**JENKINS-DOCKER-STREAMLIT-SONARCLOUD**

**Q. Jenkins-Docker- STREAMLIT-SONARCLOUD**

**1. Go to Docker Desktop Terminal:**

**Run Command: docker pull jenkins/Jenkins**

**2. On the same terminal run: docker run jenkins/jenkins**

**A screenshot of a computer

Description automatically generated**

**3. Go to localhost:8080(For Jenkins) and enter your username(Dakshita) and password(universal)**

A screenshot of a computer

Description automatically generated

**4. Generate a Sonar Cloud Token**

* **On SonarCloud-Go to My Account on Top Right Corner**

**A screenshot of a computer

Description automatically generated**

* **On Security-> give a token name and Generate Token**

**A screenshot of a security account

Description automatically generated**

**Token: 2673a342fc3ad50f47135dbd9fd8547890c39e05**

1. **Go to Jenkins-> Manage Jenkins-> Plugins->SonarQube->Let it install the plugin and restart**

**A screenshot of a chat

Description automatically generated**

1. **Sign-in Again**

**A login screen with blue and white text

Description automatically generated**

1. **Click on “Go back to the top page”**

**A screenshot of a computer error

Description automatically generated**

1. **Check If the pluggin is installed correctly**

**A screenshot of a computer

Description automatically generated**

1. **Now we want to integrate our SonarCloud Token With Jenkins**

* **Go to manage Jenkins-> Credentials**

**A screenshot of a computer

Description automatically generated**

**A close-up of a computer screen

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

* **Go to Manage Jenkins-> System-> Scroll Down to SonarCube Servers**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

* **Click on Apply and Save**

1. **To build a pipeline-> Click on New Item**

**A screenshot of a computer

Description automatically generated**

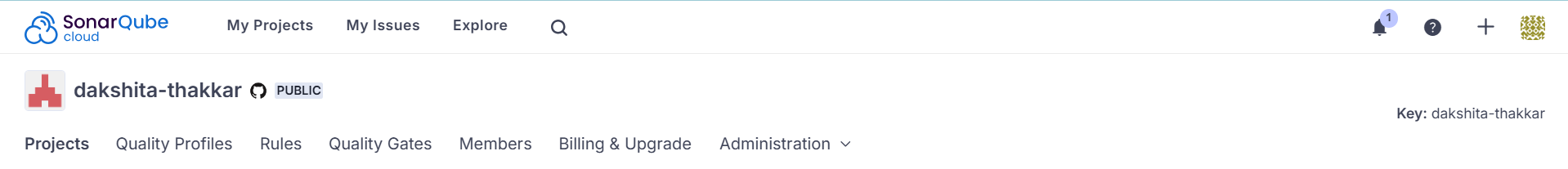
* **Give an appropriate Pipeline Name**

**A screenshot of a computer

Description automatically generated**

* **For Organization Name: Go to My Projects-> Click on Your name**A screenshot of a computer

  Description automatically generated
* **Organization Key->Top right Corner-> dakshita-thakkar**

****

* **For Project key-> Click on the project->Information**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**First Pipeline:**

pipeline {

agent any

environment {

SONAR\_TOKEN = credentials('SonarCloud-Token') // The ID of your SonarCloud token in Jenkins

}

stages {

stage('Checkout Code') {

steps {

git branch: 'master', url: 'https://github.com/dakshita-thakkar/streamlit-github.git'

}

}

stage('Install Dependencies') {

steps {

sh '''

python3 -m venv venv

source venv/bin/activate

pip install --upgrade pip

pip install -r requirements.txt

'''

}

}

stage('Run SonarCloud Analysis') {

steps {

sh '''

curl -sSLo sonar-scanner.zip https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.8.0.2856-linux.zip

unzip sonar-scanner.zip

./sonar-scanner-4.8.0.2856-linux/bin/sonar-scanner \

-Dsonar.projectKey=dakshita-thakkar\_streamlit-github \

-Dsonar.organization=dakshita-thakkar \

-Dsonar.host.url=https://sonarcloud.io \

-Dsonar.login=${SONAR\_TOKEN}

'''

}

}

}

**Final Pipeline:**

pipeline {

agent any

environment {

SONAR\_TOKEN = credentials('SonarCloud-Token') // The ID of your SonarCloud token in Jenkins

SONAR\_SCANNER\_HOME = '/var/lib/jenkins/workspace/Ubuntu-SonarCloud/sonar-scanner-4.8.0.2856-linux' // Correct path

PATH = "${SONAR\_SCANNER\_HOME}/bin:${env.PATH}" // Correct the PATH variable

}

stages {

stage('Checkout Code') {

steps {

git branch: 'master', url: 'https://github.com/dakshita-thakkar/streamlit-github.git'

}

}

stage('Install Dependencies') {

steps {

sh '''#!/bin/bash

python3 -m venv venv

. venv/bin/activate # Use '.' instead of 'source' for compatibility with sh

pip install --upgrade pip

pip install -r requirements.txt

'''

}

}

stage('Run SonarCloud Analysis') {

steps {

sh '''#!/bin/bash

# Ensure the correct path is in the environment

echo "Current PATH: $PATH"

sonar-scanner \

-Dsonar.projectKey=dakshita-thakkar\_streamlit-github \

-Dsonar.organization=dakshita-thakkar \

-Dsonar.host.url=https://sonarcloud.io \

-Dsonar.login=${SONAR\_TOKEN}

'''

}

}

}

}

**Renu Pipeline**

pipeline {

agent any

environment {

SONAR\_TOKEN = credentials('sonarcloud-token') // The ID of your SonarCloud token in Jenkins

}

stages {

stage('Checkout Code') {

steps {

git branch: 'main', url: 'https://github.com/renurrao/DevOps-Streamlit.git'

}

}

stage('Install Dependencies') {

steps {

sh '''

python3 -m venv venv

. venv/bin/activate

pip install --upgrade pip

pip install -r requirements.txt

'''

}

}

stage('Run SonarCloud Analysis') {

steps {

sh '''

curl -sSLo sonar-scanner.zip https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.8.0.2856-linux.zip

unzip sonar-scanner.zip

./sonar-scanner-4.8.0.2856-linux/bin/sonar-scanner \

-Dsonar.projectKey=renurrao\_DevOps-Streamlit \

-Dsonar.organization=renurrao \

-Dsonar.host.url=https://sonarcloud.io \

-Dsonar.login=${SONAR\_TOKEN}

'''

}

}

}

}

}

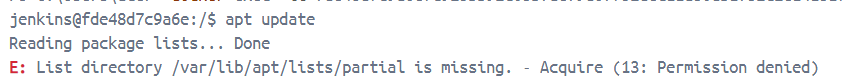
1. **We need to install Python and JDK 17 on our Container**

* **On Docker Terminal, run:**

**Command: docker exec -it --user root <container-id> bash**

**apt update**

**apt install -y python3 python3-venv**

****

**A screenshot of a computer program

Description automatically generated**

* **After installing python, restart the container: docker restart <container\_id>**

****

* **To install jdk 17, run: docker exec -it --user root <container-id> /bin/bash**

**apt update**

**apt install -y openjdk-17-jdk**

**A screenshot of a computer program

Description automatically generated**

**A close up of a code

Description automatically generated**

**JAVA Version Conflict Error:**

**A screenshot of a computer

Description automatically generated**

**Sonar Scanner Path Error**

**A screenshot of a computer

Description automatically generated**

**SonarCube Path Error**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**