AZ-400: Designing and Implementing Microsoft DevOps Solutions [AZ-400T00-A] training including Exam Voucher

Group Training

Training code	CGAAZ400CE
Spoken Language	English
Language Materials	English
Dayparts	10
Price	€2.550,00 excl. VAT No extra costs.

What is AZ-400: Designing and Implementing Microsoft DevOps Solutions [AZ-400T00-A] including Exam Voucher

This course provides the knowledge and skills to design and implement DevOps processes and practices. Students will learn how to plan for DevOps, use source control, scale Git for an enterprise, consolidate artifacts, design a dependency management strategy, manage secrets, implement continuous integration, implement a container build strategy, design a release strategy, set up a release management workflow, implement a deployment pattern, and optimize feedback mechanisms.
This course uses MOC (Microsoft Official Courseware) and will be given by an experienced MCT (Microsoft Certified Trainer).
See the below modules for more information:
Module 1: Planning for DevOps
Lessons

- Transformation Planning
- Project Selection
- Team Structures
- Migrating to Azure DevOps

Module 2: Getting Started with Source Control

Lessons

- What is Source Control
- Benefits of Source Control
- Types of Source Control Systems
- Introduction to Azure Repos
- Introduction to GitHub
- Migrating from Team Foundation Version Control (TFVC) to Git in Azure Repos

Module 3: Managing Technical Debt

Lessons

- Identifying Technical Debt
- Knowledge Sharing within Teams
- Modernizing Development Environments with Codespaces

Module 4: Working with Git for Enterprise DevOps

Lessons

- How to Structure Your Git Repo
- Git Branching Workflows
- Collaborating with Pull Requests in Azure Repos
- Why Care About Git Hooks
- Fostering Inner Source
- Managing Git Repositories

Module 5: Configuring Azure Pipelines

Lessons

- The Concept of Pipelines in DevOps
- Azure Pipelines
- Evaluate use of Hosted versus Self-Hosted Agents
- Agent Pools
- Pipelines and Concurrency
- Azure DevOps and Open-Source Projects (Public Projects)
- Azure Pipelines YAML versus Visual Designer

Module 6: Implementing Continuous Integration using Azure Pipelines

Lessons

- Continuous Integration Overview
- Implementing a Build Strategy
- Integration with Azure Pipelines
- Integrating External Source Control with Azure Pipelines
- Set Up Self-Hosted Agents

Module 7: Managing Application Configuration and Secrets

Lessons

- Introduction to Security
- Implement a Secure Development Process
- Rethinking Application Configuration Data
- Manage Secrets, Tokens, and Certificates
- Integrating with Identity Management Systems
- Implementing Application Configuration

Module 8: Implementing Continuous Integration with GitHub Actions

Lessons

- GitHub Actions
- Continuous Integration with GitHub Actions
- Securing Secrets for GitHub Actions

Module 9: Designing and Implementing a Dependency Management Strategy

Lessons

- Packaging Dependencies
- Package Management
- Migrating and Consolidating Artifacts
- Package Security
- Implementing a Versioning Strategy

Module 10: Designing a Release Strategy

Lessons

- Introduction to Continuous Delivery
- Release Strategy Recommendations
- Building a High-Quality Release pipeline
- Choosing the Right Release Management Tool

Module 11: Implementing Continuous Deployment using Azure Pipelines

Lessons

- Create a Release Pipeline
- Provision and Configure Environments
- Manage and Modularize Tasks and Templates
- Configure Automated Integration and Functional Test Automation
- Automate Inspection of Health

Module 12: Implementing an Appropriate Deployment Pattern

Lessons

- Introduction to Deployment Patterns
- Implement Blue Green Deployment
- Feature Toggles
- Canary Releases
- Dark Launching
- AB Testing
- Progressive Exposure Deployment

Module 13: Managing Infrastructure and Configuration using Azure Tools

Lessons

- Infrastructure as Code and Configuration Management
- Create Azure Resources using ARM Templates
- Create Azure Resources using Azure CLI
- Azure Automation with DevOps
- Desired State Configuration (DSC)

Module 14: Third Party Infrastructure as Code Tools Available with Azure

Lessons

- Chef
- Puppet
- Ansible
- Terraform

Module 15: Managing Containers using Docker

Lessons

- Implementing a Container Build Strategy
- Implementing Docker Multi-Stage Builds

Module 16: Creating and Managing Kubernetes Service Infrastructure

Lessons

- Azure Kubernetes Service
- Kubernetes Tooling
- Integrating AKS with Pipelines

Module 17: Implementing Feedback for Development Teams

Lessons

- Implement Tools to Track System Usage, Feature Usage, and Flow
- Implement Routing for Mobile Application Crash Report Data
- Develop Monitoring and Status Dashboards
- Integrate and Configure Ticketing Systems

Module 18: Implementing System Feedback Mechanisms

Lessons

- Site Reliability Engineering
- Design Practices to Measure End-User Satisfaction+L120
- Design Processes to Capture and Analyze User Feedback
- Design Processes to Automate Application Analytics
- Managing Alerts
- Blameless Retrospectives and a Just Culture

Module 19: Implementing Security in DevOps Projects

Lessons

- Security in the Pipeline
- Azure Security Center

Module 20: Validating Code Bases for Compliance

Lessons

- Open-Source Software
- Managing Security and Compliance Policies
- Integrating License and Vulnerability Scans

Who should attend the AZ-400: Designing and Implementing Microsoft DevOps Solutions [AZ-400T00-A] including Exam Voucher

A candidate for this certification must be familiar with both Azure administration and

development and must be expert in at least one of these areas. Interested in designing and implementing DevOps processes or in passing the Microsoft Azure DevOps Solutions certification exam.
Responsibilities for this role include designing and implementing strategies for collaboration, code, infrastructure, source control, security, compliance, continuous integration, testing, delivery, monitoring, and feedback.
Also, you will receive an Exam Voucher. Enlist today!

Prerequisites

Successful learners will have prior knowledge and understanding of:
Cloud computing concepts, including an understanding of PaaS, SaaS, and IaaS implementations.

- Both Azure administration and Azure development with proven expertise in at least one of these areas.
- Version control, Agile software development, and core software development principles. It would be helpful to have experience in an organization that delivers software.
- If you are new to Azure and cloud computing, consider AZ-900: Azure Fundamentals training.
- If you are new to Azure Administration, consider AZ-104: Microsoft Azure Administrator training.
- If you are new to Azure Developer, consider AZ-204: Developing Solutions for Microsoft Azure training.

Objectives

After completing this course, you will be able to:
Identify project metrics and Key Performance Indicators (KPI's)

- Design a tool integration strategy and a license management strategy (e.g. Azure DevOps and GitHub users)
- Design a strategy for end-to-end traceability from work items to working software
- Design an authentication and access strategy and a strategy for integrating on-premises and cloud resources
- Describe the benefits of using Source Control and Azure Repos and GitHub
- Migrate from TFVC to Git
- Manage code quality including technical debt SonarCloud, and other tooling solutions
- How to structure Git repos and branching workflows
- Leverage pull requests for collaboration and code reviews and Git hooks for automation
- Explain the role of Azure Pipelines and its components
- Implement continuous integration using Azure Pipelines
- Manage application configuration and secrets
- Create and work with GitHub Actions and Workflows
- Implement Blue Green Deployment, Canary Release and Progressive Exposure Deployment
- Deploy and manage infrastructure using Microsoft automation technologies such as ARM templates, PowerShell, and Azure CLI
- Deploy and configure infrastructure using 3rd party tools and services with Azure, such as Chef, Puppet, Ansible, and Terraform
- Implement a container strategy including how containers are different from virtual machines and how microservices use containers
- Implement containers using Docker and Docker multi-stage builds

- Deploy and configure a Managed Kubernetes cluster
- Implement tools to track system usage, feature usage, and flow
- Configure crash report integration for client applications
- Define Site Reliability Engineering
- Design processes to measure end-user satisfaction and automate application analytics
- Define an infrastructure and configuration strategy and appropriate toolset for a release pipeline and application infrastructure

---

About Capgemini Academy

Capgemini Academy's professionals offer what people in IT need. Our professionals have a keen eye for motivation, talent and are aware of specific contexts and circumstances. They move people to move. Programmes and courses that originate from daily experience of our both didactical and substantively strong trainers, light a fire within the individual IT professionals.
Real life stories of our professionals' experience that tell how to solve problems and work with the people around it, do the rest.

An organization, like ours, helps people and their organizations day by day to get the best out of themselves and each other. We prepare them to defy tomorrow's challenges. We stimulate learning and curiosity. In order for individual IT professionals and their employers, to build better, longer and more intensive relationships. For mutual benefit.

Capgemini Academy. We transform IT professionals
academy.capgemini.nl

IN/3A-018.18