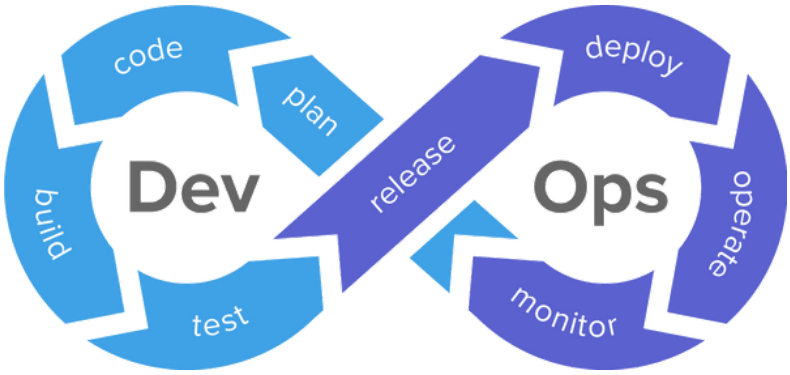
KR NETWORK CLOUD
Leading Edge IT Training

DEVOPS
TRAINING

(Duration - 150+ Hours)



KR Network Cloud

+91-9555378418 www.krnetworkcloud.org info@krnetworkcloud.org

COURSE CONTENT

MODULE - 1 GIT

UNIT - 1

INTRODUCTION TO GIT AND KEY TERMS

- SCM / Version Control
- What is Git?
- Key terms in Git

UNIT - 2

GIT REPOSITORY MANAGEMENT AND BRANCHING

- Create Repo on GitHub
- Install git client
- Git config files -
- Git init
- Upstream
- Git remote
- Git clone
- Git fork
- Branch concept
- Branching startegy
- How to create branch
- Git checkout

UNIT - 3

GIT WORKFLOW AND COLLABORATION

- Working dir
- Staging area
- Repo
- Git add
- Git commit
- Git push
- Git pull

UNIT - 4

ADVANCED GIT OPERATIONS AND REVISION CONTROL

- Git show
- Git log
- Git diff – comparing commits
- Git revert
- Git reset
- Git rebase

UNIT - 5

GIT COLLABORATION AND ADVANCED TECHNIQUES

- Git merge and merge conflicts
- Git stash
- Git tag
- Git cherry pick
- Pull request
- Students Quiz Sessions

MODULE 2 - JENKINS

UNIT - 1

INTRODUCTION TO JENKINS AND ITS POPULARITY

- What is Jenkins
- DevOps and Jenkins
- Why it is popular
- Student Quiz

UNIT - 2

JENKINS INSTALLATION, CONFIGURATION, AND JOB MANAGEMENT

- Jenkins installation – docker/ war / rpm
- Jenkins config files / dir
- Change default port
- Environment variables – inbuilt , user defined
- User management
- Installing plugins – gui , hpi
- Jenkins job
- Integration with github
- Credentials manger
- Trigger
- Workspace
- Build status
- Upstream job
- Downstream job

UNIT - 3

JENKINS INTEGRATIONS AND ADVANCED CONFIGURATIONS

- Email
- Ansible integration
- sonarqube integration
- maven integration
- Executor
- Slave configuration
- label
- view

UNIT - 4

JENKINS PIPELINE AND AUTOMATION TECHNIQUES

- Multi branch pipeline
- Pipeline – declarative / scriptive
- jenkins file
- parameters
- timeout
- agent
- stages
- steps

UNIT - 5

STUDENT QUIZ SESSIONS

MODULE 3 - DOCKER

UNIT - 1

INTRODUCTION TO CONTAINERS

- Introduction to Server Provision
- VMS Base Provisioning difficulty
- OS Virtualization Introduction
- Hardware Virtualization vs. OS virtualization
- Docker Introduction
- Docker Architecture
- Use case of Containerization Application
- Setting Up Your Environment
- Setup machine on Cloud & on-premises
- Installation Docker

UNIT - 2

LEARNING THE BASICS OF CONTAINERS

- Images and Container File System
- Docker Hub
- Image & Container basic Operation Commands
- Creating Containers
- Executing Container Commands
- Docker Logging

UNIT - 3

EXPOSE CONTAINERS FOR WORLD NETWORK

- Port Understanding
- Expose container for external Network

UNIT - 4

BUILD DOCKER IMAGE

- Image modification
- Manually Docker image Build process
- Install custom BINS/LIBS into the Docker images

UNIT - 5

DOCKER STORAGE

- Stateless and state full Application Understanding
- Provide persistent storage for container
- Create Docker Volume
- Docker Data Backup Solution

UNIT - 6

DOCKER IMAGES FILE

- Introduction to the Docker file
- Write Docker file for custom application
- Image History
- Saving and Loading Images

UNIT - 7

DOCKER NETWORK

- Understanding Bridge Network
- Creating a custom Network for Docker Container
- Establish Communication between Container

UNIT - 8

DOCKER REGISTRY

- Introduction Docker registry
- How to use Public Dockers Registry
- Setup Internal Dockers registry
- Maintain Image over internal Dockers registry

UNIT - 9

DOCKER COMPOSE

- Installing Docker Compose
- Compose Commands
- Creating a Compose File
- Using Volumes and Networking with Compose

MODULE 4 - KUBERNETES

MODULE - 1 INTRODUCTION KUBERNETES

- Introduction to Kubernetes
- Kubernetes Cluster Architecture
- Core Concept of Kubernetes Services
- ETCD & Controller & API & scheduler
- Exploring your Cluster
- Understanding YAML

MODULE - 2 INSTALLATION

- Installation minikube
- Installation using kubeadm utility in HA mode
- Install Kubernetes the hard Way
- How to use kops and create
- Kubernetes cluster

MODULE - 3 APPLICATION DEPLOYMENT

- Creating a Deployment in Kubernetes using YAML
- Creating a Service in Kubernetes
- Understanding pod & Replication & Deployment
- configuration
  Using Rolling Updates in Kubernetes
- Configure Environment variables in the application
- Configure secret resources for sensitive value
- Creating Config Map
- Scale Applications
- Multi Container PODs
- Init Containers
- Self-Healing Applications

MODULE - 4 STORAGE MANAGEMENT

- Creating Persistent volume
- Persistent Volume Claim
- Volume claim policy understanding
- Attach storage on deployment

MODULE - 5 POD SCHEDULING

- Manual Scheduling
- Labels and Selectors
- Taints and Tolerations
- Node Selectors
- Node Affinity
- DaemonSets
- Static Pods
- Multiple Schedulers
- Configuring Kubernetes Scheduler

MODULE - 6 RESOURCE ALLOCATION

- Restrict Limit Memory & CPU use
- Creating Resource Quota
- Creating Limit Quota

MODULE - 7 NETWORKING

- Logging and Monitoring
- Monitoring Cluster Component
- Managing application Logs

MODULE - 8 MONITORING KUBERNETES

- Creating a Deployment in Kubernetes using YAML
- Creating a Service in Kubernetes
- Understanding pod & Replication & Deployment
- configuration
  Using Rolling Updates in Kubernetes
- Configure Environment variables in the application
- Configure secret resources for sensitive value
- Creating Config Map
- Scale Applications
- Multi Container PODs
- Init Containers
- Self-Healing Applications

MODULE - 9 SECURITY

- Authentication
- TLS Introduction
- Certificate System Kubernetes
- Creating Certificate
- Role base Access Controls
- Cluster Role and Role Binding
- SCC: Security Constant Conta & Network Policy
- Image Security

MODULE 5 - ANSIBLE

UNIT - 1

AN INTRODUCTION TO ANSIBLE

- Automating Linux Administration with Ansible
- Quiz: Automating Linux Administration with Ansible
- Installing Ansible
- Guided Exercise: Installing Ansible
- Summary

UNIT - 2

DEPLOYING ANSIBLE

- Building an Ansible Inventory
- Guided Exercise: Building an Ansible Inventory
- Managing Ansible Configuration Files.
- Guided Exercise: Managing Ansible
- Configuration Files
- Running Ad Hoc Commands
- Guided Exercise: Running Ad Hoc Commands
- Lab: Deploying Ansible
- Summary

UNIT - 3

IMPLEMENTING PLAYBOOKS

- Writing and Running Playbooks
- Guided Exercise: Writing and Running Playbooks
- Implementing Multiple Plays
- Guided Exercise: Implementing Multiple Plays.
- Lab: Implementing Playbooks
- Summary

UNIT - 4

MANAGING VARIABLES AND FACTS

- Managing Variables
- Guided Exercise: Managing Variables
- Managing Secrets
- Guided Exercise: Managing Secrets
- Managing Facts
- Guided Exercise: Managing Facts
- Lab: Managing Variables and Facts
- Summary

UNIT - 5

IMPLEMENTING TASK CONTROL

- Writing Loops and Conditional Tasks
- Guided Exercise: Writing Loops and Conditional Tasks.
- Implementing Handlers
- Guided Exercise: Implementing Handlers.
- Handing Task Failure
- Guided Exercise: Handing Task Failure
- Lab: Implementing Task Control
- Summary

UNIT - 6

DEPLOYING FILES TO MANAGED HOST

- Managing Software and Subscriptions.
- Guided Exercise: Managing Software and Subscriptions.
- Managing Users and Authentication.
- Guided Exercise: Managing Users and Authentication.
- Managing the Boot Process and Scheduled Processes.
- Guided Exercise: Managing the Boot Process and Scheduled Processes.
- Managing Storage.
  Guided Exercise: Managing Storage
- Managing Network Configuration.
- Guided Exercise: Managing Network Configuration.
- Lab: Automating Linux Administration Tasks.
- Summary

UNIT - 7

MANAGING LARGE PROJECTS

- Selecting Hosts with Host Patterns.
- Guided Exercise: Selecting Hosts with Host Patterns.
- Managing Dynamic Inventories. Guided Exercise: Configuring Parallelism
- Managing Dynamic Inventories Configuring Parallelism.
- Guided Exercise: Configuring Parallelism.
- Including Importing Files.
- Guided Exercise: Including and Importing Files
- Lab: Managing Large Projects .
- Summary

UNIT - 8

SIMPLIFYING PLAYBOOKS WITH ROLES

- Describing Role Structure
- Quiz: Describing Role Structure
- Reusing Content with System Roles.
- Guided Exercise: Reusing Content with System Roles.
- Creating Roles
- Guided Exercise: Creating Roles.
- Deploying Roles with Ansible Galaxy
- Guided Exercise: Deploying Roles with Ansible Galaxy
- Lab: Simplifying Playbooks with Roles.
- Summary

UNIT - 9

TROUBLESHOOTING ANSIBLE

- Troubleshooting Playbooks.
- Guided Exercise: Troubleshooting Playbooks
- Troubleshooting Ansible Managed Hosts...
- Guided Exercise: Troubleshooting Ansible Managed
- Lab: Troubleshooting Ansible.
- Summary

UNIT - 10

AUTOMATING LINUX ADMINISTRATION TASKS

- Managing Software and Subscriptions.
- Guided Exercise: Managing Software and Subscriptions.
- Managing Users and Authentication.
- Guided Exercise: Managing Users and Authentication.
- Managing the Boot Process and Scheduled Processes.
- Guided Exercise: Managing the Boot Process and Scheduled Processes.
- Managing Storage.
- Guided Exercise: Managing Storage
- Managing Network Configuration.
- Guided Exercise: Managing Network Configuration.
- Lab: Automating Linux Administration Tasks.
- Summary

MODULE 6 - TERRAFORM

UNIT - 1

INTRODUCTION TO TERRAFORM

- Introduction to terraform
- Infrastructure Automation
- Install Terraform
- Providers
- Resources
- Basic Syntax

UNIT - 2

GETTING STARTED WITH TERRAFORM

- Terraform Plan, show, Apply, Destroy
- Exploring Terraform Registry
- Interpolation
- Tainting and Updating Resources
- Terraform Console and Output
- Terraform Variables
- Breaking Out Our Variables and Outputs

UNIT - 3

TERRAFORM MODULES

- Introduction to Modules
- Module repositories
- First Basic Module
- The Module code
- Main Terraform Code

UNIT - 4

TERRAFORM: WRITING IN A MORE ORGANIZED WAY

- Maps and Lookups
- Terraform Workspaces
- Breaking Out Our Variable Definitions
- Null Resources and Local-Exec
- Terraform Console

UNIT - 5

TERRAFORM WITH AWS CLOUD

- Setting up the system for AWS
- AWS Storage: The S3 Bucket and Random ID
- AWS Storage: The Root Module
- AWS Compute: AMI Data, Key Pair, and the File Function
- AWS Compute: The EC2 Instance
- AWS Compute: User Data and Template Files
- AWS Compute: The Root Module

UNIT - 6

TERRAFORM WITH AWS CLOUD ADVANCE PRACTICE

- AWS Networking: VPC, IGW, and Route Tables
- AWS Networking: Subnets, Security, and the Count Attribute
- AWS Networking: The Root Module

UNIT - 7

TERRAFORM BASIC LABS

- Remote state
- Data Sources
- Templates
- Conditionals
- Built-in Functions
- Working with state files
- Outputs, count and Join Function

UNIT - 8

TERRAFORM INTEGRATION WITH S3, GIT, JENKINS

- Adding S3 backend
- Integration with Git
- Packer introduction
- Terraform with Packer
- Terraform with Jenkins
- Terraform Formatting and Remote State
- Terraform RandomID and S3 Buckets

UNIT - 9

TERRAFORM TROUBLESHOOTING AND TESTING

- Terraform Plan revisited
- Debugging the script
- Terraform Testing
- Error Handling

UNIT - 10

TERRAFORM WITH AZURE, GOOGLE CLOUD, & PLUGINS

- Terraform with Azure cloud
- Terraform with Google Cloud
- Terraform Templates
- Terraform plugins
- Integrating Go Plugins

UNIT - 11

TERRAFORM ADVANCE FEATURES AND EXAMPLES

- Best practices in writing terraform scripts
- Terraform Workflow
- Terraform projects
- Other Hashicorp tools

MODULE 7 - MAVEN

- Maven Introduction & Installation
- Convention over configuration
- What is pom.xml
- Maven Lifecycle
- Maven Local, Central & Remote Repositries
- Maven's Setting.xml
- Group ID, Artifact ID, Snapshot vs Release
- Dependencies
- Maven Archetypes
- Dependency Management
- Distribution Management
- Maven Integration with SonarQube
- Maven Integration with Nexus
- Maven Integration to Tomcat

MODULE 8 - SONARQUBE

- Why is source code scanning required?
- Introduction & Installation of SonarQube
- Architecture
- SonarQube Home Directory Structure
- Sonar scanning Rules, Quality Profiles & Quality Gates
- SonarQube Integration with Maven
- Checking SonarQube reports
- Custom Quality Gates
- User and Group management for projects
- Integration with Jenkins
- SonarQube webhooks

MODULE 9 - NEXUS REPOSITORY ARTIFACTORY TOOL

- What is a repository?
- Repository Management tools
- What is Nexus and why do we use Nexus?
- Installation and configuration of Nexus: As a Container, Pod & Service
- Nexus Administration: Managing Nexus Configuration and security.
- Hosted vs Proxy vs Group Repositories
- Introduction to Repository replications
- Creating Hosted and Proxy Repositories for maven
- Using Nexus as docker hosted repository.
- Integrating Nexus with Maven
- Build and deploy Snapshot and Release artifacts to Nexus.
- Nexus troubleshooting

PREREQUISITE FOR THIS TRAINING

- Student must have good knowledge of Linux Operating System & Understand the basics cocept of any public cloud (AWS/Azure/GCP etc)

LEARNING OBJECTIVE :

For Git

This course equips you with Git, a version control system, teaching essential commands for managing code changes and branching strategies for efficient collaboration.

For Jenkins

For streamlined workflows, you'll explore Jenkins, a popular DevOps tool, learning installation, job creation, and pipeline automation to integrate with Git and other tools for smoother development processes

For Docker

Master containerization with Docker from the ground up. Learn to create efficient images, manage storage effectively, and configure container networks. This hands-on training will equip you to build and deploy containerized applications with confidence.

For Kubernetes

Explore the container orchestration with Kubernetes. Gain the skills to manage and deploy containerized applications at scale. This training will unlock all the aspects of Kubernetes, allowing you to efficiently manage complex deployments.

For Ansible

The Red Hat Enterprise Linux Automation with Ansible (RH294) course teaches you to automate Linux system administration. You'll learn Ansible to manage systems, write automation playbooks, configure apps, and orchestrate deployments.

For Terraform

You will learn to automate infrastructure provisioning and management using Terraform. You'll grasp core concepts, write reusable modules, and leverage Terraform with popular cloud platforms (AWS, Azure, Google Cloud). Hands-on labs will solidify your skills, and you'll learn best practices for managing infrastructure as code.

For Maven

Master Maven, a build automation tool, to manage project lifecycles and dependencies. Learn key concepts like pom.xml, repositories, and archetypes to streamline development and ensure consistent builds

For Nexus Repository

Nexus Repository, an artifact management tool, to control and optimize access to libraries and components. You'll explore installation, configuration, and repository types (hosted, proxy) for efficient build processes

For SonarQube

Enhance code quality with SonarQube, a static code analysis tool. Learn installation, configuration, and utilize quality profiles and gates to identify and address potential issues within your codebase

KR NETWORK CLOUD

LEADING EDGE IT TRAINING

Contact KR Network Cloud using any of the following

+91 9555378418

C-3/207, Second Floor, Kanishk Complex, Near Maharaja Banquet Nirman Vihar Metro Station, Delhi-110092

www.krnetworkcloud.org

info@krnetworkcloud.org

Check out the social media to get the latest update

THANK YOU