

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
Linux
DevOps

# Pipeline Snippet Generator

## Contents

- [Overview](#)
- [Tutorial](#)
- [Exercises](#)

## Overview

Jenkins Pipeline Snippet Generator is a very useful piece of "Dynamic Documentation".  
Jenkins, and it's associated plugins, all register their methods with the Snippet Generator as templates which you can use to build code snippets for your Pipelines in a kind of What You See Is What You Get kind of environment.

As it is dynamic it is constantly being updated when you add new plugins or make changes to existing Pipelines.

It is most commonly used as a reference source for when you can't quite remember the syntax of various commands or their parameters, but is equally good for exploring the capabilities of your Jenkins server and finding new methods provided by plugins.

Your particular Pipeline Snippet Generator can be found:

`${YOUR_JENKINS_URL}/pipeline-syntax`

Or you can access a Project Specific one via the **Pipeline Syntax** link below the **Pipeline > Script** input when creating/editing a Pipeline.

### Pipeline

Definition

Pipeline script

Script

```
1 pipeline {
2   agent any
3   options {
4     skipStagesAfterUnstable()
5   }
6   stages {
7     stage('Build') {
8       agent { label 'agent1' }
9       steps {
10        echo '----- B
11        checkout([$class: 'GitS
12        sh 'javac HelloWorld.ja
13        sh 'jar cvfe HelloWorld
14        archiveArtifacts artifa
15        sh 'sleep 15'
16        echo '----- B
17      }
18    }
19  }
```

☒ Use Groovy Sandbox

Pipeline Syntax

This will take you to a form where you can explore the various methods at your disposal:

## Jenkins Introduction

## Jenkins Pipeline

- ☒ Jenkins Agents
- ☒ Pipeline
- ☒ Pipeline Snippet Generator
- ☐ Credentials

## Markdown

## IDE Cheatsheet

Back

Snippet Generator

Declarative Directive Generator

Declarative Online Documentation

Steps Reference

Global Variables Reference

Online Documentation

Examples Reference

IntelliJ IDEA GDSL

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various steps. Pick a step you are interested in from the list, configure it, click **Generate Pipeline Script** and you will see a Pipeline Script statement that would call the step with that configuration. You may copy and paste the whole statement into your script, or pick up just the options you care about. (Most parameters are optional and can be omitted in your script, leaving them at default values.)

Steps

Sample Step

acceptGitLabMR: Accept GitLab Merge Request

Merge commit message

Generate Pipeline Script

Global Variables

There are many features of the Pipeline that are not steps. These are often exposed via global variables, which are not supported by the snippet generator. See the [Global Variables Reference](#) for details.

Here we can use the **Sample Step** drop down select to change the underlying form template and generate various snippets.

For instance, by selecting **dir: Change current directory** you are given a **Path** input box which you can provide with something like **\$JENKINS\_HOME/jobs/\$JOB\_NAME/builds/\$BUILD\_NUMBER/archive** which would generate:

```
dir('$JENKINS_HOME/jobs/$JOB_NAME/builds/$BUILD_NUMBER/archive') {  
    // some block  
}
```

**// some block** is a comment where you would put whatever steps you want to run within the specified directory.

Notice, however, that this isn't:

```
cd $JENKINS_HOME/jobs/$JOB_NAME/builds/$BUILD_NUMBER/archive
```

It's important to remember that these are Jenkins Methods (i.e. you are instructing Jenkins directly to change dir) and not Shell Commands.

There is a way, however, to specify a Jenkins Method that runs Shell Commands using the **sh: Shell Script** or **script: Run arbitrary Pipeline script** methods.

## Tutorial

There is no tutorial for this module.

## Exercises

There are no exercises for this module.