
Github Tutorial

鍾起鳴: r09944021@ntu.edu.tw

林承緯: r09922078@ntu.edu.tw

Slide link



Setting up your environment

Github

- [Sign up](#) for your account with your school email
- If not using school email, you need to apply [education pack](#) for more functions
- If you successfully apply education pack you will see PRO in your profile

Highlights

✳ Arctic Code Vault Contributor

☆ PRO

- Note that this account will put all your source codes in the future, please be careful about everything you do on it.
 - For example, the username should be a decent one.

Environment

- Linux or MAC OS is strongly recommended.
- Windows
 - use git bash / bash / powershell.
 - Install linux (ubuntu is suggested) on virtual machine or dual system.
 - connect to CSIE workstation.
- connect to CSIE workstation
 - ssh [your school id]@linux[machine number].csie.org
 - e.g. ssh [b09902000@linux5.csie.org](ssh:b09902000@linux5.csie.org)
 - Windows: use mobaxterm (suggested), or install openssh.

Install Git on Linux (ubuntu)

- `sudo apt update`
- `sudo apt install git`

Install vim on Linux (ubuntu)

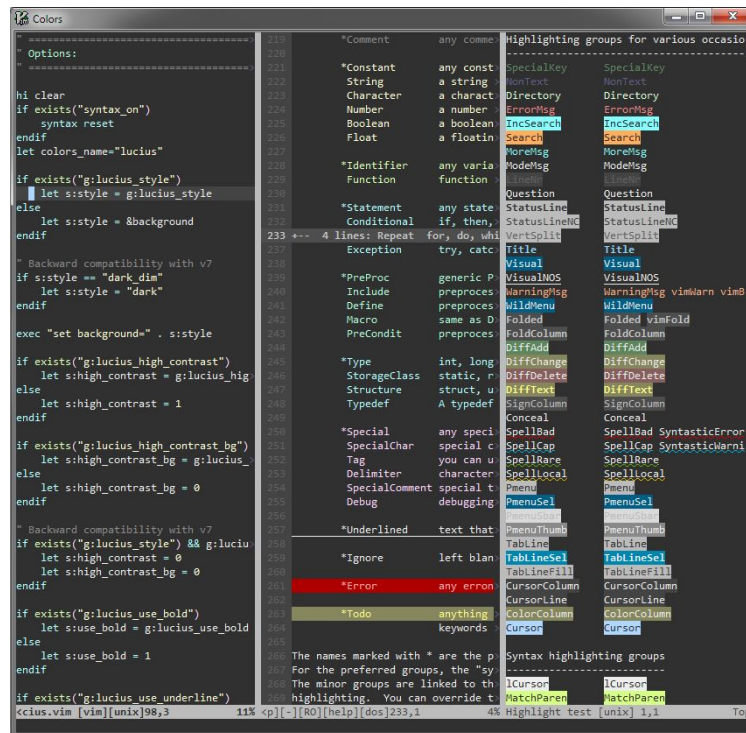
- `sudo apt update`
- `sudo apt install vim`

Suggest Editor -- Vim

- Common commands
 - "i": insert (enter editing mode)
 - "o": insert on the new line (enter editing mode)
 - "\esc": escape from editing mode
 - ":w": write to file
 - ":q": leave vim
 - ":wq": write and leave
 - "[number] yy": copy [number] lines
 - "p": paste copied lines on the current line
 - "[number] dd": delete [number] lines
 - "v + [→, ←]": underline messages
 - "u": undo (need to set up undo buffer in vimrc)
 - "\ctrl r": redo

Suggest Editor -- Vim

- Make vim more beautiful
 - [Tutorial](#)
 - [Color Sceme](#)
 - some color sceme [lucius dark](#)



<https://camo.githubusercontent.com/11a2b8c9e2544000911b0f795e9c4e3175a9c757/687474703a2f2f692e696d6775722e636f6d2f65373069392e706e67>

CSIE resources

Printer

- Printing quota:
 - Each CSIE student has 500 dollars for printing every semester (can't be cumulated).
 - NTU also gives each student 100 dollars for printing every semester (can be cumulated).
- Therefore, you can print your HW without money.

CSIE printer introduction

- Log in <https://printing.csie.ntu.edu.tw/user> .
Use your csie workstation account and password. You should received an email containing your workstation account and password after the enrollment.

PaperCut NG

摘要

費率

轉帳

交易歷史記錄

近期的列印工作

工作待決釋放

網頁列印

網頁列印

Web Print is a service to enable printing for laptop, wireless and unauthenticated users without the need to install print drivers. To upload a document for printing, click Submit a Job below.

提交工作

提交時間	印表機	文件名稱	頁數	費用	狀態
沒有活躍的工作					

摘要

費率

轉帳

交易歷史記錄

近期的列印工作

工作待決釋放

網頁列印

網頁列印

1. 印表機 2. 選項 3. 上傳

選擇一台印表機：

印表機名稱 ↑	位置/部門
<input type="radio"/> printing2\Ariel	next to 209 (two-sided)
<input type="radio"/> printing2\Elsa	next to 209 (one-sided)

« 返回活躍的工作

2. 列印選項及帳戶選擇 »

The printer is next to 209.

Workstation

- csie workstation:
ssh [your school id]@linux[machine number].csie.org

```
(base) marvinchung@506B01R09944021:~$  
(base) marvinchung@506B01R09944021:~$ ssh r09944021@linux9.csie.ntu.edu.tw  
The authenticity of host 'linux9.csie.ntu.edu.tw (140.112.30.40)' can't be established.  
ECDSA key fingerprint is SHA256:VWGyQzYLkAXXOGuch+L19cxfFaidPioxSLTEjqtIEFc.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added 'linux9.csie.ntu.edu.tw,140.112.30.40' (ECDSA) to the list of known hosts.  
r09944021@linux9.csie.ntu.edu.tw's password:  
Warning: your password will expire in 32765 days.  
Warning: your password will expire in 32765 days.  
  
#####  
#      Public Domain Workstation Lab (R217).      #  
#####  
#  UNIX Login Service:                           #  
#  FreeBSD - bsd1                                #  
#  Linux   - linux1, linux2, linux3, ... linux15  #  
#           - [NEW!] meow1, meow2 (with GPU)      #  
#           - oasis1, oasis2, oasis3 (non-computing) #  
#  Office open time:                             #  
#  08:30 ~ 17:00, otherwise please use accesscards #  
#  Contact information:                          #  
#  Web:    https://wslab.csie.ntu.edu.tw/        #  
#  E-Mail (linux): ta217@csie.ntu.edu.tw         #  
#  E-Mail (bsd) : lantw44@csie.ntu.edu.tw        #  
#  ##### Last Update: Sep 26 2018 ##  
mail: /var/spool/mail/r09944021: No such entry, file or directory  
西元2020年11月21日 (週六) 16時55分55秒 CST  
r09944021@linux9 [-] █
```

- All tips and information are in <https://wslab.csie.ntu.edu.tw/>

Github intro

Why to use GitHub

- People use GitHub to build the most advanced technologies in the world
- There's a whole set of tools on GitHub that can help you
- You put codes on GitHub and show them to the world
- If you are to become a software engineer,

then GitHub is one of the basic tools you MUST learn.

Tutorial -- Create Repository

- Each of your project is put in a repository

Create a new repository

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

Repository template

Start your repository with a template repository's contents.

No template ▾

Owner *

 MarvinChung ▾

Repository name *

Hello-Github ✓

Great repository names are short and memorable. Need inspiration? How about [supreme-sniffle](#)?

Description (optional)

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

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

☒ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**

Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**

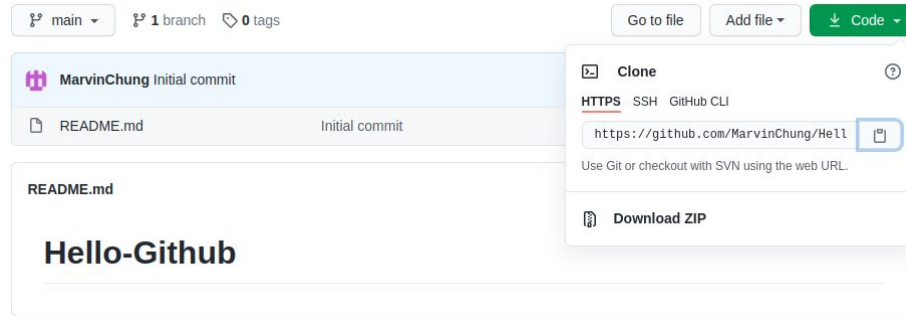
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set  `main` as the default branch. Change the default name in your [settings](#).

Create repository

Tutorial -- Clone Repository to Local

- Copy the repo url



- Open terminal, type “git clone [url]”

```
(base) marvinchung@506B01R09944021:~/Github$ git clone https://github.com/MarvinChung/Hello-Github.git
Cloning into 'Hello-Github'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 592 bytes | 592.00 KiB/s, done.
(base) marvinchung@506B01R09944021:~/Github$
```

Add my teammate

[requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Autolink references


Actions

Secrets

Moderation settings

Who has access


PUBLIC REPOSITORY



This repository is public and visible to anyone.


[Manage](#)

DIRECT ACCESS



0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

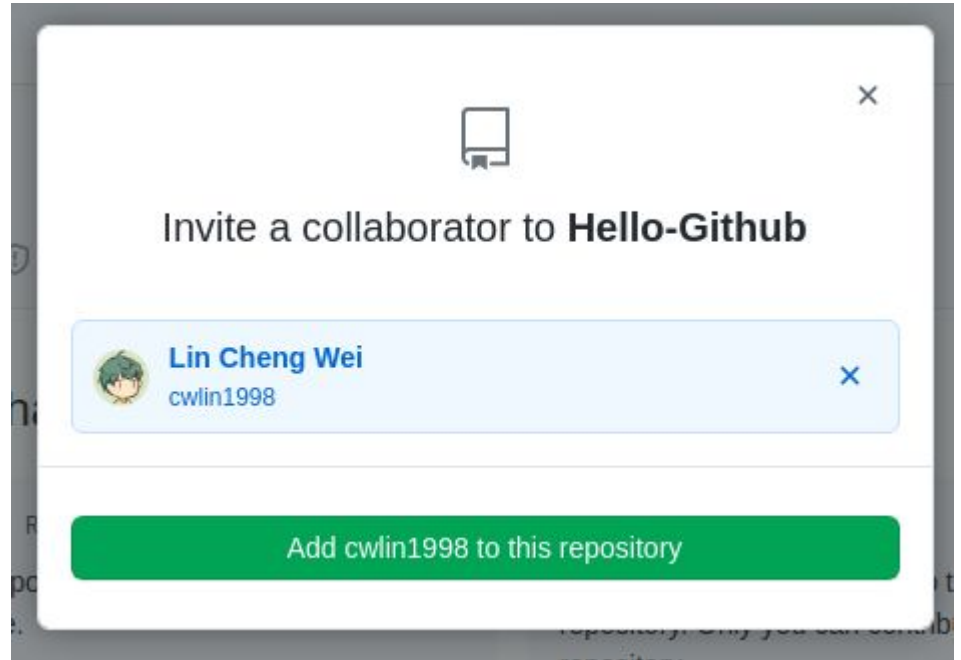


You haven't invited any collaborators yet

Invite a collaborator

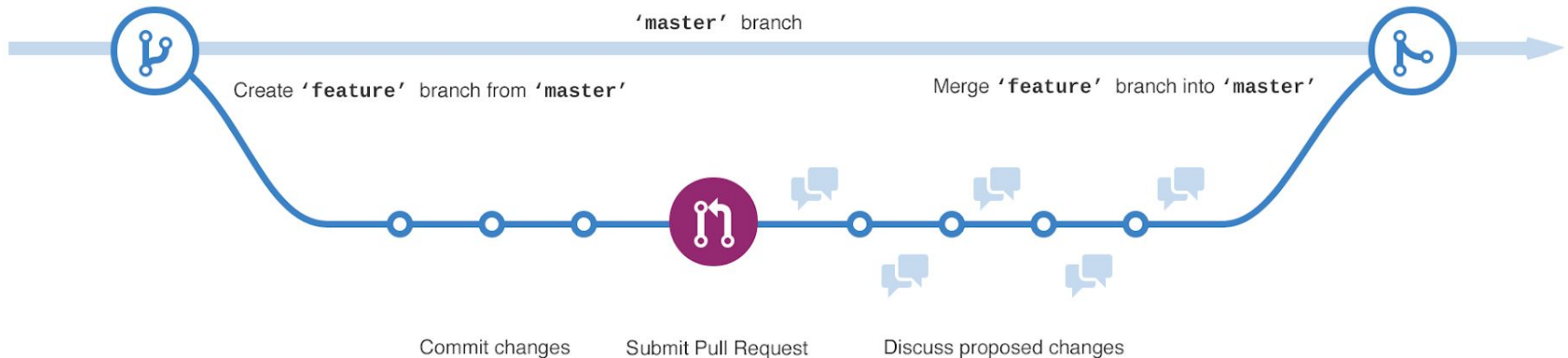
Invite my teammate

The teammate should received an email.



Tutorial -- Create a Branch

- Branching is the way to work on **different versions** at one time
- We use branches to experiment and make edits before committing them to master (main).
- master (main) branch should be clean.
- At branch master (main)
 - create branch: `git branch [branch name]`



Tutorial -- Create a Branch

- Example: branch to add TA sample codes (main is equivalent to master)

```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git branch dev
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git branch
dev
* main
```

- checkout to the dev branch

```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git checkout dev
Switched to branch 'dev'
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$
```

Tutorial -- Protect “main” Branch

- Important when developping big projects
 - Enhance code quality without creating artificial obstructions to effective collaboration
- Setting → Branches → add rule

Branch protection rule

Branch name pattern

Applies to 1 branch

main

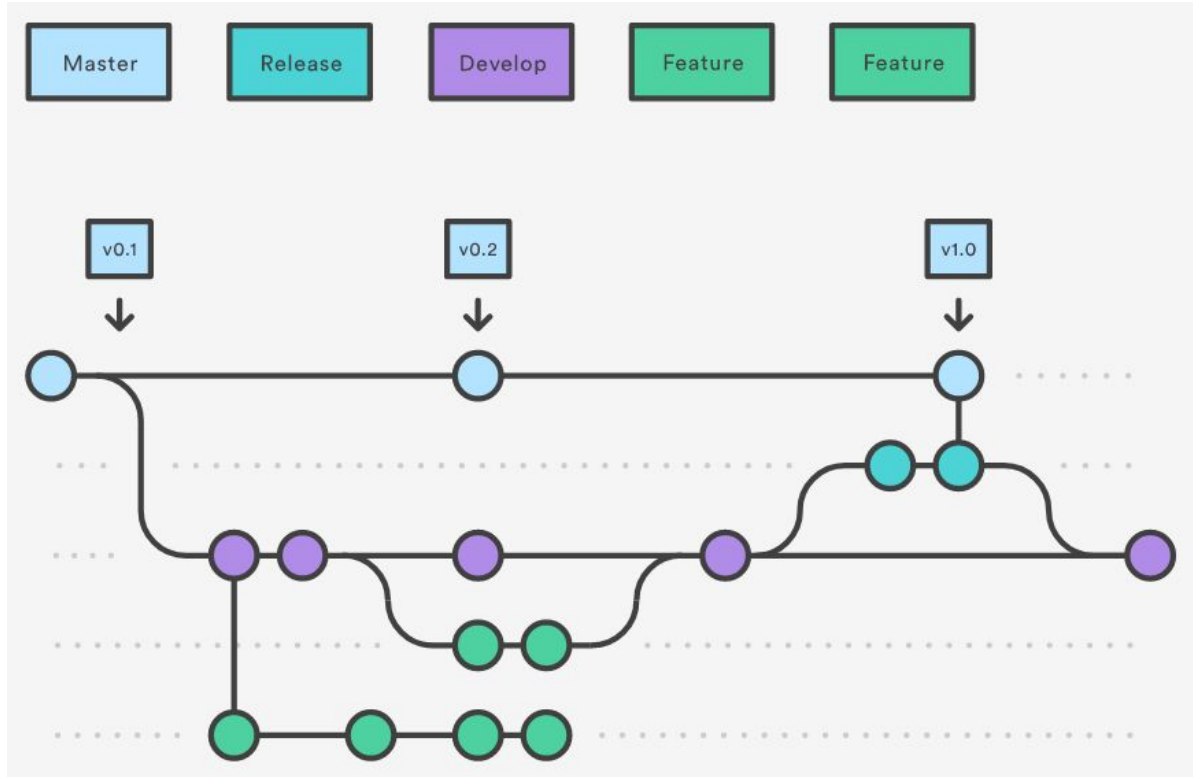
Protect matching branches

☒ **Require pull request reviews before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request with the required number of approving reviews and no changes requested before it can be merged into a branch that matches this rule.

Required approving reviews: 1 ▼

I can tell
more

Tutorial -- Branch



- master is the product
- Add features to develop branch
- After finishing and testing the feature, merge to master

[know more](#)

Write your code

```
(base) marvinchung@506B01R09944021: ~/Github/hello-Github$  
(base) marvinchung@506B01R09944021: ~/Github/Hello-Github$ vim hello_github.py
```

```
1 print("I am a good TA")  
2  
3
```


Tutorial -- Commit your codes

- After adding codes at local, you commit and push them to origin




- Example


```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git add hello_github.py  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git commit -m "add a sample code"  
[dev 14b7b1c] add a sample code  
1 file changed, 1 insertion(+)  
create mode 100644 hello_github.py  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$
```




Push to remote



```
(base) marvinchung@506801R09944021:~/Github/Hello-Github$ git push origin dev
Username for 'https://github.com': MarvinChung
Password for 'https://MarvinChung@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 316 bytes | 316.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'dev' on GitHub by visiting:
remote:   https://github.com/MarvinChung/Hello-Github/pull/new/dev
remote:
To https://github.com/MarvinChung/Hello-Github.git
 * [new branch]      dev -> dev
(base) marvinchung@506801R09944021:~/Github/Hello-Github$
```


 **MarvinChung / Hello-Github**


[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

 **dev** had recent pushes 1 minute ago [Compare & pull request](#)

 **main**  **2** branches  **0** tags [Go to file](#) [Add file](#) [Code](#)

 **MarvinChung** Initial commit a28ca9e 19 minutes ago  **1** commits

 **README.md** Initial commit 19 minutes ago

README.md 

Hello-Github

Tutorial -- Synchronize Codes from Origin

- Origin looks like this

The screenshot shows a GitHub repository page for 'MarvinChung / Hello-Github'. At the top, there are navigation tabs: Code (selected), Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. On the right, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below the navigation, a yellow banner indicates 'dev had recent pushes 14 minutes ago' with a 'Compare & pull request' button. Below this, there are buttons for 'Go to file', 'Add file', and 'Code'. A status bar shows 'This branch is 2 commits ahead of main.' with links for 'Pull request' and 'Compare'. A table of recent commits is displayed, with the most recent commit by 'cwlin1998' titled 'readme add contributor' (63c38f6, 1 minute ago, 3 commits) highlighted. Below the commits, the 'README.md' file is shown, containing the text 'Hello-Github'.

MarvinChung / Hello-Github

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

dev had recent pushes 14 minutes ago Compare & pull request

dev 2 branches 0 tags Go to file Add file Code

This branch is 2 commits ahead of main. Pull request Compare

	readme add contributor	63c38f6	1 minute ago	3 commits
README.md	readme add contributor		1 minute ago	
hello_github.py	add a sample code		17 minutes ago	

README.md

Hello-Github

My teammate has modified README.md.

Tutorial -- Synchronize Codes from Origin

- method 1:

fetch + merge

preferred, since it's safer

```
~/Github/Hello-Github$ git fetch origin dev
```

```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git merge origin/dev
Updating 14b7b1c..63c38f6
Fast-forward
 README.md | 5 ++++-
 1 file changed, 4 insertions(+), 1 deletion(-)
```

- method2:

pull

```
~/Github/Hello-Github$ git pull origin dev
```

My README.md in local

```
1 ## Hello-Github
```



```
1 ## Hello-Github
2
3 ## Contributor
4 TA
```

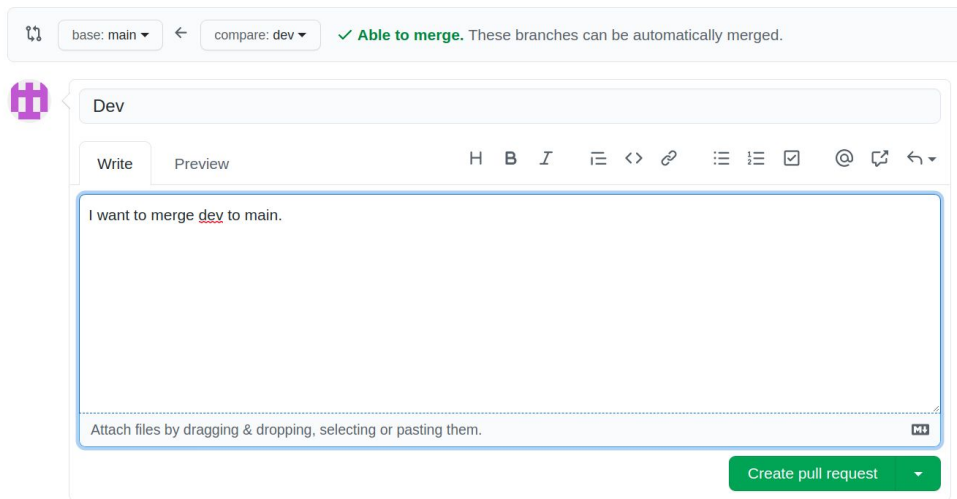
My local file has updated to latest code.

Tutorial -- Pull Request (PR)

- When your codes are ready to merge to main
- Pull Requests → New Pull Request → set dev branch merge to main

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).




base: main ← compare: dev ✓ Able to merge. These branches can be automatically merged.

Dev

Write Preview H B I ≡ <> 🔗 ≡ ≡ ☑ @ ↻ ↶

I want to merge dev to main.

Attach files by dragging & dropping, selecting or pasting them. 

Create pull request ▾

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Tutorial -- Pull Request (PR)

- A second reviewer is required
 - Second reviewer reviews
 - Files changed
 - Run codes if necessary
 - Accept if OK
 - Else, Reject the PR
- If PR is accepted, then feature branch merges into main

Dev #1

 Open MarvinChung wants to merge 2 commits into `main` from `dev`

 Conversation **1**  Commits **2**  Checks **0**  Files changed **2**



MarvinChung commented 8 minutes ago

Owner



I want to merge dev to main.



MarvinChung and others added 2 commits 39 minutes ago



 add a sample code

14b7b1c



 readme add contributor

63c38f6

My teammate reject my code



cwlin1998 requested changes 2 minutes ago

[View changes](#)

hello_github.py

```
...    ...    @@ -0,0 +1 @@
```

```
1 + print("I am a good TA")
```



cwlin1998 2 minutes ago

Collaborator



trash code



Reply...

Resolve conversation

Modify my code

```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ vim hello_github.py  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$
```

```
1 print("We are good TAs")  
2  
3
```

```
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git add hello_github.py  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git commit -m "change I to We"  
[dev 0d4c757] change I to We  
1 file changed, 3 insertions(+), 1 deletion(-)  
(base) marvinchung@506B01R09944021:~/Github/Hello-Github$ git push origin dev
```

My teammate approved my code



cwlin1998 approved these changes now

[View changes](#)

hello_github.py

```
...    ...    @@ -0,0 +1,3 @@  
1 + print("We are good TAs")
```



cwlin1998 now Collaborator



Nice!!!!!!!!!!!!



Reply...

Resolve conversation



cwlin1998 merged commit **576f2c4** into `main` now

[Revert](#)





Pull request successfully merged and closed


You're all set—the `dev` branch can be safely deleted.

[Delete branch](#)

Main branch is updated


 main ▾


 2 branches


 0 tags


Go to file


Add file ▾

 Code ▾

 **cwlin1998** Merge pull request [#1](#) from MarvinChung/dev ... 576f2c4 2 minutes ago ⌚ 5 commits

 README.md readme add contributor 35 minutes ago

 hello_github.py change I to We 4 minutes ago

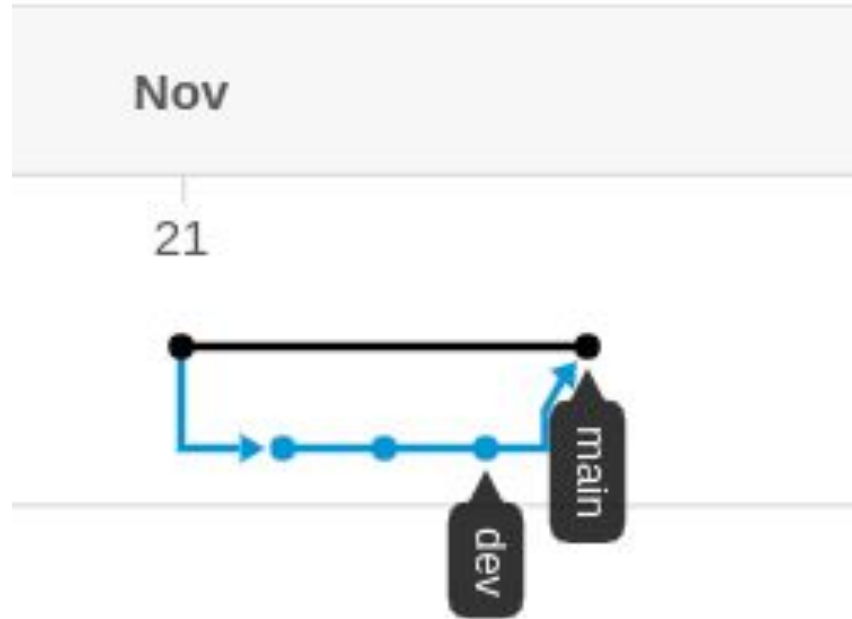
README.md 

Hello-Github

Contributer

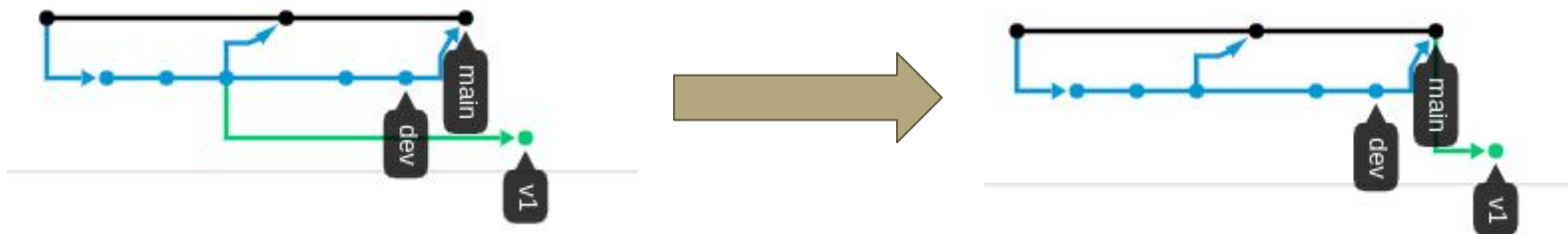
TA

Network graph



Tutorial -- Rebase

- How does other branch update with main ?



- Use rebase to update
 - pull codes from origin to origin/main (see previous slides)
 - git checkout v1
 - git rebase origin/main
 - git push origin v1 -f

Tutorial-Tag

- Use tag to label version
 - `git tag [tag-name] -m [your message]`
 - `git push origin [tag-name]`

```
(base) marvinchung@506801R09944021:~/Github/Hello-Github$ git tag v1.0 -m "robust version"
(base) marvinchung@506801R09944021:~/Github/Hello-Github$ git show
commit 14802ece6952fd7e2b2ee525e4341eb65f264b84 (HEAD -> v1, tag: v1.0, origin/v1)
Author: MarvinChung <marvin852316497@gmail.com>
Date: Sat Nov 21 18:45:05 2020 +0800
```

```
add function
```

```
diff --git a/add.py b/add.py
```

```
new file mode 100644
```

```
index 0000000..6305374
```

```
--- /dev/null
```

```
+++ b/add.py
```

```
@@ -0,0 +1,2 @@
```

```
+def add(a, b):
```

```
+    print(a+b)
```

```
(base) marvinchung@506801R09944021:~/Github/Hello-Github$ git tag
```

```
v1.0
```

```
Everything up-to-date
```

```
(base) marvinchung@506801R09944021:~/Github/Hello-Github$ git push origin v1.0
```

```
Username for 'https://github.com': MarvinChung
```

```
Password for 'https://MarvinChung@github.com':
```

```
Enumerating objects: 1, done.
```

```
Counting objects: 100% (1/1), done.
```

```
Writing objects: 100% (1/1), 170 bytes | 170.00 KiB/s, done.
```

```
Total 1 (delta 0), reused 0 (delta 0)
```

```
To https://github.com/MarvinChung/Hello-Github.git
```

```
* [new tag] v1.0 -> v1.0
```

```
(base) marvinchung@506801R09944021:~/Github/Hello-Github$
```

Tutorial-Tag

MarvinChung / Hello-Github

< > Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

📈 Insights

⚙ Settings

👤 v1 had recent pushes 11 minutes ago

Compare & pull request

📁 v1.0 ▾

🔗 3 branches

📁 1 tag

Go to file

📄 Code ▾

Switch branches/tags

✕

Find a tag

Branches

Tags

✓ v1.0

[View all tags](#)

📄 hello_rebase.py

📄 hello_world.py

14802ec 18 minutes ago 🕒 9 commits

readme add contributor

1 hour ago

add function

18 minutes ago

change I to We

1 hour ago

hello rebase

26 minutes ago

sample code 2

24 minutes ago

Useful Terminal Command (Linux/MacOS)

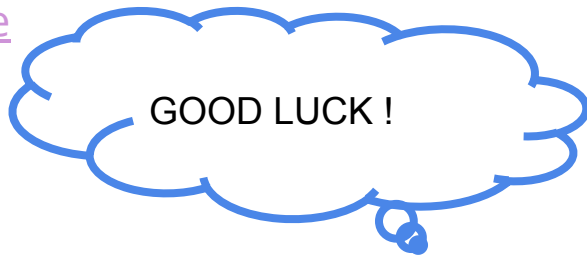
command	description	example
cd	Changes the directory of the command line path.	cd "path/to/directory/"
ls	Lists the contents of a directory.	ls "path/to/directory/"
cp	copy file	cp "filename" "newfilename"
mv	move a file	mv "filename" "path/to/new/file/location"
mv	rename a file	mv "path/to/filename" "path/to/newfilename"
rm	remove a file	rm "path/to/filename"
mkdir	create a directory	mkdir "path/to/new/directory"

Other Useful Tools

- `git status [-uno]`
 - to show status of your changing files
- `git diff [filename]`
 - to know what you modified with [filename]
- `git diff [commit hash 1] [commit hash 2] [filename]`
 - to know the difference of the same file on two different commits
- `git branch -[d/D] [branch name]`
 - delete branch, force push to origin if neccessary
- `git log [--graph]`
- `git reset [commit hash/HEAD[^]] [--hard]`
 - reset commit to specified
 - “^” the previous commit
 - “--hard” recover commit completely

Homework

- What you MUST do in your homework
 - Protect main branch
 - Branch if adding new features
 - If there are conflicts, rebase before pull request
 - Collaboration on the codes, that is, we want to see more than one person's commit.
- HW github:
<https://github.com/cwlin1998/reversi-pygame>
- 分組名單:
<https://pse.is/39qp eb>



Videos

Video link

- [Github Getting Started](#)
- [HW introduction](#)
- [code introduction](#)
- [Intro to pdb](#)

video list:

https://www.youtube.com/playlist?list=PL8RRW7e03_x2FqpgLxWehpbytFKRPy6ba