Connecting ECLIPSE and GitHub

© Manish Jaya Devadiga
manishjayadevadiga@gmail.com

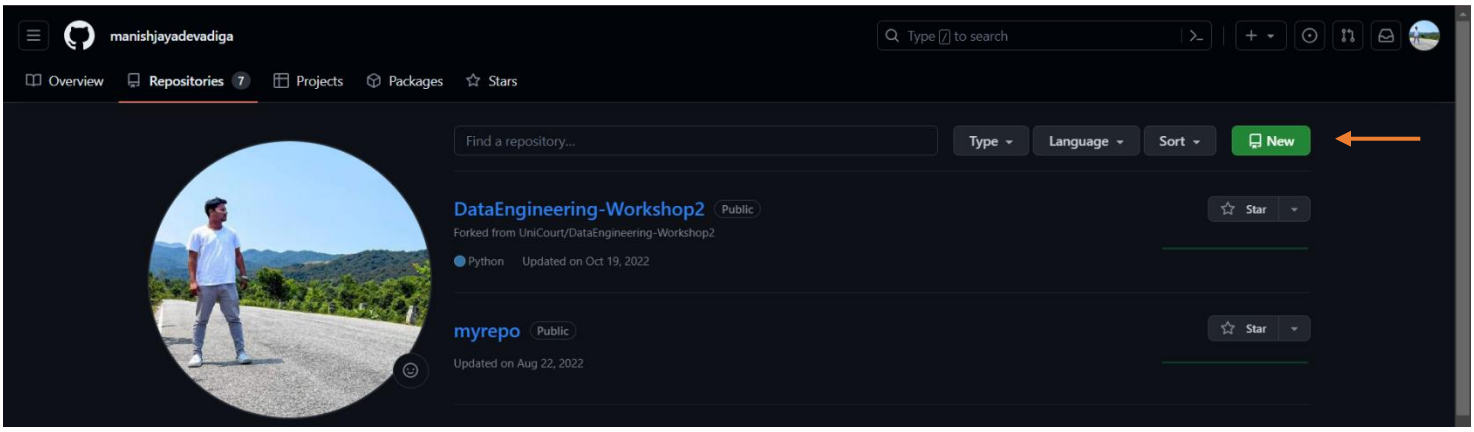
Note: Here we will see how to connect “Eclipse” and “GitHub” and upload java programs from eclipse.

A written connection pdf is also uploaded in the repository.

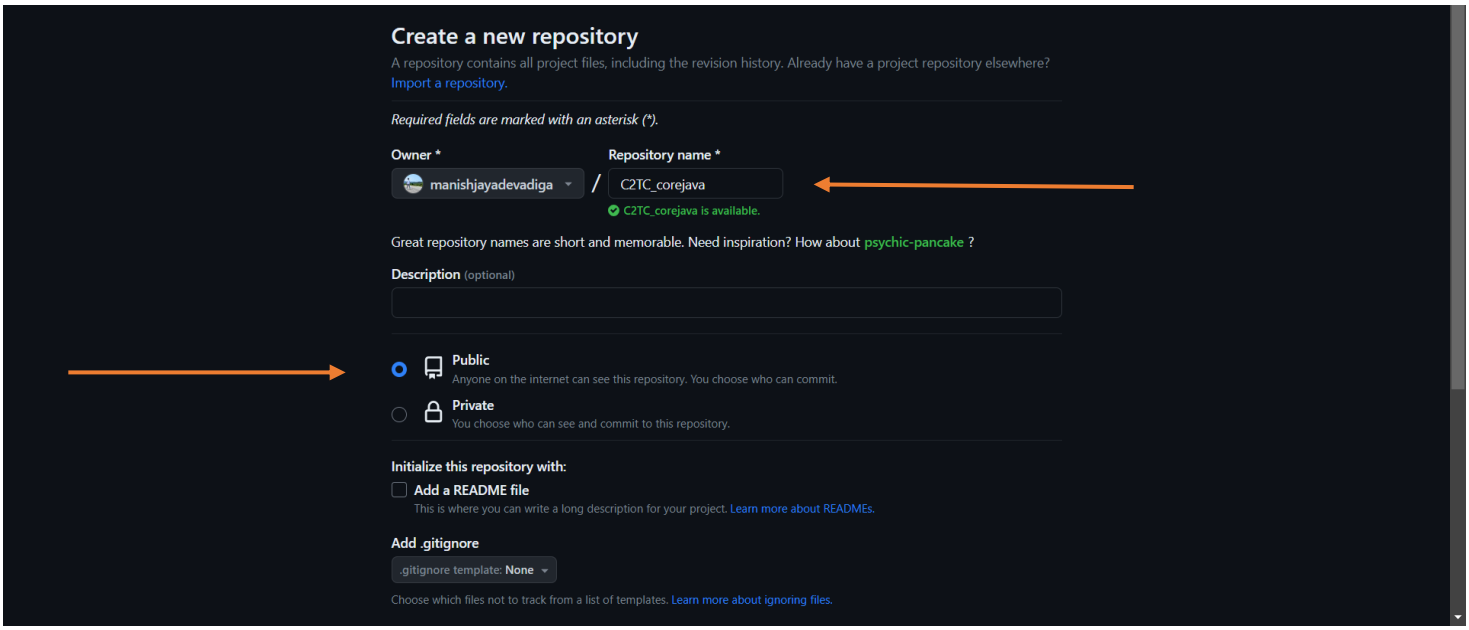
Now follow the below steps correctly for smooth and error free connection.

GitHub:

- Open GitHub and **create your account**.
- After account creation, click on “**My repository**”.
- Click “**New Repository**”.



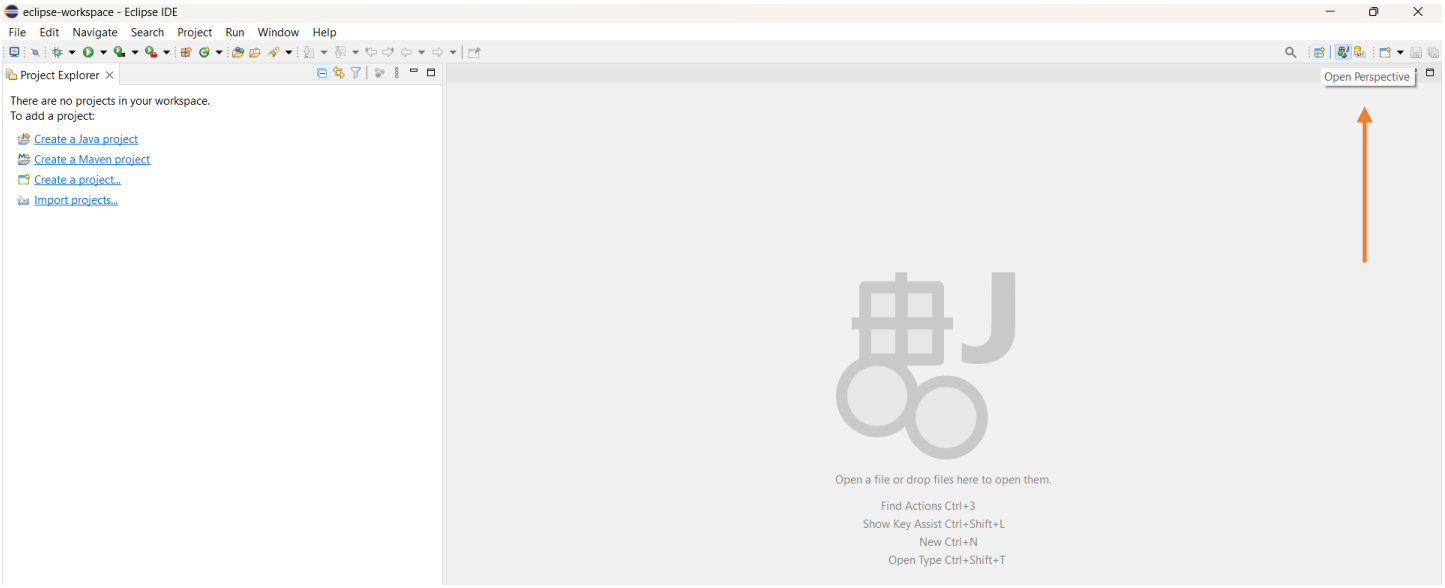
- Now give name of repository as “**C2TC_corejava**” and set it as **public**.



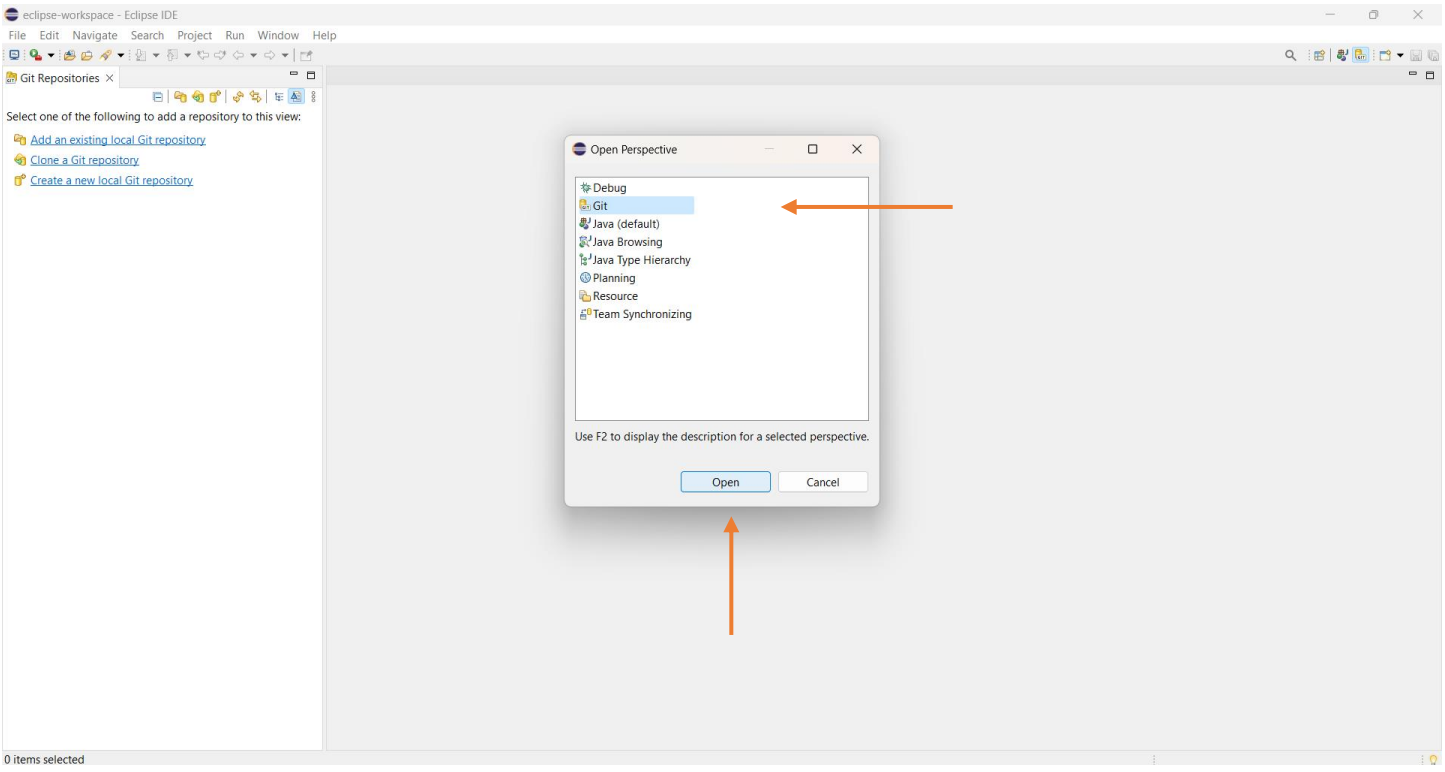
- Now click “**Create Repository**”.

Eclipse:

- Now open Eclipse.
- In the top right click “**Open Perspective**”

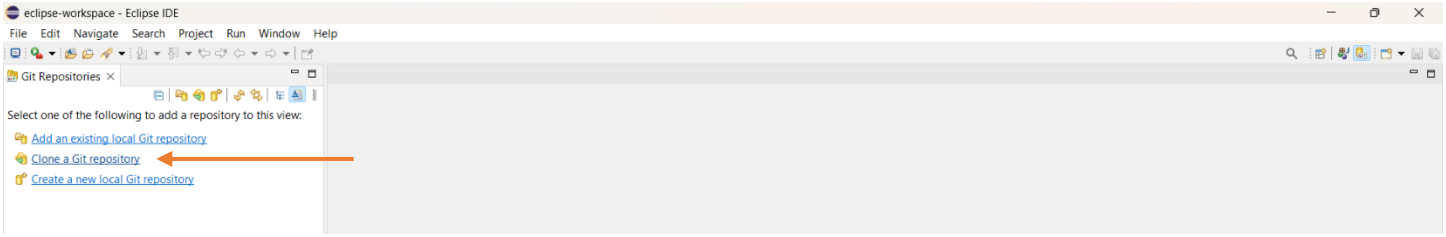


- Click on “**Git -> Open** “

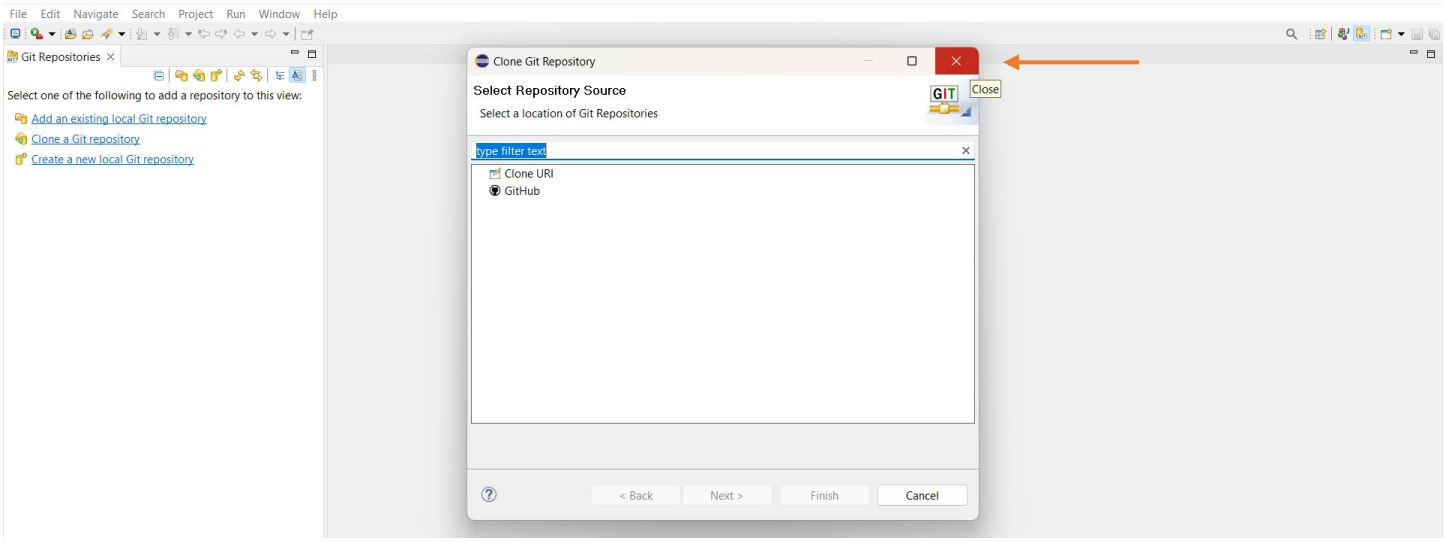


- Now click “**Windows -> Show View -> Git Repositories** “

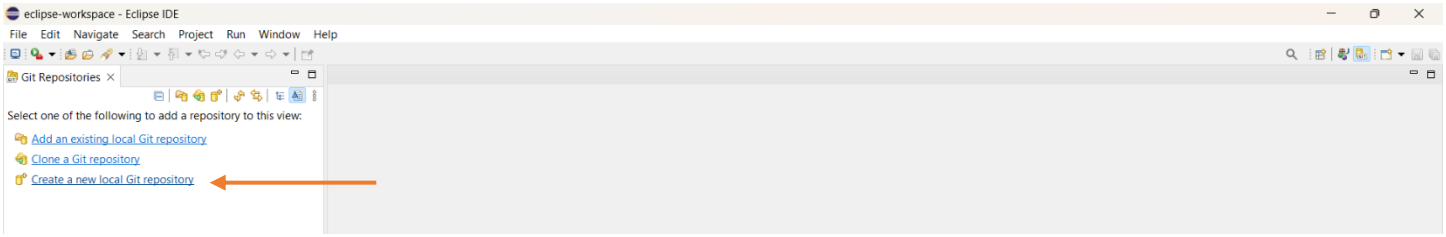
- Now in top-left corner click “**Clone a git repository**”



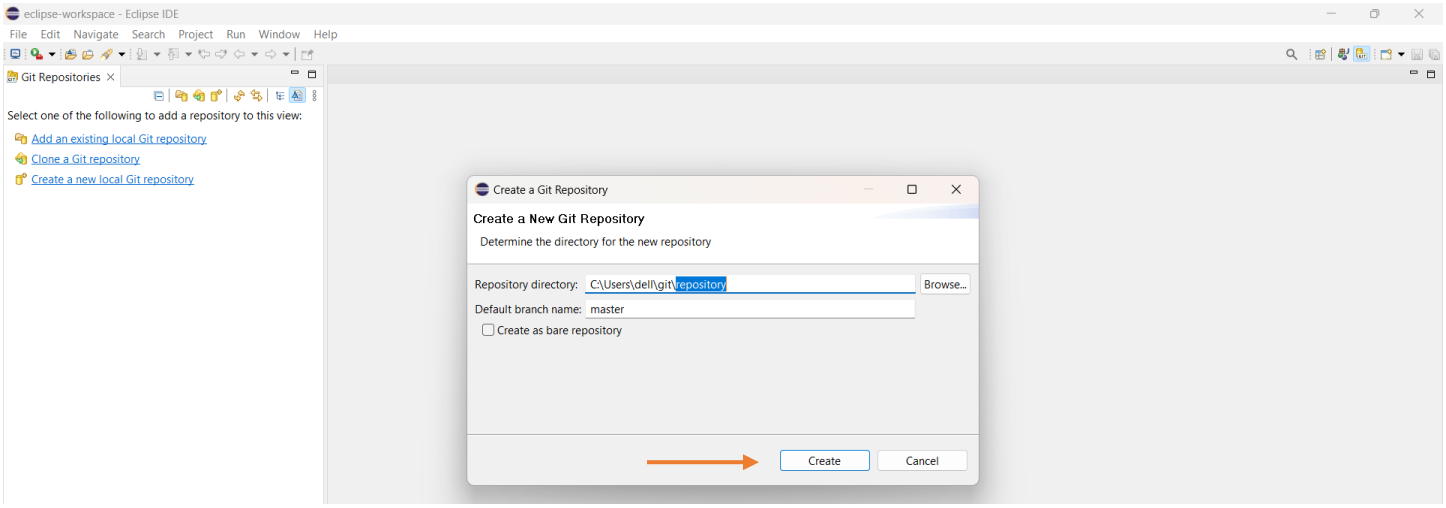
- **Close** the dialog box.



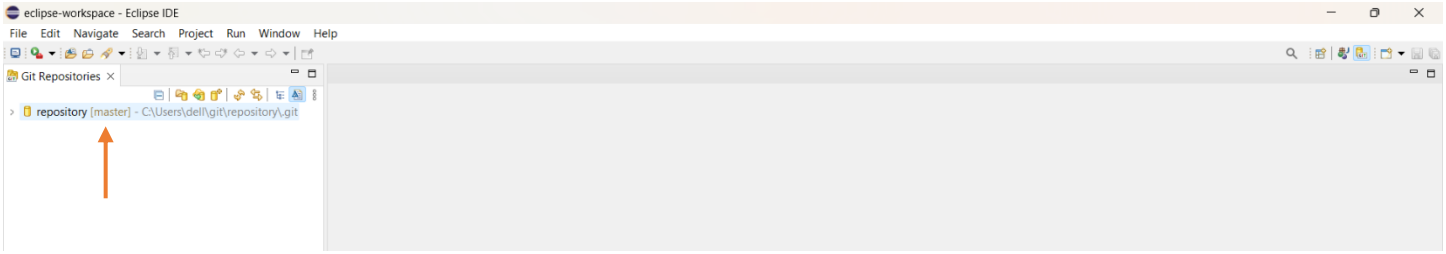
- Click on “**Create a new local Git Repository** “



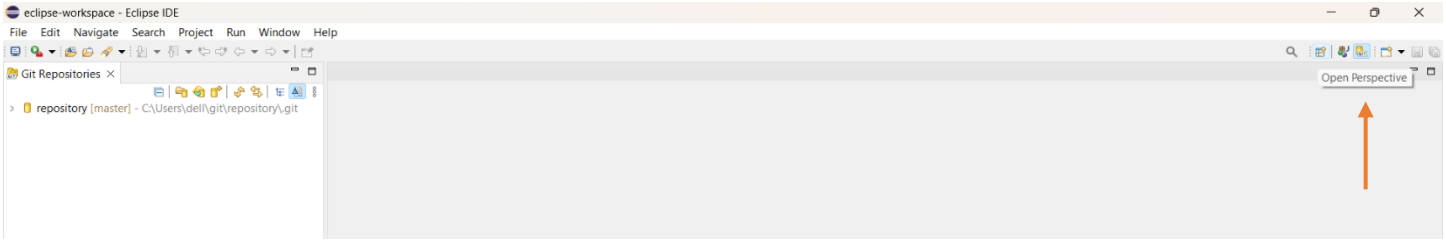
- Now leave everything as-it-is. Click “**Create**”.



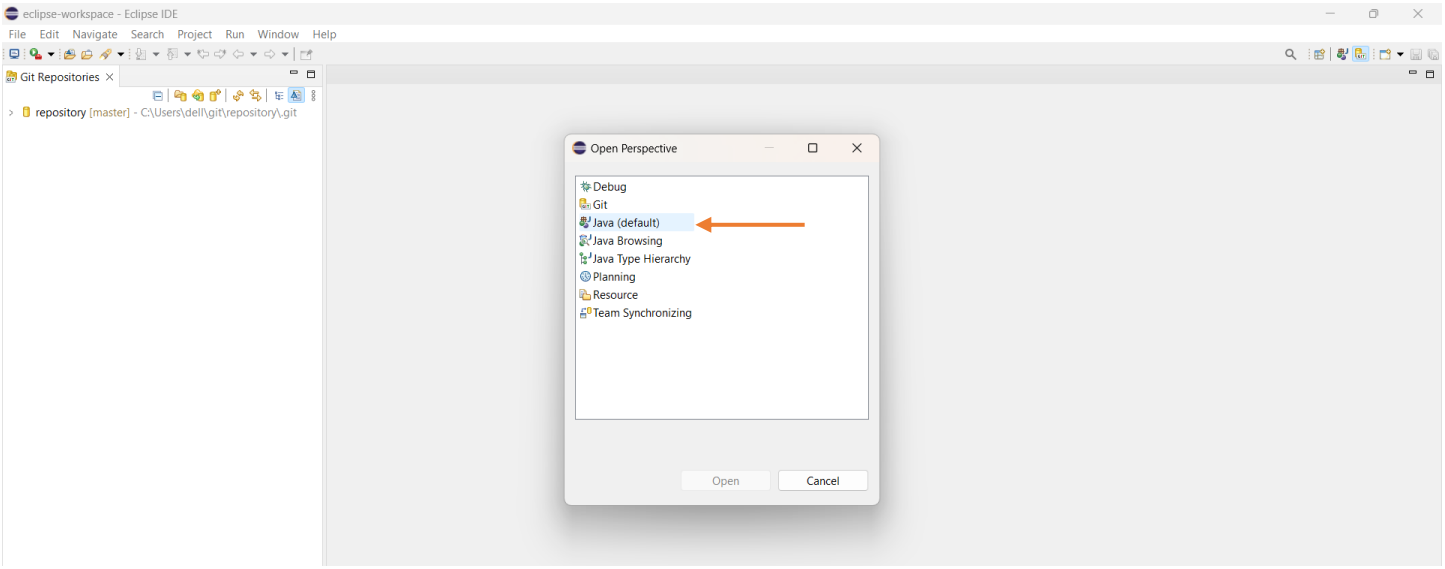
- Now in top left corner you can see “**repository[master]**”. Leave as it is.



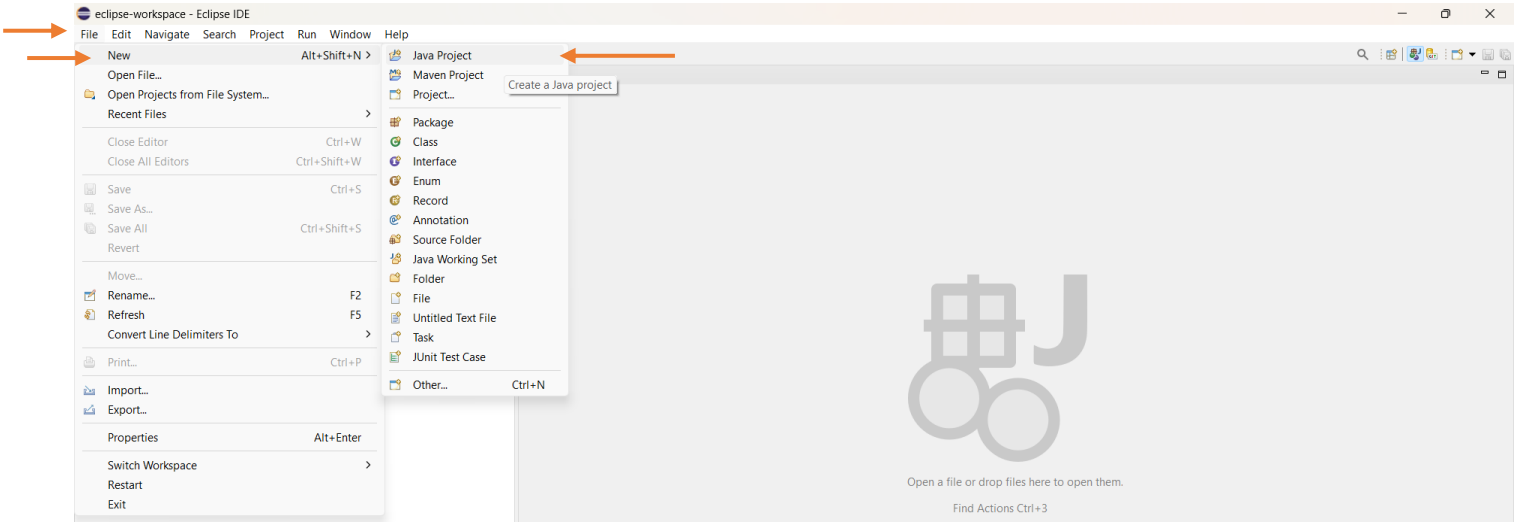
- Now click “**Open Perspective** “.



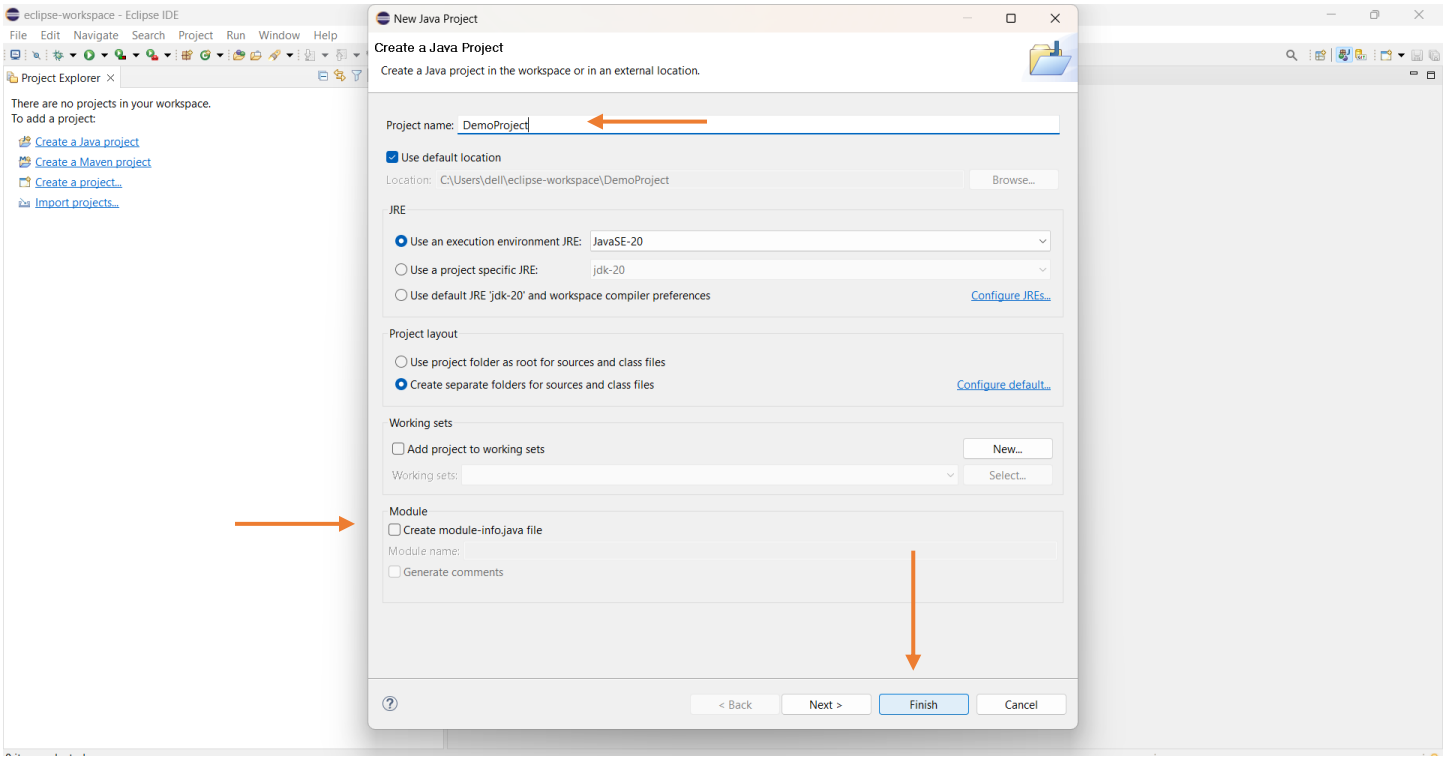
- Click “**Java(default)** “and “**Open**”.



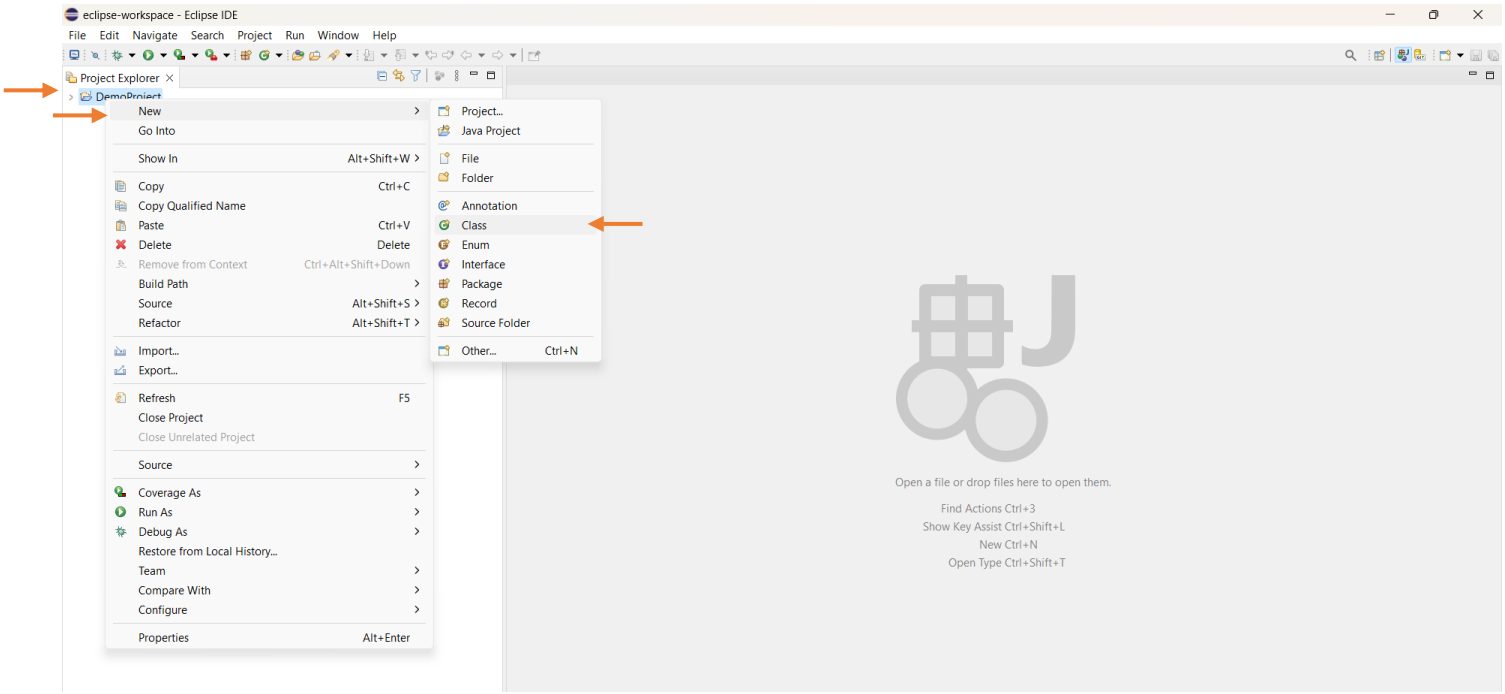
- Now we will **create Projects** to push to the GitHub Repository.
- Click on “**File -> New -> Java Project**”.



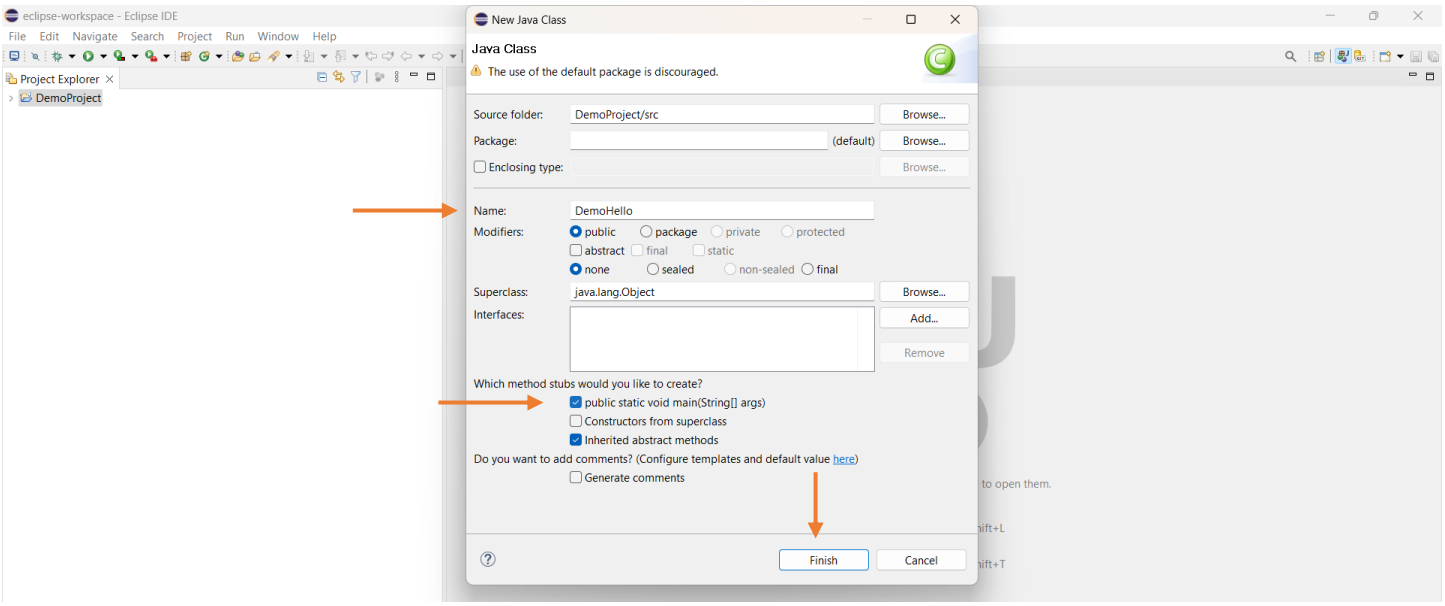
- Give the name of project as “**DemoProject** ”.
- Uncheck “**Module-info.java**”
- Click “**Finish**”.



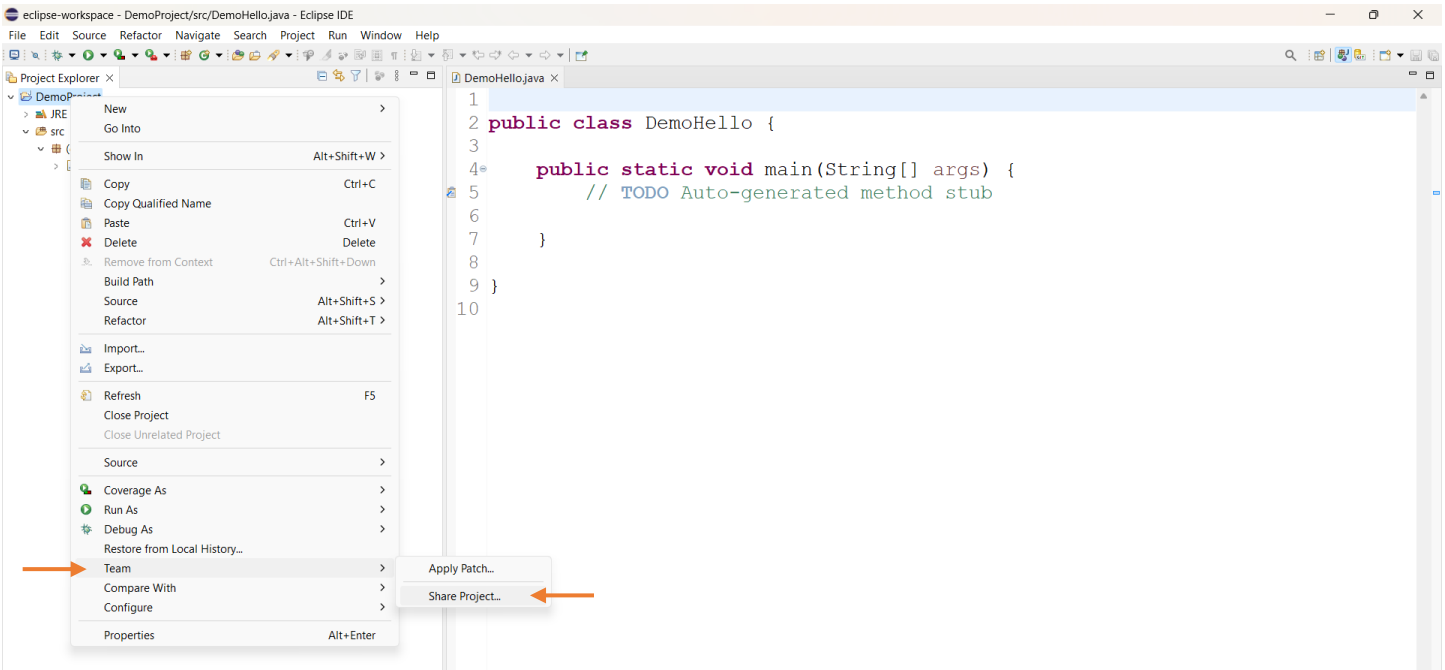
- Now let us create class inside this DemoProject.
- **Right-click** on "**DemoProject -> New -> Class** ".



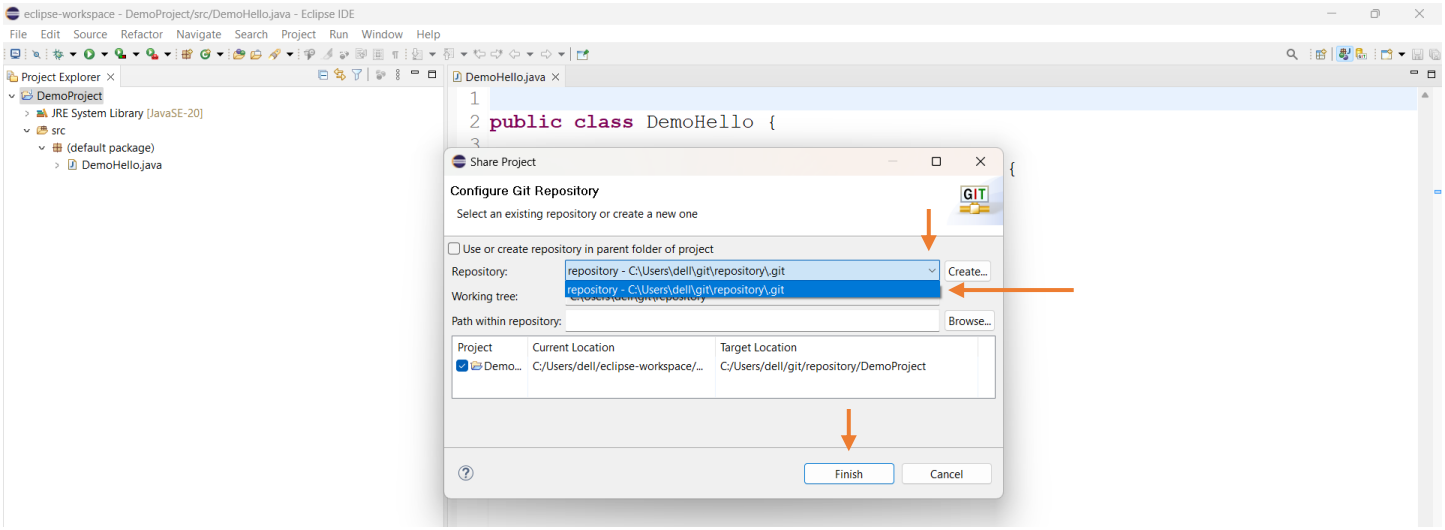
- Name the class as "**DemoWorld**"
- Check the "**public static void main (String args[])**"
- Click "**Finish**".



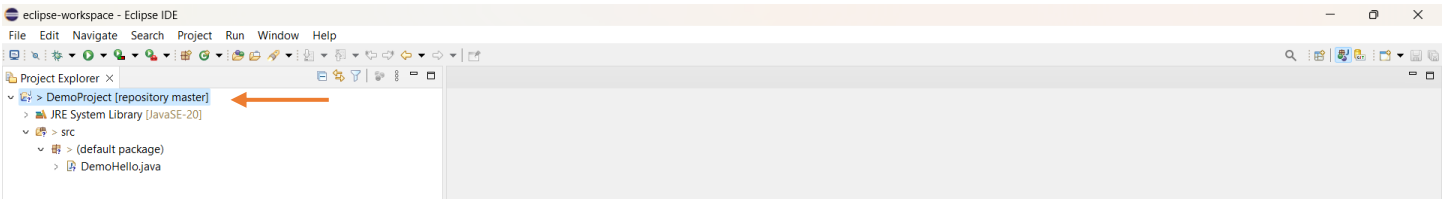
- Now write a small **Hello World** Program inside this.
- Now let's upload it to GitHub.
- Right click on "**DemoProject -> Team -> Share Project**".



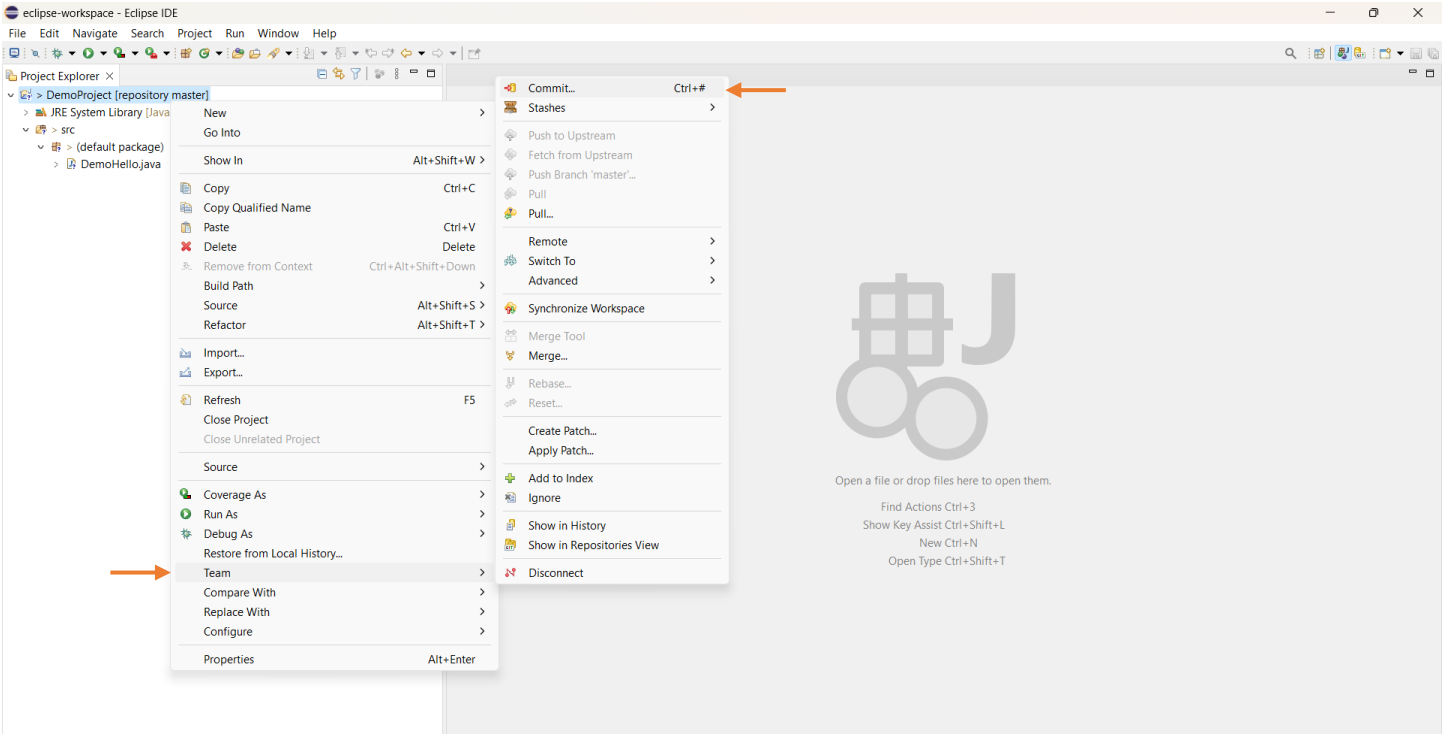
- **Select Repository** from **dropdown** and select the one which is available.
- Now Click "**Finish**".



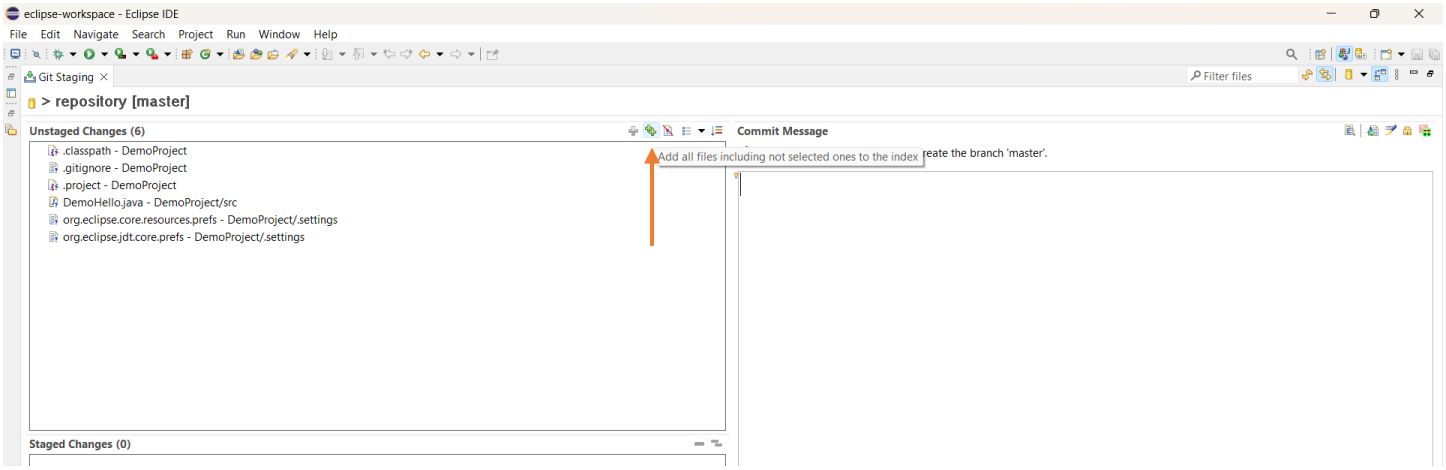
- Now in top-left corner you can see "**DemoProject[repository master]**".



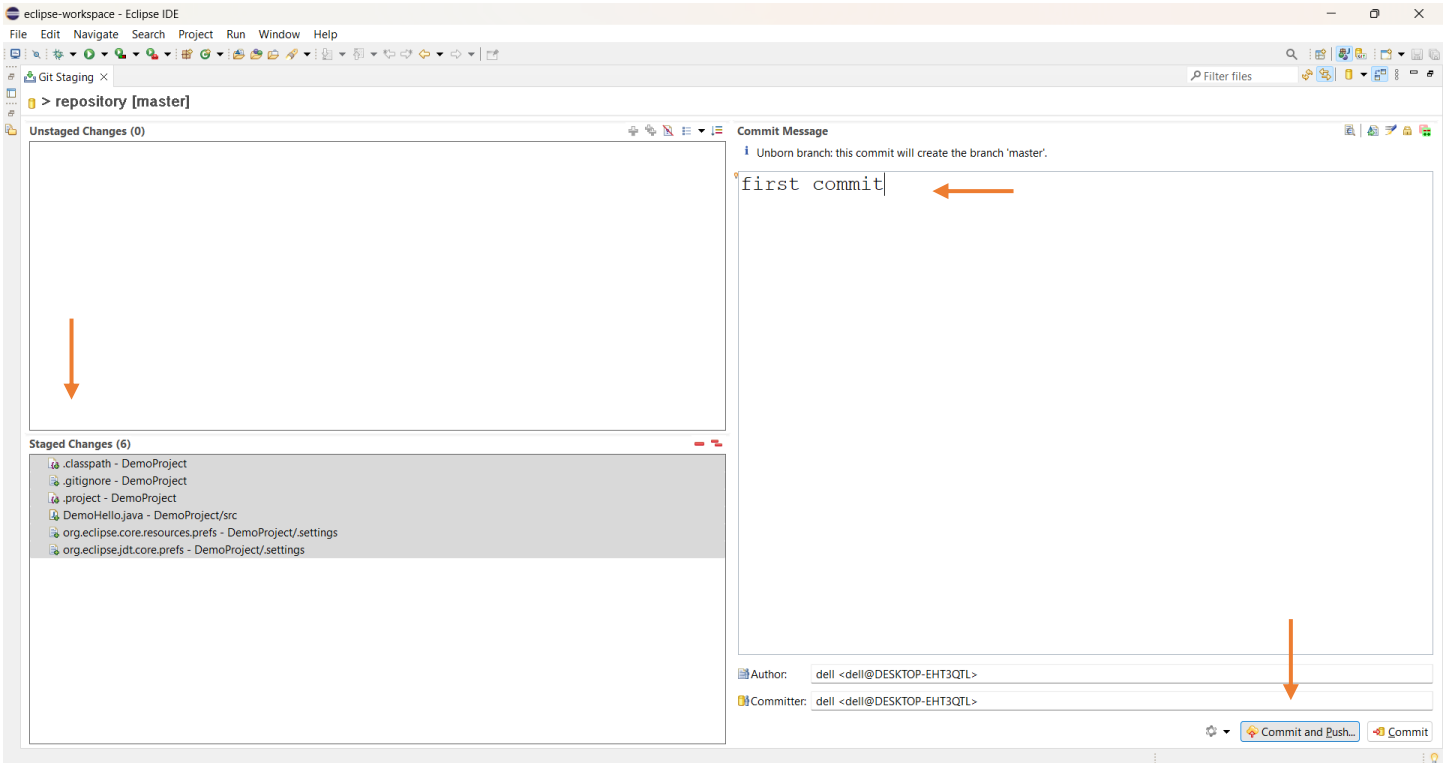
- Now Right-click on "**DemoProject -> Team -> Commit**".



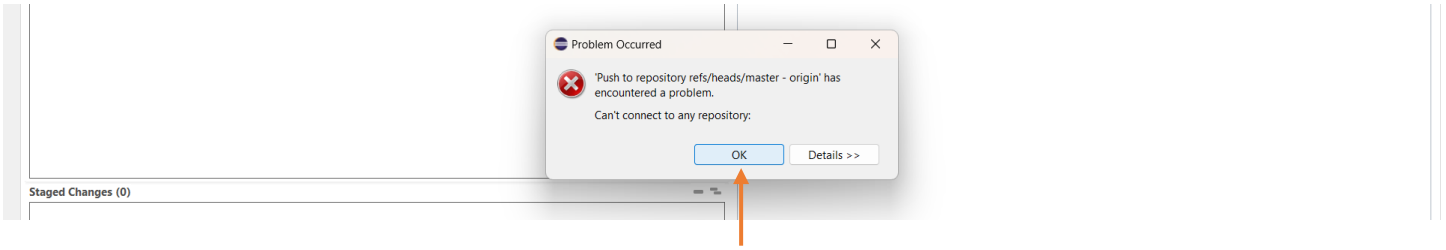
- Now **Git-Stagging window** is opened.
- Press “**++**” (**double-plus mark**).



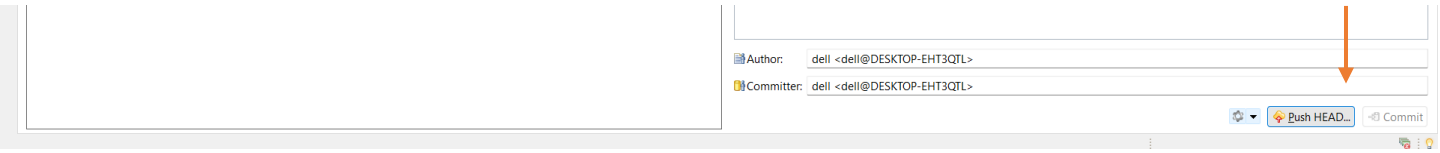
- Now you can see all the **Un-staged changes are moved to Staged Changes** below.
- Now in **Commit Message**, write “**first commit**” and click “**Commit and Push**”.



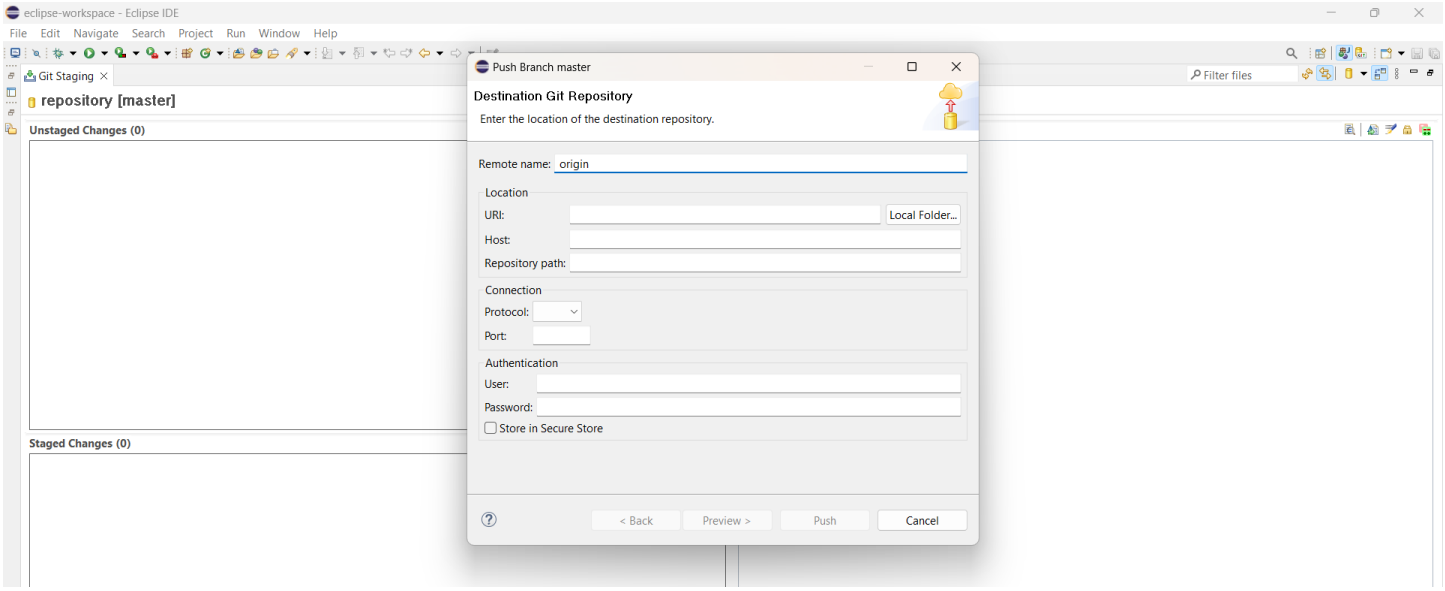
- If you get the below error, then you are going fine.
- Now Press “**OK**”.



- Now Click “**PUSH_HEAD**”.

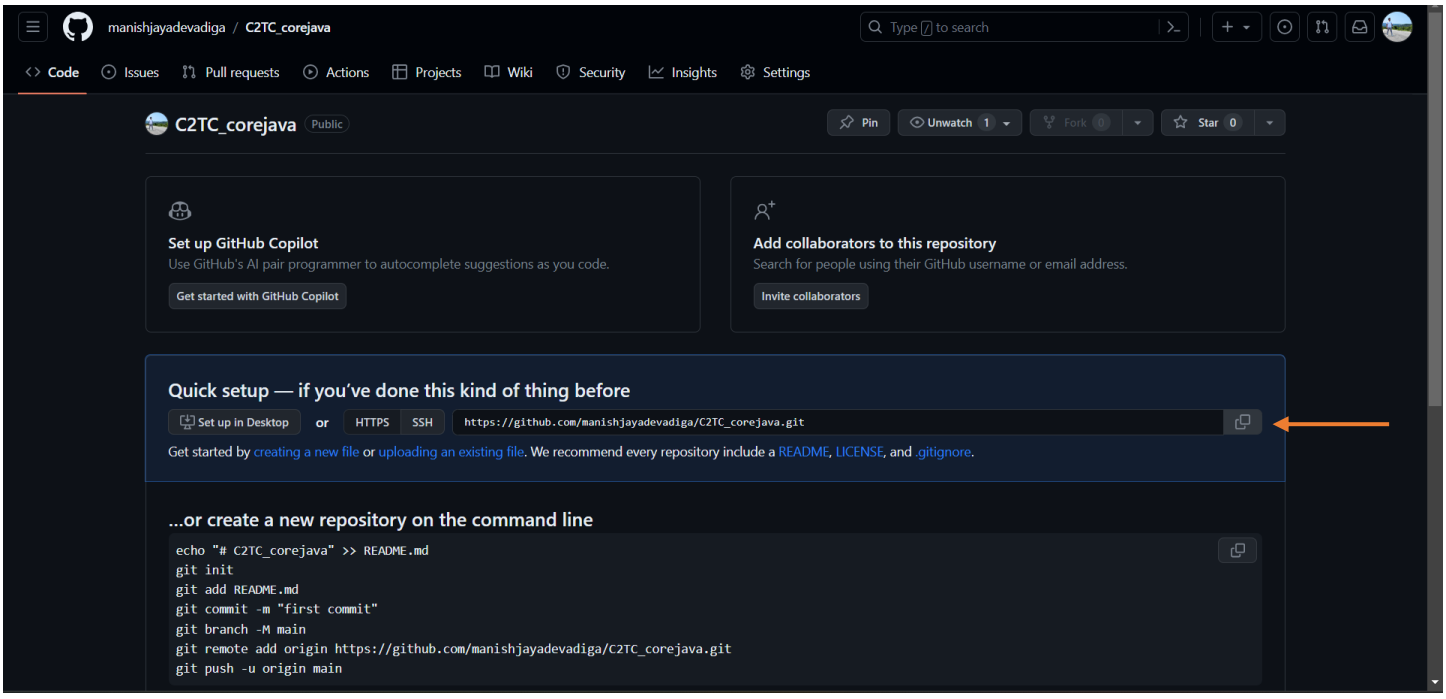


- Now Destination **Git Repository box** is opened.



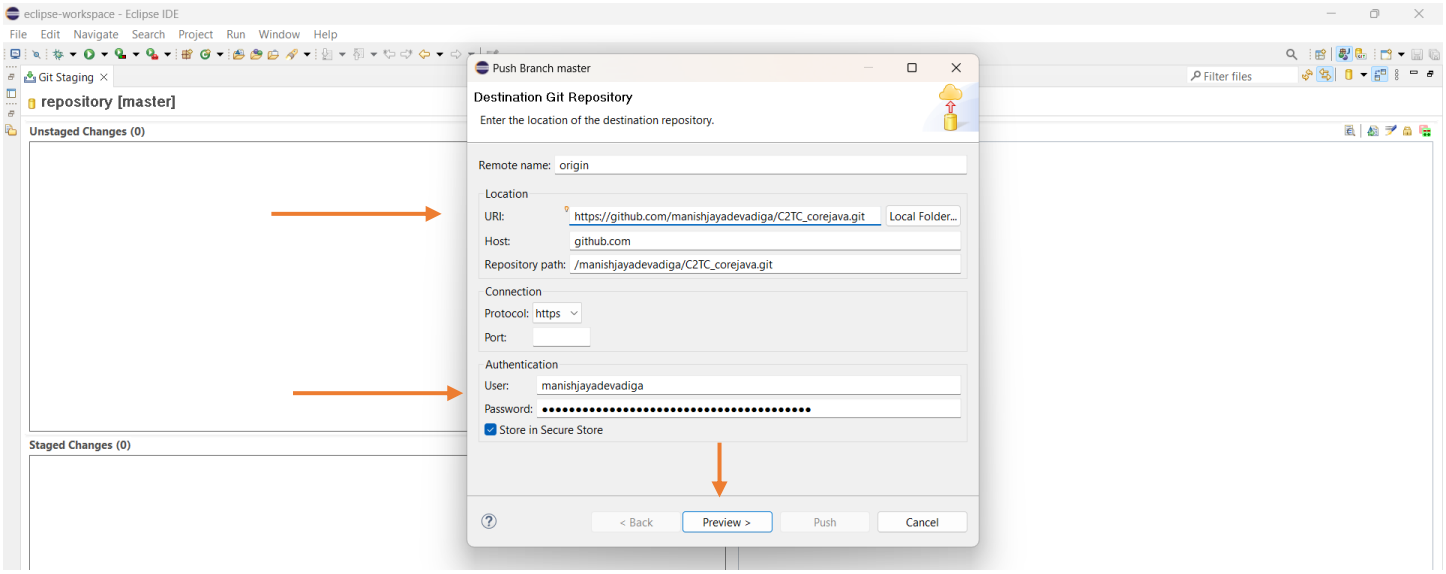
GitHub:

- Goto GitHub and “**Copy Repository Link**”.

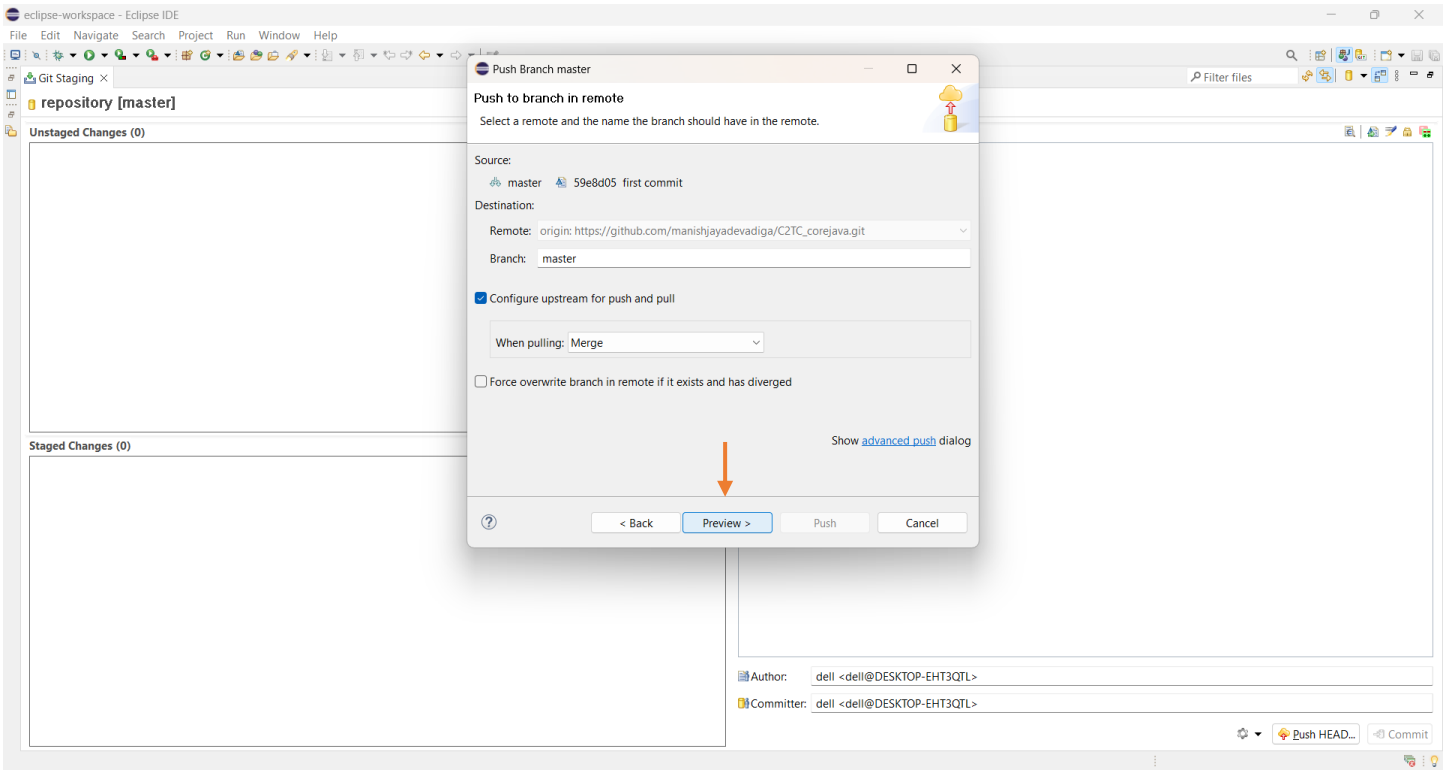


Eclipse:

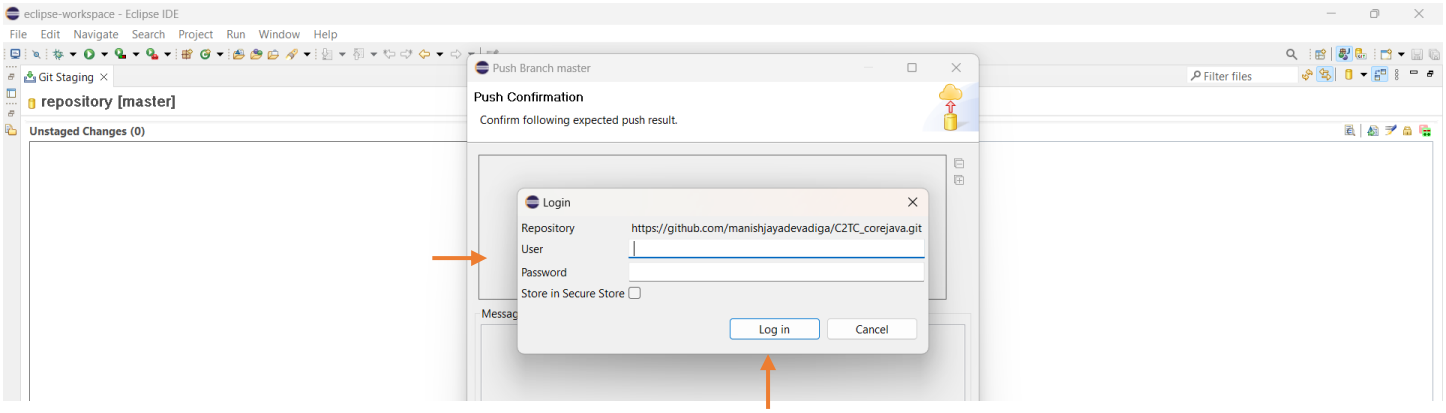
- **Paste** the copied link in "**URL section**" of the dialog box.
- Now in **Authentication**, "**Give your GitHub Username and GitHub Password**".
- Check "**Store in Secure Store**" box and click "**Preview**".



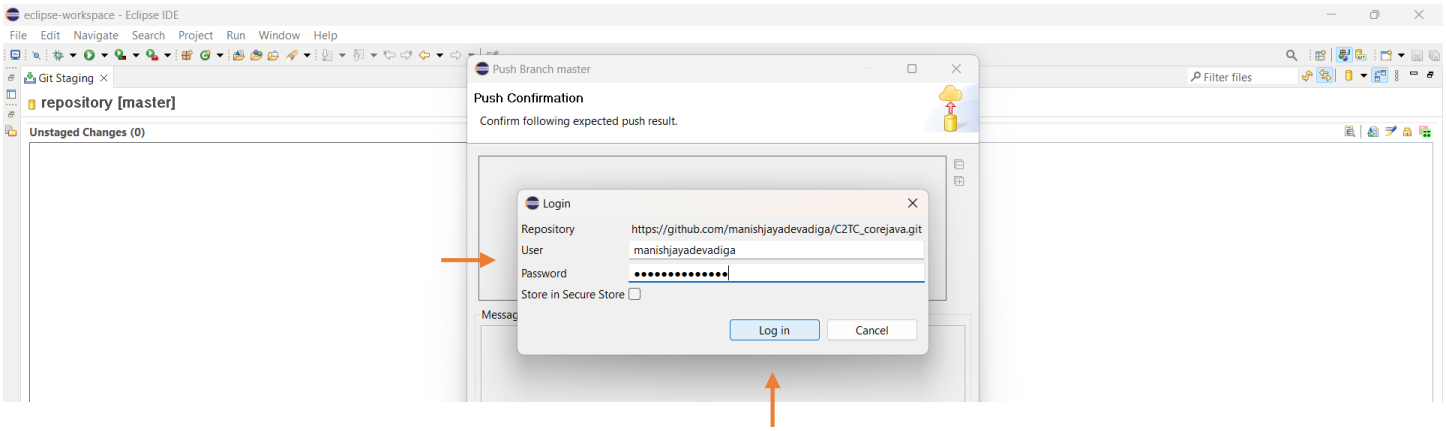
- Now click "**Preview**" again.



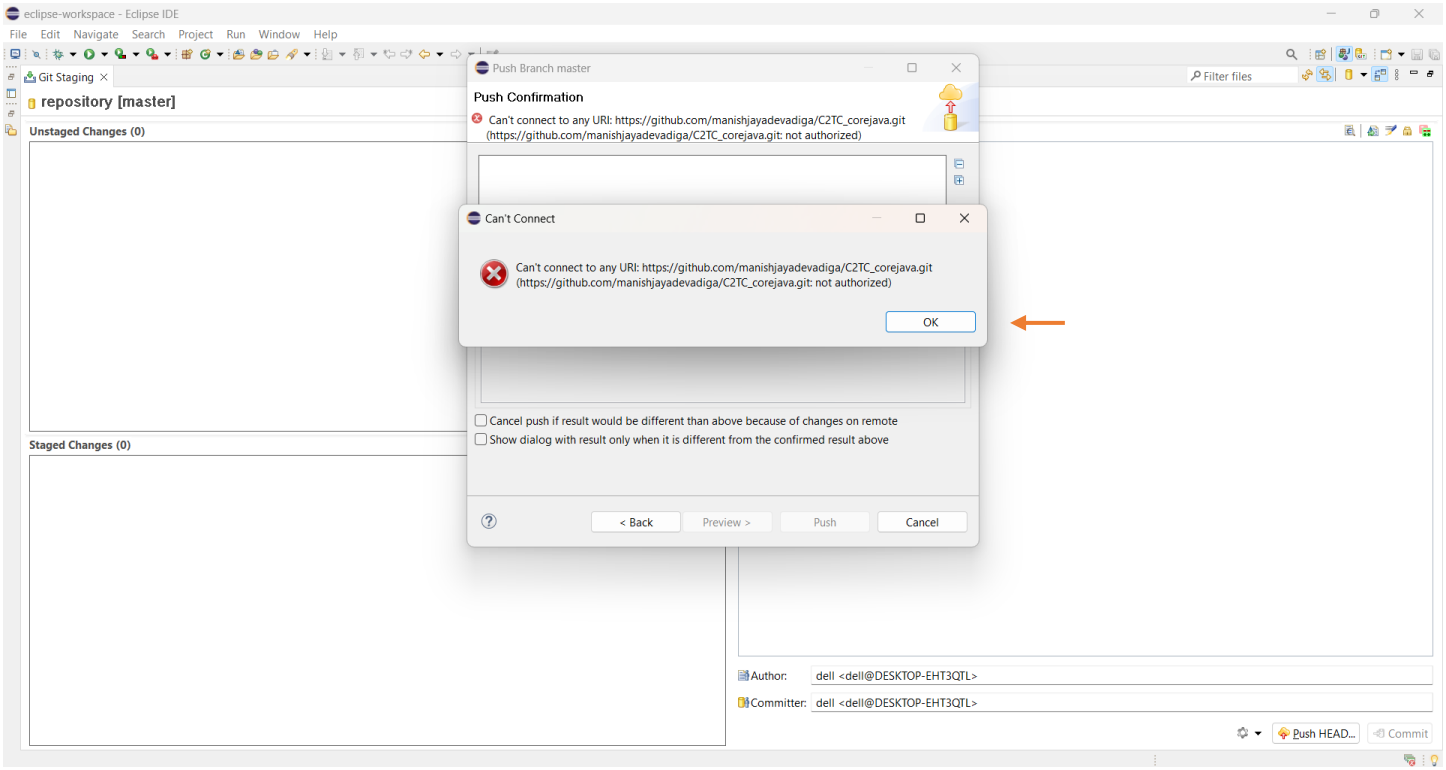
- A Login dialog box is opened.
- Give **GitHub username and GitHub password** and click "**Login**".



- **Do** the above step **again**.

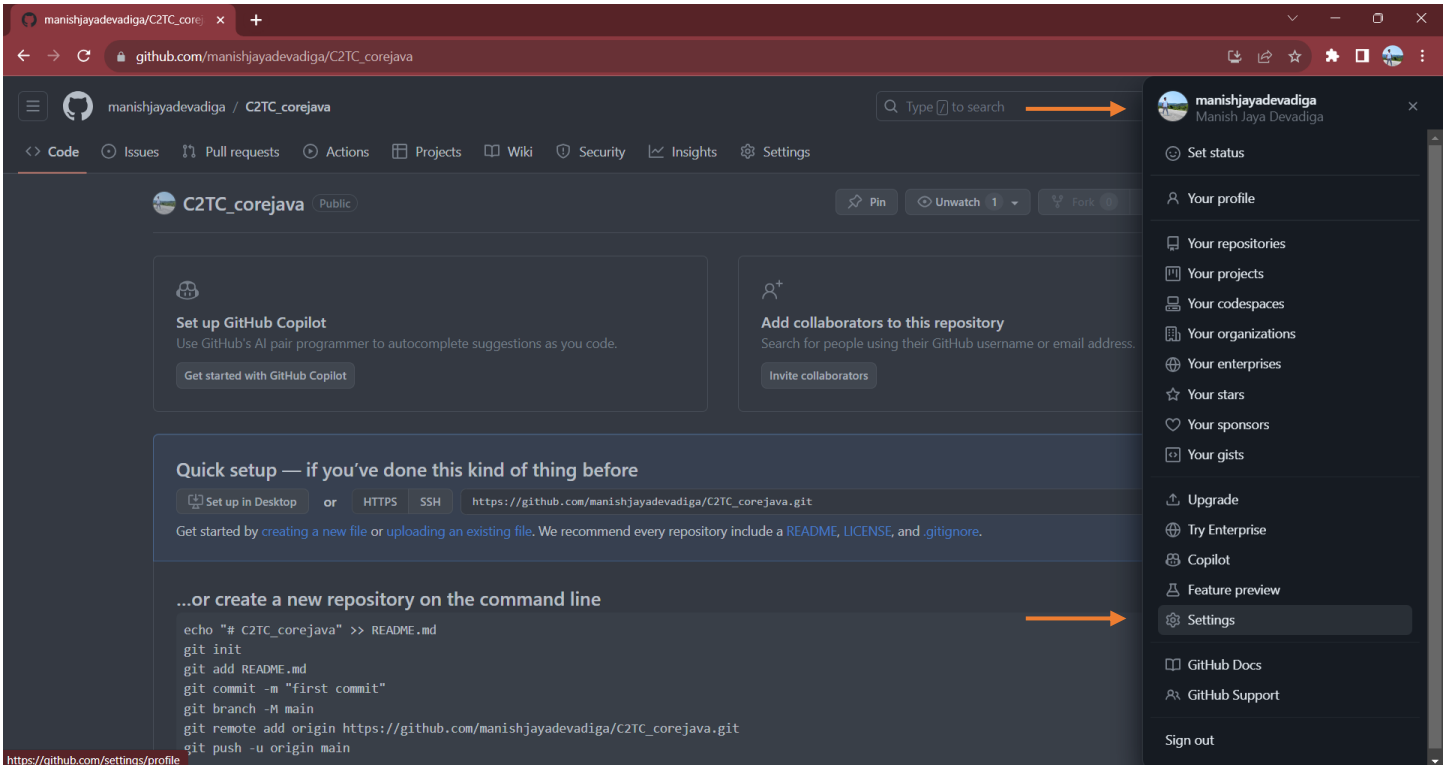


- Now **Can't Connect box** is displayed, Click "**OK**".

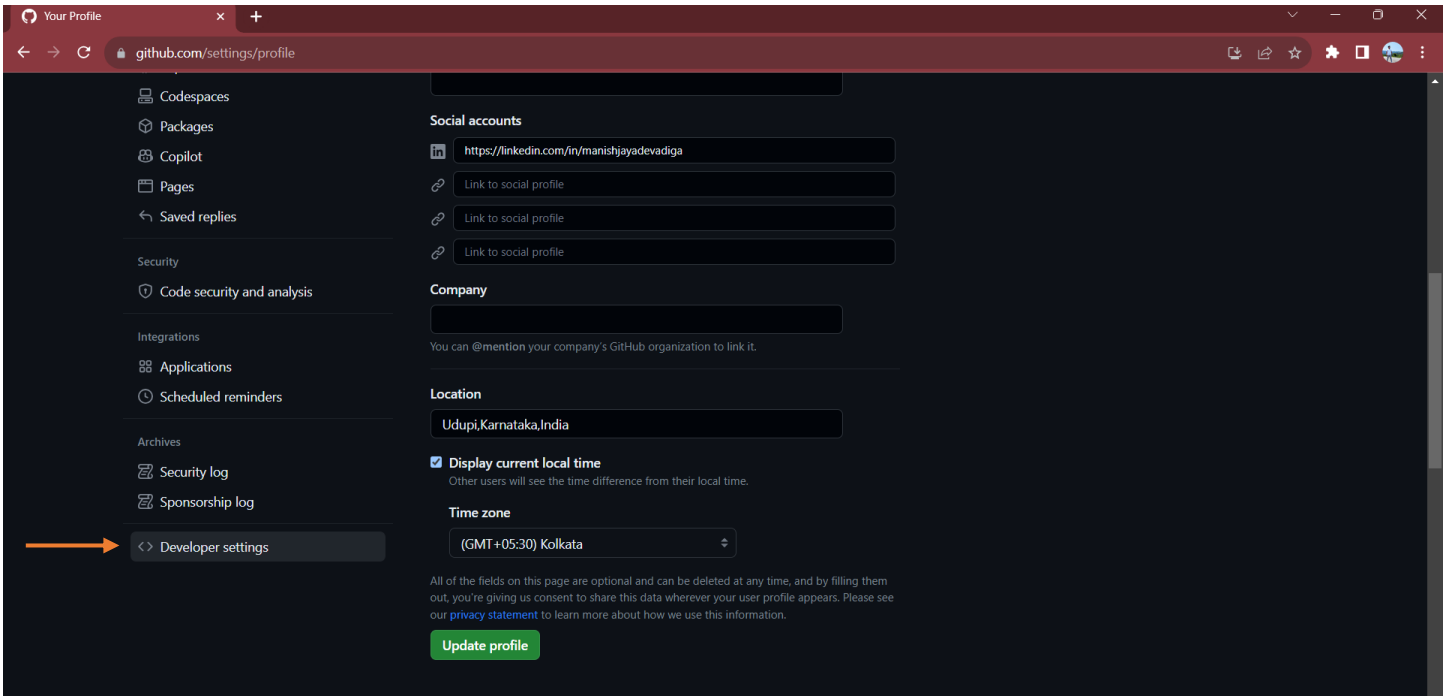


GitHub:

- Goto GitHub, "**Account (Profile Icon) -> Settings** ".

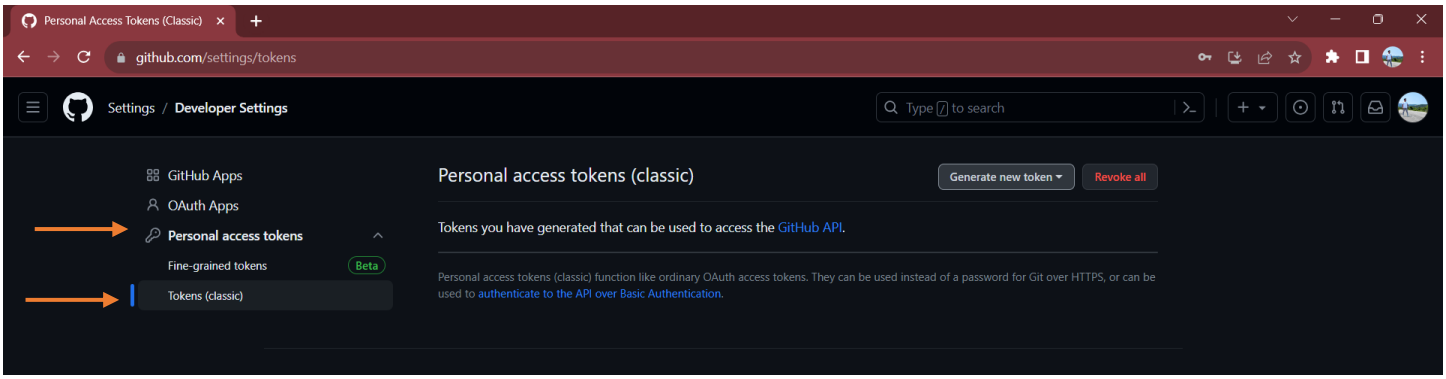


- Now Scroll Down to "**Developer Settings**".

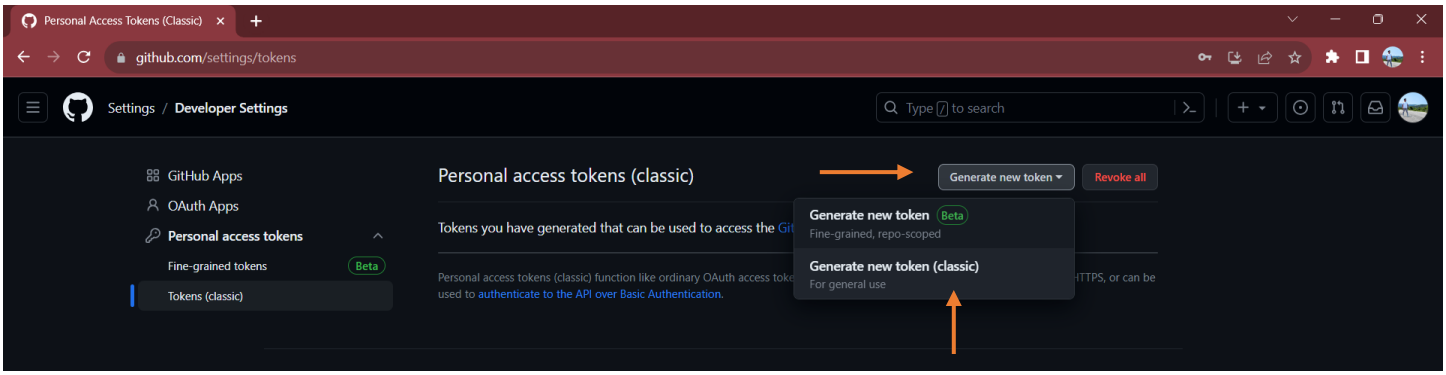


- Click "**Developer Settings**".

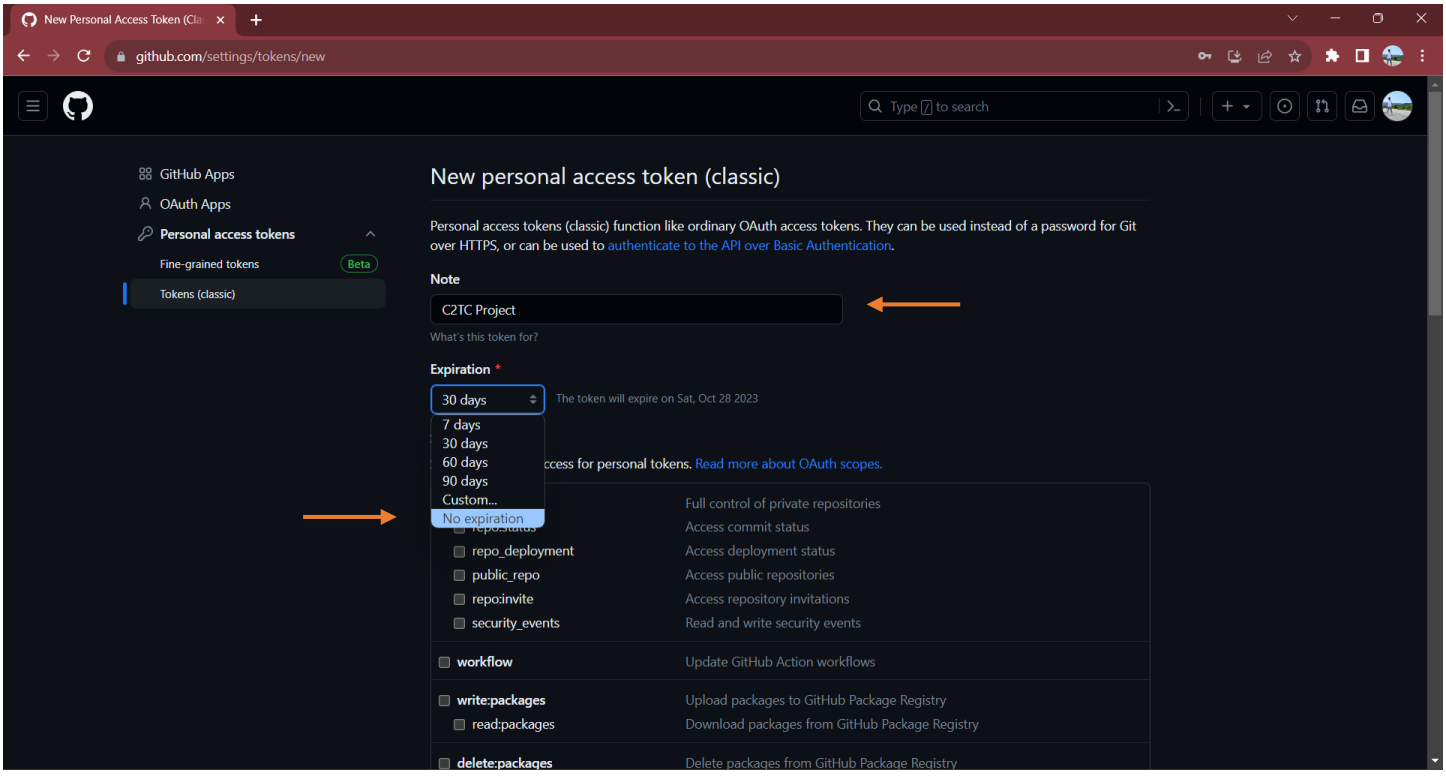
- Now click “**Personal Access tokens ->Tokens(classic)**”.



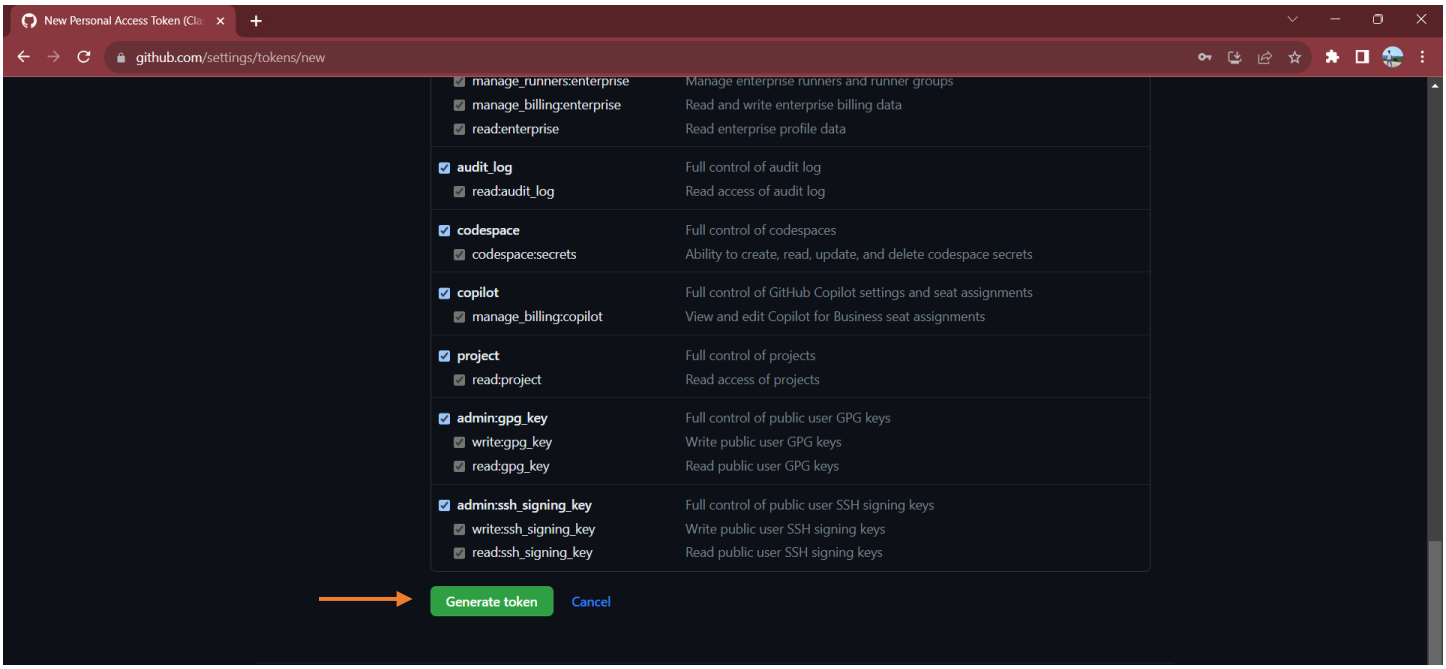
- Now click on “**Generate new token**”, and select “**Generate new token (classic)**”.



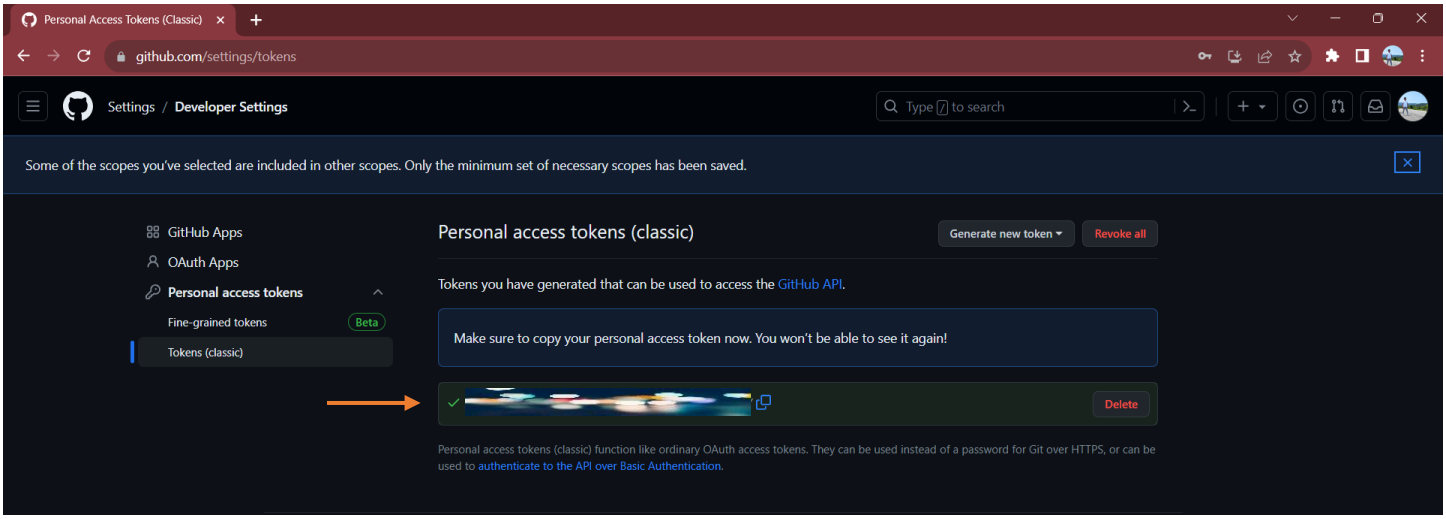
- Now give **Note** as “**C2TC Project**” and **set the Expiration** to “**No-expiration**”.



- **Check “All” the below boxes** and click “**Generate Token**”.

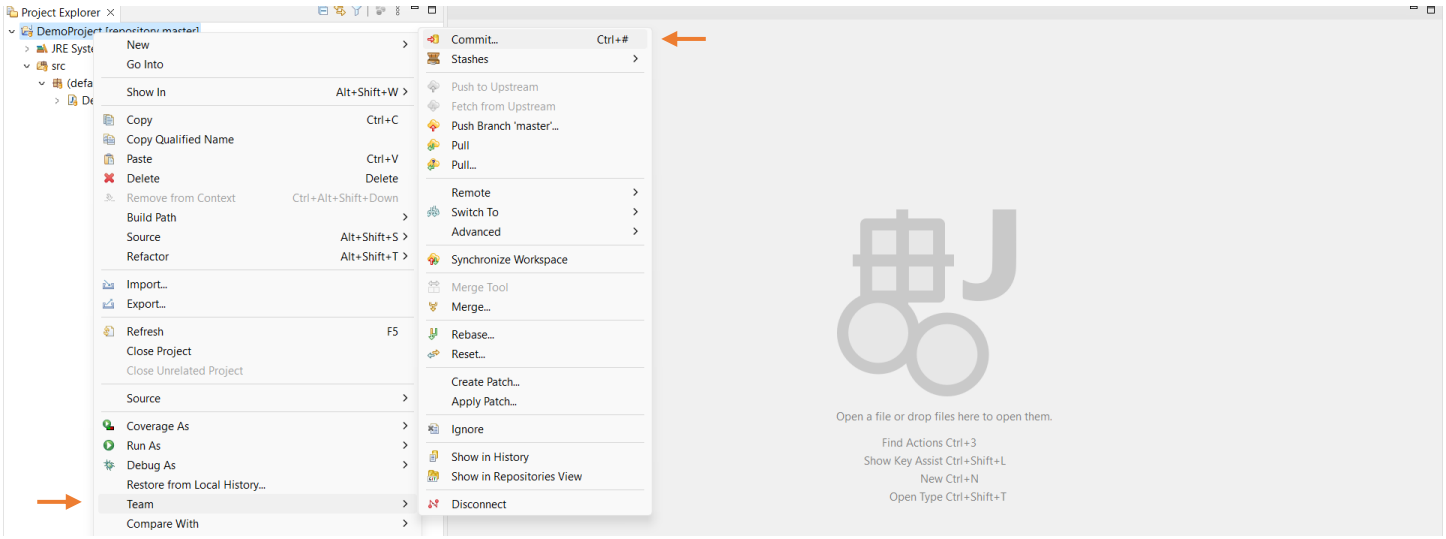


- Now you can see that the token is generated.
- **Copy this token and keep it safe**. We will not get this link(code) again and we have to use this in future steps.

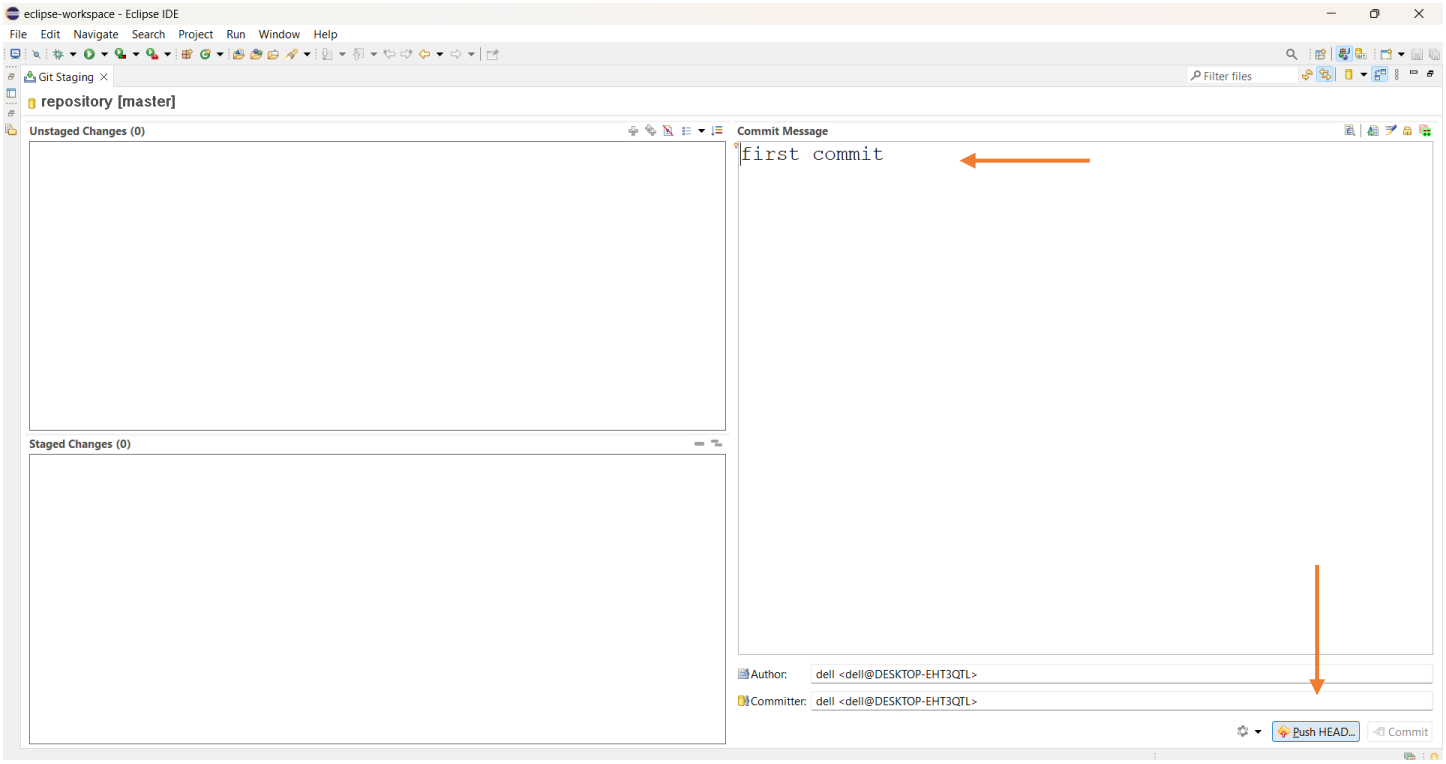


Eclipse:

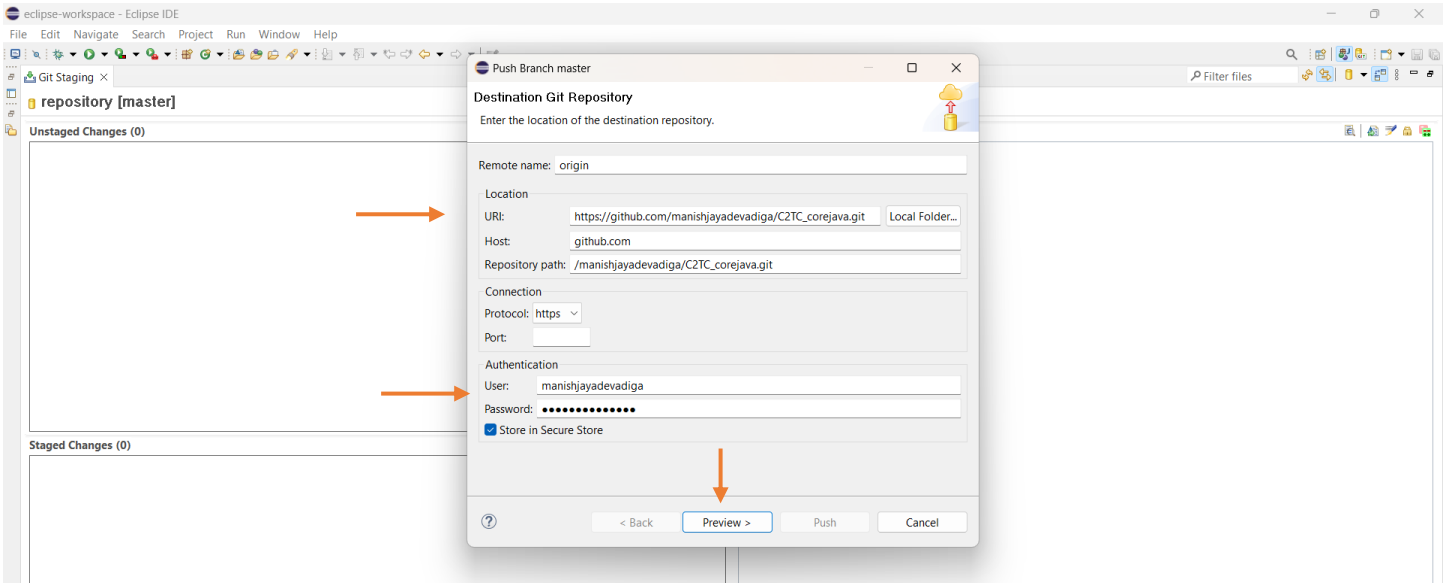
- Do the same steps which was done before.
- **Right click** on project “**DemoProject -> Team -> Commit**”.



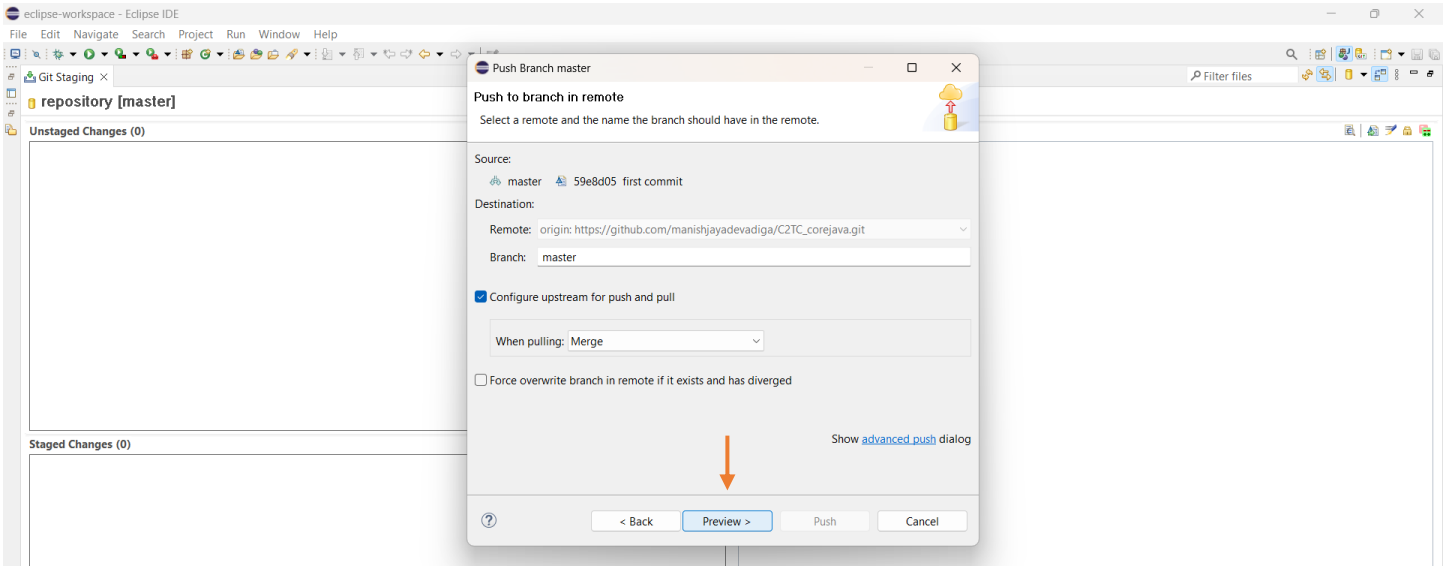
- Now write "**first commit**" and click "**PUSH_HEAD**".



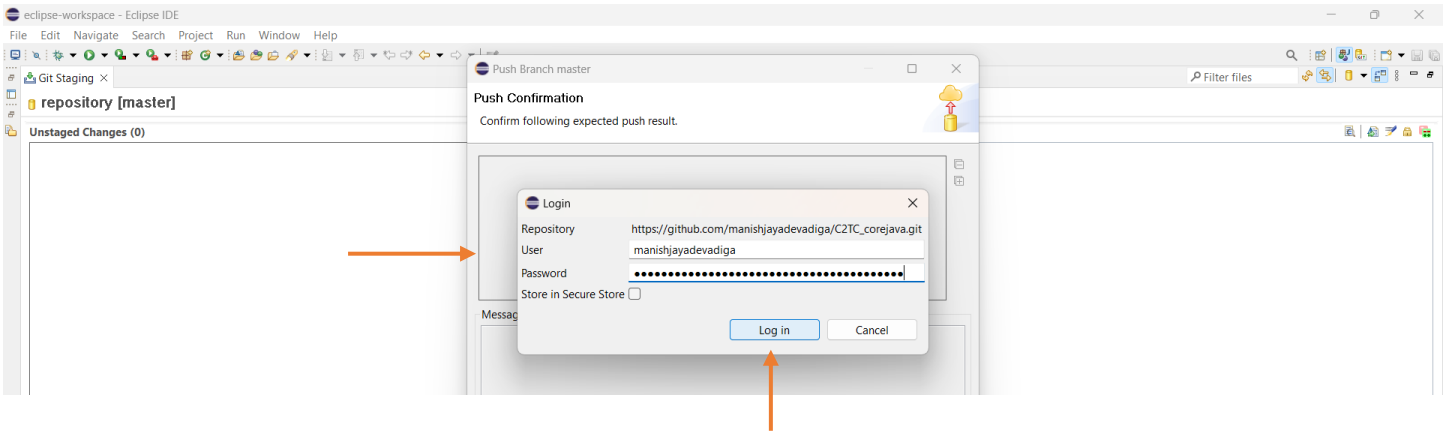
- Now Destination Git Repository box is opened.
- Give "**Repository URL, GitHub Username and GitHub Password**".
- Check "**Store in Secure Store**" box.
- Now click "**Preview**".



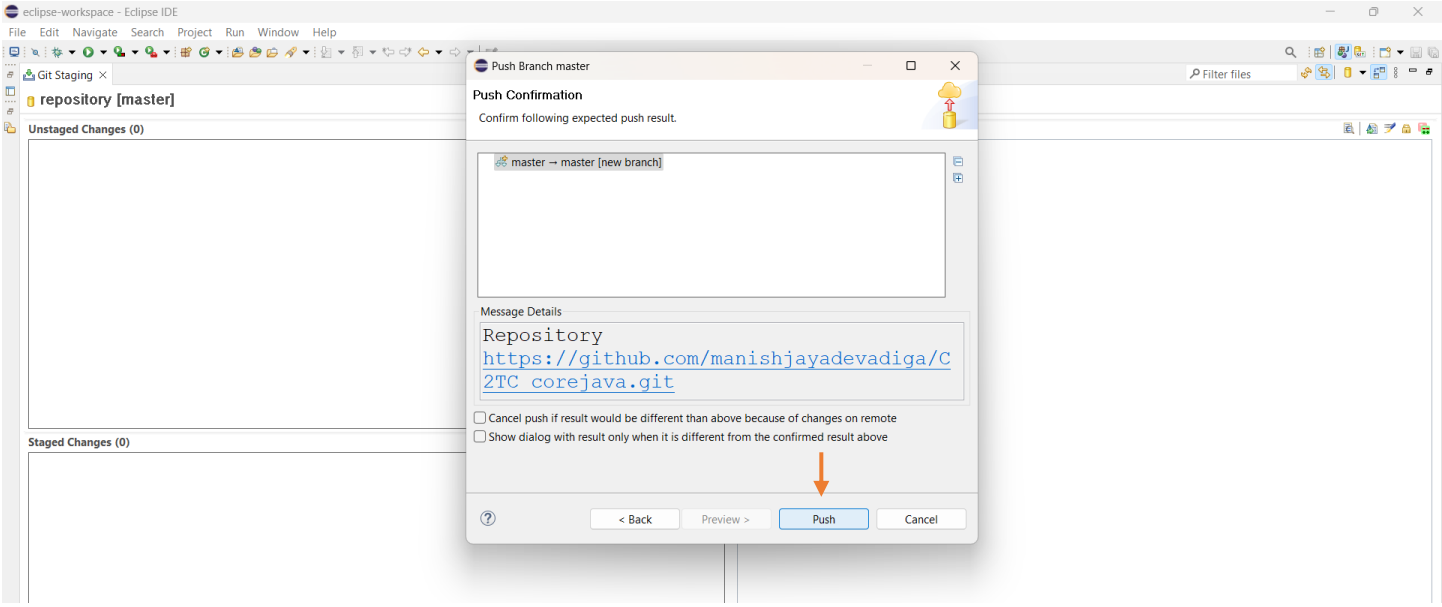
- Press “**Preview**” again.



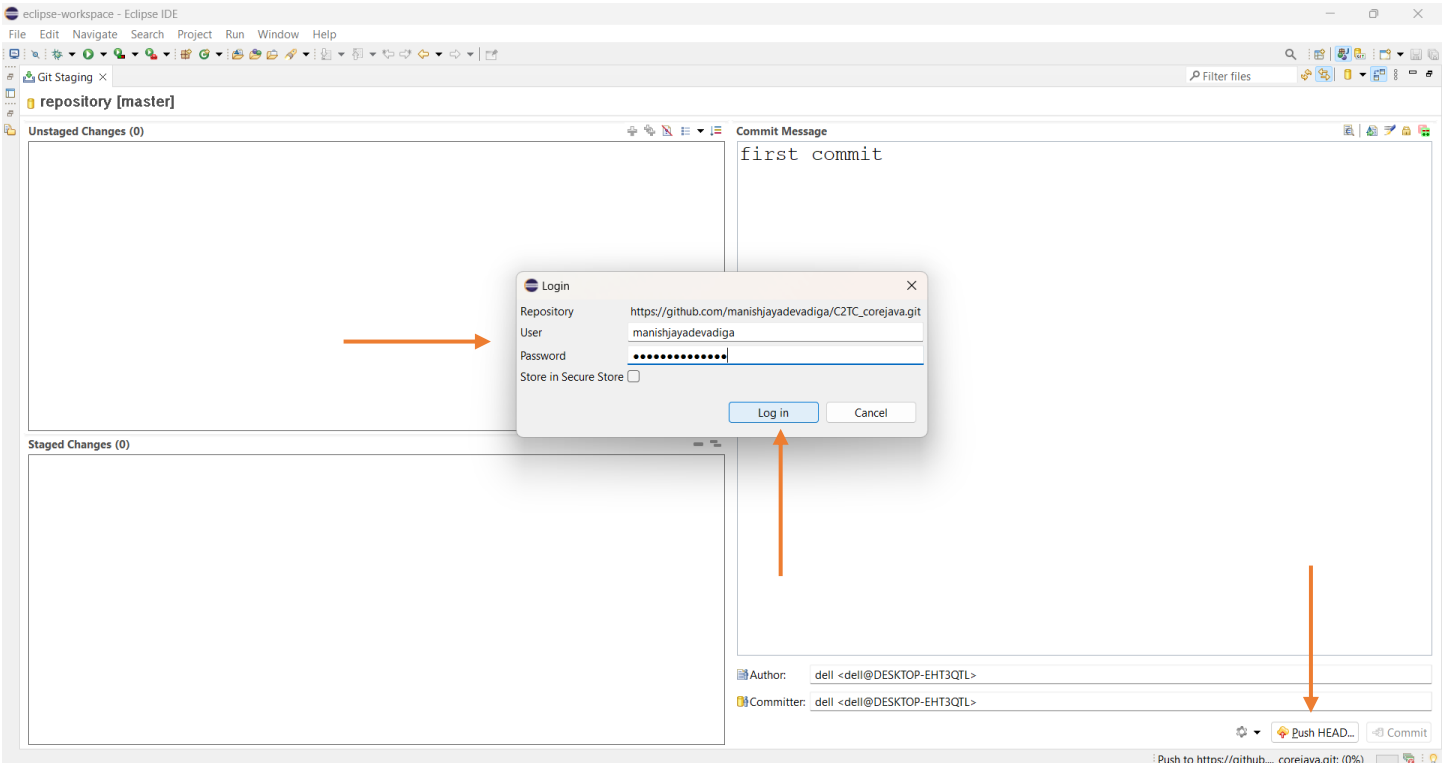
- New **Login Box** is opened.
- Give “**GitHub User Name**”.
- In the Password section “**Paste the token generated previously**”, **Uncheck** “**Secure Store**” and click “**Login**”.



- Now **Push Confirmation Box** is opened.
- You can observe "**master -> master [new branch]**".
- Don't do anything, just click "**Push**".

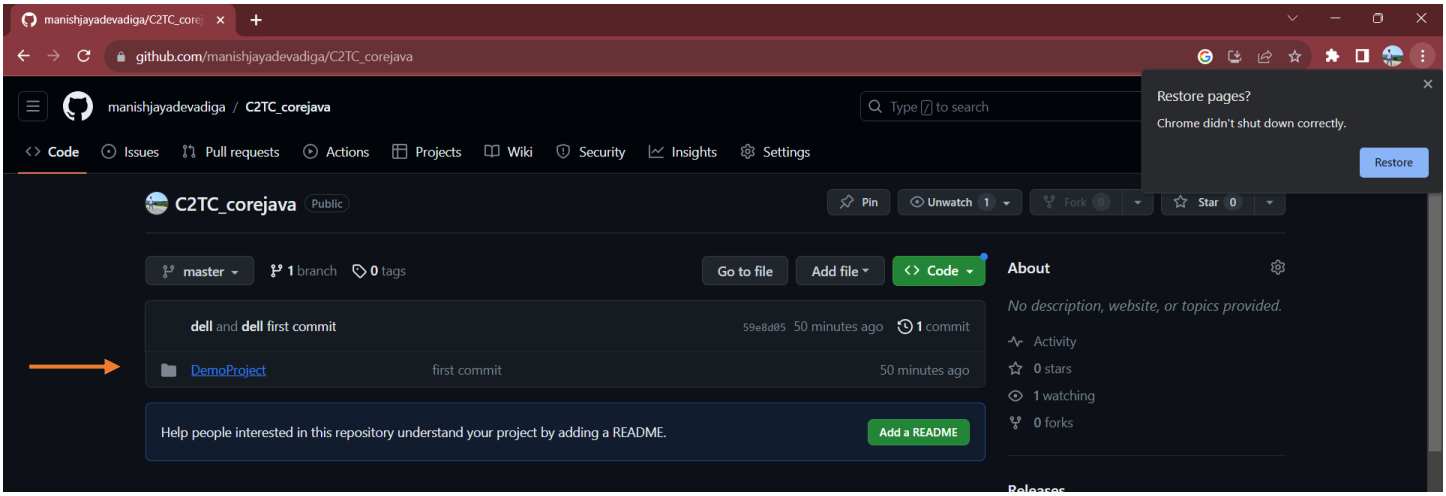


- New **Login Box** will appear.
- Give **GitHub Username and GitHub password** and click "**Login**".
- Now press "**PUSH_HEAD**".



GitHub:

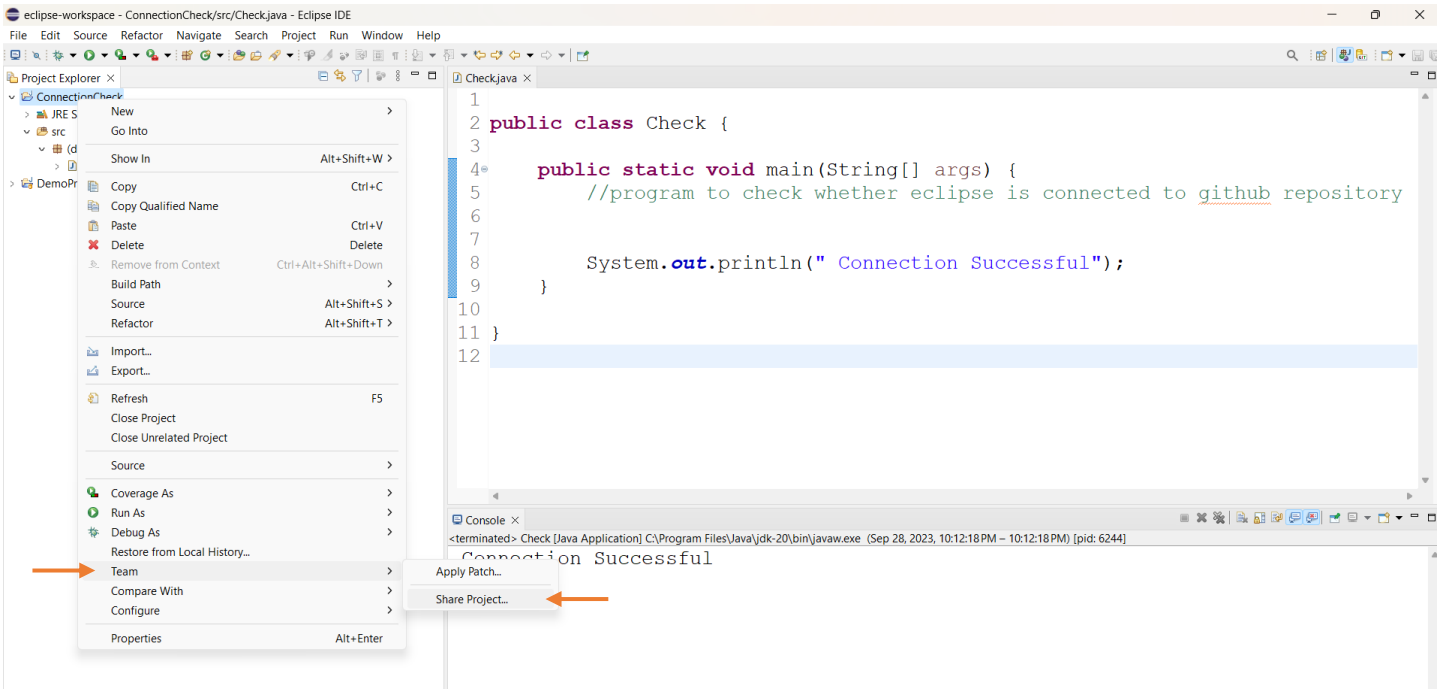
- Now open GitHub.
- Here **you can see the files (DemoProject) which is pushed from eclipse to GitHub repository** C2TC_corejava.
- Click it, you will find programs in **src** folder.



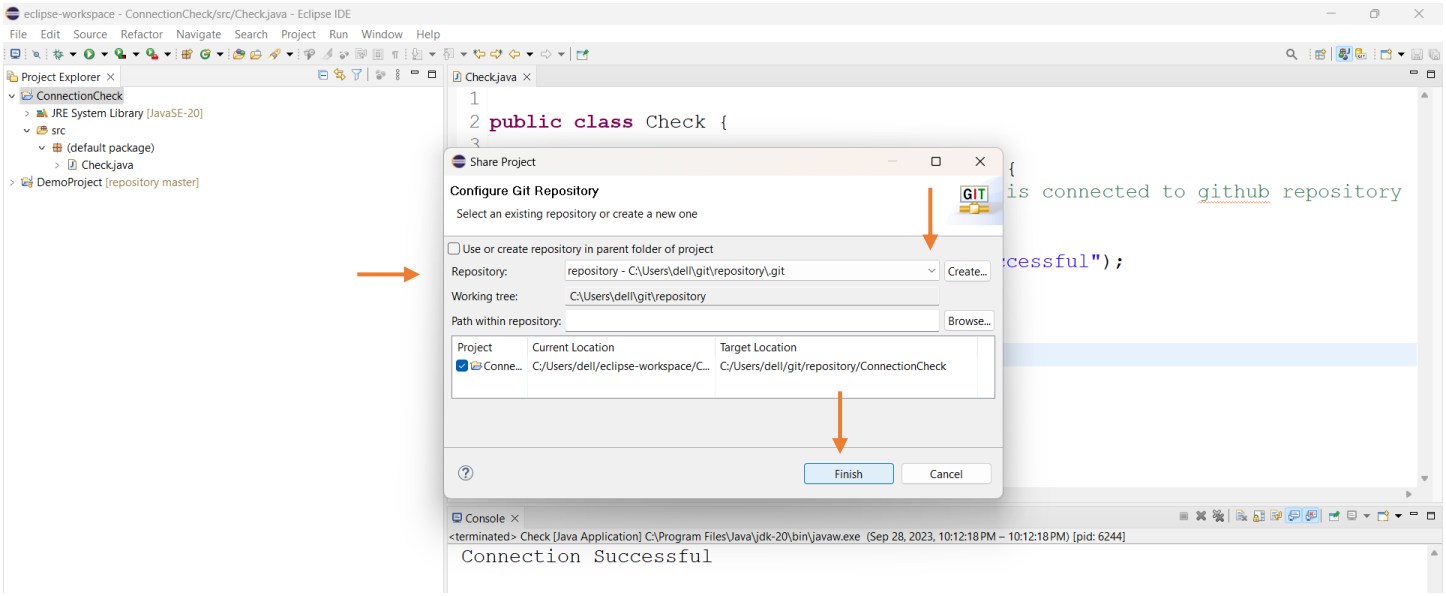
- **We are now done** with connecting Eclipse and GitHub.
- Now let us see how we can **push future files to GitHub**.

Eclipse:

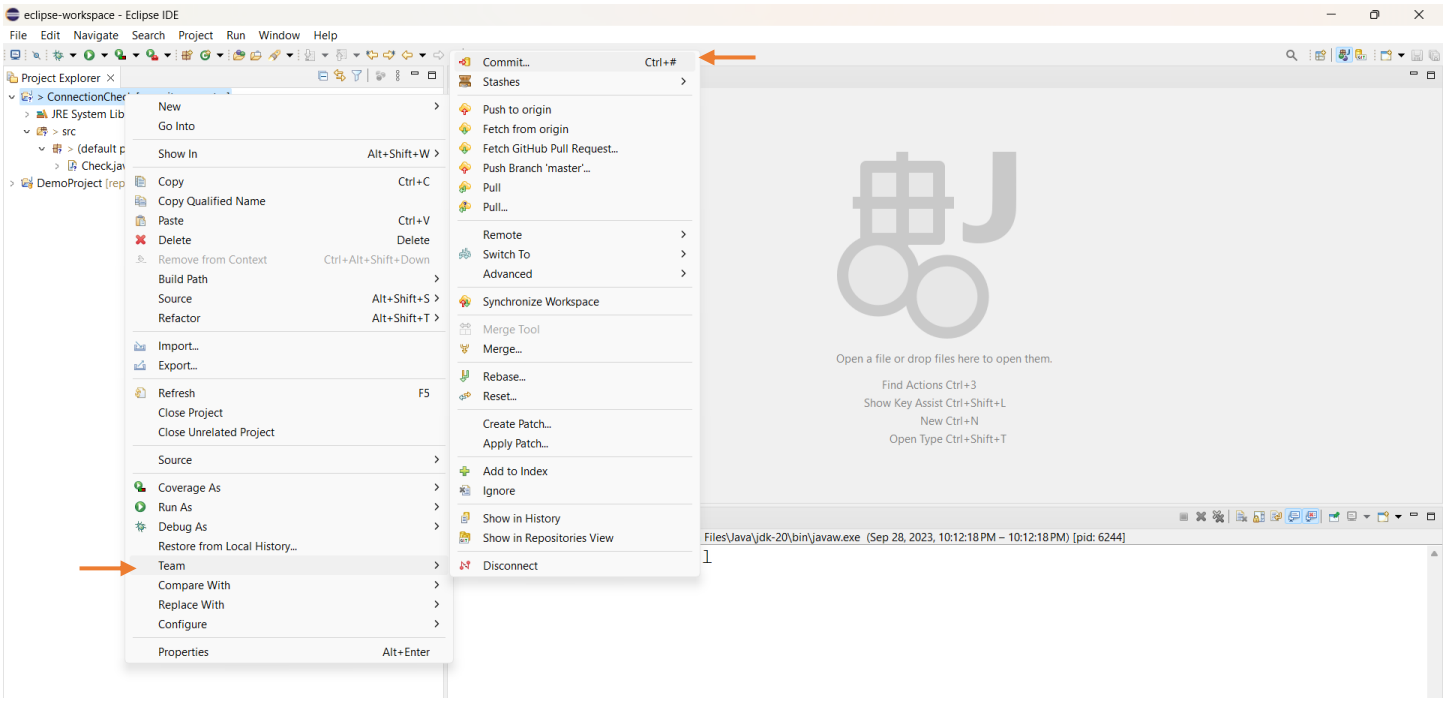
- Create a new java project "**ConnectionCheck**".
- Right click on "**ConnectionCheck -> Team -> Share Project**".



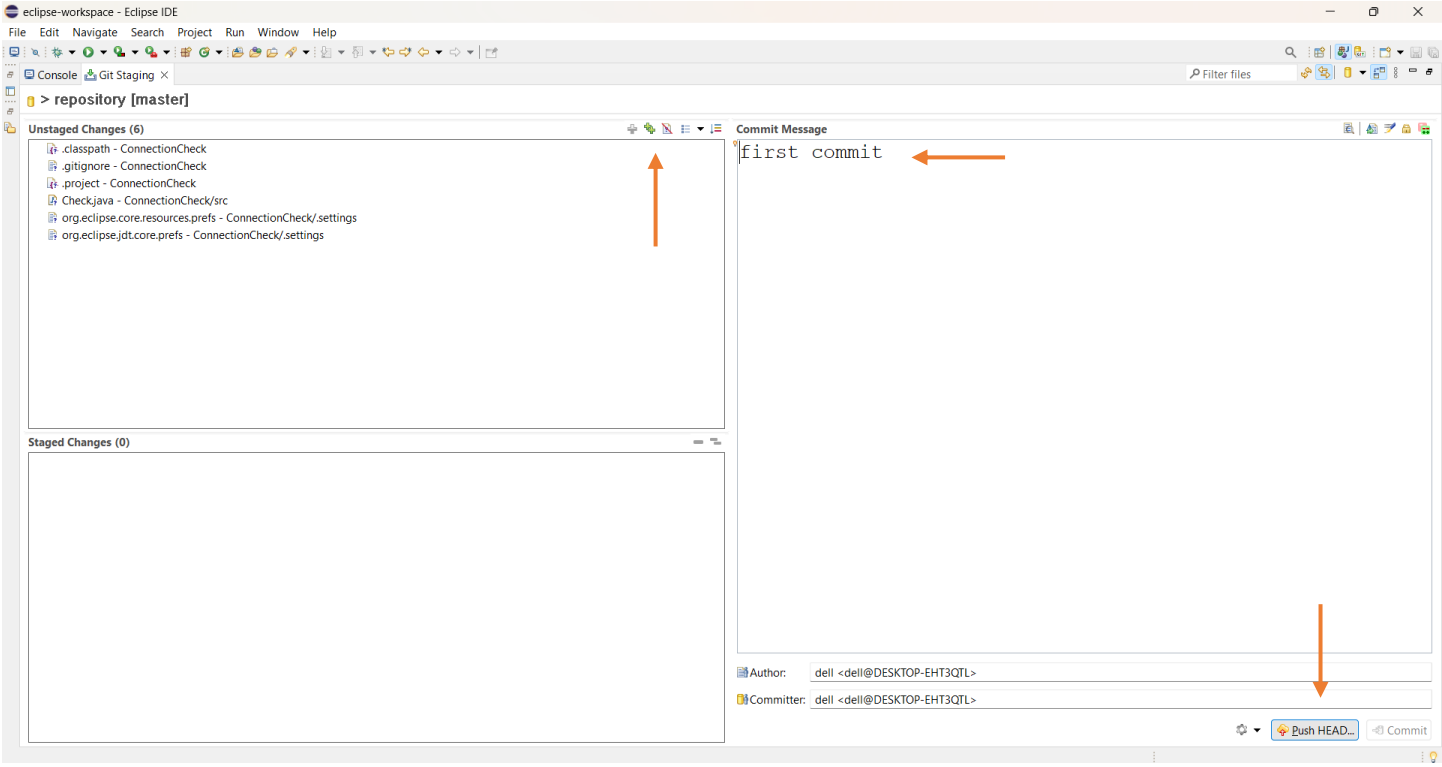
- **Select repository** from drop down and click “**Finish**”.



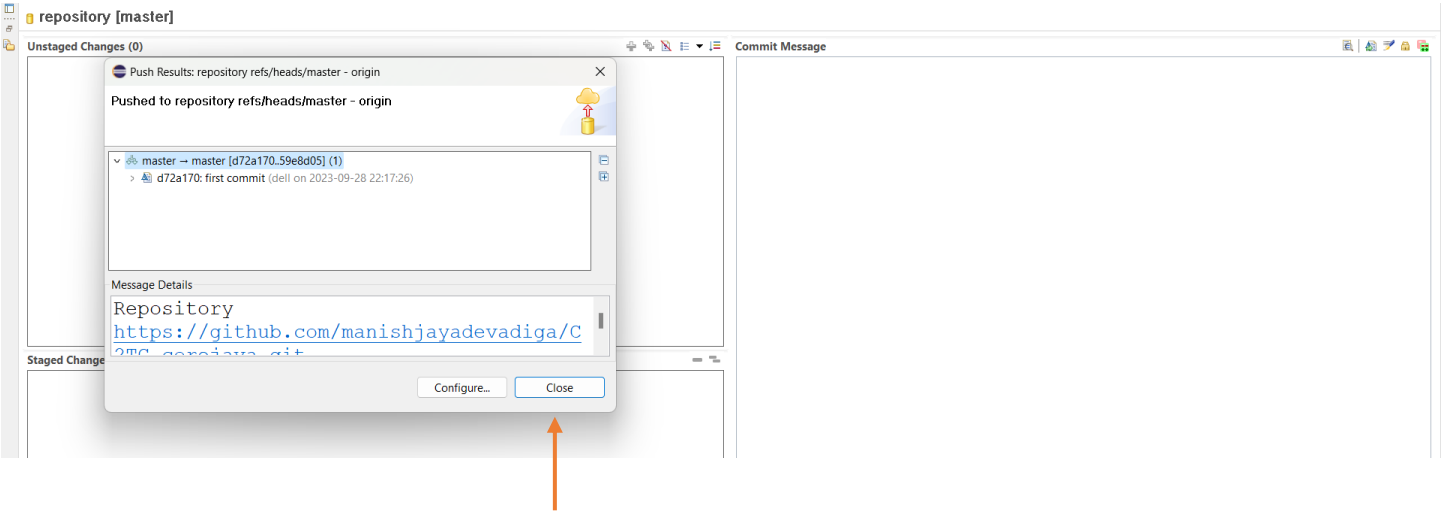
- Right click “**ConnectionCheck -> Team -> Commit** “.



- Press “**++ ->first commit**”.
- Click on “**Commit and Push**”.

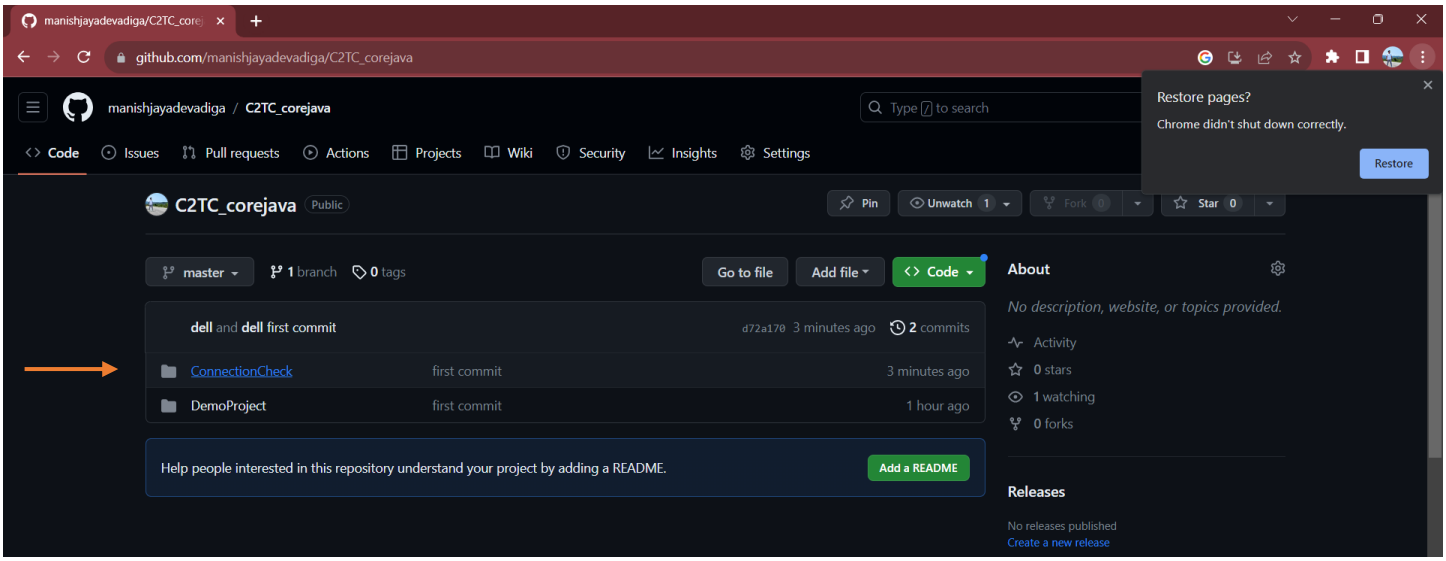


- You must see the below box. [if not, I have discussed the possible error in the next pages]
- Click "**Close**".



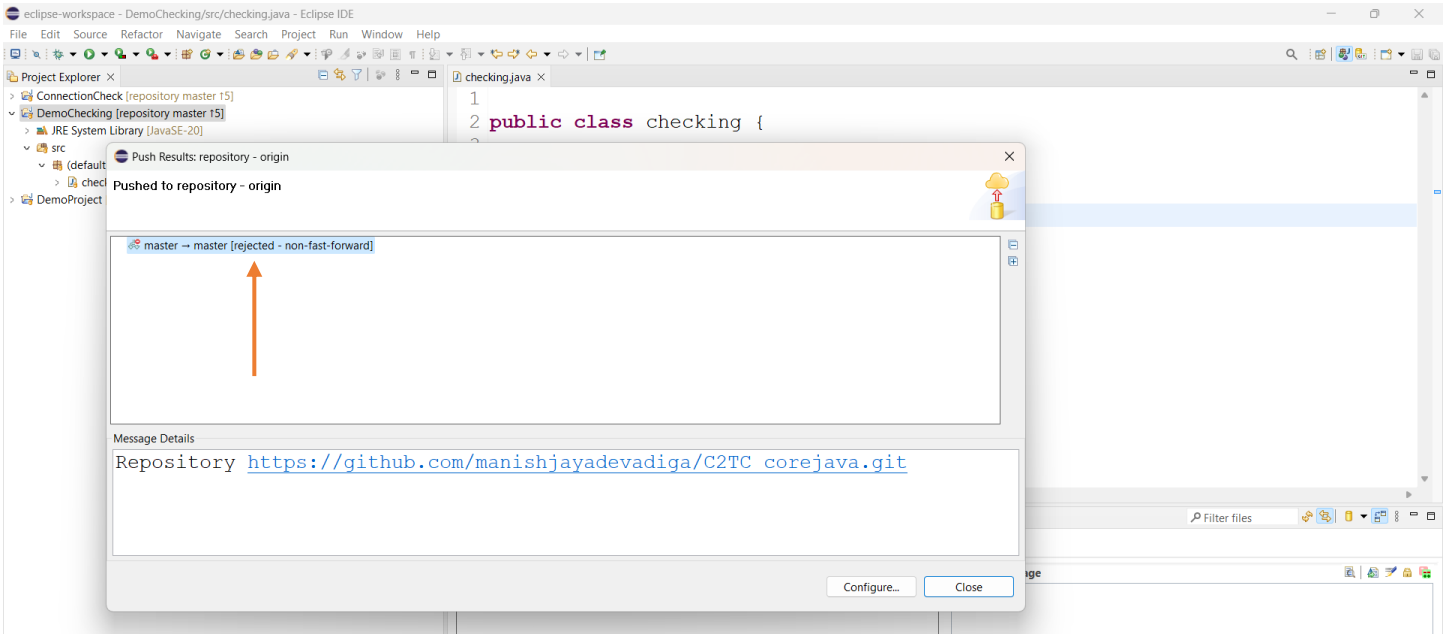
GitHub:

- Now open GitHub.
- Here **you can see the files (ConnectionCheck) which is pushed from eclipse to GitHub repository** C2TC_corejava.
- Click it, you will find programs in **src** folder.



Eclipse:

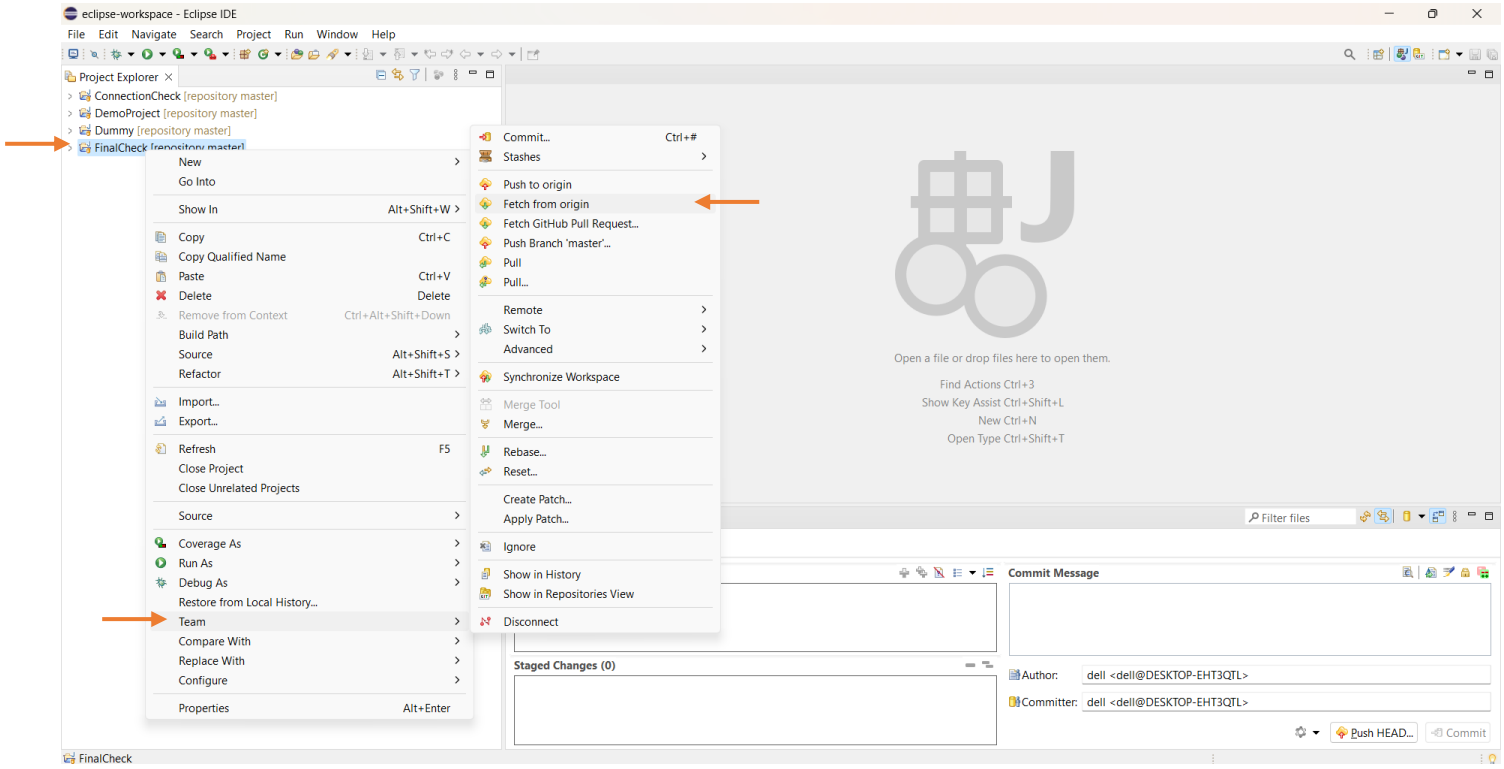
- If you get **ERROR**:**master ->master [rejected – non-fast-forward].**



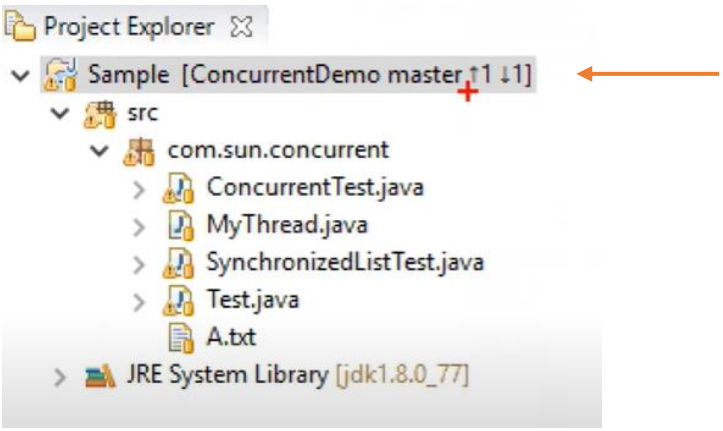
- Do the following steps carefully.
- **Close** the above dialog box.

Let's Solve This:

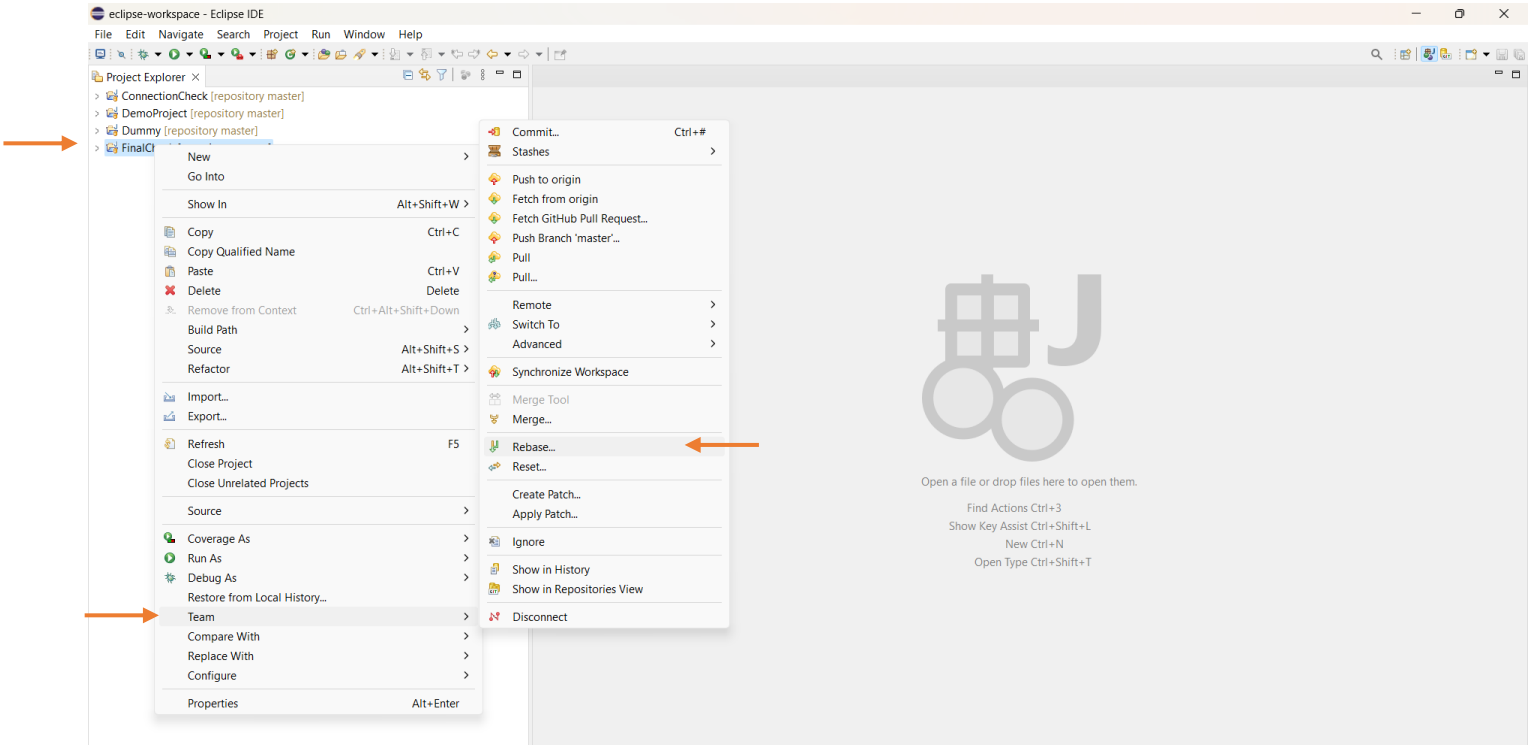
- Right click "**Project -> Team -> Fetch from Origin** ".



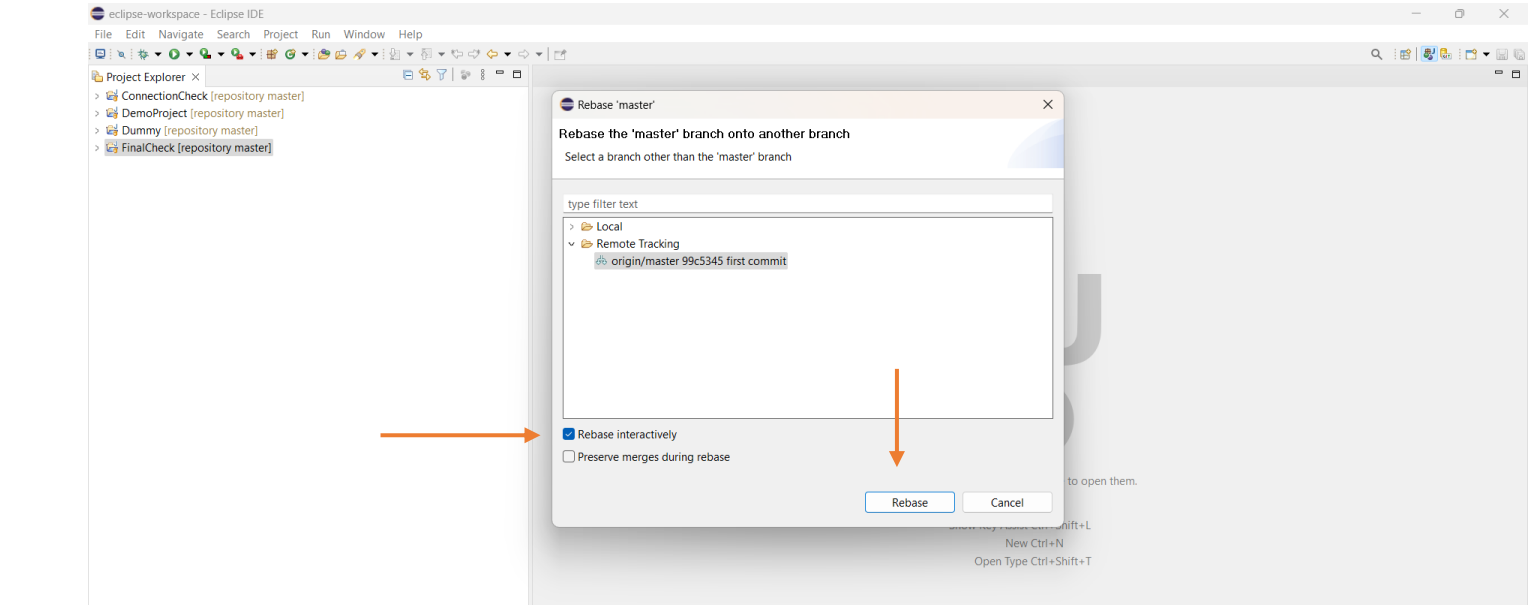
- Now in top left we will see one up arrow, one down arrow.



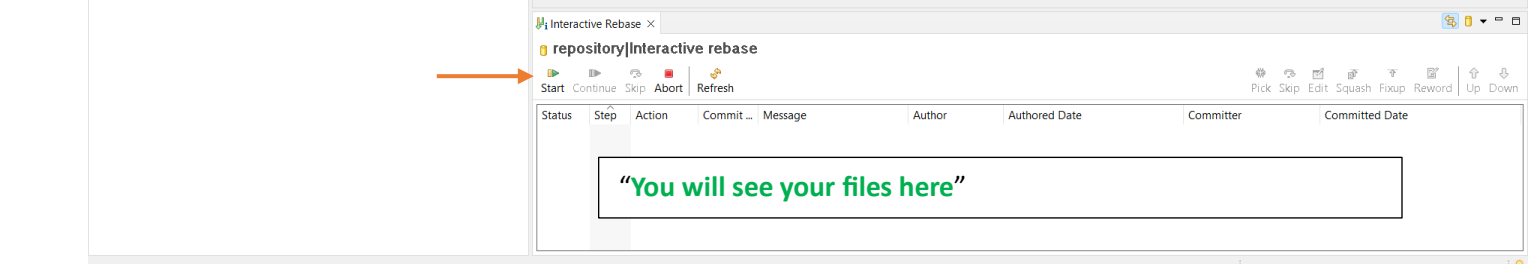
- Before pushing we need to "**perform rebase**".
- Right click on "**Project -> Team -> Rebase** ".



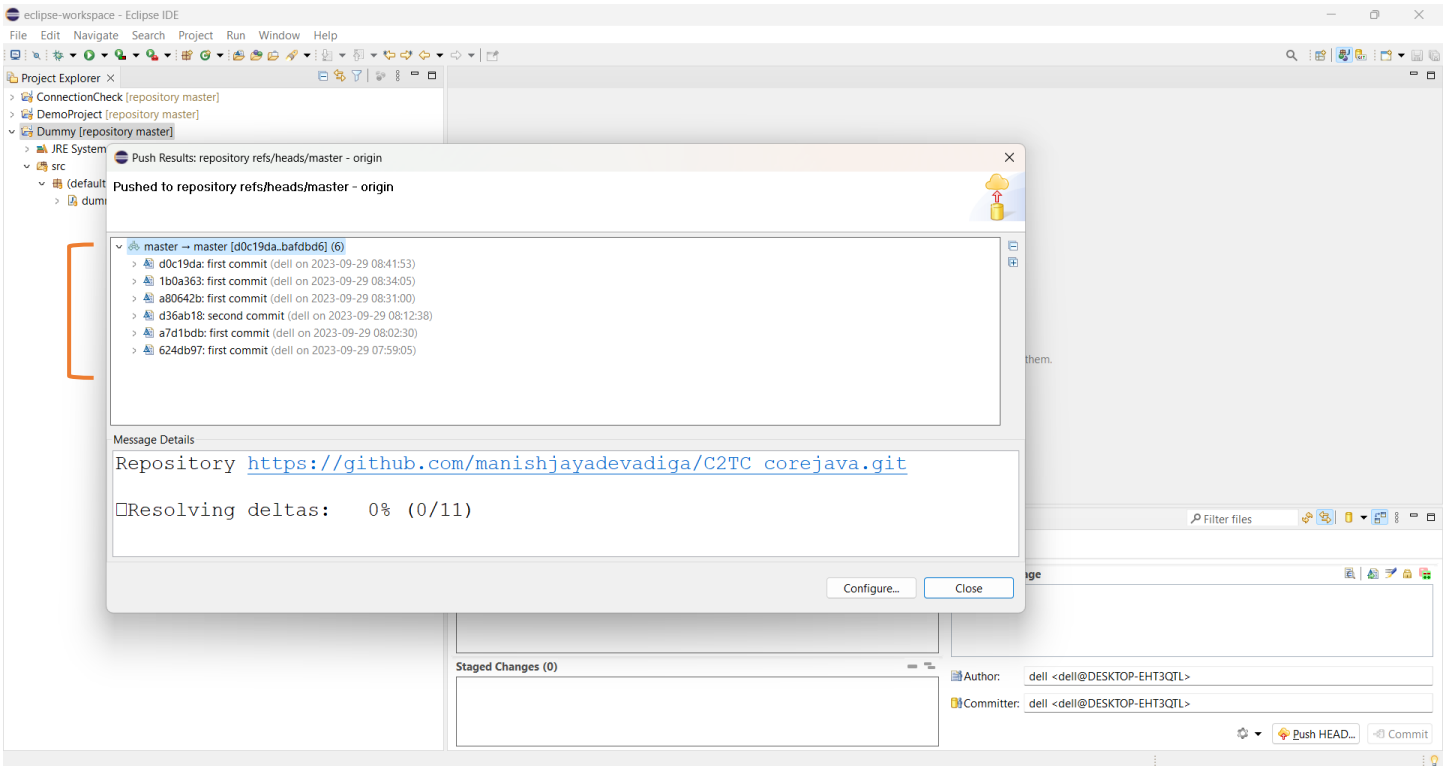
- Select "**Rebase interactively**" to avoid conflict issues.
- Click "**Rebase**".



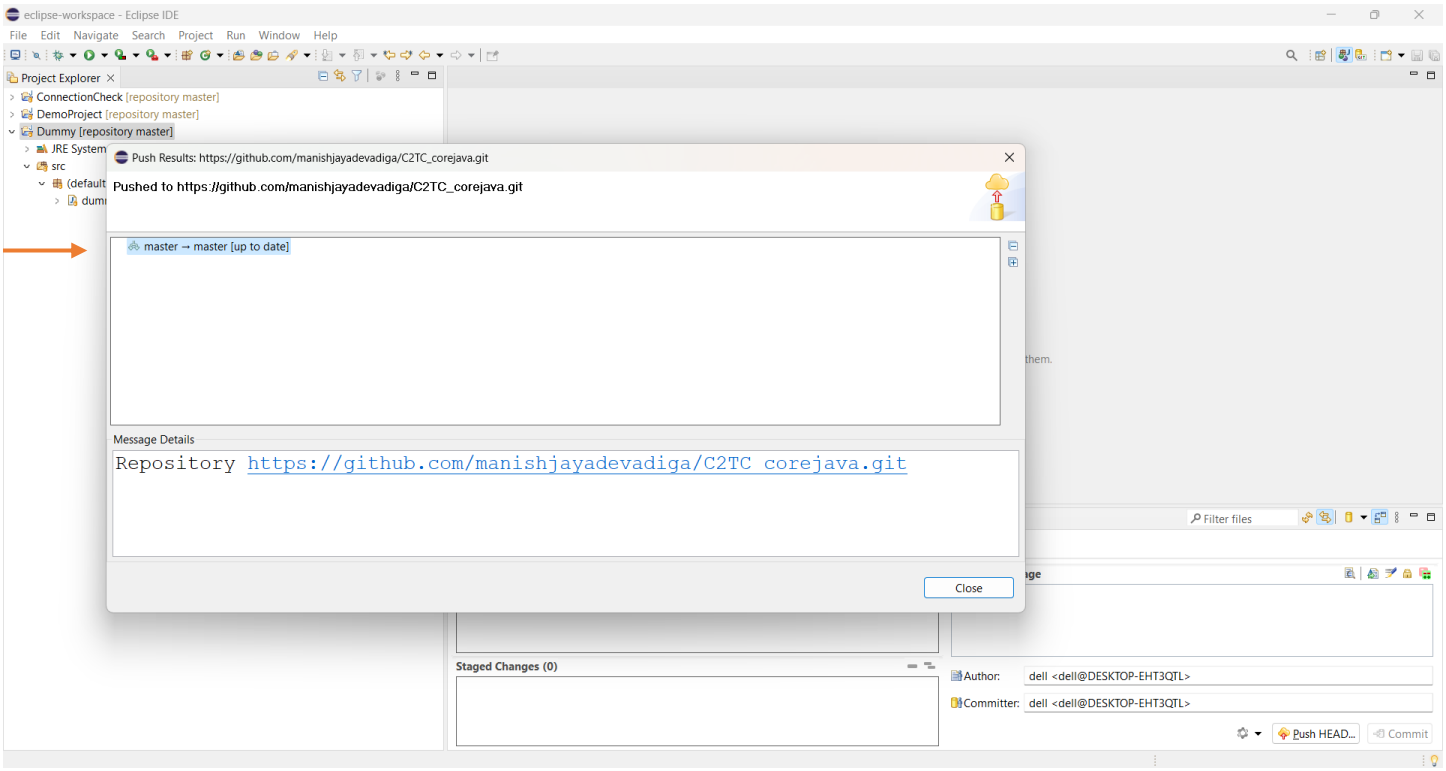
- Now **you will see your files here** (mine is empty because I have solved my error). Now press "**Start Icon**".



- Now Rebase has been finished “**Successfully”.**
- Now **perform the same Push operations** which we have been doing to push to git hub.
- **You will see the below dialog box** after Push and Commit.

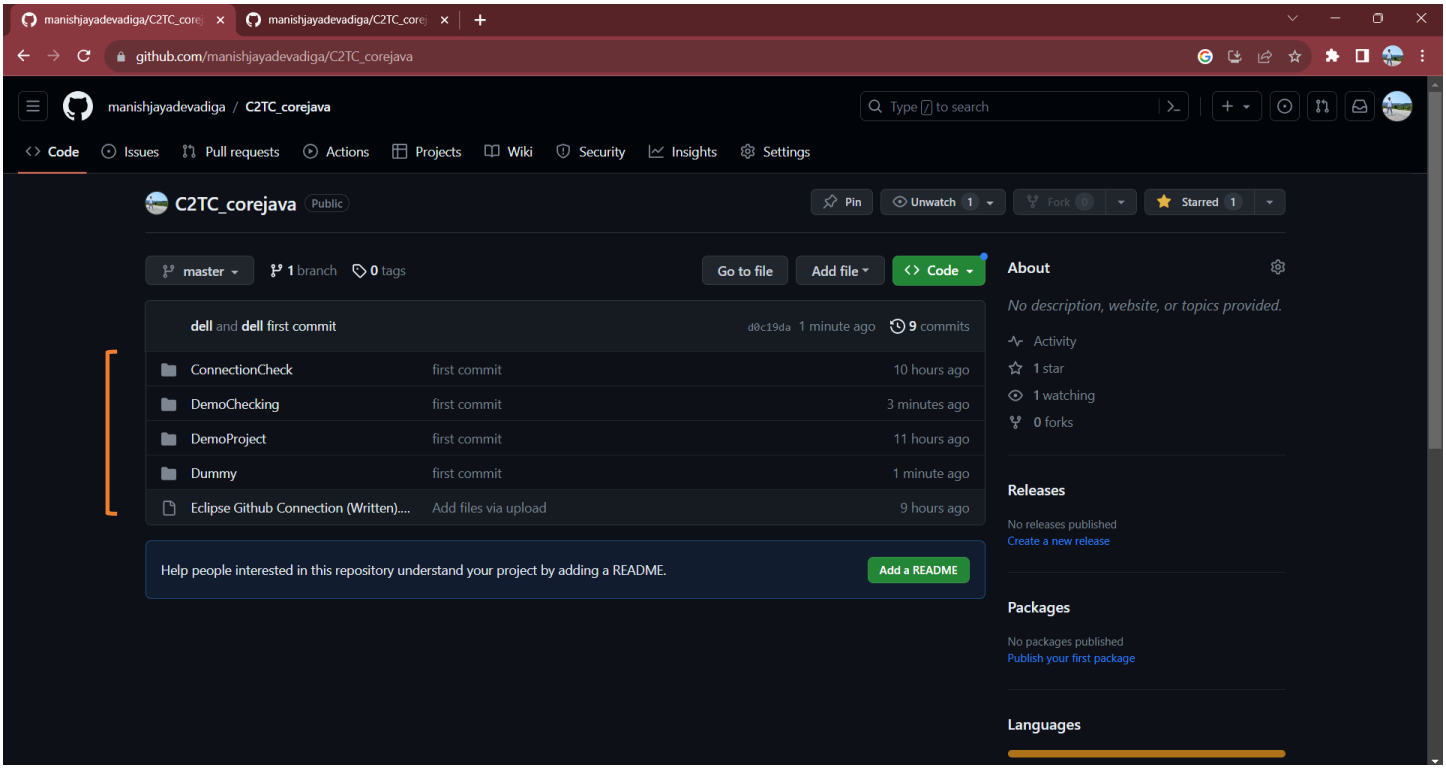


- Now “**Close**” this.
- Click “**PUSH_HEAD**”. Here **we can see the branch is up to date.**
- We have successfully solved the error. Now we can easily push our codes to GitHub.



GitHub:

- Here we can see that all the files have been pushed to GitHub repository.

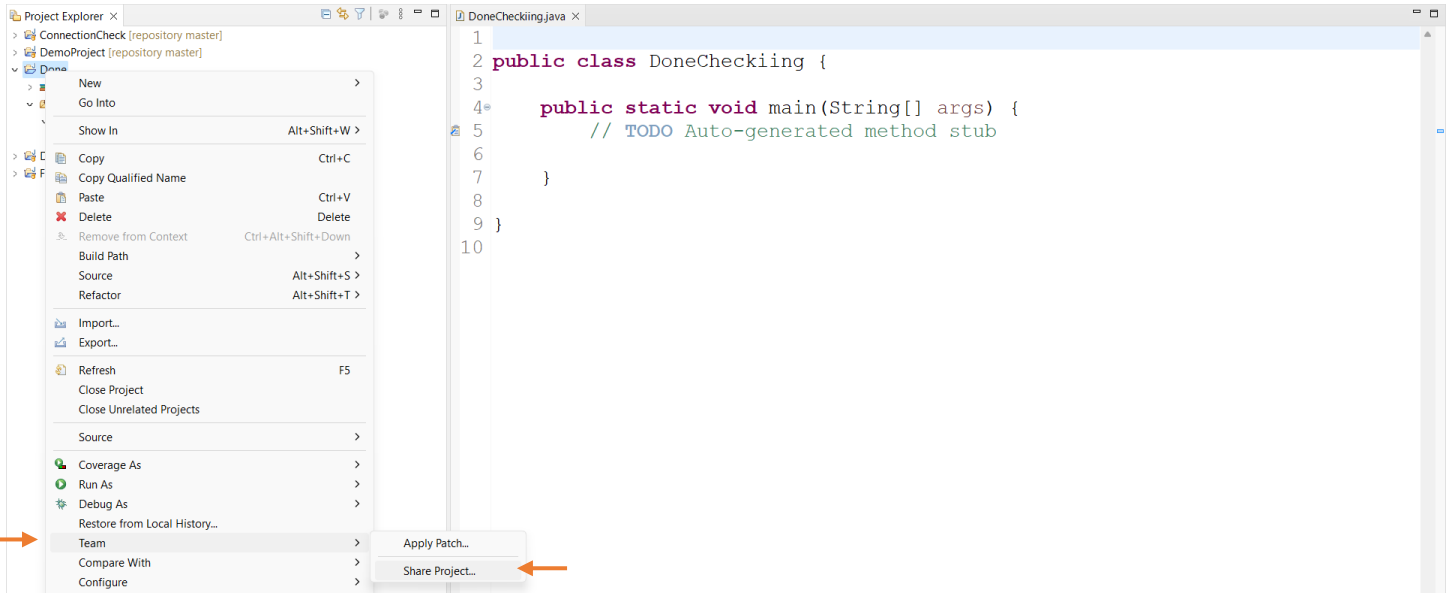


- Now "**Close Eclipse**".
- **Try to push file to GitHub** (To see if everything is working fine).

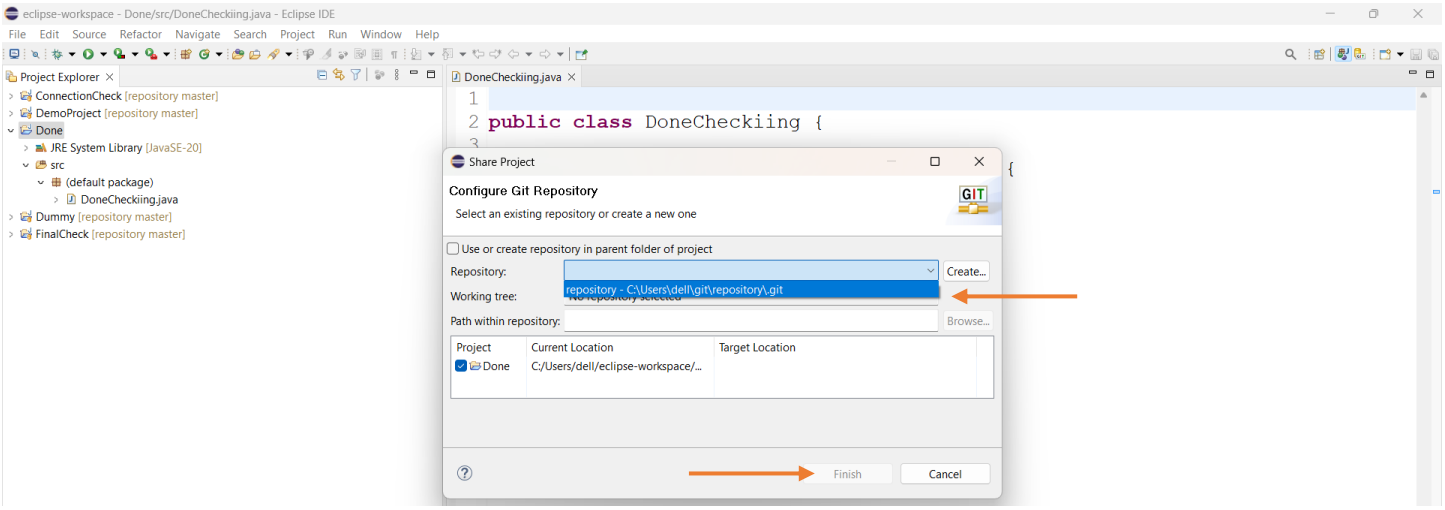
Final Check:

Eclipse:

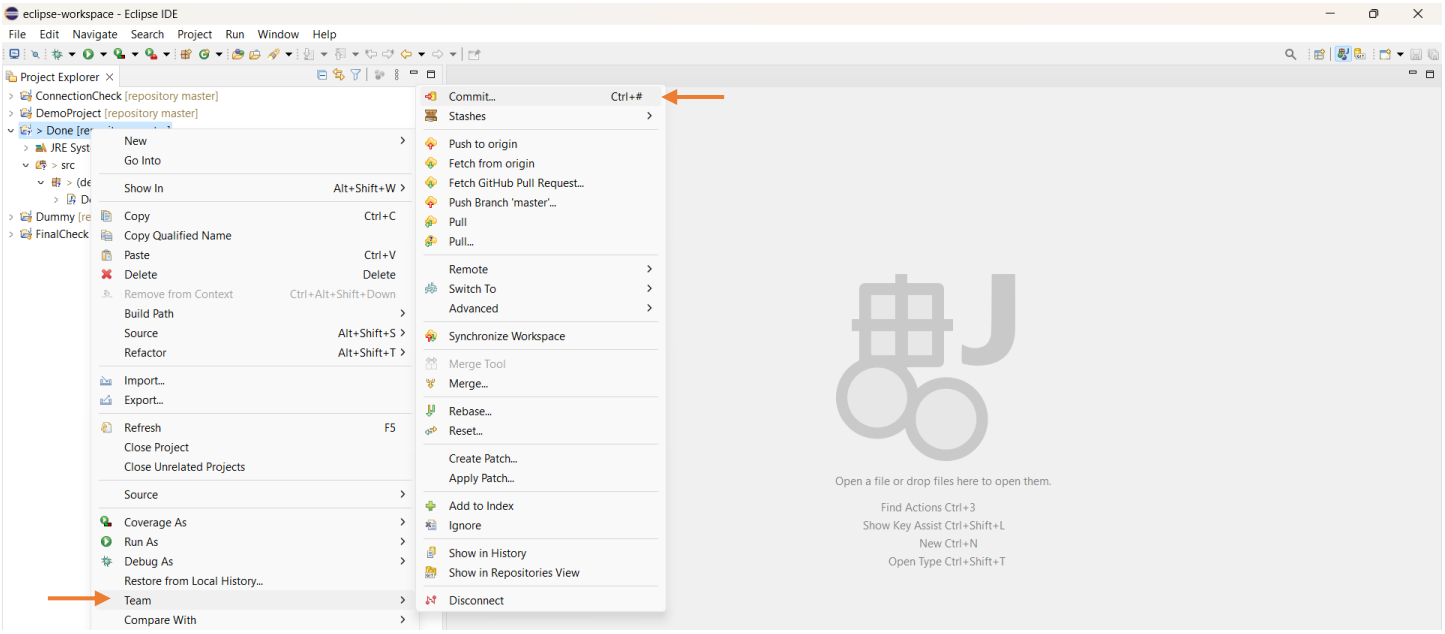
- Create a java project named "**Done**". Right click "**Done -> Team -> Share Project** ".



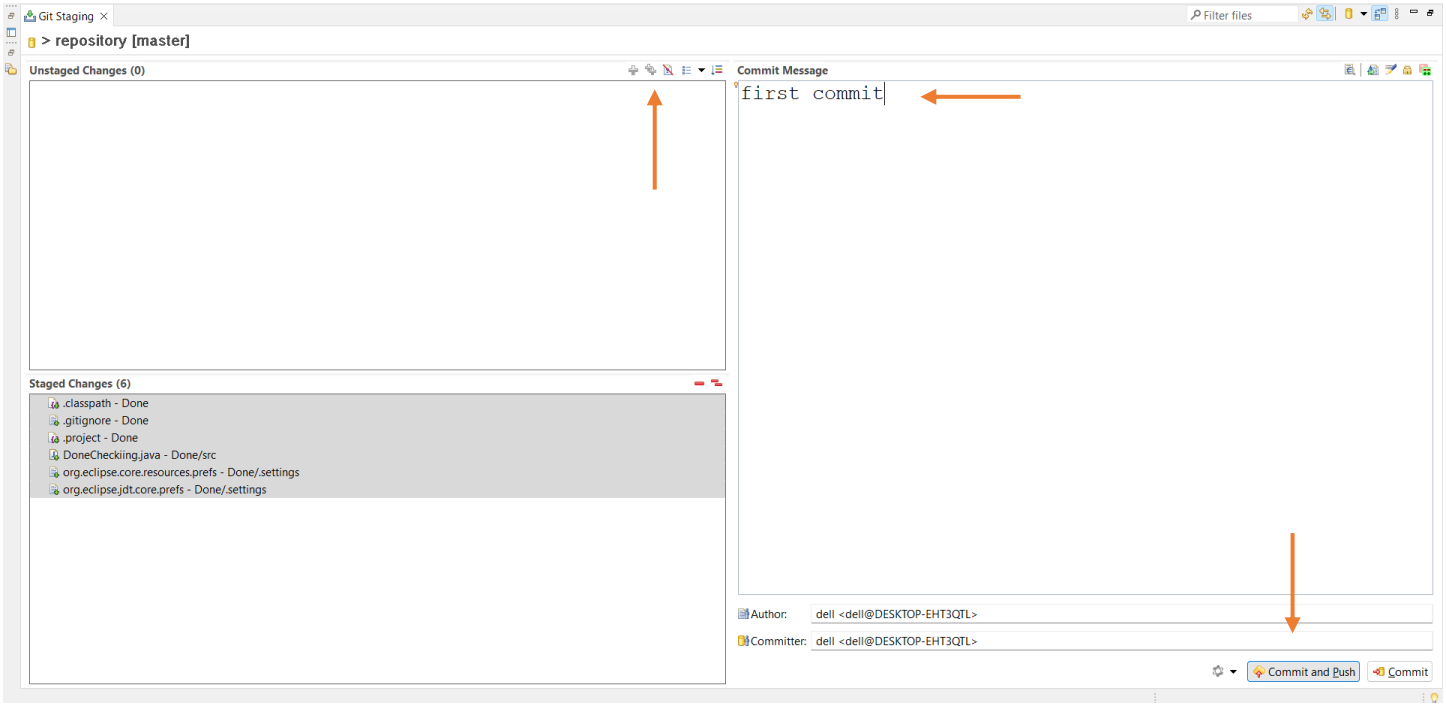
- **Select repository** from dropdown and click" **Finish**".



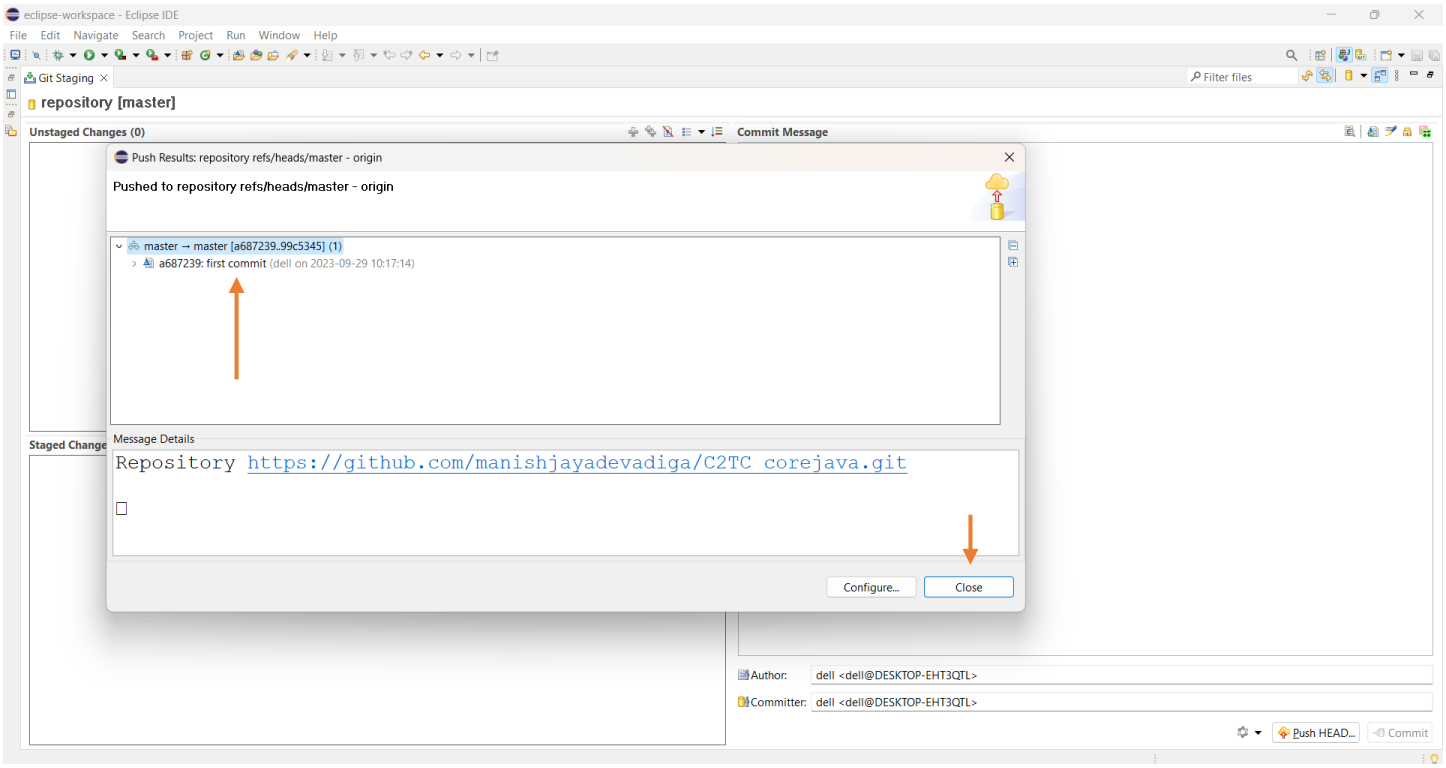
- Right click "**Done -> Team -> Commit** ".



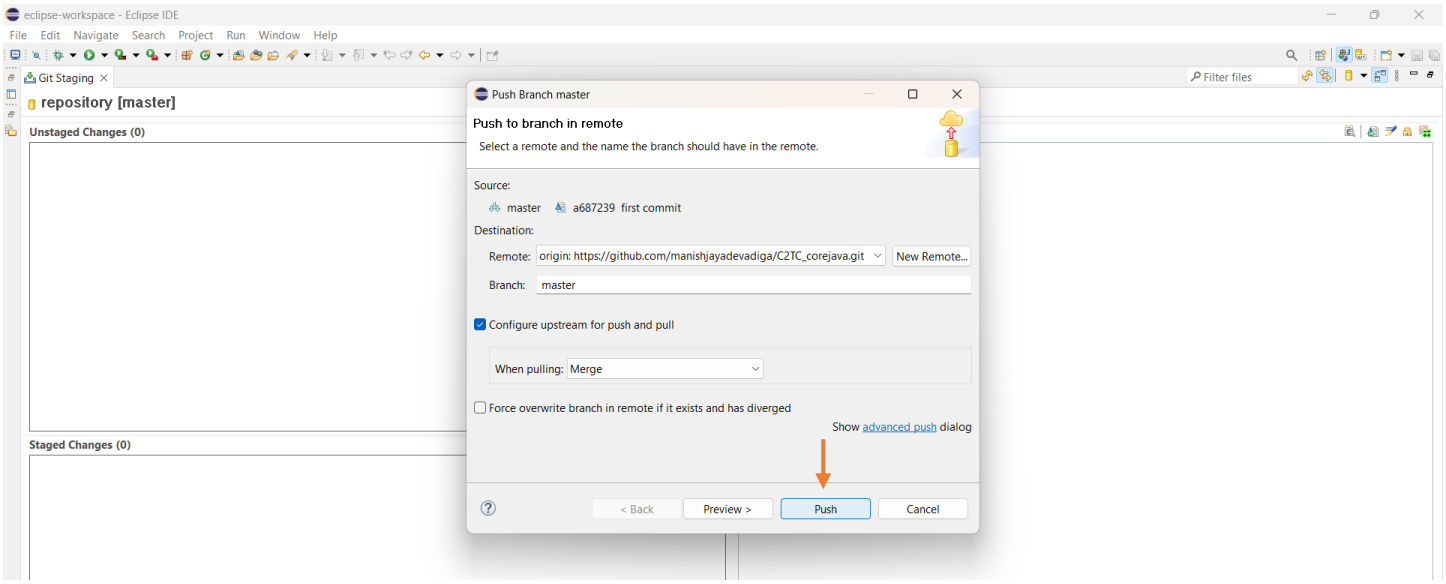
- Now Git-Stagging window will be opened.
- Press "**++ ->first commit**".
- Click on "**Commit and Push**".



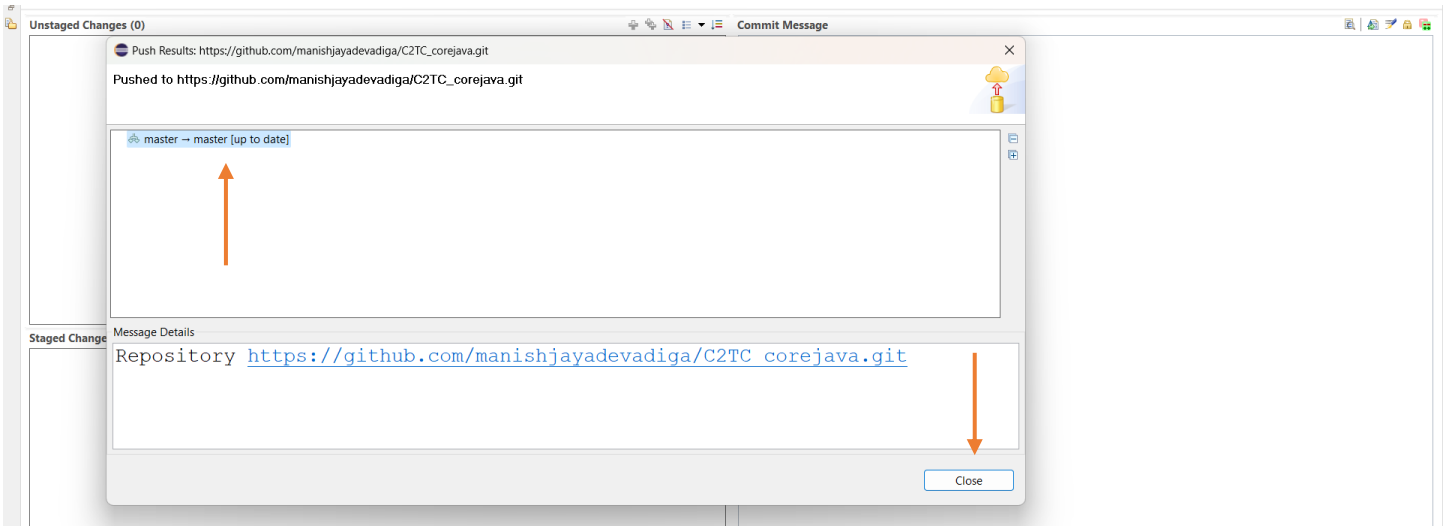
- We can see "**Pushed to repository**".
- We can see "**No error like: non-fast-forward**".
- Now "**Close**" it. And Click "**PUSH_HEAD**".



- Now click "**Push**".



- The **push is successful** and branch is up to date.
- Click "**Close**".



GitHub:

- Here we can see that Project "**Done**" have been pushed to GitHub repository successfully.