



Enterprise Computing Solutions - Education Services

TRAINING OFFERING

Sie erreichen uns unter

Arrow ECS GmbH, Elsenheimerstraße 1, 80687 München

Email: training.ecs.de@arrow.com

Phone: +49 (0)89 930 99 168



AZ-400T00: Designing and Implementing Microsoft DevOps Solutions

CODE:	LÄNGE:	PREIS:
MCS_AZ-400T00	32 Hours (4 Tage)	€2,590.00

Description

Dieser Kurs vermittelt die Kenntnisse und Fähigkeiten zur Entwicklung und Umsetzung von DevOps-Prozessen und -praktiken. Die Teilnehmer*innen lernen, wie sie DevOps planen, Quellcodeverwaltung verwenden, Git für ein Unternehmen skalieren, Artefakte konsolidieren, eine Strategie für das Abhängigkeitsmanagement entwerfen, Geheimnisse verwalten, Continuous-Integration-Ansätze und eine Strategie zum Erstellen von Containern implementieren, eine Releasestrategie entwerfen, einen Releaseverwaltungsworkflow einrichten, ein Bereitstellungsmuster implementieren und Feedbackmechanismen optimieren.

Lernziel

- Transformation mit geteilten Zielen und Fristen planen
- Auswählen eines Projekts und Identifizieren von Projektmetriken und Leistungskennzahlen (Key Performance Indicators, KPIs)
- Erstellen eines Teams und einer agilen Organisationsstruktur
- Entwickeln einer Tool-Integrationsstrategie
- Entwerfen einer Lizenzverwaltungsstrategie (z. B. für Azure DevOps- und GitHub-Benutzer*innen)
- Entwerfen einer Strategie für die End-to-End Rückverfolgbarkeit von Arbeitsaufgaben zu Arbeitssoftware
- Entwickeln einer Anmeldungs- und Zugriffsstrategie
- Entwickeln einer Strategie zur Integration von Vor-Ort und Cloud-Ressourcen
- Vorteile bei der Nutzung einer Lösungskontrolle beschreiben
- Beschreiben von Azure Repos und GitHub
- Migrieren von der TFVC nach Git
- Verwalten der Codequalität, einschließlich technischer Schulden, SonarCloud und anderer Tools
- Aufbau von Organisationswissen über die Codequalität
- Erläutern der Strukturierung von Git-Repositories
- Beschreiben von verzweigten Git-Workflows
- Nutzen von Pull Requests für Zusammenarbeit und Codeüberprüfungen
- Nutzen von Git-Hooks für die Automatisierung
- Verwenden von Git zum Fördern des Inner-Source-Ansatzes innerhalb einer Organisation
- Erläutern der Rolle von Azure Pipelines und seiner Komponenten
- Konfigurieren von Agents für die Verwendung in Azure Pipelines

- Erklären, warum kontinuierliche Integration wichtig ist
- Implementieren von CI (Continuous Integration) mit Azure Pipelines
- Definieren von Site Reliability Engineering (SRE)
- Entwurfsprozesse zum Messen der Endbenutzerzufriedenheit und Analysieren von Benutzerfeedback
- Entwurfsprozesse zur Automatisierung der Anwendungsanalyse
- Verwalten von Warnmeldungen und Reduzieren von unnötigen und nicht relevanten Warnmeldungen
- Durchführen von Nachbesprechungen ohne Schuldzuweisungen und Schaffung einer Fairnesskultur
- Infrastruktur und Konfigurationsstrategien und die richtigen Werkzeuge für eine Freigabepipeline und Anwendungsinfrastruktur definieren
- Compliance und Sicherheit in Ihre Anwendungsinfrastruktur implementieren
- Beschreiben der potenziellen Herausforderungen bei der Integration von Open-Source-Software
- Überprüfen von Open-Source-Softwarepaketen auf Sicherheit und Lizenzkonformität
- Verwalten von Sicherheits- und Compiancerichtlinien der Organisation
- Integrieren von Lizenz- und Sicherheitsrisikoüberprüfungen in Build- und Bereitstellungspipelines
- Konfigurieren von Buildpipelines für den Zugriff auf Paketsicherheits- und Lizenzbewertung

Zielgruppe

Die Teilnehmer dieses Kurses sind daran interessiert, DevOps-Prozesse zu entwerfen und zu implementieren oder die Zertifizierungsprüfung für Microsoft Azure DevOps-Lösungen abzulegen.

Voraussetzungen

Erfolgreiche Kursteilnehmer verfügen über die folgenden Vorkenntnisse und Kenntnisse:

Cloud Computing-Konzepte, einschließlich eines Verständnisses von PaaS-, SaaS- und IaaS-Implementierungen.

Erfahrung in Bezug auf Azure-Verwaltung und Azure-Entwicklung mit nachgewiesenen Kenntnissen in mindestens einem dieser Bereiche

Kenntnisse in Bezug auf Versionskontrolle, agile Softwareentwicklung und die wichtigsten Prinzipien der Softwareentwicklung Es wäre hilfreich, wenn Sie Erfahrung in einem Unternehmen hätten, das Software liefert.

Wenn Sie noch nicht mit Azure und Cloud Computing vertraut sind, sollten Sie eine der folgenden Ressourcen in Betracht ziehen:

Kostenloser Online-Lernpfad: Azure-Grundlagen (<https://docs.microsoft.com/en-us/learn/paths/azure-fundamentals/>)

Präsenzschiung: AZ-900: Azure Fundamentals (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-900t00>)

Wenn Sie noch nicht mit der Azure-Verwaltung vertraut sind, ziehen Sie Folgendes in Betracht:

Kostenloser Onlinekurs: Prerequisites for Azure Administrators (<https://docs.microsoft.com/en-us/learn/paths/az-104-administrator-prerequisites/>)

Präsenzschiung: AZ-104: Microsoft Azure Administrator (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-104t00>)

Wenn Sie noch nicht mit Azure Developer vertraut sind, ziehen Sie Folgendes in Betracht:

Kostenloser Onlinekurs: Create serverless applications (<https://docs.microsoft.com/en-us/learn/paths/create-serverless-applications/>)

Präsenzschiung: AZ-204: Entwickeln von Lösungen für Microsoft Azure (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-204t00>)

Inhalt

Modul 1: Erste Schritte zur DevOps-Transformation Lektionen

- Einführung in DevOps
- Auswählen des richtigen Projekts

- Beschreiben von Teamstrukturen
- Auswählen der DevOps-Tools
- Agile-Planung mit GitHub-Projekten und Azure Boards
- Einführung in die Quellcodeverwaltung
- Beschreiben von Typen von Quellcodeverwaltungssystemen
- Arbeiten mit Azure Repos und GitHub

Lab: Agile Planung und Portfoliomanagement mit Azure Boards Lab: Versionskontrolle mit Git in Azure Repos
 Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Verstehen, was DevOps ist und welche Schritte erforderlich sind, um DevOps zu erreichen
- Identifizieren von Teams für die Implementierung des Prozesses
- Planen der Transformation mit gemeinsamen Zielen und Zeitplänen
- Planen und Definieren von Zeitplänen für Ziele
- Verstehen der verschiedenen Projekte und Systeme als Einstieg in die Journey
- Auswählen eines Projekts zum Starten der DevOps-Transformation
- Identifizieren von Gruppen zum Minimieren anfänglicher Widerstände
- Identifizieren von Projektmetriken und Key Performance Indicators (KPI)
- Grundlegendes zu agilen Verfahren und Prinzipien der agilen Entwicklung
- Erstellen eines Teams und einer agilen Organisationsstruktur

Modul 2: Entwicklung für DevOps in Unternehmen Lektionen

- Strukturieren Ihres Git-Repositorys
- Verwalten von Git-Verzweigungen und -Workflows
- Zusammenarbeiten über Pull Requests in Azure Repos
- Erkunden von Git-Hooks
- Planen der Förderung des Inner-Source-Ansatzes
- Git-Repositorys verwalten
- Identifizieren technischer Schulden

Lab: Versionskontrolle mit Git in Azure Repos Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Grundlegendes zu Git-Repositorys
- Implementieren eines Monorepositorys oder mehrerer Repositorys
- Erläutern der Strukturierung von Git-Repositorys
- Implementieren eines Änderungsprotokolls
- Beschreiben von Git-Verzweigungsworkflows
- Implementieren von Featurebranches

- Implementieren von GitFlow
- Forken eines Repositorys
- Nutzen von Pull Requests für Zusammenarbeit und Codeüberprüfungen
- Senden von Feedback mithilfe von Pull Requests

Modul 3: Implementieren von CI mit Azure Pipelines und GitHub Actions Lektionen

- Erkunden von Azure Pipelines
- Verwalten von Azure Pipeline-Agents und Agentpools
- Beschreiben von Pipelines und Parallelität
- Erkunden von Continuous Integration
- Implementieren einer Pipelinestrategie
- Integration in Azure Pipelines
- Einführung in GitHub-Aktionen
- Informationen zu Continuous Integration (CI) mit GitHub Actions

- Entwerfen einer Strategie zur Containererstellung
 - Lab: Konfigurieren von Agent-Pools und Verstehen von Pipelinearten
 - Lab: Aktivieren von Continuous Integration mit Azure Pipelines
 - Lab: Integrieren einer externen Quellcodeverwaltung mit Azure Pipelines
 - Lab: Implementieren von GitHub Actions mithilfe von DevOps Starter
 - Lab: Bereitstellen von Docker-Containern für Azure App Service-Web-Apps
- Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Beschreiben von Azure Pipelines
- Erläutern der Rolle von Azure Pipelines und seiner Komponenten
- Entscheiden über die Zuständigkeit für die Pipelineautomatisierung
- Grundlegendes zu Azure Pipeline-Schlüsselbegriffen
- Auswählen zwischen von Microsoft gehosteten und selbstgehosteten Agents
- Installieren und Konfigurieren von Azure Pipelines-Agents
- Konfigurieren von Agentpools
- Schützen der Agents und Pools
- Verwenden und Schätzen von Parallelaufträgen

Modul 4: Entwerfen und Implementieren einer Releasestrategie Lektionen

- Einführung in Continuous Delivery (CD)
- Kennenlernen der Empfehlungen für eine Releasestrategie
- Erstellen einer qualitativ hochwertigen Releasepipeline
- Einführung in Bereitstellungsmuster
- Implementieren von Blau-Grün-Bereitstellung und Feature Toggles

- Implementieren von Canary-Releases und Dark Launching
- Implementieren von A/B-Tests und der Bereitstellung mit progressiver Exposition
Lab: Erstellen eines Releasedashboards Lab: Steuern von Bereitstellungen mithilfe von Releasegates
Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Erläutern der Continuous Delivery-Konzepte
- Implementieren von Continuous Delivery in dem Entwicklungszyklus
- Informationen zu Releases und Bereitstellungen
- Identifizieren von Projektmöglichkeiten zum Anwenden von CD
- Erläutern, was beim Entwerfen einer Releasestrategie zu berücksichtigen ist
- Definieren der Komponenten einer Releasepipeline und Verwenden von Artefaktquellen
- Erstellen eines Releasegenehmigungsplans
- Implementieren von Release Gates
- Zwischen Freigabe und Bereitstellung unterscheiden

Modul 5: Implementieren eines sicheren Continuous-Deployment-Prozesses mit Azure Pipelines Lektionen

- Erstellen einer Releasepipeline
- Bereitstellen und Testen von Umgebungen
- Verwalten und Modularisieren von Aufgaben und Vorlagen
- Automatisieren der Integritätsprüfung
- Verwalten von Anwendungskonfigurationsdaten
- Integration mit Identitätsverwaltungssystemen
- Implementieren der Anwendungskonfiguration
Lab: Konfigurieren von Pipelines-as-Code mit YAML Lab: Einrichten und Ausführen von Funktionstests
Lab: Integrieren von Azure Key Vault mit Azure DevOps Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Terminologie erklären, die in Azure-DevOps und anderen Freigabe-Managementtools verwendet wird
- Beschreiben, was Build- und Releaseaufgaben sind, was sie können und einige verfügbare Bereitstellungsaufgaben
- Implementieren von Releaseaufträgen
- Unterscheiden zwischen Multi-Agent- und Multi-Konfigurations-Releaseaufträgen
- Bereitstellen und Konfigurieren der Zielumgebung
- Sicheres Bereitstellen in einer Umgebung mithilfe einer Dienstverbindung
- Konfigurieren der Automatisierung von Funktionstests und Ausführen von Verfügbarkeitstests
- Einrichten der Testinfrastruktur
- Verwenden und Verwalten von Aufgaben- und Variablengruppen

Modul 6: Verwalten von Infrastructure-as-Code mithilfe von Azure und DSC Lektionen

- Erkunden von Infrastructure-as-Code und Konfigurationsverwaltung

- Erstellen von Azure-Ressourcen mithilfe von Azure Resource Manager-Vorlagen
- Erstellen von Azure-Ressourcen mit der Azure CLI
- Erkunden von Azure Automation mit DevOps
- Implementieren von Desired State Configuration (DSC)
- Implementieren von Bicep

Lab: Azure-Bereitstellungen mithilfe von Azure Resource Manager-Vorlagen

Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Grundlegendes zum Bereitstellen Ihrer Umgebung
- Planen Ihrer Umgebungskonfiguration
- Auswählen zwischen imperativer und deklarativer Konfiguration
- Erläutern der idempotenten Konfiguration
- Erstellen von Azure-Ressourcen mithilfe von ARM-Vorlagen
- Verstehen von ARM-Vorlagen und Vorlagenkomponenten
- Verwalten von Abhängigkeiten und Geheimnissen in Vorlagen
- Organisieren und Modularisieren von Vorlagen
- Erstellen von Azure-Ressourcen mit der Azure CLI

Modul 7: Implementieren der Sicherheit und Überprüfen von Codebasen auf Compliance Lektionen

- Einführung in Secure DevOps
- Implementieren von Open-Source-Software
- Softwarezusammensetzungsanalyse
- Statische Analysetools
- OWASP und dynamische Analysetools
- Sicherheitsüberwachung und Governance

Lab: Implementieren von Sicherheit und Compliance in Azure Pipelines

Lab: Verwalten technischer Schulden mit SonarQube und Azure DevOps

Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Identifizieren von Angriffen durch Einschleusung von SQL-Befehlen
- Grundlegendes zu DevSecOps
- Implementieren von Pipelinesicherheit
- Informationen zur Bedrohungsmodellierung
- Implementieren von Open-Source-Software
- Erläutern von Bedenken seitens Unternehmen hinsichtlich Open-Source-Komponenten
- Beschreiben von Open-Source-Lizenzen
- Grundlegendes zu Lizenzauswirkungen und -bewertungen
- Arbeiten mit statischen und dynamischen Analysetools

- Konfigurieren von Microsoft Defender für Cloud

Modul 8: Entwerfen und Implementieren einer Strategie für die Abhängigkeitsverwaltung Lektionen

- Erkunden von Paketabhängigkeiten
- Grundlegendes zur Paketverwaltung
- Migrieren, Konsolidieren und Sichern von Artefakten
- Implementieren einer Versionsverwaltungsstrategie
- Einführung in GitHub-Pakete

Lab: Paketverwaltung mit Azure Artifacts Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Strategien zur Abhängigkeitsverwaltung definieren
- Ermitteln von Abhängigkeiten
- Elemente und Komponentisierung der Abhängigkeitsverwaltung beschreiben
- Ihre Codebasis auf Abhängigkeiten überprüfen
- Implementieren der Paketverwaltung
- Verwalten von Paketfeeds
- Verwenden und Erstellen von Paketen
- Veröffentlichen von Paketen
- Identifizieren von Artefaktrepositoren
- Migrieren und Integrieren von Artefaktrepositoren

Modul 9: Implementieren eines kontinuierlichen Feedbacks Lektionen

- Implementieren von Tools zum Nachverfolgen von Verbrauch und Flow
- Entwickeln von Überwachungs- und Statusdashboards
- Teilen von Wissen innerhalb von Teams
- Entwurfsprozesse zur Automatisierung der Anwendungsanalyse
- Verwalten von Warnungen, vorwurfsfreien Nachbesprechungen und einer Fairnesskultur

Lab: Überwachen der Anwendungsleistung mit Application Insights Lab: Integration zwischen Azure DevOps und Microsoft Teams

Lab: Wissensaustausch im Team mithilfe von Azure-Projekt-Wikis

Nach Abschluss dieses Moduls werden die Teilnehmer in der Lage sein:

- Implementieren von Tools zum Nachverfolgen von Feedback
- Planen der kontinuierlichen Überwachung
- Implementieren von Application Insights
- Verwenden der Kusto-Abfragesprache (KQL)
- Implementieren des Routings für mobile Anwendungen
- Konfigurieren der App Center-Diagnose

- Konfigurieren von Warnungen
- Erstellen eines Bugtrackers
- Konfigurieren von Azure Dashboards
- Verwenden des Ansichts-Designers in Azure Monitor

Test und Zertifizierung

Vorbereitung für Examen: AZ-400

Kurstermine

Datum	Lokation	Time Zone	Sprache	Type	Durchführungsgarantie	PREIS
22 Apr 2025	Virtual Classroom	CEDT	German	Instructor Led Online		€2,590.00
22 Apr 2025	München	CEDT	German	Classroom		€2,590.00
16 Jun 2025	Virtual Classroom	CEDT	German	Instructor Led Online		€2,590.00
28 Jul 2025	Virtual Classroom	CEDT	German	Instructor Led Online		€2,590.00
15 Sep 2025	Virtual Classroom	CEDT	German	Instructor Led Online		€2,590.00
08 Dec 2025	Virtual Classroom	GMT	German	Instructor Led Online		€2,590.00

Zusätzliche Information

[Diese Schulung ist auch als Vor-Ort-Schulung verfügbar. Bitte kontaktieren Sie uns, um mehr zu erfahren.](#)