



AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 35

Course Content

Course Description:

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.

At Course Completion:

After completing this course, student will be able to:

- Plan for the transformation with shared goals and timelines
- Select a project and identify project metrics and KPIs
- Create a team and agile organization structure

Prerequisites:

Fundamental knowledge about Azure, version control, Agile software development, and core software development principles. It would be helpful to have experience in an organization that delivers software.

Target Student:

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

Topics:

Module 1: Planning for DevOps

- Transformation Planning
- Project Selection
- Team Structures
- Migrating to Azure DevOps
- Lab : Agile Planning and Portfolio Management with Azure Boards

Module 2: Getting started with Source Control

- 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
- Authenticating to Git in Azure Repos



AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 35

- Lab : Version Controlling with Git

Module 3: Scaling Git for enterprise DevOps

- How to Structure your Git Repo
- Git Branching Workflows
- Collaborating with Pull Requests in Azure Repos
- Why care about GitHooks
- Fostering Inner Source
- Lab : Code Review with Pull Requests

Module 4: Consolidating Artifacts & Designing a Dependency Management Strategy

- Packaging Dependencies
- Package Management
- Migrating and Consolidating Artifacts
- Lab : Updating Packages

Module 5: Implementing Continuous Integration with Azure Pipelines

- The concept of pipelines in DevOps
- Azure Pipelines
- Evaluate use of Hosted vs Private Agents
- Agent Pools
- Pipelines and Concurrency
- Azure DevOps and Open Source Projects (Public Projects)
- Azure Pipelines YAML vs Visual Designer
- Continuous Integration Overview
- Implementing a Build Strategy
- Integration with Azure Pipelines
- Integrate External Source Control with Azure Pipelines
- Set Up Private Agents
- Analyze and Integrate Docker Multi-Stage Builds
- Lab : Enabling Continuous Integration with Azure Pipelines
- Lab : Integrating External Source Control with Azure Pipelines

Module 6: Managing Application Config and Secrets

- Introduction to Security
- Implement secure and compliant development process
- Rethinking application config data
- Manage secrets, tokens, and certificates
- Implement tools for managing security and compliance in a pipeline
- Lab : Integrating Azure Key Vault with Azure DevOps

Module 7: Managing Code Quality and Security Policies

- Managing Code Quality
- Managing Security Policies
- Lab : Managing Technical Debt with Azure DevOps and SonarCloud

Module 8: Implementing a Container Build Strategy

- Implementing a Container Build Strategy
- Lab : Modernizing Existing ASP.NET Apps with Azure

Module 9: Manage Artifact versioning, security & compliance

- Package security
- Open source software
- Integrating license and vulnerability scans
- Implement a versioning strategy
- Lab : Manage Open Source Security and License with WhiteSource



AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 35

Module 10: Design a Release Strategy

- Introduction to Continuous Delivery
- Release strategy recommendations
- Building a High-Quality Release pipeline
- Choosing a deployment pattern
- Choosing the right release management tool

Module 11: Set up a Release Management Workflow

- Create a Release Pipeline
- Provision and Configure Environments
- Manage and Modularize Tasks and Templates
- Integrate Secrets with the release pipeline
- Configure Automated Integration and Functional Test Automation
- Automate Inspection of Health
- Lab : Configuring Pipelines as Code with YAML
- Lab : Setting up secrets in the pipeline with Azure Key vault
- Lab : Setting up and Running Functional Tests
- Lab : Using Azure Monitor as release gate
- Lab : Creating a release Dashboard

Module 12: Implement an appropriate deployment pattern

- Introduction to Deployment Patterns
- Implement Blue Green Deployment
- Feature Toggles
- Canary Releases
- Dark Launching
- AB Testing
- Progressive Exposure Deployment
- Lab : Feature Flag Management with LaunchDarkly and Azure DevOps

Module 13: Implement process for routing system feedback to development teams

- 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
- Lab : Monitoring Application Performance

Module 14: Infrastructure and Configuration Azure Tools

- Infrastructure as Code and Configuration Management
- Create Azure Resources using ARM Templates
- Create Azure Resources using Azure CLI
- Create Azure Resources by using Azure PowerShell
- Desired State Configuration (DSC)
- Azure Automation with DevOps
- Additional Automation Tools
- Lab : Azure Deployments using Resource Manager Templates

Module 15: Azure Deployment Models and Services

- Deployment Modules and Options
- Azure Infrastructure-as-a-Service (IaaS) Services
- Azure Platform-as-a-Service (PaaS) services
- Serverless and HPC Computer Services
- Azure Service Fabric
- Lab : Deploying a Dockerized Java app to Azure Web App for Containers



AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 35

Module 16: Create and Manage Kubernetes

Service Infrastructure

- Azure Kubernetes Service
- Lab : Deploying a multi-container application to Azure Kubernetes Service

Module 17: Third Party Infrastructure as Code

Tools available with Azure

- Chef
- Puppet
- Ansible
- Terraform
- Lab : Infrastructure as Code
- Lab : Automating Your Infrastructure Deployments in the Cloud with Terraform and Azure Pipelines

Module 18: Implement Compliance and Security in your Infrastructure

- Security and Compliance Principles with DevOps
- Azure security Center
- Lab : Implement Security and Compliance in an Azure DevOps Pipeline

Module 19: Recommend and design system feedback mechanisms

- The inner loop
- Continuous Experimentation mindset
- Design practices to measure end-user satisfaction
- Design processes to capture and analyze user feedback
- Design process to automate application analytics
- Lab : Integration between Azure DevOps and Teams

Module 20: Optimize feedback mechanisms

- Site Reliability Engineering
- Analyze telemetry to establish a baseline
- Perform ongoing tuning to reduce meaningless or non-actionable alerts
- Analyze alerts to establish a baseline
- Blameless Retrospectives and a Just Culture