

# BATCH-6 | DEVSECOPS & CLOUD DEVOPS

BATCH-6 | LIVE

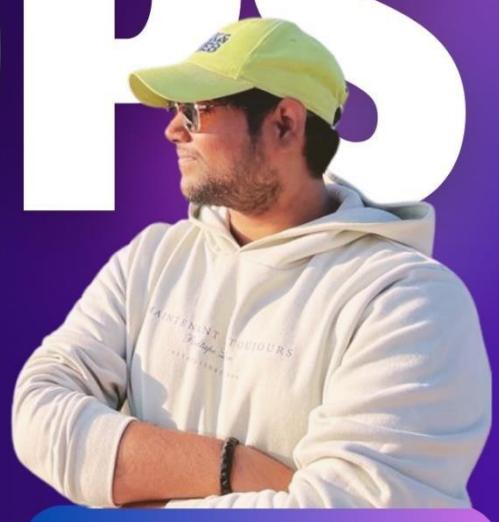
FROM SCRATCH TO EXPERT  
REAL-TIME CORPORATE



# DEVOPS



5 IN 1



ADITYA JAISWAL  
EX-CAPGEMINI, DELOITTE

Category	Details
Batch	Batch-6
Course Title	Zero To Hero
Topics	- Each module includes detailed custom recorded videos and live sessions. - At the end of each module, we will discuss projects, explain real-time architecture, and more.
Classes	<b>Weekends:</b> Live Classes (8 PM IST to 10 PM IST) <b>[We will record it in case if any one missed or want to watch later on]</b> <b>Weekdays:</b> Recorded Custom Videos
Duration	<b>Course Duration:</b> 2 Months <b>Access to Contents:</b> 2 Years
Doubts	<b>Telegram Group:</b> Dedicated Telegram Group <b>1:1 Connects:</b> Available over Discord
Detailed Docs	- All the scripts - Interview Prep Docs
Interview Prep	- 200+ Interview Questions with Answers for every topic taught - Real-Time Scenario Based Questions & Answers for every tool - Real-Time Errors & Troubleshooting on every tool - Project explanations

# Module-1 | DevSecOps Intro

## **Introduction to DevOps**

- What is DevOps?
- The Problem DevOps Solves
- DevOps Workflow
- Real-Time Corporate DevOps Workflow With Flow Diagram
- Deployment Strategies
- DevOps VS DevSecOps

## **Key DevSecOps Practices**

- Continuous Integration and Continuous Delivery (CI/CD)
- Automated Testing
- Best Practices and Challenges
- DevSecOps Best Practices
- Key Practices for Successful Implementation
- Common Challenges and Strategies for Overcoming Them

# Module-2 | Linux & Shell Scripting

## Introduction to Linux

- What is Linux?
- Overview and Key Features
- Structure of Linux
- Linux File System Hierarchy
- Key Directories and Their Purposes

## Virtual Machines (VMs)

- Structure of a VM
- Components and Architecture
- Ports in a VM
- Commonly Used Ports and Their Purposes
- Creating and Connecting to a VM
- Steps to Create a VM
- Connecting to a VM Using SSH
- Opening and Managing Ports

## Essential Linux Commands

- Most Important Linux Commands
- Basic Commands
- Advanced Commands

## Package Management

- Installing, Updating, and Removing Packages

## File/Folder Permissions

- Understanding and Setting File/Folder Permissions

## User & Group Management

- Adding, Modifying, and Deleting Users and Groups

## Networking Commands

- Common Networking Commands  **Linux Troubleshooting**

- Errors & Troubleshooting in Linux
- Common Errors and Their Solutions
- Examples of Troubleshooting Steps

## **Shell Scripting Hands-on**

- Introduction to Shell Scripting
- Basics of Shell Scripting
- 10 Corporate Shell Scripts
- Practical Scripts Used in Real-World Scenarios
- Comprehensive Guide and Documentation
- All Shell Scripts Used in the Course
- Detailed Explanations

## **Interview Preparation**

- Comprehensive Linux Guide
- Detailed Documentation Covering Key Topics
- 500 Linux Commands
- Extensive List of Commands with Explanations
- 200 Interview Questions

# Module-3 | Git

## **Introduction to Version Control Systems (VCS)**

- What is VCS & Git ?
- Use Cases and Benefits in Software Development

## **Git Basics**

- Setting Up a Git Repository
- Initializing a Repository
- Cloning a Repository
- Pushing Code to a Repository
- Adding and Committing Changes
- Pushing to Remote Repositories

## **Essential Git Commands**

- Basic Commands: add, commit, push, pull, clone
- Advanced Commands: Merge vs. Rebase, Cherry Pick, Reset & Revert, Stash & Pop, Branch Management

## **Real-Time Corporate Branching Strategy**

- Branching Strategy
- Feature Branches, Release Branches, Hotfix Branches
- Best Practices for Managing Branches in a Corporate Environment

## **Documentation**

- Comprehensive Git Guide
- Detailed Documentation Covering All Commands and Best Practices

## **Interview Preparation**

- Git Commands Overview
- Complete List of Git Commands with Explanations
- 200 Interview Questions
- Common Git Interview Questions and Model Answers
- Errors & Troubleshooting
- Common Git Errors and How to Resolve Them

# Module 4: Build Tools

## **Introduction to Build Tools**

- What Are Build Tools?
- Definition and Purpose
- Importance in the Software Development Lifecycle

## **Maven Deep Dive**

- Overview of Maven
- What is Maven?
- Key Concepts and Architecture
- Setting Up Maven
- Installation and Configuration
- Understanding pom.xml
- Building a Full-Stack Project with Maven
- Creating a Maven Project
- Managing Dependencies
- Building and Packaging the Project
- Running Tests and Generating Reports
- Maven Plugins
- Commonly Used Plugins
- Customizing Build Processes with Plugins

## **npm (Node Package Manager)**

- Overview of npm
- What is npm?
- Key Concepts and Architecture
- Setting Up npm
- Installation and Configuration
- Understanding package.json
- Managing a Node.js Project with npm
- Creating a Node.js Project
- Installing and Managing Dependencies
- Running Scripts and Tasks
- Building and Packaging the Project

## Documentation

- Comprehensive Maven Guide
- Detailed Documentation Covering Maven Setup, Configuration, and Usage
- Comprehensive npm Guide
- Detailed Documentation Covering npm Setup, Configuration, and Usage

## Interview Preparation

- Build Tool Commands Overview
- Complete List of Commands with Explanations
- 200 Interview Questions
- Common Interview Questions and Model Answers
- Errors & Troubleshooting
- Common Errors and How to Resolve Them

# Module-5 | CI/CD Tools

## (Jenkins, GitHub Actions, GitLab CI/CD)



### Jenkins

- Jenkins Setup
- Installation and Initial Configuration
- Configure Jenkins
- Setting Up Tools and Plugins
- Server Configurations
- Jenkins Jobs
- Creating Free Style, Pipeline, and Multibranch Pipeline Jobs
- Automatic Triggering Jobs
- Setting Up Generic and Multibranch Webhooks
- Integrations
- Integrate with SonarQube, Security Tools, Nexus, Docker, and Kubernetes
- Full Stack CI/CD Pipeline
- Building and Deploying an Application
- Notifications
- Sending Email Notifications
- Jenkins Backup



### GitHub Actions

- Setup Project Repository in GitHub
- Initial Setup and Configuration
- Setup Runners
- Configuring Self-Hosted and GitHub-Hosted Runners
- Create the CI/CD YAML Pipeline
- Writing and Configuring YAML Files
- Integrations
- Integrate with SonarQube, Security Tools, Docker, and Kubernetes
- Full Stack CI/CD Pipeline
- Building and Deploying an Application

## **GitLab CI/CD**

- Setup Project Repository in GitLab
- Initial Setup and Configuration
- Setup Runners
- Configuring GitLab Runners
- Create the CI/CD YAML Pipeline
- Writing and Configuring YAML Files
- Integrations
- Integrate with SonarQube, Security Tools, Docker, and Kubernetes
- Full Stack CI/CD Pipeline
- Building and Deploying an Application

## **Documentation and Interview Preparation**

- Comprehensive Documentation
- Detailed Guides and Best Practices for Jenkins, GitHub Actions, and GitLab CI/CD
- Hands-on Documentation
- 200 Interview Questions
- Errors and Troubleshooting

# Module-6 | SonarQube

## ⌚ Introduction to SonarQube

- Understanding SonarQube
- Overview and Key Features
- Sonar Scanner Tool vs. SonarQube Server
- Differences and Use Cases

## ⌚ Code Quality Issues

- Types of Issues
- Bugs, Code Smells, Vulnerabilities, Duplication, and More
- Code Quality Check & Code Coverage
- Importance and Methods of Checking Code Quality and Coverage

## ⌚ Setting Up SonarQube

- Installation and Configuration
- Step-by-Step Setup of SonarQube Server
- Configuring Sonar Scanner

## ⌚ Quality Profiles, Rules, and Gates

- Understanding Quality Profile
- Creating and Managing Quality Profiles
- Rules
- Defining and Managing Rules for Code Analysis
- Quality Gates
- Setting Up and Configuring Quality Gates

## ⌚ Implementing SonarQube Analysis using CI/CD Tools

- Integration with CI/CD Tools
- Configuring SonarQube Analysis in Jenkins, GitHub Actions, and GitLab CI/CD
- Automating Code Analysis in the CI/CD Pipeline

## 📋 Documentation and Hands-On Practice

- Comprehensive Documentation

- Complete Steps and Scripts for Setting Up and Using SonarQube
- Hands-on Exercises for Practical Learning

## Interview Preparation

- Comprehensive Interview Prep
- 200 Interview Questions and Answers
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-7 | Security Tools

## (Trivy, OWASP Dependency Check, Prowler, Dockle)

### Introduction to Security Tools

- Understanding the Tools and Their Functions
- Trivy: Container Security Scanner for Vulnerabilities
- OWASP Dependency Check: Identifies Vulnerable Dependencies in Project
- Prowler: AWS Security Best Practices Assessment Tool
- Dockle: Docker Container Linter for Security and Best Practices

### Implementation

- Implementing Trivy
- Installation and Setup
- Scanning Containers for Vulnerabilities
- Implementing OWASP Dependency Check
- Installation and Setup
- Scanning Project Dependencies for Vulnerabilities
- Implementing Prowler
- Installation and Setup
- Running Security Assessments on AWS Environments
- Implementing Dockle
- Installation and Setup
- Linting Docker Images for Security and Best

Practices

### Interview Preparation

- Comprehensive Interview Prep
- 200 Interview Questions and Answers on Security Tools
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-8 | Nexus

## **Introduction to Nexus**

- Understanding Nexus & Artifacts
- Overview of Nexus Repository Manager
- Types of Artifacts Managed by Nexus

## **Types of Repositories in Nexus**

- Repository Types
- Hosted, Proxy, and Group Repositories
- Use Cases for Each Type

## **Setting Up Nexus**

- Installation and Configuration
- Step-by-Step Setup of Nexus Repository Manager
- Initial Configuration and User Management

## **Integration with CI/CD Tools**

- Integrating Nexus with CI/CD
- Configuring Jenkins, GitHub Actions, and GitLab CI/CD to Use Nexus
- Automating Artifact Deployment to Nexus

## **Artifact Management**

- Deploying Artifacts to Nexus
- Manual and Automated Deployment Processes
- Release vs. Snapshots
- Differences and Use Cases for Release and Snapshot Repositories

## **Private Docker Registry**

- Setting Up a Private Docker Registry in Nexus
- Configuration Steps for Docker Registry
- Pushing and Pulling Docker Images

## Maintenance and Cleanup

- Cleanup Policies in Nexus
- Configuring and Managing Cleanup Policies for Repositories
- Automating Cleanup Processes

## Interview Preparation

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to Nexus
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-9 | Docker

## **Introduction to Docker**

- What is Docker?
- Overview and Key Concepts
- Benefits of Containerization
- Dockerfile vs. Docker Image vs. Docker Container
- Definitions and Differences

## **Writing Dockerfiles**

- How to Write a Dockerfile
- Basic Syntax and Commands
- CMD vs. ENTRYPOINT
- Differences and Use Cases
- ADD vs. COPY
- Differences and Use Cases

## **Docker Container Manipulation**

- Managing Containers
- Starting, Stopping, and Removing Containers
- Inspecting and Logging Containers

## **Private Docker Registry**

- Setting Up a Private Docker Registry
- Configuration and Usage
- Pushing and Pulling Images

## **Advanced Dockerfile Techniques**

- Multistage Dockerfile
- Creating Optimized and Smaller Images

## **Docker Compose**

Introduction to Docker Compose

Defining and Running Multi-Container Applications

- Writing docker-compose.yml Files
- 
-

## Docker Networking

- Understanding Docker Network
- Creating and Managing Networks
- Network Drivers and Their Uses

## Docker Volumes

- Managing Data with Docker Volumes
- Creating and Using Volumes
- Persisting Data in Containers

## Projects

- Practical Projects
- Hands-On Projects to Reinforce Learning

## Interview Preparation

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to Docker
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-10 | Kubernetes

## **Introduction to Kubernetes**

- What is Kubernetes?
- Overview and Key Concepts
- Benefits of Container Orchestration

## **Kubernetes Architecture**

- Master Node
- Components: API Server, Controller Manager, Scheduler, etcd
- Worker Node
- Components: Kubelet, Kube-Proxy, Container Runtime

## **Advanced Kubernetes Concepts**

- Taints & Tolerations
- Understanding and Configuring Taints and Tolerations

## **Cluster Setup**

- Self-Hosted Kubernetes Cluster Setup
- Step-by-Step Guide to Setting Up a Kubernetes Cluster
- EKS Cluster Setup
- Setting Up Kubernetes on AWS Using EKS

## **CI/CD Integrations**

- Integrating with CI/CD Tools
- Configuring Jenkins, GitHub Actions, and GitLab CI/CD with Kubernetes

## **Networking in Kubernetes**

- Ingress
- Setting Up and Managing Ingress Controllers and Resources

## **Troubleshooting**

Errors & Troubleshooting  
Common Issues and Solutions in Kubernetes

- 
-

## **Projects**

- Practical Projects
- Hands-On Projects to Apply Kubernetes Knowledge

## **Interview Preparation**

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to Kubernetes
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-11 | Azure DevOps

## ● **Introduction to Azure DevOps**

- Azure Cloud vs. Azure DevOps
- Differences and Use Cases

## ● **Setting Up Azure DevOps**

- Initial Setup
- Creating an Organization and Projects

## ● **Core Azure DevOps Services**

- Azure Repos
- Managing Source Code
- Azure Artifacts
- Packaging and Dependency Management
- Azure Pipelines
- Building and Deploying Applications
- Azure Kubernetes Service (AKS)
- Managing Kubernetes Clusters
- Azure Container Registry (ACR)
- Storing and Managing Container Images
- Azure App Service
- Deploying Web Applications

## ● **Configurations**

- Adding a VM as an Agent
- Configuring a Virtual Machine to Act as a Build and Deployment Agent

## ● **CI/CD Pipelines**

- Pipeline Using Classic Editor
- Setting Up a CI/CD Pipeline to Deploy Applications to Kubernetes
- Pipeline to Deploy to Azure App Service
  - Creating a CI/CD Pipeline for Web App Deployment
  - YAML Pipeline to Deploy to AKS
- 
- 
-

## ⌚ Interview Preparation

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to Azure DevOps
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-12 | Infrastructure as Code

## Terraform & Ansible

### Introduction to IaC

- What is IaC?
- Definition and Benefits
- Key Concepts and Principles

### Ansible Deep Dive

- Overview of Ansible
- What is Ansible?
- Key Features and Architecture
- Setting Up Ansible
- Installation and Configuration
- Ansible Playbooks
- Writing and Managing Playbooks
- Modules and Roles
- Using Built-in Modules and Creating Custom Roles
- Ansible in CI/CD
- Integrating Ansible with CI/CD Pipelines

### Terraform Deep Dive

- Overview of Terraform
- What is Terraform?
- Key Features and Architecture
- Setting Up Terraform
- Installation and Configuration
- Writing Terraform Configuration Files
- Understanding HCL (HashiCorp Configuration Language)
- Managing Infrastructure with Terraform
- Provisioning and Managing Infrastructure
- Terraform Modules
- Creating and Using Modules
- State Management

•

- Managing State Files and Remote State

## **Real-Time Corporate Projects**

- Practical Projects
- Hands-On Projects Using Ansible and Terraform
- Real-World Scenarios and Use Cases

## **Interview Preparation**

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to IaC, Terraform, and Ansible
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-13 | Monitoring

## **Introduction to Monitoring**

- Importance of Monitoring in DevOps
- Overview of Monitoring Principles and Practices

## **Prometheus**

- Overview of Prometheus
- What is Prometheus?
- Key Features and Architecture
- Installation and Configuration
- Monitoring with Prometheus
- Writing Prometheus Queries
- Setting Up Alert Rules

## **Grafana**

- Overview of Grafana
- What is Grafana?
- Key Features and Architecture
- Setting Up Grafana
- Installation and Configuration
- Visualizing Data with Grafana
- Creating and Managing Dashboards
- Integrating Grafana with Prometheus

## **Alertmanager**

- Overview of Alertmanager
- What is Alertmanager?
- Key Features and Architecture
- Setting Up Alertmanager
- Installation and Configuration
- Managing Alerts
- Configuring Alerting Rules
- Setting Up Notification Channels

## **Blackbox Exporter**

•

- Overview of Blackbox Exporter
- What is Blackbox Exporter?
- Key Features and Use Cases
- Setting Up Blackbox Exporter
- Installation and Configuration
- Probing with Blackbox Exporter
- Configuring Probes for Various Endpoints

## **Node Exporter**

- Overview of Node Exporter
- What is Node Exporter?
- Key Features and Use Cases
- Setting Up Node Exporter
- Installation and Configuration
- Monitoring System Metrics
- Collecting and Analyzing System Metrics

## **Real-Time Corporate Projects**

- Practical Projects
- Hands-On Projects Using Prometheus, Grafana, Alertmanager, Blackbox Exporter, and Node Exporter
- Real-World Scenarios and Use Cases

## **Interview Preparation**

- Comprehensive Interview Prep
- 200 Interview Questions and Answers Related to Monitoring Tools
- Common Errors and Troubleshooting Tips
- Real-Time Use Cases and Scenarios

# Module-14 | Resume Building, Corporate Projects Real-Time Troubleshooting

## **Resume Building & Review**

- Creating an Effective Resume
- Key Components of a DevOps Resume
- Highlighting Skills, Projects, and Experience
- Resume Review
- Common Mistakes to Avoid
- Tips for Tailoring Your Resume to Job Descriptions

## **Understanding Corporate Projects**

- Overview of Corporate Projects
- Importance of Understanding Project Architecture and Workflow
- Detailed Analysis of 5 Corporate Projects
  - Project 1: Architecture & Workflow
  - Project 2: Architecture & Workflow
  - Project 3: Architecture & Workflow
  - Project 4: Architecture & Workflow
  - Project 5: Architecture & Workflow
- Key Technologies and Tools Used in Each Project
- Roles and Responsibilities Within the Project Teams

## **Real-Time Troubleshooting**

- Troubleshooting Techniques
- Systematic Approach to Problem-Solving
- Tools and Methodologies for Effective Troubleshooting
- Real-Time Troubleshooting Scenarios
- Common Issues in DevOps Environments and How to Resolve Them
- Case Studies and Practical Examples

# Ultimate Projects

- Multi-Tier Applications (MySQL, Postgres, MongoDB)
- 11 Microservice Application
- Full Stack Projects (Java, Spring Boot, Dotnet)
- Dynamic Portfolio Website
- Virtual Browser Application
- Python-based Monitoring Application
- Terraform & Ansible Corporate Project
- Mega-Project Combining every tool we used above