AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 28

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.

Course Objectives:
Upon successful completion of this course, students will be able to:
- Understand what DevOps is and the steps to accomplish it
- Understand different projects and systems to guide the journey
- Design a tool integration strategy
- Plan for continuous monitoring
- Work with Azure DevOps Wikis
- Implement monitoring tools and technologies
- Carry out blameless retrospectives and create a just culture
- And so much more!

Prerequisites:
- 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 interested in designing and implementing DevOps processes or in passing the Microsoft Azure DevOps Solutions certification exam.

AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

Course ID #: 7000-389-ZZ-Z

Hours: 28

Topics:

Lesson 1: Introduction to DevOps
- Introduction
- What is DevOps?
- Explore the DevOps journey
- Identify transformation teams
- Explore shared goals and define timelines
- Knowledge check
- Summary

Lesson 2: Choose the right project
- Introduction
- Explore greenfield and brownfield projects
- Decide when to use greenfield and brownfield projects
- Decide when to use systems of record versus systems of engagement
- Identify groups to minimize initial resistance
- Identify project metrics and key performance indicators (KPIs)
- Knowledge check
- Summary

Lesson 3: Describe team structures
- Introduction
- Explore agile development practices
- Explore principles of agile development
- Define organization structure for agile practices
- Explore ideal DevOps team members
- Enable in-team and cross-team collaboration
- Select tools and processes for agile practices
- Knowledge check
- Summary

Lesson 4: Choose DevOps tools
- Introduction
- What is Azure DevOps?
- What is GitHub?
- Explore an authorization and access strategy
- Migrate or integrate existing work management tools
- Migrate or integrate existing test management tools
- Design a license management strategy
- Knowledge check
- Summary

Lesson 5: Plan Agile with GitHub Projects and Azure Boards
- Introduction
- Introduction to GitHub Projects and Project boards
- Introduction to Azure Boards
- Link GitHub to Azure Boards
- Configure GitHub Projects
- Manage work with GitHub Project boards
- Customize Project views
- Collaborate using team discussions
- Agile plan and portfolio management with Azure Boards
- Knowledge check
- Summary

Lesson 6: Introduction to source control
- Introduction
- Explore DevOps foundational practices
- What is source control?
- Explore benefits of source control
- Explore best practices for source control
- Knowledge check
- Summary

Lesson 7: Describe types of source control systems
- Introduction
- Understand centralized source control
- Understand distributed source control
- Explore Git and Team Foundation Version Control
- Examine and choose Git
- Describe working with Git locally
- Knowledge check
- Summary

Lesson 8: Work with Azure Repos and GitHub
- Introduction
- Introduction to Azure Repos
- Introduction to GitHub
- Migrate from TFVC to Git
- Develop online with GitHub Codespaces
- Version Control with Git in Azure Repos
- Knowledge check
- Summary

Lesson 9: Structure your Git Repo
- Introduction
- Explore monorepo versus multiple repos
- Implement a change log
- Knowledge check
- Summary

Lesson 10: Manage Git branches and workflows
- Introduction
- Explore branch workflow types
- Explore feature branch workflow
- Explore Git branch model for continuous delivery
- Explore GitHub flow
- Explore fork workflow
- Version control with Git in Azure repos
- Knowledge check
- Summary

Lesson 11: Collaborate with pull requests in Azure Repos
- Introduction
- Collaborate with pull requests
- Exercise – Azure Repos collaborating with pull requests
- Examine GitHub mobile for pull request approvals
- Knowledge check
- Summary

Lesson 12: Identify technical debt
- Introduction
- Examine code quality
- Examine complexity and quality metrics
- Introduction to technical debt
- Measure and manage technical debt
- Integrate other code quality tools
- Plan effective code reviews
- Knowledge check
- Summary

Lesson 13: Explore Git hooks
- Introduction
- Introduction to Git hooks
- Implement Git hooks
- Knowledge check
- Summary

Lesson 14: Plan foster inner source
- Introduction
- Explore foster inner source
- Implement the fork workflow
- Describe inner source with forks
- Knowledge check
- Summary

Lesson 15: Manage Git repositories
- Introduction
- Work with large repositories
- Pure repository data
- Manage releases with GitHub Repos
- Automate release notes with GitHub
- Knowledge check
- Summary

Lesson 16: Explore Azure Pipelines
- Introduction
- Explore the concept of pipelines in DevOps
- Describe Azure Pipelines
- Understand Azure Pipelines key terms
- Knowledge check
- Summary

Lesson 17: Manage Azure Pipeline agents and pools
- Introduction
- Choose between Microsoft-hosted versus self-hosted agents
- Explore job types
- Introduction to agent pools
- Explore predefined agent pool
- Understand typical situations for agent pools
- Communicate with Azure Pipelines
- Communicate to deploy to target servers
- Examine other considerations
- Describe security of agent pools
- Configure agent pools and understanding pipeline styles
- Knowledge check
- Summary

Lesson 18: Describe pipelines and concurrency
- Introduction
- Understand parallel jobs
- Estimate parallel jobs
- Describe Azure Pipelines and open-source projects
- Explore Azure Pipelines and Visual Designer
- Describe Azure Pipelines and YAML
- Knowledge check
- Summary

Lesson 19: Explore continuous integration
- Introduction
- Introduction to continuous integration
- Learn the four pillars of continuous integration
- Explore the benefits of continuous integration
- Describe build properties
- Enable Continuous Integration with Azure Pipelines
- Knowledge check
- Summary

Lesson 20: Implement a pipeline strategy
- Introduction
- Configure agent demands
- Implement multi-agent builds
- Explore source control types supported by Azure Pipelines
- Knowledge check
- Summary

Lesson 21: Integrate with Azure Pipelines
- Introduction
- Describe the anatomy of a pipeline
- Understand the pipeline structure
- Details templates
- Explore YAML resources
- Use multiple repositories in your pipeline
- Integrate external source control with Azure Pipelines
- Knowledge check
- Summary

Lesson 22: Introduction to GitHub Actions
- Introduction
- What are Actions?
- Explore Actions flow
- Understand workflows
- Describe standard workflow syntax elements
- Explore events
- Explore jobs
- Explore runners
- Examine release and test an action
- Knowledge check
- Summary

Lesson 23: Learn continuous integration with GitHub Actions
- Introduction
- Describe continuous integration with actions
- Examine environment variables
- Share artifacts between jobs
- Examine Workflow badges
- Describe best practices for creating actions
- Mark releases with Git tags
- Create encrypted secrets
- Use secrets in a workflow
- Implement GitHub Actions for CI/CD
- Knowledge check
- Summary

Lesson 24: Design a container build strategy
- Introduction
- Examine structure of containers
- Work with Docker containers
- Understand Dockerfile core concepts
- Examine multi-stage Dockerfiles
- Examine considerations for multiple stage builds
- Explore Azure container-related services
- Deploy Docker containers to Azure App Service web apps
- Knowledge check
- Summary

Lesson 25: Introduction to continuous delivery
- Introduction
- Explore traditional IT development cycle
- What is continuous delivery?
- Move to continuous delivery
- Understand releases and deployments
- Understand release process versus release
- Knowledge check
- Summary

Lesson 26: Create a release pipeline
- Introduction
- Describe Azure DevOps release pipeline capabilities
- Explore release pipelines
- Explore artifact sources
- Choose the appropriate artifact source
- Exercise – Select an artifact source
- Examine considerations for deployment to stages
- Exercise – Set up stages
- Explore build and release tasks
- Explore custom build and release tasks
- Explore release jobs
- Configure Pipelines as Code with YAML
- Knowledge check
- Summary

Lesson 27: Explore release recommendations
- Introduction
- Understand the delivery cadence and three types of triggers
- Exercise – Select your delivery and deployment cadence
- Explore release approvals
- Exercise – Set up manual approvals
- Explore release gates
- Use release gates to protect quality
- Control deployments using release gates
- Knowledge check
- Summary

Lesson 28: Provision and test environments
- Introduction
- Provision and configure target environments
- Exercise – Set up service connections
- Configure automated integration and functional test automation
- Understand Shift-left
- Set up and run availability tests
- Explore Azure Load Testing
- Set up and run functional tests
- Knowledge check
- Summary

Lesson 29: Manage and modularize tasks and templates
- Introduction
- Examine tasks groups
- Exercise – create and manage task groups
- Explore variables in release pipelines
- Understand variable groups
- Exercise – create and manage variable groups
- Knowledge check
- Summary

Lesson 30: Automate inspection of health
- Introduction
- Automate inspection of health
- Explore events and notifications
- Explore service hooks
- Exercise – Set up service hooks and monitor the pipeline
- Configure Azure DevOps notifications
- Configure GitHub notifications
- Explore how to measure quality of your release process
- Examine release notes and documentation
- Examine considerations for choosing release management tools
- Explore common release management tools
- Create a release dashboard
- Knowledge check
- Summary

Lesson 31: Introduction to deployment patterns
- Introduction
- Explore microservices architecture
- Examine classical deployment patterns
- Understand modern deployment patterns
- Knowledge check
- Summary

Lesson 32: Implement blue-green deployment and feature toggles
- Introduction
- What is blue-green deployment?
- Explore deployment slots
- Exercise – set up a blue-green deployment
- Introduction to feature toggles
- Describe feature toggle maintenance
- Knowledge check
- Summary

Lesson 33: Implement canary releases and dark launching
- Introduction
- Explore canary releases
- Examine Traffic Manager
- Understand dark launching
- Knowledge check
- Summary

Lesson 34: Implement A/B testing and progressive exposure deployment
- Introduction
- What is A/B testing?
- Explore CI-CD with deployment rings
- Exercise – Ring-based deployment
- Knowledge check
- Summary

Lesson 35: Integrate with identity management systems
- Introduction
- Integrate GitHub with single sign-on (SSO)
- Explore service principals
- Explore Managed Identity
- Knowledge check
- Summary

Lesson 36: Manage application configuration management
- Introduction
- Rethink application configuration data
- Explore separation of concerns
- Understand external configuration store patterns
- Introduction to Azure App Configuration
- Examine Key-value pairs
- Examine App configuration feature management
- Integrate Azure Key Vault with Azure Pipelines
- Manage secrets, tokens and certificates
- Examine DevOps inner and outer loop

- Integrate Azure Key Vault with Azure DevOps
- Enable Dynamic Configuration Feature Flags
- Knowledge check
- Summary

Lesson 37: Explore infrastructure as code and configuration management
- Introduction
- Explore environment deployment
- Examine environment configuration
- Understand imperative versus declarative configuration
- Understand idempotent configuration
- Knowledge check
- Summary

Lesson 38: Create Azure resources using Azure Resource Manager templates
- Introduction
- Why use Azure Resource Manager templates?
- Explore template components
- Manage dependencies
- Modularize templates
- Manage secrets in templates
- Azure deployments using Resource Manager templates
- Knowledge check
- Summary

Lesson 39: Create Azure resources by using Azure CLI
- Introduction
- What is Azure CLI?
- Work with Azure CLI
- Exercise – Run templates using Azure CLI
- Knowledge check
- Summary

Lesson 40: Explore Azure Automation with DevOps
- Introduction
- Create automation accounts
- What is a runbook?
- Understand automation shared resources
- Explore runbook gallery
- Examine webhooks
- Explore source control integration
- Explore PowerShell workflows
- Create a workflow
- Exercise – Create and run a workflow runbook
- Examine checkpoint and parallel processing
- Knowledge check
- Summary

Lesson 41: Implement Desired State Configuration (DSC)
- Introduction
- Understand configuration drift
- Explore Desired State Configuration (DSC)
- Explore Azure Automation State configuration (DSC)
- Examine DSC configuration file
- Exercise – Import and compile
- Exercise – Onboard machines for management
- Explore hybrid management
- Implement DSC and Linux Automation on Azure
- Knowledge check
- Summary

Lesson 42: Implement Bicep
- Introduction
- What is Bicep?
- Install Bicep?
- Exercise – Create Bicep templates
- Understand Bicep file structure and syntax
- Exercise – Deploy a Bicep file from Azure Pipelines
- Exercise – Deploy a Bicep file from GitHub workflows
- Knowledge check
- Summary

Lesson 43: Introduction to Secure DevOps
- Introduction
- Describe SQL injection attack
- Understand DevSecOps
- Explore Secure DevOps Pipeline
- Explore key validation points
- Explore continuous security validation
- Understand threat modeling
- Exercise – threat modeling
- Knowledge check
- Summary

Lesson 44: Implement open-source software
- Introduction
- Explore how software is built
- What is open-source software
- Explore corporate concerns with open-source software components
- Introduction to open-source licenses
- Explore common open-source licenses
- Examine license implications and ratings
- Knowledge check
- Summary

Lesson 45: Software Composition Analysis
- Introduction
- Inspect and validate code bases for compliance
- Explore software composition analysis
- Integrate WhiteSource with Azure DevOps pipeline
- Implement GitHub Dependabot alerts and security updates
- Integrate software composition analysis checks into pipelines
- Examine tools for assess package security and license rate
- Interpret alerts from scanner tools
- Implement Security and Compliance in an Azure Pipeline
- Knowledge check
- Summary

Lesson 46: Static analyzers
- Introduction
- Explore SonarCloud
- Explore CodeQL in GitHub
- Manage technical debt with SonarCloud and Azure DevOps
- Knowledge check
- Summary

Lesson 47: OWASP and Dynamic Analyzers
- Introduction
- Plan Implement OWASP Secure Coding Practices
- Explore OWASP ZAP penetration test
- Explore OWASP ZAP results and bugs
- Knowledge check
- Summary

Lesson 48: Security Monitoring and Governance
- Introduction
- Implement pipeline security
- Explore Microsoft Defender for Cloud
- Examine Microsoft Defender for Cloud usage scenarios
- Explore Azure Policy
- Understand policies
- Explore initiatives
- Explore resource locks
- Explore Azure Blueprints
- Understand Microsoft Defender for Identity
- Knowledge check
- Summary

Lesson 49: Explore package dependencies
- Introduction
- What is dependency management?
- Describe elements of a dependency management strategy
- Identify dependencies
- Understand source and package componentization
- Decompose your system
- Scan your codebase for dependencies
- Knowledge check
- Summary

Lesson 50: Understand package management
- Introduction
- Explore packages
- Understand package feeds
- Explore package and feed managers
- Explore common public package sources
- Explore self-hosted and SaaS based package sources
- Consume packages
- Introduction to Azure Artifacts
- Publish packages
- Package Management with Azure Artifacts
- Knowledge check
- Summary

Lesson 51: Migrate consolidating and secure artifacts
- Introduction
- Identify existing artifact repositories
- Migrate and integrating artifact repositories
- Secure access to package and feeds
- Examine roles
- Examine permissions
- Examine authentication
- Knowledge check
- Summary

Lesson 52: Implement a versioning strategy
- Introduction
- Understand versioning of artifacts
- Explore semantic versioning
- Examine release views
- Promote packages
- Exercise – Promote a package
- Explore best practices for versioning
- Exercise – Push from the pipeline
- Knowledge check
- Summary

Lesson 53: Introduction to GitHub Packages
- Introduction
- Publish packages
- Install a package
- Delete and restore a package
- Explore package access control and visibility
- Knowledge check
- Summary

Lesson 54: Implement tools to track usage and flow
- Introduction
- Understand the inner loop
- Introduction to continuous monitoring
- Explore Azure Monitor and Log Analytics
- Examine Kusto Query Language (KQL)
- Explore Application Insights
- Implement Application Insights
- Exercise – Add Application Insights to an ASP.NET core application
- Monitor application performance with Application Insights
- Knowledge check
- Summary

Lesson 55: Develop monitor and status dashboards
- Introduction
- Explore Azure Dashboards
- Examine view designer in Azure Monitor
- Explore Azure Monitor workbooks
- Explore Power BI
- Build your own custom application
- Knowledge check
- Summary

Lesson 56: Share knowledge within teams
- Introduction
- Share acquired knowledge within development teams
- Introduction to Azure DevOps project wikis
- Integrate with Azure Boards
- Integrate Azure DevOps and Teams
- Share team knowledge using Azure Project Wiki
- Knowledge check
- Summary

Lesson 57: Design processes to automate application analytics
- Introduction
- Explore rapid responses and augmented search
- Integrate telemetry
- Examine monitoring tools and technologies
- Knowledge check
- Summary

Lesson 58: Manage alerts, blameless retrospectives and a just culture
- Introduction
- Examine when get a notification
- Explore how to fix it
- Explore smart detection notifications
- Improve performance
- Understand server response time degradation
- Reduce meaningless and non-actionable alerts
- Examine blameless retrospective
- Develop a just culture
- Knowledge check
- Summary

Register for this class by visiting us at:
www.tcworkshop.com or calling us at 800-639-3535