VKonSec
We Konnect & Secure

DevOps

PLAN · CODE · BUILD · TEST · RELEASE · DEPLOY · OPERATE · MONITOR

Benefits of Doing a DevOps Course:

1. **High Demand in the Industry**:
   a. DevOps professionals are in high demand across all sectors, especially in IT, software development, and cloud infrastructure. Completing a DevOps course positions you to take on roles like **DevOps Engineer**, **Cloud Engineer**, or **Site Reliability Engineer**.
2. **Faster Deployment and Automation Skills**:
   a. You will learn to implement continuous integration (CI) and continuous deployment (CD) pipelines, allowing for **faster software releases** and automation of repetitive tasks. This increases efficiency in the software development lifecycle.
3. **Collaboration Between Teams**:
   a. DevOps emphasizes collaboration between development, operations, and security teams. By mastering these skills, you help bridge gaps, streamline workflows, and ensure smoother collaboration across departments.
4. **Cost-Effective Solutions**:
   a. With DevOps practices, companies can reduce errors, enhance software quality, and improve **resource utilization**, resulting in cost savings. Learning these methodologies helps you add value to organizations by optimizing their development processes.
5. **Cloud Integration**:
   a. DevOps courses often teach cloud platforms such as **AWS**, **Azure**, or **Google Cloud**, allowing you to integrate DevOps pipelines in cloud-based

environments. Cloud knowledge is essential as more businesses migrate to the cloud.

6. **Career Advancement**:
   a. DevOps professionals tend to earn higher salaries due to the technical nature of their job. Certified DevOps practitioners are highly sought after, making this certification a **career-booster** in terms of job roles and compensation.

7. **Increased Reliability and Security**:
   a. DevOps introduces **Infrastructure as Code (IaC)** and **automated monitoring**, which helps improve the reliability and security of systems. These practices are beneficial for organizations aiming for **scalability** and **robustness**.

Prerequisites for a DevOps Course:

1. **Basic Knowledge of Operating Systems**:
   a. Understanding of **Linux** and **Windows** operating systems, as DevOps tools often require configuring servers and managing different environments.

2. **Familiarity with Scripting Languages**:
   a. Proficiency in **Bash**, **Python**, or **Shell scripting** is highly recommended. These are essential for automating processes and building custom scripts.

3. **Networking Basics**:
   a. Knowledge of **networking concepts** like DNS, TCP/IP, and firewall configurations will help you understand how different services and applications interact in a DevOps environment.

4. **Version Control (Git)**:
   a. Experience with version control tools like **Git** is necessary, as DevOps relies heavily on collaboration and tracking changes in code.

5. **Containerization and Virtualization**:

a. Familiarity with tools like **Docker** and **Kubernetes** is beneficial, as these are commonly used to manage applications in a containerized environment.

6. **Understanding of Software Development Life Cycle (SDLC)**:
   a. Knowledge of the development process is essential, as DevOps integrates development with operations and ensures smooth transitions between different stages of the SDLC.

7. **Experience with Cloud Platforms** (optional but beneficial):
   a. Experience with **AWS**, **Azure**, or **Google Cloud** is recommended as many DevOps practices are implemented in cloud environments.

Recommended Prerequisite Courses:

- **Linux Essentials**
- **Python for Automation**
- **Networking Fundamentals**
- **Git and GitHub** basics

By ensuring you meet these prerequisites, you can better understand the full scope of DevOps practices and be well-prepared to handle real-world challenges in modern software and cloud environments.

Here is the **DevOps syllabus broken down in hours** based on a standard class schedule, focusing on theory and practical lab sessions:

DevOps Course Syllabus with Time Breakdown in Hours

1. Introduction to DevOps (4 Hours)

- **Sub-topics**:
    - What is DevOps? (1 hour)

o History and Evolution of DevOps (1 hour)
o DevOps Culture and Practices (1 hour)
o Business Benefits of DevOps (1 hour)

2. Version Control with Git and GitHub (8 Hours)

- **Sub-topics**:
  o Basics of Version Control (1 hour)
  o Installing Git and GitHub Setup (1 hour)
  o Branching, Merging, and Rebasing (2 hours)
  o Git Workflows (e.g., Gitflow) (2 hours)
  o Hands-on Lab: Using GitHub for Source Code Management (2 hours)

3. Continuous Integration with Jenkins (8 Hours)

- **Sub-topics**:
  o CI/CD Pipelines Overview (1 hour)
  o Installing and Configuring Jenkins (2 hours)
  o Jenkins Pipeline as Code (2 hours)
  o Integration with GitHub (1 hour)
  o Automated Testing with Jenkins (1 hour)
  o Hands-on Lab: Setting Up a Simple CI Pipeline (1 hour)

4. Infrastructure as Code (IaC) with Terraform/Ansible (12 Hours)

- **Sub-topics**:
  o Introduction to Infrastructure as Code (2 hours)
  o Terraform: Setting Up Infrastructure on AWS/Azure (4 hours)
  o Ansible for Configuration Management (3 hours)
  o Automation of Provisioning Infrastructure (2 hours)
  o Hands-on Lab: Automating Infrastructure Setup (1 hour)

5. Containerization with Docker (8 Hours)

- **Sub-topics**:
    - Introduction to Containers and Docker (2 hours)
    - Docker Architecture and Installation (1 hour)
    - Creating and Managing Docker Containers (2 hours)
    - Docker Networking and Volumes (1 hour)
    - Docker Compose for Multi-Container Applications (1 hour)
    - Hands-on Lab: Dockerize a Simple Application (1 hour)

6. Orchestration with Kubernetes (12 Hours)

- **Sub-topics**:
    - Introduction to Kubernetes and Container Orchestration (2 hours)
    - Kubernetes Architecture (Pods, Services, Nodes) (2 hours)
    - Kubernetes Installation and Setup (Minikube) (2 hours)
    - Deploying and Managing Applications on Kubernetes (3 hours)
    - Scaling, Monitoring, and Logging (2 hours)
    - Hands-on Lab: Deploy a Containerized Application to Kubernetes (1 hour)

7. Monitoring and Logging (8 Hours)

- **Sub-topics**:
    - Introduction to Monitoring Tools (Prometheus, Grafana) (2 hours)
    - Centralized Logging with ELK Stack (2 hours)
    - Application Performance Monitoring (APM) (2 hours)
    - Hands-on Lab: Setting Up Monitoring and Logging for an Application (2 hours)

8. Cloud Platforms (AWS/Azure) and DevOps (12 Hours)

- **Sub-topics**:
    - Cloud Computing Fundamentals (2 hours)

- o Introduction to AWS/Azure Cloud Services (4 hours)
- o CI/CD on Cloud (Using AWS CodePipeline, Azure DevOps) (3 hours)
- o Cloud Infrastructure Management with Terraform (2 hours)
- o Hands-on Lab: Deploying Infrastructure and Applications on Cloud (1 hour)

9. Security in DevOps (DevSecOps) (8 Hours)

- **Sub-topics**:
  - o Introduction to DevSecOps (2 hours)
  - o Integrating Security in CI/CD Pipeline (2 hours)
  - o Security Testing Automation (2 hours)
  - o Hands-on Lab: Implementing Security Checks in Pipelines (2 hours)

10. Advanced Topics (8 Hours)

- **Sub-topics**:
  - o Microservices Architecture (2 hours)
  - o Service Mesh (Istio/Linkerd) (2 hours)
  - o Serverless Computing (AWS Lambda) (2 hours)
  - o Continuous Delivery Best Practices (2 hours)

Total Duration: ~80 Hours

- **Theory (40%)**: About 32 hours will focus on theoretical concepts.
- **Hands-on Labs (60%)**: Around 48 hours will involve practical labs and real-world implementation.

The exact time division may vary depending on the pace of the class and the depth required for certain topics. This syllabus ensures that the learner gets a solid foundation in DevOps with both theoretical knowledge and hands-on skills.