Course: DevOps   **Name: CH.Lakshmi Priyank**

Module: Jenkins   **Mail-ID: chlakshmipriyanka9@gmail.com**
 Topic: Email configuration   **Batch no: 115**
**Trainer Name: Mr. Madhukar sir**   **Project No:01**
Date of submission: 6 – Dec – 2023

<u>STEPS FOR THE EMAIL CONFIGURATION THROUGH JENKINS PIPELINE:</u>

Open the instance connect through the server and log in to Jenkins.

- After login through Jenkins go through the manage Jenkins and click on the system configuration.

^ Go through the system configuration and scroll down and there we can see the **E-mail notification.**

^ In that we can see the SMTP server and Port number.

^ The default port number of the SMTP server is 465 and the mail of the SMTP server will be delivered default only. We need to type only the port number.
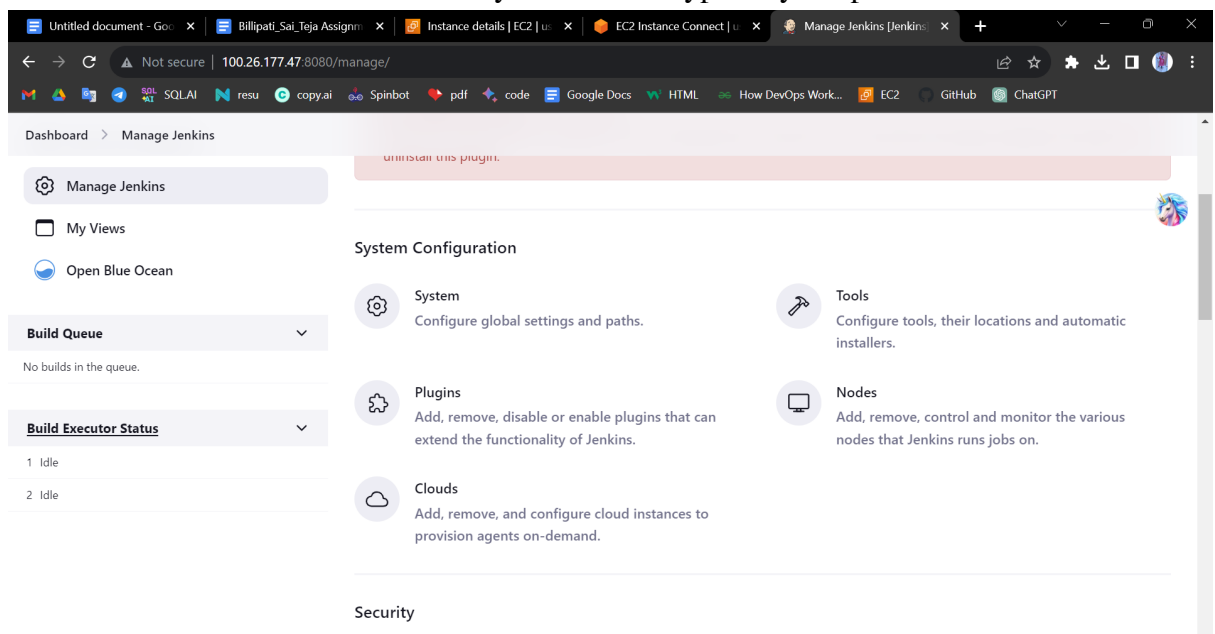
^ Click on the Advanced option below the SMTP port.

^ In that advanced option there will be credentials we add the Jenkins deployer credentials and save it.

^ Now after the credentials creation below the credentials there will be an option for SMTP authentication.

^ There we need to give the E-mail address credentials of which mail we need.

^ Below there will be a reply e-mail address we need to give the same email address that is given in the credentials.

^ We need to test whether the given E-mail address is working or not so for that, we will do a test configuration.

^ But when we give the E-mail address and click on the test configuration we get an error.

^ To sort out this error we need to go through the E-mail credentials and click on the manage account.
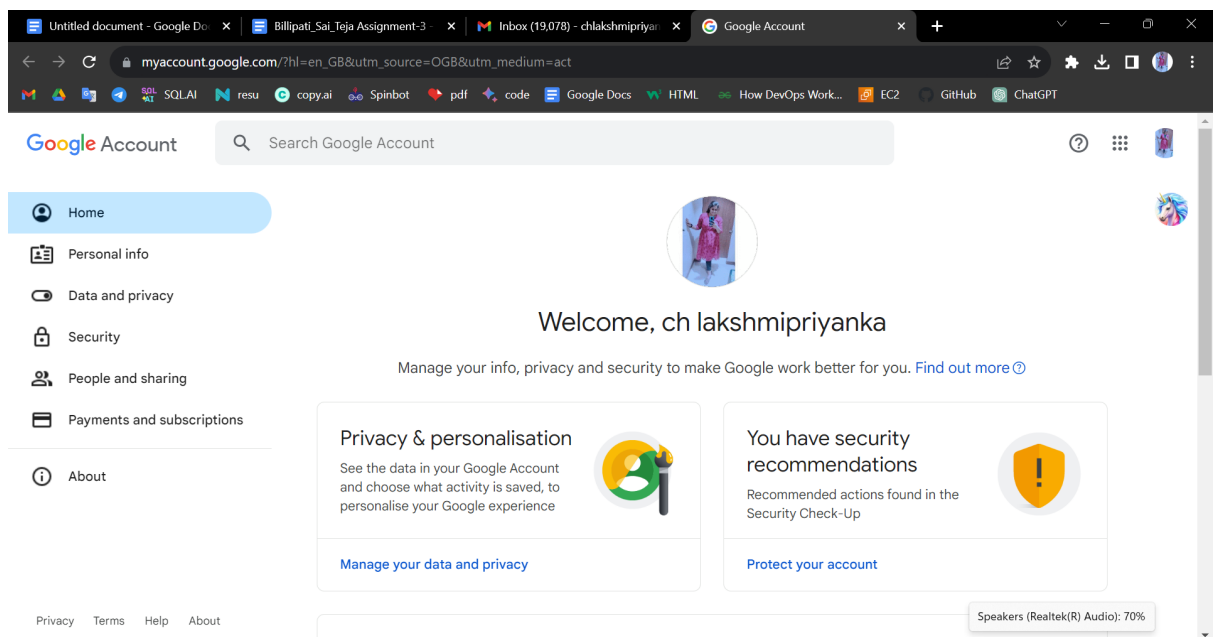
Go through the security and click on the 2–step verification on it.

In the 2-step verification, the interface will be like shown below and in that click on get started.

^ When we click on get started it will ask us to sign in through the phone to turn on the 2 - verification step click on ok and proceed.

^ We get a notification on our phone to verify whether we or any unknown user is using click yes on the mobile.

^ When we click on yes on our mobile it shows the mobile device that currently we access so confirm the device and click on continue.

^ After that it will show the backup option to retrieve the account which will choose the mobile number and proceed.

^ Whether we forgot the E-mail credentials we can retrieve back through the number that we were given.

^ We have 2 options text or call we can take which is comfortable for us and click on send.

^ We get a code to the given mobile enter it and click on next.

^ Now the 2-step verification will be turned on and the device on which we are enabled is also shown below.

^ Go through the Google prompt click on the app passwords and create a password!

^ Create a new password Jenkins and click on create

^ When creating a new password it will generate an app password copy that password.

    ^ When creating a new password it will generate an app password copy that password.

^ Now again go to the system configuration and scroll down to the E-mail notification.

^ In that click on the advance option and go through the Use **SMTP Authentication.**

^ Enter Email: katikameswar92@gmail.com
    Password: (this is the password generated by the app password)

^ Give the default port number 465 in the port no.

Now in the reply address, we need to give the E-mail address to which the notifications to sent.

^ After all the credentials are entered, we need to test whether the notification is working or not and sending the emails correctly or not.

^ For this, we had the option to **Test the e-mail recipient** and the email and click on the test configuration.

^ The email is successfully sent to the give credentials of E-mail.

^ Go through the default triggers and select the triggers when the file has been built.

^ Now click on the apply and save and test.

E-mail configuration in the Jenkins pipeline:

^ Click on the current project or create a new project which is a pipeline project and go through the pipeline script.

PIPELINE SCRIPT FOR THE E-MAIL CONFIGURATION:

```
pipeline {
    agent any
    stages {
    stage('checkout') {
    steps {
    checkout scmGit (branches: [[name: '*/DEV']], extensions: [], userRemoteConfigs:
    [[credentialsId: '35ec10a9-48ac-4423-a432-6a37b31cc3ce', url:
    'https://github.com/Venna12/taxi booking.git']])
    }
    }
    stage('Build') {
    steps {
```

```
sh 'mvn package'

}

}
}
```

E-MAIL CONFIGURATION SCPRIT:

```
_post {

always {

script {

if (currentBuild.currentResult == 'SUCCESS') {

step ([$class: 'Mailer', notifyEveryUnstableBuild: true, recipients:
'chlakshmipriyanka9@gmail.com', sendToIndividuals: true])

}

else if (currentBuild.currentResult == 'FAILURE') {

step ([$class: 'Mailer', notifyEveryUnstableBuild: true, recipients:
'chlakshmipriyanka9@gmail.com', sendToIndividuals: true])

}

}

}

}

}
```
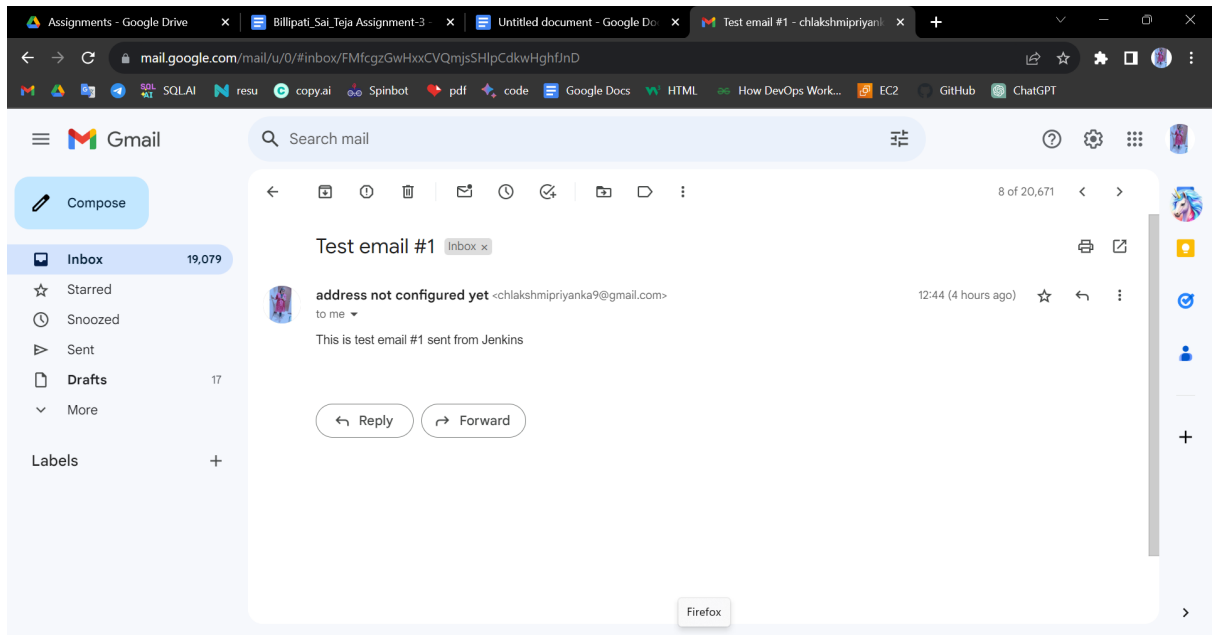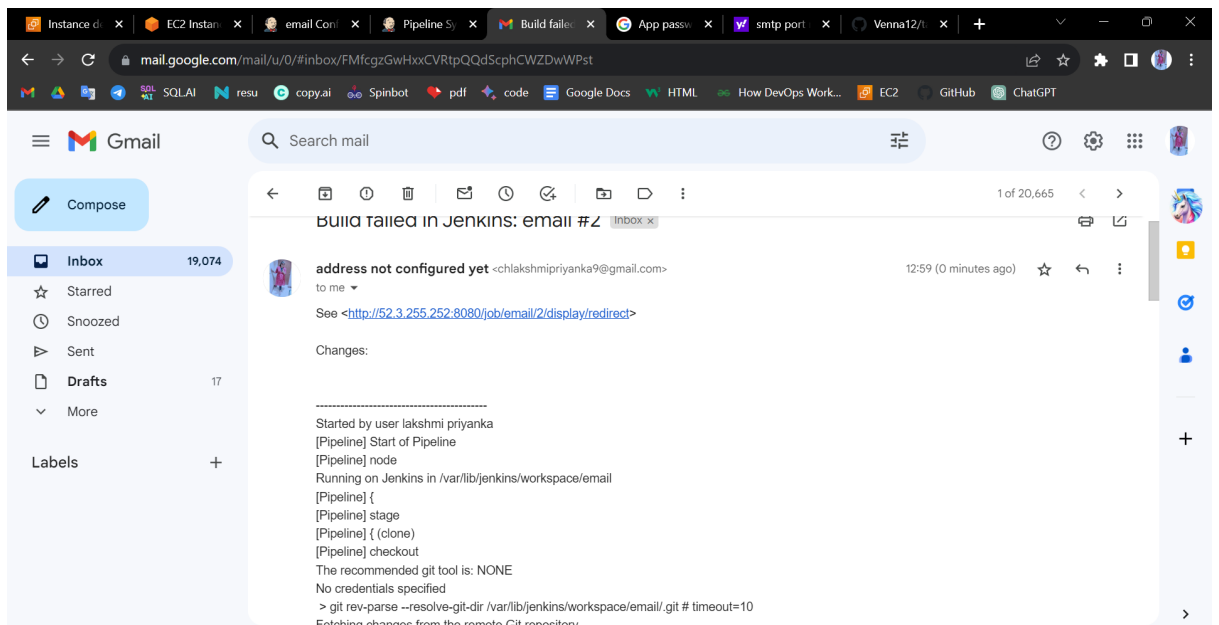
- Now build the file

^ The build has successfully completed, and E-mail was also successfully notified.

.

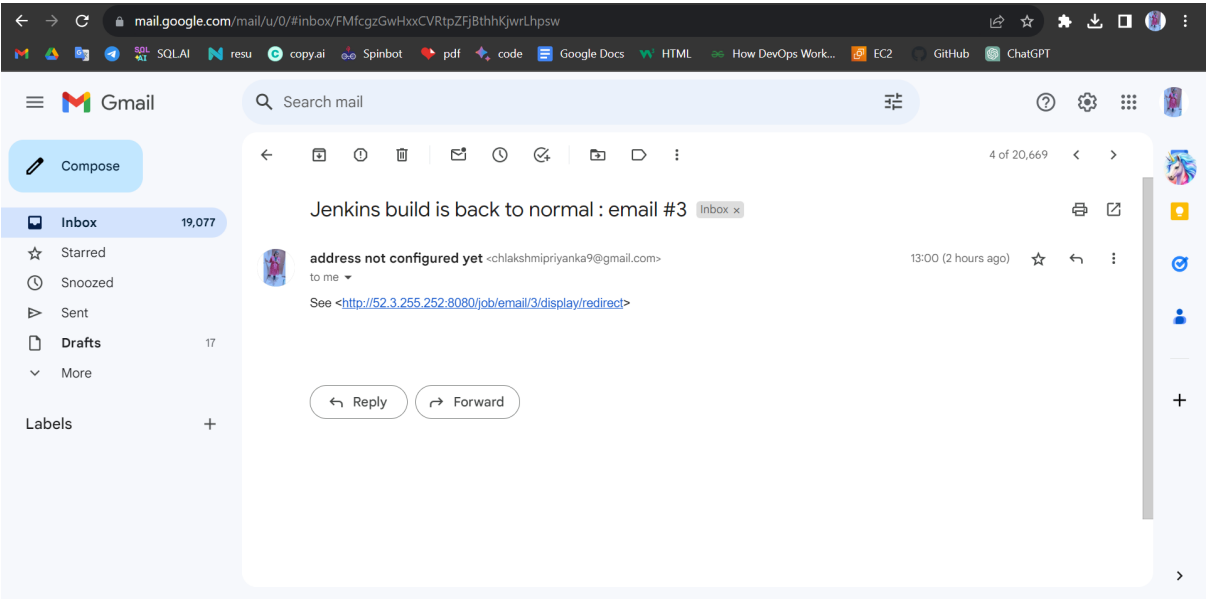^ The build was successfully completed, and E-mail was also successfully

notified.

^ When the build is failed

^ The build failure occurs means the errors which errors and where are the errors the
entire

information will be sent to the E-mail address.

FINAL OUTPUT:

THE END