ASSESSMENT 2 – BUILD A DOCKER JENKINS PIPELINE TO IMPLEMENT CI/CD WORKFLOW

1) Launched the Devops in AWS V2 Lab from simplilearn.

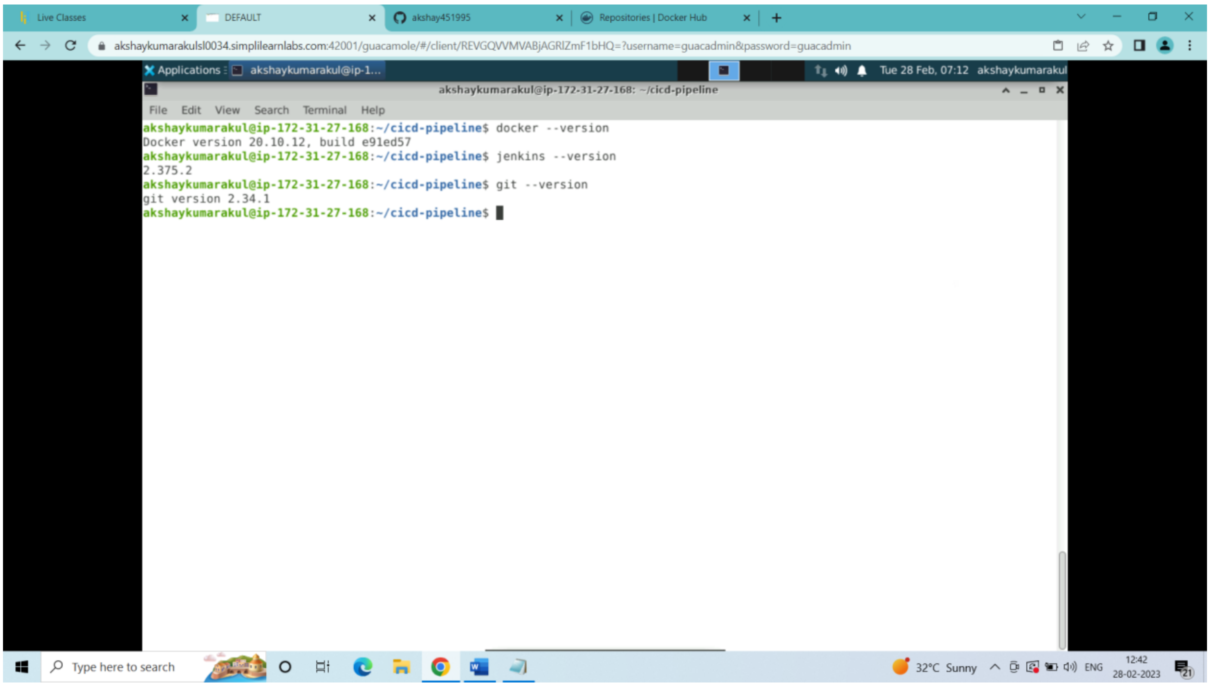
2) Started the ubuntu machine from simplilearn in RDP access through auth url.

3) Updated the machine:

   sudo apt-get update -y

4) Tools required in the machine for Docker-Jenkins pipeline:
   Docker, Git, Jenkins

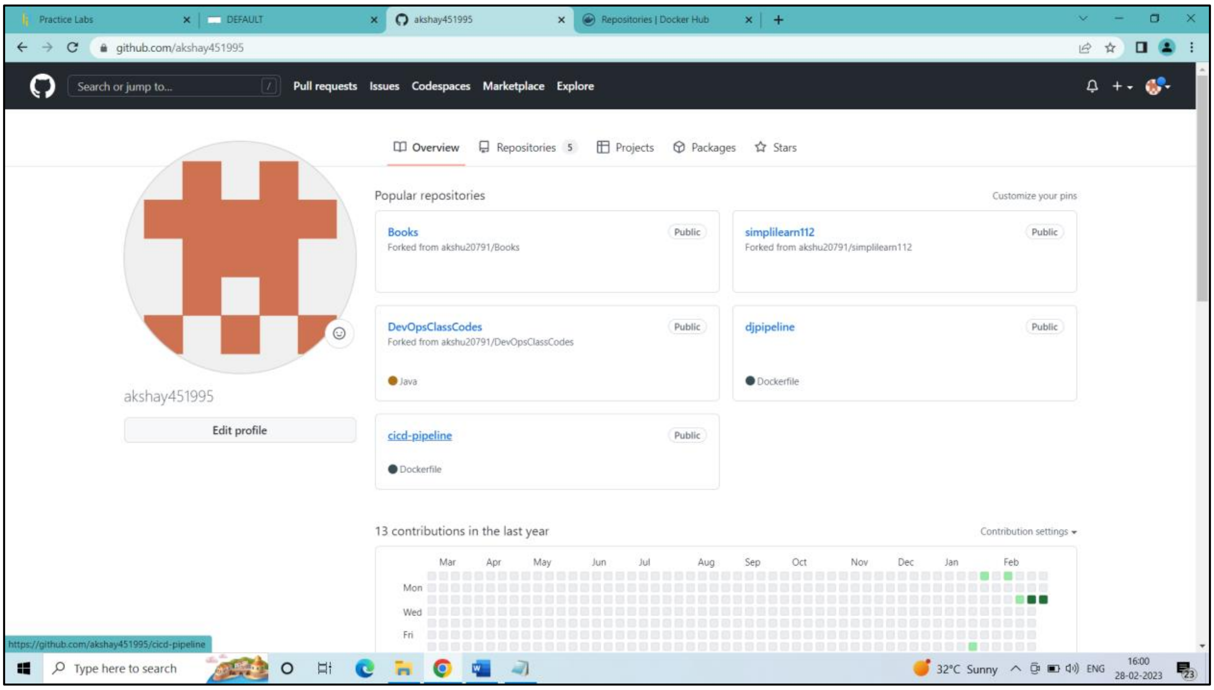
5) Docker, Jenkins and Git are already installed in the machine:
   Used the following commands to check the versions of the docker, jenkins and git:
   docker –version
   jenkins –version
   git --version



Live Classes
DEFAULT
akshay451995
Repositories | Docker Hub
akshaykumarakulsl0034.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username=guacadmin&password=guacadmin
Applications
akshaykumarakul@ip-1...
Tue 28 Feb, 07:12 akshaykumarakul
akshaykumarakul@ip-172-31-27-168: ~/cicd-pipeline
File Edit View Search Terminal Help
Type here to search
32°C Sunny
ENG
12:42
28-02-2023

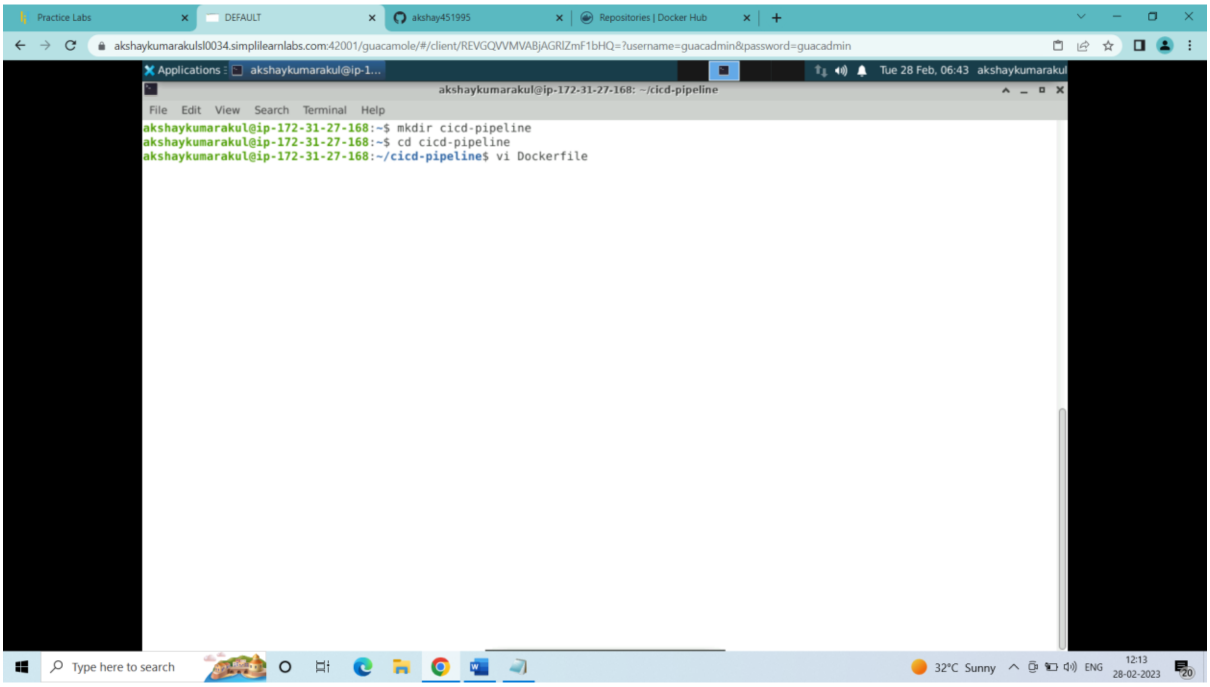
```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ docker --version
Docker version 20.10.12, build e91ed57
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ jenkins --version
2.375.2
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git --version
git version 2.34.1
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ 
```

6) Created a new repository in github account
   github repository name: cicd-pipeline



7) Created a directory named cicd-pipeline in the terminal of the ubuntu machine and entered it
mkdir cicd-pipeline
cd cicd-pipeline

8) Create a Dockerfile and Jenkinsfile inside the cicd-peline directory


Practice Labs
DEFAULT
akshay451995
Repositories | Docker Hub
akshaykumarakulsl0034.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username=guacadmin&password=guacadmin
Applications
akshaykumarakul@ip-1...
Tue 28 Feb, 06:43
akshaykumarakul
akshaykumarakul@ip-172-31-27-168: ~/cicd-pipeline
File
Edit
View
Search
Terminal
Help
Type here to search
32°C Sunny
ENG
12:13
28-02-2023

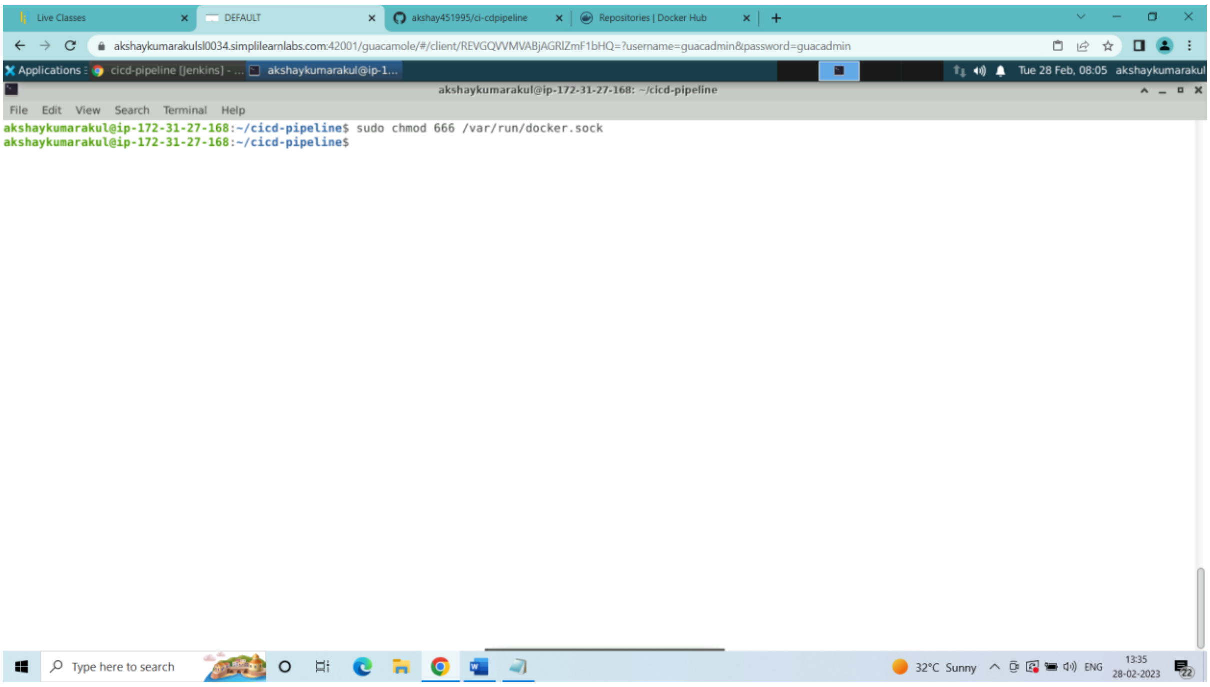
```
akshaykumarakul@ip-172-31-27-168:~$ mkdir cicd-pipeline
akshaykumarakul@ip-172-31-27-168:~$ cd cicd-pipeline
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ vi Dockerfile
```

Inside the Dockerfile:
FROM ubuntu

RUN echo "Built container successfully" > file1
(save and quit the file)

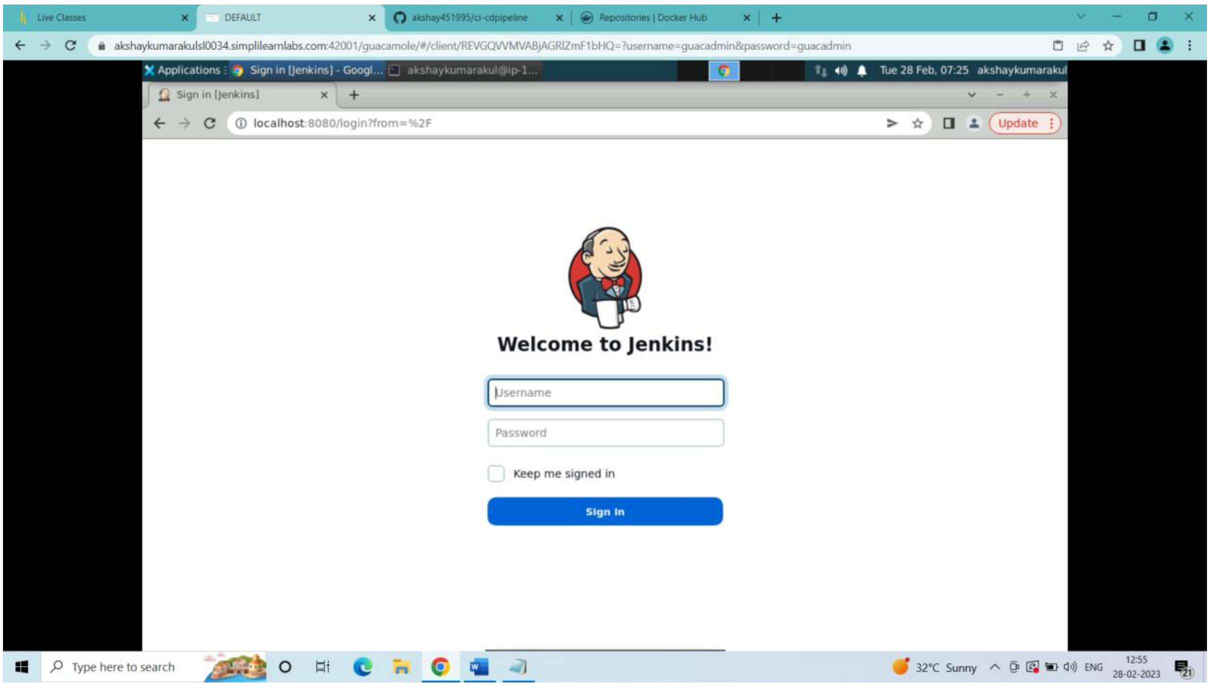
(This is the code to build an image with ubuntu as base image with a file inside it named as file1 which contains the message "Built container successfully")

9) Change the permissions of the docker.sock file
sudo chmod 666 /var/run/docker.sock



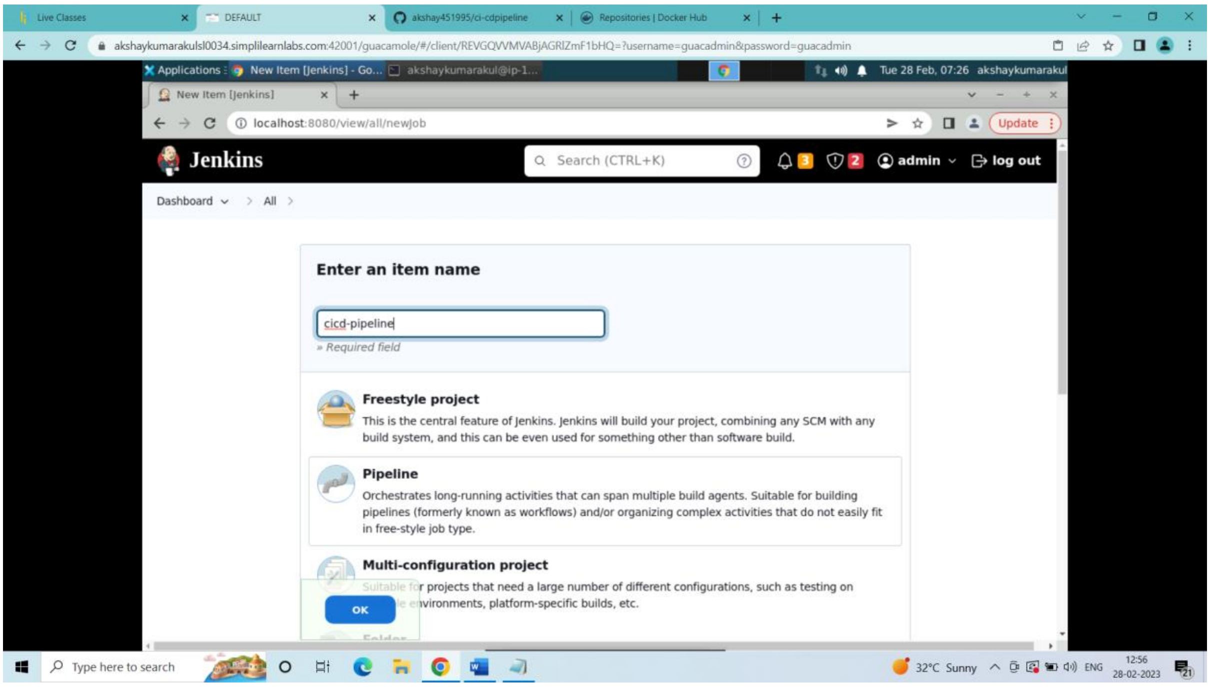
```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ sudo chmod 666 /var/run/docker.sock
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$
```

10) Opened the web browser in the simplilearn machine and in address bar type:
{{ localhost:8080 }}



login jenkins with credentials

(inside jenkins dashboard)
create a new pipeline job:
-- create new item:
select pipeline and name the pipeline ci-cdpipeline and click ok



Enter an item name

cicd-pipeline

» Required field

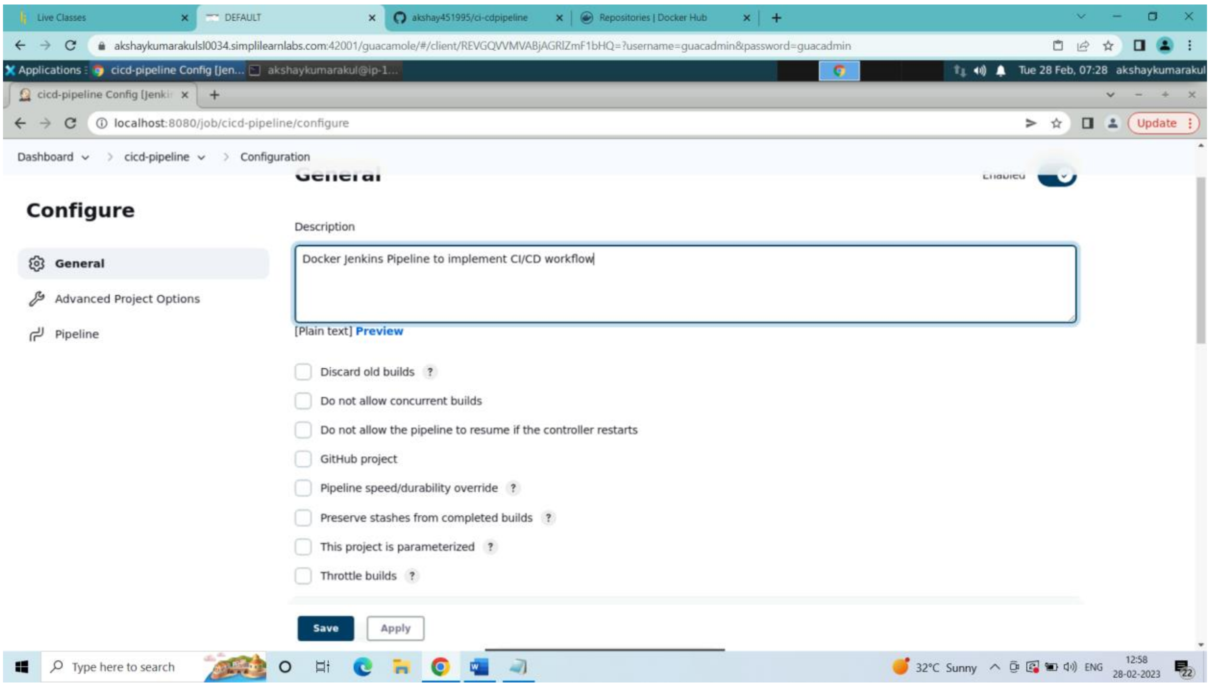
Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

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

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

11) Inside the configuration of the cicd-pipeline job:

i) Add the description of the pipeline:
Docker Jenkins Pipeline to implement CI/CD workflow



Configure

Description

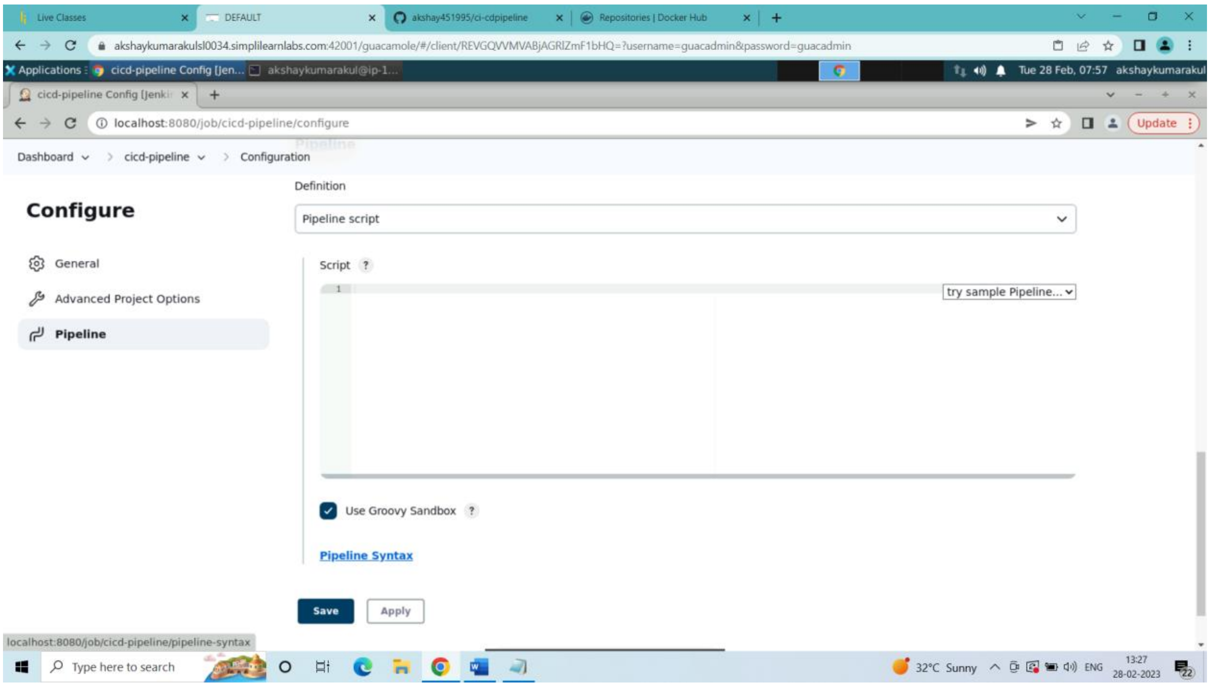
Docker Jenkins Pipeline to implement CI/CD workflow

[Plain text] Preview

□ Discard old builds
□ Do not allow concurrent builds
□ Do not allow the pipeline to resume if the controller restarts
□ GitHub project
□ Pipeline speed/durability override
□ Preserve stashes from completed builds
□ This project is parameterized
□ Throttle builds

ii) In the pipeline configure:
a) in definition: selected Pipeline script
b) clicked on Pipeline syntax to generate the syntax for login script for dockerhub login

(inside the pipeline syntax)
(i want to generate password variable for providing dockerhub password during dockerhub login)


Live Classes
DEFAULT
akshay451995/ci-cdpipeline
Repositories | Docker Hub
akshaykumarakulsl0034.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username=guacadmin&password=guacadmin
Applications
cicd-pipeline Config [Jen...
akshaykumarakul@ip-1...
Tue 28 Feb, 07:57
cicd-pipeline Config [Jenki
localhost:8080/job/cicd-pipeline/configure
Update
Dashboard
cicd-pipeline
Configuration
Configure
General
Advanced Project Options
Pipeline
try sample Pipeline...
Save
Apply
localhost:8080/job/cicd-pipeline/pipeline-syntax
Type here to search
32°C Sunny
ENG
13:27
28-02-2023

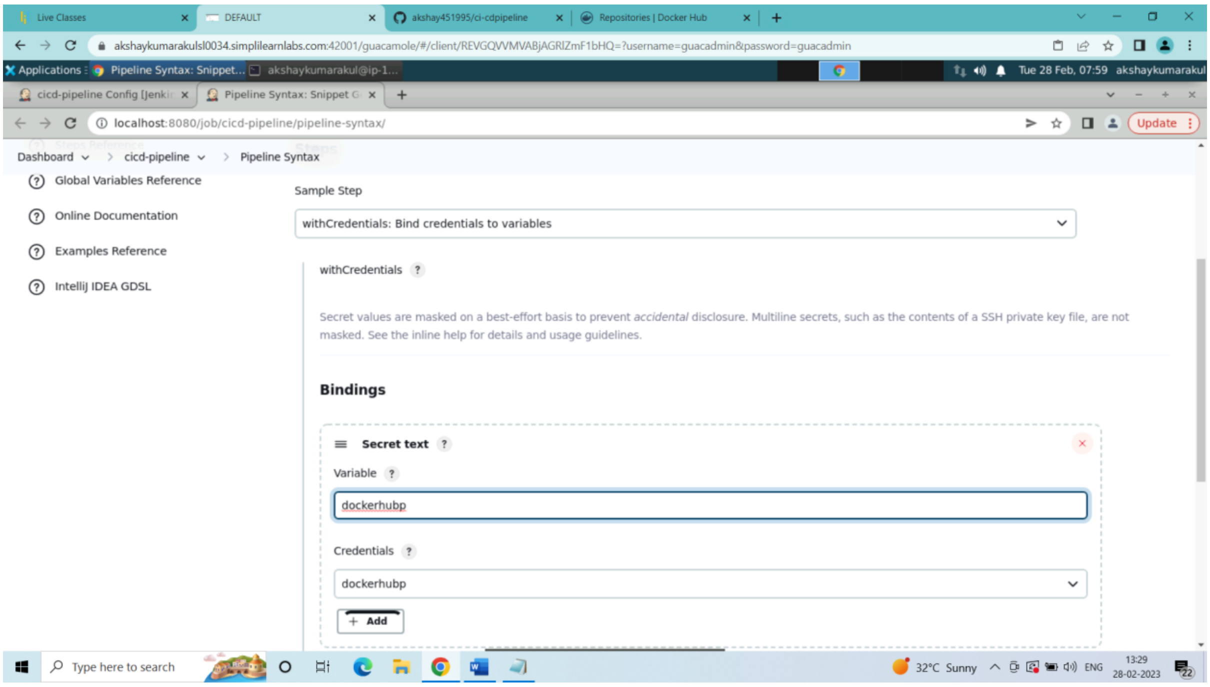
Definition
Pipeline script

Script
1

☑ Use Groovy Sandbox

Pipeline Syntax

c) clicked on Snippet Generator -- in Sample step-- select -- withCredentials: Bind credentials to variables-- in bindings -- click on add -- select Secret text --in Variable -- wrote dockerhubp--(any variable name can be given in place of dockerhubp)-- in Credentials -- click on add -- and select jenkins -- in kind-- select Secret text -- in Secret -- i have entered my dockerhub password -- in id i have given the same name as dockerhubp-- then click on add


Live Classes
DEFAULT
akshay451995/ci-cdpipeline
Repositories | Docker Hub
akshaykumarakulsl0034.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username=guacadmin&password=guacadmin
Applications
Pipeline Syntax: Snippet...
akshaykumarakul@ip-1...
Tue 28 Feb, 07:59
cicd-pipeline Config [Jenki
Pipeline Syntax: Snippet G
localhost:8080/job/cicd-pipeline/pipeline-syntax/
Update
Dashboard
cicd-pipeline
Pipeline Syntax
Global Variables Reference
Online Documentation
Examples Reference
IntelliJ IDEA GDSL
+ Add
Type here to search
32°C Sunny
ENG
13:29
28-02-2023

Sample Step
withCredentials: Bind credentials to variables

withCredentials

Secret values are masked on a best-effort basis to prevent *accidental* disclosure. Multiline secrets, such as the contents of a SSH private key file, are not masked. See the inline help for details and usage guidelines.

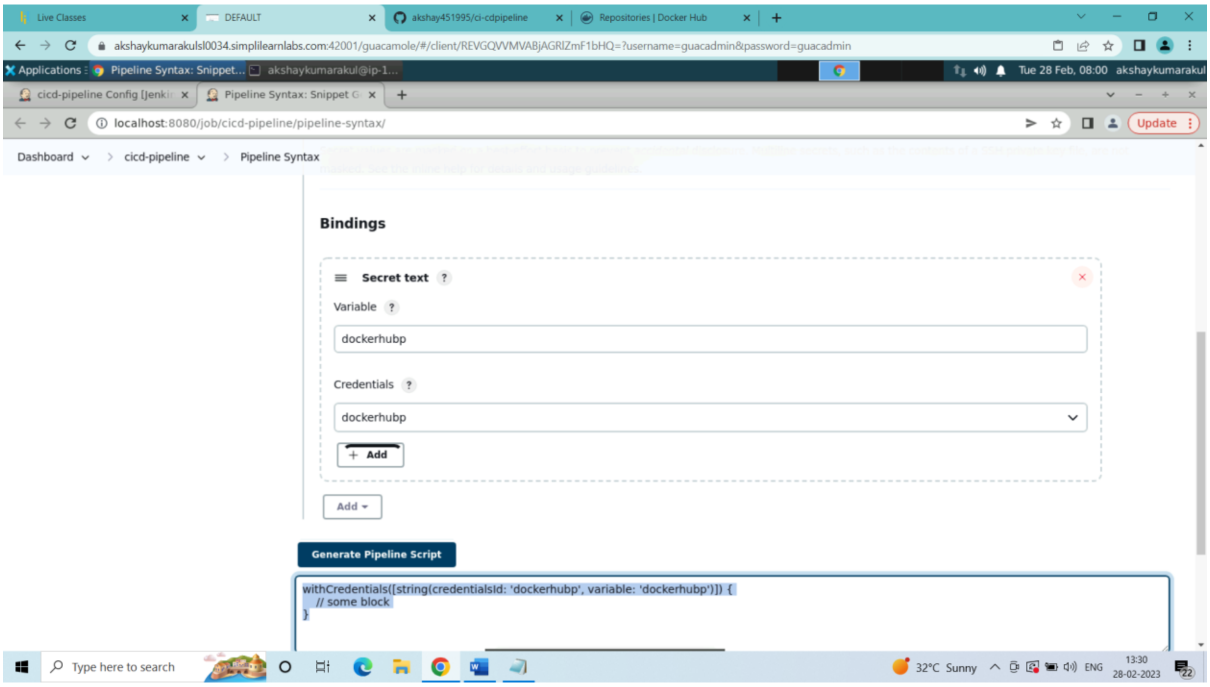
Bindings

Secret text

Variable
dockerhubp

Credentials
dockerhubp

inside the snippet generator click on Generate Pipeline Script below the bindings-- u get a script for performing the docker hub login for your scripted pipeline -- add this script to your pipeline script in the step of performing dockerhub login--



Bindings

Secret text

Variable: dockerhubp

Credentials: dockerhubp

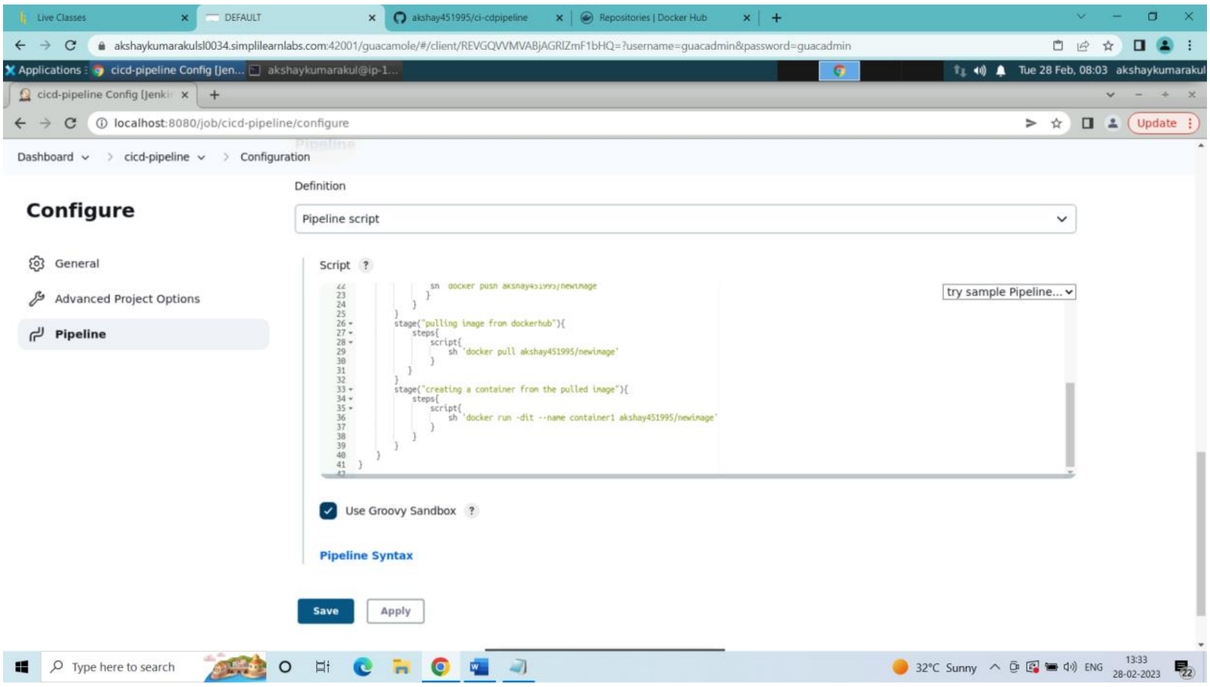
```
withCredentials([string(credentialsId: 'dockerhubp', variable: 'dockerhubp')]) {
    // some block
}
```

d) inside the pipeline script wrote the following script:

```
pipeline{
  agent any
  stages{
    stage("build docker image from dockerfile"){
      steps{
        script{
          sh 'docker build -t akshay451995/newimage .'
        }
      }
    }
    stage("push image to docker hub"){
      steps{
        script{
          withCredentials([string(credentialsId: 'dockerhubp', variable: 'dockerhubp')]) {
            sh 'docker login -u akshay451995 -p ${dockerhubp}'
}
        sh 'docker push akshay451995/newimage'
        }
      }
    }
    stage("pulling image from dockerhub"){
      steps{
        script{
          sh 'docker pull akshay451995/newimage'
        }
      }
    }
```

```
        stage("creating a container from the pulled image"){
            steps{
                script{
                    sh 'docker run -dit --name container10 akshay451995/newimage'
                }
            }
        }
    }
}
```

(This is the pipeline script to create and build a docker image using the Dockerfile from github repository, login to the dockerhub account and push the image to dockerhub repository, pull the image from the dockerhub repository and create a container using the image)



Dashboard > cicd-pipeline > Configuration

Configure

Definition: Pipeline script

Script

```
                    sh 'docker push akshay451995/newimage'
                }
            }
        }
        stage("pulling image from dockerhub"){
            steps{
                script{
                    sh 'docker pull akshay451995/newimage'
                }
            }
        }
        stage("creating a container from the pulled image"){
            steps{
                script{
                    sh 'docker run -dit --name container1 akshay451995/newimage'
                }
            }
        }
    }
}
```

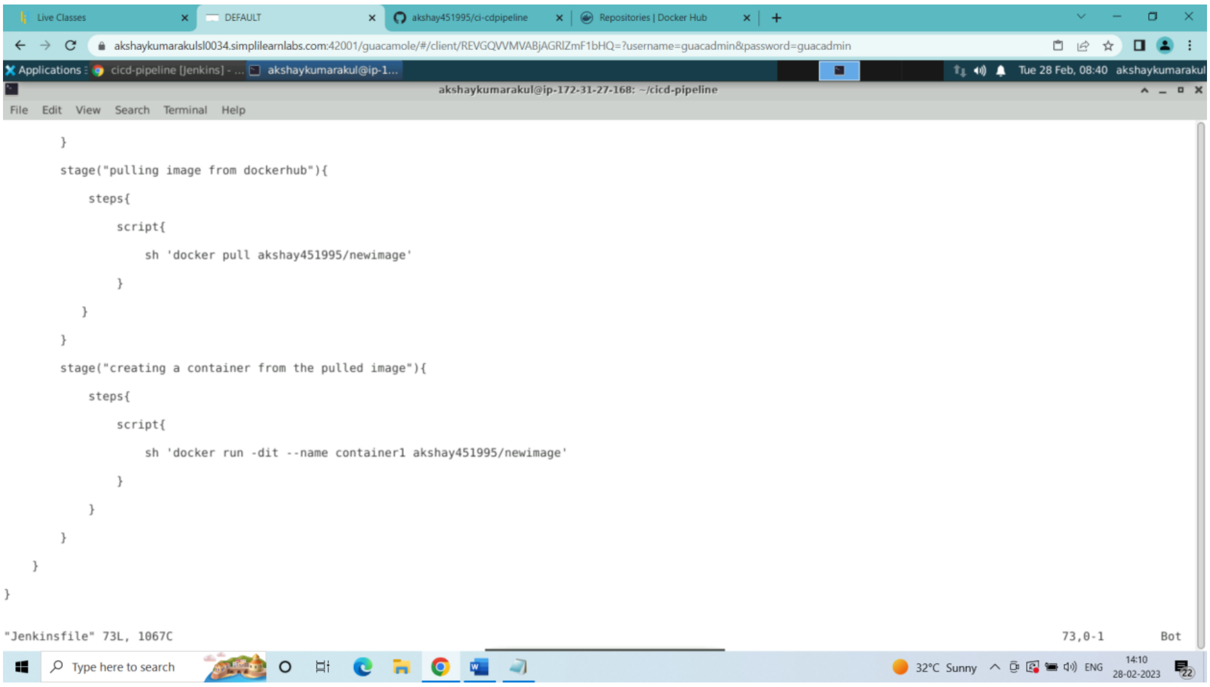
☑ Use Groovy Sandbox

12) Copied the pipeline script to Jenkinsfile created in the ubuntu machine terminal inside the cicd-pipeline directory:

```
vi Jenkinsfile

pipeline{
    agent any
    stages{
        stage("build docker image from dockerfile"){
            steps{
                script{
                    sh 'docker build -t akshay451995/newimage .'
                }
```

```
          }
        }
        stage("push image to docker hub"){
          steps{
            script{
              withCredentials([string(credentialsId: 'dockerhubp', variable: 'dockerhubp')]) {
                sh 'docker login -u akshay451995 -p ${dockerhubp}'
}
            sh 'docker push akshay451995/newimage'
            }
          }
        }
        stage("pulling image from dockerhub"){
          steps{
            script{
              sh 'docker pull akshay451995/newimage'
            }
         }
        }
        stage("creating a container from the pulled image"){
          steps{
            script{
              sh 'docker run -dit --name container10 akshay451995/newimage'
            }
          }
        }
    }
}
```

(saved and quit the file)


akshaykumarakul@ip-172-31-27-168: ~/cicd-pipeline
File Edit View Search Terminal Help

```
        }
        stage("pulling image from dockerhub"){
            steps{
                script{
                    sh 'docker pull akshay451995/newimage'
                }
            }
        }
        stage("creating a container from the pulled image"){
            steps{
                script{
                    sh 'docker run -dit --name container1 akshay451995/newimage'
                }
            }
        }
    }
}
"Jenkinsfile" 73L, 1067C                                        73,0-1        Bot
```

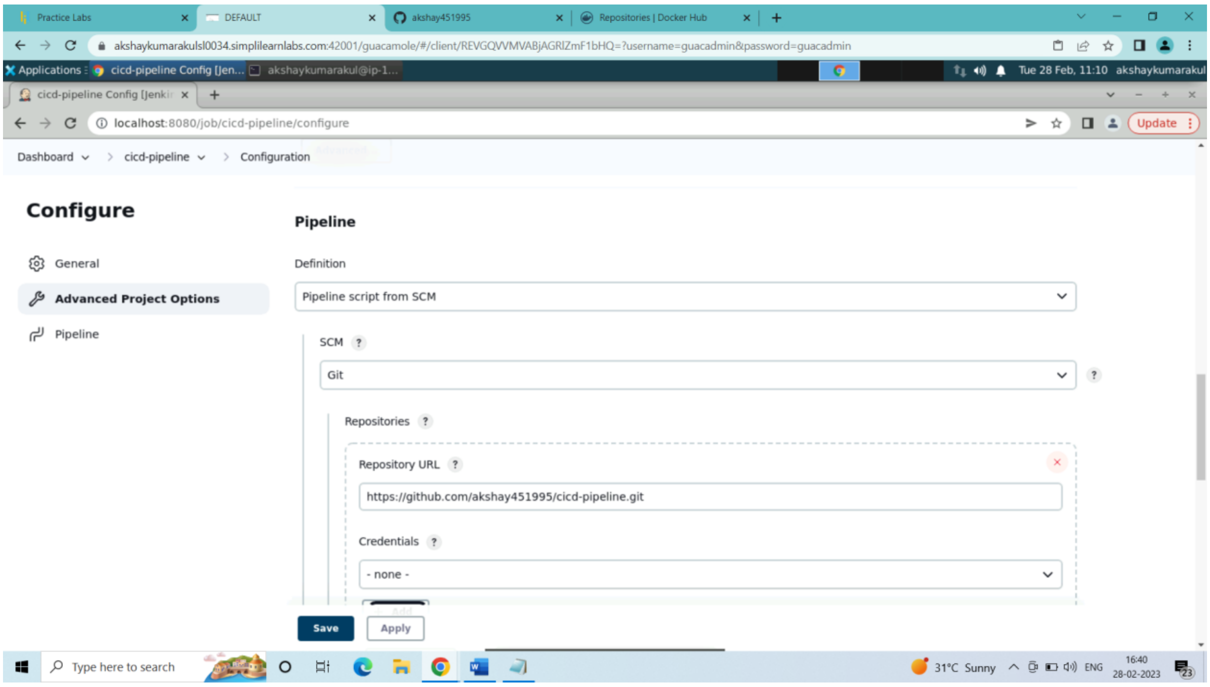
13) Inside the Jenkins cicd-pipeline job configure:

In the Pipeline configure:

a) in definition: selected Pipeline script from SCM

b) in SCM: selected git

c) in repositories URL: https://github.com/akshay451995/cicd-pipeline.git


localhost:8080/job/cicd-pipeline/configure
Dashboard > cicd-pipeline > Configuration
Configure
General
Advanced Project Options
Pipeline
Save
Apply

Pipeline

Definition: Pipeline script from SCM

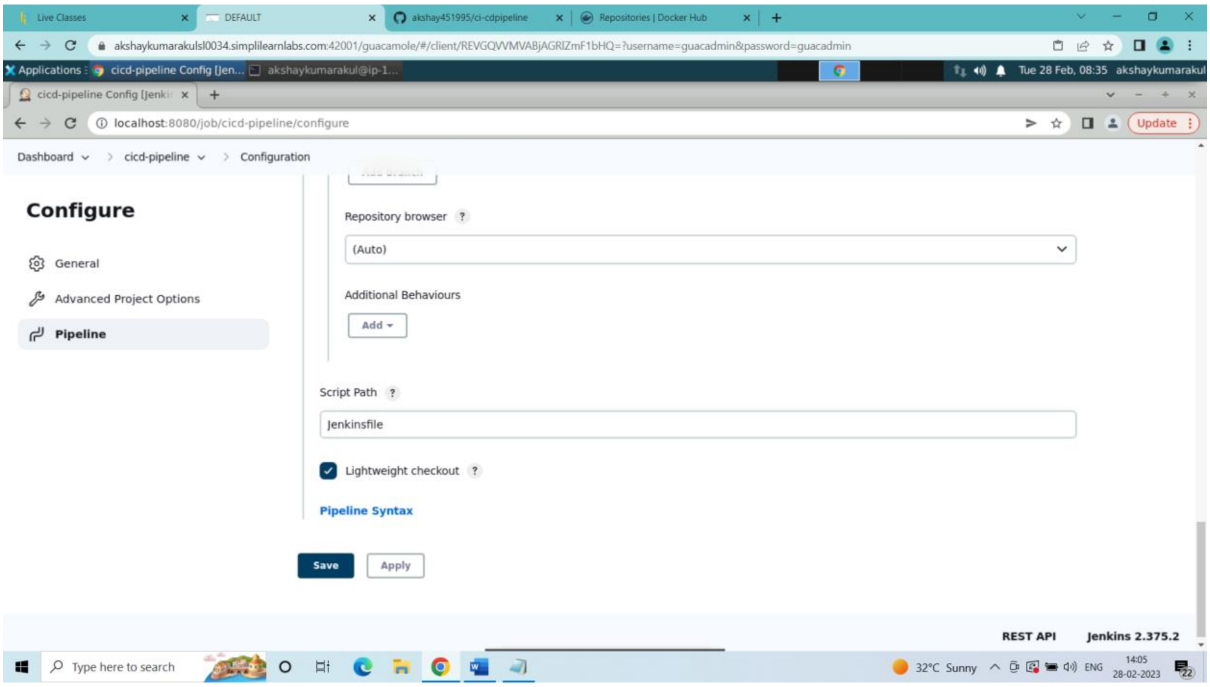
SCM: Git

Repositories

Repository URL: https://github.com/akshay451995/cicd-pipeline.git

Credentials: - none -

d) in Script Path: Jenkinsfile



Repository browser: (Auto)

Script Path: Jenkinsfile

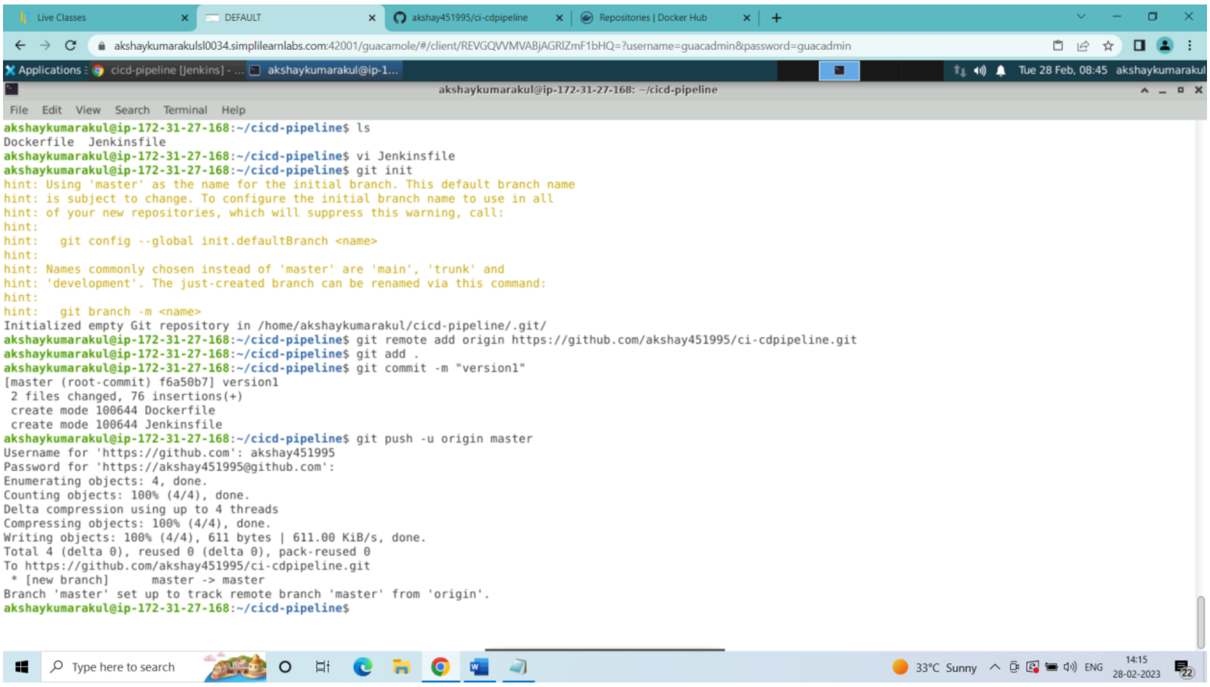
☑ Lightweight checkout

Now clicked on save and saved the cicd-pipeline job configuration.

14) Now inside the simplilearn ubuntu machine terminal:
Inside the directory cicd-pipeline, initiated a git repository:

```
git init
git remote add origin https://github.com/akshay451995/cicd-pipeline.git
git add .
git commit -m “version1”
git push -u origin master
```

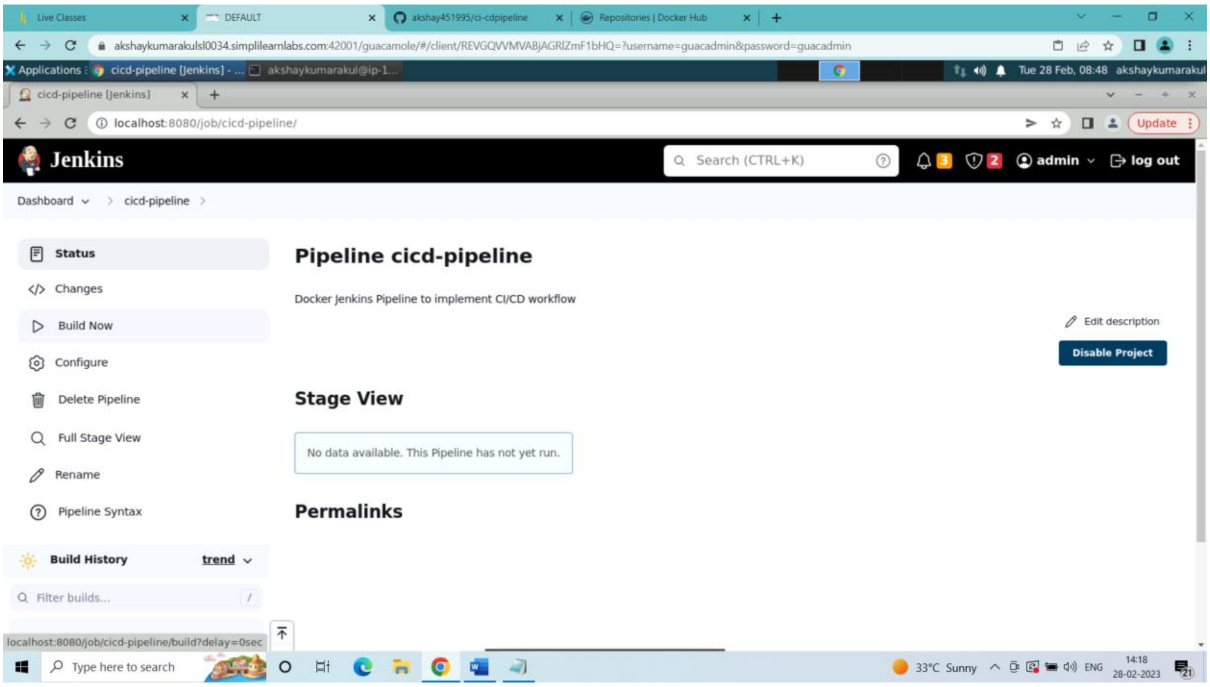
(wrote username and password for the github account in the username and password prompt)



Live Classes
DEFAULT
akshay451995/ci-cdpipeline
Repositories | Docker Hub
akshaykumarakulsl0034.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username=guacadmin&password=guacadmin
Applications
cicd-pipeline [Jenkins] - ...
akshaykumarakul@ip-1...
Tue 28 Feb, 08:45 akshaykumarakul
akshaykumarakul@ip-172-31-27-168: ~/cicd-pipeline
File Edit View Search Terminal Help
Type here to search
33°C Sunny
ENG
14:15
28-02-2023

```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ ls
Dockerfile  Jenkinsfile
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ vi Jenkinsfile
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/akshaykumarakul/cicd-pipeline/.git/
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git remote add origin https://github.com/akshay451995/ci-cdpipeline.git
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git add .
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git commit -m "version1"
[master (root-commit) f6a50b7] version1
 2 files changed, 76 insertions(+)
 create mode 100644 Dockerfile
 create mode 100644 Jenkinsfile
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git push -u origin master
Username for 'https://github.com': akshay451995
Password for 'https://akshay451995@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 611 bytes | 611.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/akshay451995/ci-cdpipeline.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$
```

15) Now inside the Jenkins dashboard cicd-pipeline job clicked on build now to execute the pipeline job:



Pipeline cicd-pipeline

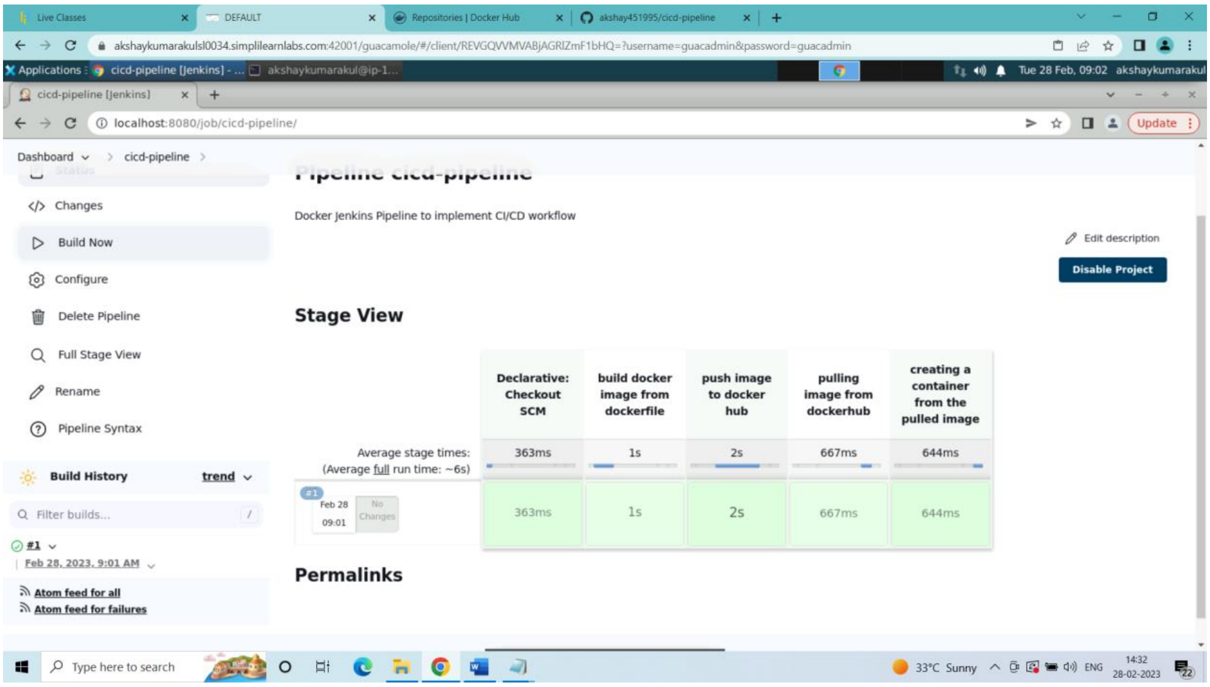
Docker Jenkins Pipeline to implement CI/CD workflow

Stage View

No data available. This Pipeline has not yet run.

Permalinks

Pipeline Job is executed:



localhost:8080/job/cicd-pipeline/

Dashboard > cicd-pipeline >

Pipeline cicd-pipeline

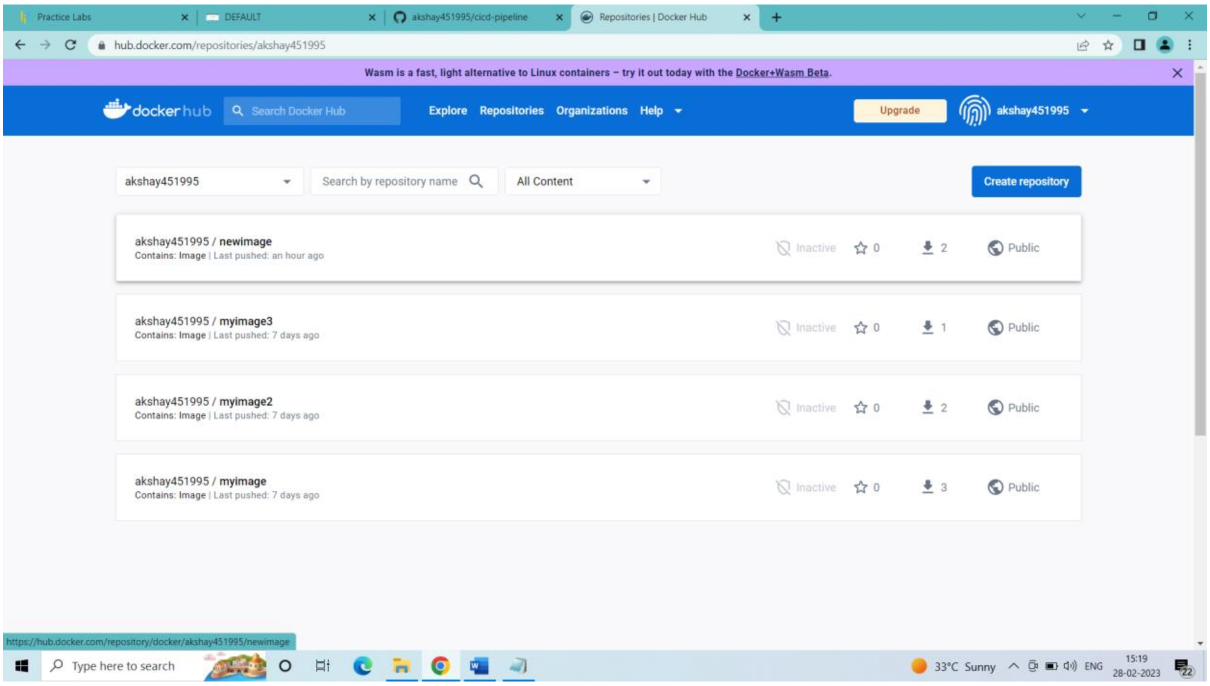
Docker Jenkins Pipeline to implement CI/CD workflow

Stage View

	Declarative: Checkout SCM	build docker image from dockerfile	push image to docker hub	pulling image from dockerhub	creating a container from the pulled image
Average stage times: (Average full run time: ~6s)	363ms	1s	2s	667ms	644ms
#1 Feb 28 09:01 No Changes	363ms	1s	2s	667ms	644ms

#1 Feb 28, 2023, 9:01 AM

16) Checked if the image was created in the ubuntu machine with the container and also in the dockerhub

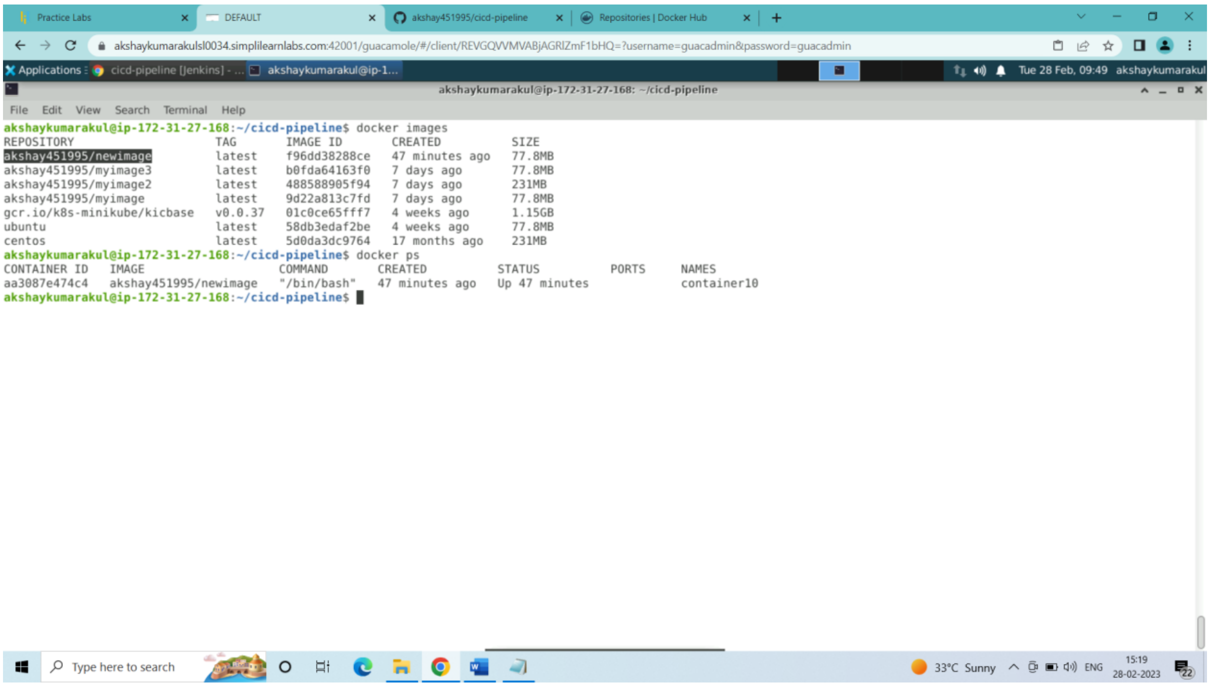


hub.docker.com/repositories/akshay451995

Wasm is a fast, light alternative to Linux containers – try it out today with the Docker+Wasm Beta.

akshay451995 / newimage Contains: Image | Last pushed: an hour ago	Inactive	0	2	Public
akshay451995 / myimage3 Contains: Image | Last pushed: 7 days ago	Inactive	0	1	Public
akshay451995 / myimage2 Contains: Image | Last pushed: 7 days ago	Inactive	0	2	Public
akshay451995 / myimage Contains: Image | Last pushed: 7 days ago	Inactive	0	3	Public

https://hub.docker.com/repository/docker/akshay451995/newimage



```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
akshay451995/newimage         latest    f96dd38288ce   47 minutes ago   77.8MB
akshay451995/myimage3         latest    b0fda64163f0   7 days ago       77.8MB
akshay451995/myimage2         latest    488588905f94   7 days ago       231MB
akshay451995/myimage          latest    9d22a813c7fd   7 days ago       77.8MB
gcr.io/k8s-minikube/kicbase   v0.0.37   01c0ce65fff7   4 weeks ago      1.15GB
ubuntu                        latest    58db3edaf2be   4 weeks ago      77.8MB
centos                        latest    5d0da3dc9764   17 months ago    231MB
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ docker ps
CONTAINER ID   IMAGE                   COMMAND       CREATED          STATUS          PORTS     NAMES
aa3087e474c4   akshay451995/newimage   "/bin/bash"   47 minutes ago   Up 47 minutes             container10
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ 
```

17) Now I will configure the Jenkins in such a way that for every commit made to github the Jenkins runs the pipeline project:

Open the configuration of the cicd-pipeline project in Jenkins:

select the build triggers:

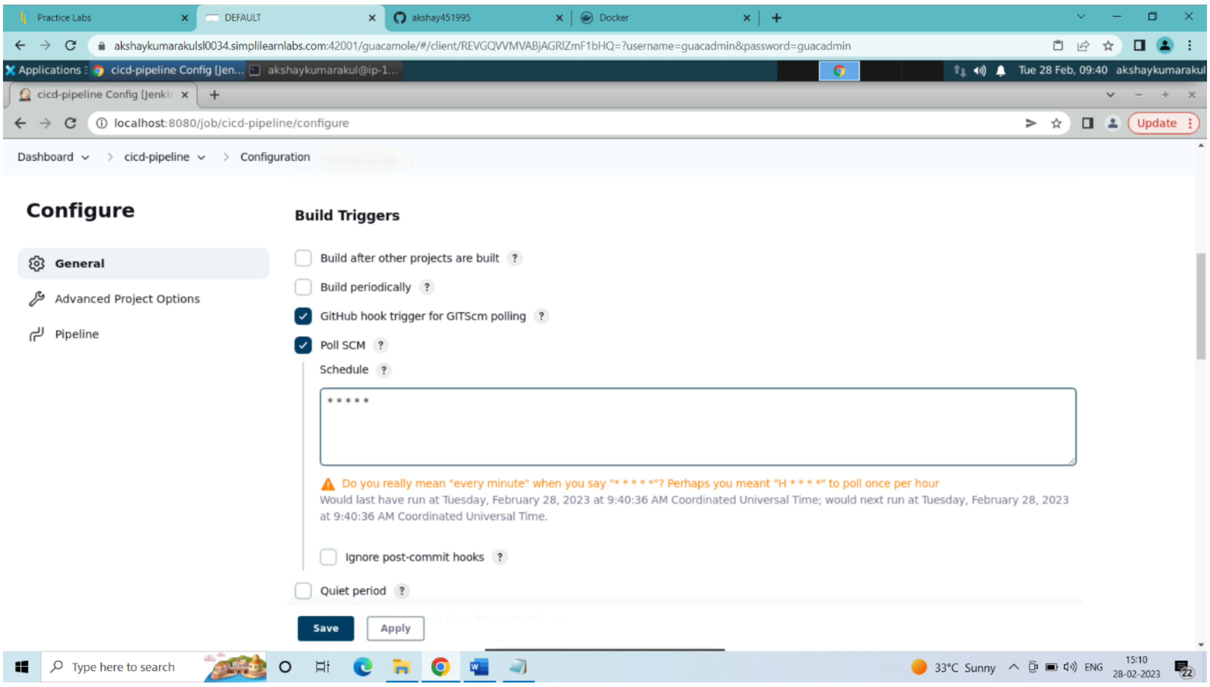
a) GitHub hook trigger for GITScm polling?

b) Poll SCM

(inside the Poll SCM)

Schedule

* * * * *



Configure

Build Triggers

□ Build after other projects are built

□ Build periodically

☑ GitHub hook trigger for GITScm polling

☑ Poll SCM

Schedule

```
* * * * *
```

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour

Would last have run at Tuesday, February 28, 2023 at 9:40:36 AM Coordinated Universal Time; would next run at Tuesday, February 28, 2023 at 9:40:36 AM Coordinated Universal Time.

□ Ignore post-commit hooks

□ Quiet period

c) click on save

18) Now to test the ci/cd pipeline by making changes in the terminal to the Dockerfile and Jenkinsfile and commit to github repository, the pipeline executes the next build in about 1 minute of the commit

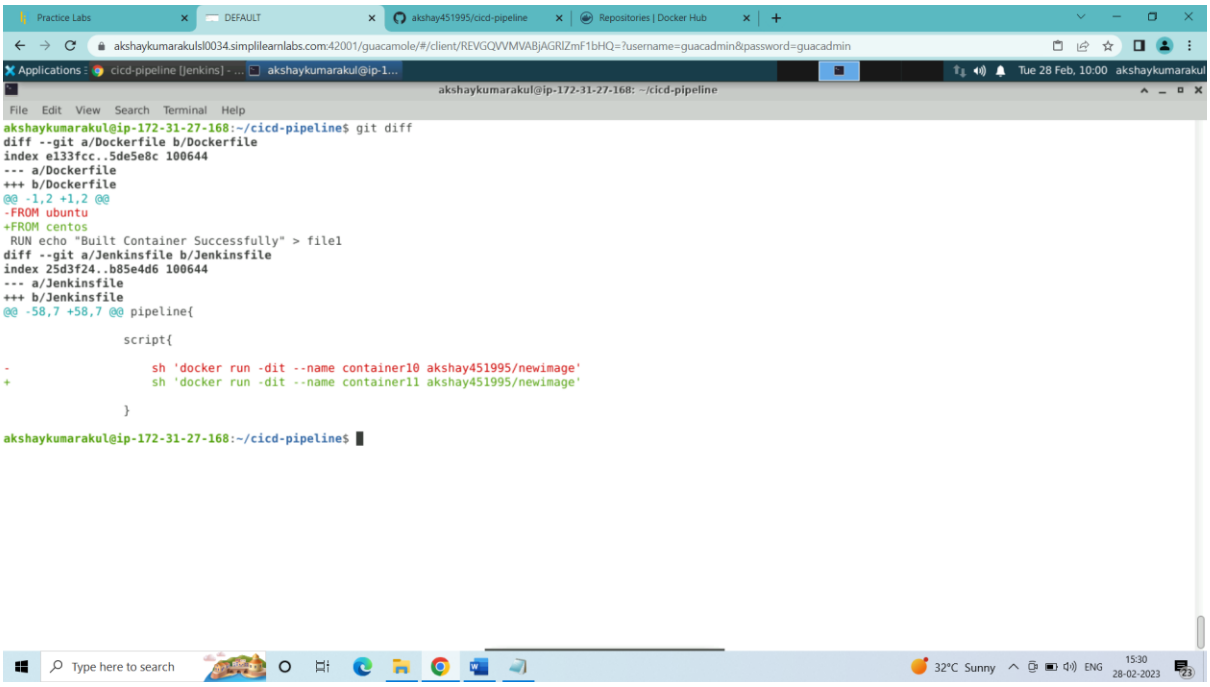
Changes I have done to the Dockerfile and Jenkinsfile:

vi Dockerfile

(inside the Dockerfile I have changed the base image form ubuntu to centos)

vi Jenkinsfile

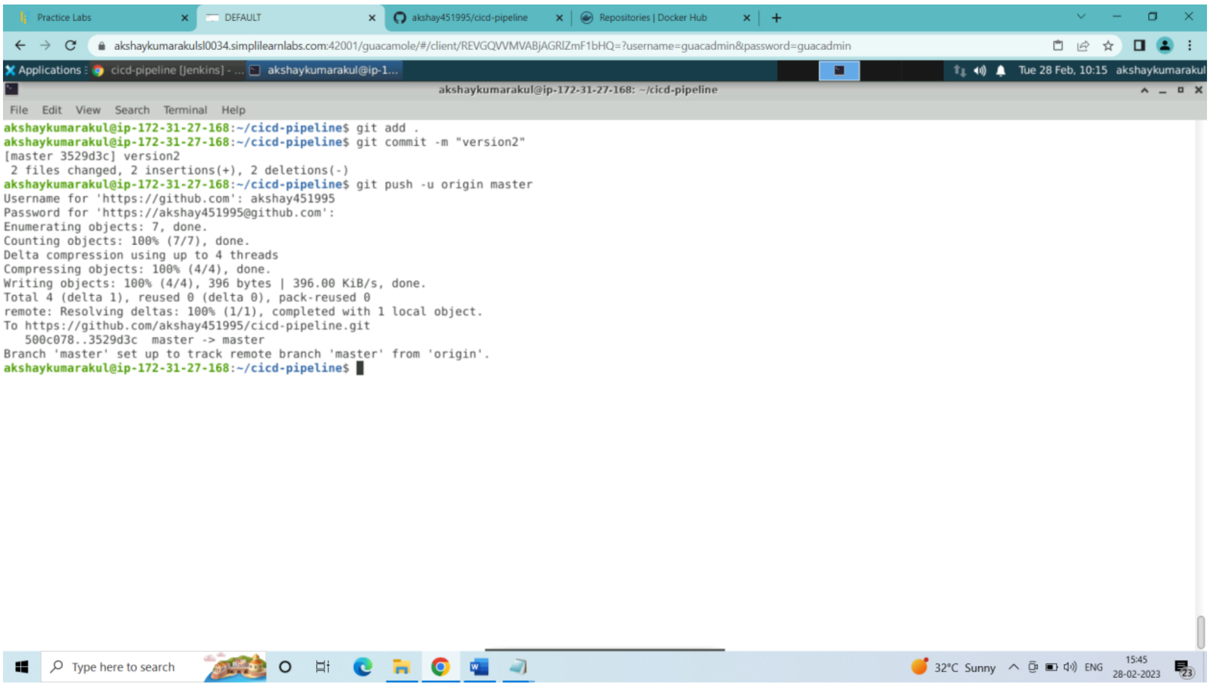
(inside the Jenkinsfile I have changed the container name from container10 to container11)



```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git diff
diff --git a/Dockerfile b/Dockerfile
index e133fcc..5de5e8c 100644
--- a/Dockerfile
+++ b/Dockerfile
@@ -1,2 +1,2 @@
-FROM ubuntu
+FROM centos
 RUN echo "Built Container Successfully" > file1
diff --git a/Jenkinsfile b/Jenkinsfile
index 25d3f24..b85e4d6 100644
--- a/Jenkinsfile
+++ b/Jenkinsfile
@@ -58,7 +58,7 @@ pipeline{

                 script{

-                    sh 'docker run -dit --name container10 akshay451995/newimage'
+                    sh 'docker run -dit --name container11 akshay451995/newimage'

                 }

akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ 
```

19) Now added and commited the changes to the github repository:

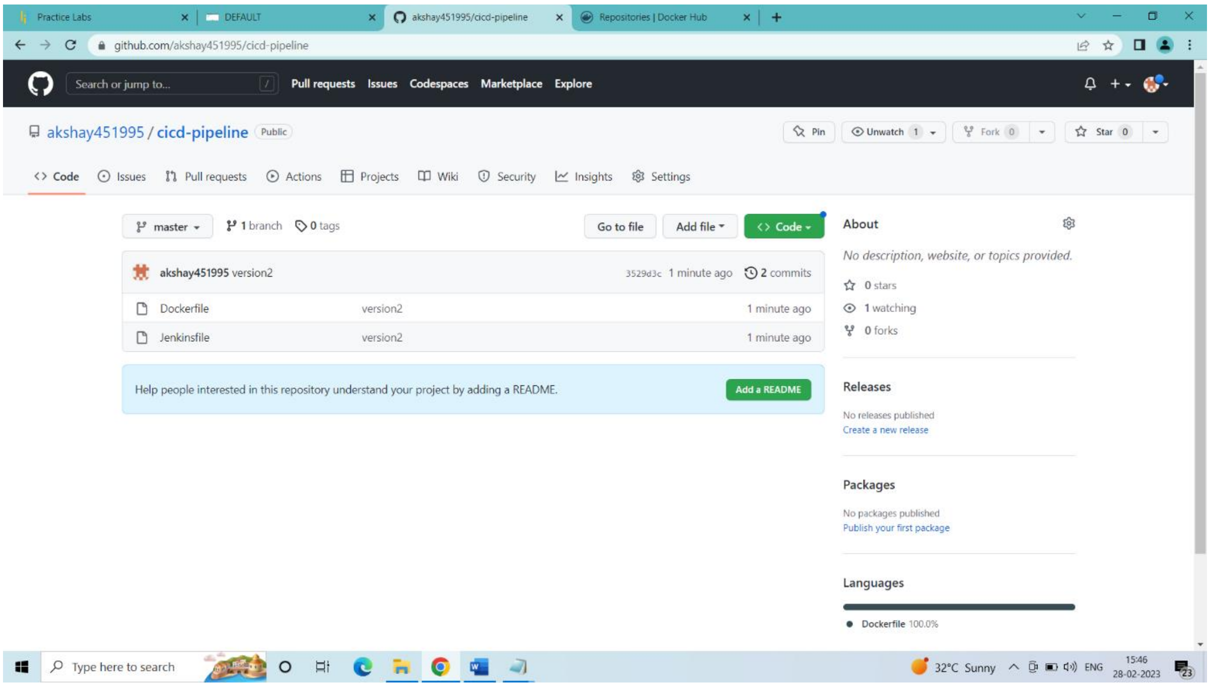
git add .
git commit -m “version2”
git push -u origin master
(wrote username and password for the github account in the username and password prompt)



```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git add .
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git commit -m "version2"
[master 3529d3c] version2
 2 files changed, 2 insertions(+), 2 deletions(-)
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ git push -u origin master
Username for 'https://github.com': akshay451995
Password for 'https://akshay451995@github.com':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 396 bytes | 396.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/akshay451995/cicd-pipeline.git
   500c078..3529d3c  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ 
```

Changes in the github Repository:

Commit version2 has been done



github.com/akshay451995/cicd-pipeline

akshay451995 / cicd-pipeline Public

master · 1 branch · 0 tags

akshay451995 version2 · 3529d3c 1 minute ago · 2 commits

Dockerfile	version2	1 minute ago
Jenkinsfile	version2	1 minute ago

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

About

No description, website, or topics provided.

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

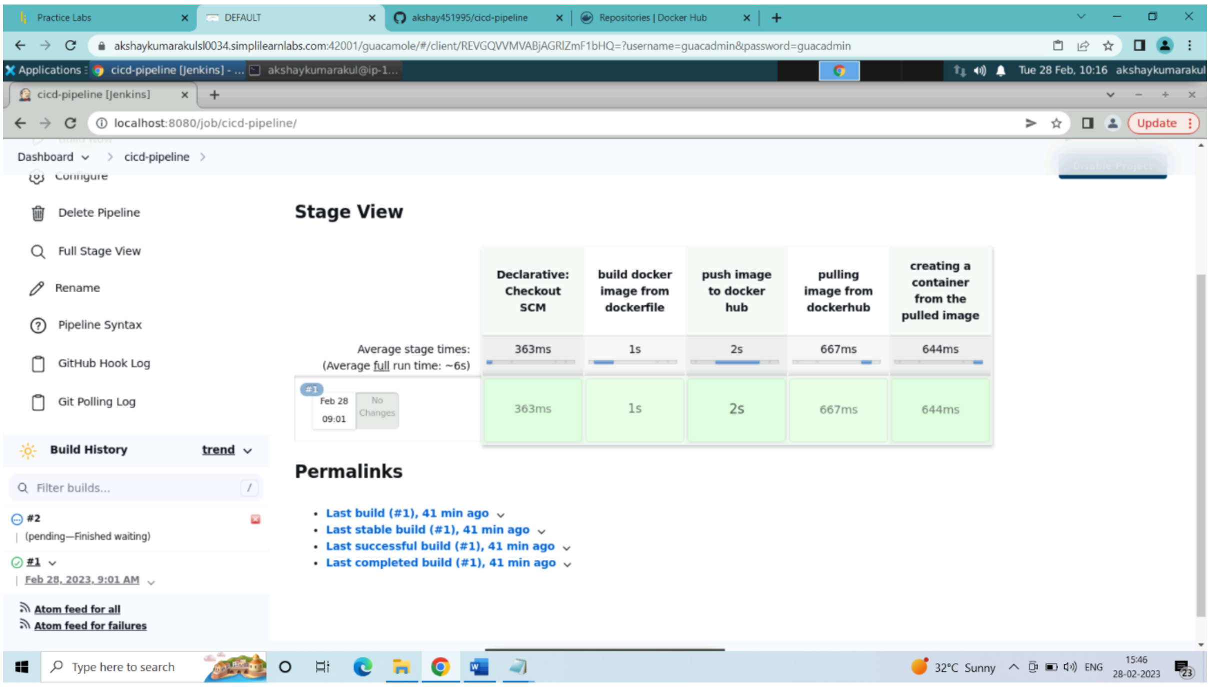
No packages published

Publish your first package

Languages

Dockerfile 100.0%

Pipeline is being executed in Jenkins automatically in about 1 minute after the commit:



localhost:8080/job/cicd-pipeline/

Dashboard > cicd-pipeline >

Stage View

	Declarative: Checkout SCM	build docker image from dockerfile	push image to docker hub	pulling image from dockerhub	creating a container from the pulled image
Average stage times: (Average full run time: ~6s)	363ms	1s	2s	667ms	644ms
#1 Feb 28 09:01 No Changes	363ms	1s	2s	667ms	644ms

Permalinks

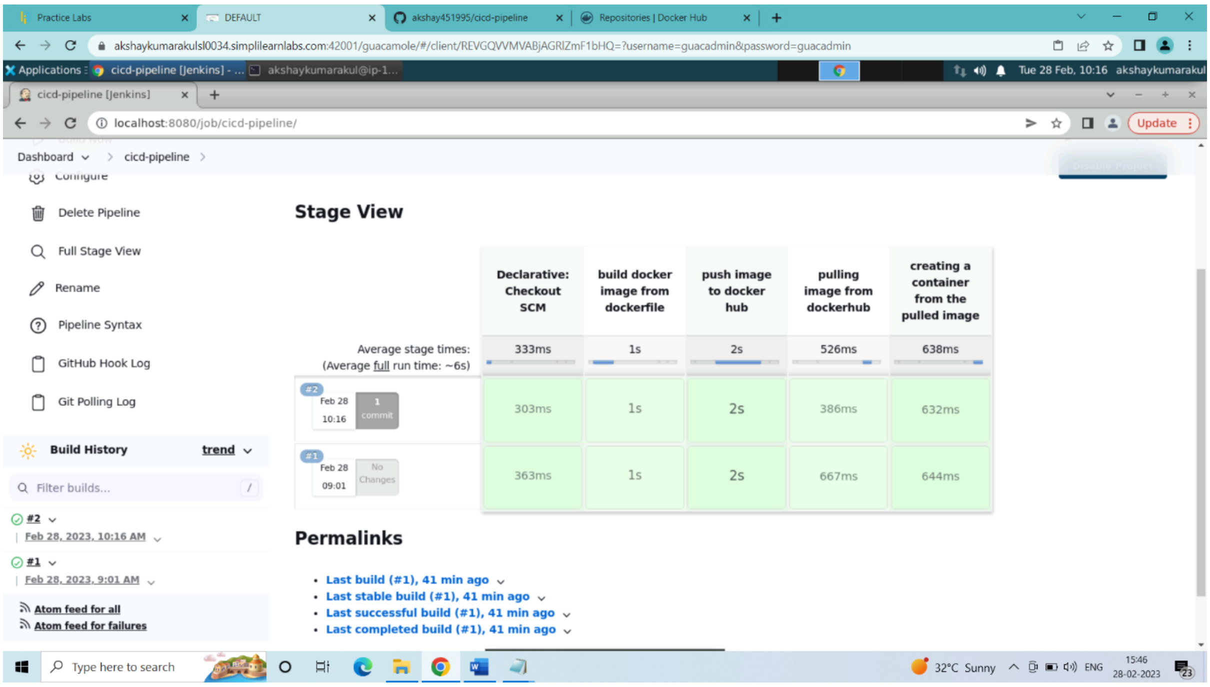
- Last build (#1), 41 min ago
- Last stable build (#1), 41 min ago
- Last successful build (#1), 41 min ago
- Last completed build (#1), 41 min ago

Build History

#2 (pending—Finished waiting)

#1 Feb 28, 2023, 9:01 AM

Pipeline is executed successfully



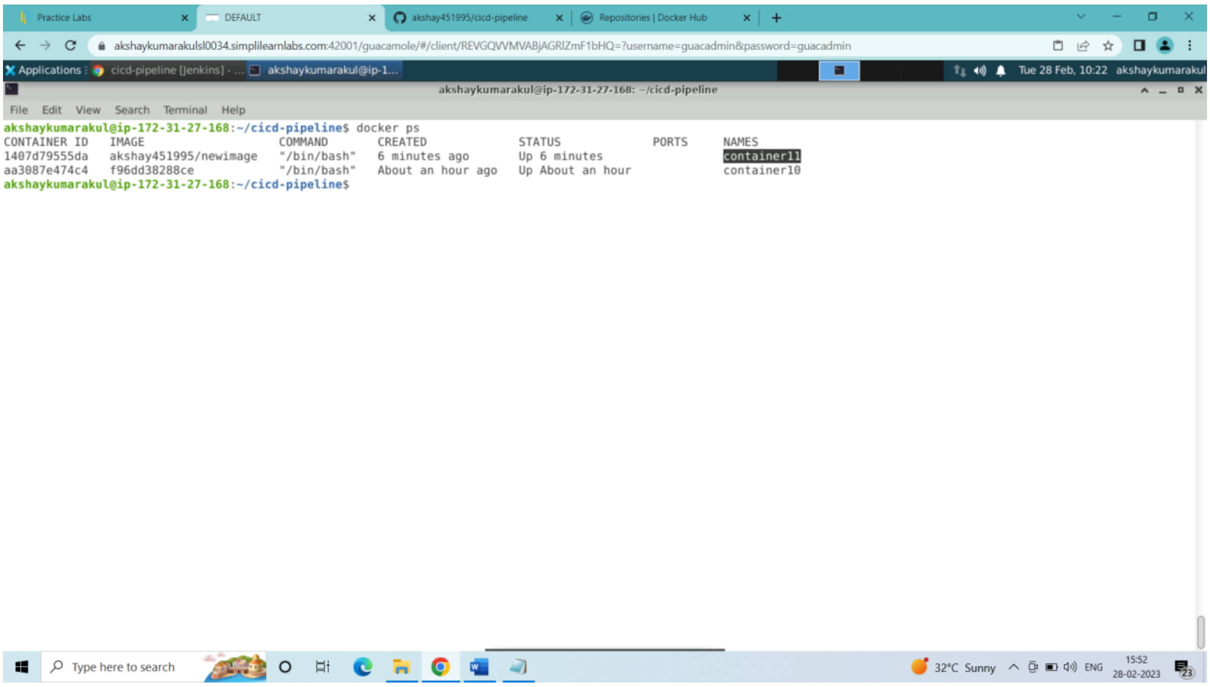
Stage View

	Declarative: Checkout SCM	build docker image from dockerfile	push image to docker hub	pulling image from dockerhub	creating a container from the pulled image
Average stage times: (Average full run time: ~6s)	333ms	1s	2s	526ms	638ms
#2 Feb 28 10:16 1 commit	303ms	1s	2s	386ms	632ms
#1 Feb 28 09:01 No Changes	363ms	1s	2s	667ms	644ms

Permalinks

- Last build (#1), 41 min ago
- Last stable build (#1), 41 min ago
- Last successful build (#1), 41 min ago
- Last completed build (#1), 41 min ago

Now check for the new container in the terminal:



```
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$ docker ps
CONTAINER ID   IMAGE                     COMMAND       CREATED              STATUS              PORTS     NAMES
1407d79555da   akshay451995/newimage     "/bin/bash"   6 minutes ago        Up 6 minutes                  container11
aa3087e474c4   f96dd38288ce              "/bin/bash"   About an hour ago    Up About an hour              container10
akshaykumarakul@ip-172-31-27-168:~/cicd-pipeline$
```

New container was built successfully

SUCCESSFULLY COMPLETED BUILDING DOCKER JENKINS PIPELINE TO IMPLEMENT CI/CD WORKFLOW