AZ-400: Designing and Implementing Microsoft DevOps solutions

Microsoft - Azure Apps & Infrastructure

Live Training (também disponível em presencial)

- **Localidade:**
- **Data:** 13 Jul 2026
- **Preço:** 1310 € (Os valores apresentados não incluem IVA. Oferta de IVA a particulares e estudantes.)
- **Horário:** Laboral das 9h30 às 17h30
- **Nível:** Avançado
- **Duração:** 40h

Sobre o curso

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

---

Destinatários

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

Objetivos

- Prepare for Exam AZ-400: Designing and Implementing Microsoft DevOps Solution
- Examine the key Git features that organizations must plan for when designing their enterprise DevOps
- Introduce the continuous integration concept using Azure Pipelines and GitHub Actions and provides instruction on configuring those services and building applications
- Explain the concepts of continuous delivery and releases strategy considerations for setting up deployment stages and your delivery and deployment cadence, and lastly, setting up your release approvals
- Implement alerting mechanisms, report on your quality, and get notified by using service hooks
- Explore the “Infrastructure as Code” (IaC) concept and how to manage your operations environment the

same way you do applications or other code for general release
- Examine dependency management in software development, how to identify them in your codebase and how to package and manage dependencies in package feeds
- Introduce the continuous feedback concept and describes how to implement it in your DevOps cycle
- Explore an infrastructure and configuration strategy and appropriate toolset for a release pipeline and application infrastructure

---

Condições

Para particulares

- 10% do valor total pago no ato da inscrição; restante valor até 7 dias antes do início do curso.
- Formandos não residentes em Portugal: pagamento de 50% no ato da inscrição.
- Possibilidade de pagamento em até 12 prestações mensais sem juros via Cofidis Pay (até 2.500€, sujeito a aprovação).
- Possibilidade de beneficiar do Cheque Formação+Digital até 750€ (conforme elegibilidade).
- Isenção de IVA para particulares.

Para empresas

- Empresas nacionais: pagamento a 30 dias, contra fatura (acresce IVA à taxa legal em vigor).
- Empresas da UE e fora da UE: valores isentos de IVA e pagamento a pronto.

Pré-requisitos

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 taking:

- Instructor-led course: AZ-900: Azure Fundamentals

If you are new to Azure Administration, consider taking:

- Instructor-led courses: AZ-104: Microsoft Azure Administrator

If you are new to Azure Developer, consider taking:

- Instructor-led course: AZ-204: Developing Solutions for Microsoft Azure

---

Programa

- Development for enterprise DevOps
- Implement CI with Azure Pipelines and GitHub Actions
- Design and implement a release strategy
- Implement a secure continuous deployment using Azure Pipelines
- Manage infrastructure as code using Azure and DSC
- Implement security and validate code bases for compliance
- Design and implement a dependency management strategy
- Implement continuous feedback

Development for enterprise DevOps

- Introduction to DevOps
- Plan Agile with GitHub Projects and Azure Boards
- Design and implement branch strategies and workflows
- Collaborate with pull requests in Azure Repos
- Explore Git hooks
- Plan foster inner source
- Manage and configure repositories
- Identify technical debt

Implement CI with Azure Pipelines and GitHub Actions

- Explore Azure Pipelines
- Manage Azure Pipeline agents and pools
- Describe pipelines and concurrency
- Design and implement a pipeline strategy
- Integrate with Azure Pipelines
- Introduction to GitHub Actions
- Learn continuous integration with GitHub Actions
- Design a container build strategy

Design and implement a release strategy

- Create a release pipeline
- Explore release recommendations
- Provision and test environments
- Manage and modularize tasks and templates
- Automate inspection of health

Implement a secure continuous deployment using Azure Pipelines

- 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

Manage infrastructure as code using Azure and DSC

- 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

Implement security and validate code bases for compliance

- Introduction to Secure DevOps
- Implement open-source software
- Software Composition Analysis
- Security Monitoring and Governance

Design and implement a dependency management strategy

- Explore package dependencies
- Understand package management
- Migrate consolidate and secure artifacts
- Implement a versioning strategy
- Introduction to GitHub Packages

Implement continuous feedback

- 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 cultu