

## Docker assignment 3

Create Jenkins declarative pipeline to build and three httpd container name c1,c2,c3 with port 80,90,8001 respectively from single repo which has three branch 2025Q1, 2025Q2, 2025Q3 and each has serving unique index.html file and verify the access of each here also add the deployment with docker volume

Dashboard > docker-assignment > docker-assignment-3 >



Status



docker-assignment-3

</> Changes



Build Now



Configure



Delete Pipeline



Stages



Rename



Pipeline Syntax

### Permalinks

- Last build (#4), 1 min 39 sec ago
- Last stable build (#4), 1 min 39 sec ago
- Last successful build (#4), 1 min 39 sec ago
- Last failed build (#2), 13 min ago
- Last unsuccessful build (#2), 13 min ago
- Last completed build (#4), 1 min 39 sec ago

Builds



Filter



Today



#4 3:30 PM



#3 3:25 PM



Dashboard > docker-assignment > docker-assignment-3 > Configuration

### Configure



General



Triggers



Pipeline



Advanced

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Definition

Pipeline script

Script ?

```
1 pipeline {
2   agent {
3     label 'built-in'
4   }
5   stages {
6     stage('Clone') {
7       steps {
8         git url: 'https://github.com/Poojak134/docker-httpd.git', branch: 'main'
9       }
10    }
11    stage('Pull') {
12      steps {
13        sh '''
14          sudo docker pull httpd:latest
15          sudo docker rm -f c1 c2 c3 || true
16        '''
17      }
18    }
19  }
20 }
```

☒ Use Groovy Sandbox ?

Graph

```
graph LR; Start((Start)) --> Clone((Clone)); Clone --> Pull((Pull)); Pull --> SetupContainers[Setup Containers]; SetupContainers --> End((End)); SetupContainers --> c1[c1-branch 2025Q1]; SetupContainers --> c2[c2-branch 2025Q2]; SetupContainers --> c3[c3-branch 2025Q3];
```

Clone 0.73 sec

Pull 2.7 sec

Setup Containers 4.3 sec

c1-branch 2025Q1 4.2 sec

c2-branch 2025Q2 4.2 sec

c3-branch 2025Q3 0.77 sec

← → ↻ ⚠ Not secure 13.201.166.222:8081/index.html

## Script

```
pipeline {
  agent {
    label 'slave'
  }
  stages {
    stage('Clone') {
      steps {
        git url: 'https://github.com/Poojak134/docker-httpd.git', branch: 'main'
      }
    }
    stage('Pull') {
      steps {
        sh '''
          sudo docker pull httpd:latest
          sudo docker rm -f c1 c2 c3 || true
        '''
      }
    }
    stage('Setup Containers') {
      parallel {
        stage('c1-branch 2025Q1') {
          steps {
            dir('workspace-c1') {
              git url: 'https://github.com/Poojak134/docker-httpd.git', branch: '2025Q1'
              sh '''
                sudo docker run -dp 80:80 --name c1 -v c1-webdata:/usr/local/apache2/htdocs
                httpd:latest
                sudo docker cp index.html c1:/usr/local/apache2/htdocs/index.html
              '''
            }
          }
        }
      }
    }
  }
}
```

```

        sudo docker exec c1 chmod 777 /usr/local/apache2/htdocs/index.html
    ""
}
}
}
stage('c2-branch 2025Q2') {
    steps {
        dir('workspace-c2') {
            git url: 'https://github.com/Poojak134/docker-httpd.git', branch: '2025Q2'
            sh ""

            sudo docker run -dp 90:80 --name c2 -v c2-webdata:/usr/local/apache2/htdocs
            httpd:latest

            sudo docker cp index.html c2:/usr/local/apache2/htdocs/index.html

            sudo docker exec c2 chmod 777 /usr/local/apache2/htdocs/index.html
        }
    }
}
stage('c3-branch 2025Q3') {
    steps {
        dir('workspace-c3') {
            git url: 'https://github.com/Poojak134/docker-httpd.git', branch: '2025Q3'
            sh ""

            sudo docker run -dp 8001:80 --name c3 -v c3-webdata:/usr/local/apache2/htdocs
            httpd:latest

            sudo docker cp index.html c3:/usr/local/apache2/htdocs/index.html

            sudo docker exec c3 chmod 777 /usr/local/apache2/htdocs/index.html
        }
    }
}

```

```
}  
}  
}  
}  
}  
}
```