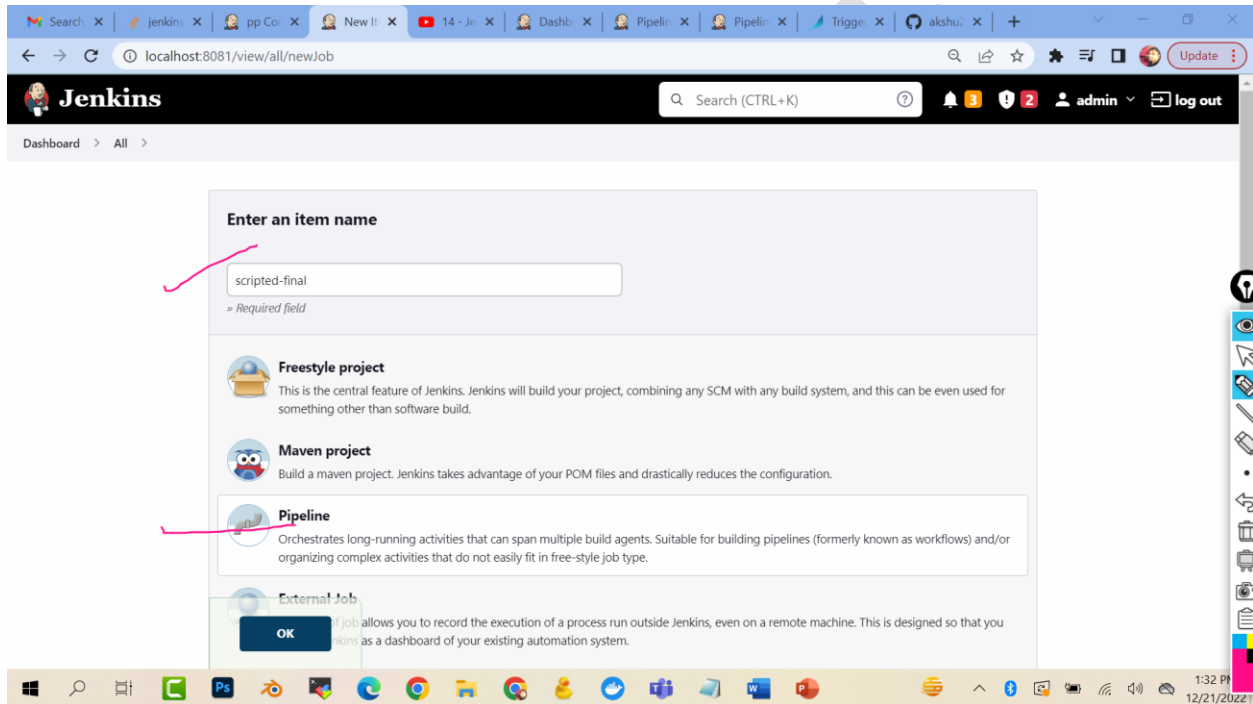


PARAMETIZATION CONCEPT

Problem statement : We have a scripted pipeline and we want to build it for specific branches.
For example with one scripted pipeline I should be able to use it for multiple branches in Jenkins

Create new item



Search | jenkins | pp Co | scrip | 14 - Je | Dashb | Pipelin | Pipelin | Trigg | akshu | +

localhost:8081/job/scripted-final/configure

Dashboard > scripted-final >

Configuration

- General
- Advanced Project Options
- Pipeline

Pipeline

Definition

Pipeline script

Script ?

1 try sample Pipeline...

☒ Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

1:32 PM 12/21/2022

Search | jenkins | pp Co | scrip | Pipel | 14 - Je | Dashb | Pipelin | Pipelin | Trigg | akshu | +

localhost:8081/job/scripted-final/pipeline-syntax/

Dashboard > scripted-final > Pipeline Syntax

Jenkins

Search (CTRL+K)

admin log out

Dashboard > scripted-final > Pipeline Syntax

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

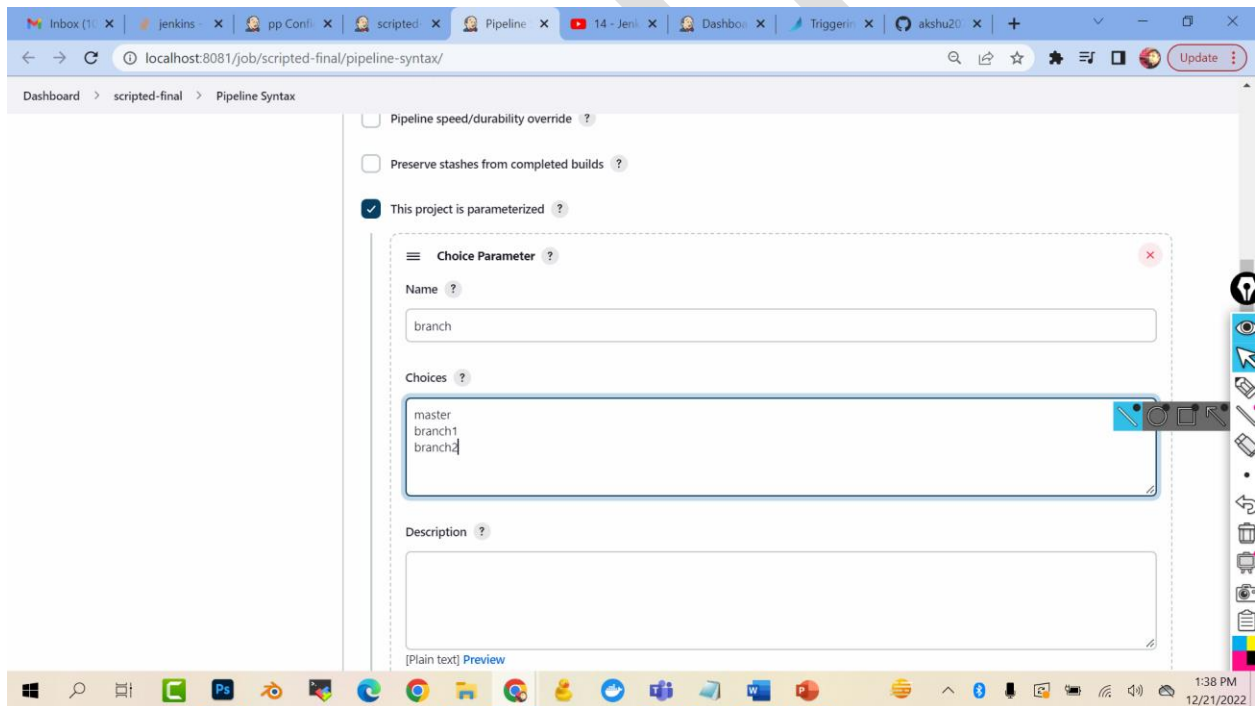
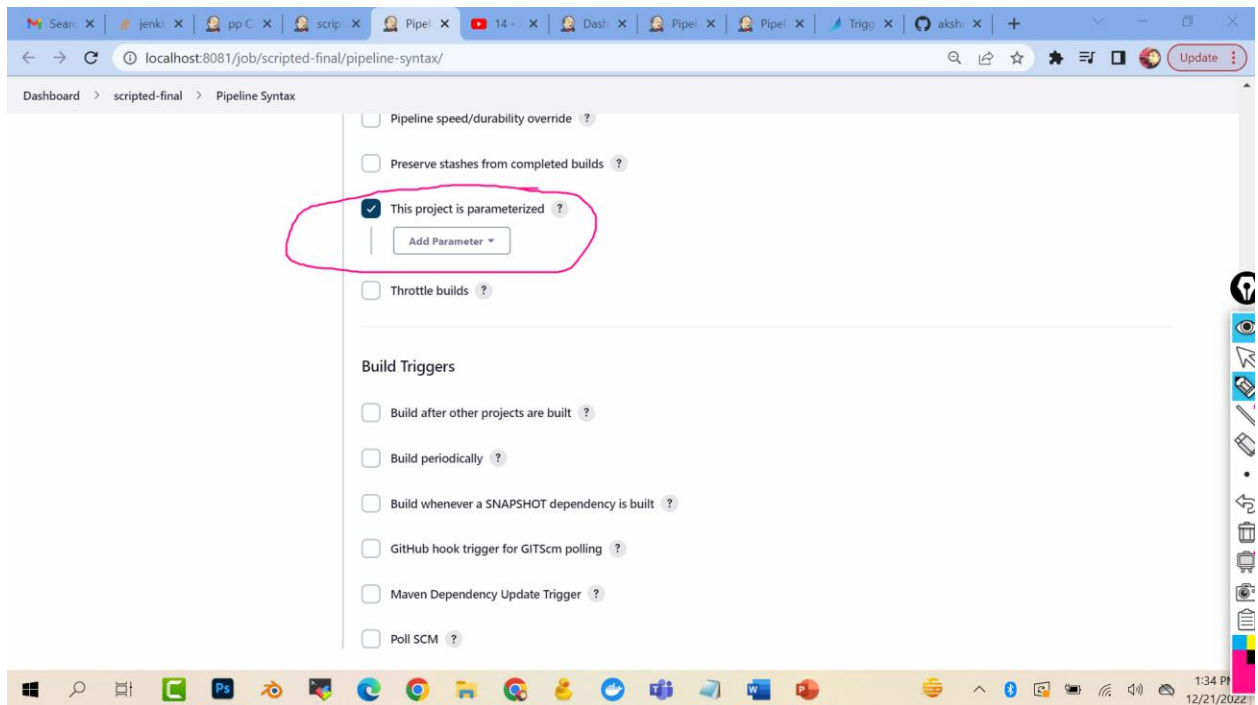
Sample Step

properties: Set job properties

properties ?

- ☐ Discard old builds ?
- ☐ Do not allow concurrent builds
- ☐ Do not allow the pipeline to resume if the controller restarts
- ☐ GitHub project

1:32 PM 12/21/2022



localhost:8081/job/scripted-final/pipeline-syntax/

Dashboard > scripted-final > Pipeline Syntax

- ☐ Build periodically ?
- ☐ Build whenever a SNAPSHOT dependency is built ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Maven Dependency Update Trigger ?
- ☐ Poll SCM ?

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.

Taskbar: Windows, Search, File Explorer, VS Code, Docker, Jenkins, Chrome, Edge, Firefox, Telegram, Word, PowerPoint, Jenkins 2.361.4, 1:38 PM, 12/21/2022

localhost:8081/job/scripted-final/pipeline-syntax/

Dashboard > scripted-final > Pipeline Syntax

- ☐ Build periodically ?
- ☐ Build whenever a SNAPSHOT dependency is built ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Maven Dependency Update Trigger ?
- ☐ Poll SCM ?

Generate Pipeline Script

```
properties([parameters([choice(choices: ['master', 'branch1', 'branch2'], name: 'branch'))])])
```

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.

Taskbar: Windows, Search, File Explorer, VS Code, Docker, Jenkins, Chrome, Edge, Firefox, Telegram, Word, PowerPoint, Jenkins 2.361.4, 1:41 PM, 12/21/2022

Configuration

Pipeline

Definition

Pipeline script

Script ?

```
1 properties([parameters([choice(choices: ['master', 'branch1', 'branch2'], name: 'branch'))]))
```

try sample Pipeline...

☒ Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

1:42 PM 12/21/2022

Configuration

Pipeline

Definition

Pipeline script

Script ?

```
1 properties([parameters([choice(choices: ['master', 'branch1', 'branch2'], name: 'branch'))]))
2 pipeline{
3   agent any
4   tools{
5     maven 'mymaven'
6   }
7   stages{
8     stage("checkout"){
9       steps{
10        echo "pulling from $params.branch"
11        git url:'https://github.com/akshu20791/DevOpsClassCodes', branch "$params.branch"
12      }
13    }
14    stage("akshat compilee"){
15      steps{
16        bat "mvn compile"
17      }
18    }
19  }
20 }
```

try sample Pipeline...

☒ Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

REST API Jenkins 2.361.4

1:43 PM 12/21/2022

```
properties([parameters([choice(choices: ['master', 'branch1', 'branch2'], name: 'branch'))]))
```

```
pipeline{
```

```
    agent any
```

```
    tools{
```

```
        maven 'mymaven'
```

```
    }
```

```
    stages{
```

```
        stage("checkout"){
```

```
            steps{
```

```
                echo "pulling from $params.branch"
```

```
                git url:'https://github.com/akshu20791/DevOpsClassCodes', branch: "$params.branch"
```

```
            }
```

```
        }
```

```
        stage("akshat compile"){
```

```
            steps{
```

```
                bat "mvn compile"
```

```
            }
```

```
        }
```

```
        stage("akshat testing"){
```

```
            steps{
```

```
                bat "mvn test"
```

```
            }
```

```
        }
```

```
        stage("akshat qa"){
```

```
            steps{
```

```
                bat "mvn pmd:pmd"
```

```
            }
```

```
        }
```

```
        stage("akshat package"){
```

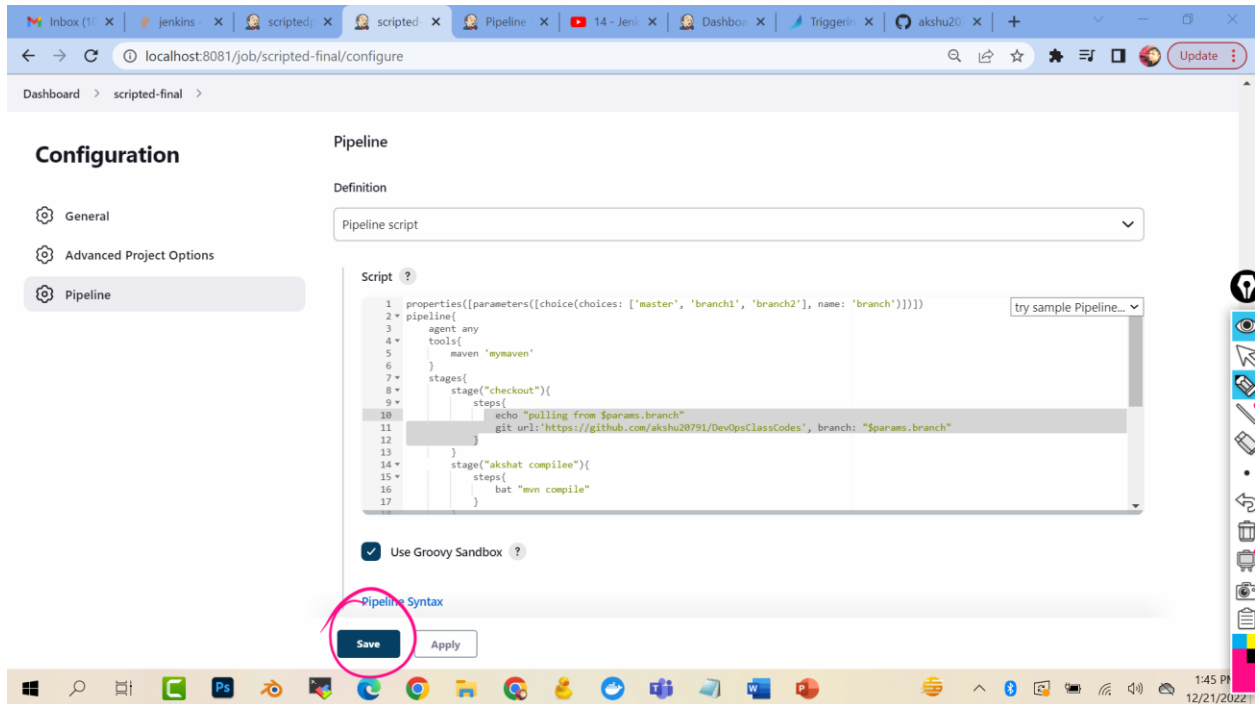
```
            steps{
```

```
                bat "mvn package"
```

```
            }
```

```
        }
```

```
}  
  
}
```



This screenshot shows the Jenkins Pipeline Configuration page for a job named 'scripted-final'. The left sidebar contains a 'Configuration' section with tabs for 'General', 'Advanced Project Options', and 'Pipeline'. The 'Pipeline' tab is selected. The main area is titled 'Pipeline' and shows the 'Definition' as 'Pipeline script'. Below this, the 'Script' field contains a Groovy script for a pipeline with a 'checkout' stage and an 'akshat compile' stage. A 'Save' button is circled in red. The bottom of the page shows a Windows taskbar with various application icons.

Configuration

Pipeline

Definition

Pipeline script

Script ?

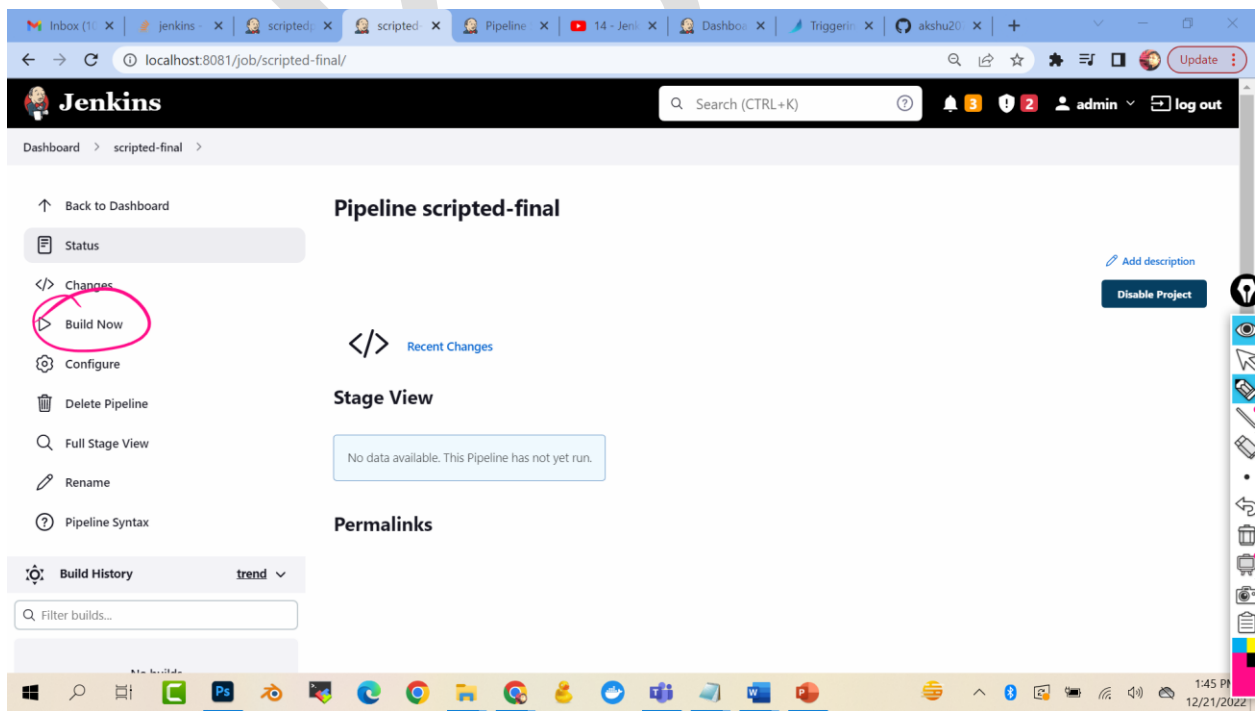
```
1 properties([parameters([choice(choices: ['master', 'branch1', 'branch2'], name: 'branch')))])  
2 pipeline(  
3   agent any  
4   tools(  
5     maven 'mymaven'  
6   )  
7   stages(  
8     stage("checkout"){  
9       steps(  
10        echo "pulling from $params.branch"  
11        git url: 'https://github.com/akshu20791/DevOpsClassCodes', branch: "$params.branch"  
12      )  
13    }  
14    stage("akshat compile"){  
15      steps(  
16        bat "mvn compile"  
17      )  
18    }  
19  )  
20 )
```

try sample Pipeline...

☒ Use Groovy Sandbox ?

Pipeline Syntax

Save Apply



This screenshot shows the Jenkins Pipeline View page for the 'scripted-final' pipeline. The left sidebar contains a 'Pipeline scripted-final' section with tabs for 'Status', 'Changes', 'Build Now', 'Configure', 'Delete Pipeline', 'Full Stage View', 'Rename', and 'Pipeline Syntax'. The 'Build Now' button is circled in red. The main area shows the 'Stage View' with a message: 'No data available. This Pipeline has not yet run.' The bottom of the page shows a Windows taskbar with various application icons.

Jenkins

Dashboard > scripted-final >

Back to Dashboard

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Pipeline scripted-final

Recent Changes

Stage View

No data available. This Pipeline has not yet run.

Permalinks

Build History trend

Filter builds...

Build once and then refresh the page

Refresh the page ..you will see that build is changed to build with parameters

The screenshot shows the Jenkins web interface for a pipeline named 'scripted-final'. The left sidebar contains a list of actions: Status, Changes, Build with Parameters (circled in red), Configure, Delete Pipeline, Full Stage View, Rename, and Pipeline Syntax. The main area displays the 'Stage View' with a table of stage times. The 'Build History' section shows a single build from Dec 21, 2022, at 1:45 PM.

Declarative: Tool Install	checkout	akshat compilee	akshat testing	akshat qa	akshat package
120ms	4s	4s	11s	10s	8s

Average stage times: (Average full run time: ~42s)

Build History: #1 Dec 21, 2022, 1:45 PM

The screenshot shows the Jenkins web interface for the same pipeline, but now displaying the 'Build with Parameters' form. The left sidebar shows the 'Build with Parameters' option selected. The main area displays the 'Pipeline scripted-final' title and a form with a 'branch' dropdown set to 'master' and a 'Build' button. The 'Build History' section shows the same build from Dec 21, 2022, at 1:45 PM.

Pipeline scripted-final

This build requires parameters:

branch: master

Build

Build History: #1 Dec 21, 2022, 1:45 PM

Build

The screenshot shows the Jenkins Pipeline Stage View for the 'scripted-final' job. The interface includes a sidebar with navigation options: Build with Parameters, Configure, Delete Pipeline, Full Stage View, Rename, and Pipeline Syntax. The main area displays the 'Stage View' with a table of stage times and a 'Build History' section on the left. The 'Build History' section shows two builds: #2 (Dec 21, 2022, 1:48 PM) and #1 (Dec 21, 2022, 1:45 PM). The 'Stage View' table shows the following data:

Stage	Declarative: Tool Install	checkout	akshat compilee	akshat testing	akshat qa	akshat package
Average stage times (Average full run time: ~42s)	135ms	4s	2s	11s	10s	8s
#2 (Dec 21 13:48)	151ms	4s	497ms			
#1 (Dec 21 13:45)	120ms	4s	4s	11s	10s	8s

The 'Permalinks' section lists the following links:

- Last build (#2), 15 ms ago
- Last stable build (#1), 2 min 3 sec ago
- Last successful build (#1), 2 min 3 sec ago
- Last completed build (#1), 2 min 3 sec ago

This screenshot is similar to the one above, but with build #2 highlighted in the 'Build History' section. The 'Stage View' table shows the following data:

Stage	Declarative: Tool Install	checkout	akshat compilee	akshat testing	akshat qa	akshat package
Average stage times (Average full run time: ~42s)	135ms	4s	5s	8s	10s	8s
#2 (Dec 21 13:48)	151ms	4s	5s	5s		
#1 (Dec 21 13:45)	120ms	4s	4s	11s	10s	8s

The 'Permalinks' section lists the following links:

- Last build (#2), 15 ms ago
- Last stable build (#1), 2 min 3 sec ago
- Last successful build (#1), 2 min 3 sec ago
- Last completed build (#1), 2 min 3 sec ago

Console output

The screenshot shows the Jenkins console output for a pipeline build. The browser address bar indicates the URL is `localhost:8081/job/scripted-final/2/console`. The left sidebar shows the 'Pipeline Steps' view. The main console area displays the following output:

```
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (checkout)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] echo
pulling from master
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\workspace\scripted-final\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/akshu20791/DevOpsClassCodes # timeout=10
Fetching upstream changes from https://github.com/akshu20791/DevOpsClassCodes
> git.exe --version # timeout=10
> git --version # 'git version 2.38.1.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/akshu20791/DevOpsClassCodes +refs/heads/*:refs/remotes/origin/* #
timeout=10
> git.exe rev-parse "refs/remotes/origin/master^[commit]" # timeout=10
Checking out Revision bced7e042a12bfda32507b7e1cdf5ea5475cefb9 (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f bced7e042a12bfda32507b7e1cdf5ea5475cefb9 # timeout=10
> git.exe branch -a -v --no-abbrev # timeout=10
> git.exe branch -D master # timeout=10
> git.exe checkout -b master bced7e042a12bfda32507b7e1cdf5ea5475cefb9 # timeout=10
Commit message: "test"
```

The screenshot shows the Jenkins Pipeline Stage View for the 'scripted-final' pipeline. The browser address bar indicates the URL is `localhost:8081/job/scripted-final/`. The left sidebar shows the 'Stage View' view. The main stage view displays the following information:

Recent Changes

Stage View

	Declarative: Tool Install	checkout	akshat compilee	akshat testing	akshat qa	akshat package
Average stage times: (Average full run time: ~38s)	135ms	4s	5s	8s	10s	8s
#2 Dec 21 13:48 No Changes	151ms	4s	5s	5s	10s	7s
#1 Dec 21 13:45 No Changes	120ms	4s	4s	11s	10s	8s

Build History

Filter builds...

- #2 Dec 21, 2022 1:48 PM
- #1 Dec 21, 2022 1:45 PM

Permalinks

Akshat