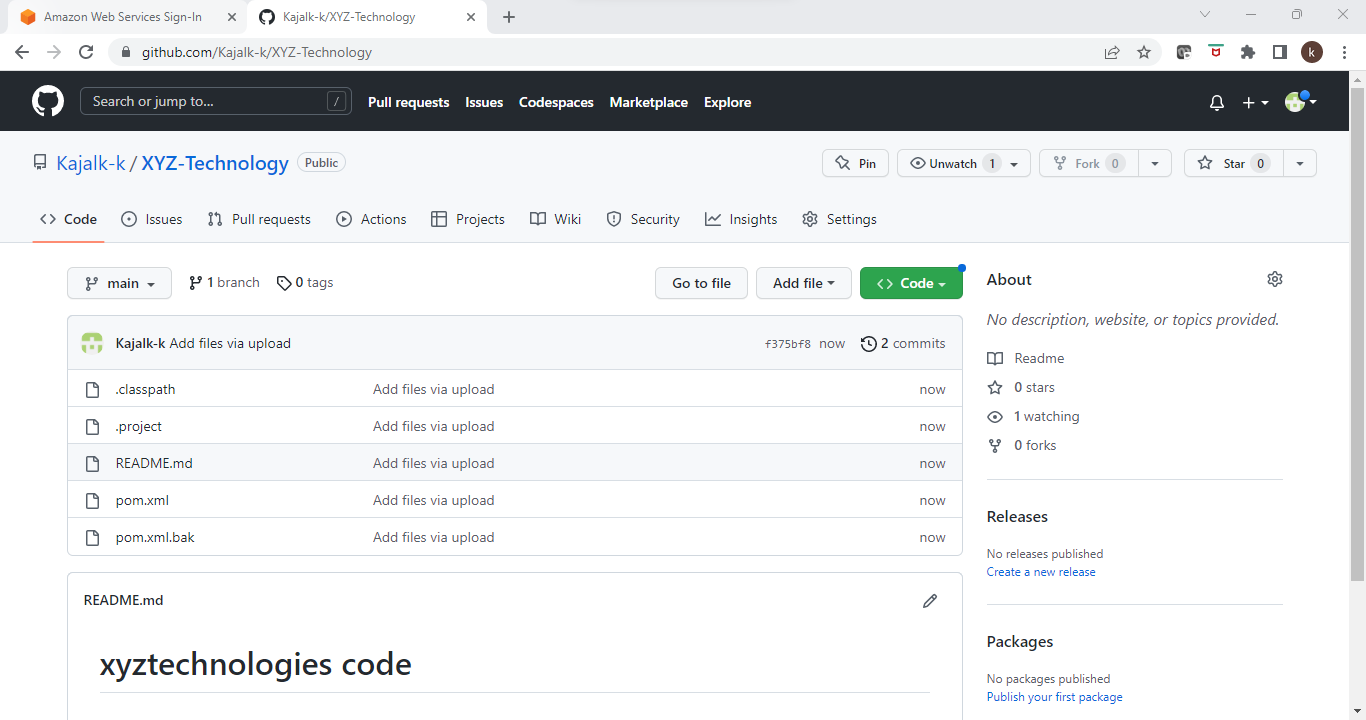
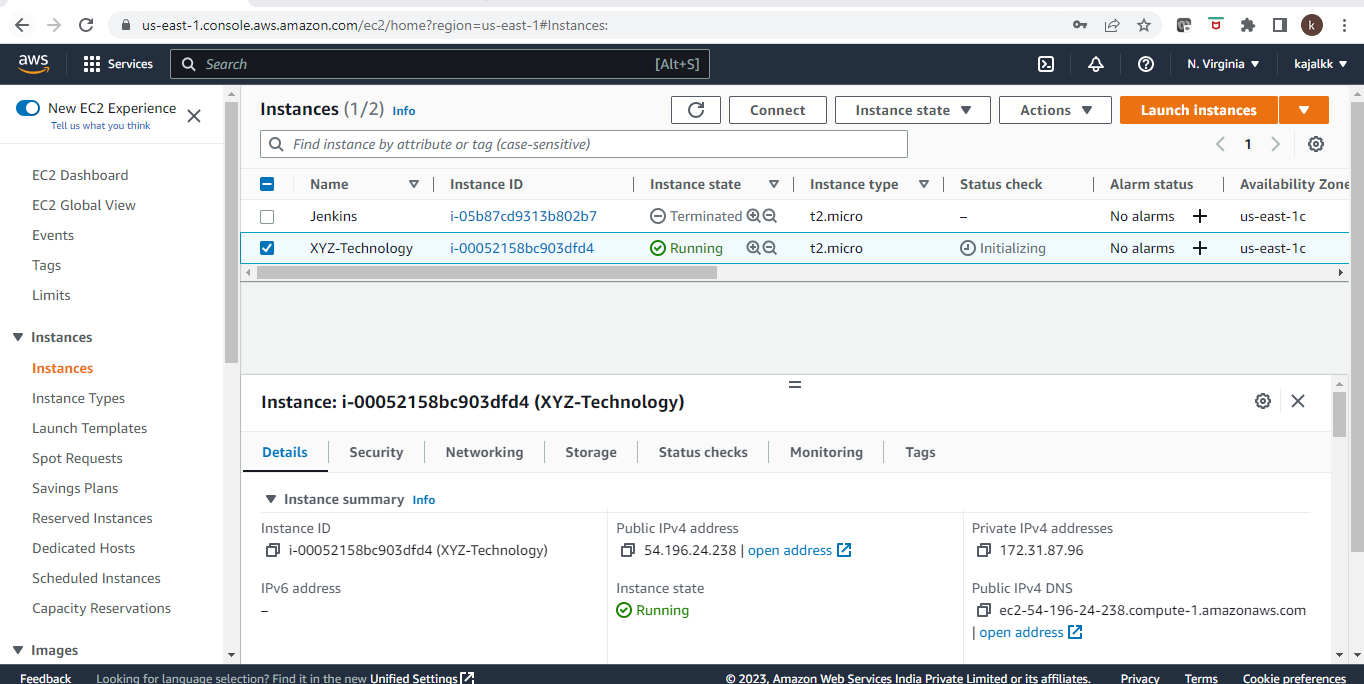
**Industry Grade Project 2 : XYZ Technology**

**Task 1:**

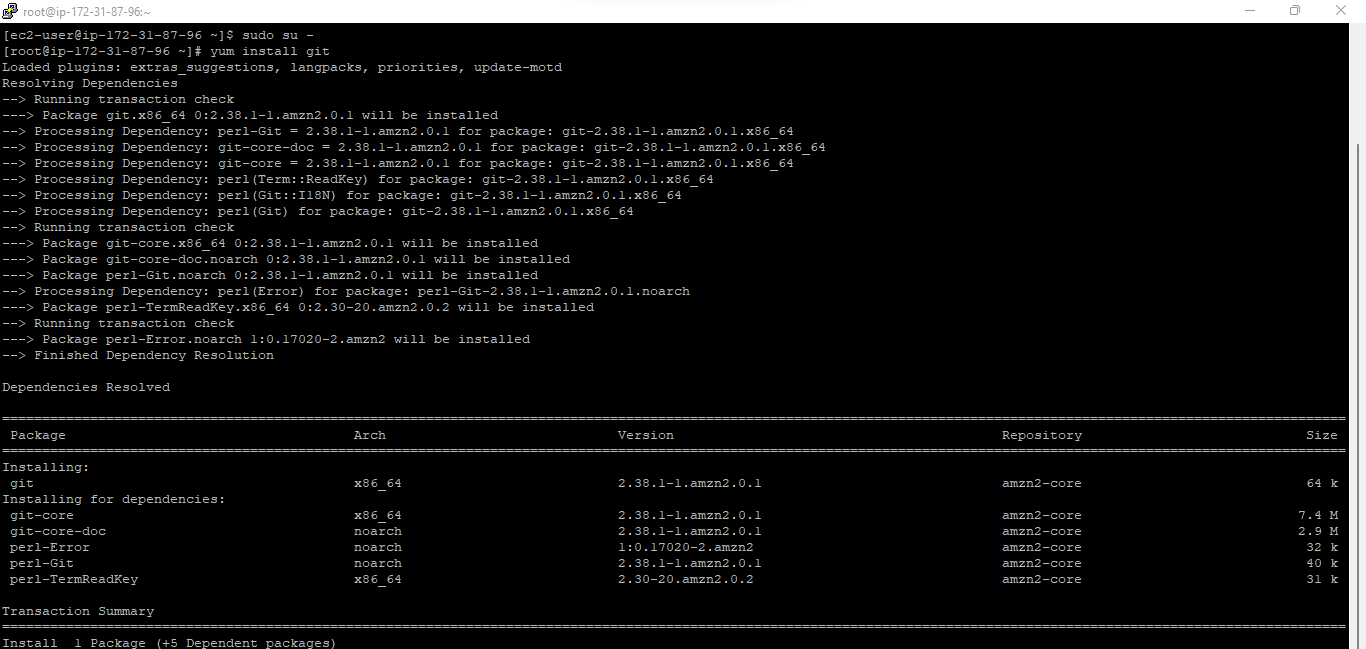
* Create a Repository in Github using the source code provide in the Project resources:

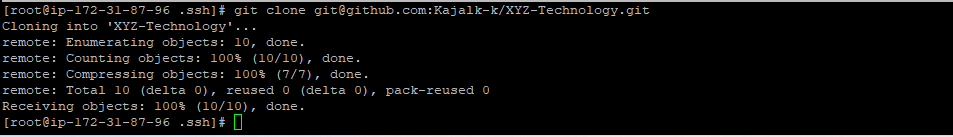


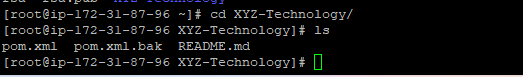
* Create an AWS instance to work on :



* Now install git in the aws instance where we have to work using the mentioned command: ***Sudo yum install git***
* Initialize git using: ***git init***
* Configure the remote url for your repo, email id and username for your github account.
* Now clone the repo in the machine using the command: git clone <url for the repo >

******

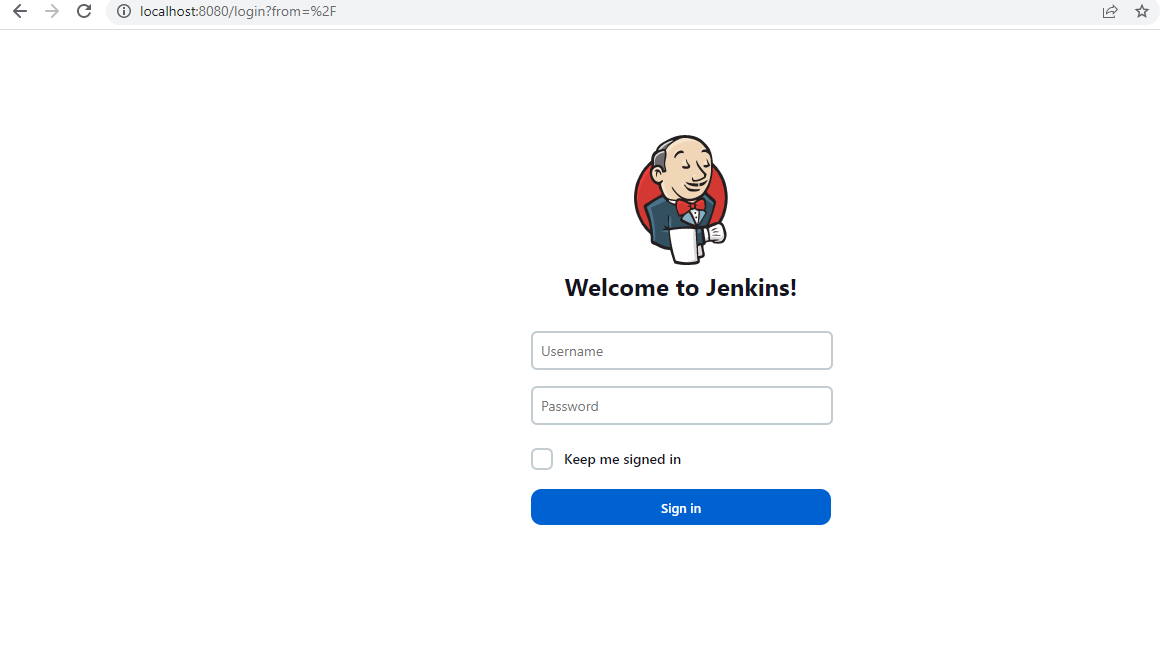
******

******

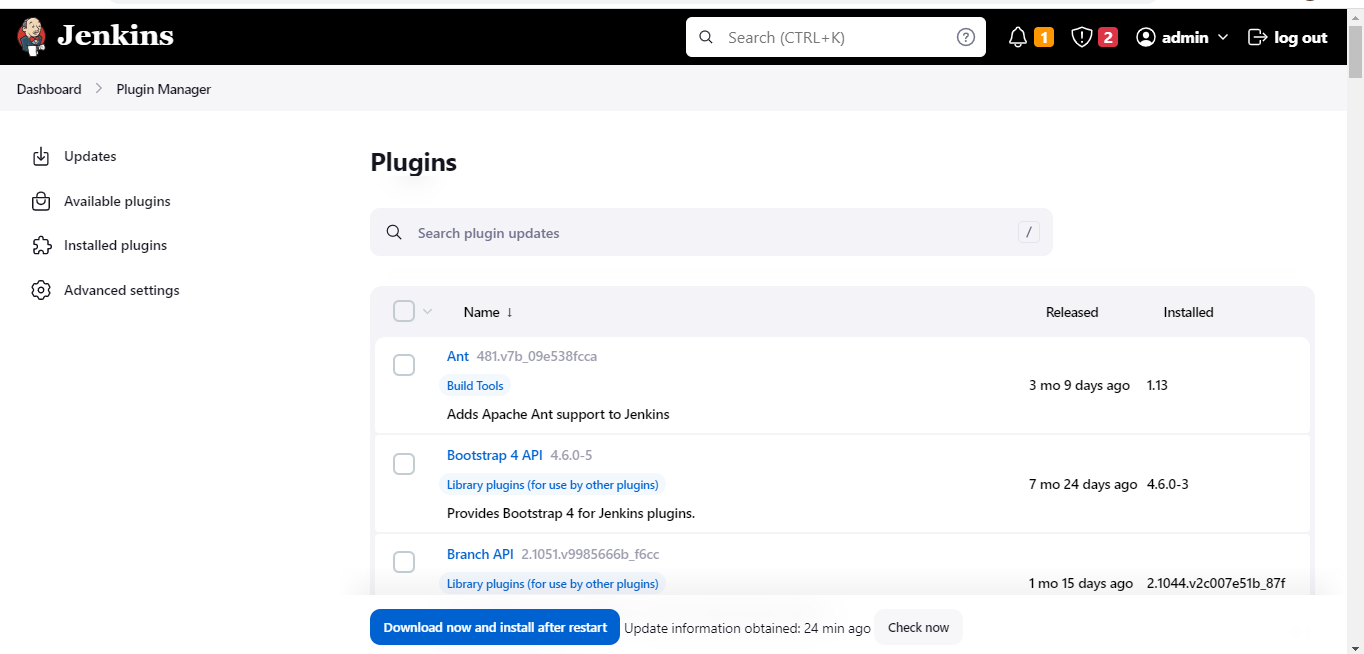
***Task 2:***

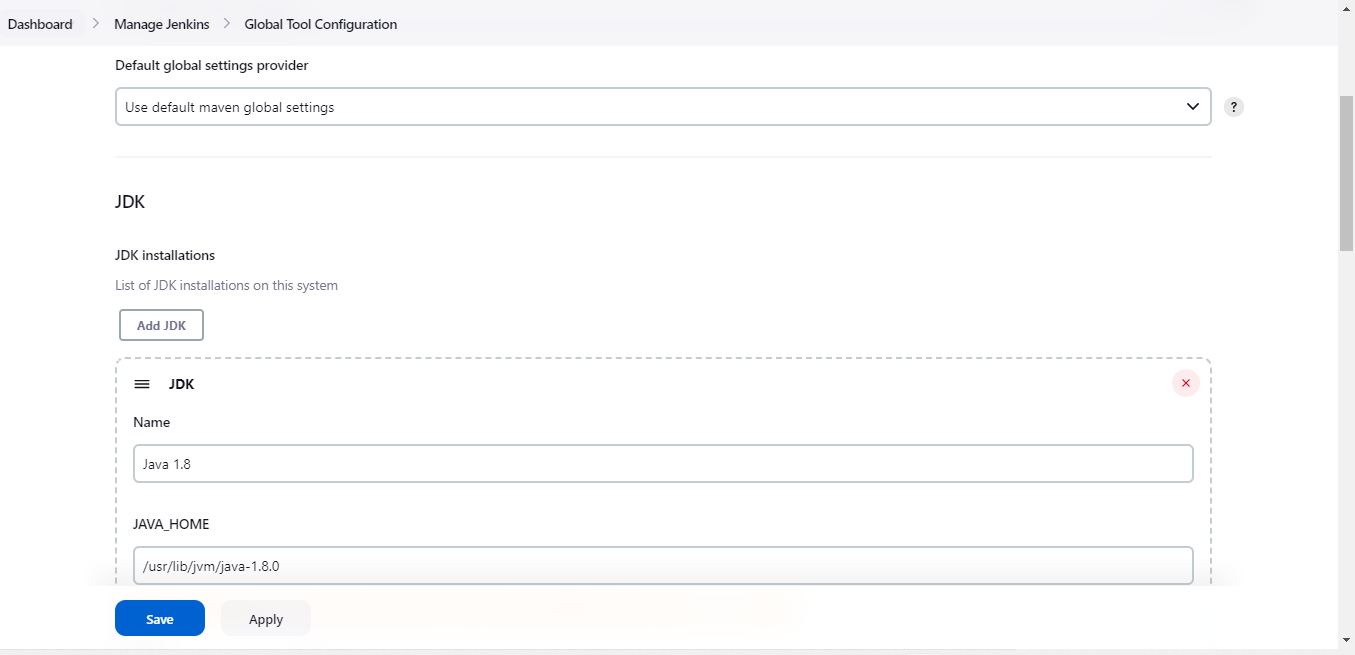
* Once we have the code now we need Jenkins to create the pipeline for our project. Using the Jenkins pipeline we will build and test our project .
* Install the Jenkins and then open the UI of the Jenkins to create the pipeline.

Global tool configuration: Java

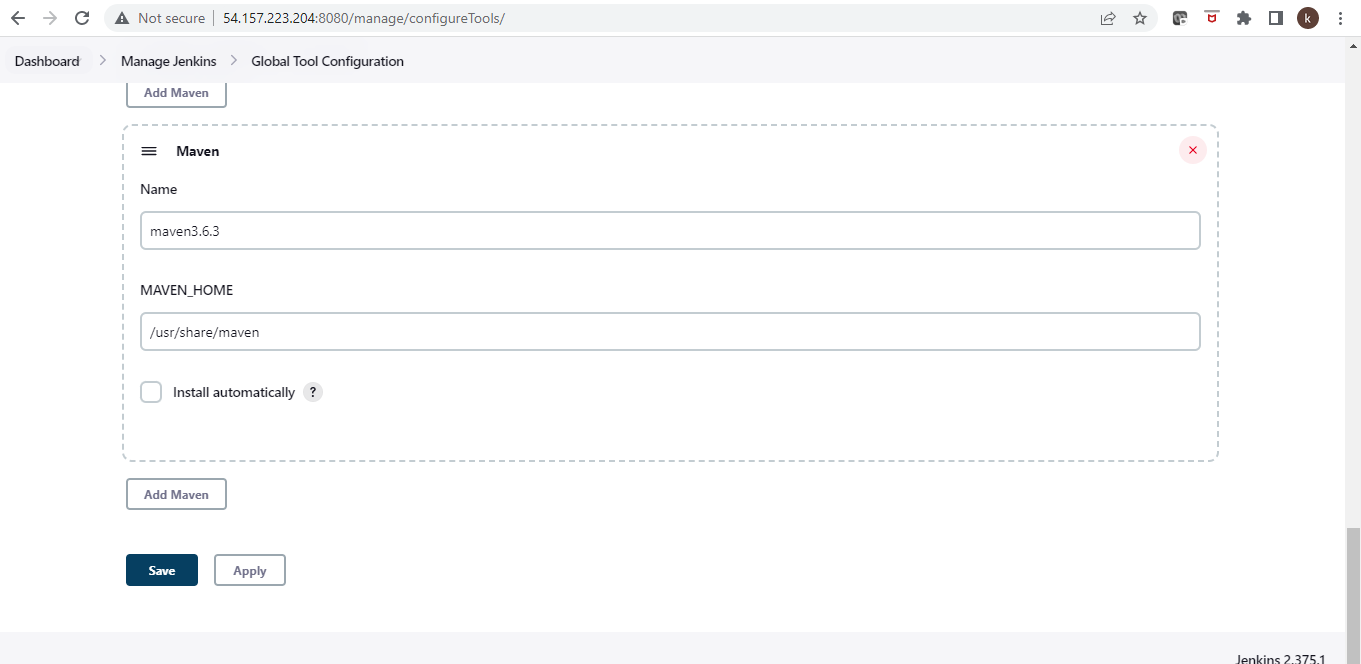


* Manage Plugins in the Jenkins:

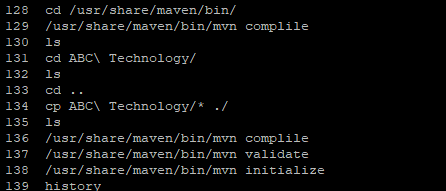


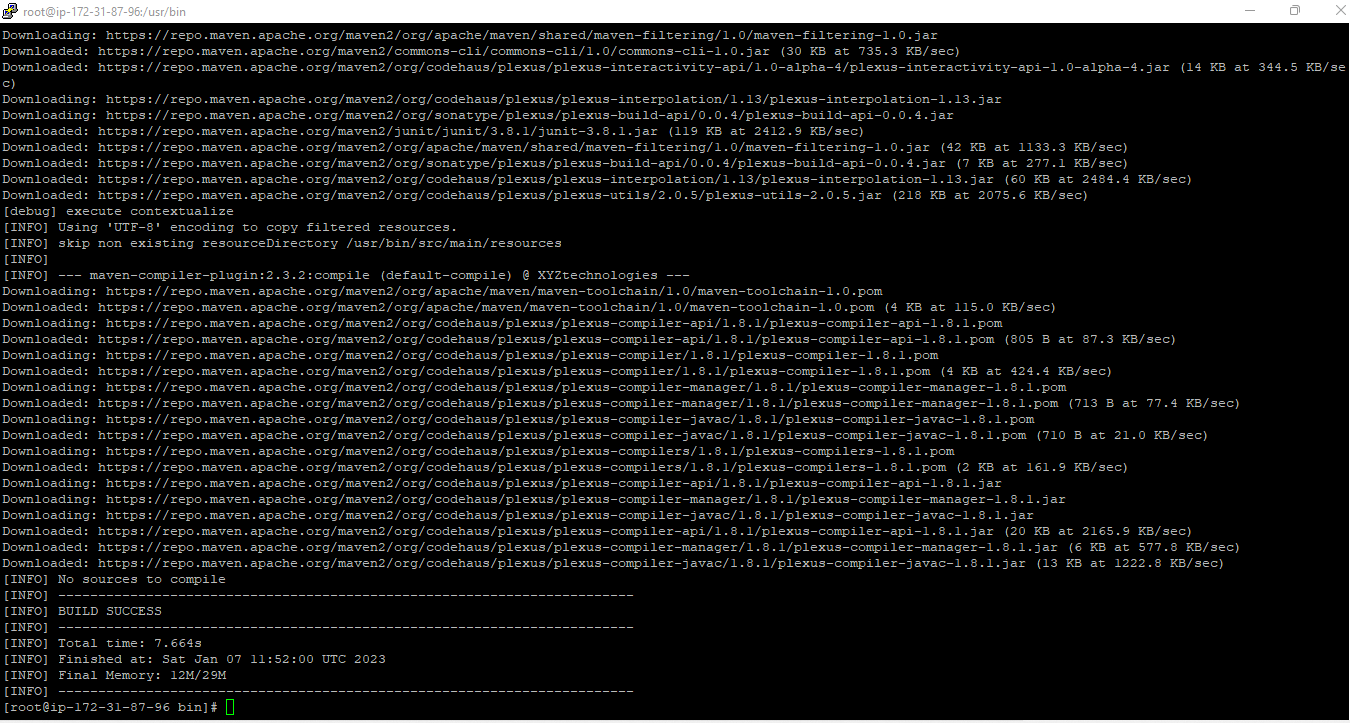
* Make the global tool configuration for java , maven and git

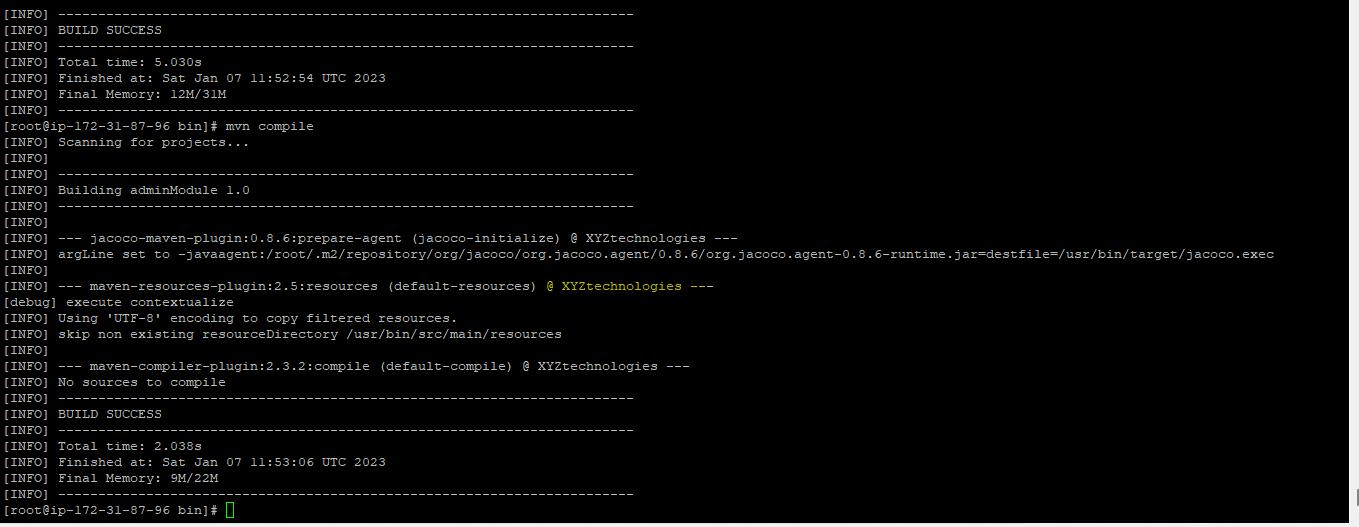
Global tool configuration: Maven

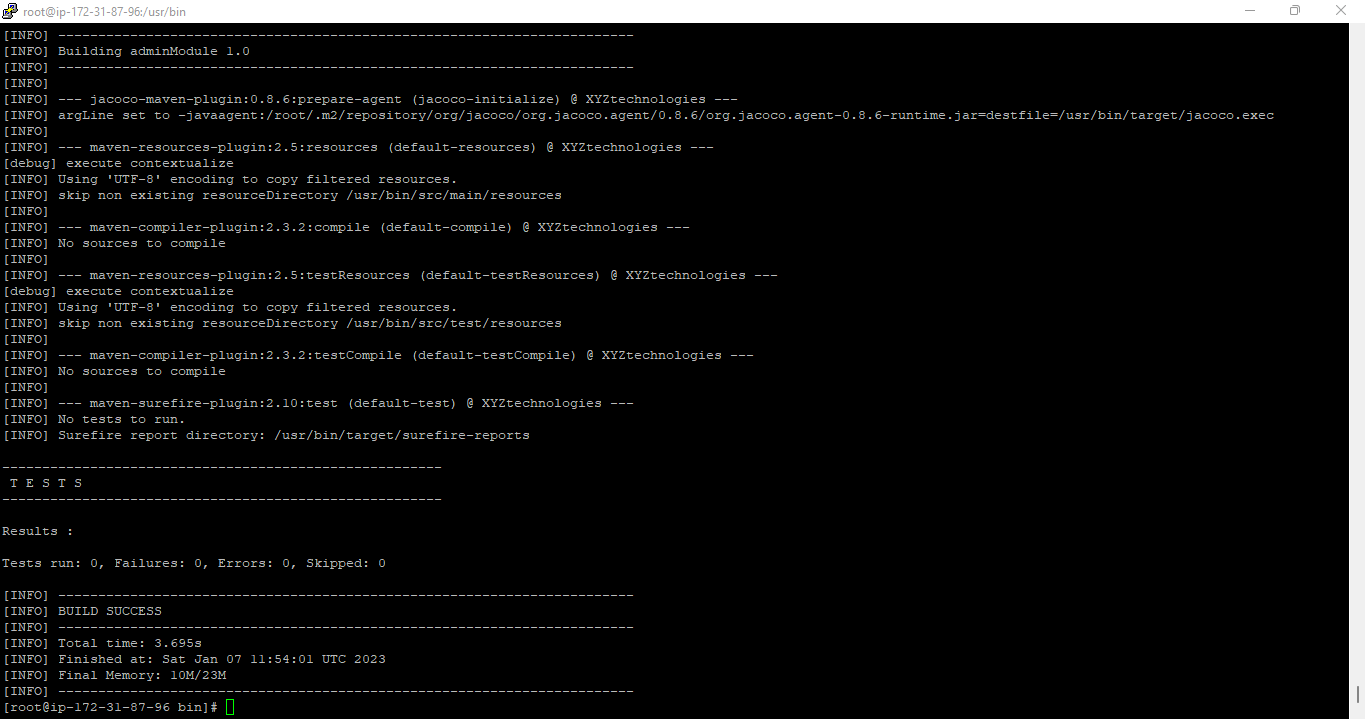


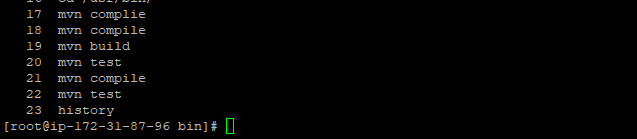
* Create the pipline using the following steps:
* Go to add new project , give the source code details , build steps ad mentioned below and build the pipeline:

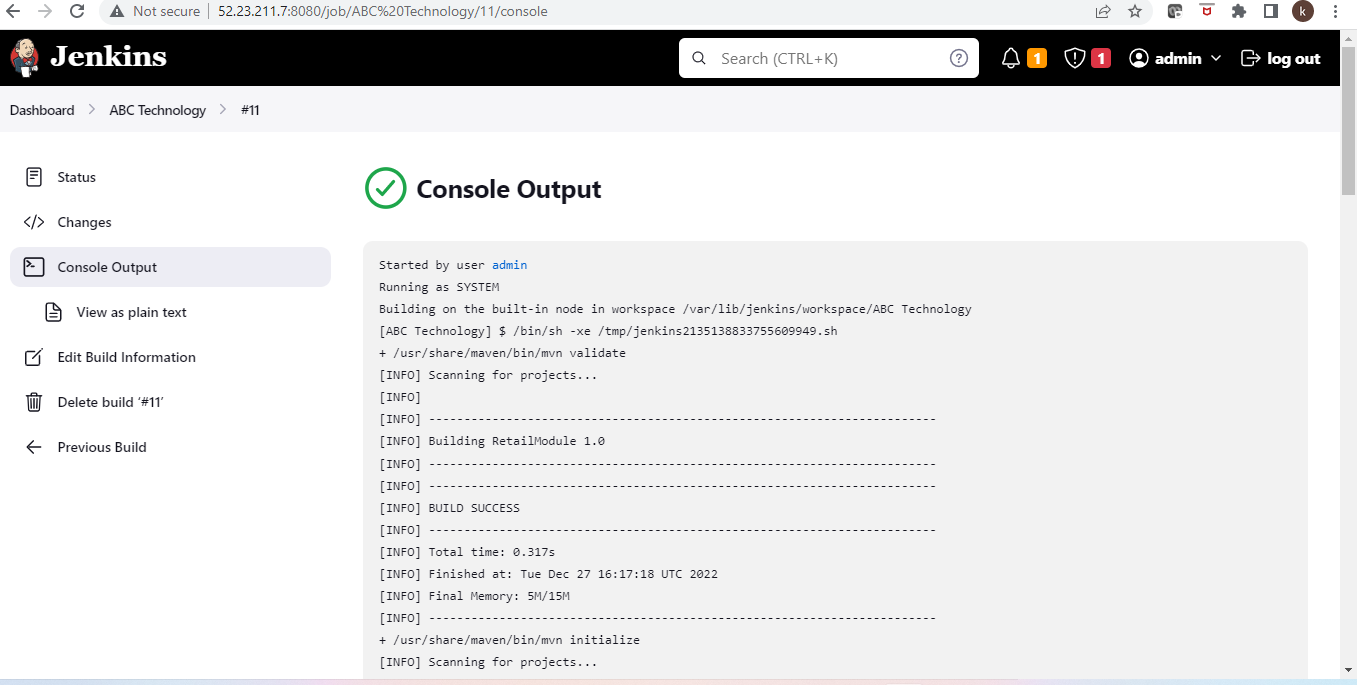


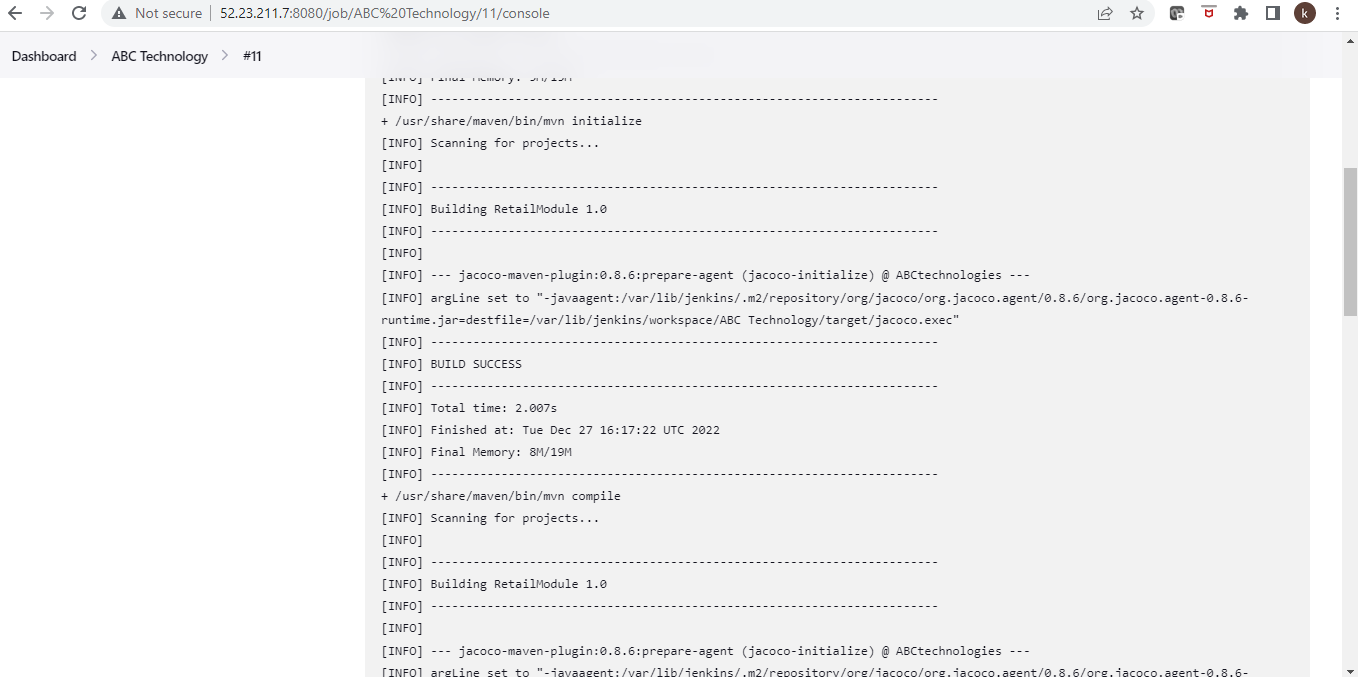


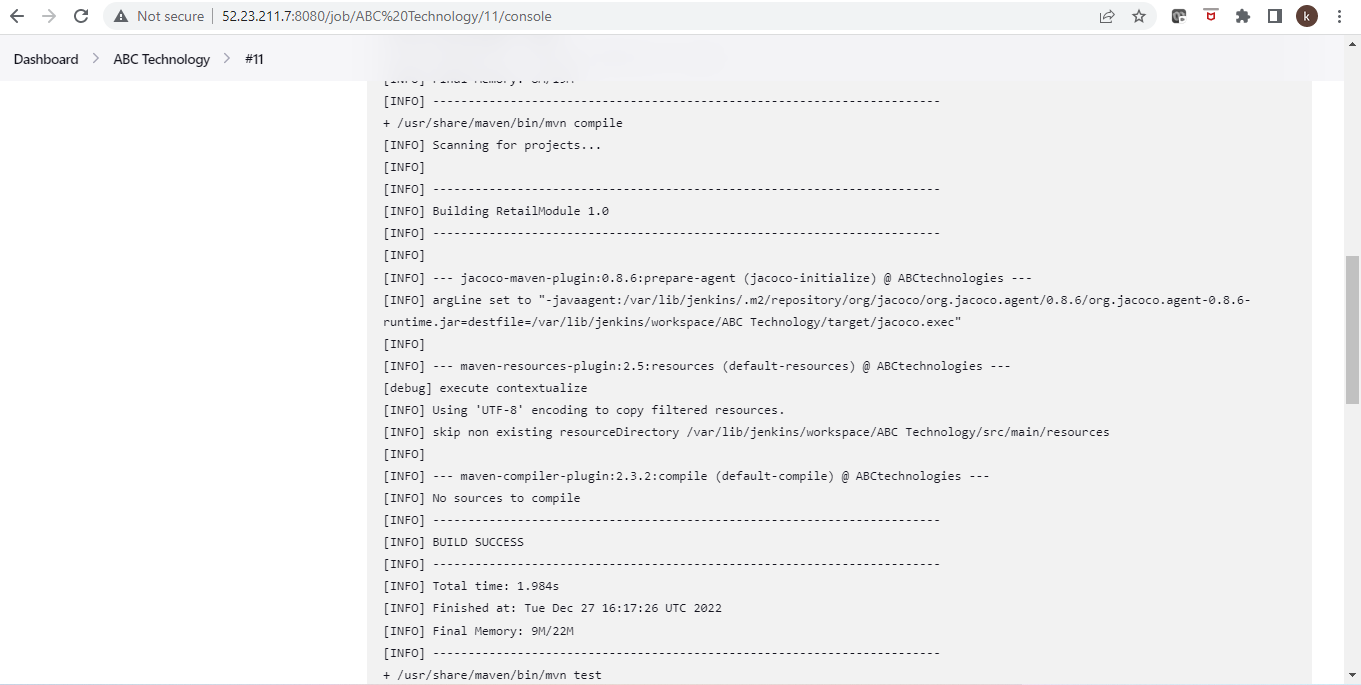


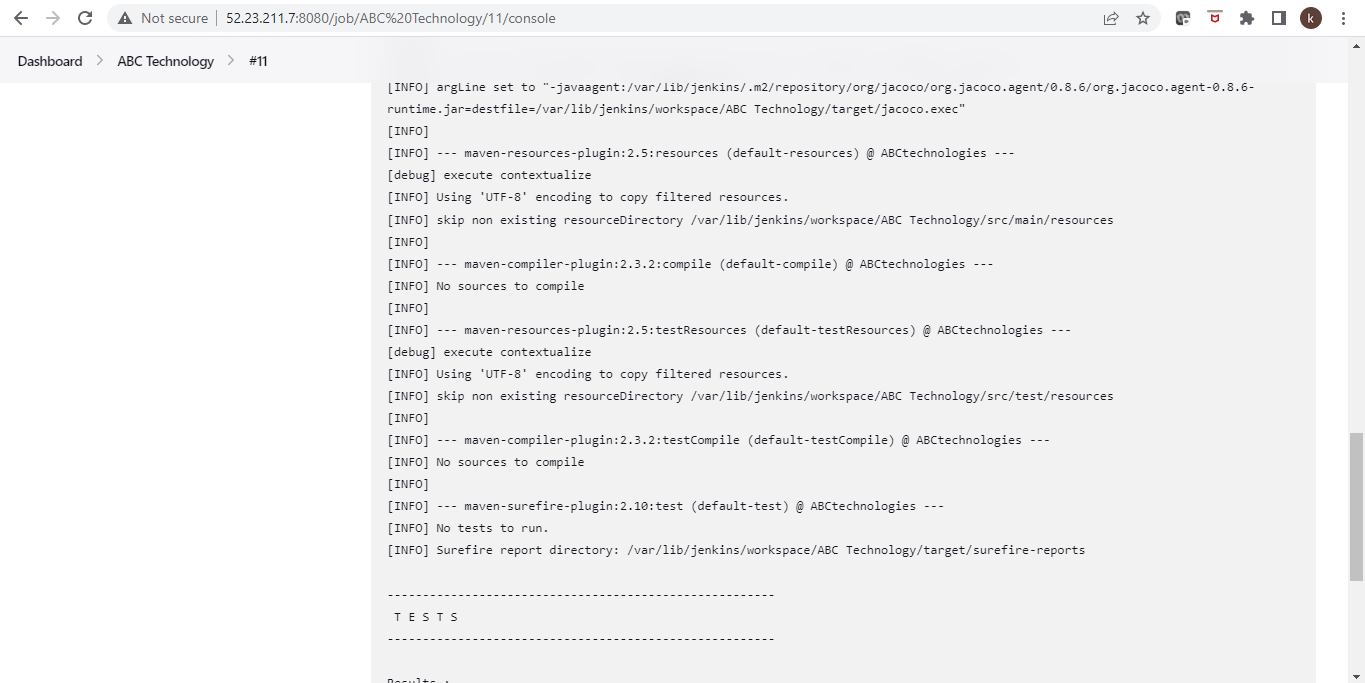


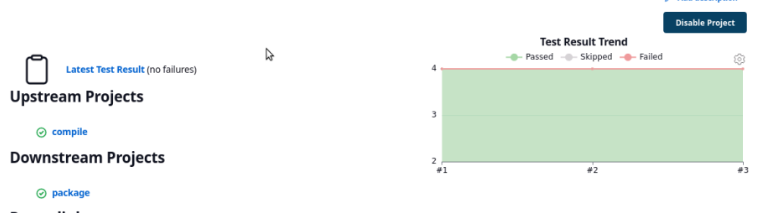


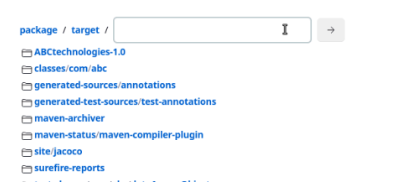










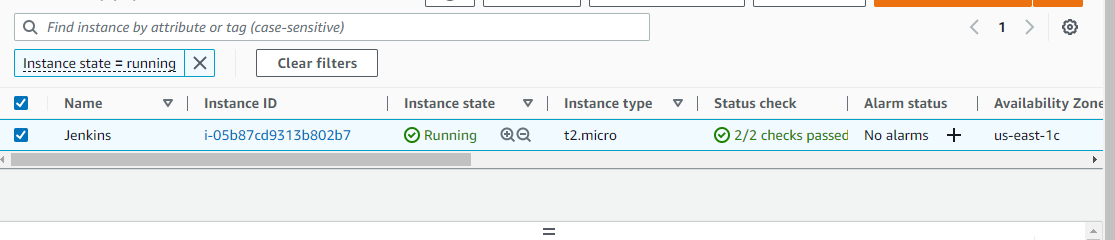


* For Poll SCM used the mentioned schedule:

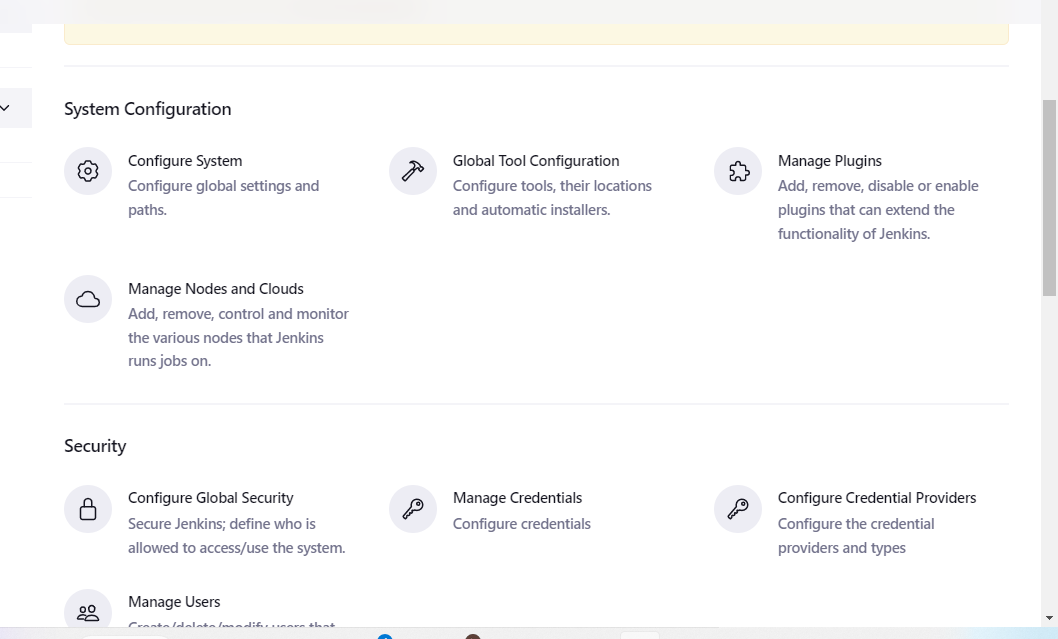
**H/2\*\*\*\***

* /usr/share/maven/bin/mvn validate
* /usr/share/maven/bin/mvn initialize
* /usr/share/maven/bin/mvn compile
* /usr/share/maven/bin/mvn test
* /usr/share/maven/bin/mvn package

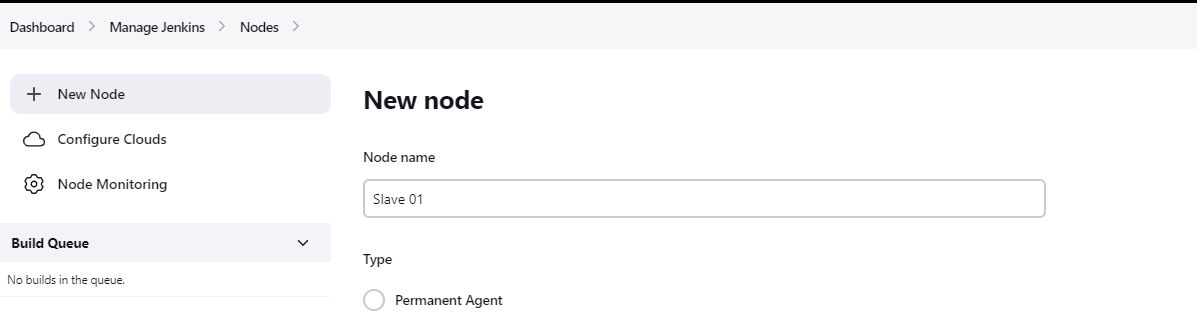
* For creating a master slave Jenkins architecture crating another ec2 instance in aws account.



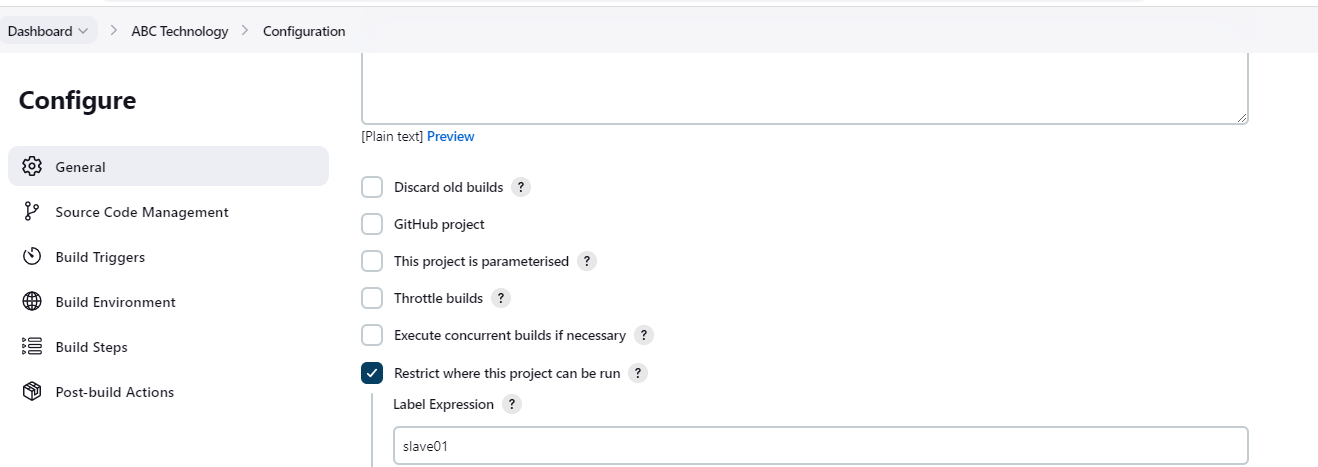
* Once the machine is up and running fine setup Jenkins and java in it.
* Post that follow the mentioned steps to set up agent and master linked:
* Go to manage Jenkins and select “manage Nodes and clouds” Option:



* Click on add new node, give the node a name and configure its details.

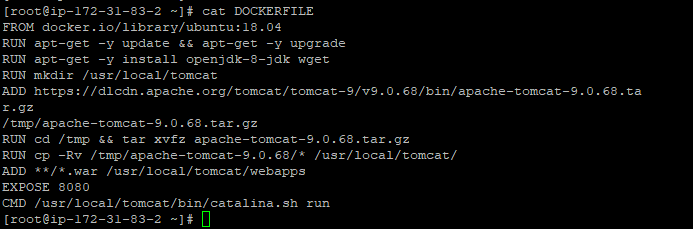


* Now we can restrict the builds to run only on the slave as per our need:

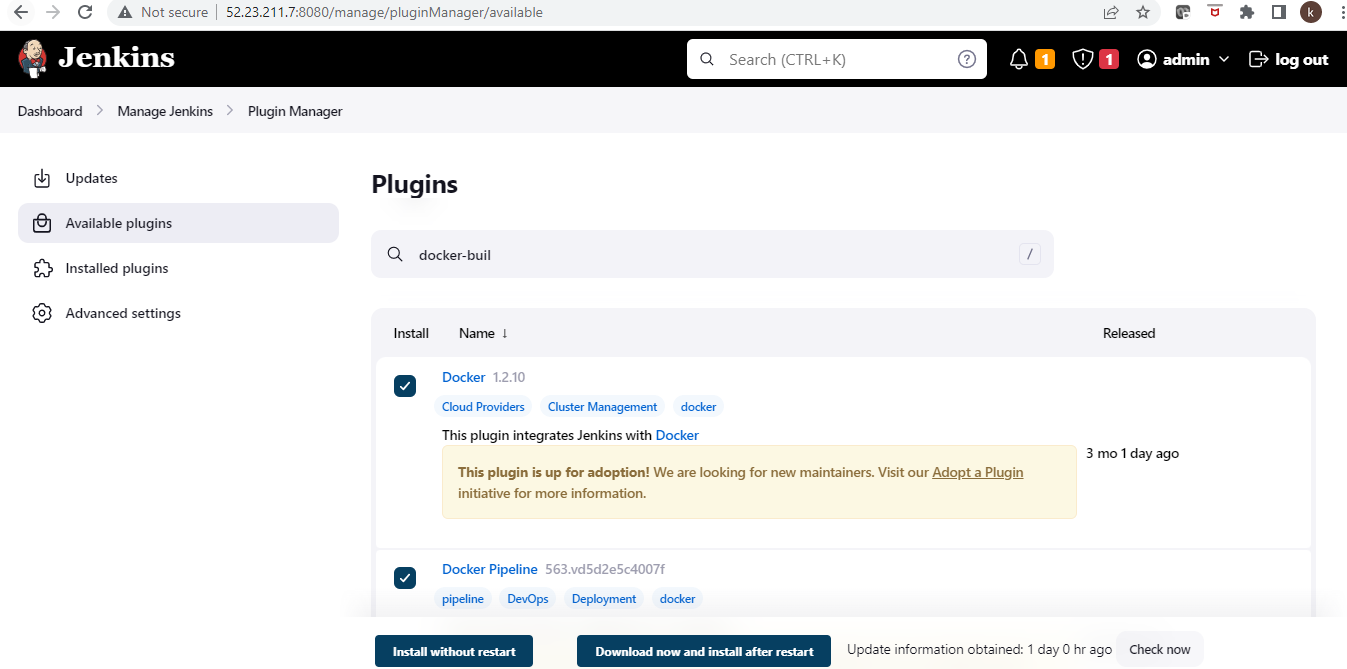


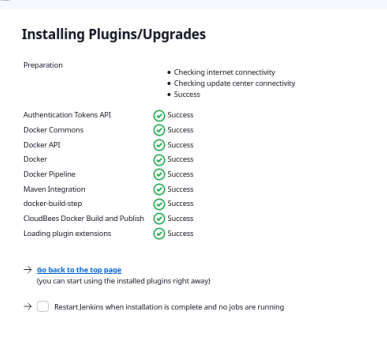
***Task 3:***

* Create a docker file with the mentioned content:

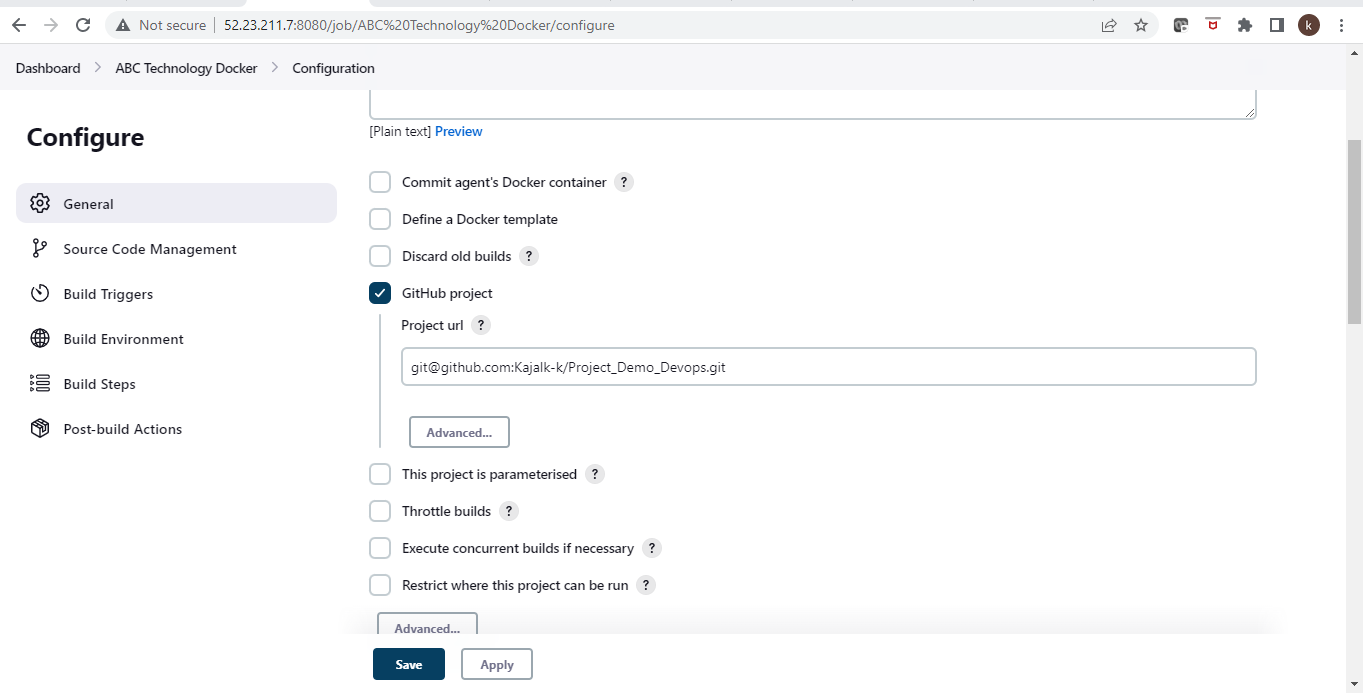


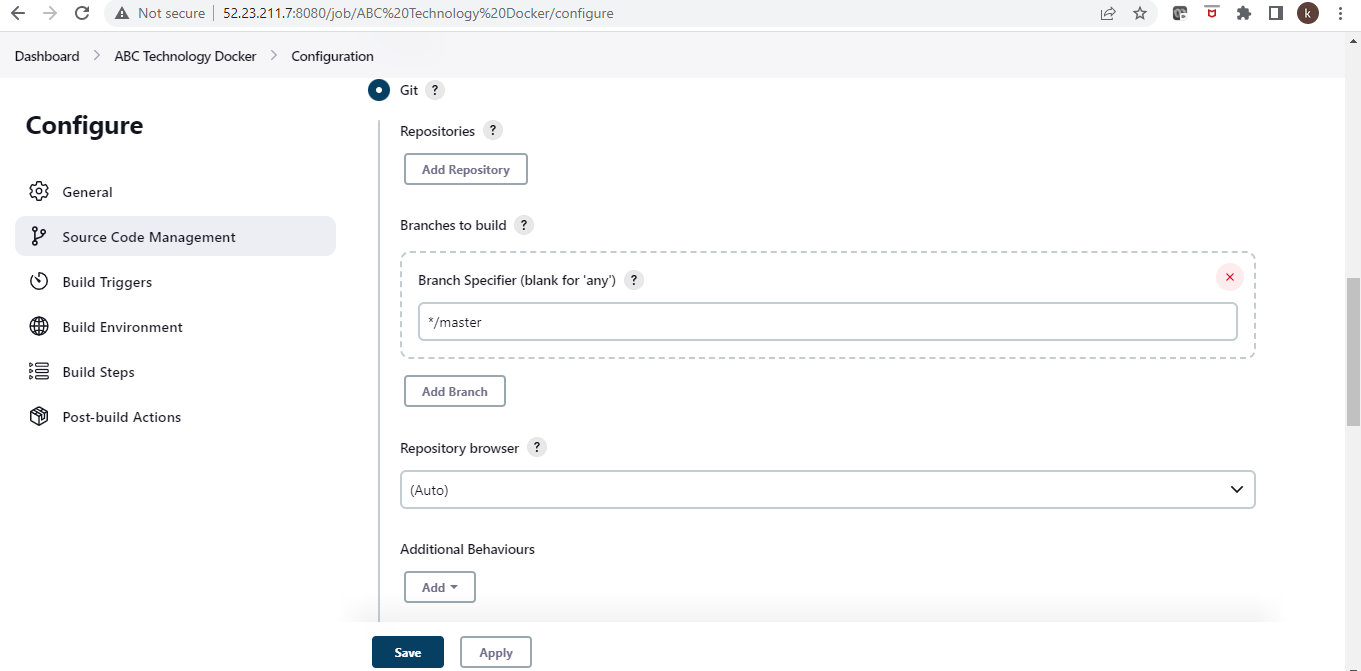
* Install the docker plugin in Jenkins:

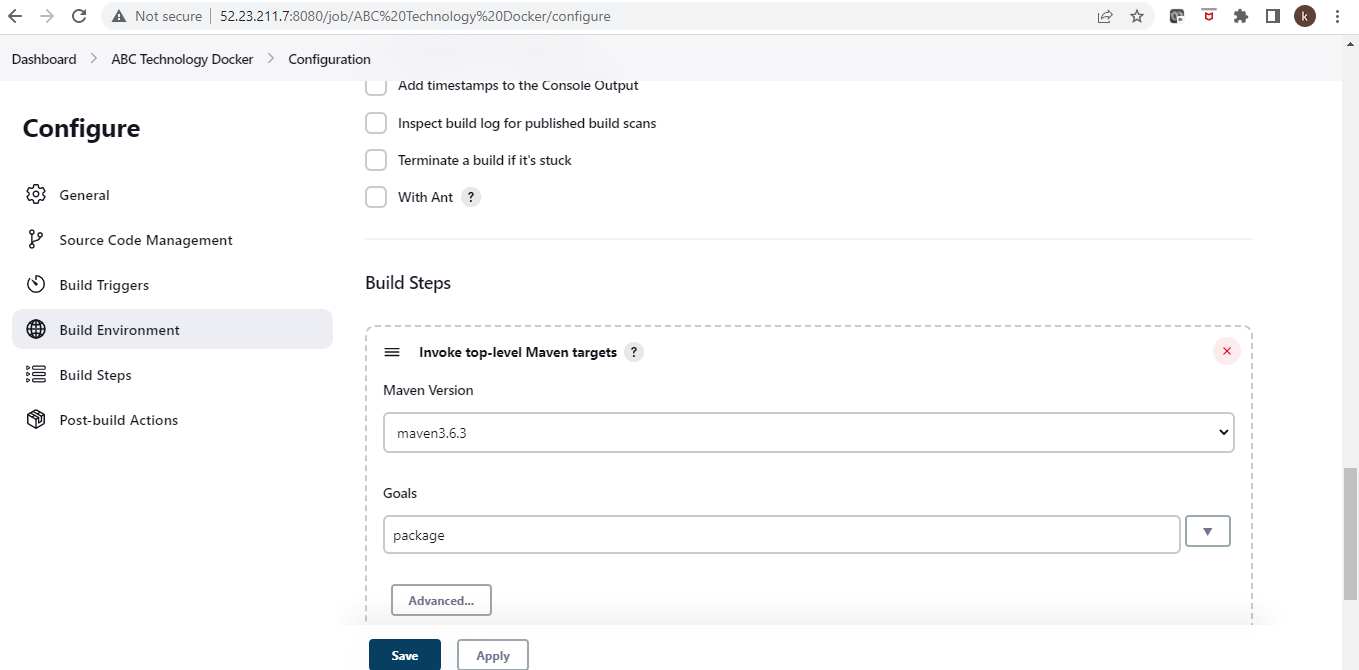


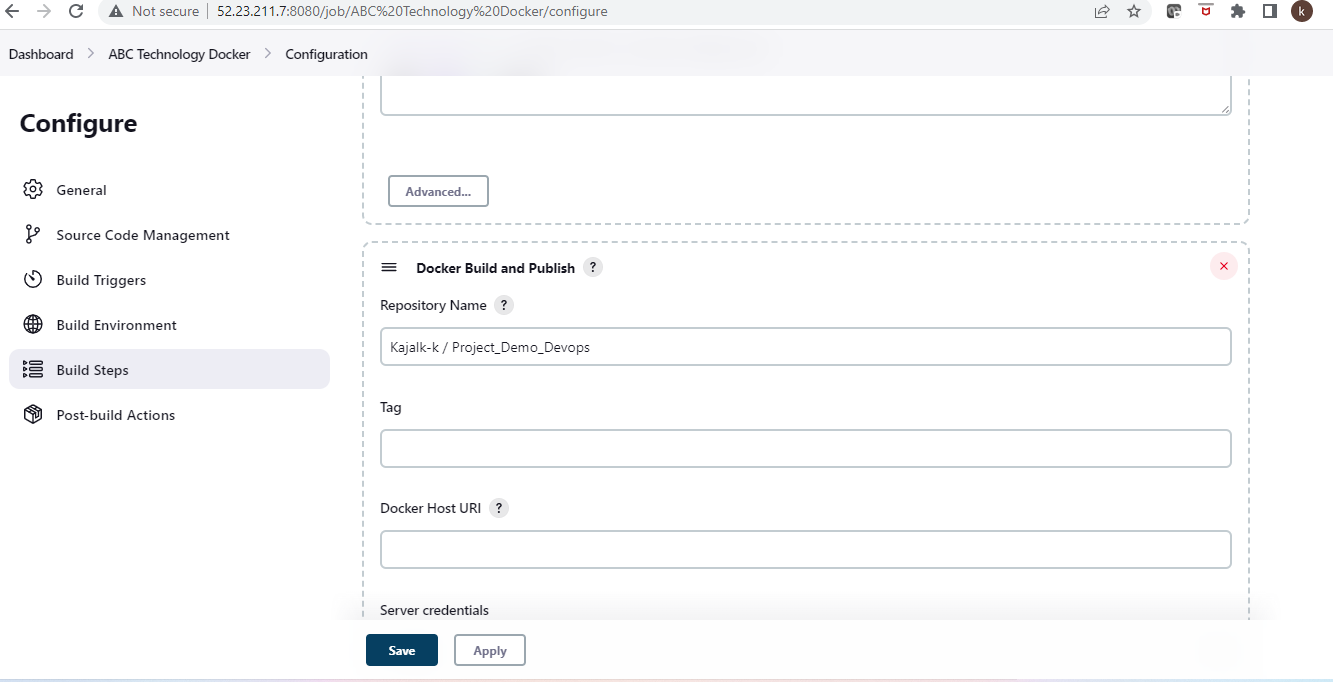


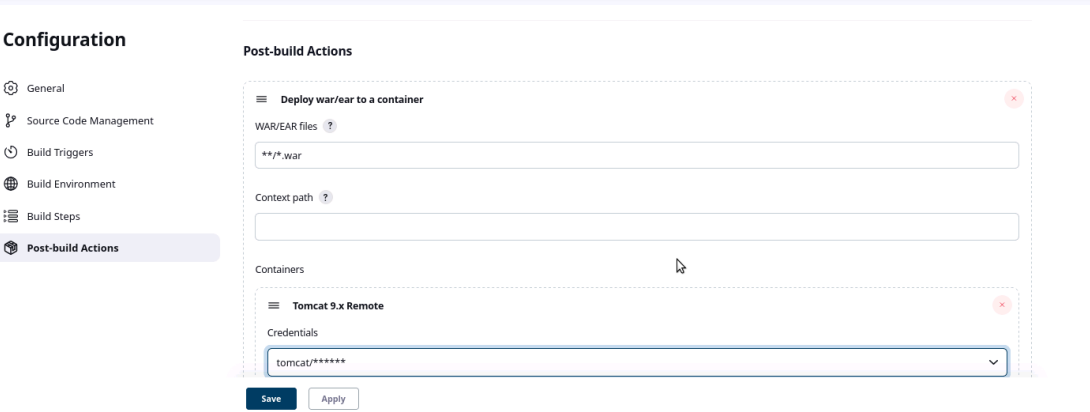
* Create a new pipeline for docker to deploy the project

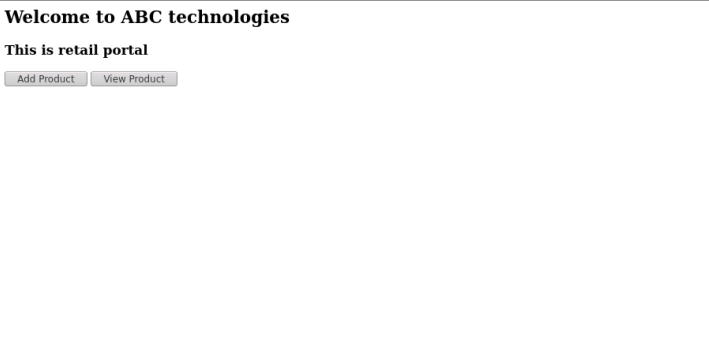










* Validating results
* 

***TASK 4:***

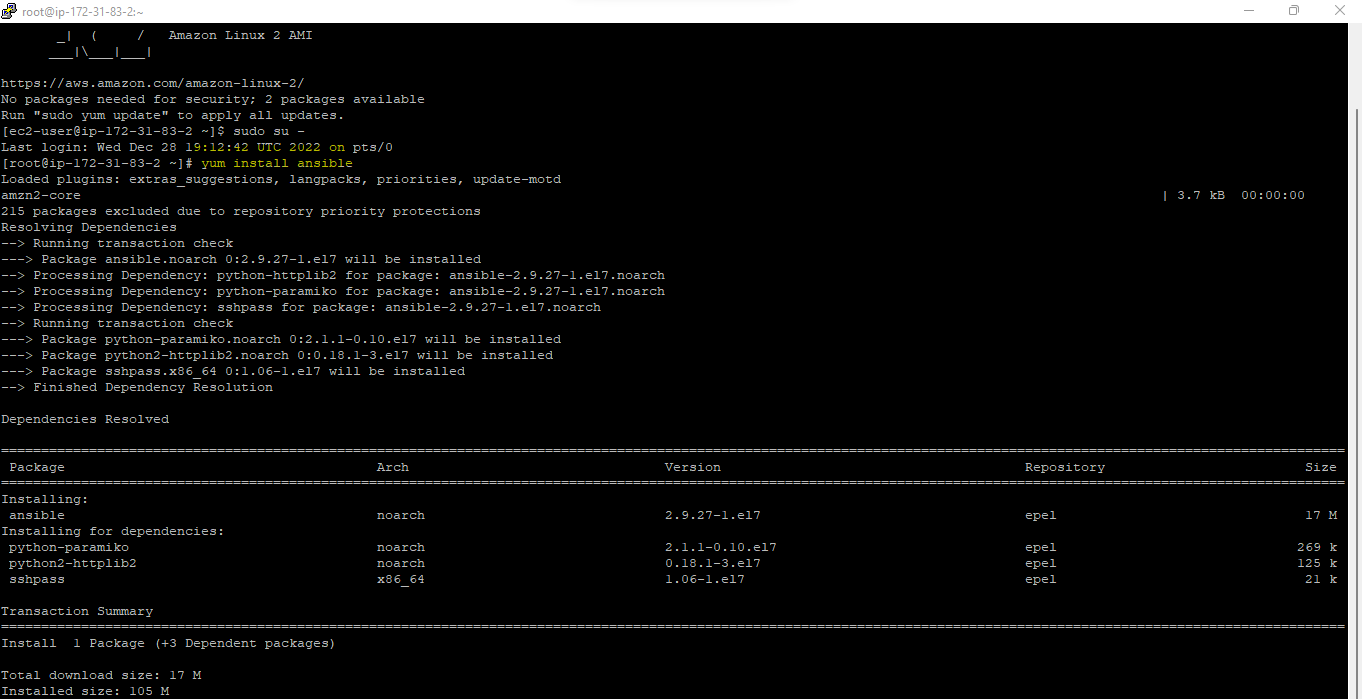
1.Deploy Artifacts on Kubernetes

2. Write pod, service, and deployment manifest file

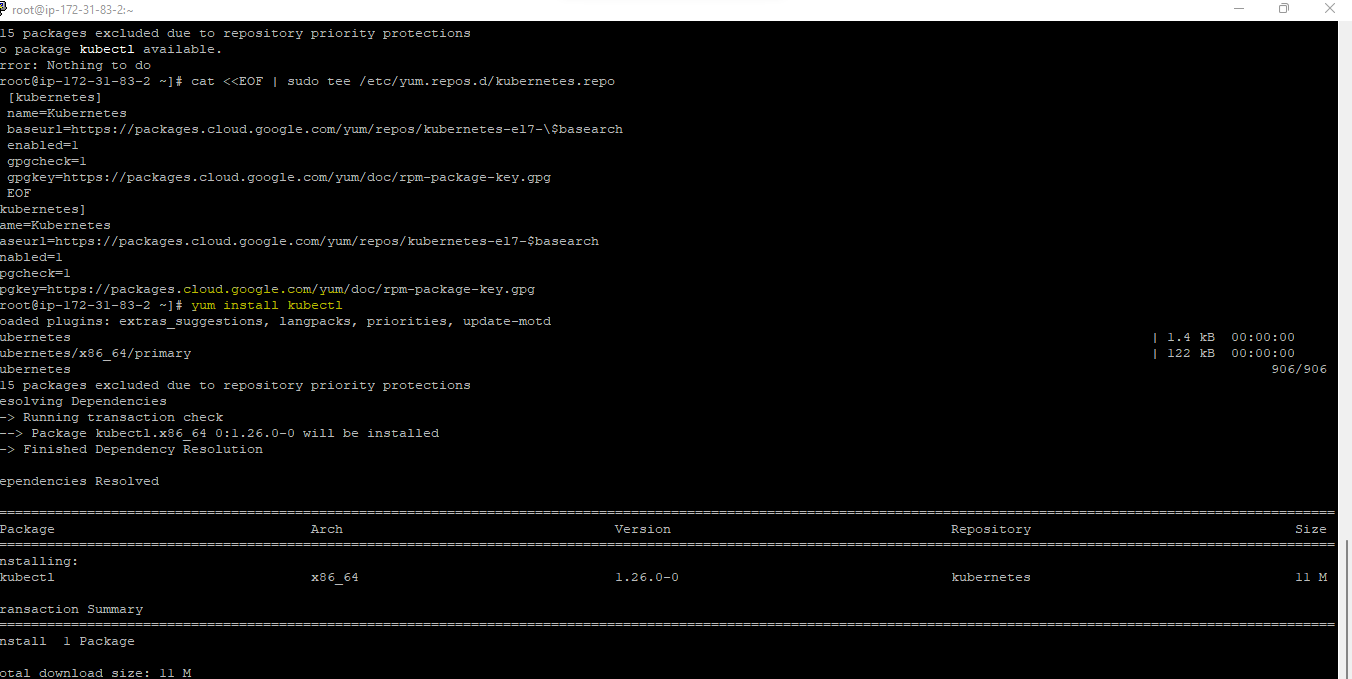
3.Integrate Kubernetes with Ansible

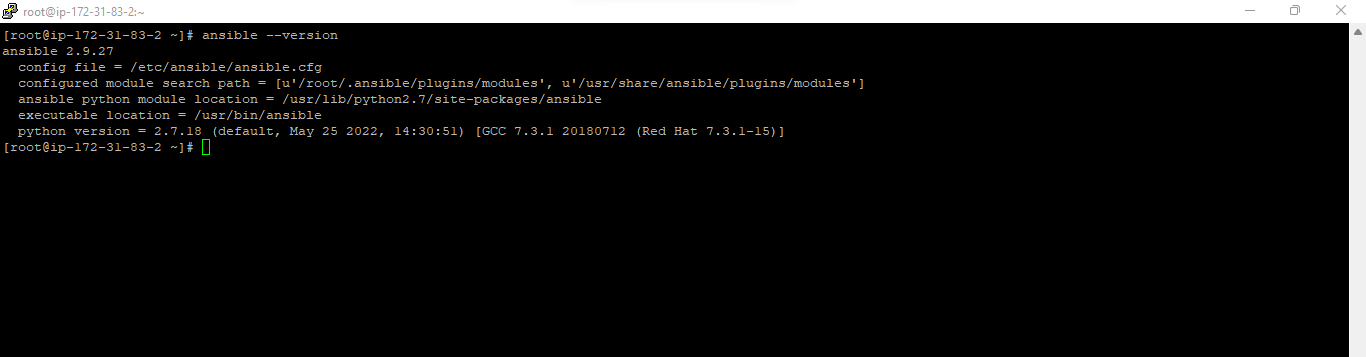
4.Ansible playbook to create deployment and service

* Install Ansible in the instance:



* Install Kubernetes:



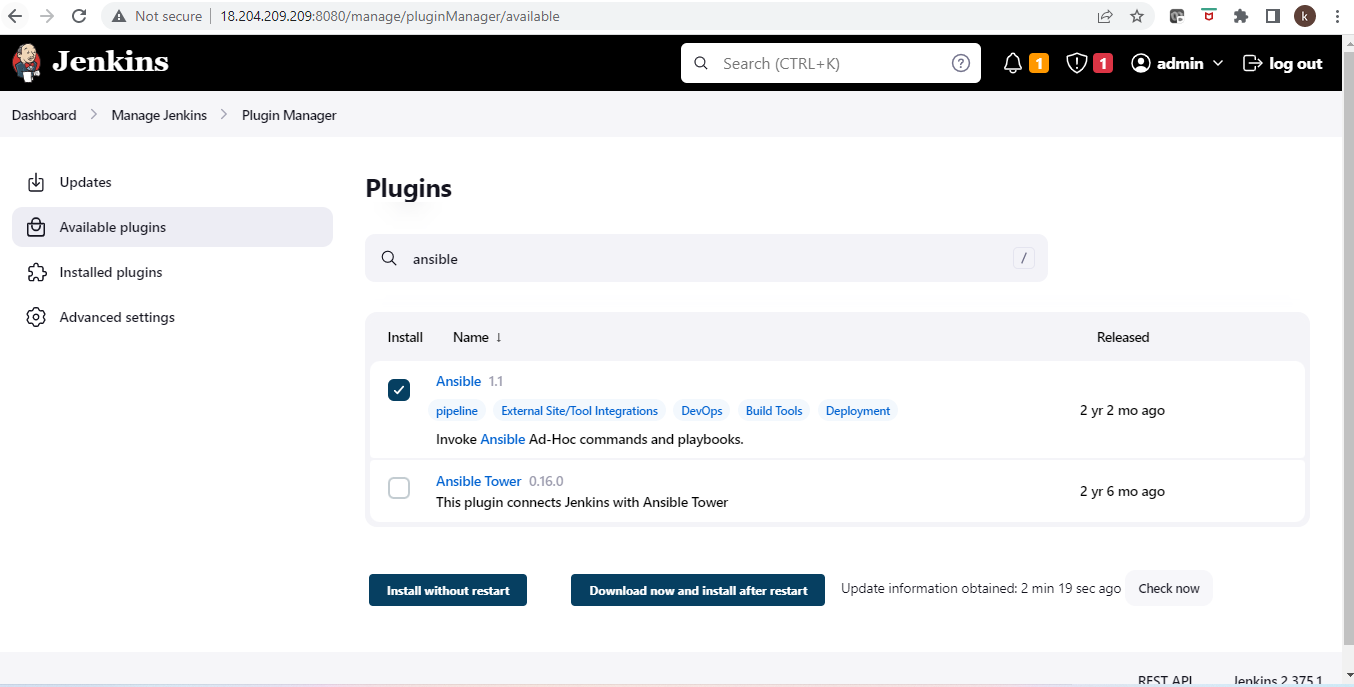
Ansible version: ansible 2.9.27

Kubernetes version:

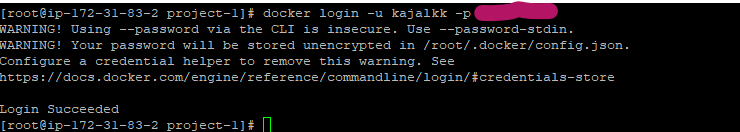
Client Version: v1.18.3

Server Version: v1.18.20

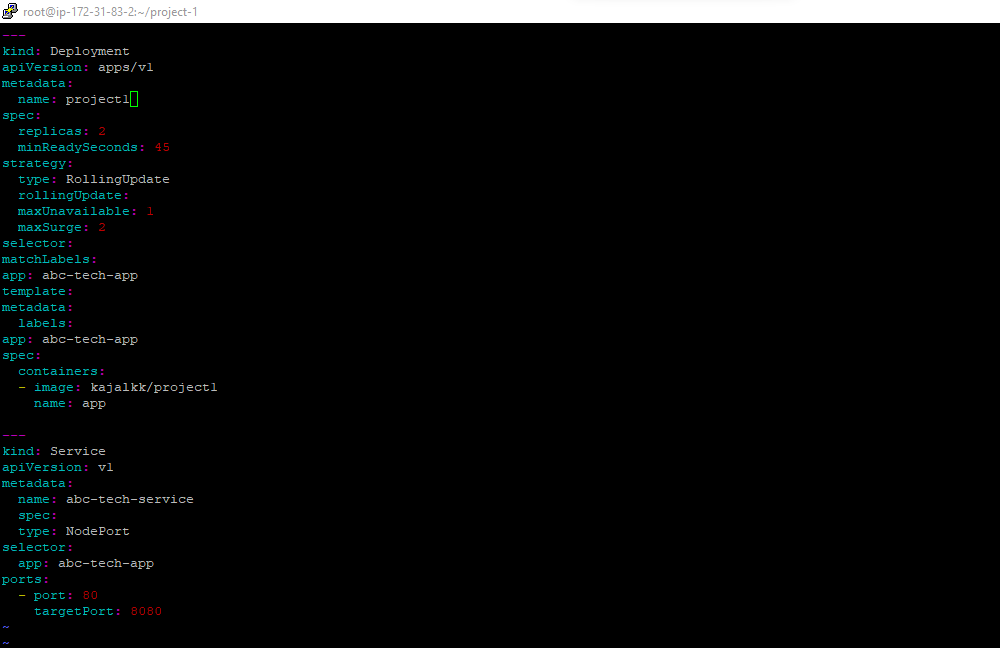
* Install Ansible plugin in Jenkins:



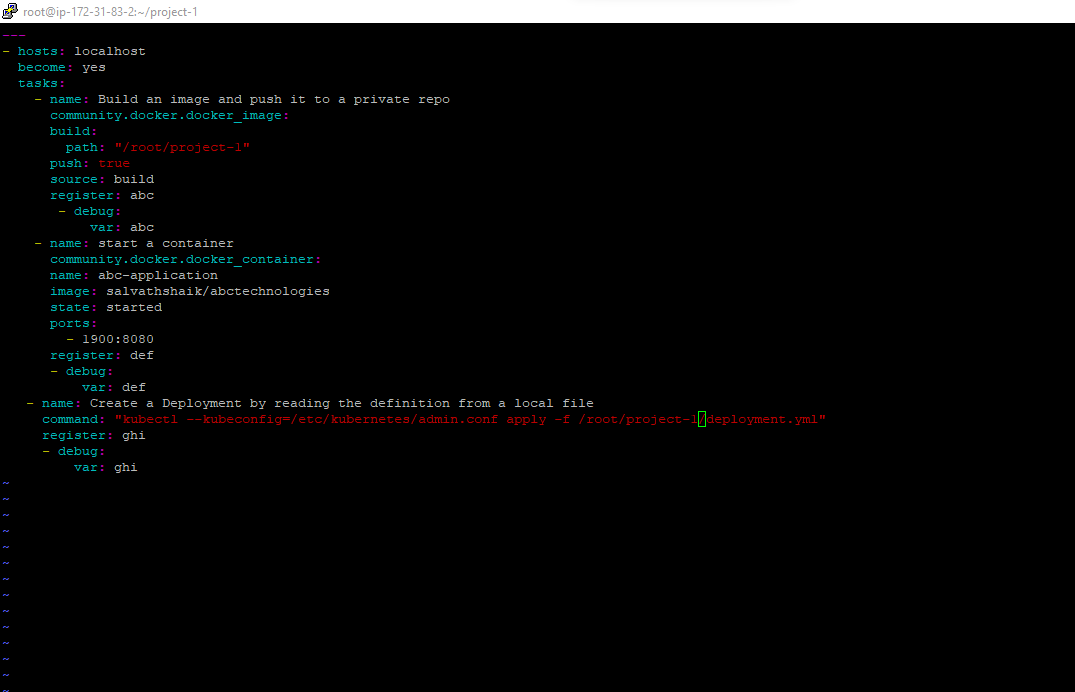
* Login to the docker using the CLI command:



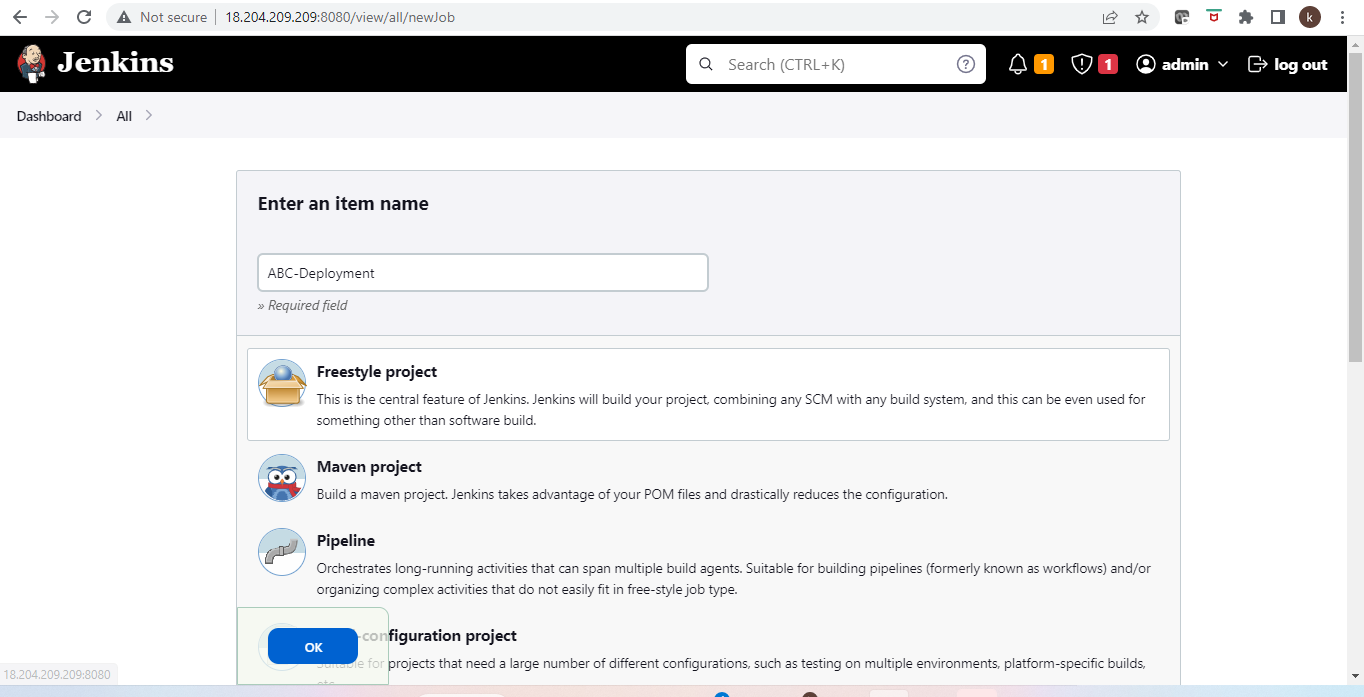
* The deployment file for pod, service and deployment :

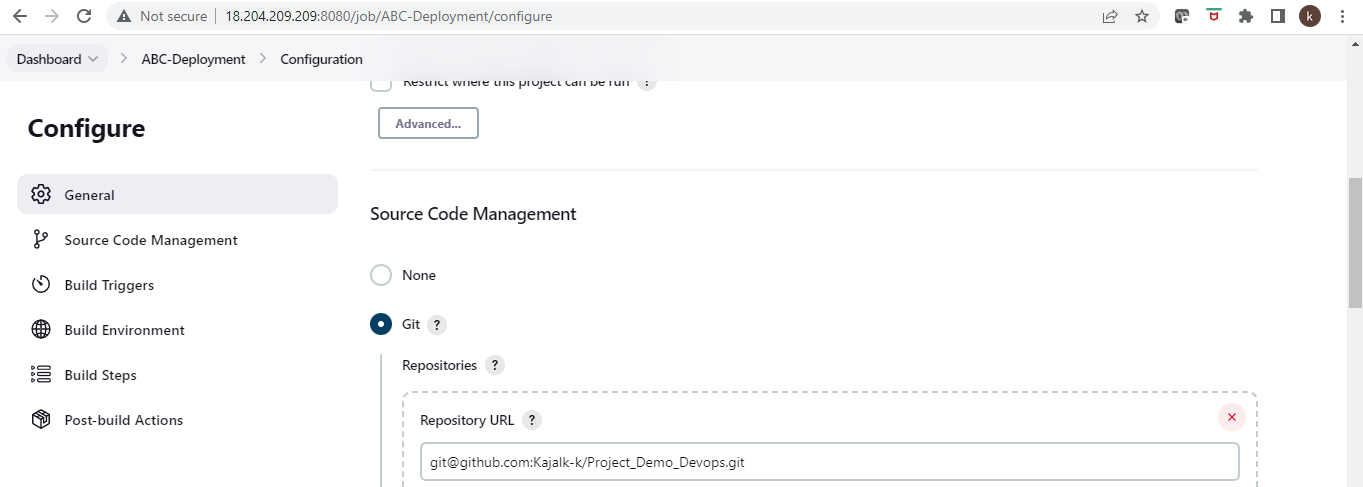


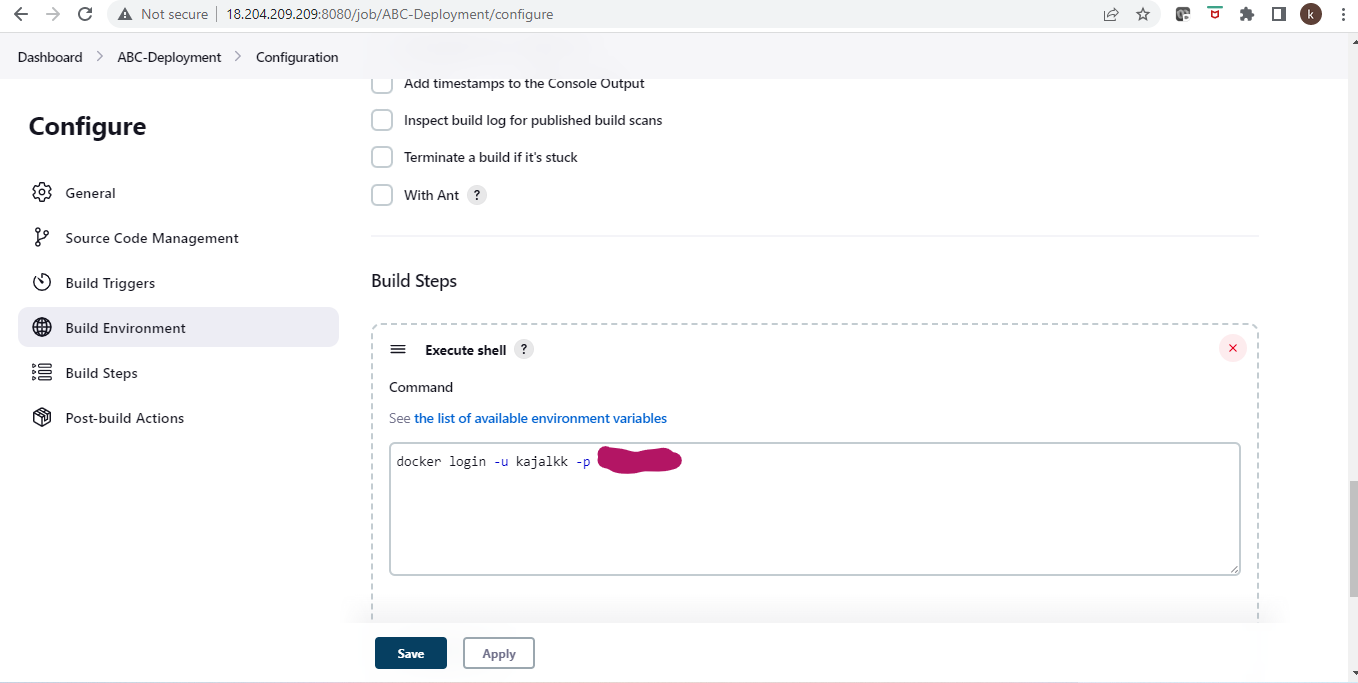
* The playbook needed to create deployment:



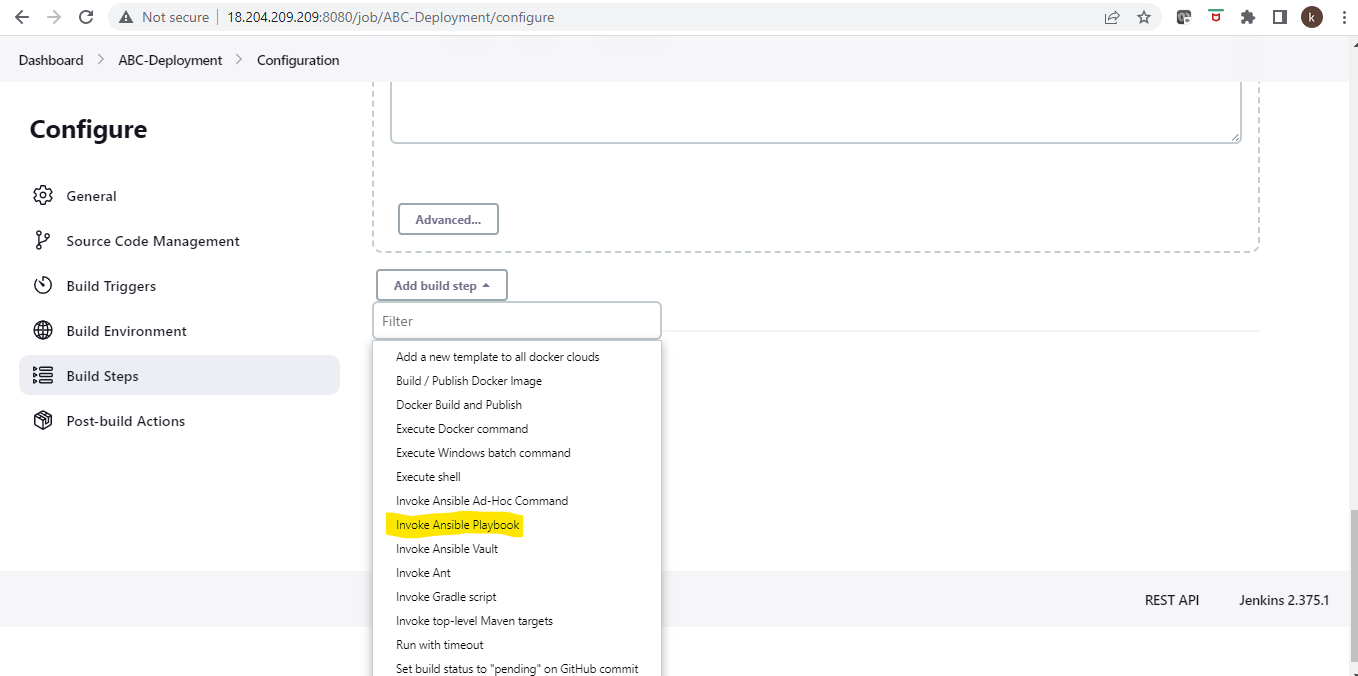
* Create a new job in Jenkins as “ABC-Deploy”







* Invoke ansible playbook in build step:



* Provide playbook path and host file path:

