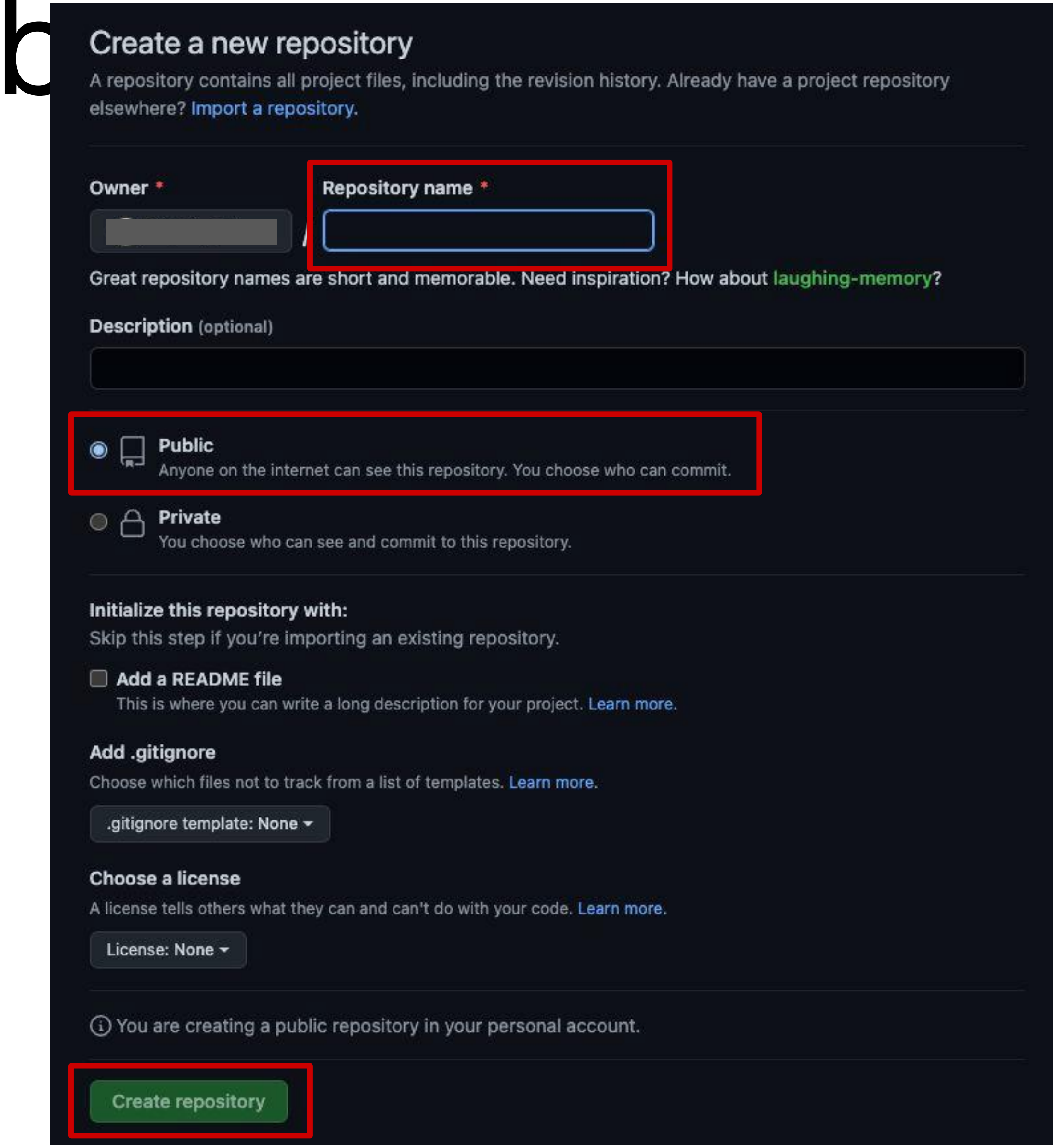
Eclipse with GitHub

陳卉縈 112356043@nccu.edu.tw
王瀚 111306078@g.nccu.edu.tw
劉亭妤113356048@g.nccu.edu.tw

Build team’s GitHub

- Login GitHub (https://github.com/)
- New 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.

Owner * [redacted] / Repository name *

Great repository names are short and memorable. Need inspiration? How about laughing-memory?

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.

.gitignore template: None

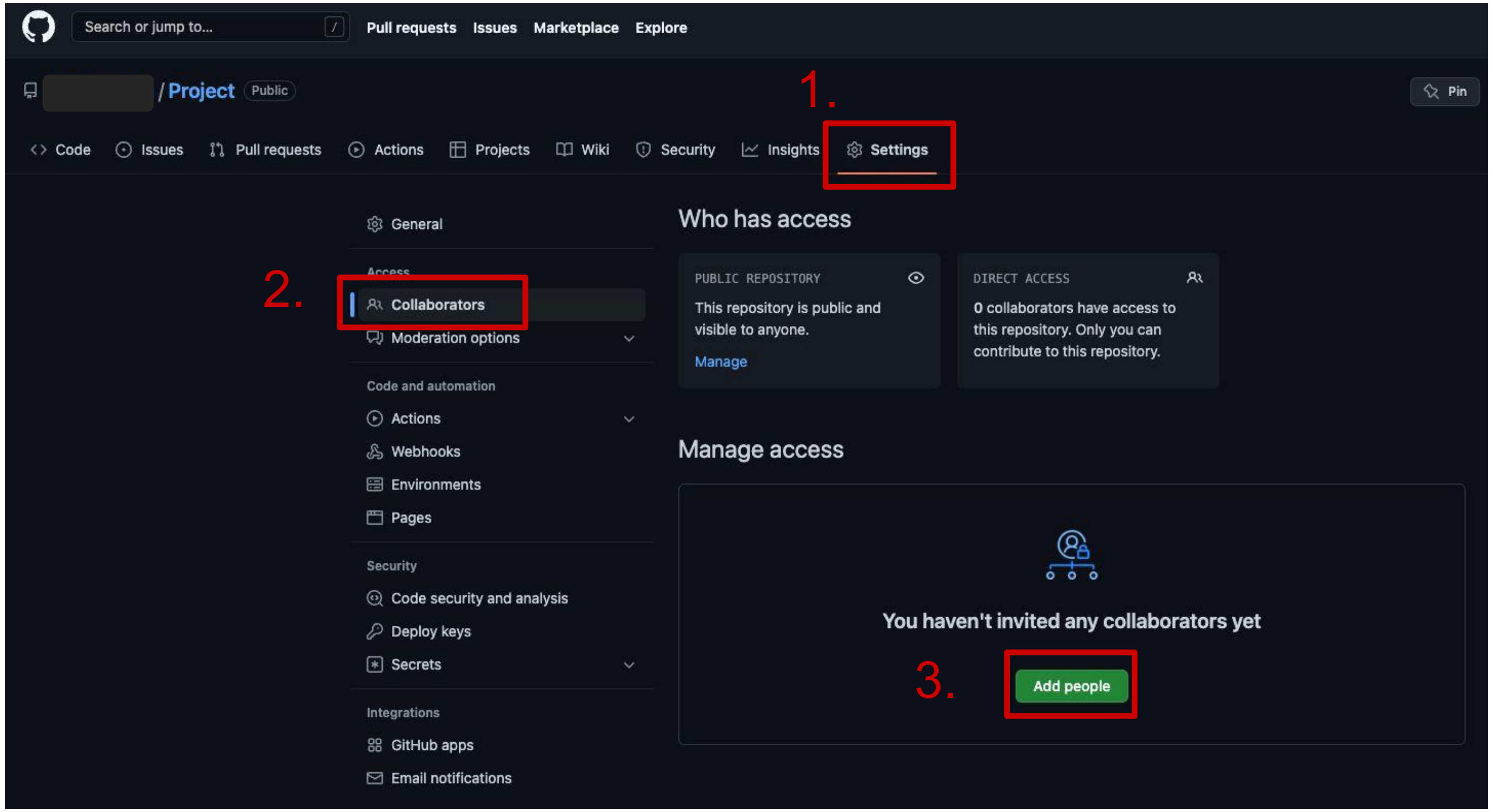
Choose a license
A license tells others what they can and can't do with your code. Learn more.

License: None

ⓘ You are creating a public repository in your personal account.

Build team's GitHub

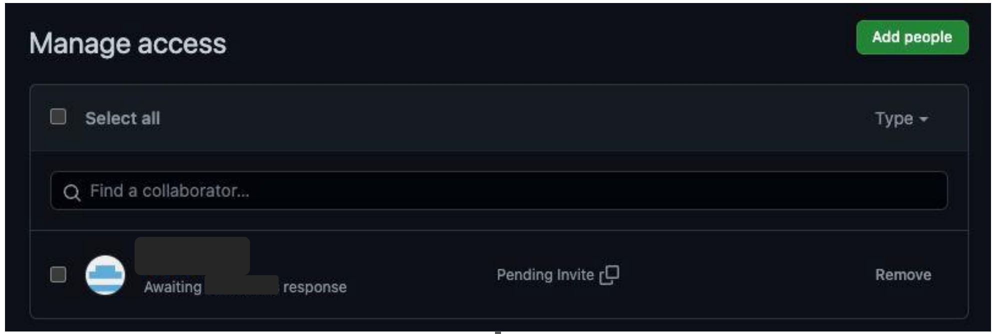
- Add your members to repository



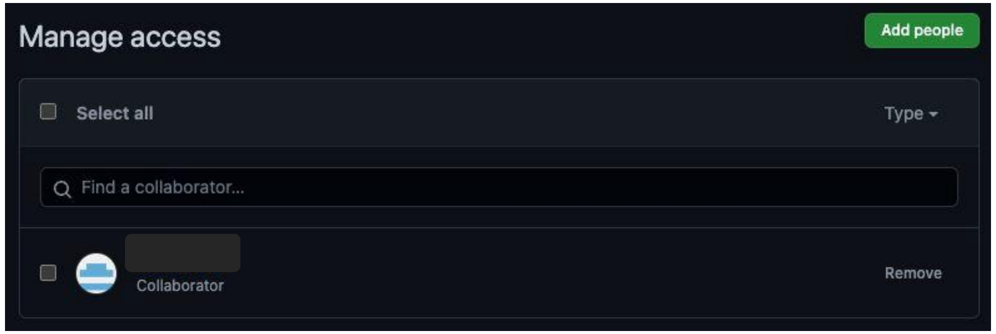
This repository is public and visible to anyone.

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

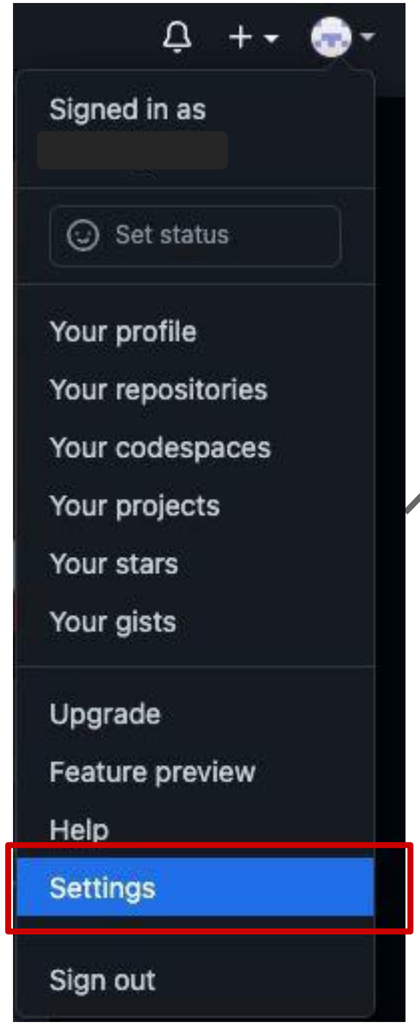
You haven't invited any collaborators yet

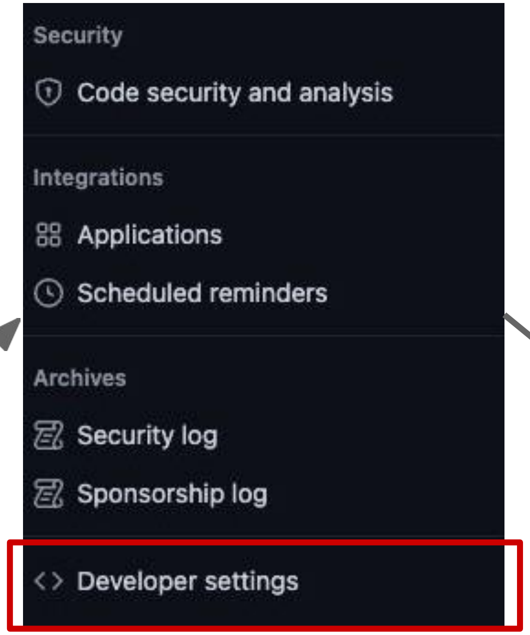


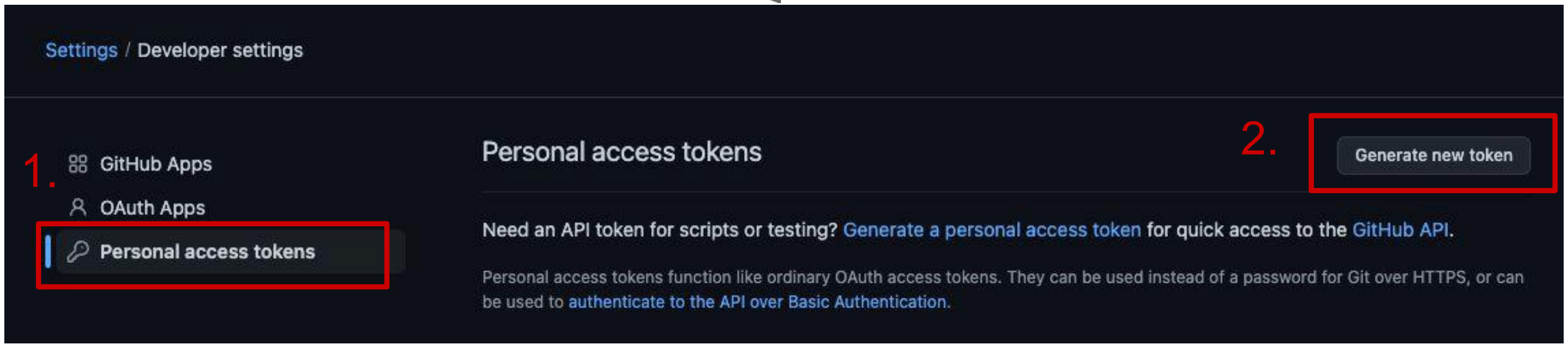
After your members join



Get the access token





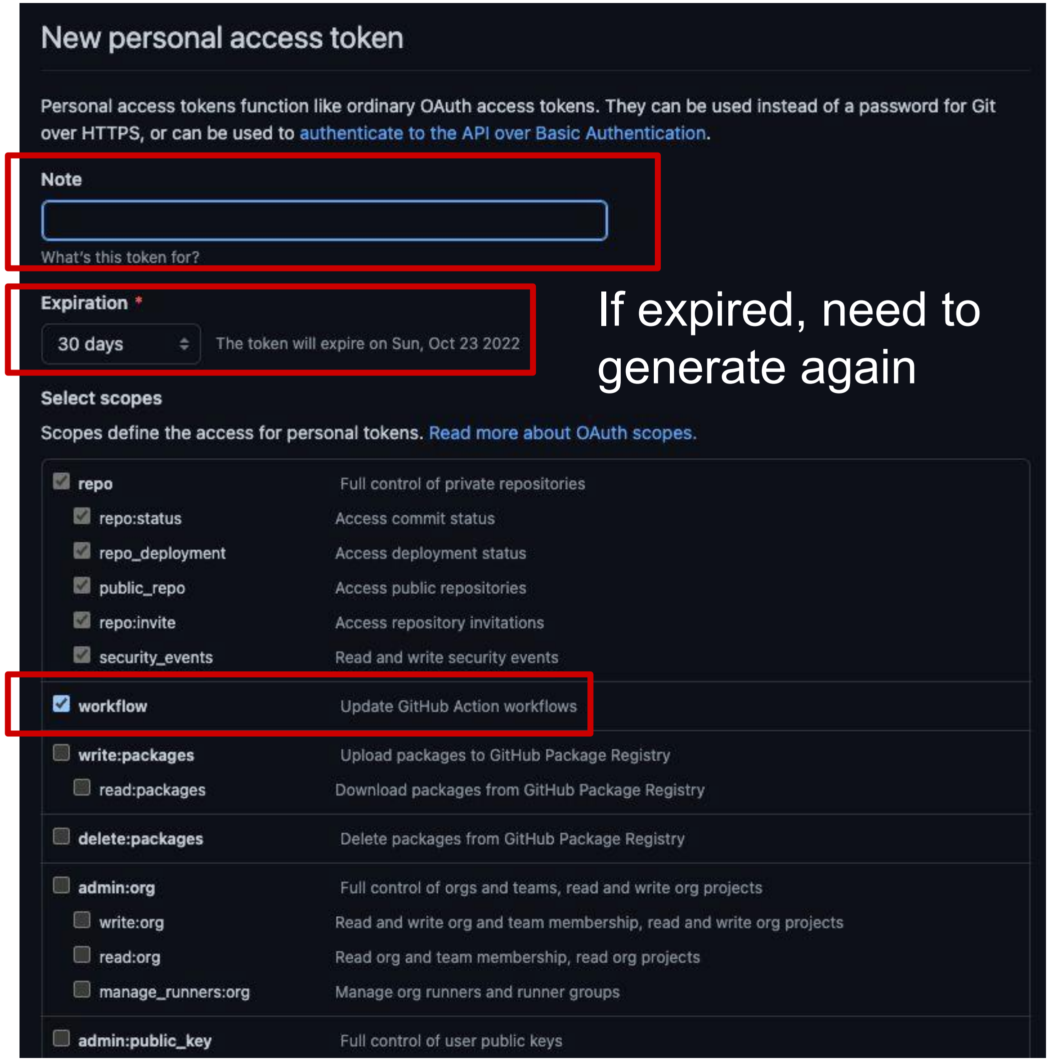


Personal access tokens

Need an API token for scripts or testing? Generate a personal access token for quick access to the GitHub API.

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Get the access toke



New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

What's this token for?

Expiration *

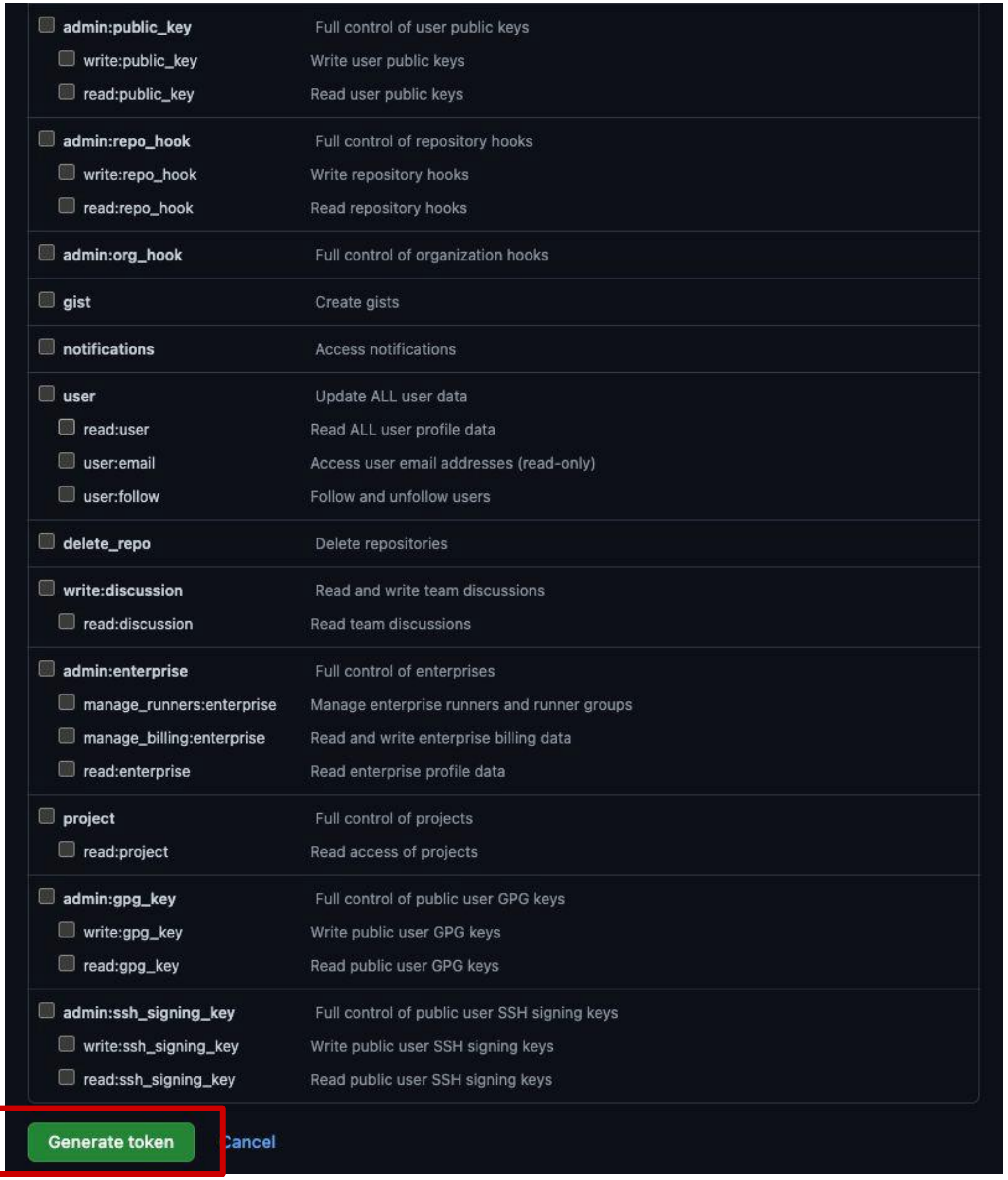
30 days — The token will expire on Sun, Oct 23 2022

Select scopes

Scopes define the access for personal tokens. Read more about OAuth scopes.

☑ **repo**	Full control of private repositories
☑ repo:status	Access commit status
☑ repo_deployment	Access deployment status
☑ public_repo	Access public repositories
☑ repo:invite	Access repository invitations
☑ security_events	Read and write security events
☑ **workflow**	Update GitHub Action workflows
□ **write:packages**	Upload packages to GitHub Package Registry
□ read:packages	Download packages from GitHub Package Registry
□ **delete:packages**	Delete packages from GitHub Package Registry
□ **admin:org**	Full control of orgs and teams, read and write org projects
□ write:org	Read and write org and team membership, read and write org projects
□ read:org	Read org and team membership, read org projects
□ manage_runners:org	Manage org runners and runner groups
□ **admin:public_key**	Full control of user public keys

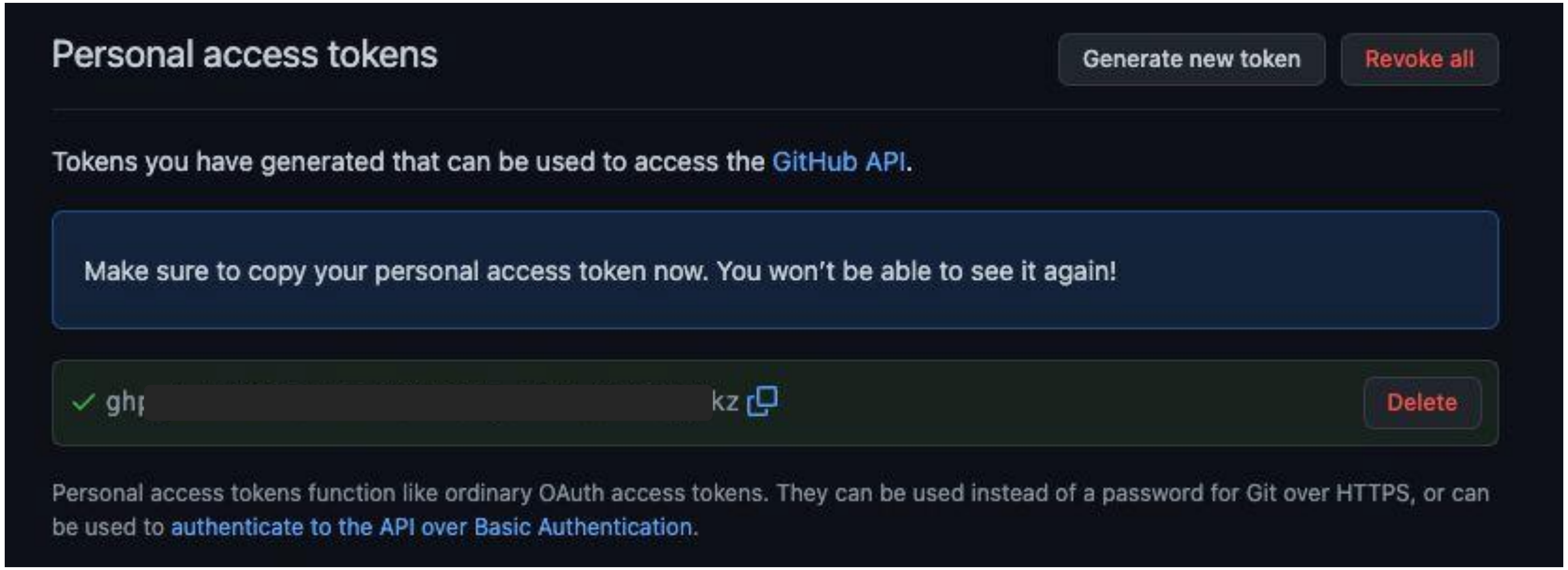
If expired, need to generate again



□ **admin:public_key**	Full control of user public keys
□ write:public_key	Write user public keys
□ read:public_key	Read user public keys
□ **admin:repo_hook**	Full control of repository hooks
□ write:repo_hook	Write repository hooks
□ read:repo_hook	Read repository hooks
□ **admin:org_hook**	Full control of organization hooks
□ **gist**	Create gists
□ **notifications**	Access notifications
□ **user**	Update ALL user data
□ read:user	Read ALL user profile data
□ user:email	Access user email addresses (read-only)
□ user:follow	Follow and unfollow users
□ **delete_repo**	Delete repositories
□ **write:discussion**	Read and write team discussions
□ read:discussion	Read team discussions
□ **admin:enterprise**	Full control of enterprises
□ manage_runners:enterprise	Manage enterprise runners and runner groups
□ manage_billing:enterprise	Read and write enterprise billing data
□ read:enterprise	Read enterprise profile data
□ **project**	Full control of projects
□ read:project	Read access of projects
□ **admin:gpg_key**	Full control of public user GPG keys
□ write:gpg_key	Write public user GPG keys
□ read:gpg_key	Read public user GPG keys
□ **admin:ssh_signing_key**	Full control of public user SSH signing keys
□ write:ssh_signing_key	Write public user SSH signing keys
□ read:ssh_signing_key	Read public user SSH signing keys

Get the access token

- Make sure to copy your access token to your note!
- This will use when you want to push code to GitHub.



Personal access tokens

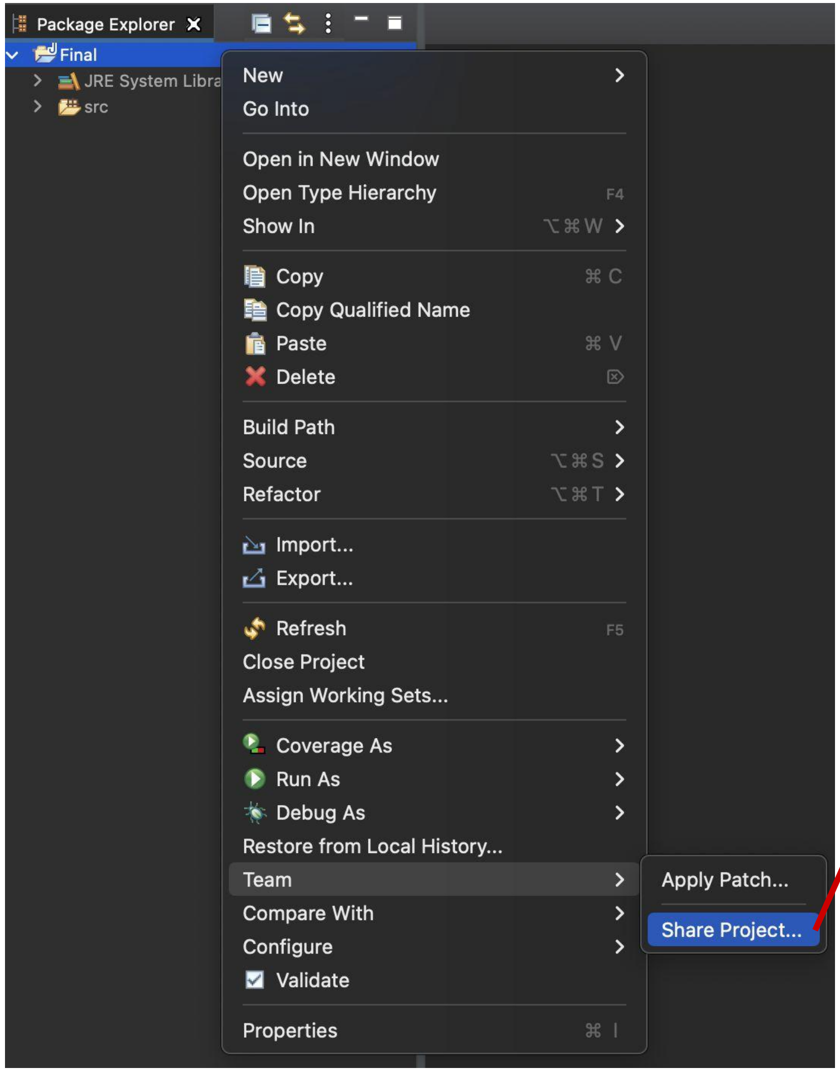
Tokens you have generated that can be used to access the GitHub API.

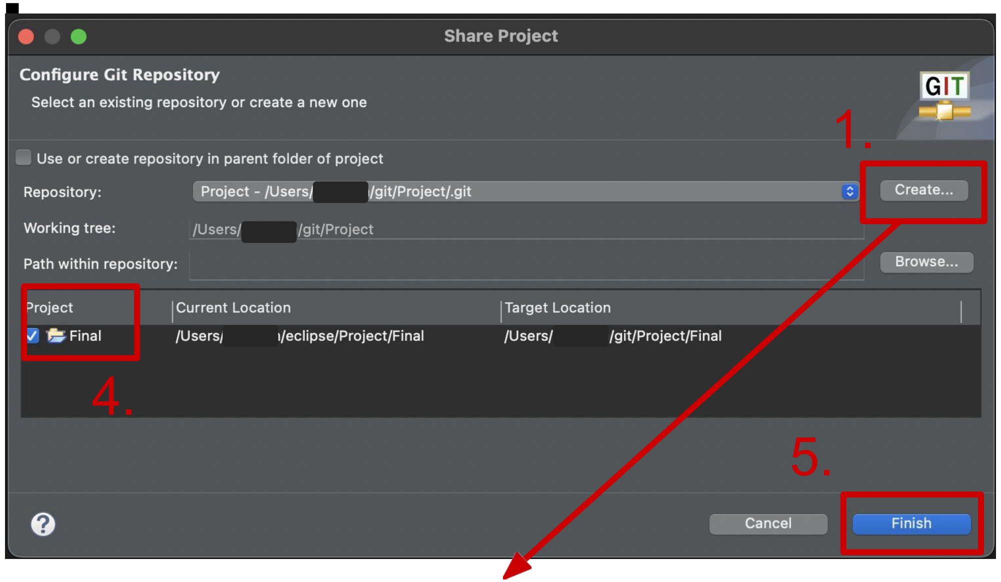
Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp[redacted]kz

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Eclipse with GitHu





Configure Git Repository

Select an existing repository or create a new one

□ Use or create repository in parent folder of project

Repository: Project - /Users/[redacted]/git/Project/.git

Working tree: /Users/[redacted]/git/Project

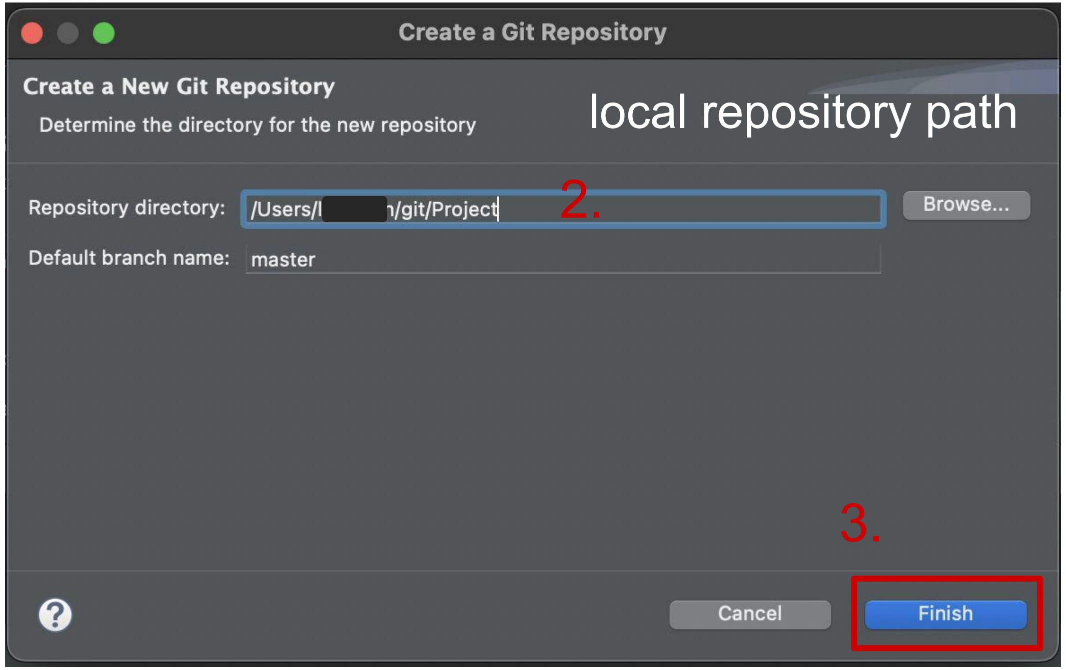
Path within repository:

Project	Current Location	Target Location
☑ Final	/Users/[redacted]/eclipse/Project/Final	/Users/[redacted]/git/Project/Final

1.

4.

5.



Create a New Git Repository

Determine the directory for the new repository

local repository path

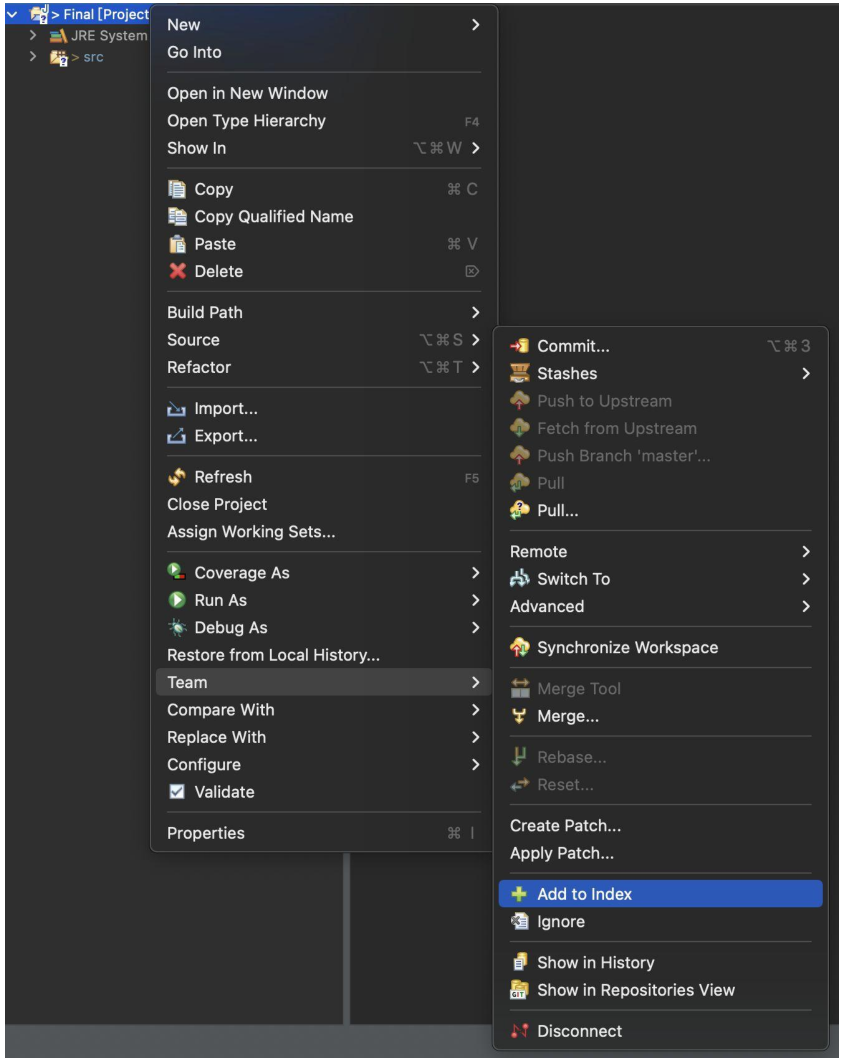
Repository directory: /Users/[redacted]/git/Project

Default branch name: master

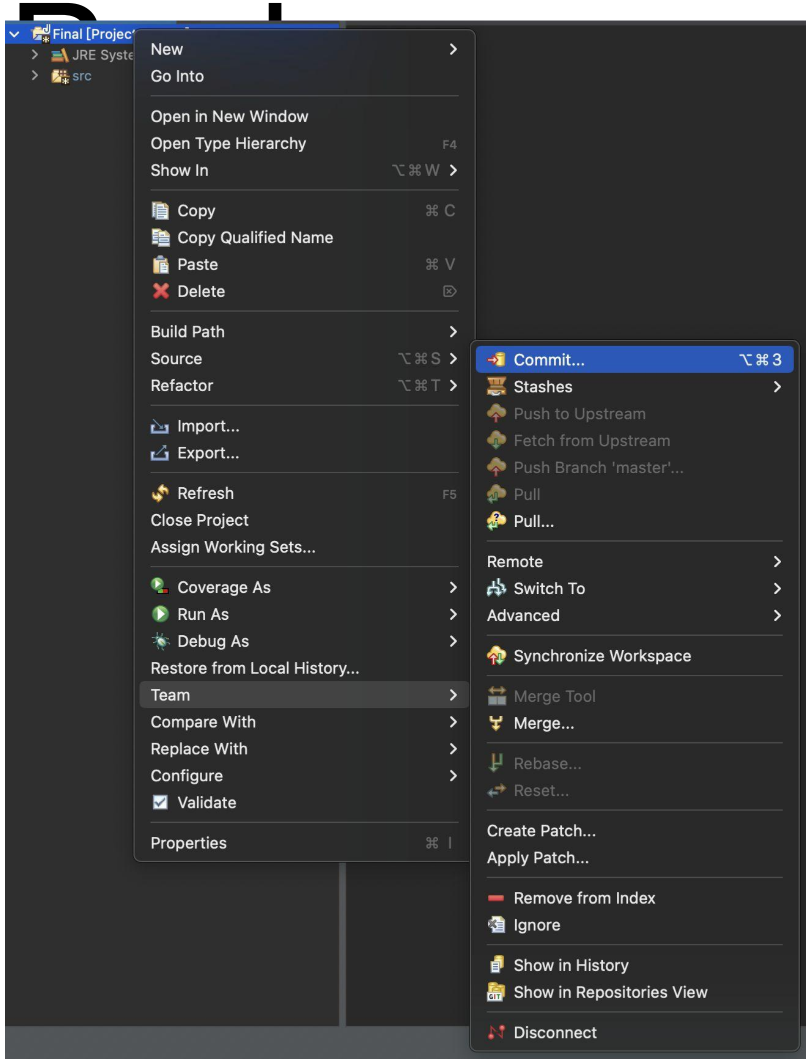
2.

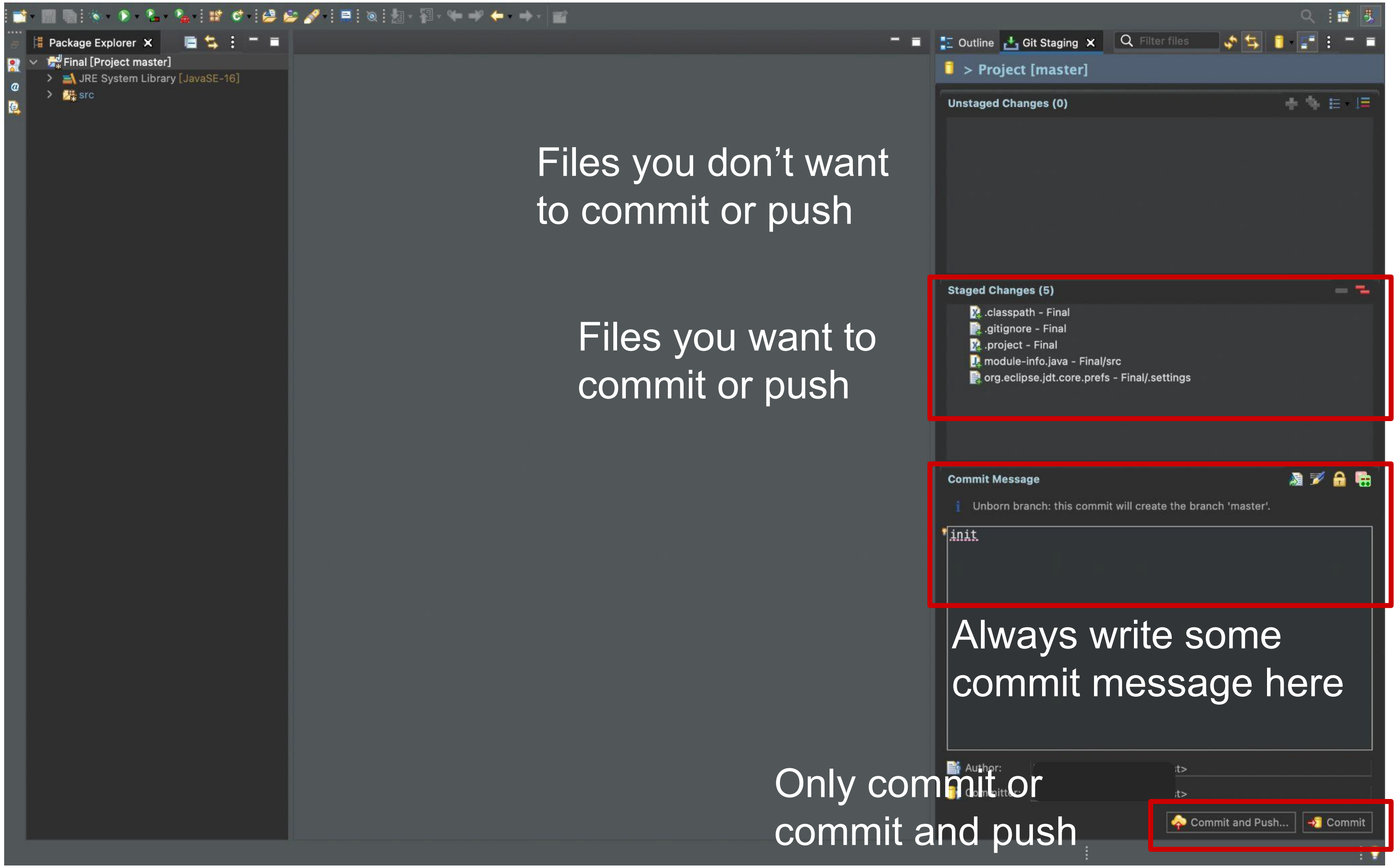
3.

Eclipse with GitHub



Eclipse with GitHub - Commit and Push





.classpath - Final
.gitignore - Final
.project - Final
module-info.java - Final/src
org.eclipse.jdt.core.prefs - Final/.settings

Unborn branch: this commit will create the branch 'master'.

init

Files you don't want to commit or push

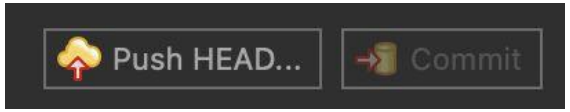
Files you want to commit or push

Always write some commit message here

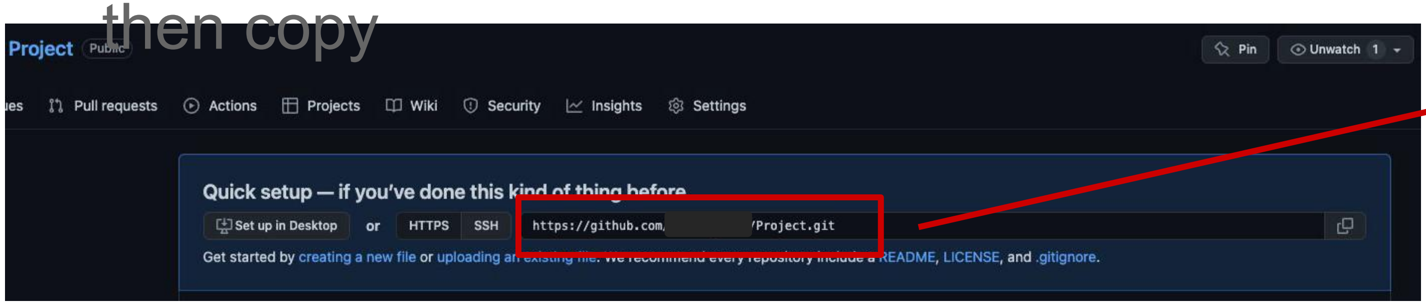
Only commit or commit and push

Eclipse with GitHub - Commit and Push

- Click **Commit** will change to
- Click **Commit and Push**



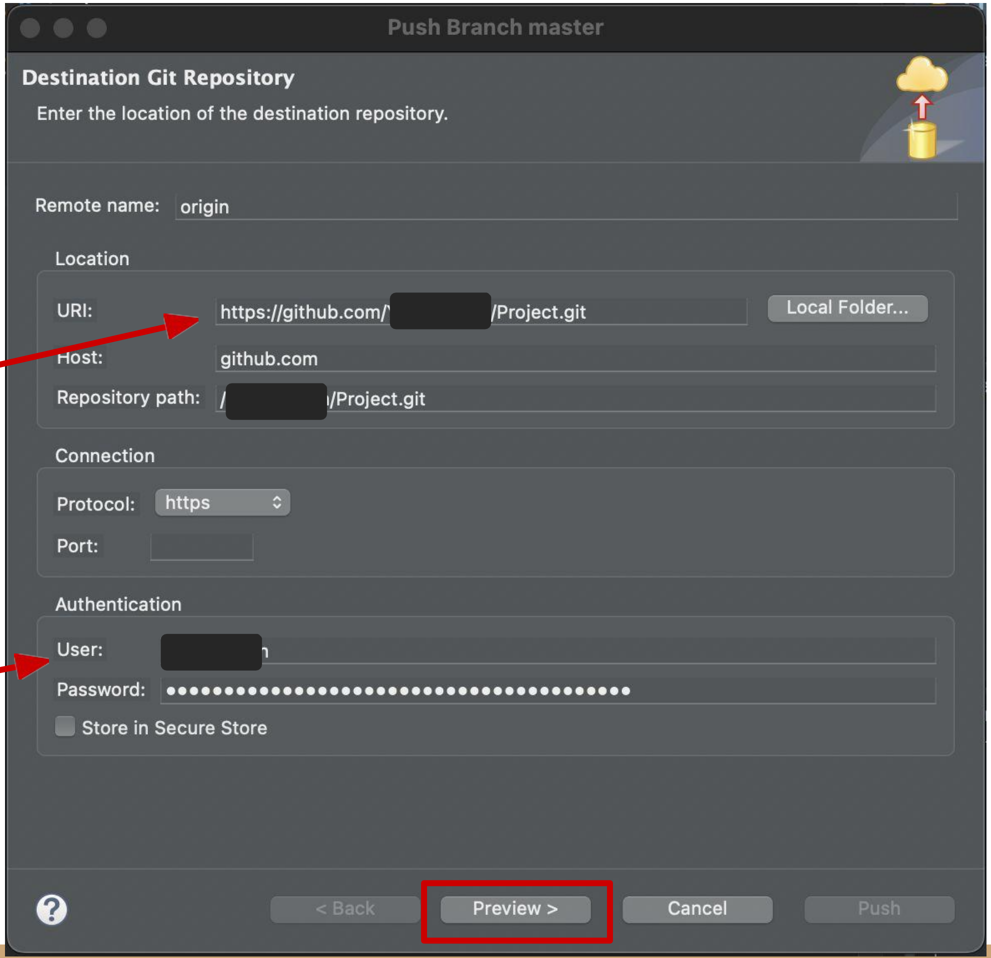
then copy



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/[redacted]/Project.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.



Push Branch master

Destination Git Repository

Enter the location of the destination repository.

Remote name: origin

Location

URI: https://github.com/[redacted]/Project.git Local Folder...

Host: github.com

Repository path: /[redacted]/Project.git

Connection

Protocol: https

Port:

Authentication

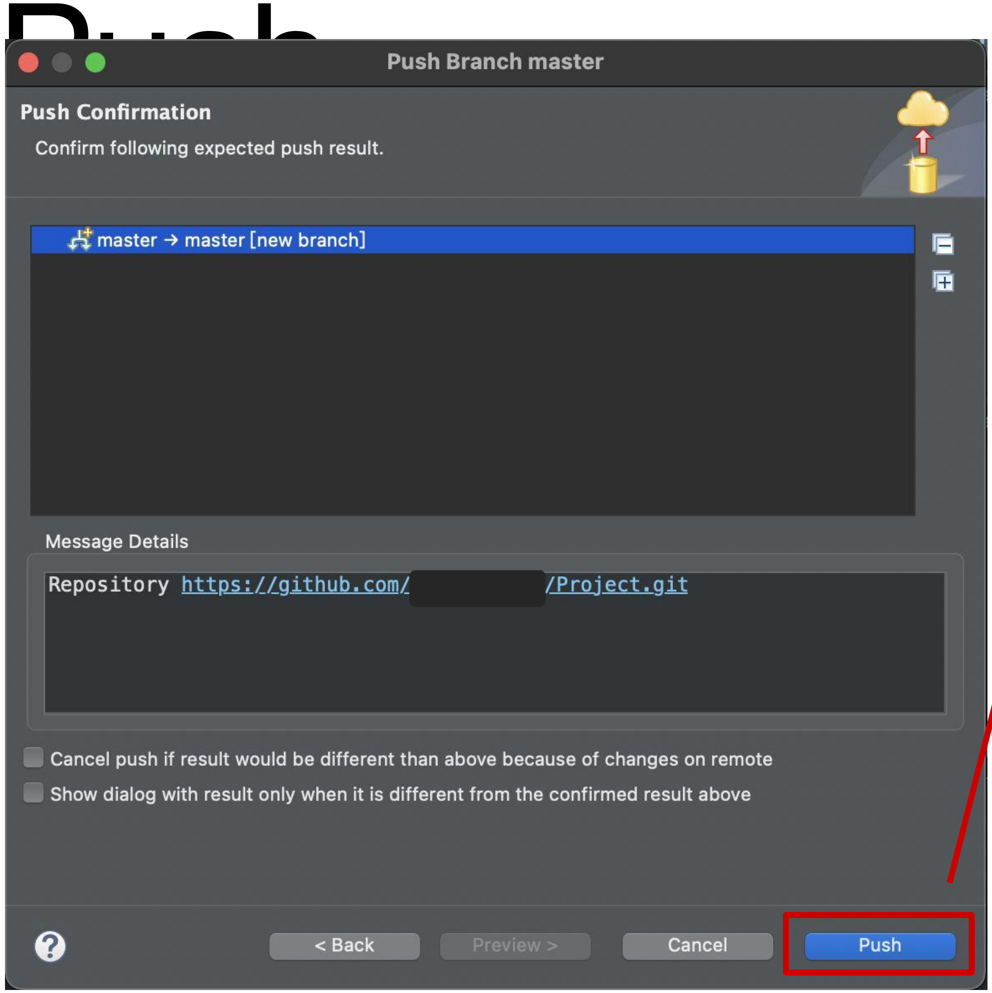
User: [redacted]n

Password: ••

☐ Store in Secure Store

Your GitHub username
Your access token

Eclipse with GitHub - Commit and Push



Push Branch master

Push Confirmation

Confirm following expected push result.

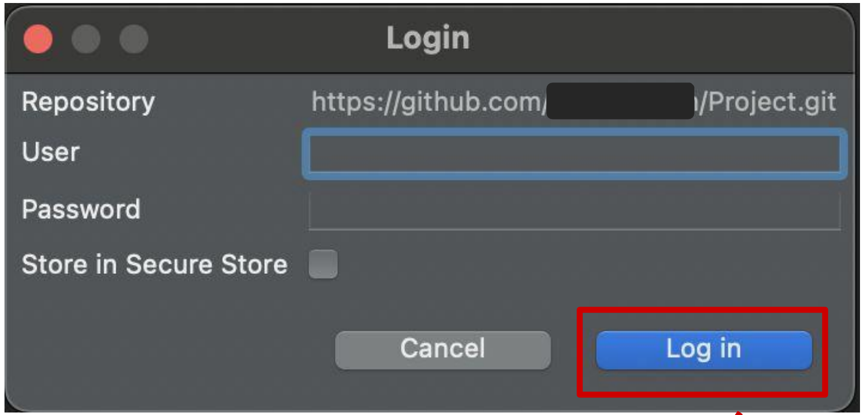
master → master [new branch]

Message Details

```
Repository https://github.com/[redacted]/Project.git
```

□ Cancel push if result would be different than above because of changes on remote

□ Show dialog with result only when it is different from the confirmed result above



Login

Repository https://github.com/[redacted]/Project.git

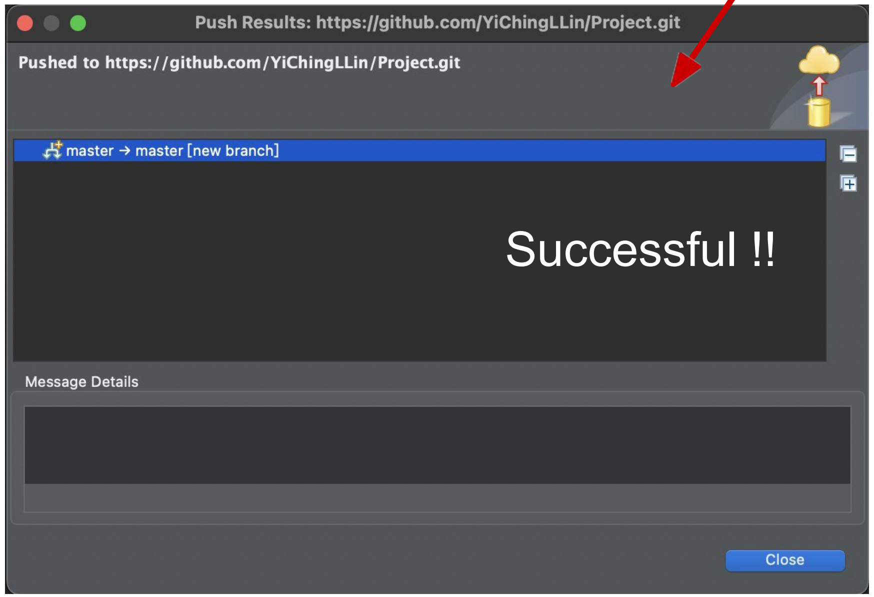
User

Password

Store in Secure Store □

Your GitHub username

Your access token



Push Results: https://github.com/YiChingLLin/Project.git

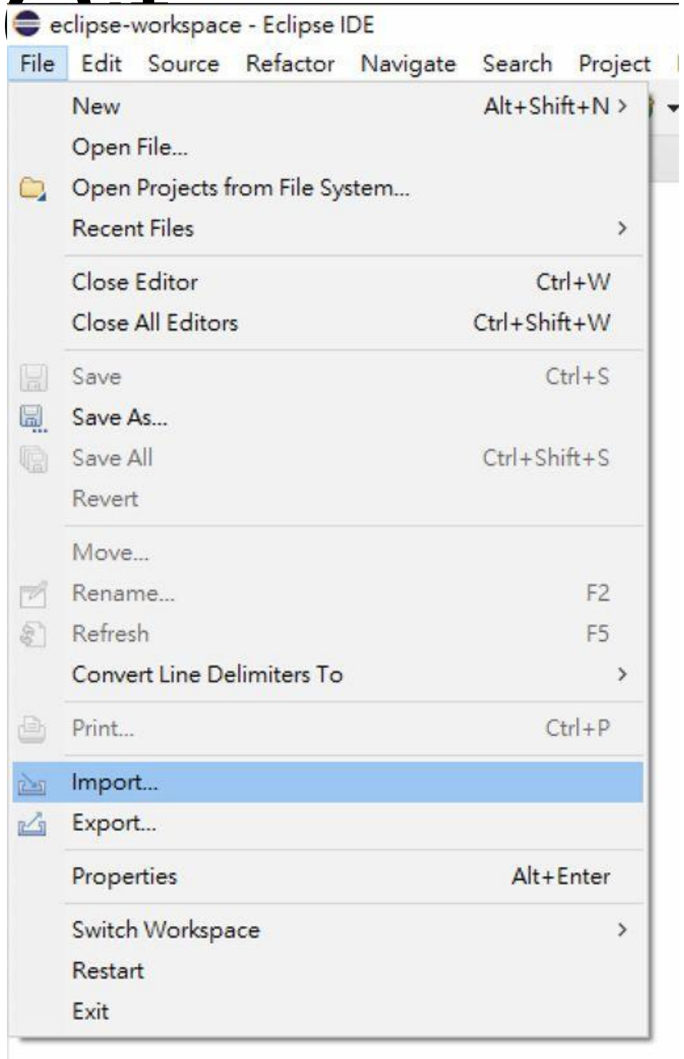
Pushed to https://github.com/YiChingLLin/Project.git

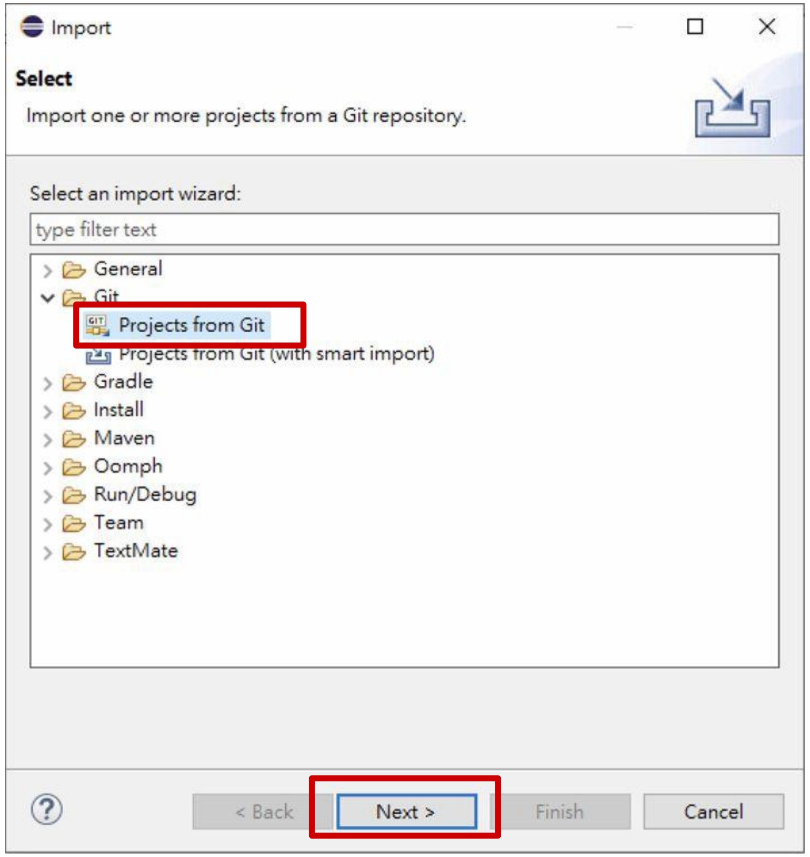
master → master [new branch]

Successful !!

Message Details

Eclipse with GitHub - Import project from Git

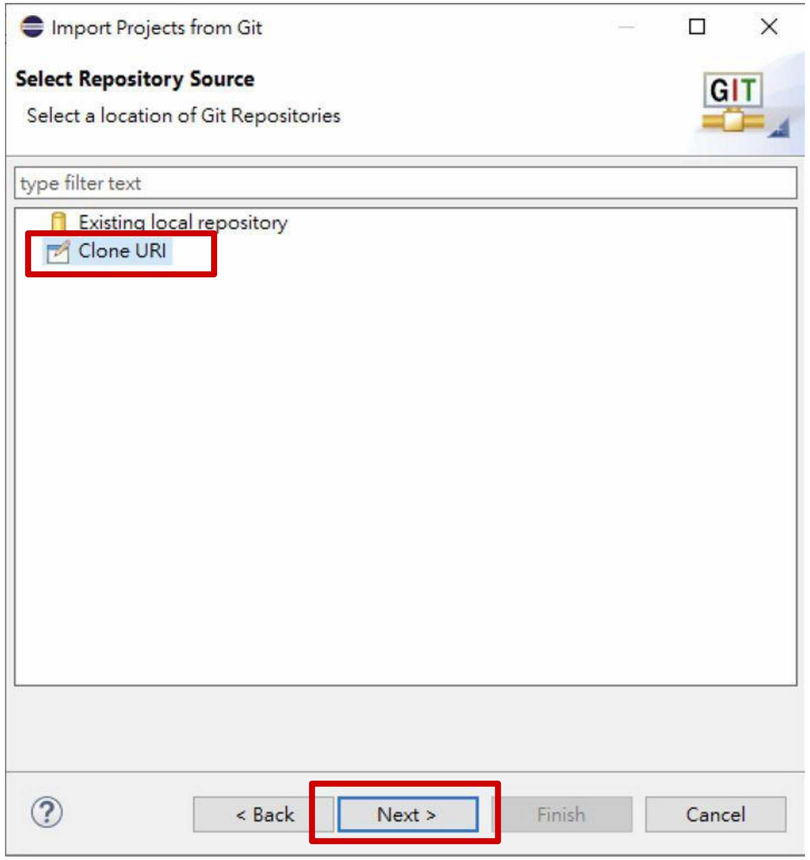




Select

Import one or more projects from a Git repository.

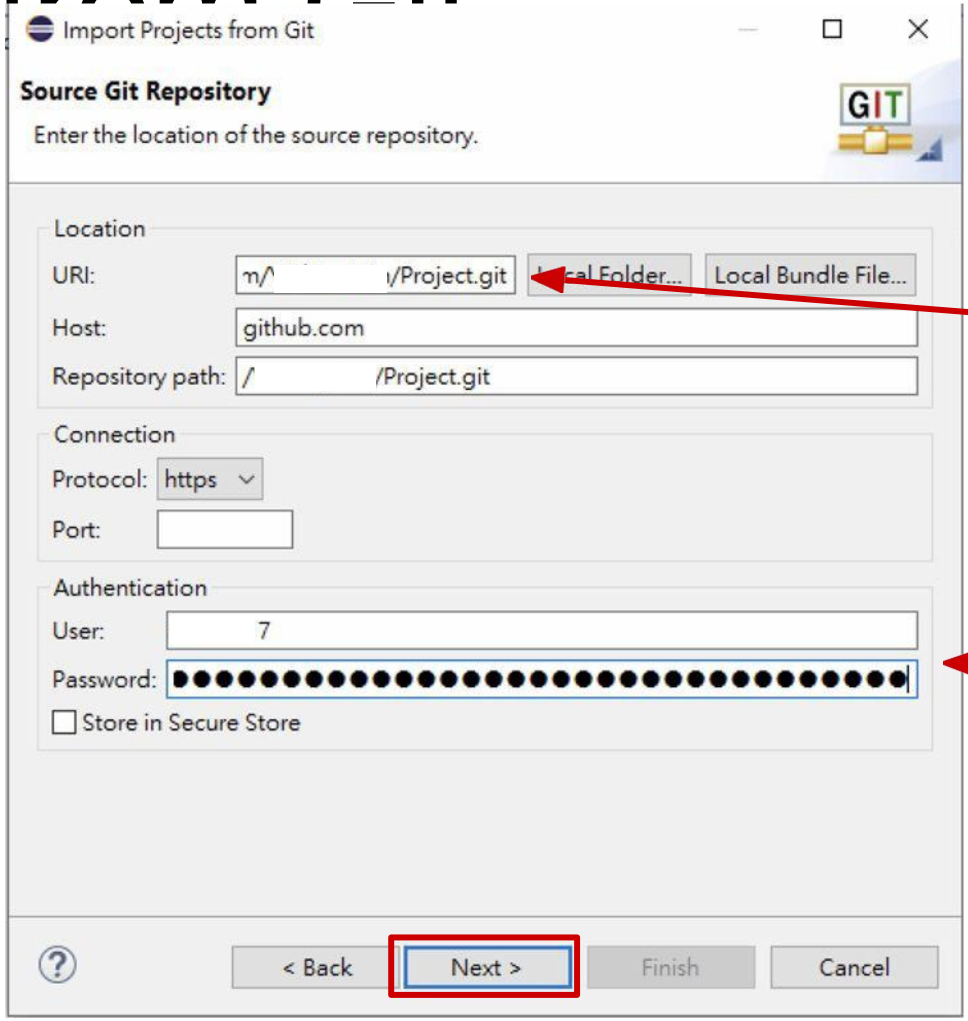
Select an import wizard:



Select Repository Source

Select a location of Git Repositories

Eclipse with GitHub - Import project from Git



Import Projects from Git

Source Git Repository

Enter the location of the source repository.

Location

URI: m/ /Project.git Local Folder... Local Bundle File...

Host: github.com

Repository path: / /Project.git

Connection

Protocol: https

Port:

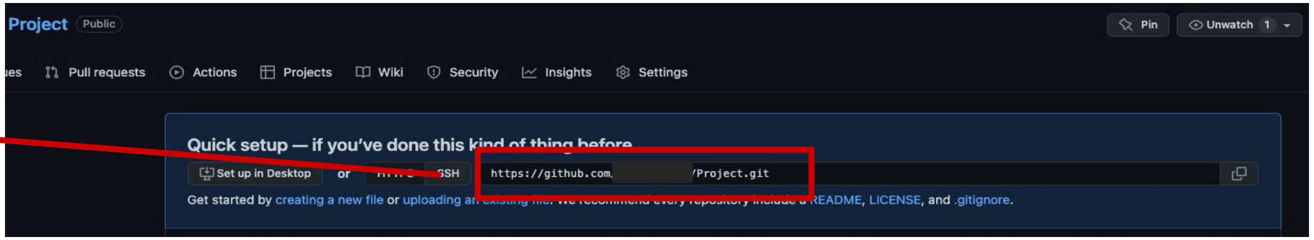
Authentication

User: 7

Password: ●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

□ Store in Secure Store

< Back Next > Finish Cancel



Project Public

Pin Unwatch 1

ues Pull requests Actions Projects Wiki Security Insights Settings

Quick setup — if you've done this kind of thing before

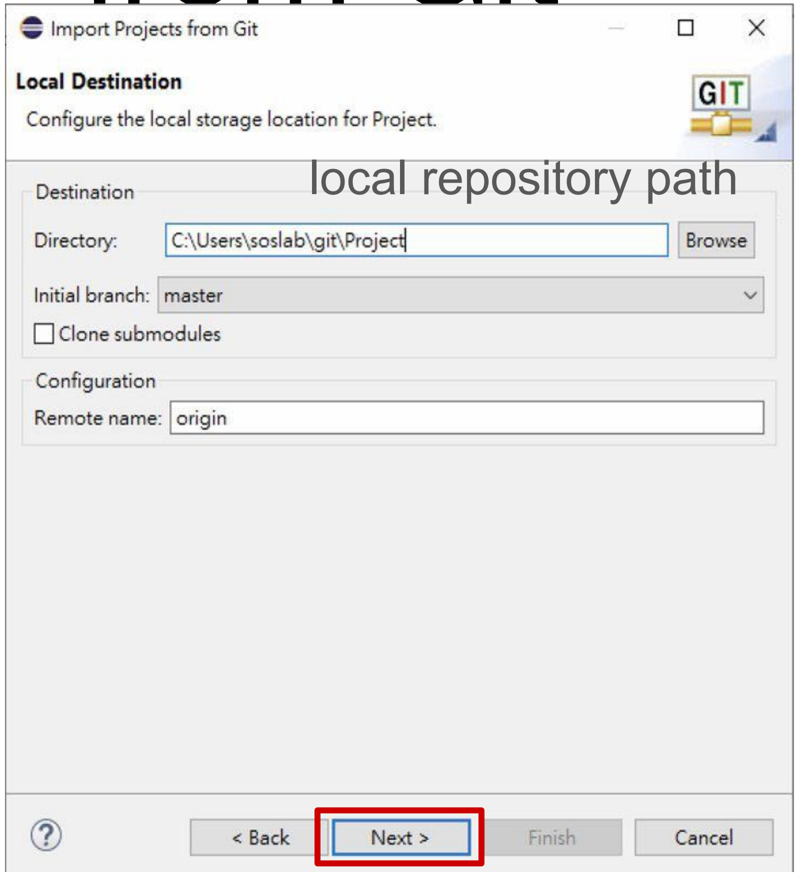
Set up in Desktop or HTTPS SSH https://github.com/[redacted]/Project.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

Your GitHub username

Your access token

Eclipse with GitHub - Import project from Git



Import Projects from Git

Local Destination

Configure the local storage location for Project.

local repository path

Destination

Directory: C:\Users\soslab\git\Project [Browse]

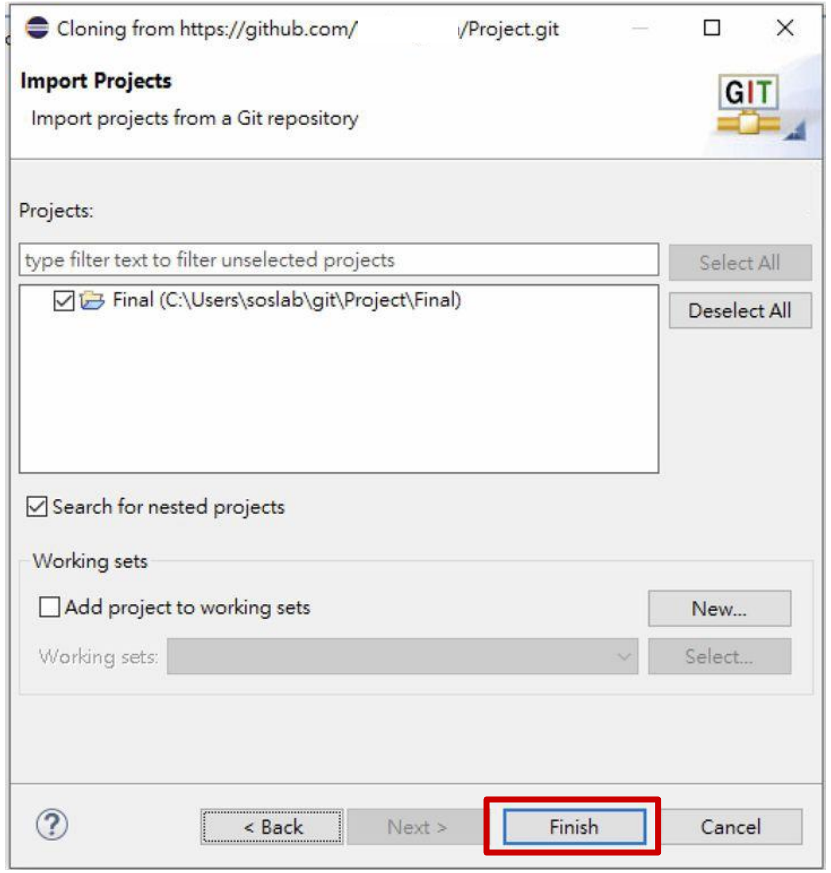
Initial branch: master

☐ Clone submodules

Configuration

Remote name: origin

[< Back] [Next >] [Finish] [Cancel]



Cloning from https://github.com/ /Project.git

Import Projects

Import projects from a Git repository

Projects:

type filter text to filter unselected projects [Select All] [Deselect All]

☑ Final (C:\Users\soslab\git\Project\Final)

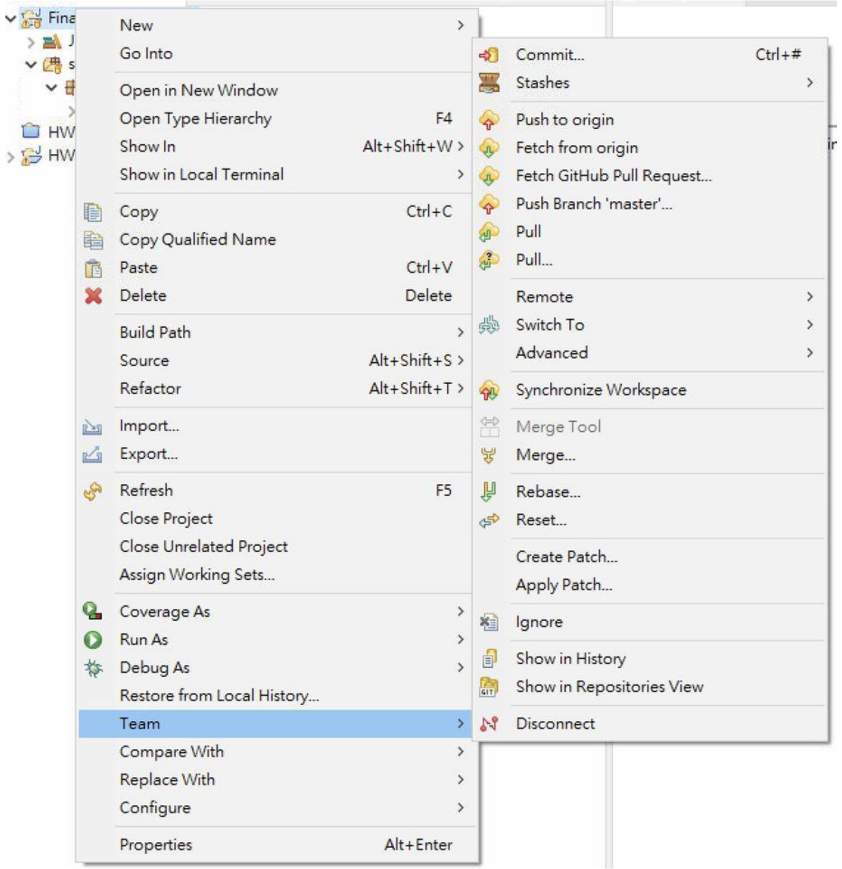
☑ Search for nested projects

Working sets

☐ Add project to working sets [New...]

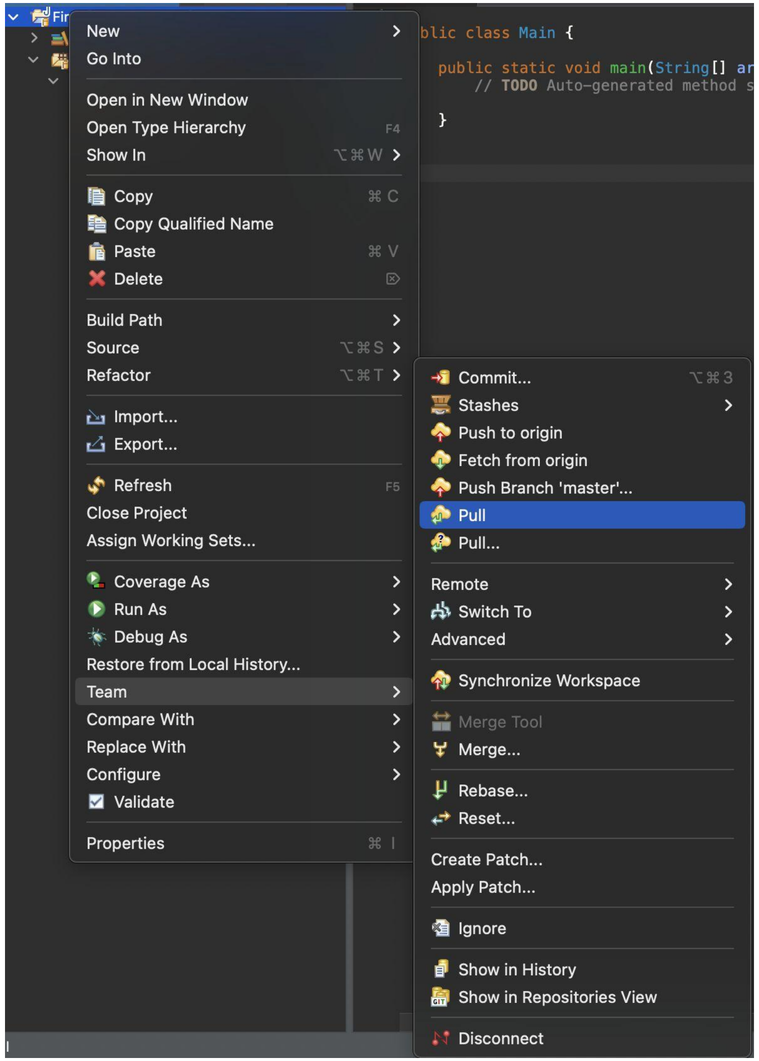
Working sets: [Select...]

[< Back] [Next >] [Finish] [Cancel]



If it has “Add to index”, click it first

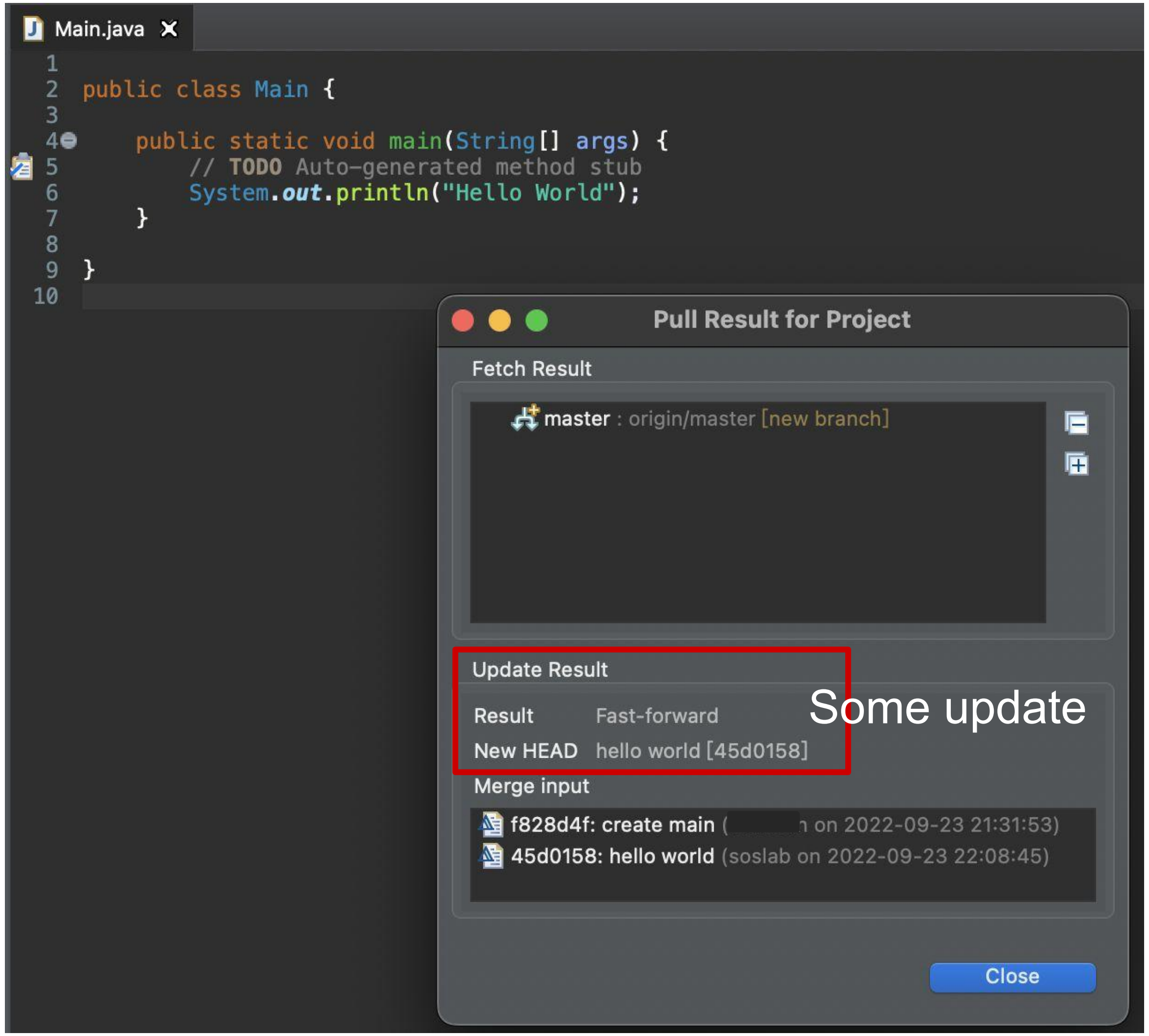
Eclipse with GitHub - Pull



```
blic class Main {

  public static void main(String[] ar
      // TODO Auto-generated method s

  }
```



Main.java

```

public class Main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println("Hello World");
    }

}
```

Pull Result for Project

Fetch Result

master : origin/master [new branch]

Update Result

Result Fast-forward

New HEAD hello world [45d0158]

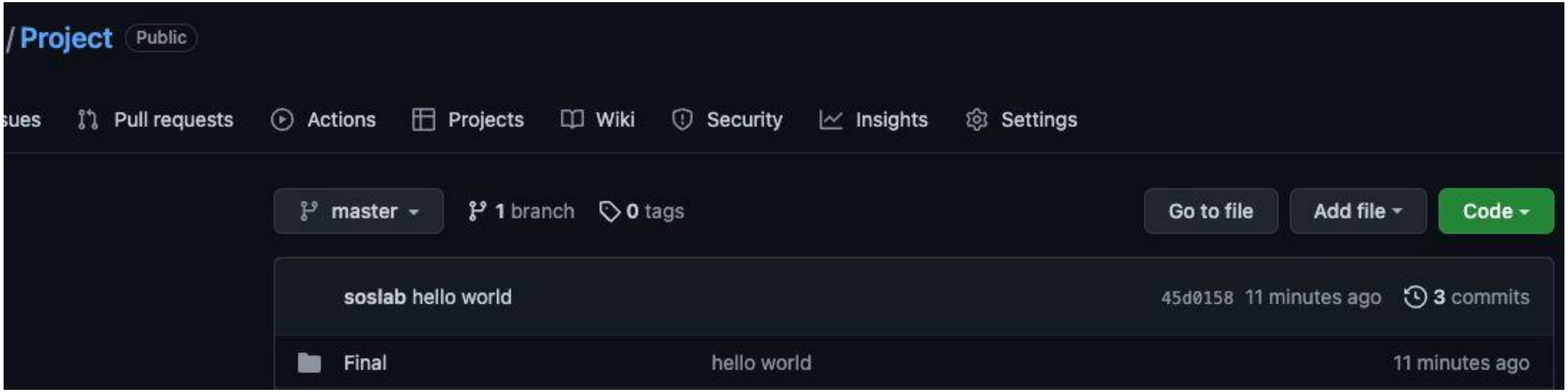
Some update

Merge input

f828d4f: create main ([redacted] on 2022-09-23 21:31:53)

45d0158: hello world (soslab on 2022-09-23 22:08:45)

GitHub



soslab hello world 45d0158 11 minutes ago 3 commits

Final hello world 11 minutes ago

Notice

- Check your group list

  https://docs.google.com/spreadsheets/d/11dr1xjKxa2Hc-jNssPWi1qvPwOzehsFc6w2ylzdq-Pw/edit?usp=sharing

- Send your GitHub link and contact information via Google form!

  https://forms.gle/sZbMwafigtvh64Us8