

# PROJECT -1

---

**BUILDING A CI/CD PIPELINE FOR A RETAIL COMPANY**

Akanksha Sukre

Date: 16/02/2023

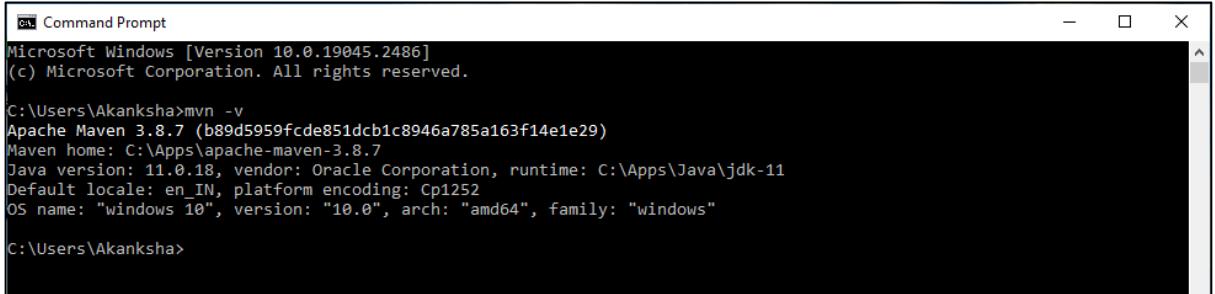
## Contents

Task 1: Build the code using Maven commands .....	3
1) Install Maven and build code .....	3
Task 2: Set up a GIT repository and push the source code .....	4
1) GitHub Repo Configuration.....	4
2) Jenkins Configuration on Master Node .....	6
3) Create Jobs in Jenkins .....	8
4) Create build pipeline with above three jobs.....	14
5) Set up master-slave nodes to distribute tasks in the pipeline.....	17
6) GITScm Polling for CI.....	20
Task 3: Create CI/CD pipeline in Jenkins to build and deploy on Docker container .....	21
1) Integrate Docker with Jenkins.....	21
2) CI/CD job on Jenkins to build and deploy on Docker container .....	22
Task 4: Integrate Docker and Kubernetes with Ansible and create CI/CD pipeline in Jenkins to deploy the application on Kubernetes.....	26
1) Integrate Ansible with Jenkins .....	26
2) Create CI/CD pipeline to create docker image and container using Ansible playbook .....	27
3) Deploy artifacts to Kubernetes .....	30
4) Integrate Ansible with Kubernetes. Create CI/CD Pipeline to deploy artifacts on Kubernetes using Ansible playbook. .....	32
5) Create end to end pipeline: Build and Publish docker image & Deploy to Kubernetes using Ansible.....	35
Task 5: Monitoring the Kubernetes Cluster using Prometheus and creating dashboard in Grafana to view important metrics.....	38
1) Install Node Exporter on master and slave node.....	38
2) Add Node URL to target in prometheus.yml .....	40
3) Create Dashboard in Grafana .....	40

## Task 1: Build the code using Maven commands

### 1) Install Maven and build code

#### i. Install maven and check version

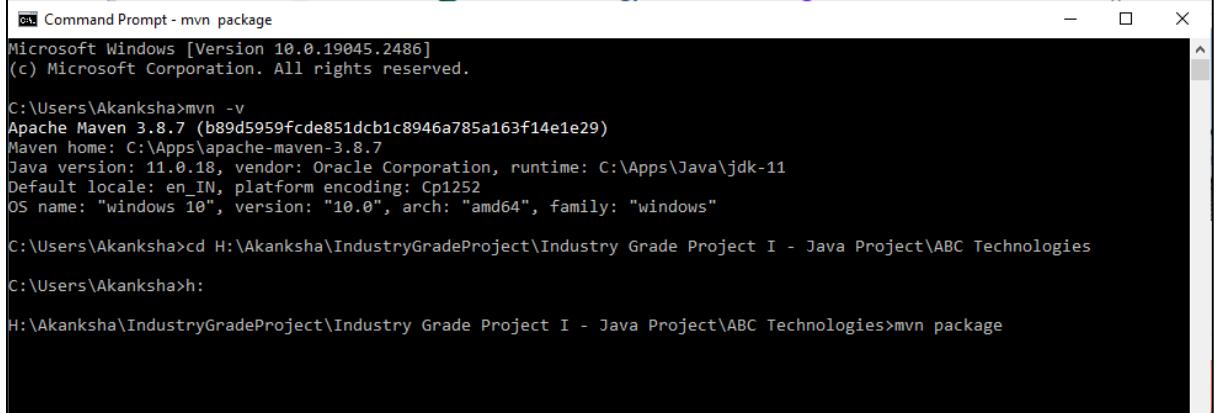


```
Microsoft Windows [Version 10.0.19045.2486]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Akanksha>mvn -v
Apache Maven 3.8.7 (b89d5959fcde851dcbb1c8946a785a163f14e1e29)
Maven home: C:\Apps\apache-maven-3.8.7
Java version: 11.0.18, vendor: Oracle Corporation, runtime: C:\Apps\Java\jdk-11
Default locale: en_IN, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\Akanksha>
```

#### ii. Run command: mvn package



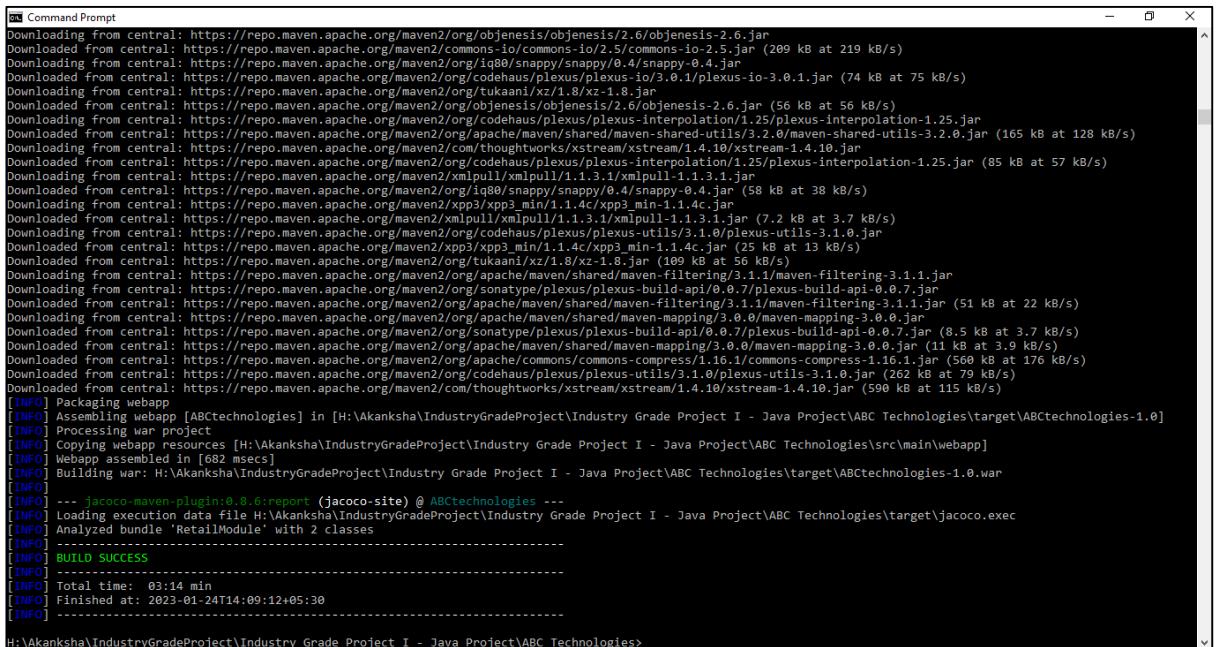
```
Microsoft Windows [Version 10.0.19045.2486]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Akanksha>mvn -v
Apache Maven 3.8.7 (b89d5959fcde851dcbb1c8946a785a163f14e1e29)
Maven home: C:\Apps\apache-maven-3.8.7
Java version: 11.0.18, vendor: Oracle Corporation, runtime: C:\Apps\Java\jdk-11
Default locale: en_IN, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\Akanksha>cd H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies

C:\Users\Akanksha>h:
H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies>mvn package
```

#### iii. Build successful and war file generated



```
Command Prompt
Downloaded from central: https://repo.maven.apache.org/maven2/org/objenesis/objenesis/2.6/objenesis-2.6.jar
Downloaded from central: https://repo.maven.apache.org/maven2/commons-io/commons-io/2.5/commons-io-2.5.jar (209 kB at 219 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.0.1/plexus-io-3.0.1.jar (74 kB at 75 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/tukaani/xz/1.8/xz-1.8.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/objenesis/objenesis/2.6/objenesis-2.6.jar (56 kB at 56 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interpolation/1.25/plexus-interpolation-1.25.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interpolation/1.4.10/xstream-1.4.10.jar (165 kB at 128 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.4.10/xstream-1.4.10.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interpolation/1.25/plexus-interpolation-1.25.jar (85 kB at 57 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/xmlpull/xmlpull/1.1.3.1/xmlpull-1.1.3.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/ig80/snappy/snappy/0.4/snappy-0.4.jar (58 kB at 38 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/xpp3/min/1.1.4c/xpp3_min-1.1.4c.jar
Downloaded from central: https://repo.maven.apache.org/maven2/xmlpull/xmlpull/1.1.3.1/xmlpull-1.1.3.1.jar (7.2 kB at 3.7 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.1.0/plexus-utils-3.1.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/xpp3/min/1.1.4c/xpp3_min-1.1.4c.jar (25 kB at 13 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/tukaani/xz/1.8/xz-1.8.jar (108 kB at 56 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/3.1.1/maven-filtering-3.1.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus-build-api/0.0.7/plexus-build-api-0.0.7.jar (51 kB at 22 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/3.1.1/maven-filtering-3.1.1.jar (8.5 kB at 3.7 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-build-api/0.0.7/plexus-build-api-0.0.7.jar (11 kB at 3.9 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-mapping/3.0.0/maven-mapping-3.0.0.jar (560 kB at 176 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.1.0/plexus-utils-3.1.0.jar (262 kB at 79 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.4.10/xstream-1.4.10.jar (590 kB at 115 kB/s)
[INFO] Packaging webapp
[INFO] Assembling webapp [ABCtechnologies] in [H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies\target\ABCtechnologies-1.0]
[INFO] Processing war project
[INFO] Copying webapp resources [H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies\src\main\webapp]
[INFO] Webapp assembled in [682 msec]
[INFO] Building war: H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies\target\ABCtechnologies-1.0.war
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.6:report (jacoco-site) @ ABCtechnologies ---
[INFO] Loading execution data file H:\Akanksha\IndustryGradeProject\Industry Grade Project I - Java Project\ABC Technologies\target\jacoco.exec
[INFO] Analyzed bundle 'RetailModule' with 2 classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 03:14 min
[INFO] Finished at: 2023-01-24T14:09:12+05:30
[INFO] -----
```



A screenshot of a Windows File Explorer window showing the directory structure of a Java project. The path is: Home (H:) > Akanksha > IndustryGradeProject > Industry Grade Project I - Java Project > ABC Technologies > target. The table lists files and folders with their names, last modified dates, types, and sizes.

Name	Date modified	Type	Size
ABCtechnologies-1.0	24-01-2023 14:09	File folder	
classes	24-01-2023 14:08	File folder	
generated-sources	24-01-2023 14:08	File folder	
generated-test-sources	24-01-2023 14:08	File folder	
maven-archiver	24-01-2023 14:09	File folder	
maven-status	24-01-2023 14:08	File folder	
site	24-01-2023 14:09	File folder	
surefire-reports	24-01-2023 14:08	File folder	
test-classes	24-01-2023 14:08	File folder	
ABCtechnologies-1.0.war	24-01-2023 14:09	WAR File	6,966 KB
jacoco.exec	24-01-2023 14:08	EXEC File	5 KB

## Task 2: Set up a GIT repository and push the source code

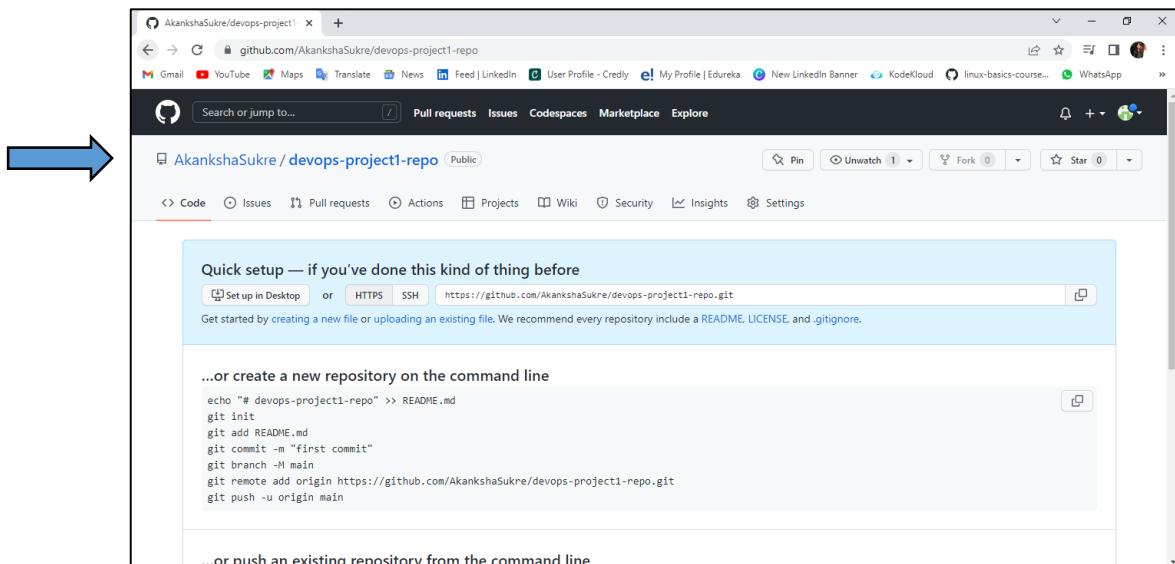
### 1) GitHub Repo Configuration

#### i. Create a new repository on GitHub



A screenshot of a web browser showing the GitHub 'Create a new repository' page. The URL is 'github.com/new'. The form fields include 'Owner' (set to 'AkankshaSukre'), 'Repository name' ('devops-project1-repo'), 'Description (optional)' ('This is a repository for DevOps Project'), and 'Visibility' (radio button selected for 'Public'). A blue arrow points to the 'Owner' dropdown.

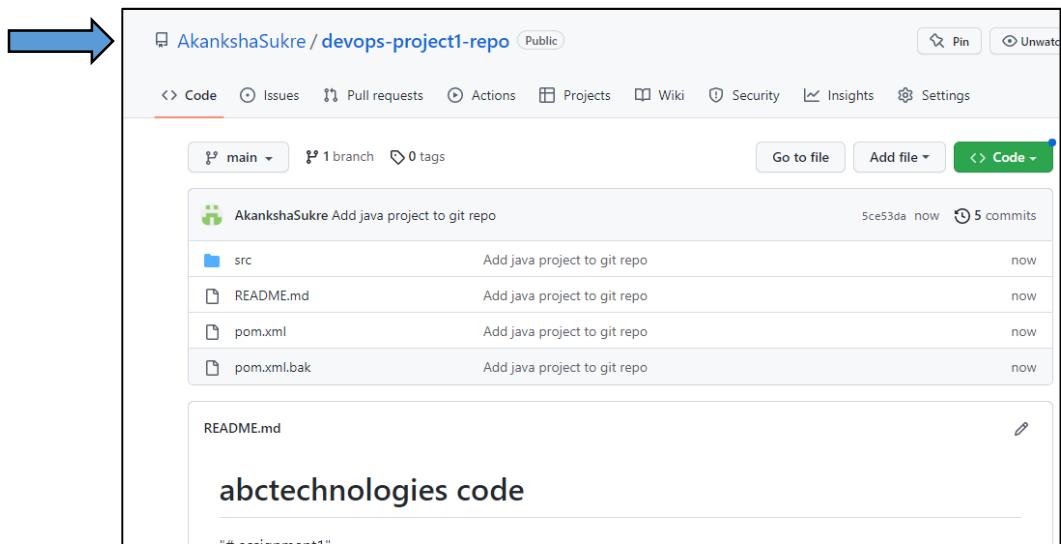
ii. New GIT repo created: **devops-project1-repo**



iii. Add project files to newly created repo and commit changes

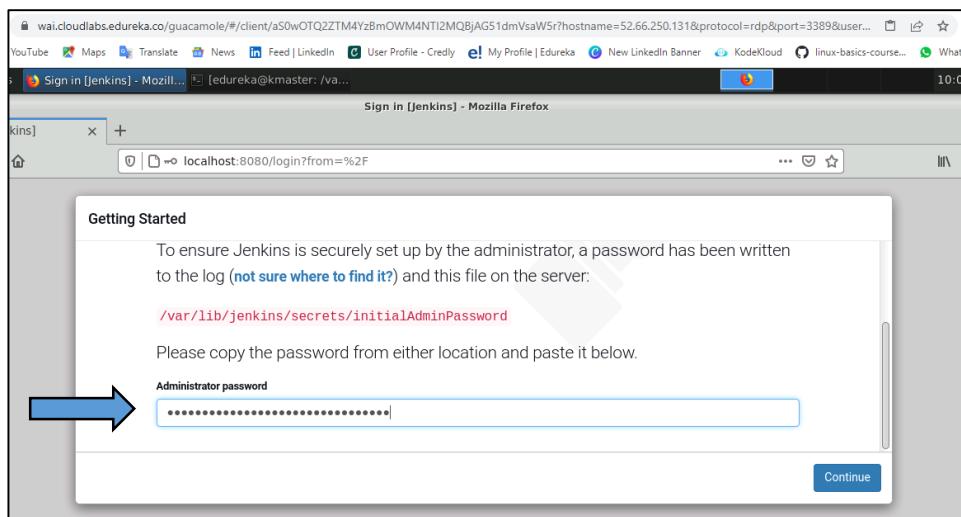
The image contains two screenshots of a GitHub repository interface. The top screenshot shows the 'Code' tab of a repository named 'devops-project1-repo'. It features a large central area for dragging files and a list of three Java files: 'RetailAccessObject.java', 'RetailDataImp.java', and 'RetailModule.java'. A blue arrow points to the 'choose your files' link. The bottom screenshot shows a 'Commit changes' dialog box. It includes fields for a commit message ('Add java project to git repo'), an optional description, and two radio button options: 'Commit directly to the main branch.' (selected) and 'Create a new branch for this commit and start a pull request.' A blue arrow points to the 'Commit changes' button at the bottom of the dialog.

iv. Project successfully pushed to GIT repo

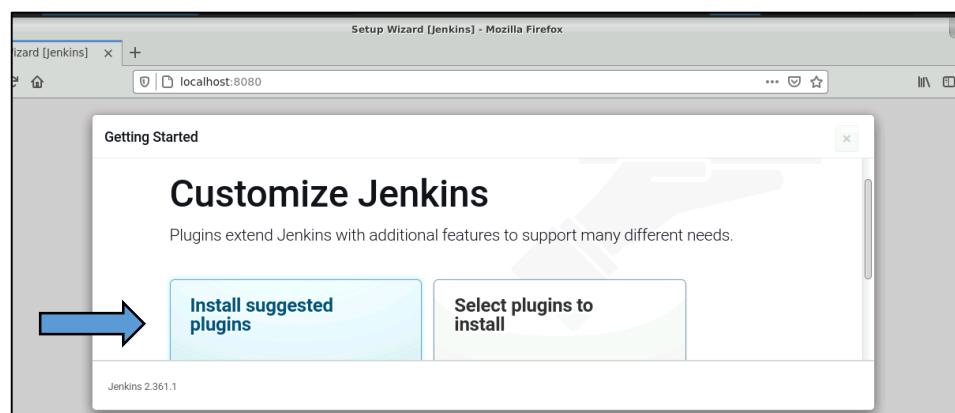


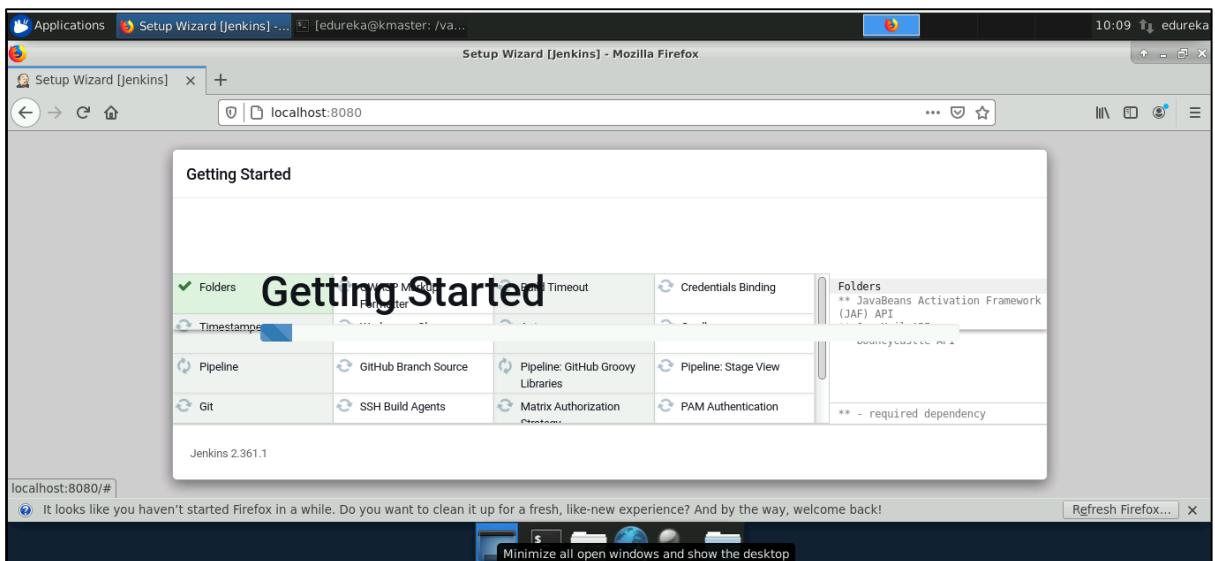
## 2) Jenkins Configuration on Master Node

i. Set up Jenkins on Master Node

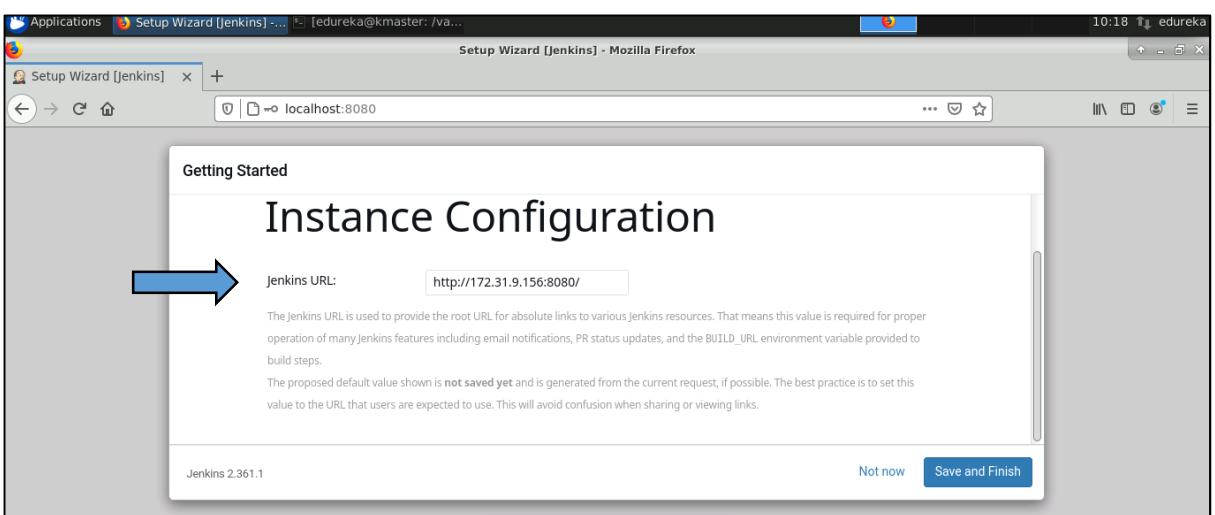


ii. Install required plugins in Jenkins

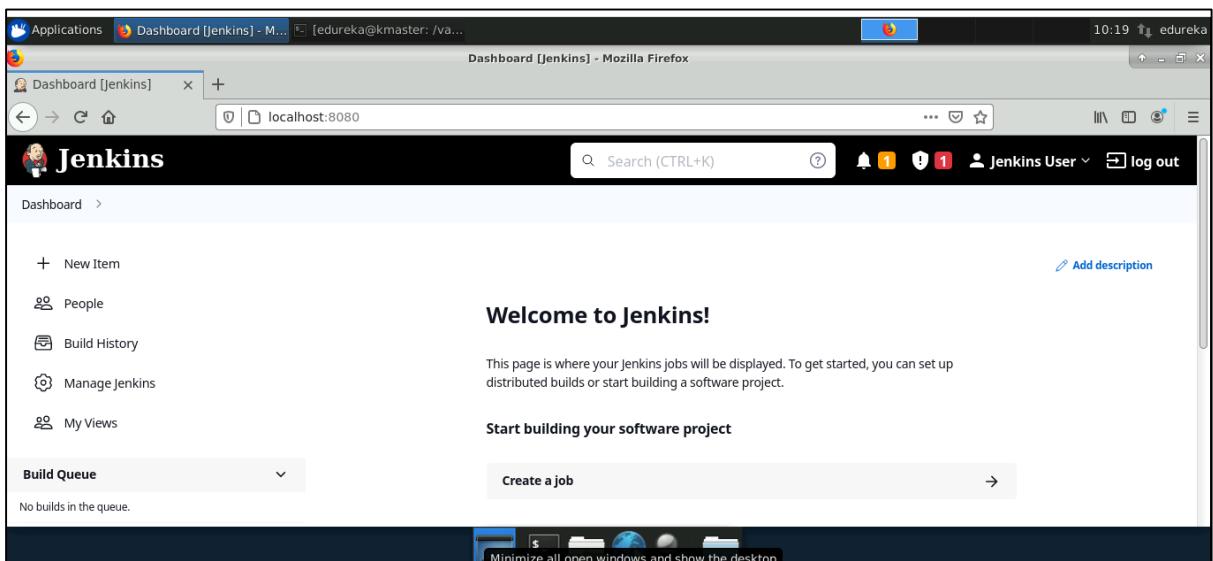




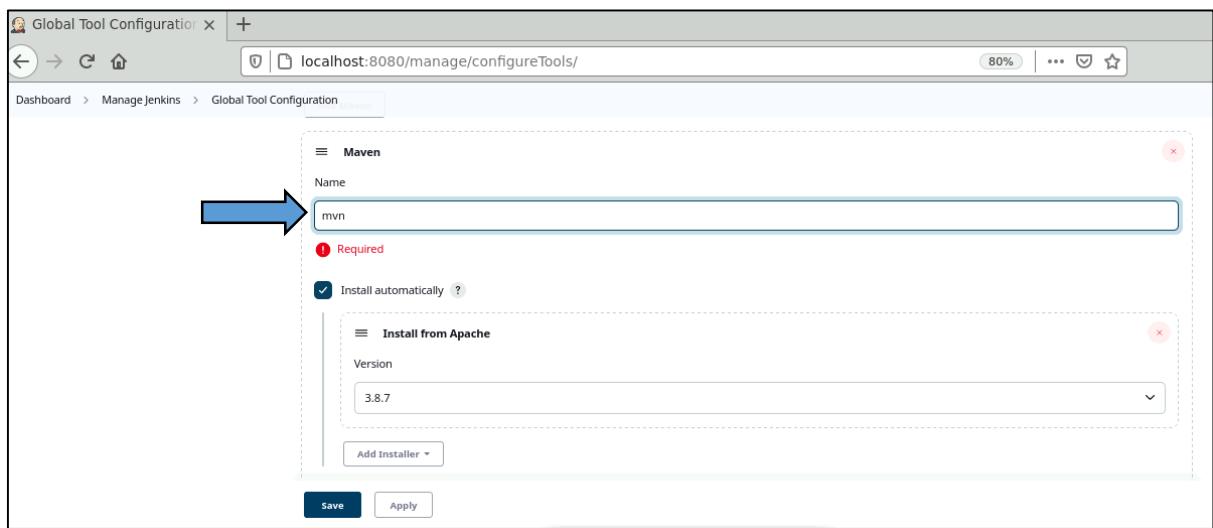
iii. Set up Jenkins URL: <http://172.31.9.156:8080>



iv. Jenkins is ready to use

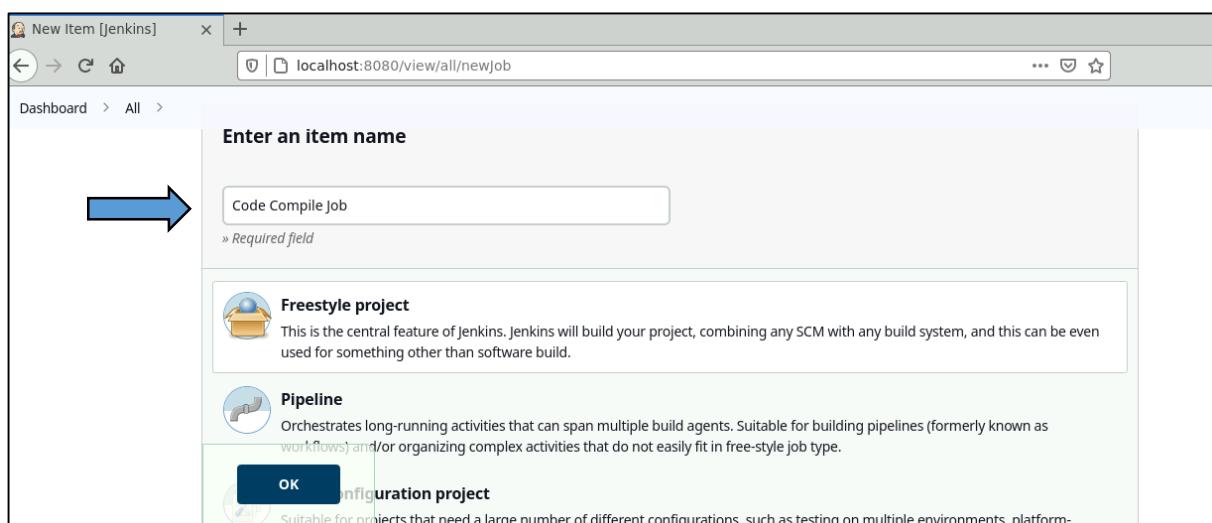
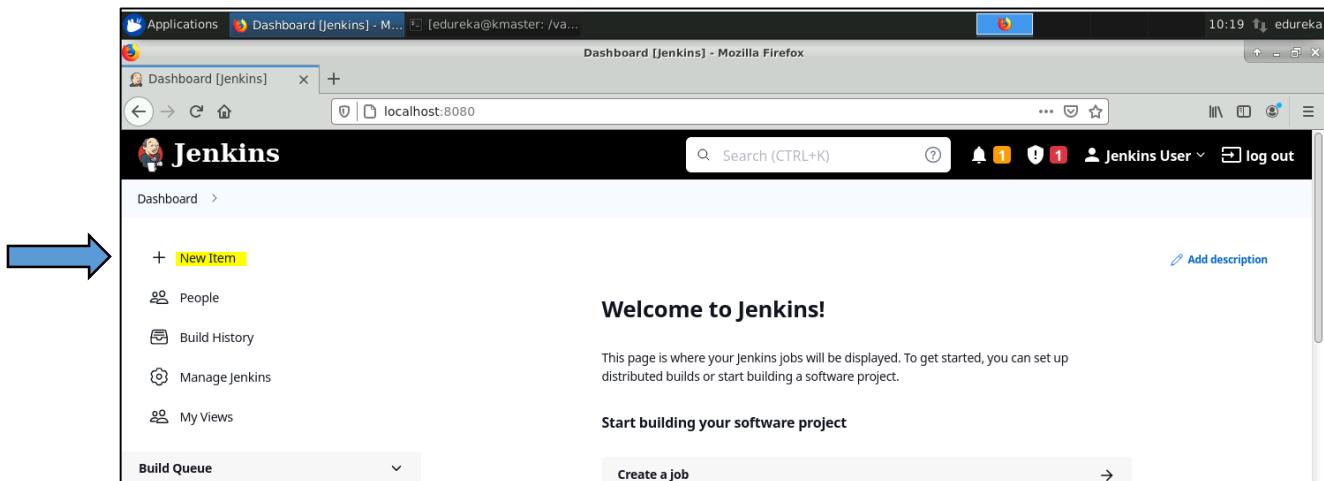


## v. Add Maven to Global Tool Configuration

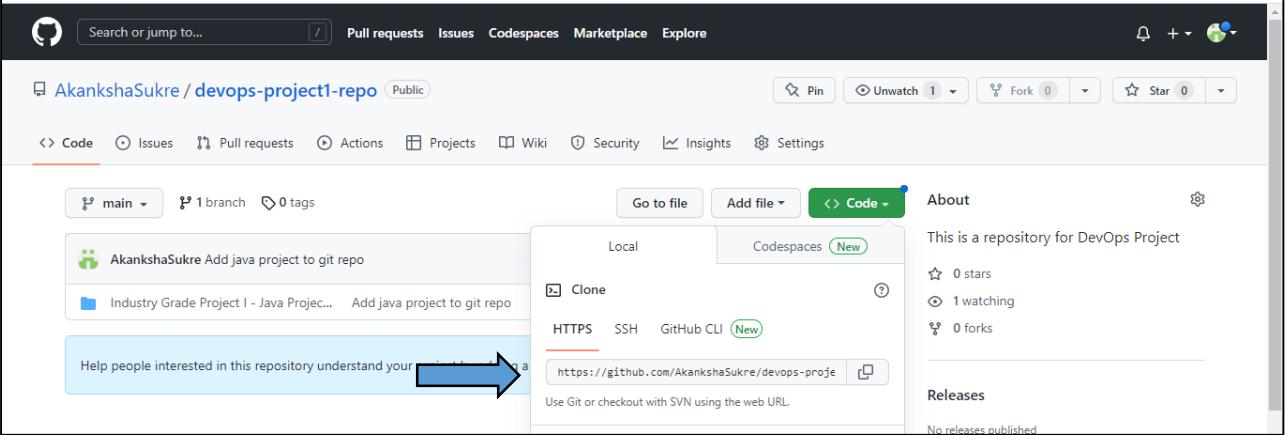


## 3) Create Jobs in Jenkins

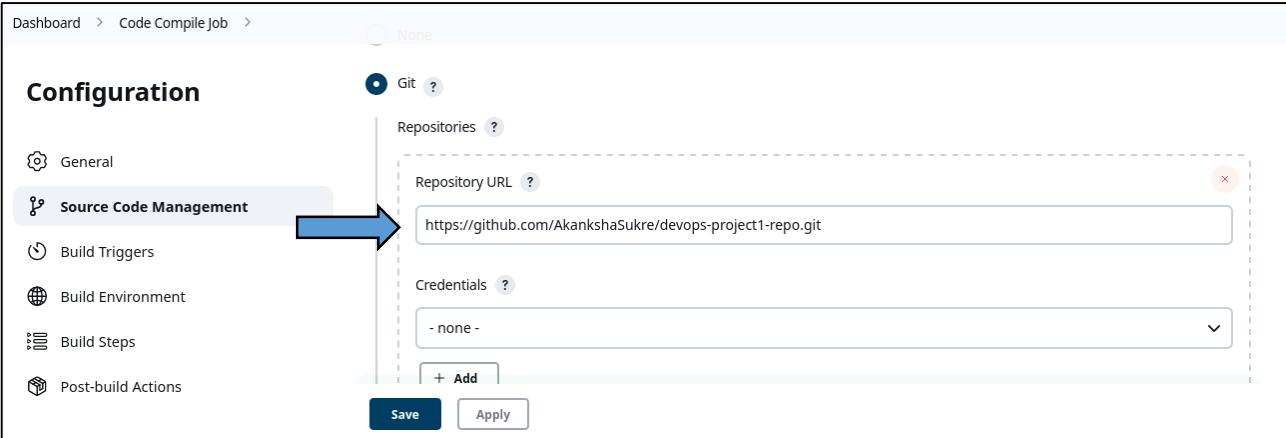
### i. Create Compile Job



ii. Copy Git Repo URL for Configuring SCM in Compile Job

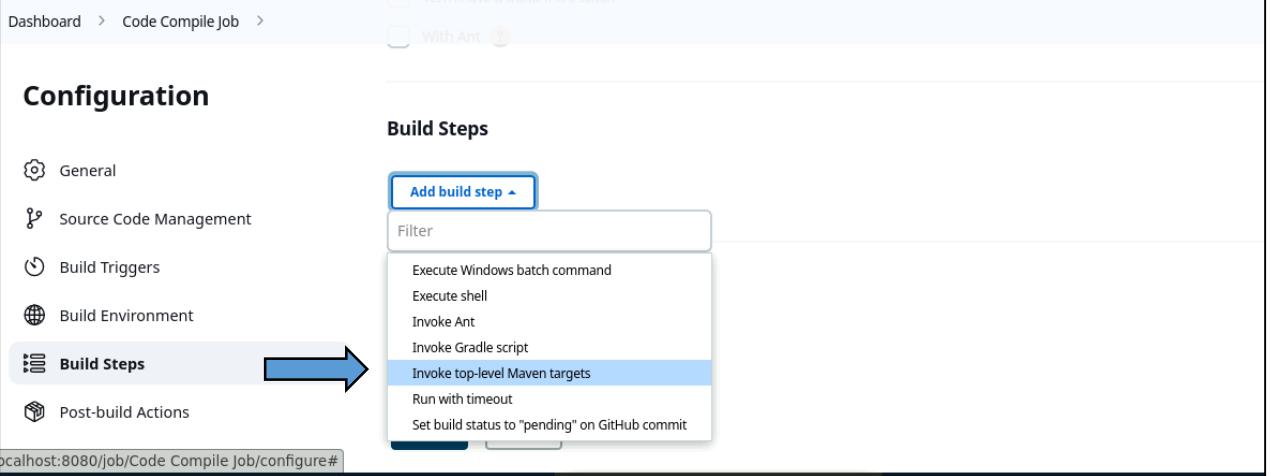


The screenshot shows a GitHub repository page for 'AkankshaSukre / devops-project1-repo'. A blue arrow points from the 'Clone' section to the HTTPS URL: <https://github.com/AkankshaSukre/devops-project1-repo>.



The screenshot shows the 'Source Code Management' section of the Jenkins job configuration. A blue arrow points from the 'Repository URL' field to the value: <https://github.com/AkankshaSukre/devops-project1-repo.git>.

iii. Add Build step to invoke top-level Maven targets



The screenshot shows the 'Build Steps' section of the Jenkins job configuration. A blue arrow points from the 'Add build step' button to the 'Invoke top-level Maven targets' option in the dropdown menu.

iv. Select Maven version and add goal : **compile**

The screenshot shows the Jenkins job configuration interface. On the left, there's a sidebar with links: General, Source Code Management, Build Triggers, Build Environment, **Build Steps** (which is selected and highlighted with a blue arrow), and Post-build Actions. The main area is titled 'Build Steps' and contains a section for 'Invoke top-level Maven targets'. It has fields for 'Maven Version' (set to 'mvn') and 'Goals' (set to 'compile'). There are 'Advanced...' and 'Add build step' buttons, and 'Save' and 'Apply' buttons at the bottom.

v. Build the compile job

The screenshot shows the Jenkins project page for 'Project Code Compile Job'. It includes a 'Status' summary, 'Changes', 'Workspace', and a prominent yellow 'Build Now' button. A blue arrow points to the 'Build Now' button. Below it, there's a 'Permalinks' section.

The screenshot shows the Jenkins console output for job #5. It displays the progress of Maven downloading dependencies from central repositories. The log output includes:

```
Progress (1): 1.7 kB  
Progress (1): 1.7 kB  
  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-error-diagnostics/2.0.6/maven-error-diagnostics-2.0.6.pom (1.7 kB at 4.8 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/commons-cli/commons-cli/1.0/commons-cli-1.0.pom  
Progress (1): 2.1 kB  
  
Downloaded from central: https://repo.maven.apache.org/maven2/commons-cli/commons-cli/1.0/commons-cli-1.0.pom (2.1 kB at 5.9 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-descriptor/2.0.6/maven-plugin-descriptor-2.0.6.pom  
Progress (1): 2.0 kB  
  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-descriptor/2.0.6/maven-plugin-descriptor-2.0.6.pom (2.0 kB at 5.7 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interactivity-api/1.0-alpha-4/plexus-interactivity-api-1.0-alpha-4.pom  
Progress (1): 4.1/7.1 kB  
Progress (1): 7.1 kB  
  
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interactivity-api/1.0-alpha-4/plexus-interactivity-api-1.0-alpha-4.pom (7.1 kB at 20 kB/s)  
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-monitor/2.0.6/maven-monitor-2.0.6.pom  
Progress (1): 1.3 kB
```

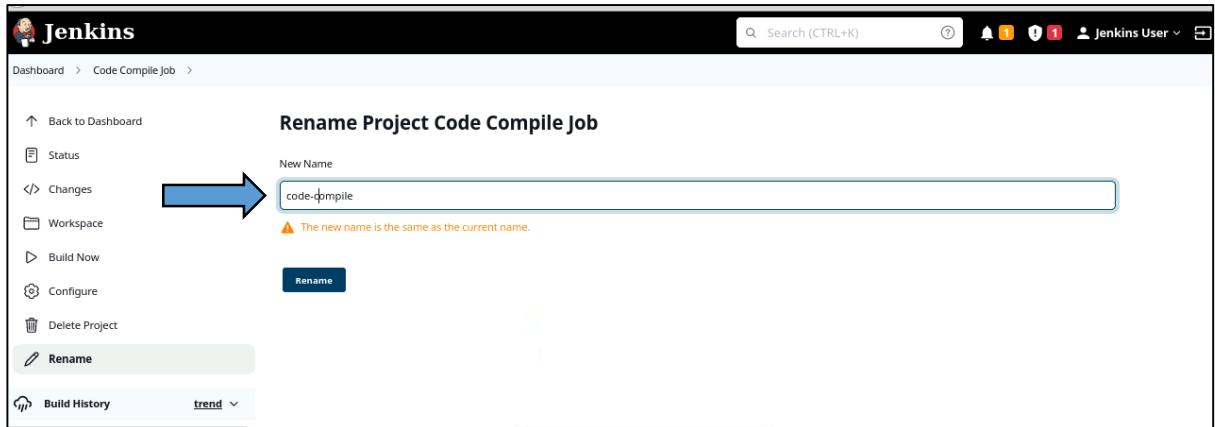
vi. Build successful

The screenshot shows the Jenkins console output for job #5, indicating a successful build. The log output includes:

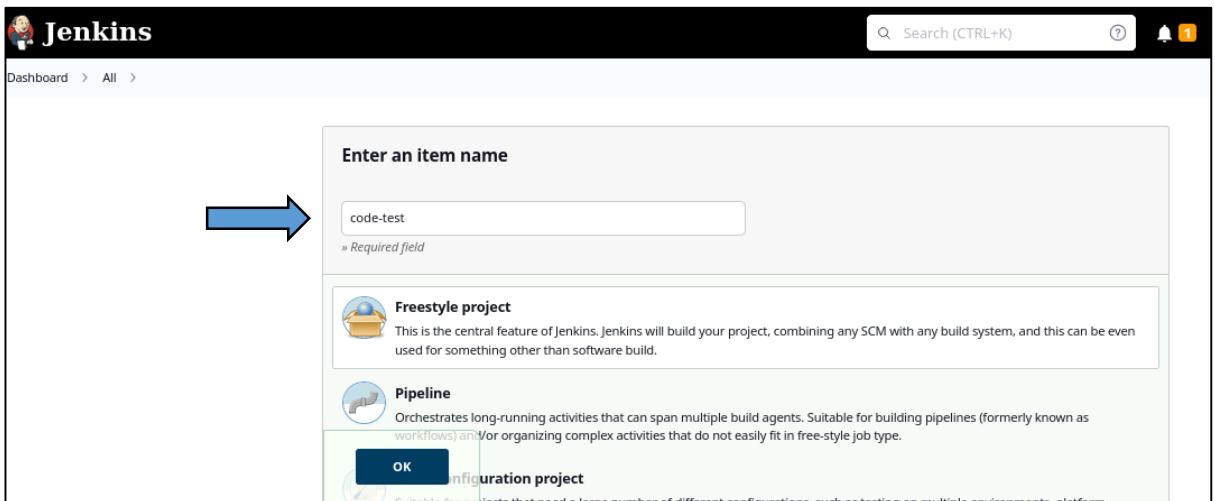
```
Progress (4): 358 kB | 45 kB | 634/640 kB | 121 kB  
Progress (4): 358 kB | 45 kB | 638/640 kB | 121 kB  
Progress (4): 358 kB | 45 kB | 640 kB | 121 kB  
  
Downloaded from central: https://repo.maven.apache.org/maven2/commons-logging/commons-logging-api/1.1/commons-logging-api-1.1.jar (45 kB at 21 kB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/log4j/log4j/1.2.12/log4j-1.2.12.jar (358 kB at 165 kB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.2/junit-3.8.2.jar (121 kB at 53 kB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/collections/google-collections/1.0/google-collections-1.0.jar (640 kB at 277 kB/s)  
[INFO] Changes detected - recompiling the module!  
[INFO] Compiling 3 source files to /var/lib/jenkins/workspace/Code Compile Job/target/classes  
[INFO] .....  
[INFO] BUILD SUCCESS  
[INFO] .....  
[INFO] Total time: 01:57 min  
[INFO] Finished at: 2023-02-03T10:46:24Z  
[INFO] .....  
Finished: SUCCESS
```

A blue arrow points to the 'Finished: SUCCESS' message.

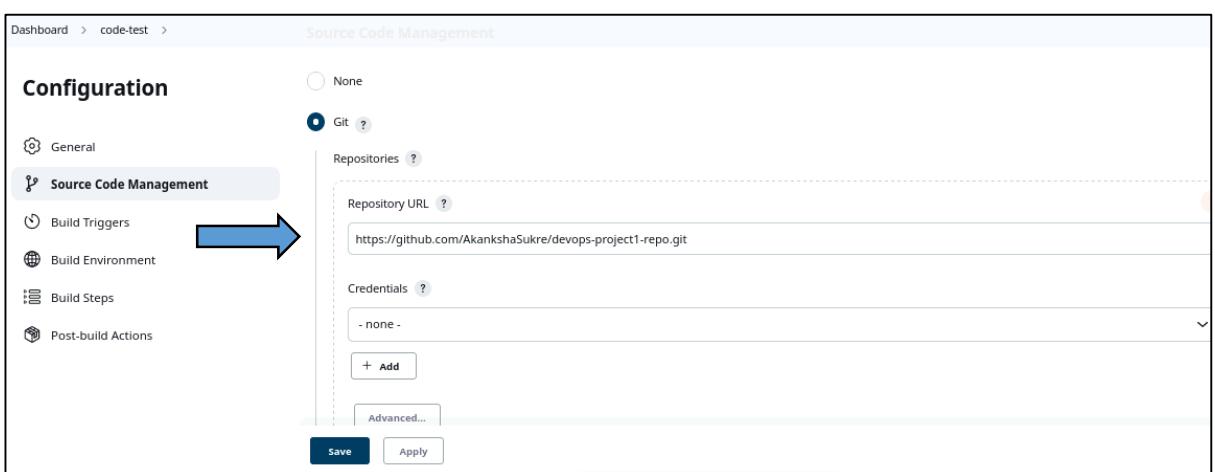
vii. Renamed project for maintaining consistency throughout



viii. Create test job



ix. Configure SCM with GIT Repo URL



x. Add build step as top-level maven targets with goal : **test**

The screenshot shows the Jenkins configuration interface for a job named 'code-test'. The 'Build Steps' section is currently selected. A blue arrow points from the 'Post-build Actions' section to the 'Invoke top-level Maven targets' section. In the 'Goals' field, 'test' is typed.

xi. Add Post build action to publish JUnit test reports

The screenshot shows the Jenkins configuration interface for a job named 'code-test'. The 'Post-build Actions' section is currently selected. A blue arrow points from the 'Build Steps' section to the 'Publish JUnit test result report' section. In the 'Fileset includes' field, 'target/surefire-reports/\*.xml' is typed.

xii. Build the test Job

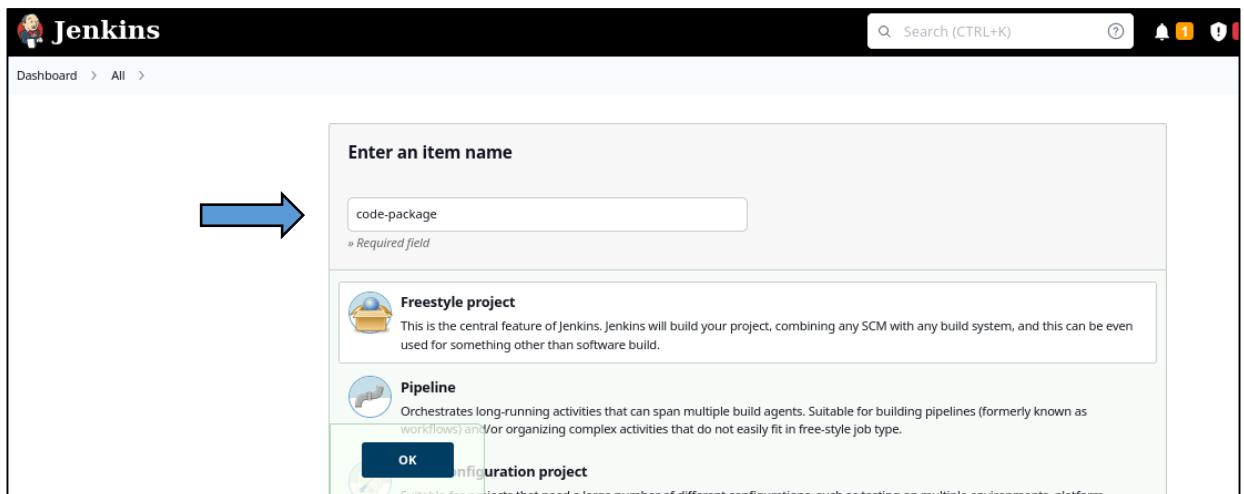
The screenshot shows the Jenkins dashboard for the 'code-test' job. A blue arrow points from the 'Build Now' button to the 'Status' section. The status is 'This job is to test the java code using Maven'.

xiii. Build successful

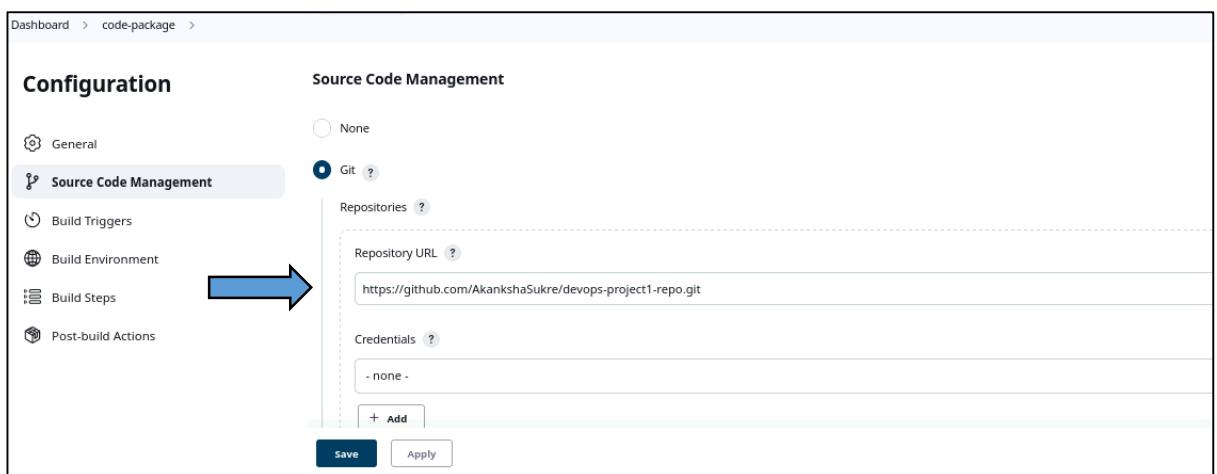
The screenshot shows the Jenkins log for the 'code-test' job, build #2. A blue arrow points to the 'TESTS' section, which shows the execution of 'ProductImplTest' and its results. The final status is 'SUCCESS'.

```
-----  
T E S T S  
-----  
Running com.abc.dataAccessObject.ProductImplTest  
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.178 sec  
Results :  
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0  
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 4.117 s  
[INFO] Finished at: 2023-02-03T11:02:29Z  
[INFO] -----  
Recording test results  
[Checks API] No suitable checks publisher found.  
Finished: SUCCESS
```

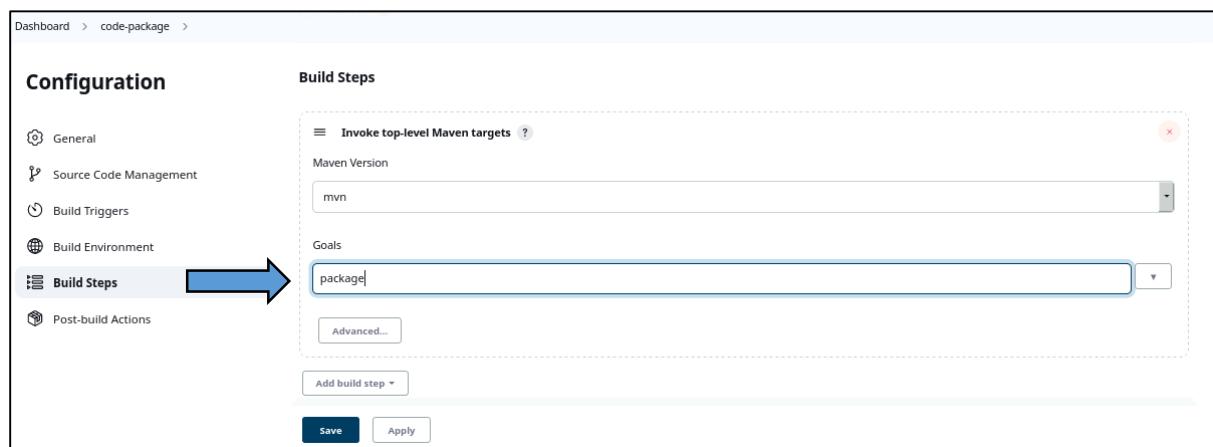
xiv. Create package job



xv. Configure SCM with GIT Repo URL



xvi. Add build step as top-level maven target with goal: **package**



## xvii. Build package Job

The screenshot shows the Jenkins project configuration page for 'code-package'. On the left, there's a sidebar with options like Status, Changes, Workspace, Build Now (which is highlighted with a yellow background and a blue arrow pointing to it), Configure, Delete Project, and Rename. Below the sidebar is a 'Build History' section with a single build entry: '#1 Feb 3, 2023, 11:07 AM'. The main area is titled 'Permalinks'.

## xviii. Build successful and war file generated

The screenshot shows the Jenkins build log for job #1. The log output is as follows:

```
[INFO] Packaging webapp
[INFO] Assembling webapp [ABCtechnologies] in [/var/lib/jenkins/workspace/code-package/target/ABCtechnologies-1.0]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/code-package/src/main/webapp]
[INFO] Webapp assembled in [134 msecs]
[INFO] Building war: /var/lib/jenkins/workspace/code-package/target/ABCtechnologies-1.0.war
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.6:report (jacoco-site) @ ABCtechnologies ---
[INFO] Loading execution data file /var/lib/jenkins/workspace/code-package/target/jacoco.exec
[INFO] Analyzed bundle 'RetailModule' with 2 classes
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  30.974 s
[INFO] Finished at: 2023-02-03T11:07:47Z
[INFO]
Finished: SUCCESS
```

## xix. Three jobs ready : compile,test and package

The screenshot shows the Jenkins dashboard. On the left, there's a sidebar with 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Build Queue' (which shows 'No builds in the queue.'), and 'Build Executor Status' (which shows '1 Idle'). The main area has tabs 'All' and '+'. A table lists three jobs:

S	W	Name	Last Success	Last Failure	Last Duration
✓	⌚	code-compile	23 min #5	26 min #4	2 min 7 sec
✓	💡	code-package	53 sec #1	N/A	34 sec
✓	☁️	code-test	5 min 44 sec #2	14 min #1	7.2 sec

At the bottom, there are icons for 'Icon legend', 'Atom feed for all', 'Atom feed for failures', and 'Atom feed for just latest builds'.

## 4) Create build pipeline with above three jobs

### i. Add post-build action in Compile job to run Test job

The screenshot shows the Jenkins configuration interface for the 'code-compile' job. The left sidebar lists 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build Steps', and 'Post-build Actions'. A blue arrow points from the sidebar to the 'Post-build Actions' section. The 'Post-build Actions' section contains a 'Build other projects' configuration with 'code-test' listed under 'Projects to build'. Below it are three radio button options: 'Trigger only if build is stable' (selected), 'Trigger even if the build is unstable', and 'Trigger even if the build fails'. At the bottom are 'Save' and 'Apply' buttons.

## ii. Add post-build action in Test job to build Package job

The screenshot shows the Jenkins configuration interface for the 'code-test' job. The left sidebar lists 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build Steps', and 'Post-build Actions'. A blue arrow points from the sidebar to the 'Post-build Actions' section. The 'Post-build Actions' section contains a 'Build other projects' configuration with 'code-package' listed under 'Projects to build'. Below it are three radio button options: 'Trigger only if build is stable' (selected), 'Trigger even if the build is unstable', and 'Trigger even if the build fails'. At the bottom are 'Save' and 'Apply' buttons.

## iii. Install Build pipeline plugin

The screenshot shows the Jenkins Plugin Manager. The top navigation bar includes 'Dashboard', 'Manage Jenkins', and 'Plugin Manager'. The 'Available' tab is selected. A search bar contains the text 'build pipeline'. A plugin card for 'Build Pipeline' version 1.5.8 is highlighted with a yellow background, showing it is 'Released'. The card describes the plugin as rendering upstream and downstream connected jobs for a build pipeline. At the bottom of the card is a note: 'This plugin renders upstream and downstream connected jobs that typically form a build pipeline. In addition, it offers the ability to define manual triggers for jobs that require intervention prior to execution, e.g. an approval process outside of Jenkins.' Below the card is an 'Install' button with a checked checkbox.

The screenshot shows the Jenkins Update Center. The top navigation bar includes 'Dashboard', 'Manage Jenkins', and 'Update Center'. The 'Available' tab is selected. A search bar contains the text 'Build Pipeline'. A plugin card for 'Build Pipeline' version 1.5.8 is highlighted with a yellow background, showing it is 'Success'. The card describes the plugin as rendering upstream and downstream connected jobs for a build pipeline. At the bottom of the card is a note: 'This plugin renders upstream and downstream connected jobs that typically form a build pipeline. In addition, it offers the ability to define manual triggers for jobs that require intervention prior to execution, e.g. an approval process outside of Jenkins.' Below the card is an 'Install' button with a checked checkbox. At the bottom of the page are links to 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.

iv. Create a new view of type: **Build Pipeline View**

Jenkins

Dashboard > Jenkins User > My Views

View name: ci-pipeline

Type:

- Build Pipeline View
- Include a global view
- List View

localhost:8080

v. Set up initial job to run as Compile Job

Based on upstream/downstream relationship

This layout mode derives the pipeline structure based on the upstream/downstream trigger relationship between jobs. This is the only out-of-the-box supported layout mode, but is open for extension.

Upstream / downstream config

Select Initial Job: code-compile

Trigger Options

Build Cards

vi. Build pipeline is ready

Build Pipeline

This is the pipeline view for jobs created in Jenkins

Trigger a Pipeline Pipeline History Configure Add Step Run History Configure Add Step Delete Manage

Pipeline #5 #5 code-compile Feb 3, 2023 10:44:17 AM 2 min 7 sec Jenkins trigger

code-test #3 code-test Feb 3, 2023 11:15:04 AM 51 sec Jenkins trigger

code-package #2 code-package Feb 3, 2023 11:15:19 AM 7 sec Jenkins trigger

REST API Jenkins 2.361.1

Build Pipeline

This is the pipeline view for jobs created in Jenkins

Trigger a Pipeline Pipeline History Configure Add Step Run History Configure Add Step Delete Manage

Pipeline #6 #6 code-compile Feb 3, 2023 11:16:40 AM 5.6 sec Jenkins trigger

#3 code-test Feb 3, 2023 11:15:04 AM 51 sec Jenkins trigger

#2 code-package Feb 3, 2023 11:15:19 AM 7 sec Jenkins trigger

## 5) Set up master-slave nodes to distribute tasks in the pipeline

### i. Go to “Manage Jenkins” and Click on “Manage Nodes and Clouds”

The screenshot shows the Jenkins Manage Jenkins dashboard. On the left, there are dropdown menus for 'Build Queue' and 'Build Executor Status'. On the right, there are sections for 'System Configuration' (Configure System, Global Tool Configuration, Manage Plugins), and a central section for 'Manage Nodes and Clouds' which includes a brief description and a 'Manage Nodes and Clouds' button. A large blue arrow points to the 'Manage Nodes and Clouds' button.

### ii. Click on “New Node”

The screenshot shows the 'Manage nodes and clouds' page. On the left, there are links for 'Back to Dashboard', 'Manage Jenkins', '+ New Node' (which is highlighted with a blue arrow), 'Configure Clouds', and 'Node Monitoring'. On the right, there is a table showing one node: 'Built-In Node' (Architecture: Linux (amd64), Clock Difference: In sync, Free Disk Space: 4.60 GB). Below the table, it says 'Data obtained' and '15 min' for clock difference and free disk space. A 'Build Queue' section at the bottom shows 'No builds in the queue.'

### iii. Create a node named “slave”

The screenshot shows the 'New node' configuration page. On the left, there are links for 'Back to Dashboard', 'Manage Jenkins', '+ New Node' (highlighted with a blue arrow), 'Configure Clouds', and 'Node Monitoring'. On the right, there are fields for 'Node name' (set to 'slave'), 'Type' (selected 'Permanent Agent'), and a detailed description of what a permanent agent is. At the bottom, there is a 'Create' button. The 'Build Queue' and 'Build Executor Status' sections at the bottom show 'No builds in the queue.' and '1 Idle' and '2 Idle' respectively.

iv. Add slave node details as shown below

The screenshot shows the Jenkins 'Nodes > slave' configuration page. It includes fields for Usage (set to 'Use this node as much as possible'), Launch method (set to 'Launch agents via SSH'), Host (set to '172.31.18.87'), and Credentials (set to 'edureka'). A blue arrow points to the 'Host' input field.

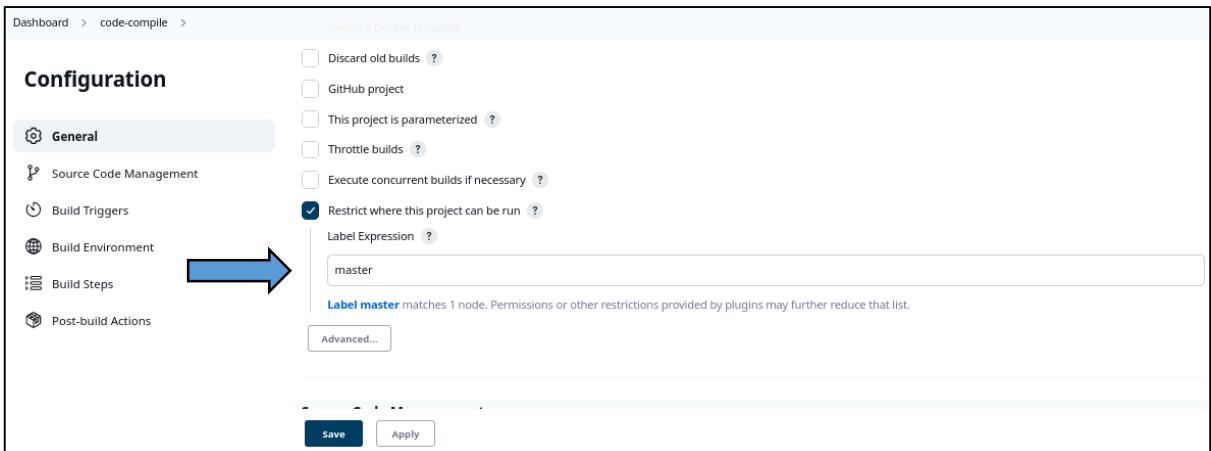
v. Node is added successfully

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
1	Built-In Node	Linux (amd64)	In sync	4.55 GB	0 B	4.55 GB	0ms
2	slave	Linux (amd64)	In sync	4.74 GB	0 B	4.74 GB	83ms

vi. Add label to Built-In Node as “master”

The screenshot shows the Jenkins 'Nodes > Built-In Node' configuration page. It includes sections for Status, Configure, Build History, Load Statistics, and Script Console. The 'Labels' section shows 'master' listed under 'Projects tied to Built-In Node'. A blue arrow points to the 'master' label.

vii. Configure “compile job” to run on master node



viii. Configure “test job” and “package” job to run on slave node

The screenshot shows the Jenkins configuration interface for the 'code-test' project. In the 'General' section, the 'Restrict where this project can be run' checkbox is checked. Below it, the 'Label Expression' field contains 'slave'. A blue arrow points from the 'Source Code Management' section on the left towards the 'Label Expression' field. The 'Source Code Management' section is expanded, showing 'Git' selected. At the bottom are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins configuration interface for the 'code-package' project. In the 'General' section, the 'Restrict where this project can be run' checkbox is checked. Below it, the 'Label Expression' field contains 'slave'. A blue arrow points from the 'Source Code Management' section on the left towards the 'Label Expression' field. The 'Source Code Management' section is expanded, showing 'Git' selected. At the bottom are 'Save' and 'Apply' buttons.

## 6) GITScm Polling for CI

- i. Add build trigger in compile job to Poll SCM. Configure a schedule to check for changes in repo every minute

The screenshot shows the Jenkins 'Configuration' screen for a project named 'code-compile'. On the left, there's a sidebar with options: General, Source Code Management, Build Triggers (which is highlighted with a blue arrow), Build Environment, Build Steps, and Post-build Actions. In the main area, under 'Build Triggers', there are several checkboxes: 'Trigger builds remotely (e.g., from scripts)', 'Build after other projects are built', 'Build periodically', and 'GitHub hook trigger for GITScm polling'. Below these is a checkbox for 'Poll SCM' with a question mark icon, which is checked. To its right is a 'Schedule' button with a question mark icon, followed by a text input field containing '\*\*\*\*\*'. At the bottom are 'Save' and 'Apply' buttons.

- ii. Test by updating Readme file in Repo

The screenshot shows a GitHub repository page for 'AkankshaSukre'. At the top, it says 'main' (branch), '1 branch', '0 tags'. Below that is a list of commits:

- src: Add java project to git repo (last week)
- README.md: Update README.md (now)
- pom.xml: Add java project to git repo (last week)
- pom.xml.bak: Add java project to git repo (last week)

At the bottom of the list is another 'README.md' entry with a pencil icon.

- iii. Check Git Polling Log

The screenshot shows the 'Git Polling Log' section of the Jenkins dashboard. On the left is a sidebar with links: Back to Dashboard, Status, Changes, Workspace, Build Now, Configure, Delete Project, and Git Polling Log (which is highlighted with a blue arrow). The main area is titled 'Git Polling Log' and contains the following text:

```
Started on Feb 13, 2023, 2:03:00 PM
Using strategy: Default
[poll] Last Built Revision: Revision 5ce53daf4e58af87a326d79d4504bc49c4ecb923 (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
> git --version # timeout=10
> git --version # 'git version 2.17.1'
> git ls-remote -h -- https://github.com/AkankshaSukre/devops-project1-repo.git # timeout=10
Found 1 remote heads on https://github.com/AkankshaSukre/devops-project1-repo.git
[poll] Latest remote head revision on refs/heads/main is: feb87d074bb868ed55c039a96a2b8aff85f192bf
Done. Took 0.37 sec
Changes found
```

- iv. Pipeline executed successfully

The screenshot shows the 'Build History' section of the Jenkins dashboard. On the left is a sidebar with links: People, Build History (which is highlighted with a blue arrow), Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. The main area is a table with columns: S, W, Name, Last Success, Last Failure, and Last Duration. The data is as follows:

S	W	Name	Last Success	Last Failure	Last Duration
✓	⌚	code-compile	1 min 51 sec #9	10 days #4	17 sec ▶
✓	☁️	code-package	51 sec #10	9 days 17 hr #7	14 sec ▶
✓	⌚	code-test	1 min 26 sec #6	10 days #1	28 sec ▶

## Task 3: Create CI/CD pipeline in Jenkins to build and deploy on Docker container

### 1) Integrate Docker with Jenkins

#### i. Install Docker plugin in Jenkins

The screenshot shows the Jenkins Plugin Manager interface. A blue arrow points to the 'Docker' plugin entry in the list. The 'Docker' plugin is version 1.3.0, released 9 days 12 hr ago. It is described as integrating Jenkins with Docker. Below it is the 'Docker Commons' plugin, version 1.21, released 5 mo 10 days ago. At the bottom are 'Install without restart', 'Download now and install after restart', and 'Check now' buttons.

#### ii. Go to Manage Jenkins → Configure Clouds → Add a new cloud Docker

The screenshot shows the Jenkins 'Configure Clouds' page. A blue box highlights the 'Add a new cloud' button. Below it is a dropdown menu set to 'Docker'. At the bottom are 'Save' and 'Apply' buttons.

#### iii. Make the highlighted changes in docker service file on Docker host(Jenkins master node)

```
edureka@kmaster: ~/devops-project
[1]+  Stopped                  sudo nano /lib/systemd/system/docker.service
edureka@kmaster:~/devops-projects$ sudo nano /lib/systemd/system/docker.service
edureka@kmaster:~/devops-projects$ cat /lib/systemd/system/docker.service
[Unit]
Description=Docker Application Container Engine
Documentation=https://docs.docker.com
BindsTo=containerd.service
After=network-online.target firewalld.service containerd.service
Wants=network-online.target
Requires=docker.socket

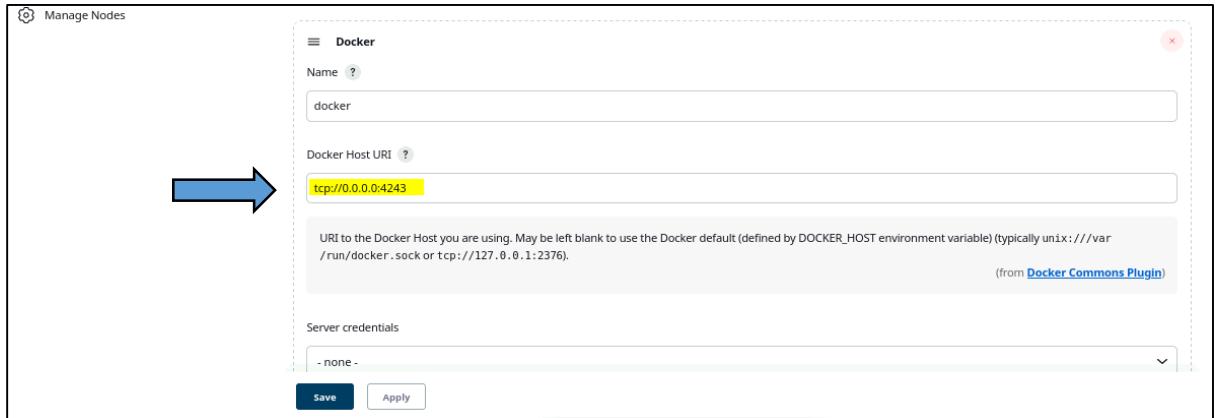
[Service]
Type=notify
# the default is not to use systemd for cgroups because the delegate issues still
# exists and systemd currently does not support the cgroup feature set required
# for containers run by docker
ExecStart=/usr/bin/dockerd -H tcp://0.0.0.0:4243 -H unix:///var/run/docker.sock
ExecReload=/bin/kill -s HUP $MAINPID
TimeoutSec=0
RestartSec=2
Restart=always

# Note that StartLimit* options were moved from "Service" to "Unit" in systemd 229.
# Both the old, and new location are accepted by systemd 229 and up, so using the old location
# to make them work for either version of systemd.
StartLimitBurst=3
```

#### iv. Restart the docker service

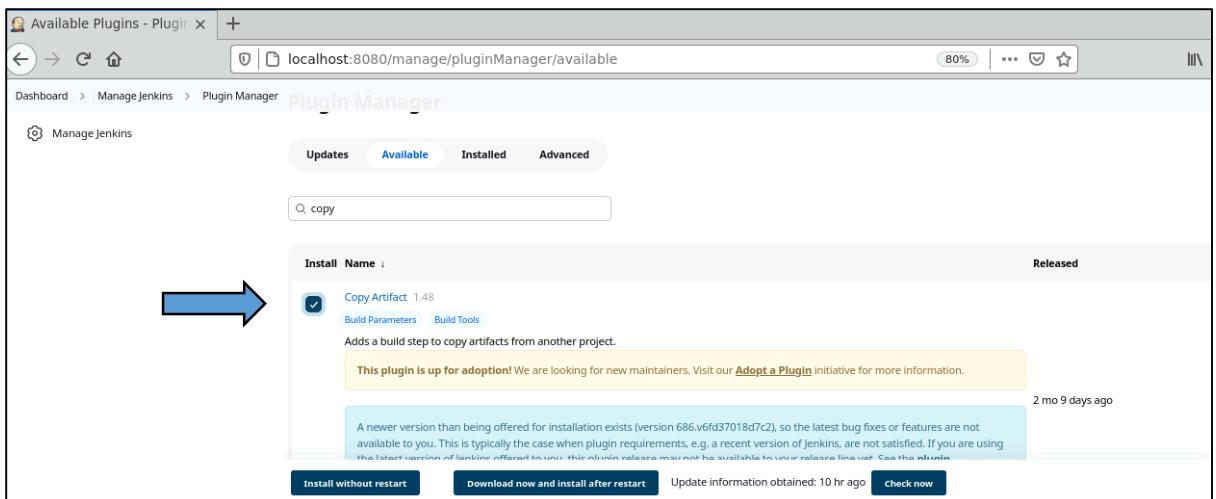
```
edureka@kmaster:~/devops-project$ sudo systemctl daemon-reload
edureka@kmaster:~/devops-project$ sudo service docker restart
edureka@kmaster:~/devops-project$ curl http://localhost:4243/version
{"Platform":{"Name":"","Components":[{"Name":"Engine","Version":"19.03.6","Details":{"ApiVersion":"1.40","Arch":"amd64","BuildTime":"2020-02-19T01:06:16.000000000+00:00","Experimental":false,"GitCommit":"369ce74a3c","GoVersion":"golang1.12.17","KernelVersion":"4.15.0-1021-aws","MinAPIVersion":1.1,"Os":"linux"}, {"Name":"containerd","Version":"1.3.3-0ubuntu1-18.04.2","Details":{"GitCommit":""}}, {"Name":"runc","Version":"spec: 1.0.1-dev","Details":{"GitCommit":""}}, {"Name":"dockerc init","Version":"0.18.0","Details":{"GitCommit":""}}], "Version": "19.03.6", "ApiVersion": "1.40", "MinAPIVersion": "1.12", "GitCommit": "369ce74a3c", "GoVersion": "golang1.12.17", "Os": "linux", "Arch": "amd64", "KernelVersion": "4.15.0-1021-aws", "BuildTime": "2020-02-19T01:06:16.000000000+00:00"}}
edureka@kmaster:~/devops-project$
```

#### v. Add the Docker Host URI in Docker cloud properties in Jenkins and save the configuration



### 2) CI/CD job on Jenkins to build and deploy on Docker container

#### i. Install the Copy Artifact plugin in Jenkins



ii. Create a new Jenkins job: **docker-cicd** and add copy artifact as the first build step

The screenshot shows the Jenkins 'Configuration' page for a job named 'docker-cicd'. The left sidebar lists 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build Steps' (which is selected), and 'Post-build Actions'. The right panel shows a 'Build Steps' section with a 'Copy artifacts from another project' step. A blue arrow points to the 'Source Code Management' link in the sidebar.

iii. Add second build step as Build/Publish Docker Image. Give Dockerfile path in the required field and specify the cloud "docker" to build the image

The screenshot shows the Jenkins 'Configuration' page for the 'docker-cicd' job. The 'Build Steps' section contains a 'Build / Publish Docker Image' step. The 'Directory for Dockerfile' field is set to '/home/edureka/devops-project'. The 'Cloud' dropdown is set to 'docker'. A blue arrow points to the 'Source Code Management' link in the sidebar.

iv. Add docker hub credentials and push latest image to registry

The screenshot shows the Jenkins 'Configuration' page for the 'docker-cicd' job. The 'Build Steps' section contains a 'Build / Publish Docker Image' step. The 'Image' field is set to 'akankshasukre/abc-tech'. The 'Push image' checkbox is checked. The 'Registry Credentials' dropdown is set to 'akankshasukre/\*\*\*'. A blue arrow points to the 'Build Steps' link in the sidebar.

v. Add third build step to execute Shell command to create container

The screenshot shows the Jenkins 'Configuration' page for the 'docker-cicd' job. The 'Build Steps' section contains an 'Execute shell' step. The 'Command' field contains the following Docker command:

```
docker stop abc.tech || true
docker run --rm -p 7001:8080 -d --name abc-tech abc-tech:latest
```

Checkboxes for 'Disable caching' and 'Pull base image' are visible at the top of the 'Execute shell' section. A blue arrow points to the 'Build Steps' link in the sidebar.

vi. Build the job docker-cicd

The screenshot shows the Jenkins dashboard for the project 'docker-cicd'. On the left, there's a sidebar with options like Status, Changes, Workspace, Build Now (which is highlighted in yellow), Configure, Delete Project, and Rename. The main area is titled 'Permalinks' and lists the last seven builds:

- Last build (#35), 2 min 14 sec ago
- Last stable build (#35), 2 min 14 sec ago
- Last successful build (#35), 2 min 14 sec ago
- Last failed build (#33), 11 min ago
- Last unsuccessful build (#33), 11 min ago
- Last completed build (#35), 2 min 14 sec ago

vii. Build successful

The screenshot shows the Jenkins build log for job #35. It displays the command-line output of the build process, which includes pushing Docker images and running a Docker container. A large blue arrow points from the right towards the bottom of the log output.

```
2a89delea448: Pushing [====>] 494.1kB/7.133MB
2a89delea448: Pushing [=====>] 690.7kB/7.133MB
2a89delea448: Pushing [=====>>] 1.674MB/7.133MB
d543b8cad89e: Layer already exists
2a89delea448: Pushing [=====>>>] 2.657MB/7.133MB
2a89delea448: Pushing [=====>>>>] 4.23MB/7.133MB
2a89delea448: Pushing [=====>>>>>] 5.999MB/7.133MB
2a89delea448: Pushing [=====>>>>>>] 7.137MB
414826190114: Pushed
2a89delea448: Pushed
latest: digest: sha256:a04af6003b62438d9ab34281528672a3cd34372580db89776d584dc2e2e63e48 size: 2206
Docker Build Done
[docker-cicd] $ /bin/sh -xe /tmp/jenkins10604719188306844522.sh
+ docker stop abc-tech
abc-tech
+ docker run --rm -p 7001:8080 -d --name abc-tech abc-tech:latest
ee549f1b8980da5feec44e75f00b74c2c14721861b73d891c4ee1f9c7473560f
Finished: SUCCESS
```

viii. Latest image built and published to docker hub as seen below

The screenshot shows the Docker Hub repository page for 'akankshasukre/abc-tech:latest'. It displays the image details: OS/ARCH (linux/amd64), COMPRESSED SIZE (250.13 MB), LAST PUSHED (3 minutes ago by akankshasukre), and TYPE (Image). Below this, the 'IMAGE LAYERS' section shows three layers: 1 ARG RELEASE (0 B), 2 ARG LAUNCHPAD\_BUILD\_ARCH (0 B), and 3 LABEL org.opencontainers.image.ref.name=ubuntu (0 B). A red 'Delete Tag' button is visible on the right.

**Note: Dockerfile is created and tested on Docker Host before creating this CI/CD pipeline in Jenkins.**

Dockerfile is as below:

```
FROM tomcat:8.5-jdk11-temurin-focal

LABEL maintainer="Akanksha K Sukre"

WORKDIR /home/edureka/devops-project

ADD /target/ABCtechnologies-1.0.war /usr/local/tomcat/webapps/

EXPOSE 8080

CMD ["catalina.sh", "run"]
```

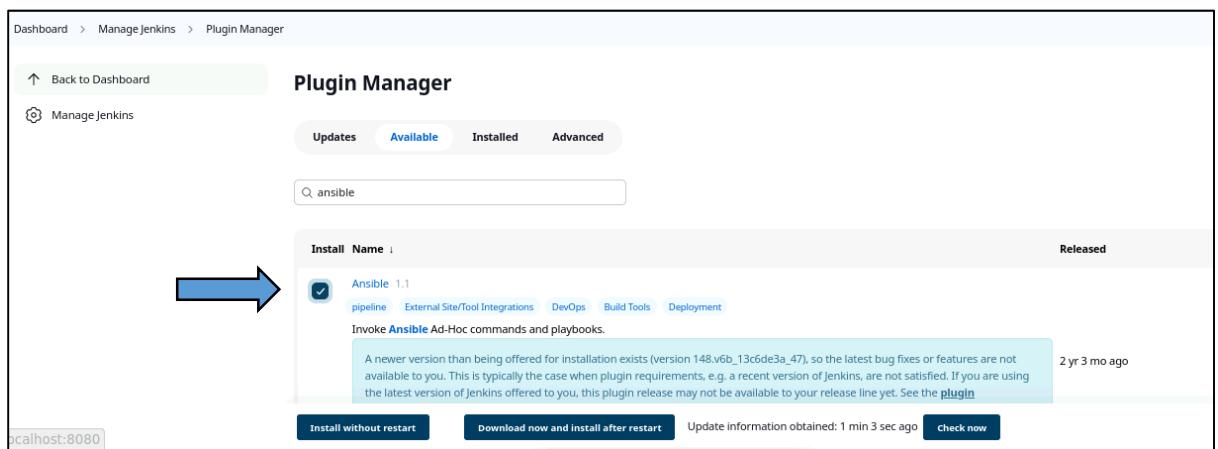
- ix. Container created on Docker host and App is running on the tomcat server at **http://localhost:7001/ABCtechnologies-1.0**



## Task 4: Integrate Docker and Kubernetes with Ansible and create CI/CD pipeline in Jenkins to deploy the application on Kubernetes

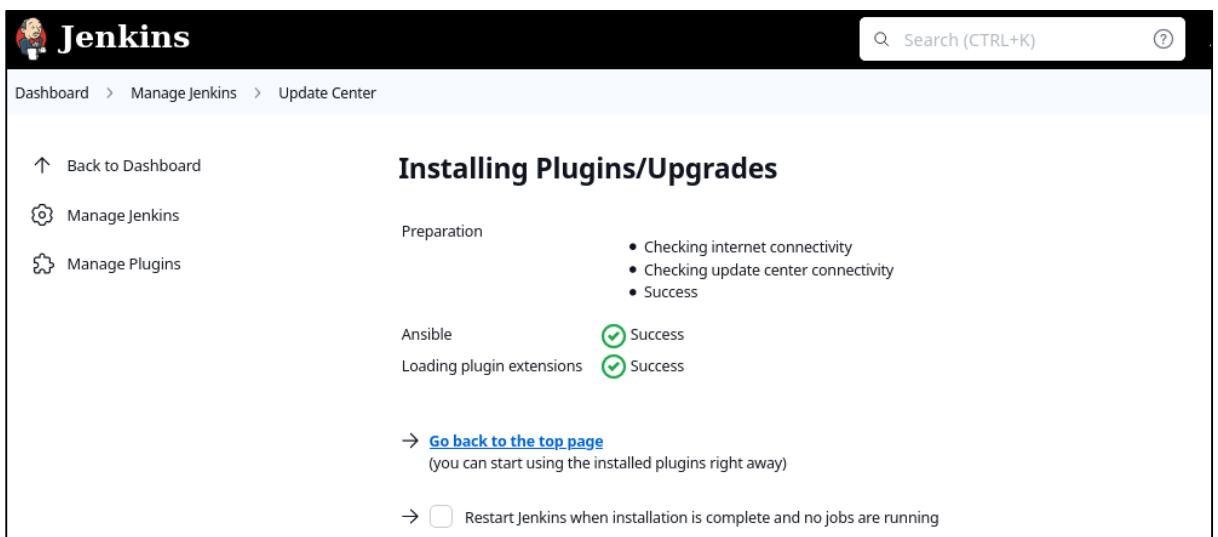
### 1) Integrate Ansible with Jenkins

#### i. Install Ansible plugin in Jenkins



The screenshot shows the Jenkins Plugin Manager interface. A blue arrow points to the 'Ansible' plugin entry in the list. The 'Available' tab is selected. The 'Ansible' plugin is version 1.1, released 2 years 3 months ago. It is categorized under 'pipeline', 'External Site/Tool Integrations', 'DevOps', 'Build Tools', and 'Deployment'. A note indicates that a newer version is available for installation. Buttons at the bottom include 'Install without restart', 'Download now and install after restart', and 'Check now'.

#### ii. Ansible plugin installed successfully



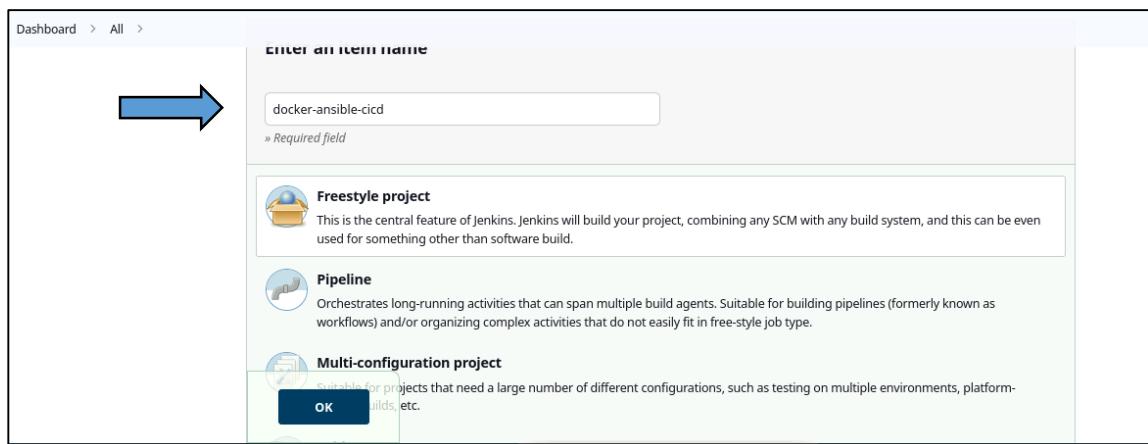
The screenshot shows the Jenkins Update Center. The 'Installing Plugins/Upgrades' section displays the successful installation of the 'Ansible' plugin. Preparation steps listed are: Checking internet connectivity, Checking update center connectivity, and Success. Under Ansible, 'Loading plugin extensions' is shown as successful. A note says you can start using the installed plugins right away. A checkbox option to restart Jenkins is present at the bottom.

2) Create CI/CD pipeline to create docker image and container using Ansible playbook

- i. Write Ansible playbook to create docker image and container. Playbook contents are as follows:

```
- hosts: localhost
  vars:
    ansible_python_interpreter: /usr/bin/python
    docker_pwd: !vault |
      $ANSIBLE_VAULT;1.1;AES256
      6336626166616661623231643363623764326433616363337376531386535316566663130656330
      32393437383861386437383661323932336330646431380a666634373166306237356138666334
      386136343537616634353266633238646134616366356463632373965646230353830373064636130
      626334633306438330a336436626462303166396137306136353732643839353065386564386237
      3838
  become_method: sudo
  tasks:
    - name: Log into DockerHub
      docker_login:
        username: akankshasukre
        password: '{{ docker_pwd }}'
    - name: Build Docker image from Dockerfile
      docker_image:
        name: abc-tech
        state: present
        source: build
        build:
          path: .
          push: yes
          tag: latest
          repository: akankshasukre/abc-tech
    - name: Running the container
      docker_container:
        image: abc-tech:latest
        name: abc-tech
        recreate: yes
        state: started
        ports:
          - "7001:8080"
```

- ii. Create a new job: **docker-ansible-cicd**



iii. Add a build step to invoke Ansible playbook

The screenshot shows the Jenkins configuration interface for a project named 'docker-ansible-cicd'. On the left, there's a sidebar with options like General, Source Code Management, Build Triggers, Build Environment, Build Steps (which is selected and highlighted with a blue arrow), and Post-build Actions. The main area is titled 'Build Steps' and contains a list of available steps. One step, 'Invoke Ansible Playbook', is highlighted with a yellow box and a blue arrow pointing to it from the sidebar.

iv. Add path to playbook and hosts file in the required field as shown below:

This screenshot shows the configuration for the 'Invoke Ansible Playbook' step. It has two fields: 'Playbook path' containing '/home/edureka/devops-project/main.yaml' and 'Inventory' which is set to 'File or host list' with the value '/etc/ansible/hosts'. A blue arrow points from the sidebar's 'Build Steps' section to this configuration screen.

v. Build the job

The screenshot shows the Jenkins project dashboard for 'Project docker-ansible-cicd'. It features a sidebar with links like Back to Dashboard, Status (which is selected and highlighted with a green box), Changes, Workspace, Configure (highlighted with a yellow box), Delete Project, and Rename. Below the sidebar, there's a 'Permalinks' section with a 'Build Now' button, which is also highlighted with a yellow box. At the bottom, there's a 'Build History' section and a 'trend' dropdown.

## vi. Build is successful

The screenshot shows the Jenkins interface with a build history. The current build's console output is displayed, showing the execution of an Ansible playbook. The output includes logs for gathering facts, logging into DockerHub, building a Docker image, and running the container. The status indicates success with 'ok' for all tasks.

```

Building on the built-in node in workspace /var/lib/jenkins/workspace/docker-ansible-cicd
[docker-ansible-cicd] $ sshpass ***** ansible-playbook /home/edureka/devops-project/main.yaml -i /etc/ansible/hosts -f 5 -u akankshasukre -k --vault-password-file /var/lib/jenkins/workspace/docker-ansible-cicd/vault12449626431283027893.password -e =
PLAY [localhost] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Log into DockerHub] ****
ok: [localhost]

TASK [Build Docker image from Dockerfile] ****
[WARNING]: The default for build.pull is currently 'yes', but will be changed to 'no' in Ansible 2.12. Please set build.pull explicitly to the value you need.
ok: [localhost]

TASK [Running the container] ****
changed: [localhost]

```

## vii. View the image and container created on the docker host

The screenshot shows two terminal sessions. The top session lists Docker images with details like tag, image ID, creation time, and size. The bottom session lists Docker containers with details like name, image, command, creation time, status, ports, and names.

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
abc-tech	latest	7919121af32b	About a minute ago	478MB
tomcat	8.5-jdk11-temurin-focal	c4976357245b	6 days ago	471MB
tomcat	9-jdk11	27f1e37a7820	7 days ago	468MB
rancher/mirrored-flannelcni-flannel-cni-plugin	v1.1.2	7a2dcab94698	2 months ago	7.97MB
rancher/mirrored-flannelcni-flannel	v0.20.2	b5f6c9203f83	2 months ago	59.6MB
k8s.gcr.io/kube-proxy	v1.18.20	27f8b8d51985	20 months ago	117MB
k8s.gcr.io/kube-apiserver	v1.18.20	7d8d2960de69	20 months ago	173MB
k8s.gcr.io/kube-controller-manager	v1.18.20	e7c545a60706	20 months ago	162MB
k8s.gcr.io/kube-scheduler	v1.18.20	a05ala79adaa	20 months ago	96.1MB
k8s.gcr.io/pause	3.2	80d28bedfe5d	2 years ago	683kB
k8s.gcr.io/coredns	1.6.7	67da37a9a360	3 years ago	43.8MB
k8s.gcr.io/etcld	3.4.3-0	303ce5db0e90	3 years ago	288MB

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
8dd364b38c3d	abc-tech:latest	"catalina.sh run"	2 minutes ago	Up 2 minutes	0.0.0.0:7001->8080/tcp	abc-tech
2d378d6973c3	67da37a9a360	"/coredns -conf /etc."	22 minutes ago	Up 22 minutes		k8s_coredns_core
dns-66bfff467f8-wkqhx	_kube-system_b642060f-2dcc-4211-bc8c-25122e24b7f7_9					
eb4419046603	k8s.gcr.io/pause:3.2	"/pause"	22 minutes ago	Up 22 minutes		k8s_POD_coredns-
66bfff467f8-wkqhx	_kube-system_b642060f-2dcc-4211-bc8c-25122e24b7f7_49					
97b6fbff409e	67da37a9a360	"/coredns -conf /etc..."	22 minutes ago	Up 22 minutes		k8s_coredns_core
dns-66bfff467f8-zvpsr	_kube-system_ea24494f-9bee-4deb-a262-dacace7e6356_10					

## viii. The app is running successfully as seen below



### 3) Deploy artifacts to Kubernetes

- Create pod-definition.yaml file. The file is as below:

```
apiVersion: v1
kind: Pod
metadata:
  name: abc-tech
  labels:
    type: webapp
spec:
  containers:
    - name: abc-tech-app
      image: akankshasukre/abc-tech:latest
      ports:
        - containerPort: 8080
```

- Create pod using this file and verify using command: kubectl get pods and describe pod

→

```
edureka@kmaster:~/devops-project/kubernetes$ kubectl create -f pod-definition.yaml
pod/abc-tech created
edureka@kmaster:~/devops-project/kubernetes$ kubectl get pods
NAME      READY   STATUS    RESTARTS   AGE
abc-tech   1/1     Running   0          6s
edureka@kmaster:~/devops-project/kubernetes$ kubectl describe pod abc-tech
Name:           abc-tech
Namespace:      default
Priority:      0
Node:          kslave1/172.31.18.87
Start Time:    Sat, 11 Feb 2023 14:13:10 +0000
Labels:         type=webapp
Annotations:   <none>
Status:        Running
IP:            10.244.1.3
IPs:
  IP: 10.244.1.3
Containers:
  abc-tech-app:
    Container ID:  docker://29d152ce36015eaf1c0bb8effee9455d96b5c950a2ecbe94c91b3b61b96e3e54
    Image:         akankshasukre/abc-tech:latest
    Image ID:     docker-pullable://akankshasukre/abc-tech@sha256:743d8b5a03e4790efdf5096c1aa46dbd549d7bc78dd772d8eb7ec1
    Port:          8080/TCP
```

- Create deployment-definition.yaml file to create new deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: abc-tech-deployment
  labels:
    type: webapp
spec:
  replicas: 3
  selector:
    matchLabels:
      type: webapp
  template:
    metadata:
      labels:
        name: abc-tech
        type: webapp
    spec:
      containers:
        - name: abc-tech-app
          image: akankshasukre/abc-tech:latest
          ports:
            - containerPort: 8080
          imagePullPolicy: Always
```

- iv. Run below commands to create new deployment using this file and later view the created deployment.

```
edureka@kmaster:~/devops-project/kubernetes$ kubectl delete pod abc-tech
pod "abc-tech" deleted
edureka@kmaster:~/devops-project/kubernetes$ vi deployment-definition.yaml
edureka@kmaster:~/devops-project/kubernetes$ kubectl create -f deployment-definition.yaml
deployment.apps/abc-tech-deployment created
edureka@kmaster:~/devops-project/kubernetes$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
abc-tech-deployment   3/3     3          3          12s
```



```
edureka@kmaster:~/devops-project/kubernetes$ kubectl get all
NAME                                         READY   STATUS    RESTARTS   AGE
pod/abc-tech-deployment-7bc6c66448-24dhl   1/1    Running   0          35s
pod/abc-tech-deployment-7bc6c66448-dzt5l   1/1    Running   0          35s
pod/abc-tech-deployment-7bc6c66448-qhprp   1/1    Running   0          35s

NAME            TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1   <none>        443/TCP   27d

NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/abc-tech-deployment   3/3     3          3          35s

NAME           DESIRED  CURRENT  READY   AGE
replicaset.apps/abc-tech-deployment-7bc6c66448  3        3        3        35s
edureka@kmaster:~/devops-project/kubernetes$
```

- v. Create service-definition.yaml file to create a service in Kubernetes cluster.

```
apiVersion: v1
kind: Service
metadata:
  name: abc-tech-svc
  labels:
    type: webapp
spec:
  ports:
    - nodePort: 30500
      port: 80
      protocol: TCP
      targetPort: 8080
  selector:
    type: webapp
  type: NodePort
```

- vi. Run kubectl command to create service and get service.



```
edureka@kmaster:~/devops-project/kubernetes$ kubectl create -f service-definition.yaml
service/abc-tech-svc created
edureka@kmaster:~/devops-project/kubernetes$ kubectl get svc
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
abc-tech-svc   NodePort   10.100.51.55  <none>        80:30500/TCP   4s
kubernetes     ClusterIP  10.96.0.1    <none>        443/TCP       27d
edureka@kmaster:~/devops-project/kubernetes$ █
```

```
edureka@kmaster:~/devops-project/kubernetes$ kubectl describe service
Name:           abc-tech-svc
Namespace:      default
Labels:          type=webapp
Annotations:    <none>
Selector:       type=webapp
Type:           NodePort
IP:             10.100.51.55
Port:           <unset>  80/TCP
TargetPort:     8080/TCP
NodePort:       <unset>  30500/TCP
Endpoints:     10.244.1.4:8080,10.244.1.5:8080,10.244.1.6:8080
Session Affinity: None
External Traffic Policy: Cluster
Events:         <none>

Name:           kubernetes
Namespace:      default
Labels:          component=apiserver
                 provider=kubernetes
Annotations:    <none>
Selector:       <none>
Type:           ClusterIP
IP:             10.96.0.1
Port:           https  443/TCP
TargetPort:     6443/TCP
```

vii. Verify the deployment by hitting the URL as seen below:

<http://172.31.18.87:30500/ABCtechnologies-1.0>



- 4) Integrate Ansible with Kubernetes. Create CI/CD Pipeline to deploy artifacts on Kubernetes using Ansible playbook.

- i. Create hosts file in Ansible with Kubernetes master and worker node details.

```
[masters]
master ansible_host=172.31.9.156 ansible_user=edureka ansible_connection=ssh ansible_python_interpreter=/usr/bin/python
[workers]
worker1 ansible_host=172.31.18.87 ansible_ssh_user=edureka ansible_connection=ssh ansible_python_interpreter=/usr/bin/python
```

- ii. Create Ansible playbook main.yaml to create deployment and service in Kubernetes as below:

```

- name: Create deployment and service in kubernetes
  hosts: master
  become: true
  user: edureka
  tasks:
    - name: Create deployment
      command: kubectl apply -f deployment-definition.yaml --kubeconfig=/home/edureka/.kube/admin.conf
      args:
        chdir: /home/edureka/devops-project/kubernetes

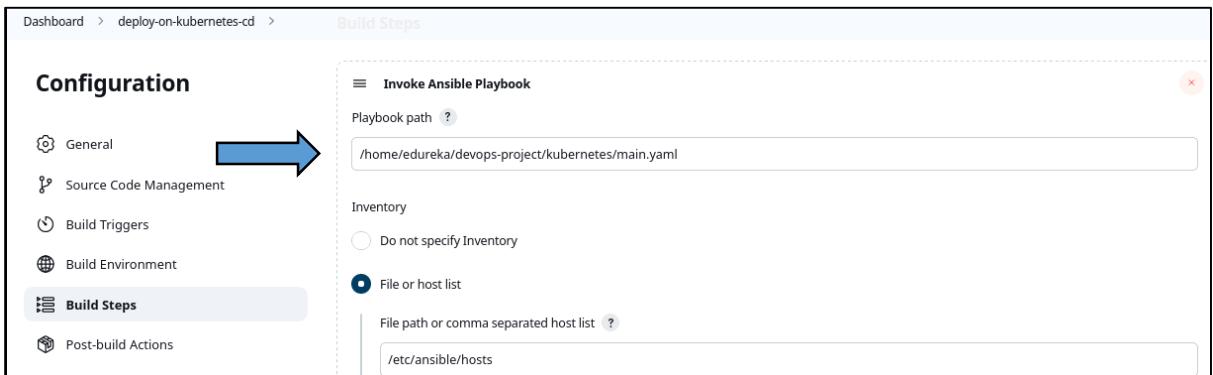
    - name: Update deployment with new pods if docker hub image is updated
      command: kubectl rollout restart deployment abc-tech-deployment --kubeconfig=/home/edureka/.kube/admin.conf

    - name: Create webapp service
      command: kubectl apply -f service-definition.yaml --kubeconfig=/home/edureka/.kube/admin.conf
      args:
        chdir: /home/edureka/devops-project/kubernetes
  
```

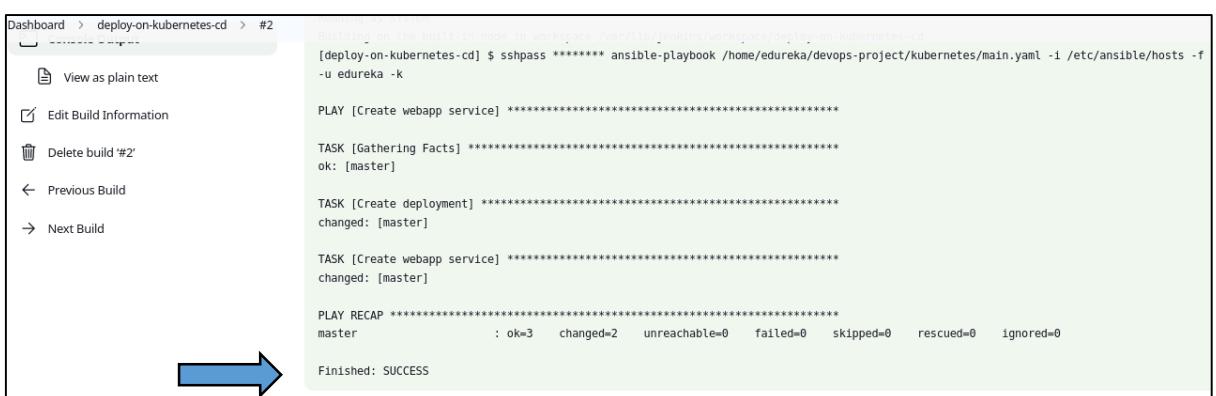
```

edureka@kmaster:~/devops-project/kubernetes$ ls -l
total 16
-rw-r--r-- 1 edureka edureka 399 Feb 11 14:19 deployment-definition.yaml
-rw-r--r-- 1 root     root    494 Feb 13 12:23 main.yaml
-rw-r--r-- 1 edureka edureka 201 Feb 11 14:11 pod-definition.yaml
-rw-r--r-- 1 edureka edureka 221 Feb 11 14:21 service-definition.yaml
edureka@kmaster:~/devops-project/kubernetes$ 
  
```

- iii. Create a new Jenkins job: **deploy-on-kubernetes-cd**. Add build step to invoke Ansible playbook and give the path to main.yaml file in Jenkins master.



- iv. Run the job and build is successful as seen below.



v. Verify the deployment using kubectl command as shown below:

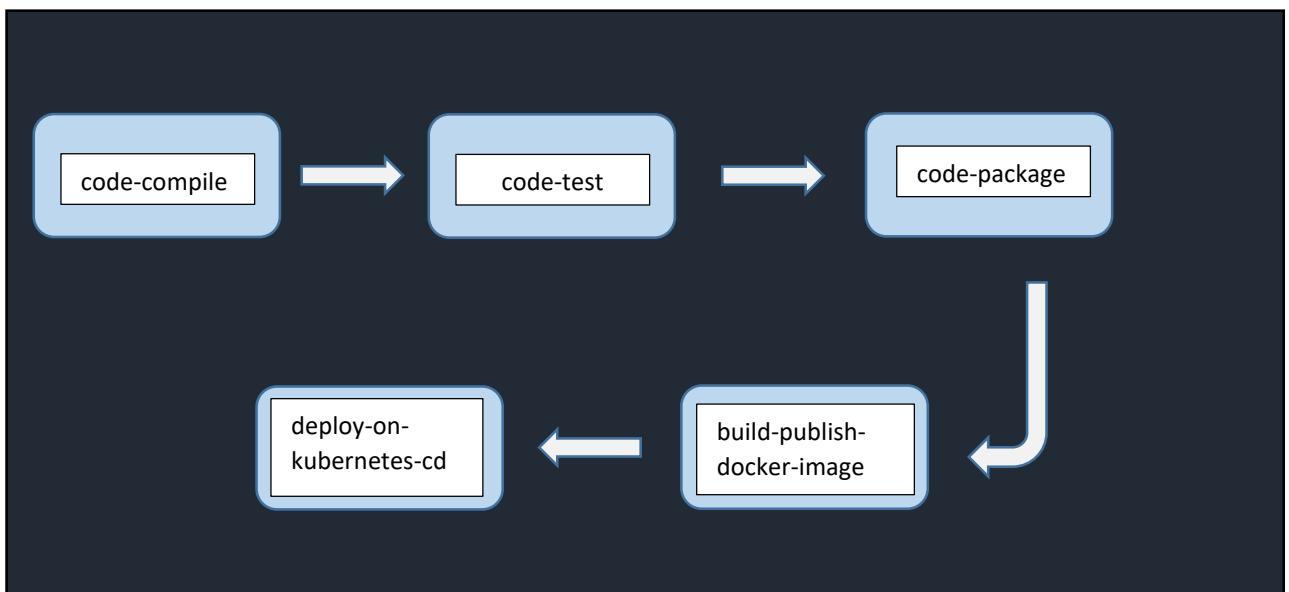
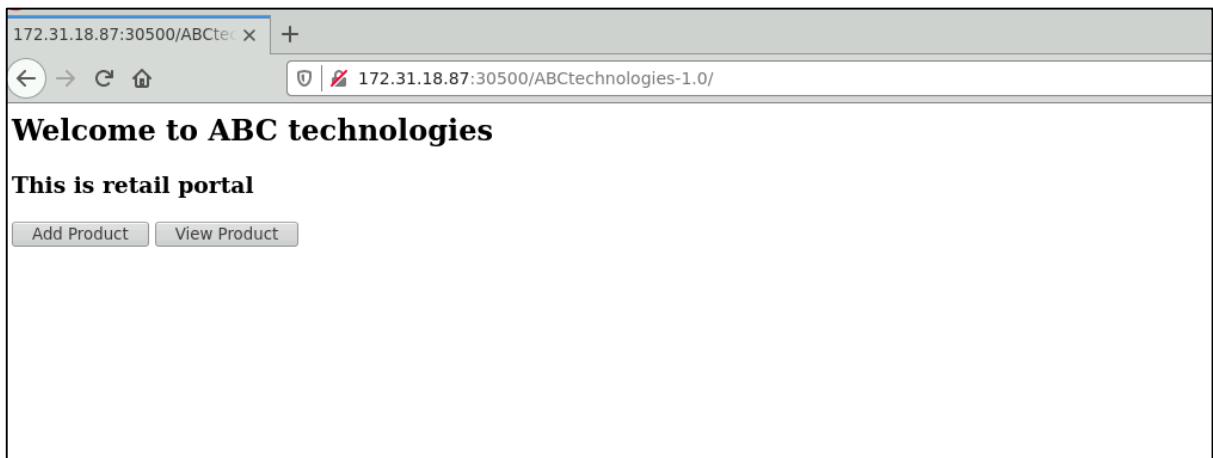
```
edureka@kmaster:~/devops-project/kubernetes$ kubectl get all
NAME                                         READY   STATUS    RESTARTS   AGE
pod/abc-tech-deployment-7bc6c66448-6967w   1/1    Running   0          5m18s
pod/abc-tech-deployment-7bc6c66448-jkpvk   1/1    Running   0          5m18s
pod/abc-tech-deployment-7bc6c66448-n5znj   1/1    Running   0          5m18s

NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/abc-tech-svc  NodePort  10.102.138.180 <none>        80:30500/TCP  5m17s
service/kubernetes ClusterIP  10.96.0.1     <none>        443/TCP     29d

NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/abc-tech-deployment   3/3     3           3          5m18s

NAME                DESIRED   CURRENT   READY   AGE
replicaset.apps/abc-tech-deployment-7bc6c66448  3         3         3       5m18s
edureka@kmaster:~/devops-project/kubernetes$
```

vi. Verify that the application is running on desired port.



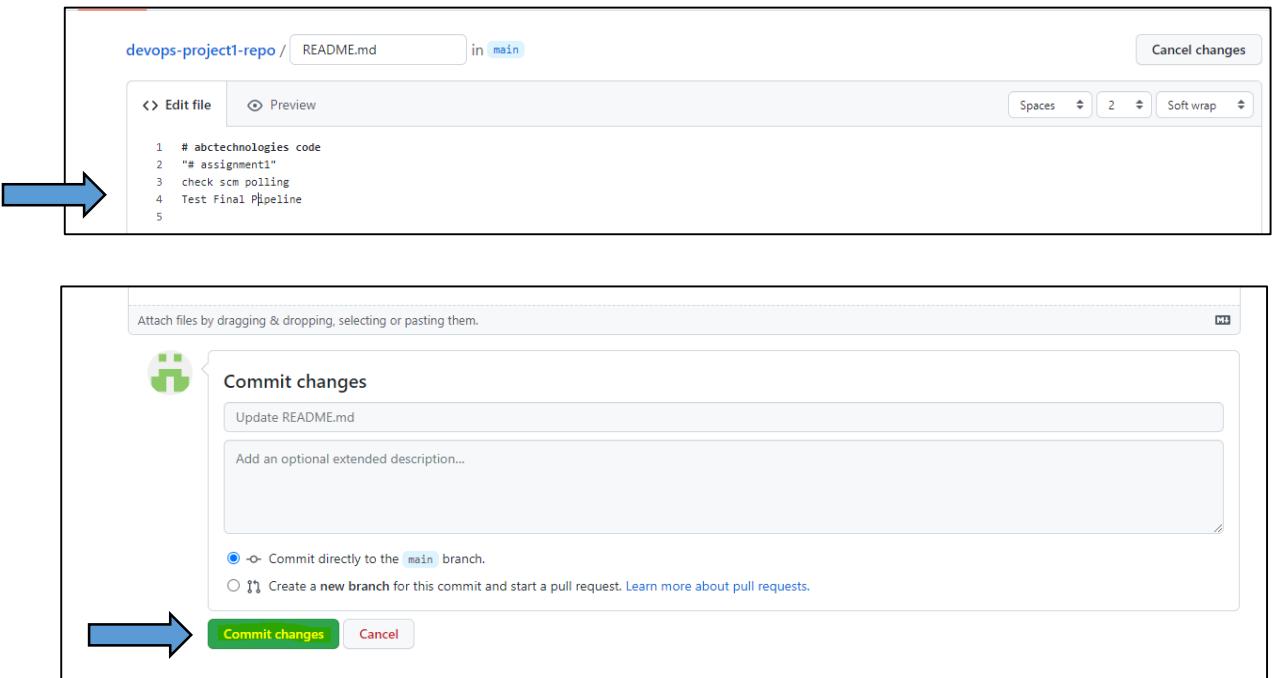
END TO END PIPELINE FLOW

## 5) Create end to end pipeline: Build and Publish docker image & Deploy to Kubernetes using Ansible

- Add a post-build action in Package job to run the job: build-publish-docker-image
- Create a new job: **build-publish-docker-image** in Jenkins and add build step to copy artifacts from Package job and then to invoke following Ansible playbook:

```
- hosts: localhost
  vars:
    ansible_python_interpreter: /usr/bin/python
    docker_pwd: !vault |
      $ANSIBLE_VAULT;1.1;AES256
      63366261666166616232316433636237643264336163633337376531386535316566663130656330
      3239343738386138643738366132393232336330646431380a666634373166306237356138666334
      38613634353761663435326663323864613461636635643632373965646230353830373064636130
      6263346333306438330a336436626462303166396137306136353732643839353065386564386237
      3838
  become_method: sudo
  tasks:
    - name: Log into DockerHub
      docker_login:
        username: akankshasukre
        password: '{{ docker_pwd }}'
    - name: Build Docker image from Dockerfile
      docker_image:
        name: abc-tech
        state: present
        source: build
        build:
          path: .
        push: yes
        tag: latest
        repository: akankshasukre/abc-tech
```

- Add a post-build action in this job to build the kubernetes job i.e. **deploy-on-kubernetes-cd**
- Test the final pipeline by committing a new change in GIT repo



The screenshot shows a GitHub commit interface. At the top, it says "devops-project1-repo / README.md in main". Below that, there's an "Edit file" button and a "Preview" button. The code editor shows the following content:

```
1 # abctechnologies code
2 "# assignment1"
3 check scm polling
4 Test Final Pipeline
5
```

A blue arrow points to the "Commit changes" button at the bottom of the interface.

v. Check the Git Polling Log of Compile job

Dashboard > code-compile > Git Polling Log

### Git Polling Log

- [Back to Dashboard](#)
- [Status](#)
- [Changes](#)
- [Workspace](#)
- [Build Now](#)
- [Configure](#)
- [Delete Project](#)
- [Git Polling Log](#)

```

Started on Feb 13, 2023, 3:38:00 PM
Using strategy: Default
[poll] Last Built Revision: Revision 29f3c91fcfad24254014797373dbfee75c3c6953 (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
> git --version # timeout=10
> git --version # 'git version 2.17.1'
> git ls-remote -h -- https://github.com/AkankshaSukre/devops-project1-repo.git # timeout=10
Found 1 remote heads on https://github.com/AkankshaSukre/devops-project1-repo.git
[poll] Latest remote head revision on refs/heads/main is: cec6f0fcf910193dbe7cf2c76ece6c3e43856e9
Done. Took 0.4 sec
Changes found
  
```

vi. Compile job is triggered which further triggers the test job

Dashboard > Build History

S	W	Name	Last Success	Last Failure	Last Duration
		build-publish-docker-image	38 min #3	52 min #1	26 sec
		code-compile	20 sec #11	10 days #4	6 sec
		code-package	38 min #11	9 days 19 hr #7	10 sec
		code-test	39 min #7	10 days #1	12 sec
		deploy-on-kubernetes-cd	6 min 55 sec #8	8 min 47 sec #7	7.3 sec
		docker-ansible-cicd	1 hr 17 min #9	4 days 21 hr #5	25 sec

vii. After the test job, package job runs and then triggers the build-publish-docker-image job.

Dashboard > Build History

S	W	Name	Last Success	Last Failure	Last Duration
		build-publish-docker-image	39 min #3	53 min #1	26 sec
		code-compile	58 sec #11	10 days #4	6 sec
		code-package	28 sec #12	9 days 19 hr #7	9.3 sec
		code-test	43 sec #8	10 days #1	6.8 sec
		deploy-on-kubernetes-cd	7 min 33 sec #8	9 min 24 sec #7	7.3 sec
		docker-ansible-cicd	1 hr 17 min #9	4 days 21 hr #5	25 sec
		docker-cicd	6 days 0 hr #35	6 days 0 hr #33	17 sec

viii. Finally the last job(deploy-on-kubernetes-cd) is triggered which completes the pipeline

Dashboard > Build History

S	W	Name	Last Success	Last Failure	Last Duration
		build-publish-docker-image	1 min 13 sec #4	54 min #1	25 sec
		code-compile	1 min 58 sec #11	10 days #4	6 sec
		code-package	1 min 28 sec #12	9 days 19 hr #7	9.3 sec
		code-test	1 min 43 sec #8	10 days #1	6.8 sec
		deploy-on-kubernetes-cd	38 sec #9	10 min #7	6.8 sec

```

deploy-on-kubernetes-cd x +
localhost:8080/job/deploy-on-kubernetes-cd/9/console
Dashboard > deploy-on-kubernetes-cd > #9

ok: [master]

TASK [Create deployment] *****
changed: [master]

TASK [Update deployment with new pods if docker hub image is updated] *****
changed: [master]

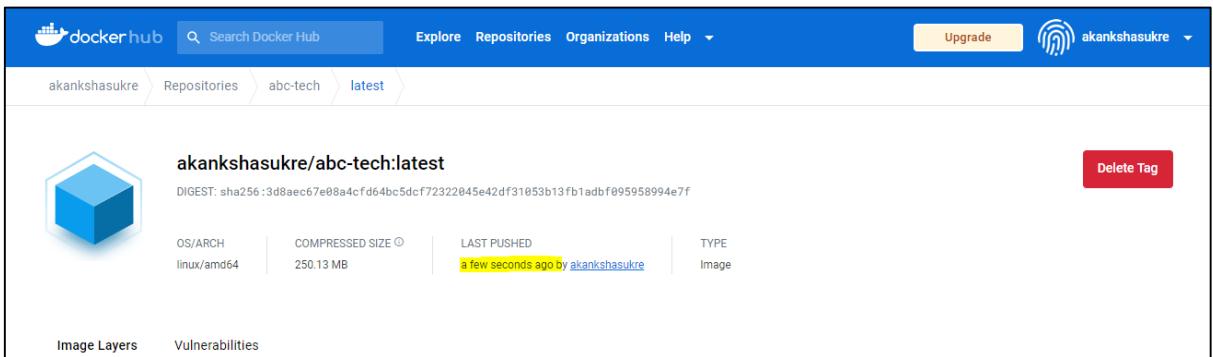
TASK [Create webapp service] *****
changed: [master]

PLAY RECAP *****
master : ok=4    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

Finished: SUCCESS

```

- ix. Verify the deployment. Latest image is pushed to docker hub repository.



- x. The application is running successfully on the given port as expected. Thus, the deployment is complete.

```

edureka@kmaster:~/devops-project/kubernetes$ kubectl get all
NAME                           READY   STATUS    RESTARTS   AGE
pod/abc-tech-deployment-75bcb44865-84fv9   1/1     Running   0          95s
pod/abc-tech-deployment-75bcb44865-87csc   1/1     Running   0          100s
pod/abc-tech-deployment-75bcb44865-d8rzj   1/1     Running   0          107s

NAME                  TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/abc-tech-svc NodePort    10.102.138.180  <none>           80:30500/TCP   3h4m
service/kubernetes   ClusterIP  10.96.0.1      <none>           443/TCP      29d

NAME                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/abc-tech-deployment   3/3     3           3           3h4m

NAME                           DESIRED   CURRENT   READY   AGE
replicaset.apps/abc-tech-deployment-75bcb44865   3        3        3       107s
replicaset.apps/abc-tech-deployment-7bc6c66448   0        0        0       3h4m
replicaset.apps/abc-tech-deployment-95d8d6d8b   0        0        0       9m42s

```



## Task 5: Monitoring the Kubernetes Cluster using Prometheus and creating dashboard in Grafana to view important metrics.

### 1) Install Node Exporter on master and slave node

#### i. Download node exporter on master node

```
edureka@kmaster:~$ wget https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz
--2023-02-13 15:54:22-- https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-production-release-asset-2e65be/9524057/7c60f6f9-7b41-446c-be81-a6c24a9d0383?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVVEH53A%2F20230213%2Fus-east-1%2F53%2Faws4_request&X-Amz-Date=20230213T155227Z&X-Amz-Expires=300&X-Amz-Signature=4637b6fe8490bed10bc4acc8ce42db71d71fea20642a04d030dfd881eec6432&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=9524057&response-content-disposition=attachment%3Bfilename%3Dnode_exporter-1.3.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream [following]
--2023-02-13 15:54:22-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/9524057/7c60f6f9-7b41-446c-be81-a6c24a9d0383?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVVEH53A%2F20230213%2Fus-east-1%2F53%2Faws4_request&X-Amz-Date=20230213T155227Z&X-Amz-Expires=300&X-Amz-Signature=4637b6fe8490bed10bc4acc8ce42db71d71fea20642a04d030dfd881eec6432&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=9524057&response-content-disposition=attachment%3Bfilename%3Dnode_exporter-1.3.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.110.133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9033415 (8.6M) [application/octet-stream]
Saving to: 'node_exporter-1.3.1.linux-amd64.tar.gz'

node_exporter-1.3.1.linux-amd64.tar.g 100%[=====] 8.61M --.-KB/s in 0.05s
2023-02-13 15:54:22 (168 MB/s) - 'node_exporter-1.3.1.linux-amd64.tar.gz' saved [9033415/9033415]
```

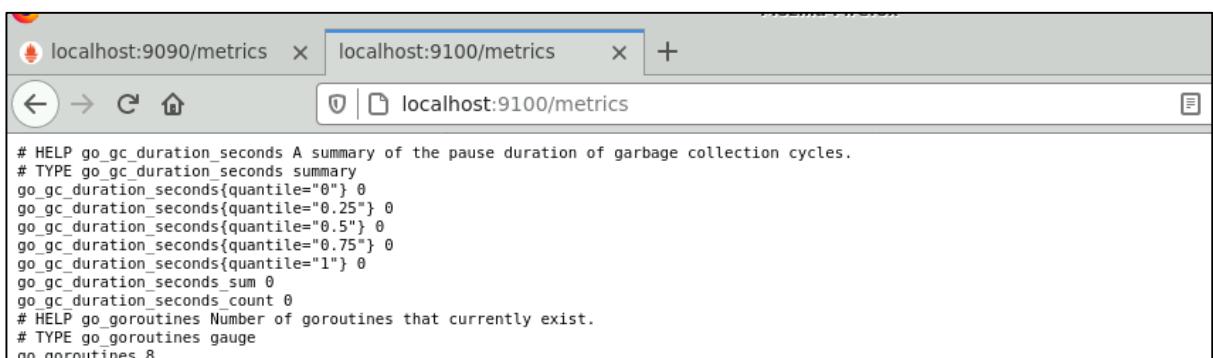
#### ii. Extract the node-exporter zip

```
edureka@kmaster:~$ tar xvfz node_exporter-*.*-amd64.tar.gz
node_exporter-1.3.1.linux-amd64/
node_exporter-1.3.1.linux-amd64/LICENSE
node_exporter-1.3.1.linux-amd64/NOTICE
node_exporter-1.3.1.linux-amd64/node_exporter
edureka@kmaster:~$
```

#### iii. Run node-exporter on master and verify functioning

```
edureka@kmaster:~$ cd node_exporter-1.3.1.linux-amd64/
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ ls -l
total 17820
-rw-r--r-- 1 edureka edureka 11357 Dec  5 2021 LICENSE
-rw-r--r-- 1 edureka edureka   463 Dec  5 2021 NOTICE
-rwxr-xr-x 1 edureka edureka 18228926 Dec  5 2021 node_exporter
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ ./node_exporter
ts=2023-02-13T15:55:28.967Z caller=node_exporter.go:182 level=info msg="Starting node_exporter" version="(version=1.3.1, branch=HEAD, revision=a2321e7b940ddcff26873612bccdf7cd4c42b666)"
ts=2023-02-13T15:55:28.967Z caller=node_exporter.go:183 level=info msg="Build context" build_context="(go=golang 1.17.3, user=root@243aaafa5525c, date=2021-12-05 11:09:49)"
ts=2023-02-13T15:55:28.969Z caller=filesystem common.go:111 level=info collector=filesystem msg="Parsed flag --collector.filesystem.mount-points-exclude" flag=~"/(dev|proc|run|credentials|.+|sys|var|lib|docker/.+)(|$)"
ts=2023-02-13T15:55:28.969Z caller=filesystem_common.go:113 level=info collector=filesystem msg="Parsed flag --collector.filesystem.fs-types-exclude" flag=(aufofs|binfmt_misc|bpf|cgroup2|configs|debugfs|devpts|devtmpfs|fusectl|hugetlbfs|iso9660|mqueue|nsfs|overlay|proc|procfs|ptstore|rpc_pipes|securityfs|selinuxfs|squashfs|sysfs|tracefs)$
ts=2023-02-13T15:55:28.969Z caller=node_exporter.go:108 level=info msg="Enabled collectors"
```

#### iv. Hit the URL to view node-exporter metrics



## v. Install node-exporter on slave node as well

```
dureka@kslave1:~$ wget https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz
--2023-02-13 16:30:26-- https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/9524057/7c60f6f9-7b41-446c-be81-a6c24a9d0383?X-Amz-Content-Type=application/x-tar&X-Amz-Content-Disposition=attachment;filename=node_exporter-1.3.1.linux-amd64.tar.gz&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=9524057&response-content-type=application/x-tar
--2023-02-13 16:30:26-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/9524057/7c60f6f9-7b41-446c-be81-a6c24a9d0383?X-Amz-Content-Type=application/x-tar&X-Amz-Content-Disposition=attachment;filename=node_exporter-1.3.1.linux-amd64.tar.gz&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=9524057&response-content-type=application/x-tar
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9033415 (8.6M) [application/octet-stream]
Saving to: 'node_exporter-1.3.1.linux-amd64.tar.gz'

node_exporter-1.3.1 100%[=====] 8.61M 5.36MB/s in 1.6s

2023-02-13 16:30:28 (5.36 MB/s) - 'node_exporter-1.3.1.linux-amd64.tar.gz' saved [9033415/9033415]

dureka@kslave1:~$
```

```
edureka@kslave1:~$ tar xvf node_exporter-*.*-amd64.tar.gz
node_exporter-1.3.1.linux-amd64/
node_exporter-1.3.1.linux-amd64/LICENSE
node_exporter-1.3.1.linux-amd64/NOTICE
node_exporter-1.3.1.linux-amd64/node_exporter
edureka@kslave1:~$ edureka@kslave1:~$ cd node_exporter-1.3.1.linux-amd64/
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ ./node_exporter
ts=2023-02-13T16:30:51.793Z caller=node_exporter.go:182 level=info msg="Starting node_exporter" version="(version=1.3.1, branch=7b40ddcf26873612bccdf/cdc42b6b6)"
ts=2023-02-13T16:30:51.793Z caller=node_exporter.go:183 level=info msg="Build context" build_context="(go=golang1.17.3, user=root@2025-11-09:49)"
ts=2023-02-13T16:30:51.793Z caller=filesystem_common.go:111 level=info collector=filesystem msg="Parsed flag --collector.filesystem"
flag=/dev/proc/run/credentials/+|sys|var/lib/docker/+($|/)
ts=2023-02-13T16:30:51.794Z caller=filesystem_common.go:113 level=info collector=filesystem msg="Parsed flag --collector.filesystem"
flag=(autofs|binfmt_misc|bpf|cgroup2|configfs|debugfs|devpts|devtmpfs|fusectl|hugetlbfs|iso9660|mqueue|nsfs|overlay|proc|proc|securityfs|selinuxfs|squashfs|sysfs|tracefs)$
```

## vi. Install node exporter as a service on master and slave nodes

```
-rwxr-xr-x 1 edureka edureka 18228926 Dec 5 2021 node_exporter
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo cp node_exporter /usr/local/bin
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo useradd --no-create-home --shell /bin/false node_exporter
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo chown node_exporter:node_exporter /usr/local/bin/node_exporter
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo nano /etc/systemd/system/node_exporter.service
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl daemon-reload
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl enable node_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/node_exporter.service → /etc/systemd/system/node_exporter.service.
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl start node_exporter.service
edureka@kmaster:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2023-02-14 17:08:33 UTC; 4s ago
     Main PID: 9449 (node_exporter)
        Tasks: 4 (limit: 4915)
       CGroup: /system.slice/node_exporter.service
               └─9449 /usr/local/bin/node_exporter
Feb 14 17:08:33 kmaster node_exporter[9449]: ts=2023-02-14T17:08:33.692Z caller=node_exporter.go:115 level=info collector=thermal_zone
Feb 14 17:08:33 kmaster node_exporter[9449]: ts=2023-02-14T17:08:33.692Z caller=node_exporter.go:115 level=info collector=time
Feb 14 17:08:33 kmaster node_exporter[9449]: ts=2023-02-14T17:08:33.692Z caller=node_exporter.go:115 level=info collector=timex
Feb 14 17:08:33 kmaster node_exporter[9449]: ts=2023-02-14T17:08:33.692Z caller=node_exporter.go:115 level=info collector=udp_queues
Feb 14 17:08:33 kmaster node_exporter[9449]: ts=2023-02-14T17:08:33.692Z caller=node_exporter.go:115 level=info collector=uname
```

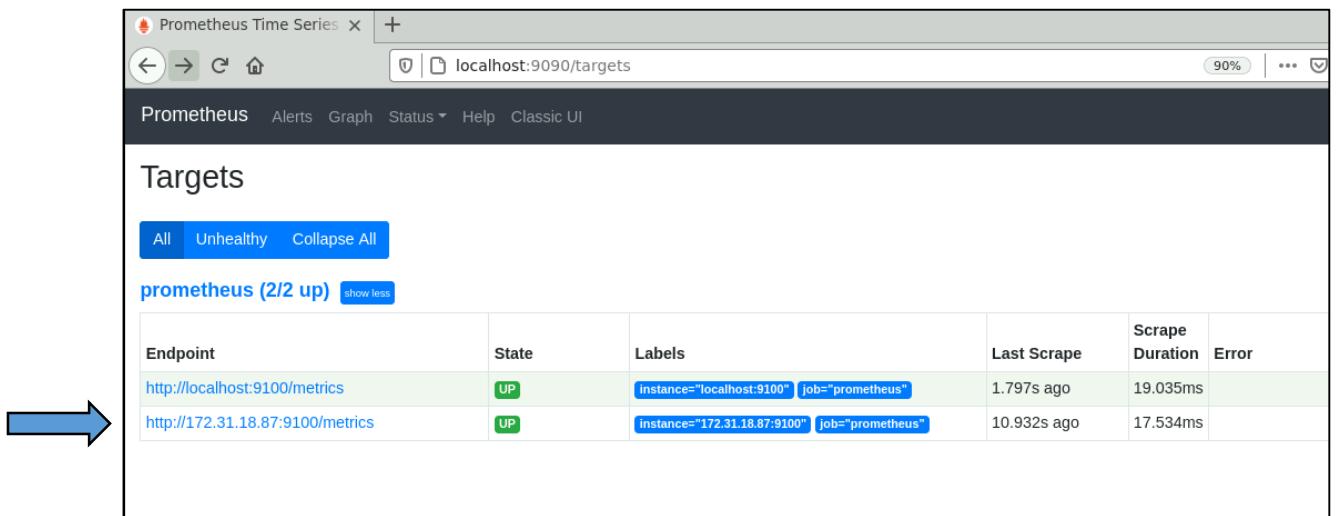
```
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo cp node_exporter /usr/local/bin
[sudo] password for edureka:
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo cp node_exporter /usr/local/bin
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo useradd --no-create-home --shell /bin/false node_exporter
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo chown node_exporter:node_exporter /usr/local/bin/node_exporter
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo nano /etc/systemd/system/node_exporter.service
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl daemon-reload
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl enable node_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/node_exporter.service → /etc/systemd/system/node_exporter.service.
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl start node_exporter
edureka@kslave1:~/node_exporter-1.3.1.linux-amd64$ sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: pr
   Active: active (running) since Tue 2023-02-14 17:10:53 UTC; 9s ago
     Main PID: 7715 (node_exporter)
        Tasks: 4 (limit: 4627)
       CGroup: /system.slice/node_exporter.service
               └─7715 /usr/local/bin/node_exporter
```

## 2) Add Node URL to target in prometheus.yml

### i. Add slave node URL to Prometheus targets

```
# A scrape configuration containing exactly one endpoint to scrape:  
# Here it's Prometheus itself.  
scrape_configs:  
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.  
  - job_name: 'prometheus'  
  
  # metrics_path defaults to '/metrics'  
  # scheme defaults to 'http'.  
  
  static_configs:  
    - targets: ['localhost:9100', '172.31.18.87:9100']
```

### ii. Restart the Prometheus service and validate its working fine.



Prometheus Time Series x +

localhost:9090/targets 90% ...

Prometheus Alerts Graph Status Help Classic UI

## Targets

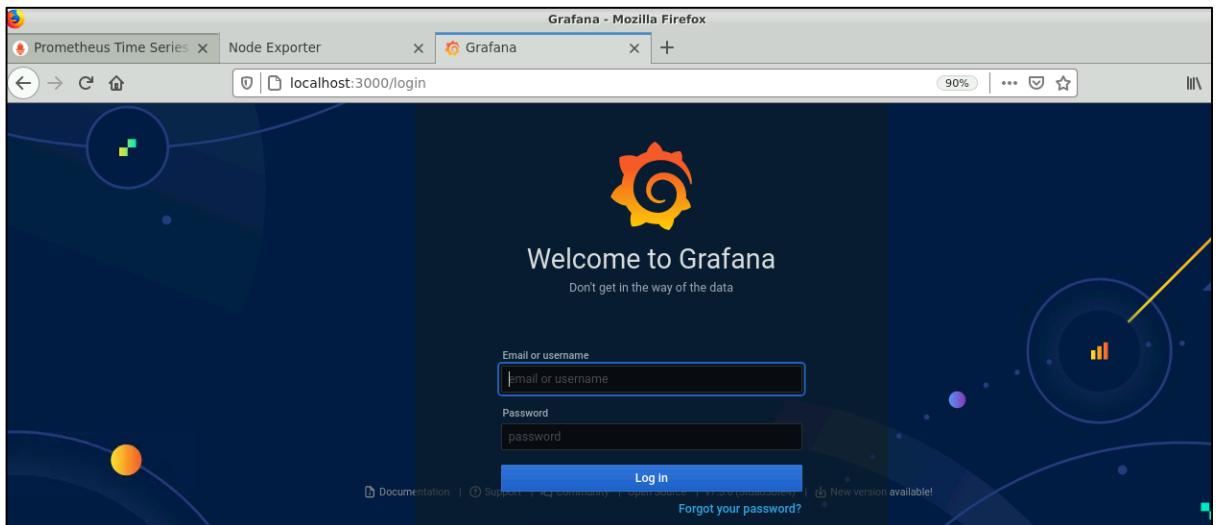
All Unhealthy Collapse All

**prometheus (2/2 up)** show less

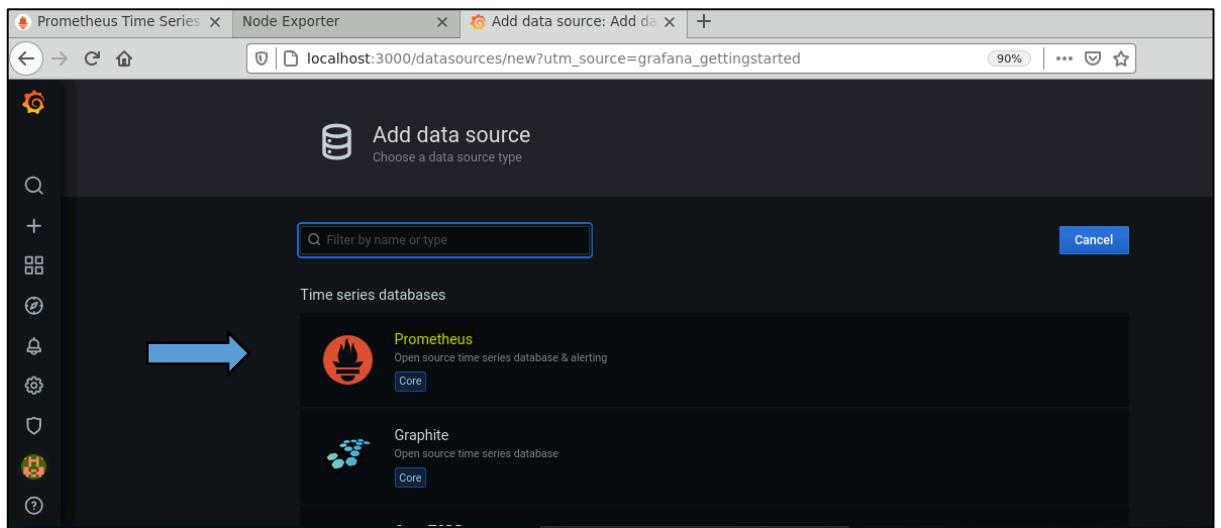
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9100/metrics	UP	instance="localhost:9100" job="prometheus"	1.797s ago	19.035ms	
http://172.31.18.87:9100/metrics	UP	instance="172.31.18.87:9100" job="prometheus"	10.932s ago	17.534ms	

## 3) Create Dashboard in Grafana

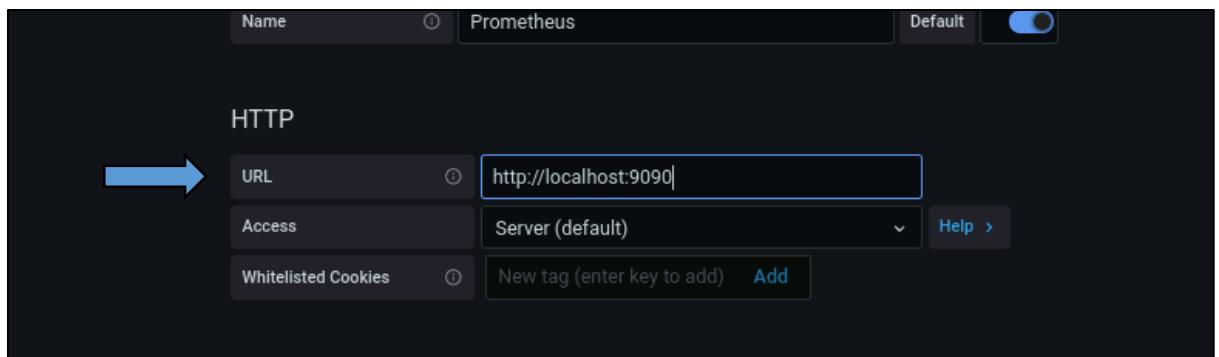
### i. Access the Grafana URL on master node: <http://localhost:3000/login>



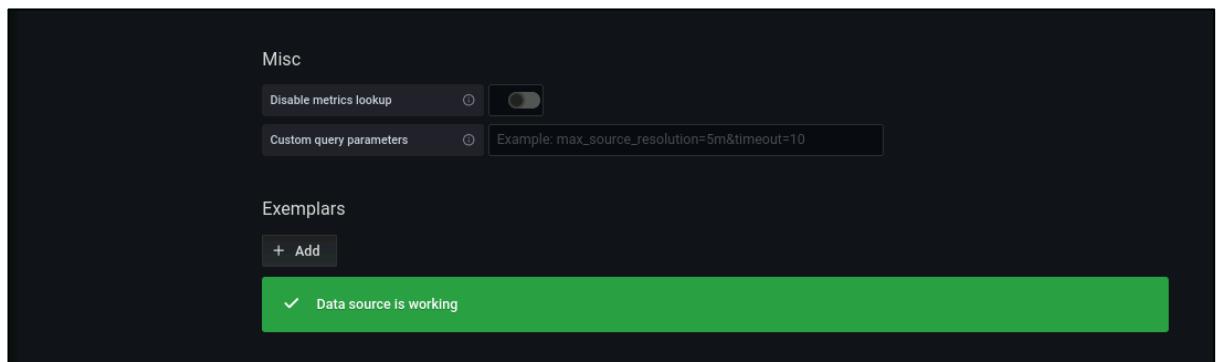
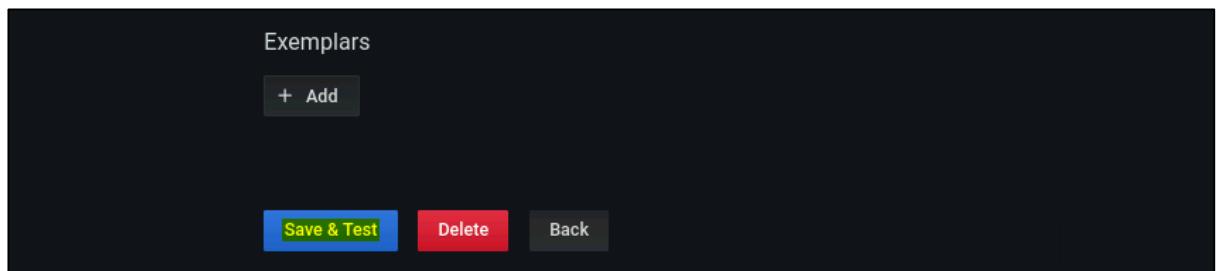
ii. Add a data source as Prometheus



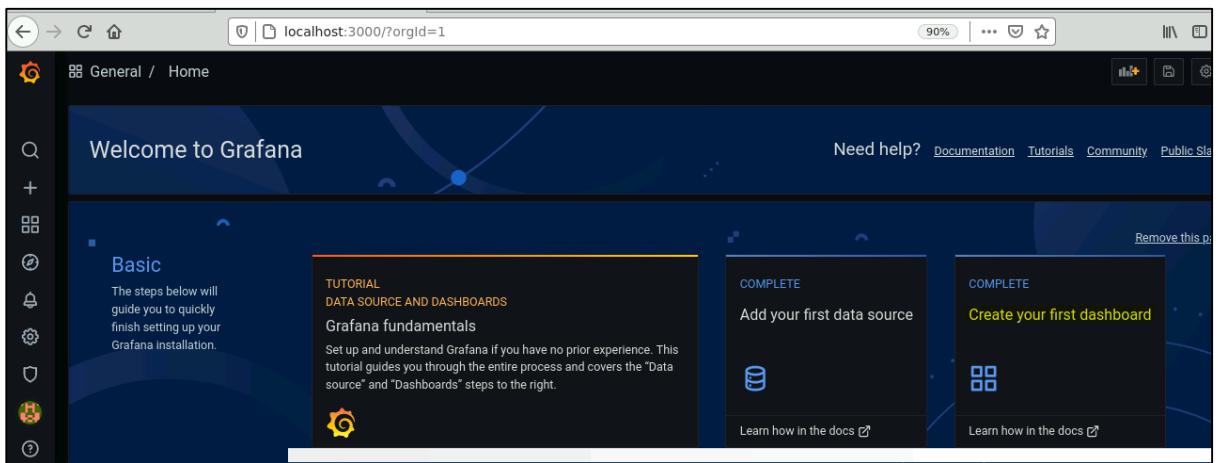
iii. Add Prometheus URL in data source configuration



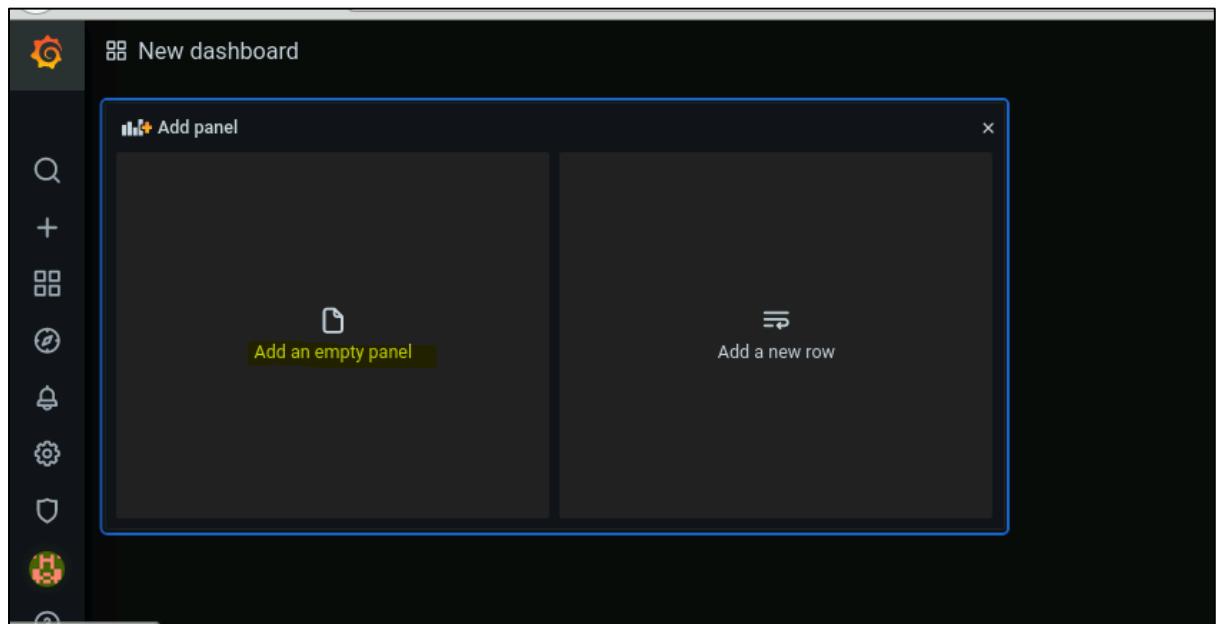
iv. Click on Save & Test



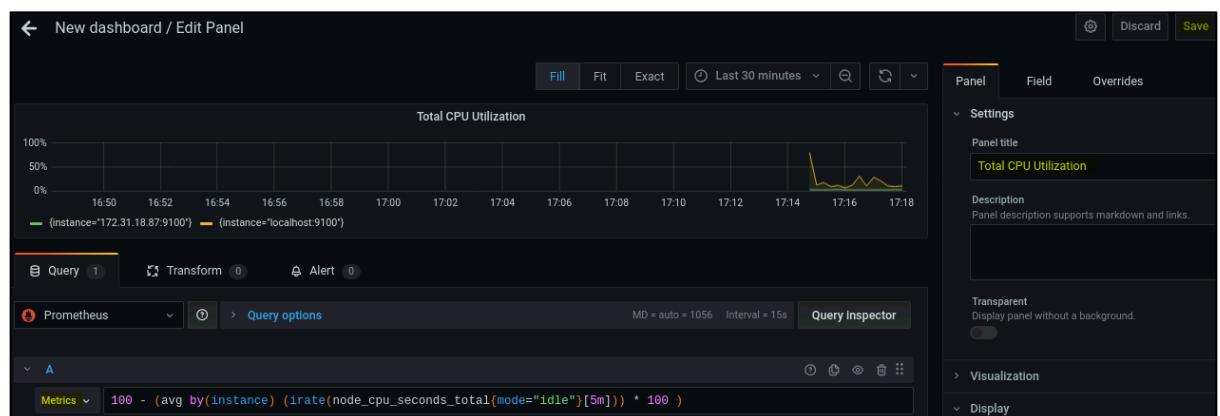
v. Create a new dashboard



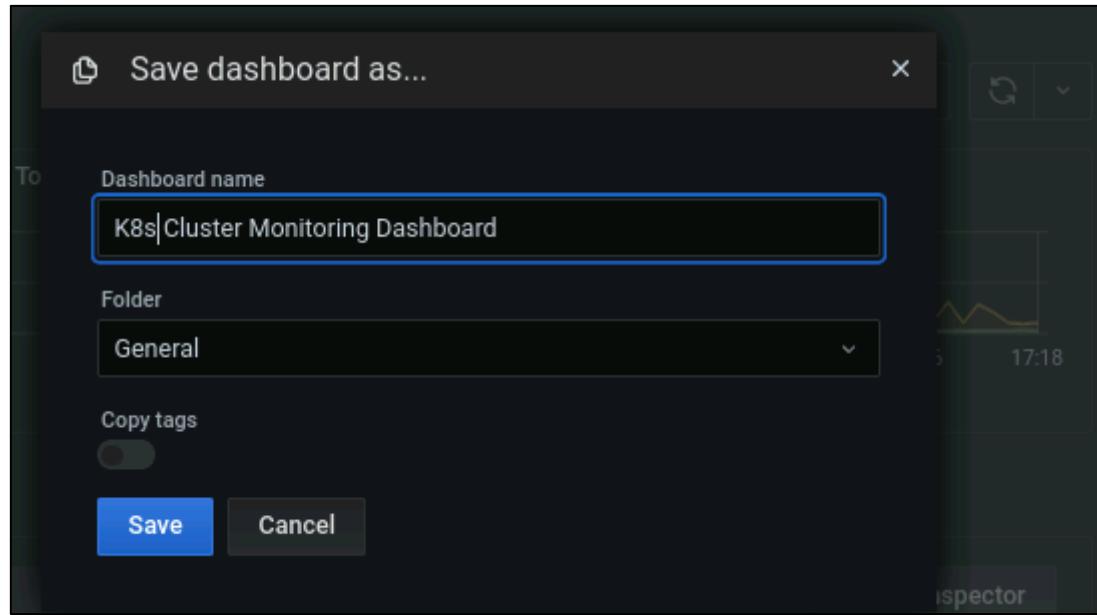
vi. Click on Add an empty panel



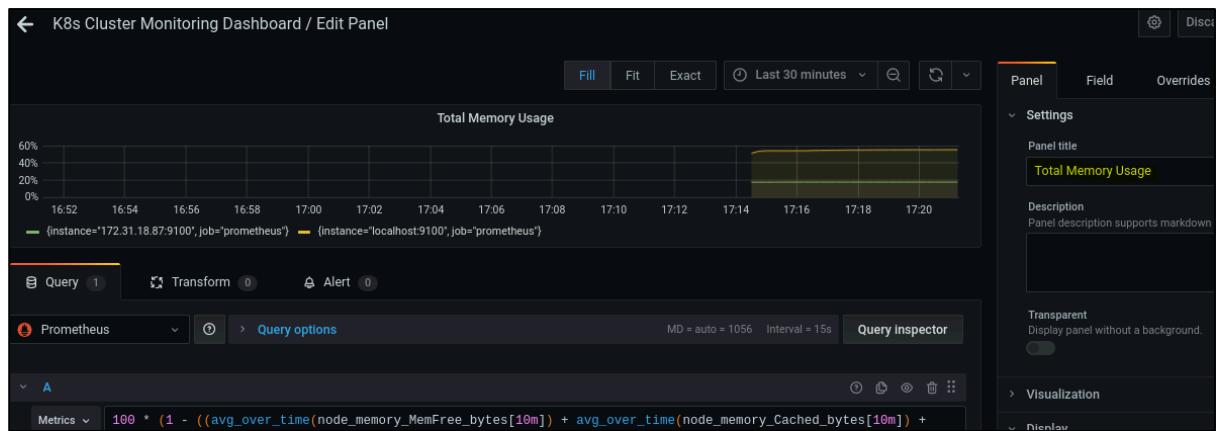
vii. Add metric for CPU utilization



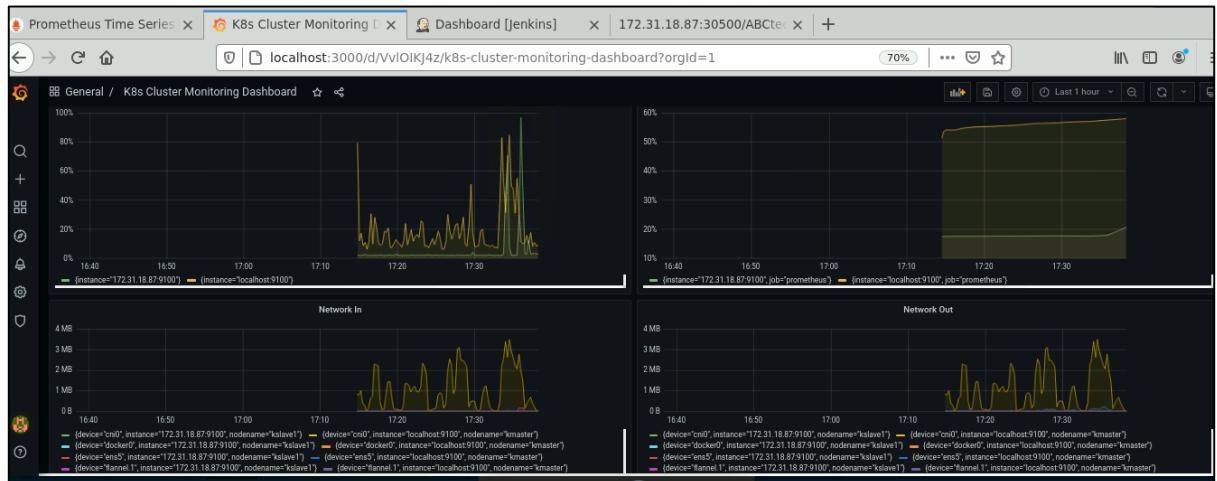
viii. Save the dashboard by giving it a name



ix. Similarly, add more panels and respective metrics to complete the dashboard.



x. Final Grafana Dashboard



### **Important URLs:**

- 1) GitHub Repo: AkankshaSukre/devops-project1-repo
- 2) Jenkins: http://localhost:8080/jenkins/jenkins
- 3) Dockerhub Repo: akankshasukre/abc-tech
- 4) Prometheus: http://localhost:9090
- 5) Node Exporter: http://localhost:9100
- 6) Grafana: http://localhost:3000
- 7) URL to access the deployed application: http://172.31.18.87:30500/ABCtechnologies-1.0/

### **Cloud Lab Details:**

<b>Master public IP.</b>	43.204.221.243
<b>Slave public IP.</b>	43.205.93.207
<b>Master private IP.</b>	172.31.9.156
<b>Slave private IP.</b>	172.31.18.87

Also, submitting the files mentioned in this solution document which are named as follows:

- 1) Dockerfile
- 2) main.yaml (docker)
- 3) pod-definition.yaml
- 4) deployment-definition.yaml
- 5) service-definition.yaml
- 6) build-publish.yaml
- 7) main.yaml (kubernetes)