

Module 3: Build Jobs and Jenkins Security

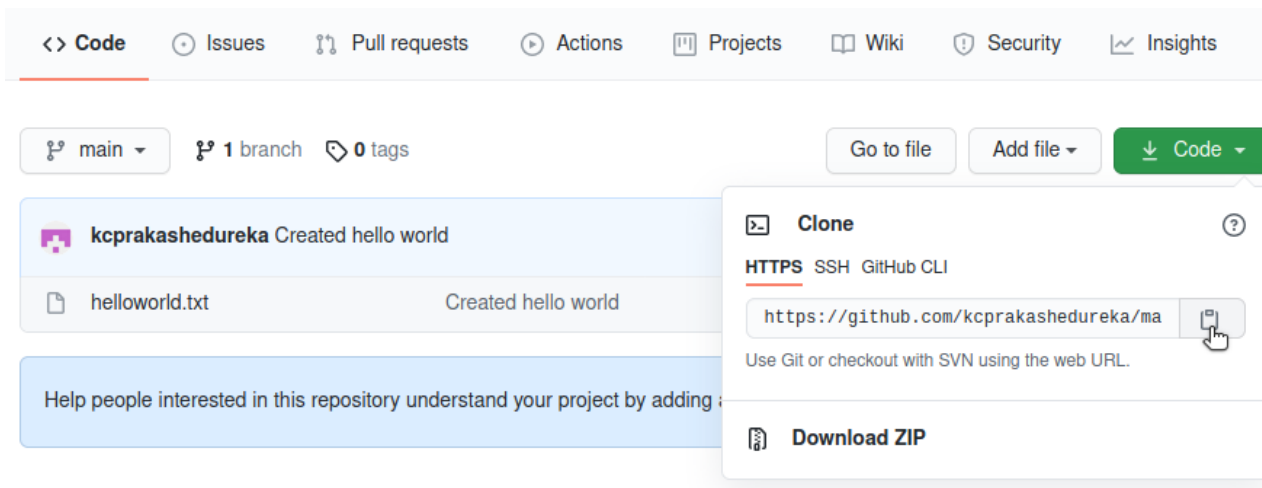
Demo Document - 3

edureka!

Demo: Creating a basic build job to clone a git and display the contents.

NOTE: - Refer **Module 3 Demo 1** and perform **steps 1 to 8** as mentioned in the document then follow the below steps. You can change the pipeline project name to 'Clone Repository Demo'.

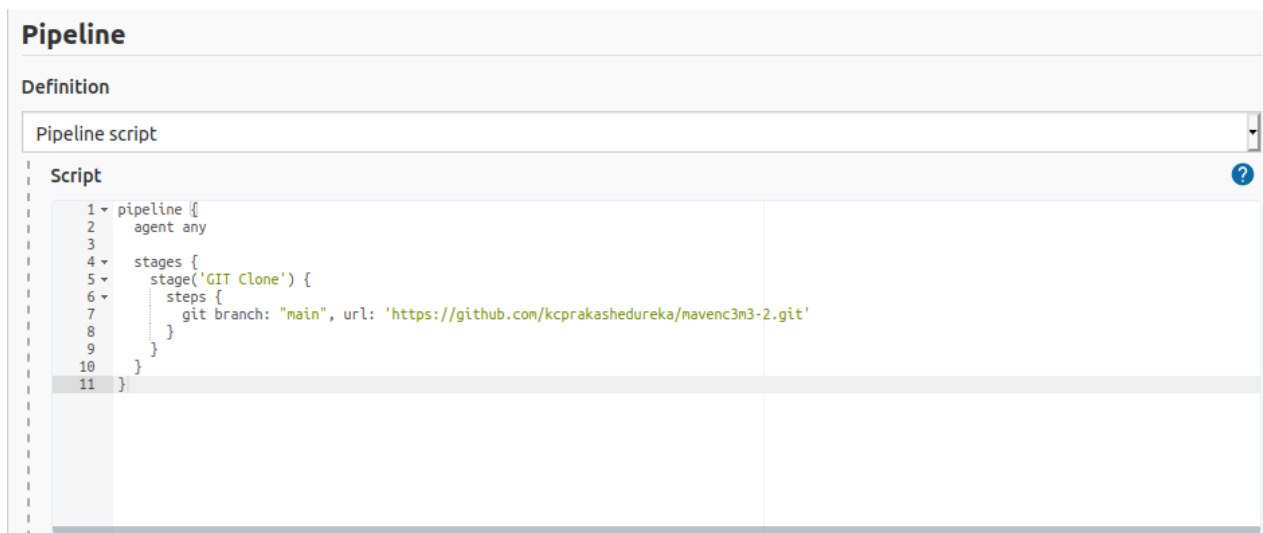
9. Login to your GITHUB account and select any remote repository that you have created in previous modules. Now click on 'Code' and copy the HTTPS GITHUB remote repository URL as mentioned in the screenshot below.



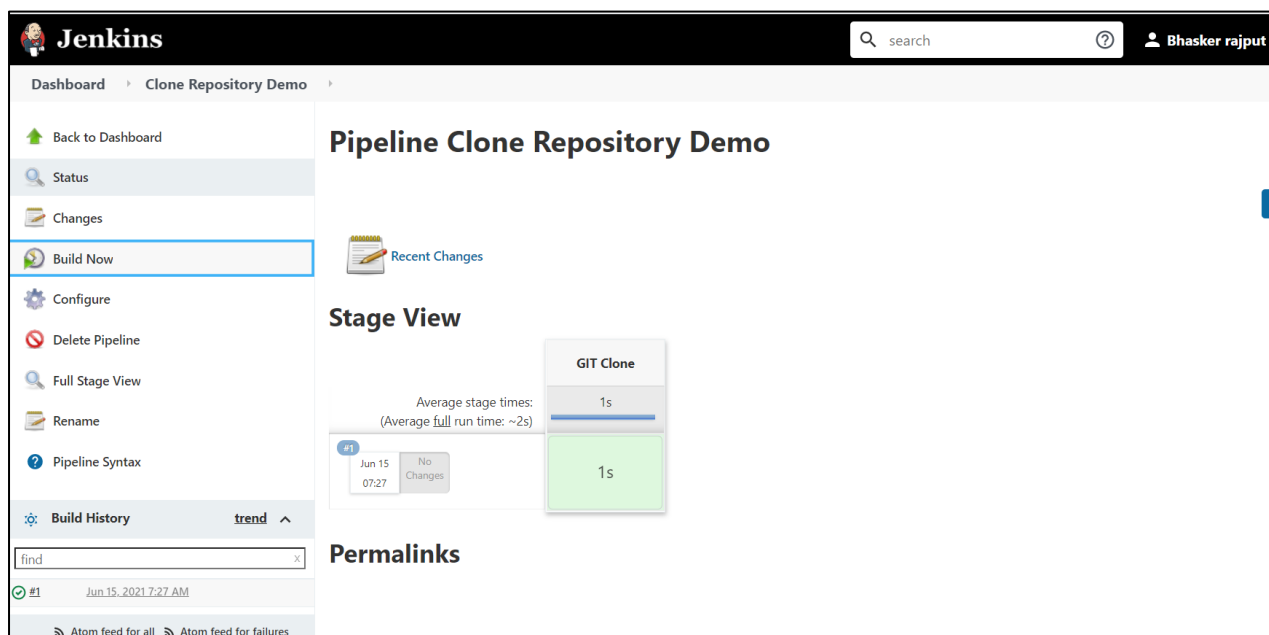
10. Now navigate back to JENKINS console and move to 'Advance Project Options' and under 'Pipeline' section write the below mentioned pipeline script. Now click on 'Appy' followed by 'OK' button.

```
pipeline {
  agent any

  stages {
    stage('GIT Clone') {
      steps {
        git branch: "main", url: 'https://github.com/kcprakashedureka/mavenc3m3-2.git'
      }
    }
  }
}
```



11. Now from the left side menu click on 'Build Now' to build the Jenkins pipeline. After few seconds, your pipeline builds successfully as mentioned in the below screenshot.



12. Now click on pipeline stage under 'stage view' and further click on logs to validate the pipeline message. In Logs you can clearly see that your Jenkins pipeline has clone the GITHUB repository as mentioned in the screenshot below.

Stage Logs (GIT Clone)

Git -- https://github.com/BhaskerRajput/gitpractical.git (self time 1s)

The recommended git tool is: NONE

No credentials specified

Cloning the remote Git repository

Cloning repository https://github.com/BhaskerRajput/gitpractical.git

> git.exe init C:\Users\bhask\jenkins\workspace\Clone Repository Demo # timeout=10

Fetching upstream changes from https://github.com/BhaskerRajput/gitpractical.git

> git.exe --version # timeout=10

> git --version # 'git version 2.32.0.windows.1'

> git.exe fetch --tags --force --progress -- https://github.com/BhaskerRajput/gitpractical.git +refs/heads/*:refs/remotes/origin/* # timeout=10

> git.exe config remote.origin.url https://github.com/BhaskerRajput/gitpractical.git # timeout=10

> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10

Avoid second fetch

> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10

Checking out Revision 506d95908104542a4fde65ec488c2d34ee0a874e (refs/remotes/origin/master)

> git.exe config core.sparsecheckout # timeout=10

> git.exe checkout -f 506d95908104542a4fde65ec488c2d34ee0a874e # timeout=10

> git.exe branch -a -v --no-abbrev # timeout=10

> git.exe checkout -b master 506d95908104542a4fde65ec488c2d34ee0a874e # timeout=10

Commit message: "test branch"

Build History

trend ^

find

07:27

Changes

15

Permalinks