DevOps Bootcamp

6-month hands-on program to acquire the complete DevOps and Cloud engineering skillset

By TechWorld with Nana

About the Program

Your last stop to mastering DevOps and Cloud Skills

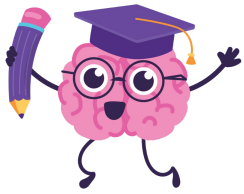
DevOps Bootcamp Description

This program will prepare you with no experience in DevOps to take over DevOps tasks in your company. You will learn the most in-demand skills, such as container orchestration with Kubernetes, how to build a complete production ready CI/CD pipeline, automation with Terraform, Ansible, Python and many other essential tools, like Docker, AWS cloud, Git, Linux etc, which they can directly apply to the projects.

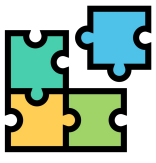
More importantly, you don't only learn the tools and their features, you learn the underlying concepts, which makes you able to transfer that knowledge to any alternative tool. You also learn the best practices behind those tools and concepts and why those best practices are important in different use cases.

All this knowledge and skills to get students ready to start implementing DevOps processes immediately in the company, is distilled in about 6 months of study. The Bootcamp format is designed to enable students to study at their own pace, revisit any videos and materials they want whenever convenient. Throughout the entire learning process you are also supported around the clock by experienced DevOps engineers in a dedicated study group.

By the end, you'll have



No knowledge gaps



Big Picture Understanding

Full DevOps Skillset

...And have become incredibly valuable on the job market!

About the Program

Key Components to accelerate your learning

Hands-On Projects

The heart of our program. You'll learn everything hands-on with real-life demos

Exercises for each Module

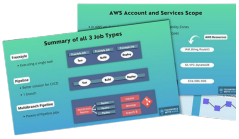
Additional practice and self-check whether you are able to do tasks independently

24/7 Support

Daily support from experienced engineers, so you don't get stuck

Engaging Community

Be part of an exclusive community that share your journey



Illustrated Handouts

Accompanying handout for each Module. Overview with Key Takeaways

DevOps Certification

Earn a verifiable, recognized digital credential to proof your knowledge

RECOMMENDED SCHEDULE

Recommended Schedule

10-15 hours / week

WEEK 1 — Overview & Basics of DevOps

WEEK 2 — OS & Linux Basics

WEEK 3 — Version Control with Git

WEEK 4 — Build & Package Manager Tools

Month 1

WEEK 5 — Cloud & IaaS

WEEK 6 — Artifact Repo Manager

WEEK 7 - 8 — Containers

WEEK 9 - 10 — Build Automation (CI/CD)

Month 2 / Month 3

WEEK 11 - 12 — AWS Services

WEEK 13 - 14 — Container Orchestration

WEEK 15 - 16 — K8s on AWS

WEEK 17 - 18 — IaC with Terraform

Month 3 / Month 4 / Month 5

WEEK 19 — Programming Basics

WEEK 20 - 21 — Automation with Python

WEEK 22 - 23 — Configuration Management

WEEK 24 - 25 — Monitoring with Prometheus

Month 5 / Month 6

BOOTCAMP SYLLABUS

1 INTRODUCTION TO DEVOPS

Key Learning Objectives

One essential part of learning DevOps is to understand how DevOps fits into the whole software development lifecycle and what are the roles and responsibilities of a DevOps engineer. This should give you an overview and introduction to DevOps, before later learning DevOps tasks in detail.

Module Curriculum

- ✔ Lesson 01: DevOps Bootcamp Overview
- ✔ Lesson 02: DevOps Overview
  - What is DevOps?
  - Roles and Responsibilities of a DevOps Engineer
  - How DevOps fits in the whole Software Development lifecycle

2 OPERATING SYSTEMS & LINUX BASICS

Key Learning Objectives

As a DevOps engineer you are responsible for preparing the infrastructure (servers) on which the application is deployed. So fundamentals of Operating Systems, Linux and feeling comfortable using the CLI is crucial.
So this is a prerequisite knowledge we teach in this program to enable learners to enroll from a non systems administrator background.

Module Curriculum 1/3

Lecture 01: What is an OS and how does it work?

- ✔ Tasks of an OS
- ✔ How an OS is constructed
- ✔ How different OSs, like Unix, Linux, Windows and MacOS differ from each other

Lesson 02: Virtualization

- ✔ Introduction to Virtual Machine
- ✔ Setup a Linux Virtual Machine

Lesson 03: Package Manager - Installing Software

- ✔ What is a Package Manager and what are Software Repositories?
- ✔ Learn all the options of installing software on Linux and how it all actually works in the background: APT, APT vs APT-GET, SNAP, Ubuntu Software Center, YUM

Lesson 04: Working with Vim Editor

- ✔ What is Vim?
- ✔ Learn most important Vim Commands to work with Vim efficiently

2 OPERATING SYSTEMS & LINUX BASICS

Module Curriculum 2/3

Lesson 05: Users & Permissions

- ✔ Linux Accounts
- ✔ Users, Groups & Permissions
- ✔ User Management in Practice
- ✔ File Ownership & Permissions
- ✔ Modifying Permissions

Lesson 06: Linux File System

Lesson 07: Basic Linux Commands

- ✔ Introduction to Command Line Interface
- ✔ Learn all the essential Linux Commands like:
  - Directory Operations
  - Navigating the Files System
  - Work with the File System (Create folders, list files, rename, remove files etc.)
  - Execute Commands as Superuser
  - Pipes, Redirects, Less, Grep

Lesson 08: Shell Scripting

- ✔ Shell vs sh vs Bash
- ✔ Write & execute a simple script
- ✔ Learn how to write Bash Scripts: Variables, Conditional Statements, Basic Operators, Passing Arguments to a Script to make it customizable and reusable, Read user input, Repeating code with shell loops, Functions

2 OPERATING SYSTEMS & LINUX BASICS

Module Curriculum 3/3

Lesson 05: Environment Variables

- ✔ What are environment variables and how to access them
- ✔ Create, Delete and Persist Env Variables
- ✔ Understand what the PATH env variables is

Lesson 05: Networking

- ✔ How computer networks work?
- ✔ What is LAN, Switch, Router, Subnet, Firewall, Gateway
- ✔ What is an IP address and port?
- ✔ What is a DNS and how does DNS resolution work?
- ✔ Useful Networking Commands

Lesson 05: SSH - Secure Shell

- ✔ What is SSH and how it works
- ✔ SSH in Action:
- ✔ Create Remote Server on Cloud
- ✔ Generate SSH Key Pair
- ✔ Execute a bash script on a remote machine

3 Version Control with Git

Key Learning Objectives

DevOps is all about automation and you write all these automation logic as code. You manage this code and configuration files with a version control tool.
So in this module you'll learn the most widely used one, which is Git. Throughout the bootcamp you will work with Git and save your code in your own Git repository to build a DevOps portfolio, which you can use to showcase your work later.

Module Curriculum

- ✔ Lesson 01: Introduction to Version Control and Git
- ✔ Lesson 02: Basic Concepts of Git
- ✔ Lesson 03: Setup git repository (remote and local)
- ✔ Lesson 04: Working with Git (git status, git commit, git add, git push)
- ✔ Lesson 05: Initialize Git project locally
- ✔ Lesson 06: Concept of Branches
- ✔ Lesson 07: Merge Requests
- ✔ Lesson 08: Deleting Branches
- ✔ Lesson 09: Avoiding Merge Commits (rebase)
- ✔ Lesson 10: Resolving Merge Conflicts
- ✔ Lesson 11: Don't track certain files (.gitignore)
- ✔ Lesson 12: Save work-in-progress changes (git stash)
- ✔ Lesson 13: Going back in history (git checkout)
- ✔ Lesson 14: Undoing commits (git revert, git reset)
- ✔ Lesson 15: Merging Branches
- ✔ Lesson 16: Git for DevOps

4 Build and Package Manager Tools

Key Learning Objectives

In this module you will learn how to build and package an application with build and package management tools as well as how to execute tests. It's a prerequisite knowledge. Understanding how this is done manually, will help you understand how it is automated later with CI/CD.

Module Curriculum

- ✔ Lesson 01: What are Build Tools and Package Managers?
- ✔ Lesson 02: How to build an artifact?
- ✔ Lesson 03: How to run the application artifact?
- ✔ Lesson 04: How to publish the application artifact to artifact repository?
- ✔ Lesson 05: Build Tools for Java (gradle and maven examples)
- ✔ Lesson 06: Dependency Management in Software Development
- ✔ Lesson 07: Package Manager in JavaScript applications - Build and run applications in JS
- ✔ Lesson 08: Build Tools & Docker
- ✔ Lesson 09: Why Build Tools are relevant for DevOps Engineers?

5 Cloud & Infrastructure as a Service

Key Learning Objectives

Nowadays many companies are using virtual infrastructure on the cloud, instead of managing their own infrastructure. So in this module you will learn about cloud concepts in general, how to set up a simple virtual server on a cloud and how to deploy an application (manually). This gives you a first introduction to this topic, later you'll learn the most popular cloud platform AWS in-depth.

Module Curriculum

- ✔ Lesson 01: Cloud & Infrastructure as a Service Concepts

Working with a cloud server:

- ✔ Lesson 02: Setup Server on DigitalOcean (Droplet)
- ✔ Lesson 03: Install Java on Cloud Server
- ✔ Lesson 04: Deploy and run an application on Cloud Server
- ✔ Lesson 05: Create a Linux User to login to Server (instead of using Root User)

6 Artifact Repository Manager with Nexus

Key Learning Objectives

Artifact Repositories and Artifact Repository Manager are important elements in the DevOps process. So in this module you will learn a popular artifact repository manager: Nexus. You will learn the general concepts, how it works and how to configure artifact repositories. Later you will learn how to integrate it in a complete CI/CD pipeline.

Module Curriculum

- ✔ Lesson 01: What is an Artifact Repository Manager?
- ✔ Lesson 02: Install and run Nexus on Cloud Server
- ✔ Lesson 03: Different Repository Types (proxy, hosted, etc.) explained
- ✔ Lesson 04: Different Repository Formats (maven, docker, npm, etc.) explained
- ✔ Lesson 05: Upload Jar File to Nexus (maven and gradle projects)
- ✔ Lesson 06: Nexus API and Repository URLs
- ✔ Lesson 07: Blob stores
- ✔ Lesson 08: Browsing Components - Components vs Assets
- ✔ Lesson 09: Cleanup Policies and Scheduled Tasks

7 Containers with Docker

Key Learning Objectives

Containers are the new standard in modern systems. This means you need to generally understand: concepts of virtualization and containers.
In this module you will learn everything to get a full picture of containers and how to work with containers.

Module Curriculum

- ✔ Lesson 01: What is a Container?
- ✔ Lesson 02: Docker Components and architecture explained
- ✔ Lesson 03: Docker vs. Virtual Machine
- ✔ Lesson 04: Main Docker Commands
- ✔ Lesson 05: Debugging a Docker Container
- ✔ Lesson 06: Demo Project Overview - Docker in Practice
- ✔ Lesson 07: Developing with Containers
- ✔ Lesson 08: Docker Compose - Running multiple services
- ✔ Lesson 09: Dockerfile - Building our own Docker Image
- ✔ Lesson 10: Private Docker Repository - Pushing our built Docker Image into a private Registry on AWS
- ✔ Lesson 11: Deploy containerized app
- ✔ Lesson 12: Docker Volumes - Persist data in Docker
- ✔ Lesson 13: Volumes Demo - Configure persistence for our demo project
- ✔ Lesson 14: Docker Best Practices

Docker & Nexus

- ✔ Lesson 15: Create Docker Images Repository on Nexus and Push/Pull Docker Image from/to Nexus Repository Manager
- ✔ Lesson 16: Deploy Nexus as Docker Container

8 Build Automation - CI/CD with Jenkins

Key Learning Objectives

Continuous Integration and Continuous Deployment is the heart of DevOps.

In this module you will learn the most used CI/CD platform in the industry, which is Jenkins. You'll learn the core concepts of CI/CD in general and how to set up a CI pipeline.

Module Curriculum 1/2

- ✔ Lesson 01: What is Build Automation? What is Jenkins?
- ✔ Lesson 02: Install Jenkins on cloud server (Docker vs Server install)
- ✔ Lesson 03: Jenkins plugins
- ✔ Lesson 04: Installing build tools in Jenkins
- ✔ Lesson 05: Jenkins Basics Demo
  - Create Freestyle Job
  - Configure Git Repository
  - Run Tests and Build Java Application
- ✔ Lesson 06: Docker in Jenkins
  - Make Docker commands available in Jenkins
  - Build Docker Image
  - Push to DockerHub Repo
  - Push to Nexus Repo
- ✔ Lesson 07: Jenkins Pipeline (Use Cases)
- ✔ Lesson 08: Create a simple Pipeline Job
- ✔ Lesson 09: Full Jenkinsfile Syntax Demo

8 Build Automation - CI/CD with Jenkins

Module Curriculum 2/2

- ✔ Lesson 10: Create a full Pipeline Job
  - Build Java App
  - Build Docker Image
  - Push to private DockerHub
- ✔ Lesson 11: Create a Multi-Branch Pipeline Job
- ✔ Lesson 12: Credentials in Jenkins
- ✔ Lesson 13: Jenkins Shared Library
- ✔ Lesson 14: WebHooks - Trigger Jenkins Jobs automatically
- ✔ Lesson 15: Versioning Application in Continuous Deployment
  - Concepts of Versioning in Software Development
  - Increment Application version from Jenkins Pipeline
  - Set new Docker Image version from Jenkins Pipeline
  - Commit Version Bump from Jenkins Pipeline

9 Cloud Platform - AWS Services

Key Learning Objectives

AWS is the most powerful and widely used cloud platform, but also one of the most complex ones. So understanding the core services of AWS and how to build a proper infrastructure using various AWS services is essential skill set of a DevOps or Cloud engineer. You learn all these in the dedicated AWS module.

Module Curriculum 1/2

- ✔ Lesson 01: Introduction to Amazon Web Services
- ✔ Lesson 02: Create an AWS Account
- ✔ Lesson 03: Identity & Access Management (IAM) - User, Groups and Permissions
- ✔ Lesson 04: Regions and Availability Zones
- ✔ Lesson 05: Virtual Private Cloud (VPC) - Manage Private Network on AWS
  - Subnets
  - Security Groups
  - Internet Gateway
  - Route Table
- ✔ Lesson 06: CIDR Blocks explained
- ✔ Lesson 07: Introduction to Elastic Compute Cloud (EC2)
  - Create an EC2 Instance
  - Run Web application on EC2 using Docker
- ✔ Lesson 08: AWS Command Line Tool
  - Install and configure AWS CLI
  - Create AWS components with AWS CLI

9 Cloud Platform - AWS Services

Module Curriculum 2/2

AWS & Jenkins - Continuous Deployment with Jenkins to AWS EC2

- ✔ Lesson 09: Automate deploying from Jenkins Pipeline to EC2 Instance
  - using docker run
  - using docker-compose
- ✔ Lesson 10: Real-life example of dynamically setting new image version in docker-compose
- ✔ Lesson 11: SSH agent plugin and SSH credential type in Jenkins

10 Container Orchestration with Kubernetes

Key Learning Objectives

In this module you will learn the most popular container orchestration platform: Kubernetes (K8s). K8s has already become a standard runtime for modern applications in cloud. K8s is also a very complex framework. In this module, you will master K8s to an advanced level und become absolutely confident working with K8s.

Module Curriculum 1/2

- ✔ Lesson 01: Introduction to Kubernetes
- ✔ Lesson 02: Understand the main Kubernetes Components
  - Node, Pod, Service, Ingress, ConfigMap, Secret, Volume, Deployment, StatefulSet
- ✔ Lesson 03: Kubernetes Architecture
- ✔ Lesson 04: Minikube and kubectl - Local Setup
- ✔ Lesson 05: Main Kubectl Commands - K8s CLI
  - Create and debug Pod in a Minicluster
- ✔ Lesson 06: Kubernetes YAML Configuration File
- ✔ Lesson 07: Demo Project: MongoDB and MongoExpress
- ✔ Lesson 08: Organizing your components with K8s Namespaces
- ✔ Lesson 09: Kubernetes Service Types
- ✔ Lesson 10: Making your App accessible from outside with Kubernetes Ingress
- ✔ Lesson 11: Persisting Data in Kubernetes with Volumes
  - Persistent Volume, Persistent Volume Claim, Storage Class
- ✔ Lesson 12: ConfigMap and Secret Kubernetes Volume Types
- ✔ Lesson 13: Deploying Stateful Apps with StatefulSet

10 Container Orchestration with Kubernetes

Module Curriculum 2/2

- ✔ Lesson 14: Deploying Kubernetes cluster on a Managed Kubernetes Service (K8s on Cloud)
- ✔ Lesson 15: Helm - Package Manager of Kubernetes
- ✔ Lesson 16: Helm Demo: Install a Stateful Application on Kubernetes using Helm
- ✔ Lesson 17: Demo: Deploy App from Private Docker Registry
- ✔ Lesson 18: Extending the Kubernetes API with Operator
- ✔ Lesson 19: Secure your cluster - Authorization with Role Based Access Control (RBAC)

Microservices in Kubernetes

- ✔ Lesson 20: Introduction to Microservices
- ✔ Lesson 21: Demo project: Deploy Microservices Application
- ✔ Lesson 22: Demo project: Create common Helm Chart for Microservices
- ✔ Lesson 23: Demo project: Deploy Microservices with helmfile

- ✔ Lesson 24: Production & Security Best Practices

11 Kubernetes on AWS - AWS EKS

Key Learning Objectives

In this module, you will learn about AWS EKS - the managed Kubernetes service of AWS. You will be able to create an AWS EKS cluster and how to deploy to it from a CI/CD pipeline, further building a production-grade CI/CD pipeline. This module builds on the skills from AWS and K8s modules and gives you advanced knowledge of working with EKS clusters.

Module Curriculum

AWS & Kubernetes

- ✔ Lesson 01: AWS Container Services: Overview (ECR, ECS, EKS, Fargate)
- ✔ Lesson 02: Create an EKS cluster with AWS Management Console (UI)
  - Create cluster VPC, cluster Roles
  - Use Cloudformation Stack
  - EC2 Worker Nodes
  - Configure kube context to connect to the cluster
- ✔ Lesson 03: Configure Autoscaling in EKS cluster
- ✔ Lesson 04: Create Fargate Profile for EKS cluster
- ✔ Lesson 05: Create an EKS cluster with eksctl (the easy way)

AWS & Kubernetes & Jenkins & Docker - CI/CD

- ✔ Lesson 06: Configure kubectl inside Jenkins
- ✔ Lesson 07: Configure kube context in Jenkins
- ✔ Lesson 08: Install aws-iam-authenticator in Jenkins
- ✔ Lesson 09: Complete Jenkins Pipeline - Deploy to EKS - using kubectl
- ✔ Lesson 10: Complete Jenkins Pipeline - Build and push docker image to ECR and deploy to EKS
- ✔ Lesson 11: Complete Jenkins Pipeline - Deploy to LKE using Kubernetes CLI plugin and kubeconfig file

12 Infrastructure as Code with Terraform

Key Learning Objectives

Manually creating and maintaining infrastructure is time consuming and error prone. In DevOps, we want to automate as much as possible and that's when Infrastructure as Code comes into the picture. In this module you will learn the most popular IaC tool: Terraform.

Module Curriculum 1/2

- ✔ Lesson 01: Introduction to Terraform
  - What is Terraform? How it works
  - Architecture
- ✔ Lesson 02: Install Terraform & Setup Terraform Project
- ✔ Lesson 03: Providers in Terraform
- ✔ Lesson 04: Resources & Data Sources
- ✔ Lesson 05: Change & Destroy Terraform Resources
- ✔ Lesson 06: Terraform commands
- ✔ Lesson 07: Terraform State
- ✔ Lesson 08: Output Values
- ✔ Lesson 09: Variables in Terraform
- ✔ Lesson 10: Environment Variables in Terraform
- ✔ Lesson 11: Create Git Repository for local Terraform Project

12 Infrastructure as Code with Terraform

Module Curriculum 2/2

Terraform & AWS

- ✔ Lesson 12 - Automate Provisioning EC2 server with Terraform
- ✔ Lesson 13 - Provisioners in Terraform
- ✔ Lesson 14 - Modularize the demo project

Terraform & AWS & Kubernetes

- ✔ Lesson 15 - Automate Provisioning EKS cluster with Terraform
  - Use existing modules from Terraform Registry
  - Create VPC
  - Provision EKS cluster

Terraform & AWS & Jenkins - Complete CI/CD

- ✔ Lesson 16: Complete CI/CD with Terraform
  - Configure Terraform in Jenkins
  - Automate provisioning EC2 instance from Jenkins pipeline and deploy the application with docker-compose

- ✔ Lesson 17: Remote State in Terraform
- ✔ Lesson 18: Terraform Best Practices

13

Programming with Python

Key Learning Objectives

As a DevOps engineer you may need to write automation scripts or small applications to automate tasks. So in this module you will learn the programming basics with Python as one of the most popular programming language. Knowing a programming language will help you stand out among other DevOps engineers.

Module Curriculum 1/2

- ✔ Lesson 01: Introduction to Python: What is Python and why to learn Python as a DevOps engineer?
- ✔ Lesson 02: Installation and Setup Local Development Environment
- ✔ Lesson 03: Write our first Python program
- ✔ Lesson 04: Python IDE vs simple File Editor
- ✔ Lesson 05: Strings and Number Data Types
- ✔ Lesson 06: Variables
- ✔ Lesson 07: Encapsulate Logic with Functions
- ✔ Lesson 08: Accepting User Input
- ✔ Lesson 09: Conditionals (if / else) and Boolean Data Type
- ✔ Lesson 10: Error Handling with Try / Except
- ✔ Lesson 11: While Loops
- ✔ Lesson 12: Lists and For Loops
- ✔ Lesson 13: Comments in Python
- ✔ Lesson 14: Sets

13 Programming with Python

Module Curriculum 2/2

- ✔ Lesson 15: Built-In Functions
- ✔ Lesson 16: Dictionary Data Type
- ✔ Lesson 17: Modularize your project with Modules
- ✔ Lesson 18: Project: Countdown App
- ✔ Lesson 19: Packages, PyPI and pip
- ✔ Lesson 20: Project: Automation with Python (Working with Spreadsheets)
- ✔ Lesson 21: Object Oriented Programming: Classes and Objects
- ✔ Lesson 22: Project: API Request to GitLab

14 Automation with Python

Key Learning Objectives

After having learnt the programming basics and being able to write Python applications, you will now learn how to use this Python knowledge for DevOps use cases and write several automation scripts to get some hands-on real-life experience.

Module Curriculum 1/2

Cloud Automation - AWS & Python

- ✔ Lesson 01: Introduction to Boto (AWS Library for Python)
- ✔ Lesson 02: Install Boto3 and connect to AWS
- ✔ Lesson 03: Getting familiar with Boto Library
  - Automate creating VPC and Subnets
- ✔ Lesson 04: Terraform vs Python - understand the differences and when to use which tool

Automation Tasks around EC2 Instance

- ✔ Lesson 05: Health Check: Automatically check the status of EC2 Instances
- ✔ Lesson 06: Scheduler: Write a scheduled task that executes the status check in a specified interval automatically
- ✔ Lesson 07: Configure Server: Automate adding tags to EC2 Instances with the environment label
- ✔ Lesson 08: Automate getting cluster information from all EKS clusters in your AWS account

14 Automation with Python

Module Curriculum 2/2

Automation Tasks around Data Backup & Restore

- ✔ Lesson 09: Backup EC2 Instances: Automate creating snapshots of EC2 Volumes
- ✔ Lesson 10: Cleanup Task: Write a cleanup script to automate cleanup of old EC2 snapshots
- ✔ Lesson 11: Restore EC2 Volume: Write a program to restore an EC2 volume with the backup snapshot and attach it to the EC2 Instance

Automation Tasks around Website Monitoring (without AWS)

- ✔ Lesson 12: Monitoring: Write a scheduled automation program that monitors the website's health
- ✔ Lesson 13: E-Mail Notification: Configure python program to automatically send an email every time the website or server is down
- ✔ Lesson 14: Recover: Restart the application and reboot the underlying remote server

www.techworld-with-nana.com

15 Configuration Management with Ansible

Key Learning Objectives

As a continuation of the projects in all the previous modules, we will use Ansible to further automate and optimize DevOps processes.
Ansible, next to Terraform is one of the most popular infrastructure as code and configuration management tools currently used in IT projects.

Module Curriculum 1/2

Core Concepts and Syntax of Ansible

- ✔ Lesson 01: Introduction to Ansible
- ✔ Lesson 02: Install & Configure Ansible
- ✔ Lesson 03: Setup Managed Server to configure with Ansible
- ✔ Lesson 04: Ansible Inventory
- ✔ Lesson 05: Ansible ad-hoc Commands
- ✔ Lesson 06: Configure AWS EC2 server with Ansible
- ✔ Lesson 07: Managing Host Key Checking and SSH keys
- ✔ Lesson 08: Ansible Tasks, Play & Playbook
- ✔ Lesson 09: Ansible Modules
- ✔ Lesson 10: Ansible Collections & Ansible Galaxy
- ✔ Lesson 11: Ansible Variables - to make your Playbook customizable
- ✔ Lesson 12: Troubleshooting in Ansible
- ✔ Lesson 13: Conditionals
- ✔ Lesson 14: Privilege Escalation
- ✔ Lesson 15: Ansible Configuration - Default Inventory File

15 Configuration Management with Ansible

Module Curriculum 2/2

Learn most common Ansible modules with hands-on demos:

- ✔ Lesson 16: Project: Deploy Nodejs Application
- ✔ Lesson 17: Project: Deploy Nexus
- ✔ Lesson 18: Configure servers with different Linux distributions on AWS and Digital Ocean platforms

In these projects we will install tools on a server, configure applications, work with a file system, move static files between machines etc.

Essentially you will learn how to map and translate shell scripts and commands into Ansible Playbooks to automate various common tasks in general.

More Advanced Topics & Integrations with other Technologies

- ✔ Lesson 19: Dynamic Inventory for EC2 Servers
- ✔ Lesson 20: Ansible Roles - to make your Ansible content more reusable and modular for better maintenance
- ✔ Lesson 21: Project: Ansible & Terraform
- ✔ Lesson 22: Project: Run Docker applications
- ✔ Lesson 23: Project: Deploying Applications in Kubernetes
- ✔ Lesson 24: Project: Run Ansible from Jenkins Pipeline

So, you don't learn Ansible just as a standalone tool in this bootcamp, but rather integrated in different technologies, like Docker, K8s, Terraform, Jenkins, AWS and so on, in various real world use cases, as it builds on the previous modules in the bootcamp!

16 Monitoring with Prometheus

Key Learning Objectives

Once software is in production, it is important to monitor it to track the performance, discover problems in your infrastructure as well as application and the K8s environment.

In this module you will learn a popular open-source monitoring tool: Prometheus along with the complete monitoring stack, using Alert Manager and Grafana

Module Curriculum

- ✔ Introduction to Monitoring with Prometheus
- ✔ Install Prometheus Stack in Kubernetes
- ✔ Data Visualization with Prometheus UI
- ✔ Introduction to Grafana
- ✔ Alert Rules in Prometheus
- ✔ Create own Alert Rules
- ✔ Introduction to Alertmanager
- ✔ Configure Alertmanager with Email Receiver
- ✔ Trigger Alerts for Email Receiver
- ✔ Monitor Third-Party Applications
- ✔ Deploy Redis Exporter
- ✔ Alert Rules & Grafana Dashboard for Redis
- ✔ Collect & Expose Metrics with Prometheus Client Library
- ✔ Scrape Own Application Metrics & Configure Own Grafana Dashboard

DevOps Certification

Complete with a certification

Upon completion of the DevOps training, you can apply for our official “Certified DevOps Practitioner” digital badge. You will need to submit the demo projects you’ve done throughout the bootcamp.

This digital credential testifies that you have acquired all the skills in the DevOps Bootcamp and are able to work implement complete DevOps processes at work.

It’s verifiable for future employers and can be shared and added to your LinkedIn profile!

A brand of
nnSoftware GmbH
1020 Vienna
AUSTRIA

More infos: www.techworld-with-nana.com

Any Questions?
Email us at support@techworld-with-nana.com

Connect with us