Experiment-2 (DevOps Lab)

Aim: Setups Eclipse for Devops

Note: Write the given steps in your lab manual. The provided image is just for your better understanding.

Step-1: Install Jdk-17 and set the java path in System environment

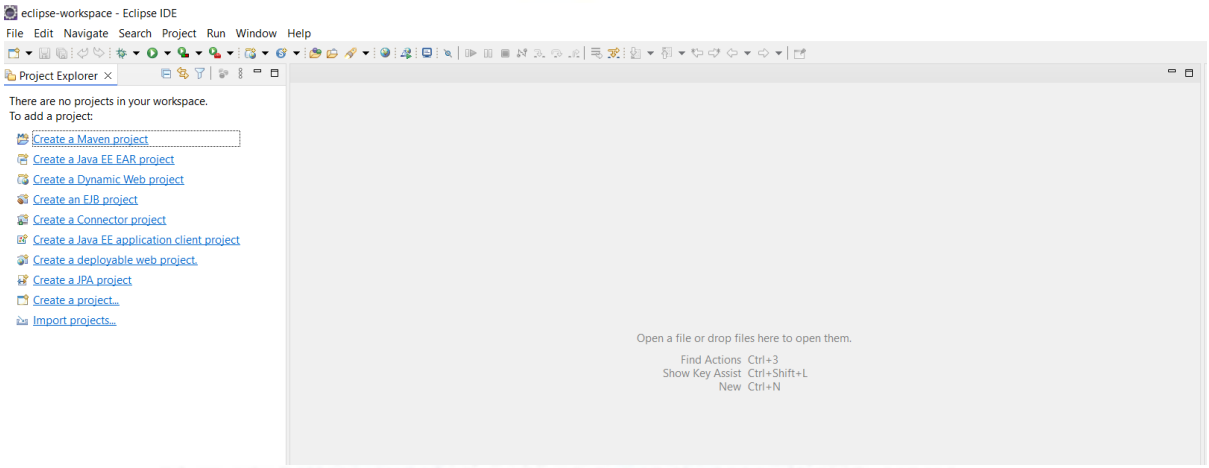
Step-2: Download eclipse zip file and extract the contents the all eclipse file

Step-3: Create a Maven Project from eclipse as:

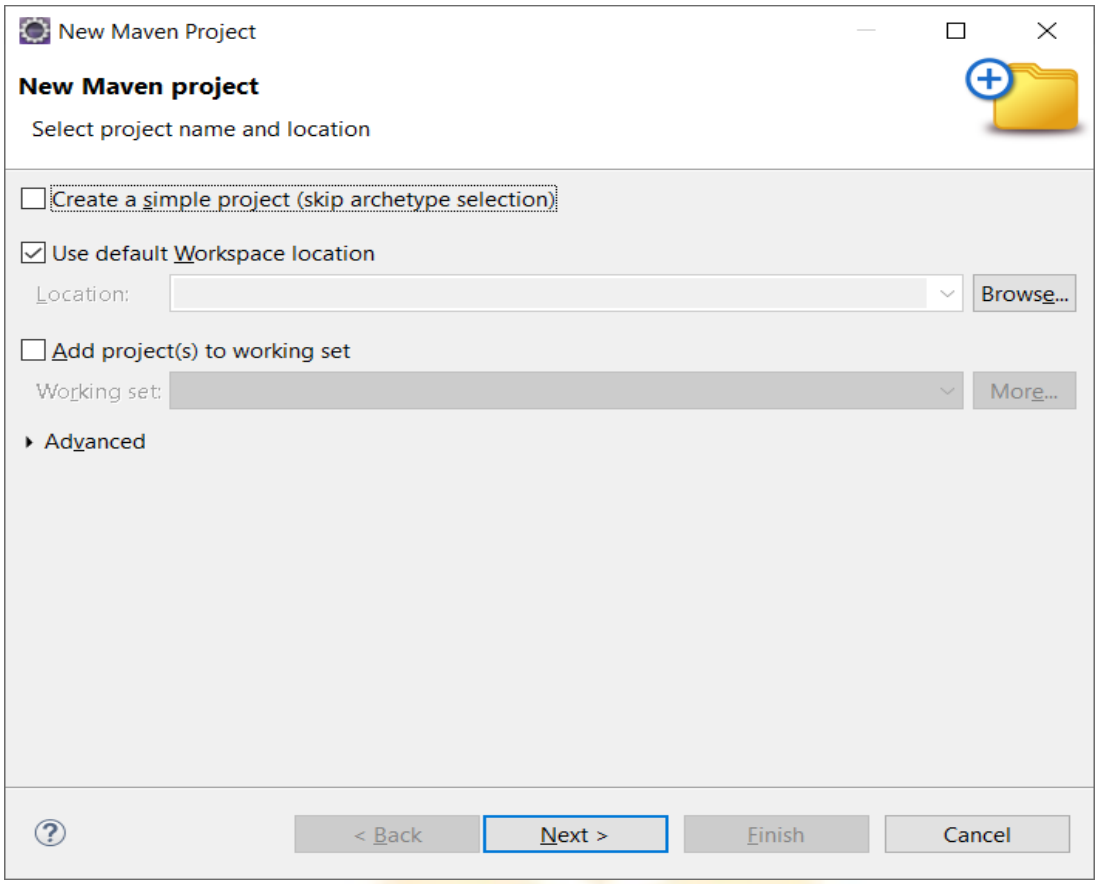
OR

Click on File in left corner -> Click on new -> click on Maven Project and follow the given image steps.

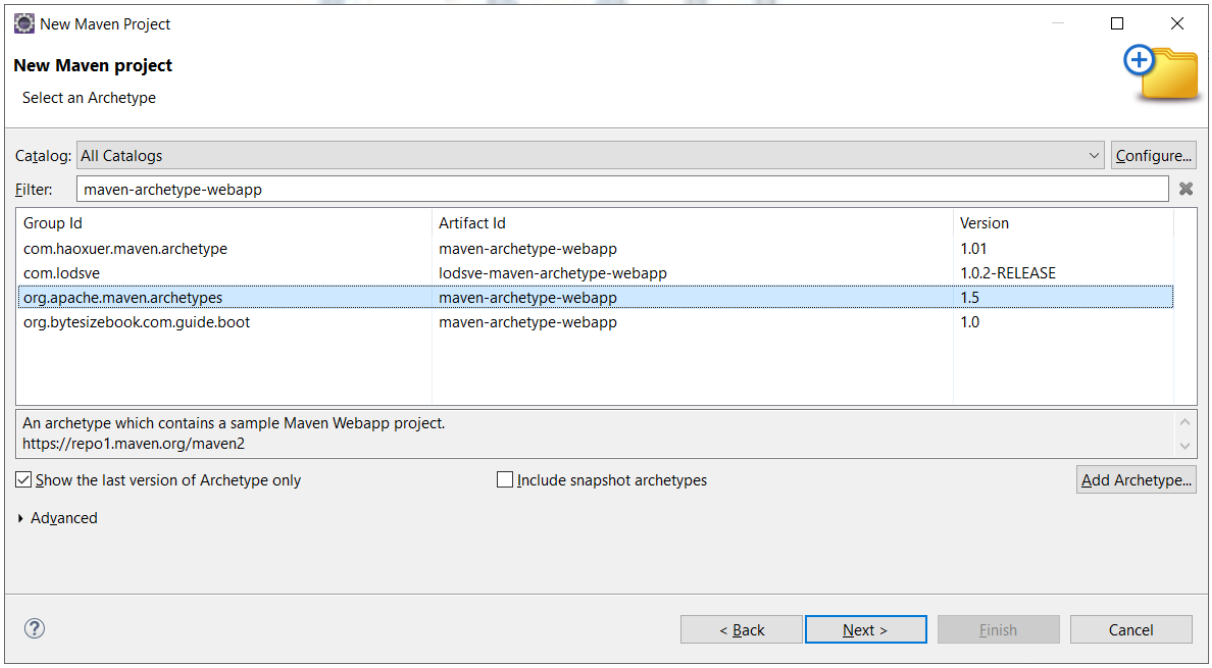
a.



b.

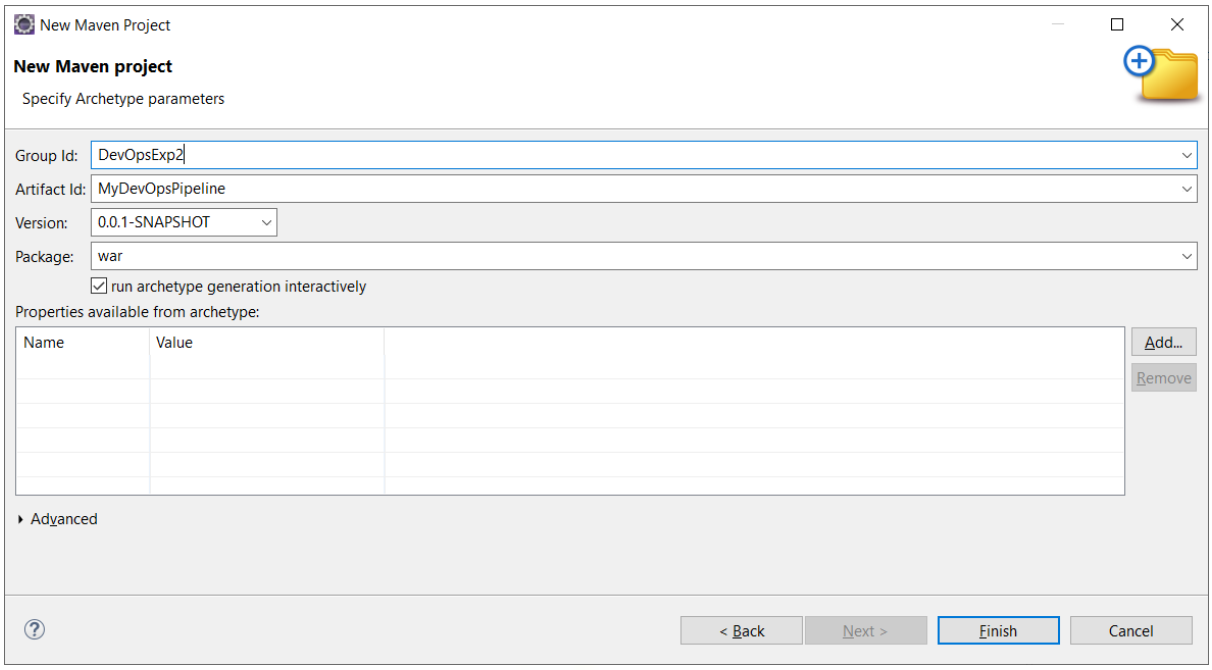


c. Click Next and Search **org.apache.maven.archetypes** and select webapp file



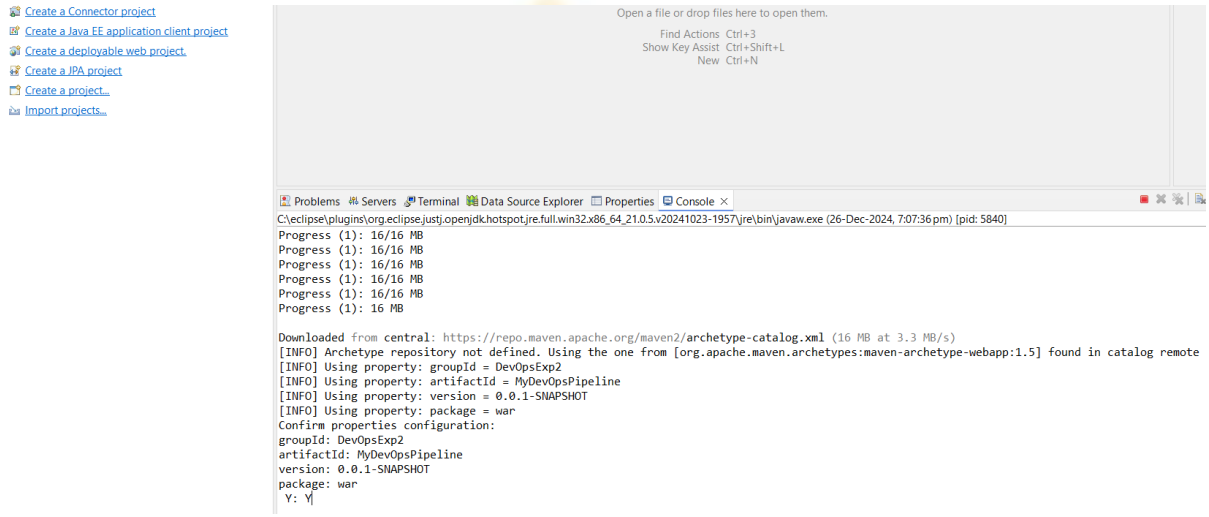
d. In a **group id** you can type anything like name and in a **artefact id:** you can type anything like your roll number

e.

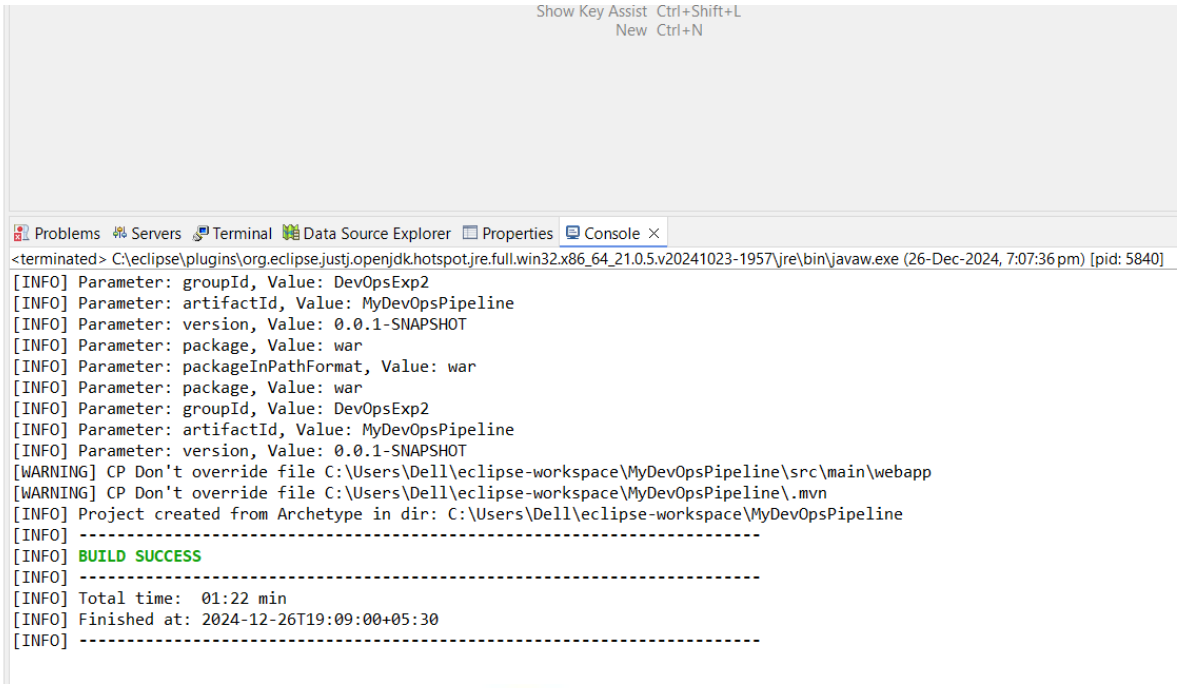


f. Click Finish

g.

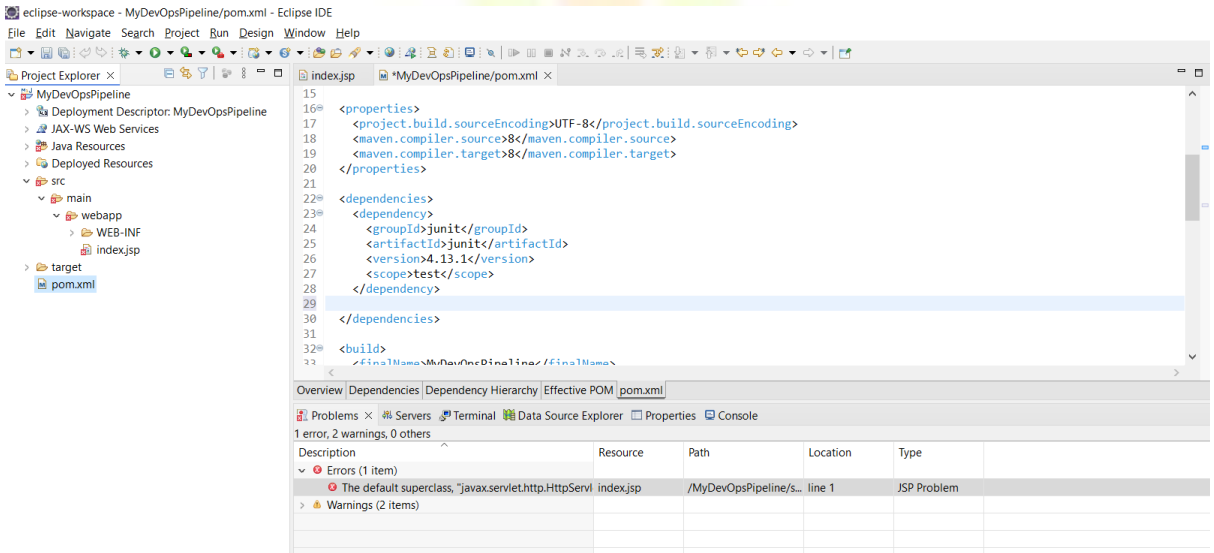


h. Type Y and Press enter, You should see a Build Success message as below:



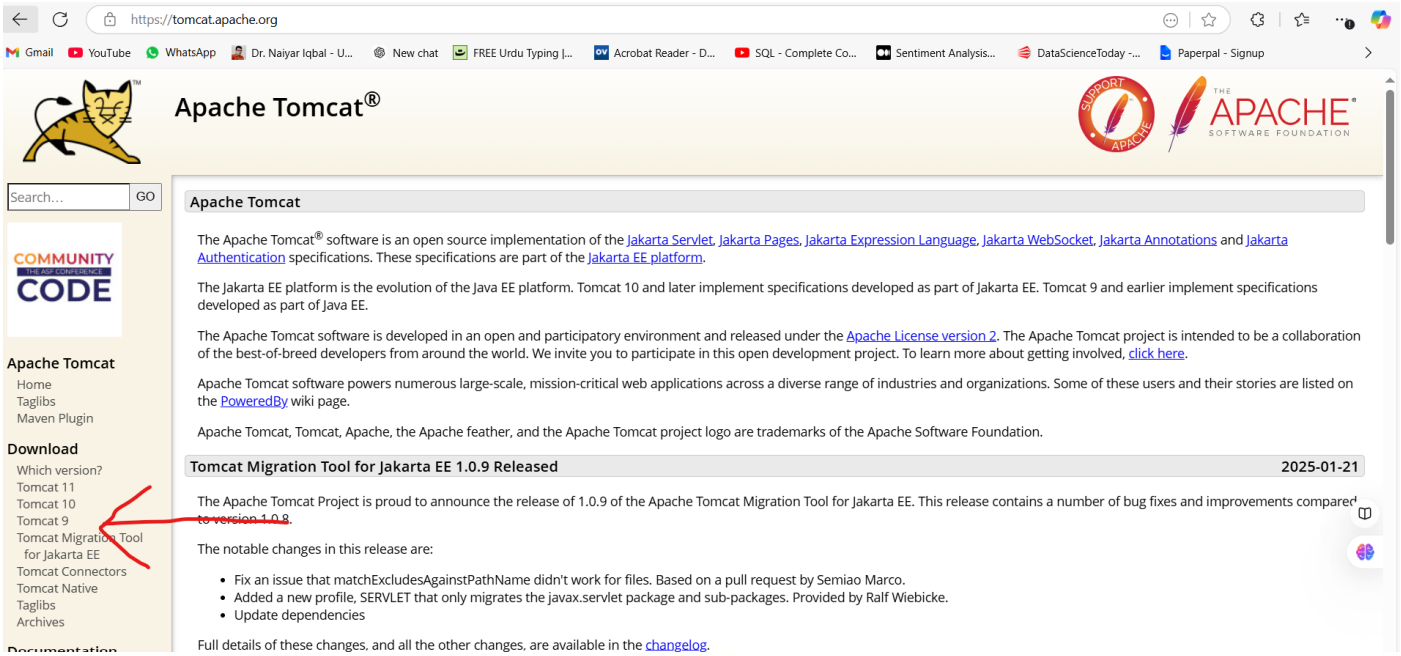
i.

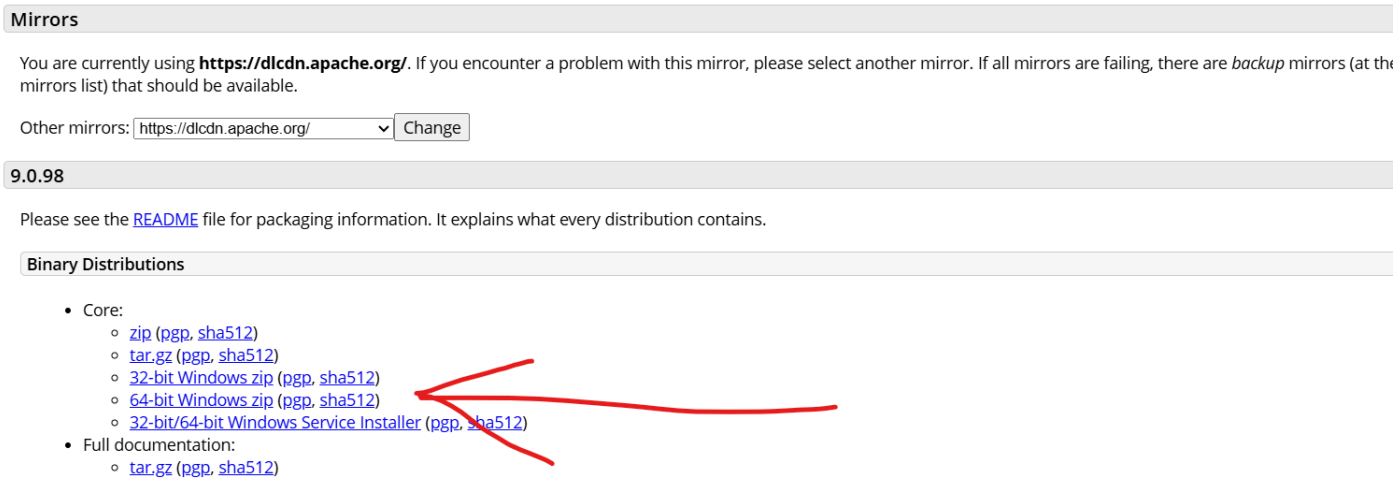
Step-4: now open your **pom.xml** file and add your **dependencies (Given file, Copy and Paste)**



Step-5: Update your project once (Right click on Project -> click on Maven -> click on Update Project)

Step-6: Download **Apache tomcat v9** from Official website.

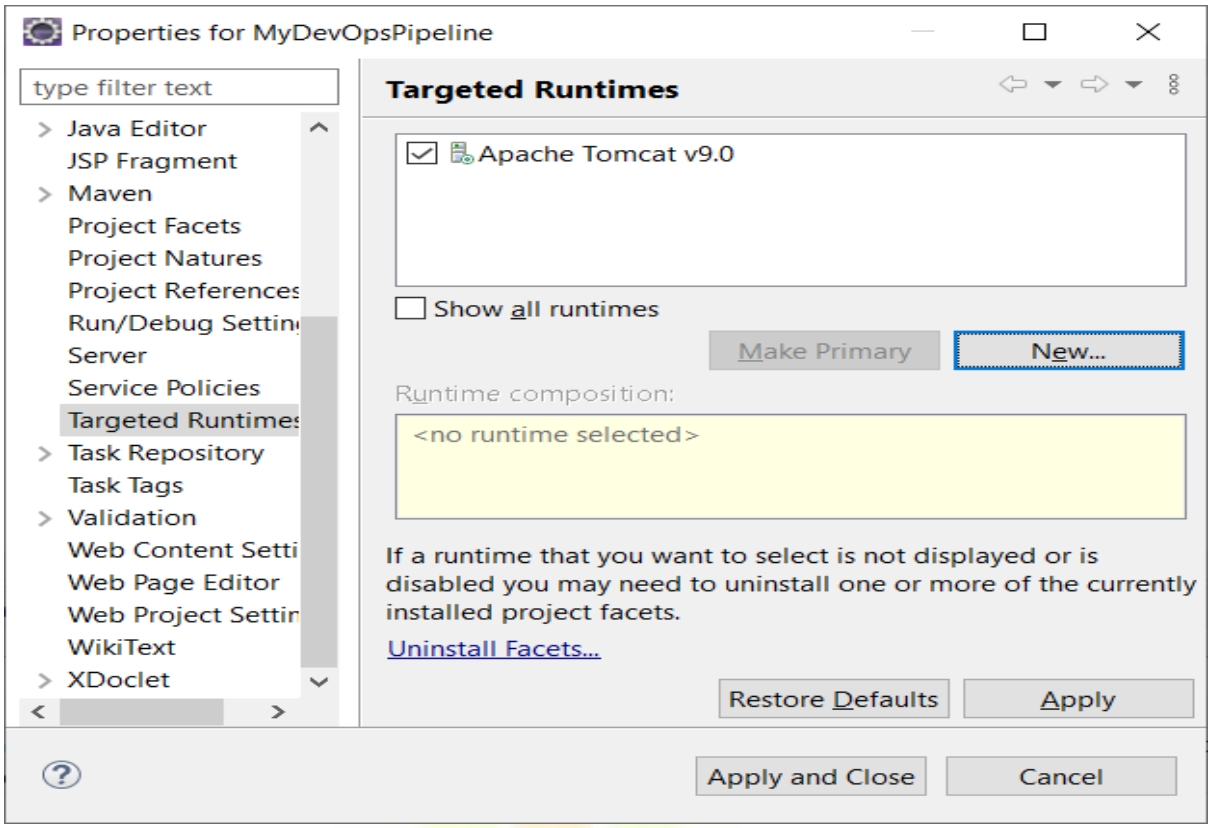




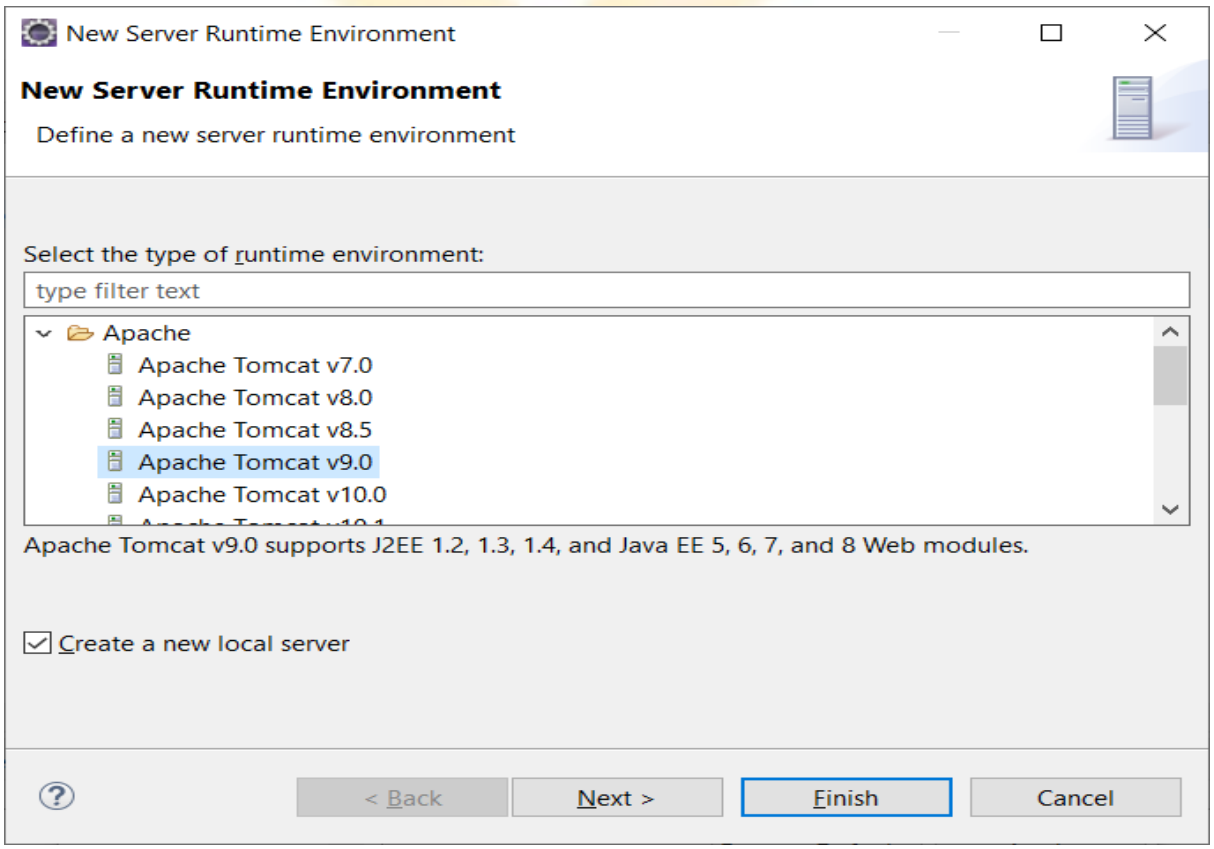
Step-7: After Download the Apache tomcat, Extract the .zip file and paste your apache-tomcat-9.0.98 folder in your folder

Step-8: Now click on your **project option in Menu -> Click on Properties -> Click on Targeted Runtime**

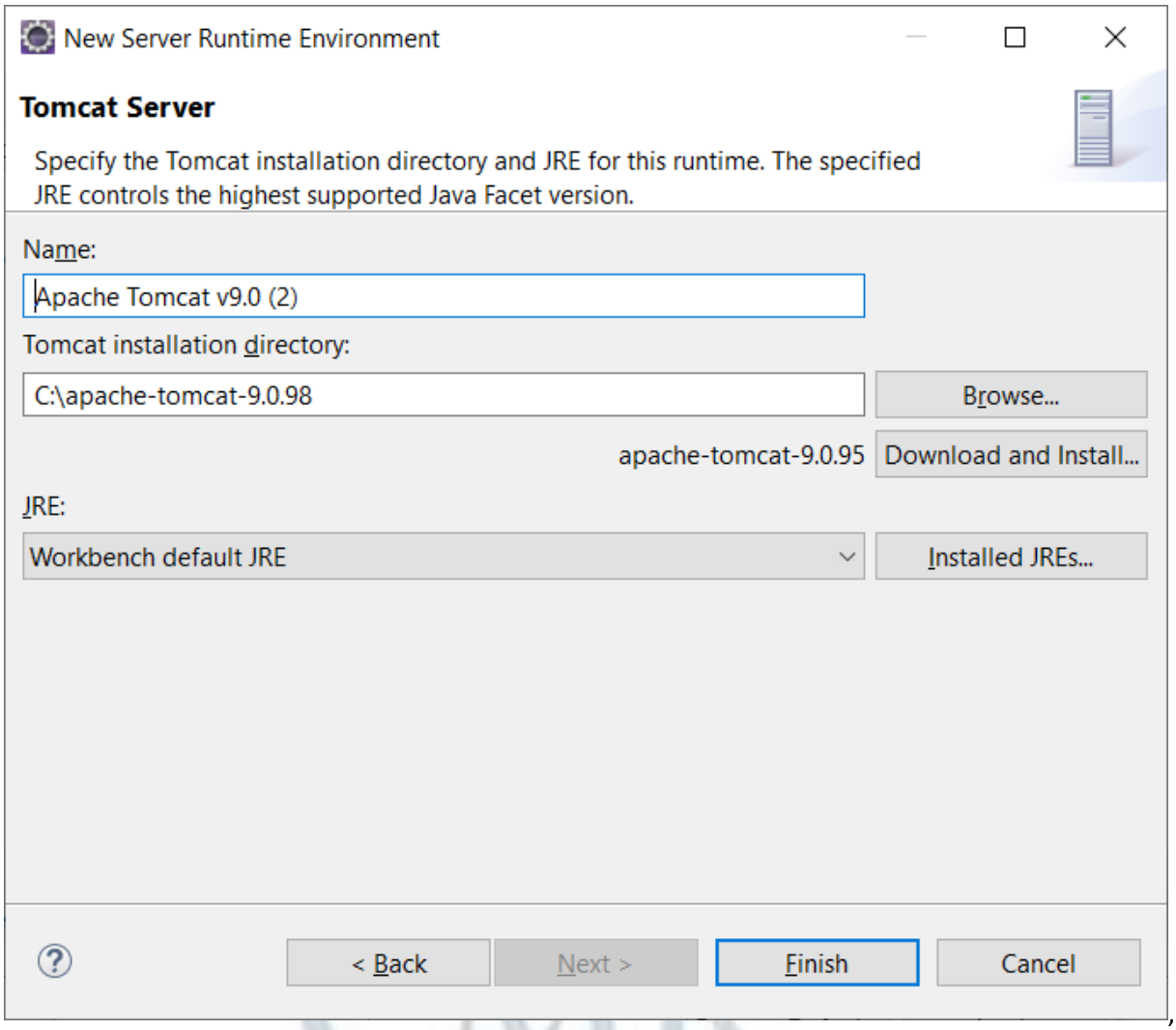
Step-9: Click on **new or follow the given image**



Step-10: Select Apache Tomcat v9.0



Step-11: Click on Browse and Select your Extracted file and then click on finish as given image

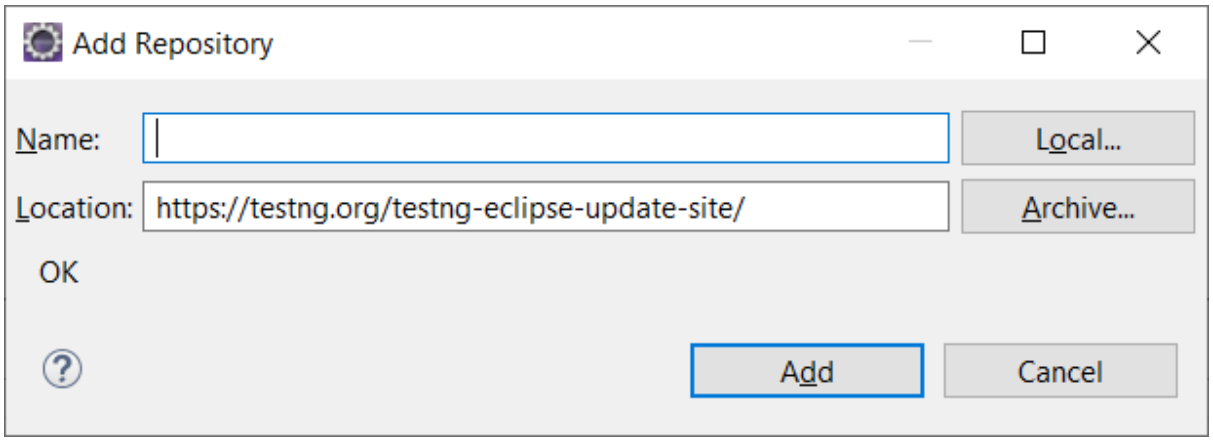


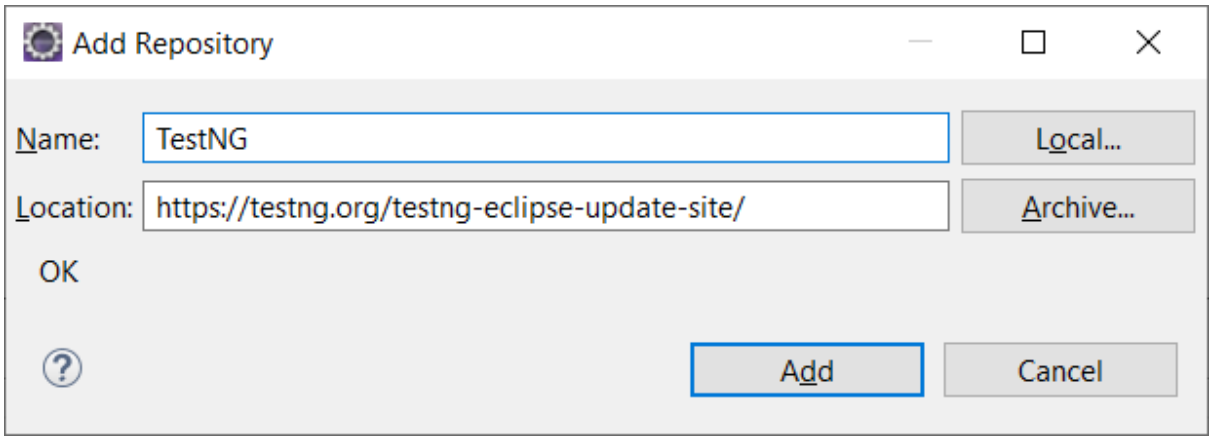
Step-12: Now Click on **help Menu ->** click on **Install new Software.**

Step-13: Click on **Add** and it will show a **popup dialog box** like given image

In the place of Name type: **TestNG**

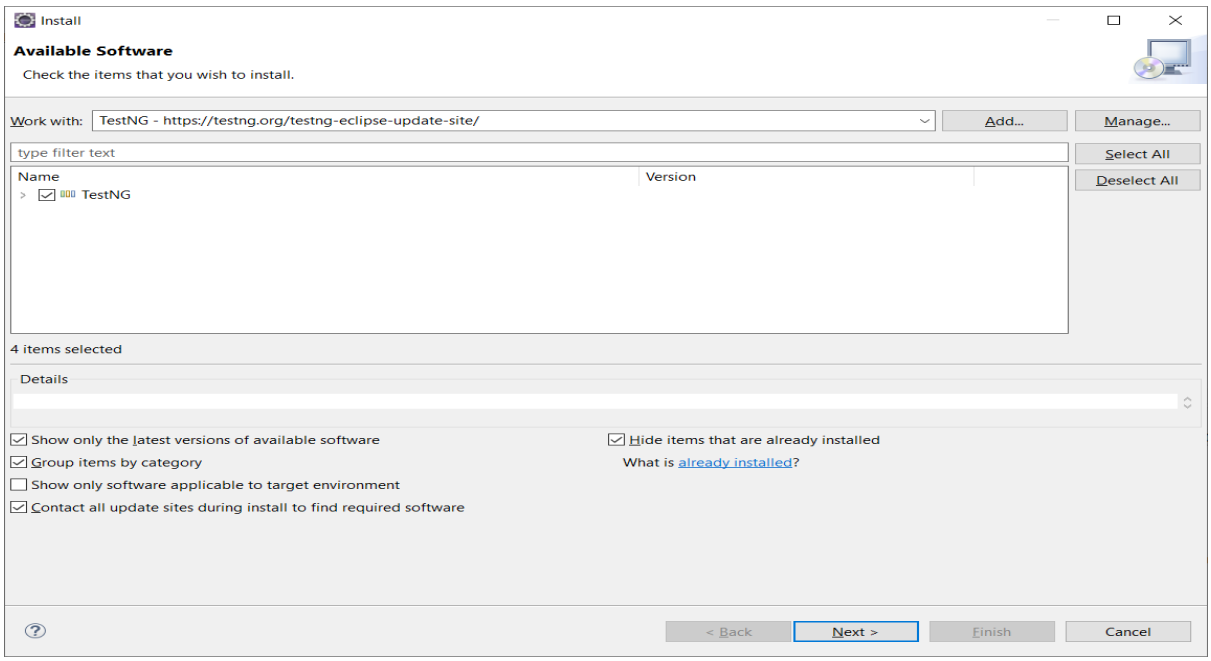
In the place of Location type: **https://testng.org/testng-eclipse-update-site/**

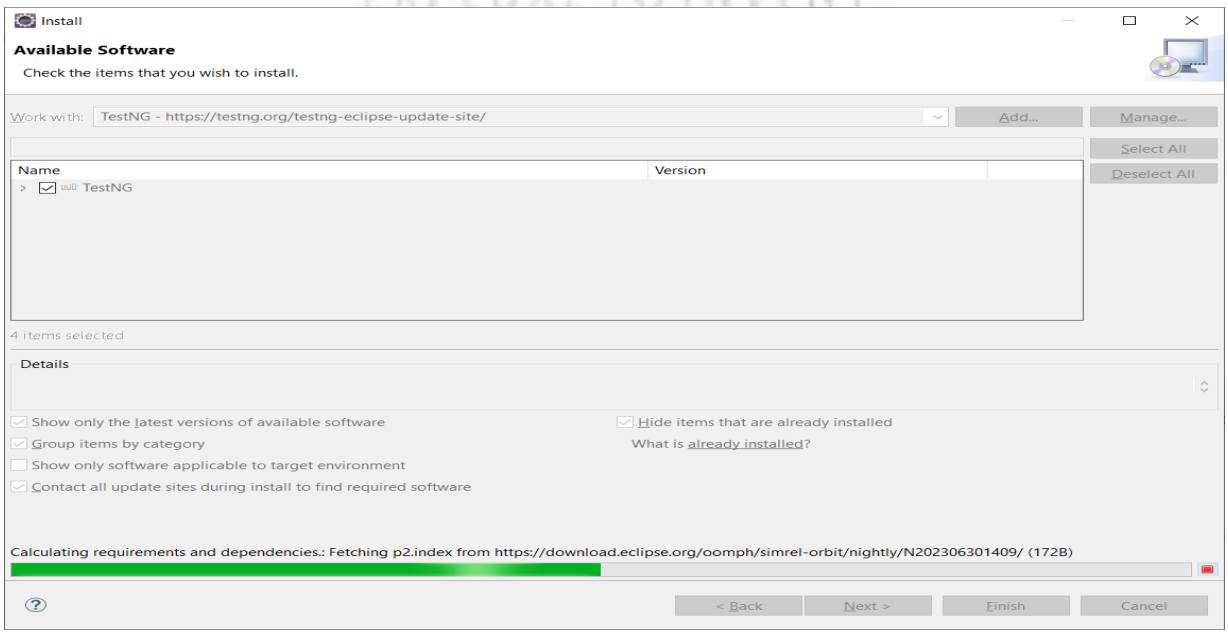




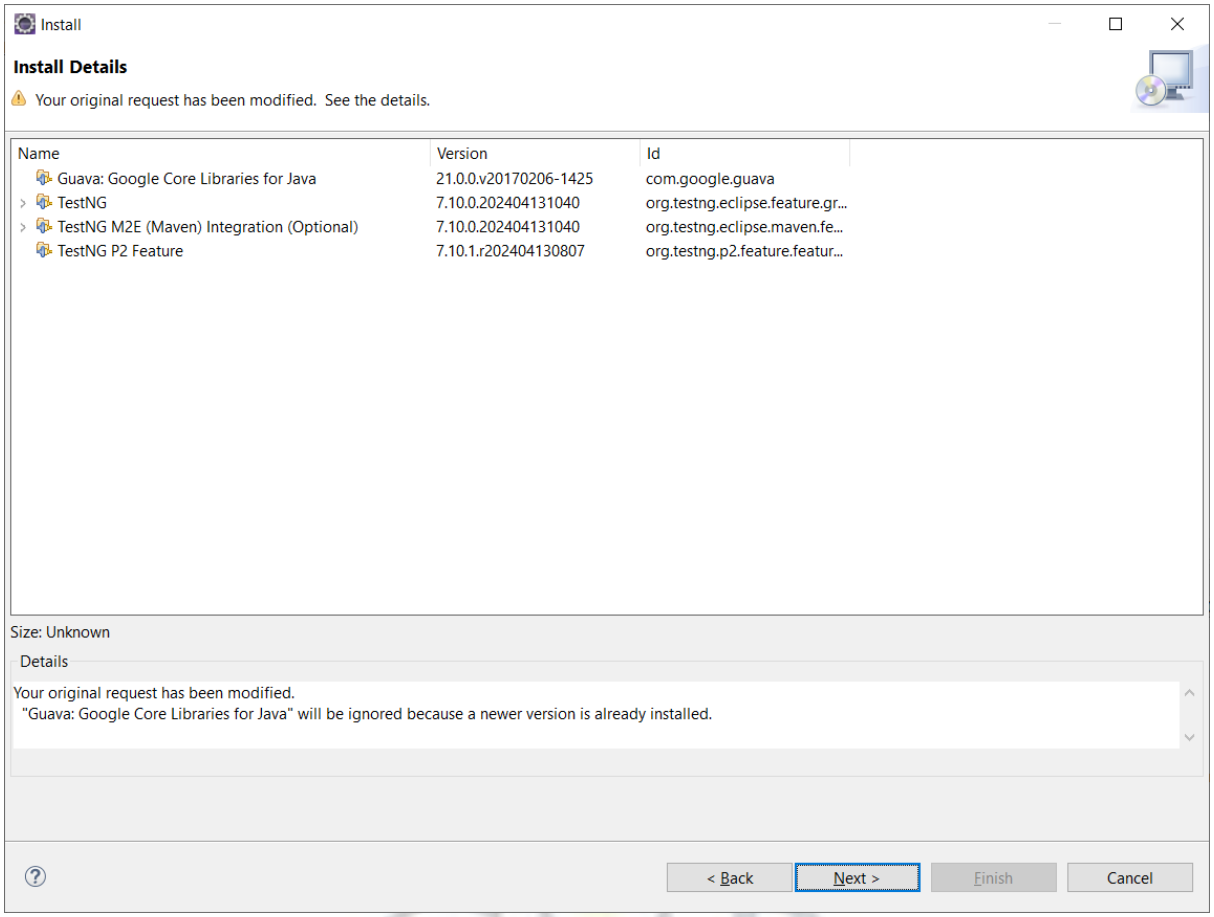
Step-14: Click on **Add ->** It will load a **testNG Dependencies ->** Select **TestNg** like given Image and then click **Next.**

It will take 10 minute to update TestNG in our Project

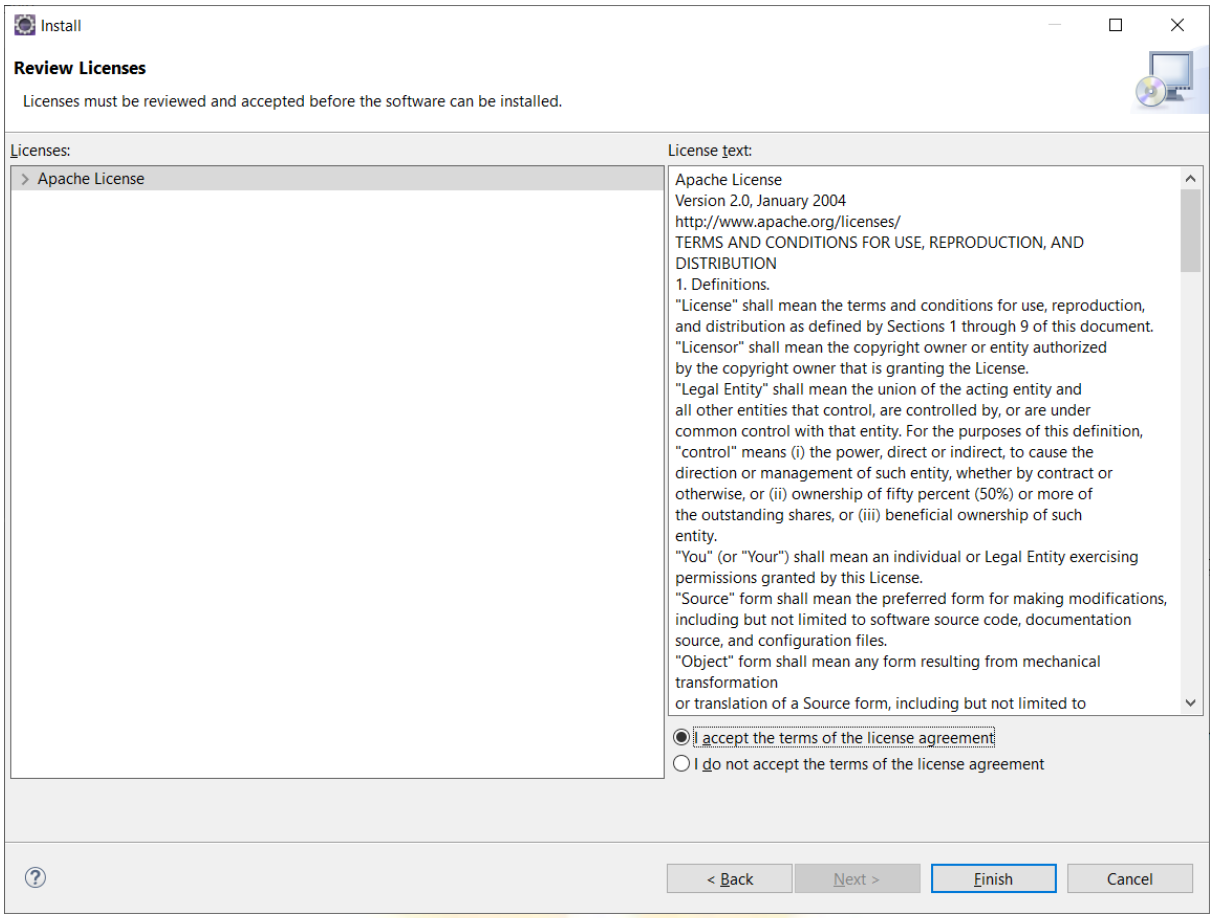




Step-15: After downloading the all dependencies it will show some file select all and click on next.



Step-16: Accept Terms and condition and click on **finish**

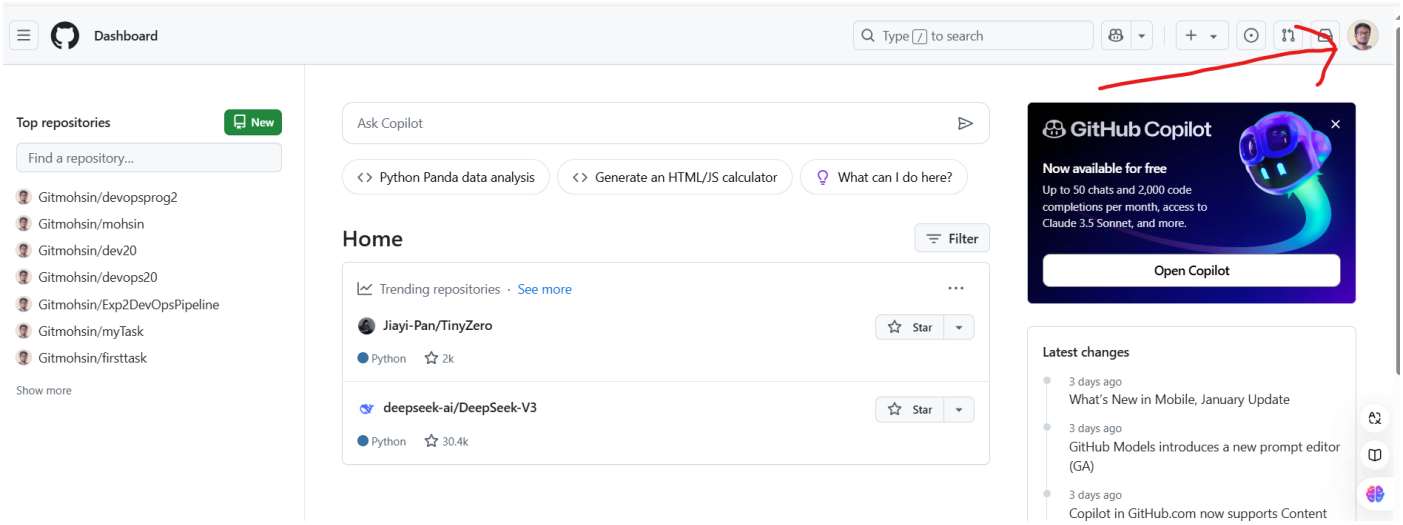


Step-17: After finish it will show **restart option (Restart the Project**) otherwise just **update** once of your project.

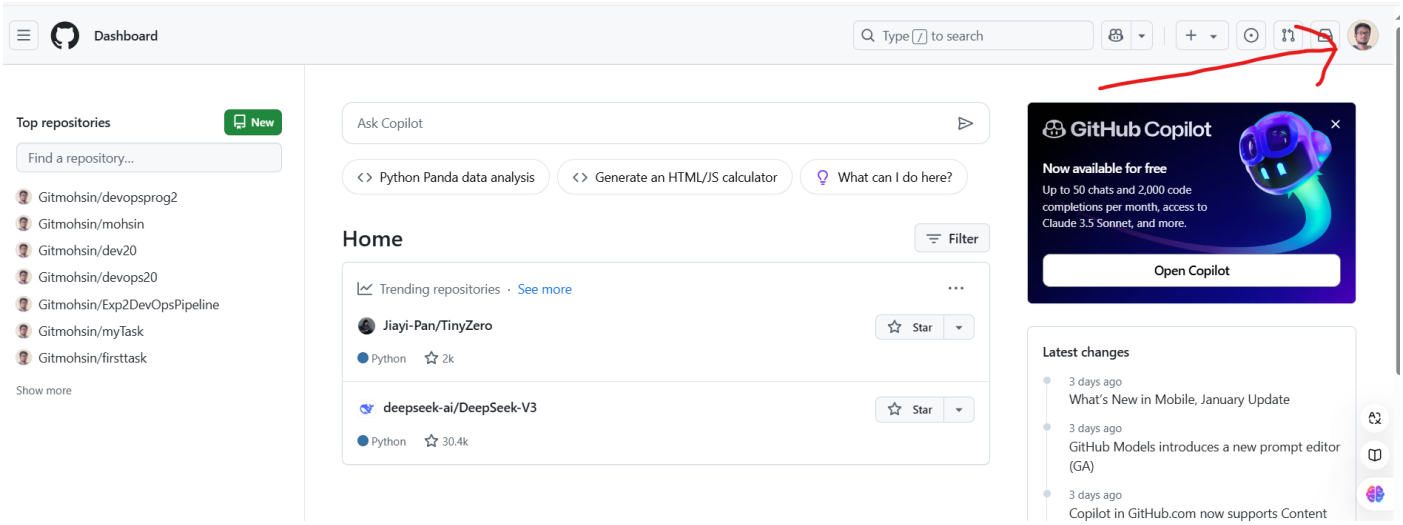
Step-18: Now Login your **GitHub Account.**

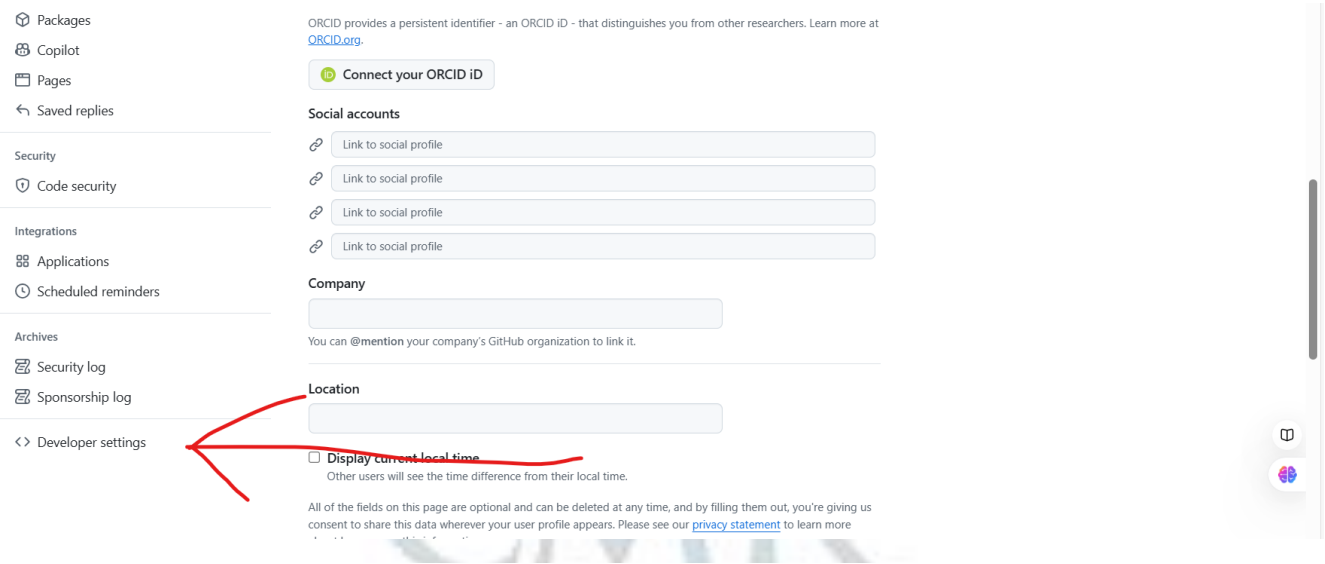
Step-19: Create a New **Repository** and Copy your **Repository** and paste in notepad

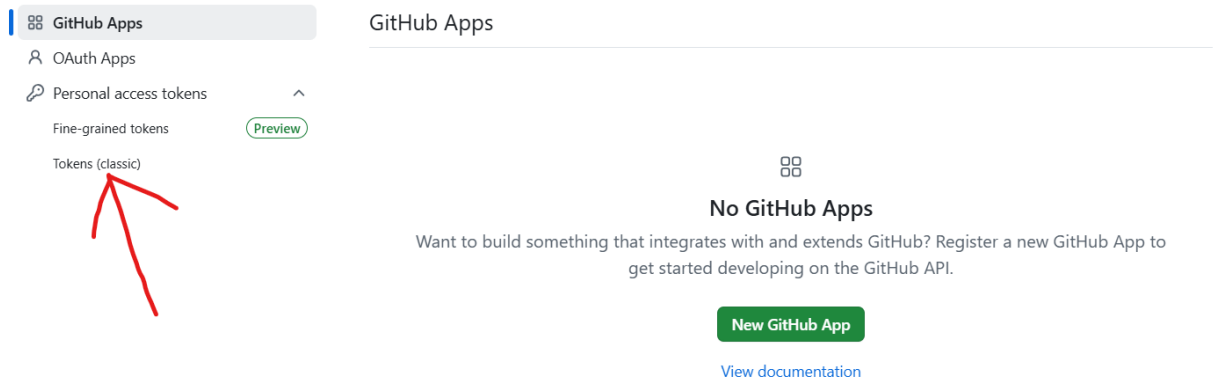
Step-20: After that **Click** on your **Profile** in Right corner **-> Click on Setting.**

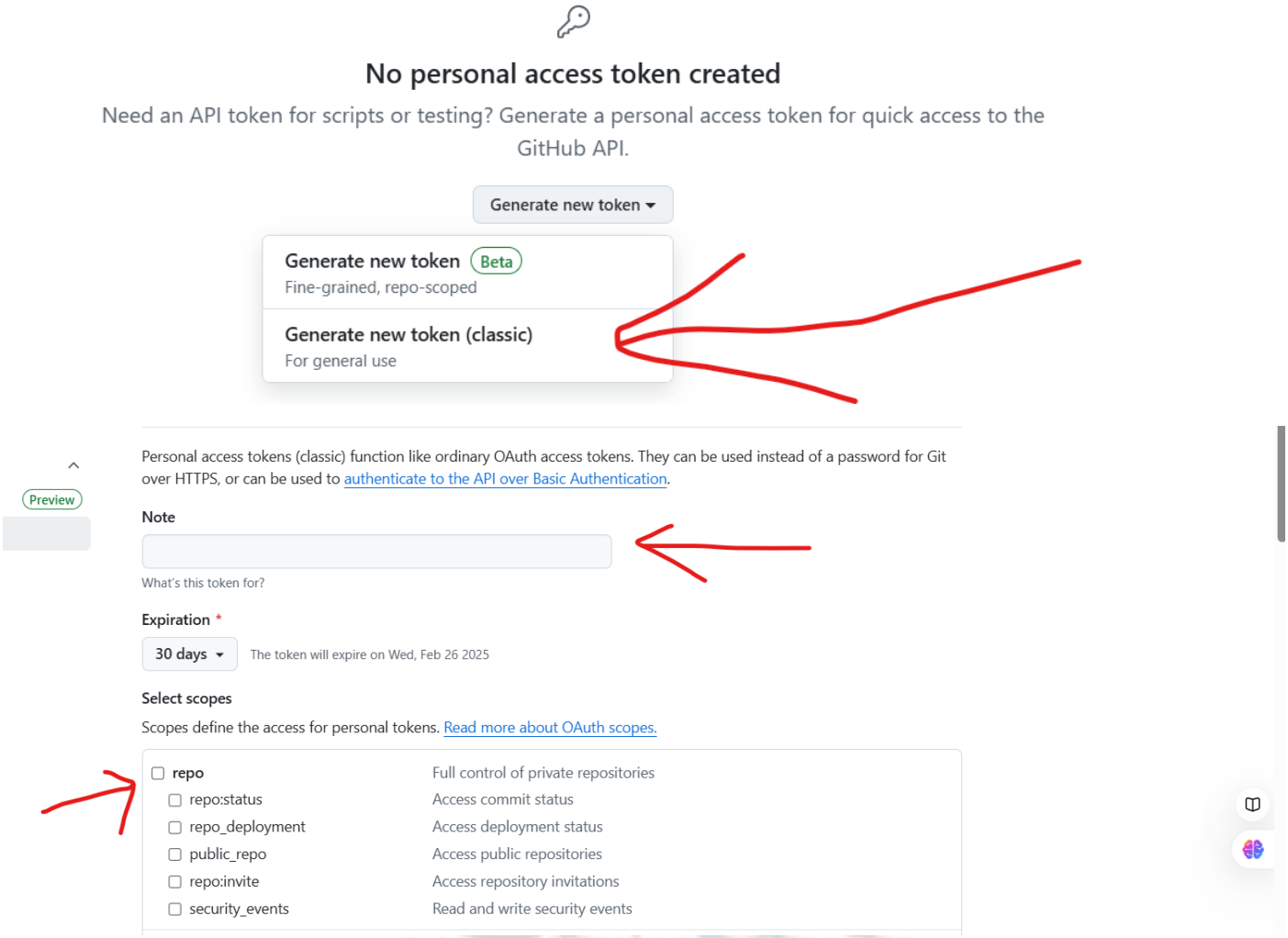


Step-21: It will show a new page, scroll down and select the **developer setting ->** click on **personal access token ->** select **Token(Classic) ->** click on **Generate new token** and select **Generate new token(Classic) ->** write your **token name** and select **repo option** and scroll down and click on **Generate Token.** (Follow the given Image)


Dashboard
Type / to search
Top repositories
New
Find a repository...
Gitmohsin/devopsprog2
Gitmohsin/mohsin
Gitmohsin/dev20
Gitmohsin/devops20
Gitmohsin/Exp2DevOpsPipeline
Gitmohsin/myTask
Gitmohsin/firsttask
Show more
Ask Copilot
Python Panda data analysis
Generate an HTML/JS calculator
What can I do here?
Home
Filter
Trending repositories · See more
Jiayi-Pan/TinyZero
Star
Python
2k
deepseek-ai/DeepSeek-V3
Star
Python
30.4k
GitHub Copilot
Now available for free
Up to 50 chats and 2,000 code completions per month, access to Claude 3.5 Sonnet, and more.
Open Copilot
Latest changes
3 days ago
What's New in Mobile, January Update
3 days ago
GitHub Models introduces a new prompt editor (GA)
3 days ago
Copilot in GitHub.com now supports Content


Packages
Copilot
Pages
Saved replies
Security
Code security
Integrations
Applications
Scheduled reminders
Archives
Security log
Sponsorship log
Developer settings
ORCID provides a persistent identifier - an ORCID iD - that distinguishes you from other researchers. Learn more at ORCID.org.
Connect your ORCID iD
Social accounts
Link to social profile
Link to social profile
Link to social profile
Link to social profile
Company
You can @mention your company's GitHub organization to link it.
Location
Display current local time
Other users will see the time difference from their local time.
All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our privacy statement to learn more

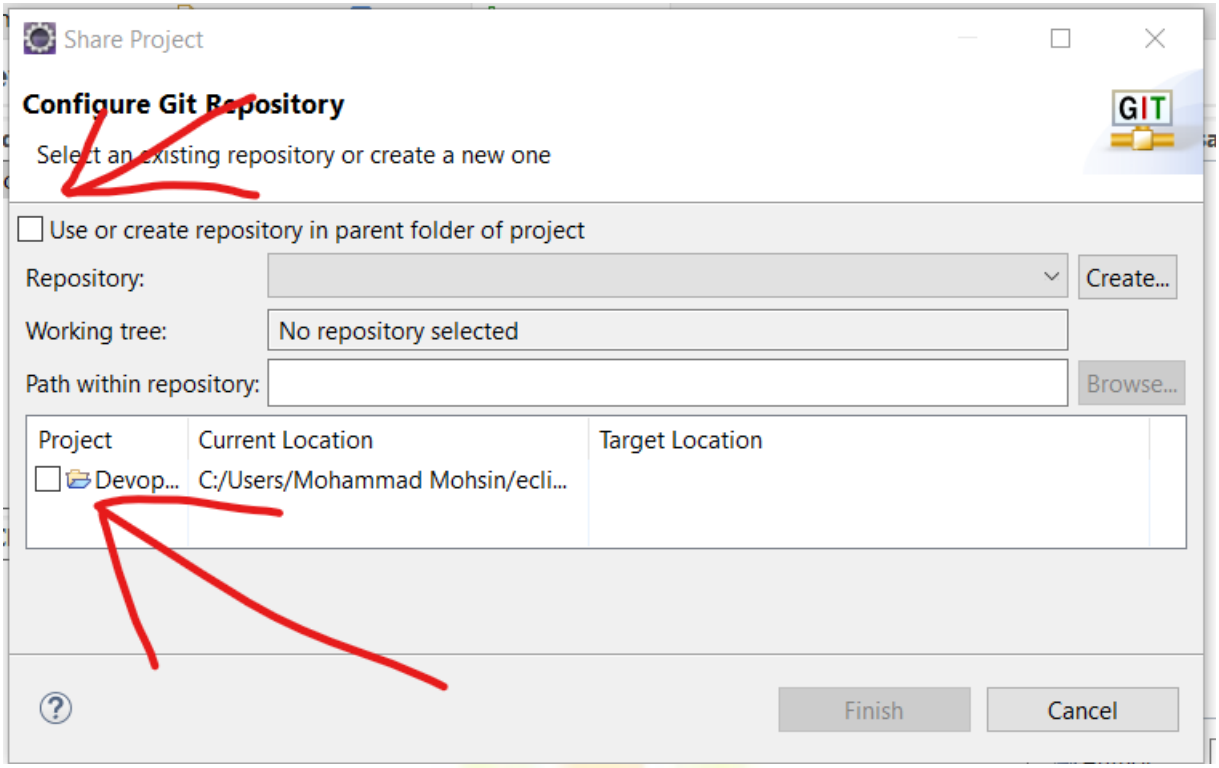

GitHub Apps
OAuth Apps
Personal access tokens
Fine-grained tokens
Preview
Tokens (classic)
GitHub Apps
No GitHub Apps
Want to build something that integrates with and extends GitHub? Register a new GitHub App to get started developing on the GitHub API.
New GitHub App
View documentation

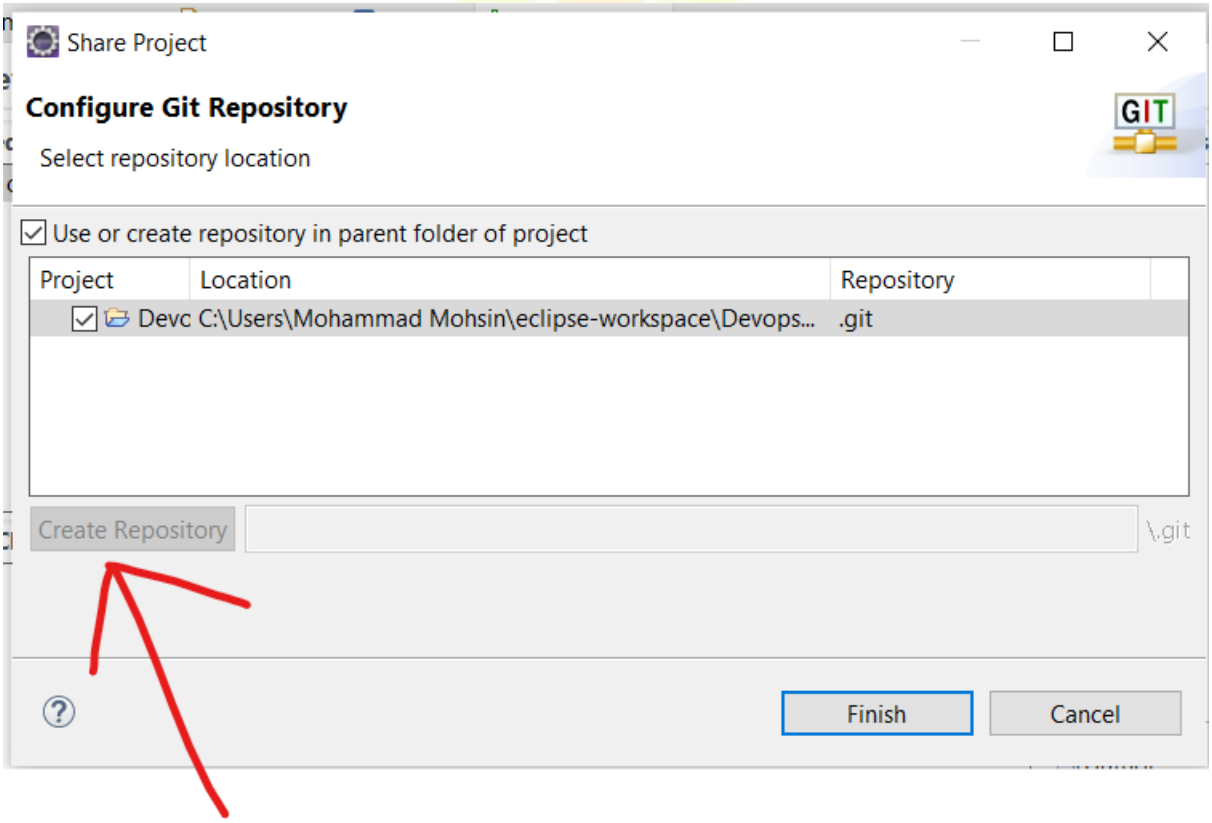


Step-22: After Generating the **token** copy the token id and paste in a **NotePad.**

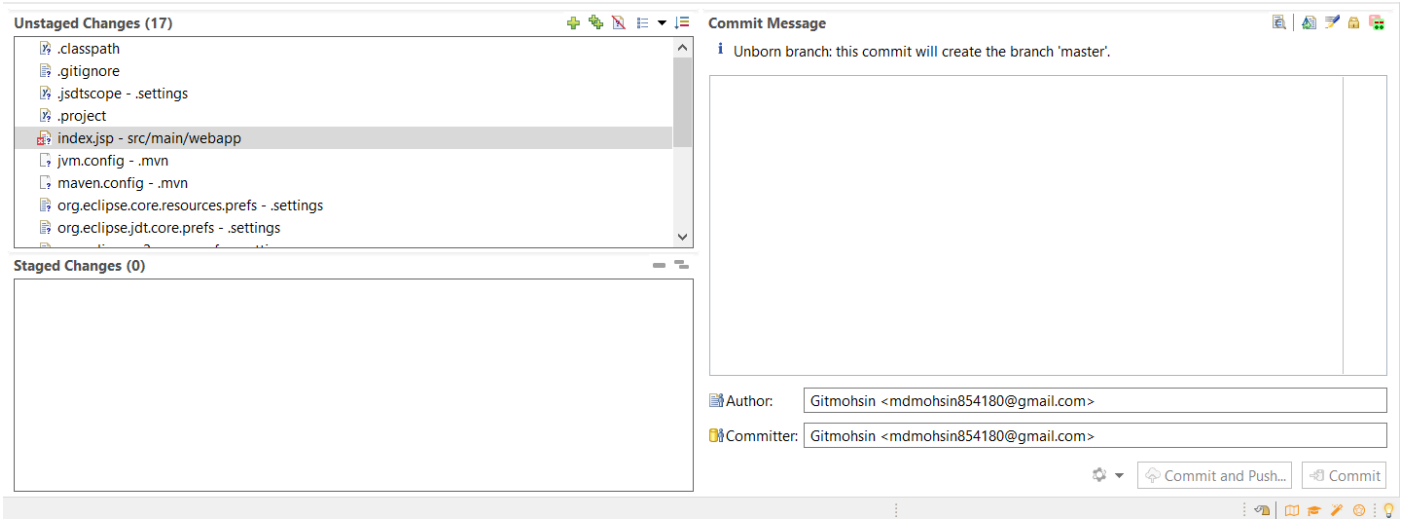
Step-23: Now come on your project and right click on your **project -> Click** on **Team -> Click** on **Share Project.**

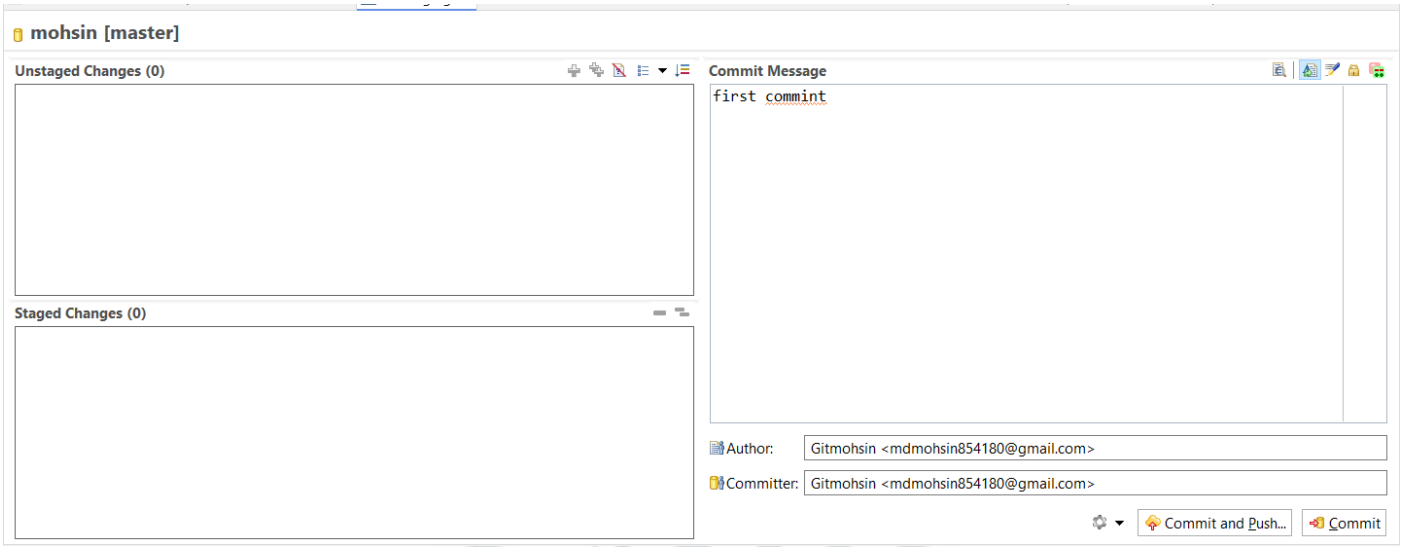
Step- 24: It will open a Dialog Box for **GitHub** Setup, select the **option Use or create repository in parent folder of project - >** Select your Project and Click on **Create Repository** and click on **Finish.**

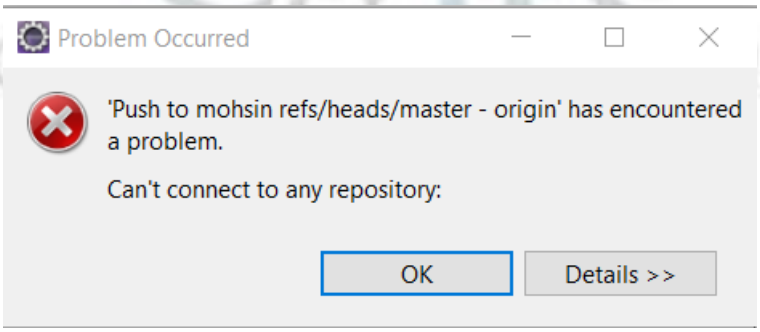


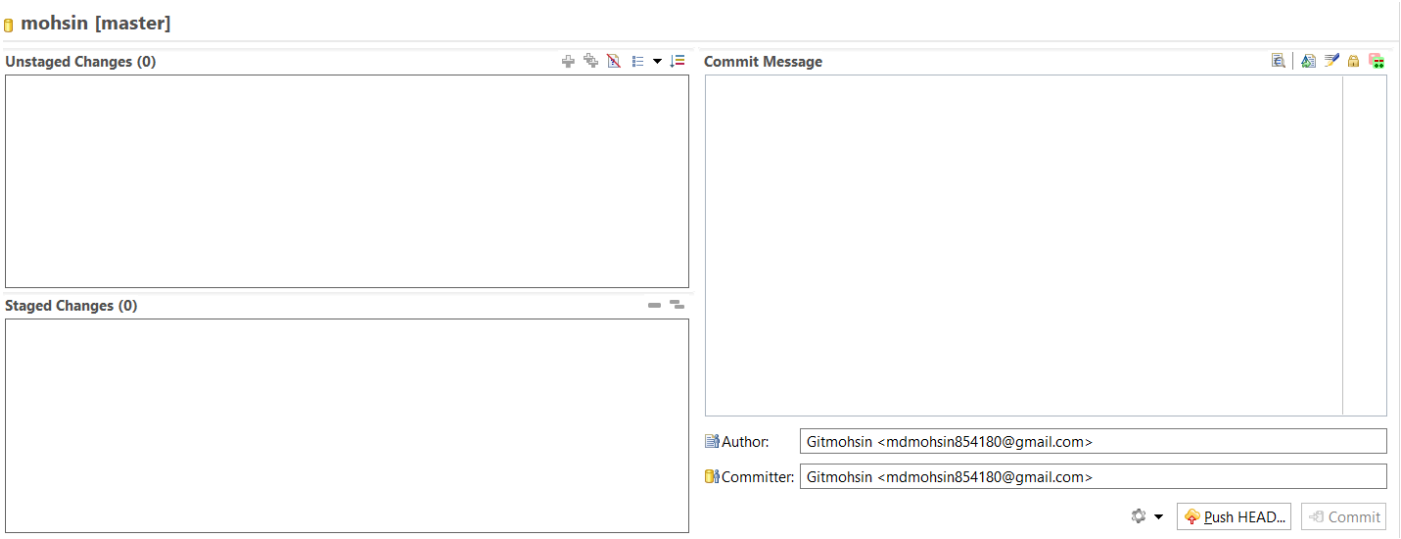


Step-25: After that again **Right click on your Project** and select the **Team** -> click on **Commit** -> and **stage your all file** -> and Write a comment (i.e. First Commit) and click on **Commit and push** -> after that it show an **error dialog ->** click **OK ->** now again click on **Push Head** Button

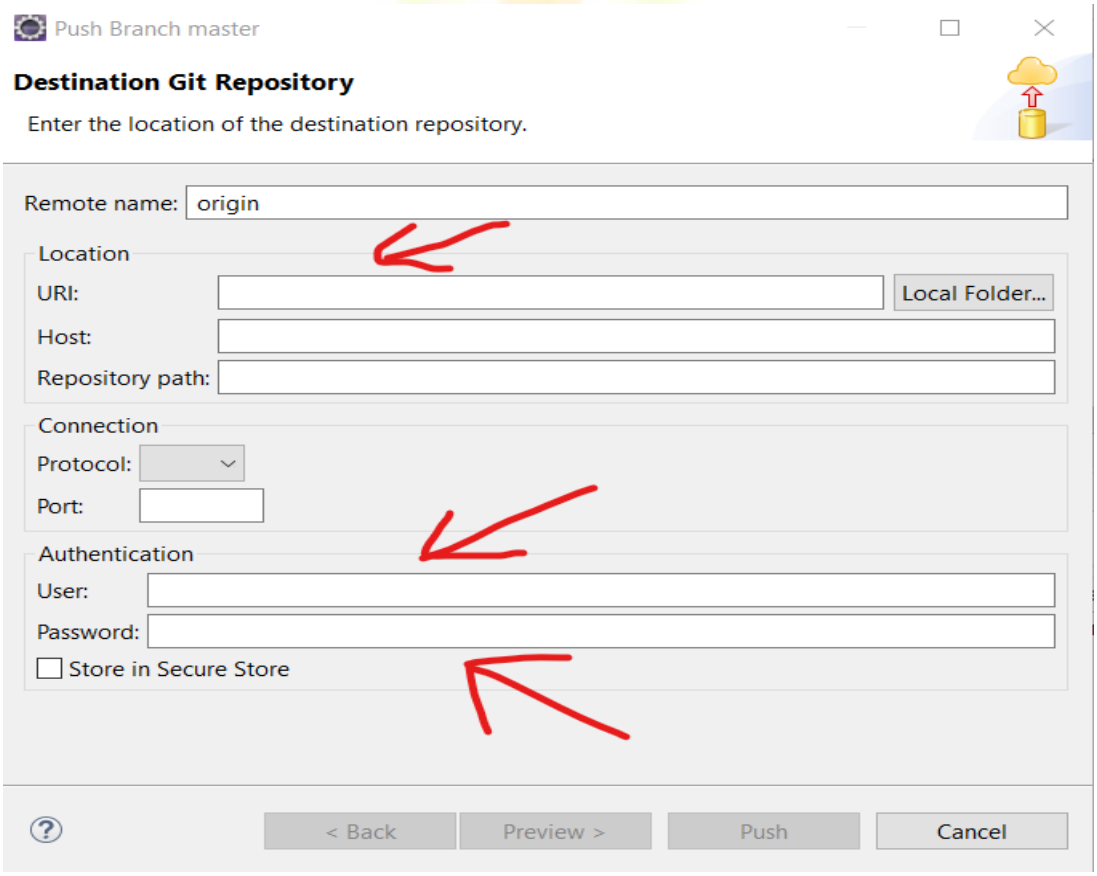

Unstaged Changes (17)
.classpath
.gitignore
.jsdtscope - .settings
.project
index.jsp - src/main/webapp
jvm.config - .mvn
maven.config - .mvn
org.eclipse.core.resources.prefs - .settings
org.eclipse.jdt.core.prefs - .settings
Staged Changes (0)
Commit Message
Unborn branch: this commit will create the branch 'master'.
Author: Gitmohsin <mdmohsin854180@gmail.com>
Committer: Gitmohsin <mdmohsin854180@gmail.com>
Commit and Push...
Commit


mohsin [master]
Unstaged Changes (0)
Staged Changes (0)
Commit Message
first commint
Author: Gitmohsin <mdmohsin854180@gmail.com>
Committer: Gitmohsin <mdmohsin854180@gmail.com>
Commit and Push...
Commit

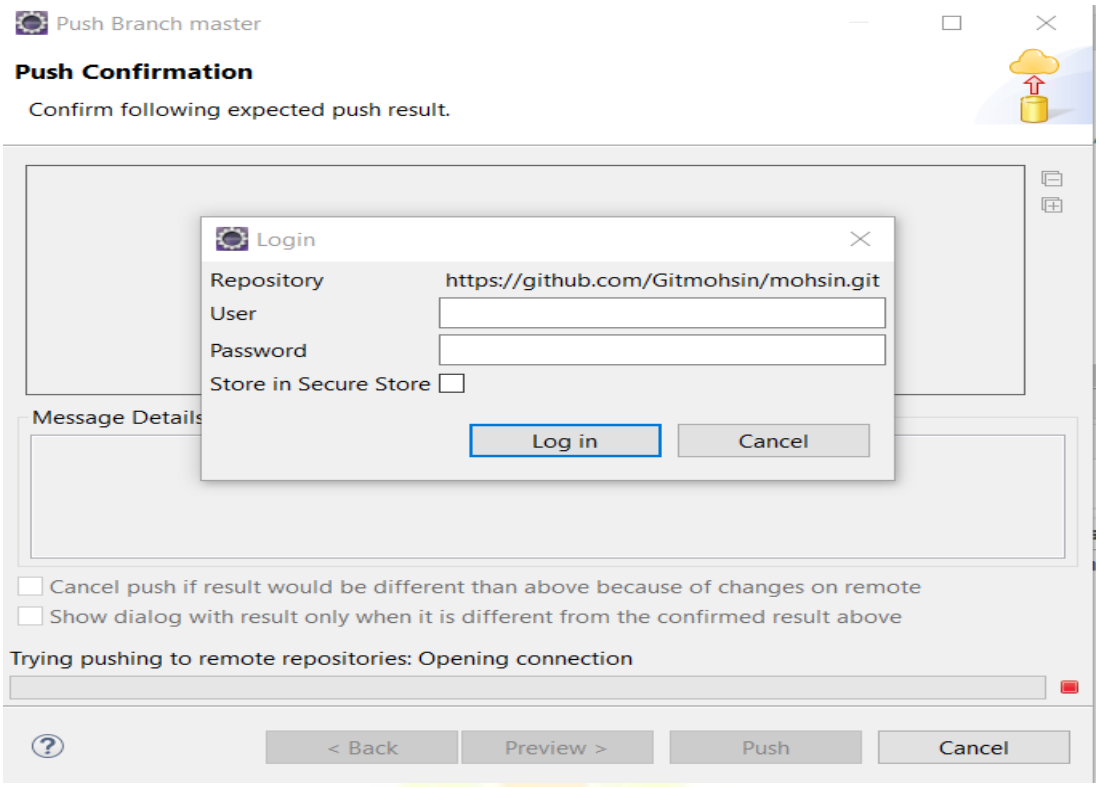

Problem Occurred
'Push to mohsin refs/heads/master - origin' has encountered a problem.
Can't connect to any repository:
OK
Details >>



Step-26: After that again click on **Push Head,** it will show a **dialog** like given image, paste your **Repository URL** in **URL** section and type your **GitHub User Id and Password** in User, password section -> Click on **preview ->** Again click on **Preview.**



Step-27: After that it will again show a **user Id** and **Password** option **->** just type your **Github id** in user section and paste your **Token id** in Password section -> click on **push ->** one more time it will ask **user id and** password just repeat your last step with u**ser id and token id ->** now check your **repository on github,** your file is **uploaded or not**



Step-28: Now you have to create a simple java code in SRC File, so first open your project from **file manager ->** open **SRC - >** Create two folder in **SRC -> first name: java, second name: test ->** now open **test folder** and create two more folder in **test folder ->** now come on your **eclipse IDE** and **Update** your project once -> After that create a java class file with a Statement **"Hello World"** in your **SRC/TEST/java** folder.

Step-29: Now Push again your **all unstage files** in your **GitHub Repository** with **different version** or **Comment** (Its just for Version Control).

Step-30: Now Check again your Repository your recent file is uploaded or not with different version.