

# Day 5

## Lab - Creating DevOps Pipeline with multiple stages

In this lab exercises, we would be creating 3 FreeStyle Jobs manually and then we will configure it in a way it runs one after the other as a DevOps Pipeline.

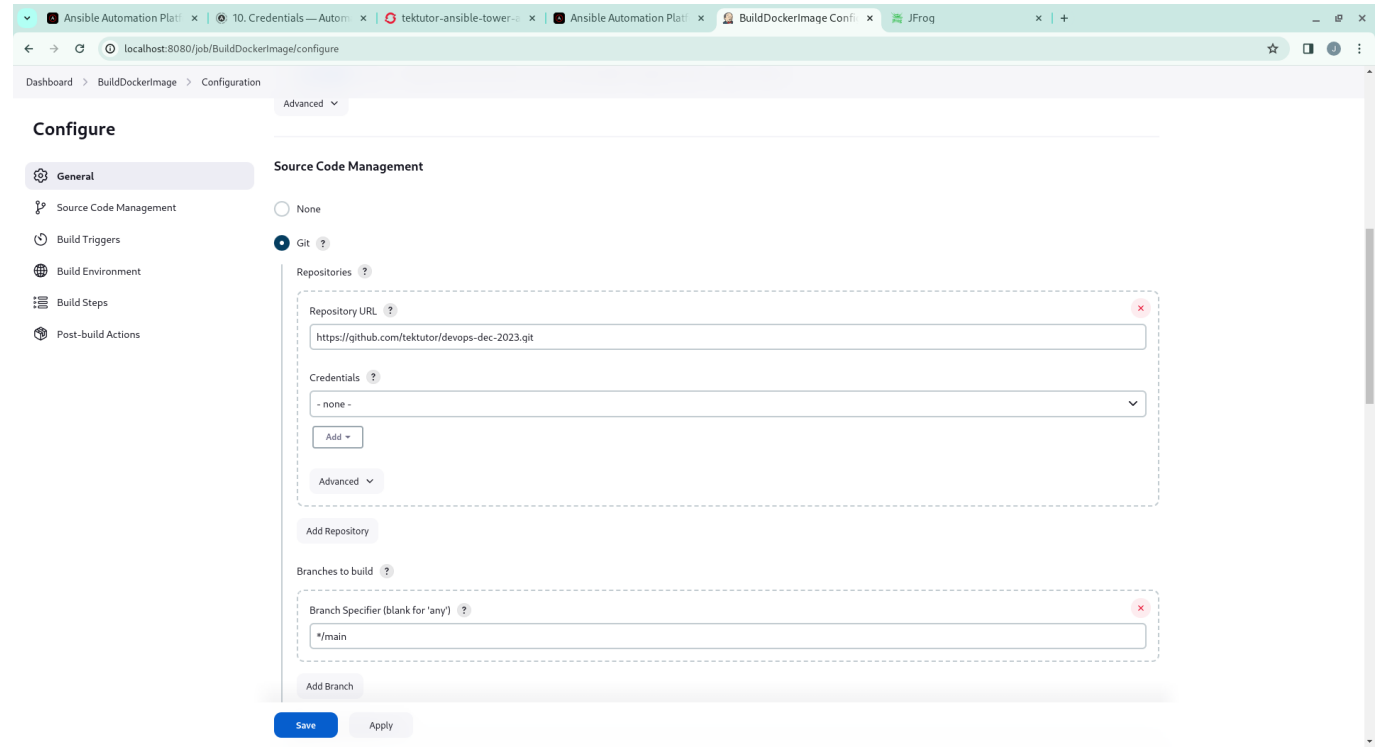
- First FreeStyle Job is BuildDockerImage ( Docker )
- Second FreeStyle Job is BuildMavenJobInSlaveDockerContainer ( Maven Build within Docker Container )
- Third FreeStyle Job is DownloadJarFromJFrogArtifactoryServerUsingAnsible ( Ansible Playbook )

### Creating the BuildDockerImage FreeStyle Job

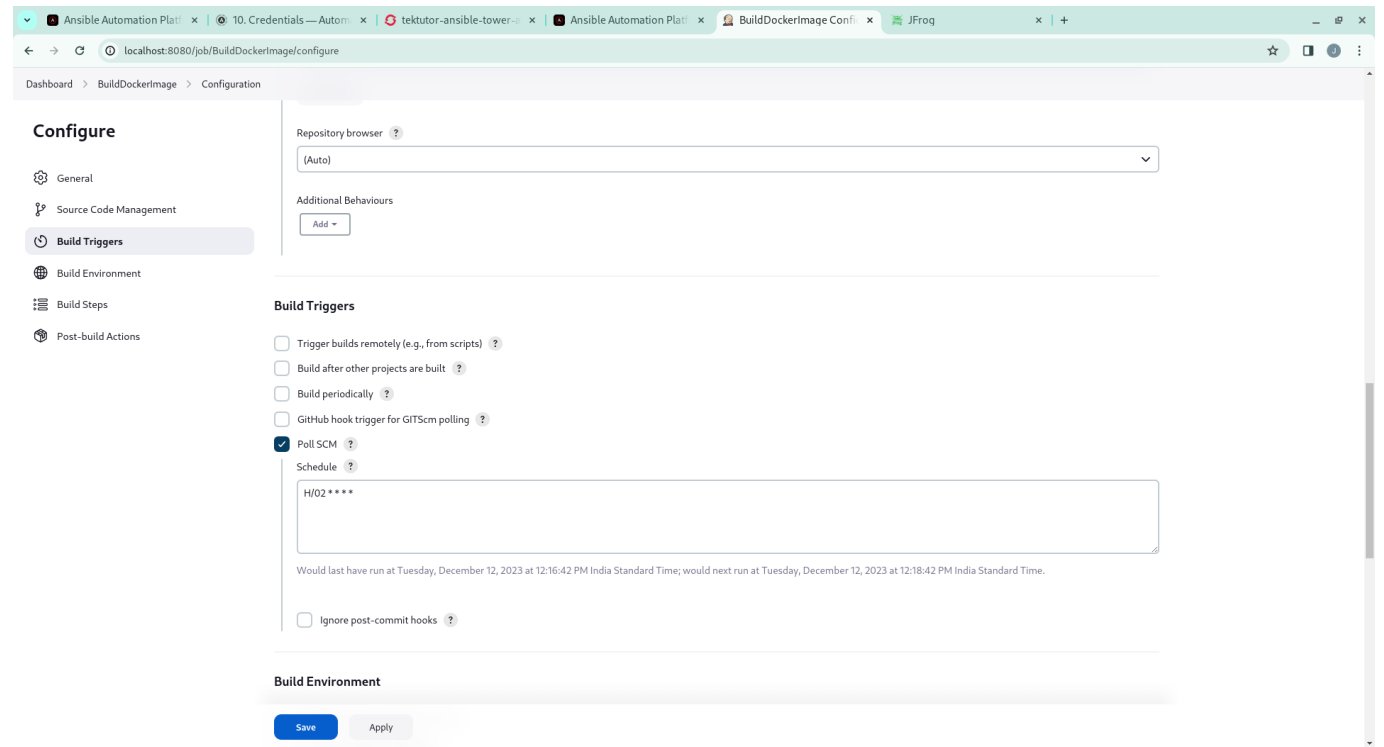
Let's create the First FreeStyle Job as shown below General Section

The screenshot shows the Jenkins web interface for configuring a new FreeStyle job named 'BuildDockerImage'. The browser address bar shows 'localhost:8080/job/BuildDockerImage/configure'. The Jenkins header includes the logo, a search bar, and the user 'Jeganathan Swaminathan'. The left sidebar shows the 'Configure' menu with options like General, Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The main content area is the 'General' tab, which is 'Enabled'. It features a 'Description' text area, a 'Plain text' preview, and several checkboxes: 'Commit agent's Docker container', 'Define a Docker template', 'Discard old builds', 'GitHub project', 'This project is parameterized', 'Throttle builds', 'Execute concurrent builds if necessary', and 'Restrict where this project can be run' (which is checked). Below the last checkbox is a 'Label Expression' field containing 'built-in'. A note at the bottom states 'Label built-in matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.' At the bottom are 'Save' and 'Apply' buttons.

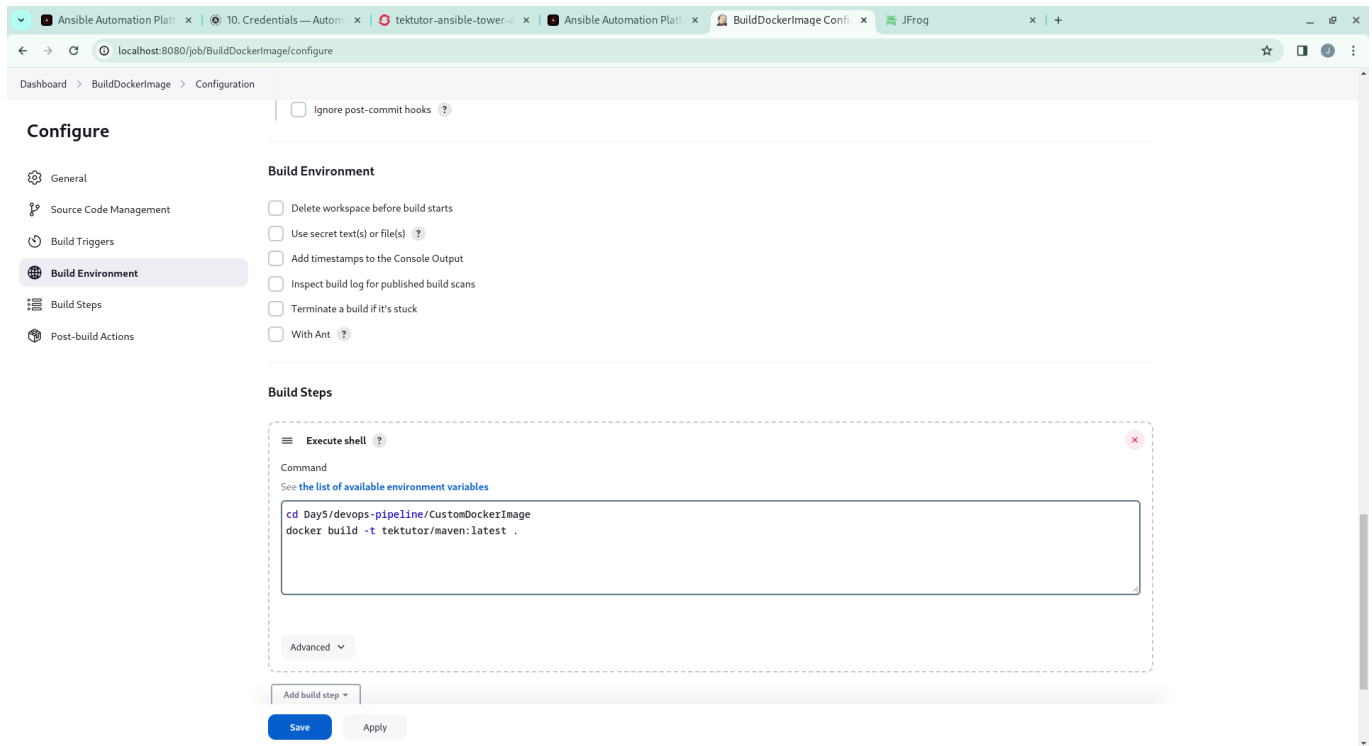
# Source Code Management



# Build Triggers



## Build Steps



The screenshot shows the Jenkins configuration page for a job named 'BuildDockerImage'. The left sidebar contains a 'Configure' menu with options: General, Source Code Management, Build Triggers, Build Environment (selected), Build Steps, and Post-build Actions. The main content area is titled 'Configure' and includes a checkbox for 'Ignore post-commit hooks'. Under the 'Build Environment' section, there are several checkboxes: 'Delete workspace before build starts', 'Use secret text(s) or file(s)', 'Add timestamps to the Console Output', 'Inspect build log for published build scans', 'Terminate a build if it's stuck', and 'With Ant'. The 'Build Steps' section shows a single step named 'Execute shell' with a command box containing the following text:

```
cd Day5/devops-pipeline/CustomDockerImage
docker build -t tekton/maven:latest .
```

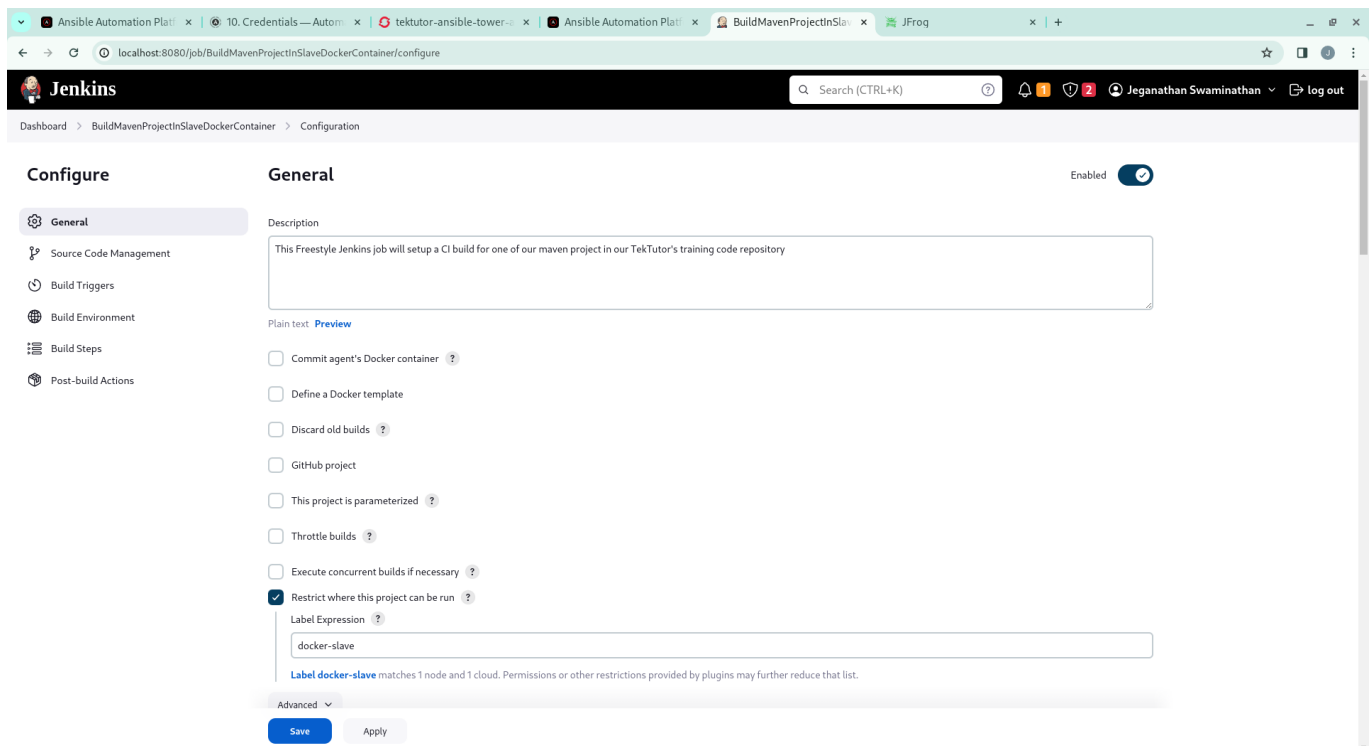
Below the command box is an 'Advanced' dropdown menu and an 'Add build step' button. At the bottom are 'Save' and 'Apply' buttons.

Save the above FreeStyle Job.

## Creating the BuildMavenJobInSlaveDockerContainer FreeStyle Job

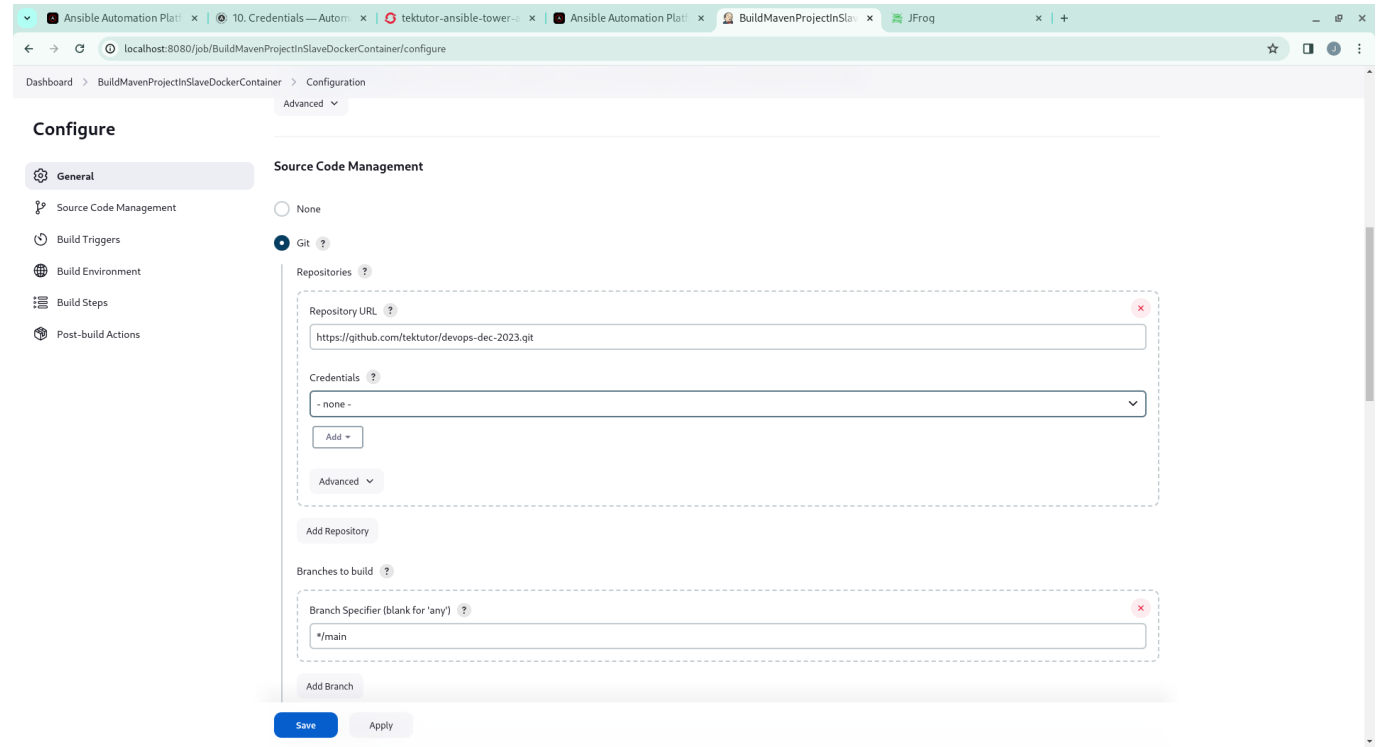
Let's create the Second FreeStyle Job as shown below

### General Section

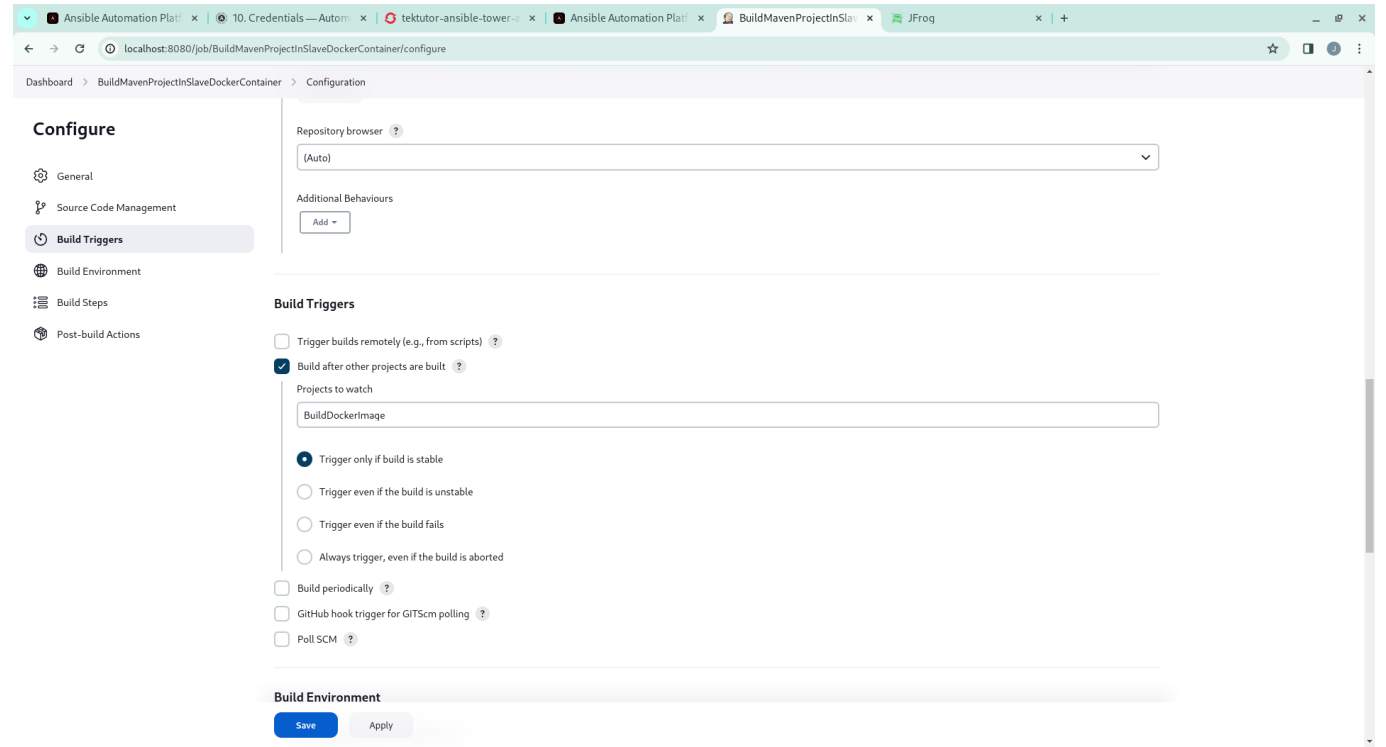


The screenshot shows the Jenkins configuration page for a job named 'BuildMavenProjectInSlaveDockerContainer'. The left sidebar contains a 'Configure' menu with options: General (selected), Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The main content area is titled 'General' and includes a toggle switch for 'Enabled'. The 'Description' field contains the text: 'This Freestyle Jenkins job will setup a CI build for one of our maven project in our TekTutor's training code repository'. Below the description is a 'Plain text' preview. The 'Restrict where this project can be run' checkbox is checked, and the 'Label Expression' field contains the text 'docker-slave'. Below the label expression field is a note: 'Label docker-slave matches 1 node and 1 cloud. Permissions or other restrictions provided by plugins may further reduce that list.' At the bottom are 'Save' and 'Apply' buttons.

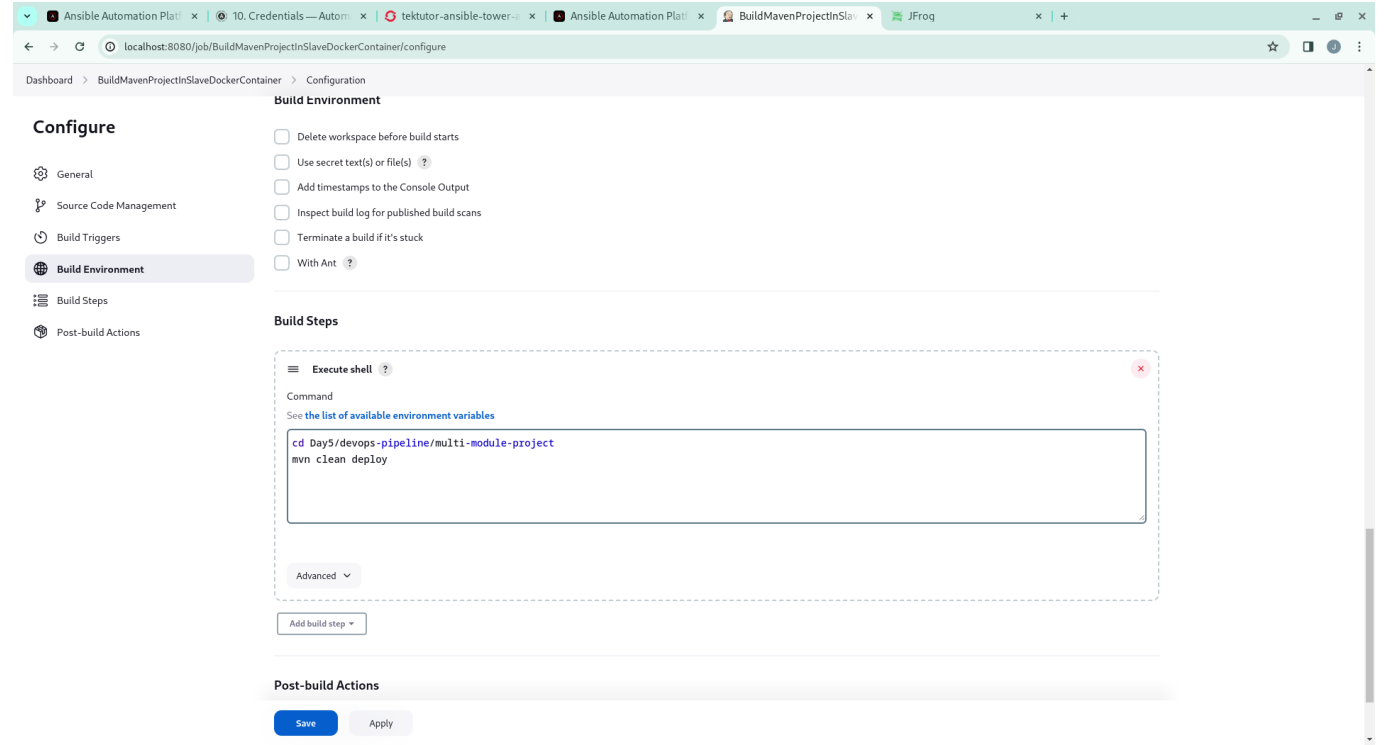
Source Code Management



Build Triggers



Build Steps

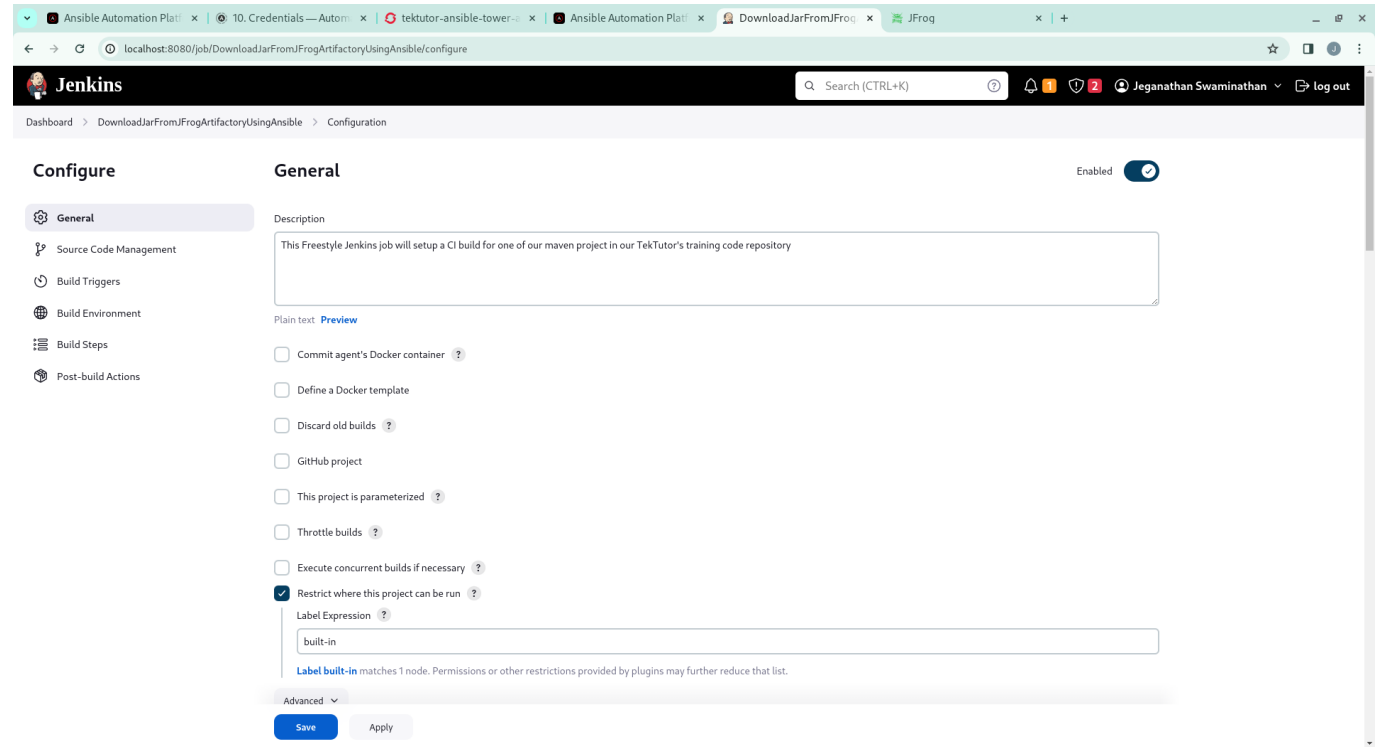


Save the FreeStyle Job

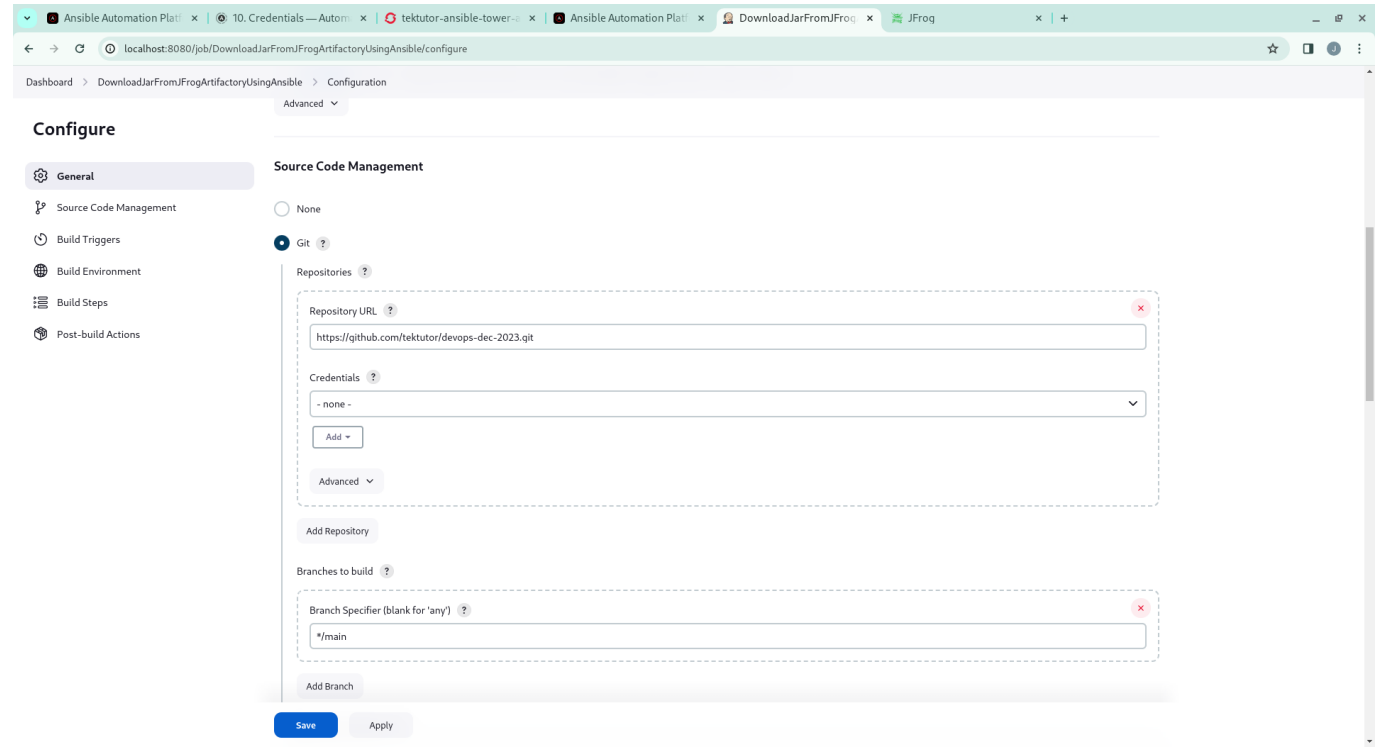
Creating the DownloadJarFromJFrogArtifactoryServerUsingAnsible FreeStyle Job

Let's create the First FreeStyle Job as shown below

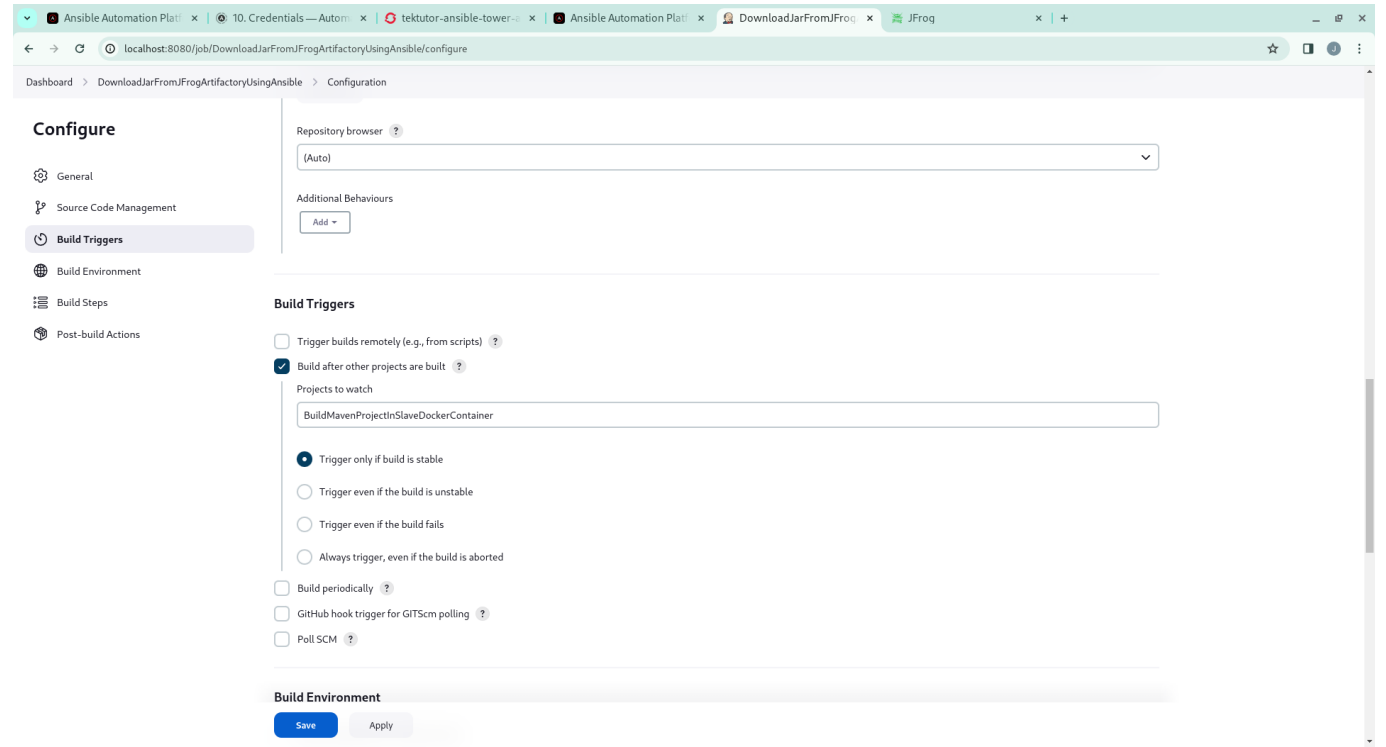
General Section



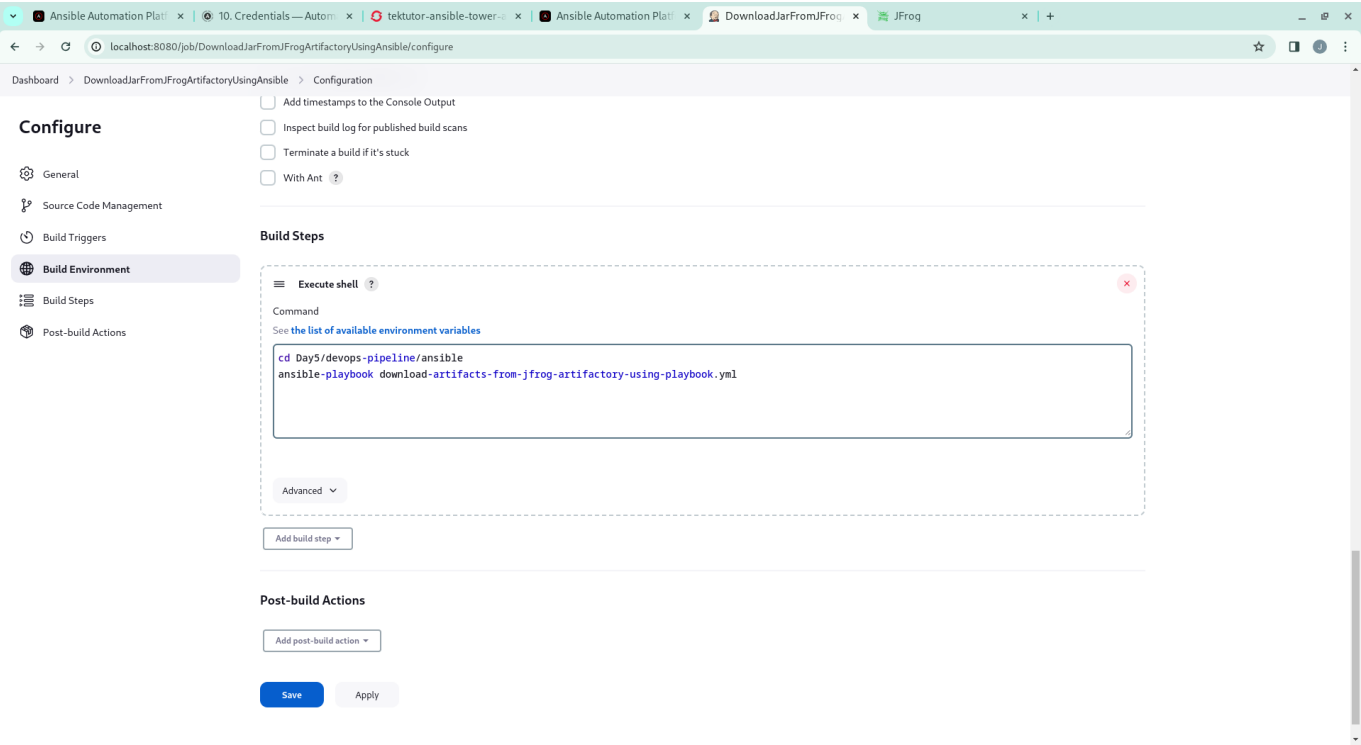
Source Code Management



Build Triggers



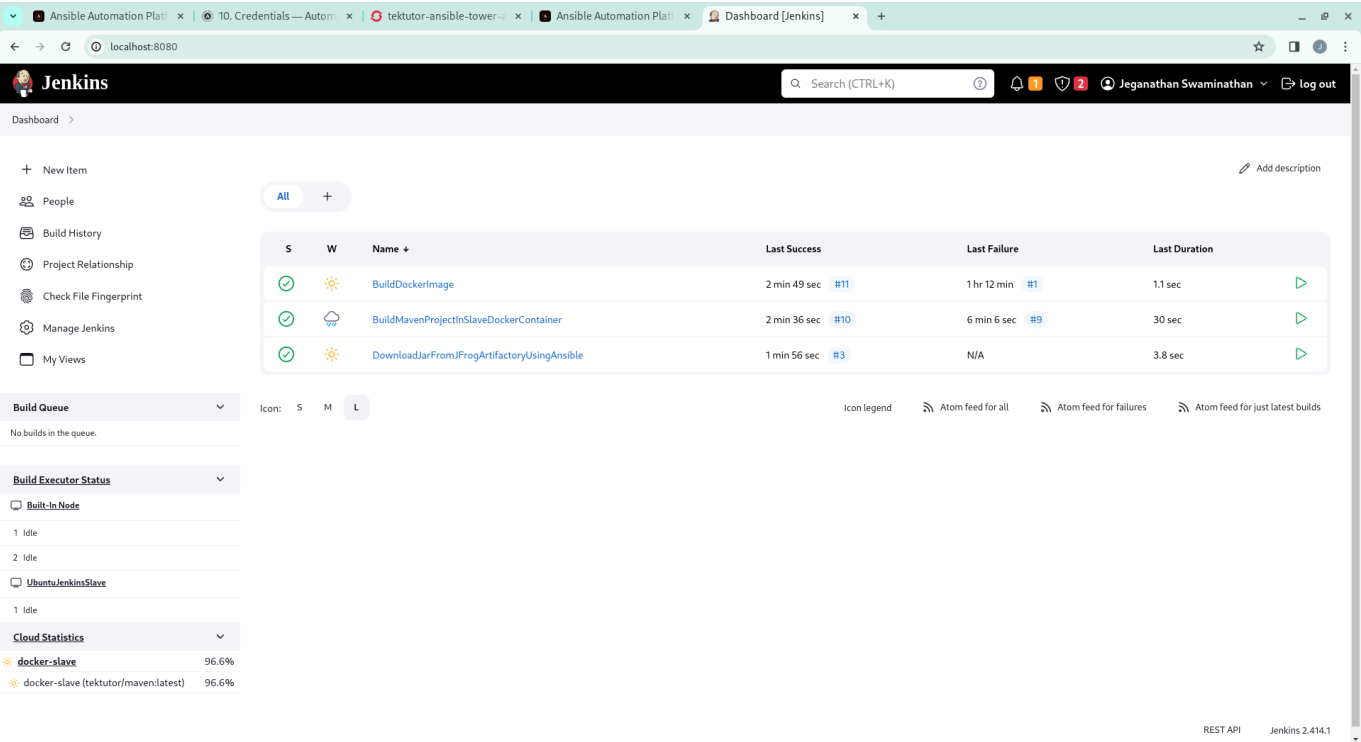
Build Steps



Save the above FreeStyle Job.

Lab - Creating the DevOps - Build Pipeline View

At this point, your Jenkins Dashboard will have all the 3 FreeStyle jobs we created today as shown below



Click on the "+" symbol and then click on "New view"

Ansible Automation Plat

10. Credentials — Autom

tektutor-ansible-tower

Ansible Automation Plat

Dashboard [Jenkins]

localhost:8080

Jeganathan Swaminathan

log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

Built-In Node

1 Idle

2 Idle

Ubuntu JenkinsSlave

1 Idle

Cloud Statistics

docker-slave

96.6%

docker-slave (tektutor/maven:latest)

96.6%

localhost:8080/newView

REST APIJenkins 2.414.1

Ansible Automation Plat

10. Credentials — Autom

tektutor-ansible-tower

Ansible Automation Plat

New view [Jenkins]

localhost:8080/newView

Jeganathan Swaminathan

log out

Dashboard

New view

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

Built-In Node

1 Idle

2 Idle

Ubuntu JenkinsSlave

1 Idle

Cloud Statistics

docker-slave

96.6%

docker-slave (tektutor/maven:latest)

96.6%

localhost:8080/newView

REST APIJenkins 2.414.1

New view

Name

DevOps Pipeline Demo

Type

Build Pipeline View

Shows the jobs in a build pipeline view. The complete pipeline of jobs that a version propagates through are shown as a row in the view.

List View

Shows items in a simple list format. You can choose which jobs are to be displayed in which view.

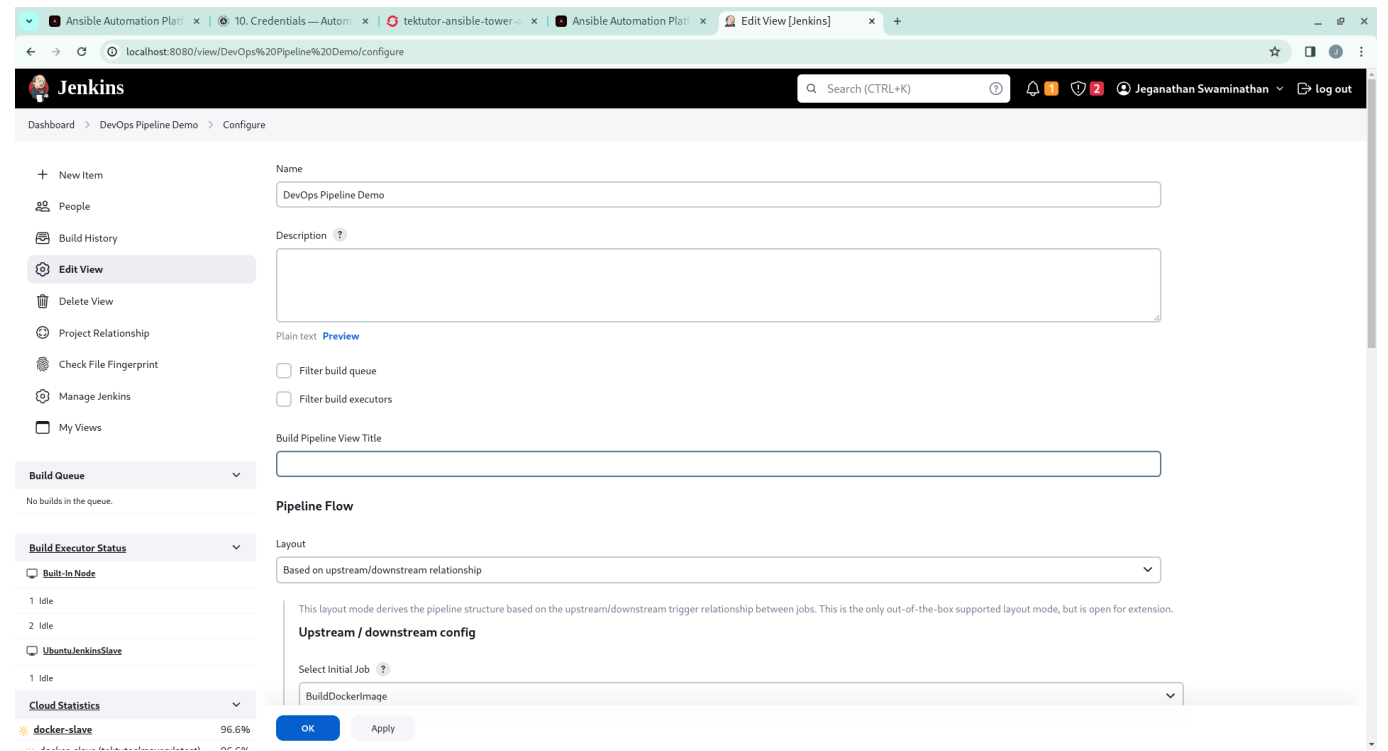
My View

This view automatically displays all the jobs that the current user has an access to.

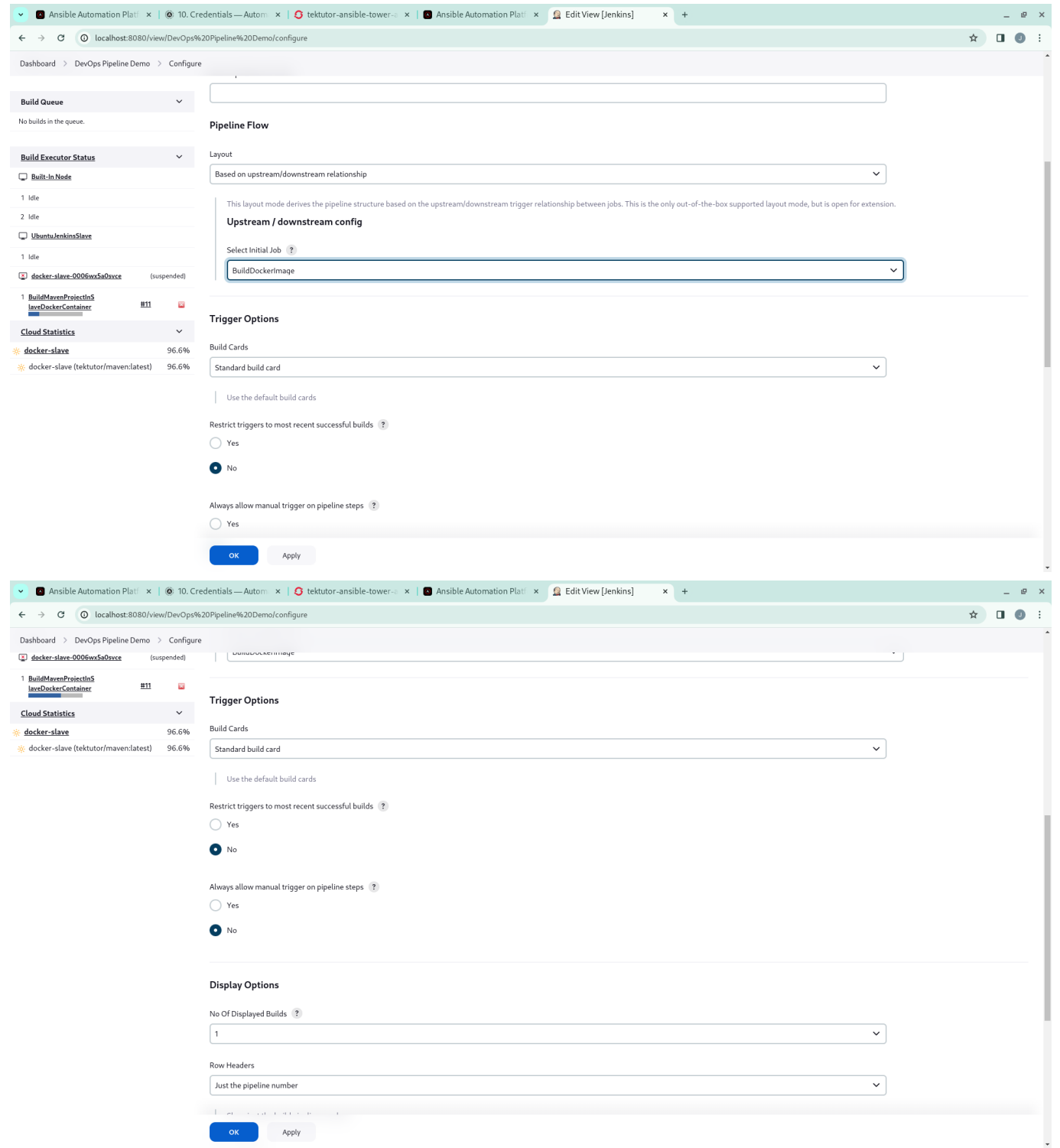
Create

Click on "Create" button





Under Pipeline Flow --> Upstream / downstream config --> Select Initial Job, you need to select "BuildDockerImage" as the first job in the pipeline.



Under Display Options --> No. of Displayed Buils, you may choose 10 or whatever number of build history your prefer.

Ansible Automation Plat...10. Credentials — Autom...tektutor-ansible-tower-...Ansible Automation Plat...Edit View [Jenkins]

localhost:8080/view/DevOps%20Pipeline%20Demo/configure

Dashboard > DevOps Pipeline Demo > Configure

Yes

No

Display Options

No Of Displayed Builds ?

1

Row Headers

Just the pipeline number

Show just the build pipeline number

Column Headers

No header

Do not show any column headers

Refresh frequency (in seconds) ?

3

URL for custom CSS files

Console Output Link Style

Lightbox

OK

Apply

Click on "Ok" to save the configurations.

Ansible Automation Plat...10. Credentials — Autom...tektutor-ansible-tower-...Ansible Automation Plat...DevOps Pipeline Demo

localhost:8080/view/DevOps%20Pipeline%20Demo/

Jenkins

Search (CTRL+K)

Jeganathan Swaminathan

log out

Dashboard > DevOps Pipeline Demo >

Build Pipeline

Run

History

Configure

Add Step

Delete

Manage

Pipeline

#12

#12 BuildDockerImage

Dec 12, 2023 12:34:10 PM

1 sec

#11 BuildMavenProjectInSlaveDockerContainer

Dec 12, 2023 12:34:24 PM

12 sec

#4 DownloadJarFromJfrogArtifactoryUsingAnsible

Dec 12, 2023 12:35:05 PM

3.5 sec

Your Dashboard would look like this now

Ansible Automation Plat

10. Credentials — Autom

tektutor-ansible-tower

Ansible Automation Plat

Dashboard [Jenkins]

localhost:8080

Search (CTRL+K)

1

2

Jeganathan Swaminathan

log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue (1)

BuildMavenProjectInSlaveDockerContainer

Build Executor Status

Built-In Node

1 Idle

2 Idle

UbuntuJenkinsSlave

1 Idle

Cloud Statistics

docker-slave

97%

docker-slave (tektutor/maven:latest)

97%

localhost:8080/view/DevOps Pipeline Demo/

REST API

Jenkins 2.414.1

DevOps Pipeline Demo

S	W	Name +	Last Success	Last Failure	Last Duration
✓	☀	BuildDockerImage	7.8 sec #13	1 hr 18 min #1	1.1 sec
✓	☁	BuildMavenProjectInSlaveDockerContainer	1 min 53 sec #11	11 min #9	32 sec
✓	☀	DownloadJarFromJfrogArtifactoryUsingAnsible	1 min 12 sec #4	N/A	3.5 sec

Icon: S M L

Icon legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

Lab - Creating a DevOps Pipeline in declarative style using Jenkinsfile(groovy)

Head over to Jenkins Dashboard

Ansible Automation Plat

10. Credentials — Autom

Services - Red Hat Open

Ansible Automation Plat

tektutor/spring-ms

Dashboard [Jenkins]

localhost:8080

Search (CTRL+K)

1

2

Jeganathan Swaminathan

log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

Built-In Node

1 Idle

2 Idle

UbuntuJenkinsSlave

1 Idle

Cloud Statistics

docker-slave

99.3%

docker-slave (tektutor/maven:latest)

99.3%

localhost:8080/view/DevOps Pipeline Demo/

REST API

Jenkins 2.414.1

DevOps Pipeline Demo

S	W	Name +	Last Success	Last Failure	Last Duration
✓	☀	BuildDockerImage	13 min #18	3 hr 27 min #1	1 sec
✓	☀	BuildMavenProjectInSlaveDockerContainer	13 min #17	2 hr 21 min #9	40 sec
✓	☁	DeclarativeDevOpsPipeline	13 min #3	29 min #1	3 min 14 sec
✓	☀	DeployAppsIntoOpenShift	13 min #4	N/A	2.4 sec
✓	☀	DownloadJarFromJfrogArtifactoryUsingAnsible	12 min #10	N/A	3.9 sec

Icon: S M L

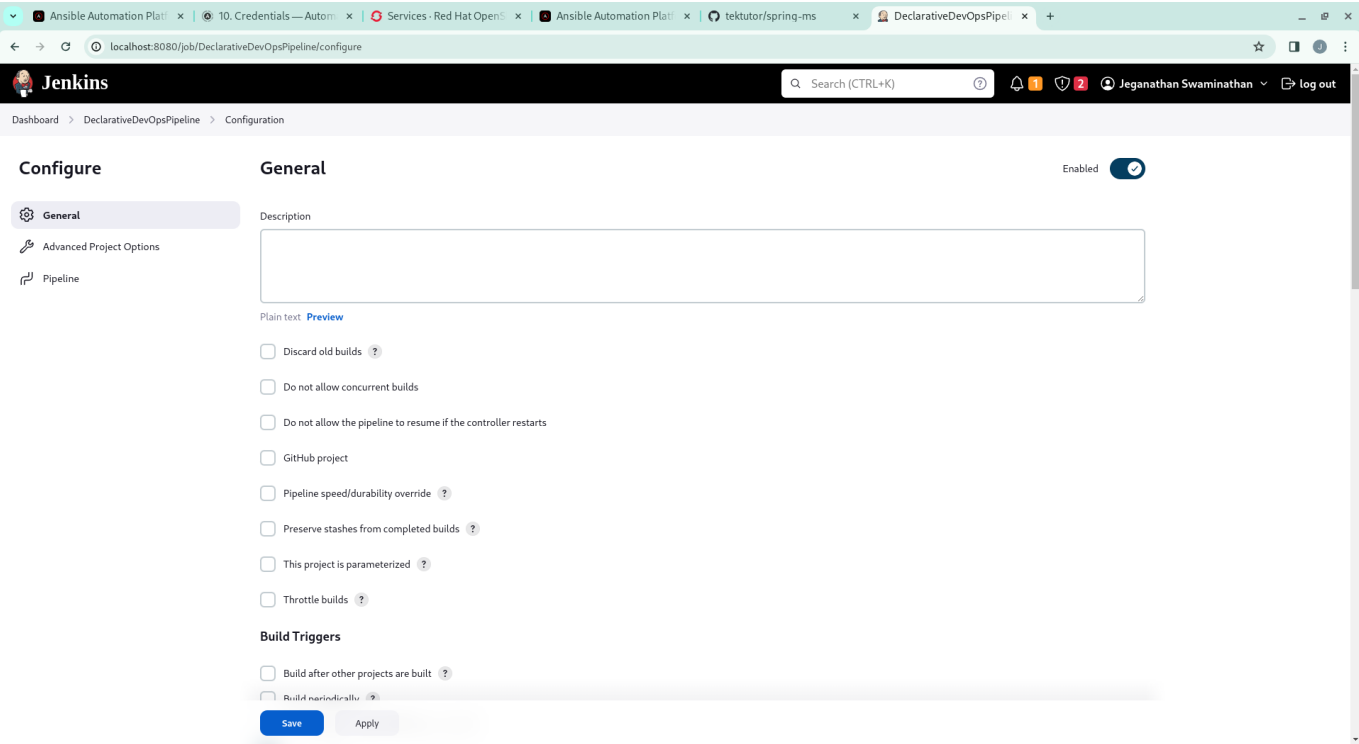
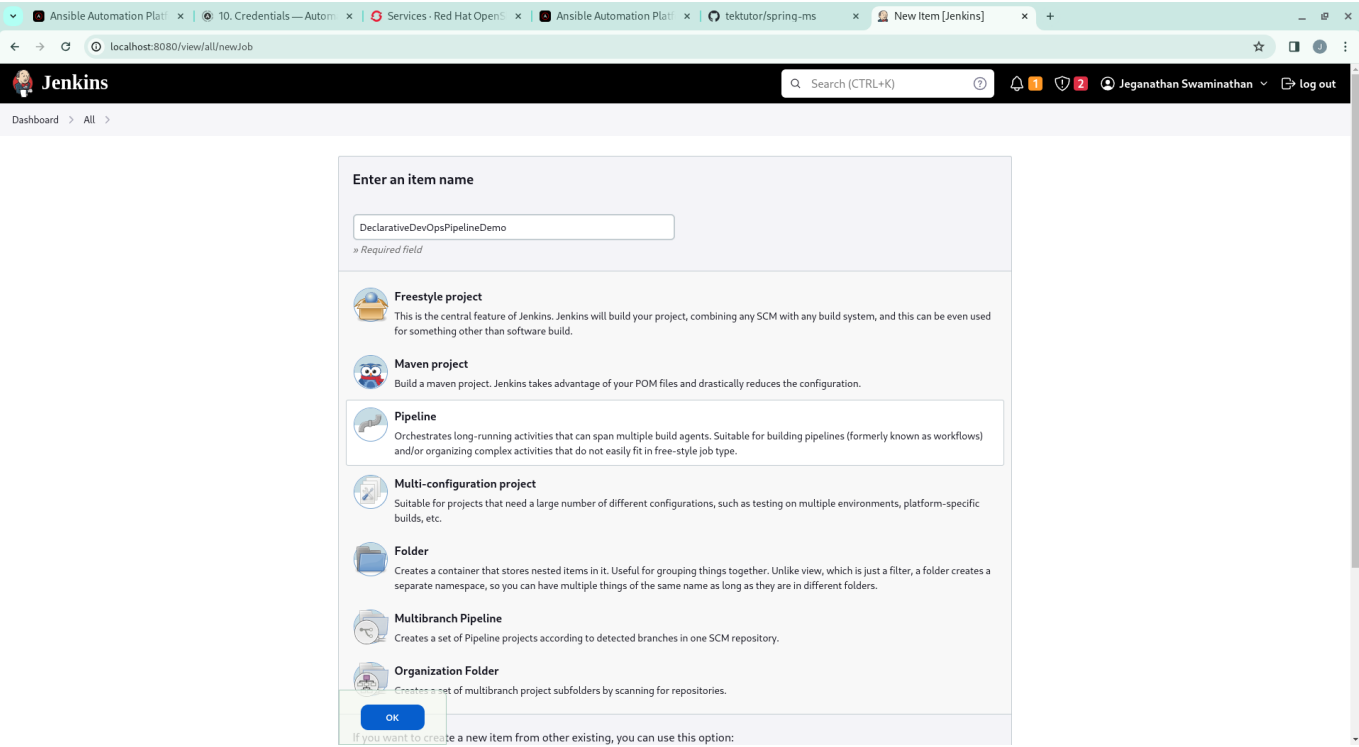
Icon legend

Atom feed for all

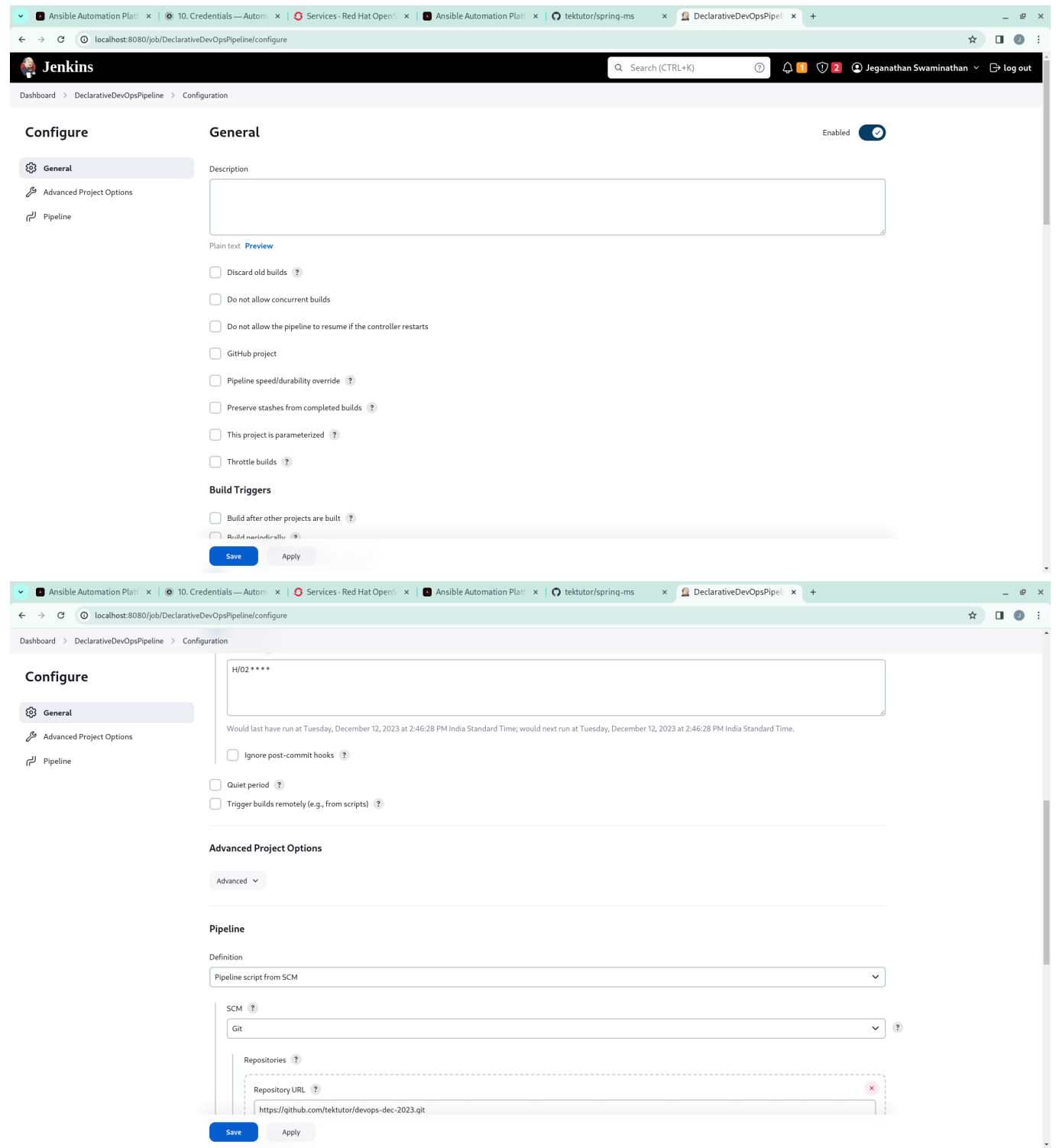
Atom feed for failures

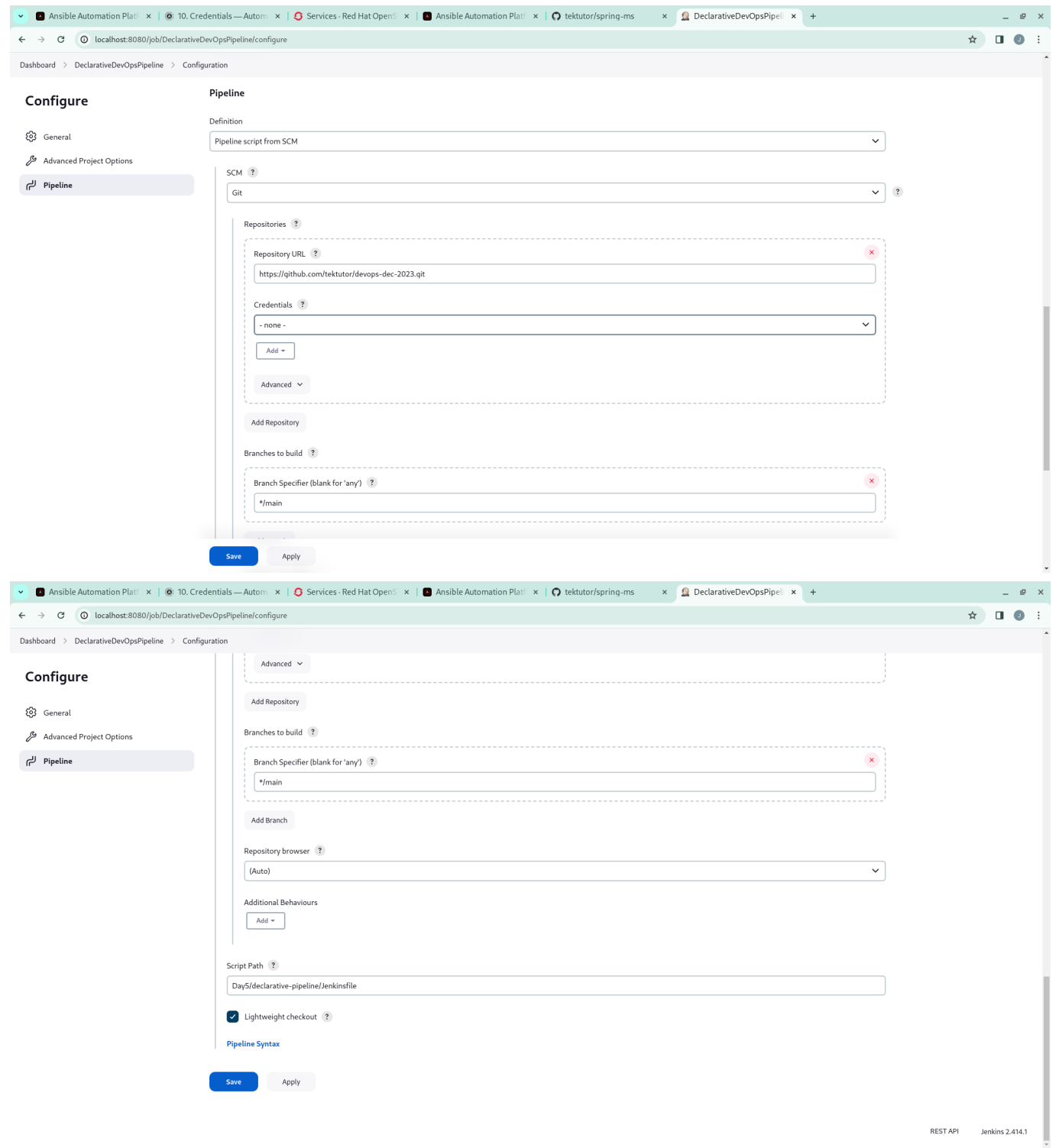
Atom feed for just latest builds

Click on "New Item" and Select "Pipeline"



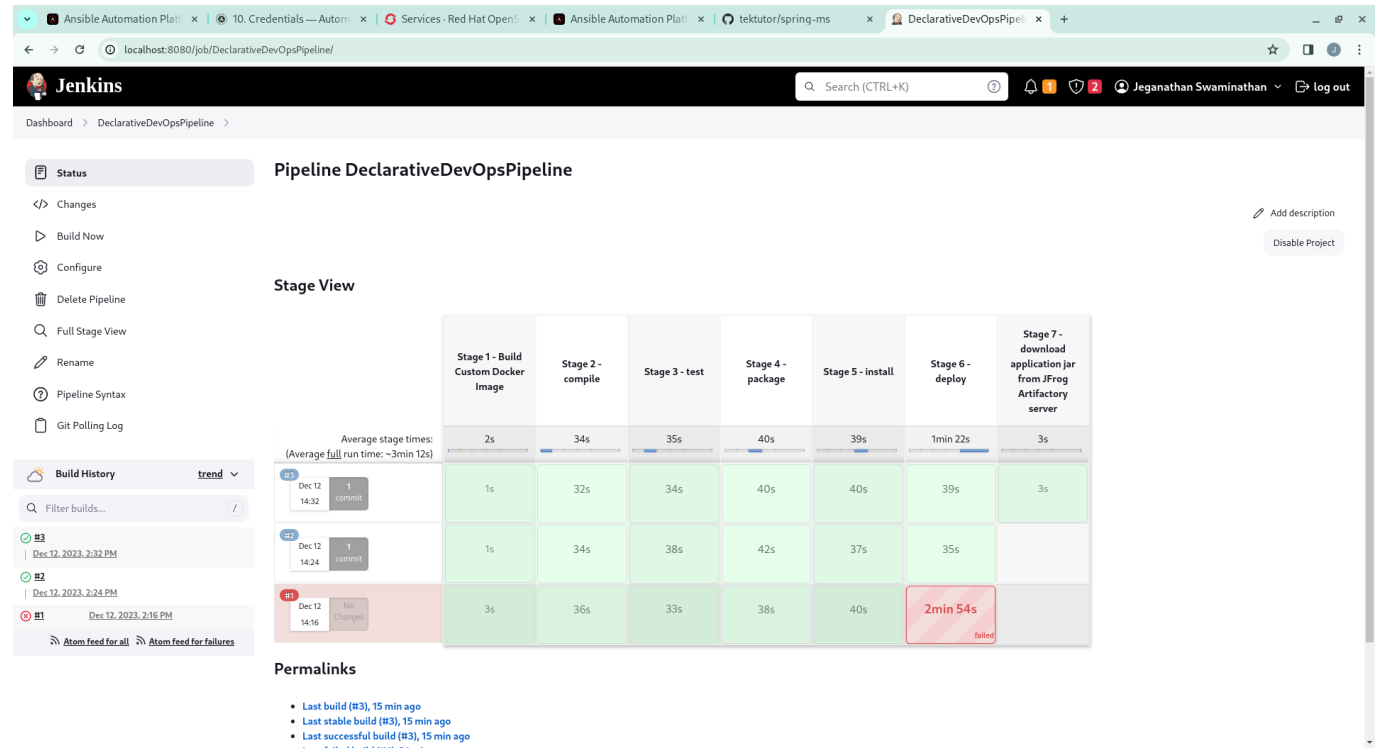
Build Triggers





Make sure you saved the configurations.

If everything went well, it will automatically trigger the pipeline and your would get an output similar to screenshot shown below



## Lab - Setting up CI for mysql db changes using Datical liquibase

Datical tool to track db schema changes in declarative style

Let's create a mysql db container

```
docker run -d --name mysql --hostname mysql -p 3306:3306 -e
MYSQL_ROOT_PASSWORD=root@123 mysql:latest
```

Expected output

```
jegan@tektutor.org: docker run -d --name mysql --hostname mysql -p
3306:3306 -e MYSQL_ROOT_PASSWORD=root@123 mysql:latest
2d1d256a3e00104d7410d8f13ff09c8261307fdda6bf3961d18dc64a9e3565b6
```

You may now check if the mysql db container is running

```
docker ps
```

Expected output

```
jegan@tektutor.org: docker ps
CONTAINER ID    IMAGE                COMMAND              CREATED
STATUS          PORTS               NAMES
```



```
2d1d256a3e00  mysql:latest  "docker-entrypoint.s..."  2 minutes ago  Up 2
minutes  0.0.0.0:3306->3306/tcp, :::3306->3306/tcp, 33060/tcp  mysql
```

Let us get inside the mysql db container, when prompts for password type 'root@123' without quotes

```
docker exec -it mysql bash
mysql -u root -p
CREATE DATABASE tektutor;
SHOW DATABASE;
```

Expected output

```
jegan@tektutor.org: docker exec -it mysql bash
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> CREATE DATABASE tektutor;
Query OK, 1 row affected (0.01 sec)

mysql> SHOW DATABASES;
+-----+
| Database                |
+-----+
| information_schema       |
| mysql                    |
| performance_schema       |
| sys                      |
| tektutor                  |
+-----+
5 rows in set (0.00 sec)

mysql> exit
Bye
bash-4.4# exit
exit
```

Let's check the datical liquibase now

```
cd ~/devops-dec-2023
git pull

cd Day5/datical/db-ci
cat liquibase.properties
```

The liquibase.properties file has the mysql connection details and it looks as shown below

```
jegan@tektutor.org:$ cat liquibase.properties
changeLogFile: dbchangelog.xml
url: jdbc:mysql://localhost:3306/tektutor
username: root
password: root@123
```

Any schema changes that we wish to perform, we need to do only via the dbchangelog.xml file. For instance, to create a training table with 3 columns, we may create a dbchangelog.xml as shown below

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xmlns:pro="http://www.liquibase.org/xml/ns/pro"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
    http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-latest.xsd
    http://www.liquibase.org/xml/ns/dbchangelog-ext
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd
    http://www.liquibase.org/xml/ns/pro
http://www.liquibase.org/xml/ns/pro/liquibase-pro-latest.xsd">

  <changeSet id="1" author="Jeganathan Swaminathan">
    <createTable tableName="training">
      <column name="id" type="int">
        <constraints primaryKey="true" nullable="false"/>
      </column>
      <column name="name" type="varchar(200)">
        <constraints nullable="false"/>
      </column>
      <column name="duration" type="varchar(200)"/>
    </createTable>
  </changeSet>
</databaseChangeLog>
```

In order to apply the table schema changes, you may run the below command

```
cd ~/devops-dec-2023/Day5/datical/db-ci
mvn liquibase:update
```

Expected output

```
jegan@tektutor.org:~/devops-dec-2023/Day5/datical/db-ci$ mvn
liquibase:update
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.tektutor:tektutor-java-app >-----
----
[INFO] Building tektutor-java-app 1.0
[INFO] -----[ jar ]-----
----
[INFO]
[INFO] --- liquibase-maven-plugin:4.22.0:update (default-cli) @ tektutor-
java-app ---
[INFO] -----
----
[INFO] there are no resolved artifacts for the Maven project.
[INFO] there are no resolved artifacts for the Maven project.
[INFO] Parsing Liquibase Properties File
[INFO]   File: liquibase.properties
[INFO] -----
----
[INFO] #####
##                                     ##
##      _      _                    _ _      ##
##      | |      ( _ )              ( _ ) |      ##
##      | |      _ _ _ _ _ _ _ _ _ _ | | _ _ _ _ _ _      ##
##      | |      | | / _ \ | | | | | | ' _ \ / _ \ / _ \ / _ \      ##
##      | | _ _ | | ( _ | | | | | | | _ ) | ( _ \ _ \ _ \      ##
##      \ _ _ / _ | \ _ , | \ _ , | | | _ . _ / \ _ , _ | _ \ _ \      ##
##                                     ##
##                                     ##
##      | _ |                                     ##
##                                     ##
##      Get documentation at docs.liquibase.com      ##
##      Get certified courses at learn.liquibase.com  ##
##                                     ##
#####
Starting Liquibase at 15:53:32 (version 4.22.0 #9559 built at 2023-05-10
20:45+0000)
[INFO] Parsing Liquibase Properties File liquibase.properties for changeLog
parameters
[INFO] Executing on Database: jdbc:mysql://localhost:3306/tektutor
[INFO] Successfully acquired change log lock
[INFO] Creating database history table with name: DATABASECHANGELOG
[INFO] Reading from DATABASECHANGELOG
[INFO] Using deploymentId: 6306214275
[INFO] Reading from DATABASECHANGELOG
Running Changeset: dbchangelog.xml::1::Jeganathan Swaminathan
```

```

[INFO] Table training created
[INFO] ChangeSet dbchangelog.xml::1::Jeganathan Swaminathan ran
successfully in 25ms
[INFO] UPDATE SUMMARY
[INFO] Run:                                1
[INFO] Previously run:                     0
[INFO] Filtered out:                       0
[INFO] -----
[INFO] Total change sets:                   1

UPDATE SUMMARY
Run:                                1
Previously run:                     0
Filtered out:                       0
-----
Total change sets:                   1

[INFO] Update summary generated
[INFO] Update command completed successfully.
Liquibase: Update has been successful.
[INFO] Successfully released change log lock
[INFO] Successfully released change log lock
[INFO] Command execution complete
[INFO] -----
----
[INFO]
[INFO] -----
----
[INFO] BUILD SUCCESS
[INFO] -----
----
[INFO] Total time:  2.046 s
[INFO] Finished at: 2023-06-09T15:53:34+05:30
[INFO] -----
----

```

The liquibase will grab the mysql connection details from the liquibase.properties file and connects to our tektutor database with mysql server and applies the changeset defined in the dbchangelog.xml file.

The pom.xml file points to the liquibase.properites and the liquibase.properites file points to dbchangelog.xml file. This is how, liquibase learns about these files.

You may now verify, if the changes are done in your mysql server

```

jegan@tektutor:~/devops-dec-2023/Day5/datical/db-ci$ docker exec -it mysql
bash
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10

```

Server version: 8.0.33 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> USE tektutor;
```

Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A

Database changed

```
mysql> SHOW TABLES;
```

```
+-----+
| Tables_in_tektutor |
+-----+
| DATABASECHANGELOG |
| DATABASECHANGELOGLOCK |
| training           |
+-----+
3 rows in set (0.01 sec)
```

```
mysql> DESCRIBE training;
```

```
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int           | NO   | PRI | NULL    |       |
| name  | varchar(200)  | NO   |     | NULL    |       |
| duration | varchar(200) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
mysql> exit
```

Bye

```
bash-4.4# exit
```

```
exit
```

Assuming, you wish to add 2 more columns to the training table on tektutor database. We can update the dbchangelog.xml file as shown below

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xmlns:pro="http://www.liquibase.org/xml/ns/pro"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
```

```

http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-latest.xsd
http://www.liquibase.org/xml/ns/dbchangelog-ext
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-ext.xsd
http://www.liquibase.org/xml/ns/pro
http://www.liquibase.org/xml/ns/pro/liquibase-pro-latest.xsd">

<changeSet id="1" author="Jeganathan Swaminathan">
  <createTable tableName="training">
    <column name="id" type="int">
      <constraints primaryKey="true" nullable="false"/>
    </column>
    <column name="name" type="varchar(200)">
      <constraints nullable="false"/>
    </column>
    <column name="duration" type="varchar(200)"/>
  </createTable>
</changeSet>
<changeSet id="2" author="Jeganathan Swaminathan">
  <addColumn tableName="training">
    <column name="from_date" type="varchar(200)"/>
    <column name="to_date" type="varchar(200)"/>
  </addColumn>
</changeSet>
</databaseChangeLog>

```

You may now update the liquibase as shown below

```

cd ~/devops-dec-2023/Day5/datical/db-ci
mvn liquibase:update

```

Expected output

```

jegan@tektutor:~/devops-dec-2023/Day5/datical/db-ci$ mvn liquibase:update
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.tektutor:tektutor-java-app >-----
----
[INFO] Building tektutor-java-app 1.0
[INFO] -----[ jar ]-----
----
[INFO]
[INFO] --- liquibase-maven-plugin:4.22.0:update (default-cli) @ tektutor-
java-app ---
[INFO] -----
----
[INFO] there are no resolved artifacts for the Maven project.
[INFO] there are no resolved artifacts for the Maven project.
[INFO] Parsing Liquibase Properties File
[INFO]   File: liquibase.properties

```

/

```
[INFO] -----
----
[INFO] BUILD SUCCESS
[INFO] -----
----
[INFO] Total time: 1.912 s
[INFO] Finished at: 2023-06-09T15:55:47+05:30
[INFO] -----
----
```

You may now verify if the schema changes are applied

```
jegan@tektutor:~/devops-dec-2023/Day5/datical/db-ci$ docker exec -it mysql
bash
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.33 MySQL Community Server - GPL
```

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> USE tektutor;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

Database changed

```
mysql> SHOW TABLES;
```

```
+-----+
| Tables_in_tektutor |
+-----+
| DATABASECHANGELOG |
| DATABASECHANGELOGLOCK |
| training           |
+-----+
3 rows in set (0.01 sec)
```

```
mysql> DESCRIBE training;
```

```
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id         | int           | NO   | PRI | NULL    |       |
| name       | varchar(200)  | NO   |     | NULL    |       |
| duration   | varchar(200)  | YES  |     | NULL    |       |
| from_date  | varchar(200)  | YES  |     | NULL    |       |
```



```
| to_date | varchar(200) | YES | | NULL | | |  
+-----+-----+-----+-----+-----+-----+  
5 rows in set (0.00 sec)  
  
mysql> exit  
Bye  
bash-4.4# exit  
exit
```

For official documentation about liquibase, you may check here <https://docs.liquibase.com/faq.html>

## Kindly share your training feedback here

---

<https://survey.zohopublic.com/zs/uB0uUy>

## Kindly complete your Post test from RPS Lab machine

---

<https://app.mymapit.in/code4/tiny/EFSOp7>