

Git and Github definition

Git: Git is a distributed version control system for tracking changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

GitHub: GitHub is a web-based Git repository hosting service, which offers all of the distributed revision control and source code management (SCM) functionality of Git as well as adding its own features.

<i>S.No.</i>	<i>Git</i>	<i>GitHub</i>
1.	Git is a software.	GitHub is a service.
2.	Git is a command-line tool	GitHub is a graphical user interface
3.	Git is installed locally on the system	GitHub is hosted on the web
4.	Git is maintained by linux.	GitHub is maintained by Microsoft.
5.	Git is focused on version control and code sharing.	GitHub is focused on centralized source code hosting.
6.	Git is a version control system to manage source code history.	GitHub is a hosting service for Git repositories.

S.No.	Git	GitHub
7.	Git was first released in 2005.	GitHub was launched in 2008.
8.	Git has no user management feature.	GitHub has a built-in user management feature.
9.	Git is open-source licensed.	GitHub includes a free-tier and pay-for-use tier.
10.	Git has minimal external tool configuration.	GitHub has an active marketplace for tool integration.
11.	Git provides a Desktop interface named Git Gui.	GitHub provides a Desktop interface named GitHub Desktop.
12.	Git competes with CVS, Azure DevOps Server, Subversion, Mercurial, etc.	GitHub competes with GitLab, Git Bucket, AWS Code Commit, etc

GitHub

Create a account

Type a user name, your email address, and a password
Choose Sign up for GitHub, and then follow the instructions.

Steps 1

Go to <https://github.com/join> in a web browser. You can use any web browser on your computer.



2

Enter your personal details. In addition to creating a username and entering an email address, you'll also have to create a password
Create your personal account

A screenshot of the GitHub account creation form. The form is titled 'Create your personal account' and contains three input fields: 'Username *', 'Email address *', and 'Password *'. Each field has a green checkmark next to it, indicating that the entered information is valid. The 'Username' field contains 'wikihowsherman', the 'Email address' field contains 'wikihowsherman@gmail.com', and the 'Password' field contains a series of dots. Below the password field, there is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)'. At the bottom of the form, there is a 'Verify account' button and a large green checkmark.

3

Click the green **Create an account** button. It's below the form.

wikihowsherman@gmail.com ✓

We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.

Password *

.....

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

Verify account

✓

By clicking "Create an account" below, you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account-related emails.

Create an account

4

Complete the CAPTCHA puzzle. The instructions vary by puzzle, so just follow the on-screen instructions to confirm that you are a human.

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

Verify account

Touch the arrows
to roll the image

?

By clicking "Create an account" below, you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account-related emails.


5

Click the **Choose** button for your desired plan. Once you select a plan, GitHub will send an email confirmation message to the address you entered. The plan options are:[2]

- **Free:** Unlimited public and private repositories, up to 3 collaborators, issues and bug tracking, and project management tools.

Choose your subscription

For the open source community, there's no wrong choice.



Free


The basics of GitHub for every developer

\$0

per month

Includes:

- ∞ Unlimited public and private repositories
- ✓ 3 collaborators for private repositories
- ✓ Issues and bug tracking
- ✓ Project management



Pro

Pro tools for developers with advanced requirements

\$7

per month
[\(view in PHP\)](#)

Includes:

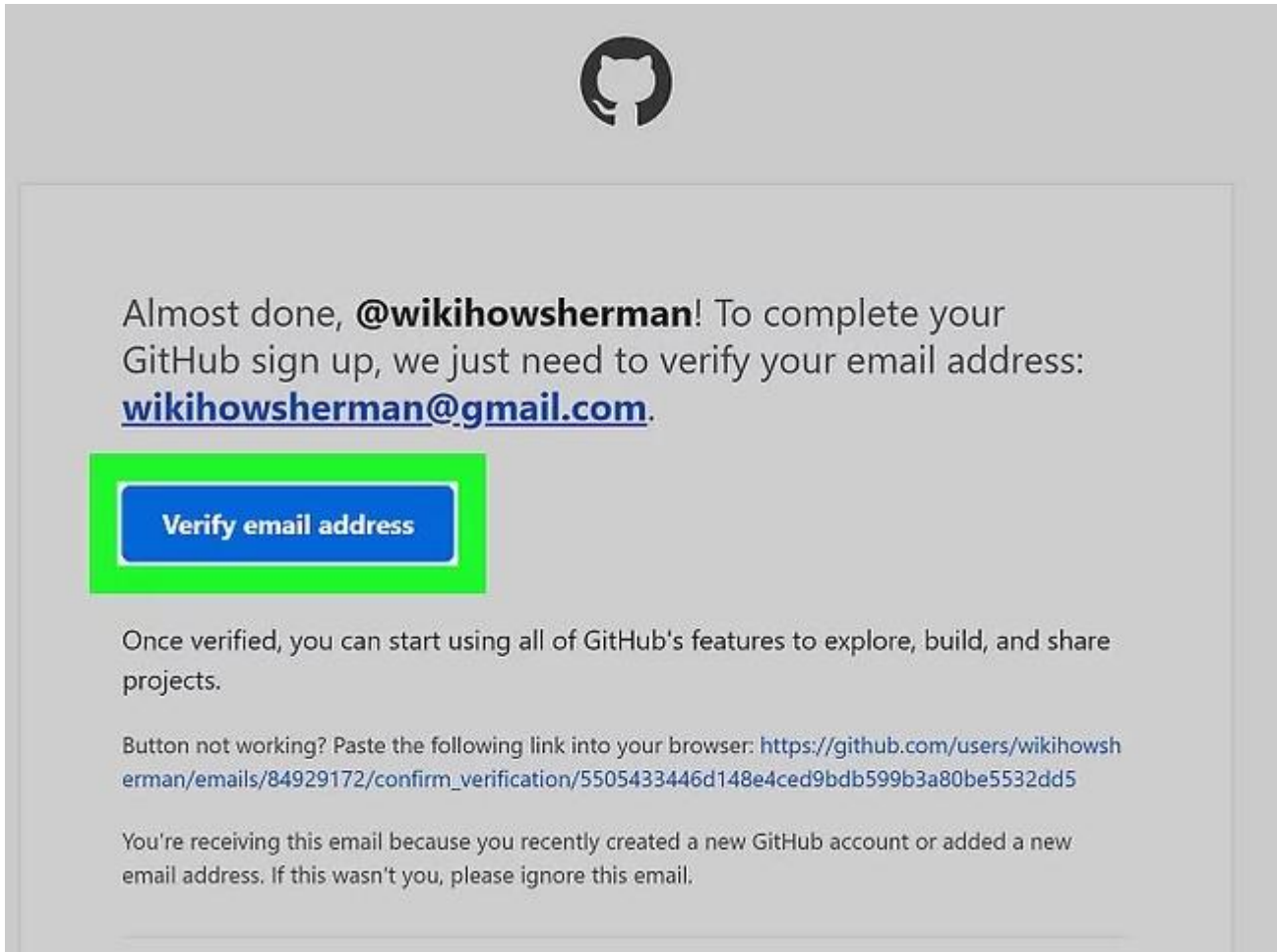
- ∞ Unlimited public and private repositories
- ∞ Unlimited collaborators
- ✓ Issues and bug tracking
- ✓ Project management
- ✓ [Advanced tools and insights](#)

Are you a [student](#)? Get access to the best developer tools for free with the [GitHub Student Developer Pack](#).

☐ **Help me set up an organization next**
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees.
[Learn more about organizations](#)

6

Click the **Verify email address** button in the message from GitHub. This confirms your email address and returns you to the sign-up process.



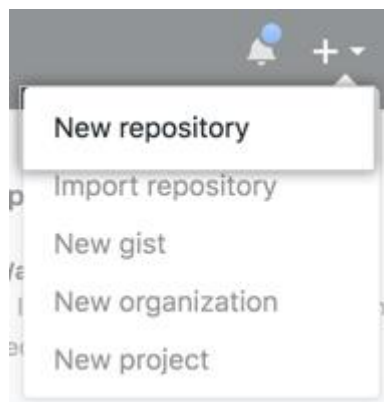
8

Select your preferences and click **Submit**. GitHub displays a quick survey that can help you tailor your experience to match what you're looking for. Once you make your selection, you'll be taken to a screen that allows you to set up your first repository.

- If you want to upgrade your Github account in the future, click the menu at the top-right corner, select **Settings**, and choose **Billing** to view your options.[3]

Create a repository

- In the upper-right corner of any page, use the drop-down menu, and select **New repository**.




- Type a short, memorable name for your repository. For example, "hello-world".


Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

 octocat

Repository name

hello-world 



Great repository names are short and memorable. Need inspiration? How about **potential-eureka**.

Description (optional)

- Optionally, add a description of your repository. For example, "My first repository on GitHub."

Create a new repository

A repository contains all the files for your project, including the revision history.




Owner	Repository name
 octocat ▾	/ hello-world 

Great repository names are short and memorable. Need inspiration? How about **potential-eureka**.

Description (optional)



My first repository on GitHub

Description (optional)

- ☒  **Public**
Anyone can see this repository. You choose who can commit.
- ☐  **Internal**
Octo Corp [enterprise members](#) can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

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

- Select **Initialize this repository with a README**.

- ☒  **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.

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

- ☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾

Add a license: None ▾



Create repository

Click **Create repository**

This will let you immediately clone the repository to your computer.

Add .gitignore: None ▼

Add a license: None ▼



Create repository

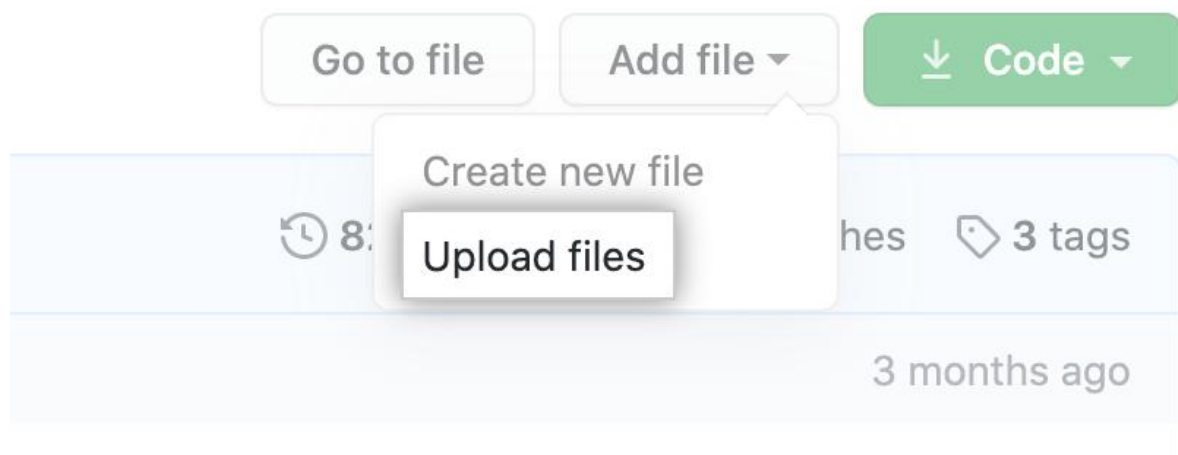
Upload the Project SRC & other project related files on the repo

Adding a file to a repository on GitHub

You can upload multiple files to GitHub at the same time.

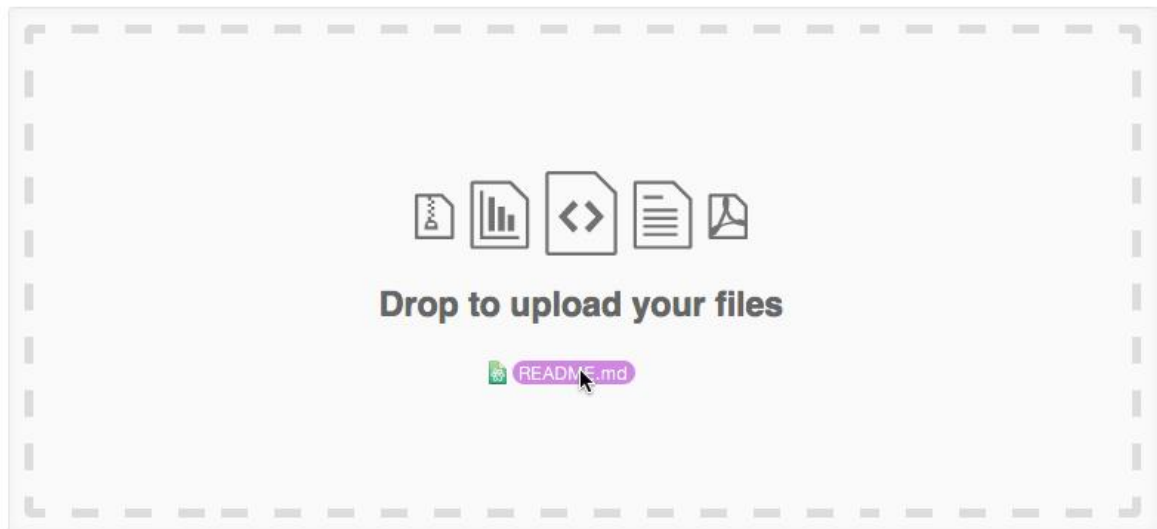
On GitHub.com, navigate to the main page of the repository.

Above the list of files, using the **Add file** drop-down, click **Upload files**.



- Drag and drop the file or folder you'd like to upload to your repository onto the file tree.

octo-repo /



. . "

Commit changes

Update issue_template.md

Add an optional extended description...

- ☐ Commit directly to the `main` branch.
- ☒ Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

octocat-patch-1

Propose new file

Cancel

Click **Commit changes**.

Commit changes

Add files via upload

Add an optional extended description...

☒ Commit directly to the `master` branch

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes
Cancel

Git in Eclipse

Eclipse ships with a plugin called Egit, which provides a fairly-complete interface to Git operations. It's accessed by switching to the Git Perspective (Window > Open Perspective > Other..., and select "Git")

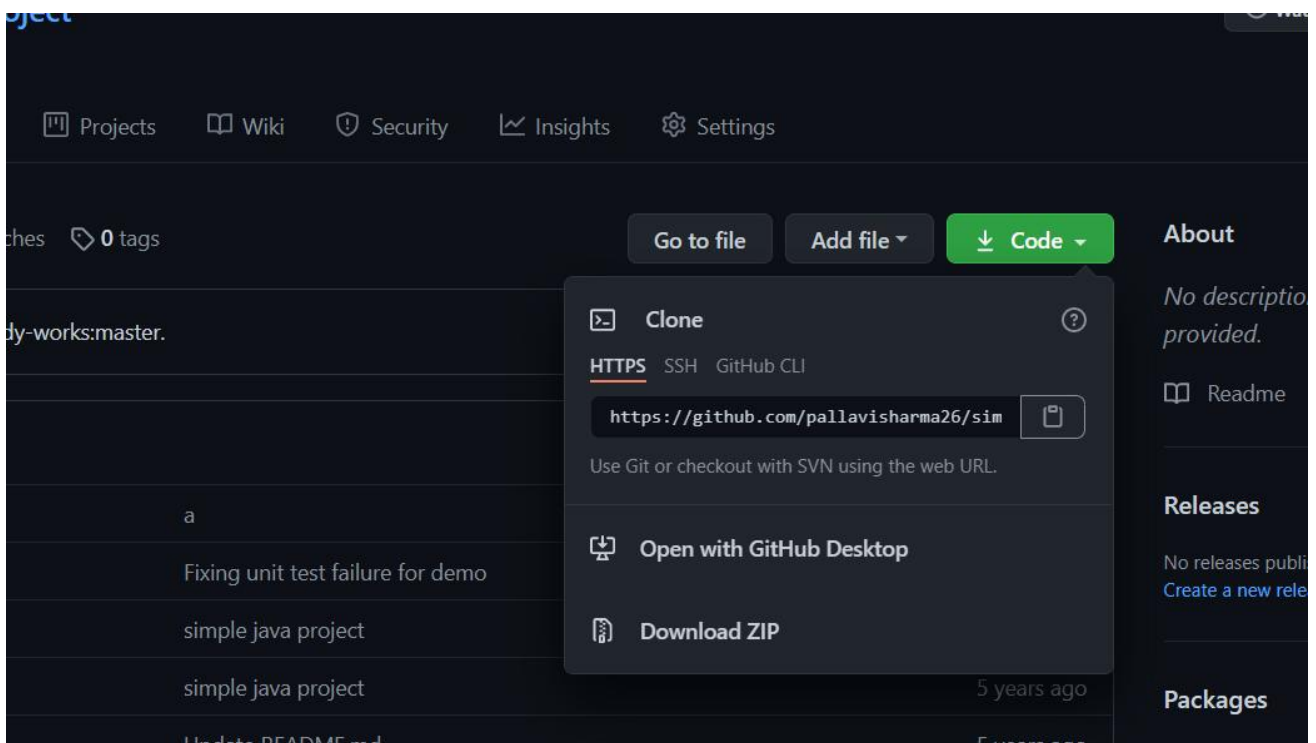
The screenshot shows the Eclipse IDE in the Git perspective. The left sidebar, titled 'Git Repositories', shows the structure of the 'libgit2' repository. It includes branches like 'development' and 'master', tags, and the working directory. The main editor area displays the 'Git Reflog' view, which lists recent commits. The top commit is a merge of pull request #1966, followed by a commit titled 'anmeldung' by Carlos Martín Nieto. The commit message and details are visible in the editor.

Git Perspective

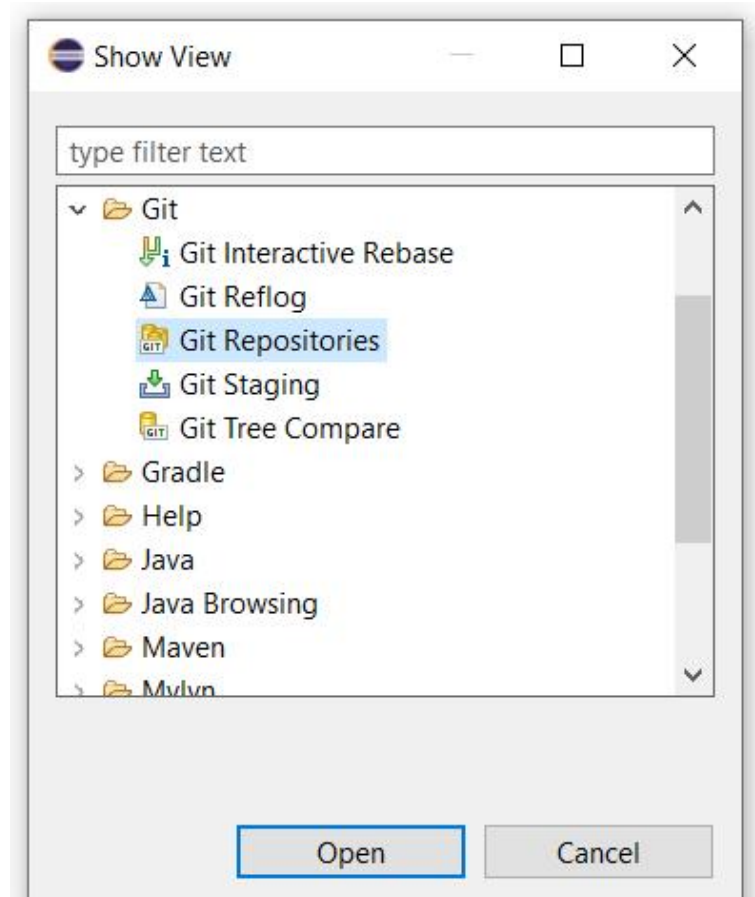
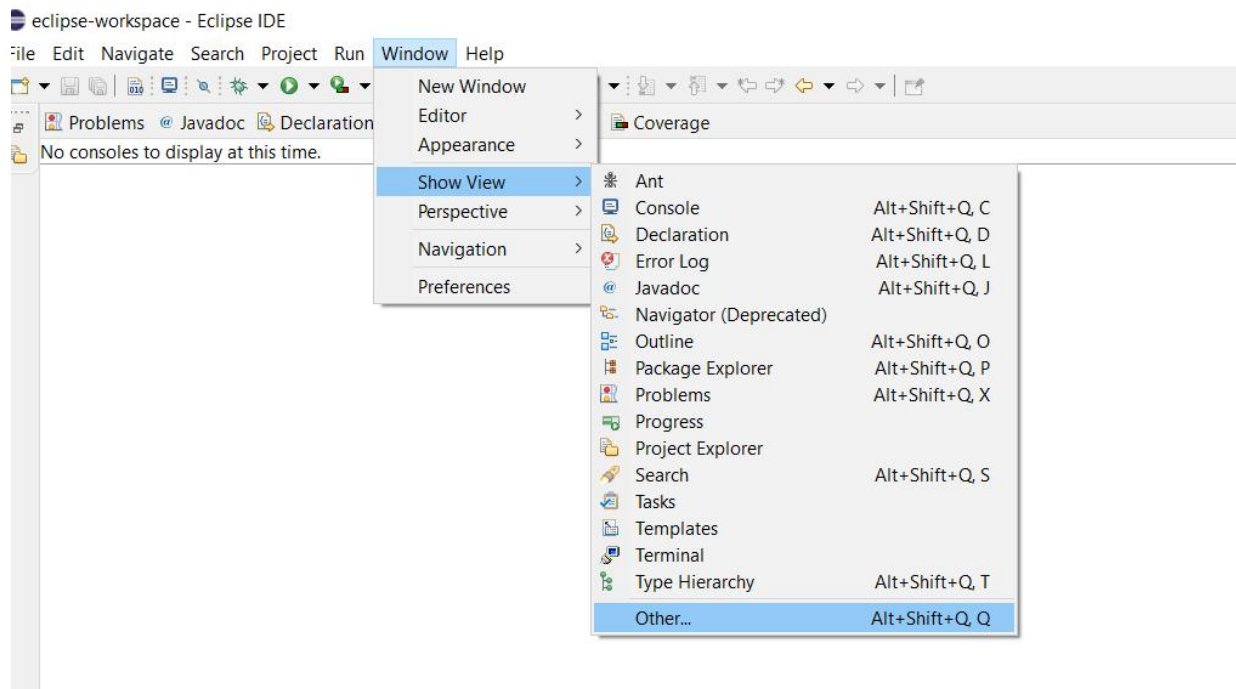
Git perspective is a group of views and editors which is used to work with local and remote Git repositories. allows interactive rebase (i.e. quick editing of commits including their removing, amending, squashing, etc.)

Cloning a project

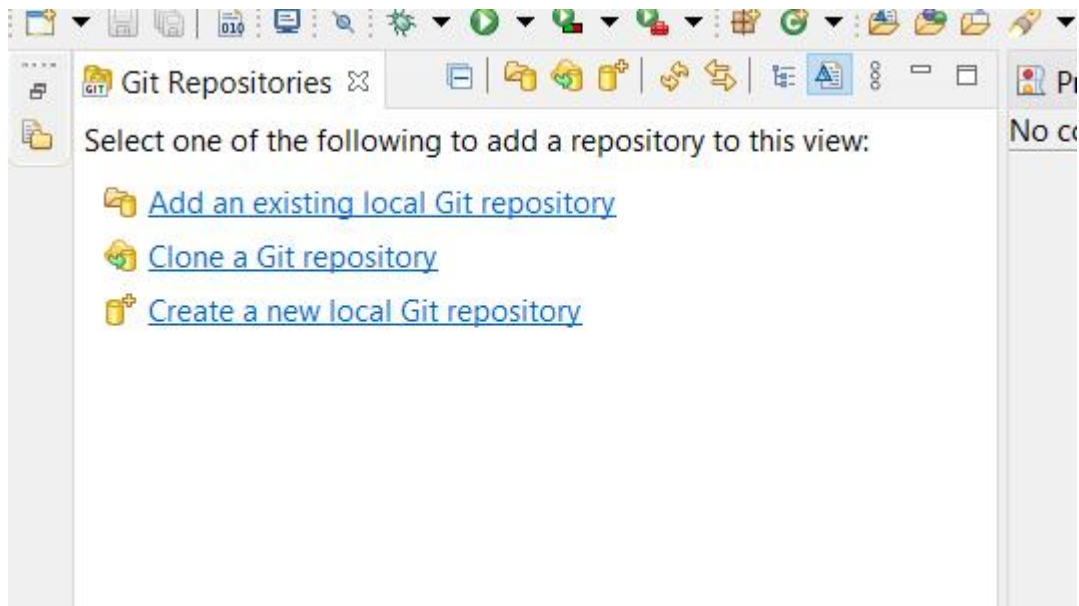
Step 1: After selecting the project, click on the Green-colored Code button then copy the hyperlink as shown in the image. You can copy the link manually or by just click on the **Copy** icon.



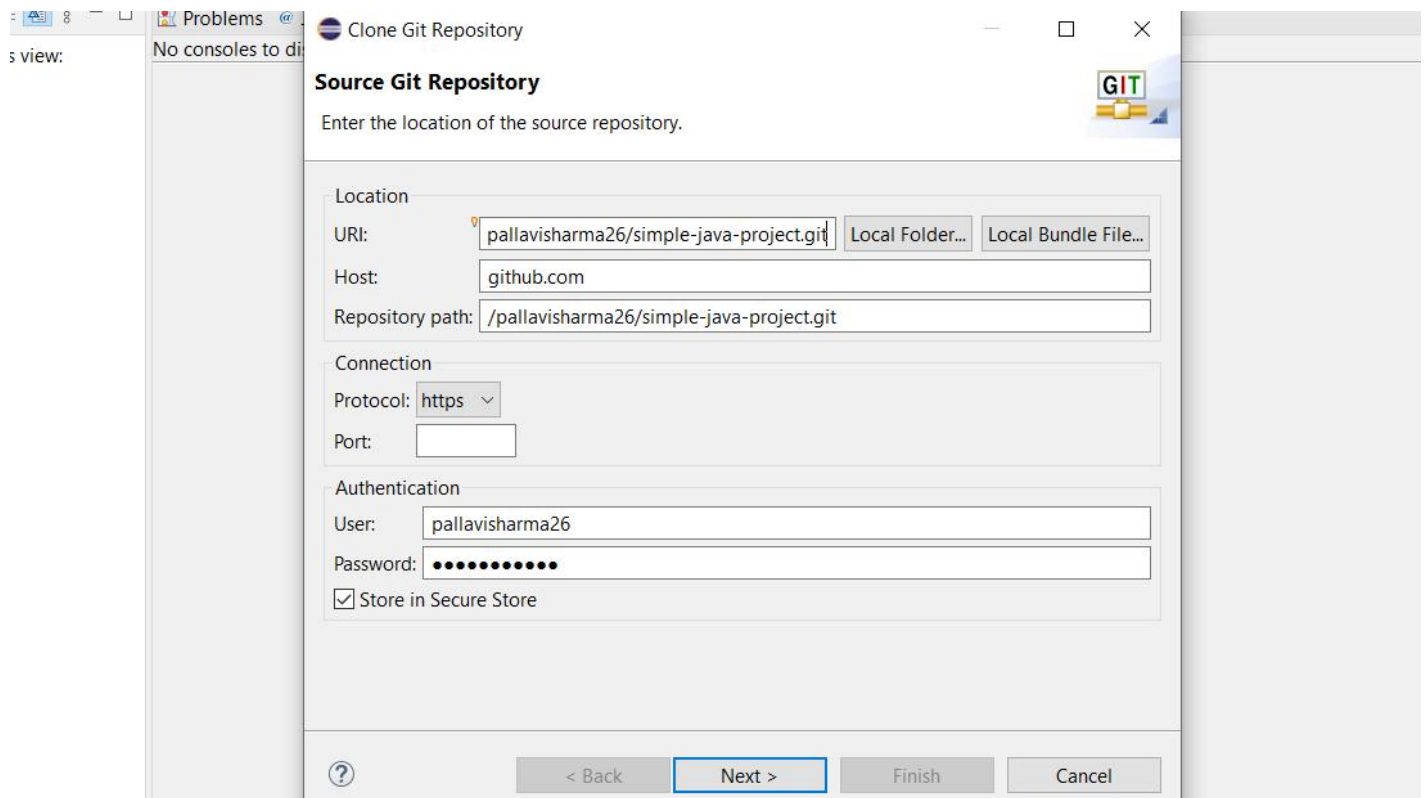
Step 2: Open Eclipse and go to **Window > Show views > Other > Git > Git Repositories** for making git repositories visible in eclipse as shown in the image.



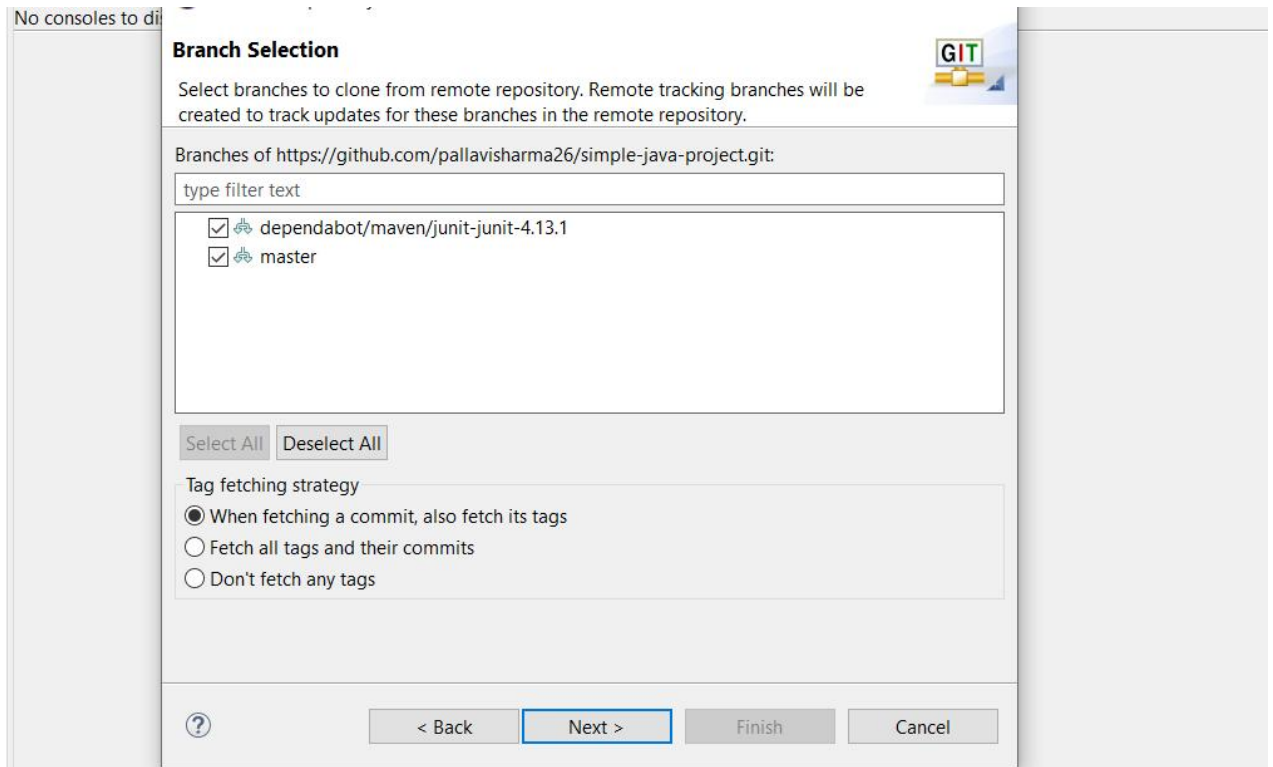
Step 3: We will after this as shown in the image now select “Clone a Git repository”



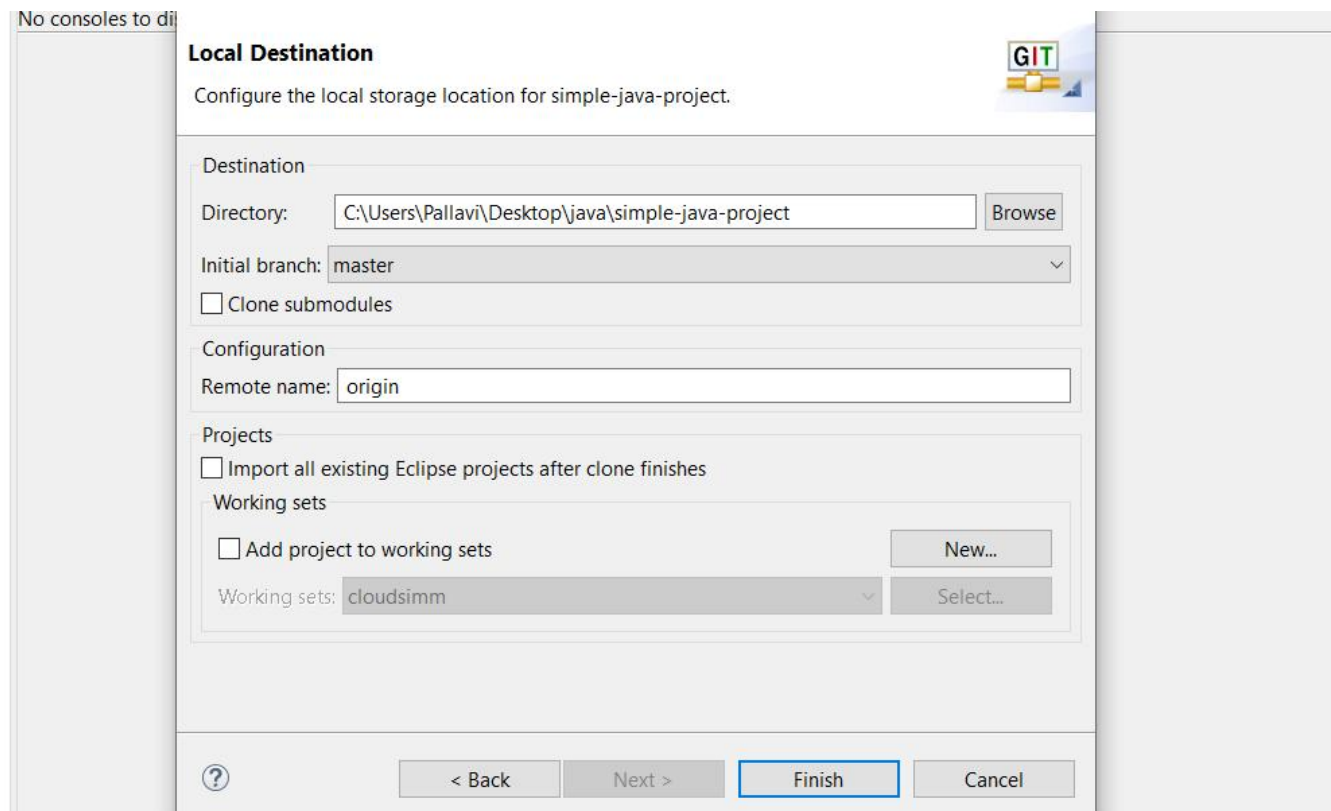
Step 4: A window will pop up in which you have to paste the GitHub Repository URL and also GitHub UserID and Password and click on the “Next” button.



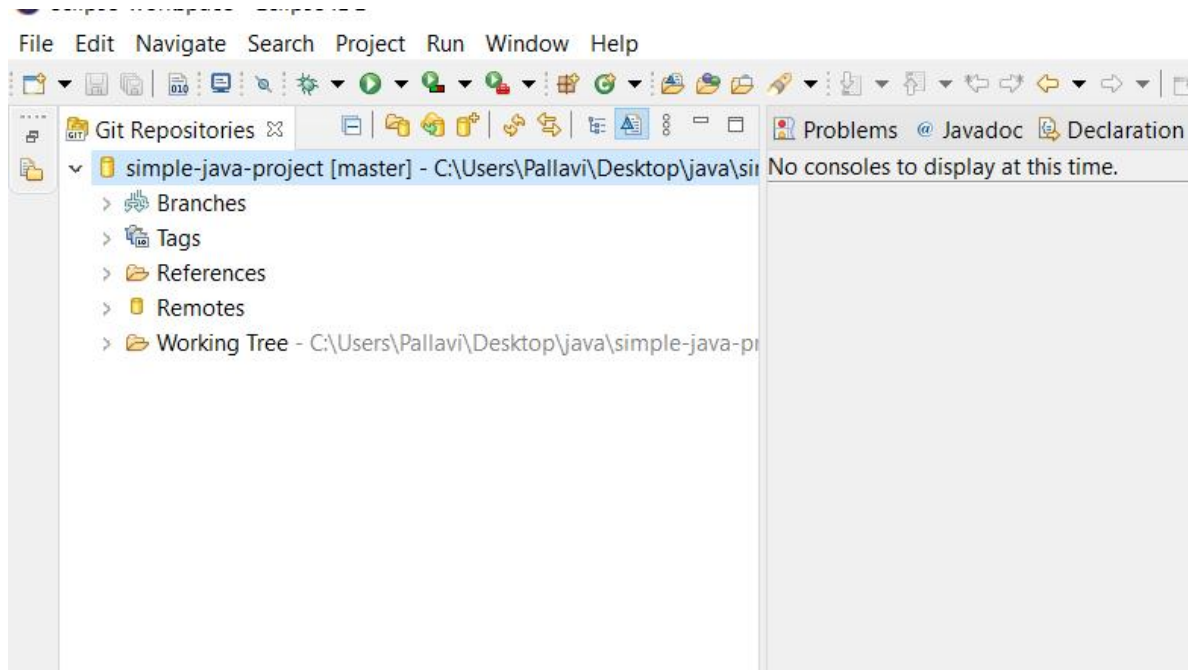
Step 5: Select “Branch” and click “Next”.



Step 6: Select the Folder directory in which you want to import the repository and click “Finish”.

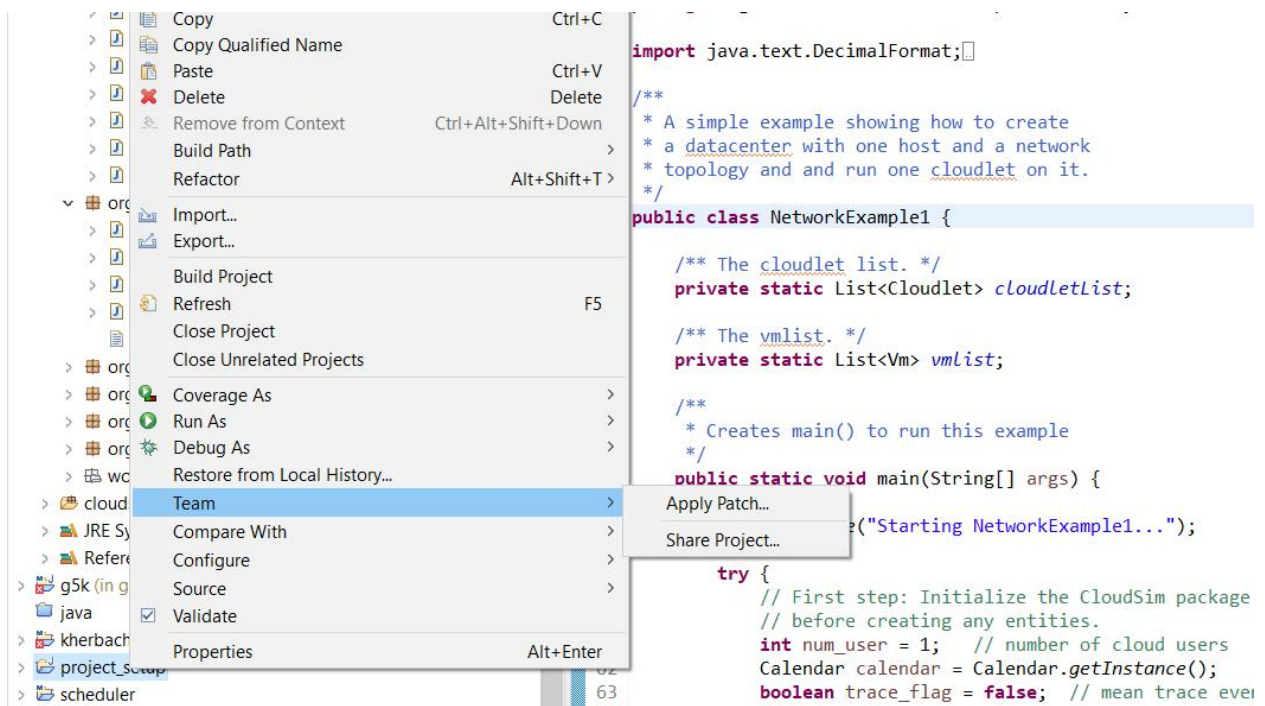


We have successfully imported the GitHub Repository and can make further changes to it.

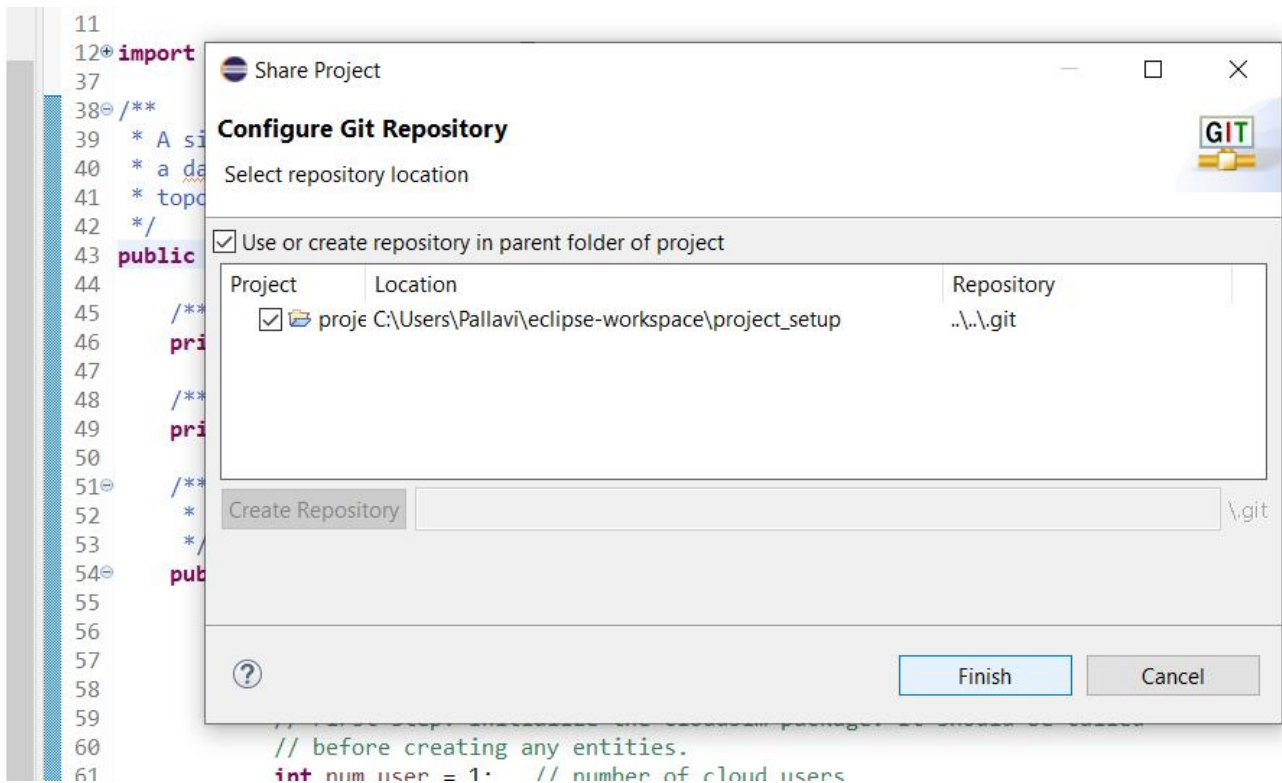


Syncing a project & Commit & Push

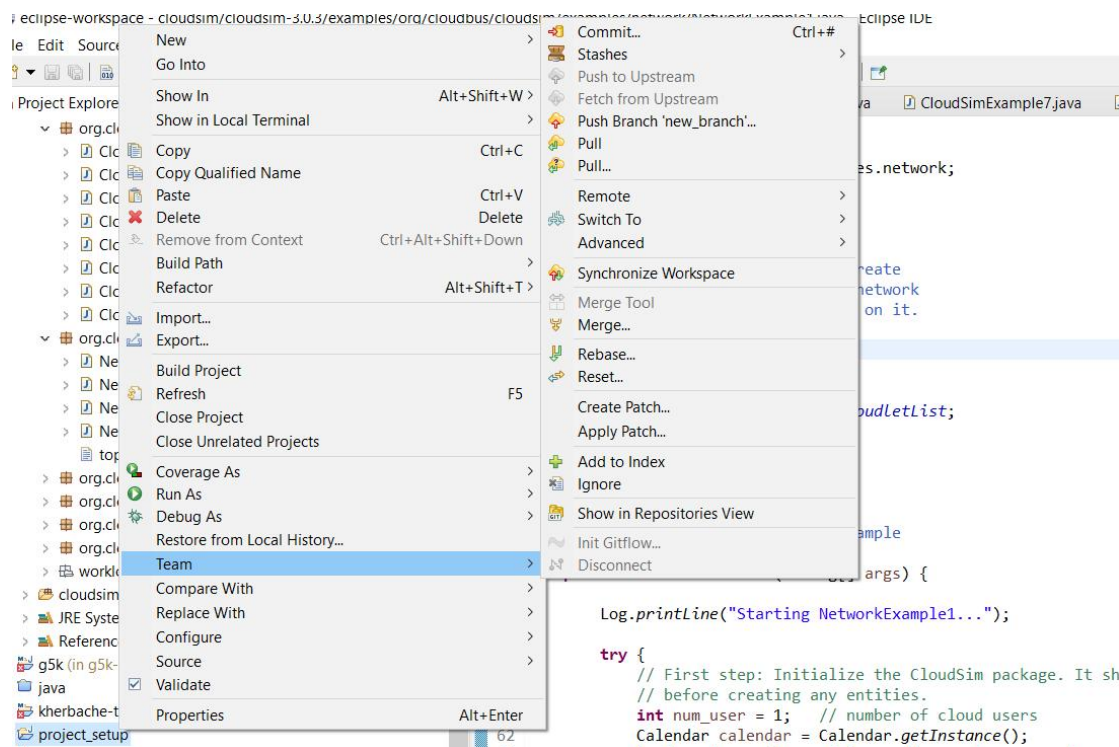
Step 1: Open Eclipse IDE and right-click on the project you want to push and go to Team->share project.



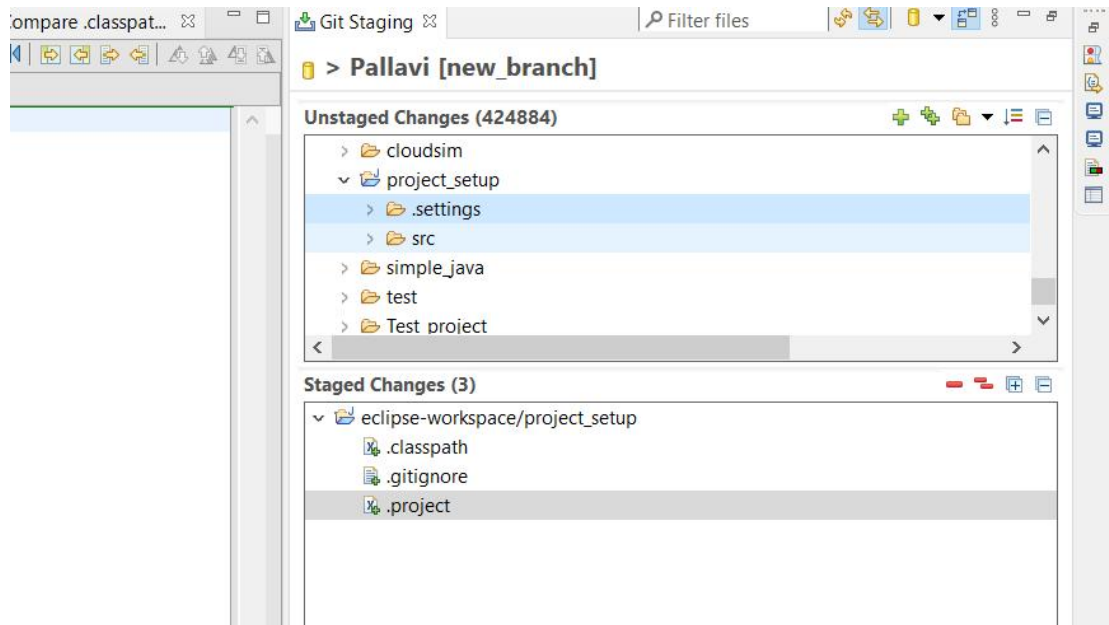
Step 2: It will add the project to the given repository as shown below:



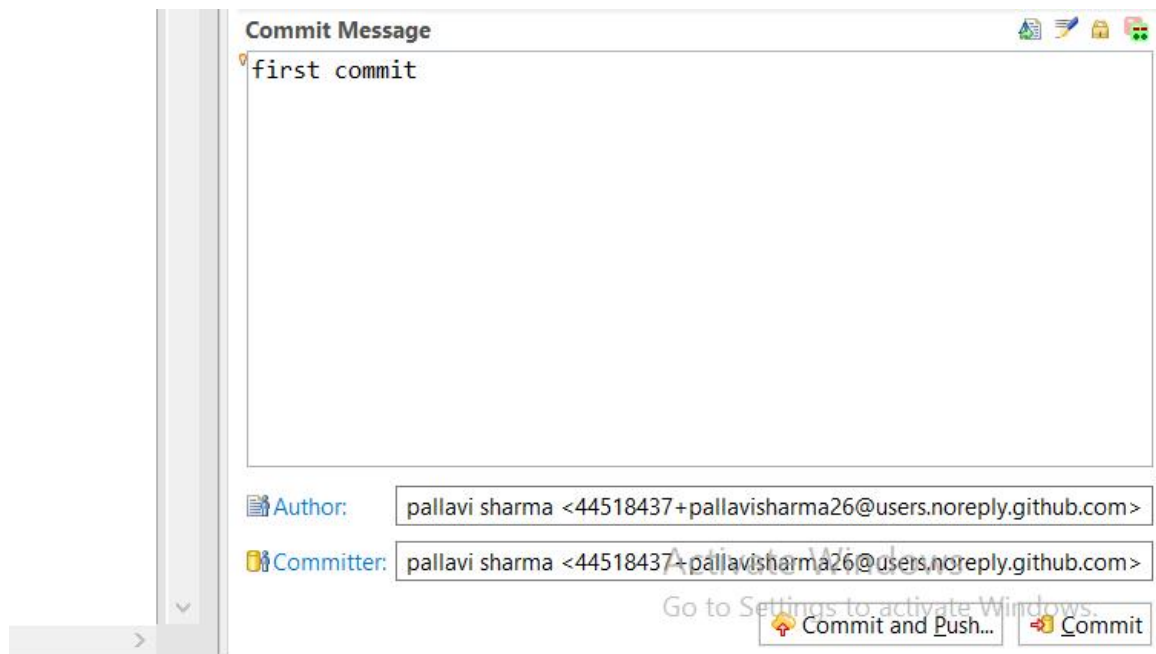
Step 3: Again right-click on the project and go to Team->commit.



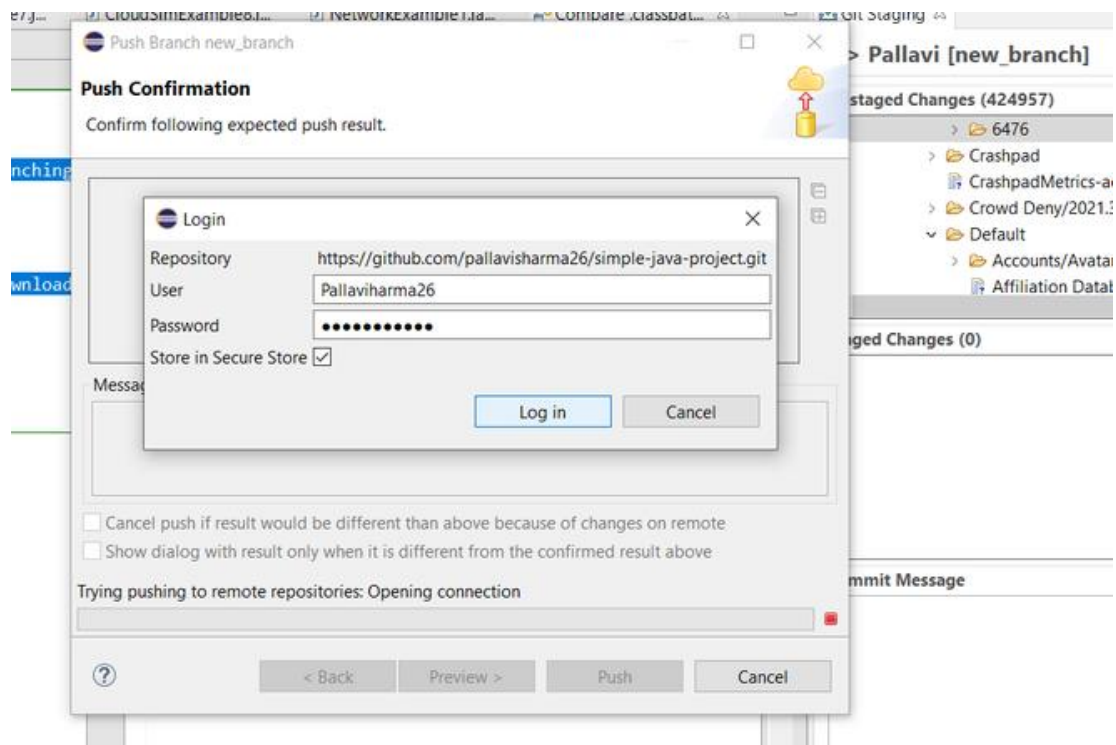
Step 4: Drag and Drop the files you want to commit from Unchanged Changes to Staged Changes.



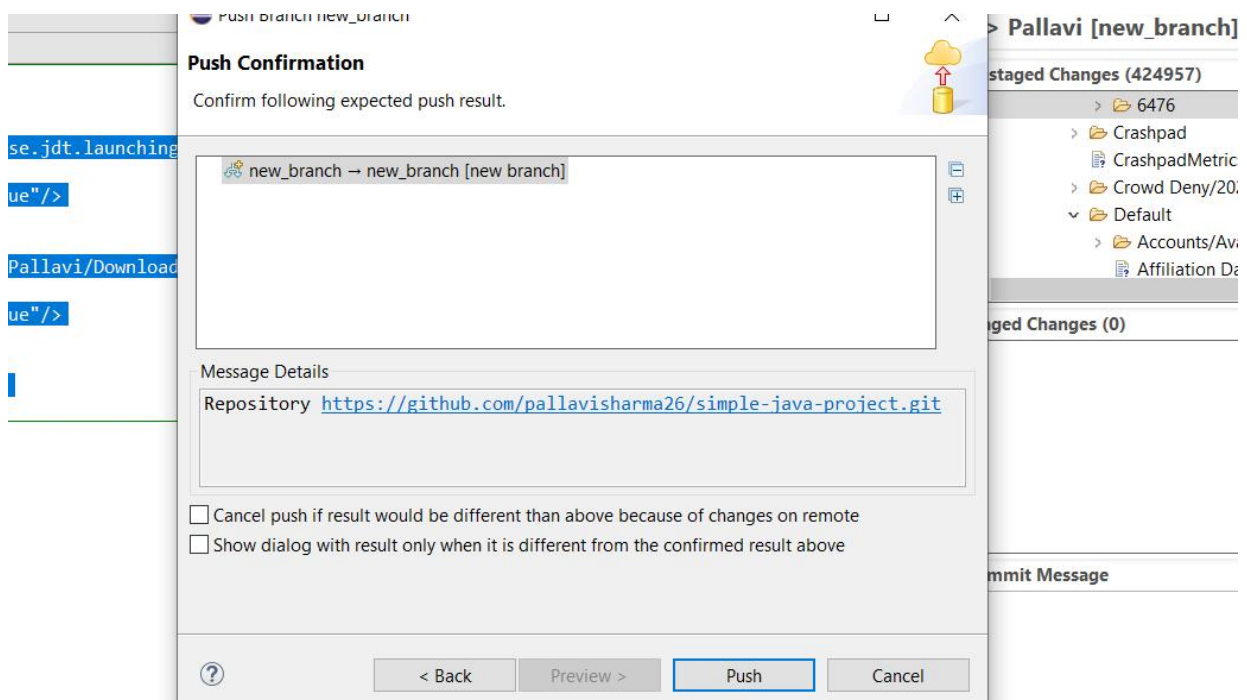
Step 5: Write the commit message in “Commit Message” and click “Commit and Push”.



Step 6: Fill in your UserID and password of GitHub and click “Log in”.



Step 7: Push to the new-branch of GitHub Repository and click “Push”.



The new branch is created on GitHub and is required to be merged by creating a pull request.

"1. gitignore file

gitignore file is a text file that tells Git which files or folders to ignore in a project. A local .gitignore file is usually placed in the root directory of a

project. You can also create a global `.gitignore` file and any entries in that file will be ignored in all of your Git repositories.

2. `readme.md` file"

3. `README.md` files are intended to provide orientation for engineers browsing your code, especially first-time users. The `README.md` is likely the first file a reader encounters when they browse a directory that contains your code. In this way, it acts as a landing page for the directory.