

## 5. To implement windows batch command on a project in Jenkins (with and without parameters)

New item -> Freestyle project -> Add Build Steps -> Windows Batch Command -> In the box enter (echo "Hello World") -> Build Now -> Console Output

The screenshot shows the Jenkins Configuration page for a project named 'act4.1'. The breadcrumb trail at the top is 'Dashboard > act4.1 > Configuration'. The page is divided into two main sections: 'Configure' on the left and 'General' on the right. Under 'Configure', there is a list of options: 'General' (selected), 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build Steps', and 'Post-build Actions'. Each option has a corresponding icon. Under the 'General' section, there is a 'Description' text area containing the text 'omompompmopompomp'. Below the description, it says 'Plain text' and has a 'Preview' link. At the bottom of the 'General' section, there is a checkbox labeled 'Discard old builds' with a help icon.

Check This project is parameterized

The screenshot shows the 'This project is parameterized' section in Jenkins. At the top, there is a checked checkbox labeled 'This project is parameterized' with a help icon. Below this, there is a dashed box containing a 'Boolean Parameter' configuration. The 'Boolean Parameter' section has a title 'Boolean Parameter' with a help icon. It includes a 'Name' field with a help icon, containing the text 'Are you student>'. Below the name field, there is a checkbox labeled 'Set by Default' with a help icon. At the bottom of the dashed box, there is a 'Description' field with a help icon. Below the dashed box, there are two buttons: 'Save' (blue) and 'Apply' (grey).

## 8. To create a pipeline script and build a pipeline of jobs in Jenkins

New Item -> Pipeline -> Create -> Enter some definition -> Scroll down to Pipeline -> Select pipeline script -> Build Now

## 9. To create a Jenkinsfile in a repository on GitHub and build a pipeline of jobs in Jenkins

First fork a repo -> <https://github.com/chrismlid/jenkins-pipeline-tutorial> ->

New Item -> Pipeline -> Create -> Enter some description -> Scroll down to pipeline -> Select pipeline from SCM -> In SCM select Git -> Enter Git URL -> Apply and Save -> Build Now

Once build is successful -> Go to github -> Setting -> Left panel search for webhooks -> Add webhook

Go to ngrok website -> Download Windows 64bit zip -> Extract .exe file ->

Go to cmd -> ngrok -> execute 'ngrok.exe http 8080' copy the forwarding link add it to the payload\_url -> Add Webhook '/github-webhook' -> Content-type (Application/JSON)

Go to Jenkins -> Click on build now

Go to GitHub -> Add a file .txt -> commit it -> Go to Jenkins -> Refresh -> Wait for 10-15 sec and it will automatically Build

## Exp 11: To Implement Docker Commands

Docker commands:

Docker --version

Docker login

Next open docker -> Go to images -> Search 'Images' -> eg: Image named registry

Cmd -> docker pull <imagename> -> docker images -q ->

Docker pull ubuntu

Docker pull ubuntu:rolling

## **Exp 7: To build a Maven and Ant project in Jenkins**

### **Maven**

New Item -> Maven Project -> Create -> (Source code management) Select Git ->

\*Note\* Maven Ant & Gradle works explicitly -> Fork a Repo on github -> Enter the cloning url ->

Scroll down to Build -> clean compile test -> Apply -> Save -> Build Now

### **Ant**

New Item -> Maven Project -> Create -> (Source code management) Select Git -> Build steps

-> select Invoke Ant -> Ant Version -> Ant -> (Targets = clean compile package) -> Build now ->

Open console output -> Scroll dwn and copy the path ->

-> Open cmd -> execute: dir

C:\ProgramData\Jenkins\.jenkins\workspace\act7.2\target\roshambo.war

### **Maven**