DevOps Certification Training Syllabus

Introduction to DevOps (Duration: 1Hr)

Learning Objective:

This session will help you understand the purpose and the scope of DevOps in the current market, tools and the skills the market is looking for and how the culture is applied in the industries

Topics:

- DevOps Principles in detail
- DevOps Engineer Skills in the market
- Knowing DevOps Delivery Pipeline
- The market trend of DevOps
- DevOps Technical Challenges
- Tools we use in DevOps

Hands-On:

A brainstorming session on the trends in current IT industries

DevOps on Cloud (Duration: 2Hrs)

Learning Objective:

This session will help you understand how DevOps is currently being used on cloud infrastructure by automating entire setup. Learn cloud setup helps faster releases, fewer software failures and how trendy it is.

Topics:

- Essentials of Cloud computing?
- Cloud and virtualization architecture
- Cloud deployment architecture
- Cloud providers – An overview
- Why we need DevOps on Cloud?
- Introducing to Amazon web services

Hands-on:

- How to setup AWS account
- Various AWS services for Devops – An overview
- DevOps using AWS – Demo

GIT – A version control tool (Duration: 5Hrs)

Learning Objective:

This session will help you understand why version control system streamlines the development of working with different people and makes very easy to collaborate on projects.

Topics:

- ✓ Knowing about Version control
- ✓ Git – A CLI
- ✓ Essentials of GIT in industry
- ✓ How to setup GIT
- ✓ Working with various commands in GIT
  - ✓ Recording Changes to the Repository
    - ✓ How to check the Status of Your Files
    - ✓ How to track New Files
    - ✓ Staging our modified files
    - ✓ Ignoring Files from GIT
    - ✓ Viewing Your Unstaged and Staged Changes
    - ✓ How to commit Your Changes
    - ✓ Skipping the Staging Area and commit
    - ✓ Removing Files from GIT
  - ✓ Viewing the Commit History
    - ✓ Limiting Log Output
    - ✓ Using a GUI to Visualize History
  - ✓ Undoing Things
    - ✓ Changing Your Last Commit
    - ✓ Unstaging a Staged File
    - ✓ Unmodifying a Modified File
  - ✓ Working with Remotes
    - ✓ Showing Your Remotes
    - ✓ Adding Remote Repositories
    - ✓ Fetching and Pulling from Your Remotes
    - ✓ Pushing to Your Remotes
    - ✓ Inspecting a Remote
    - ✓ Removing and Renaming Remotes
  - ✓ Branching and Merging in Git
    - ✓ What a Branch Is
    - ✓ Basic in Branching and Merging

- - - Branch Management in GIT
    - Branching Workflows and its usage
    - Remote Branches – create and delete
    - Rebasing
  - Git workflows
  - Git cheat sheet

Hands-on exercises:

- Installing Git
- First-Time Git Setup
- Getting a Git Repository
- Working with various git commands
- Working with Local repository vs remote repository
- Managing remote repository
- Stashing operations and various local repository operations
- Branching and merging operations
- Resolving conflicts during merges
- Managing access on repository managements

Jenkins – Continuous integration (Duration: 8 Hrs)

Learning Objective:

This session will help you understand why Jenkins is evolving as a must tool in the current DevOps practices. Understanding how Jenkins allowing current industries to set up their build pipelines with a more quickly and sophisticated build process by drastically reducing the risks in the software development lifecycles.

Topics:

- Essentials of Continuous Integration
- An example scenario where CI is used
- Know about Jenkins and its architecture in detail
- Jenkins tool Management in detail
- Know about User management in Jenkins
  - Authentication
    - Jenkins own database user creation
    - Options to enable integration with LDAP
  - Authorization

    - Matrix-based authorization
    - Project-based authorization
  - Overview of Maven
    - Maven project structure
    - Maven plugins
    - Project Object Model (POM) – the fundamental unit of work in Maven project
    - Maven build lifecycle
    - Adding external dependencies to maven pom.xml
    - Maven build and test project
  - Creating jobs and automatic build settings
    - What is Jenkins Pipeline?
    - Why Pipeline?
    - Integration with GIT
    - How to enable project-based authorization for a job
    - Source code management while creating jobs
    - Triggering automated build
    - Maven job setup
    - Know about post-build options for jobs like notifications, trigger another build, publishing reports, etc.
  - Adding a slave node to Jenkins
  - Building Delivery Pipeline
  - Notification settings in Jenkins
  - Plugin management in Jenkins

Hands-on Lab:

- Installing Jenkins
- Post-installation setup wizard
  - Unlocking Jenkins
  - Customizing Jenkins with plugins
  - Creating the first administrator user
- Administration of Jenkins
- User management – Authentication and Authorization
- Master-slave set up on Jenkins
- Creating basic jobs to pull code from GitHub

- Jobs to perform individual operations
- Setup build pipeline
- Understanding build triggers, build and post-build operations
- Deploying an application to a container using Jenkins

Docker – A containerization technology (Duration: 8Hrs)

Learning Objective:

This session will help you understand why Docker knowledge is required in order to master DevOps cultures in the current IT industry Understanding containerizing the application is also isolating that into a completely separated environment.

Topics:

- Introduction
  - Real-world Shipping Transportation Challenges
  - Introducing Docker and its technology
  - Understanding of Docker images and containers
- Working with container
  - How to Share and copy a container
  - Container Life Cycle
  - How to use Base Image and customize
  - Creation of Docker File
  - How to Publish Image on Docker Hub
- Introduction to Docker Networking
  - Network Types in docker technology
  - Docker Container Networking
  - Docker Compose – An introduction
- Docker Swarm – An introduction
  - Use Docker Compose to create PHP, WordPress, MySQL
  - How to Start Containers on a Cluster with Docker Swarm
  - Creating and Scaling an application in Docker swarm

Hands-on:

- How to setup docker-engine
- How to run docker container from pulling image from public repo
- How do we create a docker file
- Creating different docker files for different application

- ✓ Creating a docker-compose file to deploy multi-container
- ✓ Creating docker custom bridge networks
- ✓ Creating docker swarm cluster
- ✓ Orchestration of container using swarm cluster

Kubernetes (Duration: 4 Hrs)

Learning Objective:

This session will help you understand how Kubernetes helps to orchestrate the Docker containers. How the opensource system helps to automate the deployment, scaling and managing the containers.

Topics:

- ✓ Introduction to Kubernetes
- ✓ Kubernetes Cluster Architecture – An overview
- ✓ Understanding concepts of Pods, Replica sets, deployments and namespaces
- ✓ Understanding the concepts of services and networking
- ✓ Persistent volumes and persistent volume claims – an overview
- ✓ Design of Pods
- ✓ Understanding labels, selectors, jobs, and schedulers

Hands-on:

- ✓ Setting up the Kubernetes Cluster
- ✓ Deploying an app through Kubernetes Dashboard
- ✓ Accessing the application through service
- ✓ Rolling updates in Kubernetes
- ✓ Creating and adding volumes

Ansible – A configuration Management (Duration: 9Hrs)

Learning Objective:

This session will help you understand how significantly Ansible reduces your coding hours with the configuration management tool. Understanding most popular configuration management tool to set up easily, configure easily and deploy IT infrastructure easily.

Topics:

- ✓ Introducing Ansible – A configuration management tool
  - ✓ Basics / What Will Be Installed
  - ✓ Understanding Ansible architecture
  - ✓ Control Machine Requirements
  - ✓ Managed Node Requirements

- Inventory
  - Hosts and Groups
  - Host Variables
  - Group Variables
- Learn various Ansible Modules
- How to use Adhoc commands
  - Parallelism and Shell Commands
  - File Transfer
  - Managing Packages
  - Users and Groups
  - Deploying From Source Control
  - Managing Services
- Introduction to YAML script
- Playbook
  - About Playbooks
  - Playbook Language Example – YAML
  - How to Write Playbooks
  - Tasks in Playbooks
  - Understanding about various tasks in the playbook
  - Introduction to Handlers and variables
  - Learn about using handlers, variables in the playbook
  - Become (Privilege Escalation)
- Roles
  - Role of Directory Structure
  - Using Roles
  - Role Duplication and Execution
  - Role Default Variables
  - Role Dependencies
  - Role Search Path
  - Ansible Galaxy
- Including and Importing
  - Includes vs. Imports
  - Importing Playbooks

- ✓ Including and Importing Task Files
- ✓ Including and Importing Roles

Hands-on:

- ✓ How to setup Ansible server and target servers
- ✓ Writing Adhoc commands to install and configure the servers
- ✓ Writing a playbook to install and configure webservers and deplo0y an application
- ✓ How to create Ansible Role and use it
- ✓ Using an ansible role in a playbook
- ✓ How to use Ansible Galaxy to download roles.
- ✓ Example – Install and use Jenkins roles from ansible-galaxy