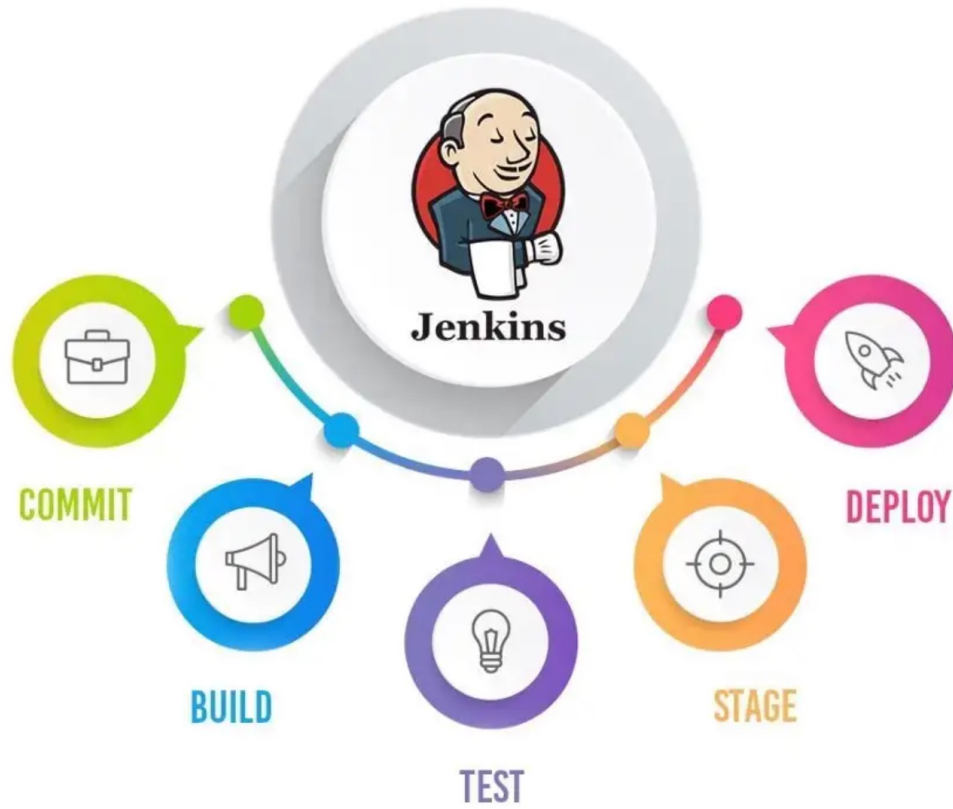


Week 6 : SOFTWARE DEVELOPMENT TOOLS AND ENVIRONMENTS



# Set Up Jenkins Credentials

---

**Jenkins** Search [?] [1] [Raj Saha] [log out]

Dashboard

New Item

People

View History

**Manage Jenkins**

My Views

New View

I will learn Jenkins and get better paying job

All + Add description

| S | W  | Name                | Last Success | Last Failure | Last Duration |
|---|----|---------------------|--------------|--------------|---------------|
| ✓ | ⚙️ | VeryFirstJenkinsJob | 19 days #3   | N/A          | 22 ms         |

Icon: S M L

Icon legend Atom feed for all Atom feed for failures Atom feed for just latest builds

Build Queue

No builds in the queue.

Build Executor Status

1 idle

2 idle

REST API Jenkins 2.332.3

1

Dashboard

Build Queue

No builds in the queue.

Build Executor Status

1 idle

2 idle

Configure global settings and paths.

Configure tools, their locations and automatic installers.

Add, remove, disable or enable plugins that can extend the functionality of Jenkins. There are updates available

Manage Nodes and Clouds

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Security

Configure Global Security

Secure Jenkins; define who is allowed to access/use the system.

**Manage Credentials**

Configure credentials

Configure Credential Providers

Configure the credential providers and types

Manage Users

Create/delete/modify users that can log in to this Jenkins

In-process Script Approval

Allows a Jenkins administrator to review proposed scripts (written e.g. in Groovy) which run inside the Jenkins process and so could bypass security restrictions. 1 scripts pending approval.

Status Information

System Information

Displays various environmental

System Log

System log captures output from

Load Statistics

Check your resource utilization and see if

2

**Jenkins** Search [?] [2] [3] [4]

Dashboard > Manage Jenkins > Credentials

Credentials

| T | P | Store | Domain | ID | Name |
|---|---|-------|--------|----|------|
|---|---|-------|--------|----|------|

Stores scoped to Jenkins

| P | Store  | Domains         |
|---|--------|-----------------|
| 👤 | System | <b>(global)</b> |

Icon: S M L

3

**Jenkins** Search [?] [2] [3] [4]

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted)

Global credentials (unrestricted)

Credentials that should be available irrespective of domain specification to requirements matching.

| ID  | Name | Kind | Description |
|---|------|------|-------------|
| This credential domain is empty. How about adding some credentials? |      |      |             |

Icon: S M L

4

# For Github

## New credentials

Kind **1** **1**

Username with password

Scope **1**

Global (Jenkins, nodes, items, all child items, etc)

Username **2**

tuchisani

☐ Treat username as secret **2**

Password **3**

ID **3**

Description

ID ? **4**

github

Description ?

my github for test jenkins

Settings / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens **3**

Tokens (classic)

## Personal access tokens (classic)

Generate new token

Revoke all

Tokens you have generated that can be used to access the GitHub API.

Make sure to copy your personal access token now. You won't be able to see it again!

ghp\_zp7azz7td4qxUcZdLyTT8PHG1LjrI8t3FnZ

Delete

Personal access tokens (classic) function like legacy OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

## Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

- ☒ **repo** Full control of private repositories
  - ☒ repo:status Access commit status
  - ☒ repo\_deployment Access deployment status
  - ☒ public\_repo Access public repositories
  - ☒ repo:invite Access repository invitations
  - ☒ security\_events Read and write security events



- ☒ **admin:repo\_hook** Full control of repository hooks
  - ☒ write:repo\_hook Write repository hooks
  - ☒ read:repo\_hook Read repository hooks

# For Github

## Global credentials (unrestricted)

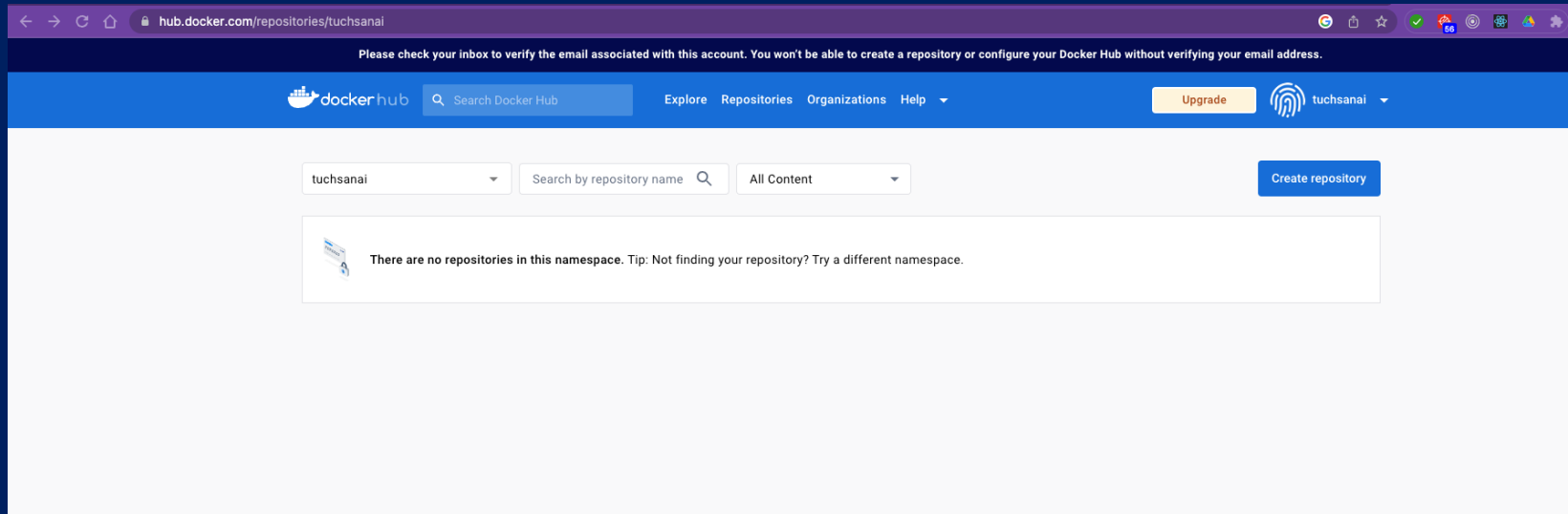
+ Add Credentials

Credentials that should be available irrespective of domain specification to requirements matching.

| ID   | Name   | Kind                   | Description                |   |
|--|--|------------------------|----------------------------|---|
|  github | tuchsanai/***** (my github for test jenkins) | Username with password | my github for test jenkins |  |

Icon: S M L

# For docker Hub



## New credentials

Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?

tuchsanai

1

☐ Treat username as secret ?

Password ?

.....

2

ID ?

dockerhub

3

Description ?

my docker hub





[Create](#)



## Global credentials (unrestricted)

[+ Add Credentials](#)

Credentials that should be available irrespective of domain specification to requirements matching.

| ID  | Name   | Kind                   | Description                |   |
|---|--|------------------------|----------------------------|---|
|  <a href="#">github</a>    | tuchsanai/***** (my github for test jenkins) | Username with password | my github for test jenkins |  |
|  <a href="#">dockerhub</a> | tuchsanai/***** (my docker hub)              | Username with password | my docker hub              |  |

Icon: S M **L**

# Intro to Jenkins Plugins

---

←

→

↻

🏠

🔒 plugins.jenkins.io

🌐

🔖

☆

Jenkins

cd

Blog

Documentation

Plugins

Community

Subprojects



# Plugins Index

Discover the 1800+ community contributed Jenkins plugins to support building, deploying and automating any project.

Browse

Find plugins...

🔍

Browse categories

Platforms

User interface

Administration

Build management

Source Code Management

New Plugins

Add Changes to Build Changelog

CodeThreat

DevOps Portal

CatLight Notification

Qualys IaC Security

Pipeline Input Step Notification

BeVigil CI

Matomo analytics

Deploy to webMethods Integration Server

Casdoor Authentication

Recently updated

Venafi CodeSign Protect

Build Failure Analyzer

Add Changes to Build Changelog

Pipeline Aggregator View

Parameterized Scheduler

Nomad

Kubernetes

SnakeYAML API

Pipeline: Basic Steps

Jakarta Mail API

Trending

Configuration as Code

AnsiColor

Job DSL

Pipeline Utility Steps

CloudBees Disk Usage Simple

SSH Agent

Kubernetes

Active Directory

Blue Ocean

Slack Notification

🗨️ Improve this page

🚨 Report a problem

Resources

Project

Community

Other



# Plugins Index


Discover the 1800+ community contributed Jenkins plugins to support building, deploying and automating any project.

Browse

docker




← → ↺ ↻ 🏠 🔒 plugins.jenkins.io/ui/search?query=docker

Jenkins 


Blog Documentation Plugins Community Subprojects Security About English Dev


Sort Relevance

- ☒ Relevance
- ☐ Most installed
- ☐ Trending
- ☐ Title
- ☐ Release date

Categories 

- ☐ Platforms
  - ☐ iOS Development
  - ☐ .NET Development
  - ☐ Android Development
  - ☐ Ruby Development
- ☐ User interface
  - ☐ User Interface
  - ☐ List view columns
- ☐ Administration
  - ☐ Agent Management
  - ☐ Page Decorators
  - ☐ Authentication and User Management
  - ☐ Cluster Management
  - ☐ Command Line Interface
- ☐ Build management
  - ☐ Build Triggers
  - ☐ Build Wrappers
  - ☐ Build Notifiers
  - ☐ Deployment
  - ☐ Build Parameters
  - ☐ Other Post-Build Actions
  - ☐ Build Tools
  - ☐ Build Reports
  - ☐ Artifact Uploaders
- ☐ Source Code Management

Browse  docker

Search by  algolia

1 to 26 of 26

Previous 1 Next


Docker

Installs: 40105  
Released: 4 weeks ago

adopt-this-plugin, Cloud Provider

This plugin integrates Jenkins with Docker

Kanstantsin Shautsou  
magnayn  
(4 other contributors)




Docker Commons

Installs: 140949  
Released: 6 months ago

Library plugins (for use by other)

Provides the common shared functionality for various Docker-related plugins.

Jesse Glick  
Kohsuke Kawaguchi  
(3 other contributors)




Docker Pipeline

Installs: 128051  
Released: 3 months ago

adopt-this-plugin, Deployment,

Build and use Docker containers from pipelines.

Jesse Glick  
Oleg Nenashev  
(6 other contributors)




Docker API

Installs: 43867  
Released: 4 days ago

adopt-this-plugin, Library plugin

This plugin provides docker-java API for other plugins.

Yoann Dubreuil  
Eric Claire




docker-build-step

Installs: 15524  
Released: 4 months ago

Build Tools, docker

This plugin allows to add various docker commands to your job as build steps.

vjuranek




CloudBees Docker Build and Publish

Installs: 9934  
Released: 6 months ago

Build Tools, docker

This plugin enables building Dockerfile based projects, as well as publishing of the built images/repos to the docker Carlos Sanchez  
Michael Neale  
(2 other contributors)




Docker Compose Build Step

Installs: 5378  
Released: 5 years ago

This plugin has no labels

Docker Compose plugin for Jenkins

João Galego




Docker Slaves

Installs: 5330  
Released: 6 years ago

docker

Uses Docker containers to run Jenkins build agents.

Yoann Dubreuil




CloudBees Docker Hub/Registry Notification

Installs: 3103  
Released: 5 years ago

docker

Run builds inside a docker container, defined by a Docker image or Dockerfile stored in project SCM.

Vincent Latombe  
Jon Hermansen




CloudBees Docker Traceability

Installs: 1398  
Released: 2 years ago

docker

Docker swarm cloud plugin.

surya gaddipati  
Roman Băriswyl



CloudShare Docker-Machine

Installs: 162  
Released: 6 years ago

Yet Another Docker

Installs: 1396  
Released: 3 years ago

Cloud Providers, Cluster Manag

Panoptica Vulnerability Scanner

Installs: 6  
Released: 4 months ago

Amazon ECR

Installs: 8908  
Released: 3 weeks ago

aws

Aqua MicroScanner

Installs: 189  
Released: 3 years ago

This plugin has no labels

CrowdStrike Security

Installs: 13  
Released: 4 months ago

This plugin has no labels

← → ↺ ⌂ Not Secure | time.tuchsanai.com:8080

# Jenkins

Dashboard >

- + New Item
- People
- Build History
- Project Relationship
- Check File Fingerprint
- Manage Jenkins >
  - My Views
- Manage Jenkins
  - My Views

**Build Queue**

No builds in the queue.

**Build Executor Status**

- 1 Idle
- 2 Idle

**System Configuration**

- Configure System
- Global Tool Configuration
- Manage Plugins**
- Manage Nodes and Clouds

**Security**

- Configure Global Security
- Manage Credentials
- Configure Credential Providers

**Status Information**

- System Information
- System Log
- Load Statistics
- About Jenkins

**Troubleshooting**

- Manage Old Data

**Tools and Actions**

- Reload Configuration from Disk
- Jenkins CLI
- Script Console
- Prepare for Shutdown

**Uncategorized**

- Docker

time.tuchsanai.com:8080/manage/pluginManager

Jenkins

Search (⌘+K)

Dashboard > Manage Jenkins > Plugin Manager

Updates

Available plugins

Installed plugins

Advanced settings

## Plugins

Q docker

| Name ↓   | Enabled                             |
|--|-------------------------------------|
| <b>Docker API</b> 3.2.13-37.vf3411c9828b9<br>This plugin provides <b>docker-java</b> API for other plugins.<br><a href="#">Report an issue with this plugin</a><br>This plugin is up for adoption! We are looking for new maintainers. Visit our <a href="#">Adopt a Plugin</a> initiative for more information. | <input checked="" type="checkbox"/> |
| <b>Docker Commons Plugin</b> 1.21<br>Provides the common shared functionality for various Docker-related plugins.<br><a href="#">Report an issue with this plugin</a>  | <input checked="" type="checkbox"/> |
| <b>Docker Pipeline</b> 563.vd5d2e5c4007f<br>Build and use Docker containers from pipelines.<br><a href="#">Report an issue with this plugin</a><br>This plugin is up for adoption! We are looking for new maintainers. Visit our <a href="#">Adopt a Plugin</a> initiative for more information.                 | <input checked="" type="checkbox"/> |
| <b>Docker plugin</b> 1.3.0<br>This plugin integrates Jenkins with <b>Docker</b><br><a href="#">Report an issue with this plugin</a><br>This plugin is up for adoption! We are looking for new maintainers. Visit our <a href="#">Adopt a Plugin</a> initiative for more information.                             | <input checked="" type="checkbox"/> |

## Plugins

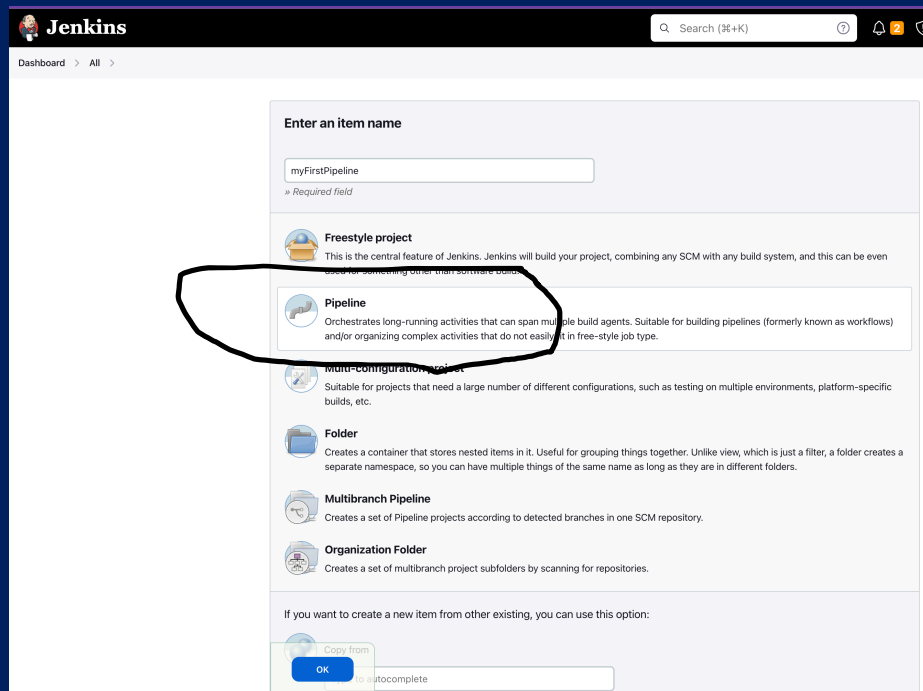
Q Docker Pipeline

| Install                  | Name ↓  | Released         |
|--------------------------|---|------------------|
| <input type="checkbox"/> | <b>Docker Pipeline</b> 563.vd5d2e5c4007f<br><a href="#">pipeline</a> <a href="#">DevOps</a> <a href="#">Deployment</a> <a href="#">docker</a><br>Build and use Docker containers from pipelines.<br>This plugin is up for adoption! We are looking for new maintainers. Visit our <a href="#">Adopt a Plugin</a> initiative for more information. | 2 mo 20 days ago |

# First pipeline using Pipeline script

---

# First pipeline



## Pipeline

### Definition

Pipeline script

Script ?

```
1- pipeline {
2-   agent any
3-   stages {
4-     stage('first pipeline') {
5-       steps {
6-
7-         echo 'welcome to first pipeline'
8-         sh 'python3 --version'
9-       }
10-    }
11-  }
12- }
13 }
```

☒ Use Groovy Sandbox ?



## Console Output

```
Started by user Tuchsana Ploysuwan
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/e
[Pipeline] {
[Pipeline] stage
[Pipeline] { (first pipeline)
[Pipeline] echo
welcome to first pipeline
[Pipeline] sh
+ python3 --version
Python 3.7.16
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

[https://github.com/Tuchsana Ploysuwan/devtool\\_week6\\_jenkins/tree/main/First\\_pipeline\\_using\\_Pipeline\\_script](https://github.com/Tuchsana Ploysuwan/devtool_week6_jenkins/tree/main/First_pipeline_using_Pipeline_script)

```
pipeline {  
  agent any  
  stages {  
    stage("build") {  
      steps {  
      }  
    }  
  }  
  node {  
    // groovy script  
  }  
}
```

- "pipeline" must be top-level
- "agent" - where to execute
- "stages" - where the "work" happens
  - "stage" and "steps"



## Declarative Pipeline fundamentals

In Declarative Pipeline syntax, the `pipeline` block defines all the work done throughout your entire Pipeline.

*Jenkinsfile (Declarative Pipeline)*

```
pipeline {  
  agent any ❶  
  stages {  
    stage('Build') { ❷  
      steps {  
        // ❸  
      }  
    }  
    stage('Test') { ❹  
      steps {  
        // ❺  
      }  
    }  
    stage('Deploy') { ❻  
      steps {  
        // ❼  
      }  
    }  
  }  
}
```

- ❶ Execute this Pipeline or any of its stages, on any available agent.
- ❷ Defines the "Build" stage.
- ❸ Perform some steps related to the "Build" stage.
- ❹ Defines the "Test" stage.
- ❺ Perform some steps related to the "Test" stage.
- ❻ Defines the "Deploy" stage.
- ❼ Perform some steps related to the "Deploy" stage.

<https://www.jenkins.io/doc/book/pipeline/>

# How To Update the Linux PATH Environment Variable

which python3

```
Last login: Mon Feb 20 06:06:03 2023 from ec2-3-0-5-37.ap-southeast-1.compute.amazonaws.com
```

```
  _ | _ | _ |  
  _ | ( _ | /  
  _ | \ _ | _ |  
Amazon Linux 2 AMI
```

```
https://aws.amazon.com/amazon-linux-2/  
30 package(s) needed for security, out of 34 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-17-81 ~]$ which python3  
/usr/bin/python3  
[ec2-user@ip-172-31-17-81 ~]$
```

You can view the `PATH` variable with the following command:

```
$ echo $PATH
```

Copy

An unchanged `PATH` may look something like this (file paths may differ slightly depending on your system):

```
[ec2-user@ip-172-31-17-81 ~]$ echo $PATH  
/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/ec2-user/.local/bin:/home/ec2-user/bin  
[ec2-user@ip-172-31-17-81 ~]$
```

# How To Update the Linux PATH Environment Variable

1. Verify that Python 3 is installed on the system: You can check if Python 3 is installed by running the following command in a terminal:

css

Copy code

```
python3 --version
```

If the output shows the version number of Python 3, it means Python 3 is installed.

2. Add the path to Python 3 to the system path: If Python 3 is installed, but Jenkins is still unable to find it, you can add the path to Python 3 to the system path. To do this, you can add the following line to the end of the `.bashrc` or `.bash_profile` file in your home directory:

ruby

Copy code

```
export PATH=$PATH:/path/to/python3/bin
```

Replace `/path/to/python3/bin` with the actual path to the directory containing the `python3` binary.

3. Restart Jenkins: After adding the path to Python 3 to the system path, you may need to restart Jenkins for the changes to take effect. You can do this by running the following command in a terminal:

Copy code

```
sudo systemctl restart jenkins
```

This command may vary depending on your system and how Jenkins is installed.

# JENKINSFILE using (Github)

---

Tuchsanai / devtool\_week6\_jenkins Private

<> Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags

Go to file Add file <> Code

|                                      |                                       |                |
|--------------------------------------|---------------------------------------|----------------|
| Tuchsanai Update .gitattributes      | 1d977a9 3 minutes ago                 | 5 commits      |
| First_pipeline_using_Pipeline_script | start                                 | 11 minutes ago |
| Pipeline_using_SCM                   | start                                 | 11 minutes ago |
| .gitattributes                       | Update .gitattributes                 | 3 minutes ago  |
| jenkinsfile_Pipeline_using_SCM       | Update jenkinsfile_Pipeline_using_SCM | 7 minutes ago  |

Add a README with an overview of your project. Add a README

main devtool\_week6\_jenkins / jenkinsfile\_Pipeline\_using\_SCM

Tuchsanai Update jenkinsfile\_Pipeline\_using\_SCM

1 contributor

12 lines (11 sloc) 220 Bytes

```
1 pipeline {
2   agent any
3   stages {
4     stage('build') {
5       steps {
6
7         echo ' welcome to github pipeline'
8         sh 'python3 --version'
9       }
10    }
11  }
12 }
```

Give feedback

[https://github.com/Tuchsanai/devtool\\_week6\\_jenkins](https://github.com/Tuchsanai/devtool_week6_jenkins)



## Enter an item name

» Required field



### Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system.



### Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (former job type).



### Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments.



### Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a name as long as they are in different folders.



### Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.



### Organization Folder

Creates a set of multibranch project subfolders by scanning for repositories.

OK

## Pipeline

### Definition

✓ Pipeline script

Pipeline script from SCM

Script ?

1

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

## Configure

- General
- Advanced Project Options
- Pipeline

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/Tuchsanai/devtool\_week6\_jenkins.git

Credentials ?

tuchsanai/\*\*\*\*\* (my github)

+ Add

Advanced...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

Script Path ?

jenkinsfile\_Pipeline\_using\_SCM

☒ Lightweight checkout ?

Pipeline Syntax

main devtool\_week6\_jenkins /jenkinsfile\_Pipeline\_using\_SCM



Tuchsana Update jenkinsfile\_Pipeline\_using\_SCM

1 contributor

12 lines (11 sloc) 220 Bytes

```
1 pipeline {
2   agent any
3   stages {
4     stage('build') {
5       steps {
6
7         echo ' welcome to github pipeline'
8         sh 'python3 --version'
9       }
10    }
11  }
12 }
```

Give feedback

Started by user Tuchsana Ploysuwan

Obtained jenkinsfile\_Pipeline\_using\_SCM from git [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in /var/lib/jenkins/workspace/s

[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 github

> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/s/.git # timeout=10

Fetching changes from the remote Git repository

> git config remote.origin.url [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git) # timeout=10

Fetching upstream changes from [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

> git --version # timeout=10

> git --version # 'git version 2.39.1'

using GIT\_ASKPASS to set credentials my github

> git fetch --tags --force --progress -- [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git) +refs/heads/\*:refs/remotes/origin/\* # timeout=10

> git rev-parse refs/remotes/origin/main^(commit) # timeout=10

Checking out Revision 1d977a9cc984a899946be4a83188d65aebffb790 (refs/remotes/origin/main)

> git config core.sparsecheckout # timeout=10

> git checkout -f 1d977a9cc984a899946be4a83188d65aebffb790 # timeout=10

Commit message: "Update .gitattributes"

> git rev-list --no-walk 1d977a9cc984a899946be4a83188d65aebffb790 # timeout=10

[Pipeline] }

[Pipeline] // stage

[Pipeline] withEnv

[Pipeline] {

[Pipeline] stage

[Pipeline] { (build)

[Pipeline] echo

welcome to github pipeline

[Pipeline] sh

+ python3 --version

Python 3.7.16

[Pipeline] }

[Pipeline] // stage

[Pipeline] }

[Pipeline] // withEnv

[Pipeline] }

[Pipeline] // node

[Pipeline] End of Pipeline

Finished: SUCCESS



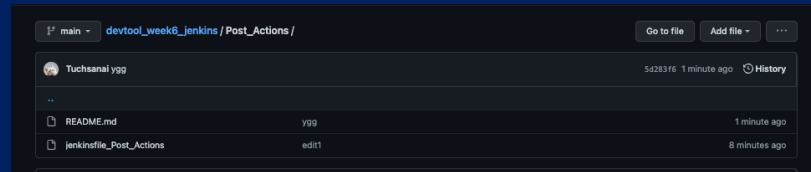
# Pipeline with post actions

---

```

pipeline {
  agent any
  stages {
    stage('build') {
      steps {
        sh 'python --version'
      }
    }
  }
  post {
    always {
      echo 'Always'
    }
    success {
      echo 'Only on SUCCESS'
    }
    failure {
      echo 'Only on Failure'
    }
    unstable {
      echo 'Only if run is unstable'
    }
    changed {
      echo 'Only if status changed from Success to Failure or vice versa w.r.t. last run.'
    }
  }
}

```



### Pipeline

Definition

Pipeline script from SCM

SCM [?](#)

Git

Repositories [?](#)

Repository URL [?](#)

[https://github.com/Tuchsanai/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsanai/devtool_week6_jenkins.git)

Credentials [?](#)

tuchsanai\*\*\*\*\* (my github)

[+ Add](#)

[Advanced...](#)

[Add Repository](#)

Branches to build [?](#)

Branch Specifier (blank for 'any') [?](#)

\*/main

[Add Branch](#)

Repository browser [?](#)

(Auto)

Additional Behaviours

[Add +](#)

Script Path [?](#)

./Post\_Actions/jenkinsfile\_Post\_Actions

☒ Lightweight checkout [?](#)

[Pipeline Syntax](#)

[Save](#) [Apply](#)

## Pipeline

### Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

[https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

Credentials ?

tuchsana|\*\*\*\*\* (my github)

+ Add

Advanced...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add +

Script Path ?

./Post\_Actions/jenkinsfile\_Post\_Actions

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Save

Apply



## Console Output

Started by user Tuchsana| Floyouvan

Obtained ./Post\_Actions/jenkinsfile\_Post\_Actions from git [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in /var/lib/jenkins/workspace/fff

[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 github

Cloning the remote Git repository

Cloning repository [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

> git init /var/lib/jenkins/workspace/fff # timeout=10

Fetching upstream changes from [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git)

> git --version # timeout=10

> git --version # 'git version 2.39.1'

using GIT\_ASKPASS to set credentials my github

> git fetch --tags --force --progress -- [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git) # timeout=10

> git config remote.origin.url [https://github.com/Tuchsana/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsana/devtool_week6_jenkins.git) # timeout=10

> git config --add remote.origin.fetch +refs/heads/\*:refs/remotes/origin/\* # timeout=10

Avoid second fetch

> git rev-parse refs/remotes/origin/main^{commit} # timeout=10

Checking out Revision 454b89843707bf7e773c992b9797f8becb59 (refs/remotes/origin/main)

> git config core.sparsecheckout # timeout=10

> git checkout -f 454b89843707bf7e773c992b9797f8becb59 # timeout=10

Commit message: 'Merge branch 'main' of [https://github.com/Tuchsana/devtool\\_week6\\_jenkins](https://github.com/Tuchsana/devtool_week6_jenkins)'

First time build. Skipping changelog.

[Pipeline] }

[Pipeline] // stage

[Pipeline] withEnv

[Pipeline] {

[Pipeline] stage

[Pipeline] { (build)

[Pipeline] sh

+ python --version

Python 2.7.18

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Declarative: Post Actions)

[Pipeline] echo

Always

[Pipeline] echo

Only if status changed from Success to Failure or vice versa w.r.t. last run.

[Pipeline] echo

Only on SUCCESS

[Pipeline] }

[Pipeline] // stage

[Pipeline] }

[Pipeline] // withEnv

[Pipeline] }

[Pipeline] // node

[Pipeline] End of Pipeline

Finished: SUCCESS

# Pipeline with Multiple Stages

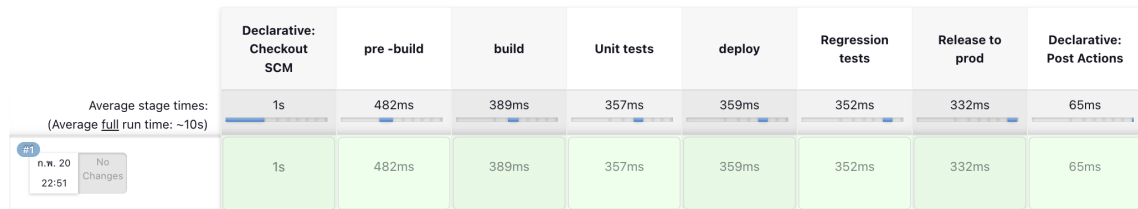
---

```

pipeline {
  agent any
  stages {
    stage('pre -build') {
      steps {
        sh 'echo Pre-build'
      }
    }
    stage('build') {
      steps {
        sh 'echo Build in progress.'
      }
    }
    stage('Unit tests') {
      steps {
        sh 'echo Running unit tests'
      }
    }
    stage('deploy') {
      steps {
        sh 'echo Deploying build'
      }
    }
    stage('Regression tests') {
      steps {
        sh 'echo Running E2E tests'
      }
    }
    stage('Release to prod') {
      steps {
        sh 'echo Releasing to prod'
      }
    }
  }
  post {
    always {
      echo 'Cleanup after everything!'
    }
  }
}

```

## Stage View



[https://github.com/Tuchsanai/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsanai/devtool_week6_jenkins.git)



Tuchsanaï 0

1a39c05 33

|                                      |                                       |
|--------------------------------------|---------------------------------------|
| First_pipeline_using_Pipeline_script | start                                 |
| Multiple_Stages                      | 0                                     |
| Pipeline_using_SCM                   | edit1                                 |
| Post_Actions                         | ygg                                   |
| .gitattributes                       | Update .gitattributes                 |
| jenkinsfile_Multiple_Stages          | 0                                     |
| jenkinsfile_Pipeline_using_SCM       | Update jenkinsfile_Pipeline_using_SCM |
| jenkinsfile_Post_Actions             | ss                                    |

[https://github.com/Tuchsanaï/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsanaï/devtool_week6_jenkins.git)

## Pipeline

### Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

[https://github.com/Tuchsanaï/devtool\\_week6\\_jenkins.git](https://github.com/Tuchsanaï/devtool_week6_jenkins.git)

Credentials ?

tuchsanaï/\*\*\*\*\* (my github)

+ Add

Advanced...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add

Script Path ?

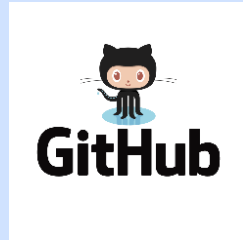
jenkinsfile\_Multiple\_Stages



Lightweight checkout ?

# Pipeline with Github Webhook

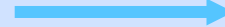
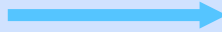
---



Branch: main

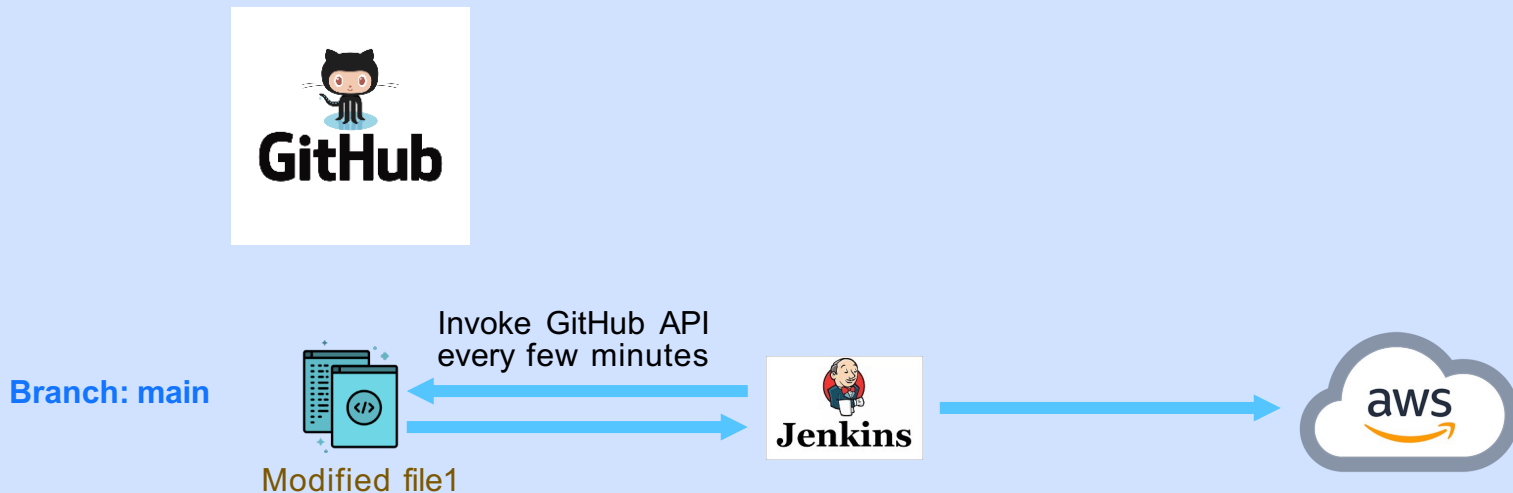


Modified file1



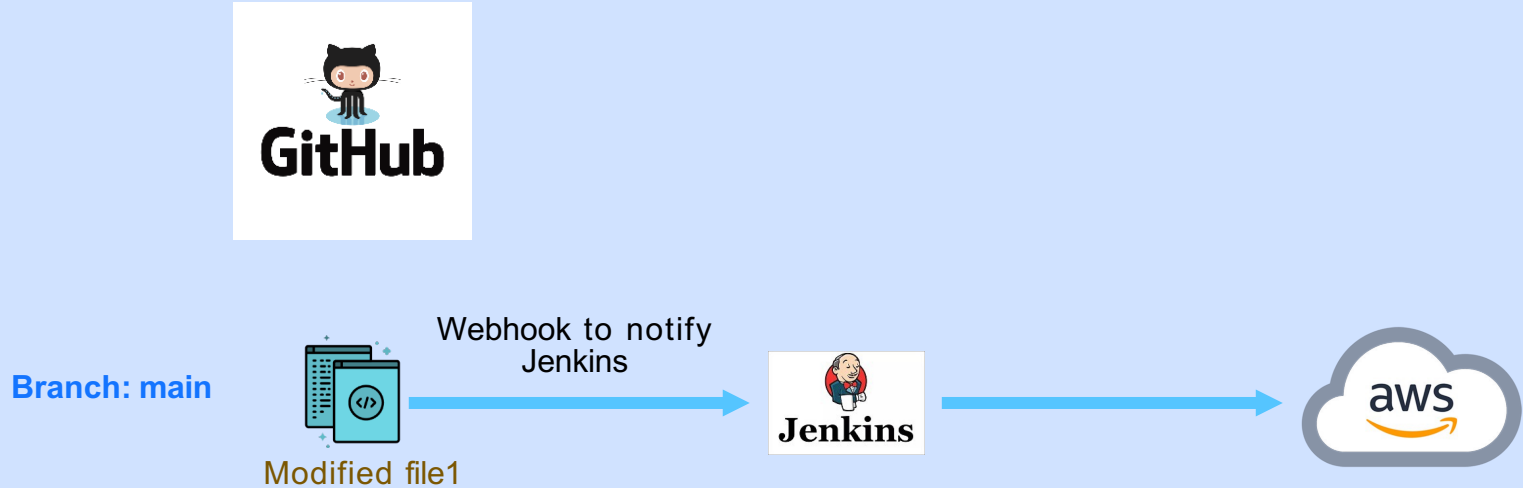


# Calling API



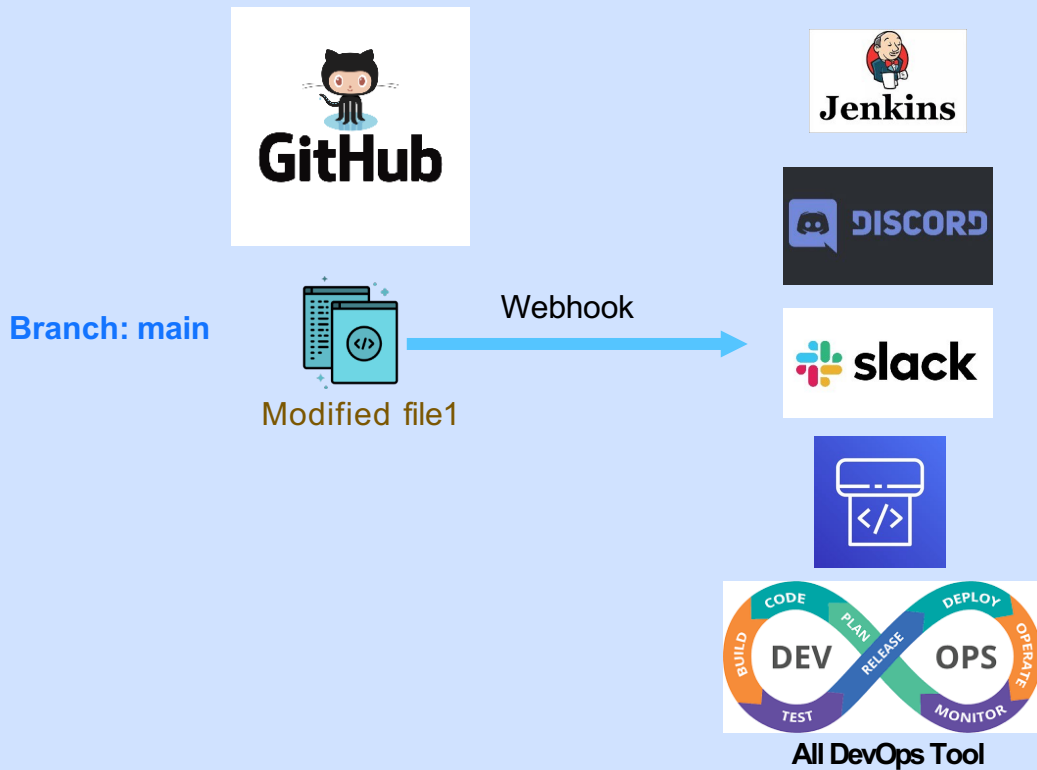
- Most of the times API will return stagnant data
- GitHub server will be bombarded
- Apps will exceed API limits

# Webhook



- GitHub will do a POST call to your app if repo changes
- Lightweight
- Realtime

# Implementation



# Python

*Jenkinsfile (Declarative Pipeline)*

```
/* Requires the Docker Pipeline plugin */
pipeline {
  agent { docker { image 'python:3.10.7-alpine' } }
  stages {
    stage('build') {
      steps {
        sh 'python --version'
      }
    }
  }
}
```

# Node.js / JavaScript

*Jenkinsfile (Declarative Pipeline)*

```
/* Requires the Docker Pipeline plugin */
pipeline {
  agent { docker { image 'node:16.17.1-alpine' } }
  stages {
    stage('build') {
      steps {
        sh 'node --version'
      }
    }
  }
}
```

# Go

*Jenkinsfile (Declarative Pipeline)*

```
/* Requires the Docker Pipeline plugin */
pipeline {
  agent { docker { image 'golang:1.19.1-alpine' } }
  stages {
    stage('build') {
      steps {
        sh 'go version'
      }
    }
  }
}
```











## **Jenkins & GitHub Integration with Selenium Python**

<https://www.youtube.com/watch?v=h2Abne2jljM&t=248s>





