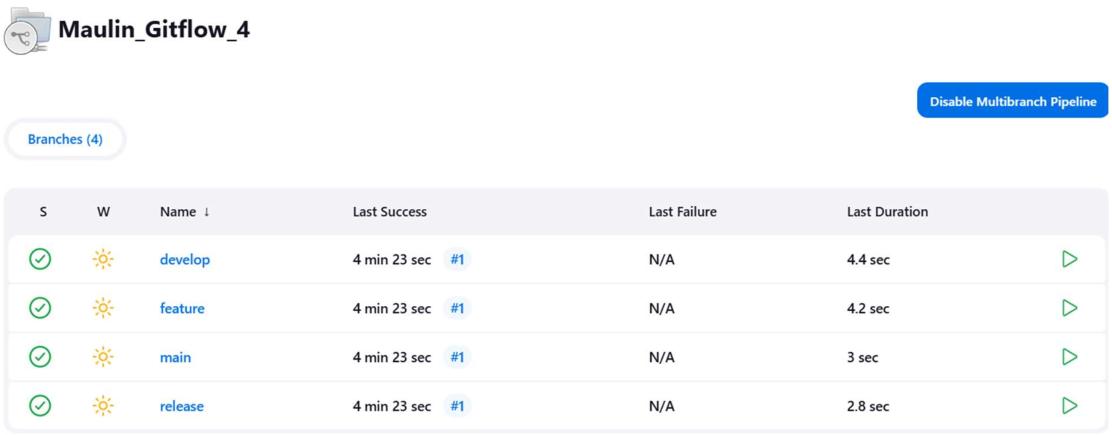
Jenkins Multi-Branch Pipeline for GitFlow with SCM

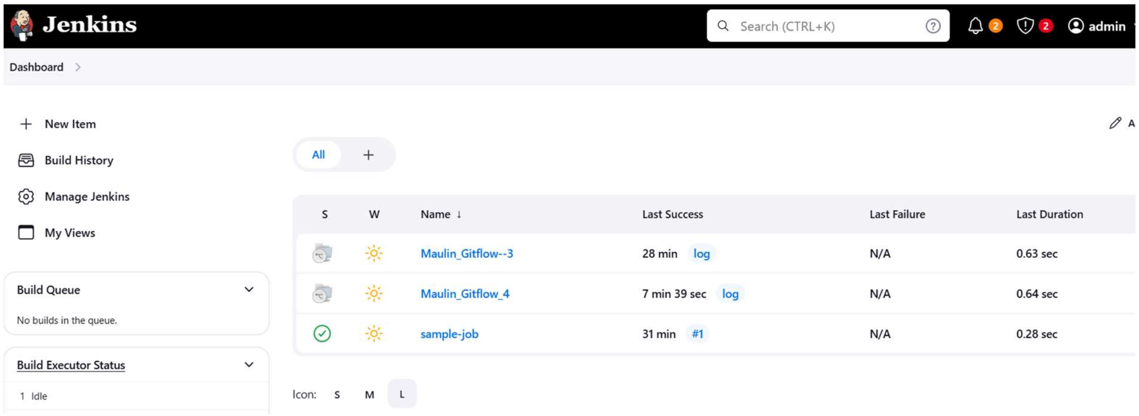


Maulin_Gitflow_4

Disable Multibranch Pipeline

Branches (4)

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☼	develop	4 min 23 sec #1	N/A	4.4 sec	▷
✓	☼	feature	4 min 23 sec #1	N/A	4.2 sec	▷
✓	☼	main	4 min 23 sec #1	N/A	3 sec	▷
✓	☼	release	4 min 23 sec #1	N/A	2.8 sec	▷



Jenkins

Search (CTRL+K)

admin

Dashboard >

- + New Item
- Build History
- Manage Jenkins
- My Views

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
	☼	Maulin_Gitflow--3	28 min log	N/A	0.63 sec
	☼	Maulin_Gitflow_4	7 min 39 sec log	N/A	0.64 sec
✓	☼	sample-job	31 min #1	N/A	0.28 sec

Icon: S M L

CODE:

```
pipeline {
  agent any
  environment {
    REPO_URL = 'https://github.com/maulin-architect/Maulin_jenkins.git'
  }
  stages {
    stage('Checkout') {
      steps {
        script {
          // Checkout the correct branch based on trigger
          checkout scm
        }
    }
}



stage('Build') {
    steps {
        echo ' Building ...'
       // sh 'make build' // Or another build command
    }
}

stage('Unit Tests') {
    steps {
         echo ' Testing ...'
       // sh 'make test-unit' // Run unit tests
    }
}

stage('Static Code Analysis') {
    steps {
        echo 'static code analysis'
        // sh 'make lint' // Or another static analysis tool
    }
}

stage('Deploy to Staging') {

    when {
        branch 'develop'
    }
```

```
  steps {
    echo 'Deploying to Statging in case of develop branch only'
    //sh 'make deploy-staging' // Deploys develop branch to staging
  }
}

stage('Integration Tests') {
  when {
    branch 'develop'
  }
  steps {
    echo ' in Develop branch doing Integration Tests'
    // sh 'make test-integration' // Run integration tests on staging
  }
}

stage('Create Release Artifact') {
  when {
    branch pattern: "release/.*", comparator: "REGEXP"
  }
  steps {
     echo ' in Release branch creating Release Artifact'
    // sh 'make build-release' // Build the release artifact
   // archiveArtifacts artifacts: 'build/*.jar', fingerprint: true
  }
}

stage('Deploy to Production') {
  when {
    branch 'main'
  }
```

```
      steps {

        echo 'Deploying'
       // sh 'make deploy-prod' // Deploy main branch to production
      }
    }
  }

  post {
    success {
      echo 'Pipeline completed successfully!'
    }
    failure {
      echo 'Pipeline failed!'
    }
  }
}
```

Scan Multibranch Pipeline Log

```
Started by user admin
[Fri Nov 15 07:57:17 UTC 2024] Starting branch indexing...
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
 > git ls-remote --symref -- https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10
 > git rev-parse --resolve-git-dir /var/lib/jenkins/caches/git-99e2b04f2d6f44693da483d3fa17e50d/.git # timeout=10
Setting origin to https://github.com/maulin-architect/Maulin_jenkins.git
 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10
Fetching & pruning origin...
Listing remote references...
 > git config --get remote.origin.url # timeout=10
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
 > git ls-remote -h -- https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10
Fetching upstream changes from origin
 > git config --get remote.origin.url # timeout=10
 > git fetch --tags --force --progress --prune -- origin +refs/heads/*:refs/remotes/origin/* # timeout=10
Checking branches...
  Checking branch develop
      ‘Jenkinsfile’ found
    Met criteria
No changes detected: develop (still at d16e29de467d3ed18e7c4ce151854f7a212e4f31)
  Checking branch main
      ‘Jenkinsfile’ found
    Met criteria
No changes detected: main (still at 8daccb571d539b67637ee3b0cce48d5859258278)
  Checking branch feature
      ‘Jenkinsfile’ found
    Met criteria
No changes detected: feature (still at b6e839951cc59f8461d77631d5e9fe618e759b32)
  Checking branch release
      ‘Jenkinsfile’ found
    Met criteria
No changes detected: release (still at 98831d5ed3097fdac2b20e95b3ed96c514c9b064)
Processed 4 branches
[Fri Nov 15 07:57:18 UTC 2024] Finished branch indexing. Indexing took 0.64 sec
Finished: SUCCESS
```

CONSOLE OUTPUT

Branch Name: main

```
Branch indexing
```

```
 > git rev-parse --resolve-git-dir /var/lib/jenkins/caches/git-99e2b04f2d6f44693da483d3fa17e50d/.git # timeout=10

Setting origin to https://github.com/maulin-architect/Maulin_jenkins.git

 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10

Fetching origin...

Fetching upstream changes from origin

 > git --version # timeout=10

 > git --version # 'git version 2.40.1'

 > git config --get remote.origin.url # timeout=10

 > git fetch --tags --force --progress -- origin +refs/heads/*:refs/remotes/origin/* # timeout=10

Seen branch in repository origin/develop

Seen branch in repository origin/main

Seen 2 remote branches

Obtained Jenkinsfile from b6e839951cc59f8461d77631d5e9fe618e759b32

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in /var/lib/jenkins/workspace/Maulin_Gitflow_3_main

[Pipeline] {

[Pipeline] stage

[Pipeline] { (Declarative: Checkout SCM)

[Pipeline] checkout

Selected Git installation does not exist. Using Default

The recommended git tool is: NONE

No credentials specified

Cloning the remote Git repository

Cloning with configured refspecs honoured and without tags

Cloning repository https://github.com/maulin-architect/Maulin_jenkins.git

 > git init /var/lib/jenkins/workspace/Maulin_Gitflow_3_main # timeout=10

Fetching upstream changes from https://github.com/maulin-architect/Maulin_jenkins.git

 > git --version # timeout=10

 > git --version # 'git version 2.40.1'
 > git fetch --no-tags --force --progress -- https://github.com/maulin-architect/Maulin_jenkins.git +refs/heads/*:refs/remotes/origin/* # timeout=10

 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10

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

Avoid second fetch

Checking out Revision b6e839951cc59f8461d77631d5e9fe618e759b32 (main)

 > git config core.sparsecheckout # timeout=10

 > git checkout -f b6e839951cc59f8461d77631d5e9fe618e759b32 # timeout=10

Commit message: "Merge pull request #1 from maulin-architect/develop"

First time build. Skipping changelog.

[Pipeline] }

[Pipeline] // stage

[Pipeline] withEnv

[Pipeline] {

[Pipeline] withEnv

[Pipeline] {

[Pipeline] stage

[Pipeline] { (Checkout)

[Pipeline] script

[Pipeline] {

[Pipeline] checkout

Selected Git installation does not exist. Using Default

The recommended git tool is: NONE

No credentials specified

 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Maulin_Gitflow_3_main/.git # timeout=10

Fetching changes from the remote Git repository

 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10

Fetching without tags

Fetching upstream changes from https://github.com/maulin-architect/Maulin_jenkins.git

 > git --version # timeout=10

 > git --version # 'git version 2.40.1'
 > git fetch --no-tags --force --progress -- https://github.com/maulin-architect/Maulin_jenkins.git
+refs/heads/*:refs/remotes/origin/* # timeout=10

Checking out Revision b6e839951cc59f8461d77631d5e9fe618e759b32 (main)

 > git config core.sparsecheckout # timeout=10

 > git checkout -f b6e839951cc59f8461d77631d5e9fe618e759b32 # timeout=10

Commit message: "Merge pull request #1 from maulin-architect/develop"

[Pipeline] }

[Pipeline] // script

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Build)

[Pipeline] echo

 Building ...

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Unit Tests)

[Pipeline] echo

 Testing ...

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Static Code Analysis)

[Pipeline] echo

static code analysis

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Deploy to Staging)

Stage "Deploy to Staging" skipped due to when conditional
```

```
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Integration Tests)
Stage "Integration Tests" skipped due to when conditional
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Create Release Artifact)
Stage "Create Release Artifact" skipped due to when conditional
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to Production)
[Pipeline] echo
Deploying
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline completed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
```

Finished: SUCCESS

Branch Name: develop

```
Branch indexing
 > git rev-parse --resolve-git-dir /var/lib/jenkins/caches/git-99e2b04f2d6f44693da483d3fa17e50d/.git # timeout=10
Setting origin to https://github.com/maulin-architect/Maulin_jenkins.git
 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10
Fetching origin...
Fetching upstream changes from origin
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
 > git config --get remote.origin.url # timeout=10
 > git fetch --tags --force --progress -- origin +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/develop
Seen branch in repository origin/main
Seen 2 remote branches
Obtained Jenkinsfile from d16e29de467d3ed18e7c4ce151854f7a212e4f31
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Maulin_Gitflow_3_develop
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning with configured refspecs honoured and without tags
Cloning repository https://github.com/maulin-architect/Maulin_jenkins.git
 > git init /var/lib/jenkins/workspace/Maulin_Gitflow_3_develop # timeout=10
Fetching upstream changes from https://github.com/maulin-architect/Maulin_jenkins.git
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
 > git fetch --no-tags --force --progress -- https://github.com/maulin-architect/Maulin_jenkins.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10
 > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
Checking out Revision d16e29de467d3ed18e7c4ce151854f7a212e4f31 (develop)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f d16e29de467d3ed18e7c4ce151854f7a212e4f31 # timeout=10
Commit message: "Update Jenkinsfile"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Checkout)
[Pipeline] script
[Pipeline] {
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Maulin_Gitflow_3_develop/.git #
timeout=10
Fetching changes from the remote Git repository
```

```
 > git config remote.origin.url https://github.com/maulin-architect/Maulin_jenkins.git # timeout=10

Fetching without tags

Fetching upstream changes from https://github.com/maulin-architect/Maulin_jenkins.git

 > git --version # timeout=10

 > git --version # 'git version 2.40.1'

 > git fetch --no-tags --force --progress -- https://github.com/maulin-architect/Maulin_jenkins.git
+refs/heads/*:refs/remotes/origin/* # timeout=10

Checking out Revision d16e29de467d3ed18e7c4ce151854f7a212e4f31 (develop)

 > git config core.sparsecheckout # timeout=10

 > git checkout -f d16e29de467d3ed18e7c4ce151854f7a212e4f31 # timeout=10

Commit message: "Update Jenkinsfile"

[Pipeline] }

[Pipeline] // script

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Build)

[Pipeline] echo

 Building ...

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Unit Tests)

[Pipeline] echo

 Testing ...

[Pipeline] }

[Pipeline] // stage

[Pipeline] stage

[Pipeline] { (Static Code Analysis)

[Pipeline] echo

static code analysis
```

```
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to Staging)
[Pipeline] echo
Deploying to Statging in case of develop branch only
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Integration Tests)
[Pipeline] echo
 in Develop branch doing Integration Tests
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Create Release Artifact)
Stage "Create Release Artifact" skipped due to when conditional
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to Production)
Stage "Deploy to Production" skipped due to when conditional
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline completed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
```

```
[Pipeline] // withEnv

[Pipeline] }

[Pipeline] // withEnv

[Pipeline] }

[Pipeline] // node

[Pipeline] End of Pipeline

Finished: SUCCESS
```

Writer :: Maulin Bhatt