

## Deploy Java Application On Tomcat Server Through Jenkins Pipeline In Local Machine(Ubuntu)

Following are the prerequisites and steps to deploy a Java application on Tomcat server using Jenkins Pipeline.

### Prerequisite:

- openjdk version 11.0.20.1
- Apache Maven 3.9.3
- Apache-tomcat-10.0.8
- Jenkins

### Steps:

- 1) Create a pipeline job in Jenkins and give it a name.  
(Java-app-tomcat-deployment)
- 2) Write a pipeline script under the Pipeline section.

Pipeline script

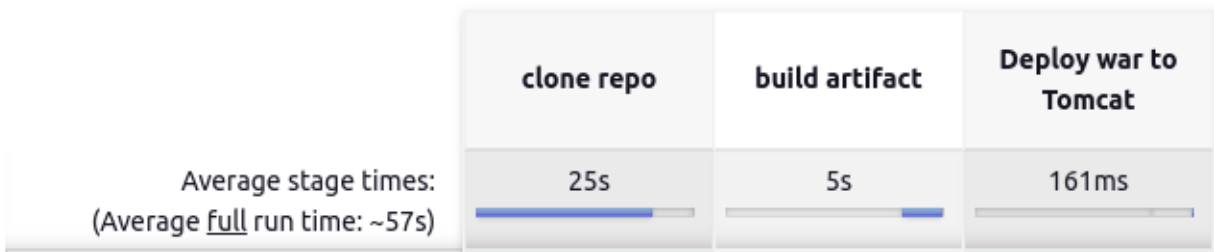
Script ?

```
1 pipeline {
2
3   agent{
4     label{
5
6       label "built-in"
7     }
8   }
9
10  stages {
11    stage("clone repo"){
12      steps{
13        sh "sudo rm -r *"
14        sh "sudo git clone https://github.com/SumitMatte/Train-Ticket-Reservation-System.git"
15        sh "sudo chown -R jenkins:jenkins *"
16        sh "sudo chmod -R 777 *"
17        sh "pwd"
18      }
19    }
20
21    stage("build artifact"){
22      steps{
23        dir("/var/lib/jenkins/workspace/tomcat-deployment/Train-Ticket-Reservation-System/"){
24          sh "pwd"
25          sh "ls -l"
```

Following is the link for the pipeline script which I have used in my Pipeline job.  
<https://drive.google.com/file/d/1W0qv-eoCYv-D6HLDxTY3JzNcF08SLDEv/view?usp=sharing>

- 3) In the pipeline script I have written three stages. In the first stage I have a clone project repository from github. In the second stage I have built an artifact using maven. In the third stage I have deployed artifact(.war) to the Tomcat Server.
- 4) Save the Pipeline Job.
- 5) Now, click on the “Build Now” button and wait until the job will be successful.
- 6) After successful completion of Job, you will see the following output.

## Stage View



- 7) In the build history click on the build number and after clicking on console output you will get all the logs related to the job which you have built.
- 8) After successfully building the job you can see the output on the browser.

