

Below is a **simple, complete GitHub + Jenkins + Docker example** showing:

- **Jenkinsfile** (Pipeline script)
- **Dockerfile**
- Project structure
- Explanation of each step

Project Structure

```
perl
my-app/
├── Jenkinsfile
├── Dockerfile
└── app.py
```

Copy code

Dockerfile (Simple Python App Example)

```
Dockerfile
FROM python:3.11-slim

WORKDIR /app

COPY app.py .

CMD ["python", "app.py"]
```

Sample app.py

```
python
print("Hello from Docker + Jenkins + GitHub!")
```

Jenkinsfile (Declarative Pipeline)

This pipeline:

1. Checks out code from GitHub
2. Builds a Docker image
3. Runs the Docker container
4. Pushes the image to Docker Hub (optional section commented)

```
groovy
pipeline {
    agent any

    environment {
        IMAGE_NAME = "yourdockerhubusername/my-app"
        IMAGE_TAG  = "latest"
    }

    stages {
        stage('Checkout') {
            steps {
                git branch: 'main',
                url: 'https://github.com/yourusername/my-app.git'
            }
        }

        stage('Build Docker Image') {
            steps {
                script {
                    sh 'docker build -t $IMAGE_NAME:$IMAGE_TAG .'
                }
            }
        }

        stage('Run Docker Container') {
            steps {
                script {
                    sh 'docker run --rm $IMAGE_NAME:$IMAGE_TAG'
                }
            }
        }

        // OPTIONAL: push to Docker Hub
        /*
        stage('Login & Push to Docker Hub') {
            steps {
                withCredentials([usernamePassword(
                    credentialsId: 'dockerhub-credentials',
                    usernameVariable: 'DOC_USER',
                    passwordVariable: 'DOCKER_PASS'
                )]) {
                    sh "echo $DOCKER_PASS | docker login -u $DOCKER_USER --password-stdin"
                    sh "docker push $IMAGE_NAME:$IMAGE_TAG"
                }
            }
        }
        */

        post {
            always {
                sh 'docker rmi $IMAGE_NAME:$IMAGE_TAG || true'
            }
        }
    }
}
```

Jenkins Configuration Requirements

1. Install required Jenkins plugins:

- Docker Pipeline
- Git

2. Jenkins node must have:

```
bash
docker --version
```

3. Add Docker Hub credentials:

- Jenkins → Manage Jenkins → Credentials
- ID: dockerhub-credentials

How to trigger with GitHub

In GitHub:

Settings → Webhooks → Add webhook

URL:

```
perl
http://<your-jenkins-url>/github-webhook/
```

Jenkins job:

- Type: Pipeline
- "Build when a change is pushed to GitHub" checked

If you want, I can also provide:

- GitHub Actions version
- Multi-stage Docker build
- Jenkins agent running inside Docker
- Kubernetes deployment (Jenkins → Docker → K8s)

Just tell me!

