APIs mit GraphQL

Eine Abfragesprache für APIs

Viele Apps fragen Daten ab, die von einer Datenbank verwaltet werden. Die App richtet ihre Anfrage dabei an einen Server, der dafür eine API anbietet und die Interaktion mit der Datenbank erledigt. GraphQL ist eine Abfragesprache, mit welcher der Client dem Server mitteilt, was er zurückbekommen möchte.

Facebook (Meta) hat GraphQL entwickelt und setzt es seit 2012 für Mobile Apps ein. Das Ziel besteht darin, eine effiziente Alternative zu REST zu haben. Während RESTful APIs mehrere Endpunkte anbieten, verwendet man bei GraphQL nur einen. Der Client legt in seiner Nachricht an den Server präzise fest, mit welchen Daten er interagieren will.

Neben der Eignung für Mobile Apps gehören auch eine schnelle Entwicklung neuer Features in Verbindung mit Continous Deployment zu den Zielen. Einige größere Unternehmen setzen GraphQL für ihre APIs ein. Dazu gehören Airbnb, Audi, Atlassian, GitHub, Twitter und weitere (siehe https://graphql.org/users/). GraphQL-Bibliotheken für die Client- oder Serverseite existieren für unterschiedliche Programmiersprachen wie z. B. Python, Javascript, Java, Kotlin, Go, Rust, C/C++ u. a.

Kursinhalt

- Grundkonzepte der GraphQL-basierten Kommunikation
- Die GraphQL Schema Definition Language (SDL)
- Das Typensystem
- Vergleich mit REST
- Client- und Serverseitige Bibliotheken für Python
- Authorization
- Pagination

Zielgruppe

Dieser Kurs richtet sich an alle, die ein Grundverständnis für GraphQL und seine Einsatzmöglichkeiten suchen. Dazu gehören Softwareentwickler, die GraphQL-APIs entwickeln wollen, aber beispielsweise auch Operations-Mitarbeiter, welche die Basics und die Unterschiede zu RESTful APIs verstehen wollen.

Voraussetzungen

Ein Grundverständnis von APIs und ihren Use Cases sollten vorhanden sein. Außerdem sind Grundkenntnisse der Programmierung – speziell in Python – für das Verständnis der Code-Beispiele nützlich. Der Kurs Python für Einsteiger – Einführung in die Programmierung vermittelt solche Grundkenntnisse.

Kursziel

Sie bauen ein Grundverständnis darüber auf, wie mögliche Anwendungsfälle aussehen und wie eine GraphQL API funktioniert. Dazu gehört der Aufbau des GraphQL Schemas, in dem definiert wird, welche Daten der Server bietet und welche Operationen man auf diesen ausführen kann. Weiterhin lernen Sie, wie man ein GraphQL-Dokument, mit dem ein Client eine Anfrage an den Server stellt, schreibt.

Dieser Kurs im Web

Alle tagesaktuellen Informationen und Möglichkeiten zur Bestellung finden Sie unter dem folgenden Link: www.experteach.ch/go/**AGRA**

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.

Premium Print Paket

Zu diesem Kurs können sie optional das hochwertige Premium Print Paket zum Preis von € 165,- (zzgl. MwSt) erwerben.

Training	Preise zzgl. MwSt.
Termine in Deutschland	**2 Tage CHF 1.975,-**
Online Training	**2 Tage CHF 1.975,-**

Termin/Kursort	Kurssprache Deutsch		
03.05.-04.05.27	Frankfurt	11.11.-12.11.27	Frankfurt
03.05.-04.05.27	Online	11.11.-12.11.27	Online

Stand 27.10.2026

Inhaltsverzeichnis

APIs mit GraphQL – Eine Abfragesprache für APIs

1 Überblick über GraphQL
1.1 Was ist GraphQL eigentlich?
1.2 Ausgangspunkt REST
1.3 Grundlagen von GraphQL - das Schema
1.3.1 Die Root Types
1.3.2 Query
1.3.3 Mutation
1.3.4 Subscription
1.3.5 Die Design-Prinzipien von GraphQL
1.4 GraphQL APIs testen (mit GraphiQL)

2 Aus der Sicht des Client
2.1 Das GraphQL-Dokument des request Strings
2.1.1 Operation Name
2.1.2 Dokumente mit mehreren Operationen
2.1.3 Verschachtelte Objekte
2.1.4 Kommentare in Queries
2.1.5 Query mit Argument
2.1.6 Variablen
2.1.7 Verwendung von Alias
2.1.8 Fragments
2.1.9 Directives
2.1.10 Enum
2.1.11 Interface und Union
2.1.12 Input Objects
2.2 Fehlermeldungen - Request Error
2.3 Introspection
2.3.1 Der Type __schema
2.3.2 Der Type __typename
2.3.3 Der Type __type

3 Aus der Sicht der Servers
3.1 Das Typen-System von GraphQL
3.1.1 Die Schema Definition
3.1.2 Scalars
3.1.3 Objects
3.2 Resolver Functions