COURSE OUTLINE

Silver
Microsoft
Partner

AZ-400T00

Designing and Implementing Microsoft DevOps solutions

DURATION	LEVEL	TECHNOLOGY		DELIVERY METHOD	TRAINING CREDITS
4 Days	Advanced	Microsoft	Azure	ILT/VILT	N/A

Overview

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.

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.

Target Audience

Students in this course are interested in designing and implementing DevOps processes or in passing the Microsoft Azure DevOps Solutions certification exam.

Job role

DevOps Engineer

Exam Requirements

AZ-400

If you are new to Azure and cloud computing, consider:

- AZ-900: Azure Fundamentals

Prerequisite courses:

- AZ-104: Azure Administrator
  OR
- AZ-204: Azure Developer

Course Topics

- Introduction to DevOps
- Choose the right project
- Describe team structures
- Choose the DevOps tools
- Plan Agile with GitHub Projects and Azure Boards
- Introduction to source control
- Describe types of source control systems
- Work with Azure Repos and GitHub
- Structure your Git Repo
- Manage Git branches and workflows
- Collaborate with pull requests in Azure Repos
- Identify technical debt
- Explore Git hooks
- Plan foster inner source
- Manage Git repositories
- Explore Azure Pipelines
- Manage Azure Pipeline agents and pools
- Describe pipelines and concurrency
- Explore continuous integration
- Implement a pipeline strategy
- Integrate with Azure Pipelines
- Introduction to GitHub Actions
- Learn continuous integration with GitHub Actions
- Design a container build strategy
- Introduction to continuous delivery
- Create a release pipeline
- Explore release recommendations
- Provision and test environments
- Manage and modularize tasks and templates
- Automate inspection of health
- Introduction to deployment patterns
- Implement blue-green deployment and feature toggles
- Implement canary releases and dark launching
- Implement A/B testing and progressive exposure deployment
- Integrate with identity management systems
- Manage application configuration data
- Explore infrastructure as code and configuration management
- Create Azure resources using Azure Resource Manager templates
- Create Azure resources by using Azure CLI
- Explore Azure Automation with DevOps
- Implement Desired State Configuration (DSC)

- Implement Bicep
- Introduction to Secure DevOps
- Implement open-source software
- Software Composition Analysis
- Static analyzers
- OWASP and Dynamic Analyzers
- Security Monitoring and Governance
- Explore package dependencies
- Understand package management
- Migrate consolidating and secure artifacts
- Implement a versioning strategy
- Introduction to GitHub Packages
- Implement tools to track usage and flow
- Develop monitor and status dashboards
- Share knowledge within teams
- Design processes to automate application analytics
- Manage alerts, blameless retrospectives and a just culture