**Jenkins Automation Project**

**1. Install Pycharm in your local PC**

Configure the git with github by the Access Token in Pycharm 🡪

Go to the setting 🡺 Version Control

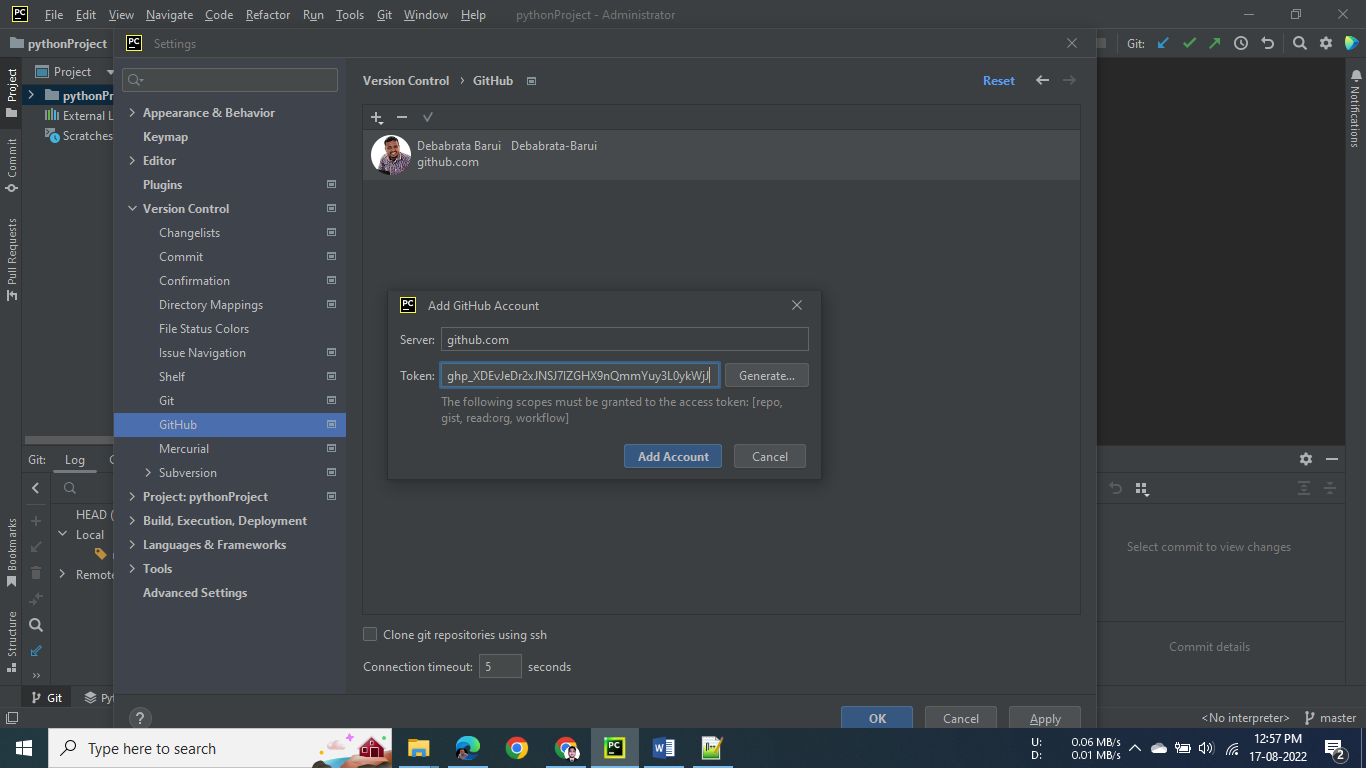
🡪 GitHub

🡪Click On Plus Button( + )

🡪Click On – Login with Token

🡪Paste the Token ID

🡪Click on Add Account amd OK button



To Clone the GitHub Repository 🡺

Click on Git Menu 🡪Click On Clone 🡪 Enter the GitHub Repository URL

**2. Install Tomcat and Download the Jenkins war file. Then Deploy it into**

\Tomcat 9.0\webapps Location..

🡪Restart the tomcat –

Go to \Tomcat 9.0\bin

Run the startup file..

3. Open the Jenkins 🡺

**Install the bellow Plugin** 🡪

Go to Manage Jenkins 🡪 Manage Plugins 🡪 Available

1. Build Pipeline -- To make automate build pipeline

2. Maven Integration -- To Attach Maven Build Tool. To build java application to war & jar file

3. Deploy to container --

4. Copy Artifact -- It is used to Copy the war file to the another job..

To configure the any plugin we should go to the

Manage Jenkins 🡪Global Tool Configuration

Go to Maven 🡪Add Installer

Name : Maven

Check the Install Automatically Option

5. To create Jobs 🡪

1.

Click On New Item on left Panel 🡪

Enter Item Name : WebProject1-Test

🡪Select On **Maven project** 🡪 OK

**Source Code Management 🡺**

Select on Git Radio Button🡪Paste the GitHub Repository Link

( <https://github.com/Debabrata-DevOps/spring-boot-war-example.git> )

**Build Triggers 🡺**

**Check On Poll SCM**

Schedule : \* \* \* \* \*

**Build 🡺**

Goals and options : test

**Post-build Actions 🡺**

**Select On Build other projects**

Enter the Next Job name which you want to build – WebProject1-Build

**Apply and Save**

2.

Click On New Item on left Panel 🡪

Enter Item Name : WebProject1-Build

🡪Select On **Maven project** 🡪 OK

**Permission to Copy Artifact 🡺**

Projects to allow copy artifacts : **WebProject1-\***

After Creating War File You can copy this war file in any (WebProject1-\*) job which is being started by WebProject1

**Source Code Management 🡺**

Select on Git Radio Button🡪Paste the GitHub Repository Link

(<https://github.com/Debabrata-DevOps/spring-boot-war-example.git> )

**Build 🡺**

Goals and options : install

**Post-build Actions 🡺**

**Select On Archive the artifacts**

Files to archive : **\*\*/\*.war**

**Select On Build other projects**

Enter the Next Job name which you want to build – WebProject1-Deploy-Test

**Apply and Save**

3.

Click On New Item on left Panel 🡪

Enter Item Name : WebProject1-Deploy-Test

🡪Select On **Freestyle project** 🡪 OK

**Permission to Copy Artifact 🡺**

Projects to allow copy artifacts : **WebProject1-Deploy-Prod**

After completing this job this war file will go to the Web Project Production Job

**Post-build Actions 🡺**

**Select On Archive the artifacts**

Files to archive : **\*\*/\*.war**

Context path : **/app**

Add Container : **Tomcat 9.x Remote**

Credentials : **admin/\*\*\*\*\*\***

**Add : Username : TomcatServerUsername(admin)**

**Password : TomcatServerPassword(admin)**

Tomcat URL : **http://127.0.0.1:8081**

**Select On Build other projects**

Enter the Next Job name which you want to build – **WebProject1-Deploy-Prod**

**Apply and Save**

**After build this job Check your tomcat server and run /app**

4. This Job is used to deploy the war file into Production Server

Click On New Item on left Panel 🡪

Enter Item Name : WebProject1-Deploy-Prod

🡪Select On **Freestyle project** 🡪 OK

**Permission to Copy Artifact 🡺**

Projects to allow copy artifacts : **WebProject1-Deploy-Prod**

After completing this job this war file will go to the Web Project Production Job

**Post-build Actions 🡺**

**Select On Archive the artifacts**

Files to archive : **\*\*/\*.war**

Context path : **/app**

Add Container : **Tomcat 9.x Remote**

Credentials : **admin/\*\*\*\*\*\***

**Add : Username : TomcatServerUsername(admin)**

**Password : TomcatServerPassword(admin)**

Tomcat URL : **http://127.0.0.1:8081**

**Apply and Save**

6. Create a New View🡪

Name : WebProject1

Type : Build Pipeline View

Click on Create …

**Upstream / downstream config**

You have to select the Job, which you want to build 1st Select Initial Job : WebProject1-Test

**Apply and OK**

7. To See Execute Pipeline🡪

Go to WebProject1 View 🡪

