, *clone* (download) the repository from the server.
2. Do your work and commit the changes at local. Once done, *push* (upload) the repository to the server.
3. If someone else modified the project, you can *pull* (sync) the repository to get the updated project.
4. Repeat 2 and 3.

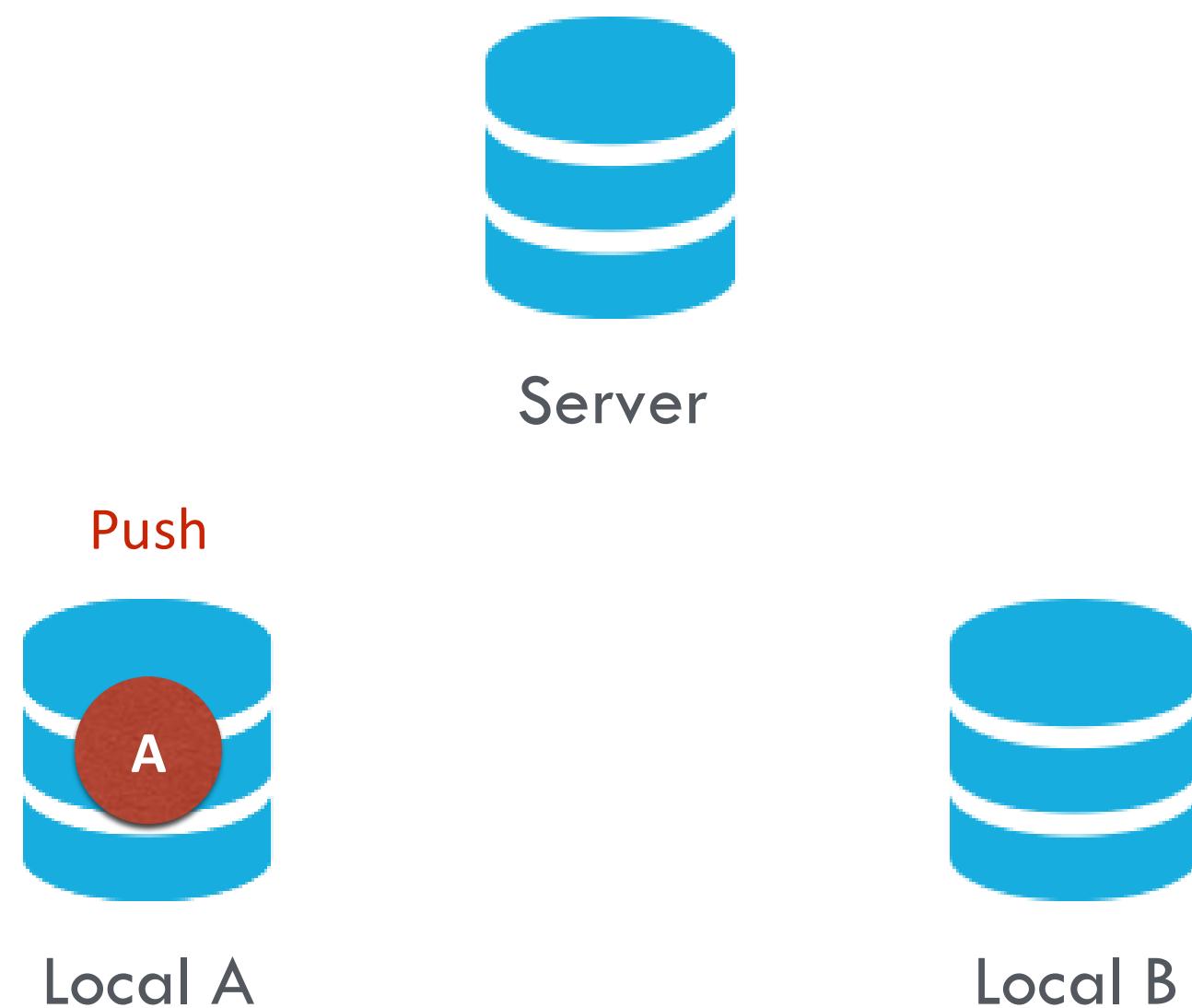
Collaboration Workflows (2/2)



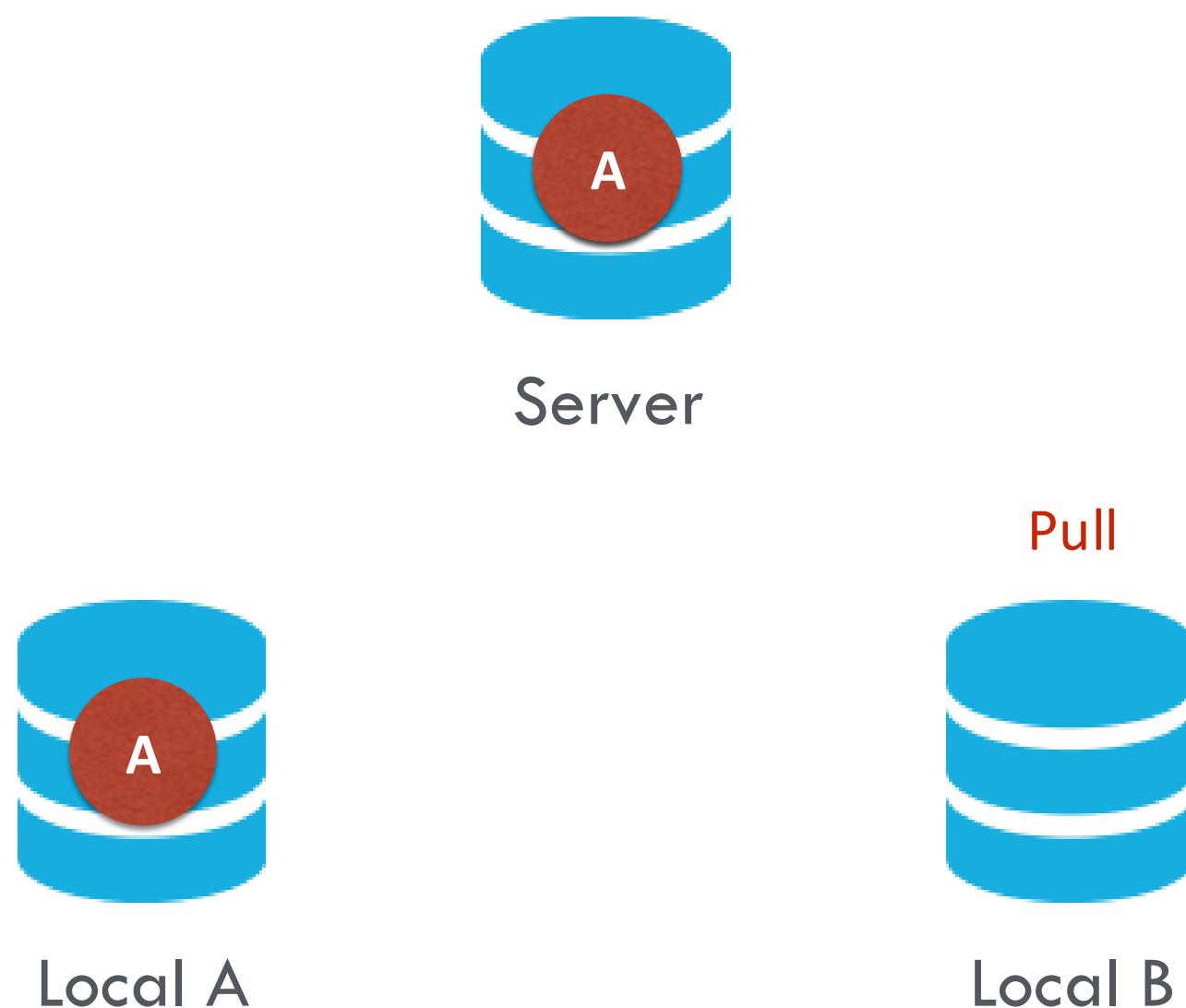
Collaboration Workflows (2/2)



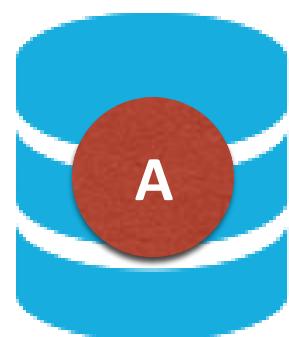
Collaboration Workflows (2/2)



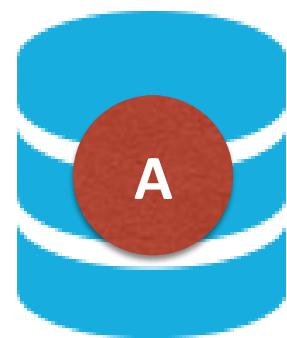
Collaboration Workflows (2/2)



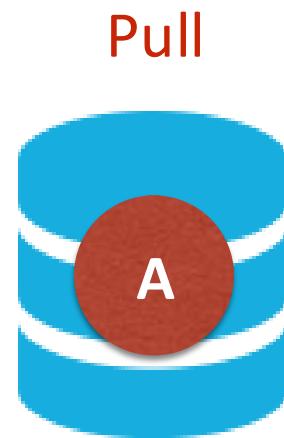
Collaboration Workflows (2/2)



Server



Local A



Local B

Why Authentication Failed?

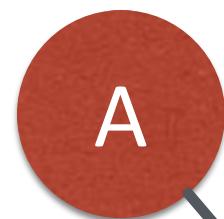
Collaboration Workflow

- If you tried to clone the code template from a server and want to push the modified file.
 - You will get authentication failed.
 - It's because it was a project of others, which means you are not able to save the changes back to the server.
- So, how can I copy a project from others on a open source platform like Github?

Introducing
Fork



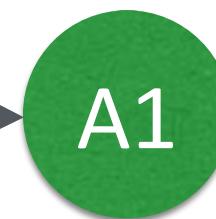
Original Project



Forked Project



Forked from Red

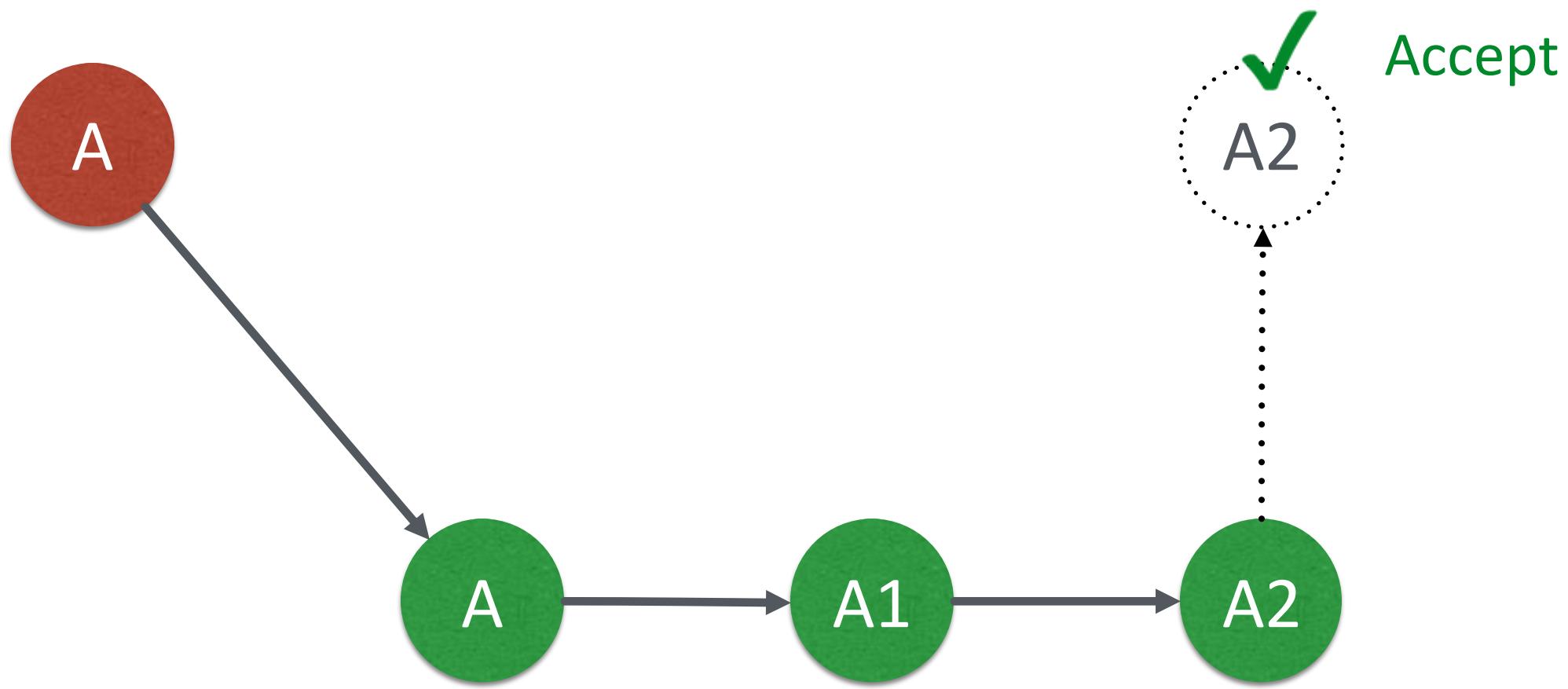


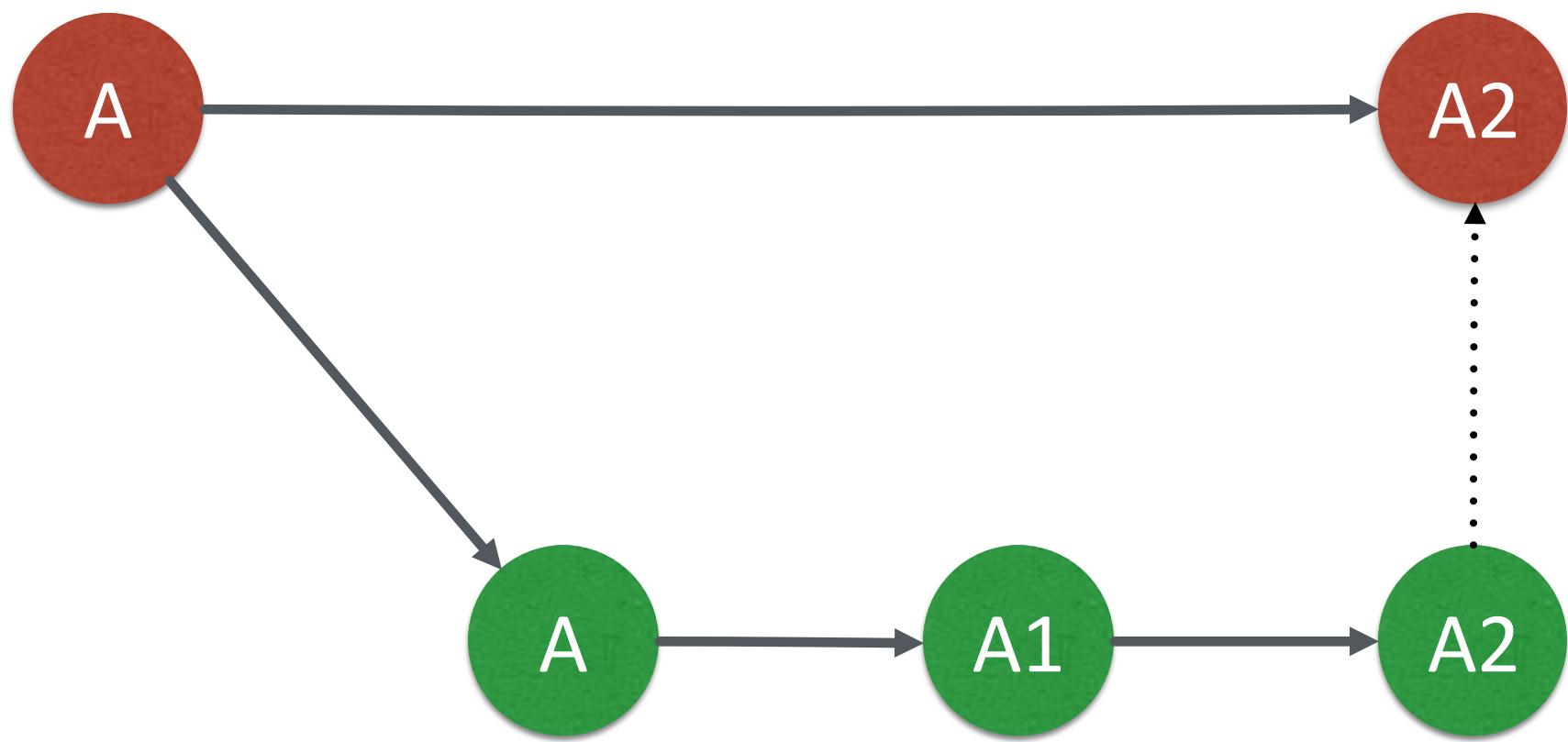
Commit

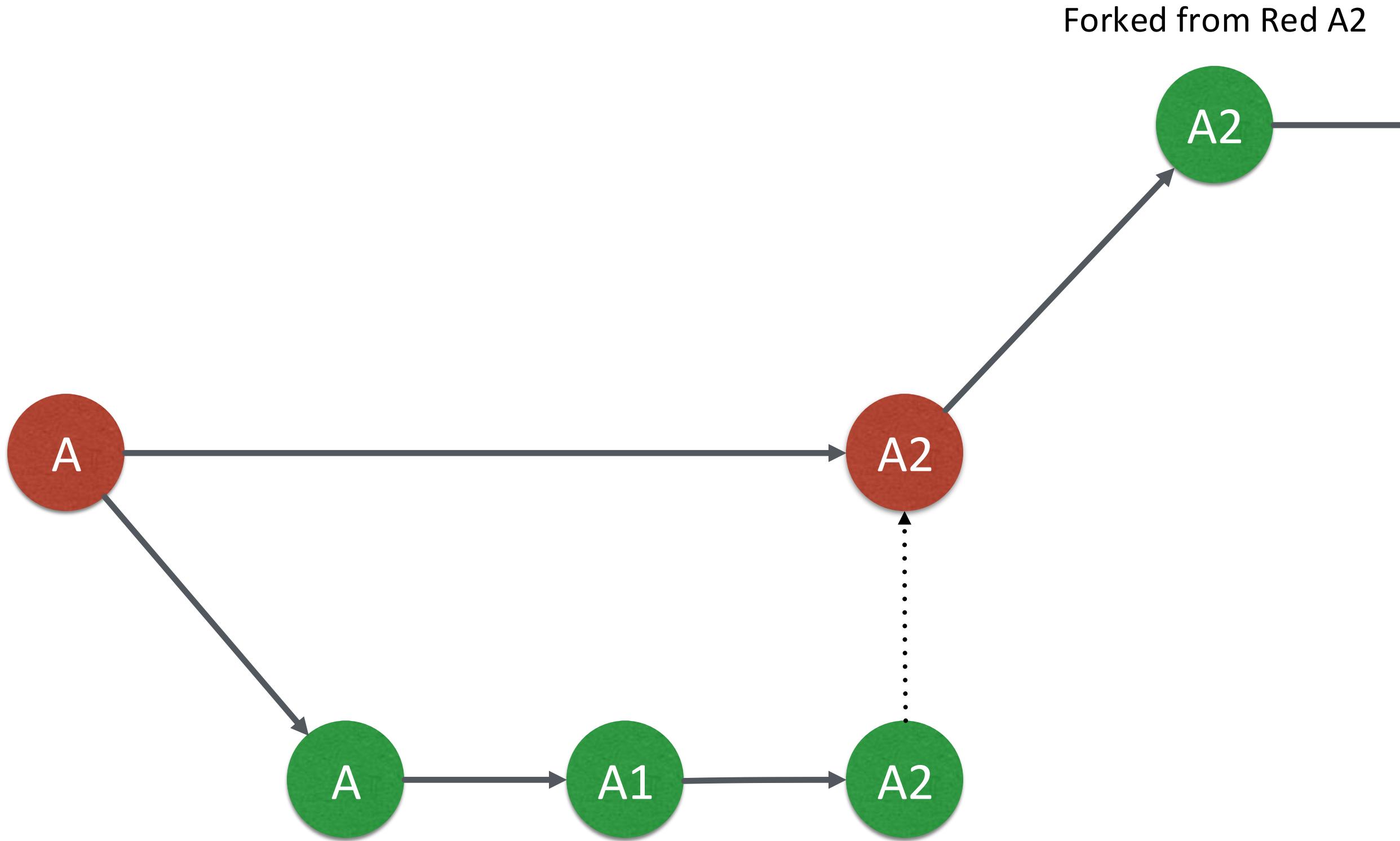


Commit

Open a Merge Request







Git Collaboration Workflow

1. *Fork* a repository to make a copy of it.
2. *Clone* the repository you forked to your workspace.
3. Do your work and *commit* the changes in your workspace.
4. *Push* the repository to the server to synchronize them.
5. Open a *merge request* to origin repository .

Basic Git Commands (2/2)

- **git clone [url]**
 - Clone a repository from remote server
- **git push [url] [branch-name]**
 - Push committed file to remote server

Outline

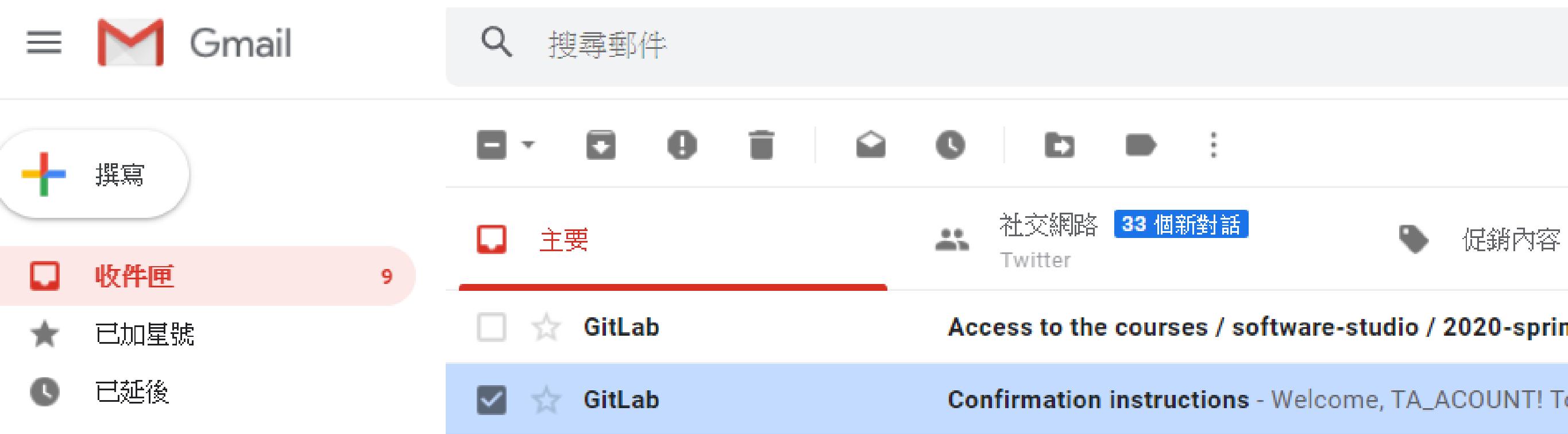
- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References



Gitlab

- We have created account for you
- Account: student ID (e.g. 106012345)
- Password: student ID (e.g. 106012345)

Gitlab



Gitlab

Confirmation instructions 收件匣 ×



GitLab <gitlab@shwu10.cs.nthu.edu.tw>

寄給我 ▾

下午 8:11 (2 分鐘前)



英文 ▾ > 中文 (繁體) ▾ [翻譯郵件](#)

[關閉下列語言的翻譯功能：英文 ×](#)



Welcome, TA_ACCOUNT!

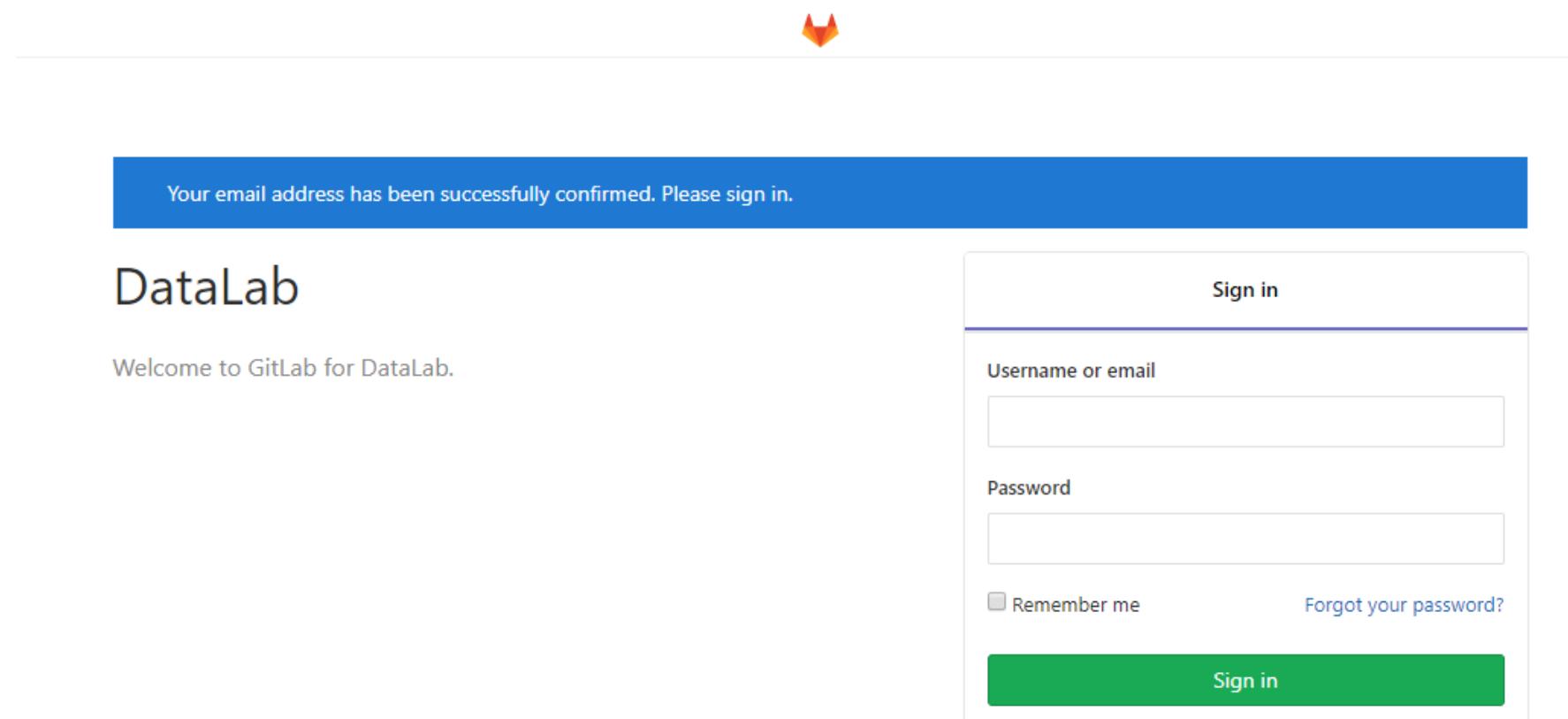
To get started, click the link below to confirm your account.

[Confirm your account](#)



You're receiving this email because of your account on shwu10.cs.nthu.edu.tw. [Manage all notifications](#) · [Help](#)

Gitlab



The image shows the GitLab sign-in page for a project named "DataLab". At the top, there is a blue header bar with the text "Your email address has been successfully confirmed. Please sign in." Below the header, the "DataLab" project name is displayed. To the right, there is a sign-in form with fields for "Username or email" and "Password", a "Remember me" checkbox, and a "Forgot your password?" link. A large green "Sign in" button is at the bottom of the form.

Your email address has been successfully confirmed. Please sign in.

DataLab

Welcome to GitLab for DataLab.

Sign in

Username or email

Password

Remember me [Forgot your password?](#)

Sign in

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

You can access course projects in this group

The screenshot shows the 'Details' page for the '2020-spring' group on GitLab. The left sidebar includes links for Overview, Details (which is selected), Activity, Issues (0), Merge Requests (0), Kubernetes, Members, and Settings. The main content area displays the group's name '2020-spring' with a React icon, its status as a 'Group', and a 'Leave group' button. It features a 'Subgroups and projects' tab (selected) and tabs for 'Shared projects' and 'Archived projects'. A search bar allows filtering by name and sorting by 'Last created'. Three projects are listed: 'lab-css-blog' (last created 6 hours ago), 'submission-exercise' (last created 6 hours ago), and 'hello-html-master' (last created 6 hours ago). The top navigation bar includes links for Projects, Groups, Activity, Milestones, Snippets, and various search/filter icons.

2020-spring

courses > ... > 2020-spring > Details

2020-spring

Group | Leave group

Subgroups and projects Shared projects Archived projects

Search by name Last created

lab-css-blog

submission-exercise

hello-html-master

Issues 0

Merge Requests 0

Kubernetes

Members

Settings

Collapse sidebar

GitLab Projects Groups Activity Milestones Snippets

alan0313 > submission-exercise > Details

The project was successfully forked.

submission-exercise Project ID: 9564

Add license 0 Commits 1 Branch 0 Tags 0 Bytes Files

Forked from courses / software-studio / 2020-spring / submission-exercise

master submission-exercise / + History Find file Web IDE

Add README.md alan0313 authored 6 hours ago 8bb12aa3

README Add CHANGELOG Add CONTRIBUTING Auto DevOps enabled

Name	Last commit	Last update
README.md	Add README.md	6 hours ago

README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting a lab or a project.

Try It !!

1. Fork this project.
2. Clone the project to your local machine.
3. Make changes to the code.
4. Commit your changes.
5. Push your changes to the remote repository.
6. Create a pull request.

1. Click to fork

GitLab Projects Groups Activity Milestones Snippets 🔍 🚧

alan0313 > submission-exercise > Details

2. Check if this repository is under your account

submission-exercise  Project ID: 9564

Add license 0 Commits 1 Branch 0 Tags 0 Bytes Files

Forked from courses / software-studio / 2020-spring / submission-exercise

master submission-exercise / + History Find file Web IDE

 Add README.md alan0313 authored 6 hours ago 8bb12aa3

 README  Add CHANGELOG  Add CONTRIBUTING  Auto DevOps enabled

Name	Last commit	Last update
README.md	Add README.md	6 hours ago

 README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting a lab or a project.

Try It !!

1. Fork this project.
2. Clone the project to your local machine.
3. Make changes to the code.
4. Commit your changes.
5. Push your changes to your forked repository.
6. Create a pull request to merge your changes into the main repository.

« Collapse sidebar

GitLab Projects Groups Activity Milestones Snippets

alan0313 > submission-exercise > General Settings

General project

Update your project name, description, avatar, and other general settings.

Permissions

Enable or disable certain project features and choose access levels.

Project visibility Private

The project is accessible only by members of the project. Access must be granted explicitly to each user.

Issues Only Project Members

Repository Only Project Members

Merge requests Only Project Members

Pipelines Enable feature to choose access level

Git Large File Storage

Manages large files such as audio, video, and graphics files

4. Set project to private

Expand Collapse

Project Repository Issues Merge Requests Wiki Snippets Settings General Members Integrations Repository

Collapse sidebar

GitLab Projects Groups Activity Milestones Snippets

Search or jump to...

submission-exercise

Project Repository Issues Merge Requests Wiki Snippets

0 0 0 0

Settings

General

Members Integrations Repository

Wiki
Pages for project documentation
Only Project Members

Snippets
Share code pastes with others out of Git repository
Only Project Members

Save changes

Merge request
Customize your merge request restrictions. Expand

Badges
Customize your project badges. [Learn more about badges.](#) Expand

Export project
Export this project with all its related data in order to move your project to a new GitLab instance. Once the export is finished, you can import the file from the "New Project" page.
General Expand

Advanced
Perform advanced options such as housekeeping, archiving, renaming, transferring, or removing your project. Expand

<< Collapse sidebar

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

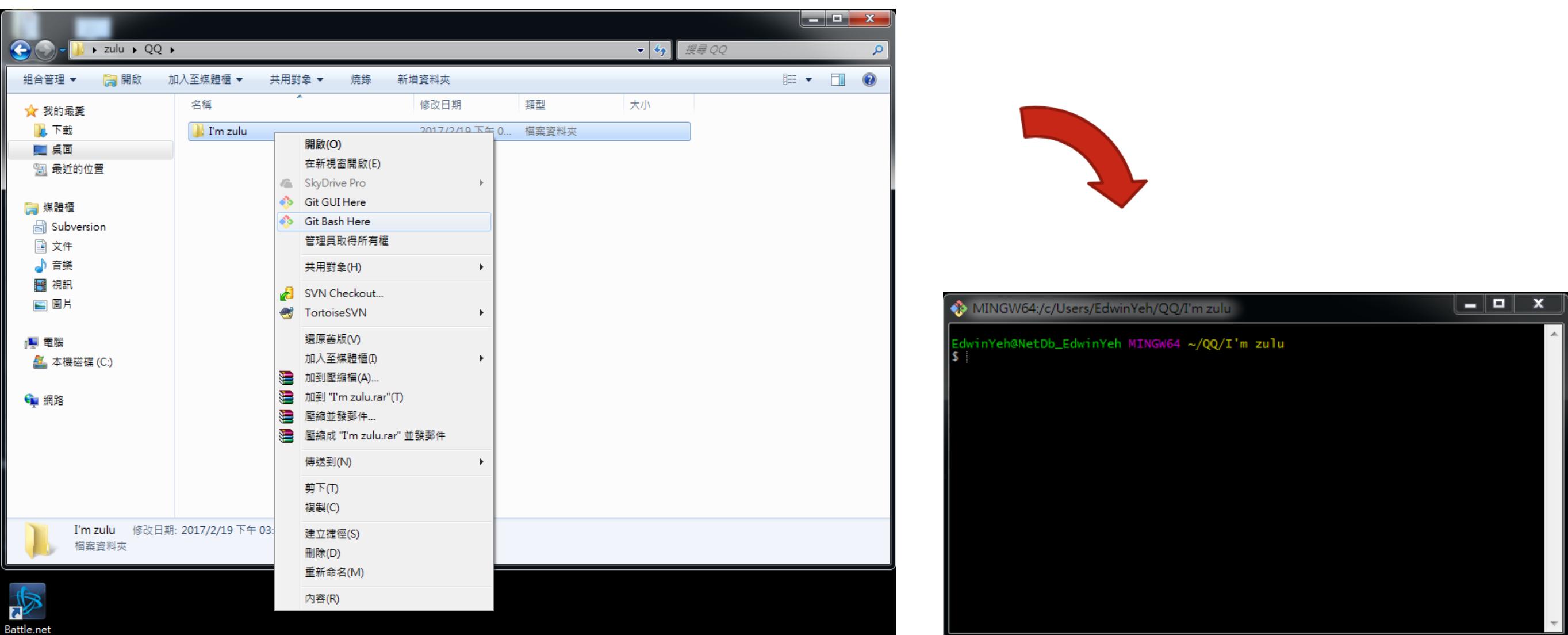
The screenshot shows a GitLab project named "submission-exercise" with Project ID 9564. The project has 0 Commits, 1 Branch, 0 Tags, and 0 Bytes Files. It was forked from "courses / software-studio / 2020-spring / submission-exercise". The repository contains a single commit by alan0313 titled "Add README.md" made 6 hours ago. The commit hash is 8bb12aa3. The project has Auto DevOps enabled.

1. Choose HTTPS

2. Copy the link

The "Clone with HTTPS" field contains the URL `https://shwu10.cs.nthu.edu.t...`. A red callout bubble points to this URL with the text "2. Copy the link". Another red callout bubble points to the "Clone with HTTPS" section with the text "1. Choose HTTPS".

If You use Windows



3. Create a folder to put your repos

```
~/SS-Projects ➔ git clone https://shwu10.cs.nthu.edu.tw/ss-student/submission-exercise.git  
Cloning into 'submission-exercise'...  
remote: Counting objects: 3, done.  
remote: Compressing objects: 100% (2/2), done.  
remote: Total 3 (delta 0), reused 3 (delta 0)  
Unpacking objects: 100% (3/3), done.  
~/SS-Projects ➔ ls  
submission-exercise  
~/SS-Projects ➔
```

4. Type "git clone {URL}"

5. The repo has been successfully cloned

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

```
~/SS-Projects/submission-exercise master vim lab1.js  
~/SS-Projects/submission-exercise master git add -A  
~/SS-Projects/submission-exercise master + 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:   lab1.js
```

1. -A means all files

```
~/SS-Projects/submission-exercise master + git commit -m "Finish lab1"  
[master c1acaf4] Finish lab1  
 1 file changed, 1 insertion(+)  
 create mode 100644 lab1.js  
~/SS-Projects/submission-exercise master
```

2. Check if your file is added to git

3. Commit your changes

```
~/SS-Projects/submission-exercise ➤ master ➤ vim lab1.html
~/SS-Projects/submission-exercise ➤ master ➤ git add -A
~/SS-Projects/submission-exercise ➤ master + ➤ git commit -m "Finish lab1"
[master 8a603d9] Finish lab1
Committer: Real Wei <realwei@Realweis-MBP.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

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 lab1.html
~/SS-Projects/submission-exercise ➤ master ➤
```

If you see these message, type
git config --global user.name "{name}"
git config --global user.email "{email}"

{email} is the email you use on gitlab

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

```
~/SS-Projects/submission-exercise ➜ master ➜ git push -u origin master  
Counting objects: 6, done.  
Delta compression using up to 4 threads.  
Compressing objects: 100% (4/4), done.  
Writing objects: 100% (6/6), 497 bytes | 0 bytes/s, done.  
Total 6 (delta 1), reused 0 (delta 0)  
To https://shwu10.cs.nthu.edu.tw/ss-student/submission-exercise.git  
  b1e0571..8a603d9 master -> master  
Branch master set up to track remote branch master from origin.  
~/SS-Projects/submission-exercise ➜ master ➜
```

Type "git push -u origin master"

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the **forked** repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

GitLab Projects Groups Activity Milestones Snippets 🔍 🛡

Search or jump to... 🔎

S submission-exercise

Project Details Activity Releases Cycle Analytics Repository Issues Merge Requests Wiki Snippets Settings

alan0313 > submission-exercise > Details

submission-exercise 🔒

Project ID: 9564

Add license 0 Commits 1 Branch 0 Tags 0 Bytes Files

Forked from courses / software-studio / 2020-spring / submission-exercise

master submission-exercise / + History Find file Web IDE

Add README.md alan0313 authored 6 hours ago 8bb12aa3

1. Click Merge Requests

README Add CHANGELOG Add CONTRIBUTING Auto DevOps enabled

Name	Last commit	Last update
README.md	Add README.md	6 hours ago

README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting a lab or a project.

Try It !!

1. Fork this project.
2. Clone the project from GitLab to your local environment.
3. Add a new file and write something.
4. Commit your work.

<< Collapse sidebar

GitLab Projects Groups Activity Milestones Snippets + This project Search

S submission-exercise

Project Repository Issues 0 Merge Requests 0 Wiki Snippets Settings

Merge requests are a place to propose changes you've made to a project and discuss those changes with others
Interested parties can even contribute by pushing commits if they want to.

New merge request

New merge request

2. New merge request

« Collapse sidebar

https://shwu10.cs.nthu.edu.tw/103060010/submission-exercise/merge_requests/new

The screenshot shows the GitLab interface for creating a new merge request. The left sidebar lists project management options like Projects, Groups, Activity, Milestones, Snippets, and a submission-exercise repository. The main area shows a 'New Merge Request' form.

3. Choose the branch you pushed in your repo

The 'Source branch' dropdown is highlighted with a red box. It shows 'alan/submission-exercise' selected with 'master' as the target branch. Below the dropdown is a commit history entry: 'Add README.md' by 'alan0313' 6 hours ago, with commit hash '8bb12aa3'. A green button at the bottom reads 'Compare branches and continue'.

4. Choose the branch named after your ID

The 'Target branch' dropdown is highlighted with a red box. It shows 'courses/software-studio/2020-spring...' selected with 'master' as the target branch. A modal window titled 'Select target branch' is open, containing a search bar 'Search branches' and a results list with 'master' checked.

5. Compare branches

A large red arrow points upwards from the 'Select target branch' modal towards the 'Compare branches and continue' button.

GitLab Projects Groups Activity Milestones Snippets + This project Search

江秉翰 > submission-exercise > Merge Requests

New Merge Request From 103060010/submit... To 103011228 Change branches

S submission-exercise

Project Repository Issues Merge Requests Wiki Snippets Settings

0 0 0 0

6. Set title to "{ID} Submission"

Title 103060010 Submission

Start the title with **WIP:** to prevent a **Work In Progress** merge request from being merged before it's ready.

Description Write Preview

Write a comment or drag your files here...

Markdown and quick actions are supported Attach a file

Source branch master

Target branch 103011228 Change branches

Squash commits when merge request is accepted. [About this feature](#)

<< Collapse sidebar

GitLab Projects Groups Activity Milestones Snippets + This project Search

S submission-exercise

Project Repository

Markdown and quick actions are supported

Attach a file

Issues 0

Merge Requests 0

Source branch master

Target branch 103011228 Change branches

Squash commits when merge request is accepted. [About this feature](#)

Snippets

Settings

Contribution Allow commits from members who can merge to the target branch. [About this feature](#)
Not available for private projects

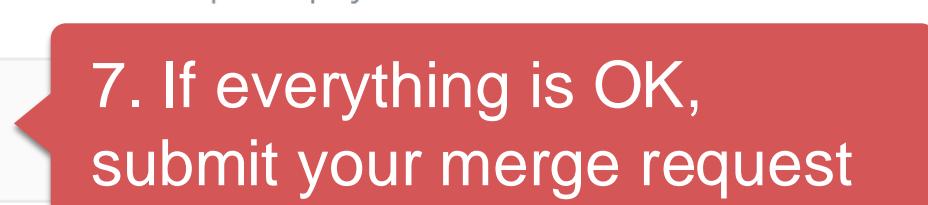
Submit merge request

Cancel

7. If everything is OK, submit your merge request

There are no commits yet.

<< Collapse sidebar



Notice



gitlab



Google 搜尋

好手氣



gitlab



全部

圖片

影片

新聞

書籍

更多

設定

工具

約有 8,300,000 項結果 (搜尋時間 : 0.45 秒)

GitLab

<https://gitlab.com/> ▾ 翻譯這個網頁

這項網站搜尋結果說明因為網站的 robots.txt 而無法提供
瞭解詳情

GitLab.com | GitLab

<https://about.gitlab.com/gitlab-com/> ▾ 翻譯這個網頁

GitLab.com. unlimited free repositories and collaborators. Sign Up. Free public & private repositories
and unlimited collaborators. Runs GitLab Enterprise Edition ...

GitLab介紹— Practical guide for git users 0.1 文档

<git-tutorial.readthedocs.io/zh/latest/gitlab.html> ▾

GitLab介紹¶. 目前最流行的線上Git專案管理系統可以說是非GitHub 莫屬，對於一般OpenSource的專案
選擇使用GitHub做為線上Git專案管理系統即可，也免收任何 ...

GitHub - gitlabhq/gitlabhq: GitLab CE | Please open new issues in our ...

<https://github.com/gitlabhq/gitlabhq> ▾ 翻譯這個網頁

README.md. GitLab. Build status CE coverage report Code Climate Core Infrastructure Initiative Best
Practices. Canonical source. The canonical source of ...

Gitlab - 維基百科，自由的百科全書 - Wikipedia

<https://zh.wikipedia.org/zh-tw/Gitlab> ▾

GitLab是一個利用Ruby on Rails開發的開源應用程式，實現一個自代管的Git專案倉庫，可通過Web介面
進行存取公開的或者私人專案。它擁有與GitHub類似的功能， ...



GitLab.com

GitLab.com offers free unlimited (private) repositories and unlimited collaborators.

- [Explore projects on GitLab.com](#) (no login needed)
- [More information about GitLab.com](#)
- [GitLab.com Support Forum](#)

By signing up for and by signing in to this service you accept our:

- [Privacy policy](#)
- [GitLab.com Terms](#).

[Sign in](#) [Register](#)

Username or email

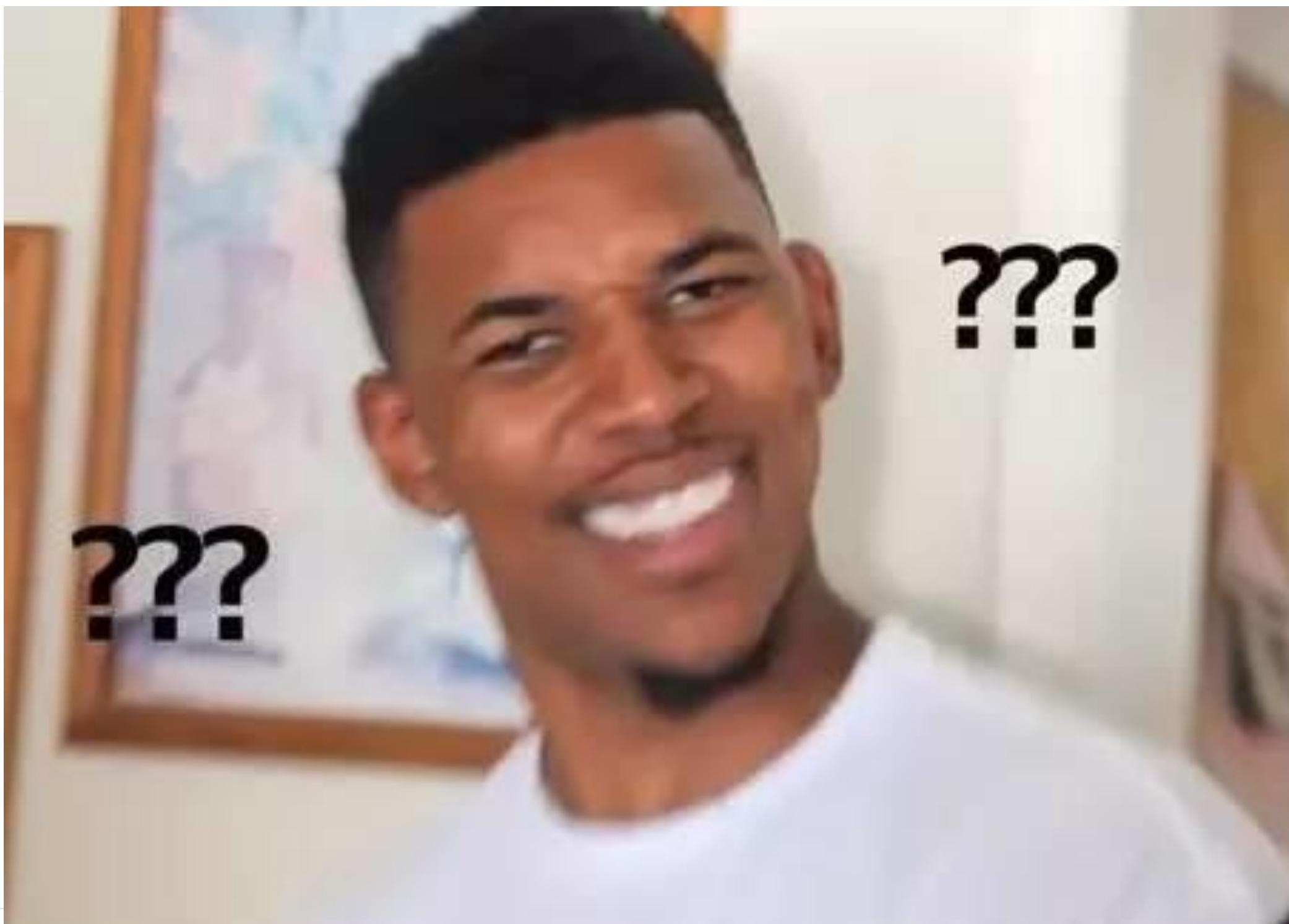
Password

Remember me [Forgot your password?](#)

[Sign in](#)

Didn't receive a confirmation email? [Request a new one.](#)

Sign in with



Resources

Here are some course materials and resources related to this course. For code and its details (such as assigned reading, project links, quiz, etc.) please refer to the GitLab. For online forum please refer to the iLMS system.



Here!!!!

Outline

- General Rule
- Introduction to Git
 - Version control
 - Git Basics
 - Try Git!
 - Remote Repositories
- How to Submit Your Code to Gitlab
- Tools & References

Tools

- Git GUI
 - GitKraken
- Editor / IDE
 - Visual Studio Code
 - Atom
 - Sublime Text
 - Brackets
 - Notepad++
 - Webstorm



axosoft

GitKraken

Repositories > GitKraken > master

Viewing 112/151 Show All

LOCAL (7/11)

- fancier-refbar-changes
- fancy-responsive-refbar-it... 42 ↗ 99+
- graph-color-test
- hopscotch 24 ↗ 99+
- init-repo-gitignore-typeahead
- invite-system 6 ↗ 99+
- jars-view-file-history
- master 5 ↗
- remote-panel-redesign 15 ↗ 13+
- settings-theme-styling
- view-file-history 24 ↗ 99+

REMOTE (6/41)

- Jeff-Schinella (0/1)
- Jordan-Wallet (0/7)
- Justin-GK (0/1)
- Ken-Price (0/2)
- Kyle-Smith (2/8)
- Max-Korp (0/2)
- Sjepan-Rajko (0/8)
- ayresa (0/3)
- cbargen (0/5)
- origin (4/4)

TAGS (99/99)

Fix un/stageall and stashing
Keep rename detection stage/unstage all
Bump to version 0.1.40
Merge pull request #597 from johnhaley81/fix-dispatc...
Merge pull request #594 from Mr-Wallet/nicer-ref-nam...
Fix `waitFor` bug in dispatcher
Merge pull request #596 from johndavidsparrow/gh-p...
Revised custom variable script and switch
Merge pull request #595 from johndavidsparrow/gh-p...
Resolved edge case where RefNodes could overlap
/universe removal of in-app invite wording
Bump to version 0.1.39
Merge pull request #591 from Mr-Wallet/fix-graph-ref...
Merge pull request #590 from srajko/div-be-gone
Merge pull request #588 from Mr-Wallet/friendlier-app...
Merge pull request #568 from Mr-Wallet/nicer-ref-nam...
Merge pull request #589 from johndavidsparrow/gh-p...
Merge pull request #592 from implausible/FixNSFW
JS tidy up in form-validation.js
Javascript update for /universe
/universe page
added maxwait to updateworkdir debounce
Update NSFW for memory leak
Fix NSFW segfault
Fix flickering GraphRefColumn every time ... 6 days ago
Preventing page reload on default pull click
Eliminate console spam when conflicts exist in a statel...
Upgrading to react-bootstrap v0.24.5

Commit: cca151e6b9e32c3f9209c25131706740050
Parent: 8efe30a11761983173f844900fa5ec5c6be2
Author: John Haley <johnh@axosoft.com>
Author Date: September 30th 2015, 2:54 pm

Bump to version 0.1.40

+ 0 added - 0 deleted ... 2 modified

```

npm-shrinkwrap.json
@@ -1,6 +1,6 @@
 1 | 1 | {
 2 | 2 |   "name": "gitkraken",
-3 | "version": "0.1.39",
+3 | "version": "0.1.40",
 4 | 4 |   "dependencies": {
 5 | 5 |     "atom-keymap": {
 6 | 6 |       "version": "5.1.11",

```

```

package.json
@@ -1,7 +1,7 @@
 1 | 1 | {
 2 | 2 |   "name": "gitkraken",
 3 | 3 |   "productName": "GitKraken",
-4 | "version": "0.1.39",
+4 | "version": "0.1.40",
 5 | 5 |   "description": "An intuitive git cli
 6 | 6 |   "main": "./src/appBootstrap/main.js"
 7 | 7 |

```

Provide Feedback



VS Code

The screenshot shows a code editor interface with a dark theme. On the left is a sidebar titled 'EXPLORE' containing a 'WORKING FILES' section with files like '03.jpg', 'jquery.dropdown.js', and 'navigation.js'. Below that is a 'TBL-STYLES' section with 'css', 'img', and 'js' sub-sections, each containing several files. The 'js' section includes 'hoverIntent.js', 'jquery.more.js', 'jquery.more.min.js', 'jquery.plugin.js', 'jquery.plugin.min.js', 'mapper.js', 'maputil.js', and 'smoothscroll.js'. The 'navigation.js' file is currently selected and highlighted in blue.

```
navigation.js js
1 var scriptbase = _spPageContextInfo.webServerRelativeUrl + "/_layouts/15/";
2
3 $(document).ready(function () {
4     $.getScript(scriptbase + "SP.Runtime.js", function () {
5         $.getScript(scriptbase + "SP.js", function () {
6             $.getScript(scriptbase + "SP.Taxonomy.js", function () {
7                 context = SP.ClientContext.get_current();
8                 //Call your code here.
9                 console.log("Navigation - ready to rock.");
10
11                 // Get default termstore
12
13                 session = SP.Taxonomy.TaxonomySession.getTaxonomySession(context);
14                 termStore = session.getDefaultSiteCollectionTermStore();
15                 context.load(session);
16                 context.load(termStore);
17                 context.executeQueryAsync(
18                     function () {
19                         console.log('Got default term store');
20                     },
21                     function(sender, args) {
22                         console.log('Could not get default term store. ' + args.get_message());
23                     }
24                 );
25
26
27             });
28         });
29     });
30 });
31
32 var topnavbar;
33
34 topnavbar += '<div class="tbl-site-navigation">';
35 topnavbar += '    <ul class="dropdown">';
36 topnavbar += '        <li class=""><a href="#">The Brand Code - a</a></li>';
37 topnavbar += '        <li class="dropdown1">';
38 topnavbar += '            <ul class="sub_menu" style="visibility: hidden;">';
39 topnavbar += '                <li class="large">';
40 topnavbar += '                    <div class="dropdownbox">';
41 topnavbar += '                        <div class="dropdownbox-title">Welcome to the Brand Code</div>';
42 topnavbar += '                        <ol>';
43 topnavbar += '                            <li><a href="">The importance of Brand Building</a></li>';
44 topnavbar += '                            <li><a href="">Introduction to the Brand Code</a></li>';
45 topnavbar += '                            <li><a href="">You and the Brand Code</a></li>';
46 topnavbar += '                        </ol>';
47 topnavbar += '                    </div>';

```



ATOM

A hackable text editor
for the 21st Century

The screenshot shows the Atom code editor interface. On the left is a sidebar with a tree view of project files, including build, docs, dot-atom, exports, keymaps, menus, node_modules, resources, script, spec, src, and static. The main editor area has a dark background and displays the file atom.coffee. The code in the file is:

```
18
19 # Essential: Atom global for dealing with packages, themes, menus, and the window system.
20 #
21 # An instance of this class is always available as the 'atom' global.
22 module.exports =
23   class Atom extends Model
24     @version: 1 # Increment this when the serialization format changes
25
26     # Load or create the Atom environment in the given mode.
27     #
28     # Returns an Atom instance, fully initialized.
29     @loadOrCreate: (mode) ->
30       startTime = Date.now()
```

The status bar at the bottom of the editor shows the word "Settings".

Reference

- Learn Git branching (interactive)
 - <http://pcottle.github.io/learnGitBranching/>
- Pro Git
 - <http://git-scm.com/book/>
- 寫給大家的 Git 教學
 - <http://www.slideshare.net/littlebtc/git-5528339>

Today's exercise

- Install Git command line tool in your computer.
 - Follow appendix A.
 - Try to submit in GitLab.