



DevOps Certification Training

In Collaboration with **IBM** and **NASSCOM FutureSkills**

Table of Contents

1. About the Program
2. Collaborating with IBM
3. Collaborating with NASSCOM FutureSkills
4. About Intellipaath
5. Key Features
6. Career Support
7. Why take up this course?
8. Who should take up this course?
9. Program Curriculum
10. Project Work
11. Certification
12. Intellipaath Success Stories
13. Contact Us





According to research, the DevOps market is expected to reach \$6.6 billion by 2022.

About the Program

The DevOps certification training course online will help you learn DevOps and master various aspects of software development, operations, continuous integration, continuous delivery, and automated build, test, and deployment. In this best DevOps training course, you will learn DevOps tools, such as Git, Puppet, Jenkins, SVN, Maven, Docker, Ansible, Nagios, and more. This is India's first DevOps course validated and certified by NASSCOM FutureSkills and IBM, and it aligns to industry standards.

IBM

Collaborating with IBM

IBM is one of the leading innovators and the biggest player in creating innovative tools. Top subject matter experts from IBM will share their knowledge in the domain through this training program that will help you gain the breadth of knowledge and industry experience.

Benefits for students from IBM

- Industry-recognized IBM certificate
- Access to IBM Watson for hands-on training and practice
- Industry in-line case studies and project work



Collaborating with FutureSkills

FutureSkills is an industry-driven learning ecosystem to get India accelerated in the journey to building skills and becoming the global hub for talent in emerging technologies.

Intellipaate is a NASSCOM FutureSkills Ecosystem Partner and NASSCOM FutureSkills Platform Partner. Our AWS certification training is aligned with competency standards developed by SSC NASSCOM in collaboration with the industry and approved by the government.

Benefits for students from FutureSkills

- Industry-recognized, FutureSkills and Intellipaate co-branded certificate
- Job-ready skills upon completing the training
- Industry in-line case studies and project work

About IntelliPaat

IntelliPaat is one of the leading e-learning training providers with more than 600,000 learners across 55+ countries. We are on a mission to democratize education as we believe that everyone has the right to quality education.

Our courses are delivered by subject matter experts from top MNCs, and our world-class pedagogy enables learners to quickly learn difficult topics in no time. Our 24/7 technical support and career services will help them jump-start their careers in their dream companies.

Key Features



**33 HRS INSTRUCTOR-LED
TRAINING**



30 HRS SELF-PACED TRAINING



**60 HRS REAL-TIME
PROJECT WORK**



LIFETIME ACCESS



24/7 TECHNICAL SUPPORT



**INDUSTRY-RECOGNIZED
CERTIFICATION**



**JOB ASSISTANCE THROUGH
80+ CORPORATE TIE-UPS**



FLEXIBLE SCHEDULING

Career Support



SESSIONS WITH INDUSTRY MENTORS

Attend sessions from top industry experts and get guidance on how to boost your career growth



MOCK INTERVIEWS

Mock interviews to make you prepare for cracking interviews by top employers



GUARANTEED INTERVIEWS & JOB SUPPORT

Get interviewed by our 400+ hiring partners



RESUME PREPARATION

Get assistance in creating a world-class resume from our career services team



Why take up this course?

The DevOps industry is expected to grow six times higher by the year 2022, according to Gartner. The average annual salary of a DevOps Engineer in the United States is US\$136,500, and in India, it is ₹666,232 (PayScale). Also, DevOps and Cloud jobs are going to be among the highest-paying jobs in the coming years, according to Forbes. All this is because a DevOps Architect is the backbone of any organization's technical team.

Who should take up this course?

- IT Professionals
- Software Testers
- Systems Admins
- Solutions Architects
- Security Engineers
- Application Developers
- Integration Specialists

Program Curriculum

DevOps Certification Course Content

1. INFRASTRUCTURE SETUP

1.1 EC2 Walkthrough

1.2 Installation of DevOps Tools in the Cloud

- Git
- Docker
- Selenium
- Maven
- Jenkins
- Puppet
- Ansible
- Kubernetes
- Nagios

2. INTRODUCTION TO DEVOPS

2.1 What is Software Development?

2.2 Software Development Life Cycle

2.3 Traditional Models for SDLC

2.4 Why DevOps?

2.5 What is DevOps?

2.6 DevOps Lifecycle

2.7 DevOps Tools

3. CONTINUOUS TESTING

3.1 What is Continuous Testing?

3.2 What is Maven?

3.3 Running Test Cases on Chromium WebDriver

3.4 What is the Headless Mode?

***Hands-on Exercise:** Using Maven to import dependencies in Eclipse and implementing a headless test using Chrome WebDriver*

4. CONTINUOUS INTEGRATION USING JENKINS

4.1 Introduction to Continuous Integration

4.2 Jenkins Master Slave Architecture

4.3 Understanding CI/CD Pipelines

4.4 Creating an End-to-end Automated CI/CD Pipeline

***Hands-on Exercise:** Creating a Jenkins Master Slave on AWS, installing plug-ins in Jenkins, creating Jenkins builds, creating scheduled builds, triggering jobs using Git web hooks, and using the pipeline plugin in Jenkins*

5. SOFTWARE VERSION CONTROL

5.1 What is Version Control?

5.2 Types of Version Control Systems

5.3 Introduction to SVN

5.4 Introduction to Git

5.5 Git Lifecycle

5.6 Common Git Commands

5.7 Working with Branches in Git

5.8 Merging Branches

5.9 Resolving Merge Conflicts

5.10 Git Workflow

***Hands-on Exercise:** Git life cycle commands, pushing code to GitHub, stashing code in Git, creating and deleting Git branches, reverting a push to GitHub, merging branches using git merge, merging branches using git rebase, and resolving merge conflicts using the git merge tool*

6. CONTINUOUS DEPLOYMENT: CONTAINERIZATION WITH DOCKER

6.1 Introduction to Docker

6.2 Understanding Docker Lifecycle

6.3 Components of the Docker Ecosystem

6.4 Common Docker Operations

- 6.5 Creating a Docker Hub Account
- 6.6 Committing Changes in a Container
- 6.7 Pushing a Container Image to Docker Hub
- 6.8 Creating Custom Docker Images Using a Dockerfile

Hands-on Exercise: Common Docker operations, creating a Docker Hub account, committing changes to a container, pushing the container to Docker Hub, creating local image repository, and building an image using a Dockerfile

7. CONTAINERIZATION WITH DOCKER: ECOSYSTEM AND NETWORKING

- 7.1 What are Docker Volumes?
- 7.2 Deploying a Multi-tier Application Using Docker Network
- 7.3 Using Docker Compose to Deploy Containers
- 7.4 What is Container Orchestration?
- 7.5 Container Orchestration Tools
- 7.6 Introduction to Docker Swarm
- 7.7 Deploying a 2-node Cluster Using Docker Swarm

Hands-on Exercise: Creating Docker volumes, using Docker Compose to deploy multiple containers, deploying a multi-node cluster using Docker Swarm, and deploying a multi-service app on Docker Swarm

8. CONFIGURATION MANAGEMENT USING PUPPET

- 8.1 Need of Configuration Management
- 8.2 Configuration Management Tools
- 8.3 What is Puppet?
- 8.4 Puppet Architecture
- 8.5 Setting up Master Slave Using Puppet
- 8.6 Puppet Manifests
- 8.7 Puppet Modules
- 8.8 Applying Configuration Using Puppet
- 8.9 Puppet File Server

Hands-on Exercise: Setting up Master Slave on AWS, testing the connection of nodes with Puppet, creating a Manifest, deploying Manifest on a node, creating a Module, deploying sample software on nodes using Puppet Modules and Manifests, and implementing a File Server Module on Puppet

9. CONFIGURATION MANAGEMENT USING ANSIBLE

- 9.1 What is Ansible?
- 9.2 Ansible vs Puppet
- 9.3 Ansible Architecture
- 9.4 Setting up Master Slave Using Ansible
- 9.5 Ansible Playbook
- 9.6 Ansible Roles
- 9.7 Applying Configuration Using Ansible

Hands-on Exercise: *Installing Ansible on AWS, creating a Playbook using YAML, creating an Ansible role, and using roles in the playbook*

10. CONTINUOUS ORCHESTRATION USING KUBERNETES

- 10.1 Introduction to Kubernetes
- 10.2 Docker Swarm vs Kubernetes
- 10.3 Kubernetes Architecture
- 10.4 Deploying Kubernetes Using Kubeadms
- 10.5 Alternate Ways of Deploying Kubernetes
- 10.6 YAML Files
- 10.7 Creating a Deployment in Kubernetes Using YAML
- 10.8 Services in Kubernetes
- 10.9 Ingress in Kubernetes
- 10.10 Case Study – Kubernetes Architecture

Hands-on Exercise: *Setting up Kubernetes using kubeadm, installing Kubernetes using kops and GCK, creating a deployment, creating services, creating an ingress, and demonstrating the use of ingress, services and deployments together*

11. CONTINUOUS MONITORING USING NAGIOS

- 11.1 What is Continuous Monitoring?
- 11.2 Introduction to Nagios
- 11.3 Nagios Architecture
- 11.4 Monitoring Services in Nagios
- 11.5 What are NRPE Plugins?
- 11.6 Monitoring System Info Using NRPE Plugins

Hands-on Exercise: *Installing Nagios and monitoring different servers using Nagios*

12. TERRAFORM MODULES & WORKSPACES

- 12.1 What is Infrastructure-as-Code?
- 12.2 IaC vs Configuration Management
- 12.3 Introduction to Terraform
- 12.4 Installing Terraform on AWS
- 12.5 Basic Operations in terraform

- Init
- Plan
- Apply
- Destroy

- 12.6 Terraform Code Basics
- 12.7 Deploying an End-to-end Architecture on AWS Using Terraform

***Hands-on Exercise:** Installing Terraform, initializing AWS Terraform Provider, creating an EC2 instance using Terraform, updating changes to EC2 using Terraform, destroying EC2 using Terraform, and deploying EC2 inside a custom VPC using Terraform*

Project Work

DevOps Projects Covered

Creating a CI/CD Pipeline to Build and Test a Website

Here, you will create a CI/CD Pipeline to integrate DevOps tools such as Git, Docker, Jenkins, Puppet, and Selenium. In this project, you will be containerizing a website which will be pushed to Git and then will be deployed automatically on the testing server, and if everything goes well, it will be pushed to production servers.

Deploying an App on Kubernetes Using CI/CD Pipelines

In this project, you will be creating a CI/CD pipeline that is integrated with Kubernetes. This project will make use of tools such as Git, Docker, Jenkins, Selenium, and Kubernetes. First, you will containerize the code pushed into Git, save the Docker image, and then push the image to Docker Hub. Finally, it will be deployed on Kubernetes.

Merging Git Branches

In this project, you will understand the importance of merging branches in Git. You will be using a production use case where merging is required. Based on the use case, you will have to decide whether you need to use git merge or rebase. Since this is a production problem, you will also have to make use of Git Workflows.

Implementing Git Workflow

In this project, you will be analyzing and using a specific Git Workflow scenario. Here, you will work with industry-standard architectures, merging branches, and simulating a working architecture in Git. You will have to be familiar with topics such as Git branches, Git workflows, merging, and rebasing to work on this case study.

Creating Custom Containers in Docker

Here, you have to create a custom container from the base Ubuntu container. In this case study, you will be required to create a Dockerfile, push the container to Docker Hub, and create custom Docker images. Through this case study, you will cover the working knowledge and implementation of Dockerfile and Docker Hub in DevOps.

Using Bind Mounts, Docker Compose, and Docker Swarm

In this case study, you will be deploying an application using Docker Compose, Bind Mounts, and Docker Swarm. You will create and use Bind Mounts in Docker, deploy multiple Docker applications with Docker Compose, and create a Docker Swarm cluster and deploy applications on it. You will also use Docker storage and Docker networks.

Using Puppet for Configuration Management

In this case study, you will be learning how to work with Docker, Puppet Manifests, conditional statements, modules, and classes to create Modular scripts and also to manage a large fleet of servers with Puppet.

Using Selenium to Create Automated Test Cases

In this case study, you will be required to create automated test cases using Selenium. You will also have to create runnable jar files and run headless tests in Chrome using non-GUI Linux. Through this case study, you will have a thorough understanding of how Selenium, runnable jar files, ChromeDriver, and headless tests work.

Certification

After the completion of the course, you will get a certificate from FutureSkills and IntelliPaat.



CERTIFICATE OF COMPLETION

This certificate is awarded to

Your Name

Who has successfully completed

Course Name

Fulfilling all the requirements stipulated by IntelliPaat to achieve professional excellence.

Issued Date: Month XX, XXXX



Success Stories



Kevin K Wada

Thank you very much for your top-class service. A special mention should be made for your patience in listening to my queries and giving me a solution, which was exactly what I was looking for. I am giving you a 10 on 10!



Sampson Basoah

The Intellipaate team helped me in selecting the perfect course that suits my profile. The whole course was practically oriented, and the trainers were always ready to answer any question. I found this course to be impactful. Thank you.



Dawn Berry

Oh, thank you for your wonderful and helpful response. I am so glad that I joined your training! It was truly the best course for DevOps.



Alok

Hi, I am a Cloud DevOps Engineer at Tech Mahindra. I completed DevOps training from Intellipaate. The course was good, and the support team was very helpful and supportive. The projects and assignments were quite good as well. The most important thing about Intellipaate is the certificate, which has huge value in the market, and this helped me a lot in my career. I surely recommend the DevOps certification training from Intellipaate.

CONTACT US

INTELLIPAAT SOFTWARE SOLUTIONS PVT. LTD.

Bangalore

AMR Tech Park 3, Ground Floor, Tower B,
Hongasandra Village, Bommanahalli,
Hosur Road, Bangalore – 560068

USA

1219 E. Hillsdale Blvd. Suite 205,
Foster City, CA 94404

If you have any further queries or just want to have a conversation with us, then do call us.

IND: +91-7022374614 | US: 1-800-216-8930