**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

1. Tools required in the machine for Docker-Jenkins pipeline:

Docker, Git, Jenkins

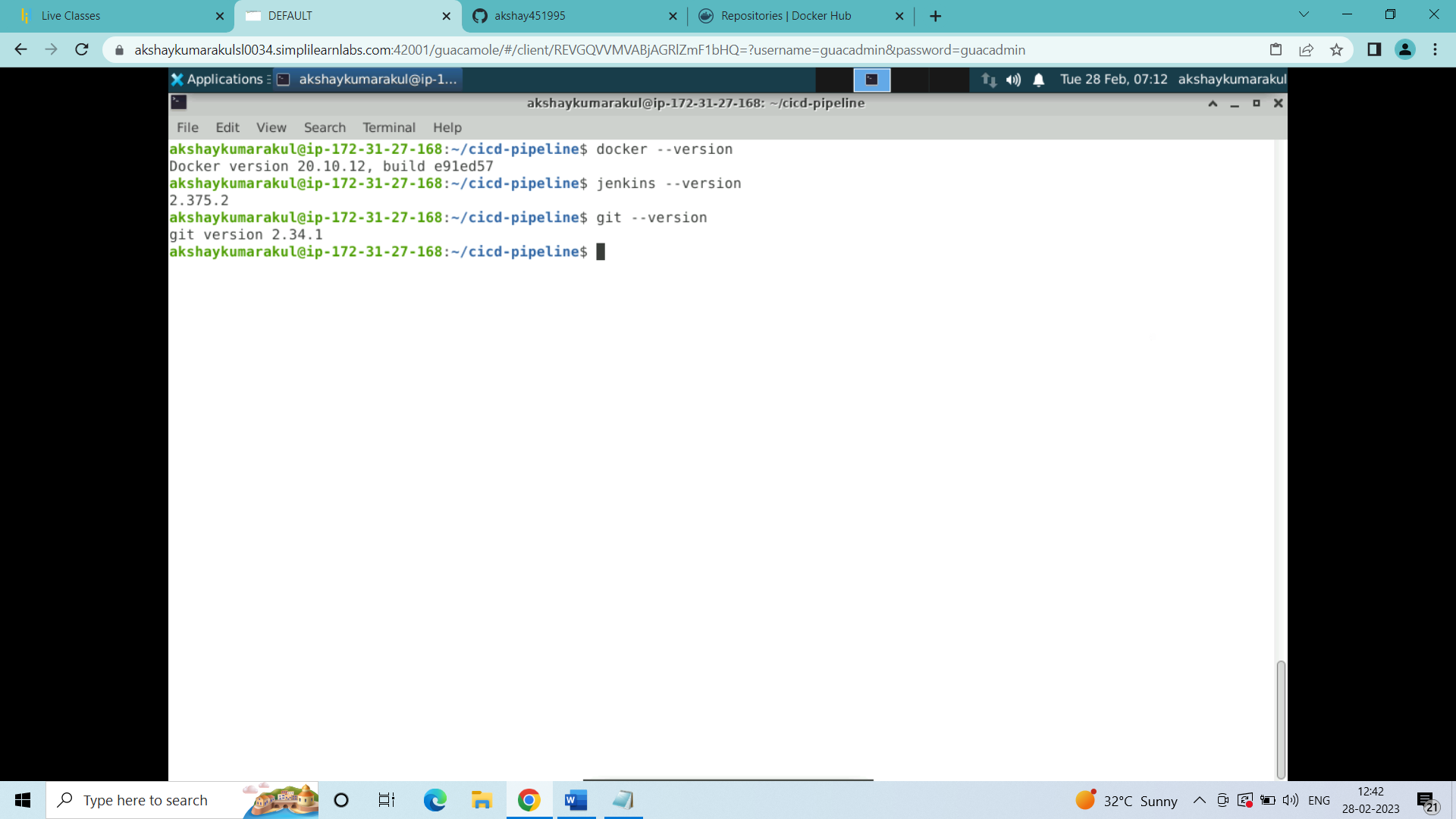
1. Docker, Jenkins and Git are already installed in the machine:

Used the following commands to check the versions of the docker, jenkins and git:

docker –version

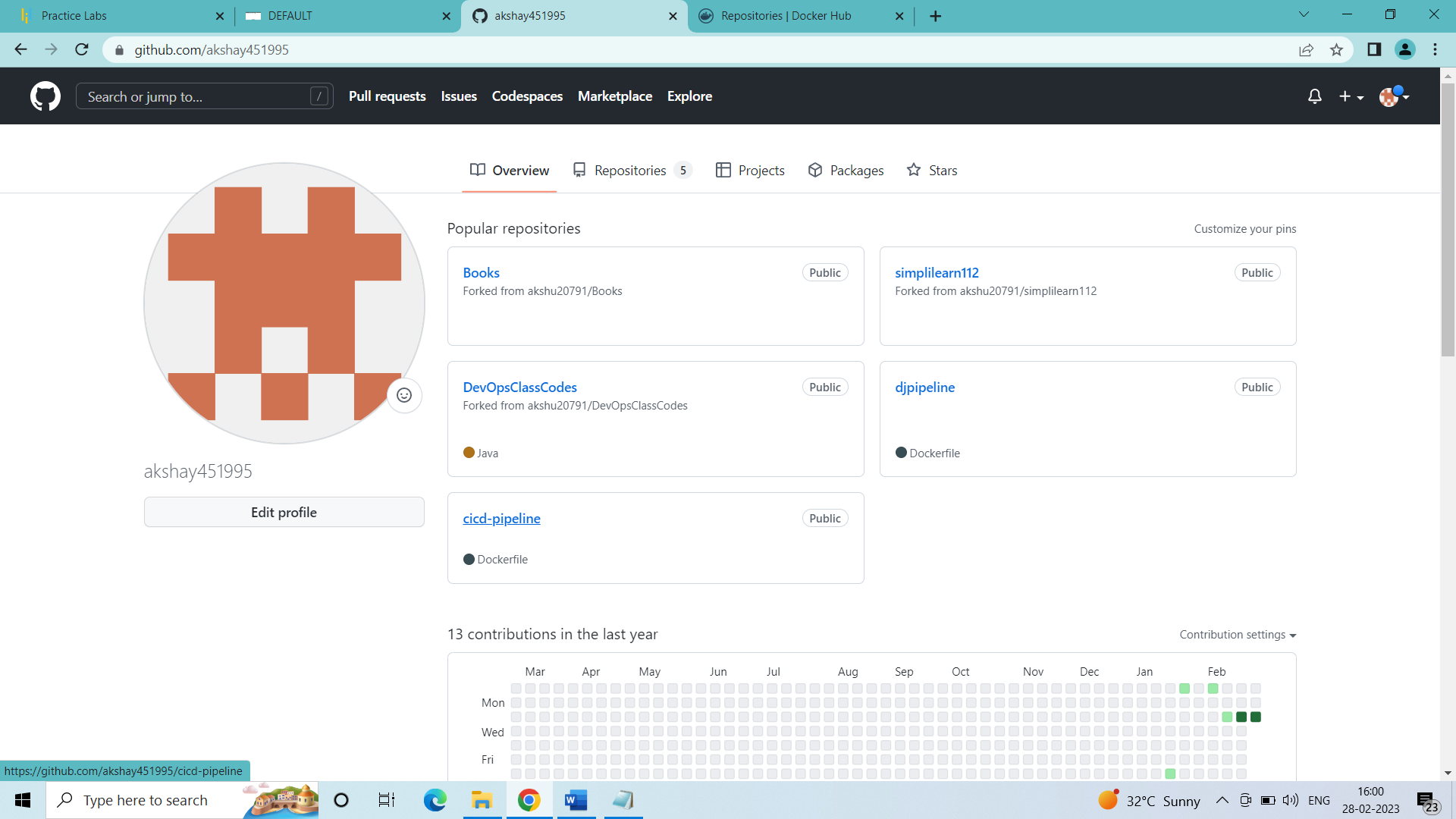
jenkins –version

git --version



1. Created a new repository in github account

github repository name: cicd-pipeline

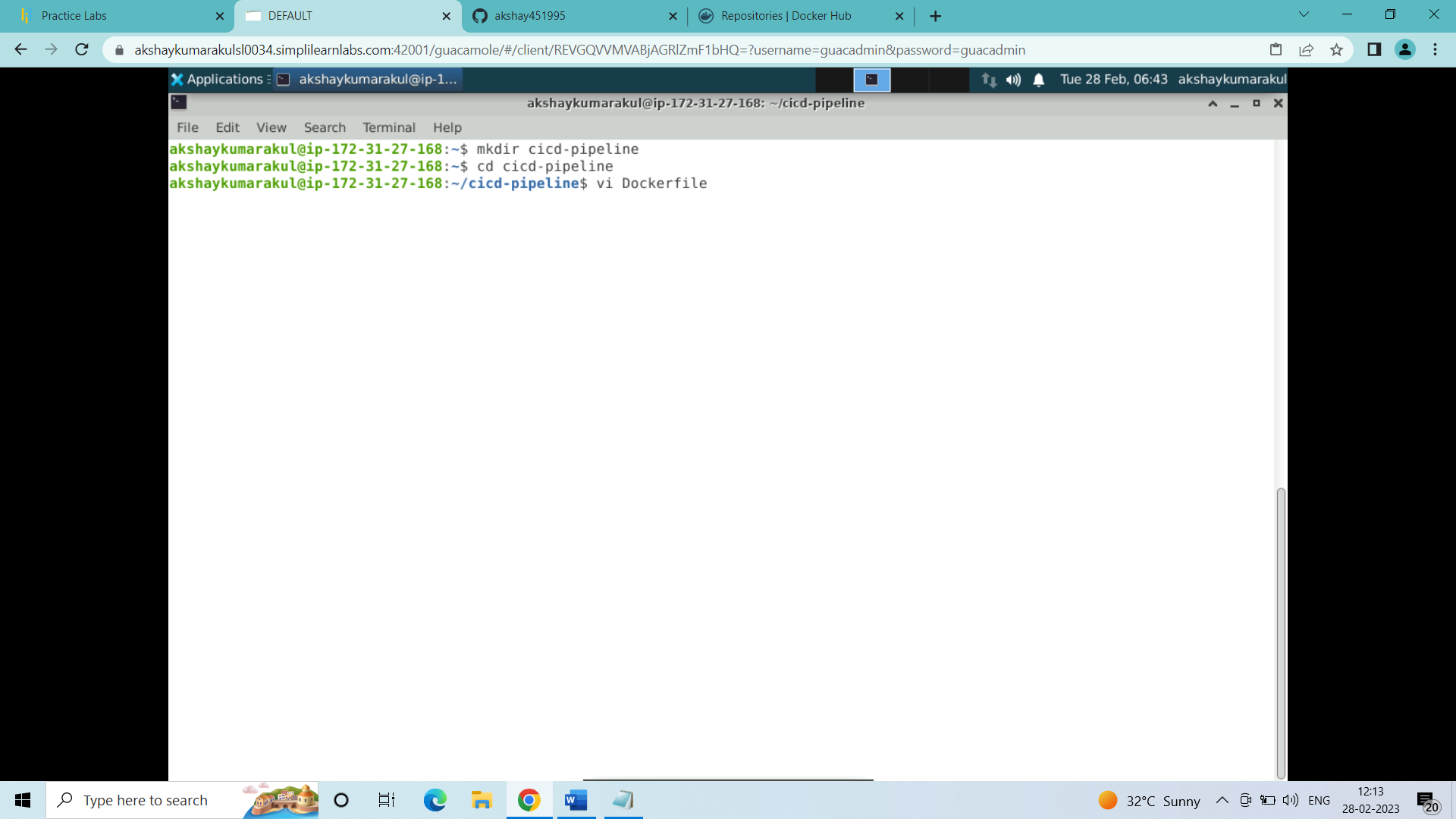


1. Created a directory named cicd-pipeline in the terminal of the ubuntu machine and entered it

mkdir cicd-pipeline

cd cicd-pipeline

1. Create a Dockerfile and Jenkinsfile inside the cicd-peline directory



### Inside the Dockerfile:

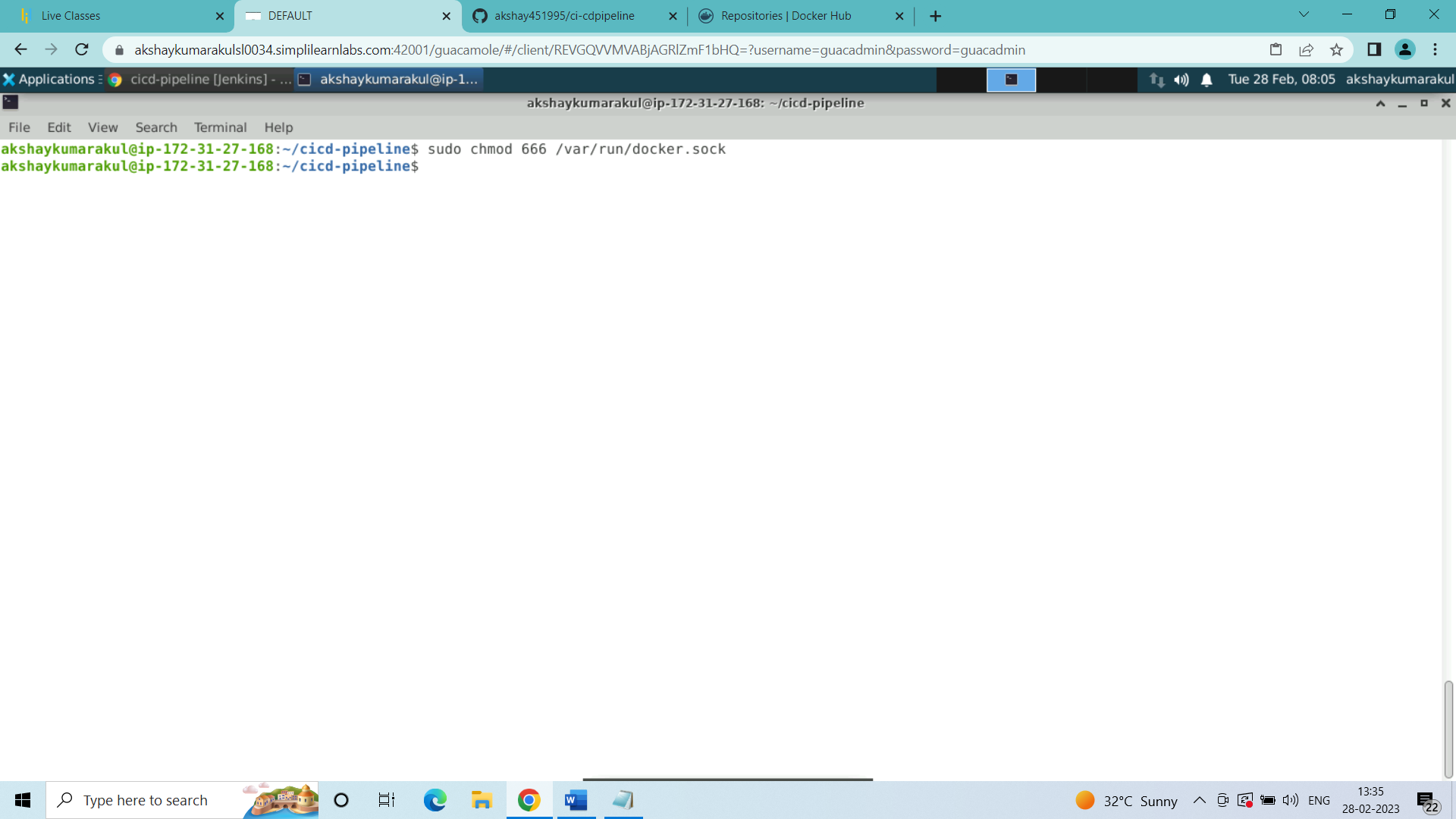
FROM ubuntu

RUN echo "Built container successfully" > file1

(save and quit the file)

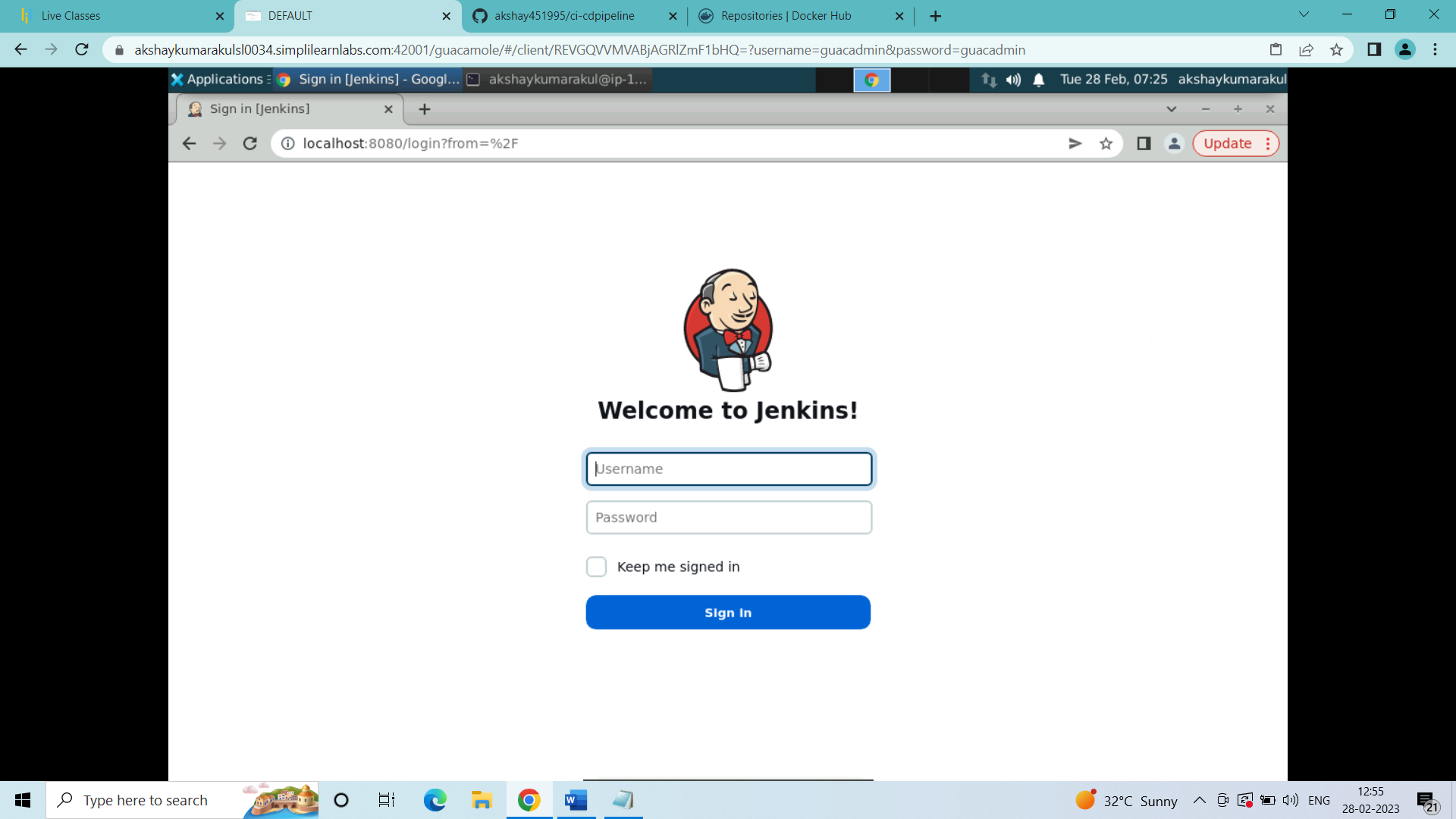
(This is the code to build an image with ubuntu as base image with a file inside it named as file1 which contains the message "Built container successfully")

1. Change the permissions of the docker.sock file

sudo chmod 666 /var/run/docker.sock 

1. Opened the web browser in the simplilearn machine and in address bar type:

{{ localhost:8080 }}



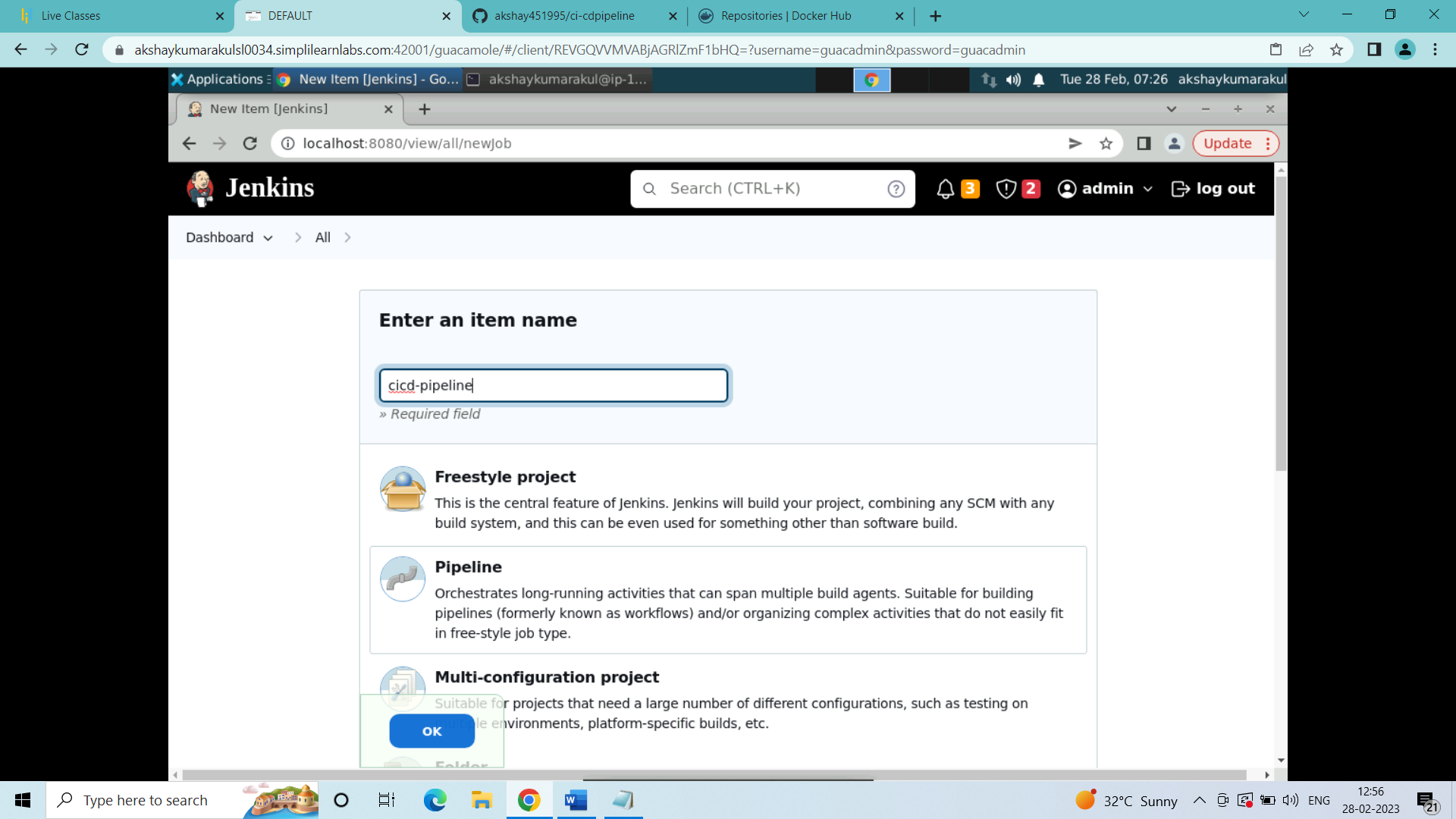
login jenkins with credentials

(inside jenkins dashboard)

create a new pipeline job:

-- create new item:

select pipeline and name the pipeline ci-cdpipeline and click ok



1. Inside the configuration of the cicd-pipeline job:

i) Add the description of the pipeline:

Docker Jenkins Pipeline to implement CI/CD workflow



ii) In the pipeline configure:

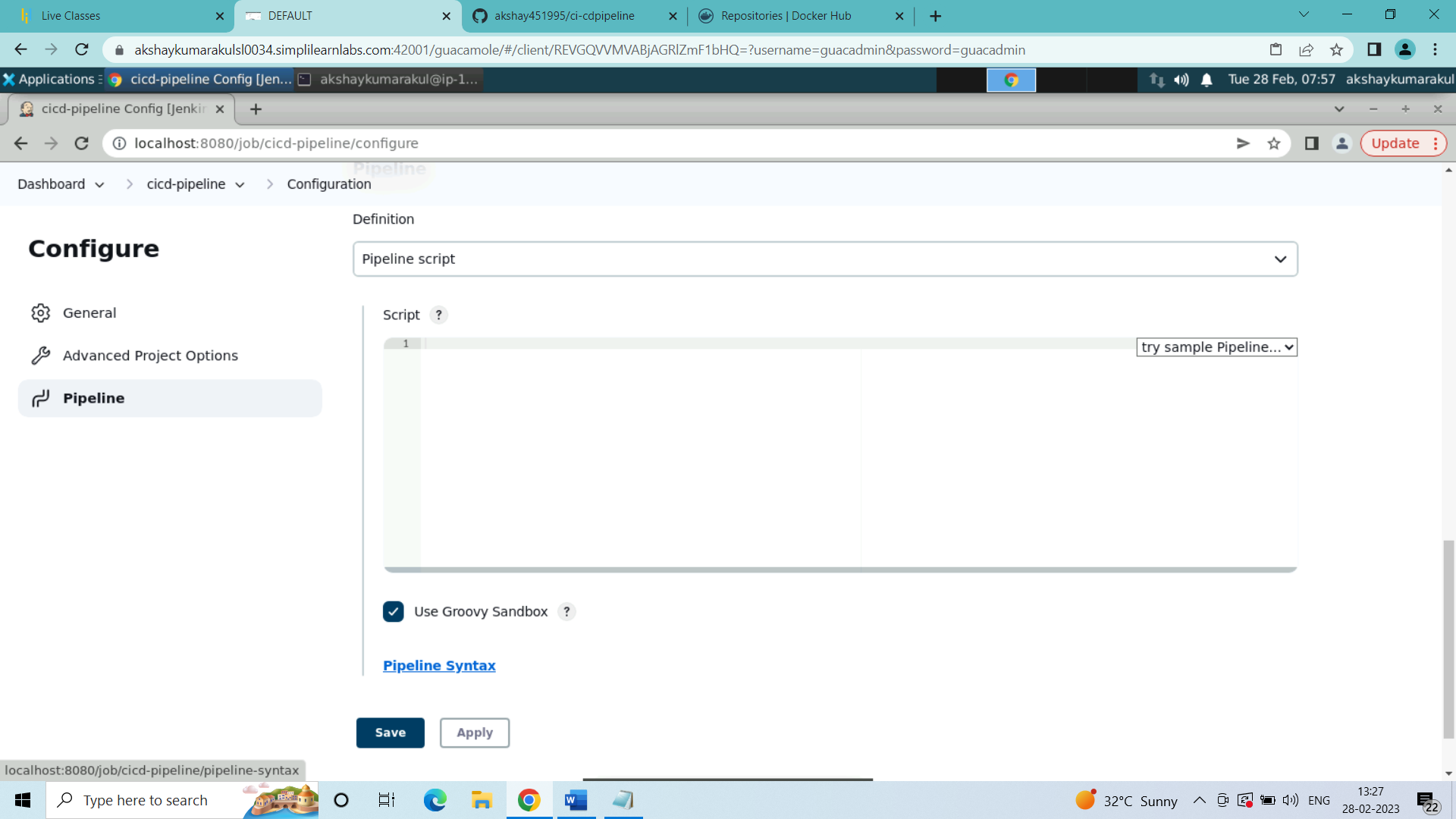
a) in definition: selected Pipeline script

b) clicked on Pipeline syntax to generate the syntax for login script for dockerhub login

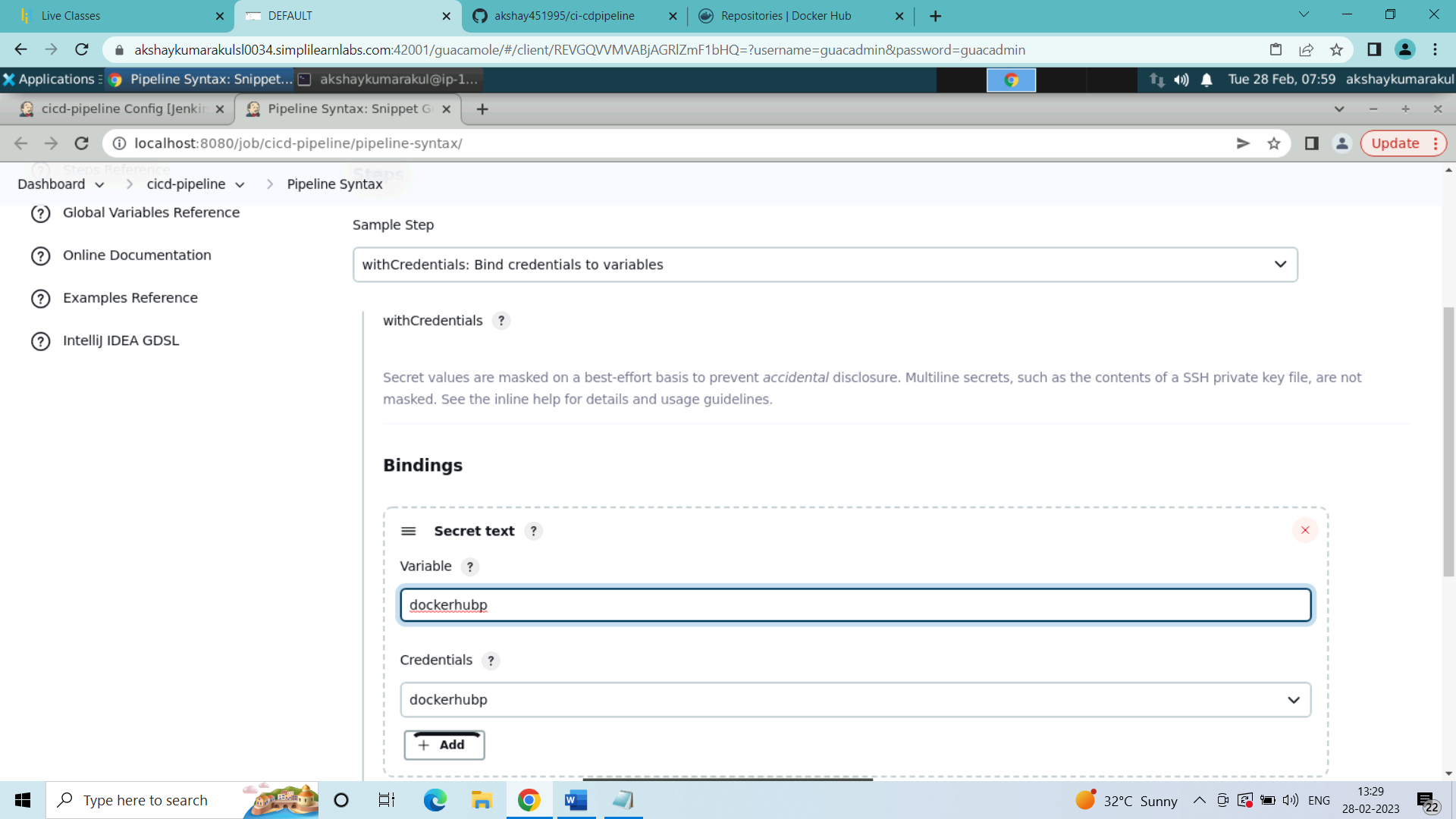
(inside the pipeline syntax)

(i want to generate password variable for providing dockerhub password during

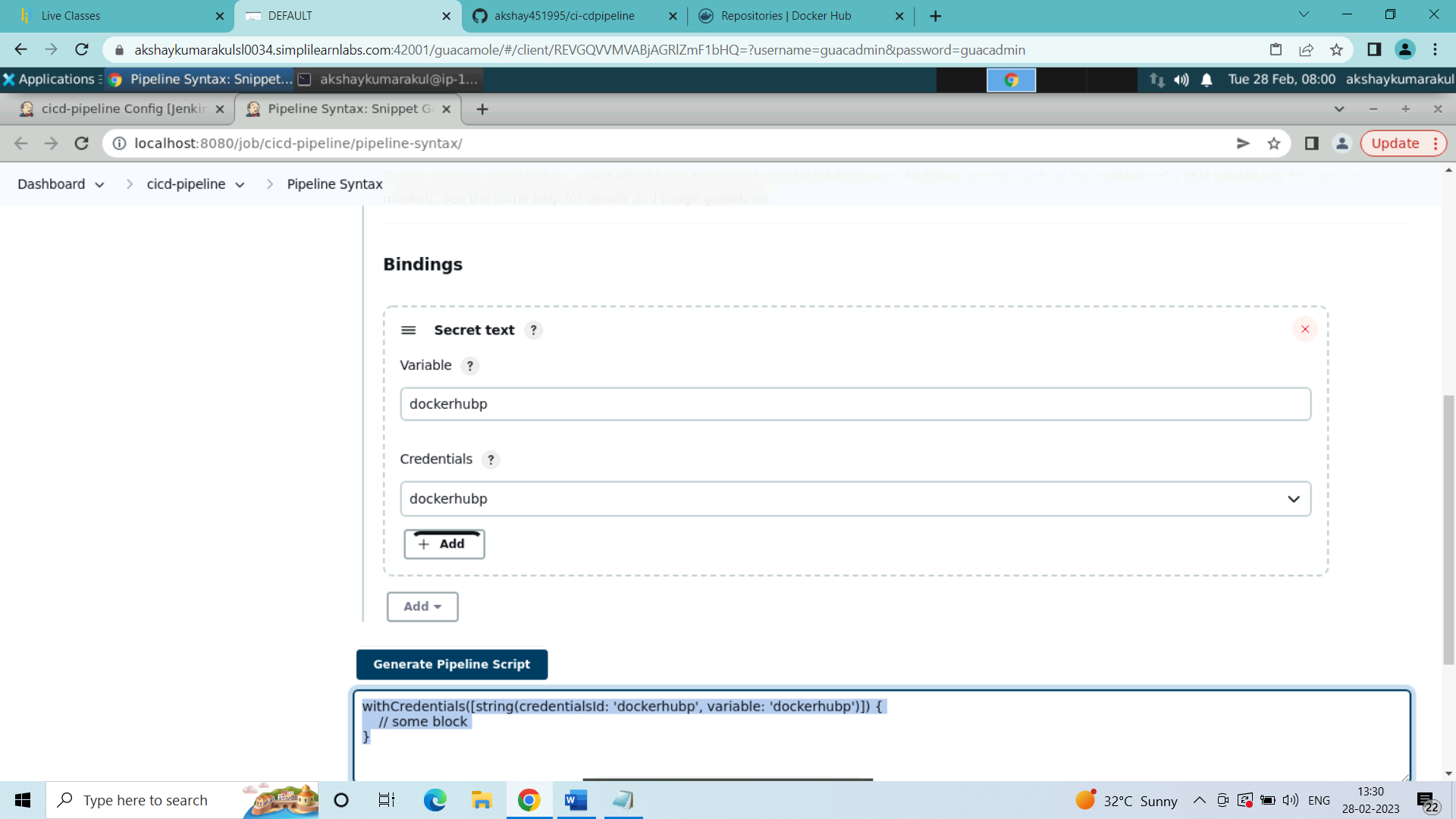
dockerhub login)



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



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--



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'

}

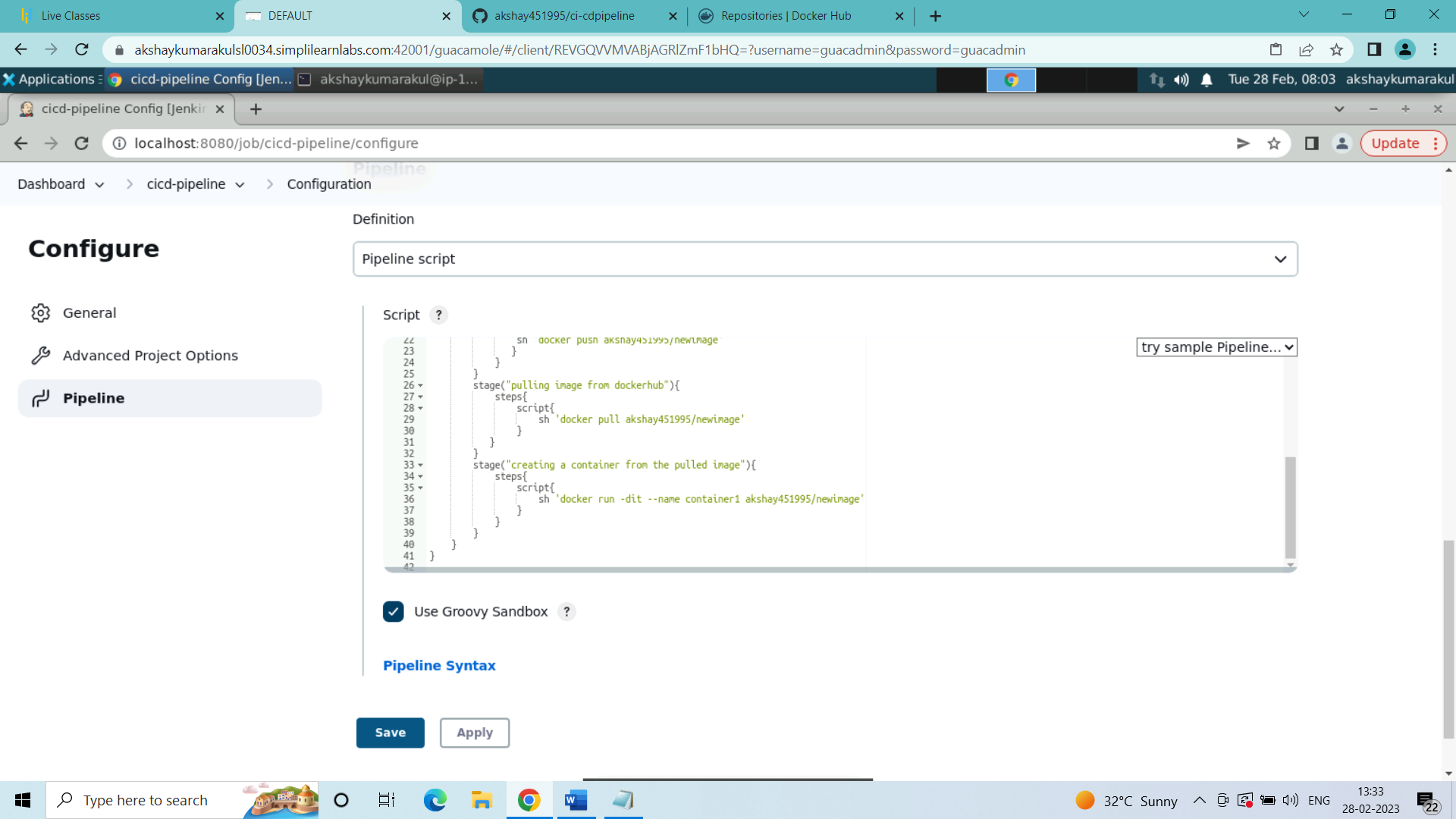
}

}

}

}

(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)



1. 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'

}

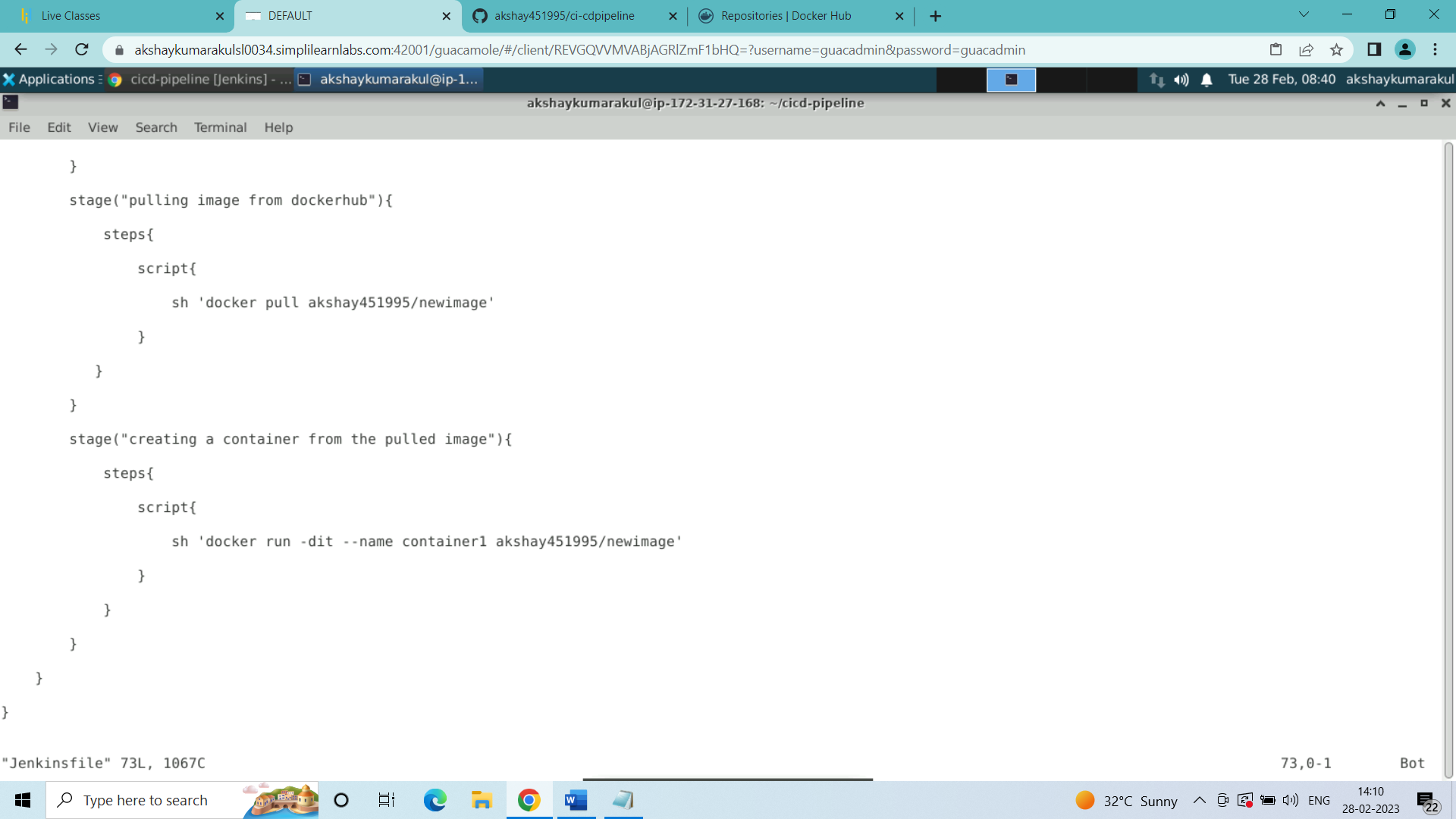
}

}

}

}

(saved and quit the file)



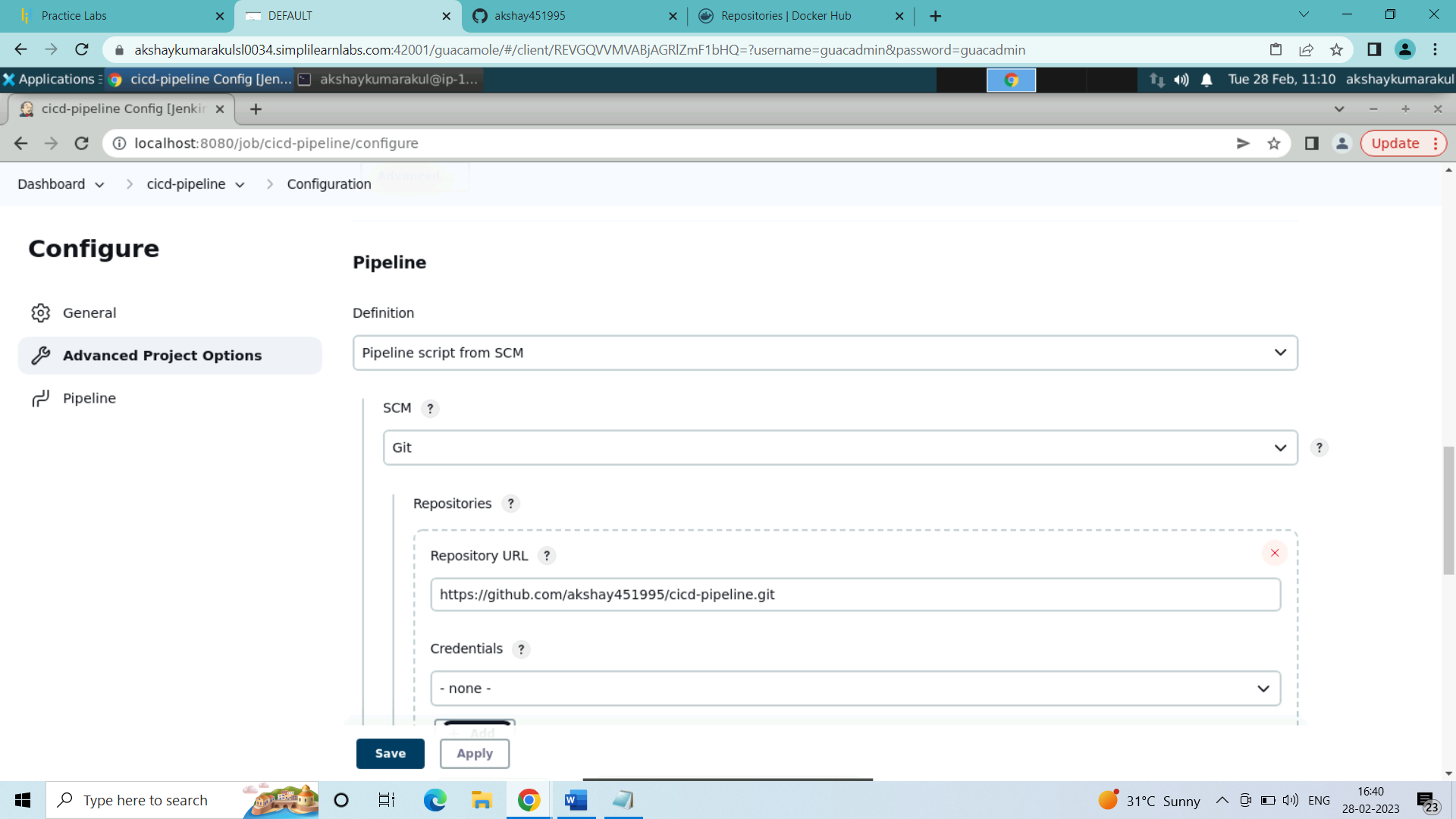
1. Inside the Jenkins cicd-pipeline job configure:

In the Pipeline configure:

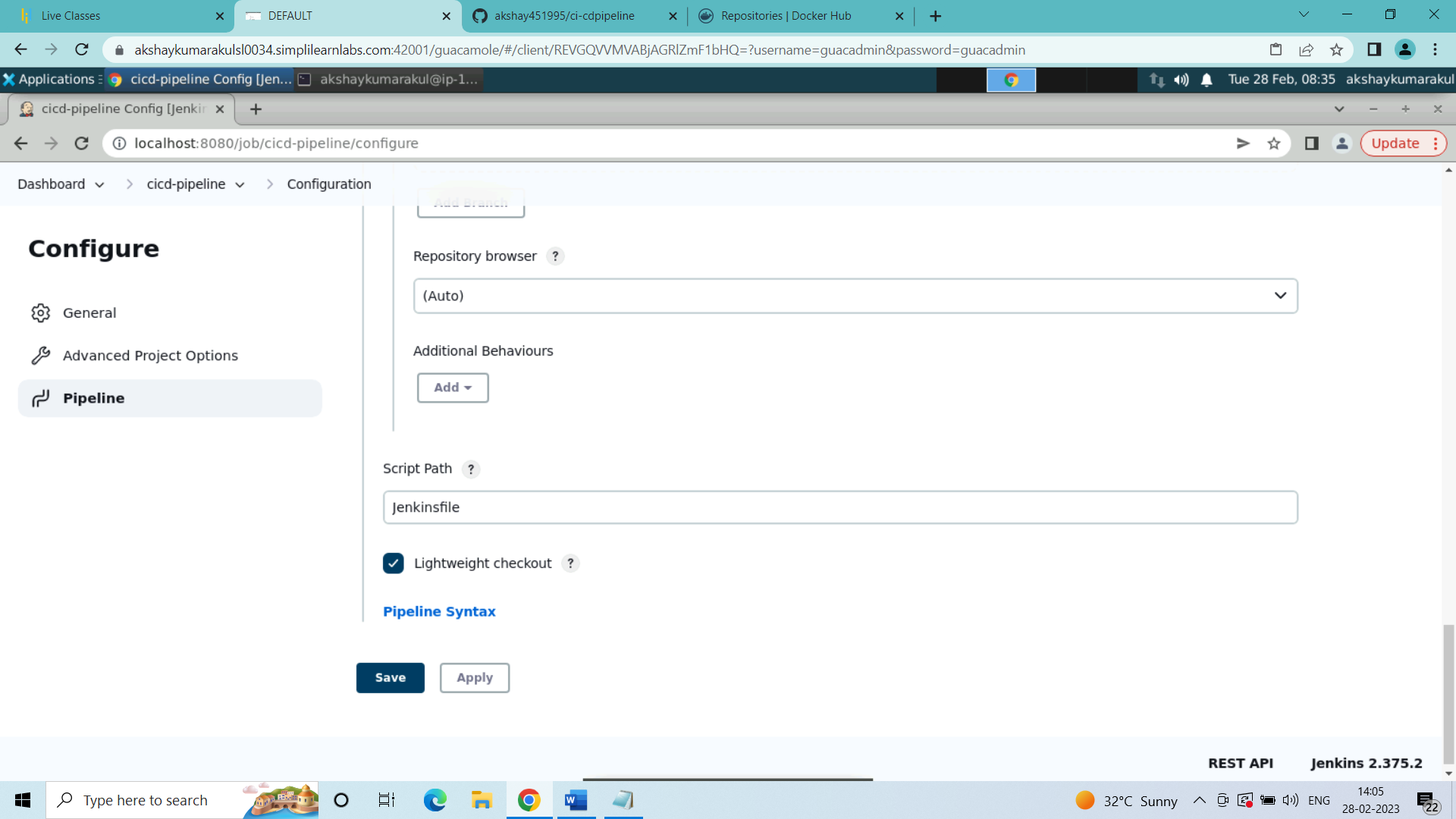
a) in definition: selected Pipeline script from SCM

b) in SCM: selected git

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



d) in Script Path: Jenkinsfile



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

1. Now inside the simplilearn ubuntu machine terminal:

Inside the directory cicd-pipeline, initiated a git repository:

git init

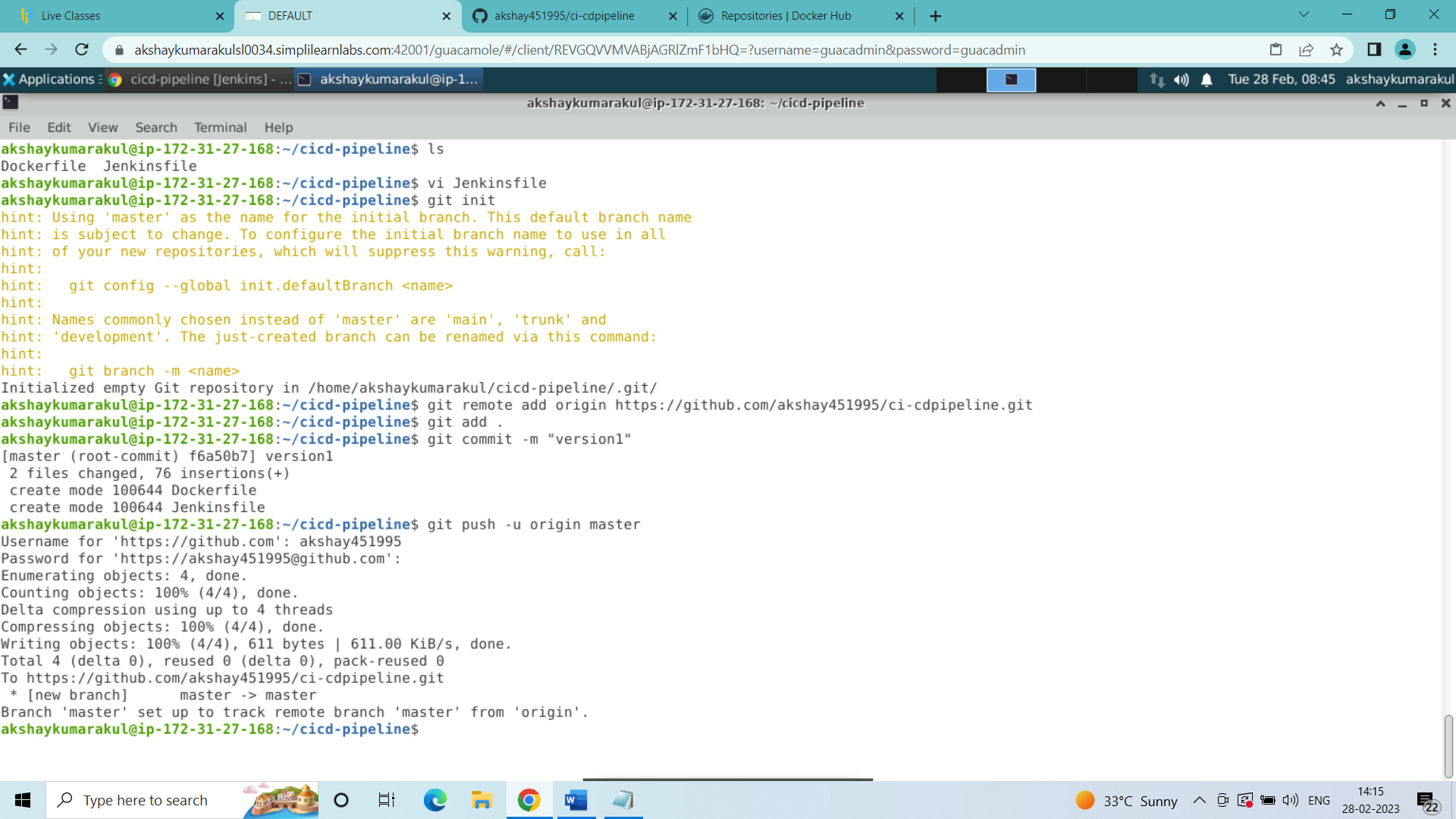
git remote add origin <https://github.com/akshay451995/cicd-pipeline.git>

git add .

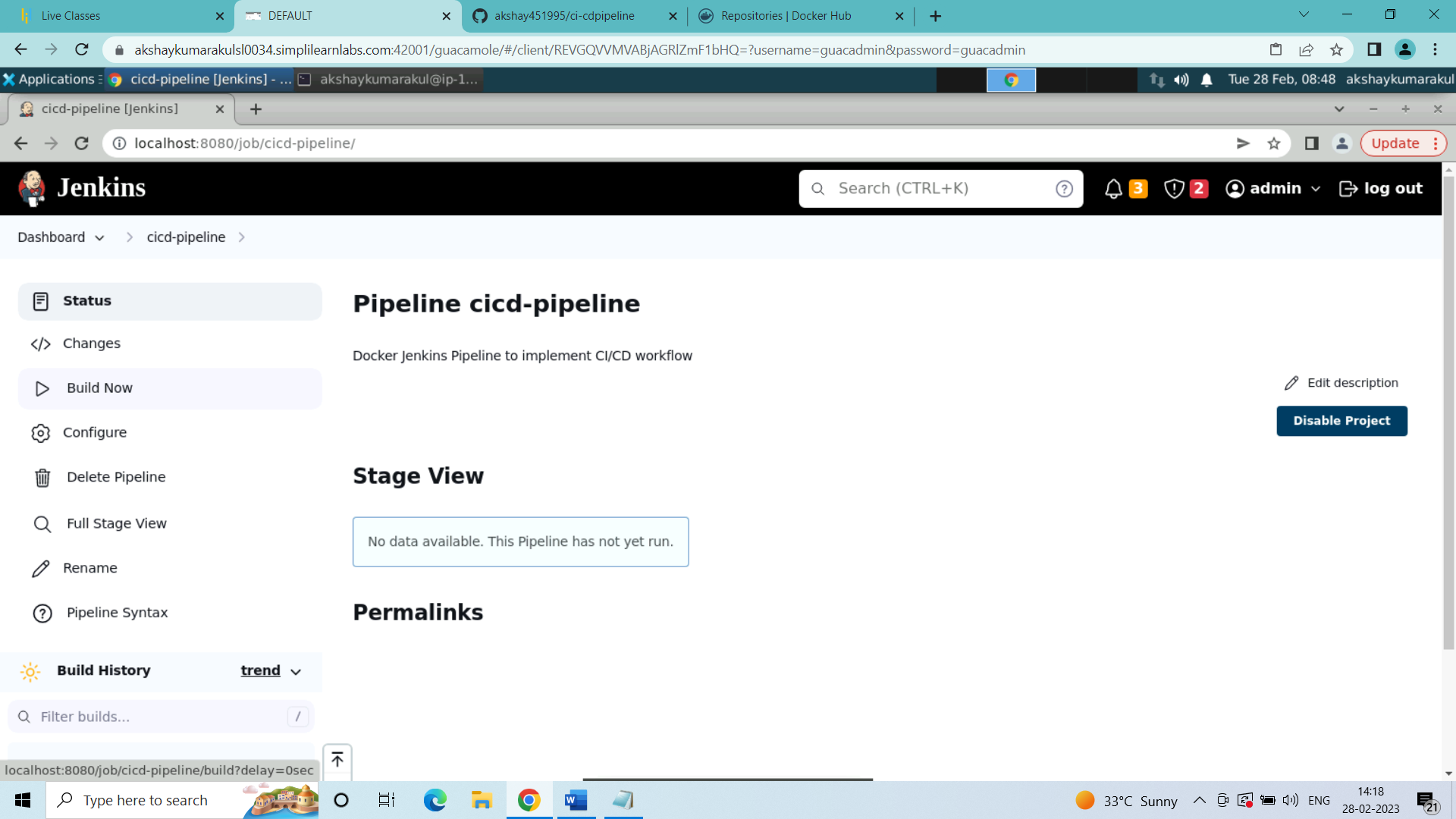
git commit -m “version1”

git push -u origin master

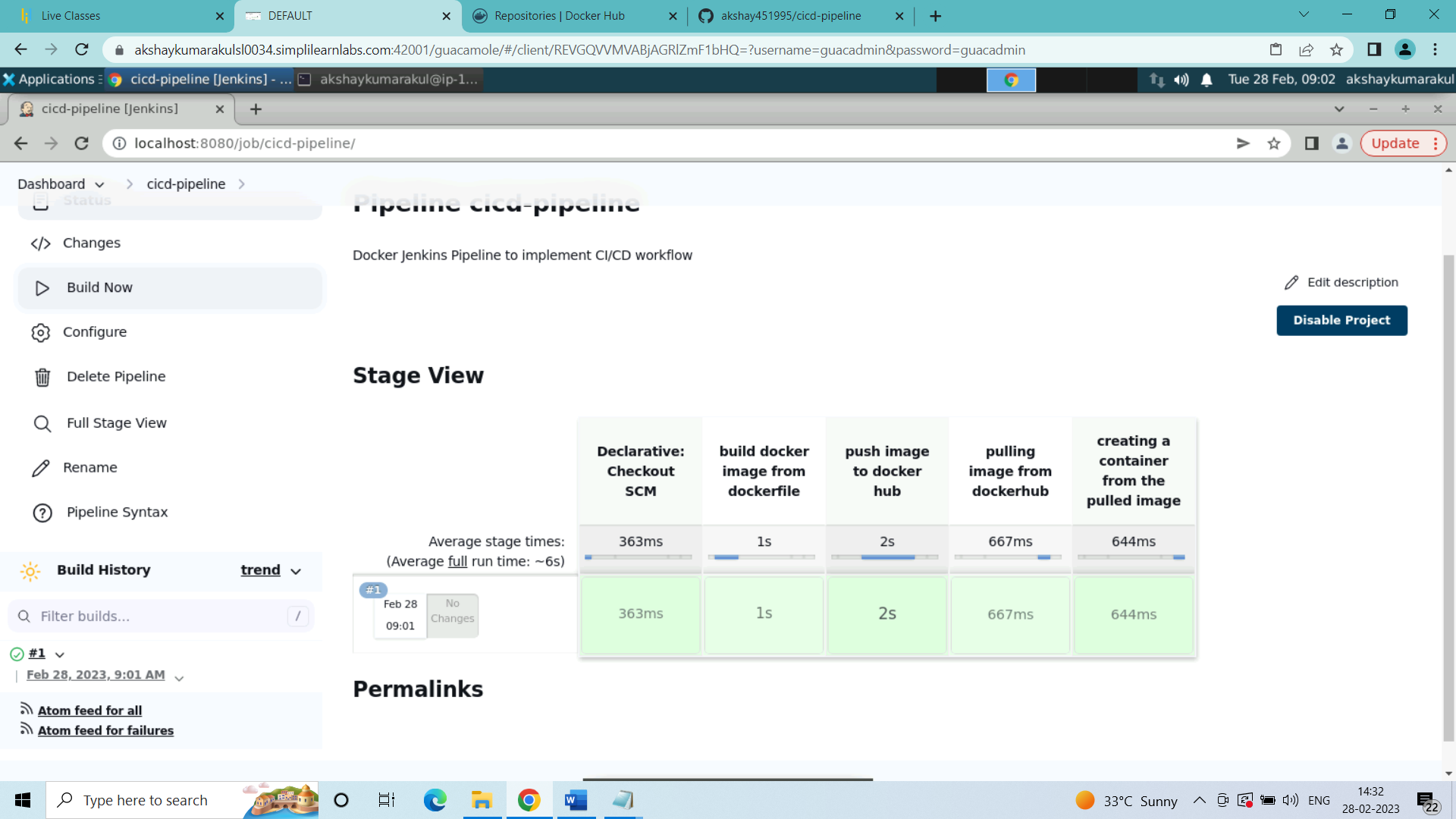
(wrote username and password for the github account in the username and password prompt)



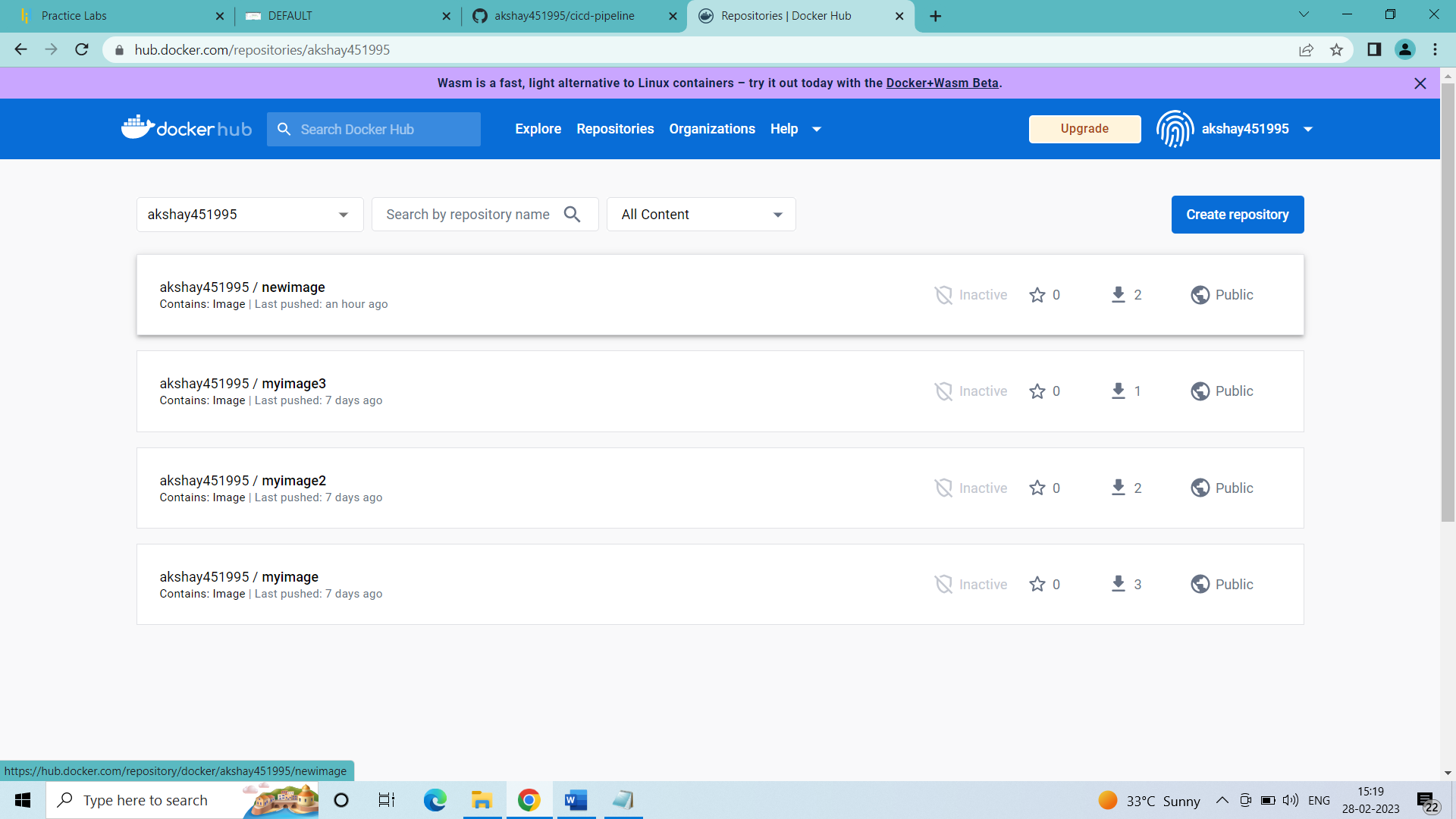
1. Now inside the Jenkins dashboard cicd-pipeline job clicked on build now to execute the pipeline job:

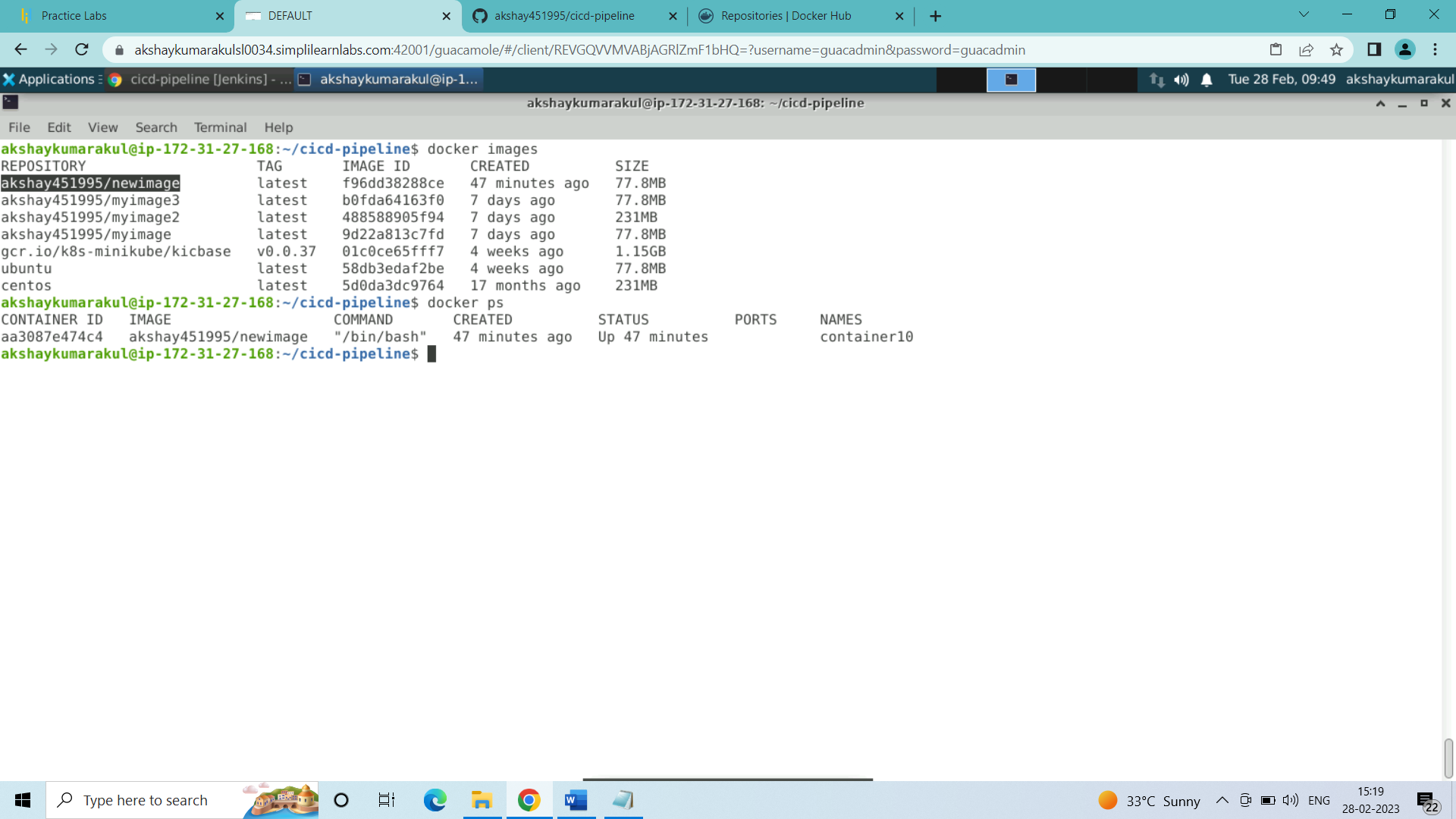


Pipeline Job is executed:



1. Checked if the image was created in the ubuntu machine with the container and also in the dockerhub





1. 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:

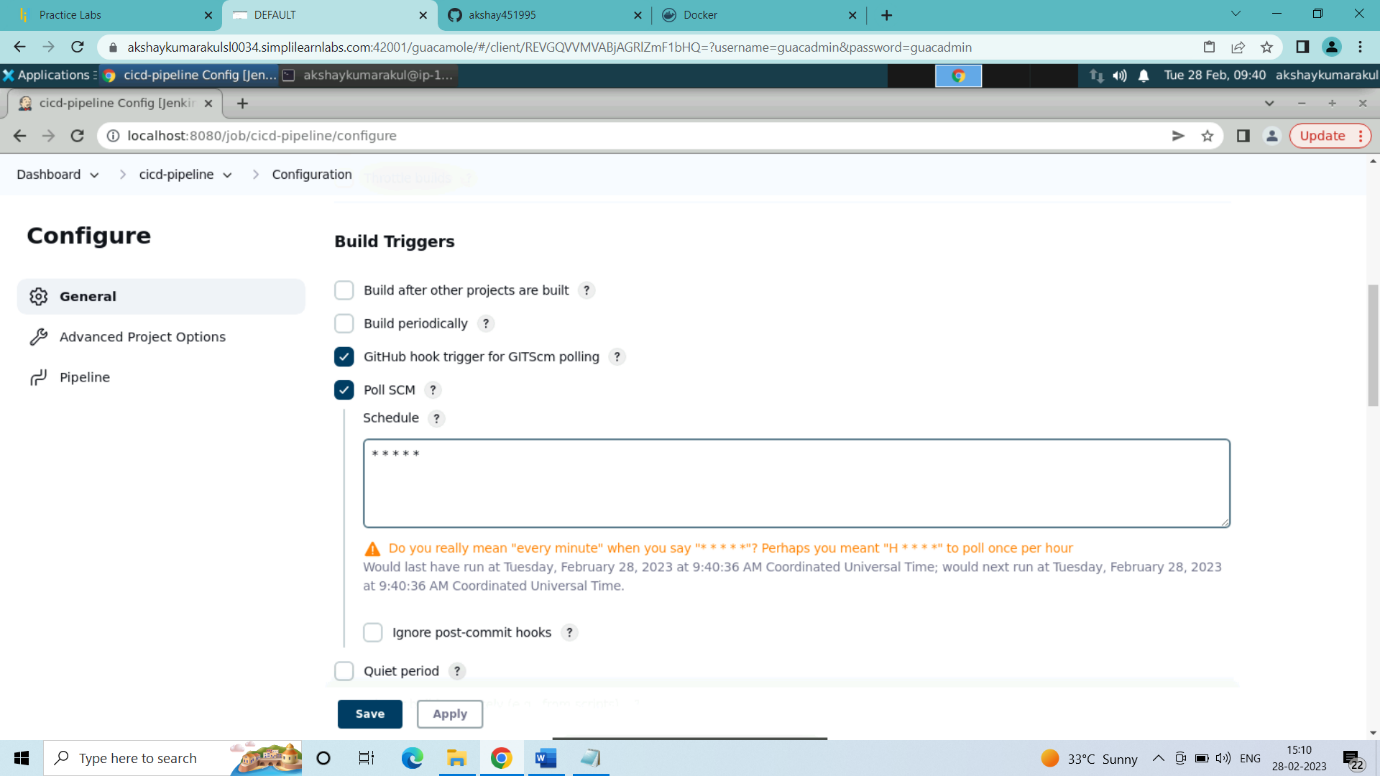
a) GitHub hook trigger for GITScm polling?

b) Poll SCM

(inside the Poll SCM)

Schedule

\* \* \* \* \*



c) click on save

1. 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

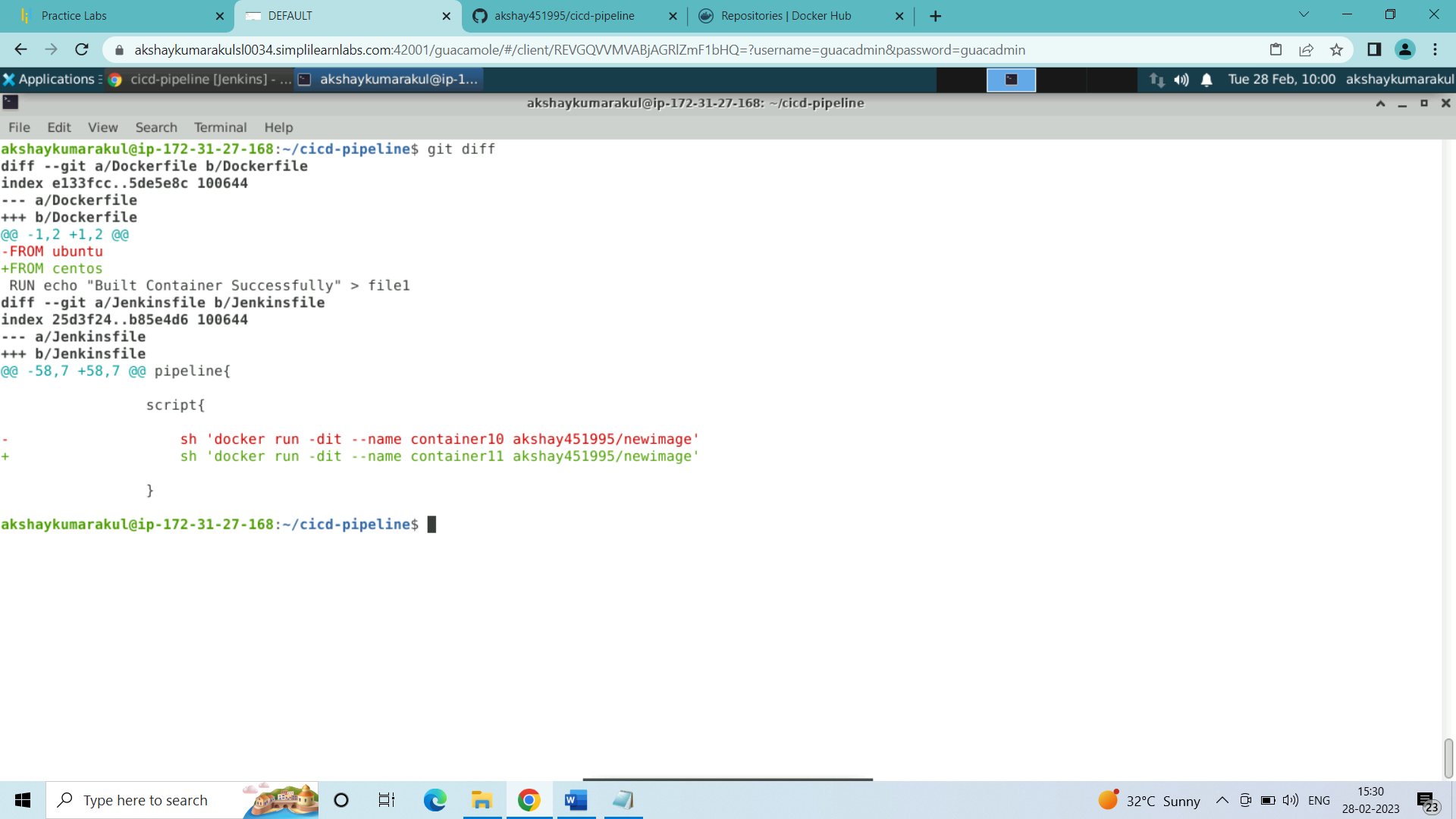
Changes I have done to the Dockerfile and Jenkinsfile:

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)



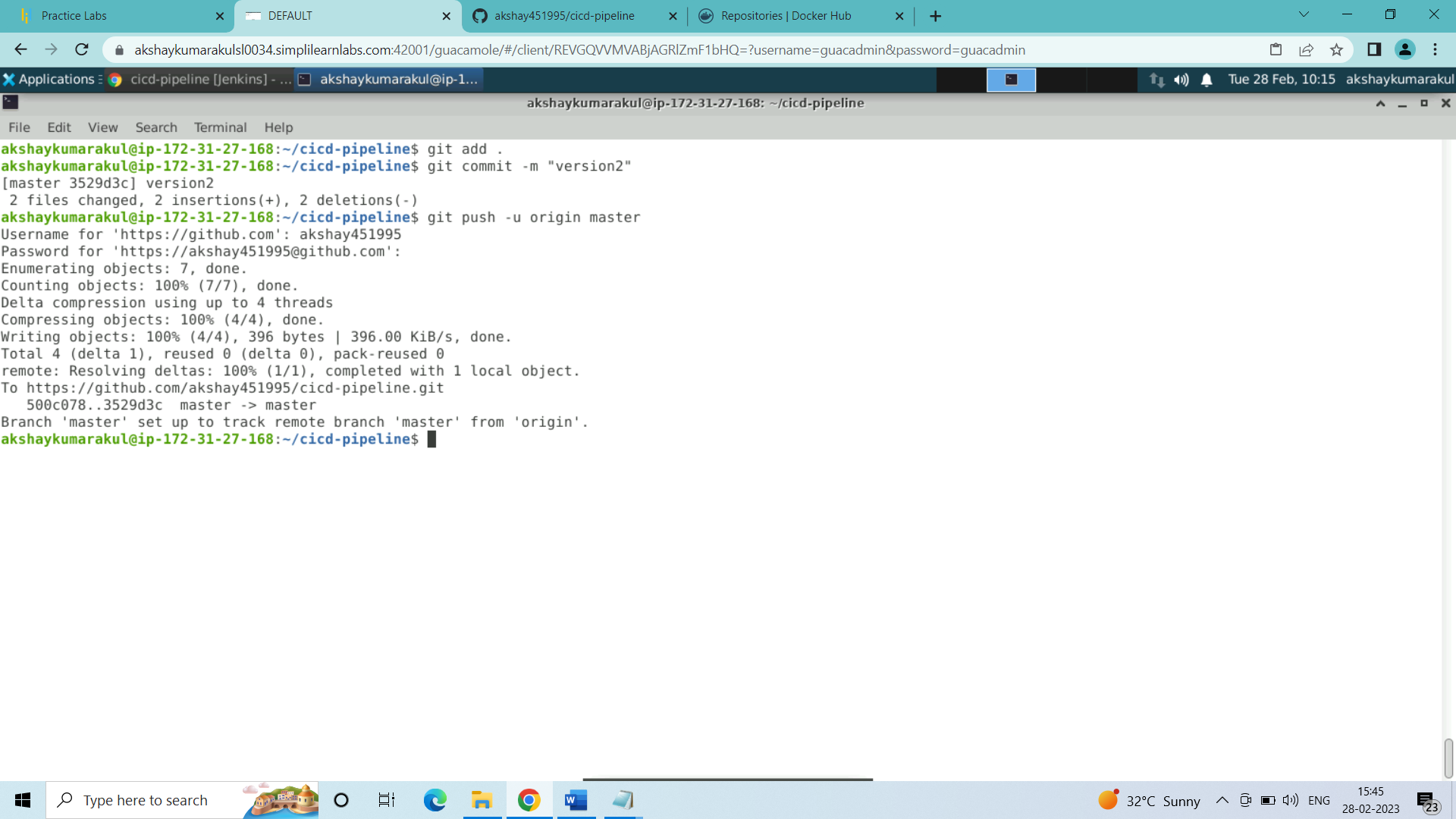
1. Now added and commited the changes to the github repository:

git add .

git commit -m “version2”

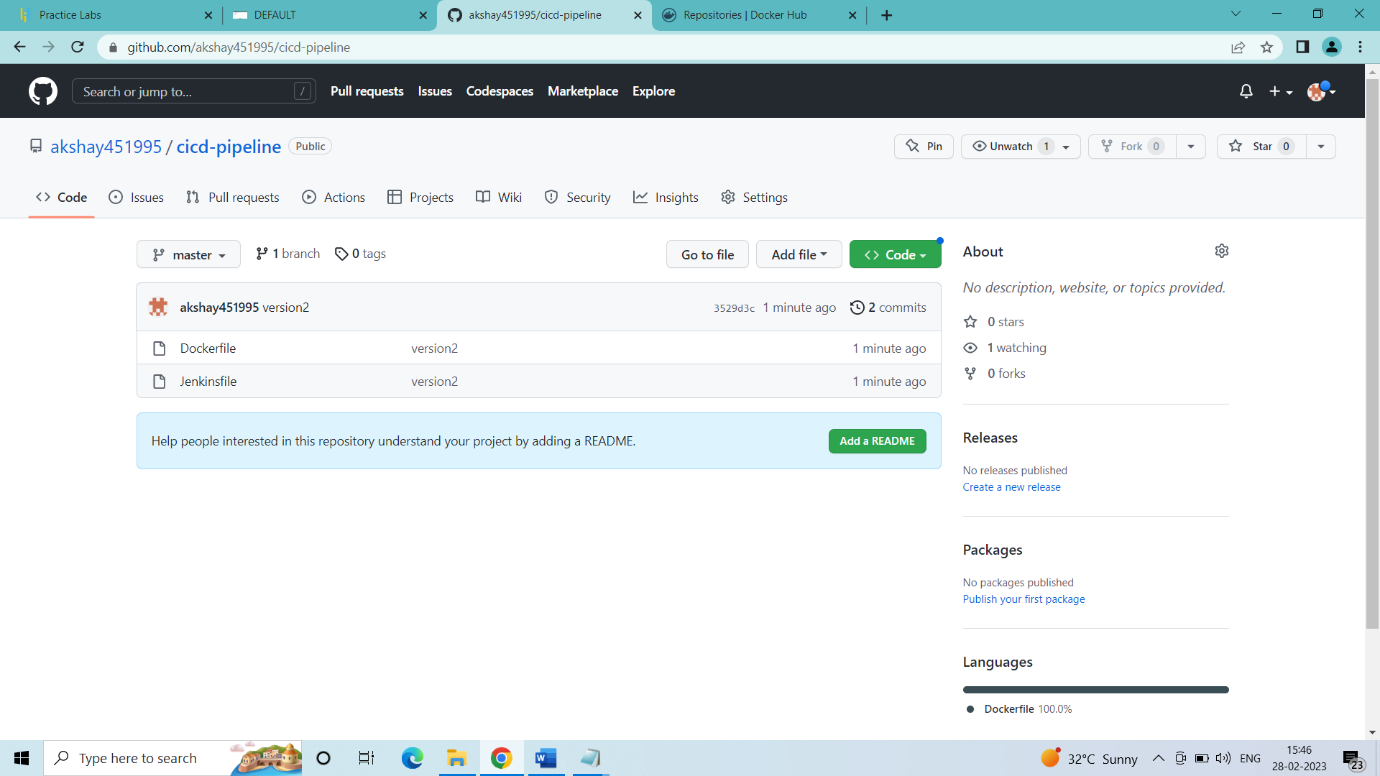
git push -u origin master

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



Changes in the github Repository:

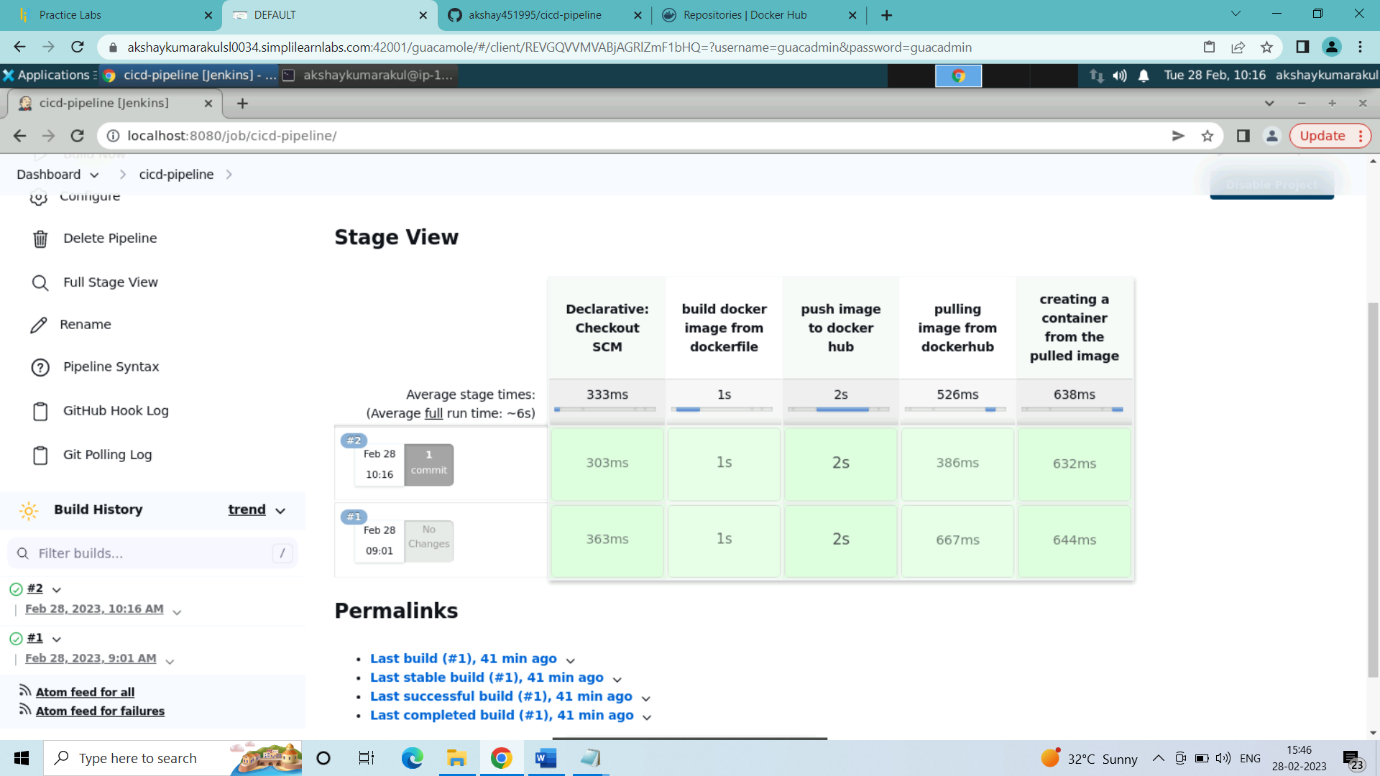
Commit version2 has been done



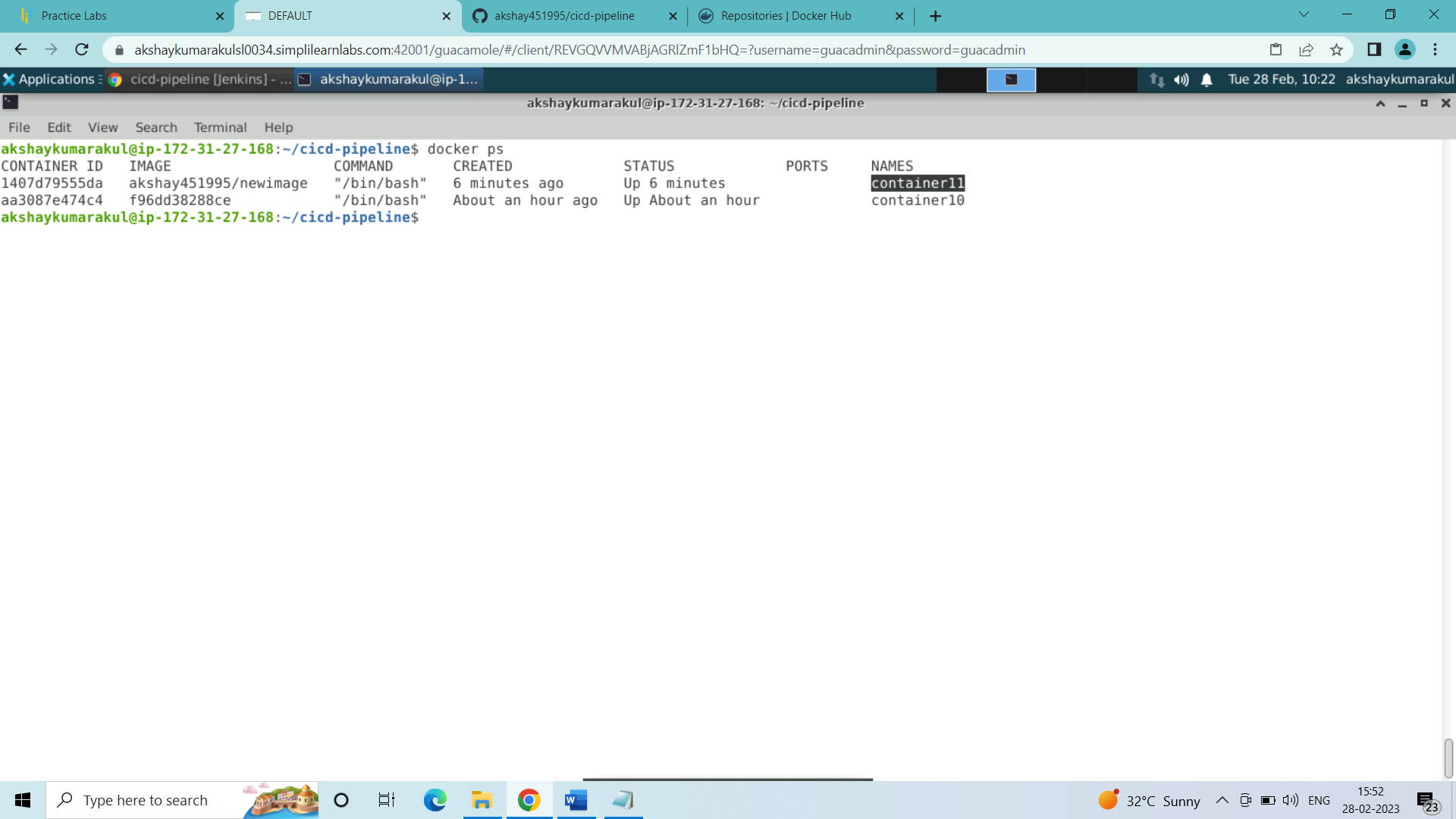
Pipeline is being executed in Jenkins automatically in about 1 minute after the commit:



Pipeline is executed successfully



Now check for the new container in the terminal:



New container was built successfully

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