Jenkins Task 2

1 - What is Jenkins pipeline?

2 - What scripting language is Jenkins pipeline syntax based on?

3 - what are the different ways to trigger pipeline ?

4 - what is different between parameter and jenkins env variable?

5 - what is organization folder job and what is used for ?

6 - Create jenkins pipeline for your repo and use script file (jenkinsfile) to write pipeline syntax, include parameter functions and env variable in it ?

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

The Solution

1 - What is Jenkins pipeline?

Jenkins Pipeline is a suite of plugins that supports the implementation and integration of continuous delivery pipelines into Jenkins. It allows you to define and manage your build, test, and deployment processes as code. Jenkins Pipeline provides a way to create workflows that can be versioned, shared, and executed in a robust and scalable manner.

2 - What scripting language is Jenkins pipeline syntax based on?

The scripting language used in Jenkins Pipeline syntax is called Groovy. Groovy is a powerful, dynamic language that runs on the Java Virtual Machine (JVM) and provides a concise and expressive syntax for writing Jenkins Pipelines.

3 - What are the different ways to trigger pipeline ?

There are several ways to trigger a Jenkins Pipeline:

A - Triggering the pipeline manually by clicking on a button or link in the Jenkins web interface.

B - Triggering the pipeline automatically on a schedule using cron syntax.

C - Triggering the pipeline when changes are pushed to a version control system (e.g., Git) by configuring webhooks or polling.

D - Triggering the pipeline when a specific event occurs, such as a new artifact being published or an external system triggering a webhook.

4 - What is different between parameter and jenkins env variable?

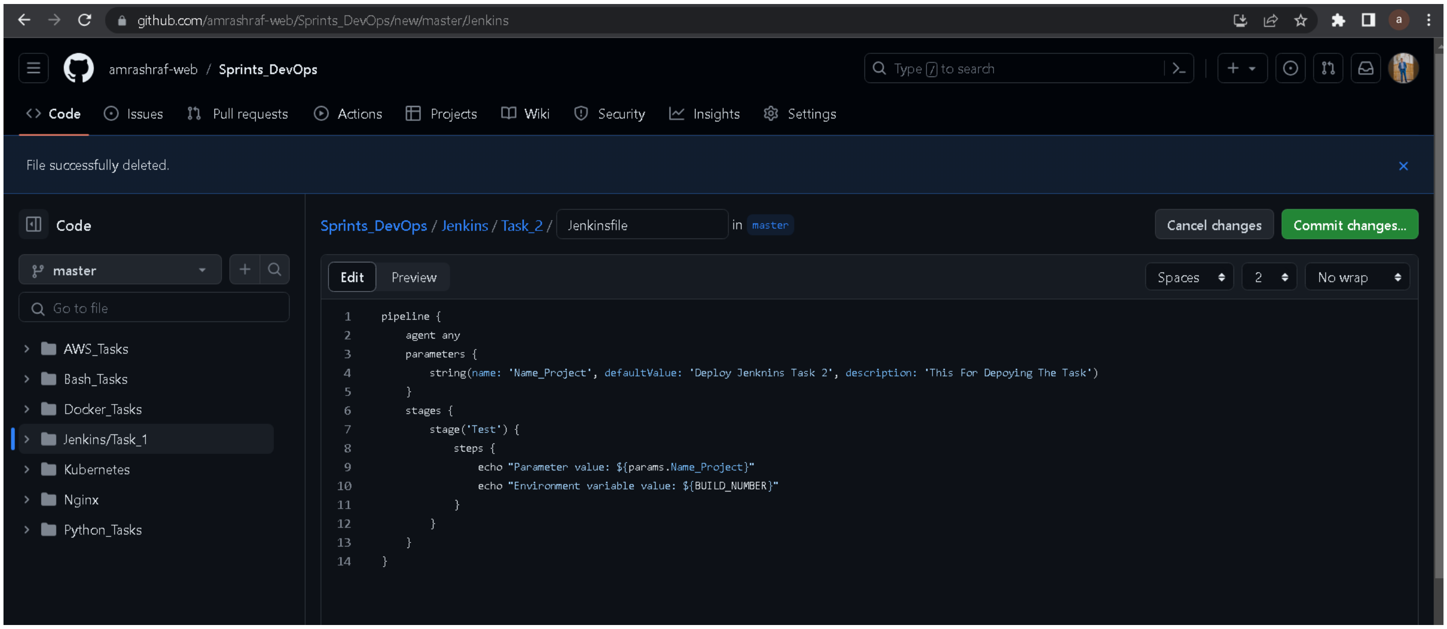
A - Parameters in Jenkins are values that can be passed into a pipeline at runtime. They allow users to customize the behavior of the pipeline by providing input values.

B - Jenkins environment variables, on the other hand, are predefined variables that store information about the Jenkins environment. They can be accessed and used within a pipeline to provide contextual information or configure specific behavior

5 - What is organization folder job and what is used for ?

An organization folder job in Jenkins is a special type of job that allows you to organize and manage pipelines for multiple repositories or projects within a single Jenkins job. It provides a way to create and manage pipelines dynamically based on the repositories or projects present in a specified source code management system (e.g., Git).

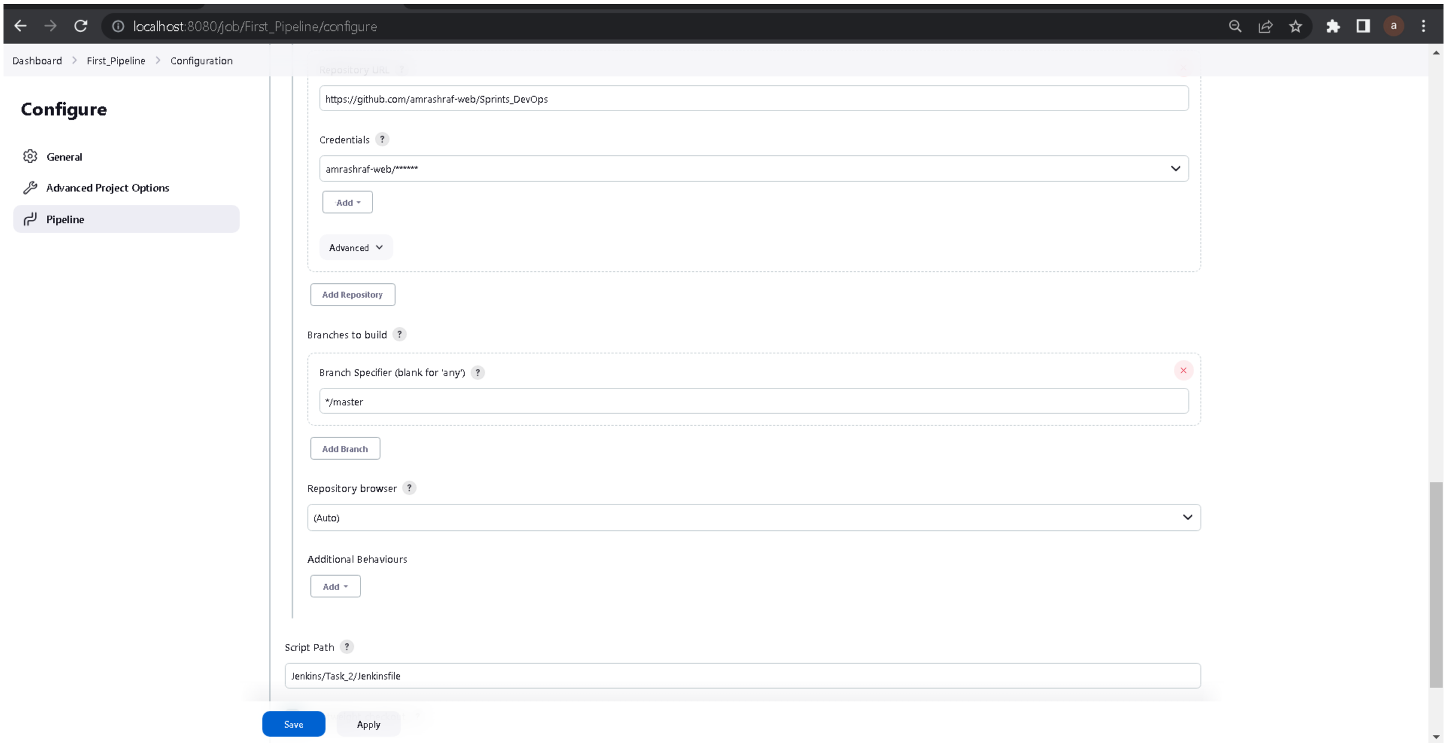
6 - Create jenkins pipeline for your repo and use script file (Jenkinsfile) to write pipeline syntax, include parameter functions and env variable in it ?



File successfully deleted.

Sprints_DevOps / Jenkins / Task_2 / Jenkinsfile in master

```
pipeline {
    agent any
    parameters {
        string(name: 'Name_Project', defaultValue: 'Deploy Jenknins Task 2', description: 'This For Depoying The Task')
    }
    stages {
        stage('Test') {
            steps {
                echo "Parameter value: ${params.Name_Project}"
                echo "Environment variable value: ${BUILD_NUMBER}"
            }
        }
    }
}
```



localhost:8080/job/First_Pipeline/configure
Dashboard > First_Pipeline > Configuration

Configure

Repository URL ?

https://github.com/amrashraf-web/Sprints_DevOps

Credentials ?

amrashraf-web/******

Branches to build ?

Branch Specifier (blank for 'any') ?

*/master

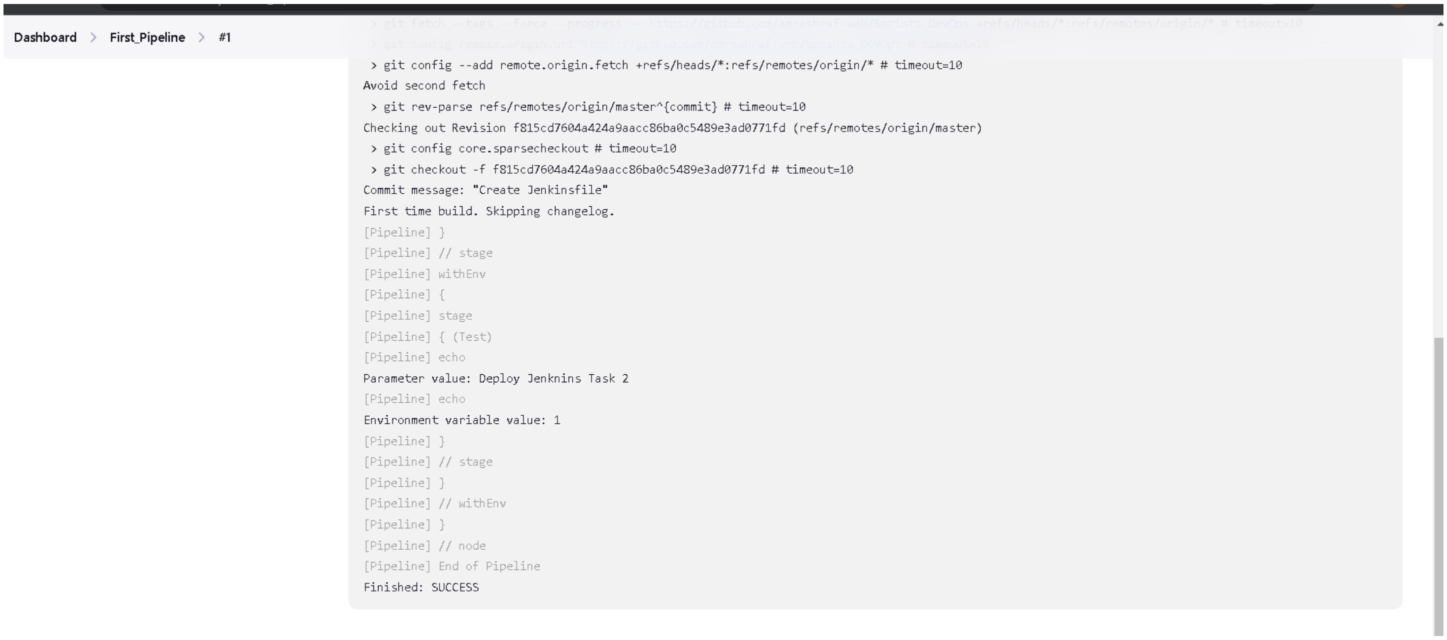
Repository browser ?

(Auto)

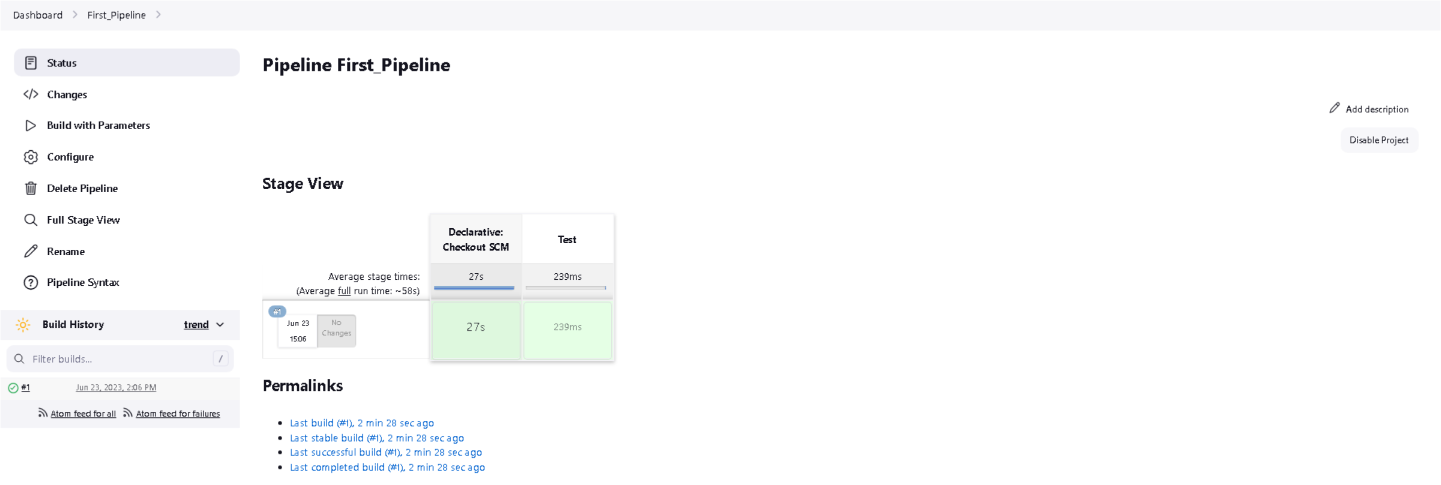
Additional Behaviours

Script Path ?

Jenkins/Task_2/Jenkinsfile



```
 > git fetch --tags --force --progress -- https://github.com/amrashraf-web/Sprints_DevOps +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git config remote.origin.url https://github.com/amrashraf-web/Sprints_DevOps # timeout=10
 > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision f815cd7604a424a9aacc86ba0c5489e3ad0771fd (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f f815cd7604a424a9aacc86ba0c5489e3ad0771fd # timeout=10
Commit message: "Create Jenkinsfile"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] echo
Parameter value: Deploy Jenknins Task 2
[Pipeline] echo
Environment variable value: 1
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```



Pipeline First_Pipeline

Stage View

	Declarative: Checkout SCM	Test
Average stage times: (Average full run time: ~58s)	27s	239ms
#1 Jun 23 15:06 No Changes	27s	239ms

#1 Jun 23, 2023, 2:06 PM

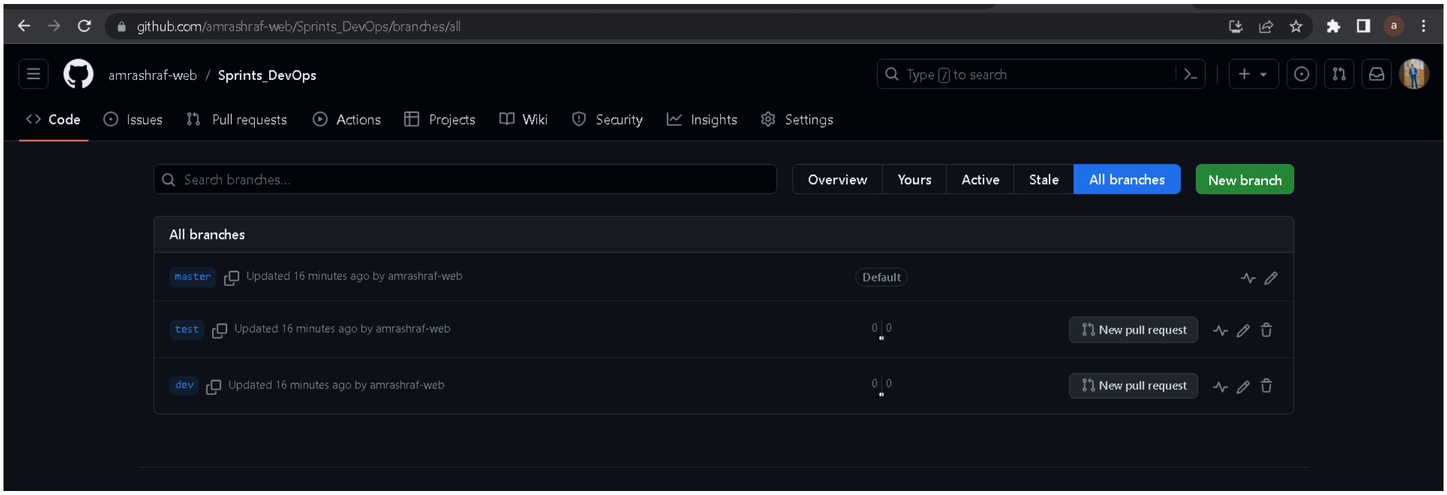
Atom feed for all Atom feed for failures

Permalinks

- Last build (#1), 2 min 28 sec ago
- Last stable build (#1), 2 min 28 sec ago
- Last successful build (#1), 2 min 28 sec ago
- Last completed build (#1), 2 min 28 sec ago

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

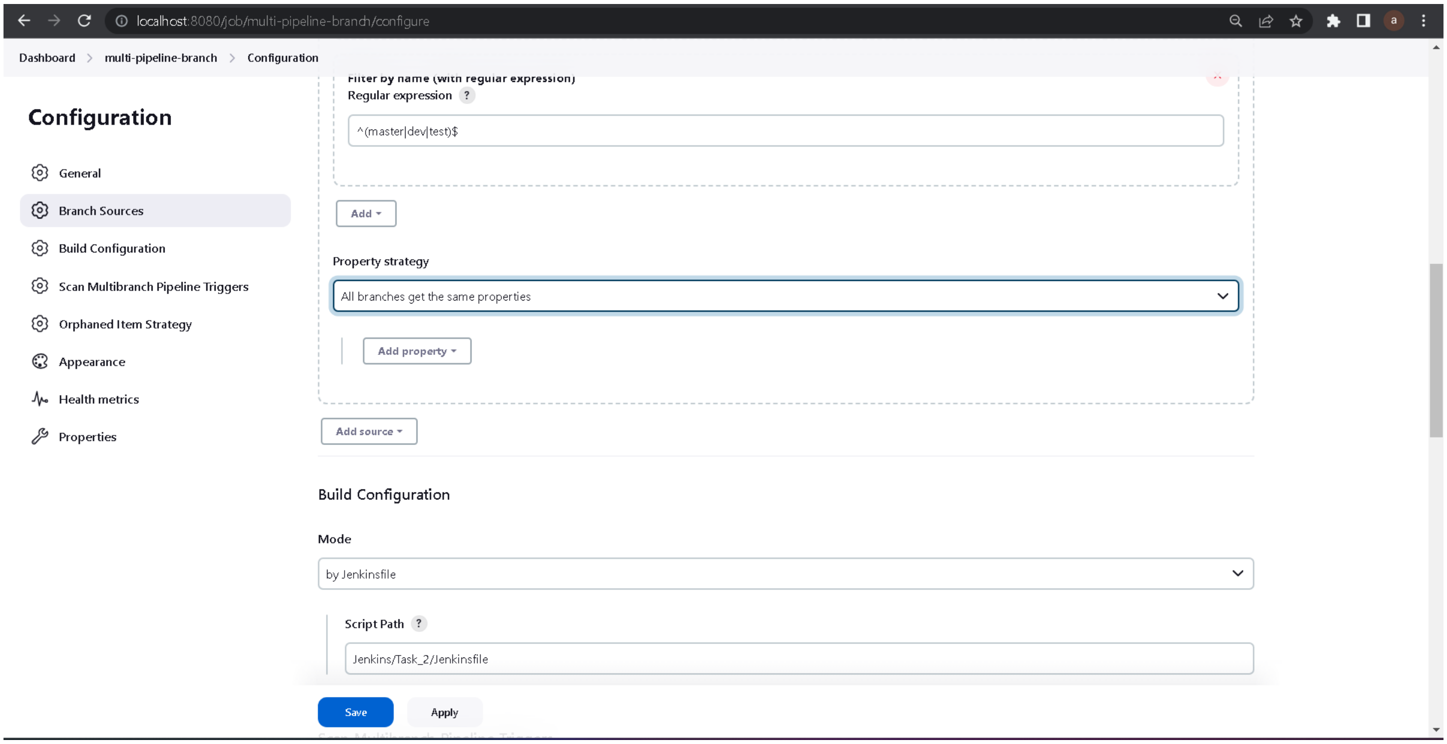
A – create two branch (dev , test)



All branches

master	Updated 16 minutes ago by amrashraf-web	Default	
test	Updated 16 minutes ago by amrashraf-web	0 | 0	New pull request
dev	Updated 16 minutes ago by amrashraf-web	0 | 0	New pull request

Create multi branch pipeline on jenkins with filter branch by name



Filter by name (with regular expression)

Regular expression

^(master|dev|test)$

Property strategy

All branches get the same properties

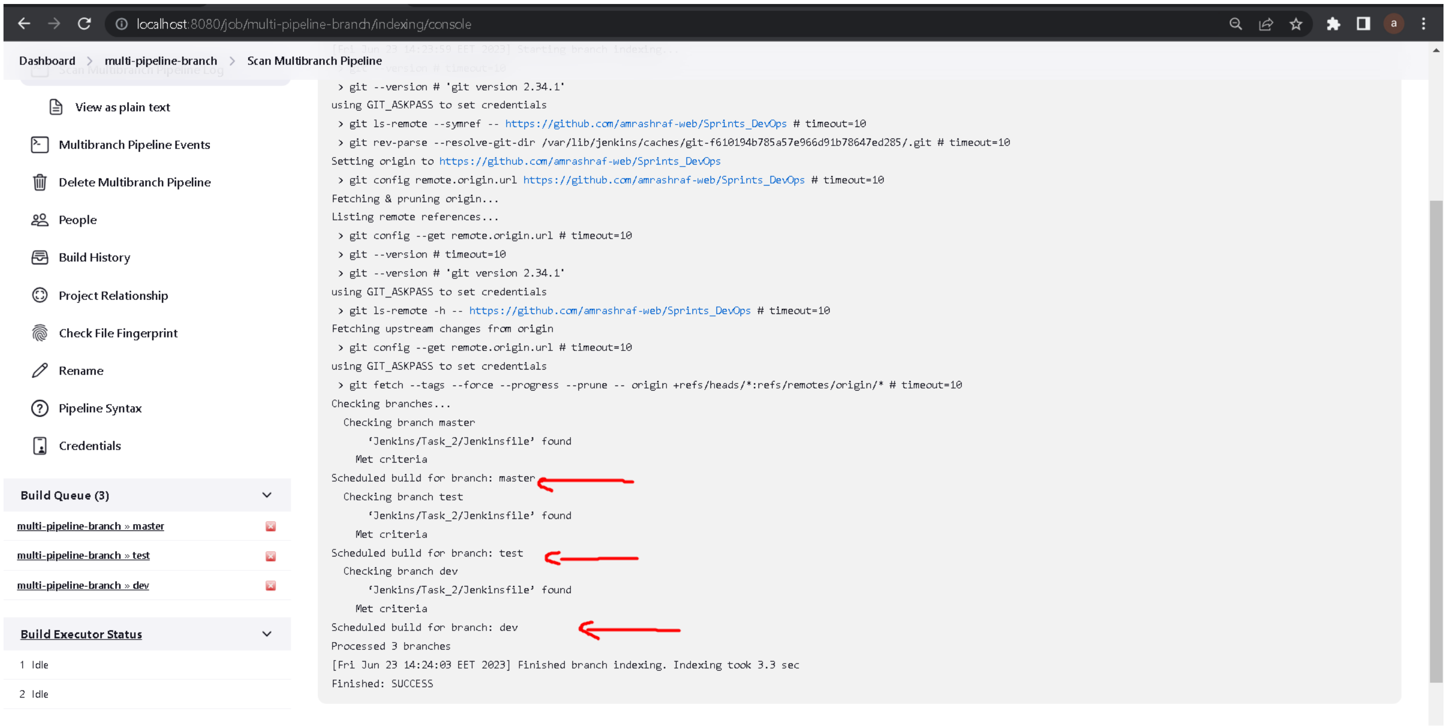
Build Configuration

Mode

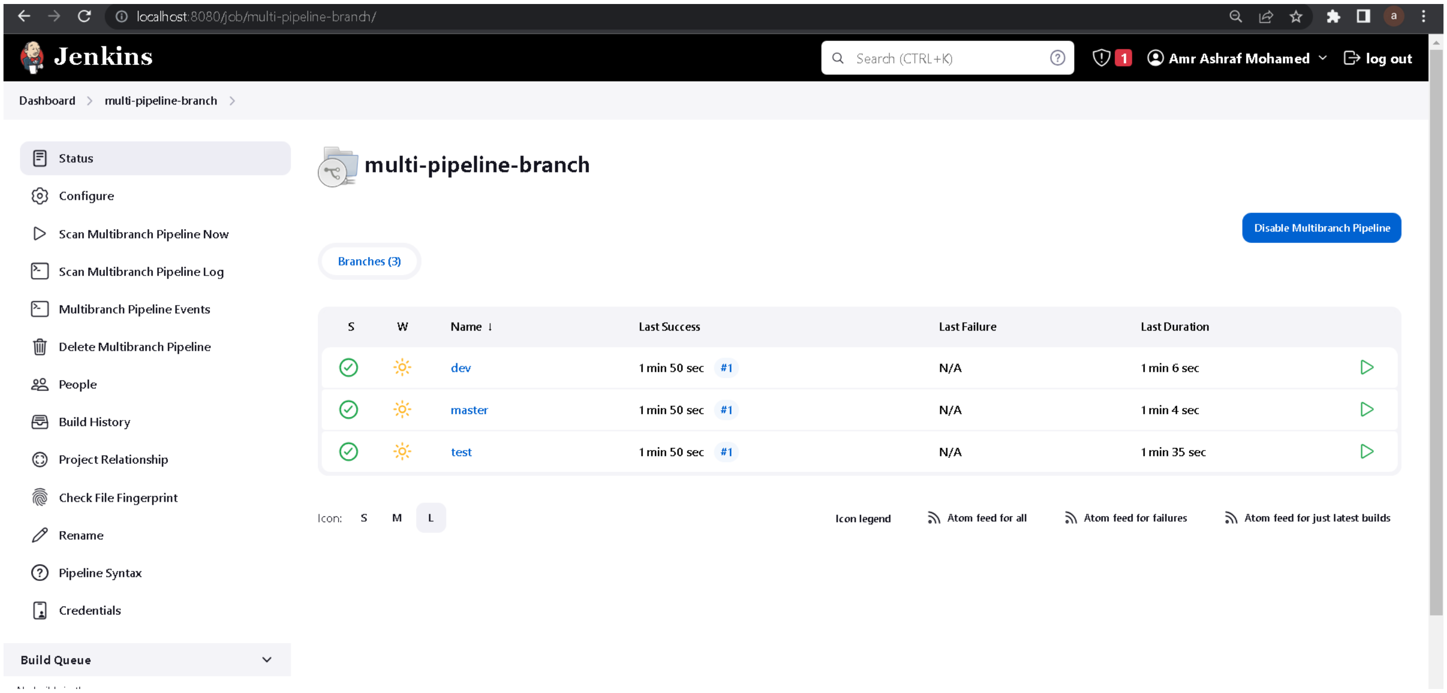
by Jenkinsfile

Script Path

Jenkins/Task_2/Jenkinsfile


localhost:8080/job/multi-pipeline-branch/indexing/console
Dashboard > multi-pipeline-branch > Scan Multibranch Pipeline
View as plain text
Multibranch Pipeline Events
Delete Multibranch Pipeline
People
Build History
Project Relationship
Check File Fingerprint
Rename
Pipeline Syntax
Credentials
Build Queue (3)
multi-pipeline-branch » master
multi-pipeline-branch » test
multi-pipeline-branch » dev
Build Executor Status
1 Idle
2 Idle

```
[Fri Jun 23 14:23:59 EET 2023] Starting branch indexing...
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_ASKPASS to set credentials
 > git ls-remote --symref -- https://github.com/amrashraf-web/Sprints_DevOps # timeout=10
 > git rev-parse --resolve-git-dir /var/lib/jenkins/caches/git-f610194b785a57e966d91b78647ed285/.git # timeout=10
Setting origin to https://github.com/amrashraf-web/Sprints_DevOps
 > git config remote.origin.url https://github.com/amrashraf-web/Sprints_DevOps # timeout=10
Fetching & pruning origin...
Listing remote references...
 > git config --get remote.origin.url # timeout=10
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_ASKPASS to set credentials
 > git ls-remote -h -- https://github.com/amrashraf-web/Sprints_DevOps # timeout=10
Fetching upstream changes from origin
 > git config --get remote.origin.url # timeout=10
using GIT_ASKPASS to set credentials
 > git fetch --tags --force --progress --prune -- origin +refs/heads/*:refs/remotes/origin/* # timeout=10
Checking branches...
  Checking branch master
      'Jenkins/Task_2/Jenkinsfile' found
    Met criteria
Scheduled build for branch: master
  Checking branch test
      'Jenkins/Task_2/Jenkinsfile' found
    Met criteria
Scheduled build for branch: test
  Checking branch dev
      'Jenkins/Task_2/Jenkinsfile' found
    Met criteria
Scheduled build for branch: dev
Processed 3 branches
[Fri Jun 23 14:24:03 EET 2023] Finished branch indexing. Indexing took 3.3 sec
Finished: SUCCESS
```



Dashboard > multi-pipeline-branch >

multi-pipeline-branch

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☀	dev	1 min 50 sec #1	N/A	1 min 6 sec	▷
✓	☀	master	1 min 50 sec #1	N/A	1 min 4 sec	▷
✓	☀	test	1 min 50 sec #1	N/A	1 min 35 sec	▷