# **CNCF and DevOps Training 2024**

## **Overview**

This is an overview of the training program to introduce you to DevOps, the Cloud, and best practices in the software development lifecycle.

## **Course Outline**

### **Step 1: Understanding the Fundamentals**

* Introduction to the Cloud Native Computing Foundation (CNCF) – Get yourself familiar with the CNCF landscape, discuss its history and role in the cloud industry today.
* Setup your environment
* Install the **tools** you’re going to need: [Docker Desktop](https://docs.docker.com/desktop/install/windows-install/), [Terraform](https://developer.hashicorp.com/terraform/tutorials/aws-get-started/install-cli), [Kubectl](https://kubernetes.io/docs/tasks/tools/), [Minikube](https://minikube.sigs.k8s.io/docs/start/), [Visual Studio Code](https://code.visualstudio.com/download), [Git](https://git-scm.com/book/en/v2/Getting-Started-Installing-Git)
* Create an account on one or more of these Cloud providers: [Microsoft Azure](https://azure.microsoft.com/en-us/free/search), [Amazon Web Services](https://aws.amazon.com/), [Google Cloud Platform](https://cloud.google.com/gcp)
* Create [GitLab](https://about.gitlab.com/), [GitHub](https://github.com/), [DockerHub](https://hub.docker.com/) and [ExamPro](https://www.exampro.co/) accounts
* Once you’re done with the setup, create a **fork** of the Summer Practice Git repository, where you will be taking **notes** of the subjects as you move through them (Notes help you a lot, especially when you revisit the topics; **Tip:** use Markdown (.md) format for your notes to have great readability)
* Start learning the basics of [Linux](https://killercoda.com/pawelpiwosz/course/linuxFundamentals) and [Git](https://killercoda.com/pawelpiwosz/course/gitFundamentals)
* Get a little experience in Vim (Open Git bash, type “vimtutor” and go through the 30-minute tutorial)

### **Step 2: Introduction to Docker**

* Docker and Containers: Understand what a container is, learn how to write Dockerfiles, build images and run containers.
* Killercoda [lab](https://killercoda.com/docker)
* Play with Docker [1](https://labs.play-with-docker.com/) and [2](https://training.play-with-docker.com)
* Write a Dockerfile for an application of yours, build the image, and test it

### **Step 3: Container orchestration – Kubernetes**

* How do you manage your containers when your system grows?
* Go through the [Kubernetes Tutorial for Beginners](https://www.youtube.com/watch?v=X48VuDVv0do&ab_channel=TechWorldwithNana) and remember to take notes along the way
* Deploy your containerized application locally on a [Minikube](https://minikube.sigs.k8s.io/docs/start/) cluster and experiment with Kubernetes objects
* After you can successfully deploy your application locally on Minikube, create a Kubernetes cluster in your Cloud of choice (Azure, AWS, GCP), and deploy your application there
* **Hint:** Use the official [Kubernetes](https://kubernetes.io/docs/home/) documentation when creating your YAML manifests

### **Step 4: Infrastructure as Code – Terraform**

* What is the best way to manage your resources in the Cloud?
* Check out this Terraform [course](https://app.exampro.co/student/journey/terraform) on ExamPro and don’t forget to take notes
* Write a Terraform configuration which creates your Kubernetes cluster in the Cloud
* **Hint:** Use the official Terraform [documentation](https://registry.terraform.io/) and be careful when selecting the SKU (Stock Keeping Unit) size – use a small size for your Kubernetes nodes

### **Step 5: Building a CI/CD Pipeline**

* Read [12 Factor App](https://12factor.net/) and remember to take notes
* Watch this introductory [course](https://www.youtube.com/watch?v=qP8kir2GUgo&ab_channel=TechWorldwithNana) from Lana about GitLab CI/CD
* Create your Continuous Integration/Continuous Deployment pipeline in GitLab; the pipeline will create your Kubernetes cluster using Terraform, then build and push your application as a Docker image to [DockerHub](https://hub.docker.com/), and finally deploy it in your Kubernetes cluster

**Cute challenges for when you’re done**

* Apply your YAML manifests in the Kubernetes cluster using [Helm](https://helm.sh/docs/intro/quickstart/)
* Deploy a second application in the cluster and try to achieve connectivity between them
* Learn about the interesting topic of [GitOps](https://www.youtube.com/watch?v=f5EpcWp0THw&ab_channel=TechWorldwithNana) with Lana and try a demo with [ArgoCD](https://www.youtube.com/watch?v=MeU5_k9ssrs&ab_channel=TechWorldwithNana)

**Additional Reads**

* Continuous Delivery, Jezz Humble (DevOps Bible)