School of Computer Science Engineering and Technology

Course- BTech
Course Code- CSET205
Year-2022
Date-15/08/2022

Type- Core
Course Name-Software Engineering
Semester- Odd
Batch- 2021-2025

Lab Assignment No. 3 - Working with the Eclipse IDE and GitHub together

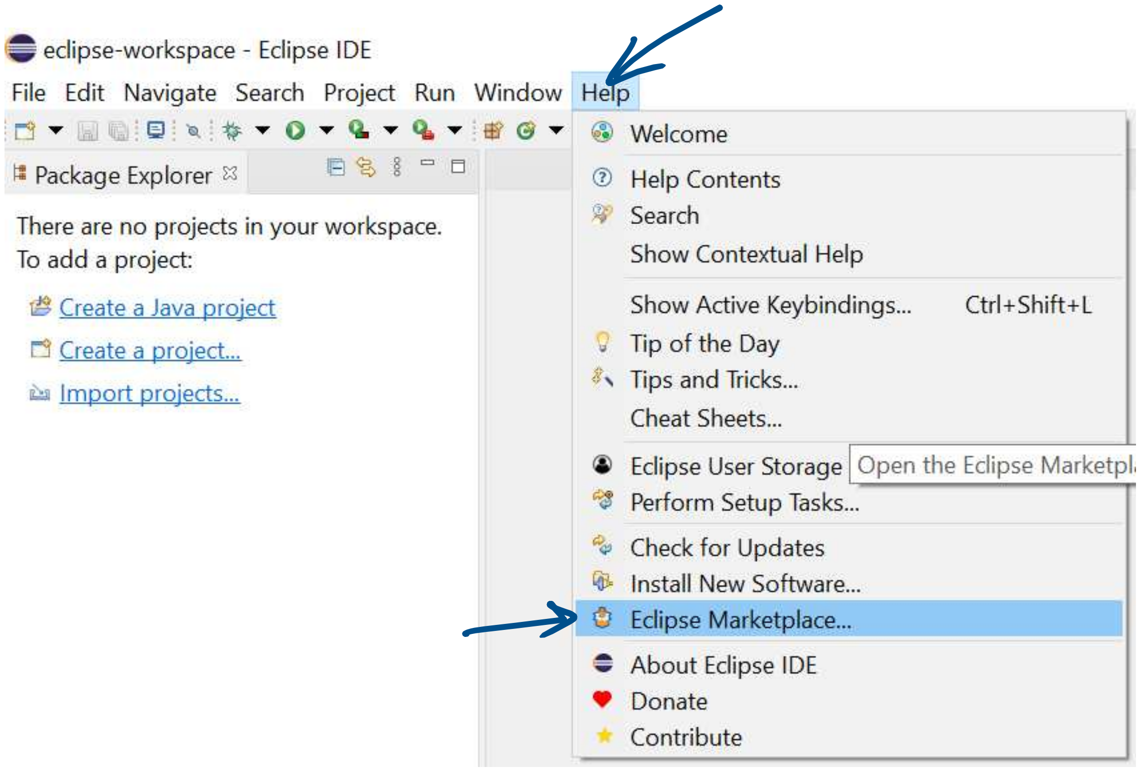
CO Mapping

Exp. No.	Name	CO1	CO2	CO3
3	Working with the Eclipse IDE and GitHub		√	

This lab enables you to understand how software developers in the industries are working across the globe, in different teams, and can develop one product together via their IDE (e.g., Eclipse) and Repository (e.g., GitHub) integrated together.

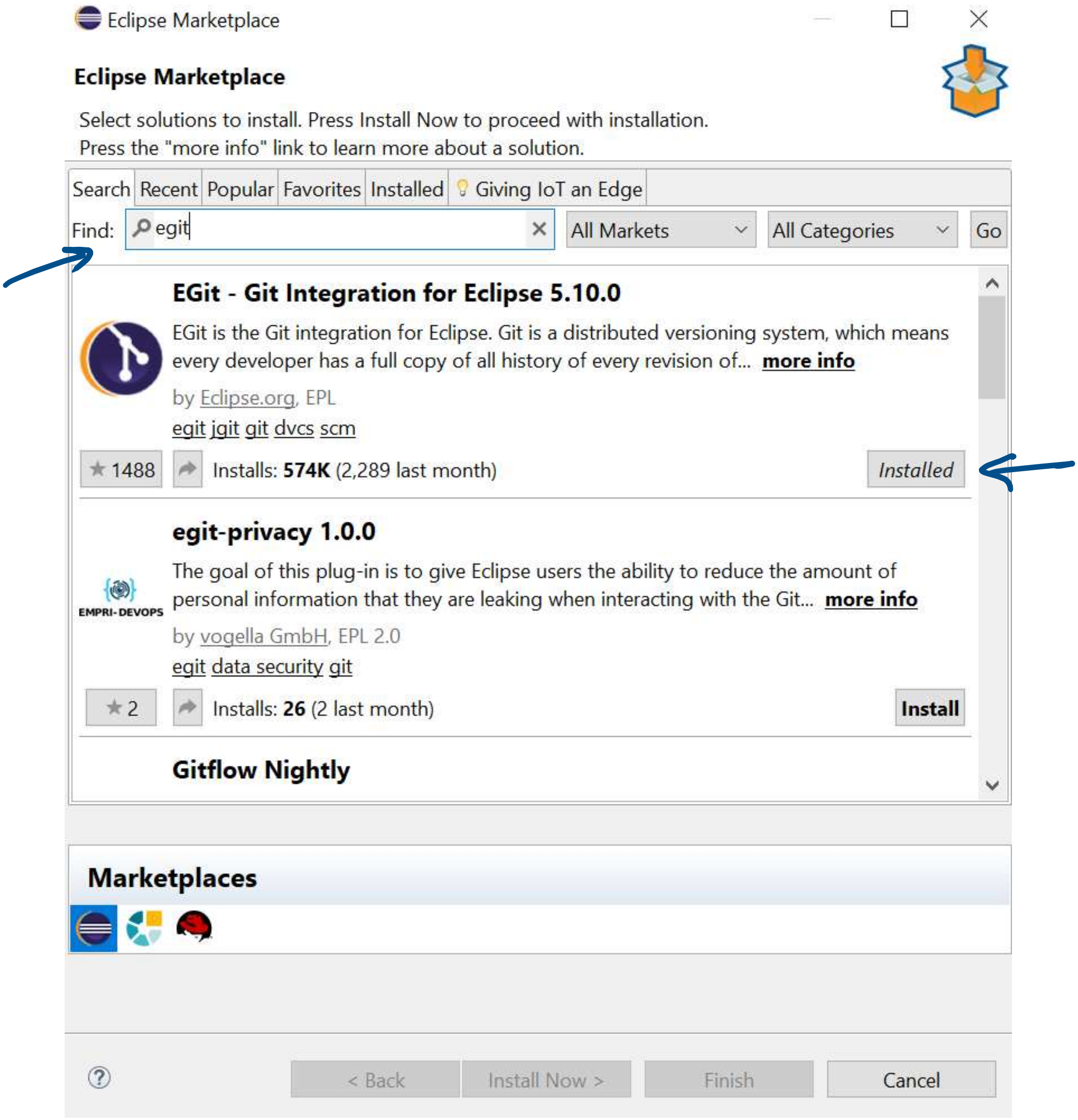
Follow the steps below to install the Git plugin

1. Open Eclipse
2. Click on Help → Eclipse Marketplace



There are no projects in your workspace.
To add a project:

3. In the search box, type ***egit,*** and look at the results. If it shows ***installed*** for EGit (like in the snapshot below), then **go to step 7**; else, go to the next step.



Eclipse Marketplace

Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: egit

EGit - Git Integration for Eclipse 5.10.0

EGit is the Git integration for Eclipse. Git is a distributed versioning system, which means every developer has a full copy of all history of every revision of... **more info**

by Eclipse.org, EPL

egit jgit git dvcs scm

★ 1488 Installs: **574K** (2,289 last month) *Installed*

egit-privacy 1.0.0

The goal of this plug-in is to give Eclipse users the ability to reduce the amount of personal information that they are leaking when interacting with the Git... **more info**

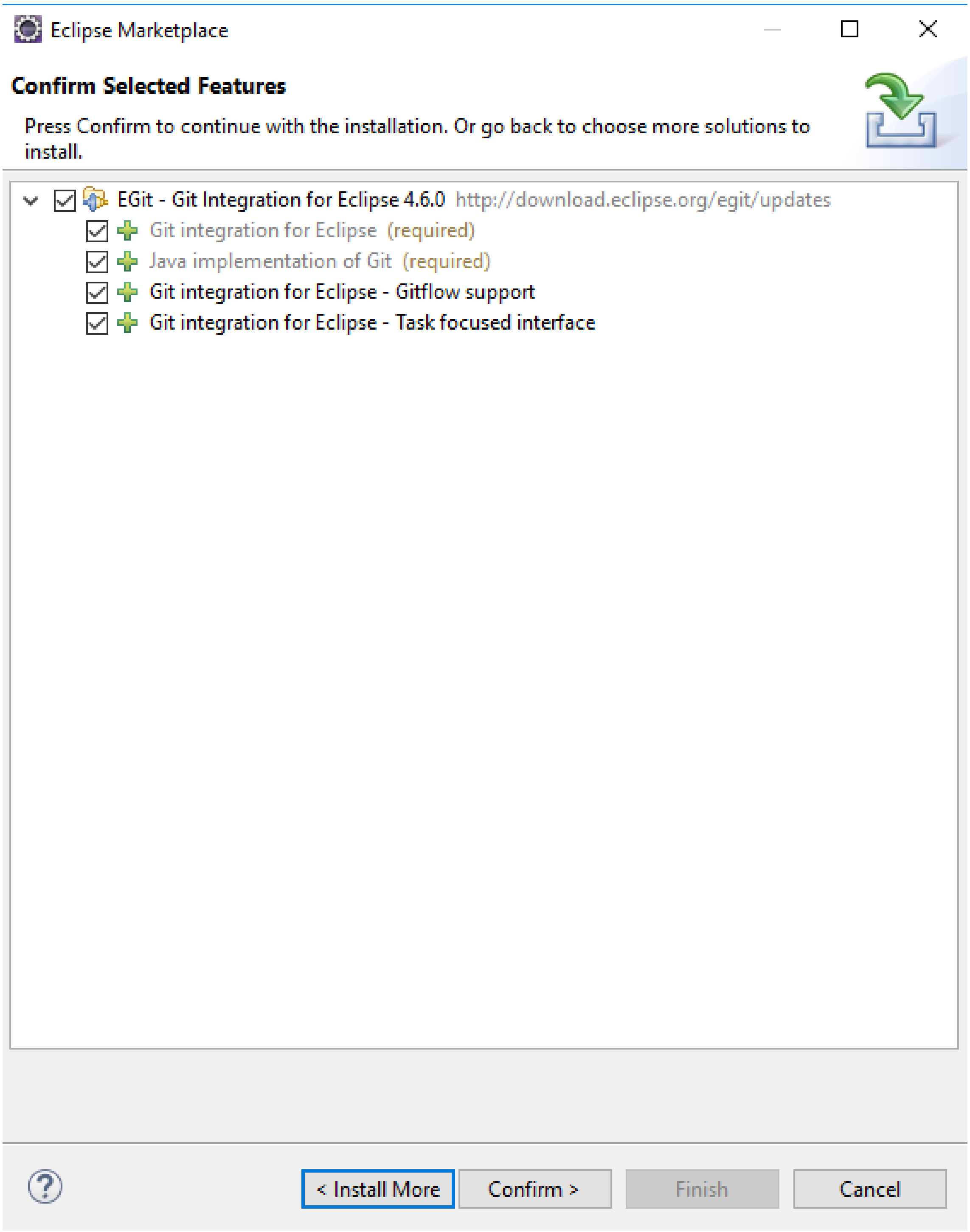
by vogella GmbH, EPL 2.0

egit data security git

★ 2 Installs: **26** (2 last month) **Install**

Gitflow Nightly

4. Click on the **Install** button. You will see the Eclipse marketplace window with the EGit installation option. **Select All** and click Confirm to proceed with the installation.



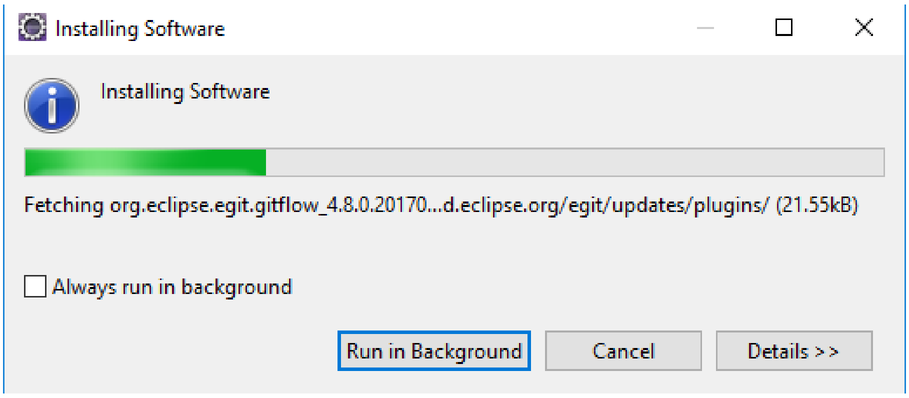
Eclipse Marketplace

Confirm Selected Features

Press Confirm to continue with the installation. Or go back to choose more solutions to install.

- ☑ EGit - Git Integration for Eclipse 4.6.0 http://download.eclipse.org/egit/updates
  - ☑ Git integration for Eclipse (required)
  - ☑ Java implementation of Git (required)
  - ☑ Git integration for Eclipse - Gitflow support
  - ☑ Git integration for Eclipse - Task focused interface

5. **Accept** the license terms click Finish. You will see the installation progress.



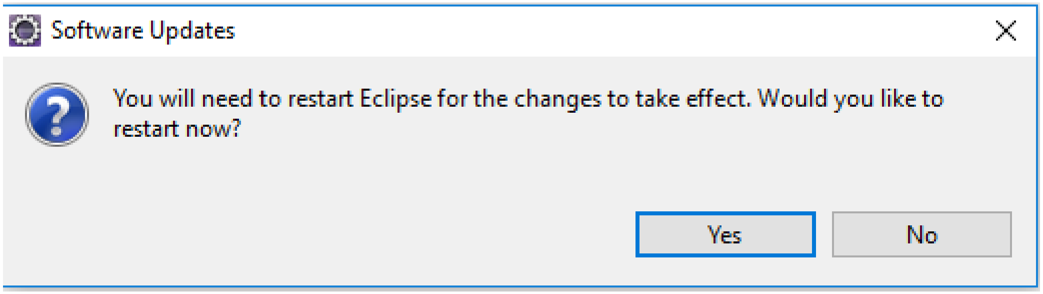
Installing Software

Installing Software

Fetching org.eclipse.egit.gitflow_4.8.0.20170...d.eclipse.org/egit/updates/plugins/ (21.55kB)

□ Always run in background

6. After installation completes, click yes to restart Eclipse.



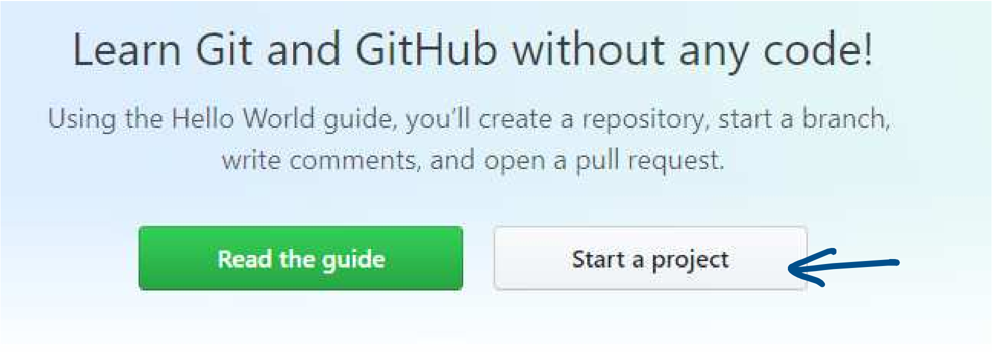
Software Updates

You will need to restart Eclipse for the changes to take effect. Would you like to restart now?

Creating a GitHub account and repository

7. If you haven't created your GitHub account, please create it from https://github.com else skip this step

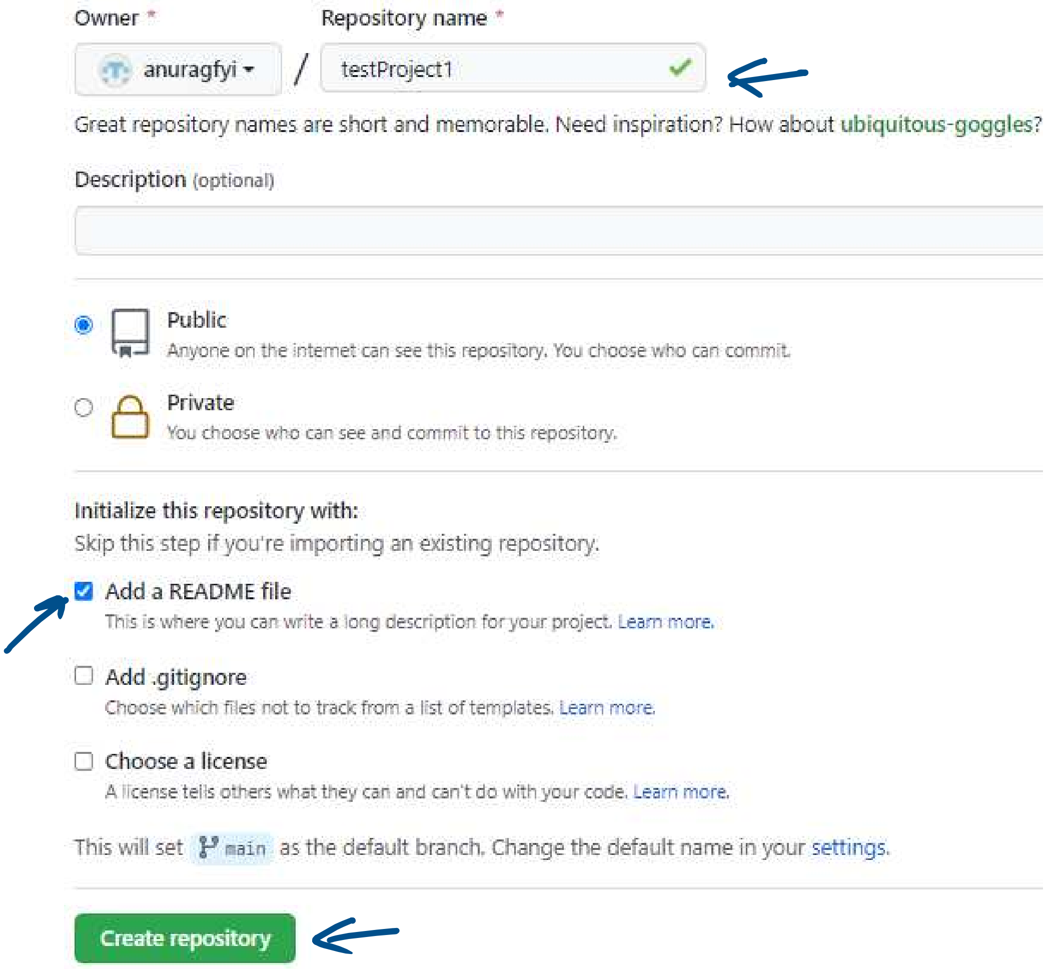
8. After the creation of the account, log in to your account, click on **Start a project**.



Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

9. Input your repository name → check **initialize this repository with a Readme** → click **create repository**.



Owner *

anuragfyi ▾ / Repository name * testProject1 ✓

Great repository names are short and memorable. Need inspiration? How about ubiquitous-goggles?

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.

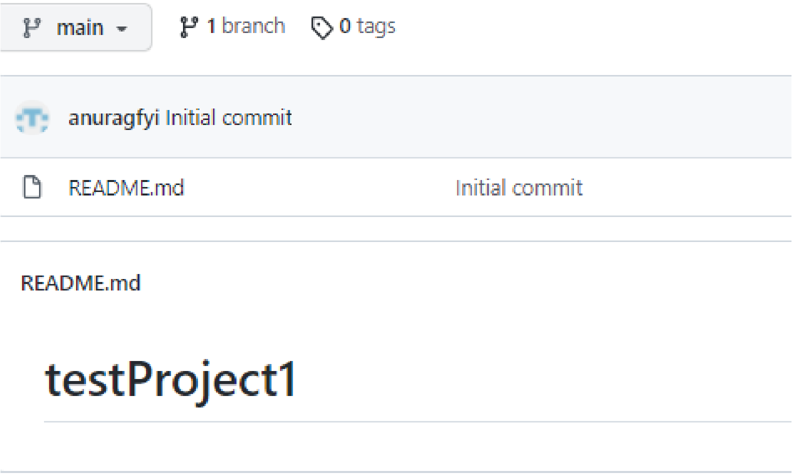
☑ Add a README file
This is where you can write a long description for your project. Learn more.

□ 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.

10. You will see the window below:



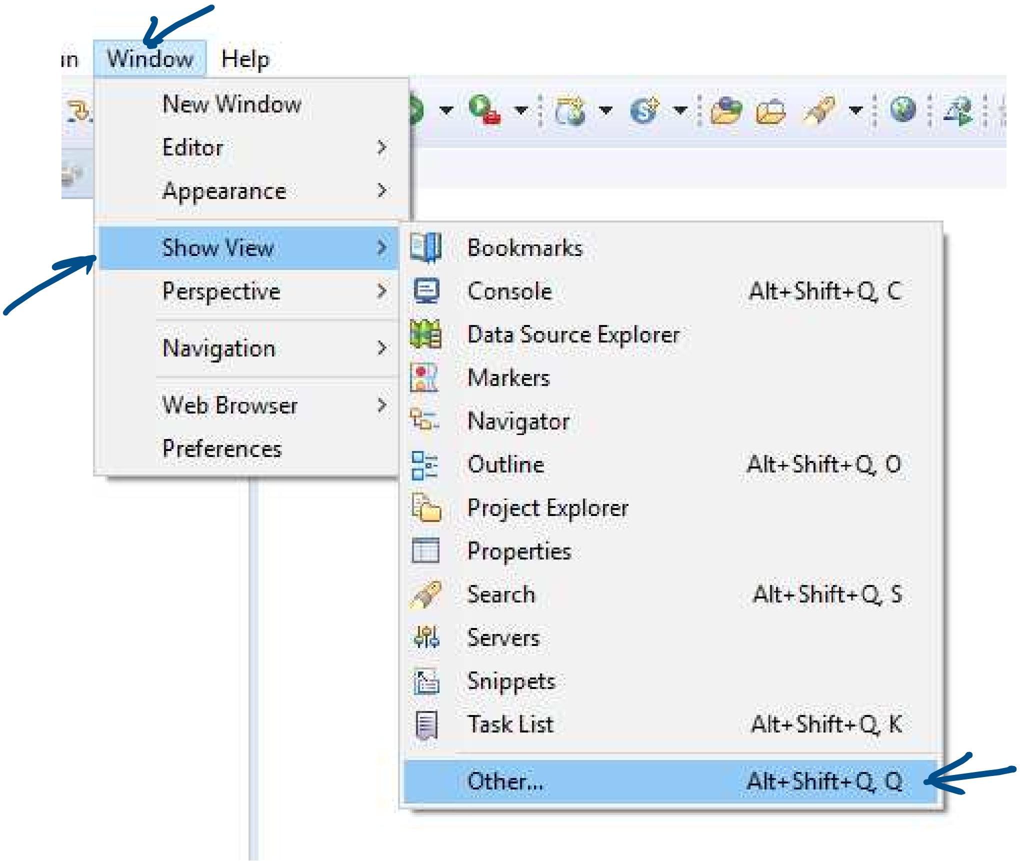
anuragfyi Initial commit

README.md	Initial commit

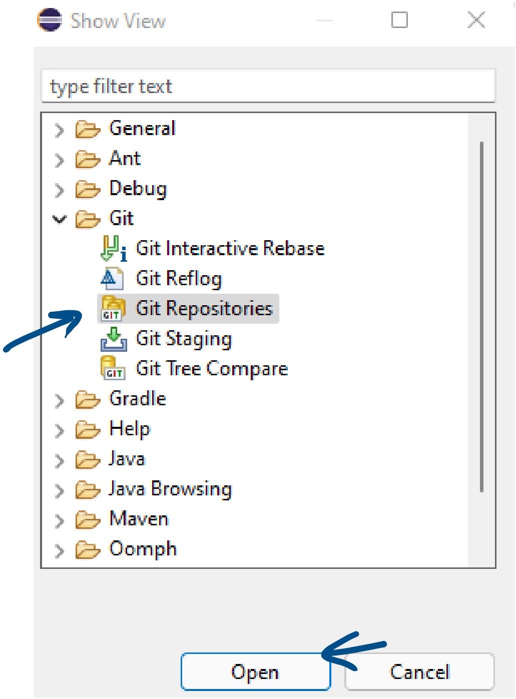
README.md

testProject1

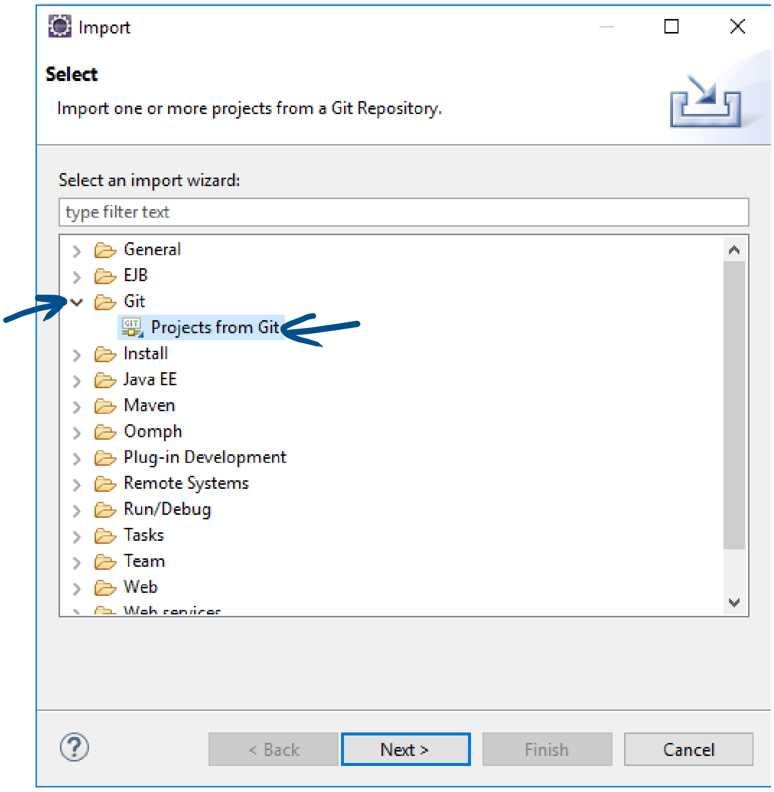
11.Go back to Eclipse and click on the **Window tab → Show view → Other**



12.In the Show view windows, expand the Git option, select Git Repositories and click on the OK button.



13.Click on the **File** tab on Eclipse and Goto **import** option. Under the Import window, expand **Git** and select **Projects from Git**. Click the Next button.



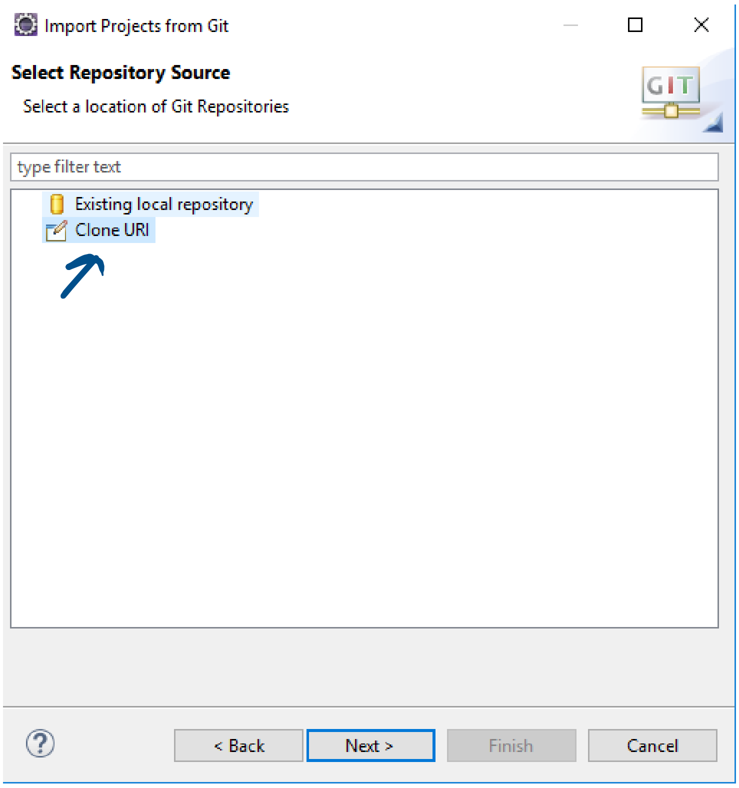
Import

Select

Import one or more projects from a Git Repository.

Select an import wizard:

14.Select **clone URI** and click the **Next** button.

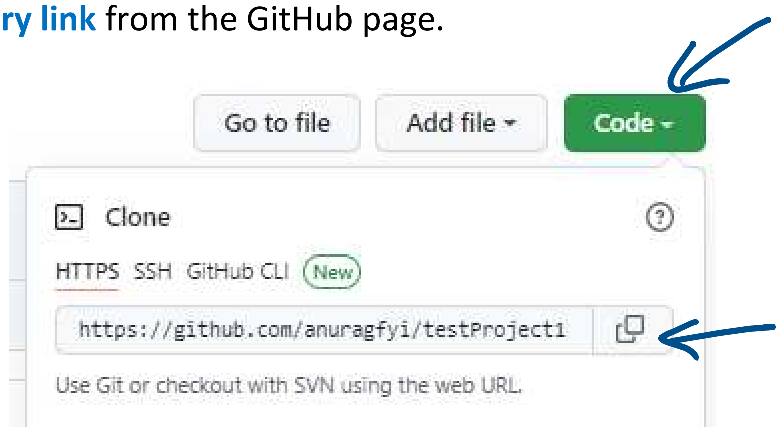


Import Projects from Git

Select Repository Source

Select a location of Git Repositories

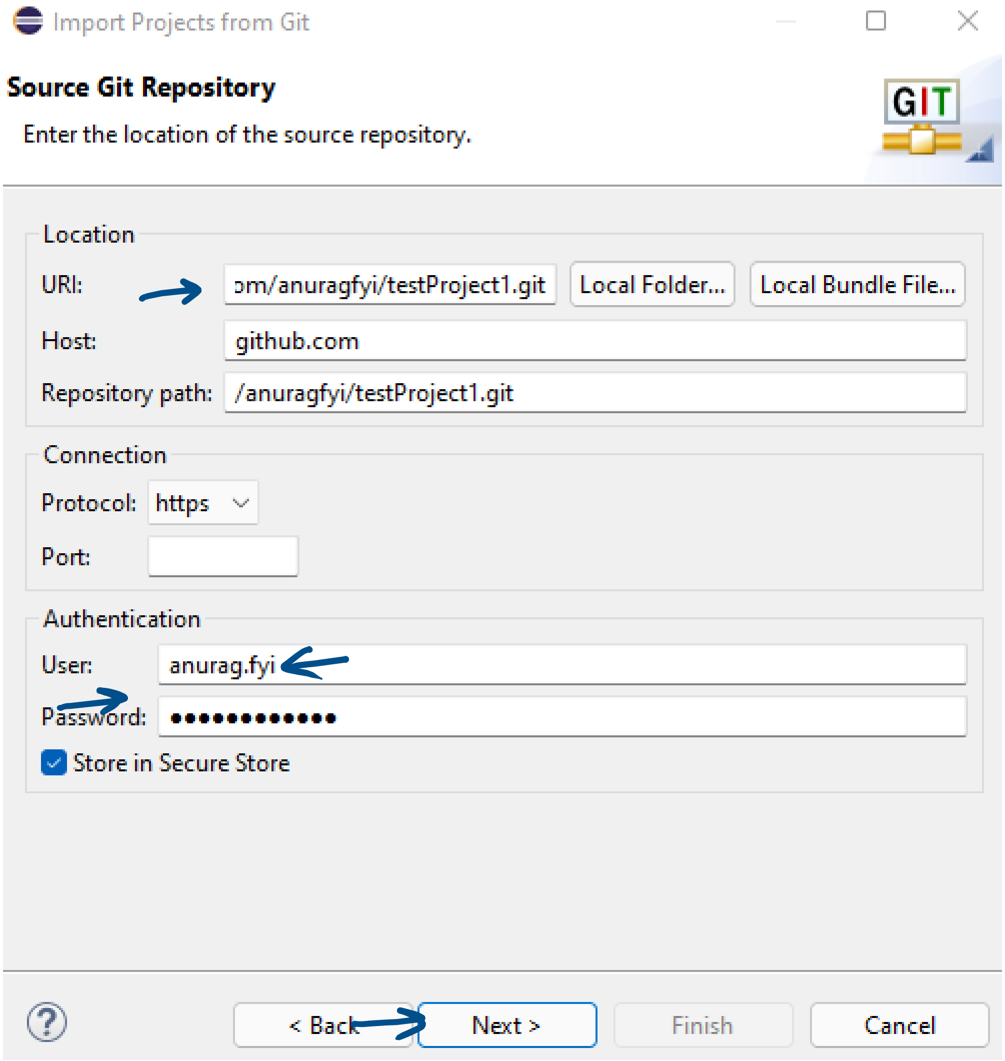
15. Go to the GitHub page again → **Click on Code → Click on button to copy your repository link** from the GitHub page.



https://github.com/anuragfyi/testProject1

Use Git or checkout with SVN using the web URL.

16. Paste the link in the URI window → type host as github.com → select HTTPS protocol → enter your GitHub credentials (don't click on the store in Secure Store checkbox – **keep it unchecked on public machines**)



Source Git Repository

Enter the location of the source repository.

Location

URI: om/anuragfyi/testProject1.git Local Folder... Local Bundle File...

Host: github.com

Repository path: /anuragfyi/testProject1.git

Connection

Protocol: https

Port:

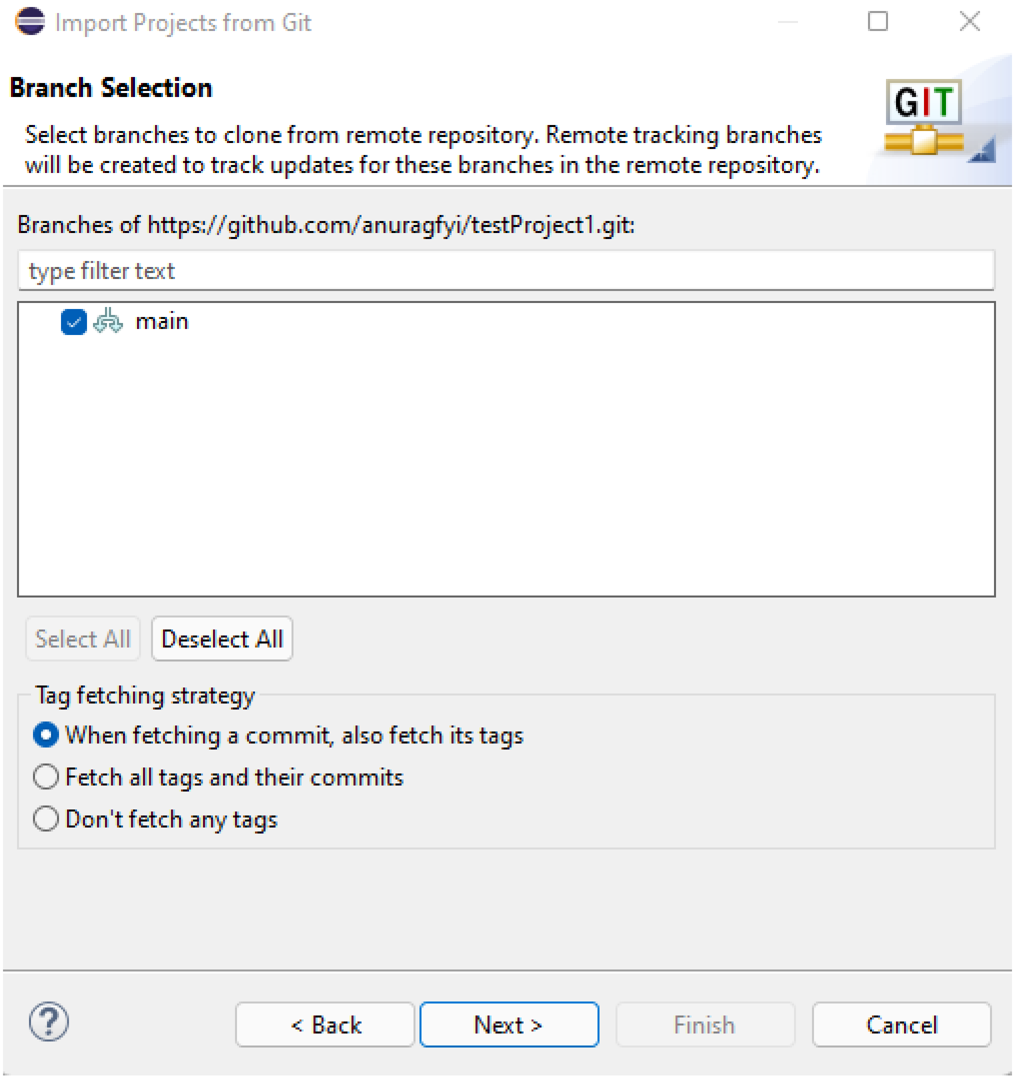
Authentication

User: anurag.fyi

Password: ●●●●●●●●●●●●

☑ Store in Secure Store

17. In branch selection windows, by default Master branch (main) is selected. Click next.



Import Projects from Git

Branch Selection

Select branches to clone from remote repository. Remote tracking branches will be created to track updates for these branches in the remote repository.

Branches of https://github.com/anuragfyi/testProject1.git:

type filter text

☑ main

Select All | Deselect All

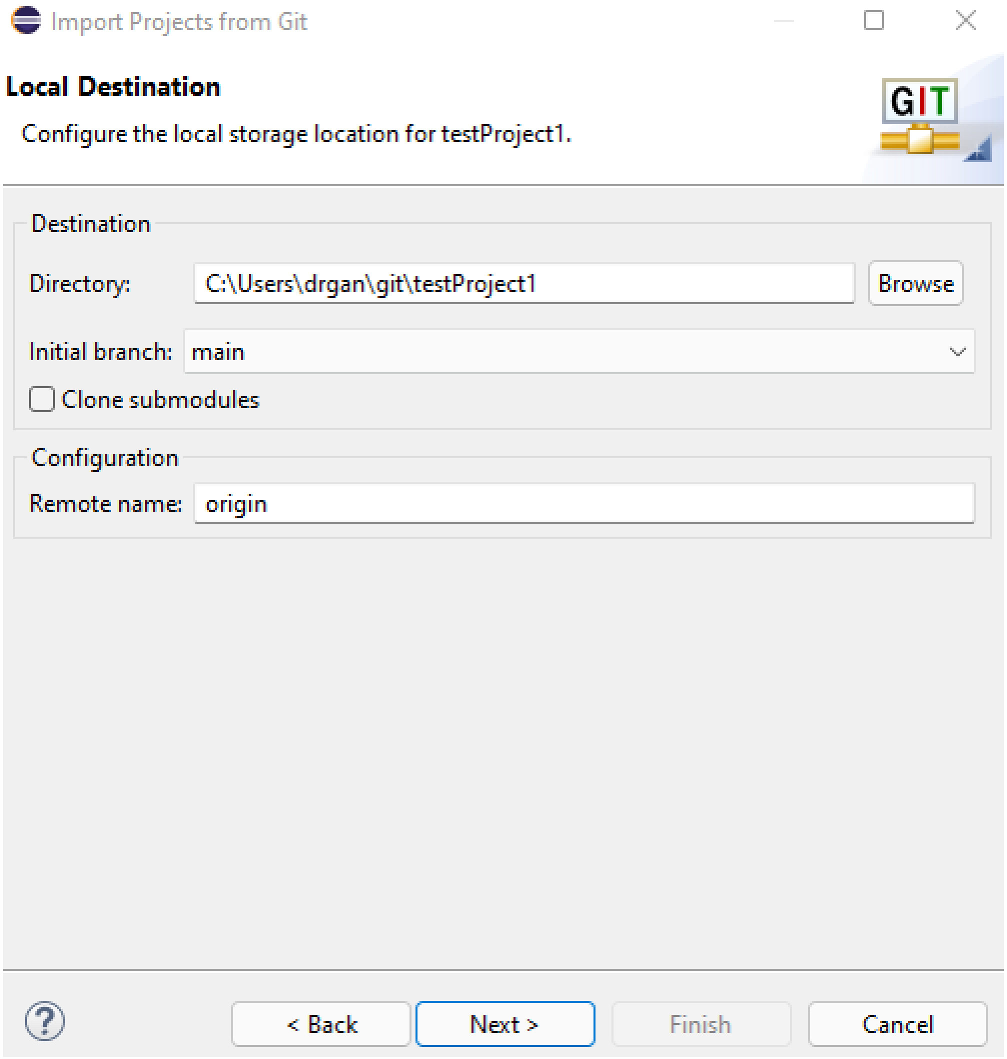
Tag fetching strategy

● When fetching a commit, also fetch its tags

○ Fetch all tags and their commits

○ Don't fetch any tags

18. In the Local Directory window, select the folder (if you want), else click next.



Import Projects from Git

Local Destination

Configure the local storage location for testProject1.

Destination

Directory: C:\Users\drgan\git\testProject1

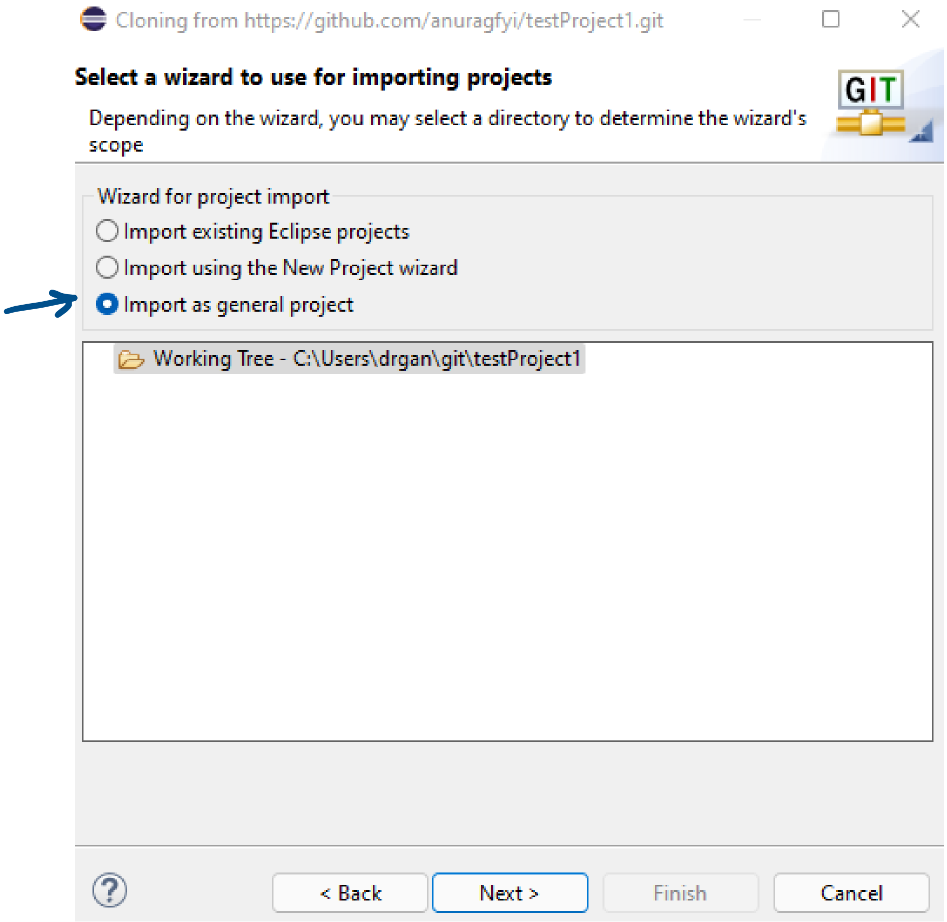
Initial branch: main

☐ Clone submodules

Configuration

Remote name: origin

19. In the project wizard window, select the **general project** option and click next.



Cloning from https://github.com/anuragfyi/testProject1.git

Select a wizard to use for importing projects

Depending on the wizard, you may select a directory to determine the wizard's scope

Wizard for project import

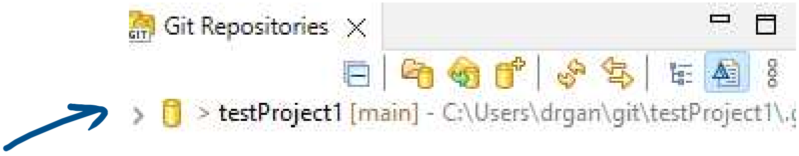
○ Import existing Eclipse projects

○ Import using the New Project wizard

● Import as general project

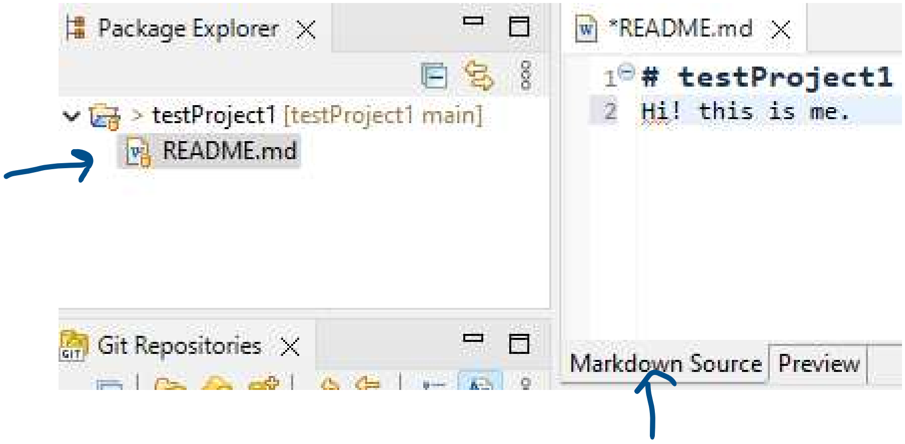
Working Tree - C:\Users\drgan\git\testProject1

20. Click the Finish button on the next screen. You will see that your repository is cloned in Eclipse.



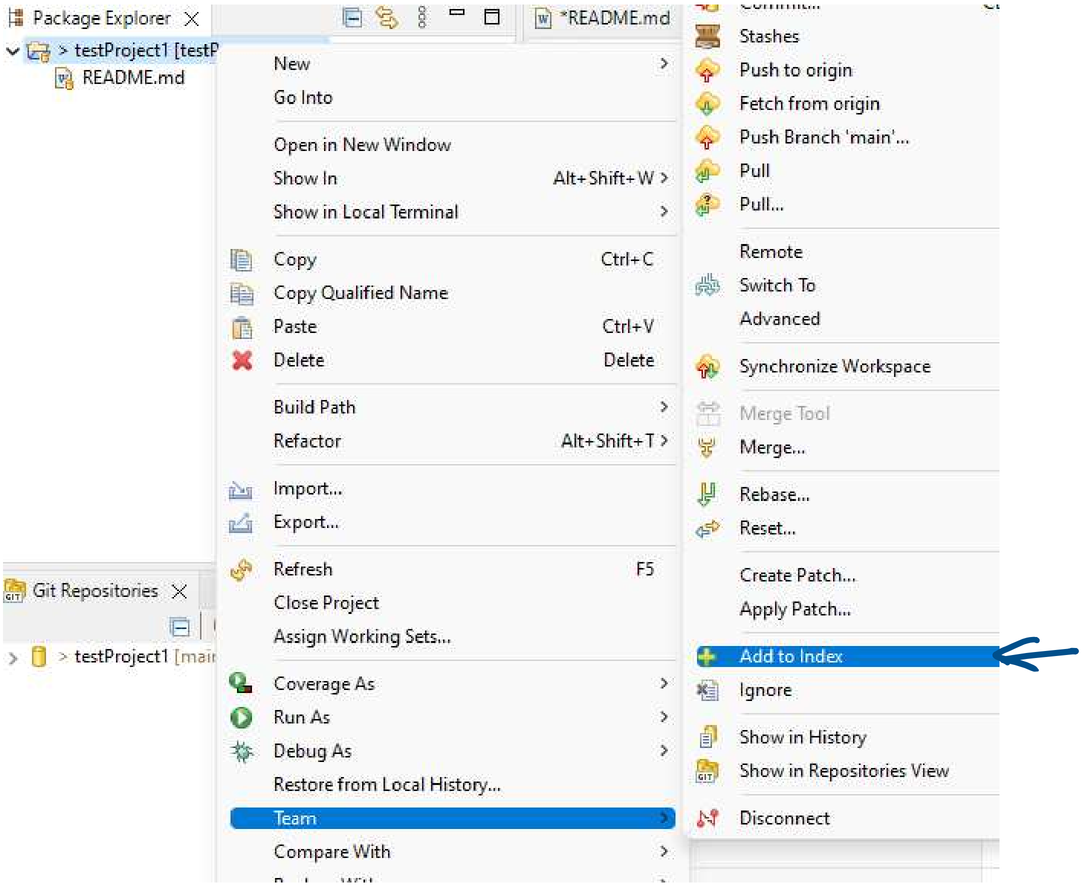
> testProject1 [main] - C:\Users\drgan\git\testProject1\.g

21. Expand and open the readme file in Eclipse and make some changes to it and save the file. Before making changes, make sure to click on **markdown source** to enable edit mode.

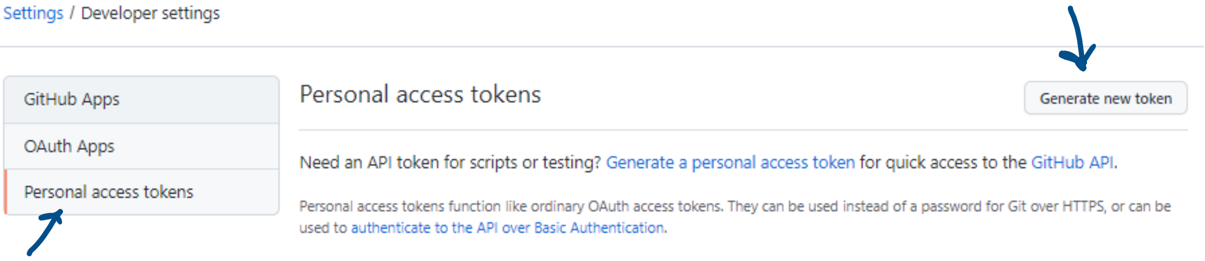


```
# testProject1
Hi! this is me.
```

22. Right-click on your project → **go to Team option → click on Add to Index** option. This will mark the new files as ready to be moved to your repository.



23.Go to Github once again. Go to **Settings > Developer settings > Personal access tokens** to create a Personal access token. When Eclipse asks for the password at the next step, enter the personal access token.



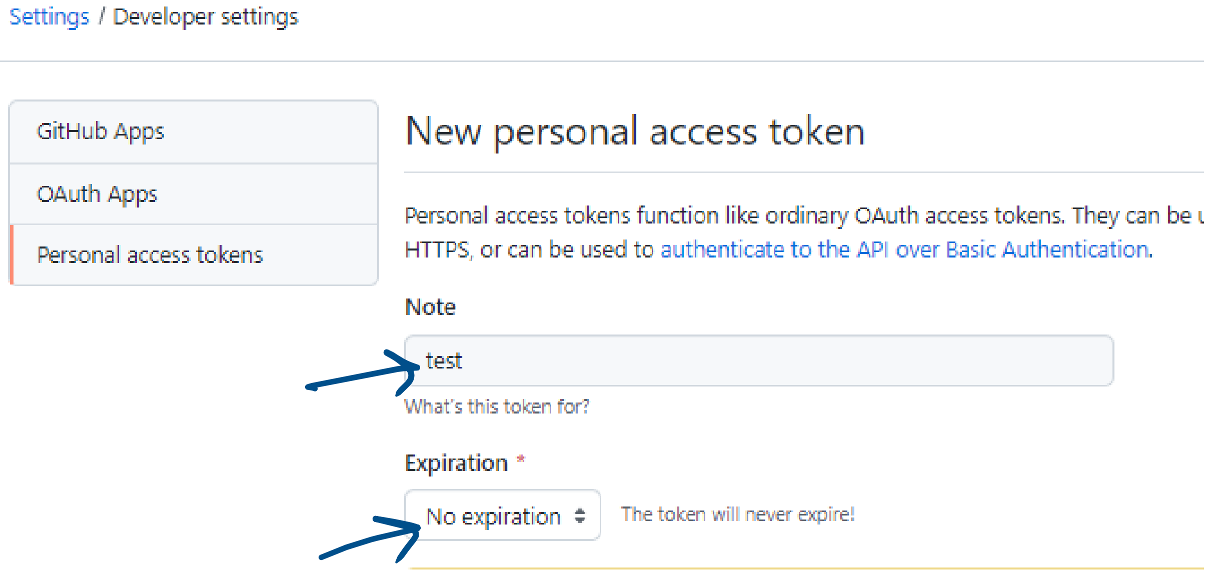
Settings / Developer settings

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.

Give it a name and set the expiration as **No expiration** and **check all the check boxes** below (not shown in the image below). Click on **Generate token** button at the bottom. **Copy your personal access token.**



Settings / Developer settings

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be u
HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

test

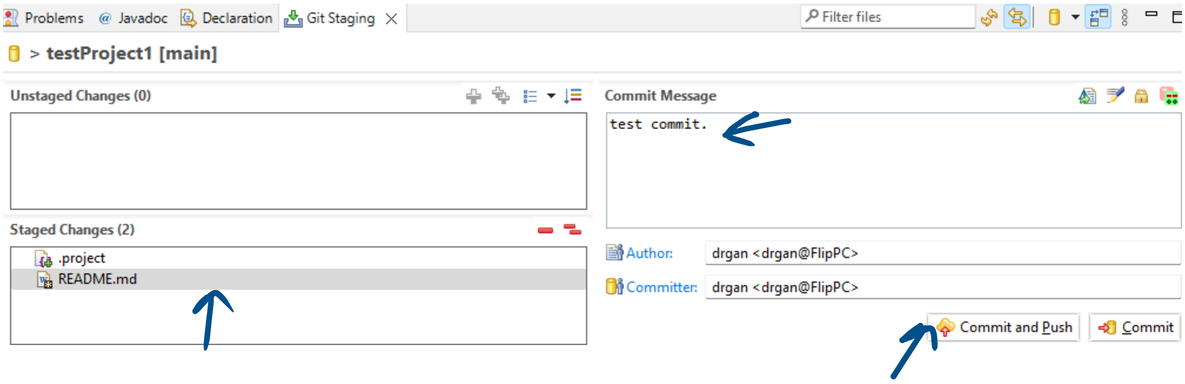
What's this token for?

Expiration *

No expiration

The token will never expire!

24. Right-click on your project → **go to Team option → click on Commit → add a commit message** and click on Commit and Push button if you want changes to be pushed to GitHub.



> testProject1 [main]

Staged Changes (2)

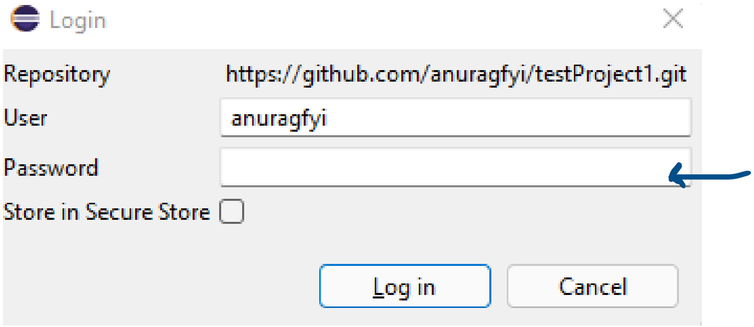
- .project
- README.md

```
test commit.
```

Author: drgan <drgan@FlipPC>

Committer: drgan <drgan@FlipPC>

25. Eclipse will ask for your username and password. For username enter your Git username and for password enter the token you copied in the last step. Click on **Login button**.



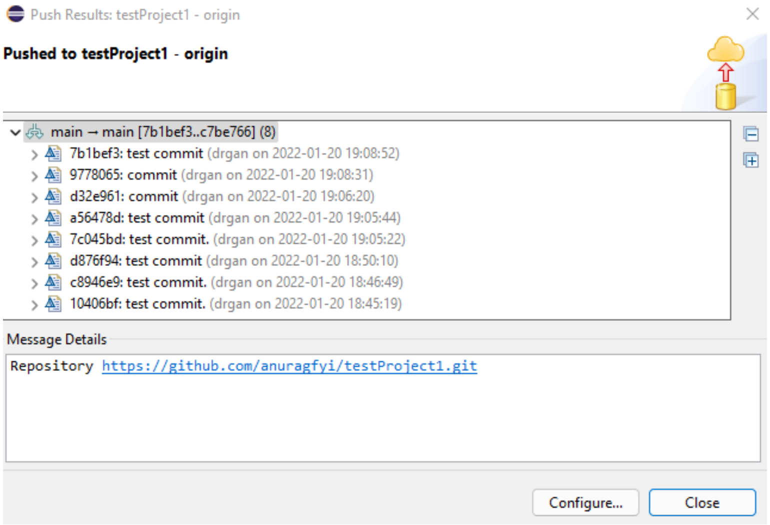
Repository https://github.com/anuragfyi/testProject1.git

User anuragfyi

Password

Store in Secure Store □

26. You will see the push results. Click **close** button.



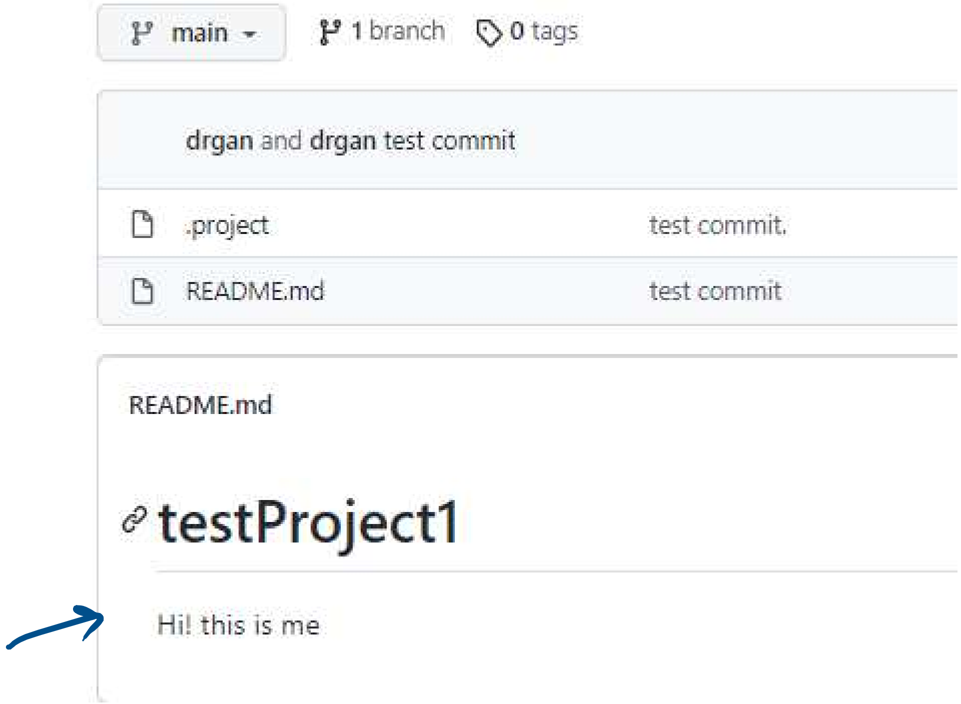
Pushed to testProject1 - origin

```
main → main [7b1bef3..c7be766] (8)
  7b1bef3: test commit (drgan on 2022-01-20 19:08:52)
  9778065: commit (drgan on 2022-01-20 19:08:31)
  d32e961: commit (drgan on 2022-01-20 19:06:20)
  a56478d: test commit (drgan on 2022-01-20 19:05:44)
  7c045bd: test commit. (drgan on 2022-01-20 19:05:22)
  d876f94: test commit (drgan on 2022-01-20 18:50:10)
  c8946e9: test commit. (drgan on 2022-01-20 18:46:49)
  10406bf: test commit. (drgan on 2022-01-20 18:45:19)
```

Message Details

```
Repository https://github.com/anuragfyi/testProject1.git
```

27. Go to the GitHub repository page opened in the browser → Refresh the page, and you will see changes in the Git file contents **(as it was updated and pushed from Eclipse IDE)**



drgan and drgan test commit

.project	test commit.
README.md	test commit

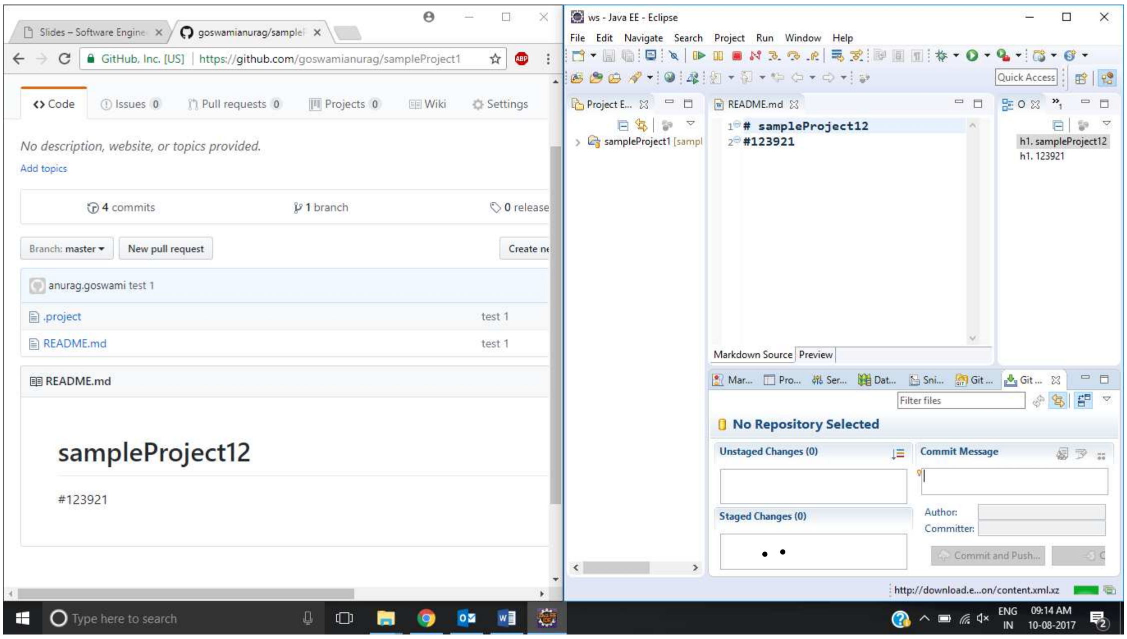
README.md

testProject1

Hi! this is me

28. Congratulations! You have successfully connected/integrated your Git repository with Eclipse IDE to work on the projects.

29. Take a snapshot of your Git repository and eclipse windows together (to show both the changes in one image) as shown in the example below:



```
# sampleProject12
#123921
```

Submit the snapshot from the step# 29 above and your original ‘Readme’ file from Git Repository (total two files) as your LMS lab task.

Full tutorial: https://wiki.eclipse.org/EGit/User_Guide