

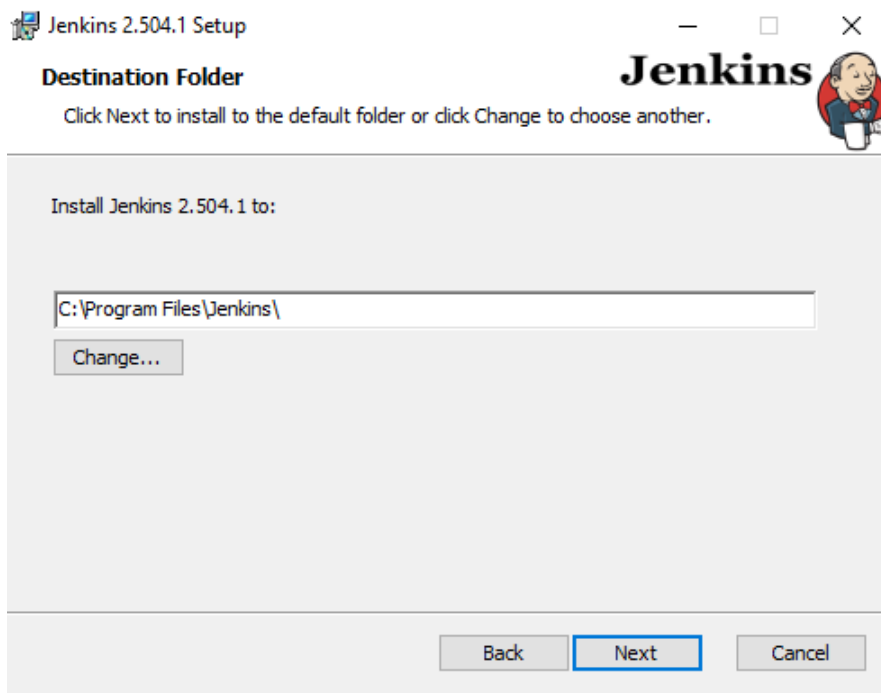
# DevOps - Program 4

## Installation of Jenkins

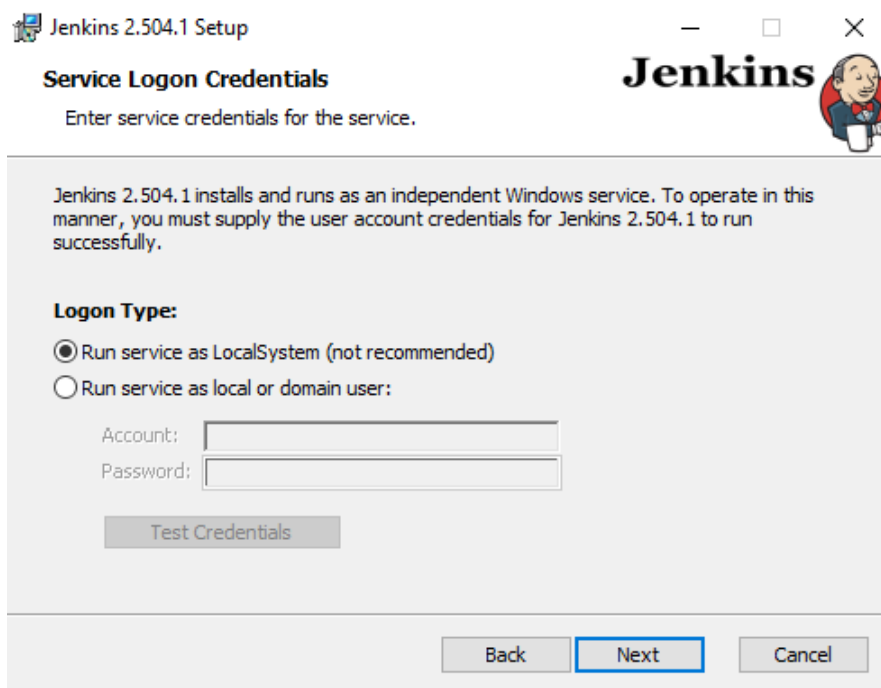
1. Download the 'Jenkins' Executable from - "<https://www.jenkins.io/download/>" (Windows - .msi)
2. Run the 'jenkins.msi' file
3. Click on 'Next'



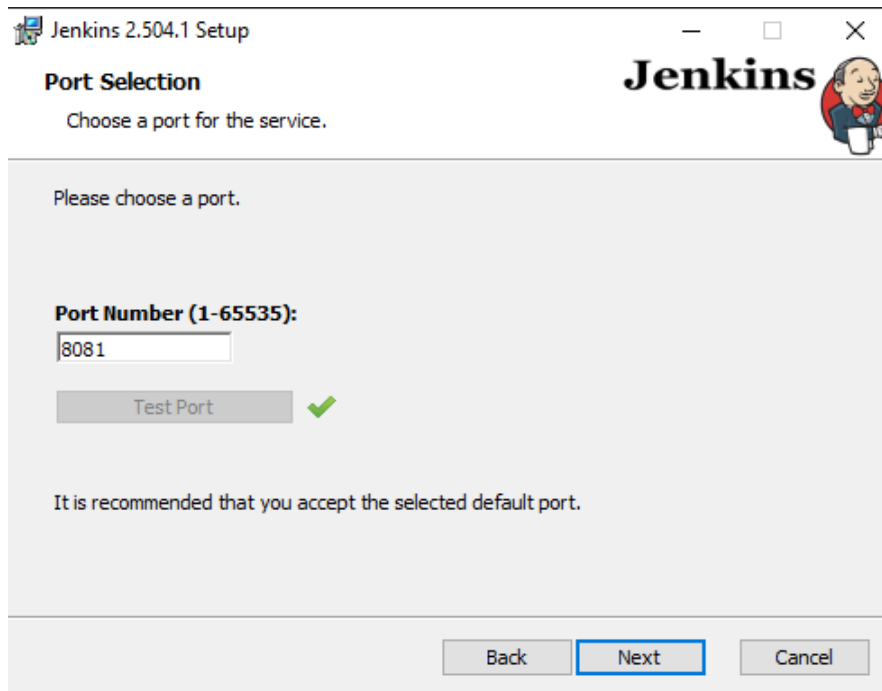
4. Click on 'Next'



5. Select 'Run Service as LocalSystem', Click on 'Next'



6. Enter Port Number as '8081' and Test Port (To Check for any Port Conflicts), Click on 'Next'



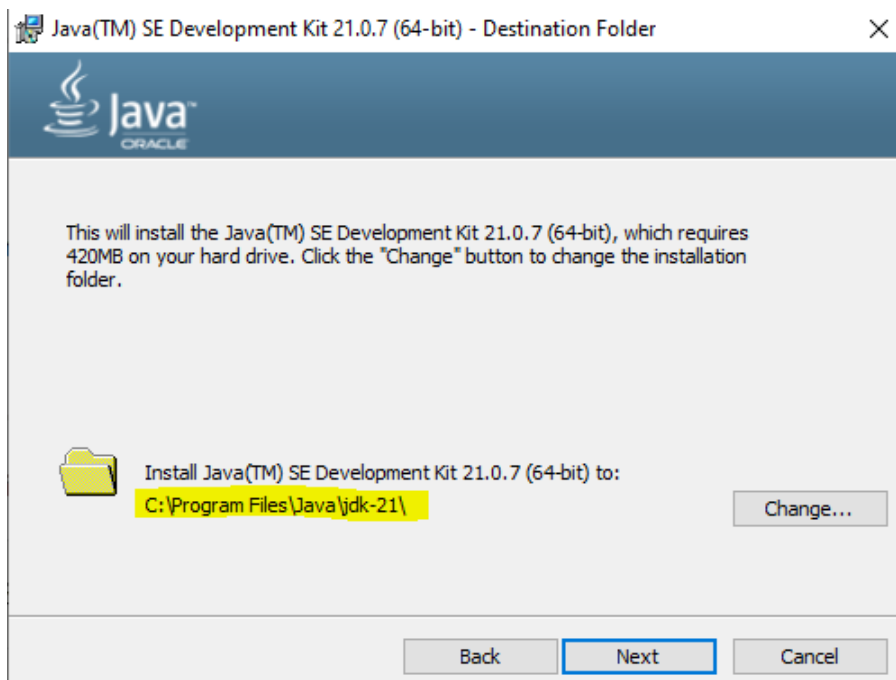
7. If we cannot see any folder 'Java', We will have to install Java JDK (Should not be less than JDK 21)

- URL For Download: <https://www.oracle.com/in/java/technologies/downloads/#jdk21-windows>
- Download the x64 MSI Installer
- Run the 'msi' file for Java JDK

(i) Click on 'Next'



(ii) Take note of the 'Installation Path' as highlighted and Click on 'Next'



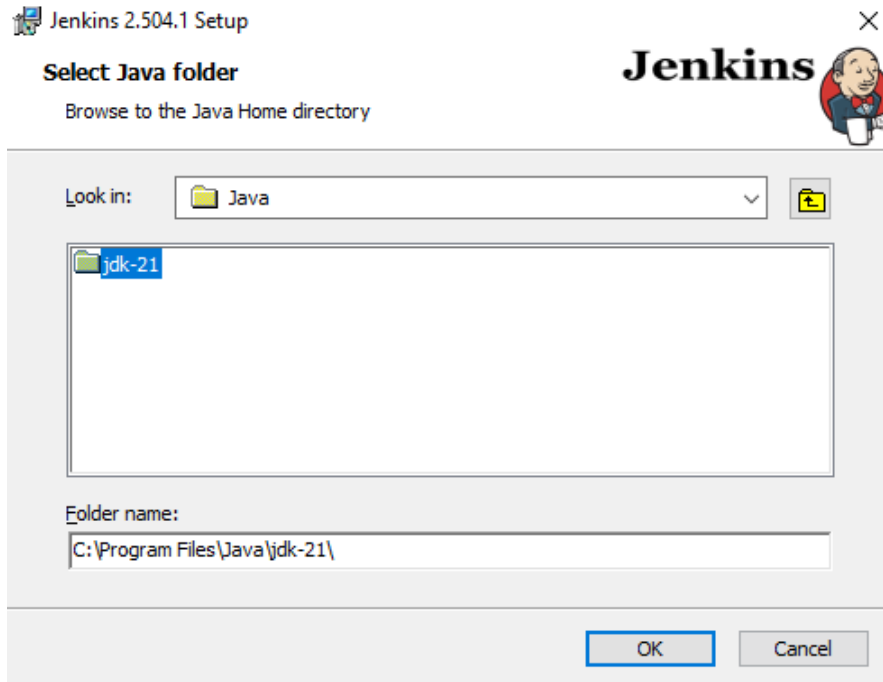
- If a popup appears to '**Make Change in the Hard-Drive**', Click on 'Yes'

(iii) JDK Installed Successfully! (Click on 'Close')

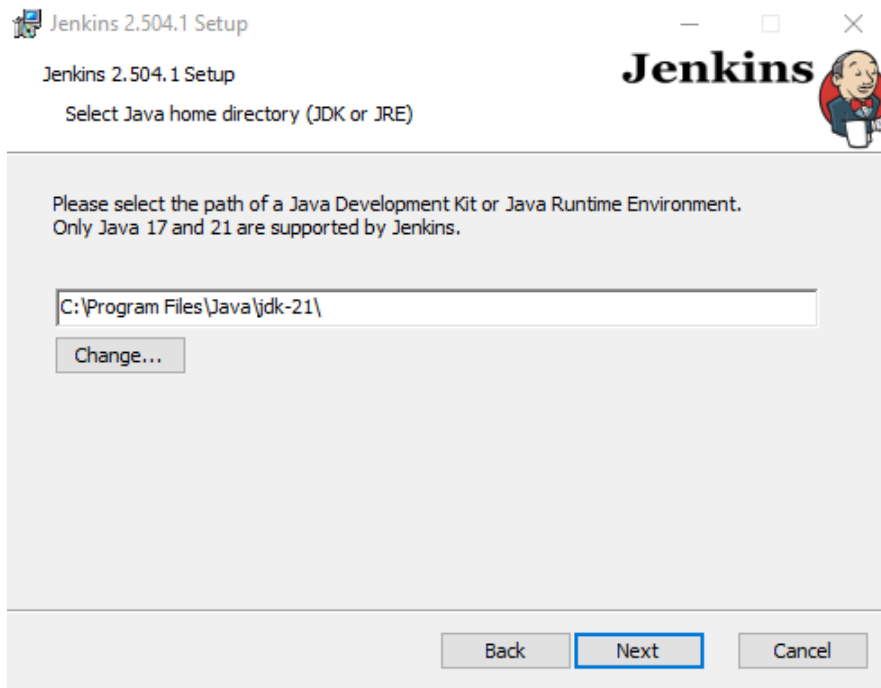


8. Make sure to add the Path of the Java JDK for Jenkins — as highlighted in Step-7 (ii)

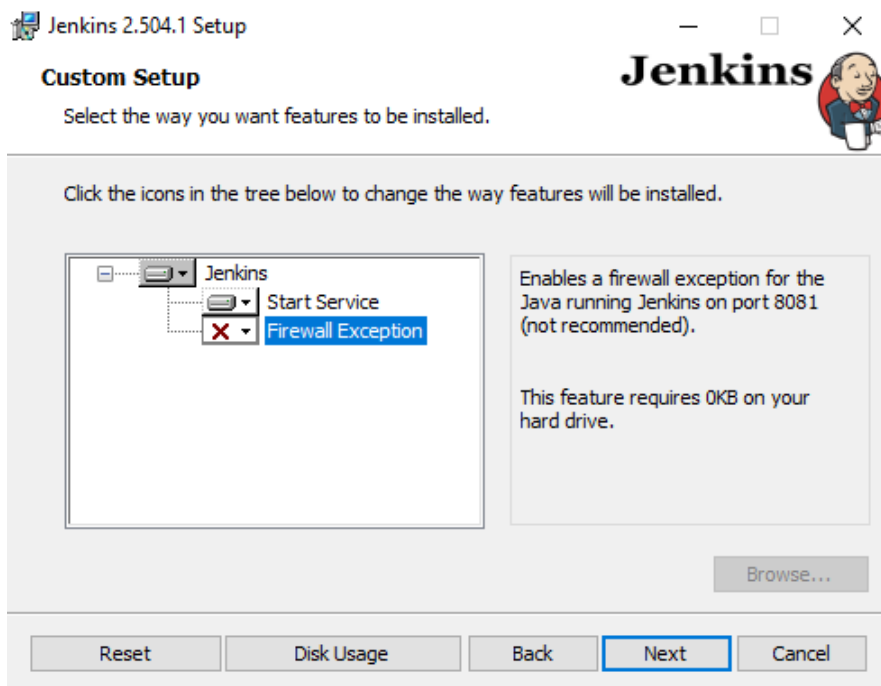
- Click on 'Java' in 'C:\Program Files\', and then Click on 'jdk-21'



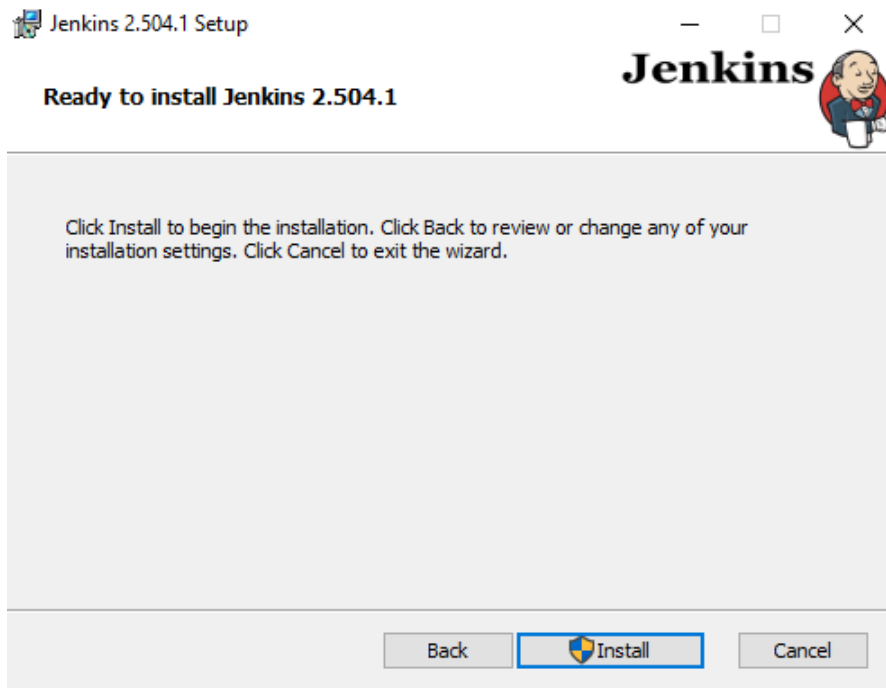
- Click on 'Ok', then Click on 'Next'



9. Click on 'Next'

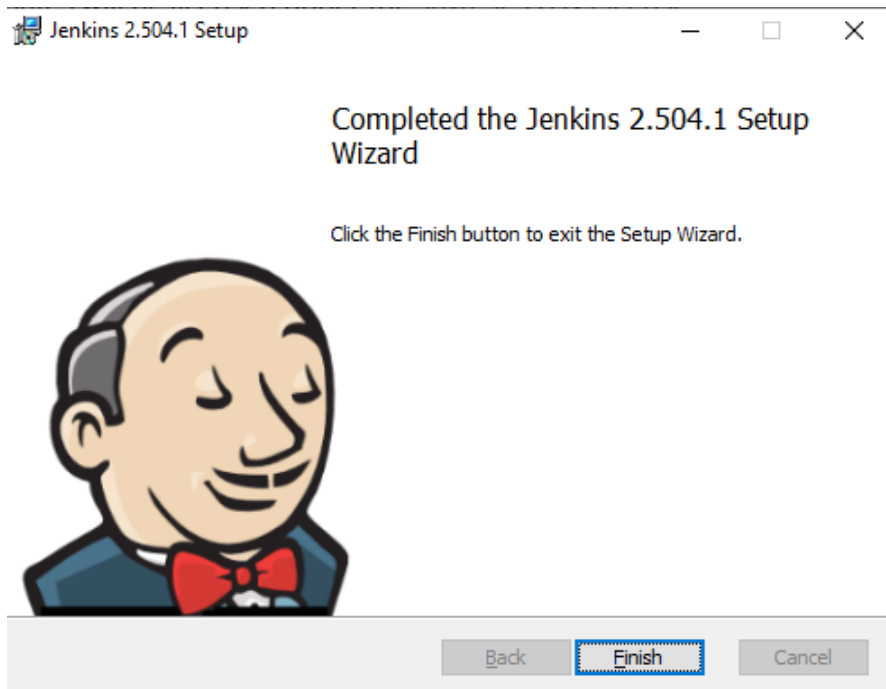


10. Click on 'Install'



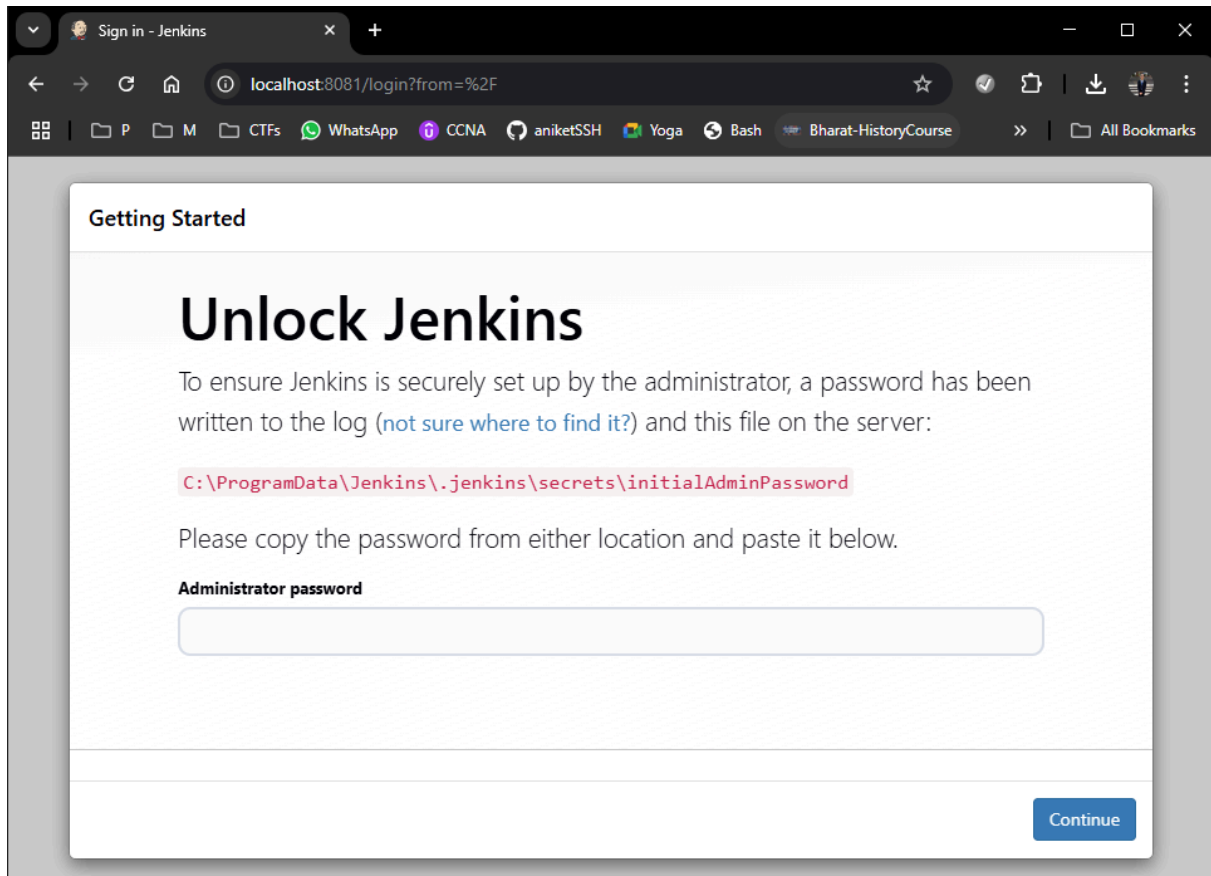
- If a popup appears to **'Make Change in the Hard-Drive'**, Click on 'Yes'

#### 11. Installation Successful! (Click on 'Finish')



#### 12. Admin Account Creation in Jenkins Web-UI

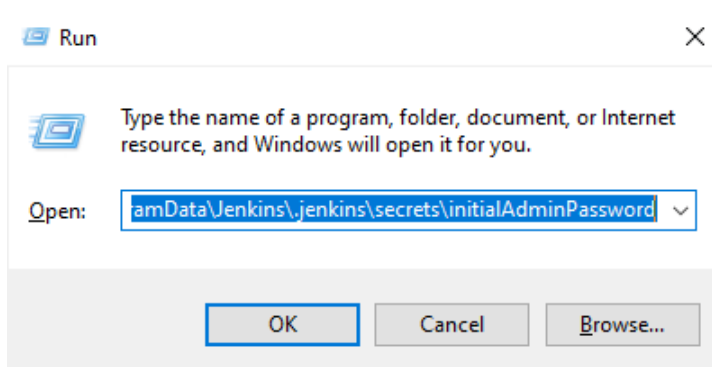
- Type '<http://localhost:8081>', It will ask us to Unlock Jenkins using a Temporary passwords present in the Specified Folder.



- Copy the path, Press 'Win + R' key

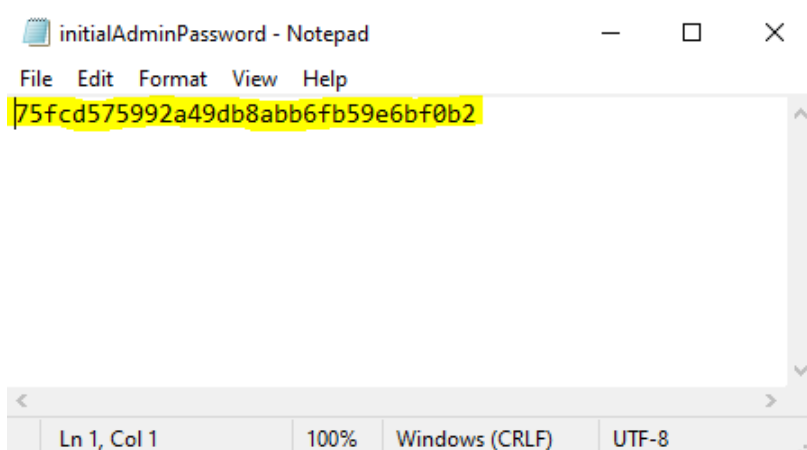
notepad.exe `C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword`

- Click on 'Ok'

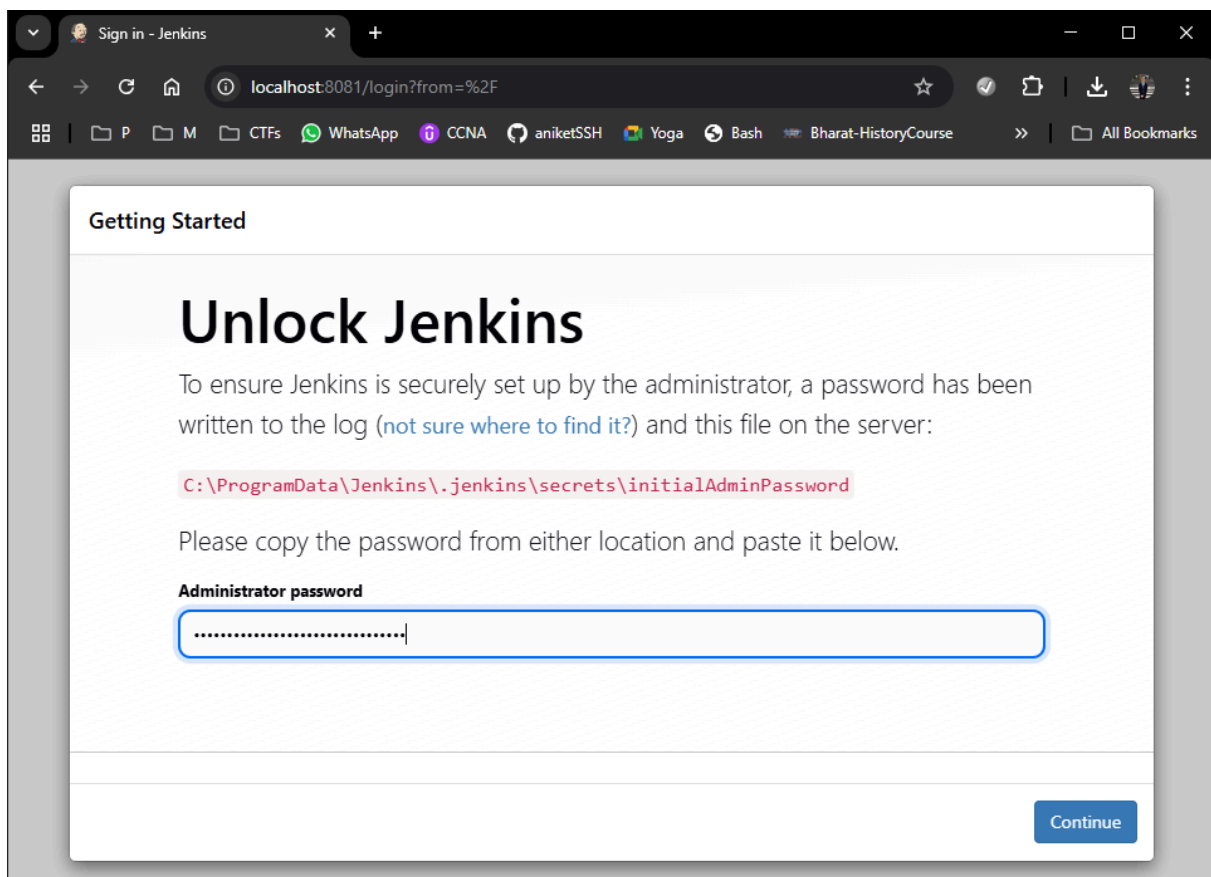


- Copy the password from 'Notepad'

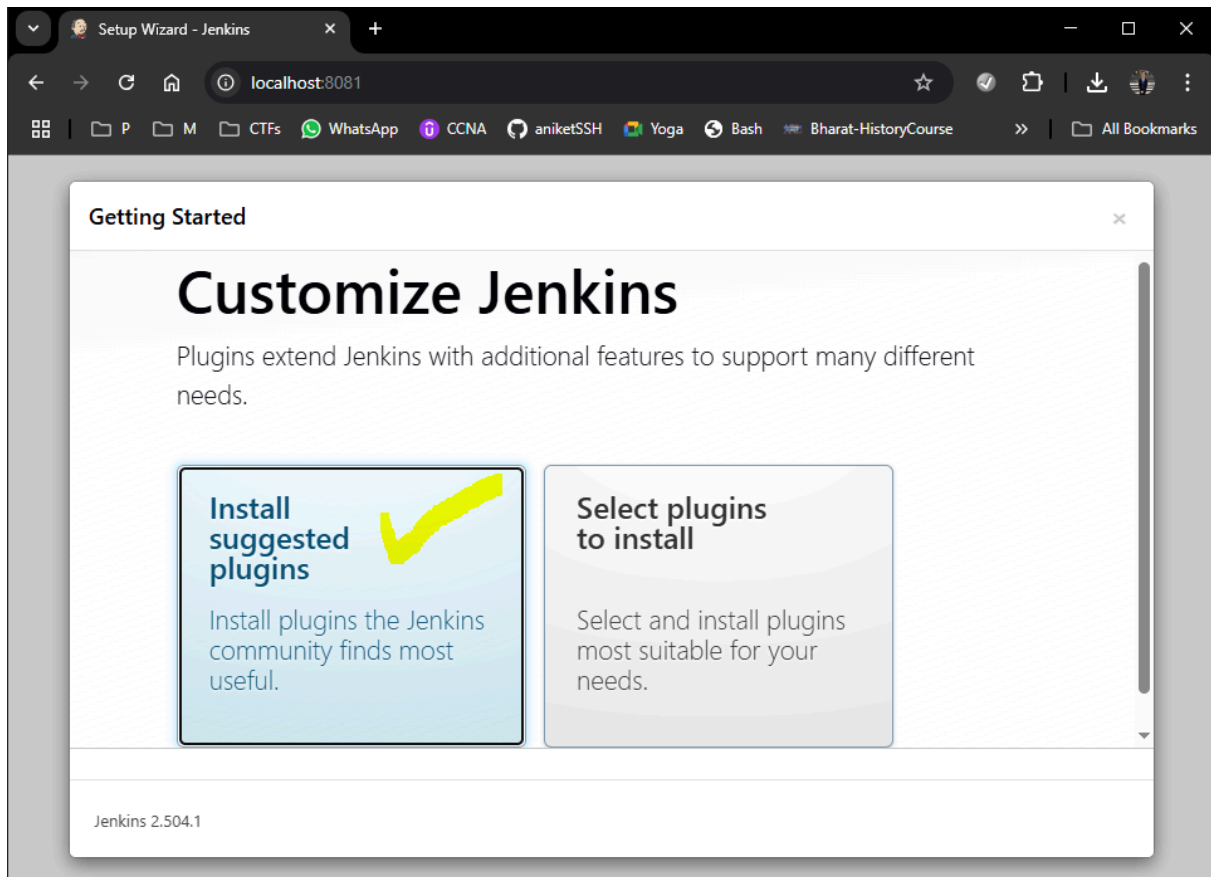




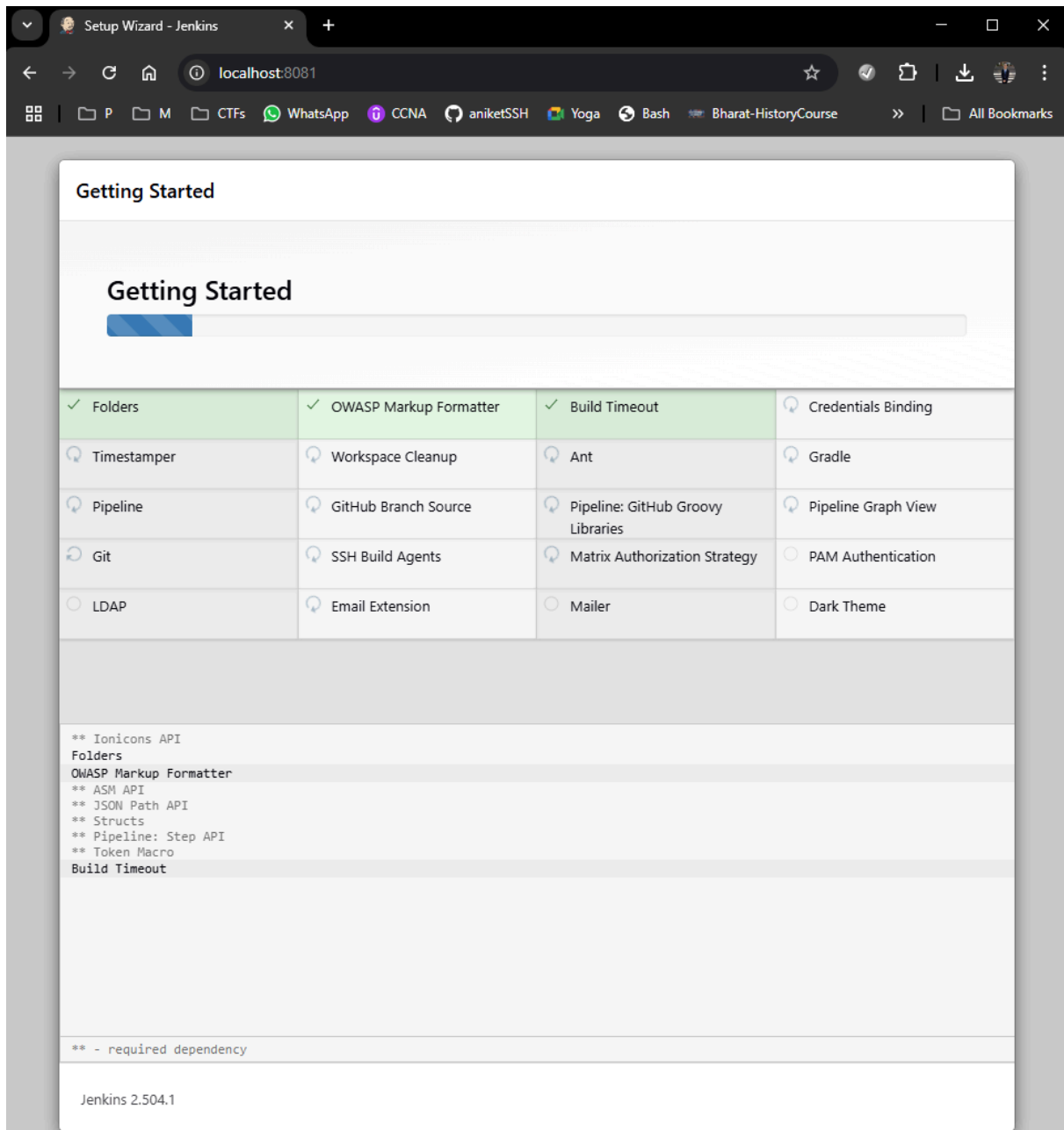
- Paste the password in the Web-UI and Click on 'Continue'



- Click on 'Install Suggested Plugins' Box (This might take a while)



- Plugin Installation in Progress



- Creating 'admin' credentials, let the Password be 'admin' as well ( [X] - Not for Production!)

# Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

- Let the Jenkins URL be the Same as 'Default'

# Instance Configuration

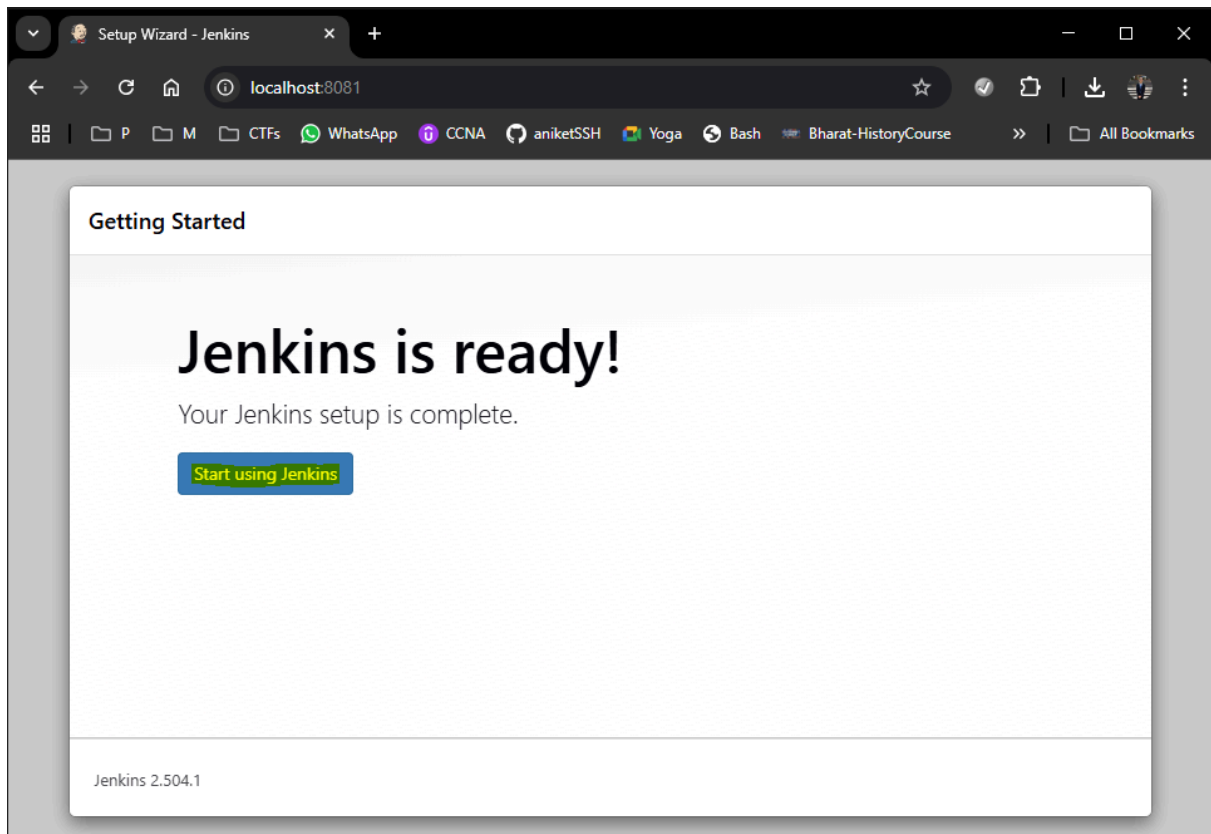
Jenkins URL:

http://localhost:8081/

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD\_URL environment variable provided to build steps.

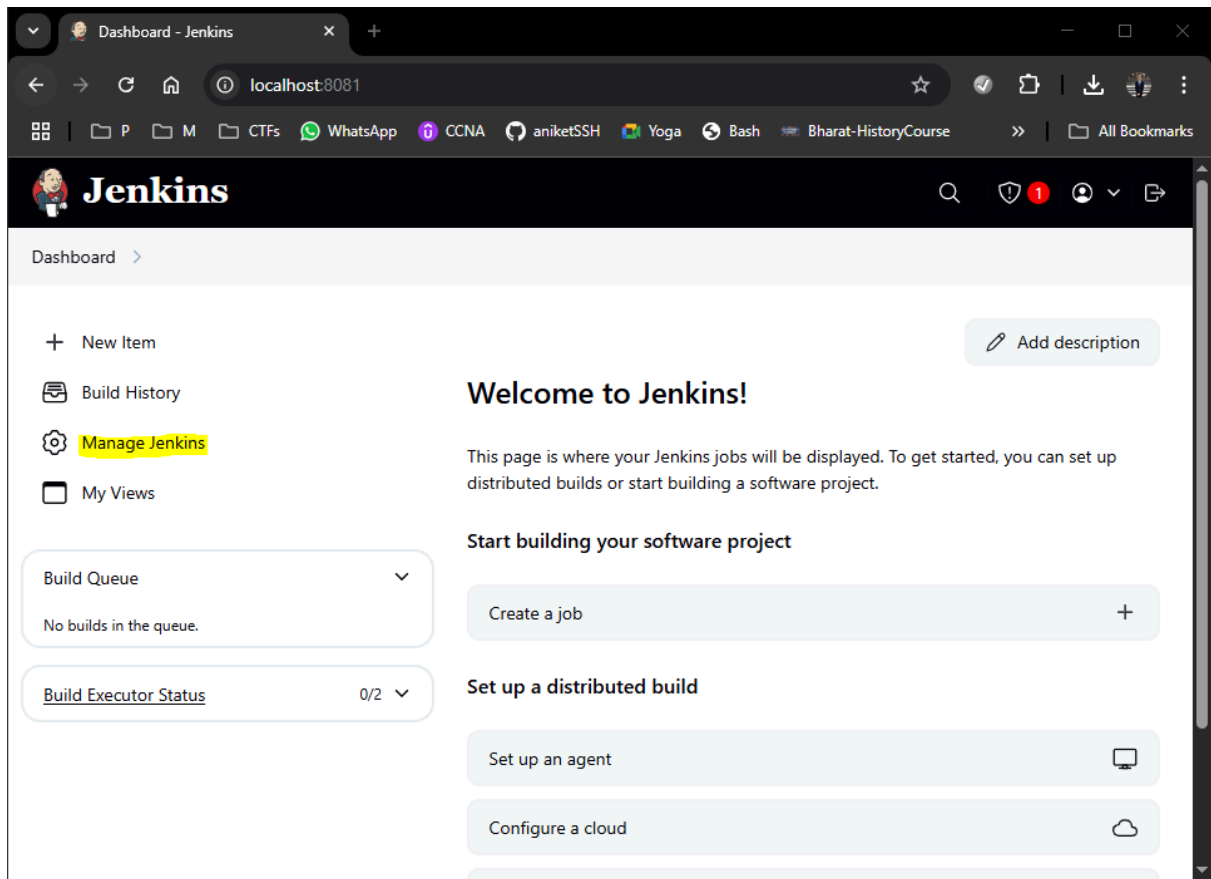
The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

- Jenkins is ready, Click on 'Start using Jenkins'

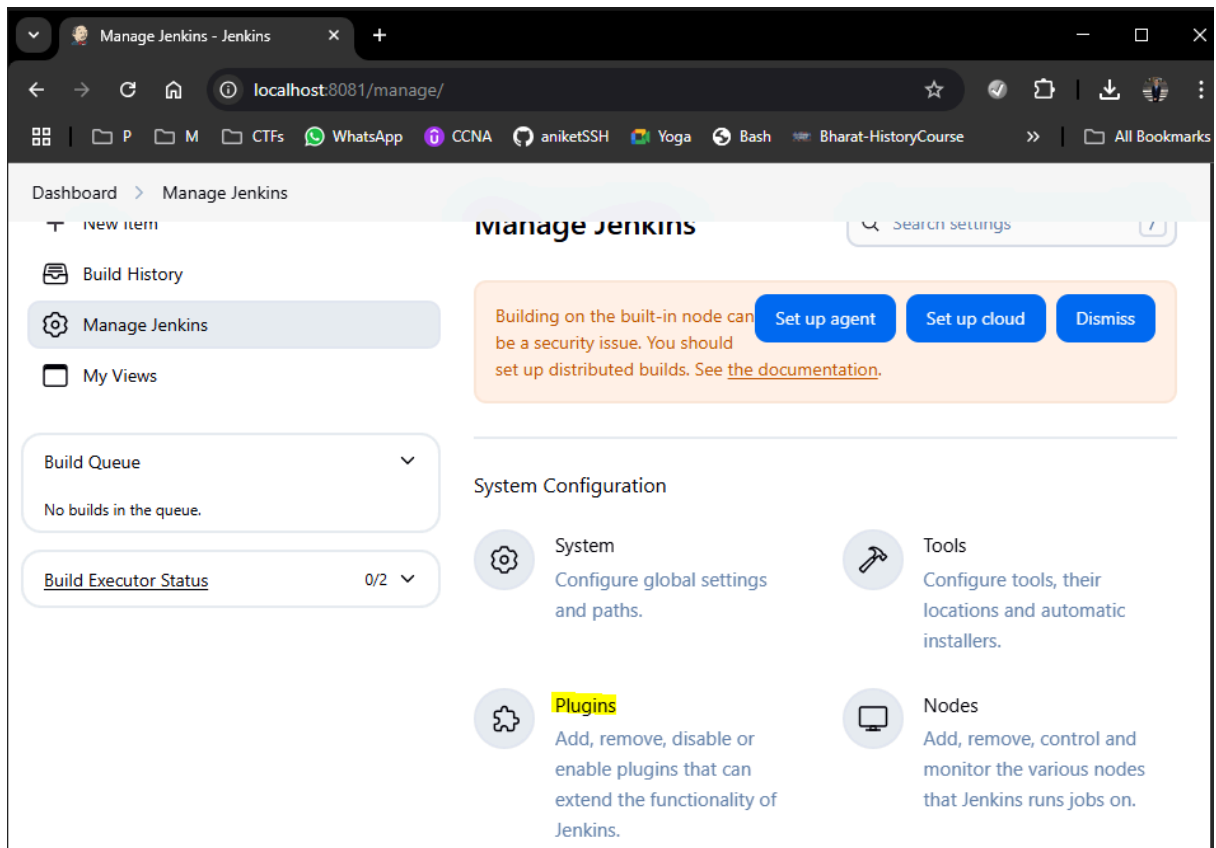


### 13. Install the 'SonarQube Scanner' Plugin in Jenkins

- Click on 'Manage Jenkins'



- Click on 'Plugins'



- Click on 'Available Jenkins', Search for 'SonarQube Scanner' (Click on the Checkbox and Install)



Available plugins - Plugins - Jen x +

localhost:8081/manage/pluginManager/available

Jenkins

Dashboard > Manage Jenkins > Plugins

## Plugins

Q SONAR

Install

Updates

Available plugins

Installed plugins


Advanced settings

Download progress

Install	Name ↓	Released
<input checked="" type="checkbox"/>	<b>SonarQube Scanner 2.18</b> <a href="#">External Site/Tool Integrations</a> <a href="#">Build Reports</a> This plugin allows an easy integration of <a href="#">SonarQube</a> , the open source platform for Continuous Inspection of code quality.	3 mo 22 days ago
<input type="checkbox"/>	<b>Sonar Quality Gates 352.vdcdb_d7994fb_6</b> <a href="#">Library plugins (for use by other plugins)</a> <a href="#">analysis</a> <a href="#">Other Post-Build Actions</a> Fails the build whenever the Quality Gates criteria in the Sonar 5.6+ analysis aren't met (the project Quality Gates status is different than "Passed")	2 mo 27 days ago
<input type="checkbox"/>	<b>Quality Gates 2.5</b>	

- Once the Plugins have the 'Success' Tag — Click on 'Restart' Checkbox



 Jenkins is restarting

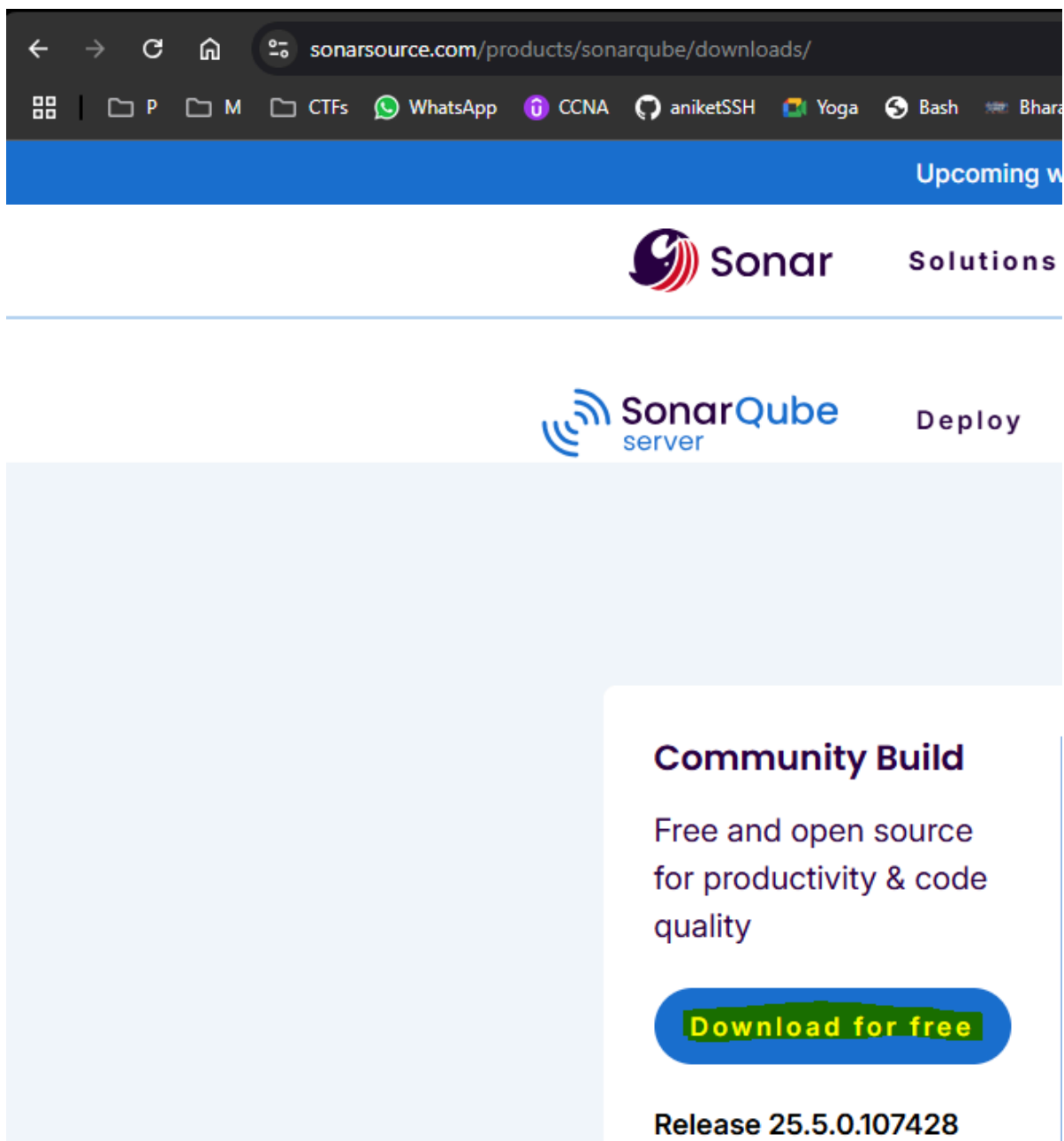
Your browser will reload automatically when Jenkins is ready.

**Safe Restart**  
Builds on agents can usually continue.

- Login back into Jenkins (Username: admin, Password: admin)

## Installation of SonarQube Server

1. Download the 'SonarQube Server' Executable from:  
["https://www.sonarsource.com/products/sonarqube/downloads/"](https://www.sonarsource.com/products/sonarqube/downloads/)
- Under "Community Build" > "Download for free" > "Click on Download Only"  
(Windows - .zip folder)

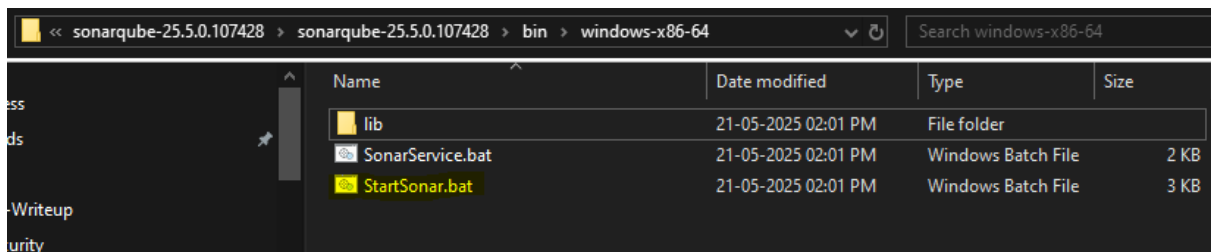


2. Unzip the 'sonarqube-<version>.zip' file to Folder.

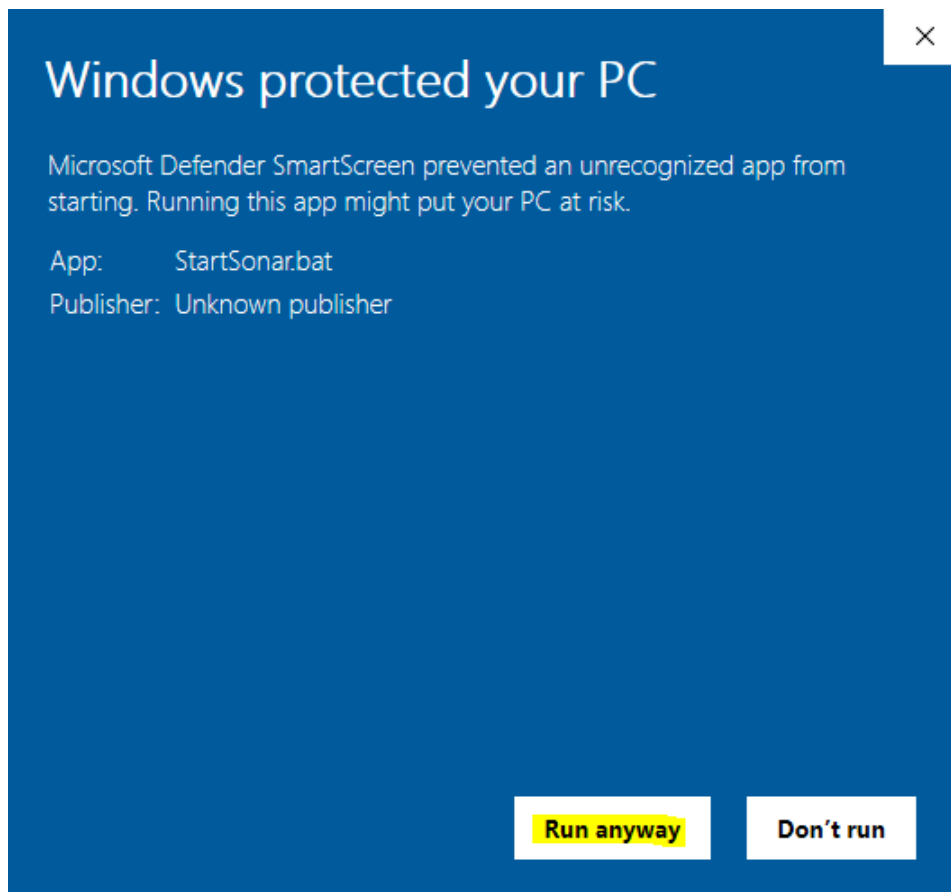
3. Go to Specified Path,

```
sonarqube-25.5.0.107428/  
└─ sonarqube-25.5.0.107428/  
    └─ bin/  
        └─ windows-x86-64/
```

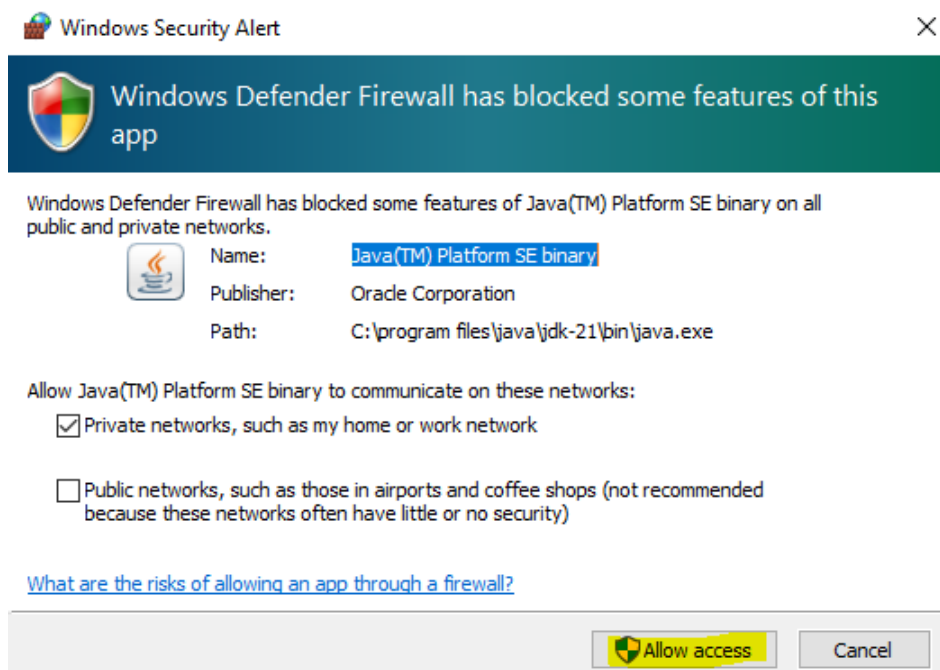
4. Click on 'StartSonar.bat' file, This will execute the SonarQube Server



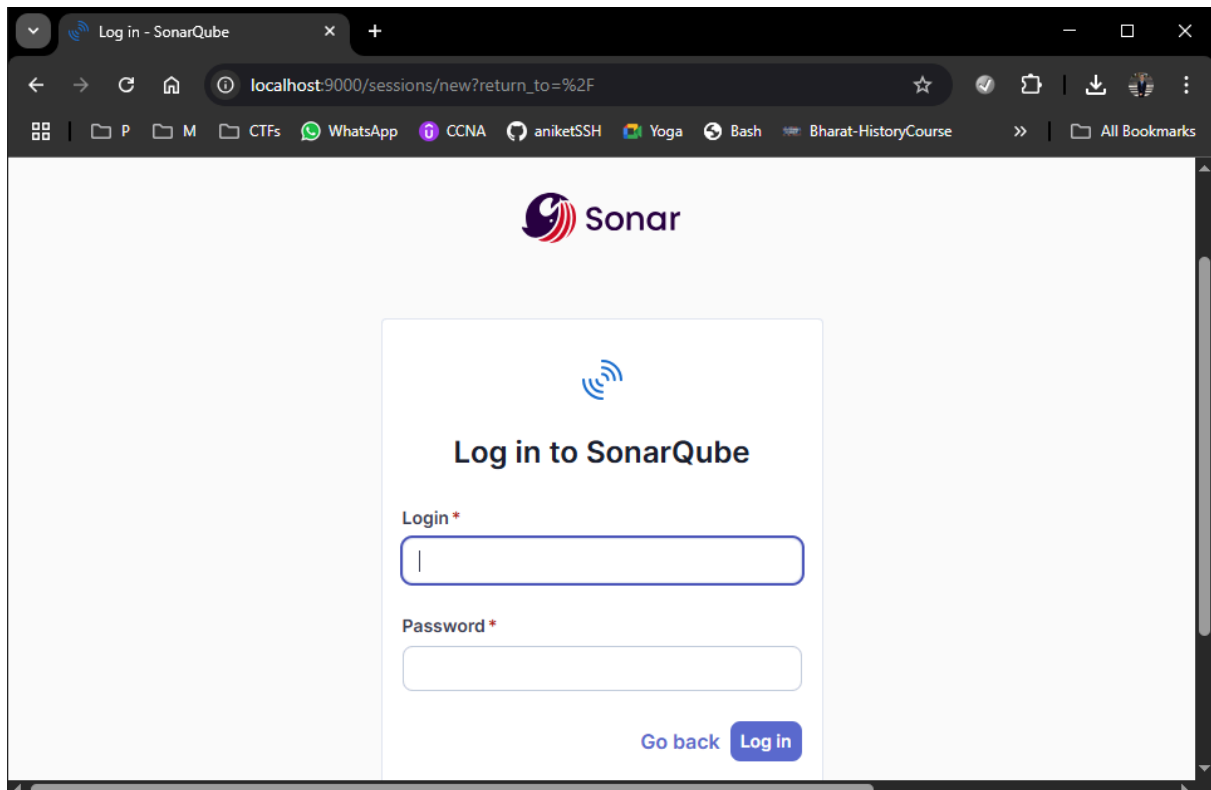
- A popup of "Windows Protected your PC" will appear, Click on "More info", Click on "Run anyway"



- Click on "Allow access" in the Windows Security Alert Popup




4. Go to "<http://localhost:9000>" (Login Page will appear for SonarQube Server)



- Credentials — Username: admin, Password: admin
- Update password for 'admin' user: (Set — New Password: 'Admin@123.456')

## Update your password

 This account should not use the default password.

### Enter a new password

Old Password \*

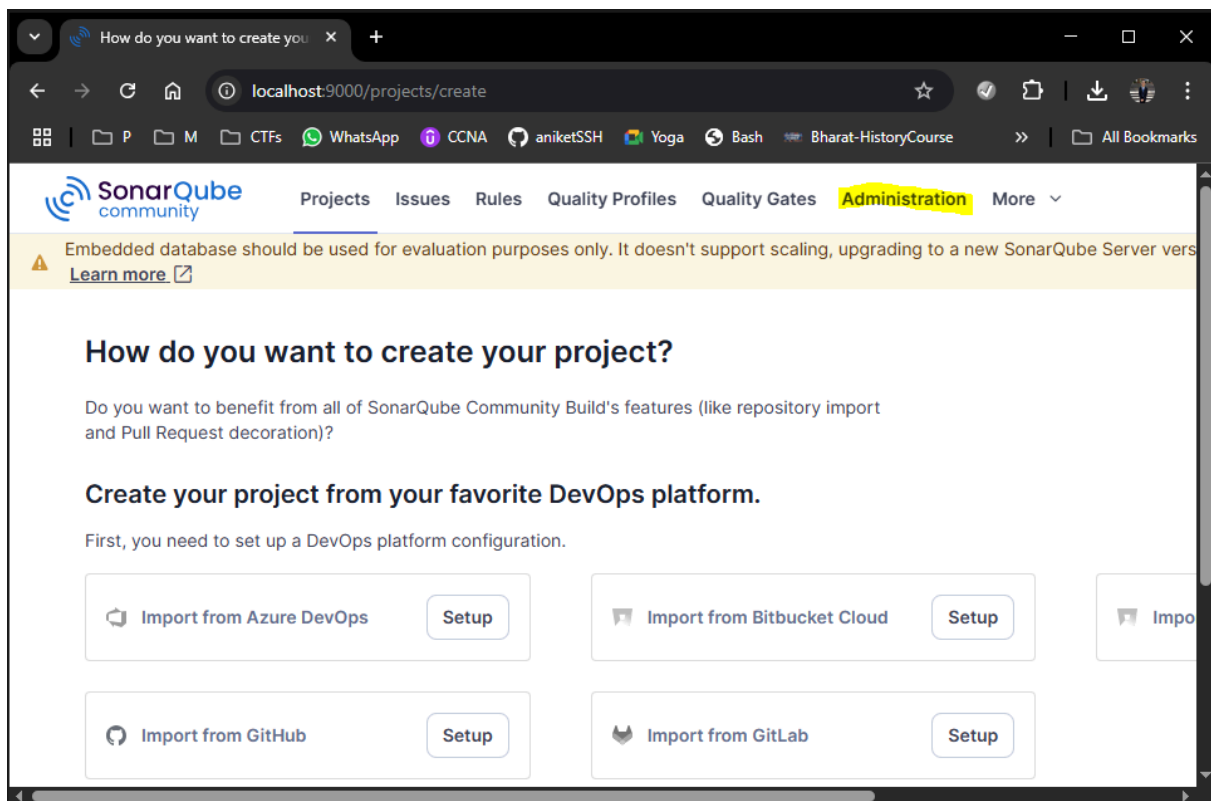
Password \*

Confirm Password \*

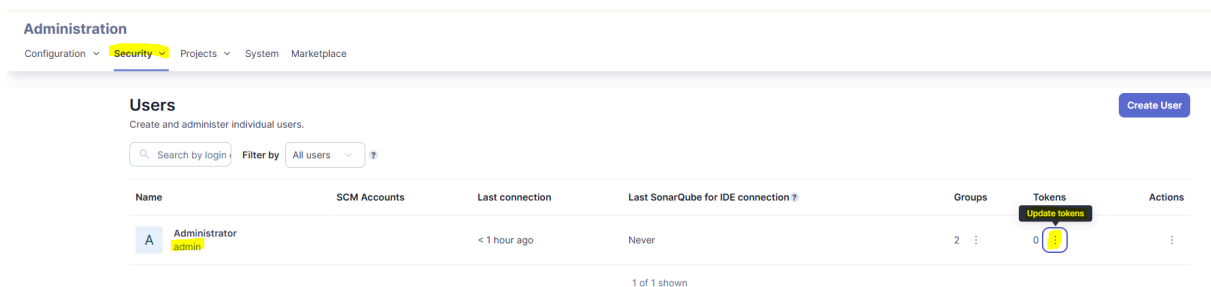
Update

#### 5. Generate Token for SonarQube Server to be used in Jenkins

- Go to the 'Administration' Tab,



- Click on 'Security' tab and then Click on 'Update tokens' (as highlighted in the screenshot)




- Add the Token name as 'jenkins' and set expiration 'Never', Click on Generate Button (Make sure to immediately copy the token and paste it in notepad since it will not be visible again - One Time)

## Tokens of *Administrator*

### Generate Tokens

**Name** **Expires in**

✓ New token "jenkins" has been created. Make sure you copy it now, you won't be able to see it again!


`squ_9da5f116384cf5f594ee57a90ba229c355c5d6aa` 

Name	Type	Project	Last use	Created	Expiration	
jenkins	User		Never	May 21, 2025	–	<input type="button" value="Revoke"/>

## Setting Up the Jenkins-SonarQube Configuration


1. Login into Jenkins Web-UI (<http://localhost:8081>)
2. Click on 'Manage Jenkins' > 'Tools'





 **Jenkins**

Dashboard > Manage Jenkins

+ New Item

 Build History

 **Manage Jenkins**

 My Views

Build Queue

No builds in the queue.


Build Executor Status

0/2


## Manage Jenkins

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).


System Configuration

 System


Configure global settings and paths.

 **Tools**

Configure tools, their locations and automatic installers.

 Clouds

Add, remove, and configure cloud instances to provision agents on-demand.

 Appearance

Configure the look and feel of Jenkins

3. We will now setup the SonnarQube Scanner which will be scanning our code.

- Click on 'Add SonarQube Scanner' (Highlighted in Screenshot)

Dashboard > Manage Jenkins > Tools

### Gradle installations

Add Gradle

### SonarScanner for MSBuild installations

Add SonarScanner for MSBuild

### SonarQube Scanner installations

**Add SonarQube Scanner**

- Add the name as 'sonar' and Click on 'Save'

The screenshot shows the Jenkins configuration page for the 'SonarQube Scanner'. The 'Name' field is set to 'sonar'. The 'Install automatically' checkbox is checked. Under the 'Install from Maven Central' section, the 'Version' dropdown is set to 'SonarQube Scanner 7.1.0.4889'. There is an 'Add Installer' button below this section. At the bottom of the configuration area, there is an 'Add SonarQube Scanner' button. Below the configuration area, there are sections for 'Ant installations' (with an 'Add Ant' button) and 'Maven Toolchains' (with a 'Save' button and an 'Apply' button).

4. We will now add the Credentials as URL of the SonarQube server in the Environment Variable of Jenkins (Manage Jenkins > System > SonarQube Servers [Environment Variables])

[+ New Item](#)[Build History](#)[Manage Jenkins](#)[My Views](#)[Build Queue](#)

No builds in the queue.

[Build Executor Status](#)

0/2

## Manage Jenkins

Building on the built-in node can be a security issue. You

### System Configuration

**System**[Configure global settings and paths.](#)**Clouds**[Add, remove, and configure cloud instances to provision agents on-demand.](#)

- Check the Environment Variable Box under 'SonarQube servers', and Click on 'Add SonarQube'

Dashboard > Manage Jenkins > System >

---

### Global properties

- ☐ Disable deferred wipeout on this node ?
- ☐ Disk Space Monitoring Thresholds
- ☐ Environment variables
- ☐ Tool Locations

---

### SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☒ Environment variables

### SonarQube installations

List of SonarQube installations

**Add SonarQube**

- Add the (name: 'sonar', URL: 'http://localhost:9000', Server Authentication Token)

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☒ Environment variables

### SonarQube installations

List of SonarQube installations

Name ✕

sonar

Server URL

Default is http://localhost:9000

http://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

**+ Add**

Advanced ▾

- Click on the '+ Add' button as Highlighted and Click on 'Jenkins Credential Provider':

(i) Kind: 'Secret text'

(ii) Secret: 'squ\_\*\*\*\*'

(iii) ID: 'sonar'

Jenkins Credentials Provider: Jenkins

### Add Credentials

Domain

Global credentials (unrestricted) ▼

Kind

Secret text ▼

Scope ?

Global (Jenkins, nodes, items, all child items, etc) ▼

Secret

.....

ID ?

sonar

Description ?

sonar

Cancel Add

- Click on 'Add'
- Select from the Drop-Down Menu in 'Server Authentication Token' → 'sonar'

Dashboard > Manage Jenkins > System >

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☒ Environment variables

SonarQube installations

List of SonarQube installations

Name

sonar

Server URL

Default is http://localhost:9000

http://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

sonar

+ Add

Advanced ▼

Add SonarQube

Metrics

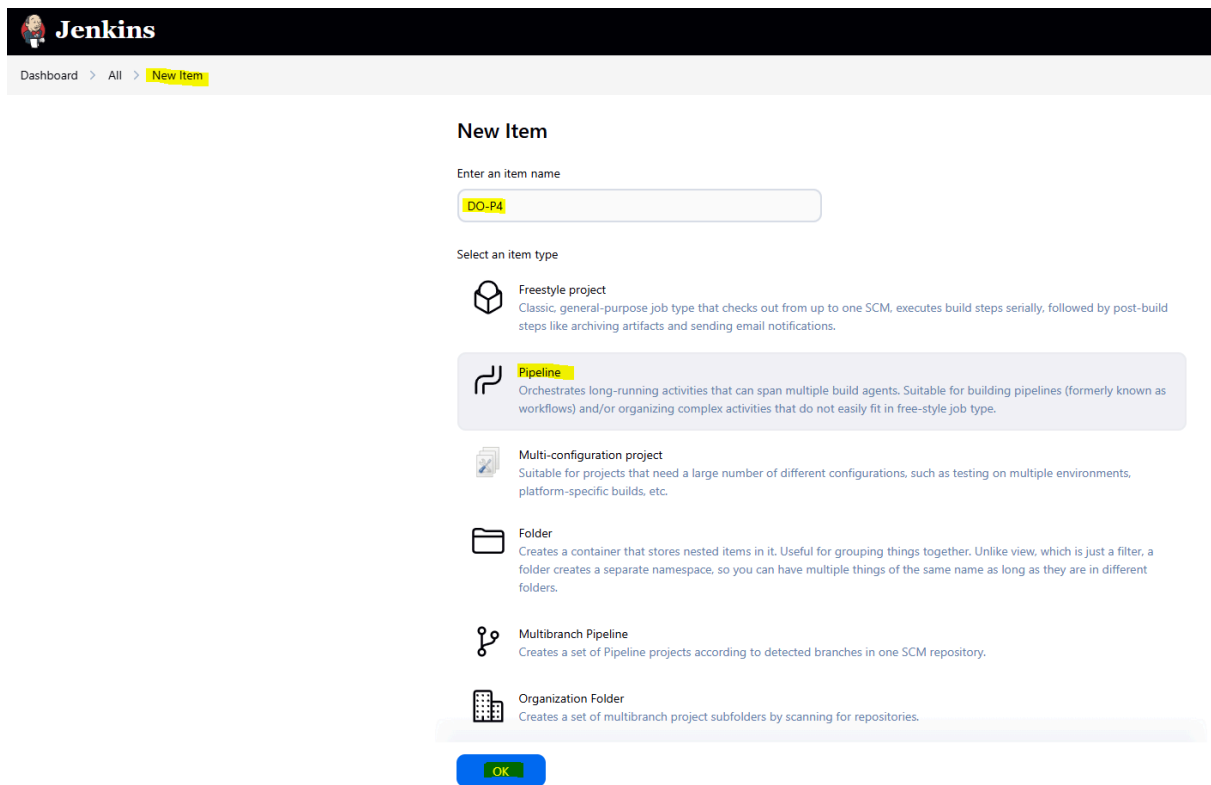
Save Apply

- Click on 'Save'

## Setting Up the Jenkins Pipeline

## 1. Initial Steps for Creating a Jenkins Pipeline

- Click on 'New Item'
- Enter the Name of the Item 'DO-P4'
- Click on 'Pipeline'
- Click on 'Ok'



**Jenkins**







Dashboard > All > **New Item**

### New Item

Enter an item name

**DO-P4**

Select an item type

-  **Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
-  **Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
-  **Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
-  **Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
-  **Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in one SCM repository.
-  **Organization Folder**  
Creates a set of multibranch project subfolders by scanning for repositories.

**OK**

## 2. Copy the Code below for the Pipeline Script

```
pipeline {
  agent any

  stages {
    stage('Prepare Workspace') {
      steps {
        // Copy local code using Windows command
        bat 'xcopy /s /e /y "C:\\Users\\Admin\\Desktop\\helloworld.py" .'
      }
    }

    stage('Code Analysis') {
```

```

environment {
    scannerHome = tool 'sonar'
}
steps {
    script {
        withSonarQubeEnv('sonar') {
            bat """
                ${scannerHome}\\bin\\sonar-scanner.bat ^
                -Dsonar.projectKey=your_project_key ^
                -Dsonar.projectName=your_project_name ^
                -Dsonar.projectVersion=1.0 ^
                -Dsonar.sources=.
            """
        }
    }
}

```

Dashboard > DO-P4 > Configuration

## Configure

☐ Trigger builds remotely (e.g., from scripts) ?

**General**

**Triggers**

Pipeline

Advanced

### Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition

Pipeline script

Script ?

```

1 pipeline {
2   agent any
3
4   stages {
5     stage('Prepare Workspace') {
6       steps {
7         // Copy local code using Windows command
8         bat 'xcopy /s /e /y "C:\Users\anike\Desktop\helloworld.py" .'
9       }
10    }
11
12    stage('Code Analysis') {
13      environment {
14        scannerHome = tool 'sonar'
15      }
16    }
17  }
18 }

```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

**Save** Apply

**NOTE: \*\* Make Sure the code path is correct (Specially Username) \*\***

- Click on 'Save'

3. Make a python file: 'helloworld.py' in the Path Specified in the Pipeline Script (Highlighted)

The screenshot shows the Visual Studio Code interface. The file explorer on the left shows a file named 'helloworld.py' on the Desktop. The main editor area shows the content of 'helloworld.py':

```

1 print("Hello World!")

```

The bottom panel shows the 'OUTPUT' window with the following text:

```

[Running] python -u "c:\Users\anike\Desktop\helloworld.py"
Hello World!

[Done] exited with code=0 in 0.084 seconds

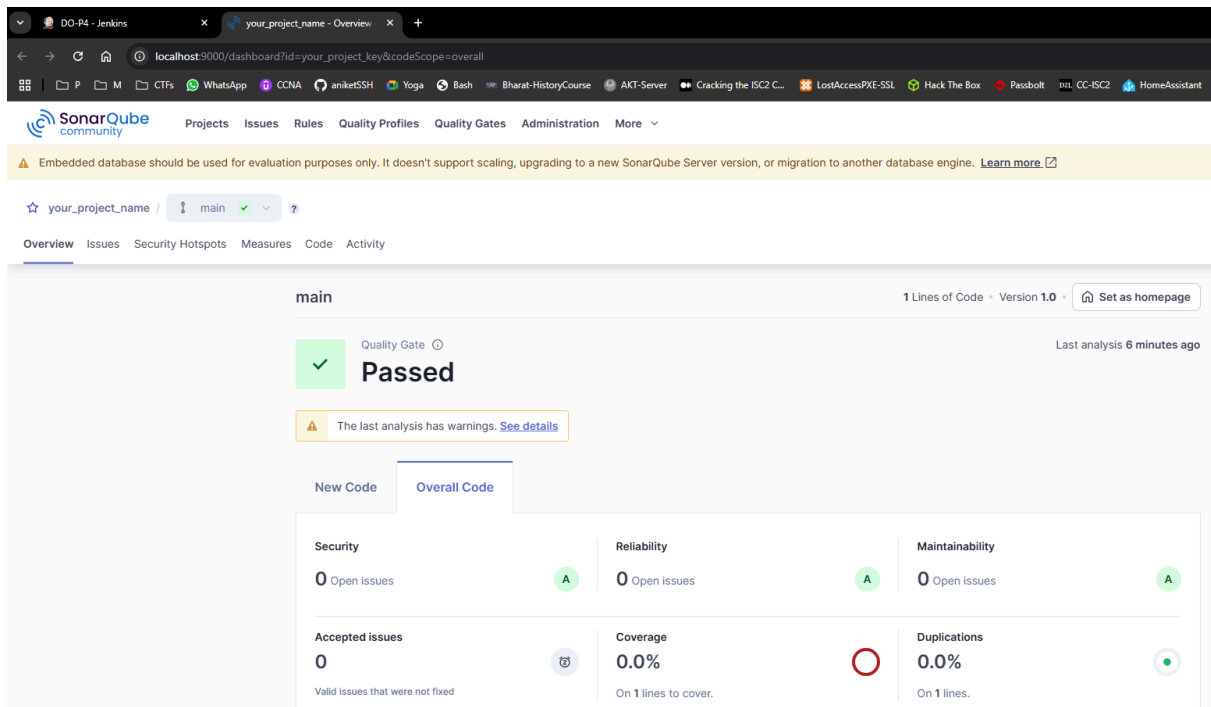
```



4. For running the Jenkins Pipeline, Click on 'Build Now'

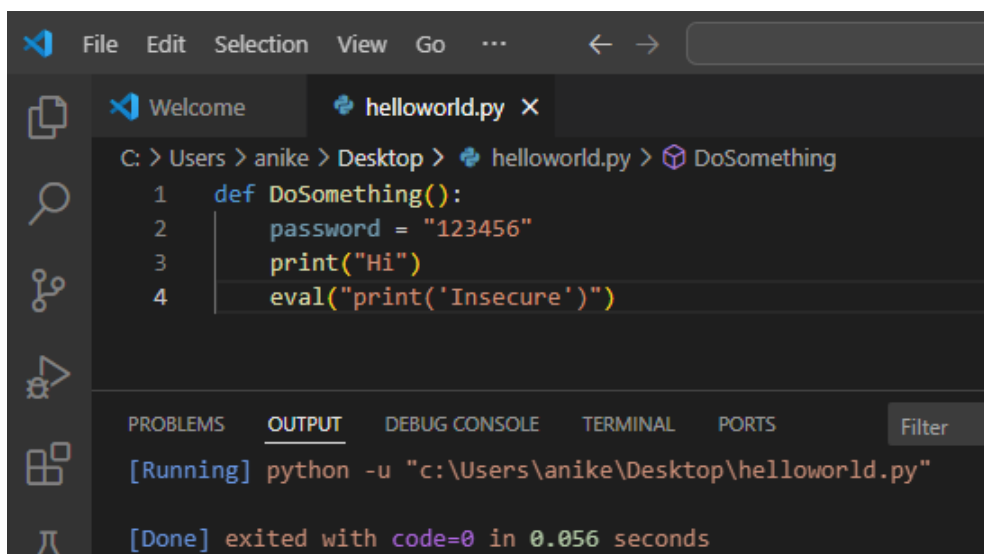
The screenshot displays the Jenkins web interface for a job named 'DO-P4'. The browser address bar shows 'localhost:8081/job/DO-P4/'. The Jenkins logo and 'Dashboard > DO-P4 >' breadcrumb are visible. On the left sidebar, the 'Build Now' button is highlighted in yellow. The main content area shows a green checkmark icon next to 'DO-P4'. Below this, the 'SonarQube Quality Gate' section displays 'your\_project\_name' as 'N/A' and 'server-side processing' as 'In progress'. The 'Permalinks' section lists four links: 'Last build (#1), 21 sec ago', 'Last stable build (#1), 21 sec ago', 'Last successful build (#1), 21 sec ago', and 'Last completed build (#1), 21 sec ago'. At the bottom, the 'Builds' section shows a single build (#1) completed at 3:14 PM.

- Right-Click on SonarQube as Highlighted in the above screenshot in a new tab.
- Once the Pipeline is executed, Click on 'SonarQube' for checking the Code Analysis Report.



- To Show the other testcase for SonarQube Analysis, Change the code in 'helloworld.py'

```
def DoSomething():
    password = "123456"
    print("Hi")
    eval("print('Insecure')")
```



- Now Re-Run the Jenkins Pipeline (After saving the code in 'helloworld.py')

(i) Click on 'Build' again

The screenshot shows the Jenkins web interface for a pipeline named 'DO-P4'. The left sidebar contains a list of actions: Status, Changes, Build Now (highlighted in yellow), Configure, Delete Pipeline, SonarQube (highlighted in yellow), Stages, Rename, and Pipeline Syntax. The main content area shows the pipeline status as 'DO-P4' with a green checkmark. Below this, the 'SonarQube Quality Gate' section displays 'your\_project\_name' as 'Failed' (in a red box) and 'server-side processing' as 'Success' (in a green box). A large yellow 'X' is drawn over the 'Failed' status. The 'Permalinks' section lists four links: 'Last build (#2), 17 sec ago', 'Last stable build (#2), 17 sec ago', 'Last successful build (#2), 17 sec ago', and 'Last completed build (#2), 17 sec ago'. At the bottom, a 'Builds' section shows a list of builds: '#2 3:25 PM' (highlighted in yellow) and '#1 3:14 PM', both with green checkmarks and expandable icons.

(ii) Right-Click on SonarQube as Highlighted in the above screenshot in a new tab.

(iii) Once the Pipeline is executed, Click on 'SonarQube' for checking the Code Analysis Report.

DO-P4 - Jenkins x your\_project\_name - Overview x +

localhost:9000/dashboard?id=your\_project\_key

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More

Embedded database should be used for evaluation purposes only. It doesn't support scaling, upgrading to a new SonarQube Server version, or migration to another database engine. [Learn more](#)

your\_project\_name / main

Overview Issues Security Hotspots Measures Code Activity

main 4 Lines of Code · Version 1.0 Set as homepage Take the Tour

Quality Gate **Failed** Last analysis 1 minute ago

The last analysis has warnings. [See details](#)

New Code (2 failed) Overall Code

New Code: Since May 21, 2025 Started 11 minutes ago

Some Quality Gate conditions on New Code were ignored because of the small number of New Lines

2 conditions failed

2 Issues

0.0% Security Hotspots Reviewed is less than 100%

Fix issues before they fail your Quality Gate

New issues **FAILED** 2 Required = 0

Accepted issues 0 Valid issues that were not fixed

Coverage 0.0% Required ≥ 80.0% On 4 New Lines to cover.

Duplications 0.0% Required ≤ 3.0% On 4 New Lines.

**\*\* END OF PROGRAM - 04 [DEV\_OPS\_LAB] \*\***