

ADVANCED JENKINS

By Dr. Vishwanath Rao

Day 1

1. Why Jenkins?
2. Current issues which Jenkins can resolve
3. Getting Started with Jenkins

a. Learning Objective

i. In this module, you will learn about Jenkins, setting up Jenkins environment on your local setup, terminologies used in Jenkins and creating Jenkins job.

b. Topics:

.Install Jenkins on a local machine

i.Starting Up with Jenkins

ii.Jenkins Architecture and terms of Jenkins

iii.Overview of Jenkins UI iv.Creating a Jenkins Job

v.Configuring a Jenkins job c. Hands-On:

.Installing the Jenkins setup, create a simple Jenkins job, delete, fail, disable the job

. a. b. c. d. e. f.

4.

5.

Introduction to GIT

What is GIT?

Common commands in GIT Integrate GIT with Jenkins Git vs GitHub

Create account on GitHub Create Repo on Github Clone GitHub repo

Plugins and its uses Learning Objective

.

.In this module, we will learn about various plugins available in Jenkins and their usage. Then we will

learn to configure Jenkins to work with other tools, working with maven builds and polling for source code changes

a. Topics:

.Introduction to Plugins i.Adding Plugins to Jenkins

ii.Commonly used plugins

- iii. Configuring Jenkins to work with java, Git and Maven
- iv. Creating a Jenkins Build and Jenkins workspace
- v. Configure Jenkins to check for source code changes periodically.
- vi. Working with Maven Build Jobs

b. Hands-On:

.Adding the plugins to the Jenkins job, creating a Jenkins job to check for source code changes, creating maven build jobs

Day 2

6. Setting Up Your Build Jobs and Security

a. Learning Objective

i. In this module, we will learn to learn to Build Jobs, add the advanced features to Jobs and Security.

b. Topics:

.Creating a Freestyle Build Job

i. Introduction to Build Triggers and to Build Steps

ii. Pre-and Post-Build Actions: Adding properties and properties files

iii. Running Your New Build Job

iv. Parameterized Builds

v. Distributed builds vi. Setting Email notification

vii. Enabling Security in Jenkins viii. Different Levels of Authentication

ix. Types of Access and Administration of the Access

c. Hands-On:

.Creating Freestyle jobs, demo on security and authorization in Jenkins, sending email notifications, running the Jenkins job

7. Implementing Automated Testing

. Learning Objective

.In this module, you will learn to setup Jenkins to Implement Automated Testing and publish reports.

a. Topics: .Introduction

i. Automating Your Unit and Integration Tests ii. Configuring Test Reports in Jenkins

iii. Displaying Test Results iv. Ignoring Tests

v. Automated Acceptance Test with JUnit b. Hands-On:

.Unit and integration testing, ignoring the tests, display test results .

Assignments will be assignment to the participants as a homework

Day 3

9. Metrics to Improve Quality

- Learning Objective

i. In this module, we shall learn to check for the code coverage and usage of PMD and FindBugs Jenkins plugin and also Jenkins support

with other build systems.

- Topics:

.Looking for foul Code through Code Coverage

i.Activating and usage of PMD Jenkins plugin

8. Assignment

ii.Activating and usage of Findbugs Jenkins plugin iii.Verifying HTML Validity

iv.Reporting with JavaNCSS

v.Jenkins with Gradle script build system vi.Jenkins with shell script build system

c. Hands-On:

.Jenkins plugin for analyzing the bugs, reporting using Jenkins, Jenkins with a different build system

10. Distributed Jenkins Configuration . Learning Objective

.In this module, we shall learn to create a distributed Jenkins system to handle concurrent build triggers, we will learn to set up master and slave configuration in Jenkins.

a. Topics:

.Introduction to Distributed Jenkins Build i.Configuring Master Jenkins node

ii.Configuring the Jenkins slave

iii.Managing nodes and distributing jobs over nodes iv.Binding Jobs on the master and slave setup

v.Labelling the nodes to run a specific job b. Hands-On:

.Configuring slave node in your Jenkins, adding labels to nodes, managing the nodes

11. Maintain and Use Jenkins

- Learning Objective

i.In this module, we shall learn the Best Practices in Jenkins culture. You will also learn the Do's and Don'ts in Jenkins setup

- Topics

.How to maintain Jenkins

i.Do's and Don'ts of Jenkins

ii.Backup of Jenkins and Migrating Jenkins from one server to another.

c. Hands-On .Backup of Jenkins

12. Assignment

. Assignments will be assignment to the participants as a homework

Day 4

13. Performing Automated Deployment and Continuous Delivery

a. Learning Objective

i. In this module, we shall learn to deploy an application to a web server using Tomcat and understanding pipeline and parallel builds.

b. Topics .Deployment Overview

i. Implementing Automated and Continuous Deployment ii. Deploying an application to an application server

iii. Install and configure tomcat.

iv. Deployment of Simple Java web application using Tomcat.

v. Jenkins Build Pipeline vi. Parallel Jenkins build

vii. Achieve generated Artifacts viii. Jenkins integrations

ix. Scaling Jenkins

c. Hands-On

.Continuous deployment using Jenkins, parallel and pipeline builds, Jenkins integrations 14. Jenkins Pipeline

. Learning Objective

. In this module, we shall learn to implement a project's entire build/test/deploy pipeline in a Jenkinsfile and create a fully automated Jenkins pipeline

a. Topics

.Overview of Pipeline as code

i. Overview of Pipeline Plugin ii. Automated Jenkins Pipeline

b. Hands-On: .Jenkins pipeline

15. Pipeline as a code

16. Move the infrastructure from one server to another

17. Setting up Environment using Environment variables

18. Case Study: Automate repetitive deployment

Day 5

19. Jenkins Tips and Tricks

- Global environment variables in Jenkins
- Create custom global environment variables
- Modify the Jenkins URL

- Meet the Jenkins' cron: Learn how to execute Jobs automatically
- Learn how to trigger Jobs from external sources: Create a generic user
- Troubleshooting: Github Actions throwing 403 forbidden errors?
- Trigger Jobs from Bash Scripts (No parameters)
- Trigger Jobs from Bash Scripts (With Parameters)

20. Introduction to Docker

21. Project/ Case Study: Implement CI/CD with Jenkins Pipeline, Docker and Maven

. Learn how to install Docker inside of a Docker Container

- Define the steps for Pipeline
- Build: Create a Jar for Maven App using Docker

c. Build: Write a bash script to automate the Jar creation

- Build: Create a Dockerfile and build an image with Jar
- Build: Create a Docker Compose file to automate the Image build process
- Build: Write a bash script to automate the Docker Image creation process
- Build: Add scripts to the Jenkinsfile
- Test: Test code using Maven and Docker
- Test: Create a bash script to automate the test process
- Test: Add test script to Jenkinsfile
- Create a remote machine to deploy containerized app
- Push: Create Docker Hub account
- Push: Create a Repository in Docker Hub

- Push: Push/Pull Docker images to Repository
- Push: Write a bash script to automate the push process
- Push: Add push script to Jenkinsfile
- Deploy: Transfer some variables to the remote machine
- Deploy: Deploy application on the remote machine manually
- Deploy: Transfer the deployment script to the remote machine
- Deploy: Execute the deploy script in the remote machine
- Deploy: Add deploy script to Jenkinsfile
- Create a Git Repository to store scripts and the code for the app
- Create the Jenkins Pipeline
- Modify the path when mounting Docker volumes
- Create the Registry Password in Jenkins
- Add the private ssh key to the Jenkins container
- Add post actions to Jenkinsfile

bb. Execute Pipeline manually

cc. Create a Git Hook to automatically trigger Pipeline

dd. Start the CI/CD process by committing new code to Git!