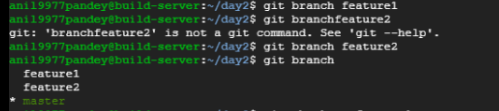
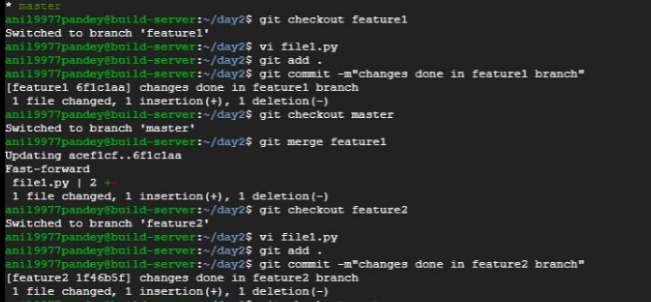
DevOps Bootcamp – Assignment 2

**Git:**

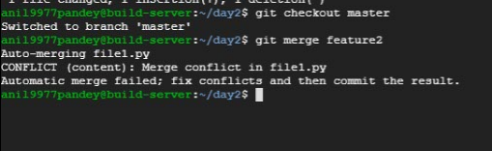
1. Create two separate branches from master



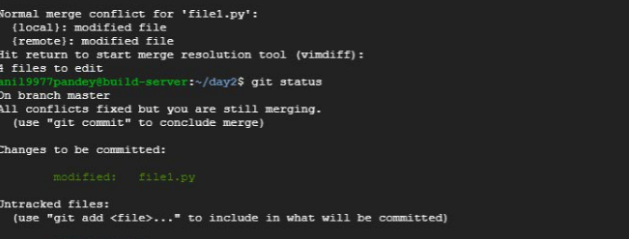
1. Make changes in the same function of the source code in both the branches
2. Merge Feature1 branch to master



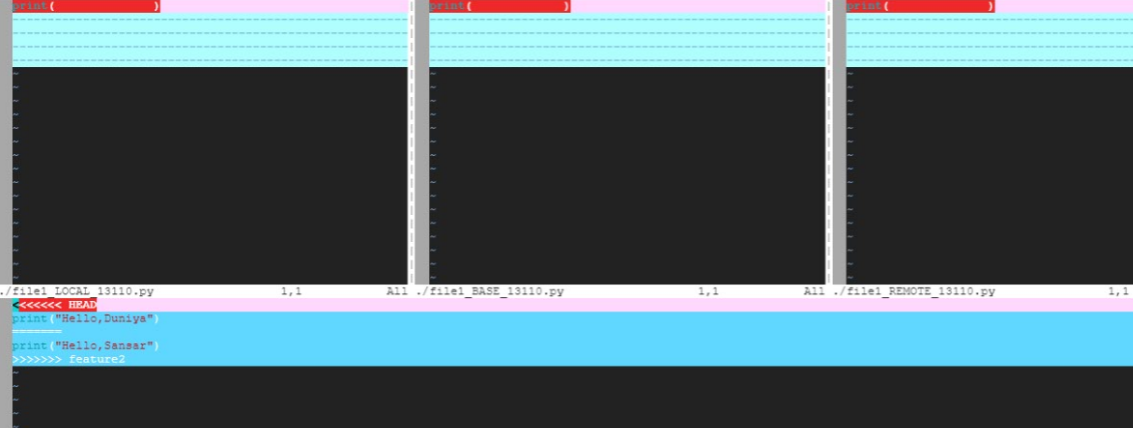
1. Try and merge branch2 into the master(merge conflict should arise)

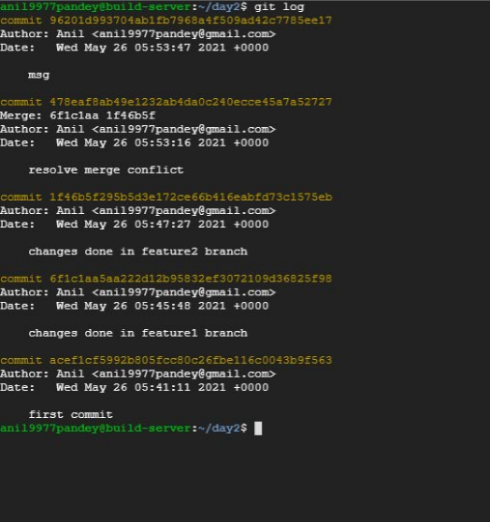


1. Install a merge tool of your choice and resolve the merge conflict using git merge tool command.



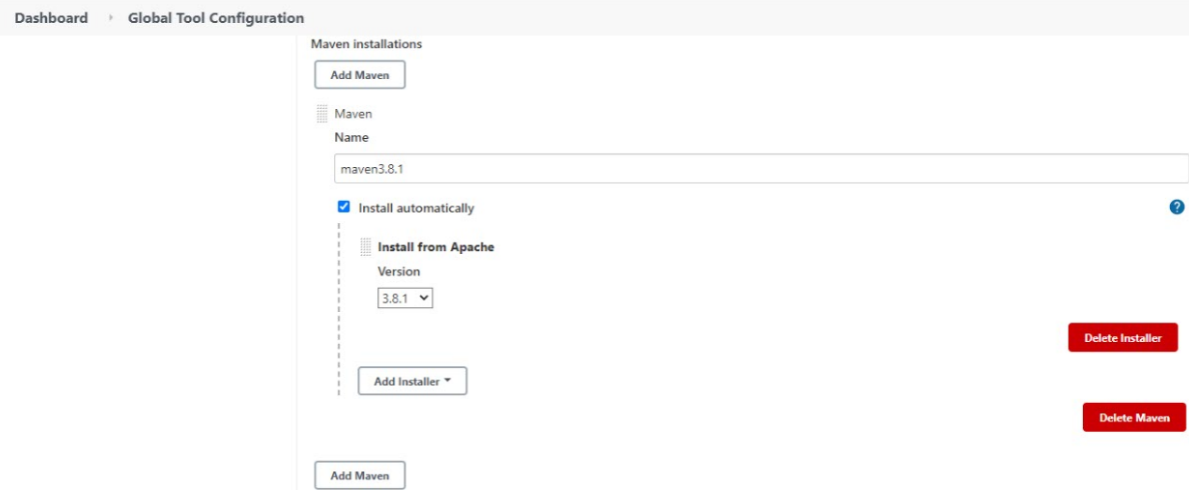
Merge Tool (Meld)





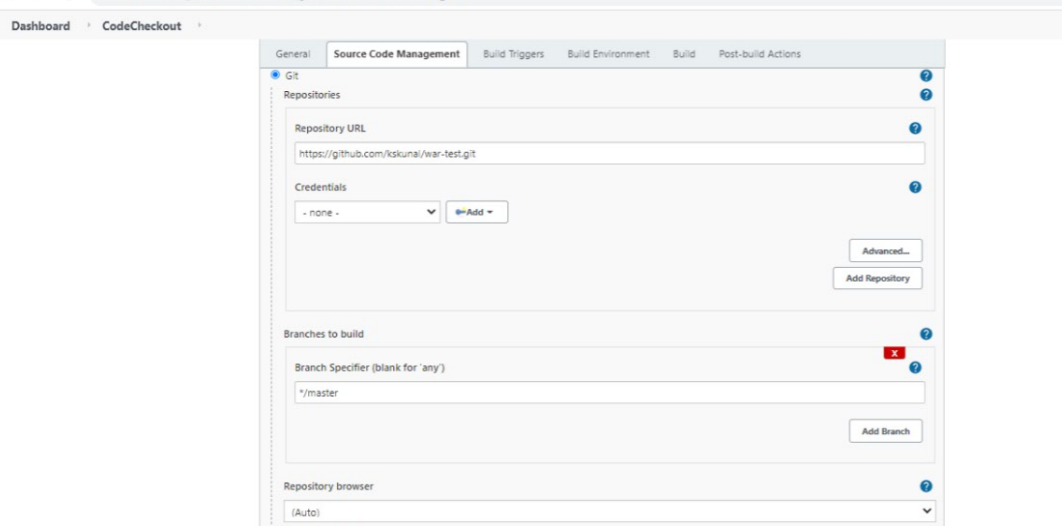
**Jenkins**

1. Create multiple freestyle project in Jenkins to checkout, build, test, package and deploy the web application in a tomcat9 server.
   1. Establish a build pipeline view with appropriate plugin installation.
   2. Configure maven as a tool in Jenkins environment.

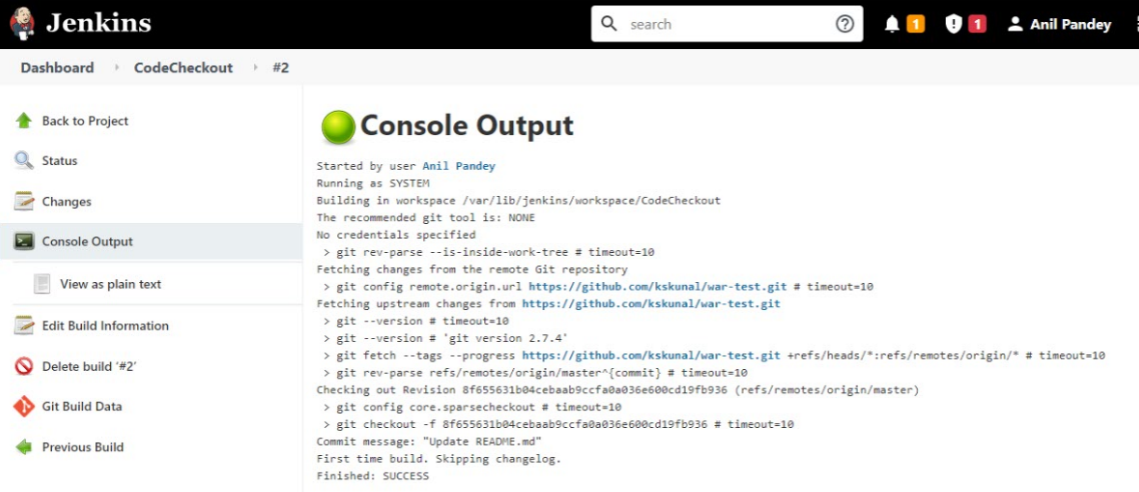


* 1. Attach the screenshot of the console output of each successful execution of the job.

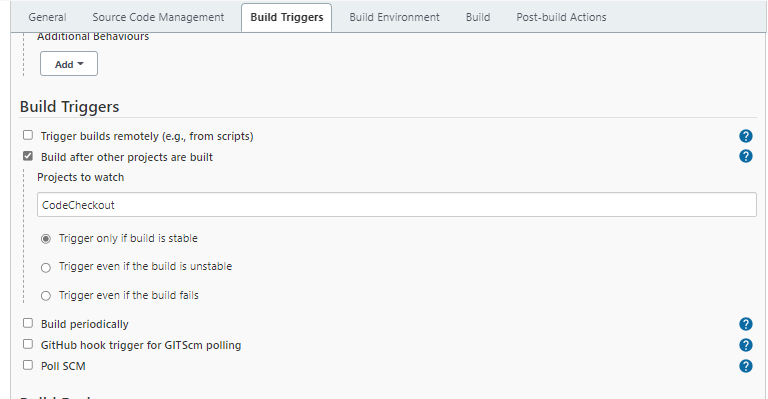
1. CodeCheckout

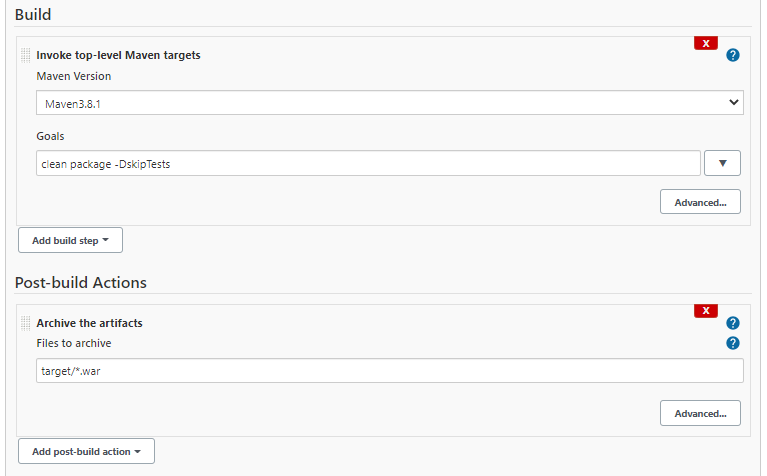


Output

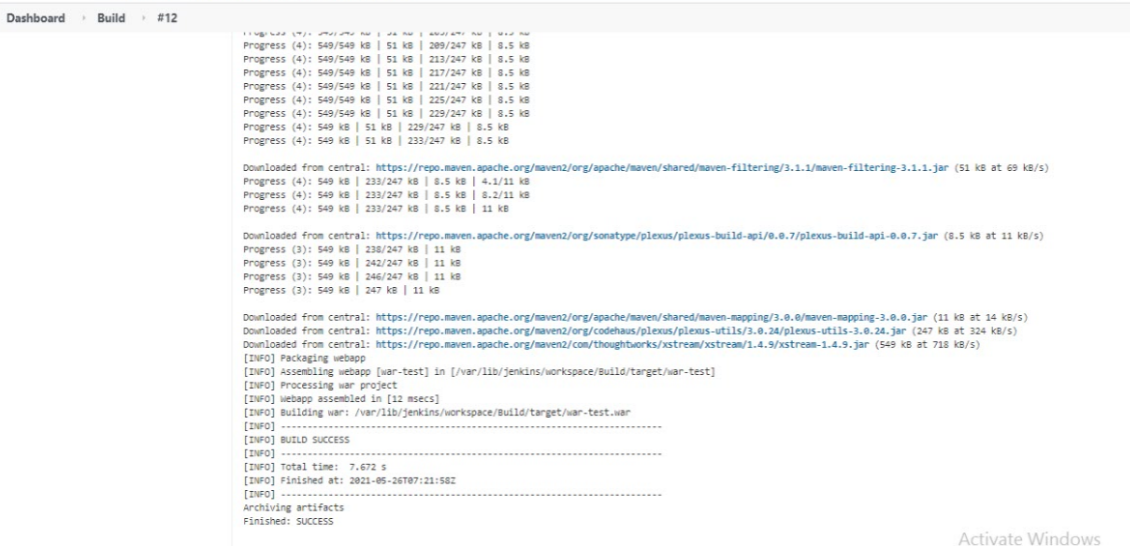


1. Build

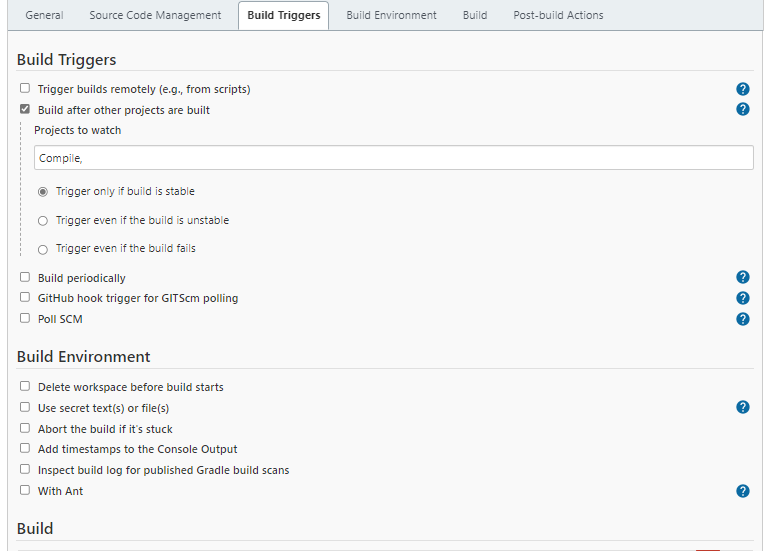


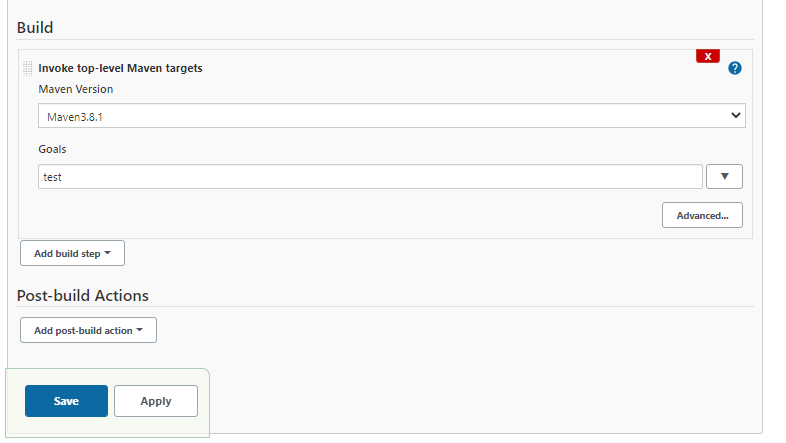


Output

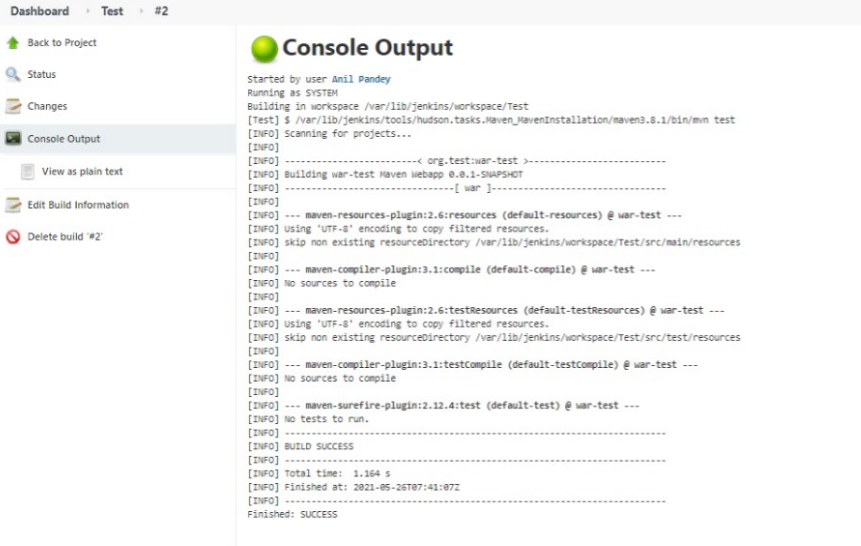


1. Test

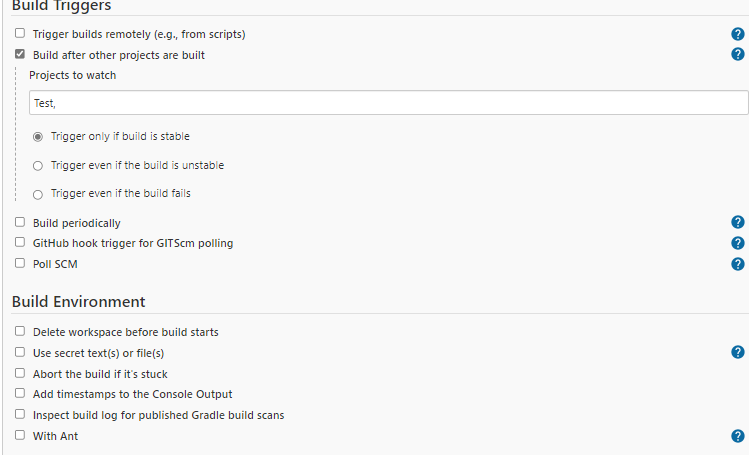


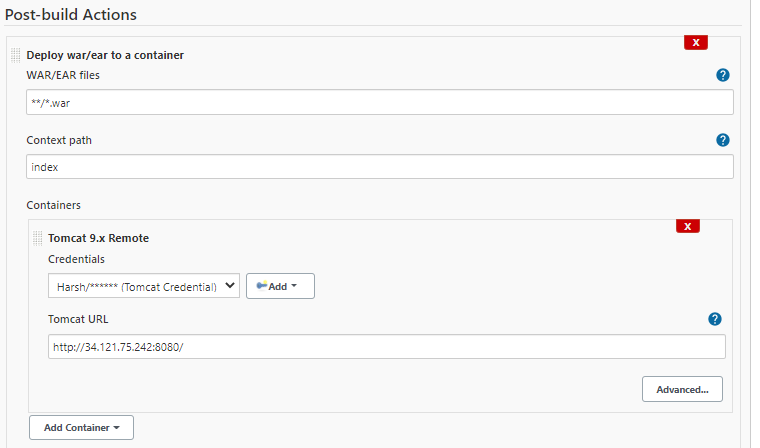


Output

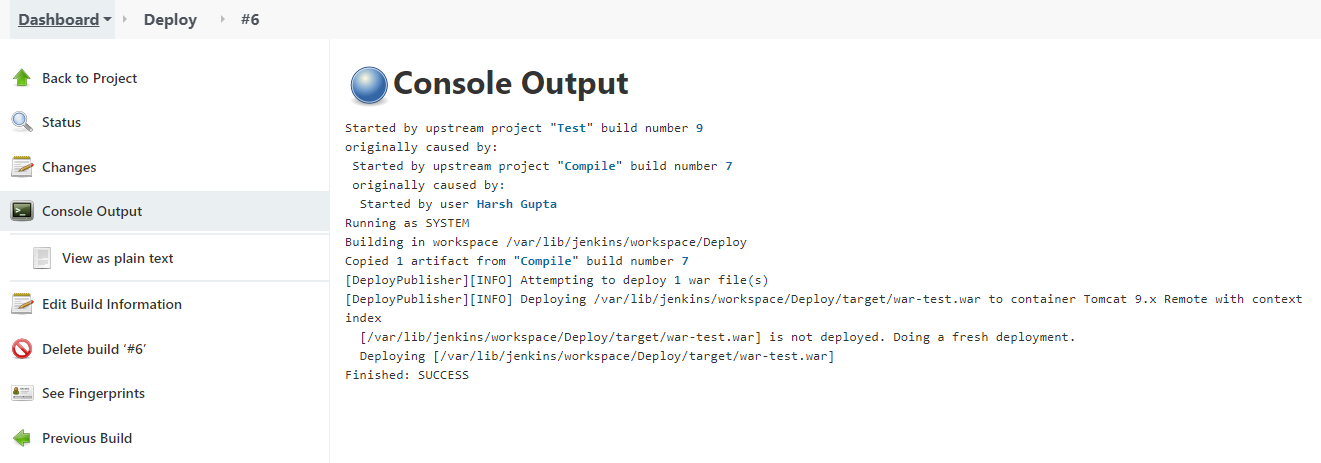


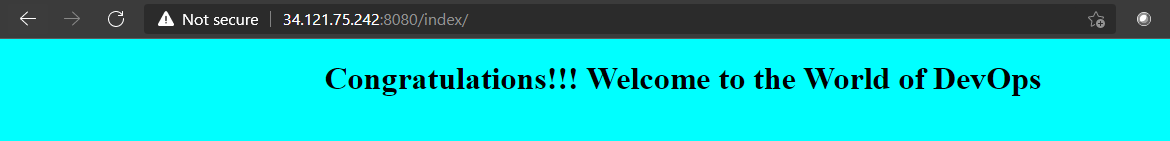
1. Deploy



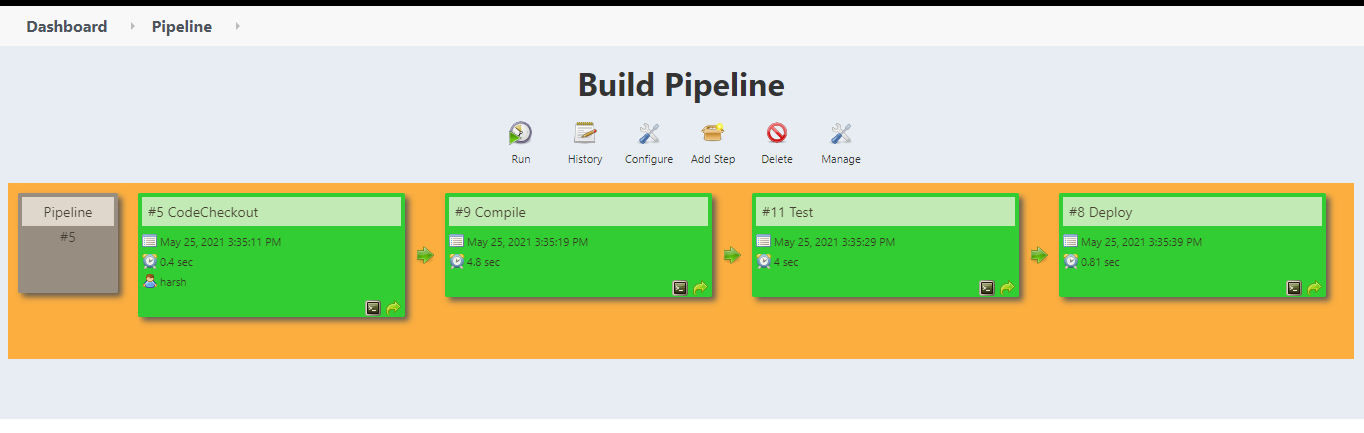


Output

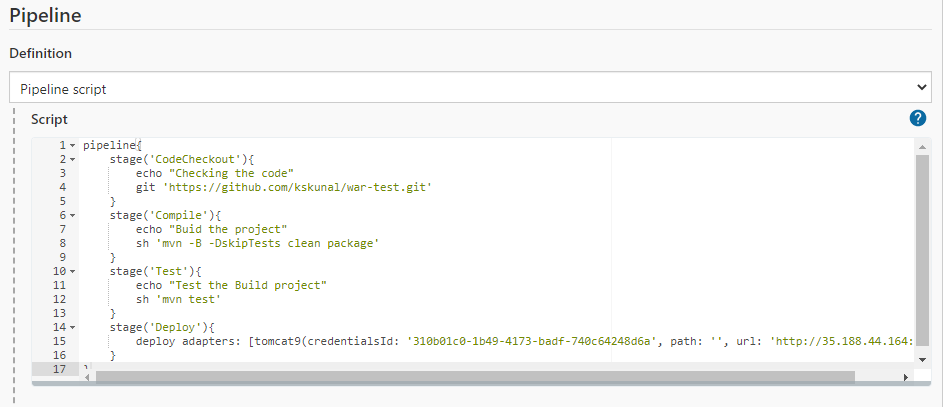




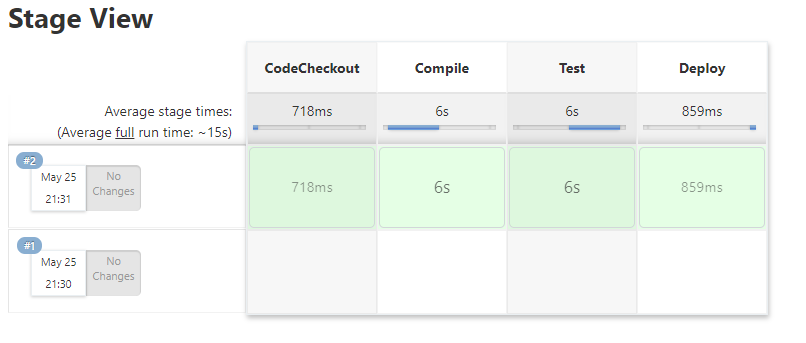
Below is the pipeline view configured and successfully run



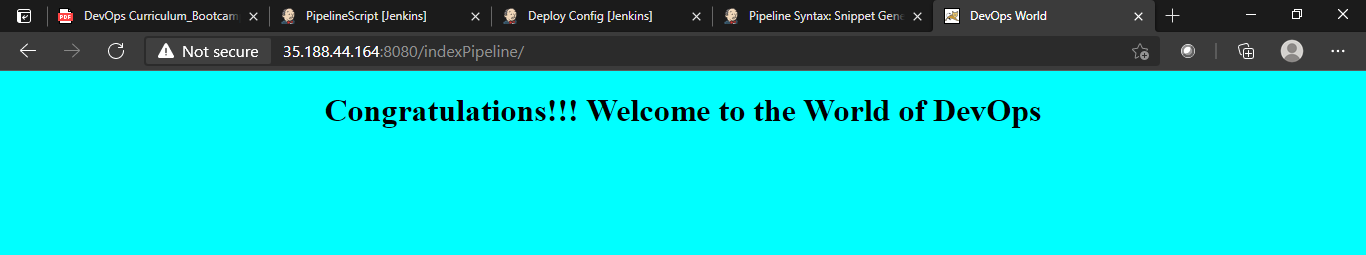
1. Create a pipeline job and write the script to achieve the same tasks. Pipeline must be triggered with a GitHub webhook upon every commit.
2. Created Pipeline Script



Successfully deployed

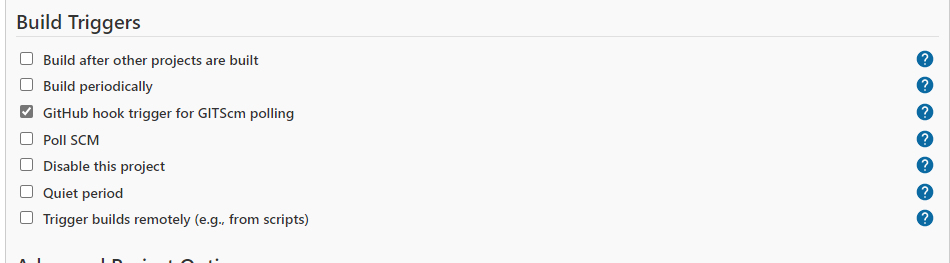


Output

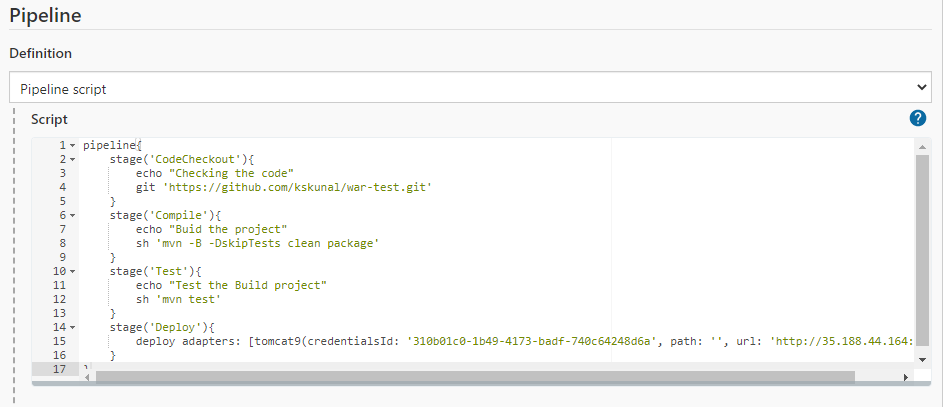


1. Created the pipeline job and enabled it to trigger for each commit

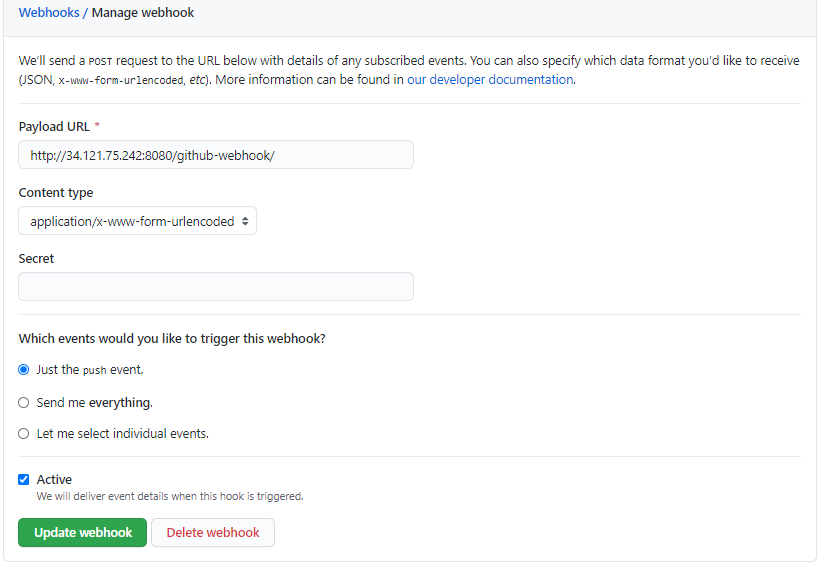
* Build GitHub Hook Tigger in Jenkins



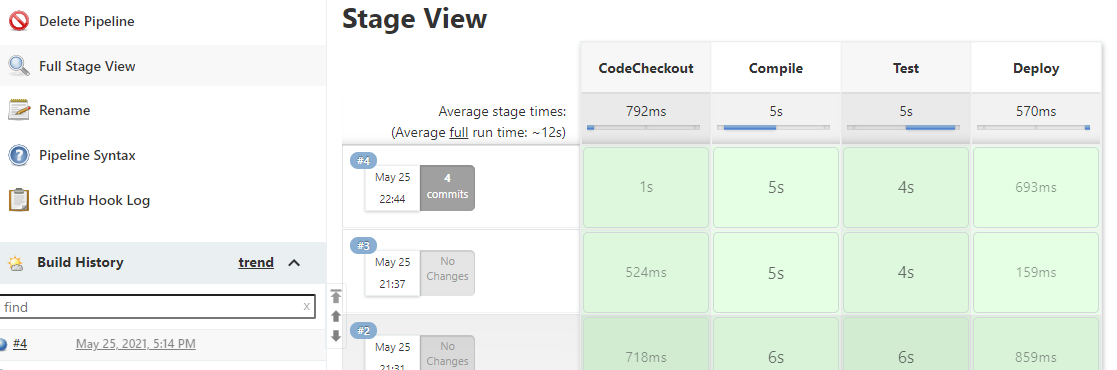
* Create a Script



* From GitHub Webhook configure



* Job start executing and Deployed successfully



Output

