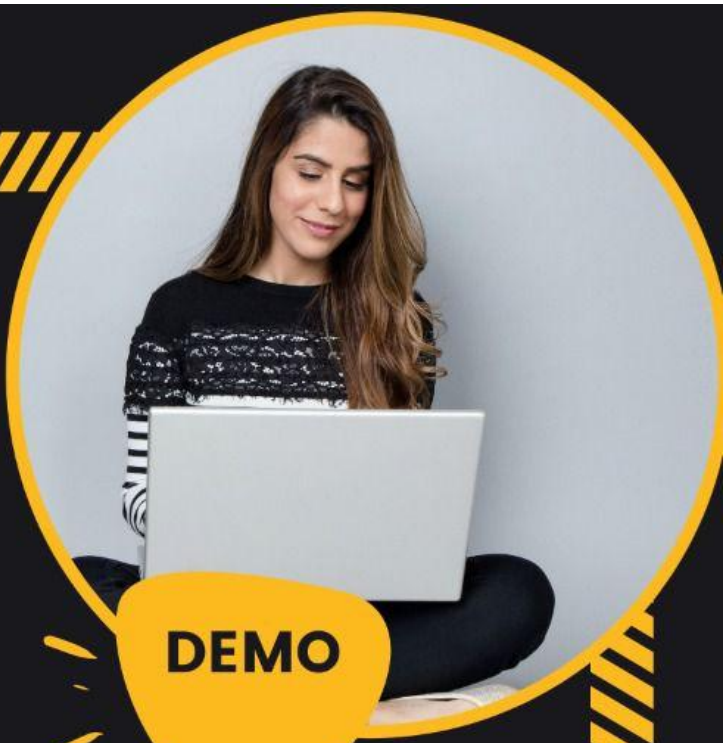

DEMO

+91 9786436525

ONLINE DEVOPS, AWS CLASSES

2 MONTH COURSE

BOOK NOW

Trainer: Dr. Vivek Yoganand B.E,. M.Tech., MBA,. LAW., PhD
Current Company: Redhat

Mobile: 9786436525

NO MORE WAIT, START YOUR CAREER!!

About the Course

In this module, you will learn about DevOps, its evolution, the interrelation between agile and DevOps, technical and security challenges in DevOps, the difference between requirements and architecture, and ways to write user acceptance tests. DevOps tools such as Git, Docker, Jenkins, Tomcat and Nagios in practical, hands on and interactive approach. The Devops training course focuses heavily on the use of Docker containers, a technology that is revolutionizing the way apps are deployed in the cloud today and is a critical skillset to master in the cloud age

DURATION : 8 weeks to 9 weeks.

WHO SHOULD ATTEND

This DevOps training course will be of benefit the following professional roles:

- Software Developers
- Technical Project Managers
- Architects
- Operations Support
- Deployment engineers
- IT managers
- Development managers

PREREQUISITES

Infra build support, preferably in Terraform and the UNIX/Linux command line tools are essential for this course.

Introduction to Devops:

- Define Devops
- What is Devops
- SDLC models,Lean,ITIL,Agile
- Why Devops?
- History of Devops
- Devops Stakeholders
- Devops Goals
- Important terminology
- Devops perspective
- Devops and Agile
- Devops Tools
- Configuration management
- Continuous Integration and Deployment

Linux Basics:

- Unix and linux difference
- Linux File system structure
- Basic linux/unix commands
- Changing file permissions and ownership
- Types of links soft and hard link
- Filter commands
- Simple filter and advance filter commands
- Start and stop services
- Find and kill the process with id and name
- Package installation using RPM and YUM

GIT: Version Control:

- Introduction
  1. Version control systems
  2. Local, Centralized and distributed
- Installing Git
  1. Installing on Linux
  2. Installing on Windows
  3. Initial setup
- Git Essentials
  1. Creating repository
  2. Cloning, check-in and committing
  3. Fetch pull and remote
  4. Branching

Ansible:

- Introduction
- Installation
- Installation demo
- First Playbook
- First Playbook demo
- Provisioning with EC2 demo
- Roles
- Best Practices
- Directory Layout
- Ansible Tower

Jenkins – Continuous Integration

- Introduction.
  1. Understanding continuous integration
  2. Introduction about Jenkins
  3. Build Cycle
  4. Jenkins Architecture
- Installation
  1. Obtaining and installing Jenkins
  2. Installing and configuring GIT
  3. Java installation and configuration
  4. Maven Installation
  5. Exploring Jenkins Dashboard.
- Jobs
  1. Creating Jobs
  2. Running the Jobs
  3. Adding and updating Plugins
  4. Disabling and deleting jobs
- Build Deployments
  1. Understanding Deployment.
  2. Tomcat installation and configuration
  3. Deployment Plugins
  4. Deploying a war file from Jenkins to Tomcat
  5. Maven Build Scripts
- Securing Jenkins
  1. Authentication
  2. Jenkins Plugin
  3. Authorization
  4. Confidentiality
  5. Creating users
  6. Best Practices for Jenkins

  Pipeline Automation

Docker– Containers:

- Introduction
  1. What is a Docker
  2. Use case of Docker
  3. Platforms for Docker
  4. Dockers vs Virtualization
- Architecture
  1. Docker Architecture.
  2. Important Docker components
  3. Understanding the Docker components
- Installation
  1. Installing Docker on Linux.
  2. Understanding Installation of Docker on Windows.
  3. Some Docker commands.
- Provisioning
  1. Docker Hub.
  2. Downloading Docker images.
  3. Running Docker images
  4. Running commands in container.
  5. Running multiple containers.
- Custom images
  1. Creating a custom image.
  2. Running a container from the custom image.
  3. Publishing the custom image.
- Docker Networking
  1. Accessing containers
  2. Linking containers
  3. Exposing container ports
  4. Container Routing

Tool Study – Monitoring – Nagios

- Introduction to Nagios
- Nagios Plugins.
- Nagios objects.
- Nagios commands.
- Nagios notification

Tool Study – Micorservice Managment tool

- Istio
- How it works
- Core concepts
- Virtual services

- Sidecars
- Traffic management
- Security
- Observability

Tool Study – Messaging/Streaming tool

- Kafka Basics and Core concepts
- Introduction
- Distributed
- Streaming Platform
- Commit Log
- Message
- Topic
- Partitions
- Producer
- Consumer
- Broker
- Cluster

Tool Study – In-memory/Distributed databases

- Redis, Postgres

Terraform

Getting Started & Setting Up Labs

- Choosing a right Infrastructure as Code tool
- Installing Terraform - Windows Users
- Installing Terraform - MacOS and Linux Users
- Choosing Right IDE for Terraform IAC development
- Setting up AWS account

Deploying Infrastructure with Terraform

- Creating first EC2 instance with Terraform
- Understanding Resources & Providers
- Understanding Resource & Providers - Part 2
- Destroying Infrastructure with Terraform
- Understanding Terraform State files
- Understanding Desired & Current States
- Challenges with the current state on computed values
- Terraform Provider Versioning

AWS Solution Architect Training

AWS Modules

- Introduction to Amazon Web Services

- Compute Services
- Pricing Models & Billing
- Elastic Block Store (EBS)
- EFS
- Simple Storage Service (S3)
- Virtual Private Cloud (VPC)
- Virtual Private Network (VPN) & Direct Connect
- Elastic Load Balancer
- Auto Scaling
- Route53
- Relational Database Services (RDS)
- Identity and Access Management
- Simple Notification Services
- Simple Queue Services
- AWS Cloud Watch Monitoring and Logging
- Migration Service: SNOWBALL
- Storage Gateway
- Elastic Cache, DynamoDB and Redshift

Kubernetes (10 hr Continous Video will be shared)

- Features of Kubernetes
- Architecture of Kubernetes
- Install and Configure Kubernetes ENV
- Introduction of Kubernetes Images
- Kubernetes Jobs
- Kubernetes Node
- Kubernetes Service
- Kubernetes Pod
- Kubernetes Volumes
- Kubernetes Replication Controls
- Kubernetes API
- Introduction to Kubectl
- Creating App
- App deployment
- Auto Scaling

Openshift (10 hr Continous Video will be shared)

- Create containerized services using Podman.
- Manage containers and container images.
- Create custom container images.
- Deploy containerized applications on Red Hat OpenShift.
- Deploy multi-container applications.

Student got an opportunity to work