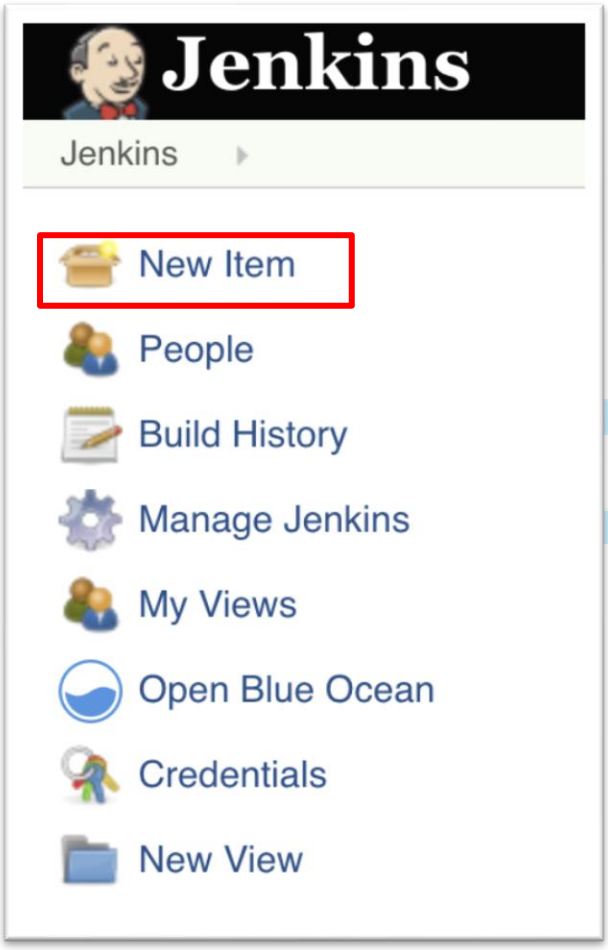
Module 4: Continuous Integration Using Jenkins

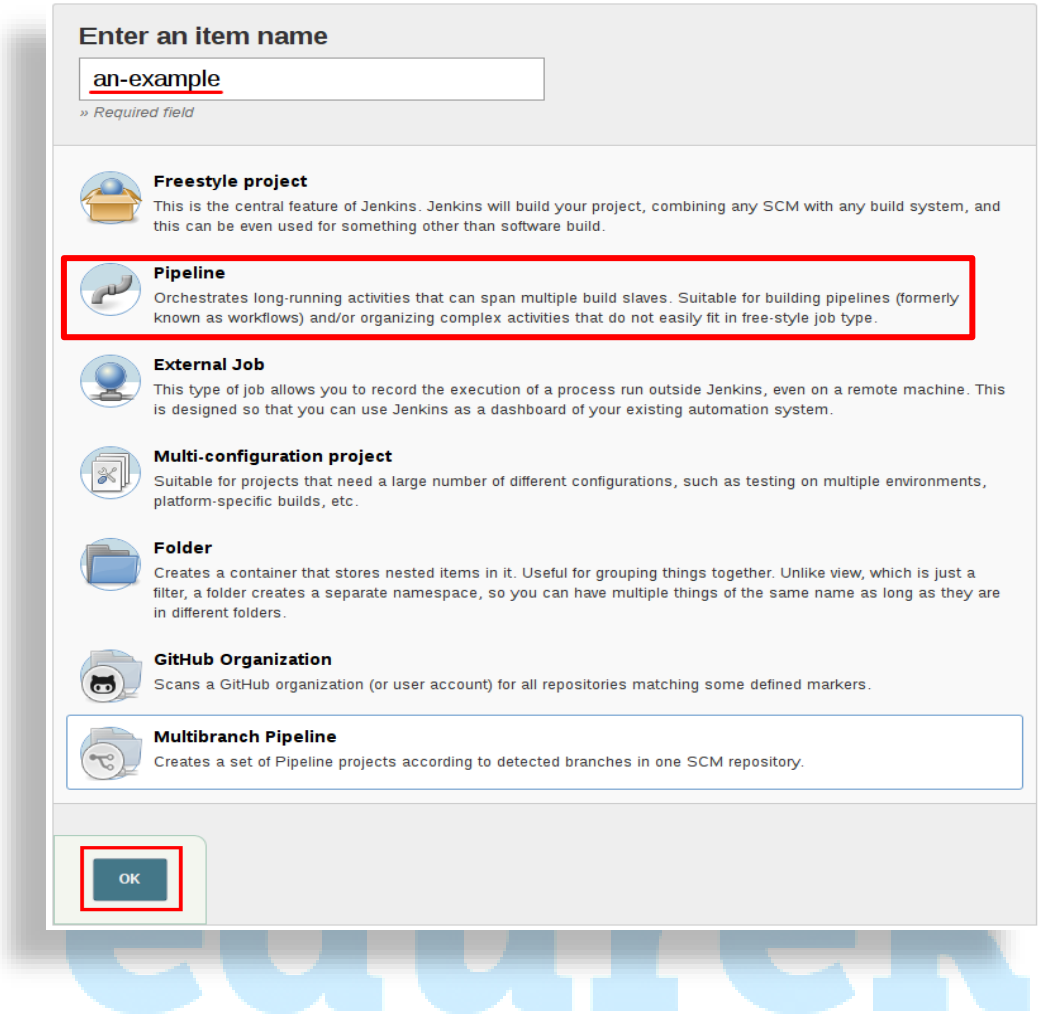
Demo Document

Demo 2: Build a Pipeline project using Groovy Scipt.

Step1: Create a Pipeline Project by navigating to Jenkins -> New Item

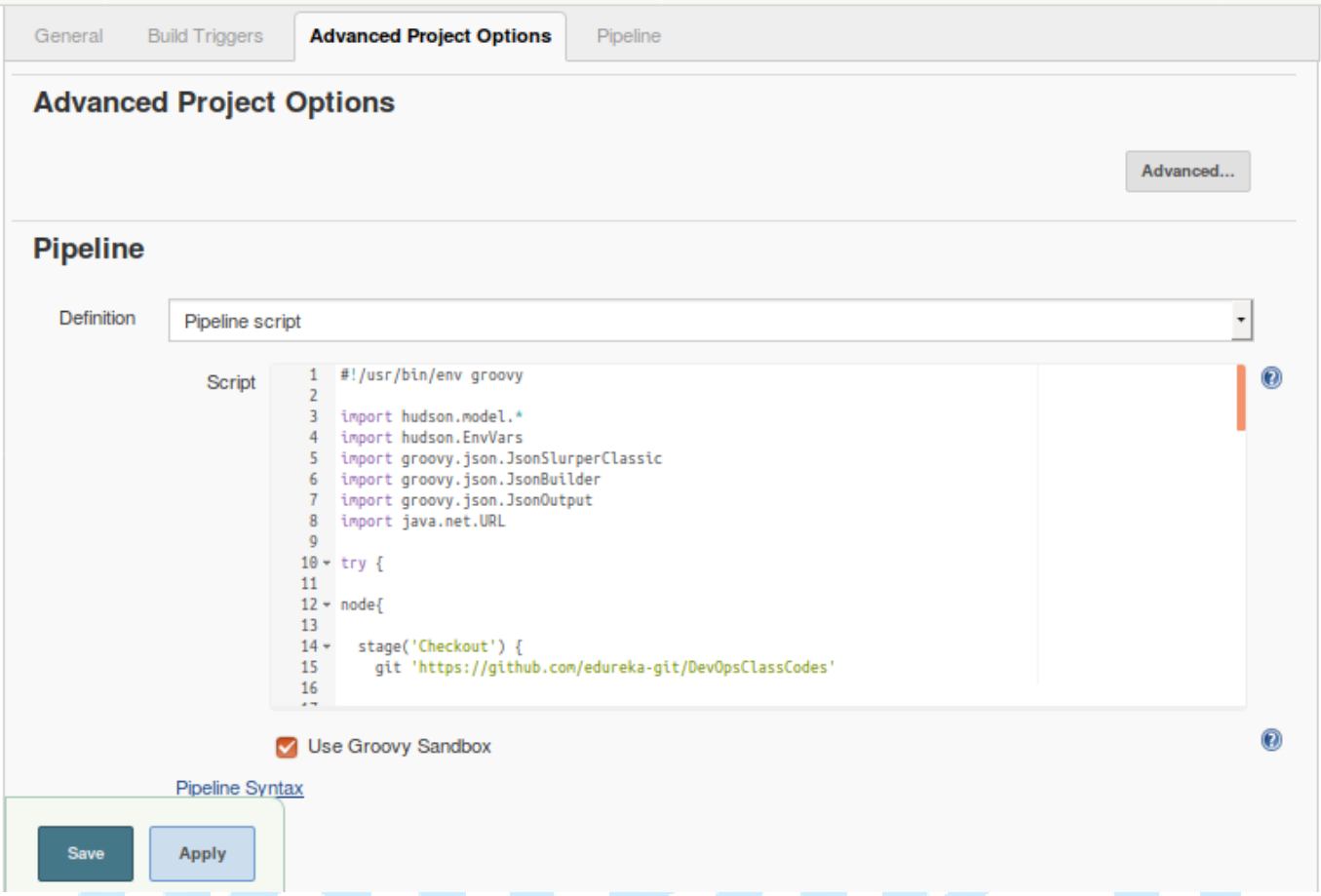


Step-2 Create a Pipeline project for your job.



Step-3: Go to your project and start configuring it.

Step-4: Go to the Advanced Project option and write your code within the Pipeline definition. The code is mentioned below if you want to copy and paste the code to generate the pipeline. Replace the “some email id” with the email_id you want to send email to if the build fails.




```
#!/usr/bin/env groovy

import hudson.model.*
import hudson.EnvVars
import groovy.json.JsonSlurperClassic
import groovy.json.JsonBuilder
import groovy.json.JsonOutput
import java.net.URL

try {

node{

  stage('Checkout') {
    git 'https://github.com/edureka-git/DevOpsClassCodes'
```

```
  }


  stage('Build') {
      dir('') {
        sh 'mvn -B -V -U -e clean package'
    }
  }


  stage ('Email') {
    emailext attachLog: true, body: 'The status of the build can be obtained
from the build log attached', subject: 'The build update is ', to: 'some
email id'
}


stage('Deployment') {
            // Deployment
                script {
                    echo "deployment"
                    sh 'cp
/var/lib/jenkins/workspace/package_1/target/addressbook.war
/opt/tomcat/webapps/'
            }
        }
        stage('publish html report') {
                echo "publishing the html report"
                publishHTML([allowMissing: false, alwaysLinkToLastBuild:
false, keepAll: false, reportDir: '', reportFiles: 'index.html', reportName:
'HTML Report', reportTitles: ''])
        }
        stage('clean up') {
                echo "cleaning up the workspace"
                cleanWs()
        }
```

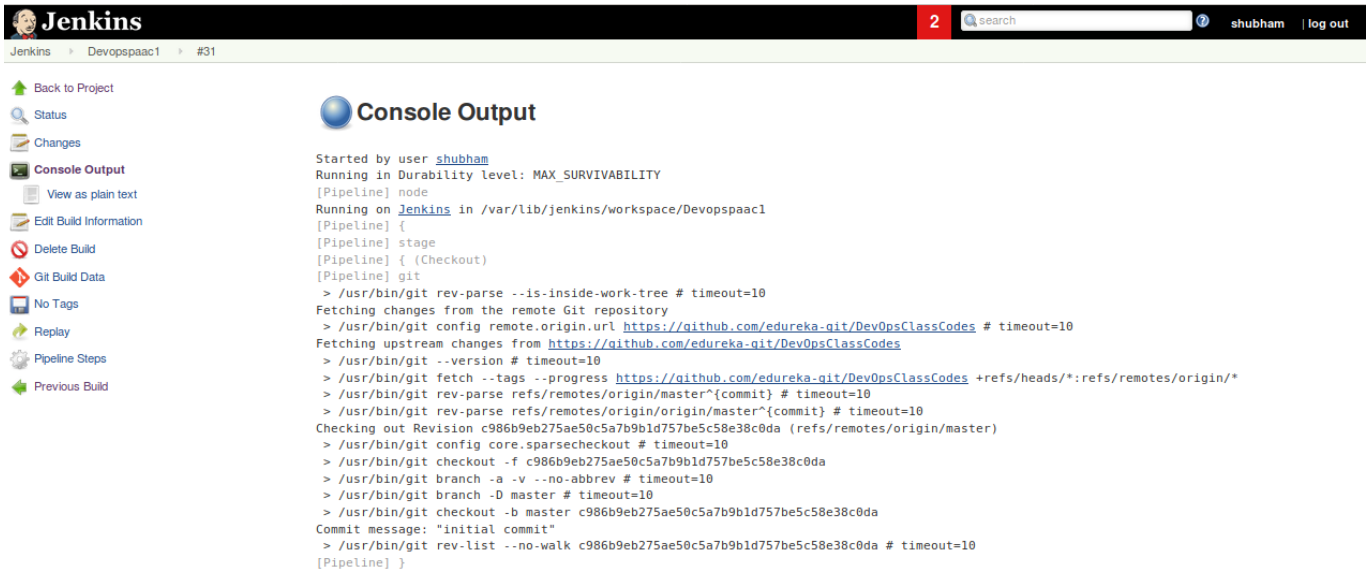
```
}// node
} // try end
finally {


 (currentBuild.result != "ABORTED") && node("master") {
         // Send e-mail notifications for failed or unstable builds.
     // currentBuild.result must be non-null for this step to work.
     step([$class: 'Mailer',
         notifyEveryUnstableBuild: true,
         recipients: 'some email id',
         sendToIndividuals: true])
 }
}
```

Step 5. If you are getting any error on deployment stage, make sure you are copying the war file to webapps folder correctly i.e., provide the war file location and tomcat directory path correctly.

Step 6. Then save the code and click on build now to build the project.

Step 7. If you're getting some error related to html report then go to manage jenkins->manage plugins->available and install html publisher plugin.

Step 8. Go to the console output and check the build of your project.



```
[Pipeline] stage
[Pipeline] { (publish html report)
[Pipeline] echo
publishing the html report
[Pipeline] publishHTML
[htmlpublisher] Archiving HTML reports...
[htmlpublisher] Archiving at PROJECT level /var/lib/jenkins/workspace/Devopspaac1 to /var/lib/jenkins/jobs/Devopspaac1/htmlreports
/HTML_20Report
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (clean up)
[Pipeline] echo
cleaning up the workspace
[Pipeline] cleanWs
[WS-CLEANUP] Deleting project workspace...[WS-CLEANUP] done
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Devopspaac1
[Pipeline] {
[Pipeline] step
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Step-9: Go back to your project and check the pipeline flow.

Pipeline Devopspaac1

Recent Changes

Stage View

	Checkout	Build	Email	Deployment	publish html report	clean up
Average stage times: (Average full run time: ~41s)	4s	11s	23s	338ms	239ms	111ms
#31 Oct 23 17:04 No Changes	4s	15s	1min 0s	569ms	404ms	200ms