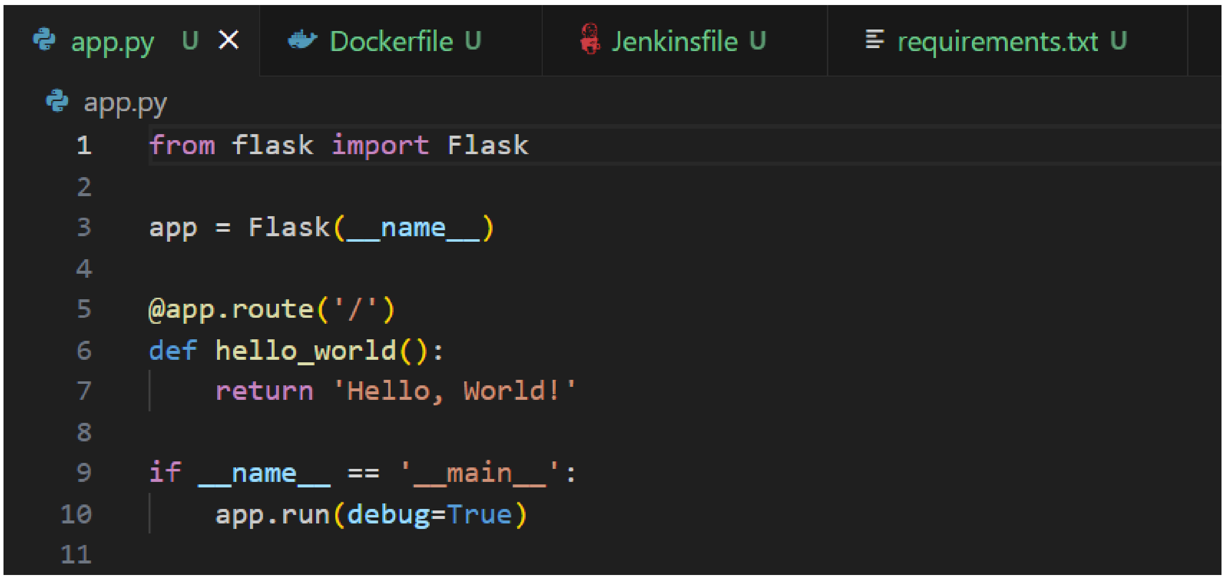
Automate docker built and push using Jenkinsfile

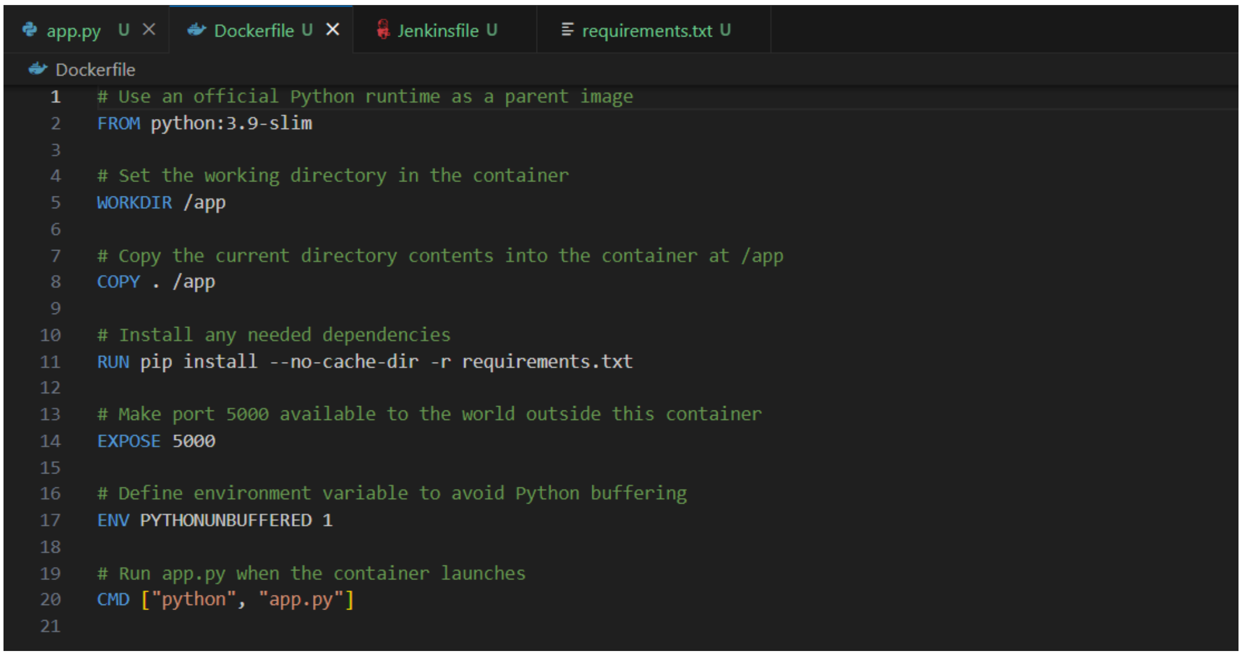
1. Set up simple flask app

   Project structure:

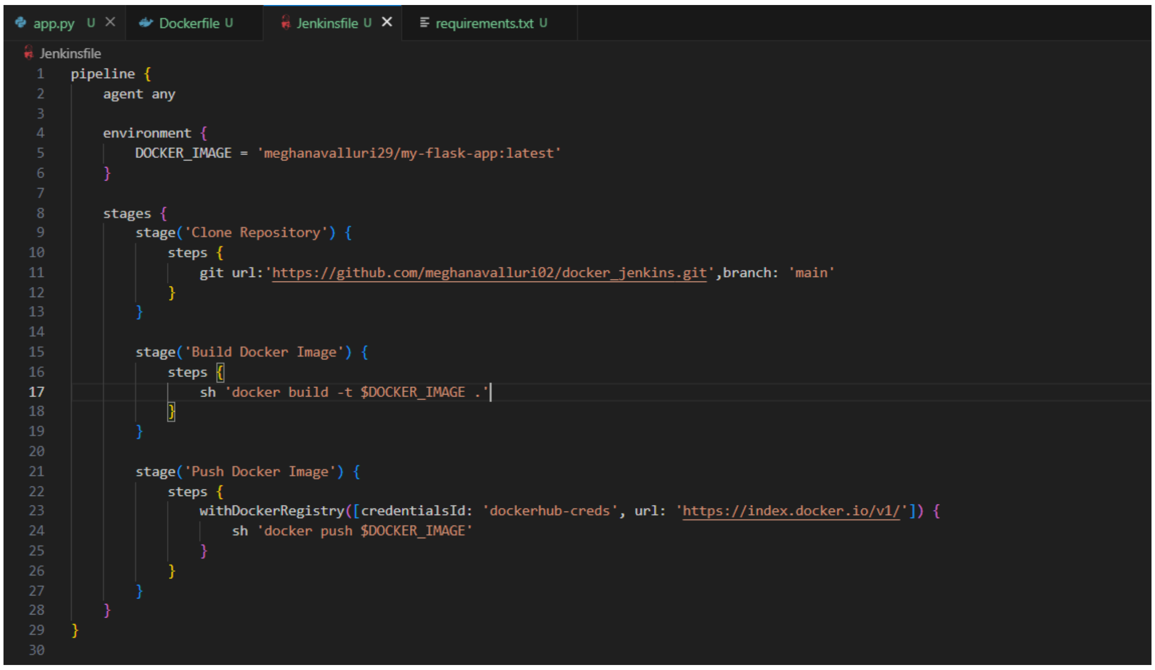
```
my-flask-app
│── app.py
│── requirements.txt
│── Dockerfile
│── Jenkinsfile
```



```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'

if __name__ == '__main__':
    app.run(debug=True)

```

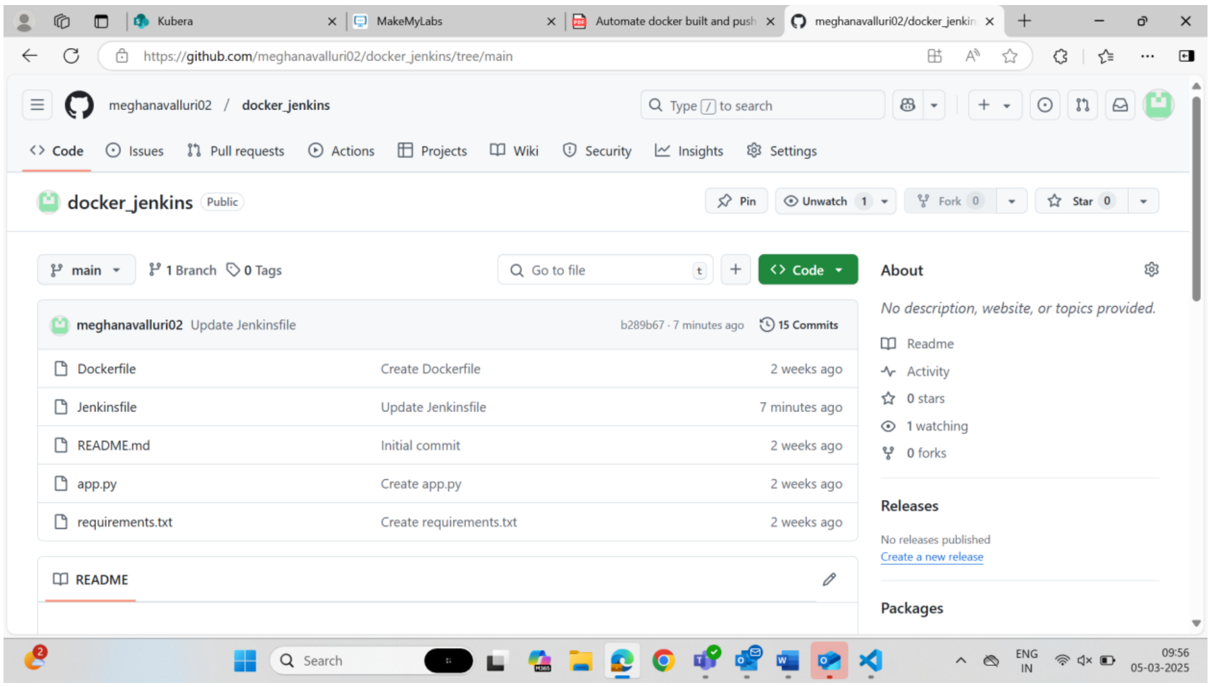


```dockerfile
# Use an official Python runtime as a parent image
FROM python:3.9-slim

# Set the working directory in the container
WORKDIR /app

# Copy the current directory contents into the container at /app
COPY . /app

# Install any needed dependencies
RUN pip install --no-cache-dir -r requirements.txt

# Make port 5000 available to the world outside this container
EXPOSE 5000

# Define environment variable to avoid Python buffering
ENV PYTHONUNBUFFERED 1

# Run app.py when the container launches
CMD ["python", "app.py"]

```



```
pipeline {
    agent any

    environment {
        DOCKER_IMAGE = 'meghanavalluri29/my-flask-app:latest'
    }

    stages {
        stage('Clone Repository') {
            steps {
                git url:'https://github.com/meghanavalluri02/docker_jenkins.git',branch: 'main'
            }
        }

        stage('Build Docker Image') {
            steps {
                sh 'docker build -t $DOCKER_IMAGE .'
            }
        }

        stage('Push Docker Image') {
            steps {
                withDockerRegistry([credentialsId: 'dockerhub-creds', url: 'https://index.docker.io/v1/']) {
                    sh 'docker push $DOCKER_IMAGE'
                }
            }
        }
    }
}
```

2. Push the code into the github:



meghanavalluri02 Update Jenkinsfile b289b67 · 7 minutes ago 15 Commits

Dockerfile	Create Dockerfile	2 weeks ago
Jenkinsfile	Update Jenkinsfile	7 minutes ago
README.md	Initial commit	2 weeks ago
app.py	Create app.py	2 weeks ago
requirements.txt	Create requirements.txt	2 weeks ago

3. Configure Docker Hub Credentials in Jenkins

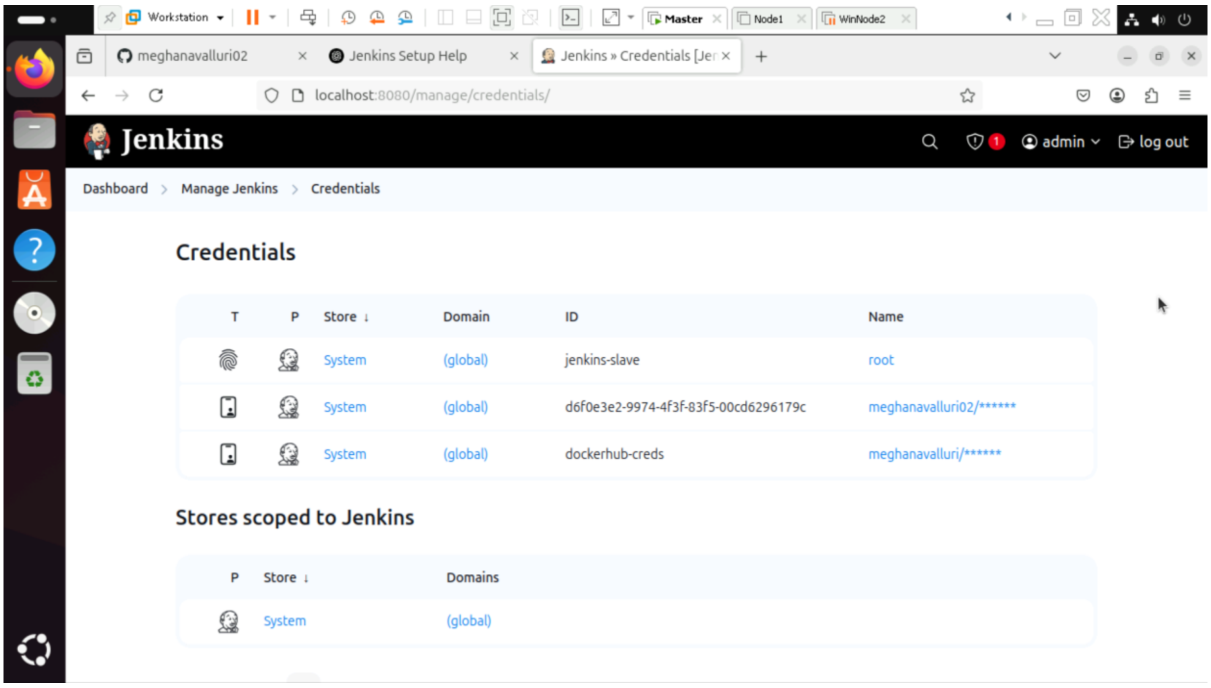
• Go to Jenkins > Manage Jenkins > Manage Credentials.

• Add new credentials:

o Username: Your Docker Hub username.

o Password: Your Docker Hub password (or token).

o ID: Name it something like dockerhub-creds (the same name used in the Jenkinsfile).



Credentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	jenkins-slave	root
		System	(global)	d6f0e3e2-9974-4f3f-83f5-00cd6296179c	meghanavalluri02/******
		System	(global)	dockerhub-creds	meghanavalluri/******

Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global)

4. Create a New Pipeline in Jenkins

• In Jenkins, click New Item > Pipeline.

• Enter a name for the pipeline.

• Under Pipeline Definition, select Pipeline script from SCM.

o Select Git as the SCM.

o Enter the GitHub repository URL (https://github.com/your-username/myflask-app.git).

o Set the branch (typically master or main).

• Click Save.

5. Click Build Now

• Click Build Now in Jenkins to trigger the build.

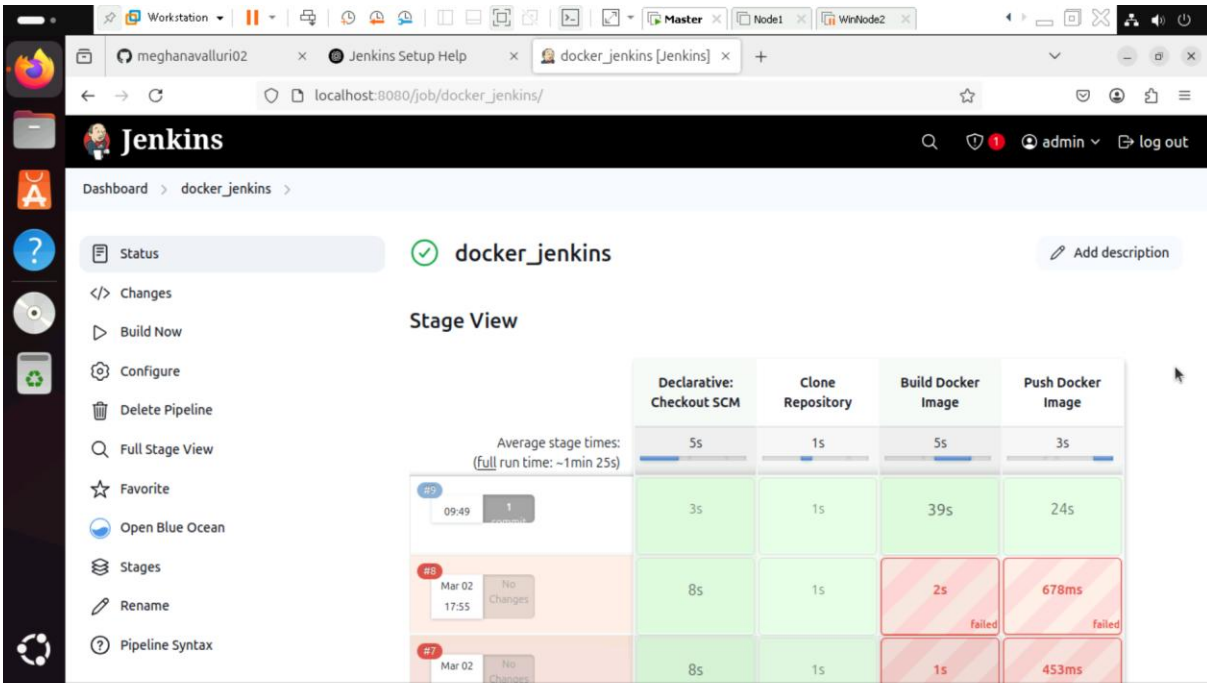
• Jenkins will:

o Checkout the code from GitHub.

o Build the Docker image.

o Push the image to Docker Hub

6.

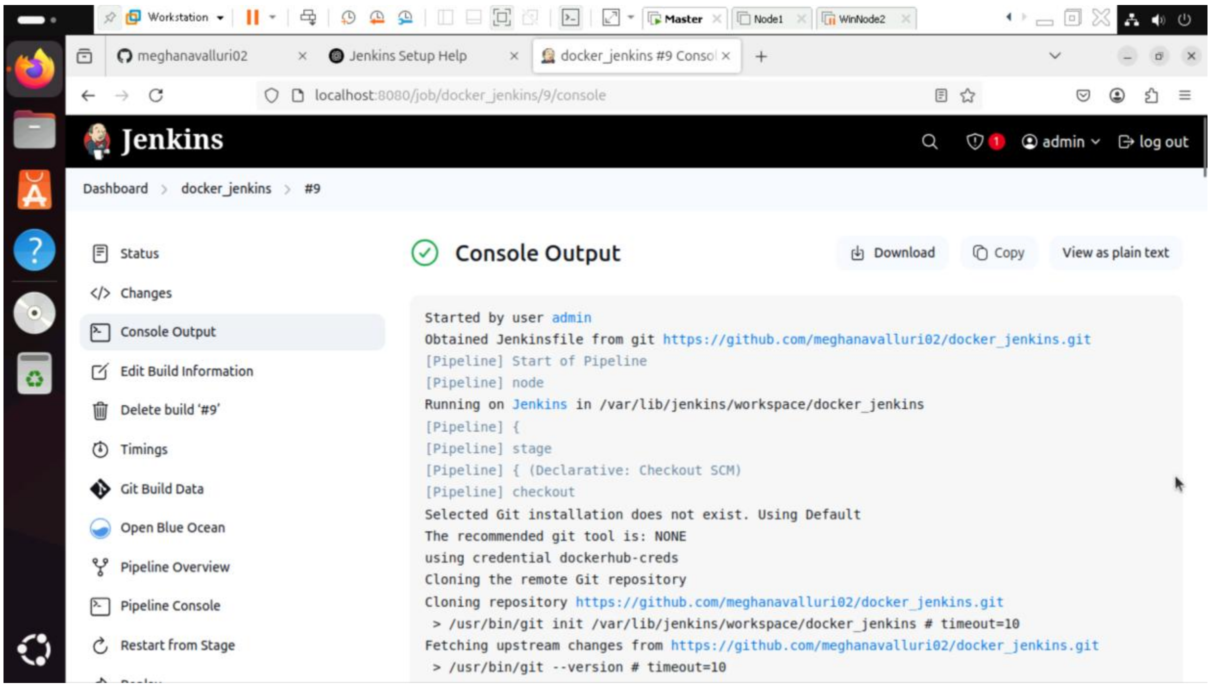


docker_jenkins

Stage View

	Declarative: Checkout SCM	Clone Repository	Build Docker Image	Push Docker Image
Average stage times: (full run time: ~1min 25s)	5s	1s	5s	3s
#9 09:49 1 commit	3s	1s	39s	24s
#8 Mar 02 17:55 No Changes	8s	1s	2s failed	678ms failed
#7 Mar 02 No Changes	8s	1s	1s	453ms

7.



Console Output

```
Started by user admin
Obtained Jenkinsfile from git https://github.com/meghanavalluri02/docker_jenkins.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/docker_jenkins
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential dockerhub-creds
Cloning the remote Git repository
Cloning repository https://github.com/meghanavalluri02/docker_jenkins.git
 > /usr/bin/git init /var/lib/jenkins/workspace/docker_jenkins # timeout=10
Fetching upstream changes from https://github.com/meghanavalluri02/docker_jenkins.git
 > /usr/bin/git --version # timeout=10
```