

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

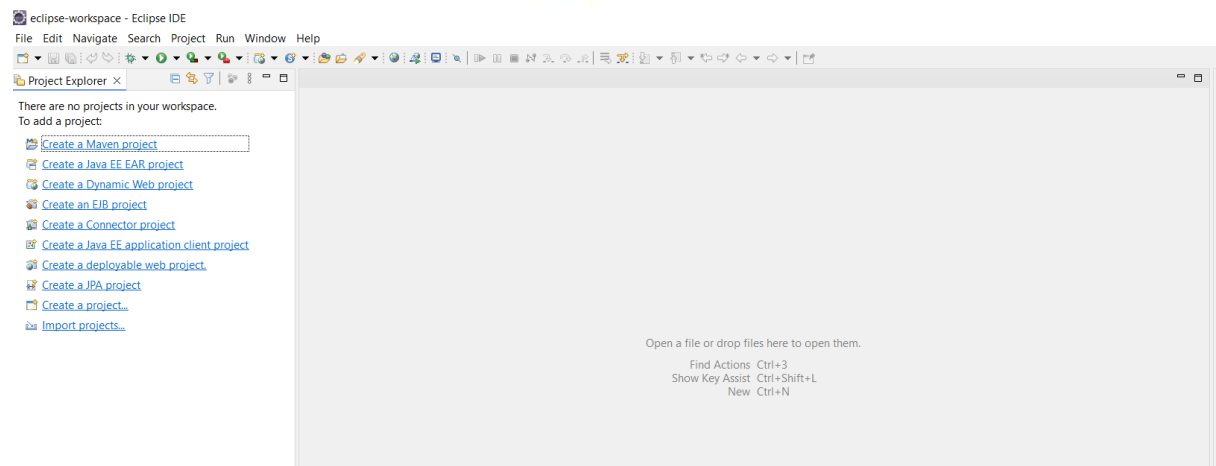
Step-2: Download eclipse zip file and extract the contents the all eclipse file

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

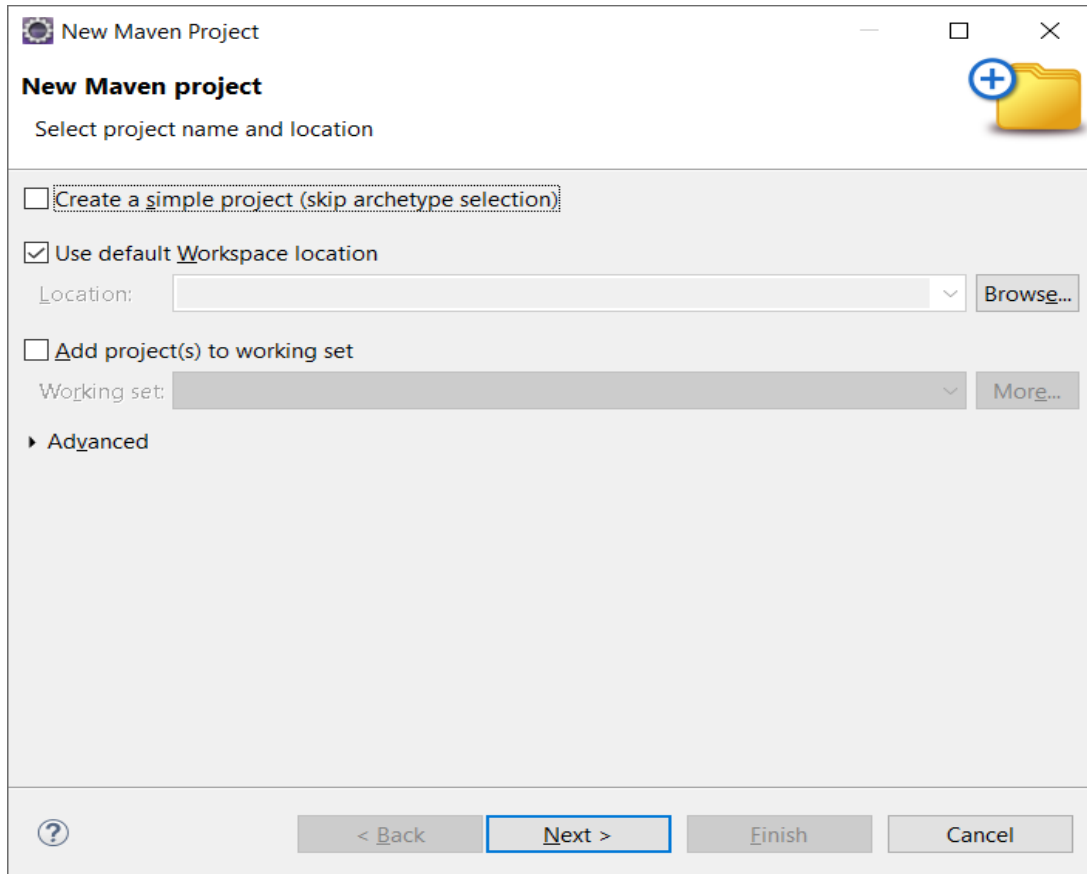
OR

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

a.



b.



New Maven Project

Select project name and location

☐ Create a simple project (skip archetype selection)

☒ Use default Workspace location

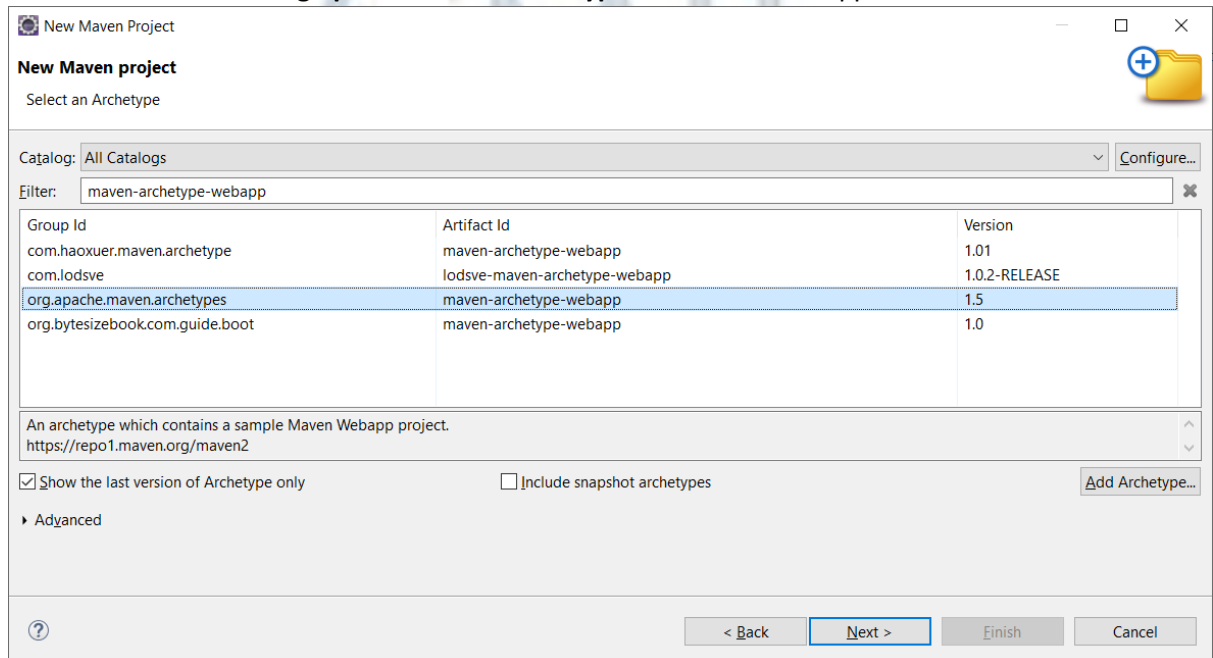
Location:

☐ Add project(s) to working set

Working set:

► Advanced

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



New Maven Project

Select an Archetype

Catalog:

Filter:

Group Id	Artifact Id	Version
com.haoxuer.maven.archetype	maven-archetype-webapp	1.01
com.lodsve	lodsve-maven-archetype-webapp	1.0.2-RELEASE
org.apache.maven.archetypes	maven-archetype-webapp	1.5
org.bytesizebook.com.guide.boot	maven-archetype-webapp	1.0

An archetype which contains a sample Maven Webapp project.
<https://repo1.maven.org/maven2>

☒ Show the last version of Archetype only ☐ Include snapshot archetypes

► Advanced

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

New Maven Project

New Maven project
Specify Archetype parameters

Group Id:

Artifact Id:

Version:

Package:

☒ run archetype generation interactively

Properties available from archetype:

Name	Value

[Add...](#)
[Remove](#)

► Advanced

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

e.

f. Click Finish

- [Create a Connector project](#)
- [Create a Java EE application client project](#)
- [Create a deployable web project](#)
- [Create a JPA project](#)
- [Create a project...](#)
- [Import projects...](#)

```

Open a file or drop files here to open them.
Find Actions Ctrl+3
Show Key Assist Ctrl+Shift+L
New Ctrl+N

C:\eclipse\plugins\org.eclipse.just\openjdkhotspot.jre.full.win32.x86_64_21.0.5.v20241023-1957\jre\bin\javaw.exe (26-Dec-2024, 7:07:36 pm) [pid: 5840]
Progress (1): 16/16 MB
Progress (1): 16/16 MB
Progress (1): 16/16 MB
Progress (1): 16/16 MB
Progress (1): 16 MB
Downloaded from central: https://repo.maven.apache.org/maven2/archetype-catalog.xml (16 MB at 3.3 MB/s)
[INFO] Archetype repository not defined. Using the one from [org.apache.maven.archetypes:maven-archetype-webapp:1.5] found in catalog remote
[INFO] Using property: groupId = DevOpsExp2
[INFO] Using property: artifactId = MyDevOpsPipeline
[INFO] Using property: version = 0.0.1-SNAPSHOT
[INFO] Using property: package = war
Confirm properties configuration:
groupId: DevOpsExp2
artifactId: MyDevOpsPipeline
version: 0.0.1-SNAPSHOT
package: war
Y: Y

```

g.

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

```

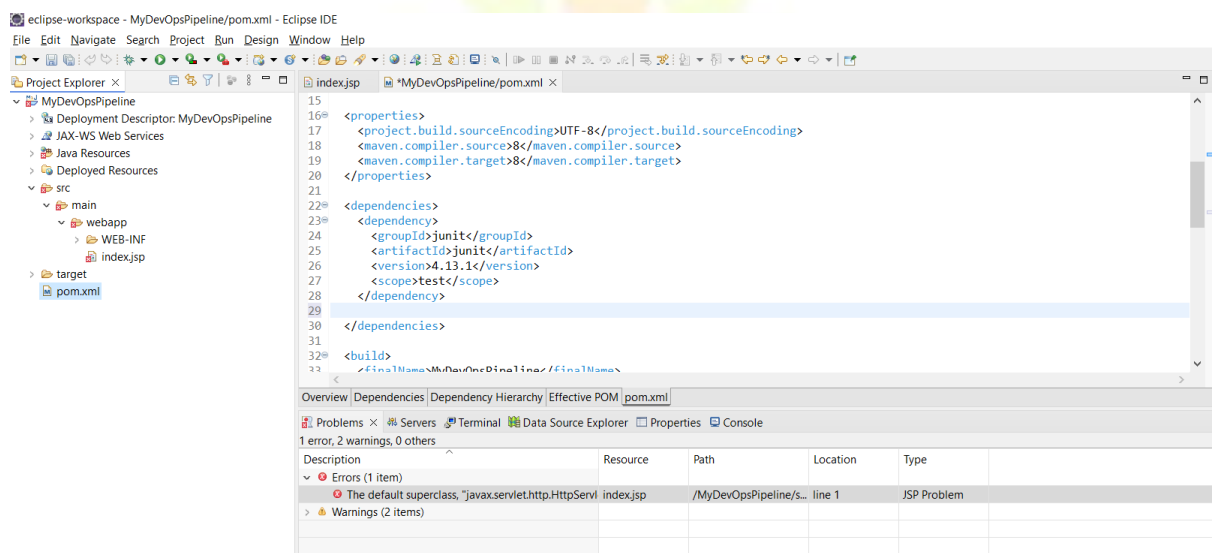
Show Key Assist Ctrl+Shift+L
New Ctrl+N

[terminated] C:\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.5.v20241023-1957\jre\bin\javaw.exe (26-Dec-2024, 7:07:36 pm) [pid: 5840]
[INFO] Parameter: groupId, Value: DevOpsExp2
[INFO] Parameter: artifactId, Value: MyDevOpsPipeline
[INFO] Parameter: version, Value: 0.0.1-SNAPSHOT
[INFO] Parameter: package, Value: war
[INFO] Parameter: packageInPathFormat, Value: war
[INFO] Parameter: package, Value: war
[INFO] Parameter: groupId, Value: DevOpsExp2
[INFO] Parameter: artifactId, Value: MyDevOpsPipeline
[INFO] Parameter: version, Value: 0.0.1-SNAPSHOT
[WARNING] CP Don't override file C:\Users\Dell\eclipse-workspace\MyDevOpsPipeline\src\main\webapp
[WARNING] CP Don't override file C:\Users\Dell\eclipse-workspace\MyDevOpsPipeline\src\main\resources
[INFO] Project created from Archetype in dir: C:\Users\Dell\eclipse-workspace\MyDevOpsPipeline
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:22 min
[INFO] Finished at: 2024-12-26T19:09:00+05:30
[INFO] -----

```

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.

https://tomcat.apache.org

Apache Tomcat®

COMMUNITY CODE

Apache Tomcat

- Home
- Taglibs
- Maven Plugin

Download

Which version?

- Tomcat 11
- Tomcat 10
- Tomcat 9
- Tomcat Migration Tool for Jakarta EE
- Tomcat Connectors
- Tomcat Native
- Taglibs
- Archives

Documentation

Apache Tomcat

The Apache Tomcat® software is an open source implementation of the [Jakarta Servlet](#), [Jakarta Pages](#), [Jakarta Expression Language](#), [Jakarta WebSocket](#), [Jakarta Annotations](#) and [Jakarta Authentication](#) specifications. These specifications are part of the [Jakarta EE platform](#).

The Jakarta EE platform is the evolution of the Java EE platform. Tomcat 10 and later implement specifications developed as part of Jakarta EE. Tomcat 9 and earlier implement specifications developed as part of Java EE.

The Apache Tomcat software is developed in an open and participatory environment and released under the [Apache License version 2](#). The Apache Tomcat project is intended to be a collaboration of the best-of-breed developers from around the world. We invite you to participate in this open development project. To learn more about getting involved, [click here](#).

Apache Tomcat software powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations. Some of these users and their stories are listed on the [PoweredBy](#) wiki page.

Apache Tomcat, Tomcat, Apache, the Apache feather, and the Apache Tomcat project logo are trademarks of the Apache Software Foundation.

Tomcat Migration Tool for Jakarta EE 1.0.9 Released

2025-01-21

The Apache Tomcat Project is proud to announce the release of 1.0.9 of the Apache Tomcat Migration Tool for Jakarta EE. This release contains a number of bug fixes and improvements compared to version 1.0.8.

The notable changes in this release are:

- Fix an issue that matchExcludesAgainstPathName didn't work for files. Based on a pull request by Semiao Marco.
- Added a new profile, SERVLET that only migrates the javax.servlet package and sub-packages. Provided by Ralf Wiebicke.
- Update dependencies

Full details of these changes, and all the other changes, are available in the [changelog](#).

Mirrors

You are currently using <https://dldcn.apache.org/>. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the mirrors list) that should be available.

Other mirrors:

9.0.98

Please see the [README](#) file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
 - [32-bit Windows zip \(pgp, sha512\)](#)
 - [64-bit Windows zip \(pgp, sha512\)](#)
 - [32-bit/64-bit Windows Service Installer \(pgp, sha512\)](#)
- Full documentation:
 - [tar.gz \(pgp, sha512\)](#)
- Deployer:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
- Embedded:
 - [tar.gz \(pgp, sha512\)](#)
 - [zip \(pgp, sha512\)](#)

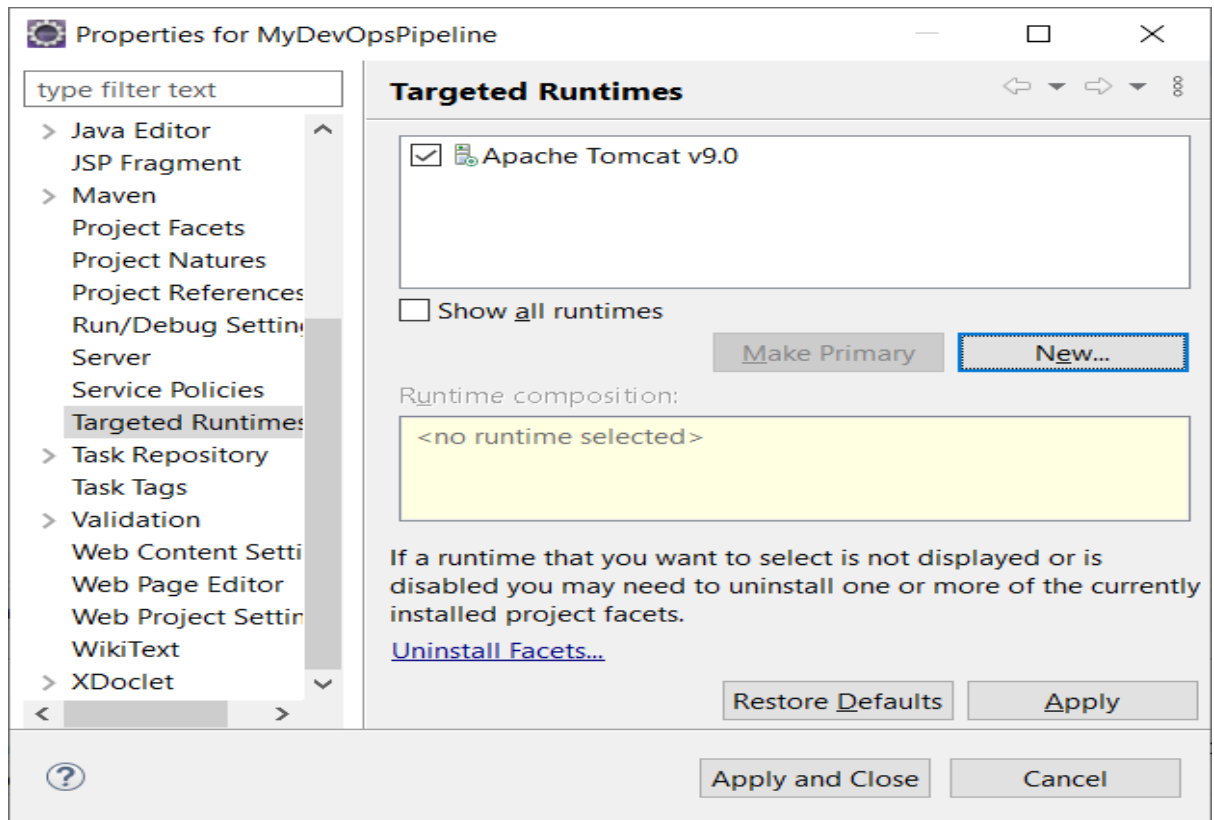
Source Code Distributions

- [tar.gz \(pgp, sha512\)](#)
- [zip \(pgp, sha512\)](#)

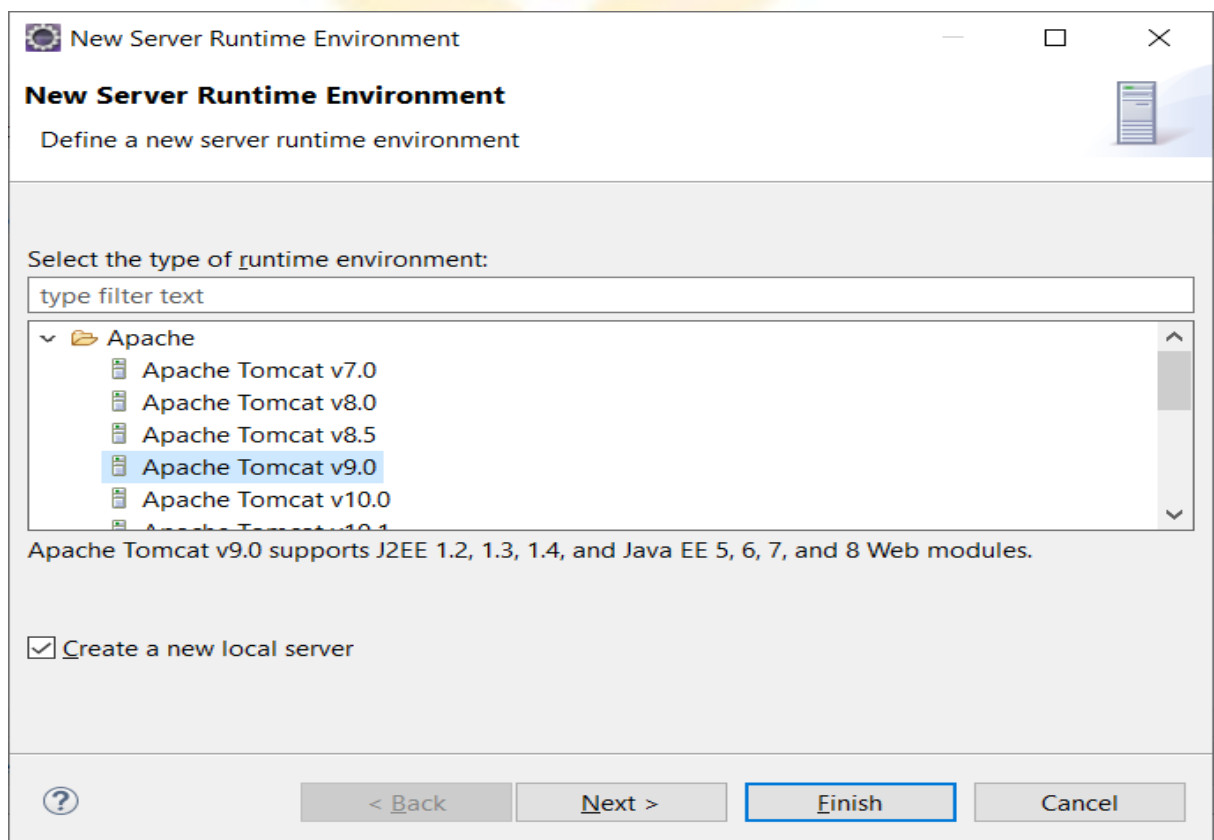
Step-7: After Download the Apache tomcat, Extract the .zip file and paste your apache-tomcat-9.0.98 folder in your folder

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

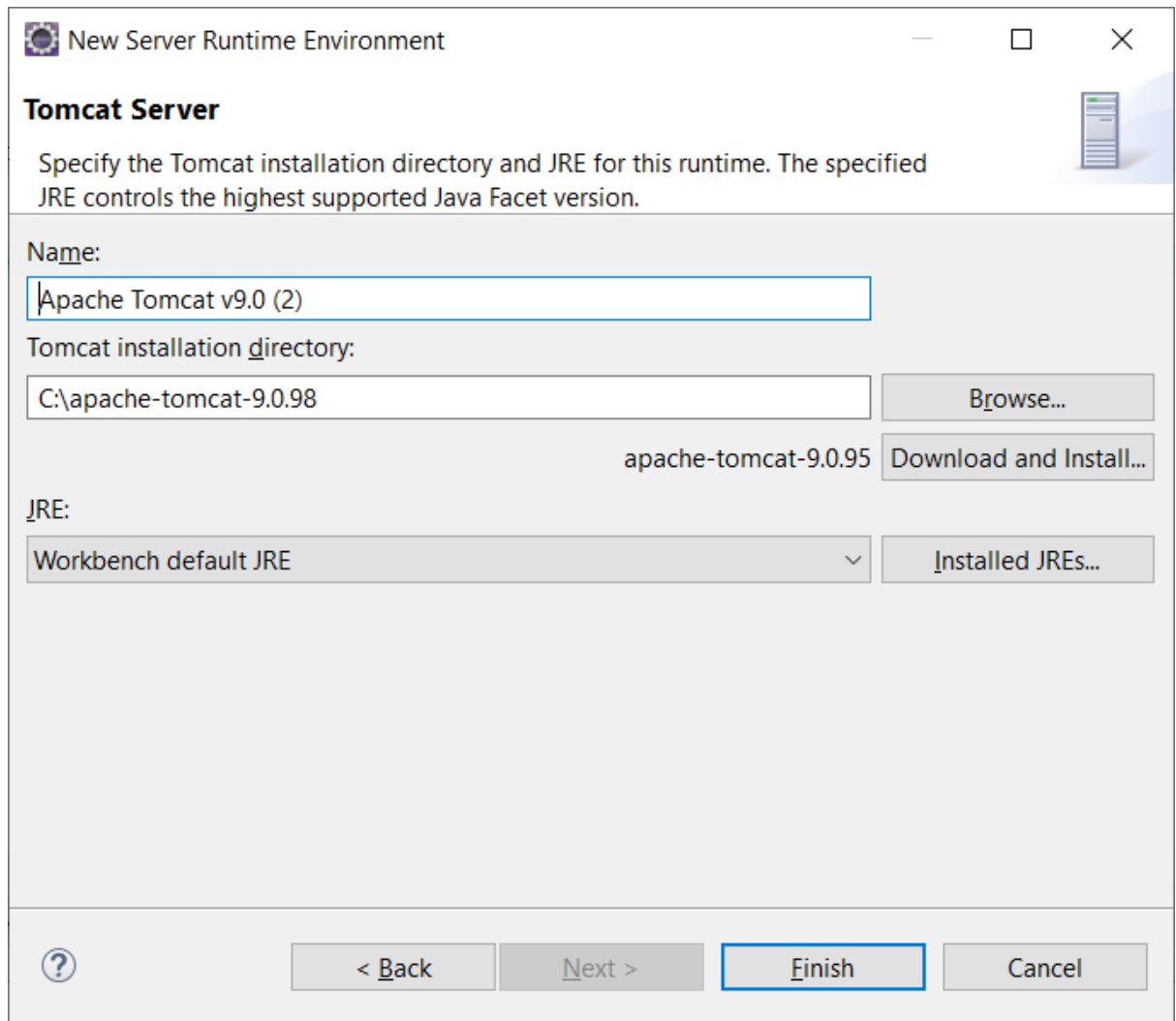
Step-9: Click on **new or follow the given image**



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

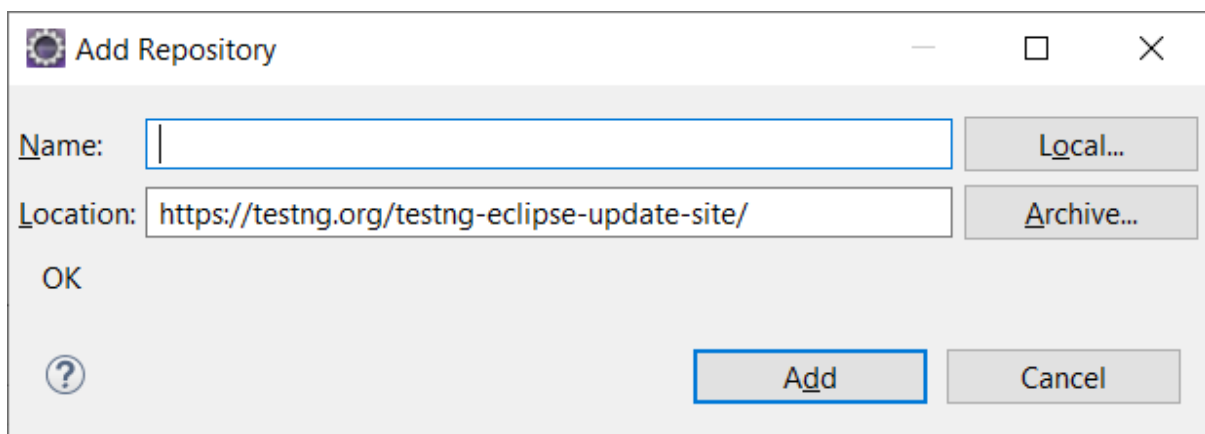


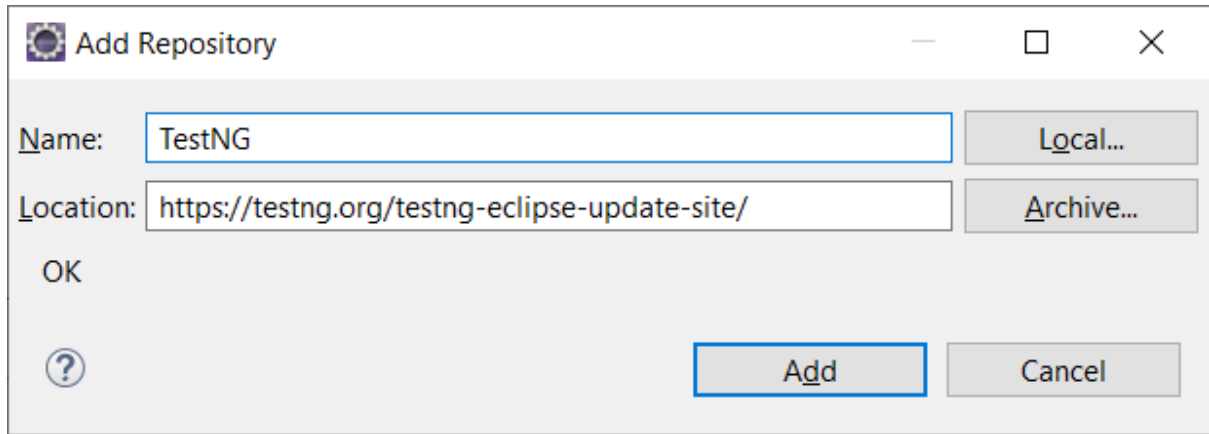
Step-12: Now Click on **help Menu** -> click on **Install new Software**.

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

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

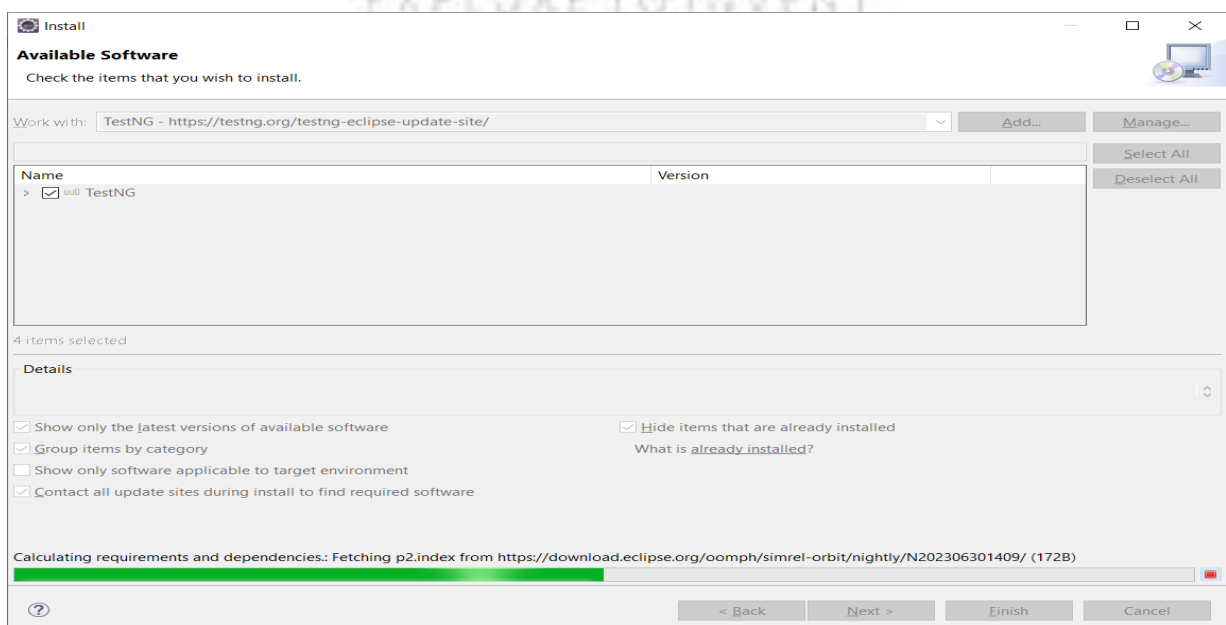
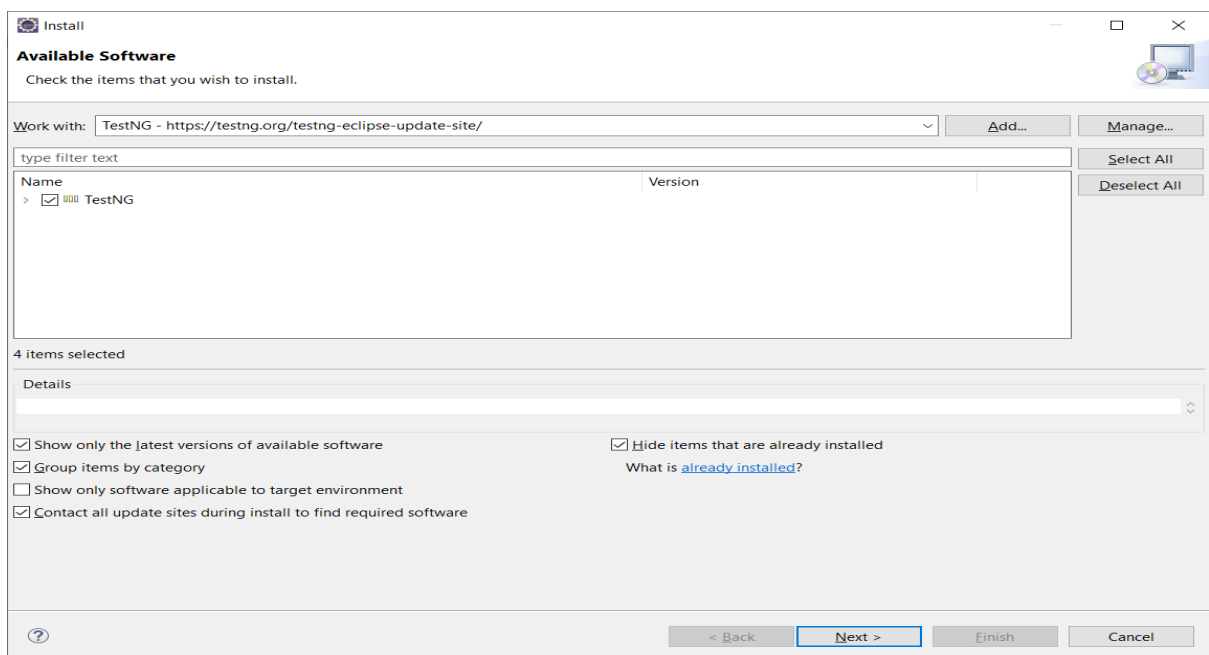
In the place of Location type: **<https://testng.org/testng-eclipse-update-site/>**



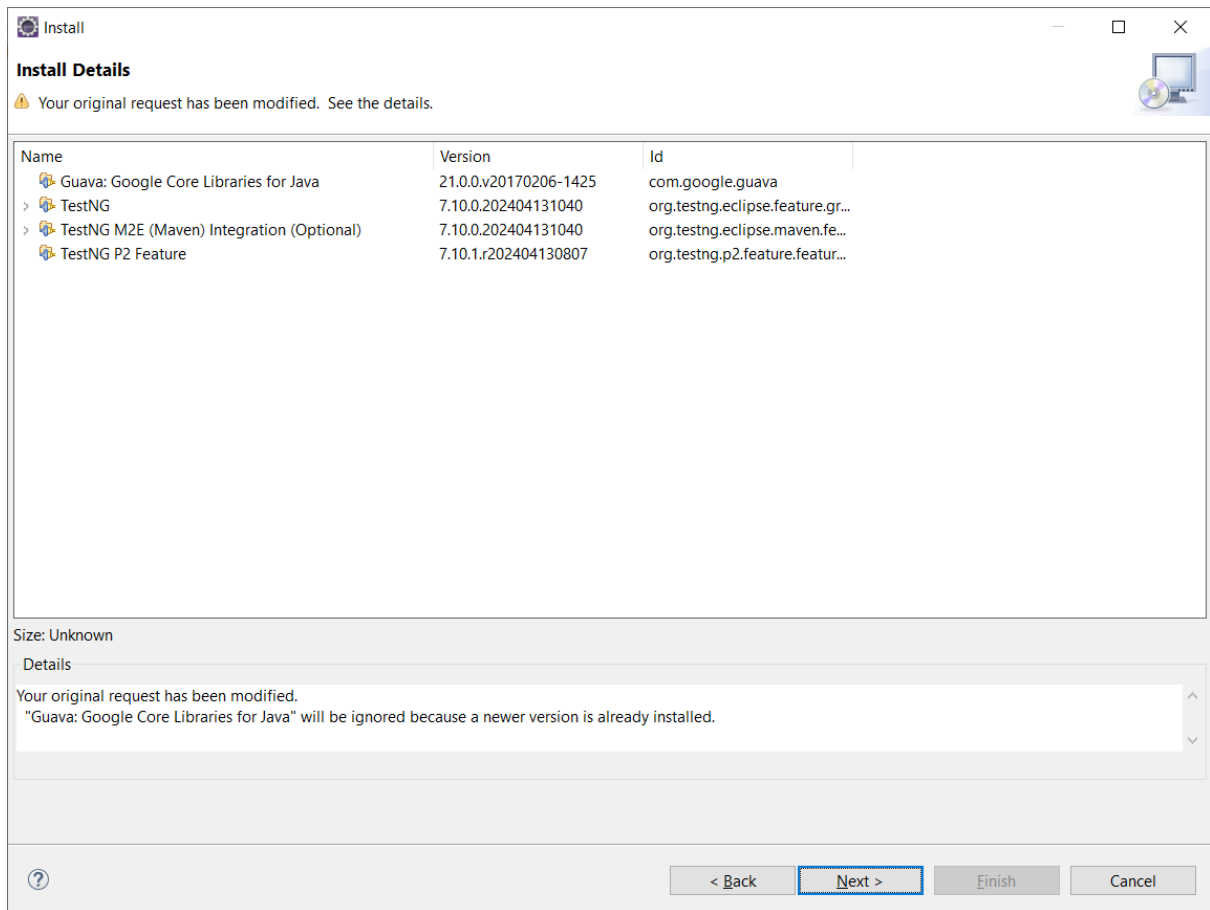


Step-14: Click on **Add** -> It will load a **testNG Dependencies** -> Select **TestNg** like given Image and then click **Next**.

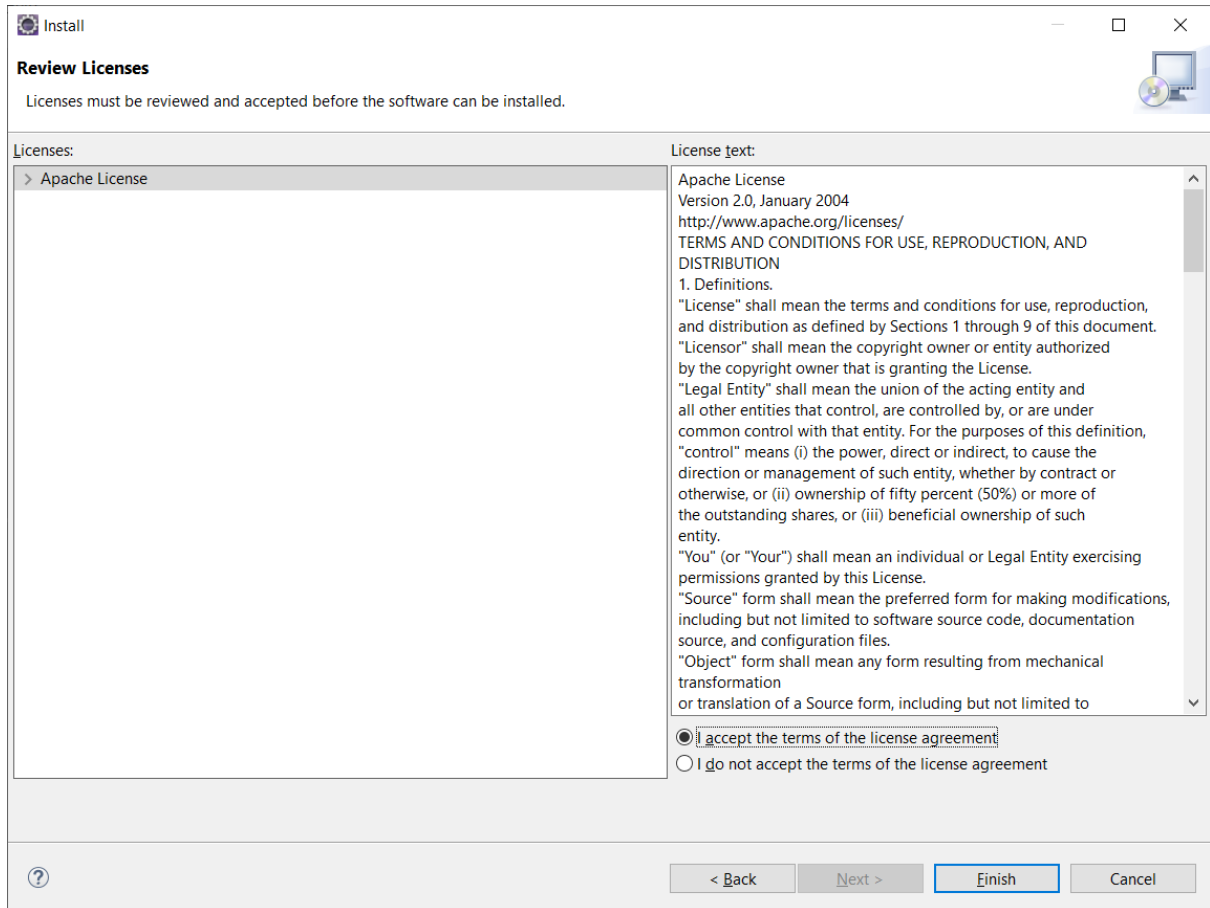
It will take 10 minute to update TestNG in our Project



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

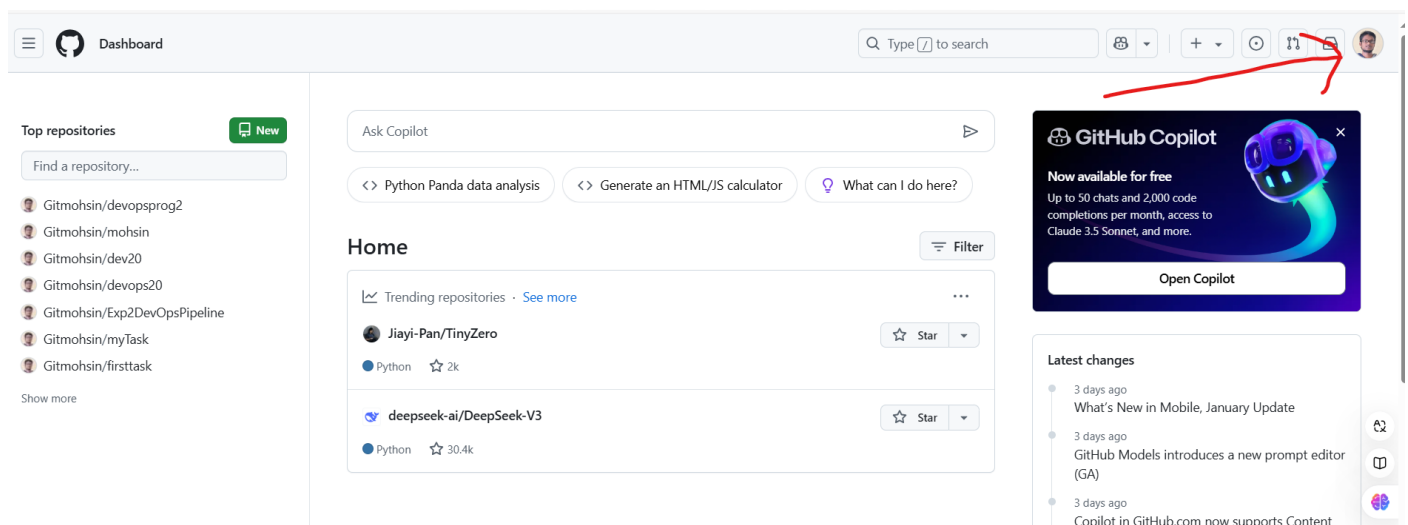


Step-17: After finish it will show **restart option (Restart the Project)** otherwise just **update** once of your project.

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

Python 2k

Star

deepseek-ai/DeepSeek-V3

Python 30.4k

Star

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

Tokens (classic)

Preview

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



No personal access token created

Need an API token for scripts or testing? Generate a personal access token for quick access to the GitHub API.

Generate new token ▾

Generate new token Beta

Fine-grained, repo-scoped

Generate new token (classic)

For general use

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

What's this token for?

Expiration *

30 days ▾

The token will expire on Wed, Feb 26 2025

Select scopes

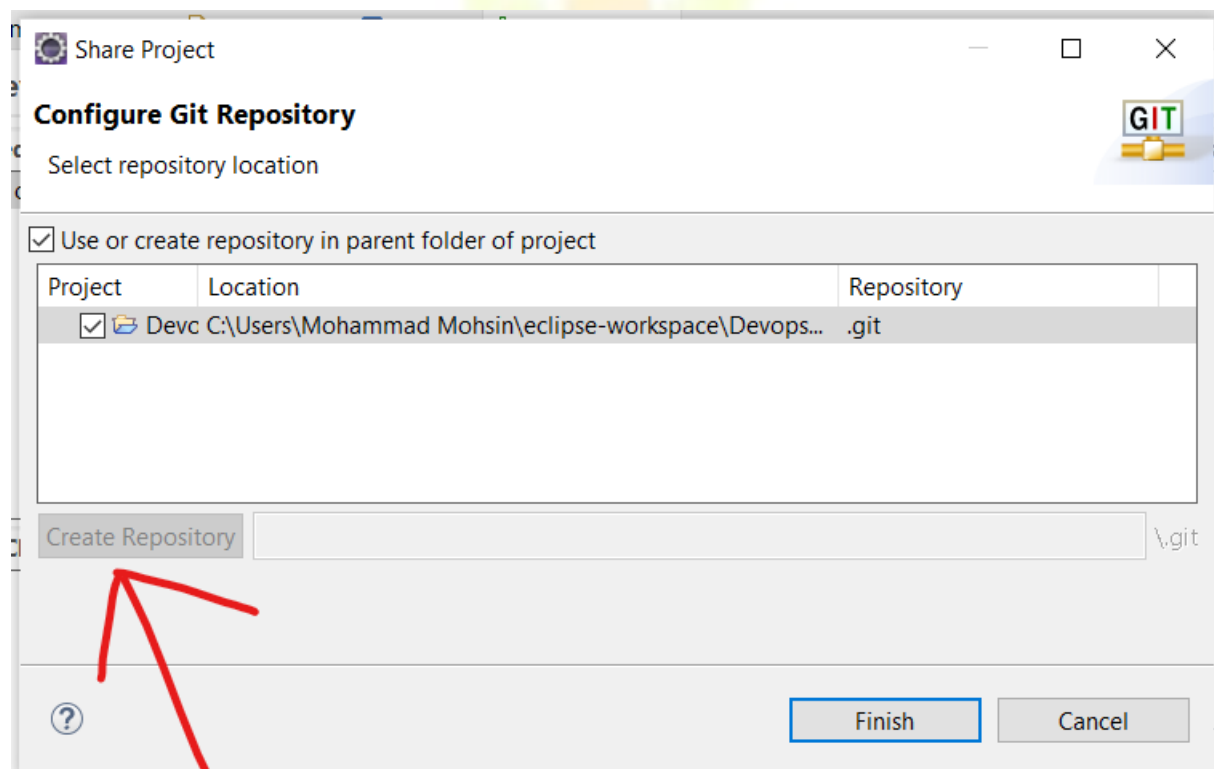
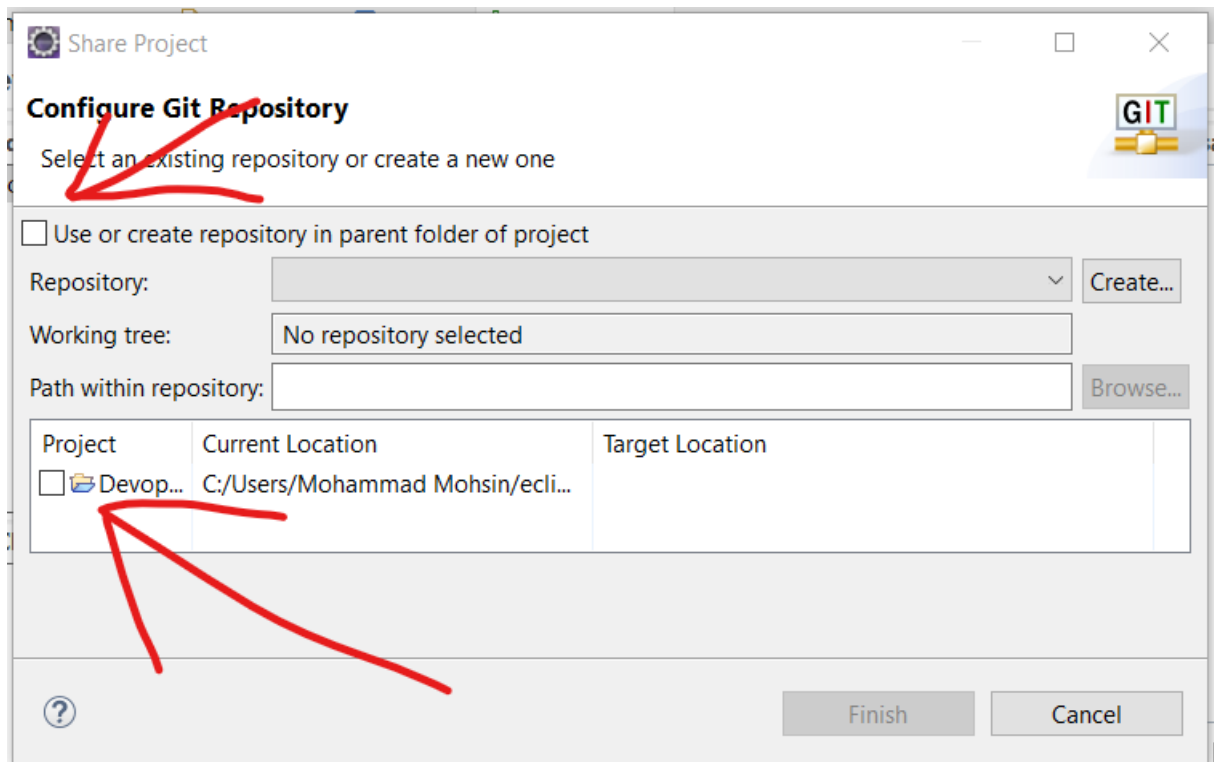
Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

- | | |
|--|--------------------------------------|
| <input type="checkbox"/> repo | Full control of private repositories |
| <input type="checkbox"/> repo:status | Access commit status |
| <input type="checkbox"/> repo_deployment | Access deployment status |
| <input type="checkbox"/> public_repo | Access public repositories |
| <input type="checkbox"/> repo:invite | Access repository invitations |
| <input type="checkbox"/> security_events | Read and write security events |

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)

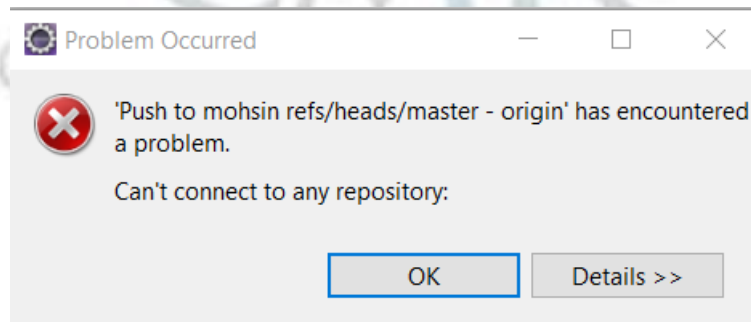
Commit Message

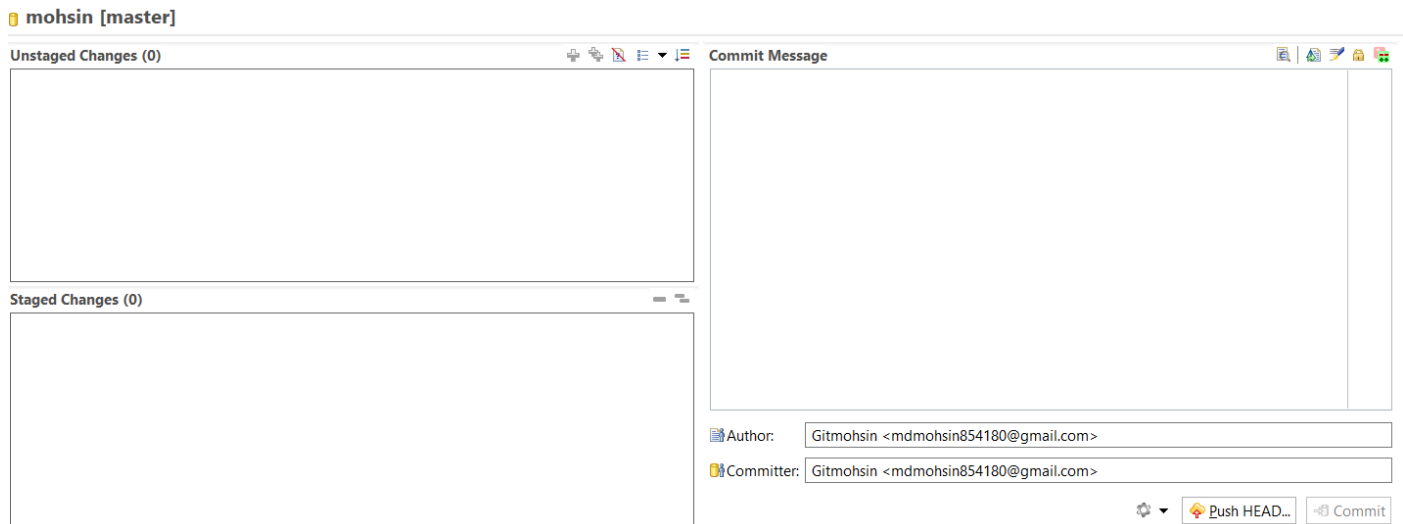
first commint

Author: Gitmohsin <mdmohsin854180@gmail.com>

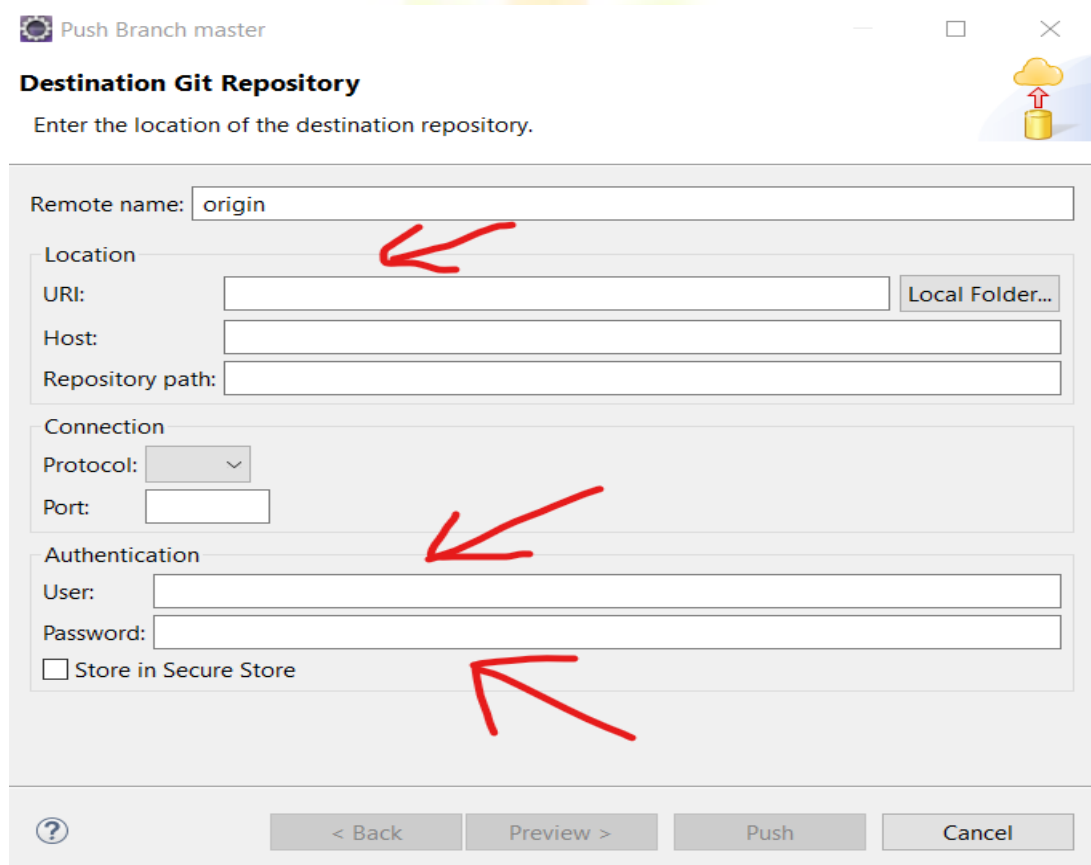
Committer: Gitmohsin <mdmohsin854180@gmail.com>

Commit and Push... Commit

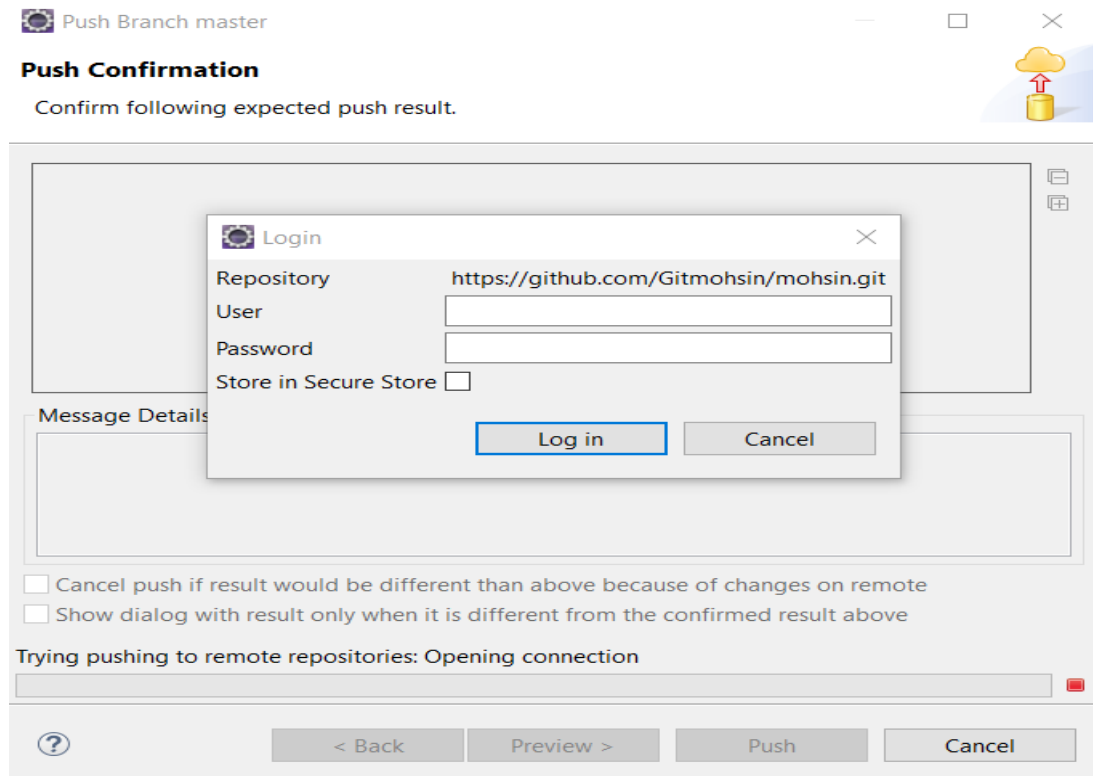




Step-26: After that again click on **Push Head**, it will show a **dialog** like given image, paste your **Repository URL** in **URL** section and type your **GitHub User Id and Password** in User, password section -> Click on **preview** -> Again click on **Preview**.



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 **user 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**.