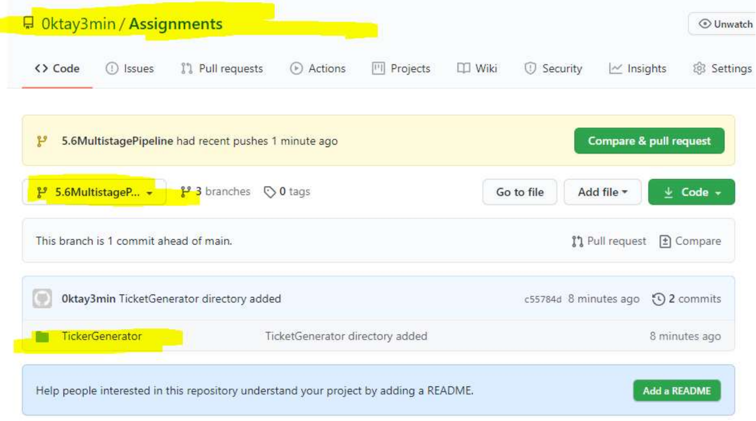
Assignment 5.6 Multistage Pipelines

Saturday, January 2, 2021 8:37 AM

Assignment 5.6 Multistage Pipelines

1-Create a Github Repository



0ktay3min / Assignments

5.6MultistagePipeline had recent pushes 1 minute ago

5.6MultistageP... 3 branches 0 tags

This branch is 1 commit ahead of main.

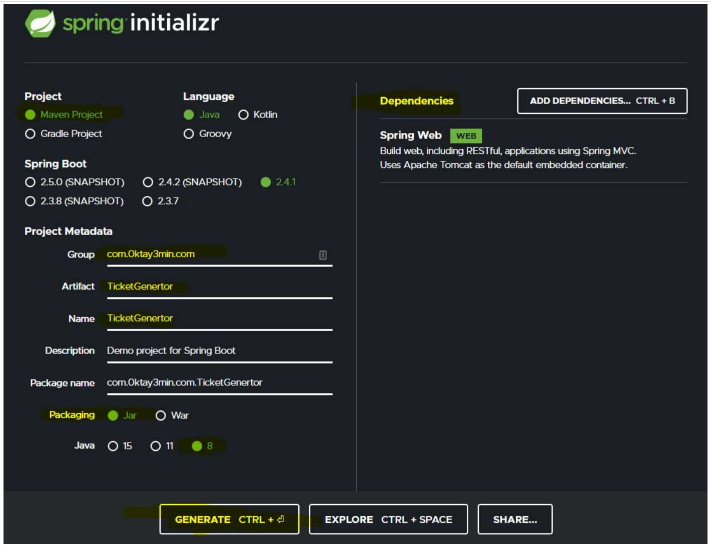
0ktay3min TicketGenerator directory added c55784d 8 minutes ago 2 commits

TickerGenerator TicketGenerator directory added 8 minutes ago

Help people interested in this repository understand your project by adding a README.

2-Go to www.start.spring.io/ and enter the following

- Select Maven as a project type
- Enter Group and Artifacts with appropriate values
- Add Web Dependencies
- Select Java 8
- Click Generator Project



spring initializr

Project
● Maven Project
○ Gradle Project

Language
● Java ○ Kotlin
○ Groovy

Spring Boot
○ 2.5.0 (SNAPSHOT) ○ 2.4.2 (SNAPSHOT) ● 2.4.1
○ 2.3.8 (SNAPSHOT) ○ 2.3.7

Project Metadata

Group com.0ktay3min.com

Artifact TicketGenertor

Name TicketGenertor

Description Demo project for Spring Boot

Package name com.0ktay3min.com.TicketGenertor

Packaging ● Jar ○ War

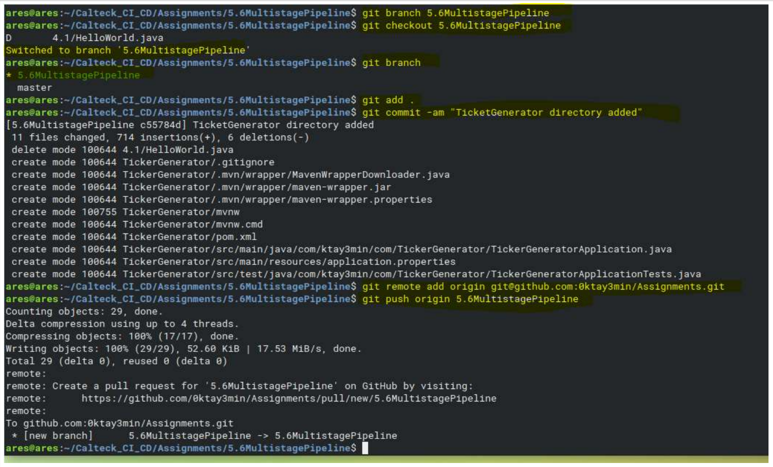
Java ○ 15 ○ 11 ● 8

Dependencies

Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

3-Save Downloaded folder and push it to your remote Git Repository

- git branch 5.6MultistagePipeline => to create a new branch called 5.6MultistagePipeline
- git checkout 5.6MultistagePipeline => to go inside the branch called 5.6MultistagePipeline
- git add . => to add all files and directories to staging area
- git commit -am "TicketGenerator Directory Added" => to commit changes
- git remote add origin git@github.com:0ktay3min/Assignments.git => to add the URL of the remote Repository
- git push origin 5.6MultistagePipleline => to push the changes in branch called 5.6MultistagePipleline to remote repository

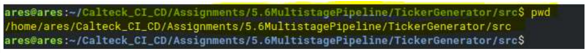


```
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git branch 5.6MultistagePipeline
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git checkout 5.6MultistagePipeline
D       4.1/HelloWorld.java
Switched to branch '5.6MultistagePipeline'
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git branch
* 5.6MultistagePipeline
  master
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git add .
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git commit -am "TicketGenerator directory added"
[5.6MultistagePipeline c55784d] TicketGenerator directory added
 11 files changed, 714 insertions(+), 6 deletions(-)
 delete mode 100644 4.1/HelloWorld.java
 create mode 100644 TickerGenerator/.gitignore
 create mode 100644 TickerGenerator/.mvn/wrapper/MavenWrapperDownloader.java
 create mode 100644 TickerGenerator/.mvn/wrapper/maven-wrapper.jar
 create mode 100644 TickerGenerator/.mvn/wrapper/maven-wrapper.properties
 create mode 100755 TickerGenerator/mvnw
 create mode 100644 TickerGenerator/mvnw.cmd
 create mode 100644 TickerGenerator/pom.xml
 create mode 100644 TickerGenerator/src/main/java/com/ktay3min/com/TickerGenerator/TickerGeneratorApplication.java
 create mode 100644 TickerGenerator/src/main/resources/application.properties
 create mode 100644 TickerGenerator/src/test/java/com/ktay3min/com/TickerGenerator/TickerGeneratorApplicationTests.java
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git remote add origin git@github.com:0ktay3min/Assignments.git
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$ git push origin 5.6MultistagePipeline
Counting objects: 29, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (17/17), done.
Writing objects: 100% (29/29), 52.60 KiB | 17.53 MiB/s, done.
Total 29 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for '5.6MultistagePipeline' on GitHub by visiting:
remote:      https://github.com/0ktay3min/Assignments/pull/new/5.6MultistagePipeline
remote:
To github.com:0ktay3min/Assignments.git
 * [new branch]      5.6MultistagePipeline -> 5.6MultistagePipeline
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline$
```

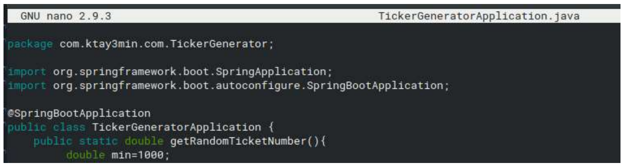
4-We need to add java code inside src folder

- Navigate to the 5.6MultistagePipeline/TickerGenerator/src/main/java/com/ktay3min/com/TickerGenerator folder
- Open the file called TickerGeneratorApplication.java using any text editor and add the following

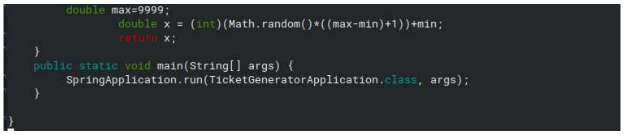
```
package com.ktay3min.com.TickerGenerator;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class TickerGeneratorApplication {
```



```
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator/src$ pwd
/home/ares/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator/src
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator/src$
```



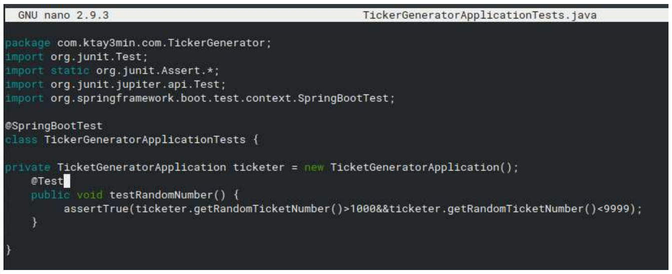
```
  GNU nano 2.9.3                                   TickerGeneratorApplication.java

package com.ktay3min.com.TickerGenerator;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class TickerGeneratorApplication {
    public static double getRandomTicketNumber(){
        double min=1000;
        public static double getRandomTicketNumber(){
            double min=1000;
            double max=9999;
                double x = (int)(Math.random()*((max-min)+1))+min;
                return x;
        }
        public static void main(String[] args) {
            SpringApplication.run(TicketGeneratorApplication.class, args);
        }
    }
```



```
        double max=9999;
                double x = (int)(Math.random()*((max-min)+1))+min;
                return x;
    }
    public static void main(String[] args) {
        SpringApplication.run(TicketGeneratorApplication.class, args);
    }

}
```

- Navigate to the 5.6MultistagePipeline/TickerGenerator/src/test/java/com/ktay3min/com/TickerGenerator
- Open the file called TickerGeneratorApplication.java using any text editor and add the following

```
package com.ktay3min.com.TickerGenerator;
import org.junit.Test;
import static org.junit.Assert.*;
import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class TickerGeneratorApplicationTests {

private TicketGeneratorApplication ticketer = new
TicketGeneratorApplication();
    @Test
    public void testRandomNumber() {
        assertTrue(ticketer.getRandomTicketNumber()>1000
&&ticketer.getRandomTicketNumber()<9999);
    }
}
```

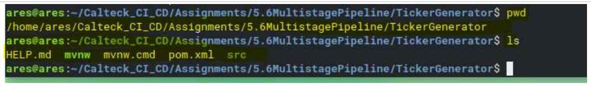


```
  GNU nano 2.9.3                                    TickerGeneratorApplicationTests.java

package com.ktay3min.com.TickerGenerator;
import org.junit.Test;
import static org.junit.Assert.*;
import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class TickerGeneratorApplicationTests {

private TicketGeneratorApplication ticketer = new TicketGeneratorApplication();
    @Test
    public void testRandomNumber() {
        assertTrue(ticketer.getRandomTicketNumber()>1000&&ticketer.getRandomTicketNumber()<9999);
    }

}
```

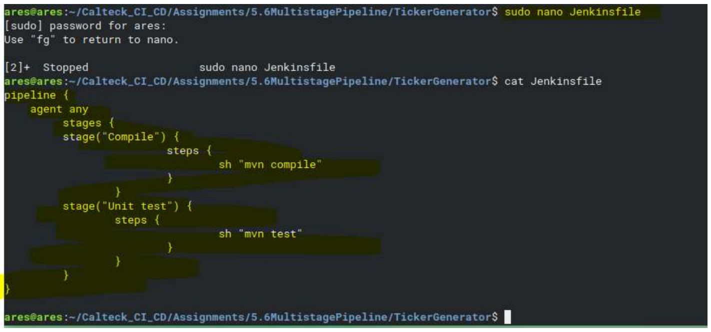
5- Navigate to the TicketGenerator directory and create a Jenkinsfile where pom.xml is located

- $ sudo nano Jenkinsfile => this will create a Jenkinsfile using nano text editor.

```
pipeline {
    agent any
        stages {
        stage("Compile") {
                steps {
                    sh "mvn compile"
                }
        }
        stage("Unit test") {
            steps {
                sh "mvn test"
                }
            }
        }
}
```



```
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ pwd
/home/ares/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ ls
HELP.md  mvnw  mvnw.cmd  pom.xml  src
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$
```

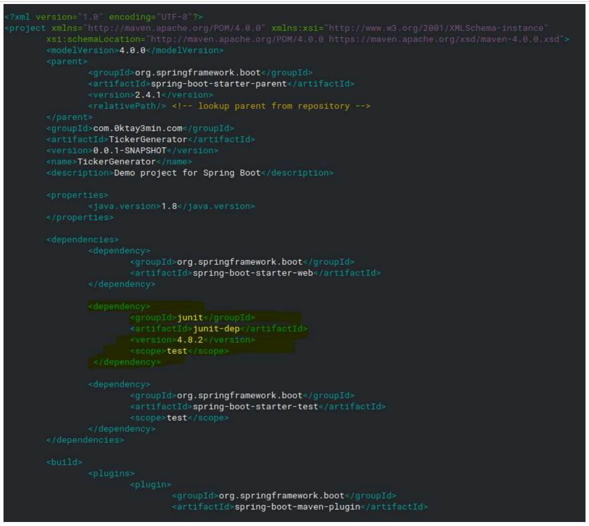


```
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ sudo nano Jenkinsfile
[sudo] password for ares:
Use "fg" to return to nano.

[2]+  Stopped                 sudo nano Jenkinsfile
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ cat Jenkinsfile
pipeline {
    agent any
         stages {
         stage("Compile") {
                        steps {
                                sh "mvn compile"
                        }
                }
         stage("Unit test") {
                steps {
                                sh "mvn test"
                        }
                }
         }
}

ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$
```

6- Now open the pom.xml file and add the following dependency

- $ sudo nano pom.xml => to open pom.xml file using nano text editor
- Add these dependencies

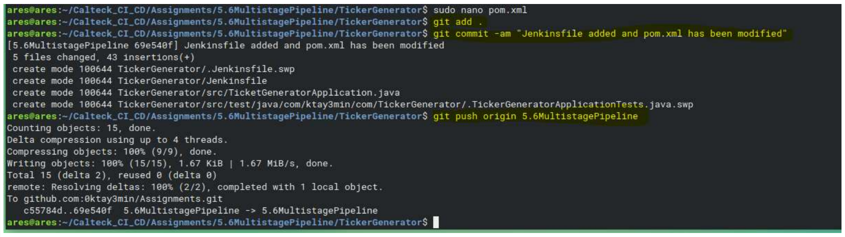
```
<dependency>
            <groupId>junit</groupId>
            <artifactId>junit-dep</artifactId>
            <version>4.8.2</version>
            <scope>test</scope>
          </dependency>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
        <modelVersion>4.0.0</modelVersion>
        <parent>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-starter-parent</artifactId>
                <version>2.4.1</version>
                <relativePath/> <!-- lookup parent from repository -->
        </parent>
        <groupId>com.0ktay3min.com</groupId>
        <artifactId>TickerGenerator</artifactId>
        <version>0.0.1-SNAPSHOT</version>
        <name>TickerGenerator</name>
        <description>Demo project for Spring Boot</description>

        <properties>
                <java.version>1.8</java.version>
        </properties>

        <dependencies>
                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-web</artifactId>
                </dependency>

                <dependency>
                        <groupId>junit</groupId>
                        <artifactId>junit-dep</artifactId>
                        <version>4.8.2</version>
                        <scope>test</scope>
                 </dependency>

                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-test</artifactId>
                        <scope>test</scope>
                </dependency>
        </dependencies>

        <build>
                <plugins>
                        <plugin>
                                <groupId>org.springframework.boot</groupId>
                                <artifactId>spring-boot-maven-plugin</artifactId>
```

7-Commit changes and push them to remote repository

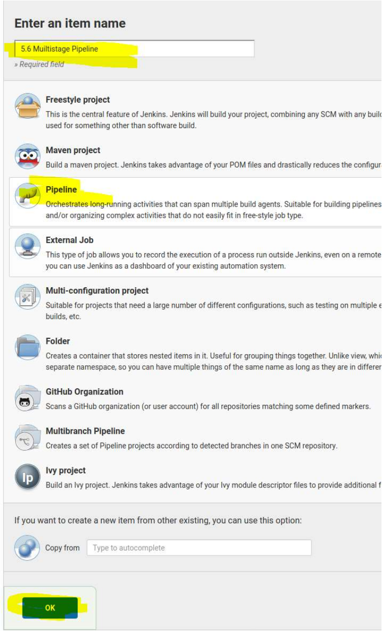
- $ git add .
- $ git commit -am "Jenkinsfile added and pom.xml has been modified"
- $ git push origin 5.6MultistagePipeline



```
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ sudo nano pom.xml
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ git add .
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ git commit -am "Jenkinsfile added and pom.xml has been modified"
[5.6MultistagePipeline 69e540f] Jenkinsfile added and pom.xml has been modified
 5 files changed, 43 insertions(+)
 create mode 100644 TickerGenerator/.Jenkinsfile.swp
 create mode 100644 TickerGenerator/Jenkinsfile
 create mode 100644 TickerGenerator/src/TicketGeneratorApplication.java
 create mode 100644 TickerGenerator/src/test/java/com/ktay3min/com/TickerGenerator/.TickerGeneratorApplicationTests.java.swp
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$ git push origin 5.6MultistagePipeline
Counting objects: 15, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (15/15), 1.67 KiB | 1.67 MiB/s, done.
Total 15 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:0ktay3min/Assignments.git
   c55784d..69e540f  5.6MultistagePipeline -> 5.6MultistagePipeline
ares@ares:~/Calteck_CI_CD/Assignments/5.6MultistagePipeline/TickerGenerator$
```

8-Now we need to create multistate pipeline in Jenkins

- Go to Jenkins dashboard
- Click "New Item"
- Enter the name of your build job
- Select Pipeline as a build job type



Enter an item name

5.6 Muiltistage Pipeline

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any buil
used for something other than software build.

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configur

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines
and/or organizing complex activities that do not easily fit in free-style job type.

External Job
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote
you can use Jenkins as a dashboard of your existing automation system.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple e
builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, whi
separate namespace, so you can have multiple things of the same name as long as they are in differer

GitHub Organization
Scans a GitHub organization (or user account) for all repositories matching some defined markers.

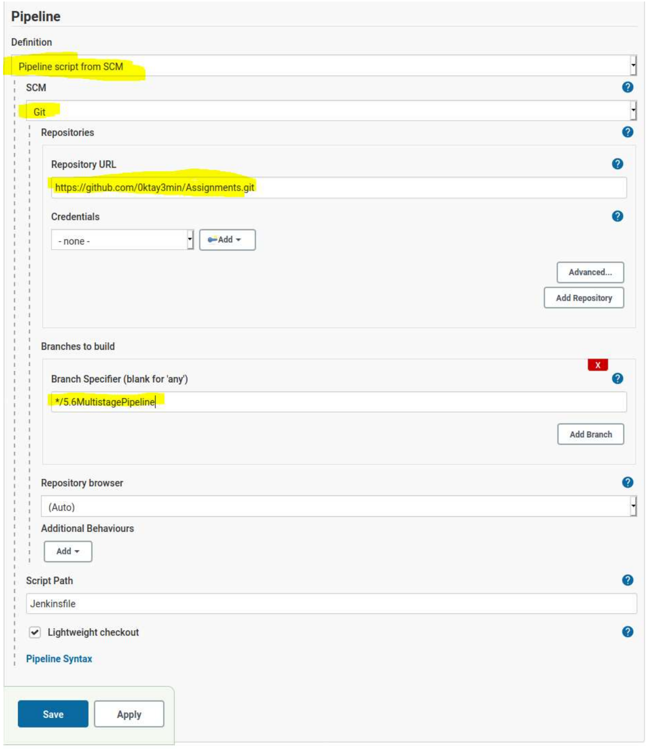
Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

Ivy project
Build an Ivy project. Jenkins takes advantage of your Ivy module descriptor files to provide additional f

If you want to create a new item from other existing, you can use this option:

Copy from: Type to autocomplete

- On the configuration page, scroll down to the Pipeline Section
- Change Definition from Pipeline script to Pipeline script from SC
- Select Git from SCM
- Add Github Repository URL
- Select correct branch name
- Click Save



Pipeline

Definition: Pipeline script from SCM

SCM: Git

Repositories

Repository URL: https://github.com/0ktay3min/Assignments.git

Credentials: - none -

Branches to build

Branch Specifier (blank for 'any'): */5.6MultistagePipeline

Repository browser: (Auto)

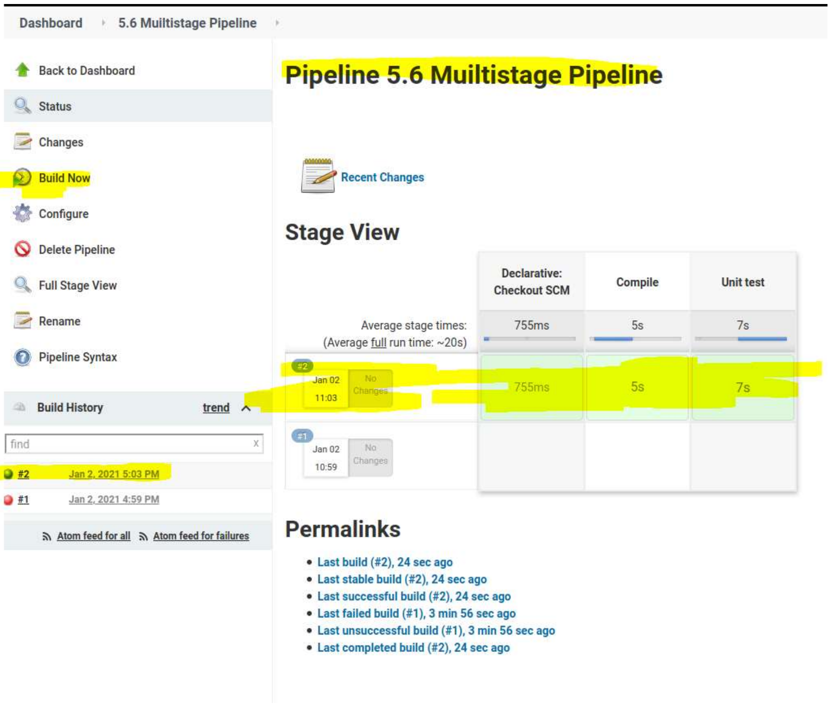
Additional Behaviours

Script Path: Jenkinsfile

☑ Lightweight checkout

Pipeline Syntax

9- Running a multistage pipeline in Jenkins
- Click Build Now in project window
- Jenkins will now build the Pipeline and output the logs



Pipeline 5.6 Muiltistage Pipeline

● #2	Jan 2, 2021 5:03 PM
● #1	Jan 2, 2021 4:59 PM

Stage View

	Declarative: Checkout SCM	Compile	Unit test
Average stage times: (Average full run time: ~20s)	755ms	5s	7s
#2 Jan 02 11:03 No Changes	755ms	5s	7s
#1 Jan 02 10:59 No Changes			

Permalinks

- Last build (#2), 24 sec ago
- Last stable build (#2), 24 sec ago
- Last successful build (#2), 24 sec ago
- Last failed build (#1), 3 min 56 sec ago
- Last unsuccessful build (#1), 3 min 56 sec ago
- Last completed build (#2), 24 sec ago