**Git (Hub)**

**What is GIT?**

Git is a version control system.

**What is version control system?u are having multiple version of document or software code, Git will help us to manage those versioning.**

During college final year project, we created a document with different name during changes.

In simple college project itself, its very different to handle those changes..when we look the document after 1 year..we don’t know which one is latest..think about big software project

Initial name: final year project

Next name: final year project\_modified

Next name: final year project\_reviewed

Next name: final year project\_Final

Nest name: final year project\_Final\_Modified

Net name: final year project\_Completed

**Why it was created?**

Created by **Linus Torvalds**

<https://www.geeksforgeeks.org/history-of-git/>

The story of Git’s creation is closely tied to the development of the Linux kernel, an open-source project led by Linus Torvalds. In the early 2000s, the Linux kernel project relied on a proprietary VCS called BitKeeper. BitKeeper was a distributed version control system that suited the needs of the kernel’s distributed development model.

However, in 2005, a licensing dispute between the Linux kernel community and BitKeeper’s owner led to the withdrawal of the tool’s free version. This left the Linux kernel project in urgent need of a new version control system. Linus Torvalds decided to create a new system from scratch, prioritizing the following goals:

With these objectives in mind, Linus Torvalds began developing Git in April 2005. Within weeks, Git was functional and began to be used for the Linux kernel’s development.

**Key Milestones in Git’s Development**

* **Initial Release (2005):** The first version of Git was released on April 7, 2005. It quickly gained traction within the Linux kernel community.
* **Git 1.0 (2005):** By December 2005, Git 1.0 was released, marking its readiness for broader use beyond the Linux kernel project.
* **GitHub Launch (2008):** The launch of GitHub, a web-based platform for hosting Git repositories, revolutionized how developers collaborated on open-source projects. GitHub’s user-friendly interface and social features accelerated Git’s adoption.
* **Wide Adoption (2010s):** Throughout the 2010s, Git became the standard VCS in the software development industry. Major companies and open-source projects transitioned to Git, recognizing its advantages over other systems

**Git vs Github**

Based on Git technology, Github product developed & enhanced with user friendly webbased UI

**Basic GITHUB Terminologies**

1.Repositories [ a folder contains all the files related to project]

2.Readme [basic instruction about the project]

3.Commit (saving the state)

**Advanced GITHUB Terminologies**

4. Branch

5.Merge

6.Fork

7.Pull Request

**What is GITHUB?**

1.Github is a web based hosting services for software code([www.github.com](http://www.github.com))

2.Its provide version control using GIT Technology

3.Its is a collaborative tool which helps multiple user to work on a single project without much difficulties

Additional Note..we can store the project code/document in another devices like

**GoogleDrive, Onedrive, Dropbox**

**Create a GITHUB account online?**

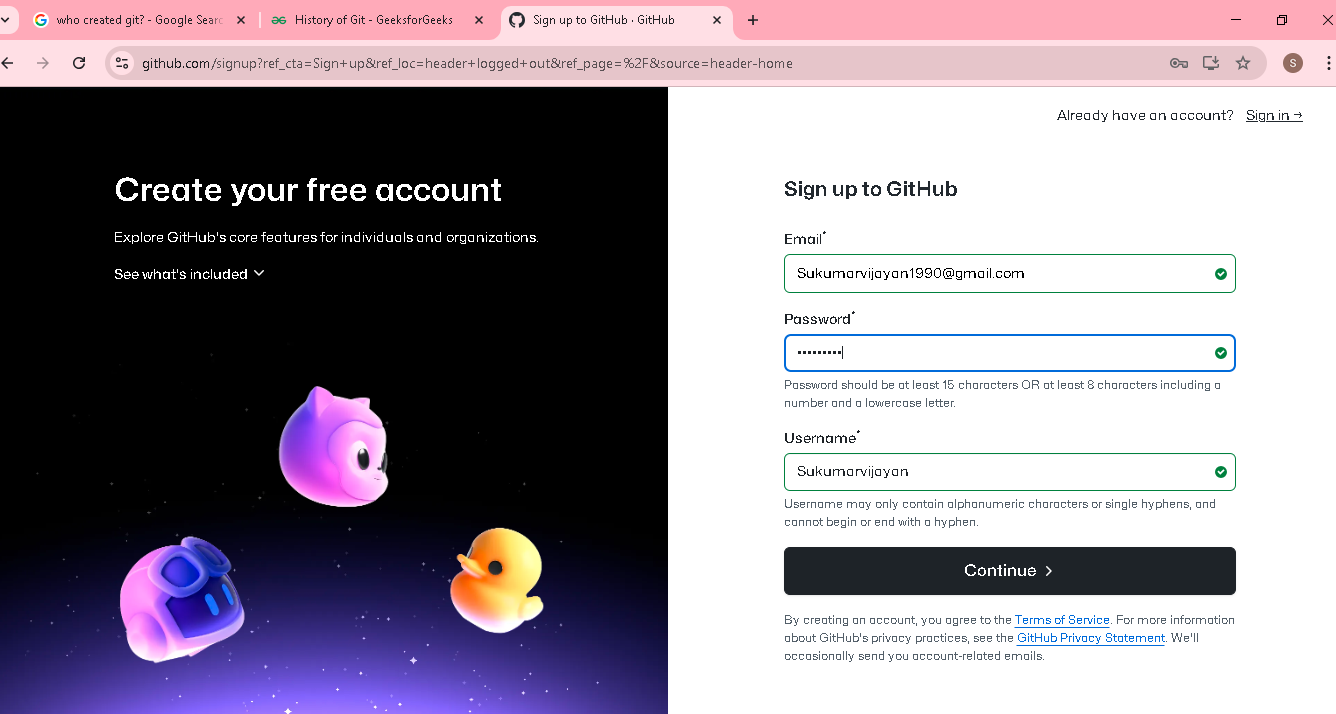
[**https://github.com/**](https://github.com/)

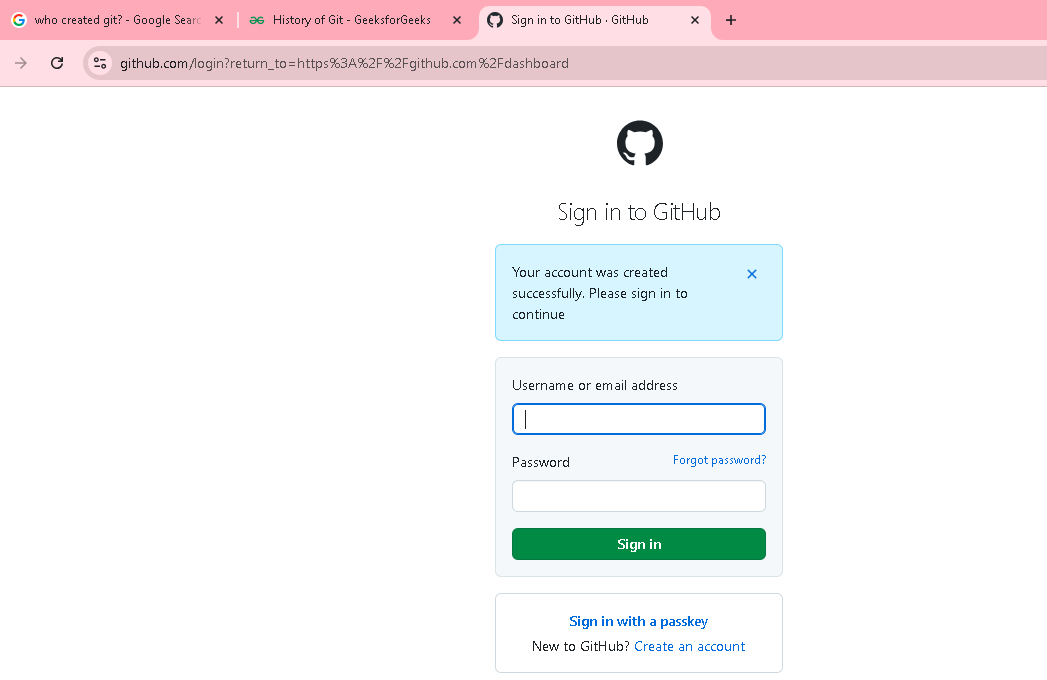
**Email:** [**Sukumarvijayan1990@gmail.com**](mailto:Sukumarvijayan1990@gmail.com)

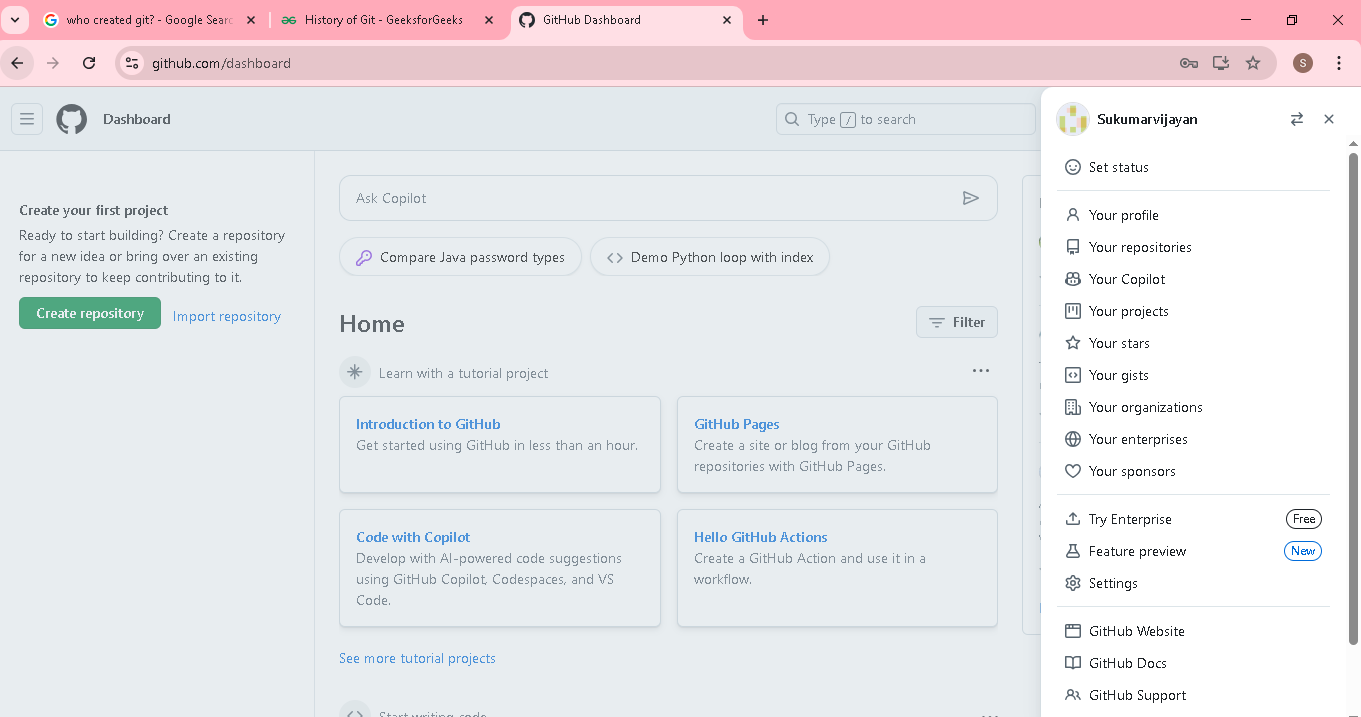
**PW: Aarinew1$**

**Username: Sukumarvijayan**

**Access token->** ghp\_n6donjMtueS3phgYPPMdzJHzrRU4kl4UFify

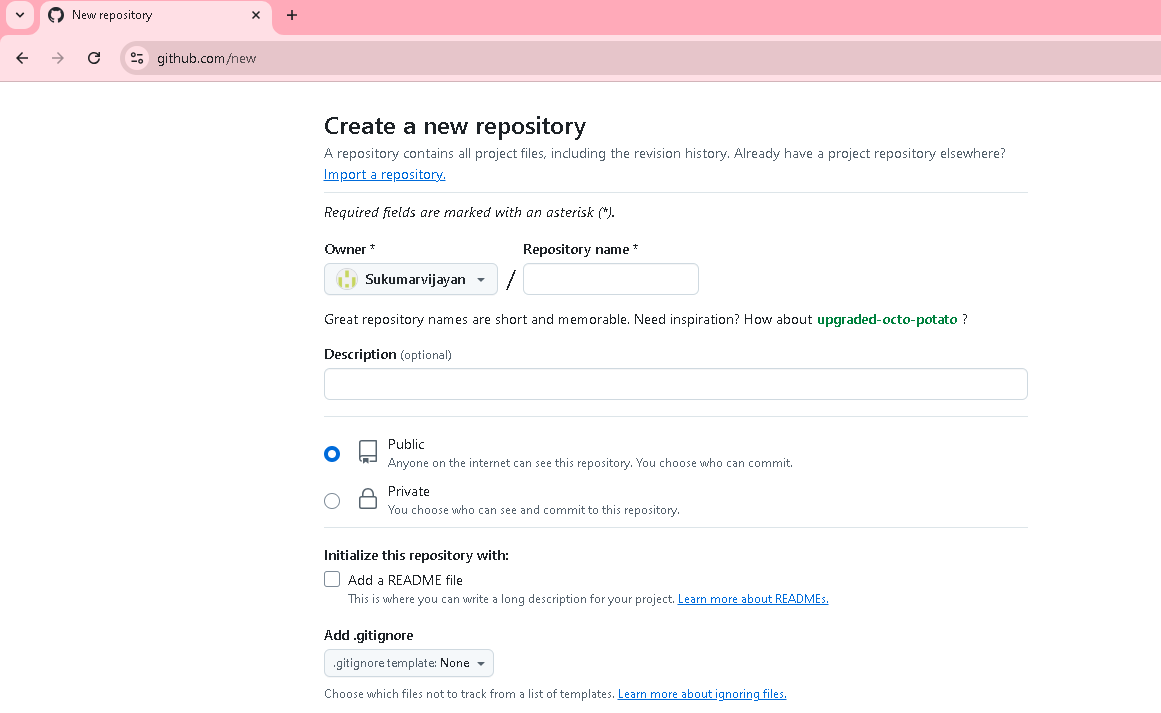
****

****

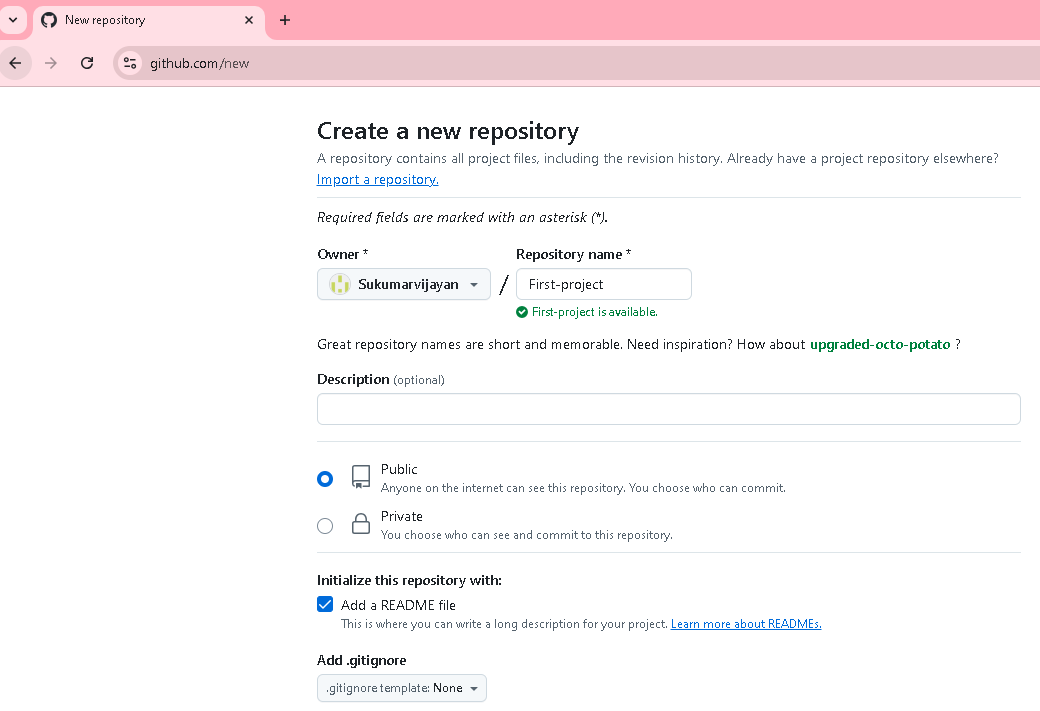
****

**Create Repository**

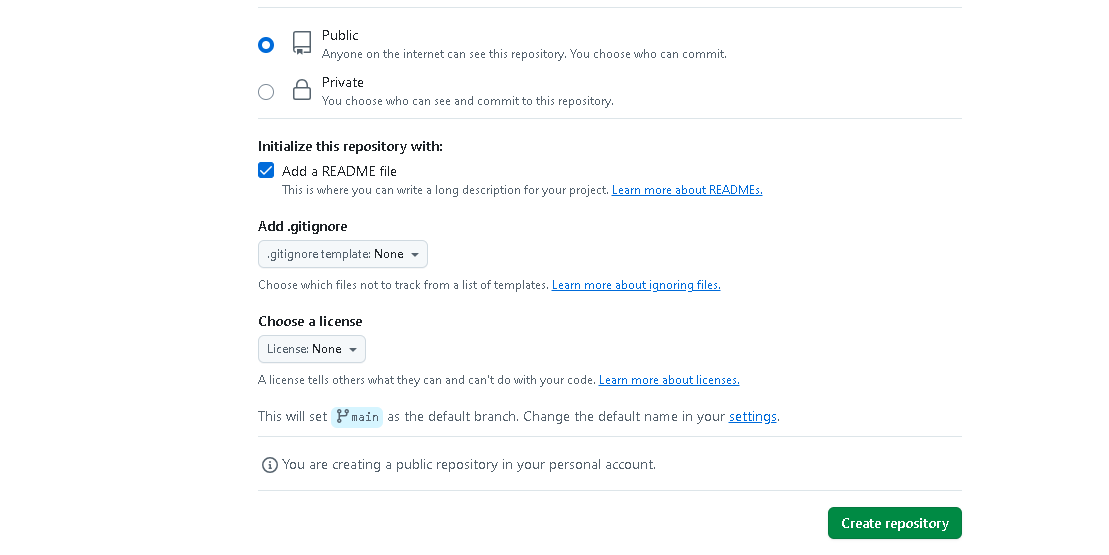
Click Create Repository button

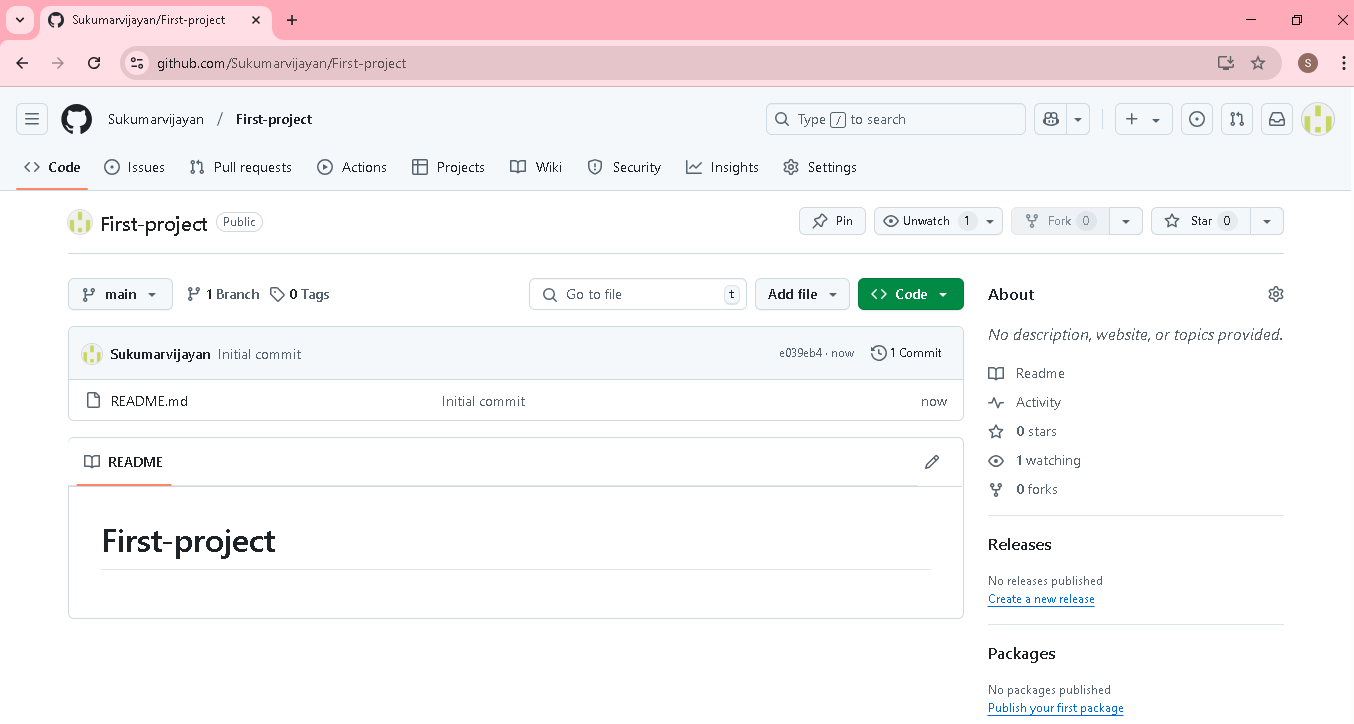
****

Fill the Repository name & check the “Add a README file”

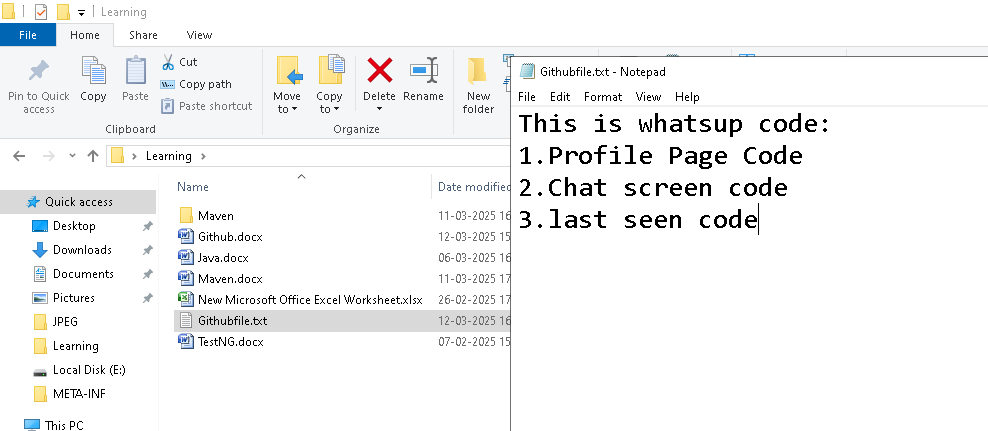


If private is selected, then also if its paid one & very very safe..no one will access to those account..in software industry uses private only with paid service for their code



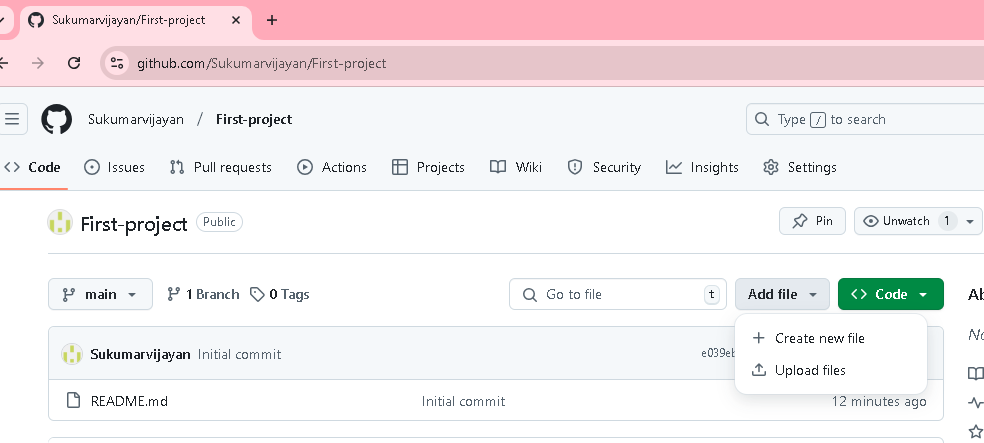


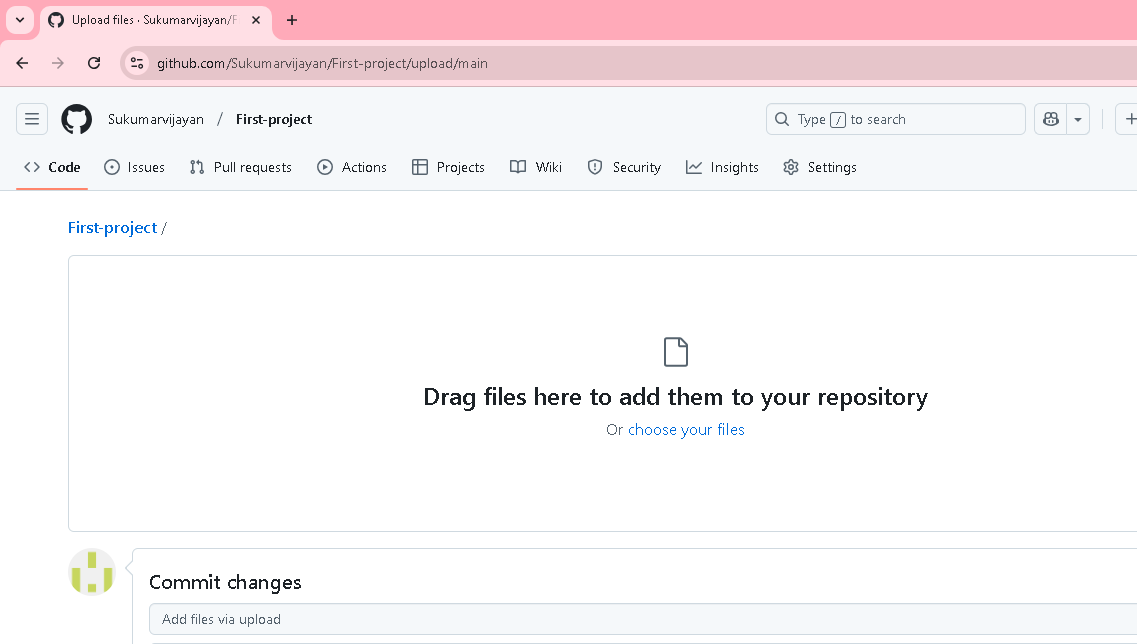
Upload simple txt file & do some changes & validate how its reflected in GITHUB



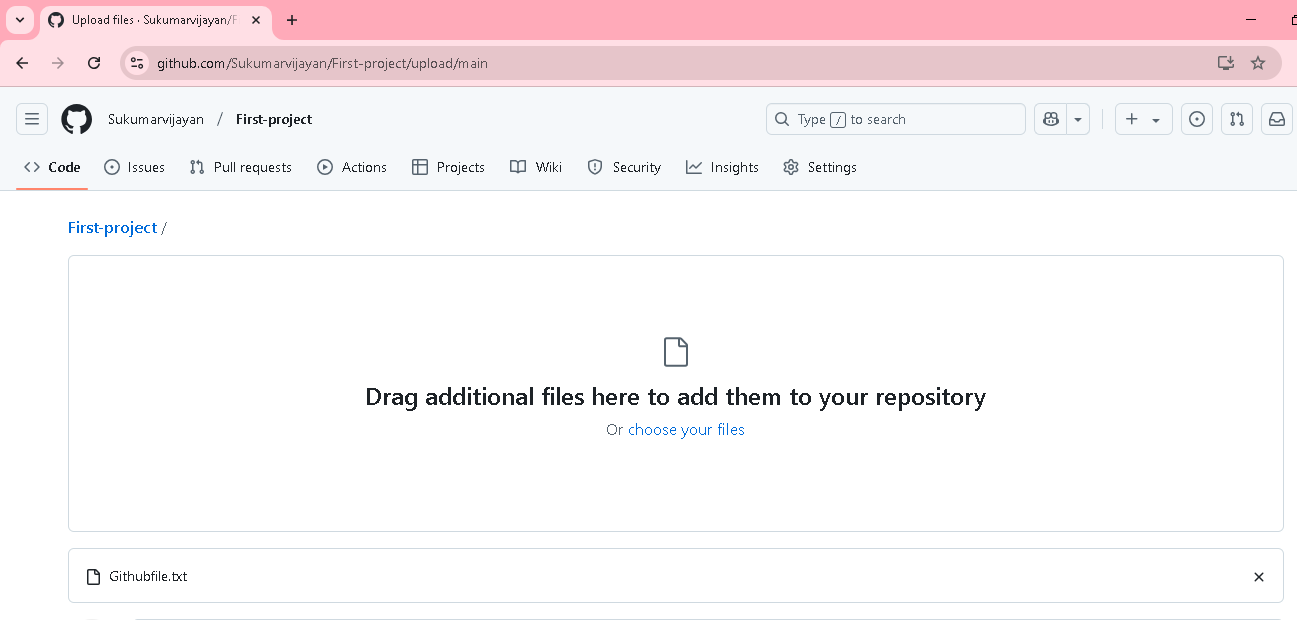
Click Add file button, in that select Upload files

Note: we can click create new file also & just copy paste the text



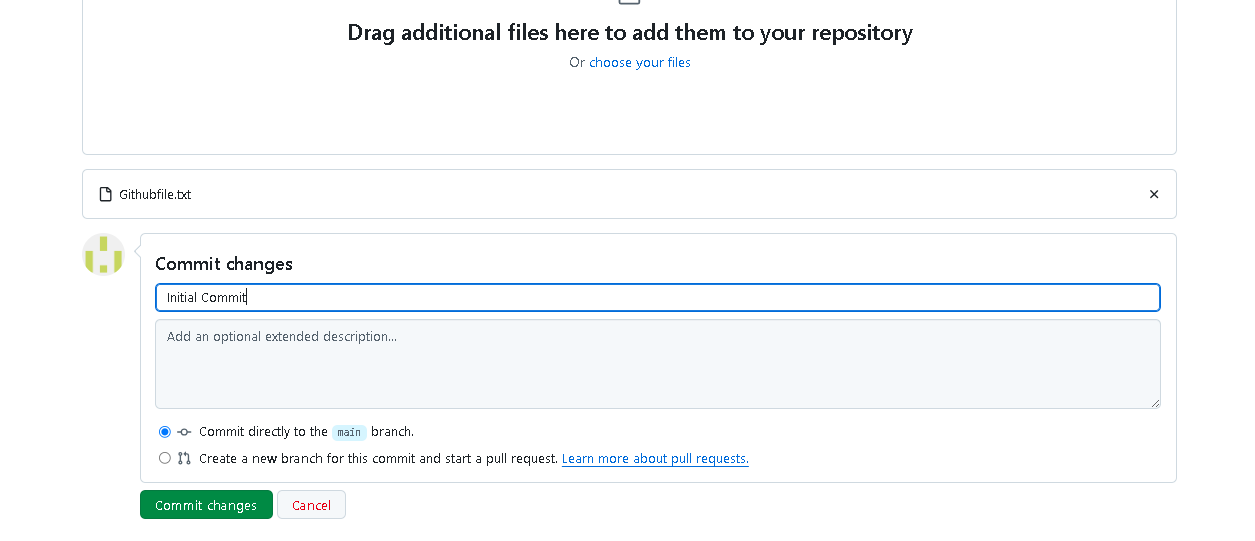


Browse & upload the files

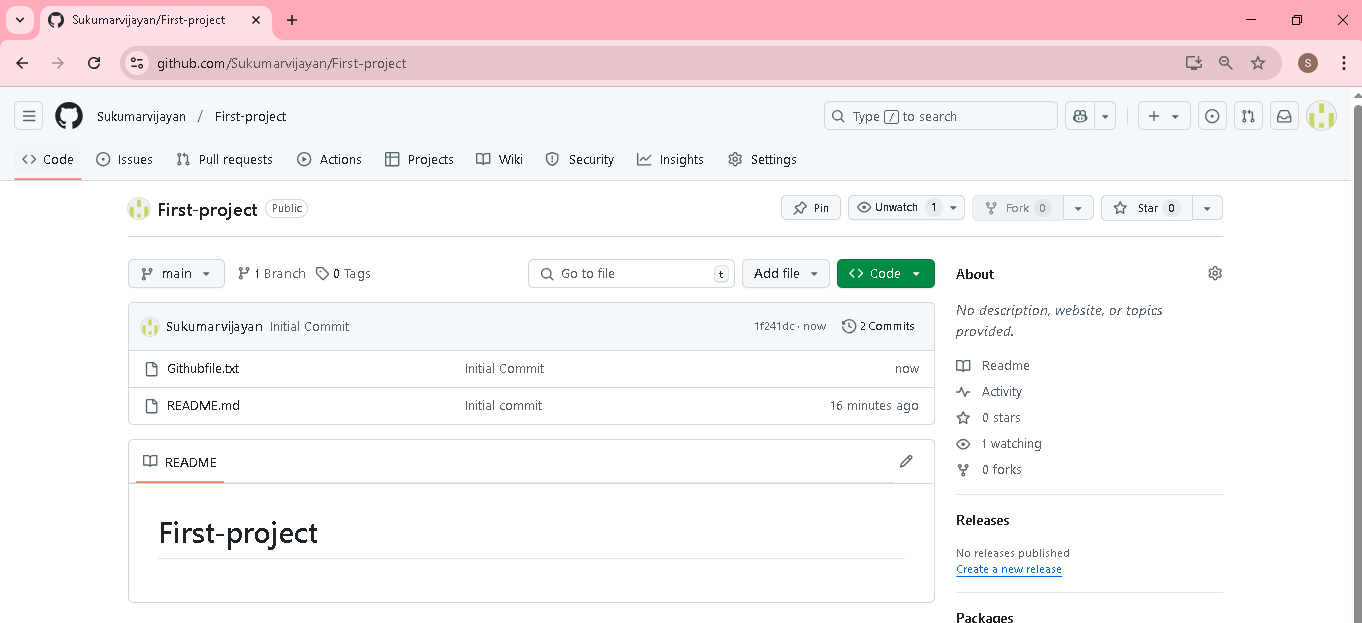


Provide the reason for commit changes

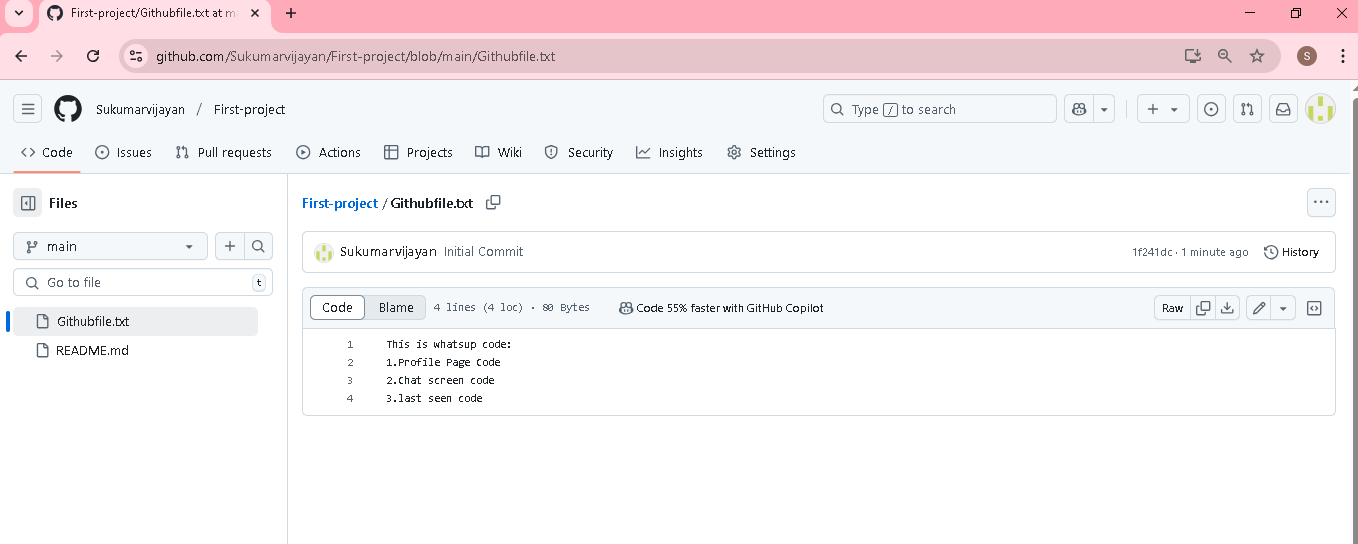
Based on changes, each time provide the reason accordingly



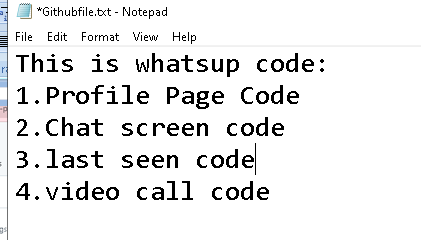
Click commit changes

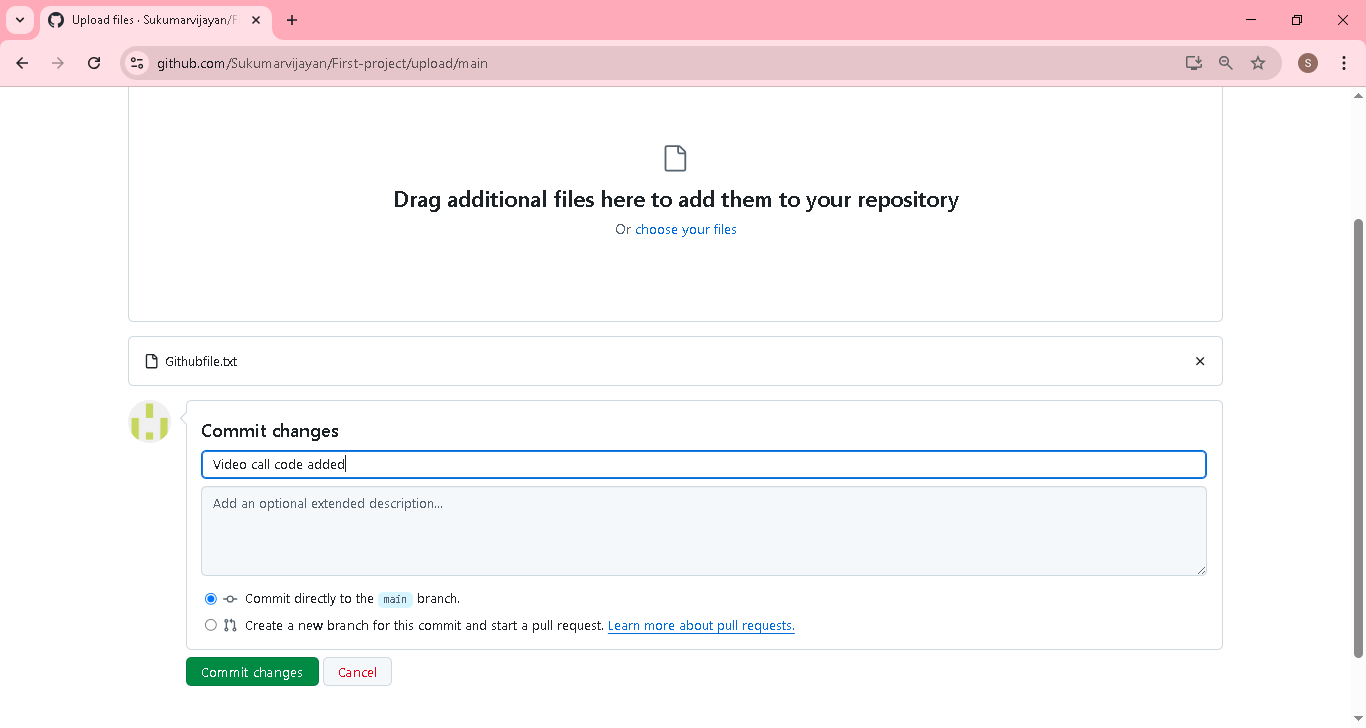


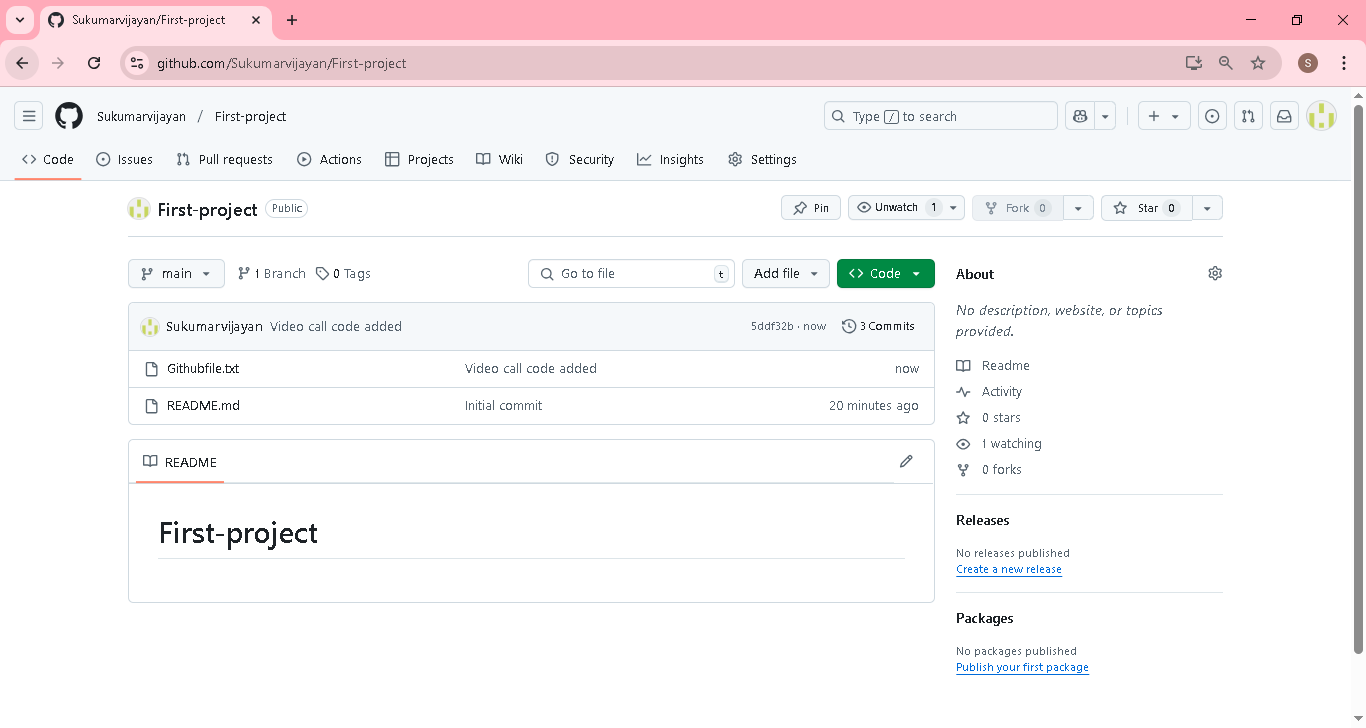
Click the file, we can see the code

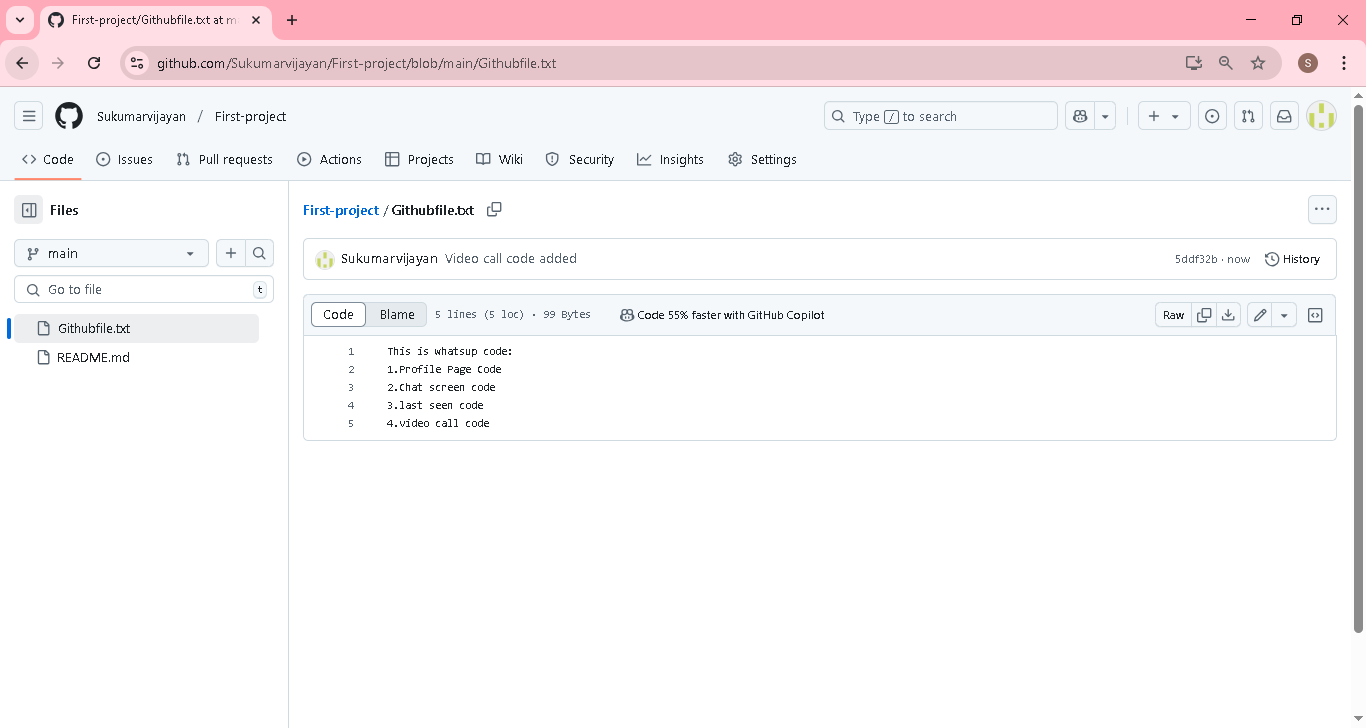


Then do the changes in text file & upload again

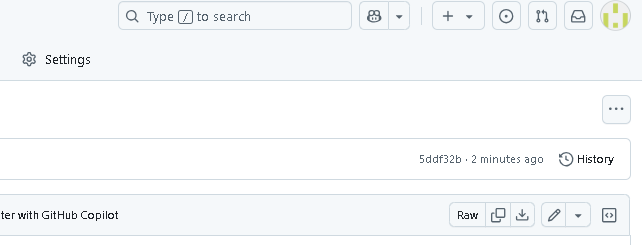




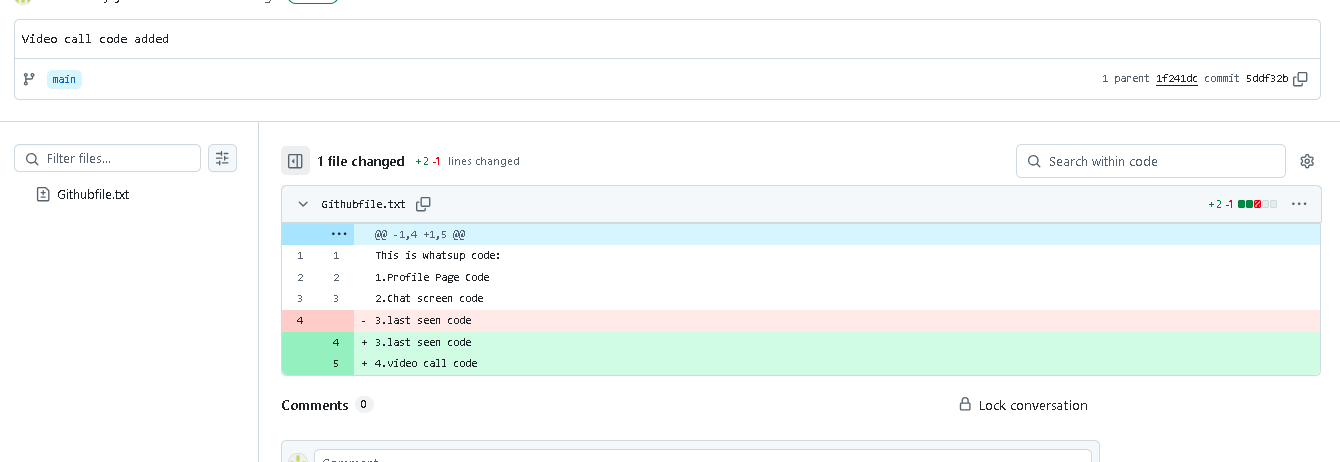




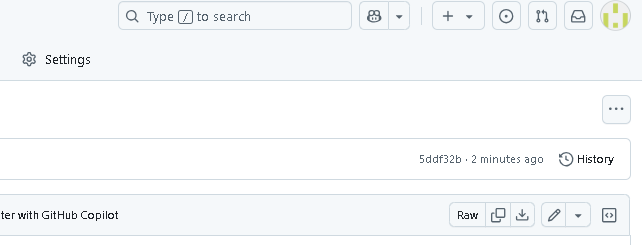
Click the version 5ddf32b[ code create automatically by guthub, its a version code]

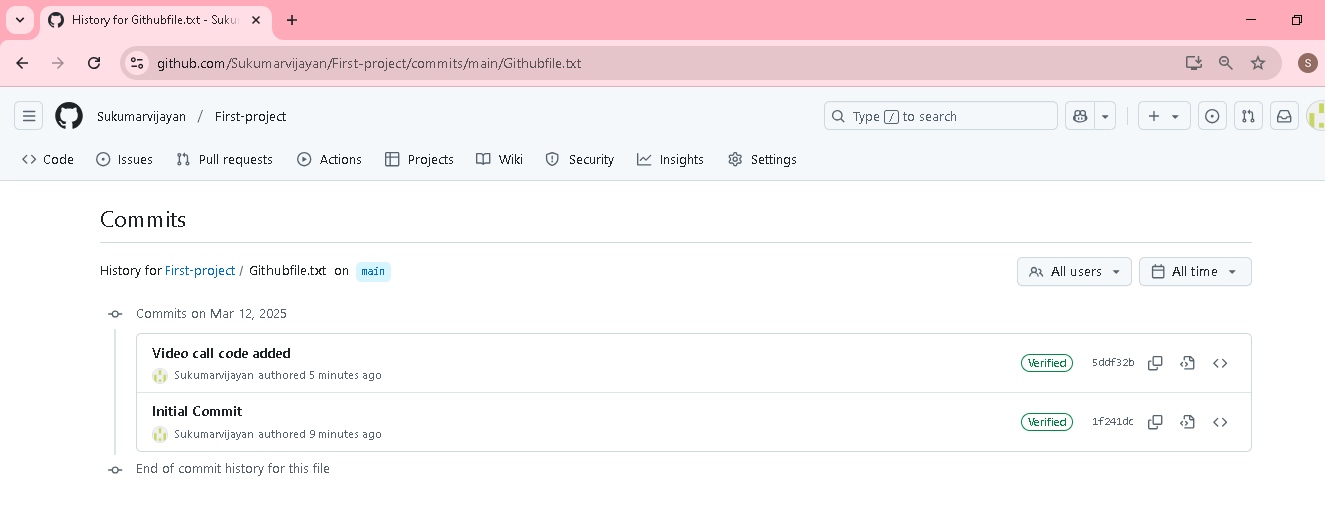


Click the version number, u can see the changes



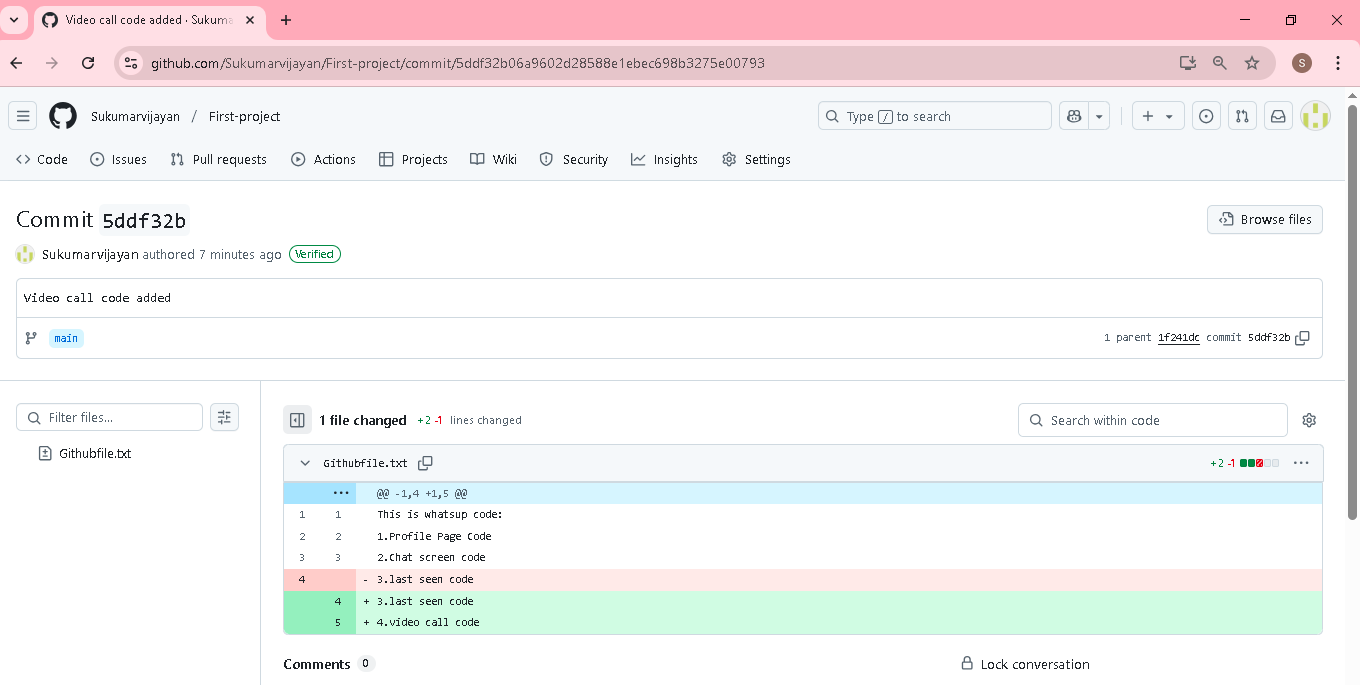
Click the history also, we can c entire version changes



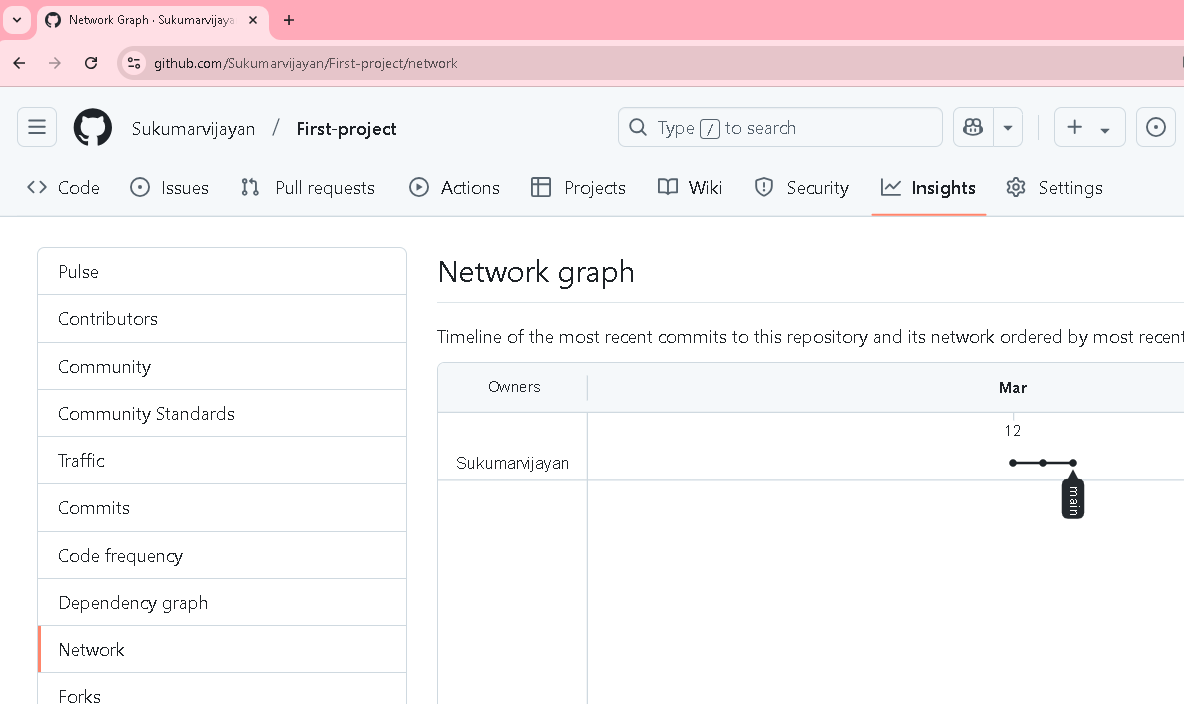


Click version number

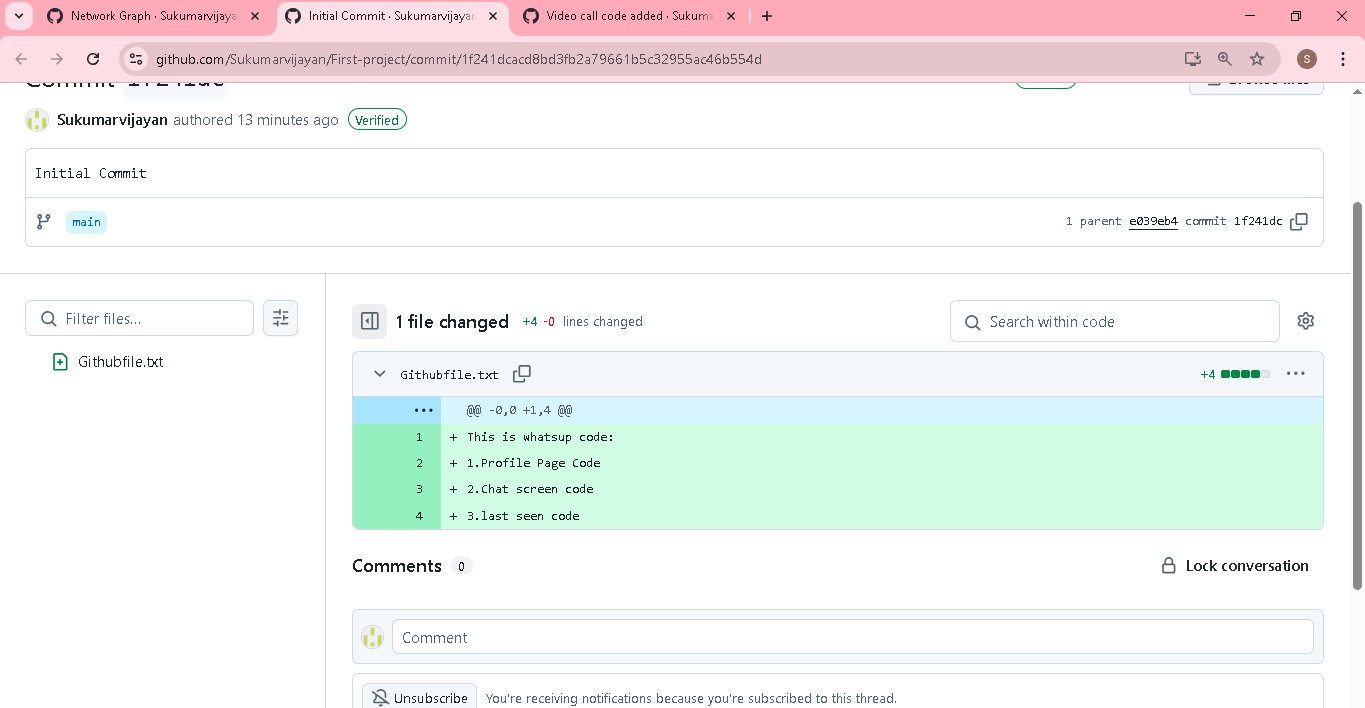
If any issue in the new version, then we can easly compare with old version..that why it a version control system

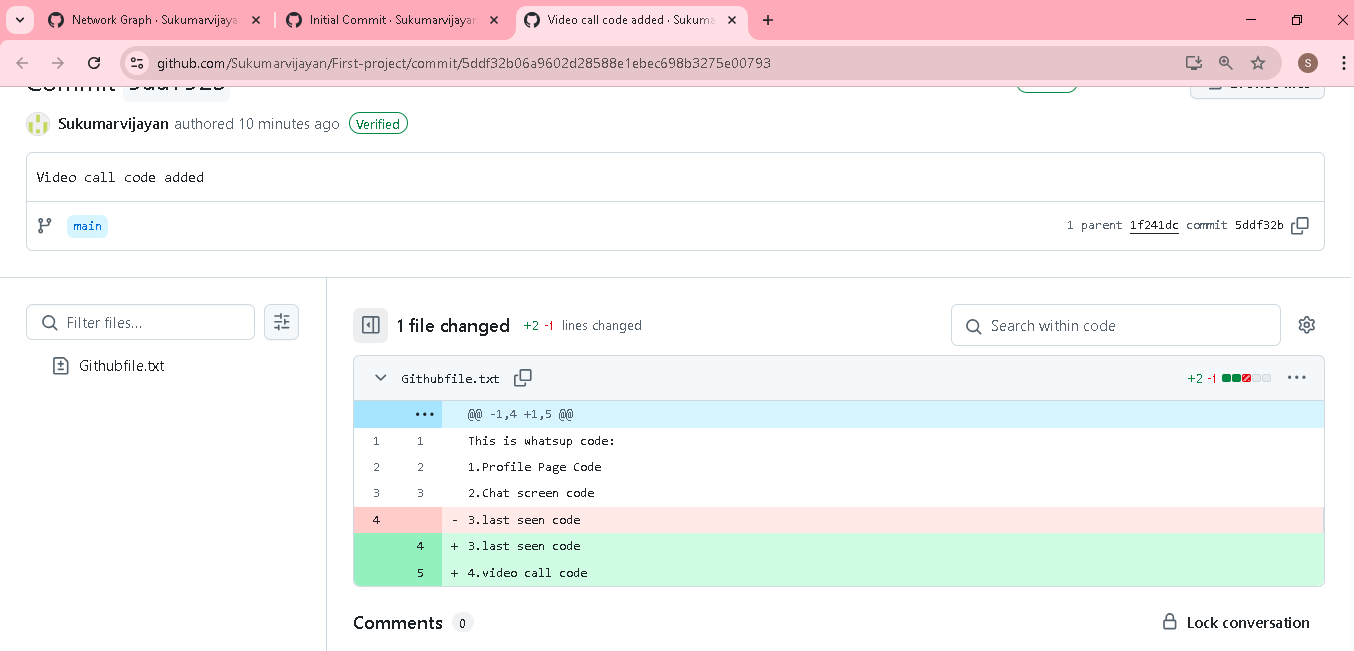


Click insight->network



By clicking the dot, it will open the code in new browser





**GitHUB Branches**

Branches is to isolate the development work without affecting other branches in the repository

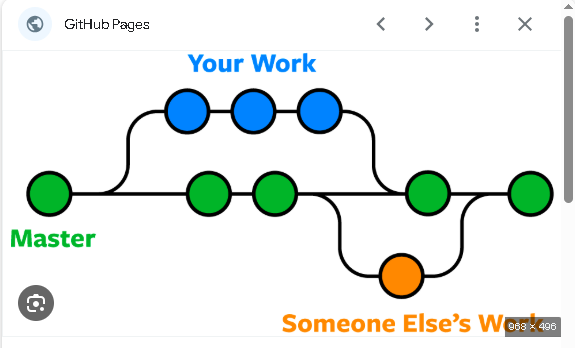
Each repository has one default branch (Master or Main) & can have multiple other branches.You can merge a branch into another branch using a pull request.

**You can use branches to:**

Develop features

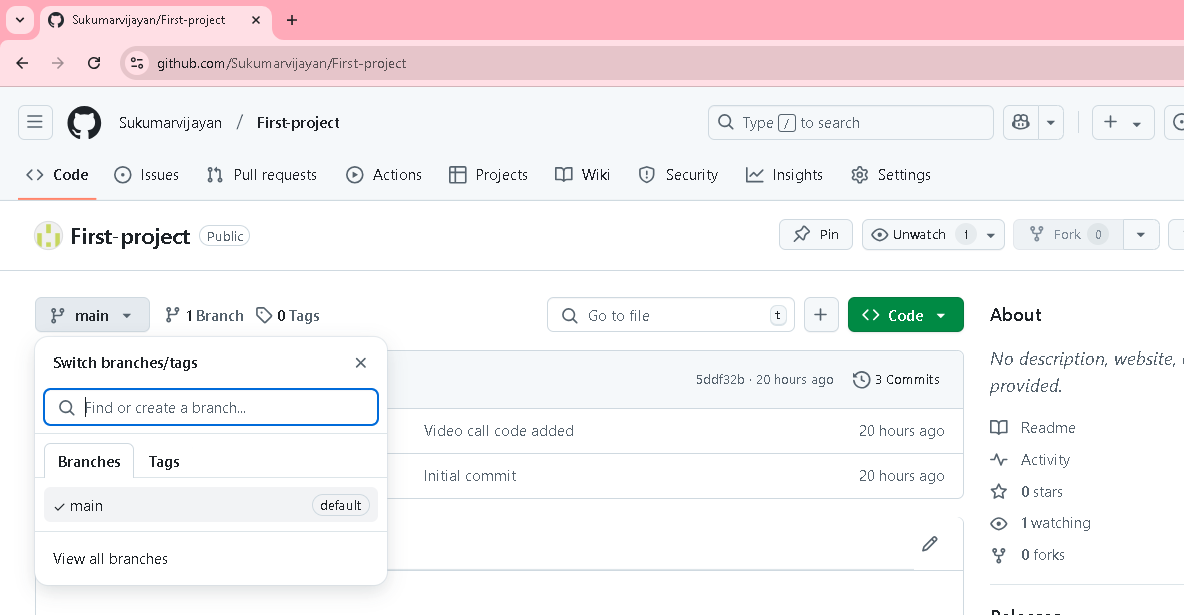
Fix bugs

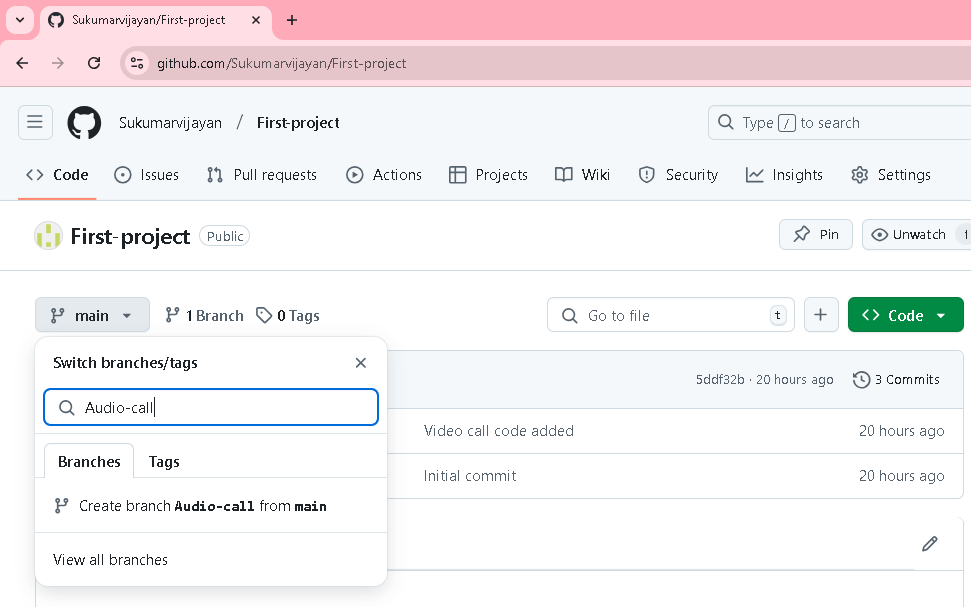
Safely experiment with new idea



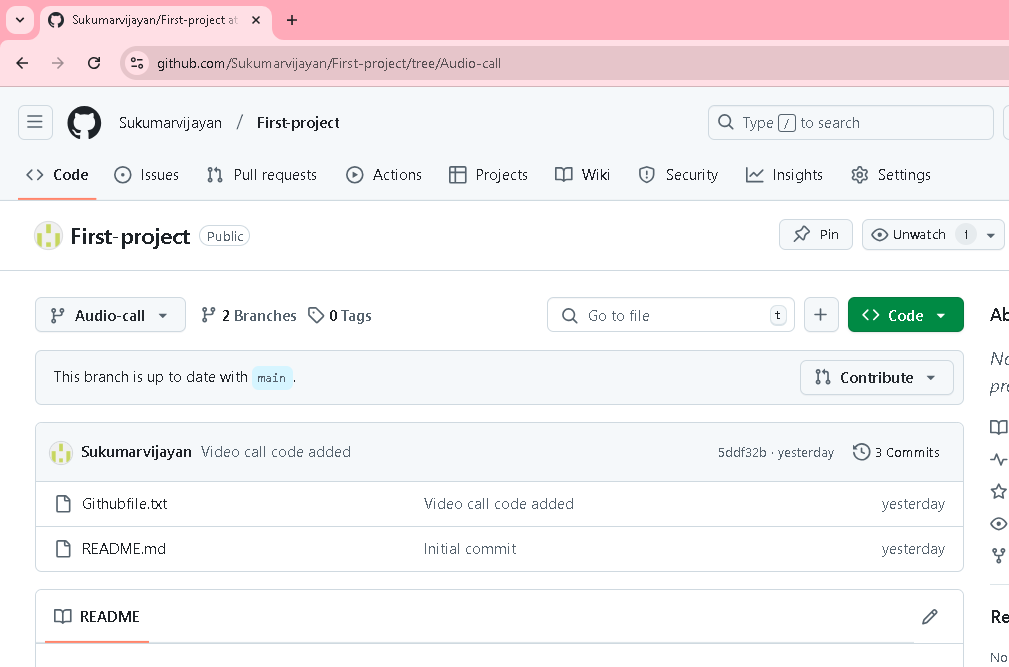
Create a branch

Click the main button & then create a branch by enter the any branch name

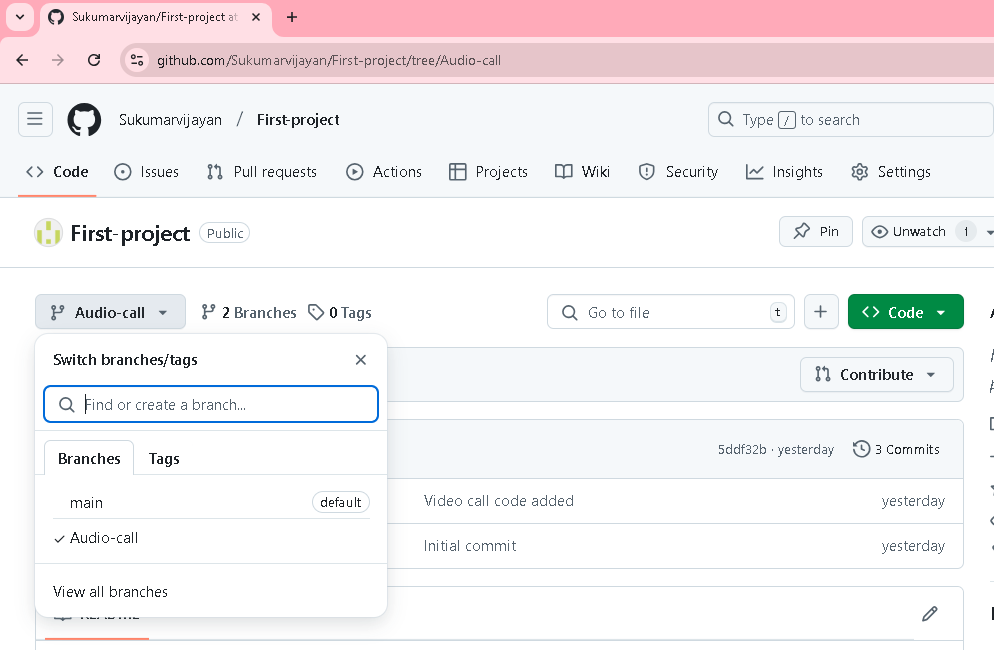




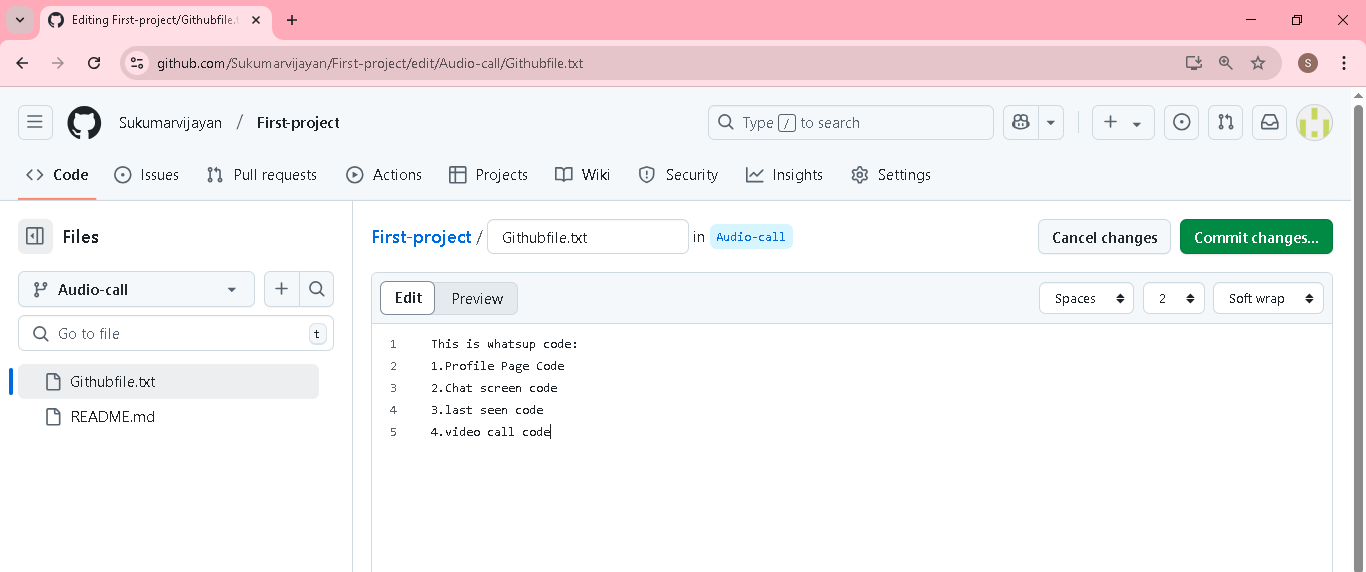
Click the create branch Audio-call from main

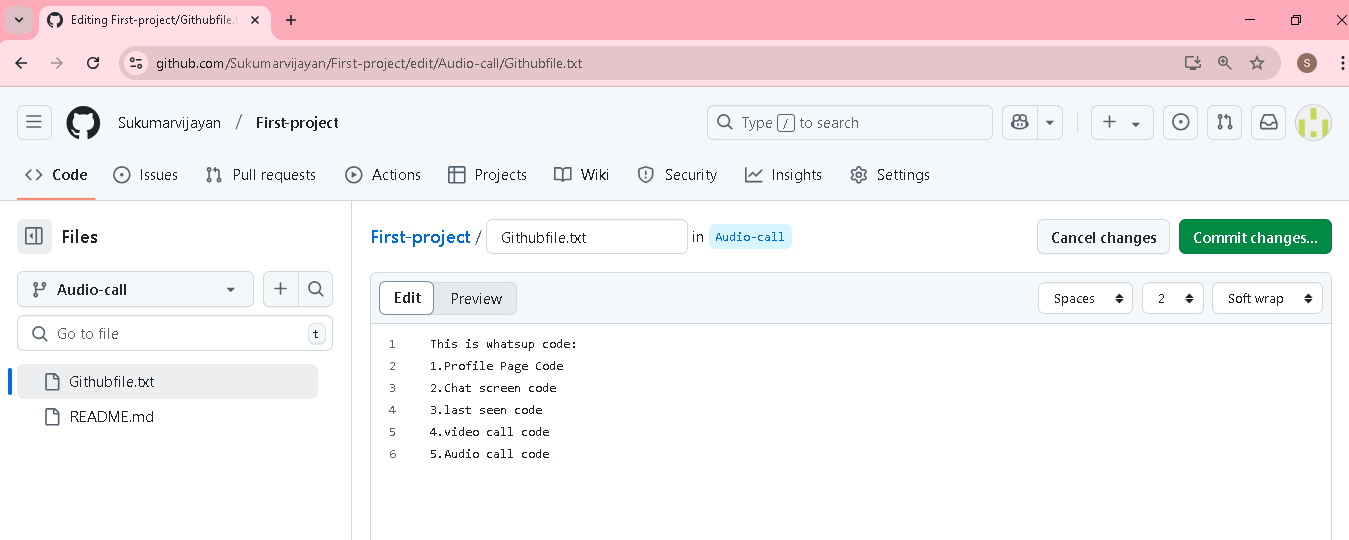


Now 2 branch available, Main & audio-call branch

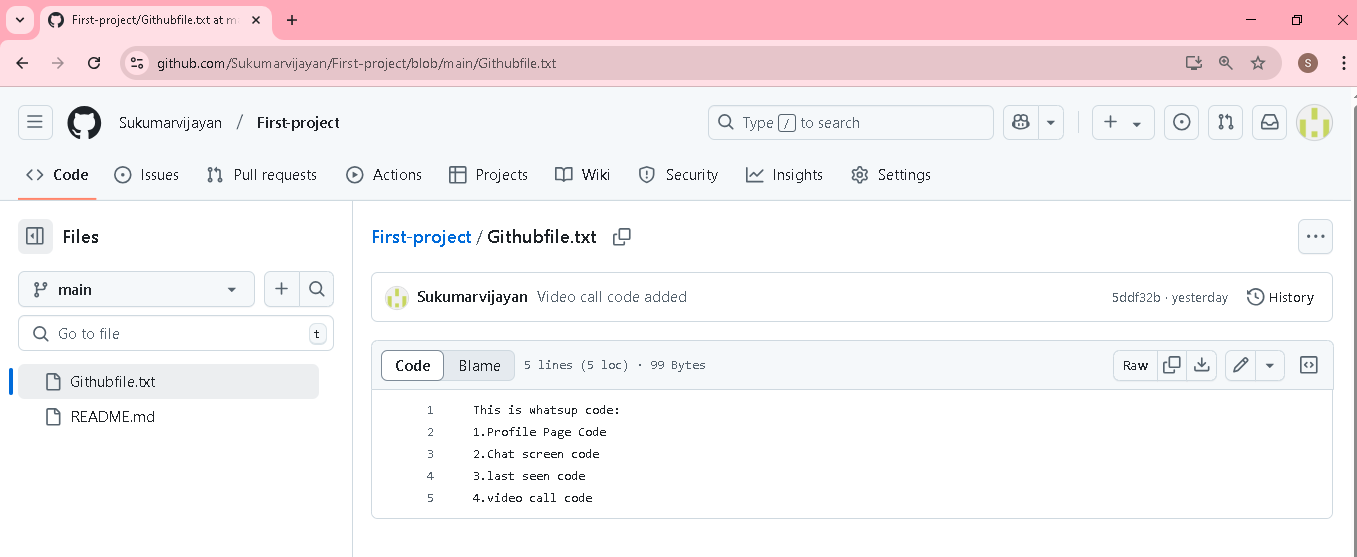


Edit the code in Audio-call branches & Commit the changes



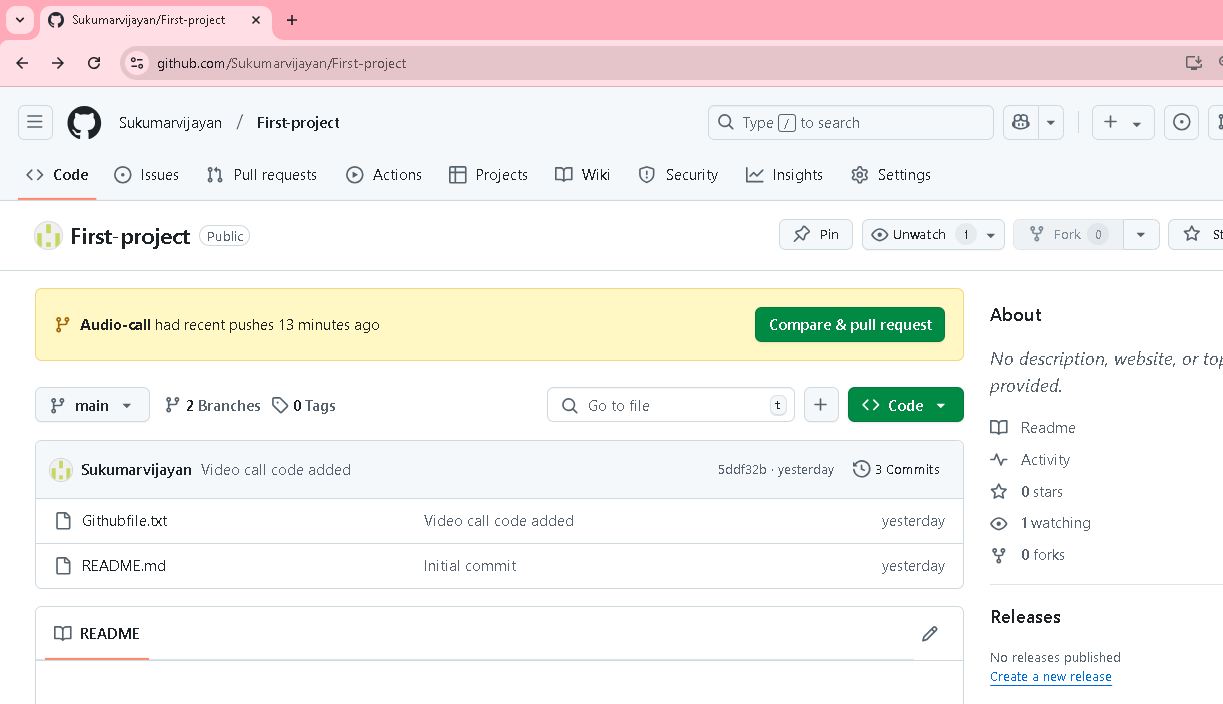


Navigate to main branch & check the new code present..it should be present



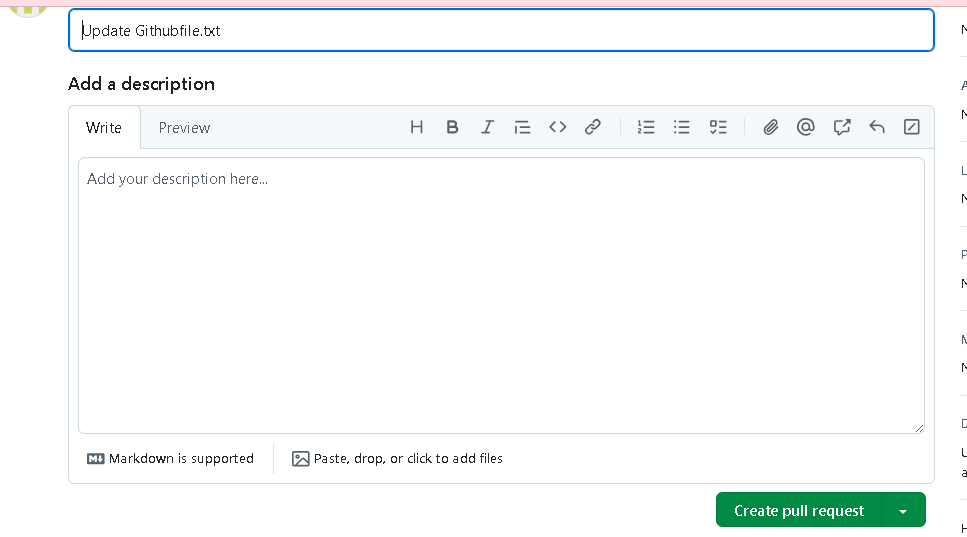
Next merge the code to main branch

Click the Compare & pull request

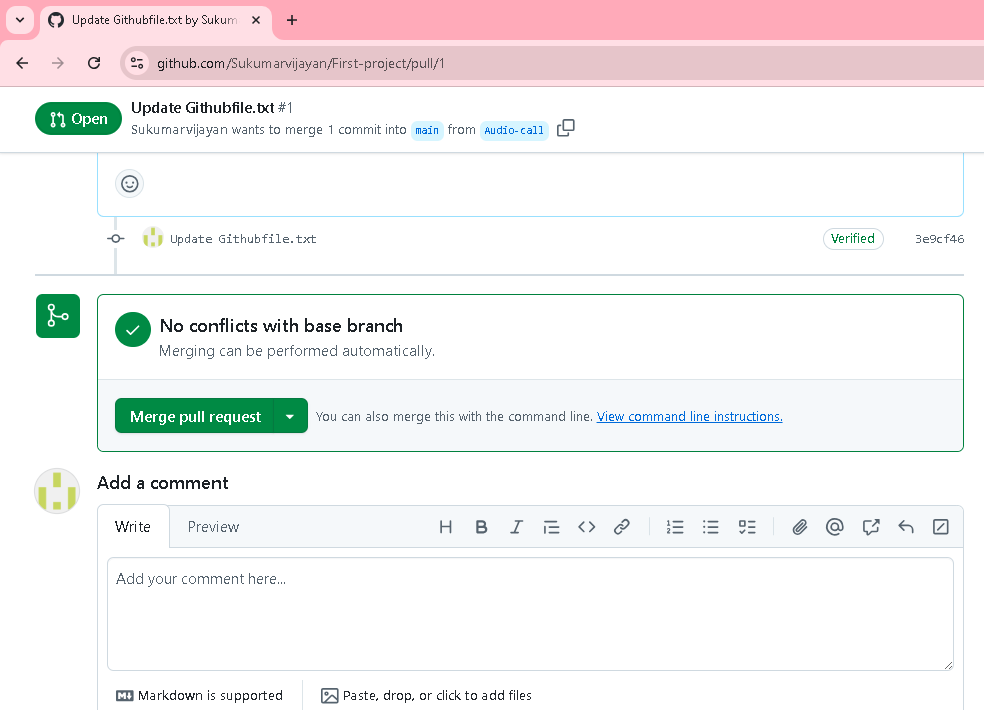




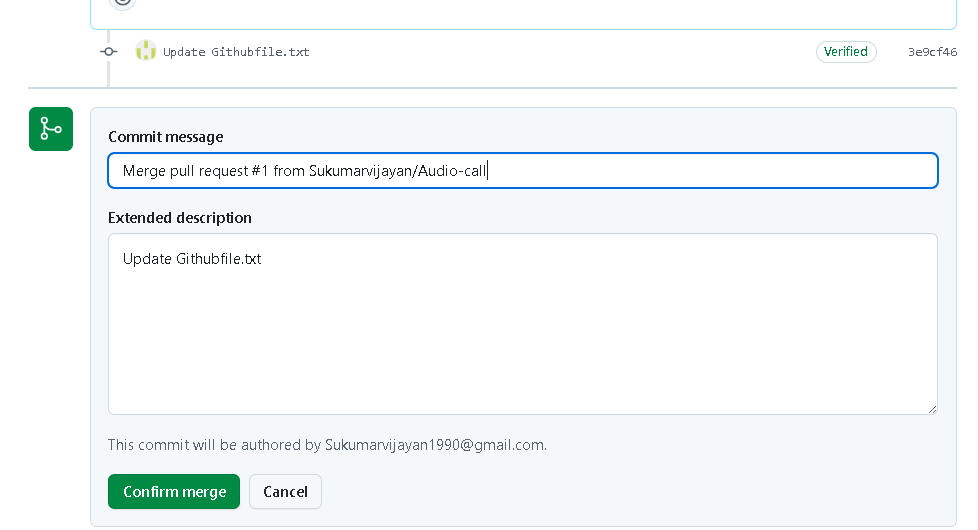
Click create pull request

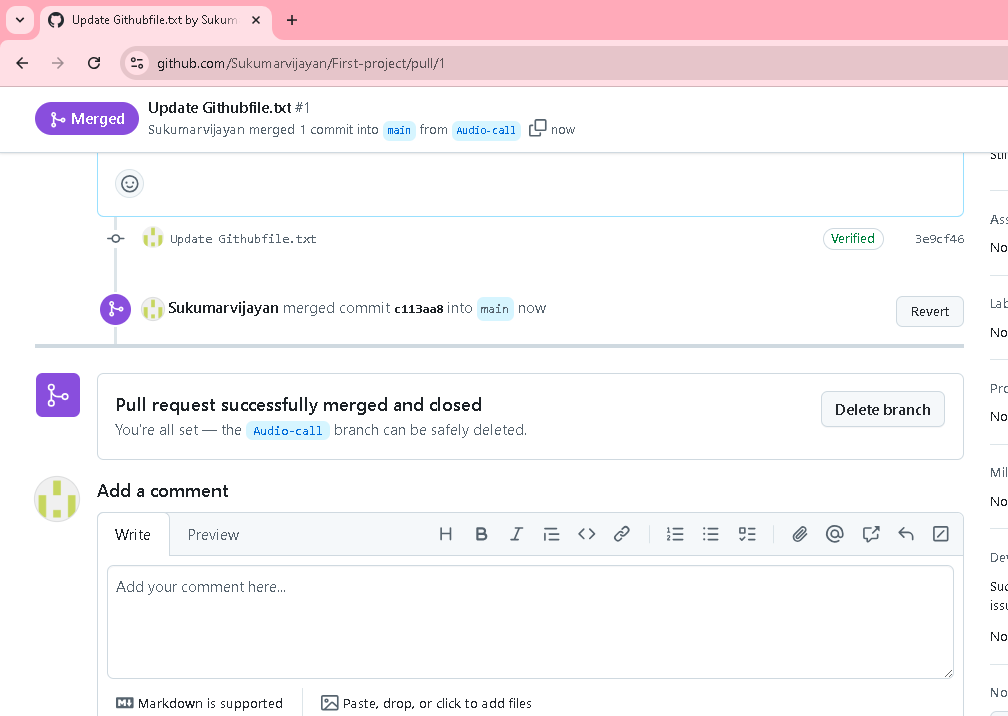


Click the merge pull request



Click Confirm Merge

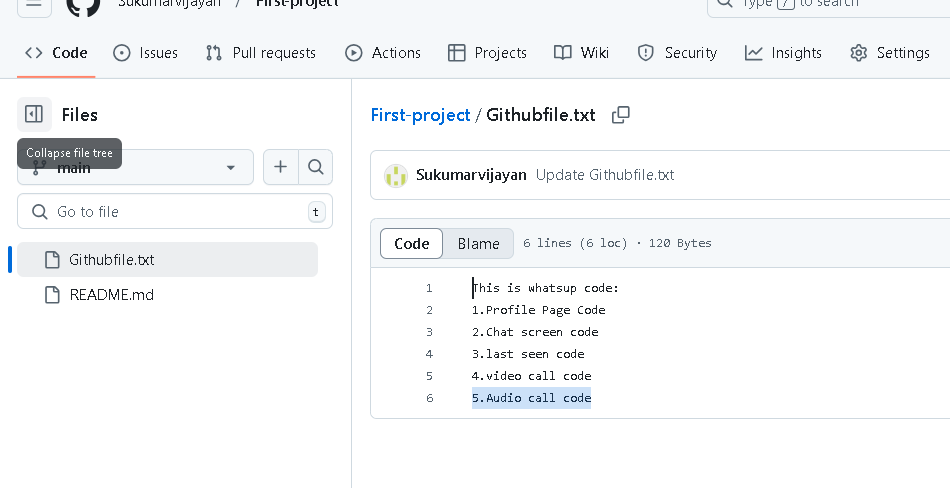




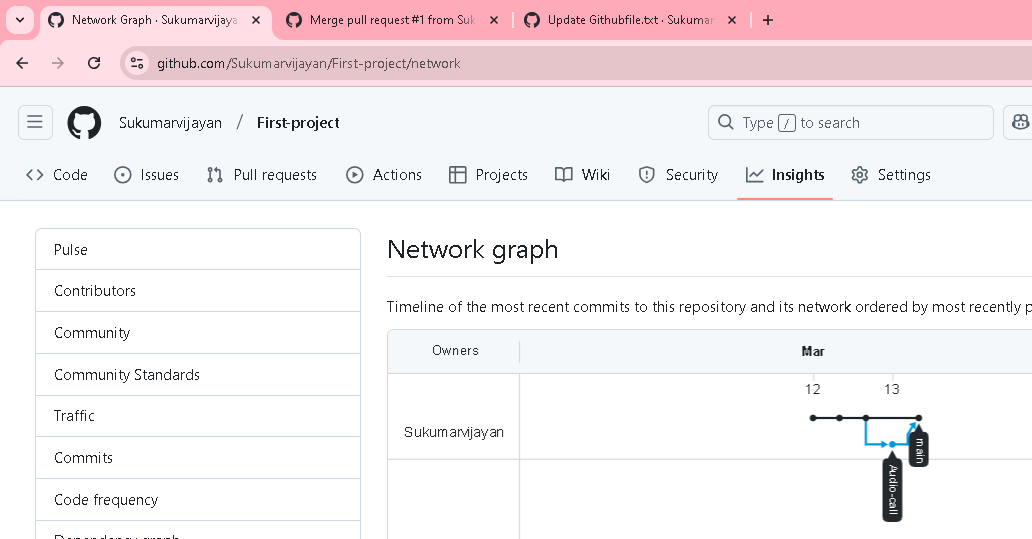
If needed After code merge, we can delete the Audio-call branch by clicking the “Delete Branch” button

After deleting the branch also, we can restore in github..by clicking the “Restore button”[button enable after deletion]

Now go to main branch & check the changes added



Go to insight->network, we can see the main branch & sub branch diagram

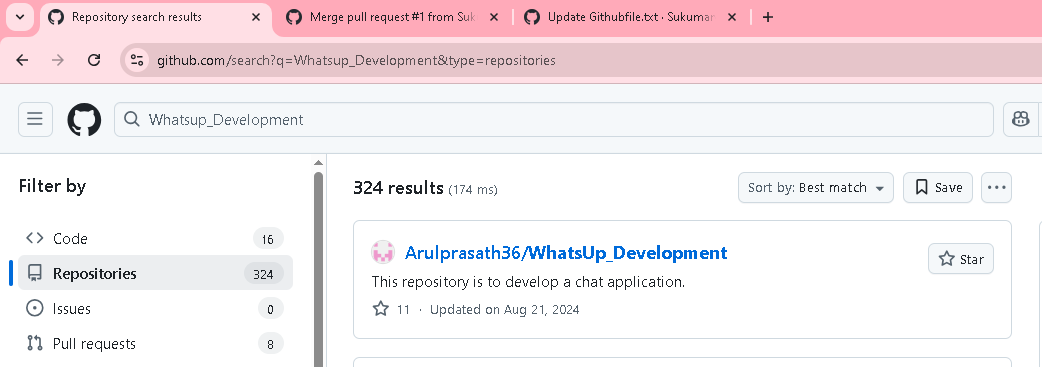


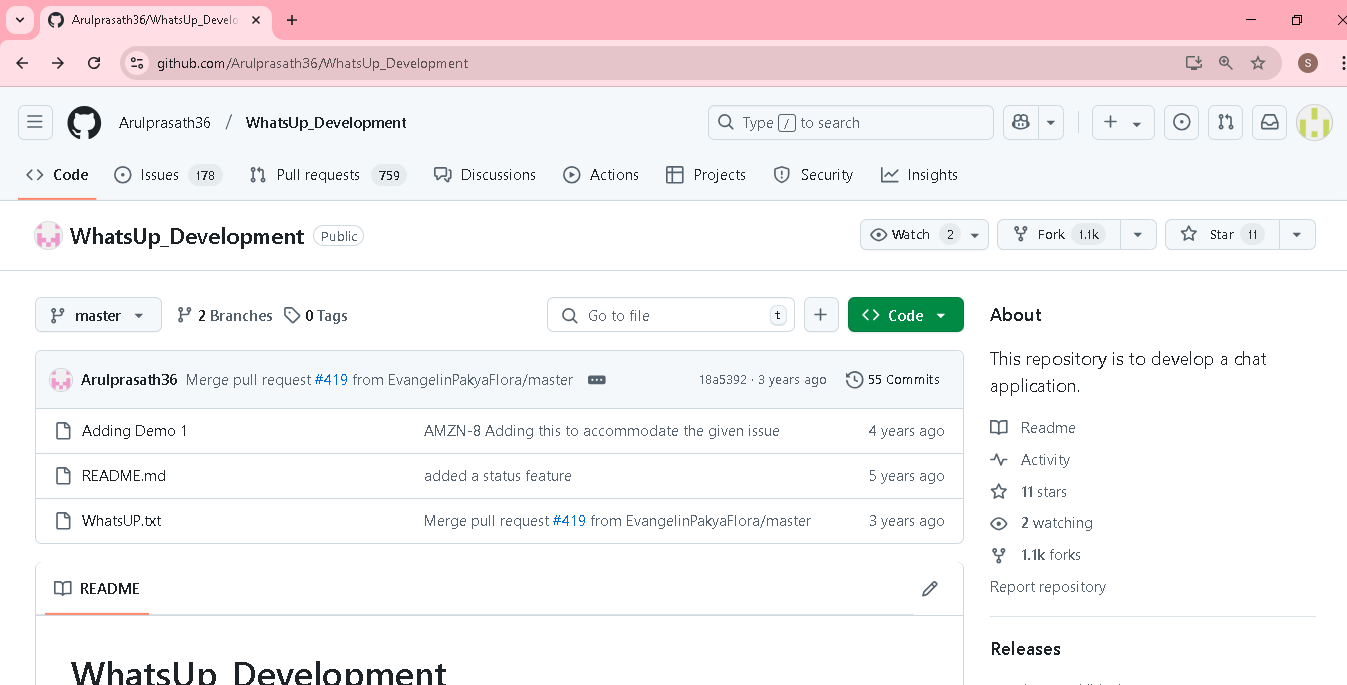
**Fork**

**Now we are going to update other user GITHub repository**

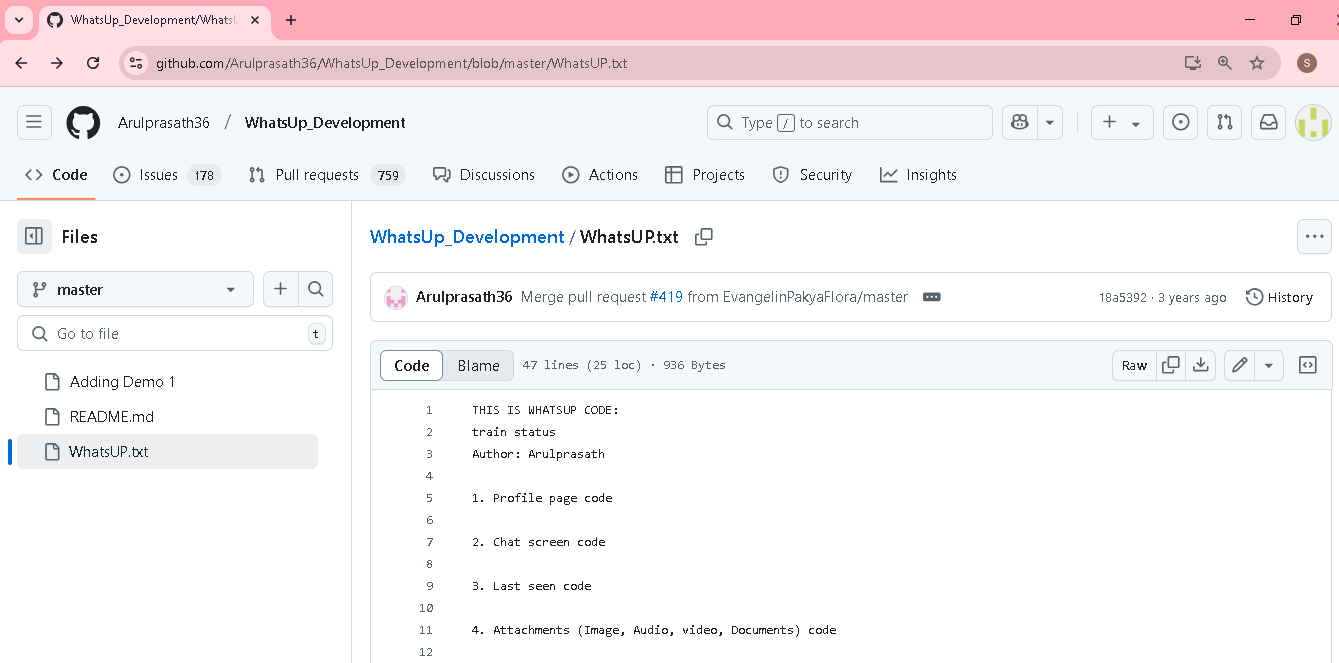
Eg..in online u r seeing arulprasad repository whatsapp application features..you liked the features & in that u r planning to add new features (group call)..bu using fork concept only we can fork(like copy) arul code to our repository & do code modification & send to other repository(arulprasad) tp merge

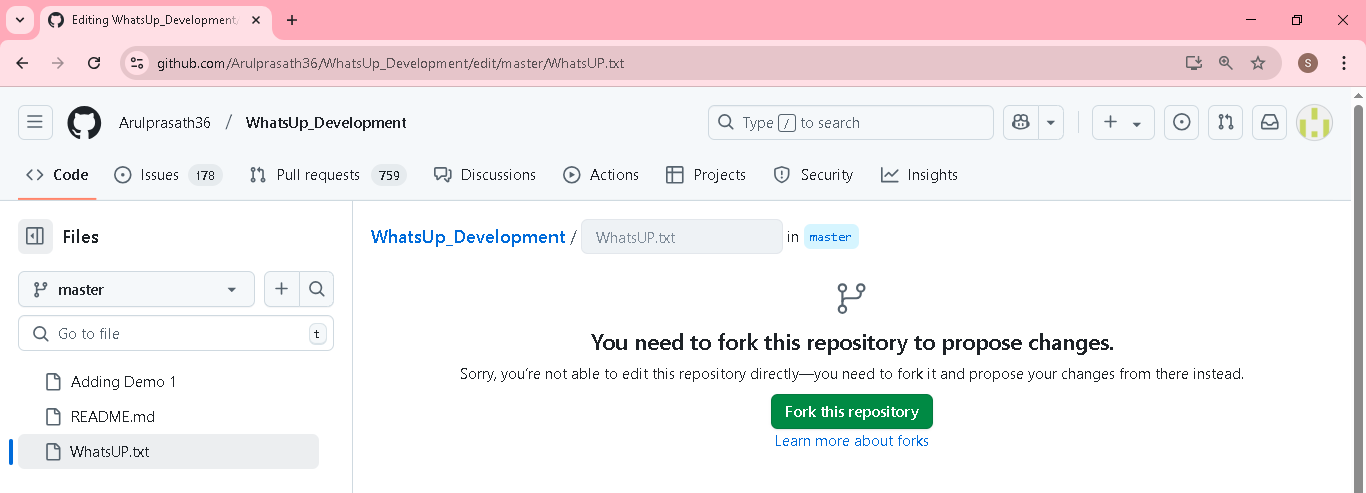
In the search Whatsup\_Development enter..we can able to below repositiry [ Trainer Arul github]





Click the edit button





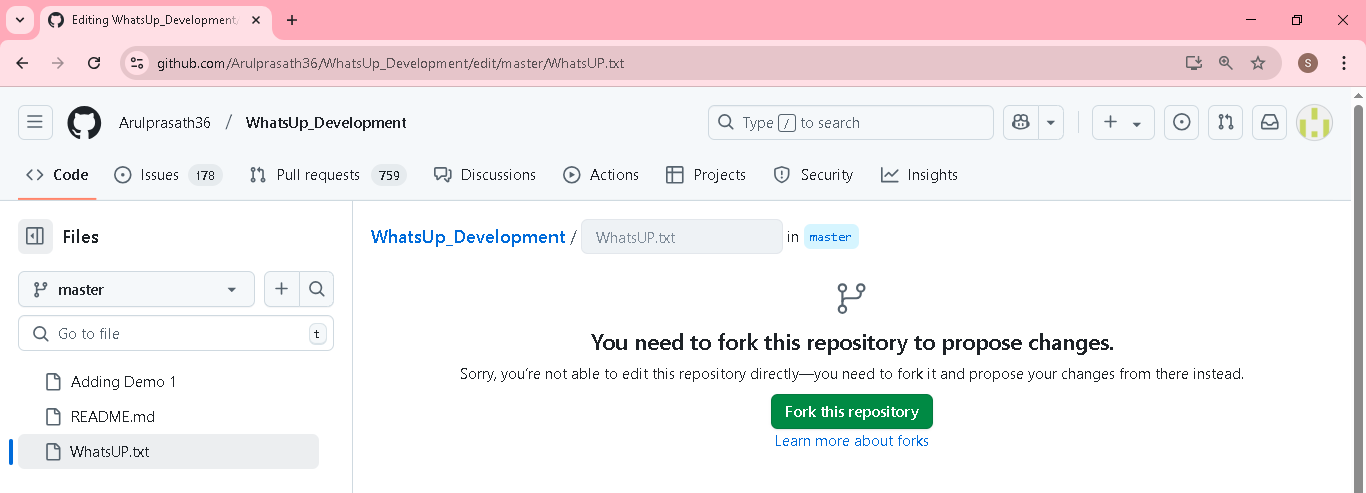
Above message received after clicking the edit button..we cant edit another repository directly..here we are using fork concept

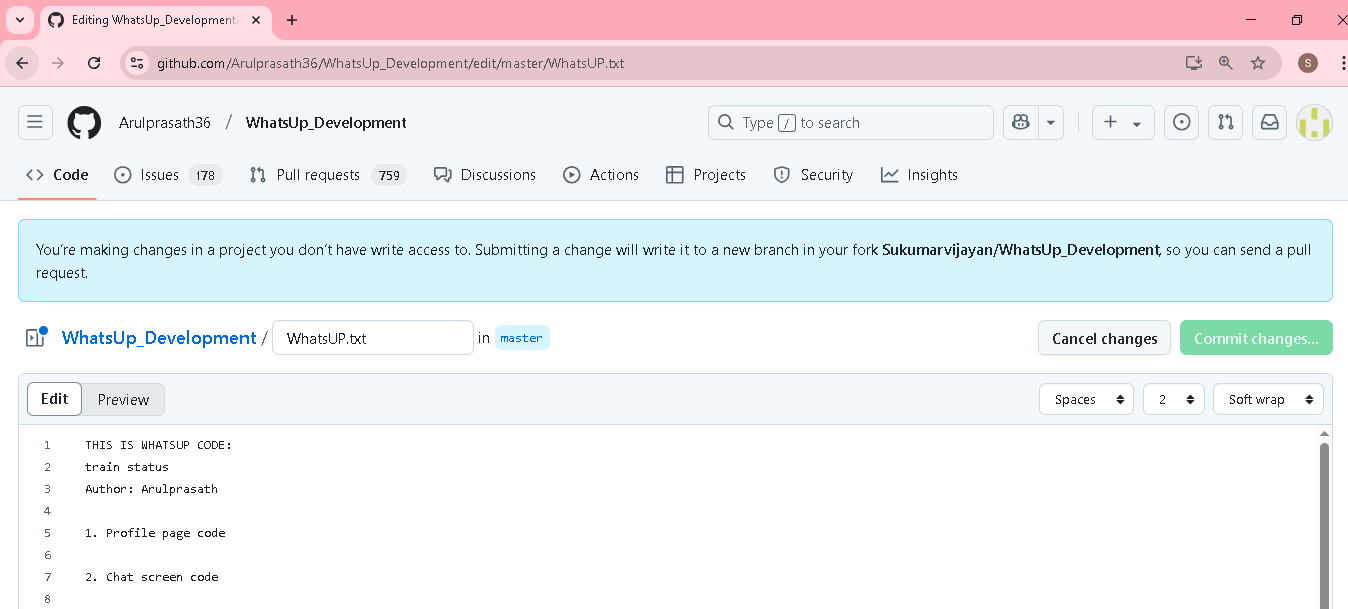
**Fork**:

**When a community wants to contribute to the repository, they don’t have permission to edit. They can fork and make the changes and make a pull request.**

**The original author of the repository can verify the pull request and accept or decline the pull request**

Click the “Fork this repository”

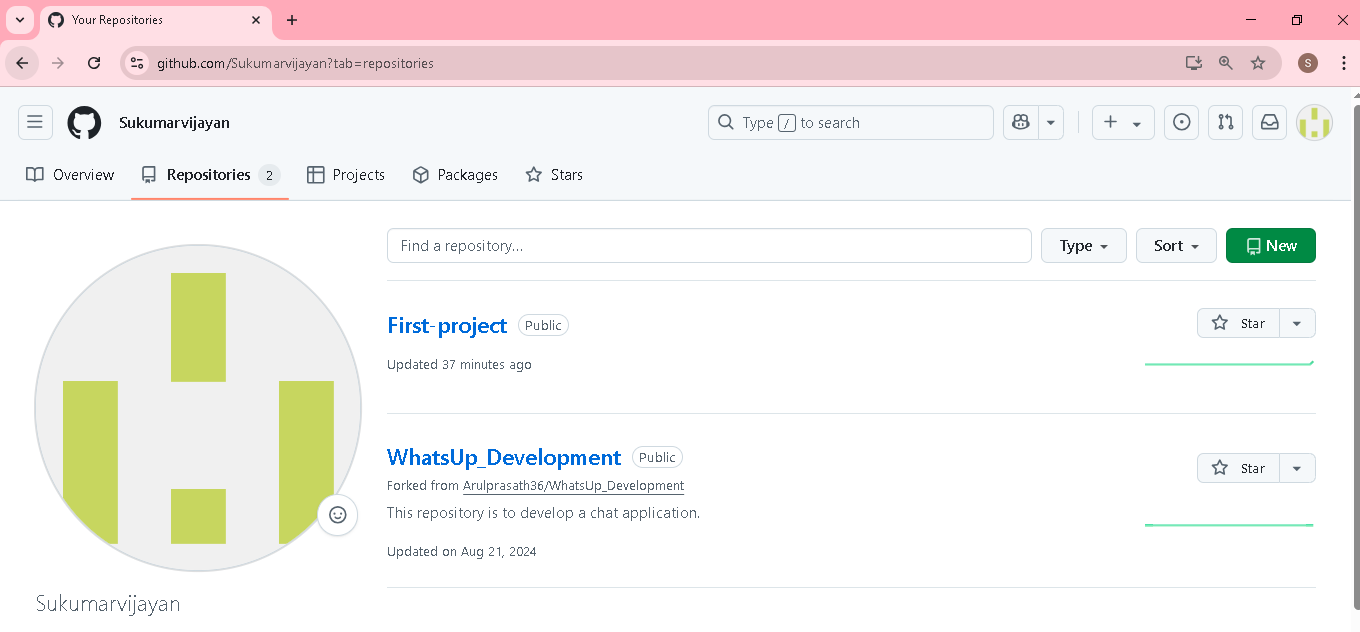


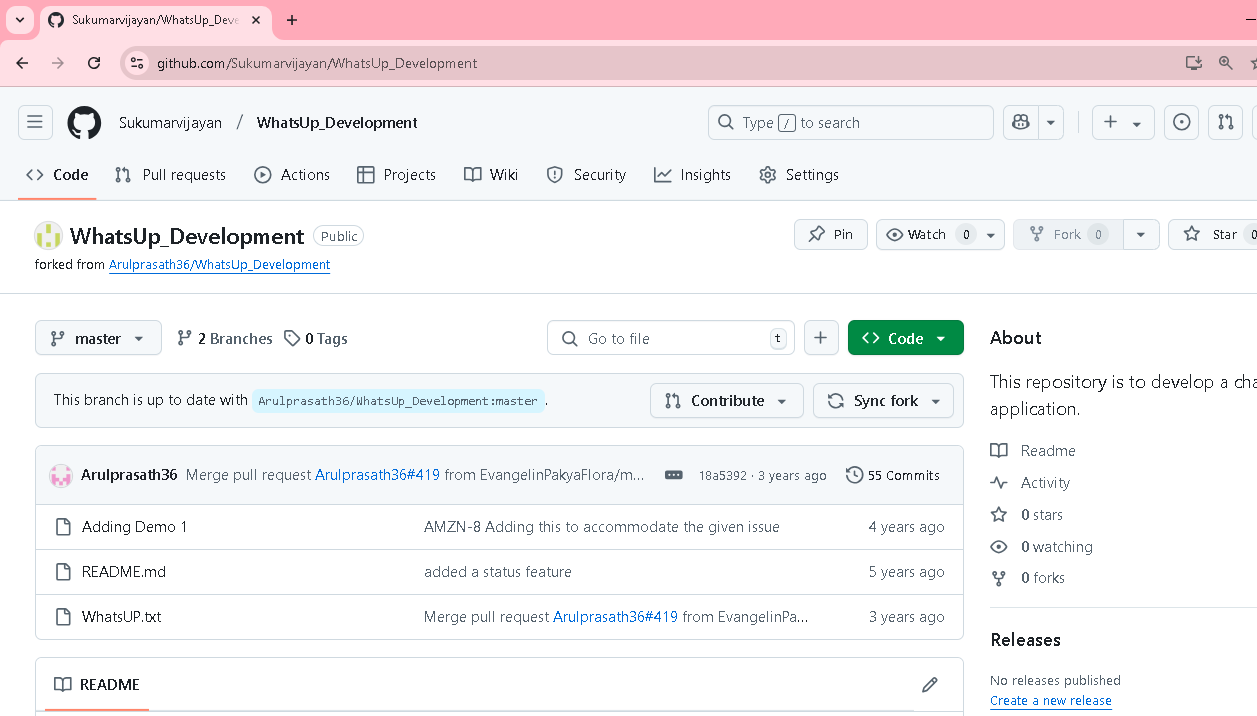


Now come to my repository..then u can see the fork from arulprasad repository & displayed in ur repository

We cant directly update other repository, we need fork from other repository to our repository.Do the code changes & pull to other repository(arulprasad)

Arulprasad only need to review & merge with main master of arulprasad repository

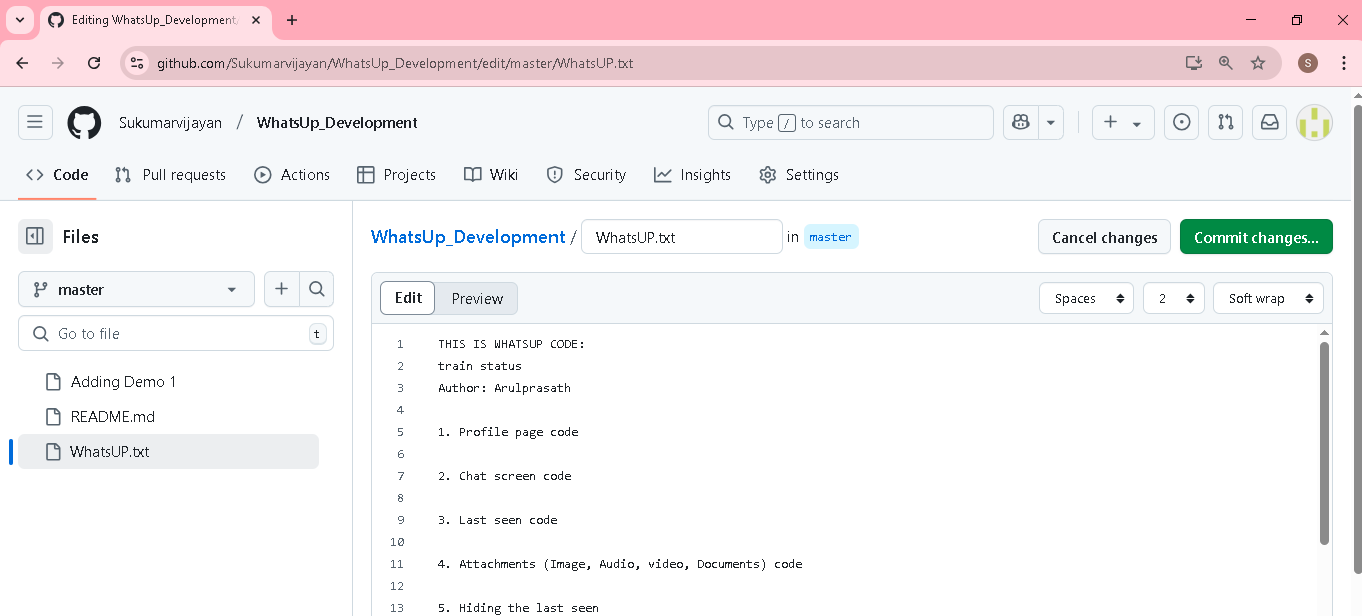


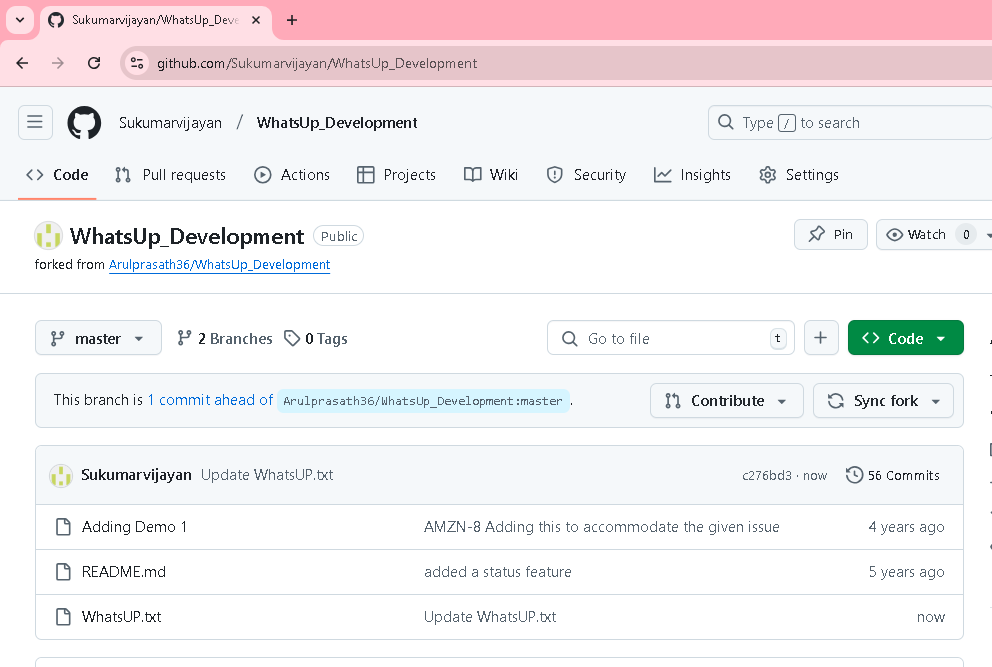


Edit & add the code

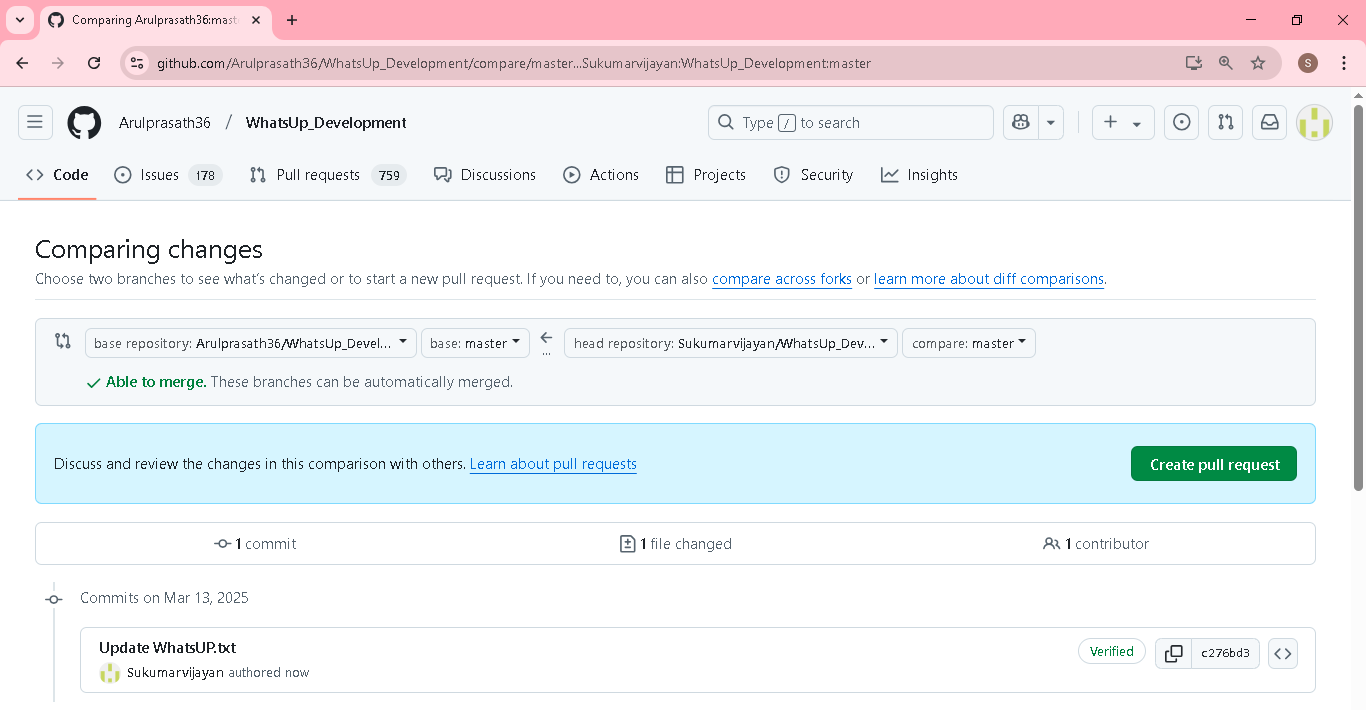


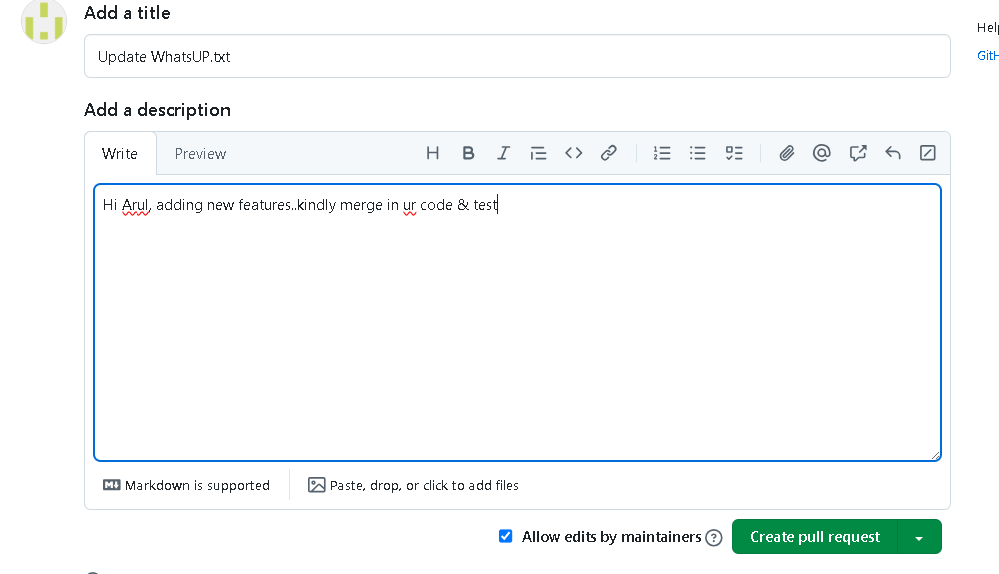
Click commit changes

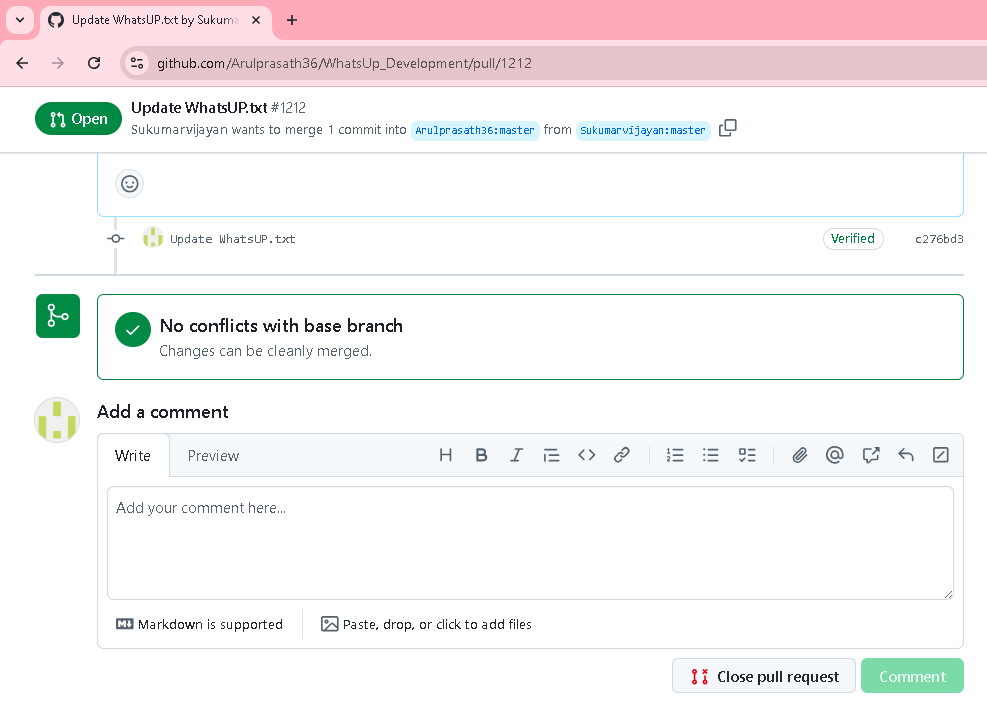




Click create pull request







Now arul only having access to review & merge the code with main code

**Merge Conflict in GITHUB**

You have opened your code to the community, you are receiving lot of pull request

When trying to merge the requests from 2 different users, you are getting an error called **“This branch has conflicts that must be resolved”**

Conflict – clash, disagreement

When 2 different persons trying to modify the same line of code. There arises a conflict

So in order to merge the code, these conflicts has to be resolved first

**Repository owner will review both the code & accept any one or modify to keep both also**

**First click “resolve conflicts” button** ->it will show the conflicts

Repository owner will review & keep best code & remove another code

& then Commit change->Merge pull request->confirm merge

**To error in Human**

Nothing is prefect in this world..there is always a room for improvement

Our code is not exception from this..what if someone wants to raise some concerns over our product.

It may be reporting a bug or requesting for help, suggesting new features and so on

**GITHUB Issue**

The issues option will help us to address this requirement.Any user can submit a new issue to our software code.

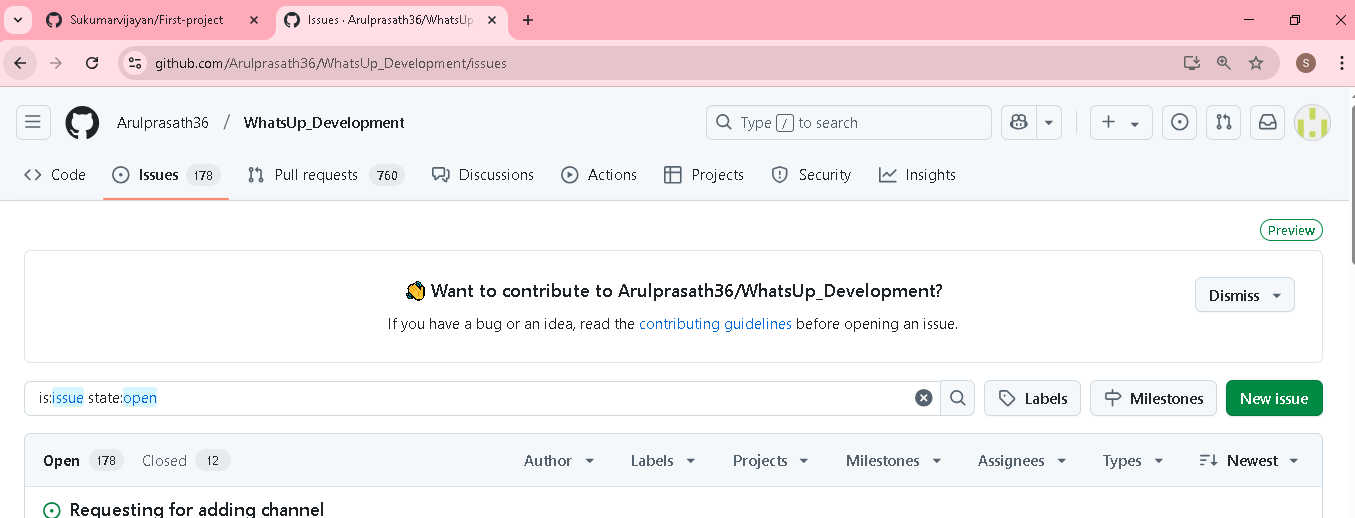
The owner then can assign the code to him/her or any of their team members, classify that issue type, close the issue, comments on the issue & so on.

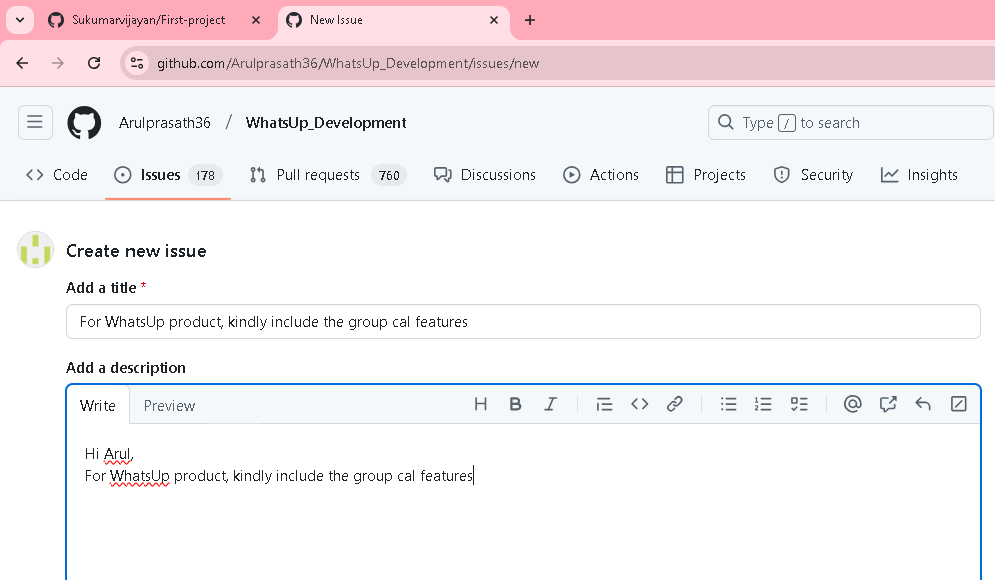
Eg...Sukumar review the arul code in Github repository..he found one issue in that code..so he can raise issue & assign to arul..

Arul can review the comments in issue..he can enhance the product or correct the code or ignore.

**Go to arul account**

**Click issue & new issue**

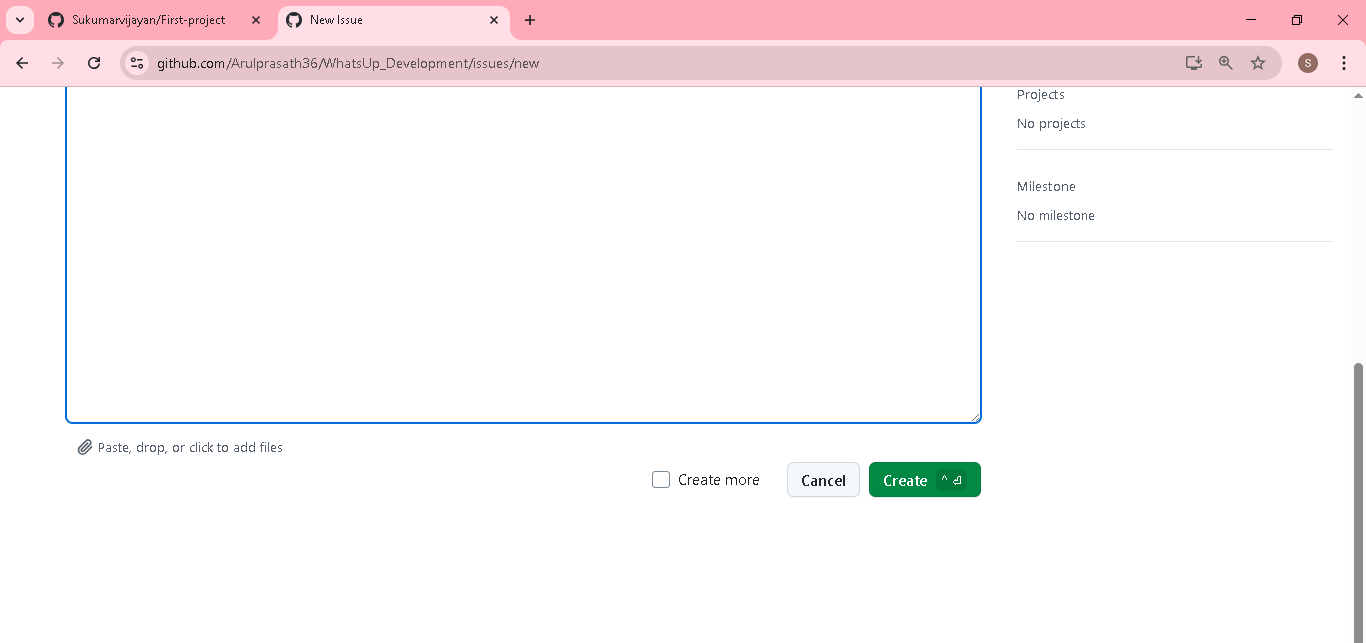
****

****

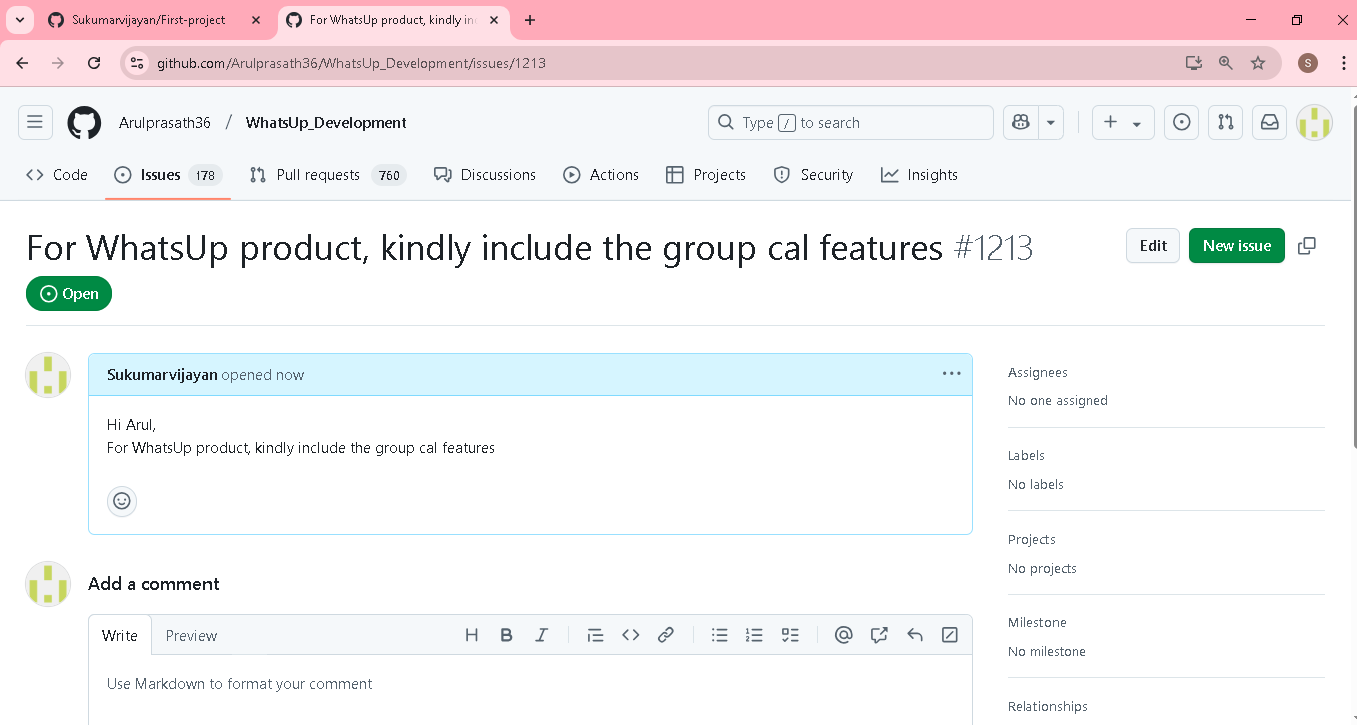
**Enter issue title, description**

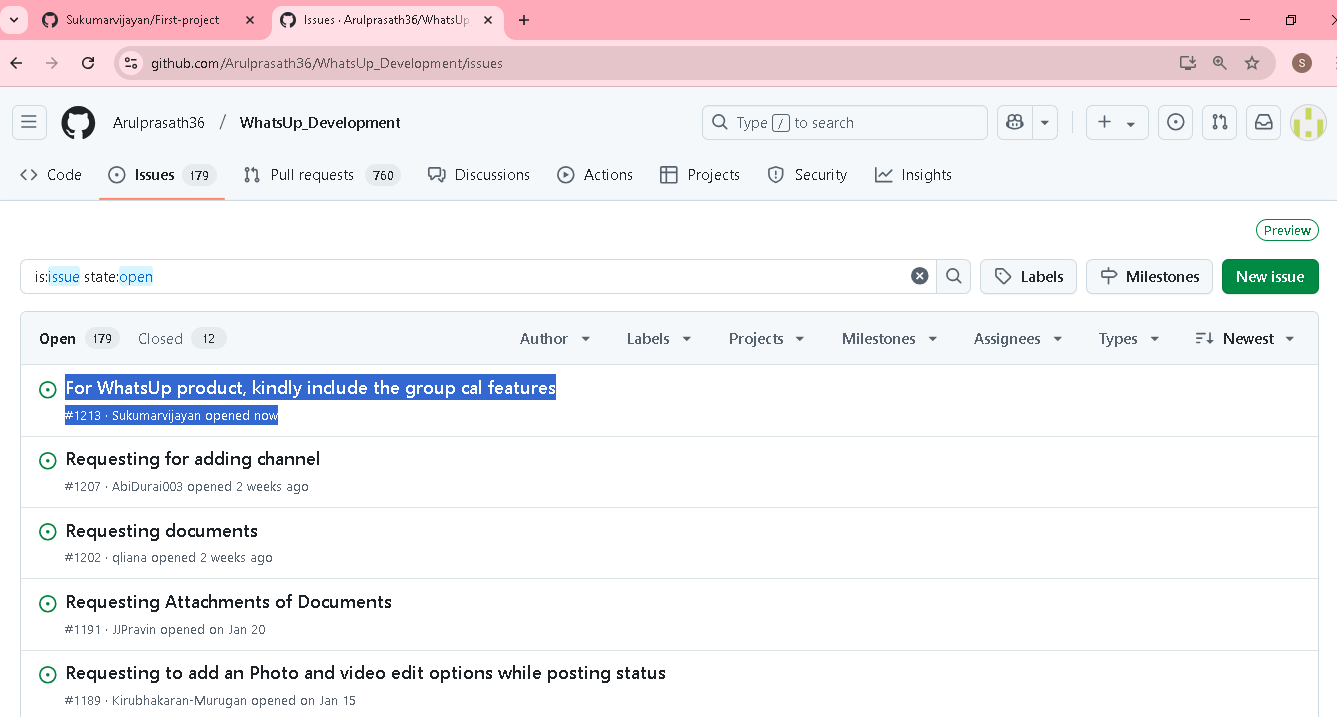
**U can provide the codes, attach any link, image in the above screen**

**Click Create**

****

**Issue created & sent to arul**

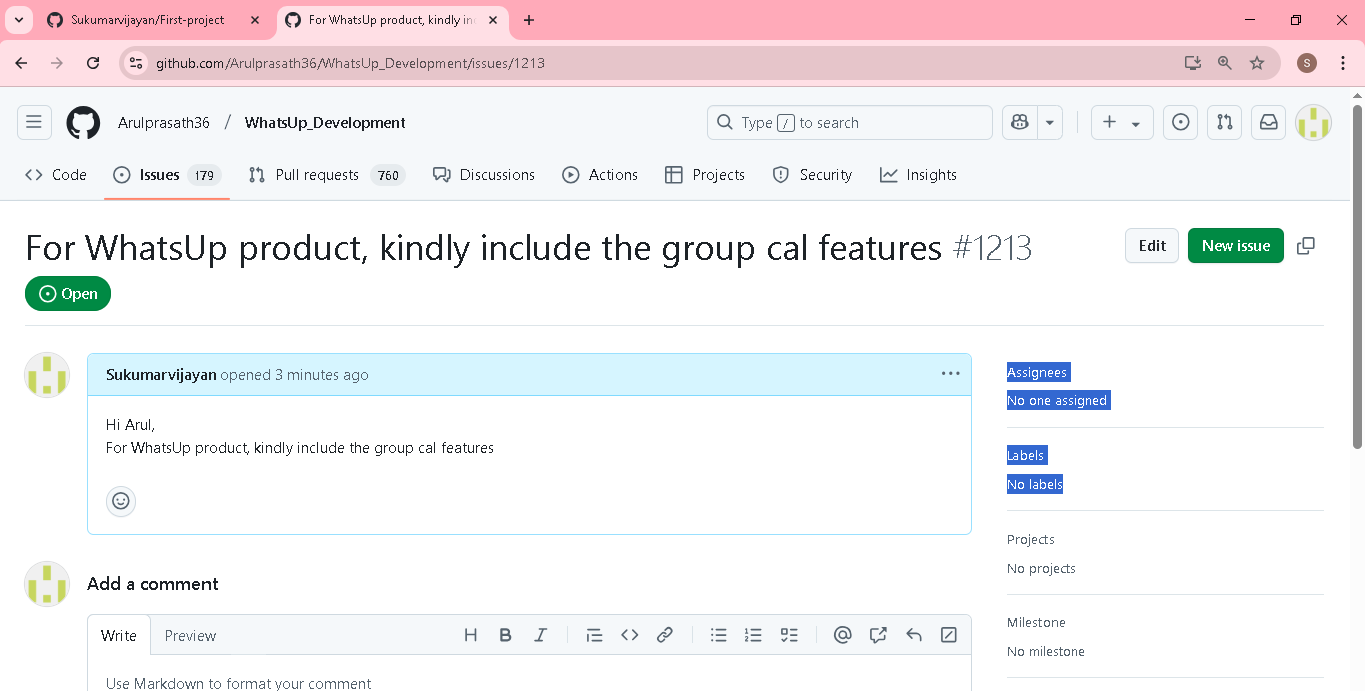
****

****

After that , arul need to review the issue .. he can enhance the product or correct the code or ignore.

Assignees, labels etc present in issue section..

So arul can assign the issue to the team , so that team can work on that issue [ in real project, multiple team wil work on same repository..owner review the issue & assign accrodgly]

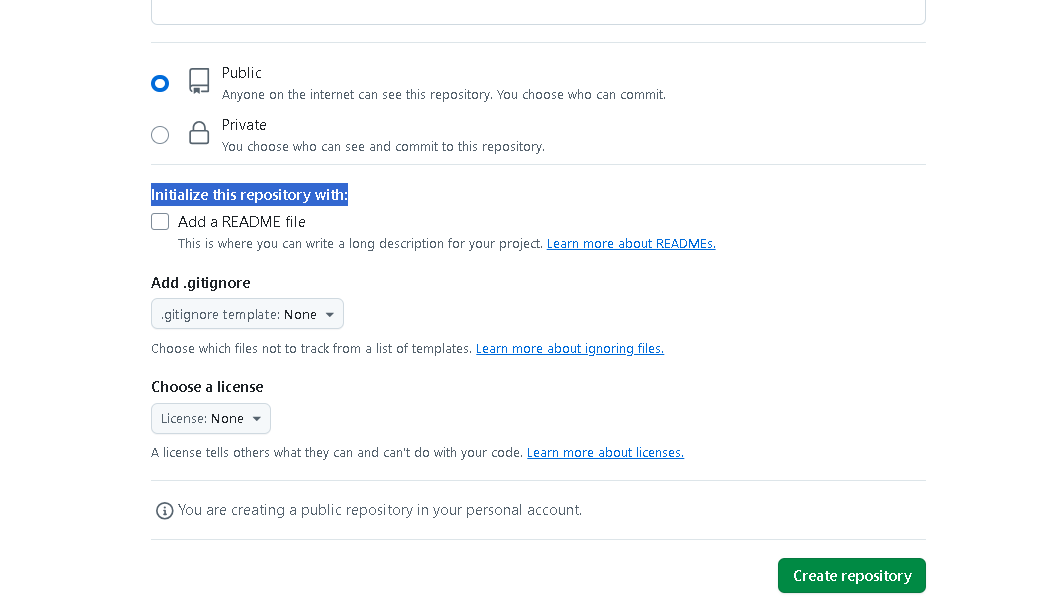


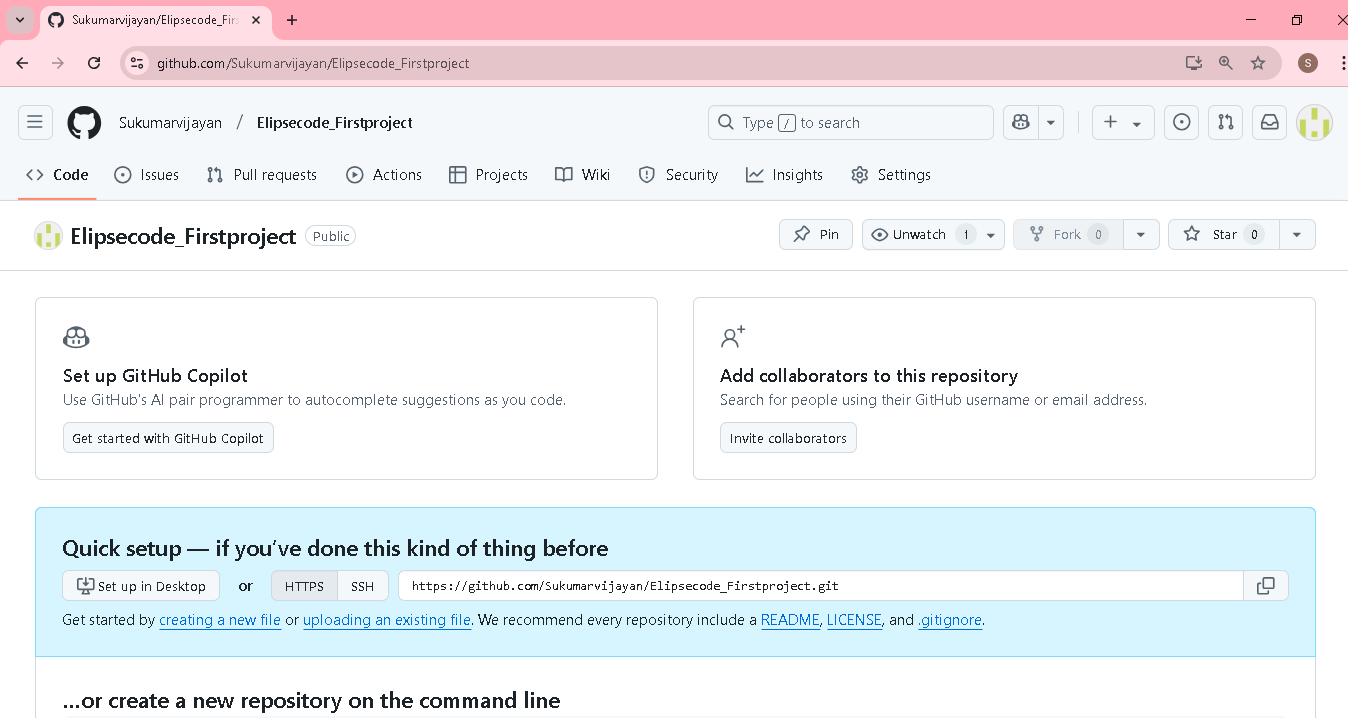
**GitHuB with Eclipse**

How to upload our code in eclipse to github

In GitHUB , create a repository [Dont select **Initialize this repository with, it will create a empty repository & contains some instructions, during terminal we wil discuss further on this]**







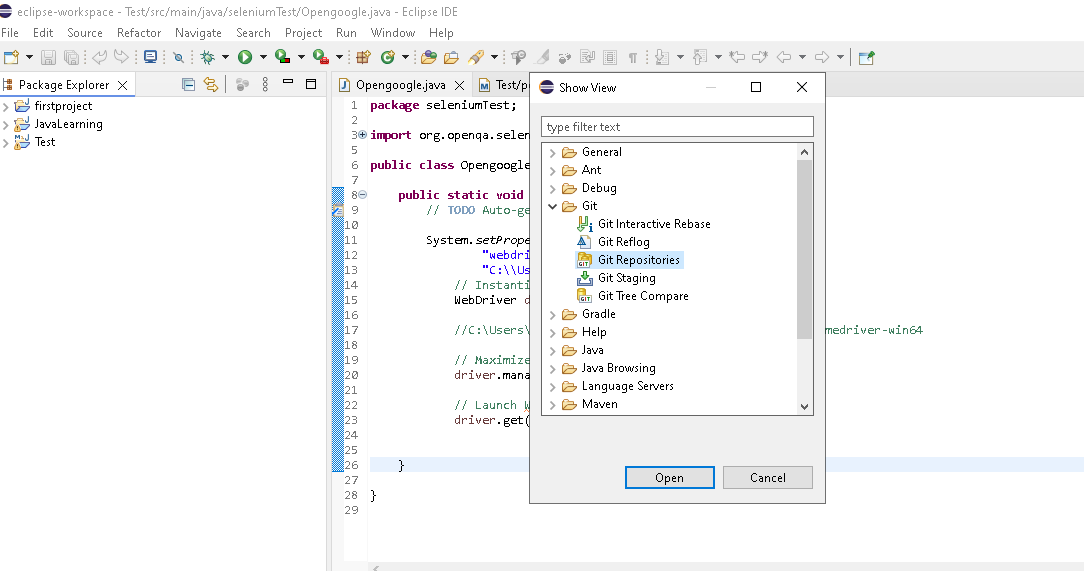
Its a repository URL

Copy this <https://github.com/Sukumarvijayan/Elipsecode_Firstproject.git>

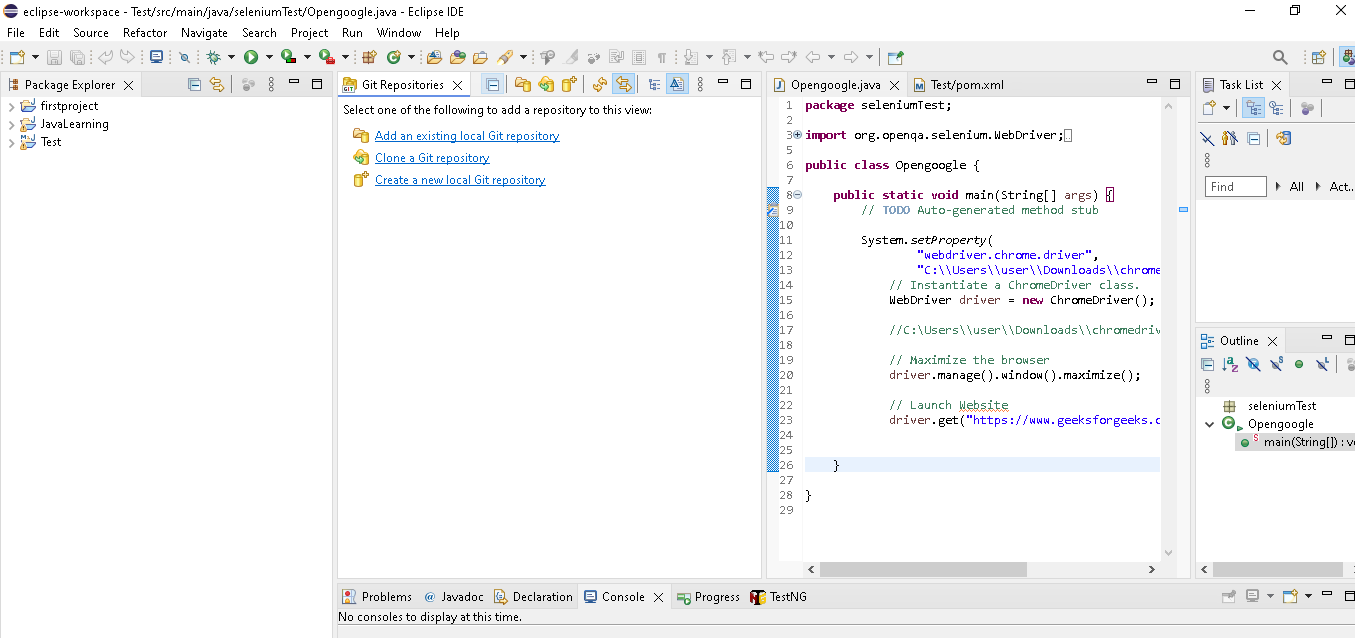
**Go to ellipse**

**Windows->show view->Other**

**In below window , select “Git Repositories” & click Open**

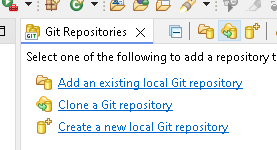
****

**Window will open like below**

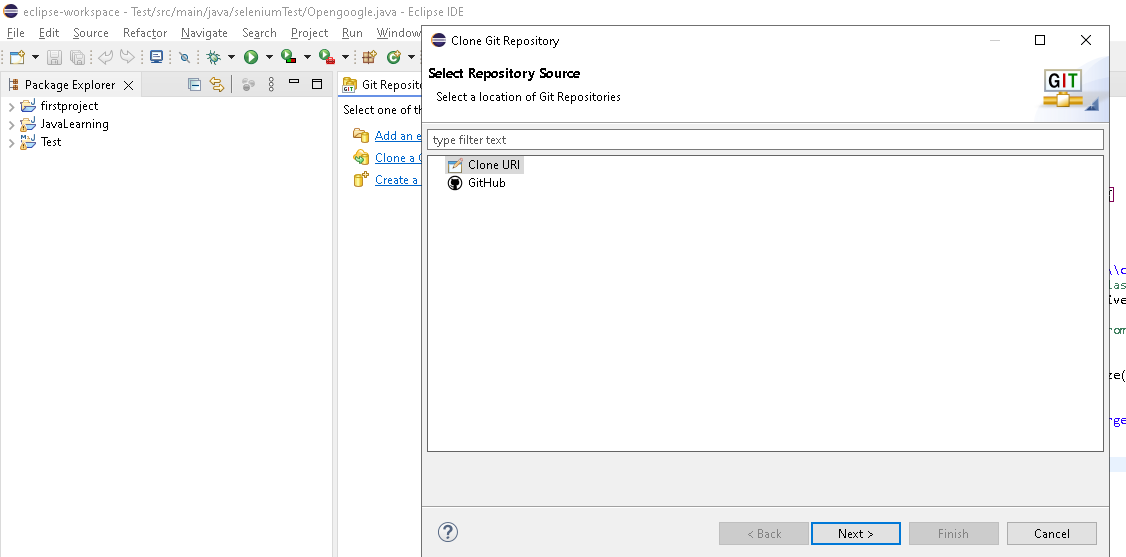
****

**3 icon display, in that select “Clone a Git repository” [ as we already create a repository in GITHUB, by using this option we are moving the GIT repository to elipse]**

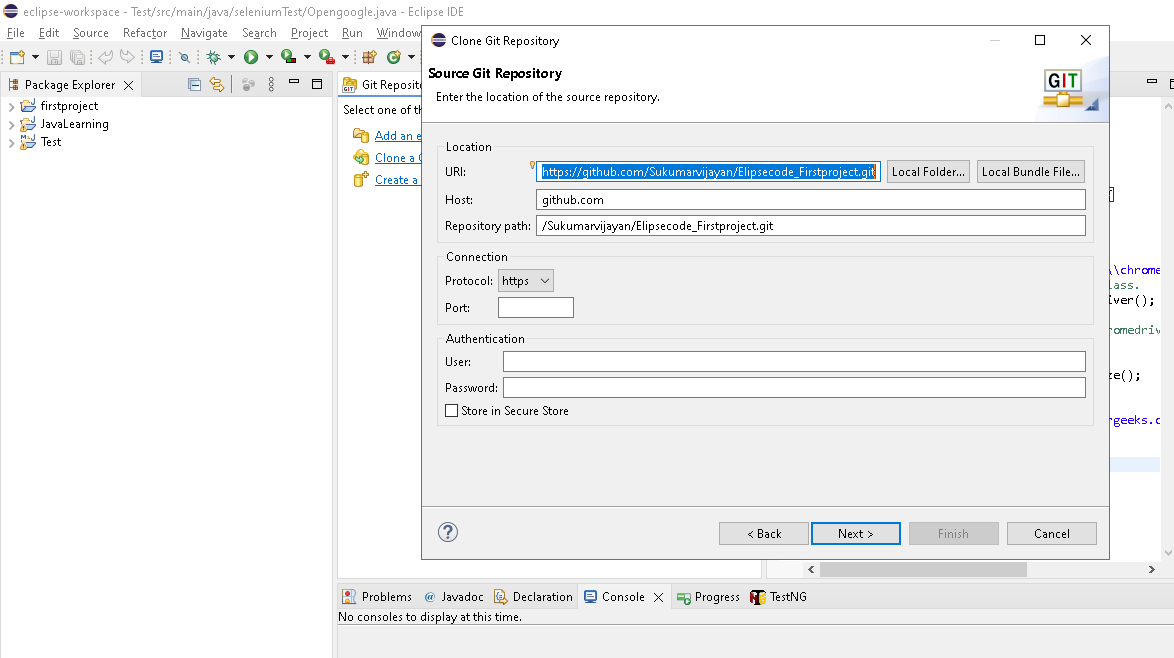
**-Create a new local Git repository [ if repository not created in GITHUB]**

****

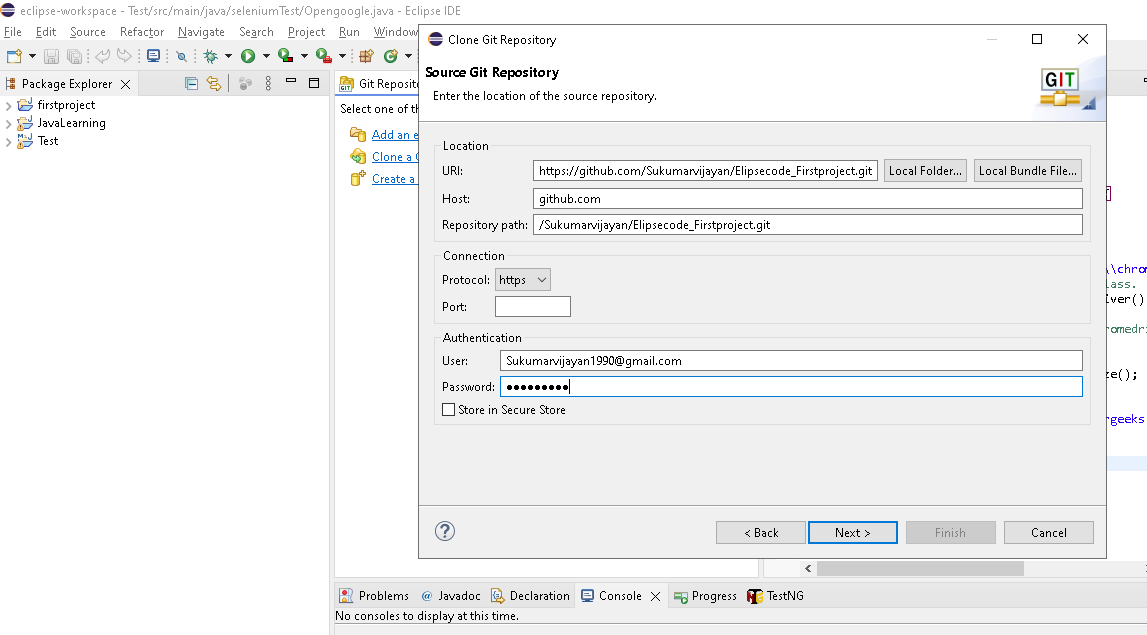
Select Clone URL & click Next



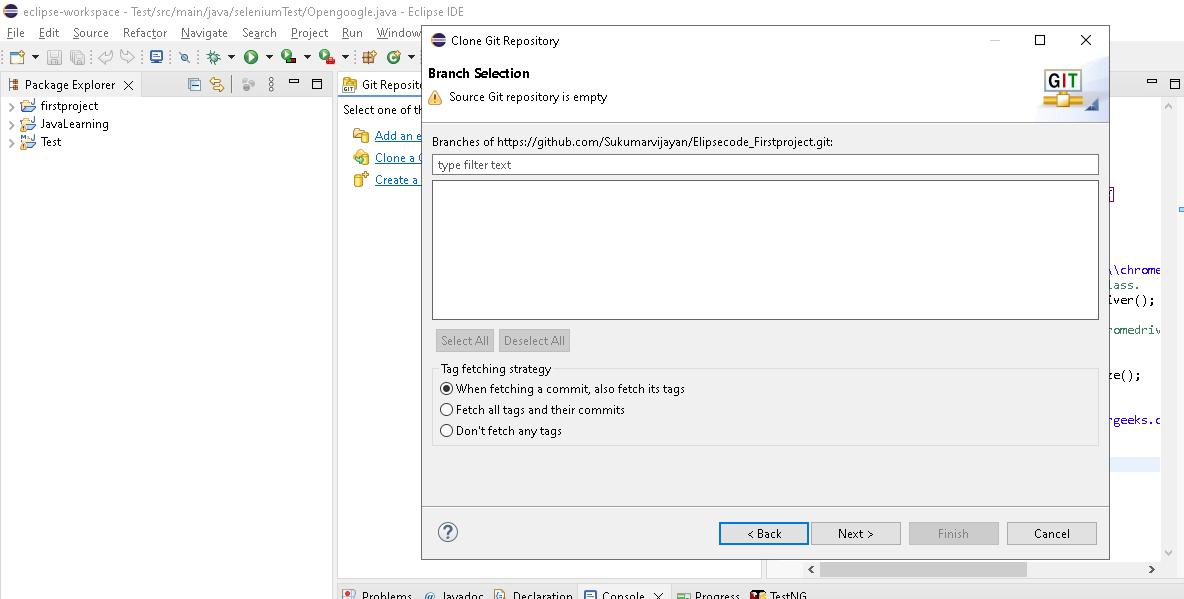
URL auto populated, as we copied the url from git



Enter GIT url & password & click Next

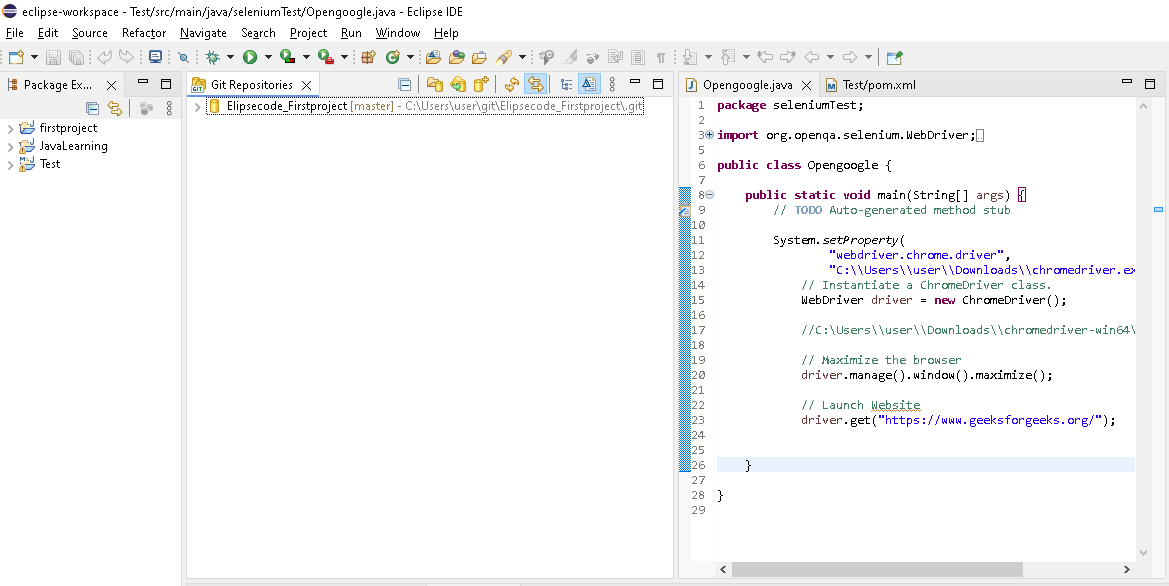


Its will show source repository is empty[i.e is GIT repo]..no problem, just click next





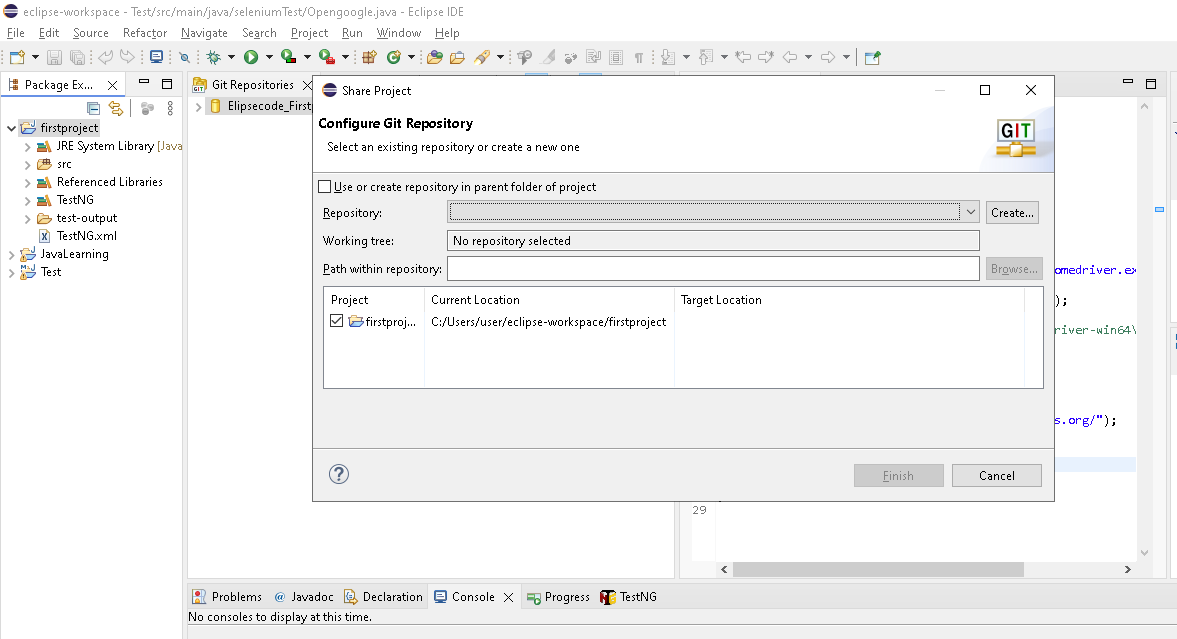
Click Finish



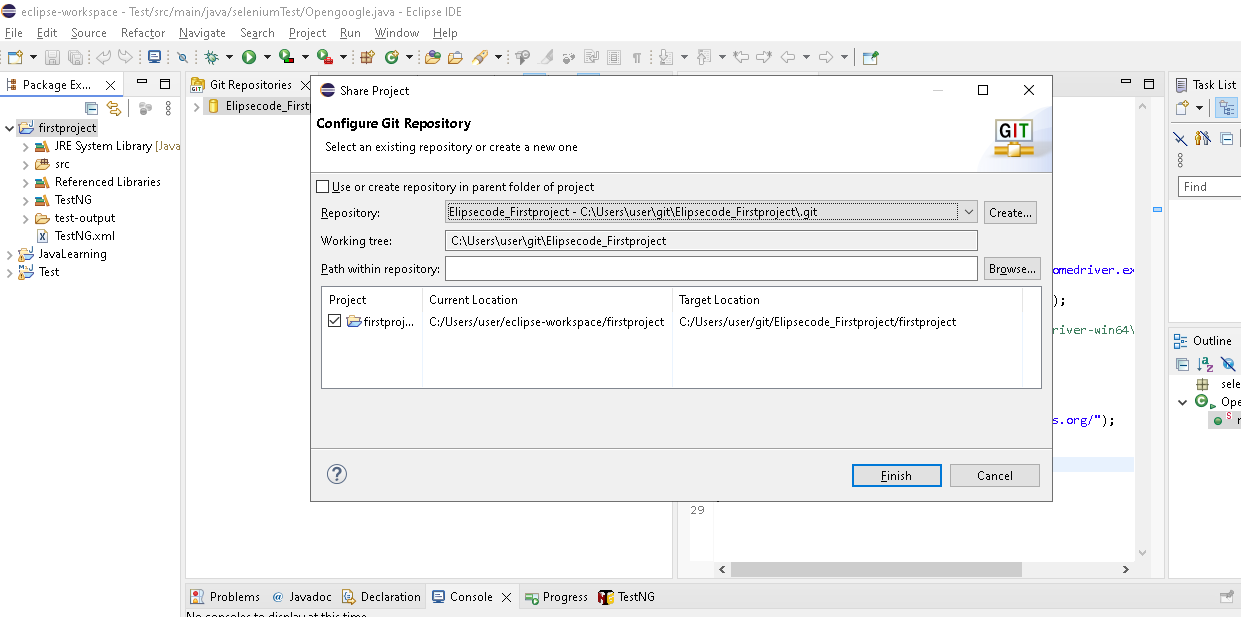
C:\users\user\git\Elipsecode\_firstproject\git ->local Git repository in ur system

First we will move the project code to local Git repository in ur system & then moving to Online Git repository

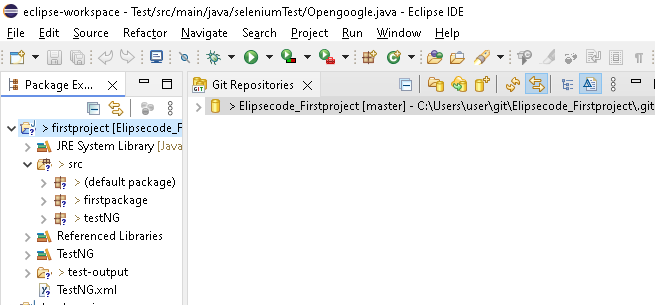
Right click the project(First project in elipse)->Team->Share project



In the repository dropdown, u need to select & click finish



After that, u can see ? in the First project in elipse

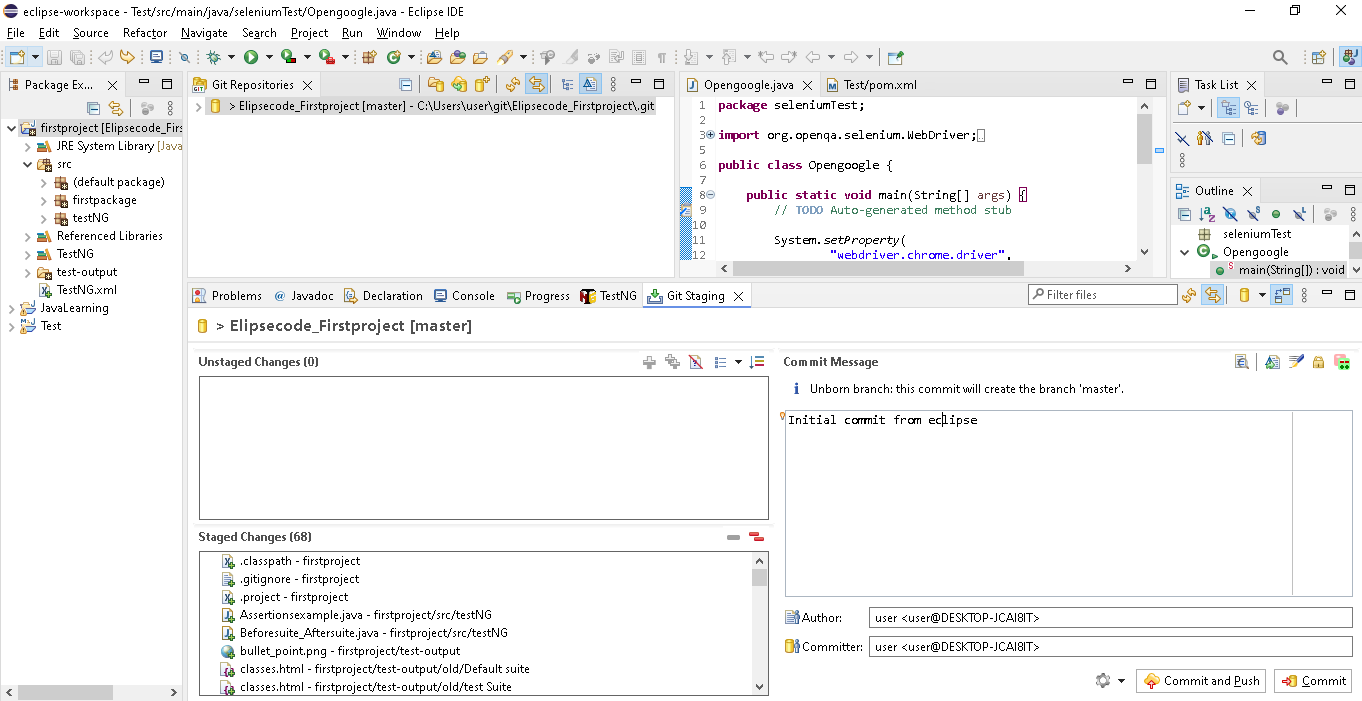


Right click the project(First project in elipse)->Team->Add to Index

? changed to +

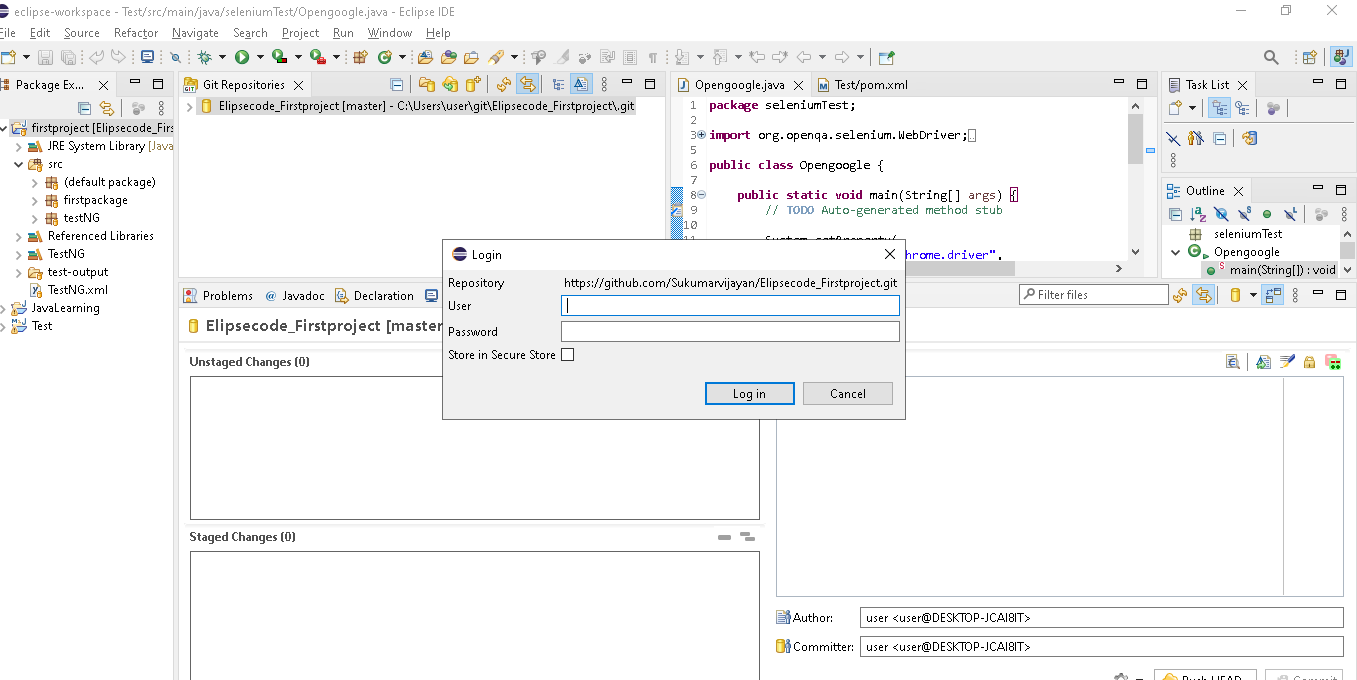


Right click the project(First project in elipse)->Team->Commit



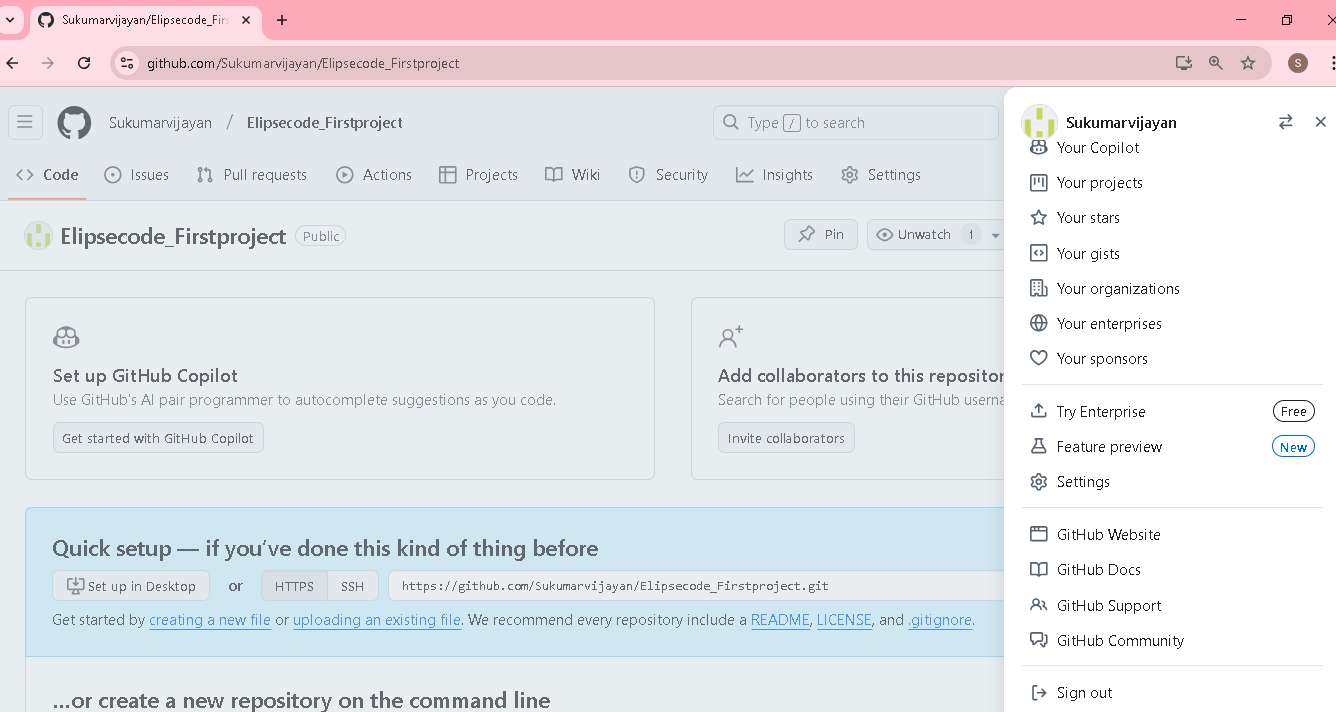
Provide the commit message

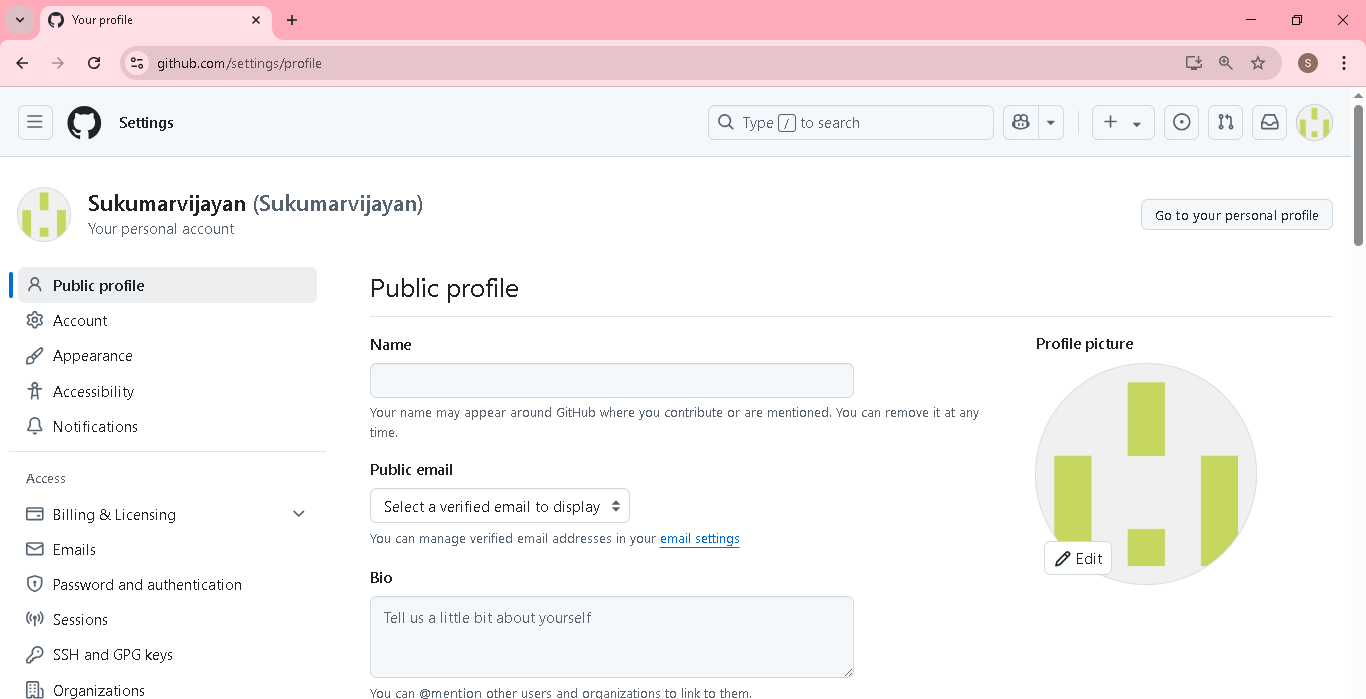
Click “Commit & push”



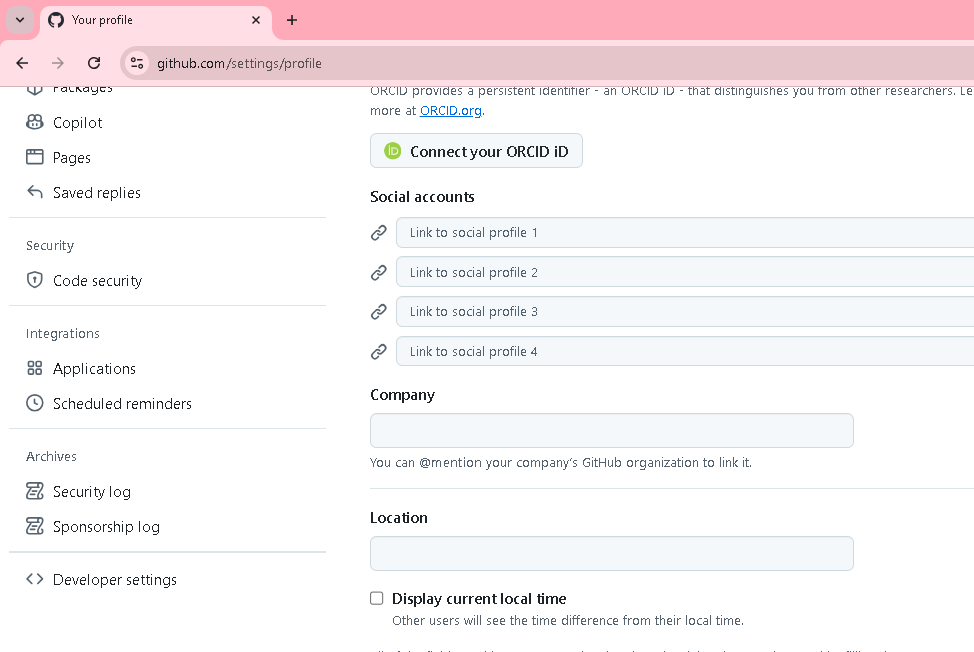
In the above username & password..dont give normal id & pw

Go to github ->Profile Icon->Setting

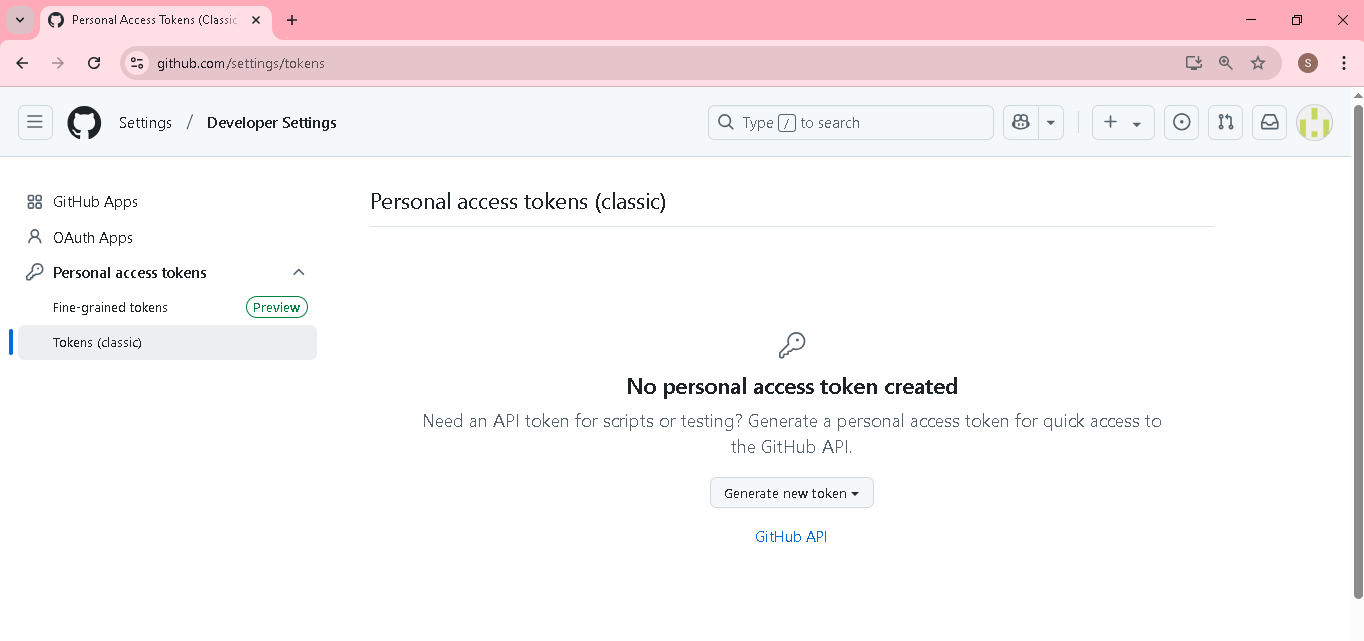


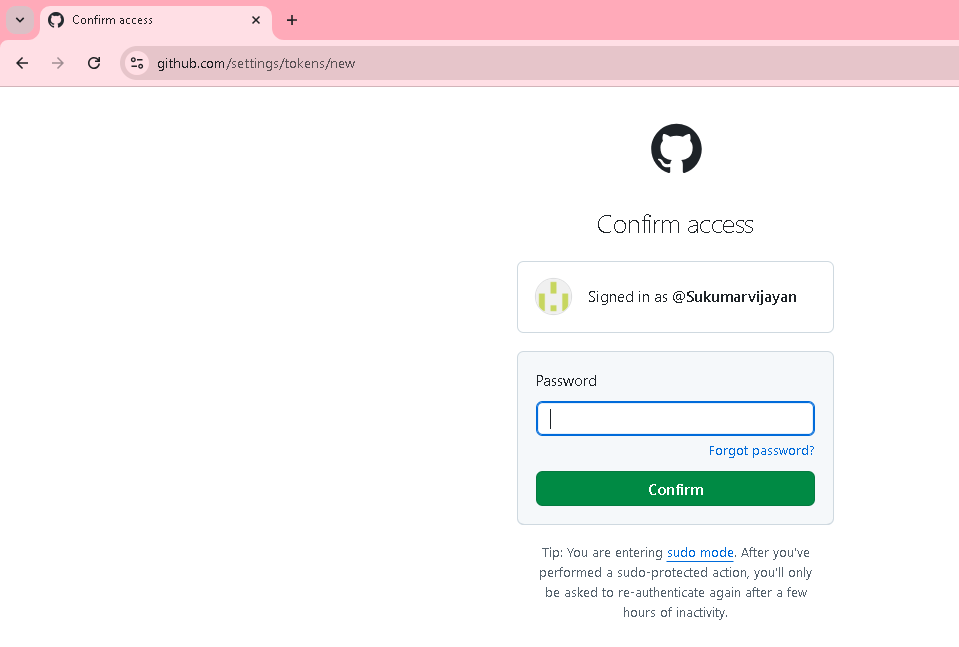


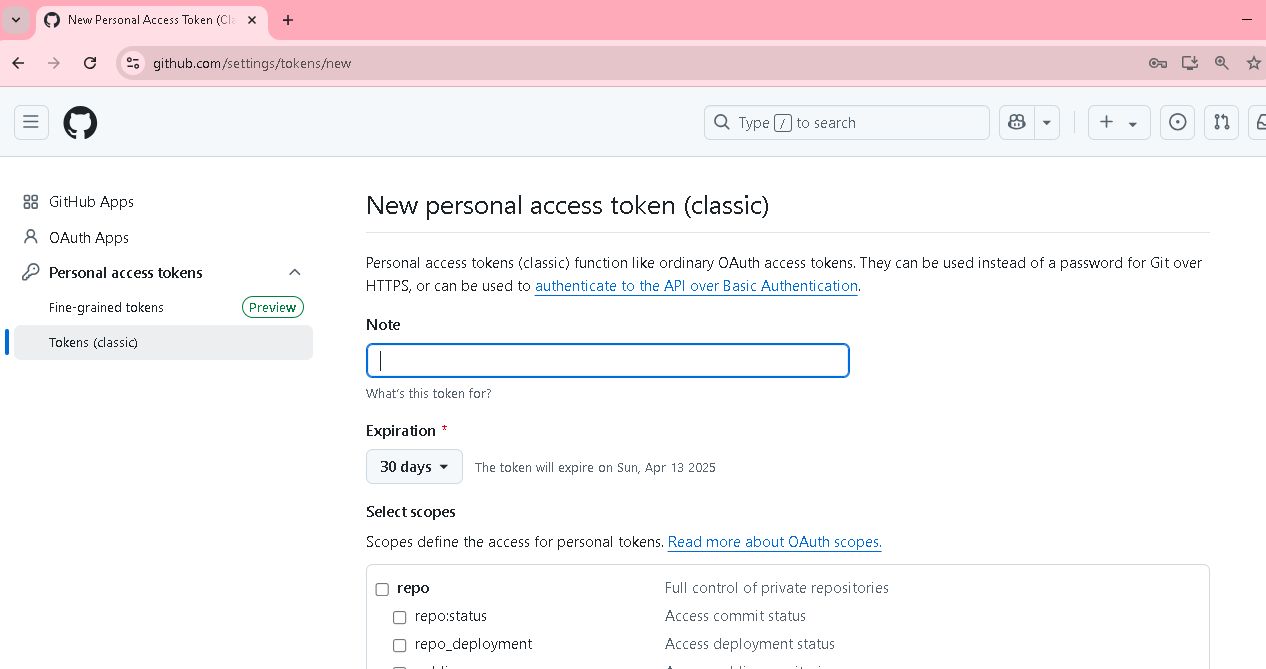
Click developer setting

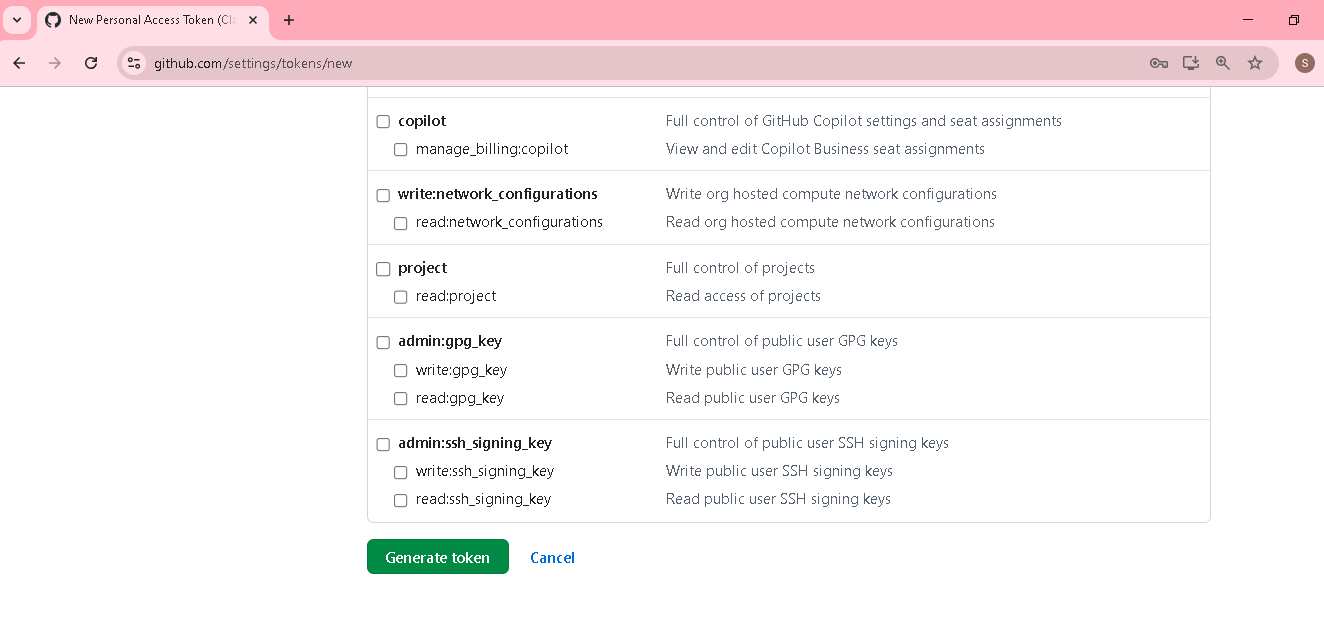


Personal Access Token ->Tokens(classic)->generate new token



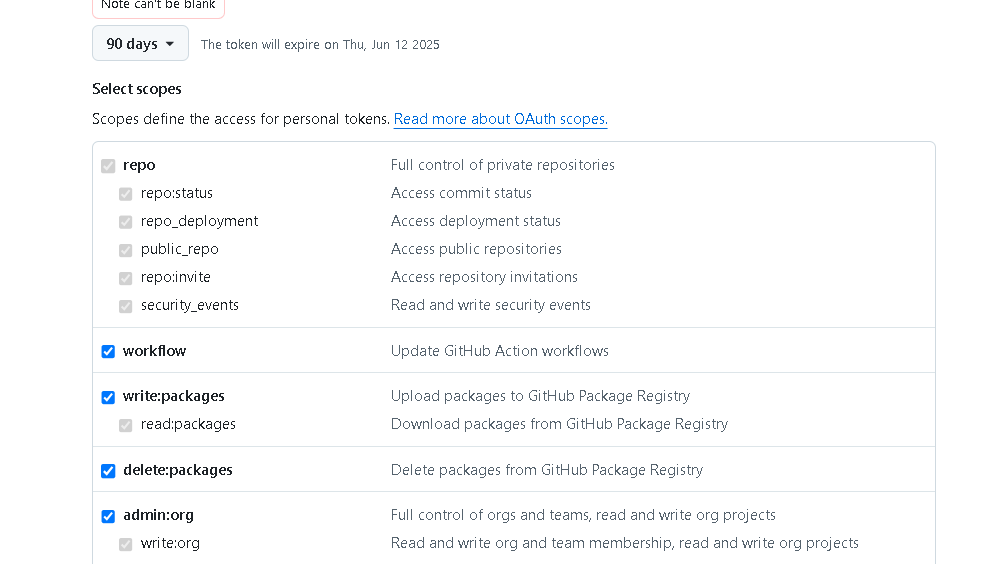


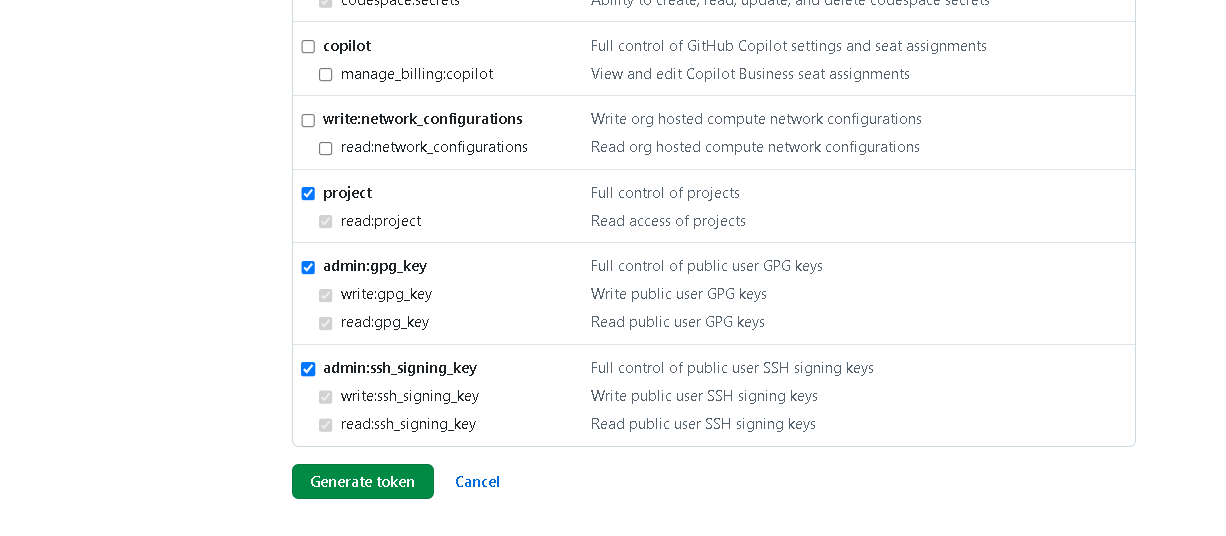




Provide any notes & token expiry timeline

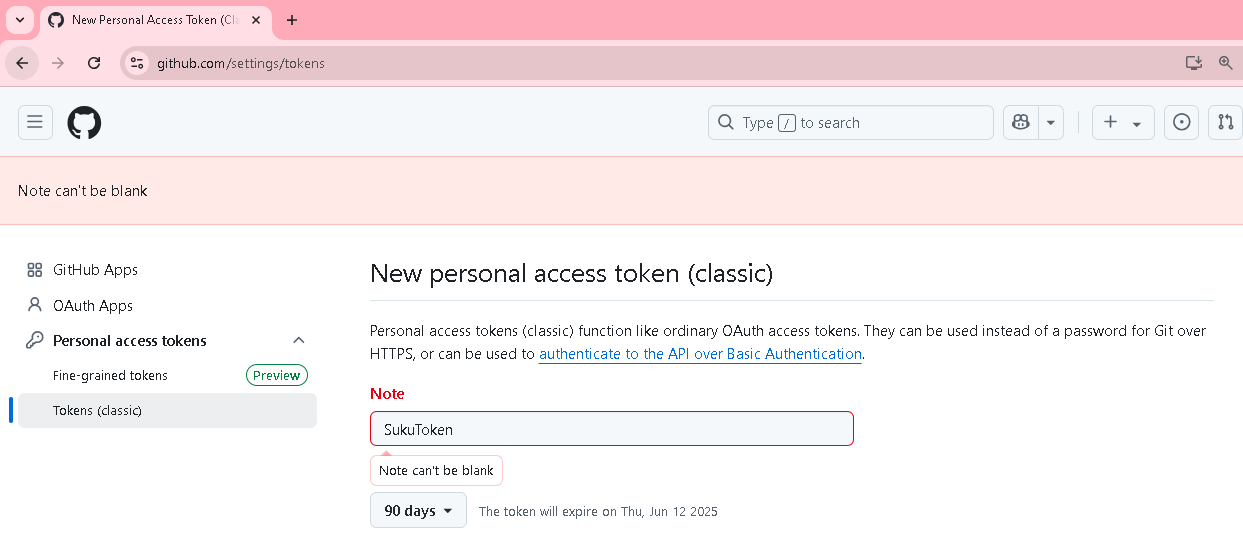
Select all checkbox

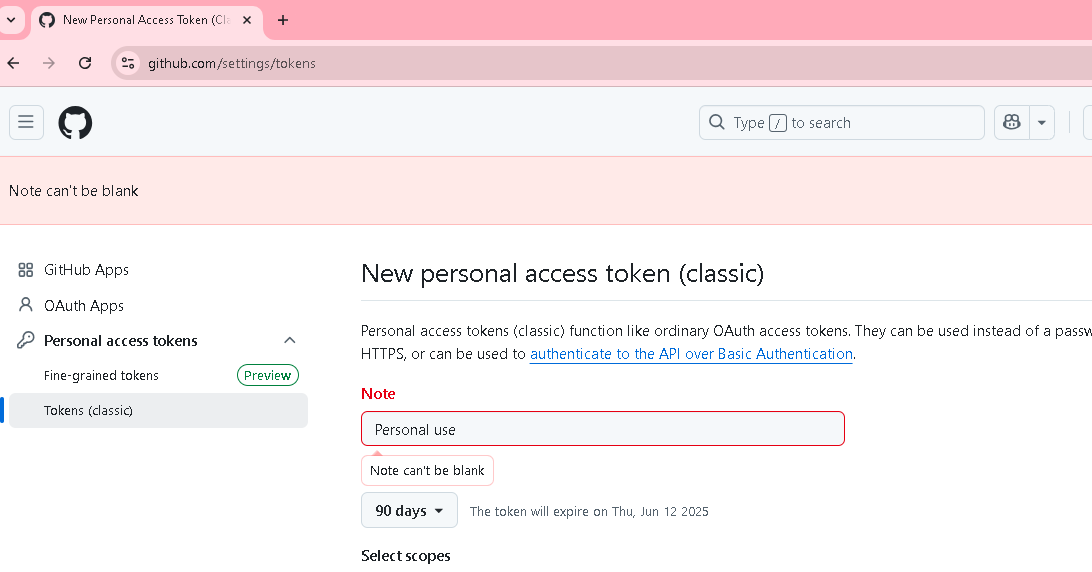




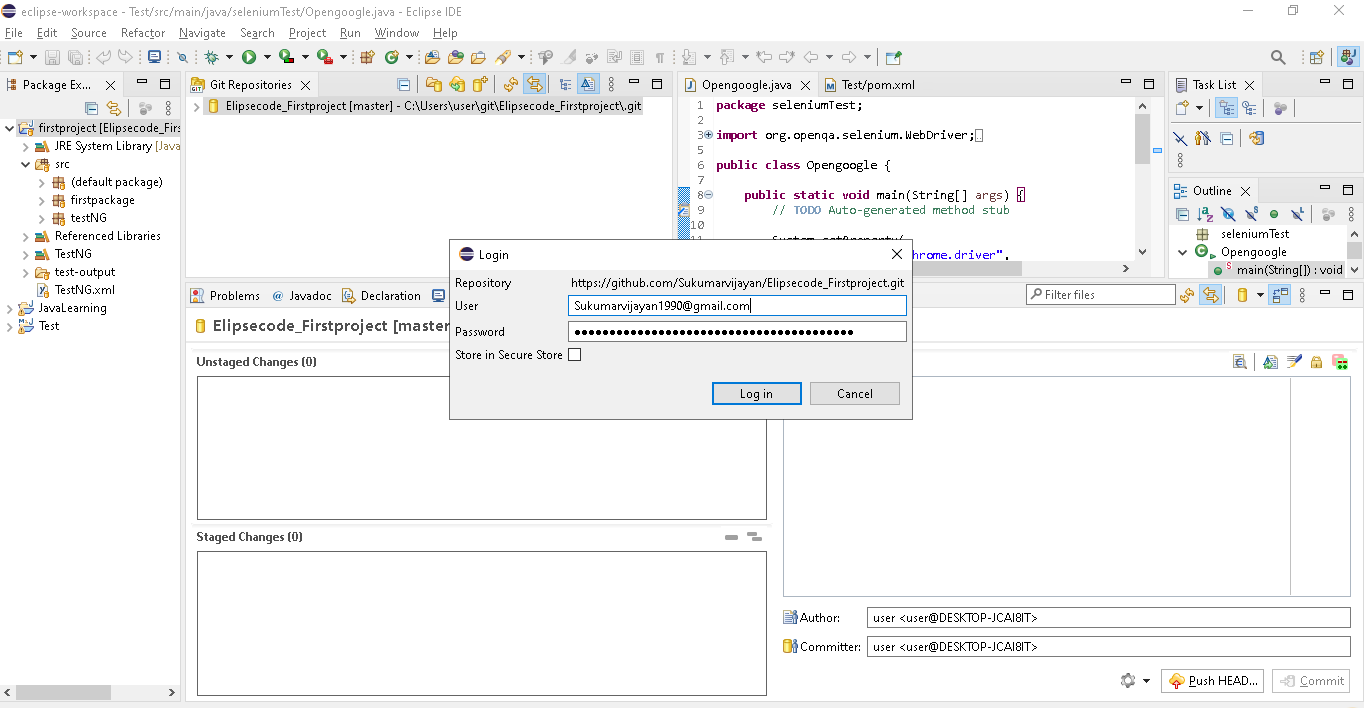


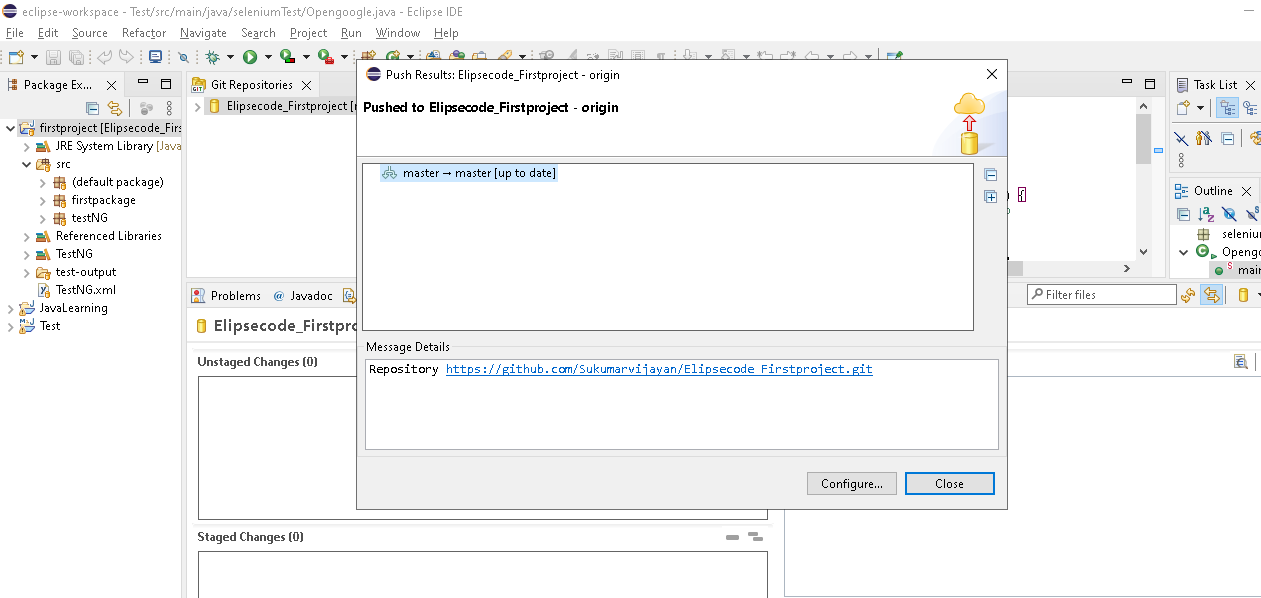
Token-> ghp\_n6donjMtueS3phgYPPMdzJHzrRU4kl4UFify





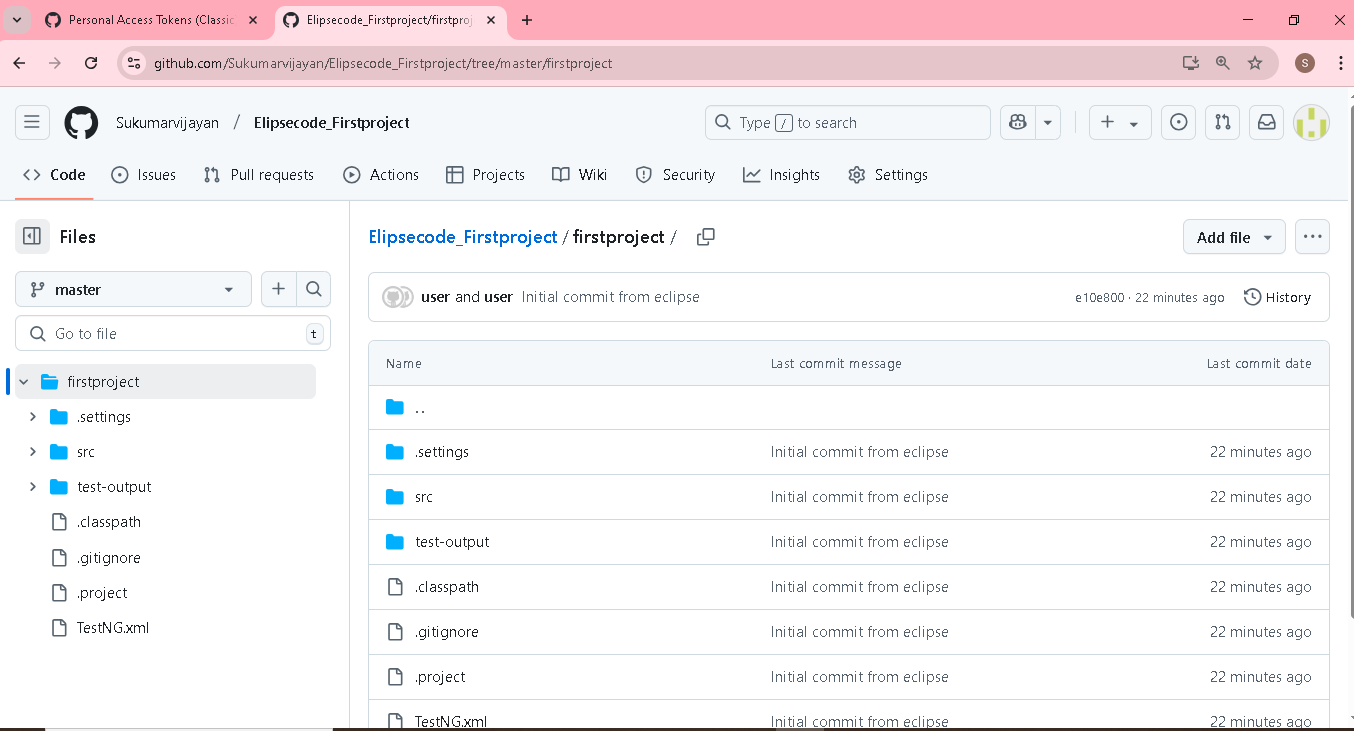
Go to elipse & provide access token



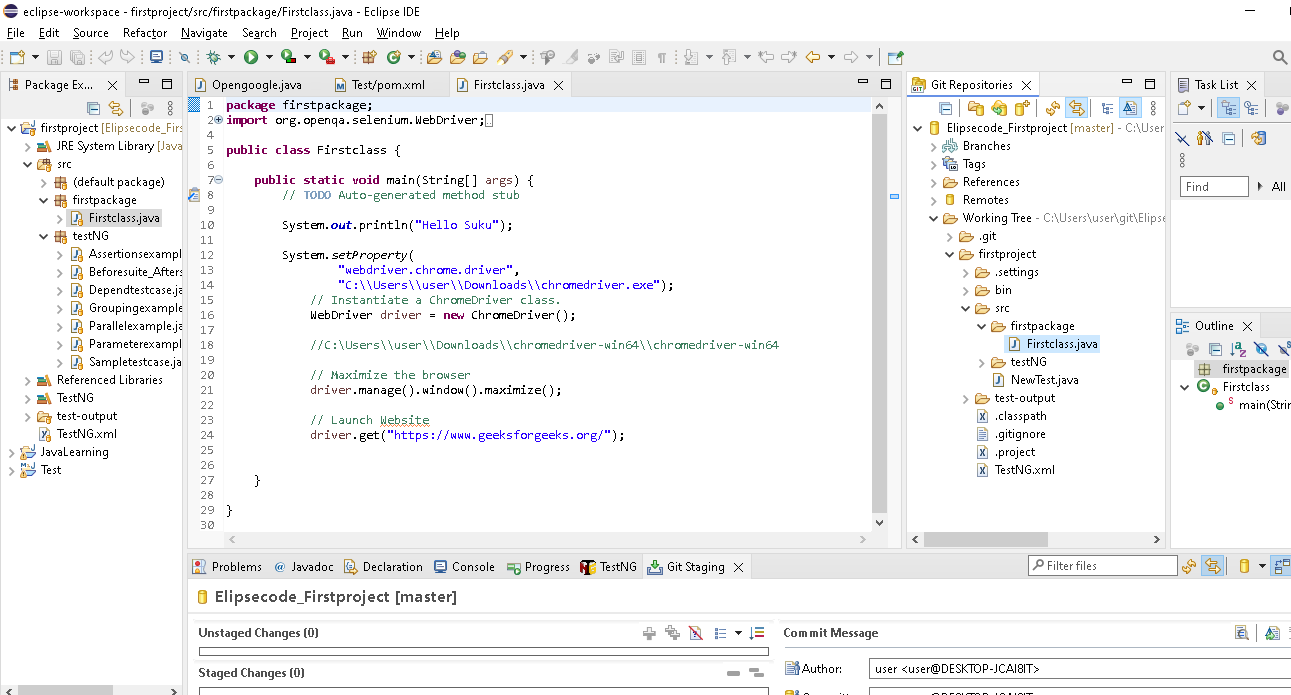


Click Configure ->Save & push

Go to github repository, check the code

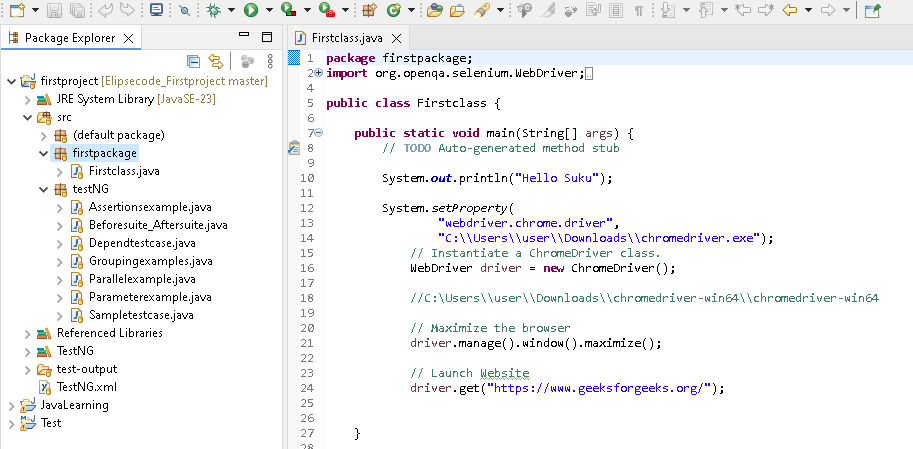




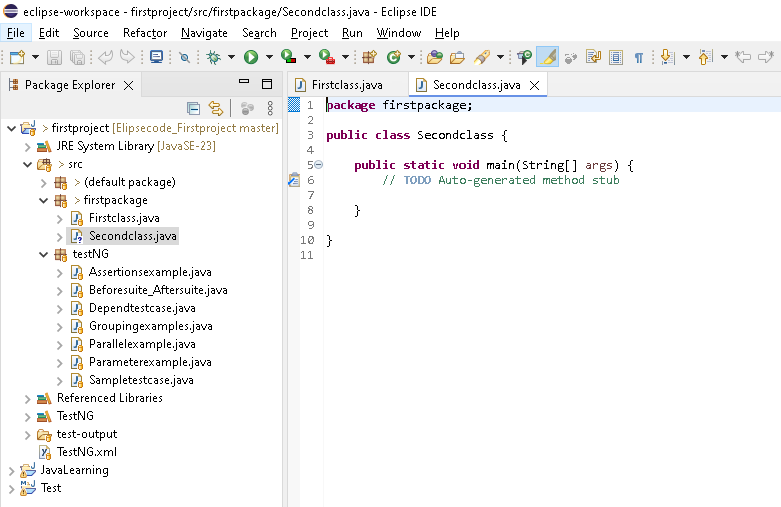


Whenever doing the changes in ellipse, by end of code completion or day, we need to push the code to GITHUB

Eg Now add new class in ellipse & push again

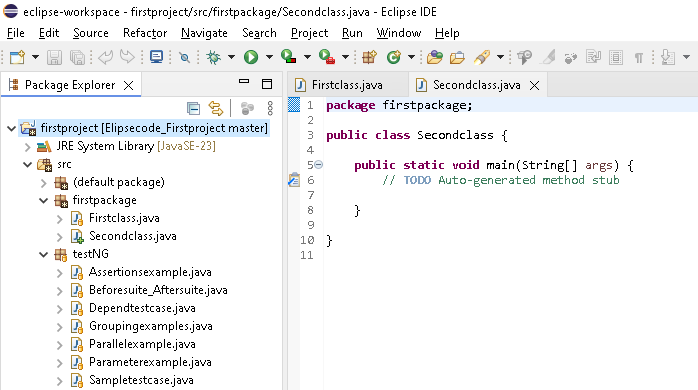


Add second class under First project & push to github



Same operation repeat

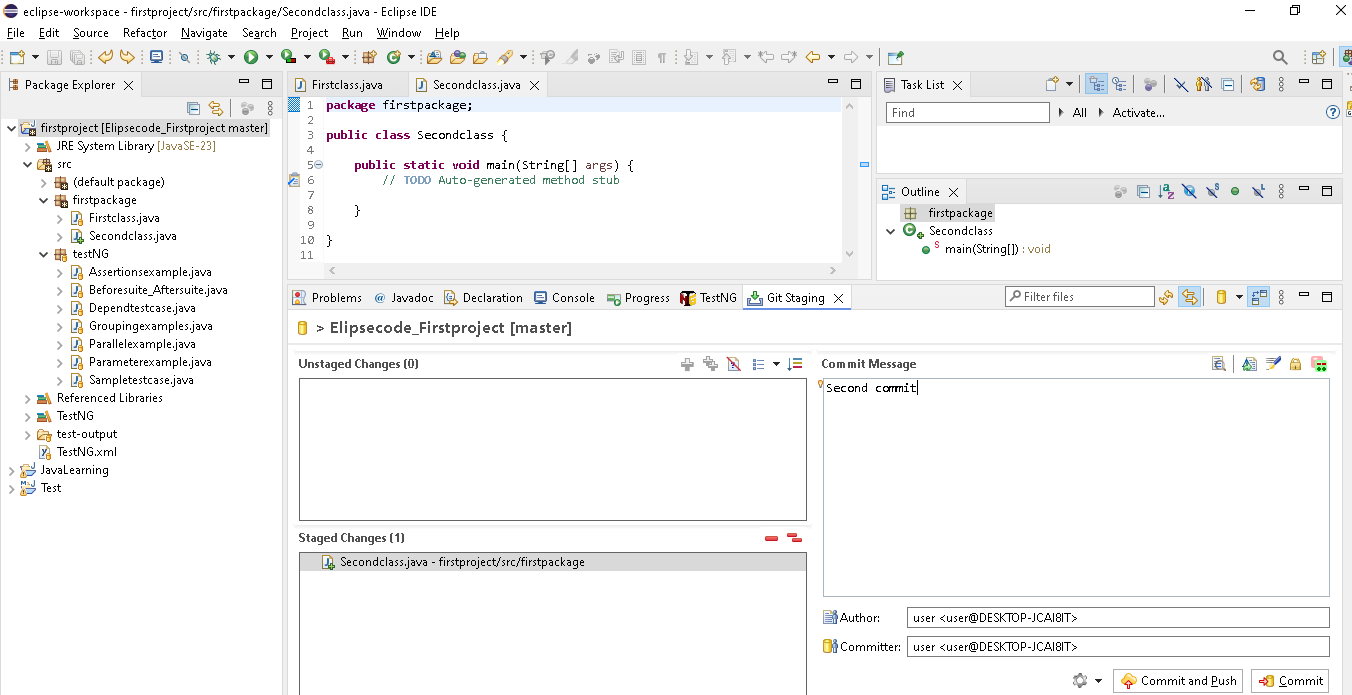
Right click the project(First project in elipse)->Team->Add to Index

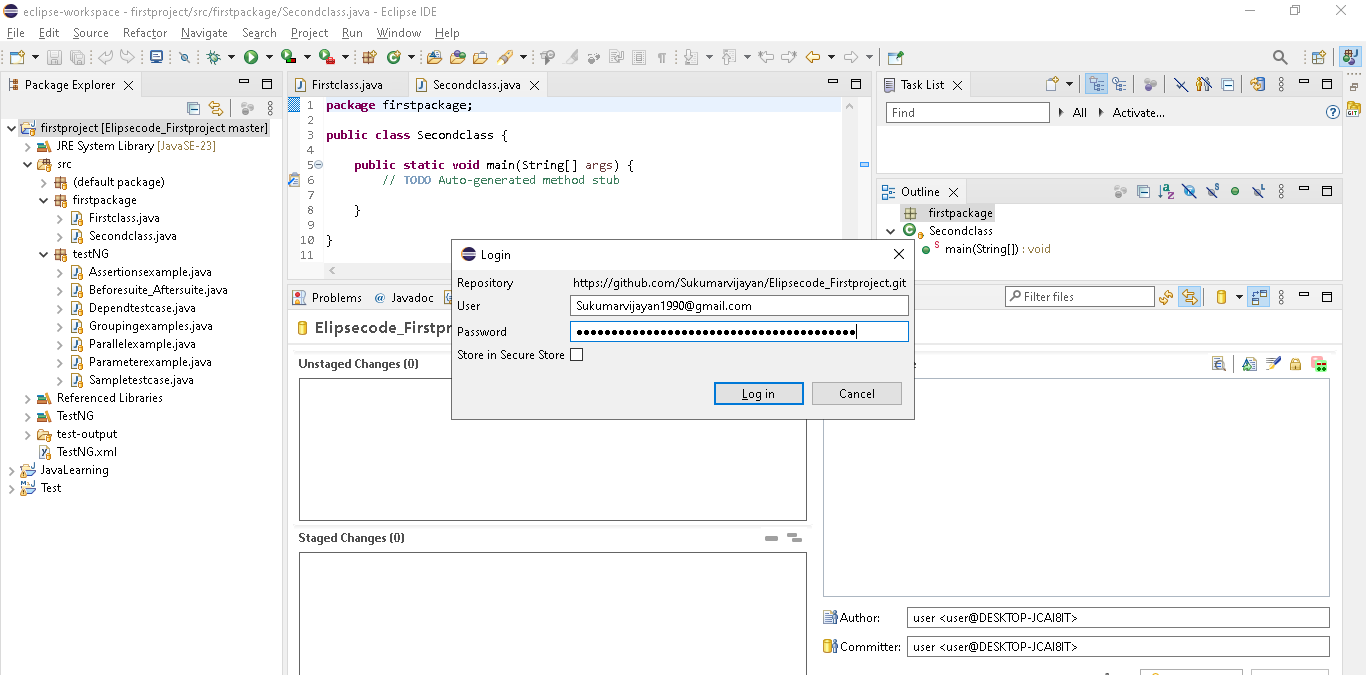


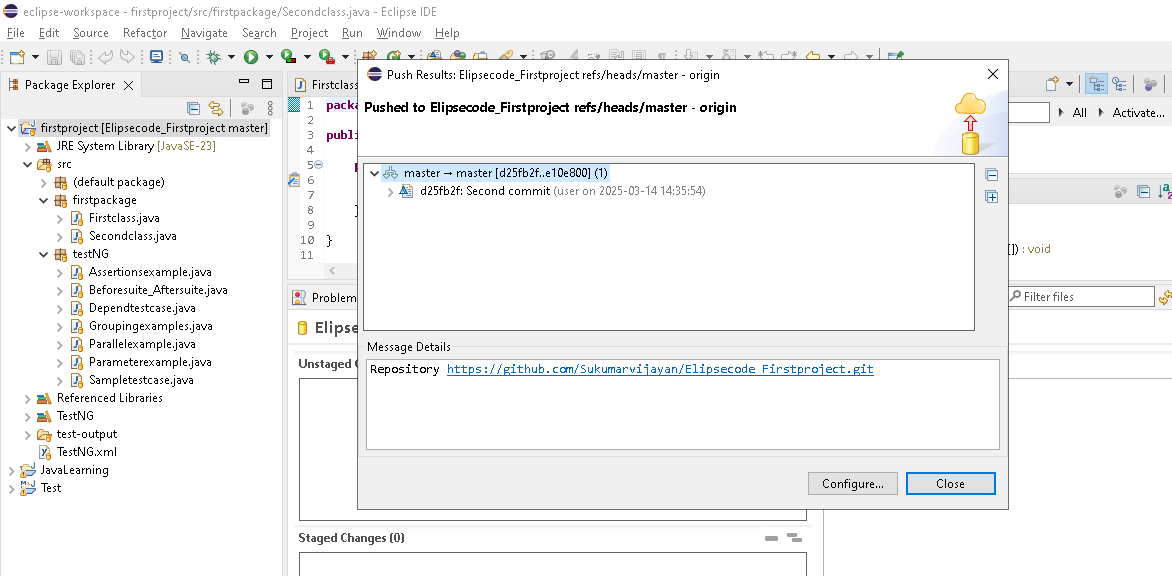
Right click the project(First project in elipse)->Team->Commit

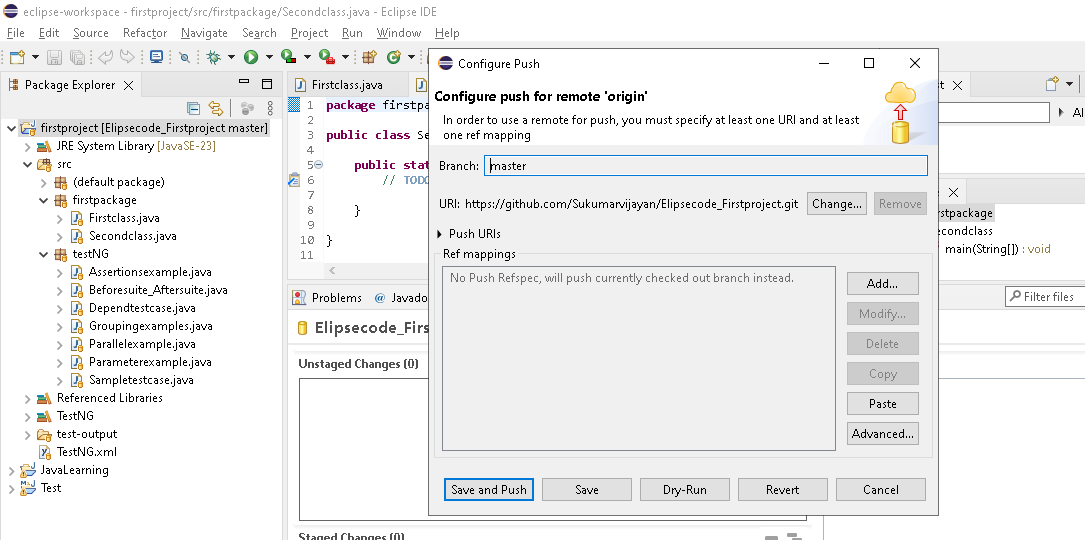
Provide the commit message

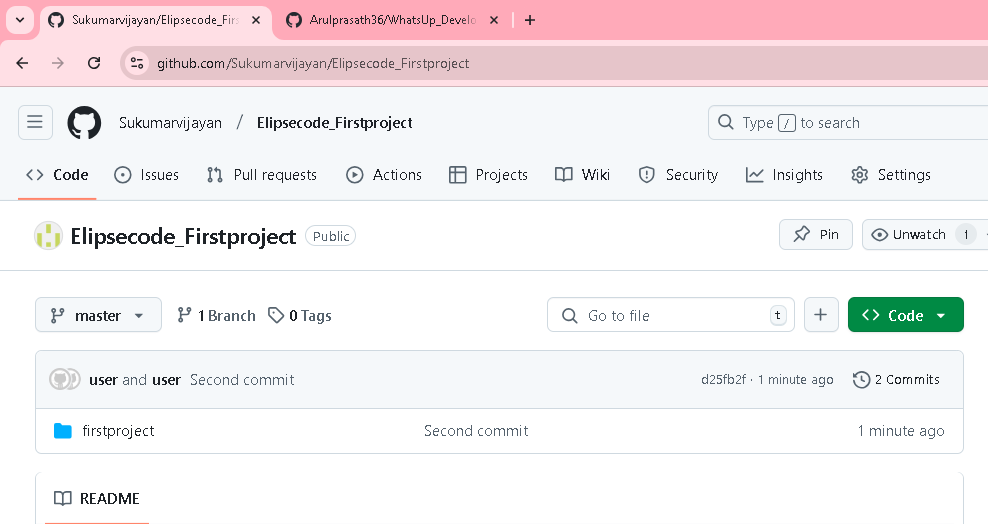
Click “Commit & push”

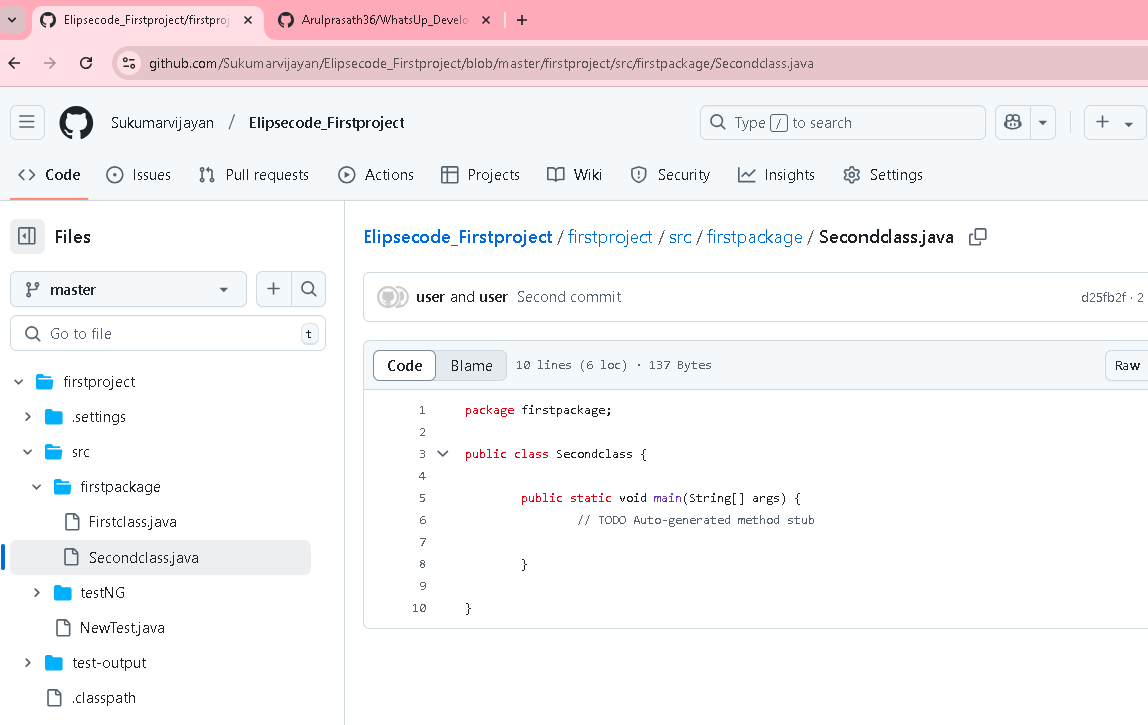






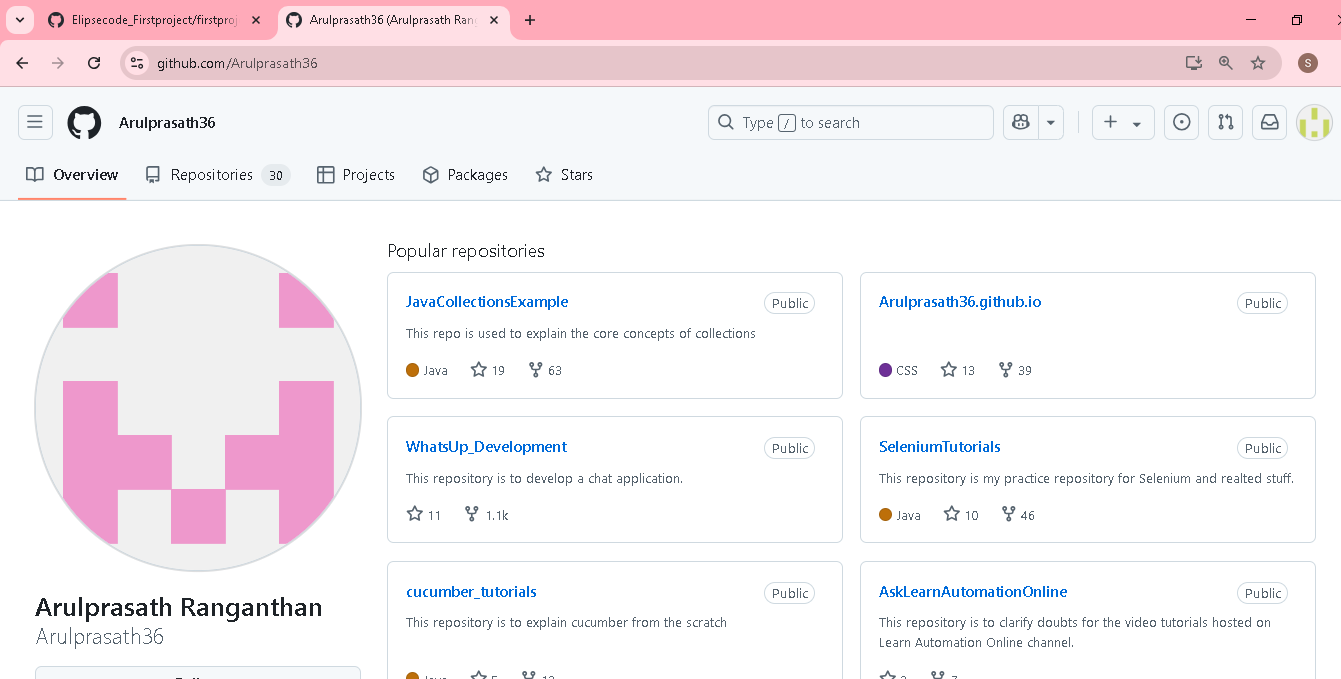


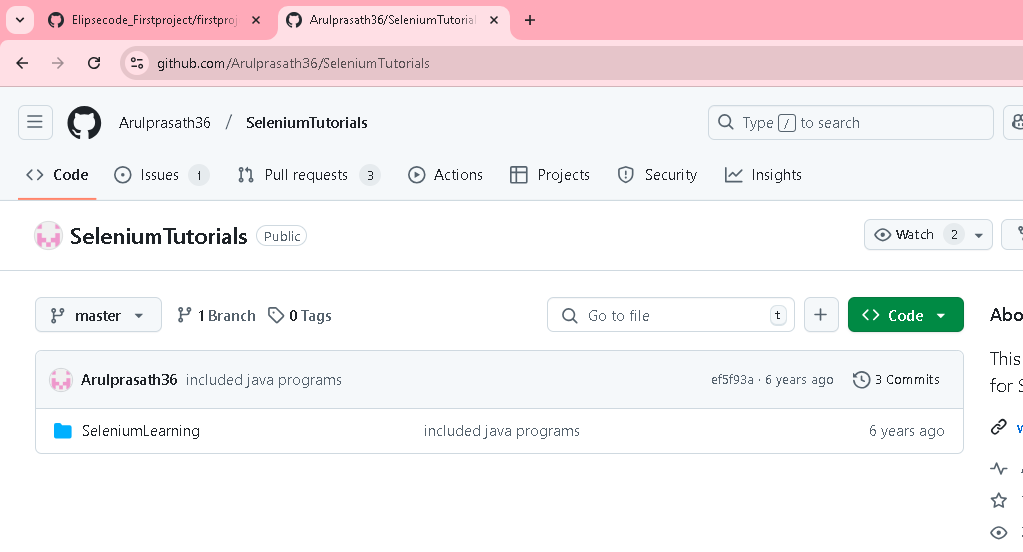




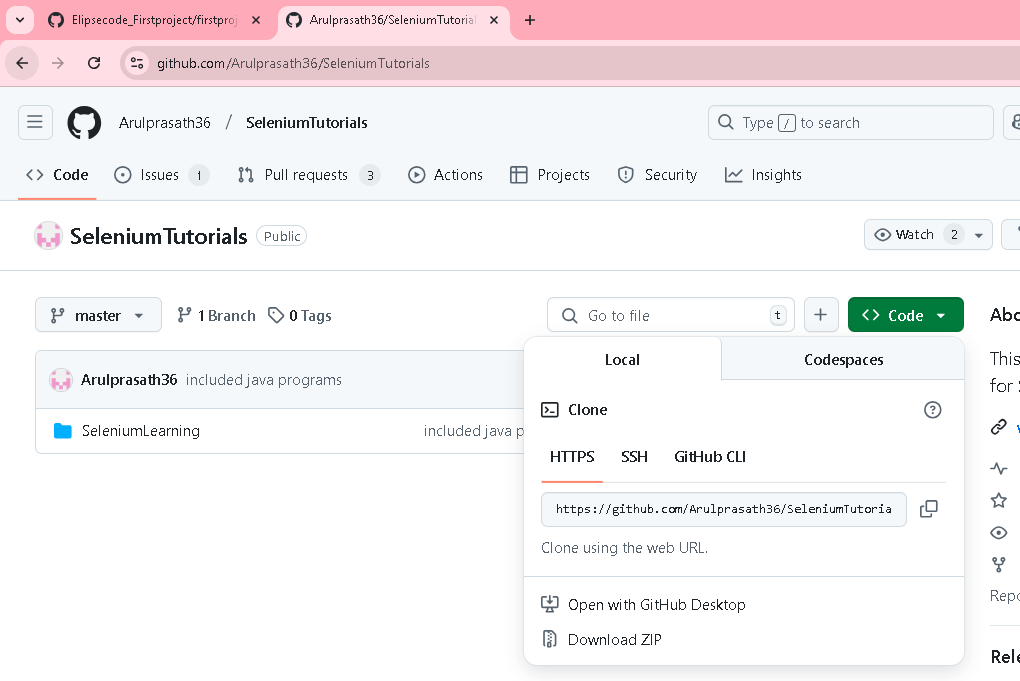
How to clone a code from GitHUB to ellipse

Go to Arul git, copy the url for selenium tutorials & download in ur elipse



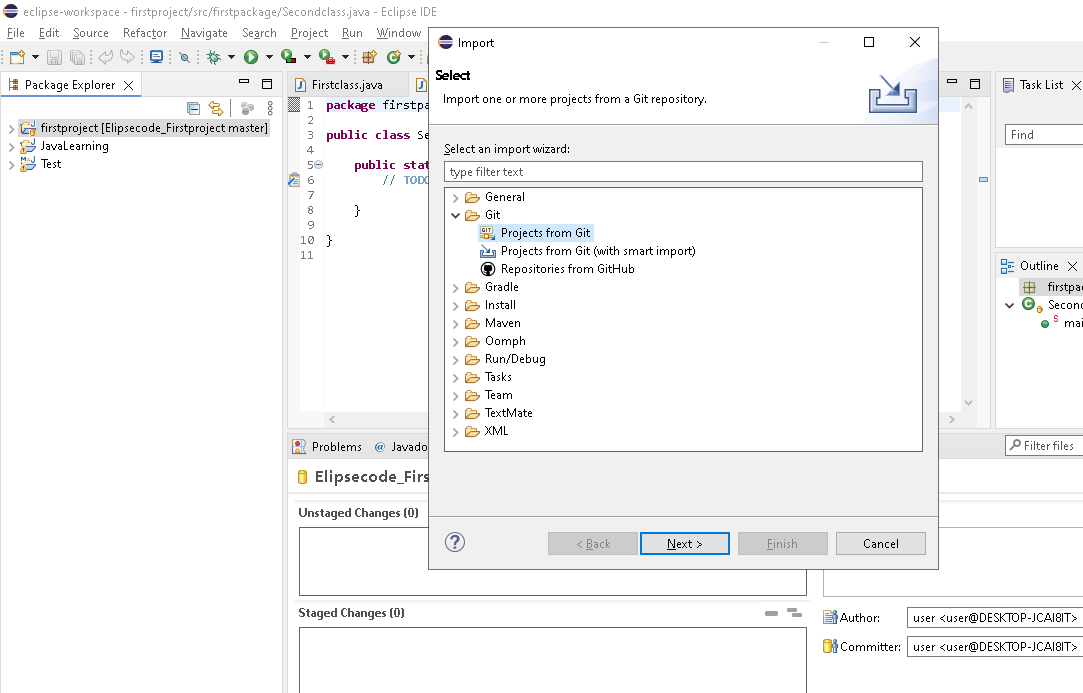


Click the code & copy the url

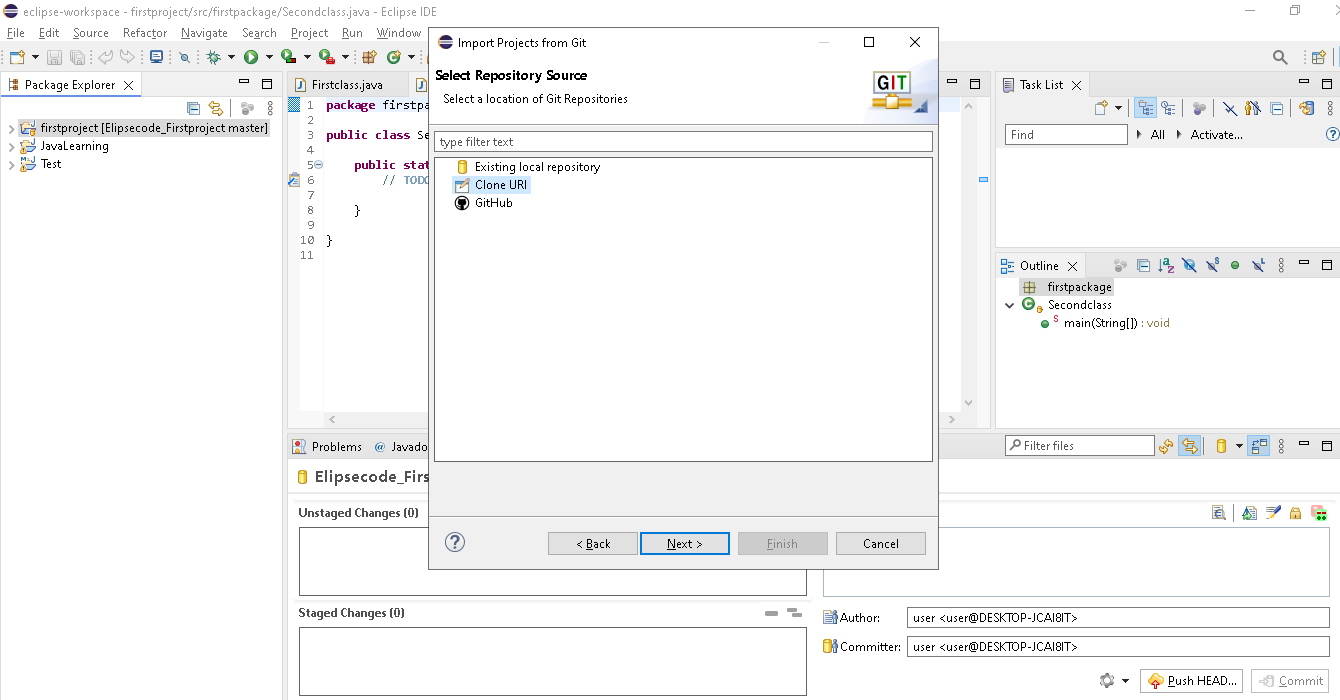


Go to ellipse

File->import ->Git->Project from git

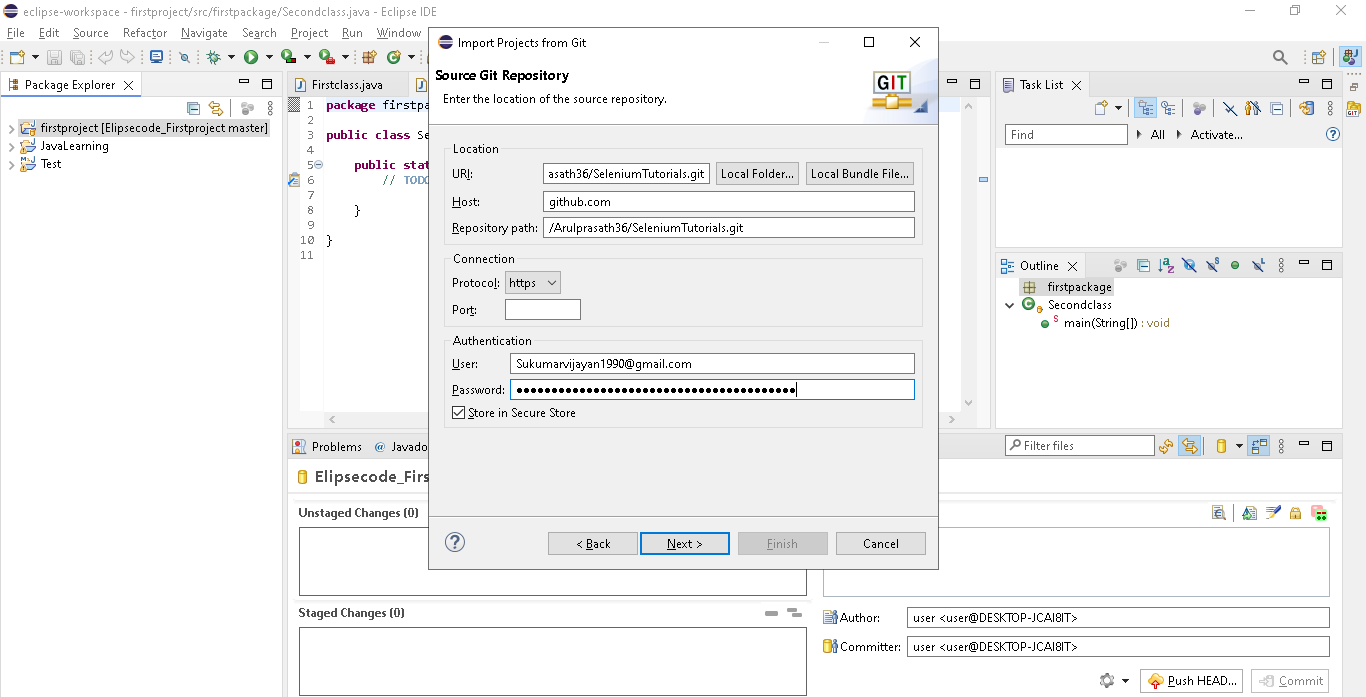


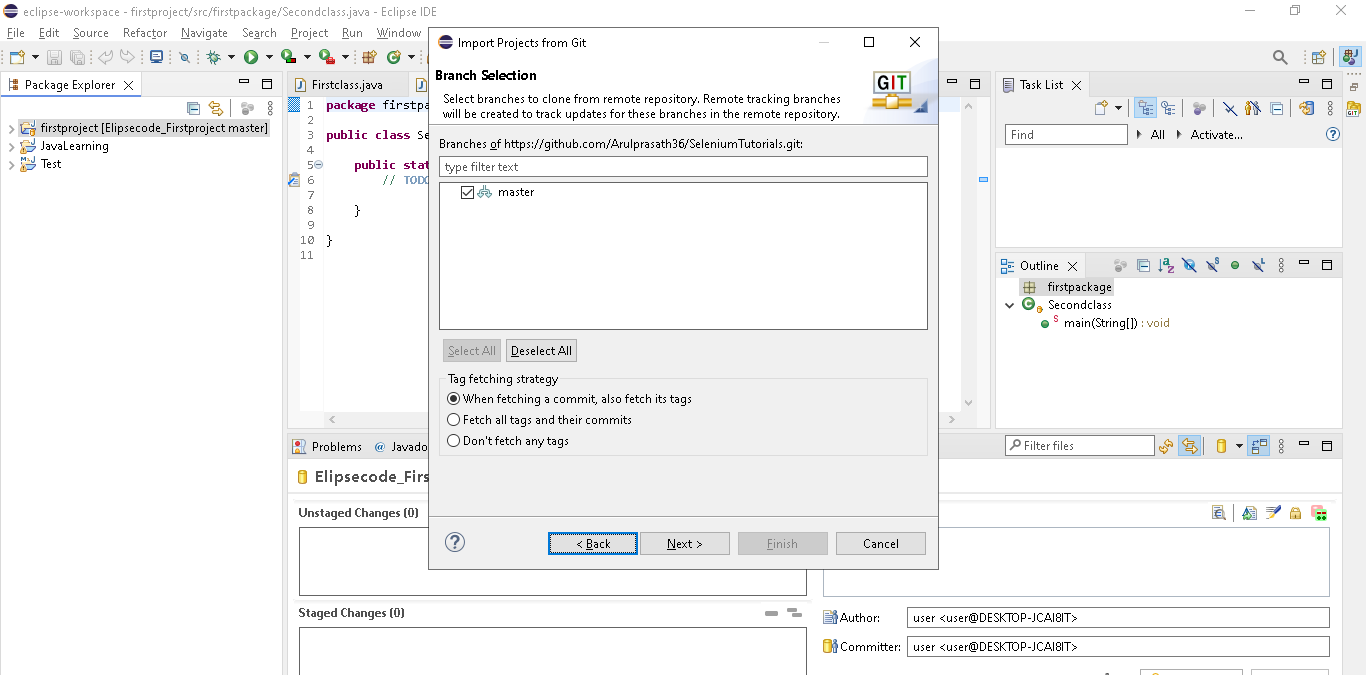
Select “Clone URL” & next

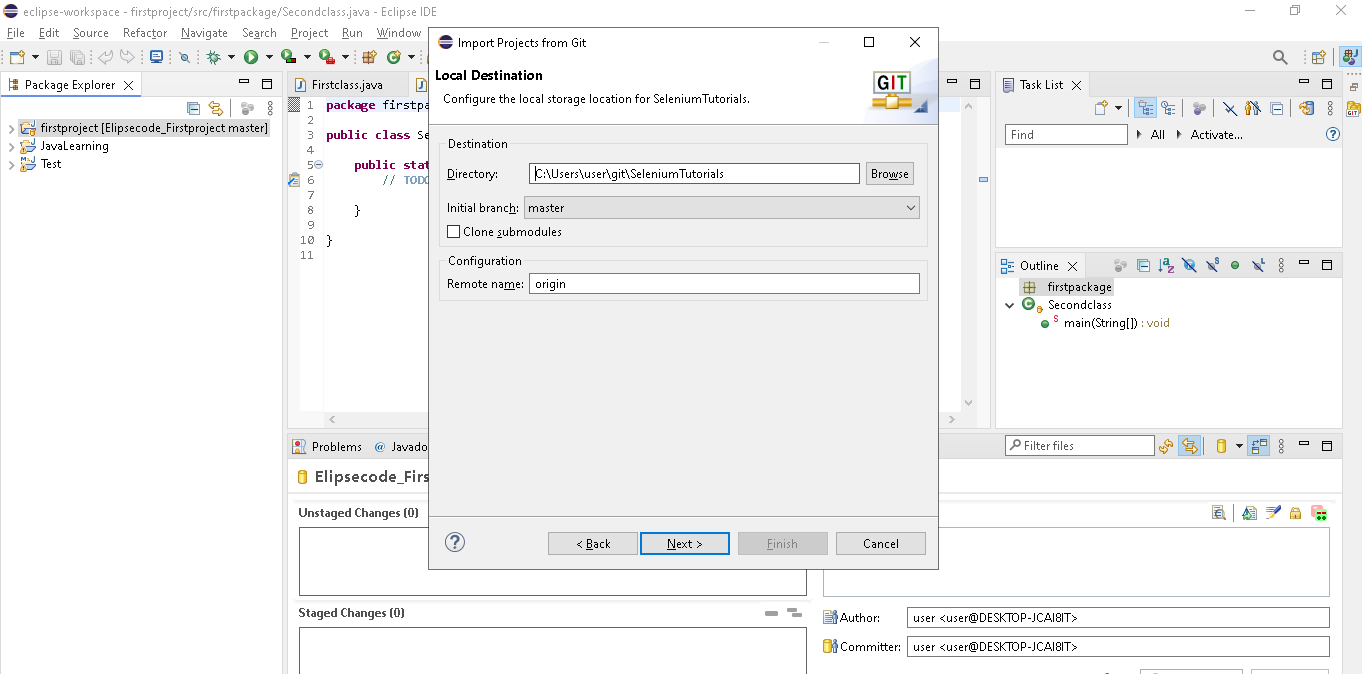


Fill the details

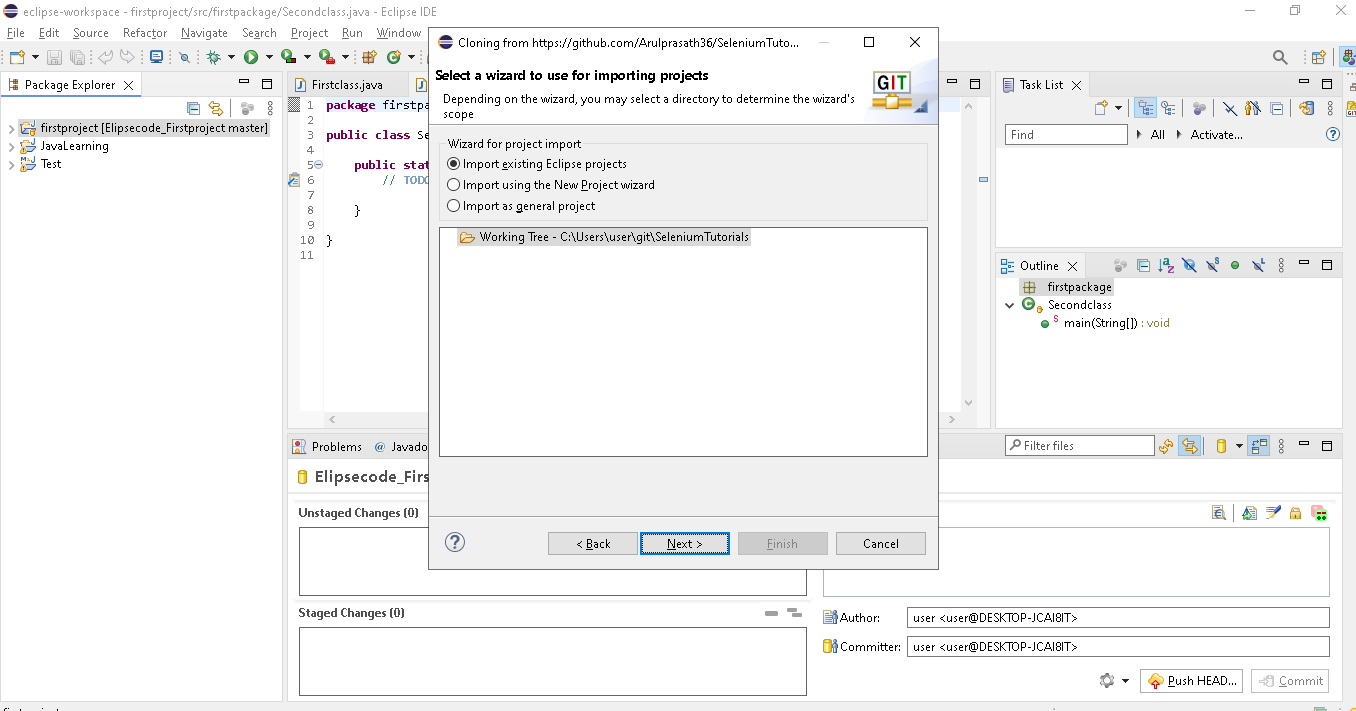
<https://github.com/Arulprasath36/SeleniumTutorials.git>

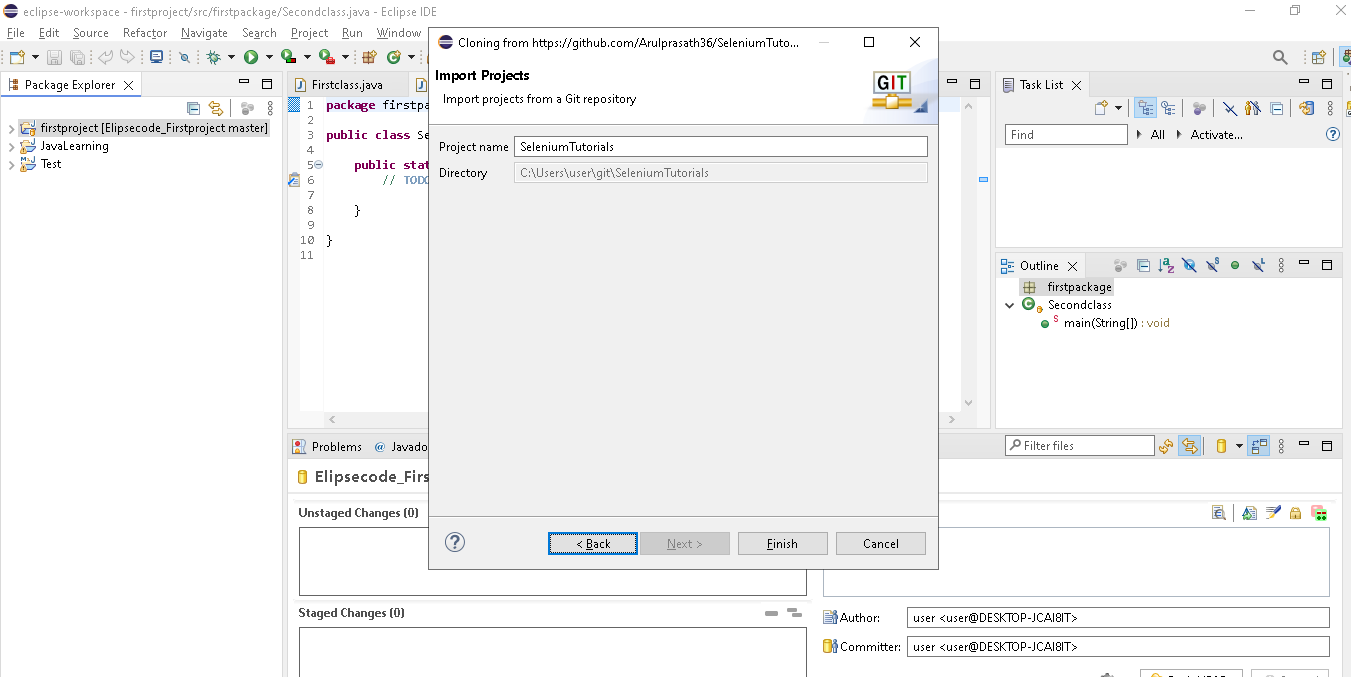




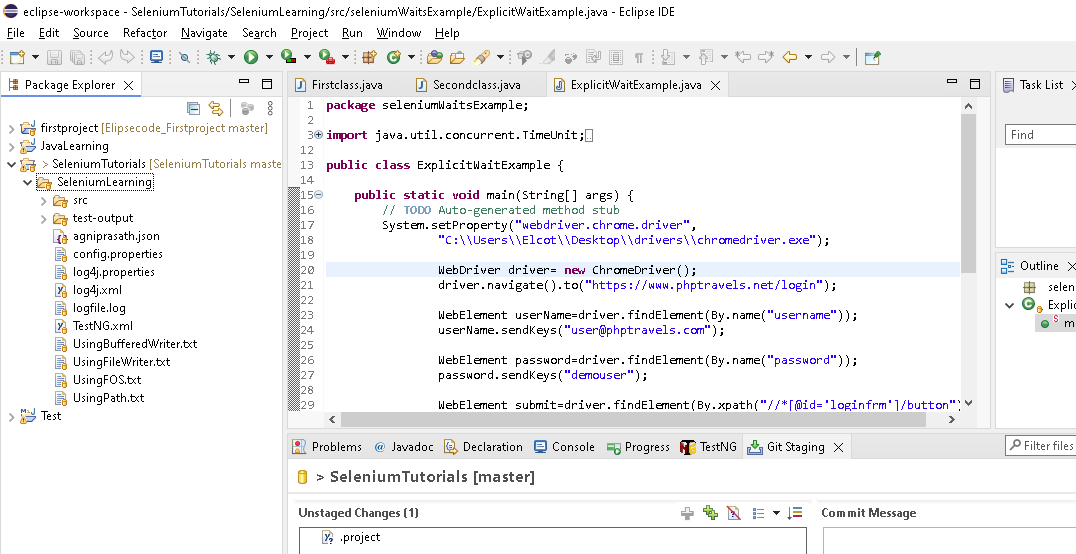


Select “Import as general project “ & click next





Project imported successfully



Delete the selenium & loaded the Javacollection

Able to execute the code successfully

