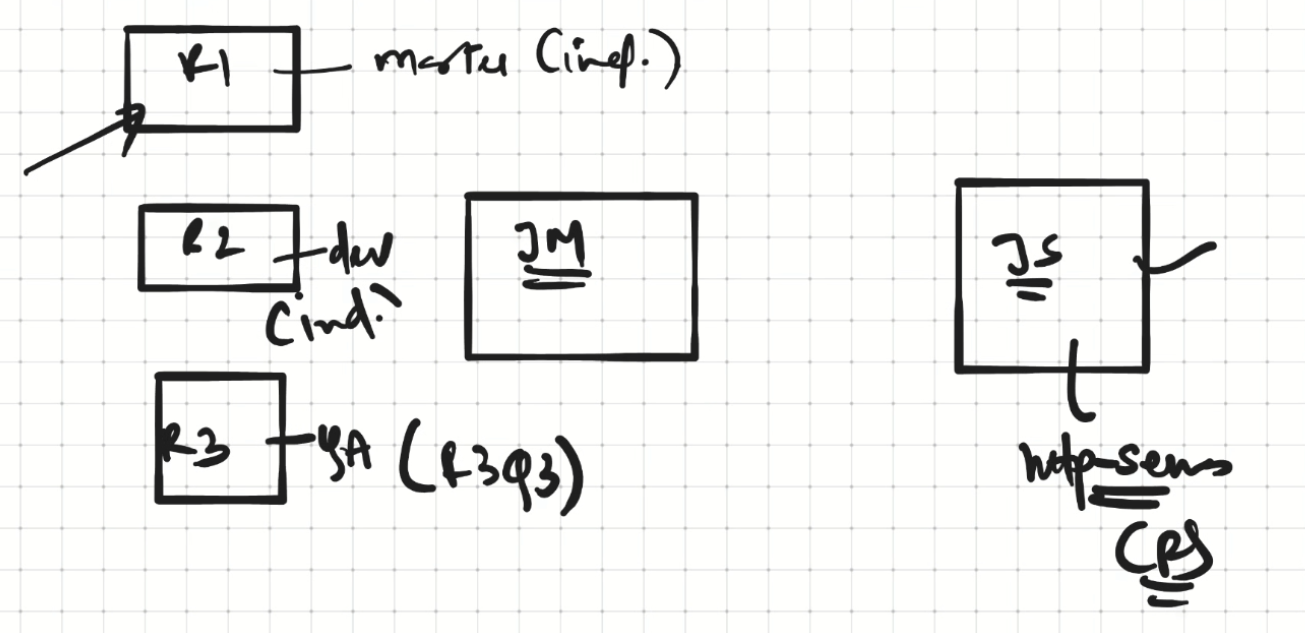
Jenkins Assignment-4



Create 3 Repositories on github and add index.html page in each repository

Repo-1 has master branch, Repo-2 has dev branch, Repo-3 has qa branch

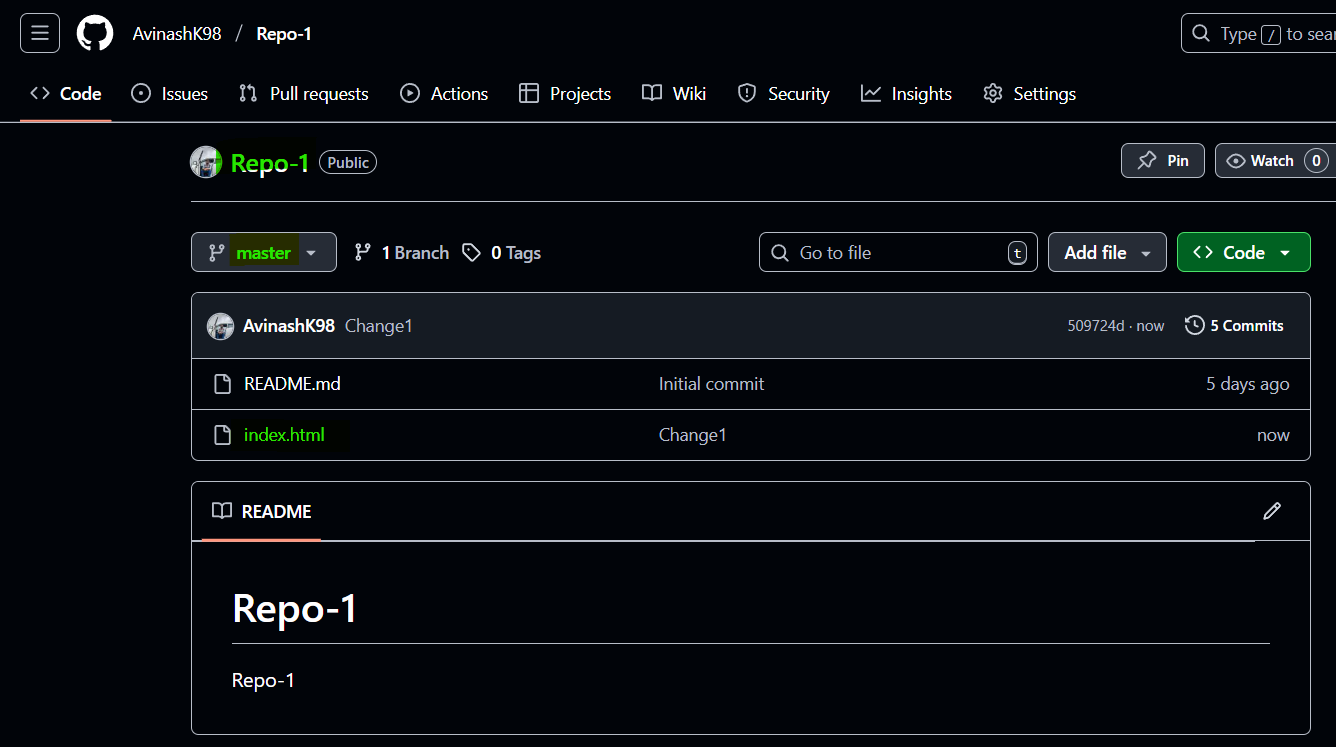
Configure a Jenkins-Master and Slave machines

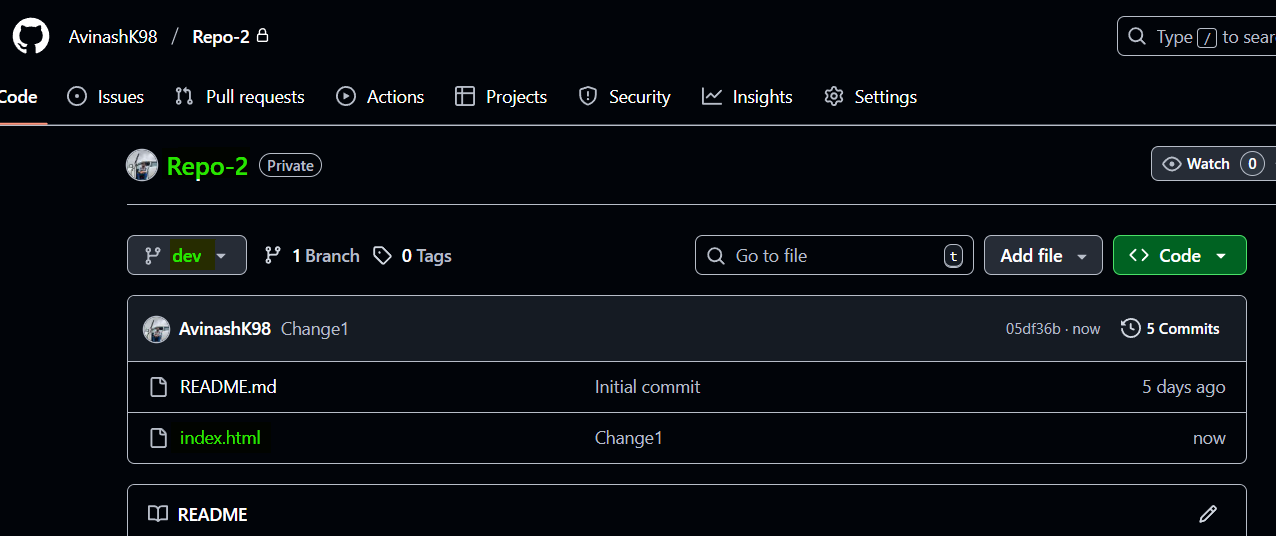
Install httpd and start the server on slave

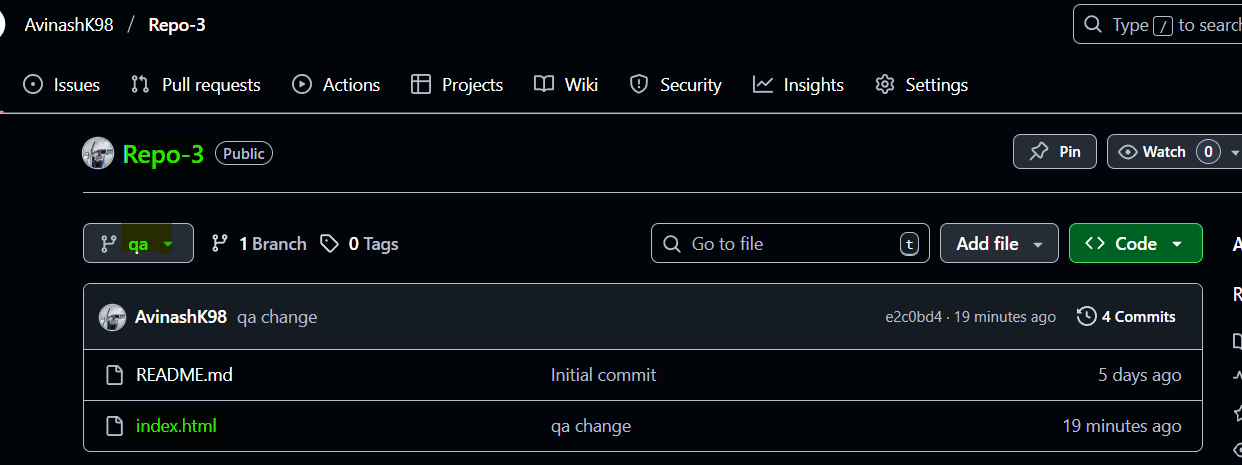
Create Jenkins job for every repository. Whenever the latest commit has been done the change should deployed on httpd server which is running on Slave machine

Solution:

1. Create 3 Repositories and add index.html page in each repository

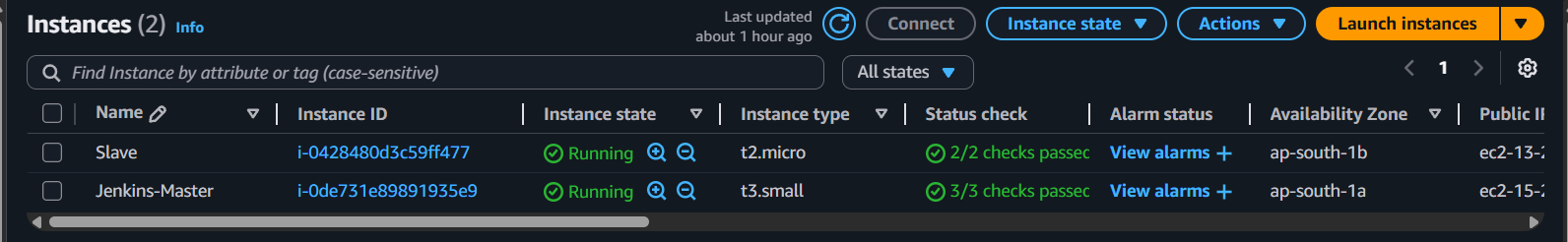






1. Configuring Jenkins-Master and Slave connection

Launch 2 instances (Jenkins-Master and Slave)



i) Launch an EC2 Instance (Jenkins-Master)

ii)Install java 17

iii)Install git

iv)Install Apache Tomcat 10, provide permissions, start the server

v)Deploy latest Jenkins on apache tomcat

vi)Launch another EC2 instance Tomcat-Server

vii)Install java 17

viii)Install Apache Tomcat 10, provide permissions, start the server

ix) Launch another EC2 Instance (Slave)  
x) Install Java 17  
xi) Install git  
xii) Create Slave Home mkdir /mnt/Jenkins-slave  
xiii) Install httpd server  
xiv) Start httpd server

Initial setup has been done

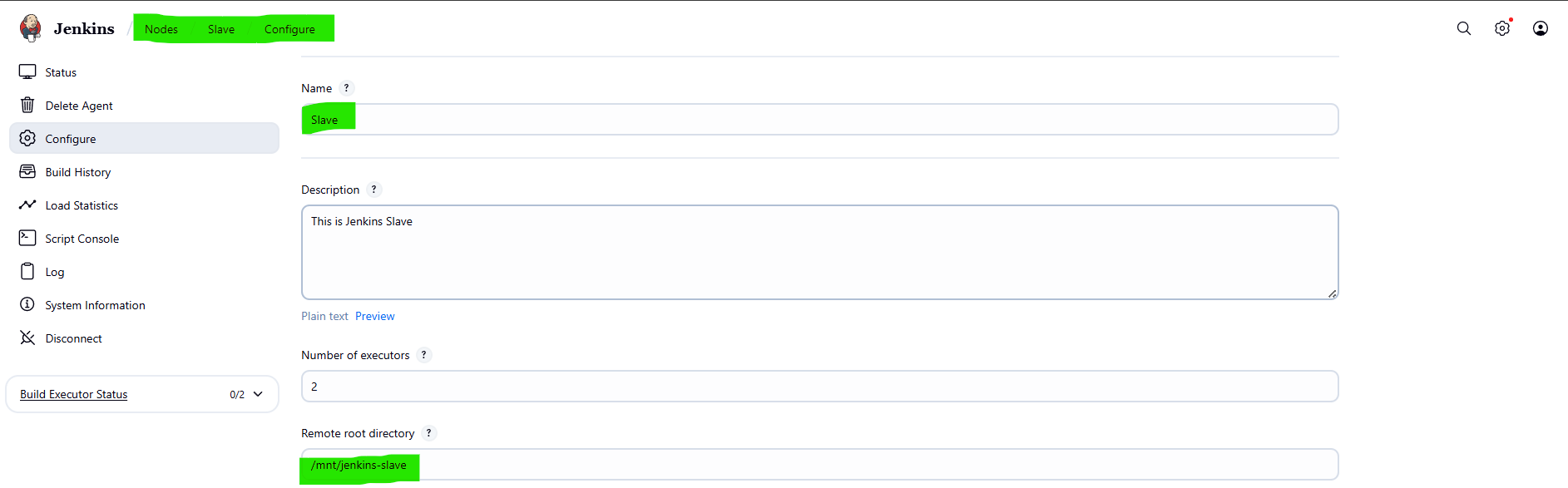
1. Creating connection between Jenkins-Master and Slave

go to manage Jenkins-> system



1. go to manage jenkins-> Nodes and configure the node

provide the path of Jenkins-slave in Remote root directory



In the host provide slave’s private ip address



and mention 0 in Disk space Monitoring Thresholds

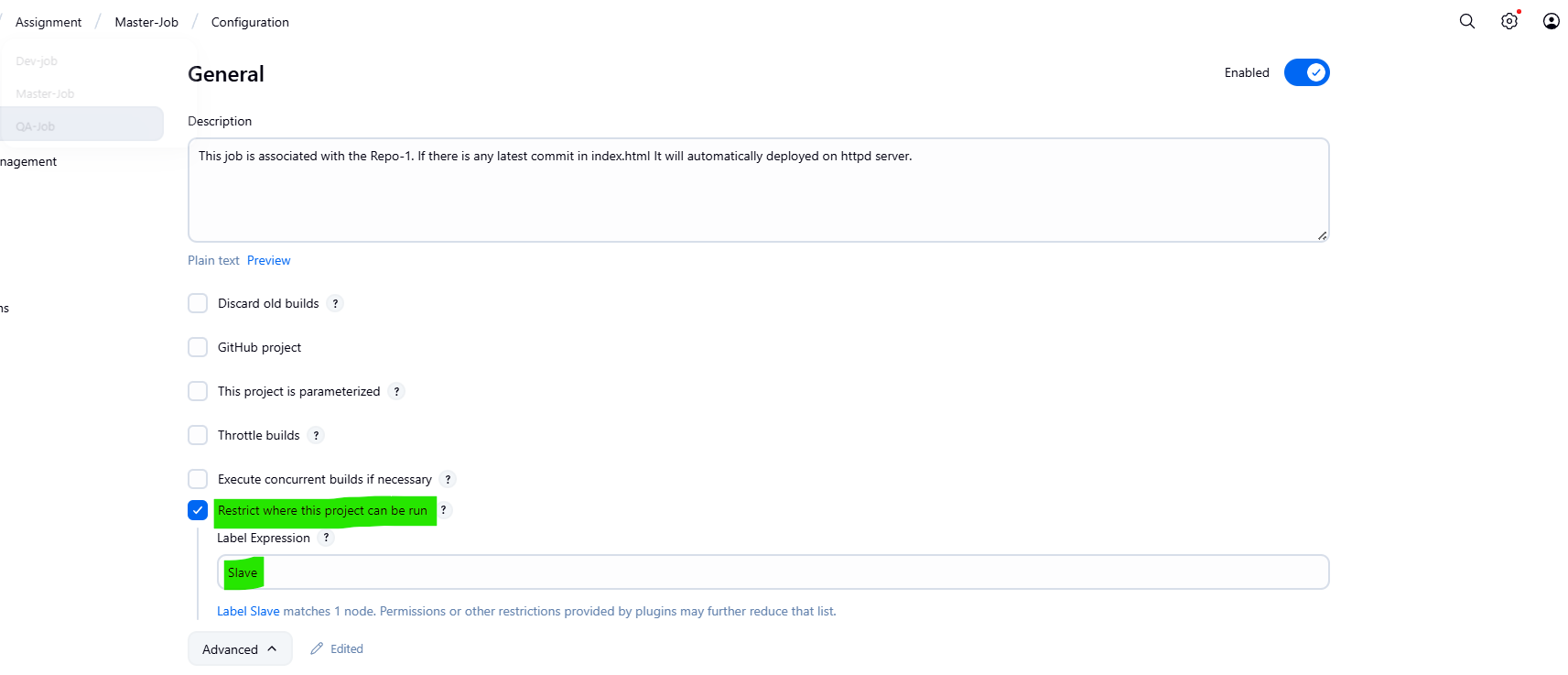


Then do Apply and Save and click on trush ssh

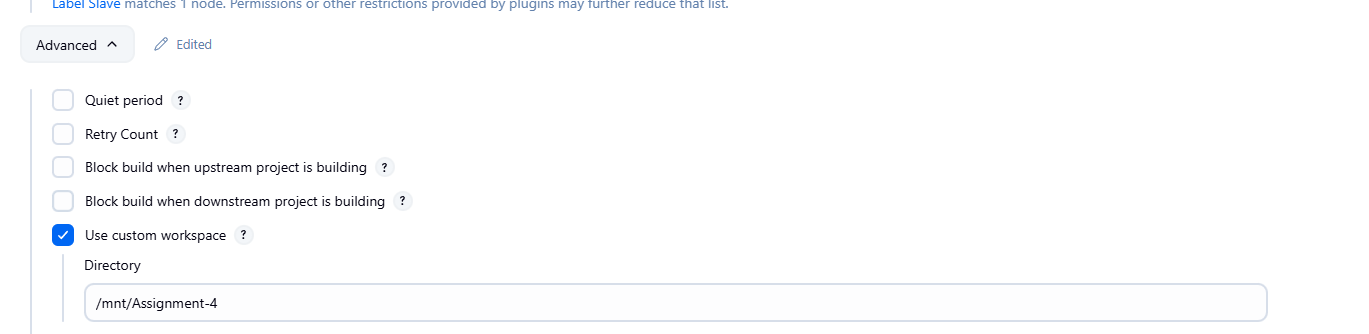


Now the connection has been established between Jenkins-Master and Slave

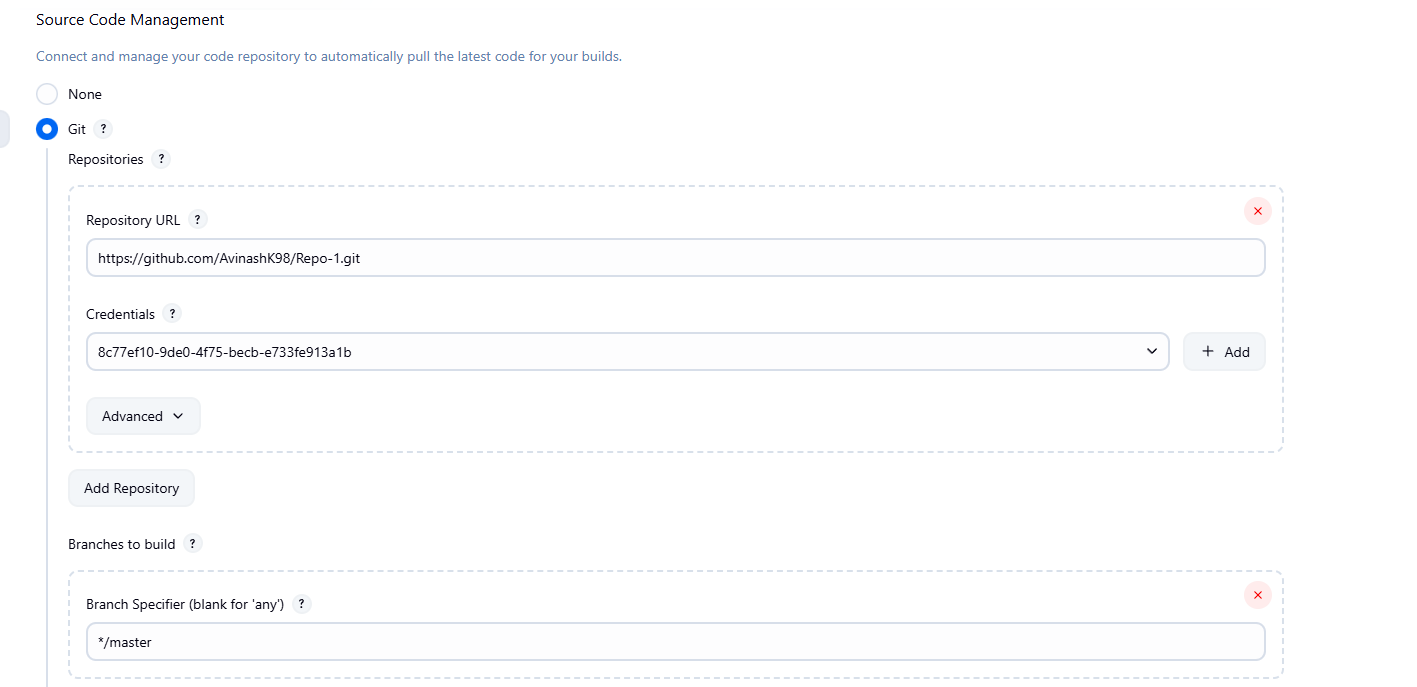
1. Create a Master-Job for Repo-1 which has index.html in master branch



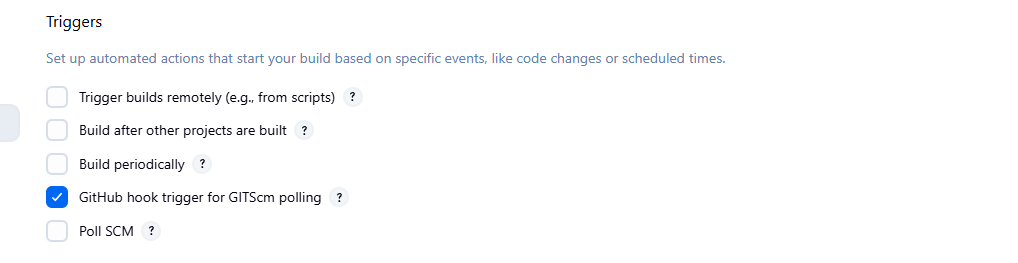
Use custom workspace for the job

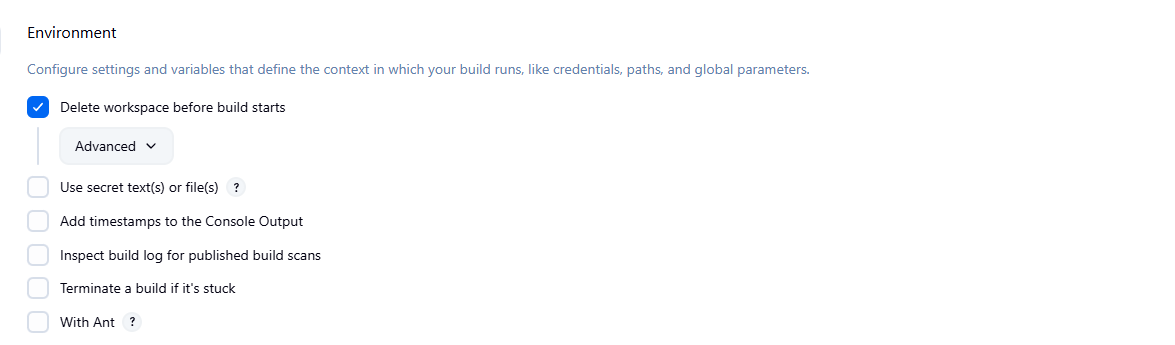


Provide Github Repo-1 url, crefdentials and specific branch



Use Github hook trigger for GITScm pooling, so whenever the new commit is happening in remote repository it will automatically pull the latest changes and deploy on the server



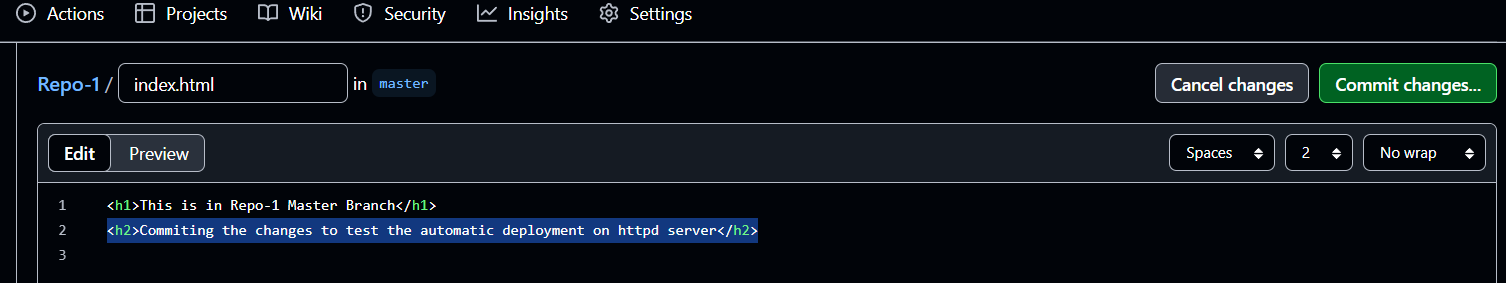


Using this shell script to grant the permission to html because the job will get triggered by Jenkins and it is an other user so need to provide the permissions. Then the next step will be copy the index.html file from workspace to the httpd server location

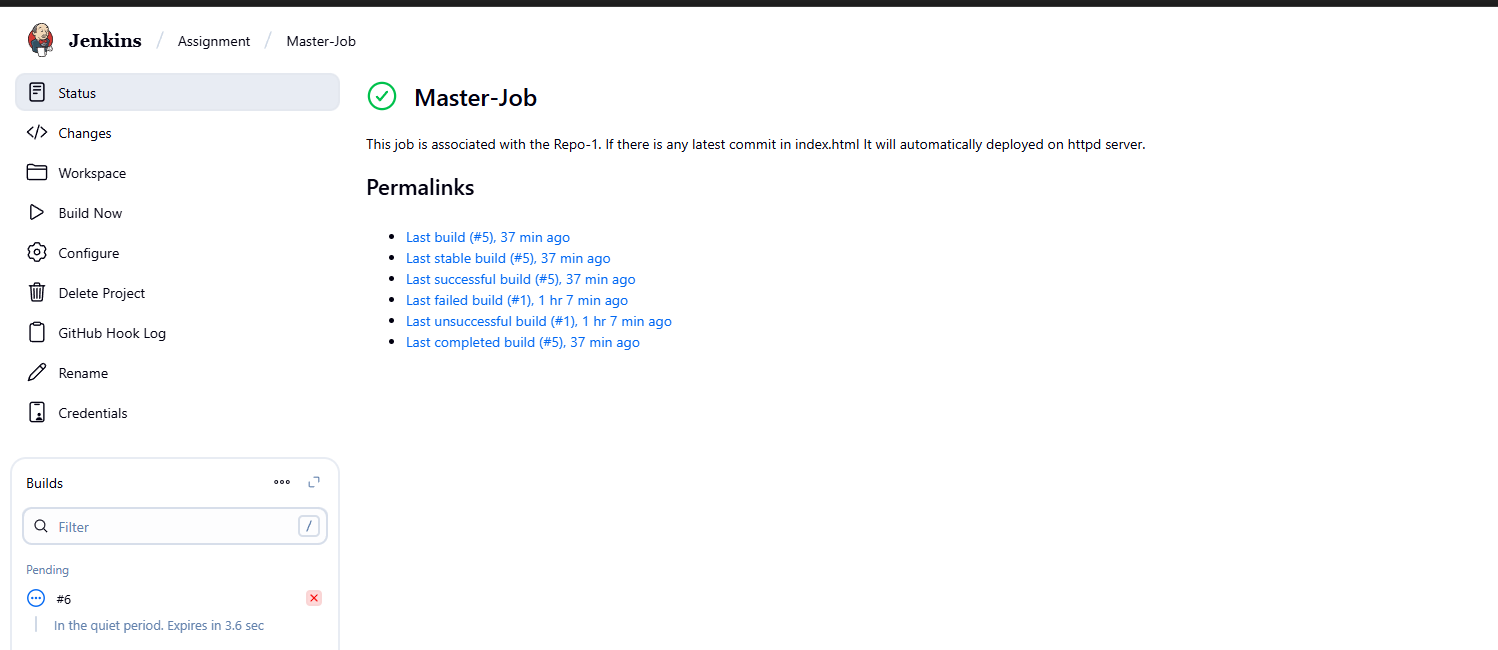


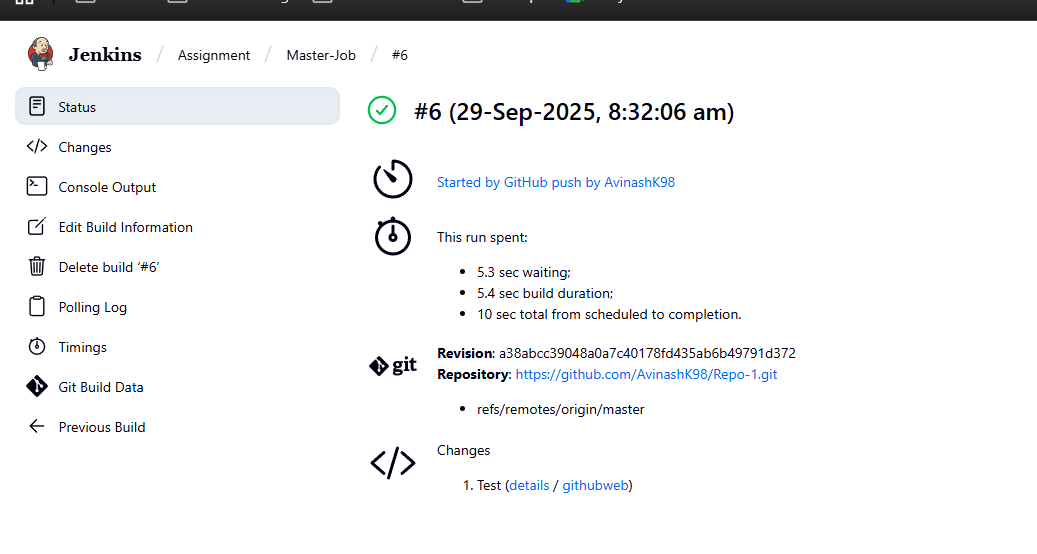
Now let’s do some changes in index.html which is in Repo-1 and check whether the job will start automatically and pull the latest changes from git hub and deploy it on the index.html on httpd server

I’ve add h2 line and done the commit



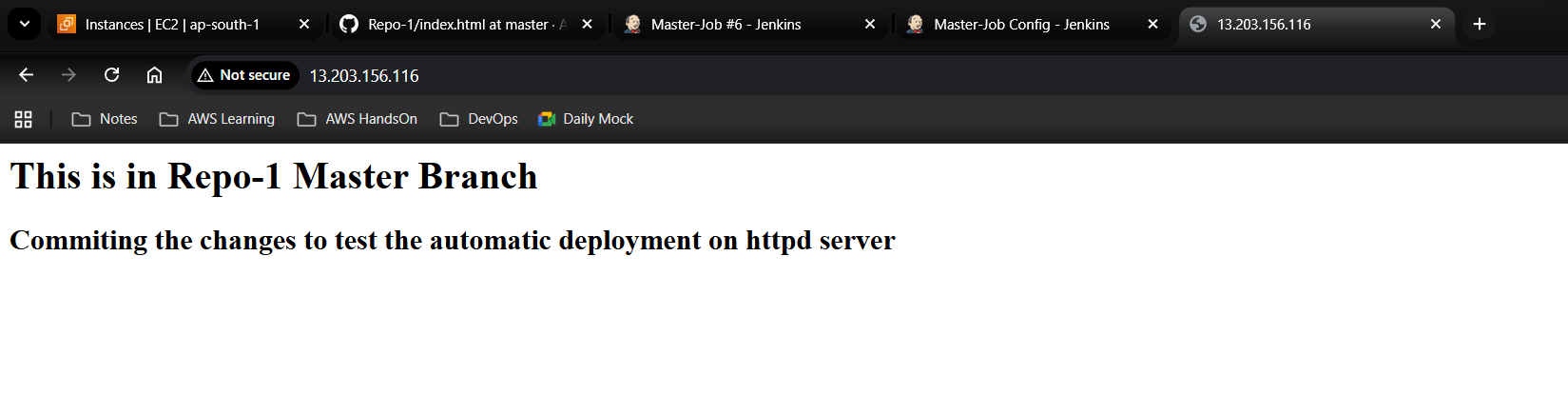
After commit, the job has been started automatically





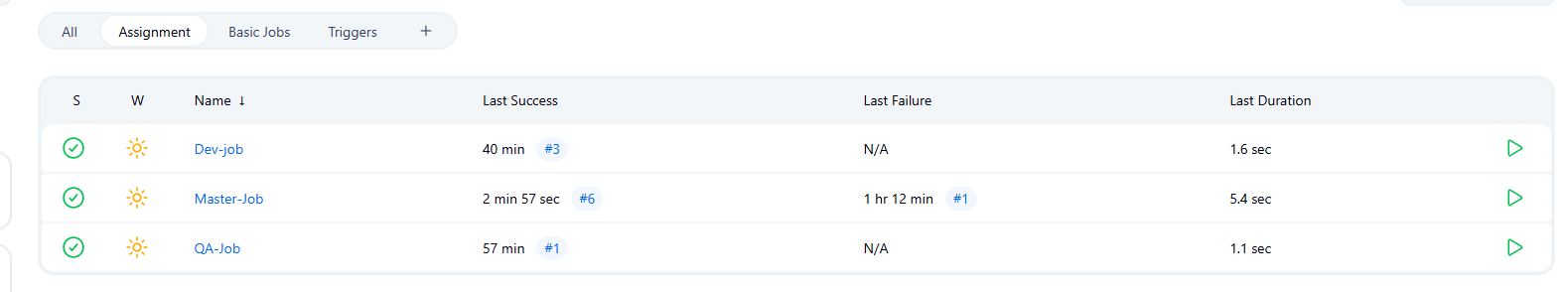
1. Check the webpage by using slave machine’s public ip

The changes have been reflected



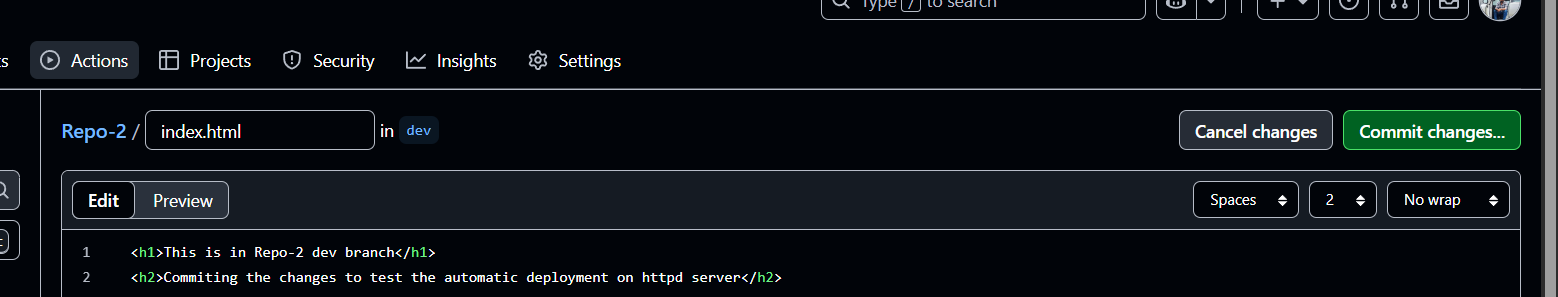
1. Do the same for Repo-2 Dev-Job and Repo-3 QA Job

All jobs are ready

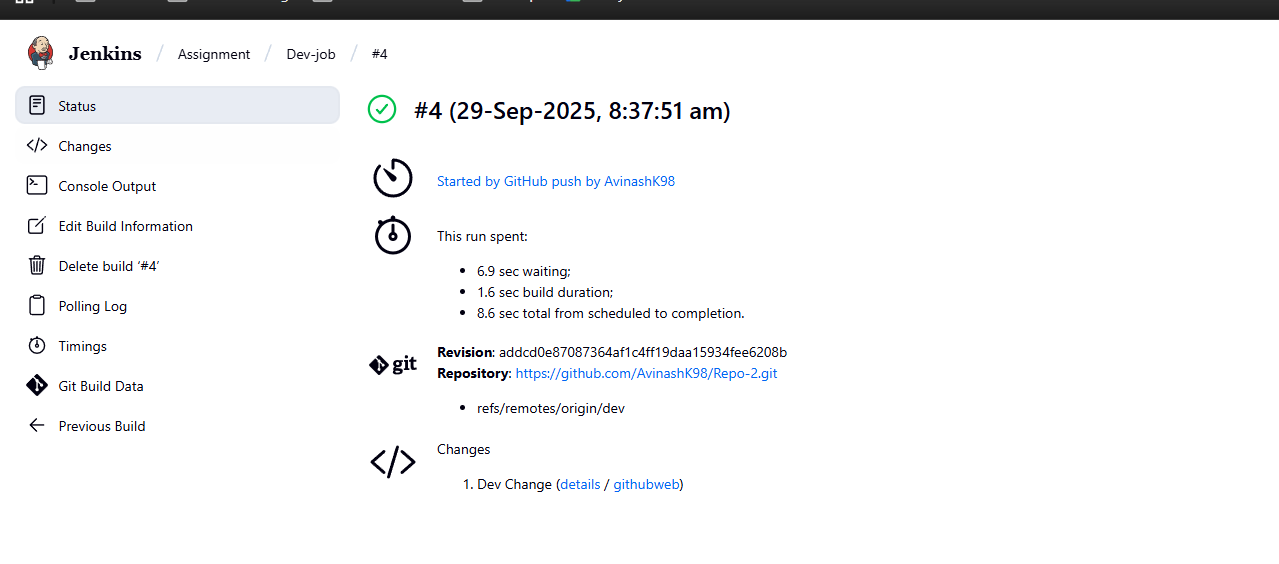


1. Doing some changes in index.html which is in Repo-2 dev branch

added h2 line and done commit

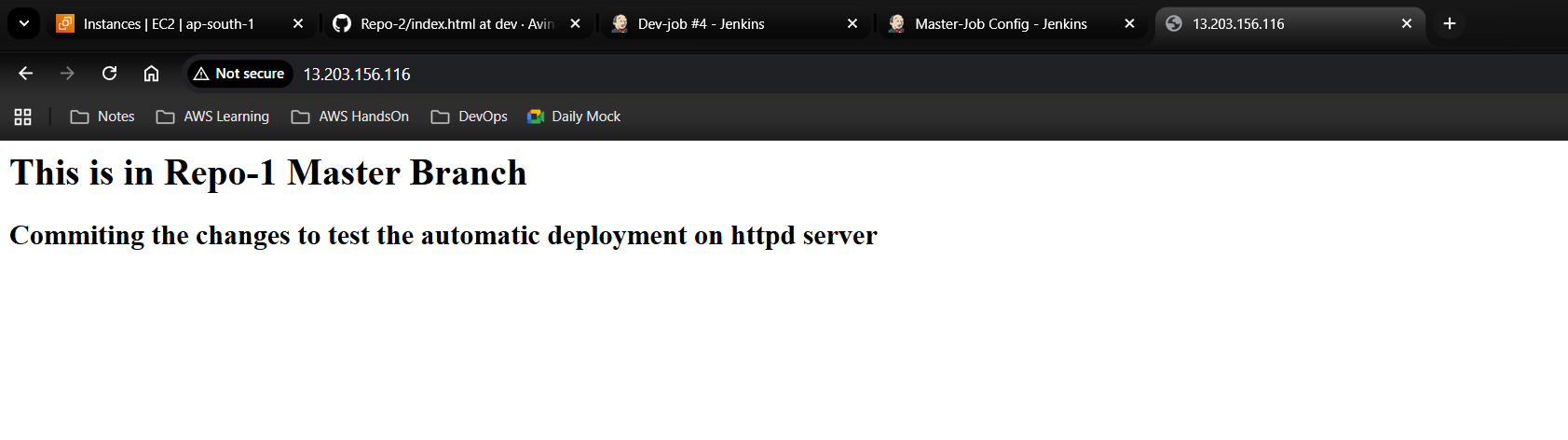


Dev-Job git triggered automatically and completed

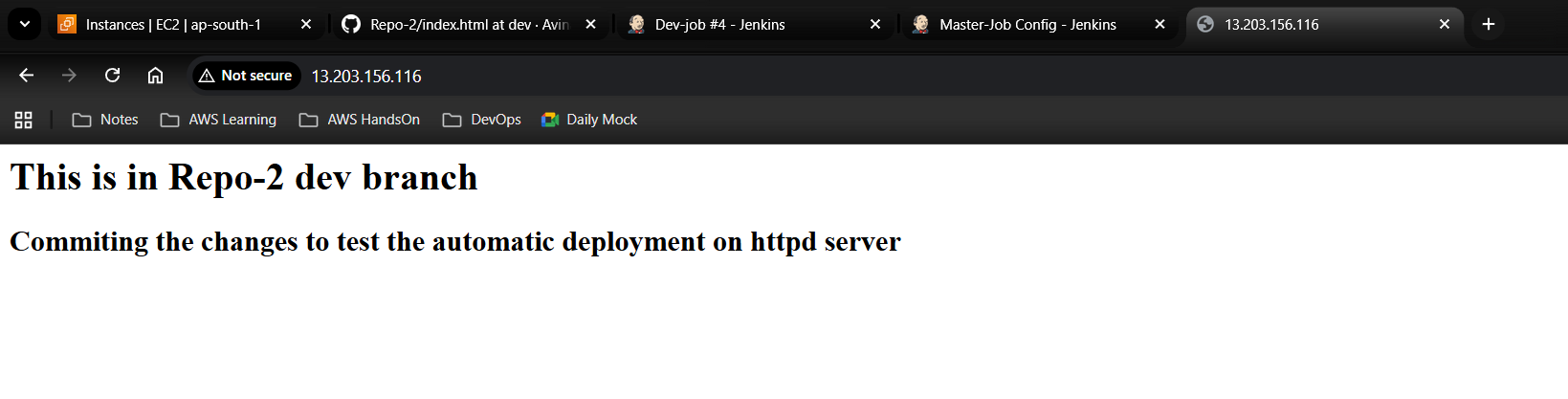


now refresh the url and check

before refresh

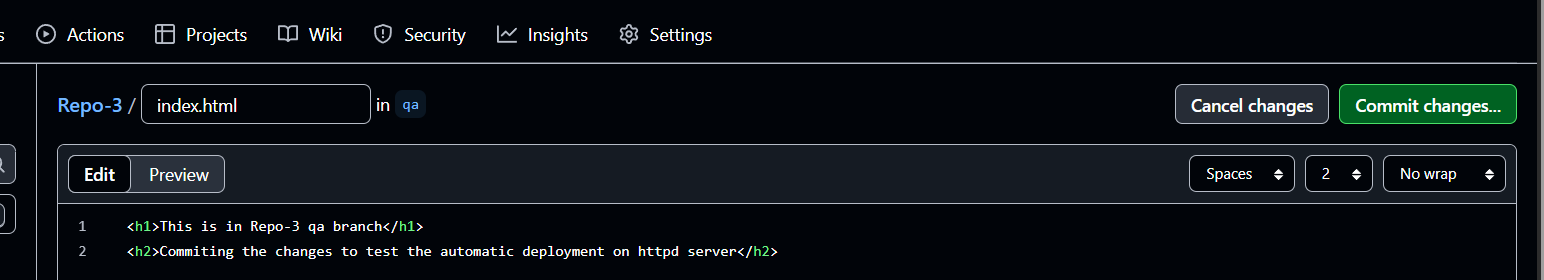


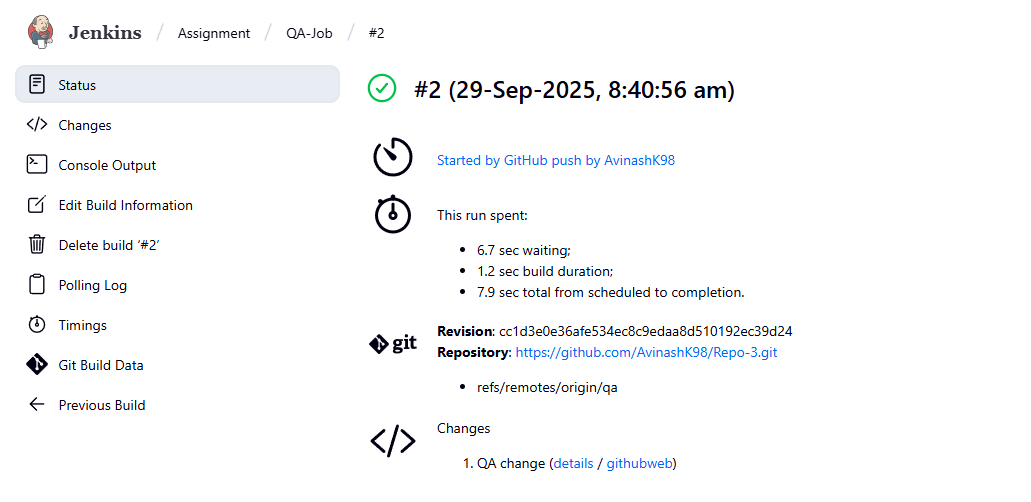
after refresh



1. Doing some changes in index.html which is in Repo-3 QA branch

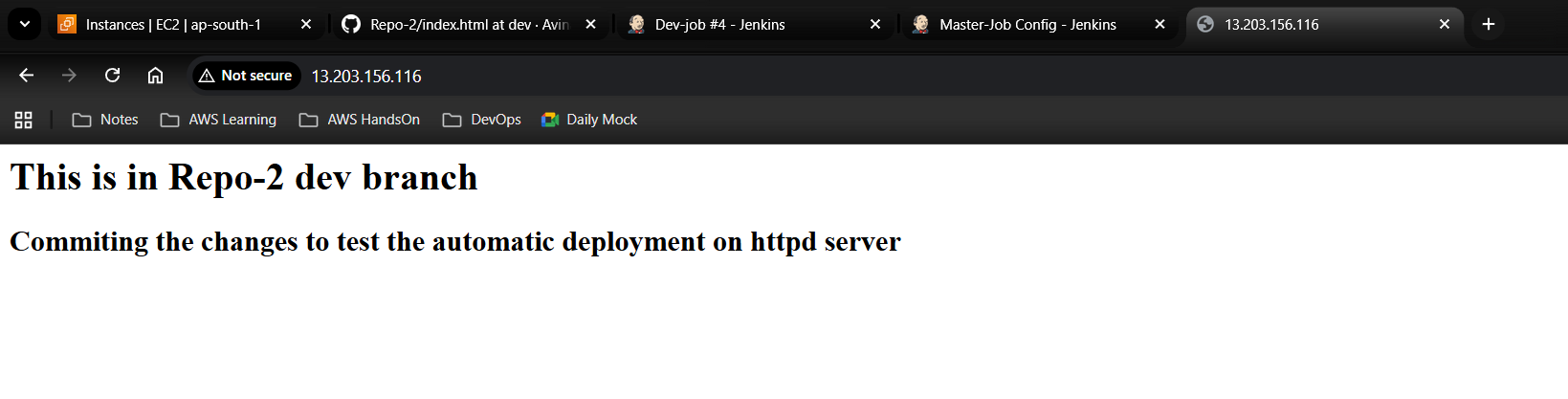
added h2 line and done commit





now refresh the url and check

before refresh



after refresh

