

DevNet Tools

Programmierung und Automatisierung

Der Kurs DevNet Tools vermittelt einen einfachen Einstieg in die Werkzeuge der Digitalisierung. Durch Themen wie DevOps kommen zudem noch kulturelle und professionelle Änderungen in Prozessen in Betracht, welche sich in der Kommunikation, Zusammenarbeit, Integration und Automation widerspiegeln. Dazu ist nicht nur ein Umdenken in Unternehmen unerlässlich, sondern es werden auch neue Tools, Schnittstellen und technische Möglichkeiten benötigt. Die meisten Hersteller setzen hierzu auf offene Schnittstellen und stellen Beschreibungen zur Verfügung, um diese zu nutzen. Durch die Vielzahl der Möglichkeiten steht man als Einsteiger vor der Herausforderung, die mannigfaltigen Begrifflichkeiten grundlegend einzusortieren. Dieser Kurs stellt daher die wichtigsten eingesetzten Werkzeuge vor. Die Fragestellung „Warum sollte ich den Weg der Digitalisierung beschreiten?“ wird genauso behandelt wie das „Wie kann ich es tun?“. Dabei wird der Teilnehmer Schritt für Schritt an die aktuellen Technologien herangeführt.

Kursinhalt

- Automatisierung
- API
- DevOps
- Geführte Beispiel-Programmierung einer kleinen Anwendung
- Datenstrukturen
- Digitalisierung
- SDN
- Orchestrierung

E-Book Das ausführliche deutschsprachige digitale Unterlagenpaket, bestehend aus PDF und E-Book, ist im Kurspreis enthalten.

Zielgruppe

Die Schulung richtet sich an alle, die einen Einstieg in die Themen Digitalisierung, Programmierung und Application Programmable Interface (API) suchen.

Voraussetzungen

Besondere Vorkenntnisse werden nicht verlangt.

Dieser Kurs im Web



Alle tagesaktuellen Informationen und Möglichkeiten zur Bestellung finden Sie unter dem folgenden Link: www.expertteach.de/go/EDEV

Vormerkung

Sie können auf unserer Website einen Platz kostenlos und unverbindlich für 7 Tage reservieren. Dies geht auch telefonisch unter 06074 4868-0.

Garantierte Kurstermine

Für Ihre Planungssicherheit bieten wir stets eine große Auswahl garantierter Kurstermine an.

Ihr Kurs maßgeschneidert

Diesen Kurs können wir für Ihr Projekt exakt an Ihre Anforderungen anpassen.

Stand 01.06.2025

Training		Preise zzgl. MwSt.
Termine in Deutschland	3 Tage	€ 2.195,-
Online Training	3 Tage	€ 2.195,-
Termin/Kursort	Kursprache Deutsch 	
03.12.-05.12.25  Frankfurt	27.05.-29.05.26  Frankfurt	
03.12.-05.12.25  Online	27.05.-29.05.26  Online	

Inhaltsverzeichnis

DevNet Tools – Programmierung und Automatisierung

1 Applikationen in der Cloud	2.10.2 Puppet und Chef	6.1.1 Download, Installation und Lieferumfang
1.1 Applikationen in der Cloud	2.10.3 Ansible	6.1.2 Python-Shell
1.1.1 Aufbau von Applikationen		6.1.3 Schnelldurchlauf im interaktiven Modus
1.1.2 Open APIs	3 Realisierung einer Microservices-Landschaft	6.1.4 Programmausführung im interaktiven Modus
1.1.3 Wo kommt das OS her? (Satellite Server, Cloud-init, ...)	3.1 Versionsverwaltung mit Git	6.1.5 Verwendung eines Editors oder einer IDE
1.1.4 Cloud-untaugliche Anwendungen	3.1.1 Git	6.1.6 Programmausführung eines Python Skriptes
1.1.5 Cloud Native Applications	3.1.2 GitHub	6.2 Datentypen
1.2 12-Factor-App	3.1.3 GitLab	6.2.1 Erste Datentypen in Python
1.2.1 Versionsverwaltung (Git)	3.2 Tools für Continuous Integration	6.3 Anweisung und Anweisungsblock
1.2.2 Saubere Trennung des eigentlichen Codes von anderem	3.2.1 GitLab für DevOps	6.3.1 Anweisungen und Blöcke in Python
1.2.3 Build, Release, Run	3.2.2 Jenkins	6.4 Ausdruck
1.2.4 Stateless Applications	3.2.3 TravisCI	6.4.1 Boolesche und Arithmetische Operationen
1.2.5 Web Services und Port-Bindung	3.3 Testing	6.4.2 Weitere Ausdrücke in Python
1.2.6 Microservices	3.3.1 Testautomatisierung	6.5 Variablen und Zuweisungen
1.2.7 Weitere Faktoren	3.4 REST API im Detail	6.5.1 Variablen in Python
1.2.8 ISTQB	3.4.1 RESTful API	6.5.2 Details der Zuweisung
1.3 Applikationssicherheit in Cloud-Umgebungen	3.4.2 Konstruktion der HTTP-Meldungen	6.6 Funktionen I: Eingebaute Funktionen
1.3.1 OWASP Top 10	3.4.3 Generische Beschreibung eines REST API	6.6.1 Die Hilfe-Funktion
1.3.2 Continuous Integration	3.4.4 Idempotenz	6.7 Vergleiche und Bedingungen
1.3.3 DevOps	3.4.5 Nested Ressource Expansion	6.7.1 Die Bedingung: if
1.3.4 Continuous Delivery	3.4.6 Authentifizierung	6.7.2 Kombinationen und Verknüpfungen
1.3.5 CI/CD	3.4.7 Content Negotiation	6.8 Datentypen II – String Details
1.3.6 Skripte, Tools und Lifecycle von Apps in der Cloud	3.4.8 Alternativen zu REST	6.9 Iterationen I – While
1.4 Ausblick	3.5 API-Gateway	6.9.1 While-Schleife
1.4.1 Container-Apps	3.5.1 Reverse Proxying mit NGINX (Beispiel)	6.9.2 For – Eine Vorschau
1.4.2 Wie geht es weiter?	3.6 API Gateway-Frameworks	6.10 Funktionen II – Eigene Funktionen
2 Einführung in Microservices	4 Installation und Bedienung	6.10.1 Verschachtlung
2.1 Definition Microservices	4.1 Installation von Postman	6.10.2 Global oder Lokal
2.2 Motivation für Microservices	4.2 Nutzung eines Accounts	6.10.3 Rekursion
2.3 Herausforderungen beim Einsatz von Microservices	4.3 Die Benutzeroberfläche von Postman	6.10.4 Arten von Argumenten
2.4 Gestaltung von Microservices	4.4 Anlegen eines Environment	6.11 Datentypen III
2.4.1 Designkriterium Größe von Microservices	4.5 Arbeiten mit Variablen	6.11.1 Liste
2.4.2 Designkriterium Teamstrukturen (Gesetz von Conway)	4.6 Nutzung von Collections	6.11.2 Weitere Methoden von Listen
2.4.3 Designkriterium Domain-driven Design und Bounded Context	4.6.1 Anlegen einer eigenen Collection	6.11.3 Tupel
2.4.4 Designkriterium User Interface	4.6.2 Einbinden einer öffentlichen Collection	6.11.4 Sequenz
2.4.5 Gründe, um weitere Microservices zu erstellen	5 Container	6.11.5 Slicing
2.5 Aufteilung des Gesamtprojekts – Large Scale Structures	5.1 Zusammenhang Microservices und Container	6.11.6 Eine Menge - Set
2.6 Anwendungsarchitekturen auf Basis von Microservices	5.2 Container-Virtualisierung – Technische Aspekte	6.11.7 Dictionaries
2.6.1 Fachliche Aufteilung	5.2.1 Container-Technologien	6.11.8 Dictionaries – Methoden
2.6.2 Design Tipps	5.3 Docker	6.12 Iterationen II
2.6.3 Konfiguration eines Microservices – Zentral oder im Service	5.3.1 Bestandteile von Docker	6.12.1 For-Details
2.6.4 Data Management in Microservices-Umgebungen	5.3.2 Docker-Nutzung in Unternehmen	6.13 Weitere Konzepte und Funktionen
2.6.5 Service Discovery	5.3.3 Docker Monitoring	6.13.1 Exceptions
2.6.6 Load Balancing – Service Discovery versus Load Balancer	5.4 Kubernetes	6.13.2 Anonyme Funktionen
2.7 Sicherheit	5.5 Service Meshes	6.13.3 Comprehensions
2.8 Service-Meshes	5.5.1 Istio	
2.9 Monitoring	5.6 Serverless Computing – Functions	7 API-Gateway
2.10 Deployment	5.7 Cloud: Optimierte für Mode-2-Anwendungen	7.1 Ohne API-Gateway...
2.10.1 Orchestrierung von Infrastruktur und Applikationen	6 Die Programmiersprache Python	7.2 Mit API-Gateway!
	6.1 Was ist Python?	7.2.1 Reverse Proxying mit NGINX (Beispiel)
		7.3 API Gateway-Frameworks
		7.4 Welches Gateway?

