Devops Course Content

Course Description

DevOps tool blends software development (Dev) and information-technology operations (Ops) to reduce system development life cycle, deliver high-quality software and best IT solutions to meet business goals. DevOps tools include Jenkins, git, chef, ansible, docker, Nagios, puppet, and aws devops. DevOps methodology includes principles of continuous development and deployment, code automation, configuration management, inter-team collaboration, testing and monitoring, and agile IT service.

The future scope of DevOps is huge because of its unique key features. Career opportunities with DevOps course certification are enormous in the IT field. According to Glassdoor - The average salary of a DevOps engineer will be $143,000 per year.

Hachion's DevOps online training helps you understand the fundamental concepts and dives you deep into the core implementations of DevOps. Our DevOps course designed with a well-structured curriculum up-to-date and practical sessions. Basic knowledge of Linux fundamentals and any scripting language like shell and text-editor will be an added advantage to learn the DevOps course easily.

Course Content

Introduction to DevOps

- What is DevOps
- Why DevOps?
- History of DevOps
- SDLC models: Waterfall, Lean, Agile
- DevOps Goals
- DevOps perspective
- Relationship between DevOps and Agile
- Configuration Management
- Continuous Integration and Deployment

Cloud Computing and Virtualization

- History and evolution of cloud
- Cloud computing concepts
- Grid Computing vs. Cloud Computing
- Characteristics and Benefits of Cloud
- Cloud service models (NIST)
- IaaS, PaaS and SaaS
- Cloud service model implementations
- Virtualization
- Virtual bootable OS Images
- Cloud Storage-
- Introduction to AWS and AZURE

Architectures

- What is Application Architecture
- How Internet works
- A Sample Project Explanation
- Monolithic Architecture
- 3 Tier Architecture
- SOA Architecture
- Microservices based Container Architecture

AWS Basics & Lab Server Setup

- How to setup Amazon account
- How to create a user in AWS
- AWS Market Place
- Amazon EC2, S3.
- Managing Login
- Amazon S3 storage
- Elastic IP's

Linux Administration

- Introduction to Linux Administration
- How to create users and groups
- How to manage the system services using the systemctl
- What are system logs and how to use them
- How to configure the hostname, time and date, locale and keyboard settings
- How to check the network interfaces and update IP address details
- DNS Configuration and other system tasks
- YUM and RPM Package Installers for RHEL

GIT: Version Control System

1. **Introduction**
   - Version control systems
   - Local, Centralized and distributed

2. **Installing Git**
   - Installing on Linux
   - Installing on Windows
   - Initial setup

3. **Git Essentials**
   - Creating repository
   - Cloning, check-in and committing
   - Fetch pull and remote
   - Branching

Maven – Build Tool

- Introduction to Maven
- Maven Lifecycle

- Maven Phases and Goals
- Maven Directory structure
- Maven Repositories
- Maven Installation
- Creation of Maven structure
- Running Maven commands { site, clean, test, package, install, deploy, compile}

Ansible - Configuration Management

1. **Introduction to Ansible**
   - Ansible Sessions Environment Setup
   - Validating Ansible Installation
   - Ansible Requirements – Python, PIP, and Virtual Environments
   - Ansible on Linux

2. **Ansible Configuration**
   - Understanding & Customizing Ansible's Configuration file
   - Ansible inventory
   - Verify Ansible inventory
   - Ansible commands
   - Ansible ad-hoc commands

3. **Ansible Playbooks**
   - Introduction to YAML
   - Ansible playbook sections
   - Ansible Variables
   - Ansible loops
   - Ansible Facts
   - Ansible conditionals
   - Debug Ansible playbooks
   - Registered variables and conditional execution
   - Looping
   - Asynchronous and Parallel
   - Ansible playbook for installing Apache webserver

4. **Introduction to Ansible roles**
   - Why do you need roles?
   - Code Organization Strategies
   - Anatomy of a Role
   - Understand the Ansible roles directory structure
   - Creating a role for Apache
   - Writing and applying playbook for app servers
   - Copying config file, notifications, and handlers

5. **Ansible Galaxy**
   - What is Ansible Galaxy
   - How to Download Ansible Roles from Galaxy

Docker

1. **Introduction**
   - What is a Docker
   - Use case of Docker
   - Platforms for Docker
   - Dockers vs Virtualization

2. **Architecture**
   - Docker Architecture.
   - Important Docker components
   - Understanding the Docker components

3. **Installation**
   - Installing Docker on Linux.
   - Understanding Installation of Docker on Windows.
   - Basic Docker commands

4. **Provisioning**
   - Docker Hub
   - Downloading Docker images.
   - Running Docker images & Docker containers
   - Running docker applications in a container

5. **Dockerfile**
   - How to create a docker image
   - Instructions of Dockerfile
   - Creation of multiple docker images from scratch using Dockerfile

6. **Docker Swarm**
   - What is Docker swarm
   - What is Docker High Availability
   - Deploy an application in high availability mode across multiple services

Kubernetes

1. **Introduction to Kubernetes**
   - What is Kubernetes & Why it's so popular
   - Difference between Docker & Kubernetes
   - Advantages of Kubernetes
   - Components of Kubernetes
   - Explanation of END to END Kubernetes LifeCycle
   - Kubernetes Deployment Types

2. **Components of Kubernetes Master**
   - Kubernetes Master
   - kube-apiserver
   - key-value store
   - kube-scheduler

- kube-controller-manager

3. **Components of Kubernetes Node**
   - Docker
   - kubelet
   - kube-proxy
   - kubectl

4. **Installation & Cluster Creation**
   - Create 3 RHEL VM's
   - Create a three-node Kubernetes cluster using Master and 2 Nodes
   - Install Kubectl, a command line tool to manage clusters
   - Start a three-node cluster
   - Get cluster details
   - List all nodes associated with the cluster

5. **Working with Kubernetes Configuration Objects**
   - Pod
   - Pod Requests and Limits
   - Pod Readiness and Liveness
   - ReplicaSet
   - Creation of Deployment (Deploying Highly available Application with Best Practices)
   - Job
   - CronJob
   - DeamonSet
   - ClusterIP Service
   - NodePort Service
   - Ingress Load balancer
   - Deploy a containerized app image in the Kubernetes cluster

JENKINS – Continuous Integration

1. **Introduction**
   - Understanding continuous integration
   - Introduction about Jenkins
   - Build Cycle
   - Jenkins Architecture

2. **Installation**
   - Obtaining and installing Jenkins Server
   - Installing and configuring GIT
   - Java installation and configuration
   - Maven Installation & configuration
   - Exploring Jenkins Dashboard

3. **Jobs**
   - Creating Simple Jobs
   - Running the Jobs

- Understanding the JOBS and WORKSPACE Directory
- Understanding what is BUILD

4. **Build Deployments**
   - Understanding Deployment
   - Tomcat installation and configuration
   - Deployment Plugins

5. **Jenkins LifeCycle**
   Creation of Complete application lifecycle from the Git -> Maven -> Junit -> Tomcat -> Documentation creation in a single Job

6. **Securing Jenkins**
   - Authentication & Authorization
   - Matrix based Authentication
   - Creating users
   - Plugin Management

7. **Jenkins Build Pipeline**
   - What is Jenkins build Pipeline
   - Creation of Jenkins Pipeline using the Pipeline Plugin
   - Creation of Jenkins Pipeline using the upstream and downstream projects
   - Coding the Jenkins Pipeline with GitHub, Maven, Ansible, Tomcat, Docker integrated

Nagios: Monitoring

1. **Introduction and Installation**
   - Obtaining Nagios
   - Compiling and installing Nagios

2. **Basic configuration**
   - Configuring a Linux server to monitor the complete statistics
   - How can Nagios be extended in the Infrastructure