



Microservices, ASP.NET Core, and Docker

Course ID #: NET-948

Hours: 28

Course Content

Course Description:

This Microservices, ASP.NET Core, and Docker training course teaches attendees how to build microservices based on ASP.NET Core and successfully deploy them using Docker containers.

At Course Completion:

After completing this course, student will be able to:

- Learn what Microservices are and how they can be used
- Understand the benefits and challenges that Microservices present
- Learn how to move Monolithic applications to Microservices
- Understand the role of containers and how they can be used with Microservices
- Learn how to work with Docker Community Edition
- Use Docker commands to work with images and containers
- Understand how to create custom Dockerfiles
- Convert Dockerfiles into custom images
- Work with and use Docker volumes
- Create docker-compose.yml files
- Build multiple images using Docker Compose
- Orchestrate multiple Docker containers using Docker Compose
- Work with Docker container registries
- Learn about ASP.NET Core MVC and Web API features
- Understand the role of Entity Framework Core
- Create ASP.NET Core Linux and Windows containers
- Learn Kubernetes concepts
- Orchestrate and monitor microservices

Target Student:

.NET web developers

Prerequisites:

Must have prior experience working with C#. A minimum of 6-months of hands-on C# experience is recommended to get the most out of the course. Prior experience with ASP.NET Web Forms, MVC or Web API is highly recommended.



Microservices, ASP.NET Core, and Docker

Course ID #: NET-948

Hours: 28

Topics:

The Case for Microservices

- Overview of Microservices
- Microservices vs. SOA
- Pros and Cons of Microservices
- Client-side vs. Server-side Routing
- Planning for Microservices
- Microservices Reference Application

Getting Started with Docker

- Application Deployment
- Shipping with Containers
- Getting Started with Docker
- Benefits to Developers and DevOps
- Overview of Images and Containers
- Layered File System
- Containers vs Virtual Machines
- Docker and Microservices

Docker Images and Containers

- Container Registries
- Working with Images
- Working with Containers
- Getting Started with a Custom Dockerfile
- Building an Image
- Updating an Image
- Pushing an Image to a Registry

Containers and Volumes

- Components Overview
- Source Code and Containers
- Introduction to Volumes
- Creating a Volume
- Inspecting Volumes
- Defining a volume in a Dockerfile
- Local Source Code and Containers

Orchestrating Containers with Docker Compose

- Container Orchestration
- Getting Started with Docker Compose
- The docker-compose.yml File
- Docker Compose Commands

Getting Started with ASP.NET Core

- Getting Started with ASP.NET Core
- Configuration and Middleware Pipeline
- Controllers and Views
- RESTful Services with Web API
- Data Access with Entity Framework Core
- OS X Cross Platform Development

ASP.NET Core and Docker

- ASP.NET Core Docker Images
- Creating ASP.NET Core Linux Images
- Creating ASP.NET Core Windows Images
- Visual Studio Docker Features
- "Containerizing" .NET Framework Applications
- Running and Debugging Containers

Monolithic to Microservices

- Monolithic Applications Review
- Microservices Review
- Communicating with Microservices
- Converting a Monolithic Feature to a Microservice
- Creating Dockerfiles and Images
- Communication with Microservices
- Fault Handling
- Health Checks

Microservices Orchestration and Monitoring

Tools Overview

- Docker Swarm
- Kubernetes
- Marathon
- cAdvisor
- SysDig

(Bonus) Introduction to Kubernetes

- Why Kubernetes
- Kubernetes Overview
- The Master
- Nodes and Pods
- Services and Deployments
- Files and Commands