

Day 27 Task: Jenkins Declarative Pipeline with Docker



Use your Docker Build and Run Knowledge

docker build - you can use `sh 'docker build . -t <tag>'` in your pipeline stage block to run the docker build command.

docker run: you can use `sh 'docker run -d <image>'` in your pipeline stage block to build the container.

How will the stages look

```
stages {
    stage('Build') {
        steps {
            sh 'docker build -t basanagoudapatil02/django-app:latest'
        }
    }
}
```


Task-01

- Create a docker-integrated Jenkins declarative pipeline
 1. Login to Jenkins and click on new item, Provide desired name and select “Pipeline” and click on “Ok” button.


Enter an item name

Declarative-Pipeline-react-django


» Required field


Freestyle project


This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



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.

2. Use the below-given syntax using sh inside the stage block

```

pipeline {
    agent any

    stages{
        stages('code'){
            steps{
                git url:'https://github.com/Basanagoudapatil02/django-todo.git',branch
'main'
            }
        }

        stage('Build'){
            Steps{
                sh'docker build . -t react-django-docker-img:latest'
            }
        }

        stage('Test'){
            Steps{
                echo "Testing"
            }
        }

        stage('Deploy'){
            Steps{
                sh "docker run -d react-django-docker-jenkins -p 8001:8001 react-
django-docker-img:latest"
            }
        }
    }
}

```

```
}  
}
```

General

Advanced Project Options

Pipeline

Definition

Pipeline script

Script ?

```
1= pipeline {  
2  agent any  
3  
4  stages{  
5    stage('Code') {  
6      steps {  
7        git url: "https://github.com/Basanagoudapatil82/django-todo.git", branch: "develop"  
8      }  
9    }  
10   stage('Build') {  
11     steps {  
12       sh 'docker build -t django-app-img:latest .'  
13     }  
14   }  
15   stage('Test') {  
16     steps {  
17       echo "Testing"  
18     }  
19   }  
20   stage('Deploy') {  
21     steps {  
22       sh 'docker run -d --name django-python-ctr django-app-img:latest'  
23     }  
24   }  
25 }  
26 }
```

☒ Use Groovy Sandbox ?

Save

Apply

Script ?

```
10 stage('Build') {  
11   steps {  
12     sh 'docker build -t django-app-img:latest .'  
13   }  
14 }  
15 stage('Test') {  
16   steps {  
17     echo "Testing"  
18   }  
19 }  
20 stage('Deploy') {  
21   steps {  
22     sh 'docker run -d --name django-python-ctr django-app-img:latest'  
23   }  
24 }  
25 }  
26 }
```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

3. Click on Save. Click on “Build Now”.

Dashboard > Declarative-Pipeline-react-django >

</> Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Build History trend

Filter builds...

Feb 18, 2023, 7:56 AM

Atom feed for all Atom feed for failures

Add description

Disable Project

Stage View

Average stage times:
(Average full run time: ~28s)

	Code	Build	Test	Deploy
Feb 18 13:26	938ms	22s	73ms	3s

Permalinks

- Last build (#1), 2 min 6 sec ago
- Last stable build (#1), 2 min 6 sec ago
- Last successful build (#1), 2 min 6 sec ago
- Last completed build (#1), 2 min 6 sec ago

Dashboard > Declarative-Pipeline-react-django > #1

- Status
- Changes
- Console Output
- View as plain text
- Edit Build Information
- Delete build #1
- Git Build Data
- Restart from Stage
- Replay
- Pipeline Steps
- Workspaces

Console Output

```

Started by user Basanagouda Patil
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Declarative-Pipeline-react-django
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Code)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Basanagoudapatil02/django-todo.git
> git init /var/lib/jenkins/workspace/Declarative-Pipeline-react-django # timeout=10
Fetching upstream changes from https://github.com/Basanagoudapatil02/django-todo.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/Basanagoudapatil02/django-todo.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/Basanagoudapatil02/django-todo.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch

'django.db.models.BigAutoField'.
B[0mRemoving intermediate container 757a2538cf61
--> 24330e998500
Step 6/7 : EXPOSE 8000
--> Running in 690df76ea45c
Removing intermediate container 690df76ea45c
--> ce5a753f6278
Step 7/7 : CMD python manage.py runserver 0.0.0.0:8000
--> Running in 6b5efd13928f
Removing intermediate container 6b5efd13928f
--> 087a9ecaaa3e
Successfully built 087a9ecaaa3e
Successfully tagged django-app-img:latest
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] echo
Testing
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] sh
+ docker run -d --name django-python-ctr django-app-img:latest
fda8cd0925e5db71b03c5c381e55db39d9f2da033aa607e5193d69b73baebba5
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

← → ↺ ⚠ Not secure | 34.236.243.202:8080/job/Declarative-Pipeline-react-django/1/console

Dashboard > Declarative-Pipeline-react-django > #1

- You will face errors in case of running a job twice, as the docker container will be already created, so for that do task 2

1. Run pipeline once again by clicking on “Build Now” button from dashboard.

Jenkins

Dashboard > Declarative-Pipeline-react-django >

- Status
- Changes
- Build Now
- Configure
- Delete Pipeline
- Full Stage View
- Rename
- Pipeline Syntax

Pipeline Declarative-Pipeline-react-django

Add description

Disable Project

Stage View

	Code	Build	Test	Deploy
Average stage times (Average full run time: ~25s)	702ms	34s	81ms	1s
Feb 18 13:02	676ms	43s	82ms	645ms
Feb 18 13:01	493ms	37s	89ms	623ms
Feb 18 13:04	838ms	22s	73ms	3s

Build History

trend

Filter builds...

- #1 Feb 18, 2023, 8:02 AM
- #2 Feb 18, 2023, 8:01 AM
- #3 Feb 18, 2023, 7:56 AM

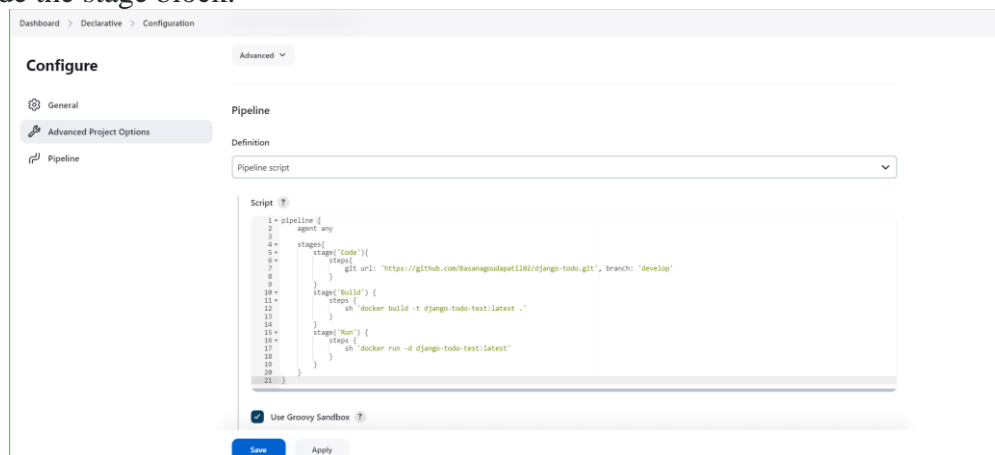
2. Error will pop up when we run same job twice.

```
Dashboard > Declarative Pipeline-react-django > #3

----> Running in pod/remote
Removing intermediate container e5d7e6e0be
----> 8515f0d095bc
Successfully built 8515f0d095bc
Successfully tagged django-app-ing:latest
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (test)
[Pipeline] who
Testing
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] sh
+ docker run -d --name django-python-ctr django-app-ing:latest
docker: Error response from daemon: Conflict. The container name "/django-python-ctr" is already in use by container
"fdac08925e5db72b03c5381e55db39d92da033a007e5193d09673ba0ba5". You have to remove (or rename) that container to be able to reuse that name.
See "docker run --help".
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
ERROR: script returned exit code 125
Finished: FAILURE
```

Task-02

- Create a docker-integrated Jenkins declarative pipeline using the docker groovy syntax inside the stage block.



Click on Save and then click on Build Now.

Dashboard > Declarative >

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Build History

trend

Filter builds...

#2 Feb 18, 2023, 8:26 AM

#1 Feb 18, 2023, 8:25 AM

Atom feed for all

Atom feed for failures

Pipeline Declarative

Stage View

	Code	Build	Run
Average stage times: (Average full run time: ~35min 27s)	379ms	17min 39s	3s
#2 Feb 18 13:56 No Changes	366ms	35min 18s	7s
#1 Feb 18 13:55 No Changes	393ms	635ms failed	46ms failed

Permalinks

- Last build (#2), 1 hr 9 min ago
- Last stable build (#2), 1 hr 9 min ago
- Last successful build (#2), 1 hr 9 min ago

Jenkins

Search (CTRL+K)

Basanagouda Patil

log out

Dashboard > Declarative > #2

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build #2

Git Build Data

Restart from Stage

Replay

Pipeline Steps

Workspaces

Previous Build

Console Output

```

Started by user Basanagouda Patil
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Declarative
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Code)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Declarative/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/Basanagoudapatil82/django-todo.git # timeout=10
Fetching upstream changes from https://github.com/Basanagoudapatil82/django-todo.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/Basanagoudapatil82/django-todo.git vrefs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/develop^{commit} # timeout=10
Checking out revision b83cc0933dc6d477b6d5d4ff0b07db0b0902 (refs/remotes/origin/develop)
> git config core.sparsecheckout # timeout=10

Removing intermediate container d8d562fd7292
--> 509b61f513f4
Step 6/7 : EXPOSE 8000
--> Running in 9c7824e8318f
Removing intermediate container 9c7824e8318f
--> 468dbd1d6f9d
Step 7/7 : CMD python manage.py runserver 0.0.0.0:8000
--> Running in fc4dfeb9dae3
Removing intermediate container fc4dfeb9dae3
--> 9c2e6cb0a112
Successfully built 9c2e6cb0a112
Successfully tagged django-todo-test:latest
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Run)
[Pipeline] sh
+ docker run -d django-todo-test:latest
2cc73521bf8ad29d8660281a09d174a883acdeb4d32eadedc0a4aee5ce19fc4d
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

Happy Learning:)

