

---

# Pipeline for our project

[Pipeline.post stages](#) Jenkinsfile  
x/Jenkinsfile

[artifacts](#)

```
pipeline {
  agent { label 'master' }
  stages {
    stage('build') {
      agent {
        docker {
          image 'python'
          args '-u root:sudo -v /var/lib/jenkins/store/demo-for-pipeline/:/store'
        }
      }
      steps {
        sh 'pip install -r requirements.txt'
        sh 'pytest --junitxml=/store/test-results/$BUILD_NUMBER.xml'

        sh 'DATE=date "+%Y-%m-%d--%H-%M-%S"; tar czf /store/artifacts/release-$DATE-$GIT_COMMIT.gz demo.py templates/'
      }
      post {
        always {
          sh 'git clean -fdx'
        }
      }
    }
    stage('deploy') {
      agent { label 'master' }
      steps {
        sh 'echo deploy'
        sh 'cd /home/gabor/work/demo-for-pipeline; /usr/bin/git pull'
        sh 'sudo /usr/sbin/service uwsgi reload'
      }
    }
  }
  post {
    cleanup {
      sh 'echo cleanup'
      dir("${env.WORKSPACE}@2") {
        deleteDir()
      }
      dir("${env.WORKSPACE}@2@tmp") {
        deleteDir()
      }
      dir("${env.WORKSPACE}@tmp") {
        deleteDir()
      }

      // sh 'rm -rf .pytest_cache/'
      // sh 'rm -rf __pycache__/'
      // sh 'rm -rf tests/__pycache__/'
      // sh 'rm -f *.pyc'
    }
  }
}
```