

## 01) What is Jenkins pipeline?

Jenkins Pipeline is a suite of plugins which supports implementing and integrating *continuous delivery pipelines* into Jenkins.

A *continuous delivery pipeline* is an automated expression of the process for getting software from version control right through to your users and customers.

## 02) What scripting language is Jenkins pipeline syntax based on?

- Groovy programming language

## 03) What are the ways you can write pipeline in Jenkins?

- Declarative
- Scripted

## 04) Create Jenkins pipeline for your repo and use script file (jenkinsfile) to write pipeline syntax ?

The screenshot displays the Jenkins web interface for a project named 'project22'. The left sidebar contains navigation links: Status, Changes, Build with Parameters, Configure, Delete Pipeline, Full Stage View, Rename, and Pipeline Syntax. Below these is the 'Build History' section, which shows a list of builds. The first build, #1, is highlighted and shows a status of 'Success' (green circle) and a timestamp of 'Feb 7, 2023, 19:26 AM'. The main content area is titled 'Pipeline project22' and features a 'Stage View' section. This section displays a bar chart of stage times for the 'Checkout' stage, with an average stage time of 2s and an average full run time of ~5s. The 'Permalinks' section at the bottom provides a link to the 'Last build (#1), 86 ms ago'. The interface also includes buttons for 'Add description' and 'Disable Project'.

- ✓ Edit Build Information
- 🗑 Delete build '#1'
- ≡ Parameters
- 🔗 Git Build Data
- 🔄 Replay
- 📋 Pipeline Steps
- 📁 Workspaces

```
Running on Jenkins in /var/jenkins_home/workspace/project22
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Checkout)
[Pipeline] git
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential github-token
Cloning the remote Git repository
Cloning repository https://github.com/AmrGomaa96/test
> git init /var/jenkins_home/workspace/project22 # timeout=10
Fetching upstream changes from https://github.com/AmrGomaa96/test
> git --version # timeout=10
> git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials
> git fetch --tags --force --progress -- https://github.com/AmrGomaa96/test +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/AmrGomaa96/test # 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 0a5e6536d1d21d53f267d5e50e30168f2da93602 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 0a5e6536d1d21d53f267d5e50e30168f2da93602 # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b main 0a5e6536d1d21d53f267d5e50e30168f2da93602 # timeout=10
Commit message: "Add files via upload"
First time build. Skipping changelog.
[Pipeline] sh
+ ls
Dockerfile
Jenkinsfile
README.md
testfile
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Dashboard > project22 > Configuration

## Configure

- ⚙ General
- 🔗 Advanced Project Options
- 📋 Pipeline

[Plain text] [Preview](#)

### Choice Parameter ?

Name ?

namespace

Choices ?

dev  
prod  
stage

Description ?

[Plain text] [Preview](#)


[Add Parameter](#)

☐ Throttle builds ?





### Build Triggers

☐ Build after other projects are built ?

## 05) Create another multibranch pipeline and filter branches to contain only (master , dev , test ) ?

**Jenkins**

Search (CTRL+K)

 amr goma ▾  log out

Dashboard > multi >

Status

Configure

Scan Repository Now

Scan Repository Log

Multibranch Pipeline Events

Delete Multibranch Pipeline

People

Build History

Project Relationship

Check File Fingerprint

GitHub

Rename

Pipeline Syntax




Credentials

multi

Branches (3) Pull Requests (0)

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☀	dev	1 min 45 sec #1	N/A	4.5 sec	▶
✓	☀	master	1 min 45 sec #1	N/A	7.5 sec	▶
✓	☀	test	1 min 45 sec #1	N/A	4.5 sec	▶

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.