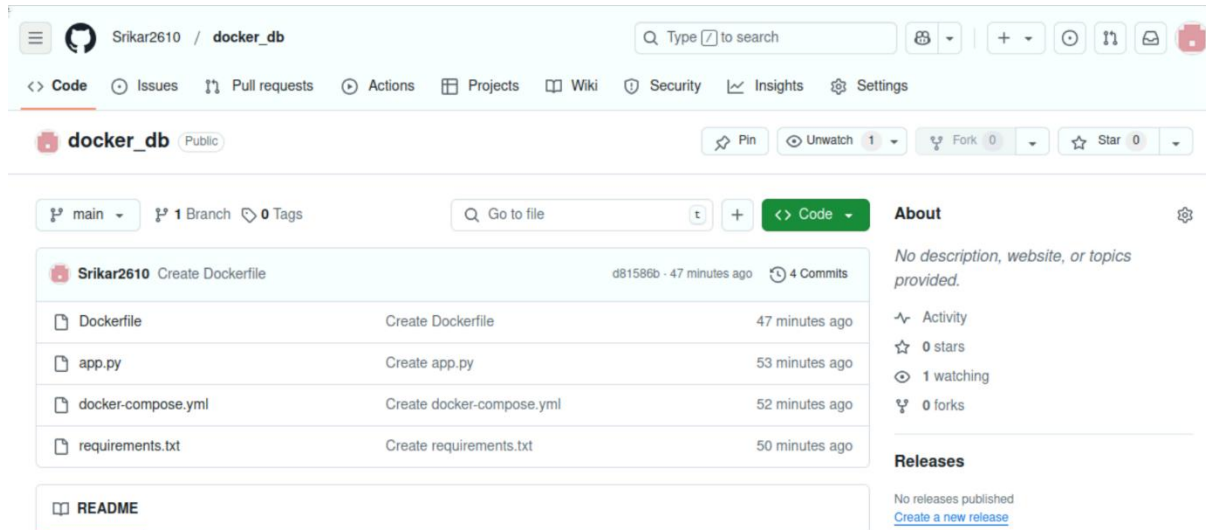


Multi-Container Flask Application with PostgreSQL Using Docker Compose

Step 1: Create a GitHub Repository and add files into it



Step 2: Build and Start the Containers

```
master@master-vm:~/Desktop$ docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~22.04.1
master@master-vm:~/Desktop$ sudo apt install docker-compose
[sudo] password for master:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-attr python3-distutils python3-docker python3-dockerpty
  python3-doccopt python3-dotenv python3-jjsonschema python3-pysistent
  python3-setuptools python3-texttable python3-websocket
Suggested packages:
  python-attr-doc python-jjsonschema-doc python-setuptools-doc
The following NEW packages will be installed:
  docker-compose python3-attr python3-distutils python3-docker
  python3-dockerpty python3-doccopt python3-dotenv python3-jjsonschema
  python3-pysistent python3-setuptools python3-texttable python3-websocket
0 upgraded, 12 newly installed, 0 to remove and 4 not upgraded.
Need to get 911 kB of archives.
After this operation, 4,842 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 python3-distutils all 3.10.8-1~22.04 [139 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 python3-websocket
```

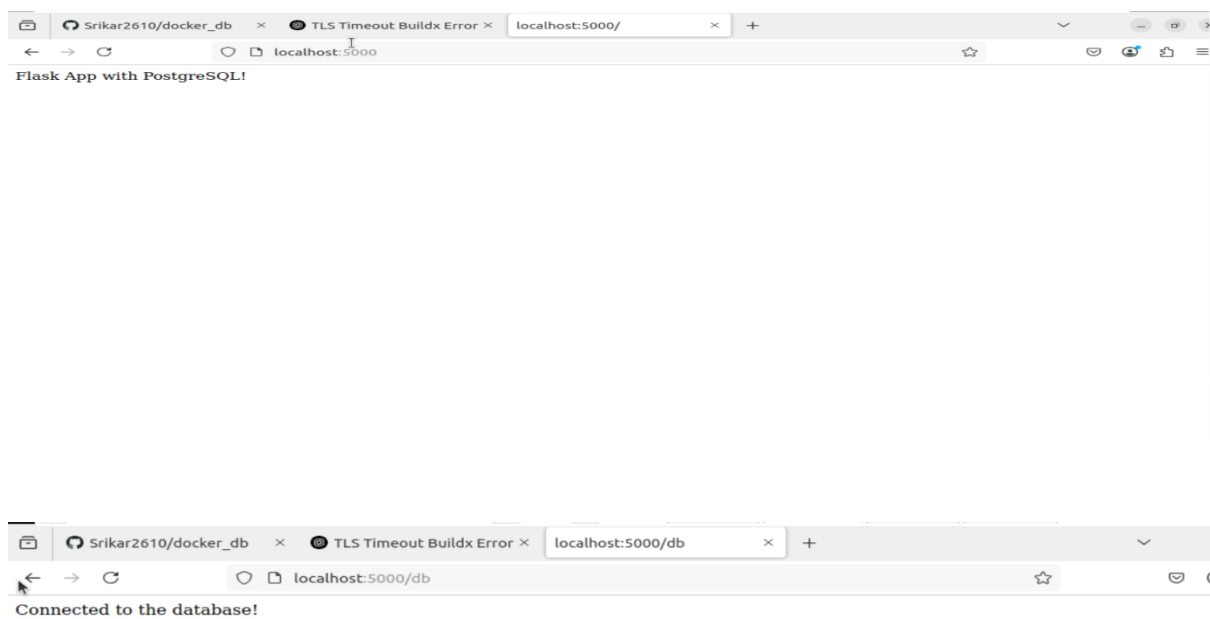
Step 3: Verify the running Container

```
master@master-vm:~/Desktop/docker_db$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
ea5a54ae0d3e	docker_db_web	"python app.py"	56 minutes ago	Up 56 m
004f25a9aa8f	postgres	"docker-entrypoint.s..."	56 minutes ago	Up 56 m

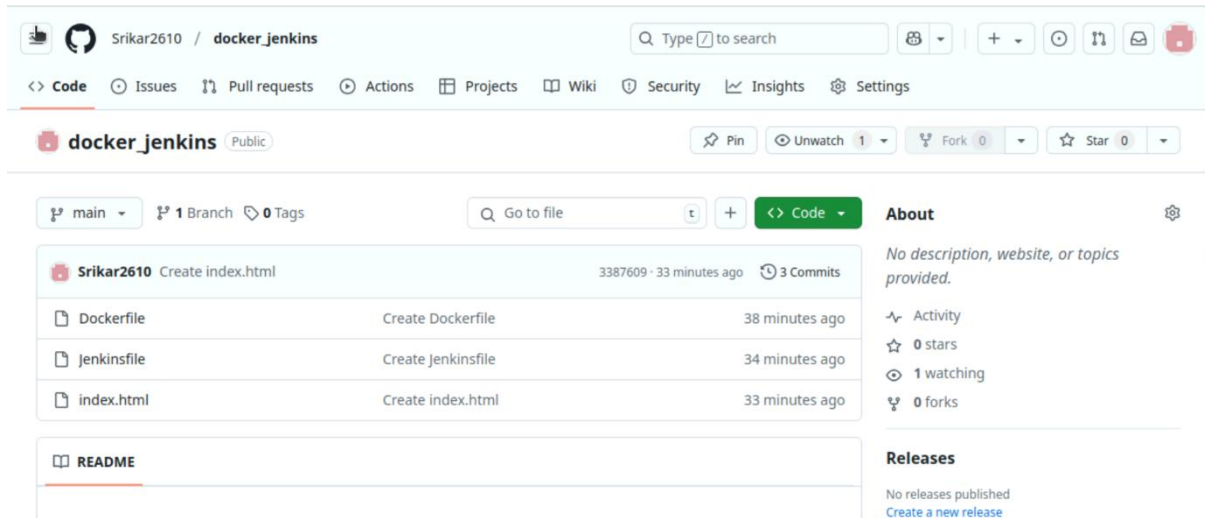
```
master@master-vm:~/Desktop/docker_db$
```

Step 4: Test the application

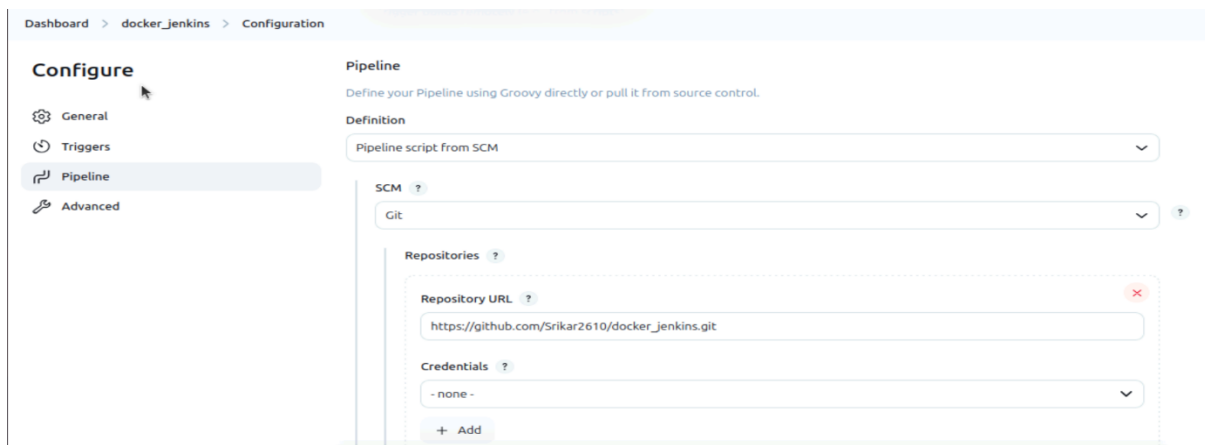


Jenkins-Dockers Using Web Application

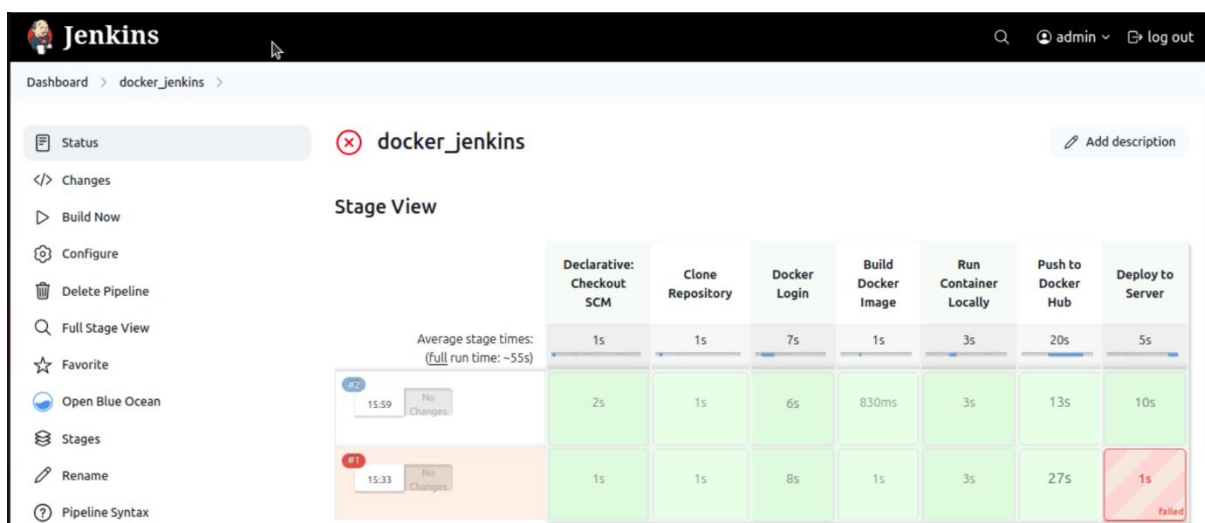
STEP 1: Push files to GitHub Repository



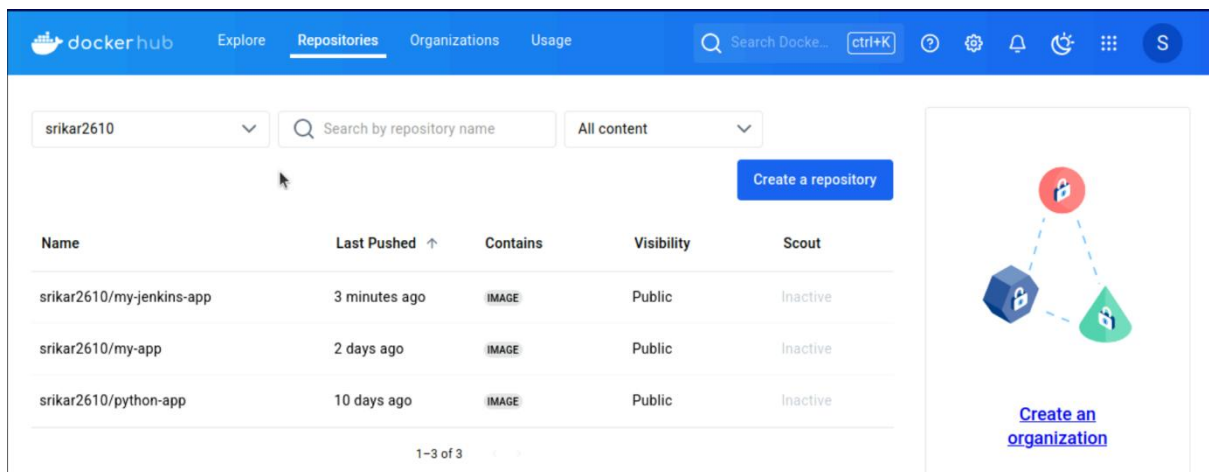
Step 2: Create a Pipeline



Step 3: Build the Pipeline



Step 4: my-jenkins-app image is build in Docker repository



The screenshot shows the Docker Hub interface for the user 'srikar2610'. The 'Repositories' tab is selected. A table lists three repositories: 'my-jenkins-app' (pushed 3 minutes ago), 'my-app' (pushed 2 days ago), and 'python-app' (pushed 10 days ago). All are public and inactive. A 'Create a repository' button is visible. On the right, there is a diagram showing a repository connected to two containers, with a link to 'Create an organization'.

Name	Last Pushed	Contains	Visibility	Scout
srikar2610/my-jenkins-app	3 minutes ago	IMAGE	Public	Inactive
srikar2610/my-app	2 days ago	IMAGE	Public	Inactive
srikar2610/python-app	10 days ago	IMAGE	Public	Inactive

OUTPUT



The screenshot shows a web browser with several tabs open: 'Port Conflict Resolu...', 'docker_jenkins [Jeni...', 'docker_jenkins/Jeni...', 'Docker Hub Contain...', and 'Jenkins + Docker Pipeli...'. The address bar shows 'localhost:8092'.

Deployment Successful with Jenkins and Docker!