

1 Learning

Du solltest ein grundlegendes Verständnis dieser Technologien präsentieren können. Notiere Verständnis-Schwierigkeiten. Bei Fragen kannst du dich per WA an mich wenden.

1.1 GraphQL

Arbeite die GraphQL Einführung unter <https://graphql.org> durch.

<https://graphql.org/learn/>

Notiere deine Meinung zu dieser API Technologie, auch im Vergleich mit REST und SOAP.

1.2 Angular

Mache dich mit der aktuellen Angular Version vertraut. Nutze dafür das Tutorial:

<https://angular.io/tutorial>

1.3 Lodash

Beschäftige dich kurz mit Lodash

<https://lodash.com/docs/4.17.15>

2 Einrichten lokale Umgebung

Du solltest zeigen dass du in der Lage bist deine Arbeitsumgebung einrichten und verwalten zu können. Notiere Schwierigkeiten und welche Online Ressourcen du verwendet hast, diese zu lösen!

2.1 Installation

Installiere lokal

- <https://nodejs.org/en/>
- <https://code.visualstudio.com>
- <https://angular.io/guide/setup-local>
- Optional <https://www.mongodb.com/de> (Free Edition)

2.2 Test

Teste dass alle Komponenten korrekt installiert sind und du sie starten kannst.

3 Application Development

Zeige dass du mithilfe von vorhandenen Online-Tutorials diese Aufgaben zu lösen kannst.

3.1 Express GraphQL Server

Entwickle einen Beispiel-Express Server der eine Apollo GraphQL Instanz verwendet.

<https://expressjs.com>

<https://www.apollographql.com>

<https://www.apollographql.com/docs/apollo-server/integrations/middleware/>

Schreibe ein einfaches GraphQL Schema

<https://www.apollographql.com/docs/apollo-server/schema/schema/>

inkl. der notwendigen Resolver

<https://www.apollographql.com/docs/apollo-server/data/resolvers/>

und verwende dieses in deiner Anwendung.

Erstelle ein Schema mit einem Typ „car“ und mindestens den Fields „Brand“, „Color“, „engine power“ und „has trailer hitch“. Finde die richtigen Typen für die Fields. Verwende mind. 1 Enum Typ.

Die Resolver können auf ein hard-codiertes Array zugreifen, dass im Code steht.

3.2 Angular UI

Entwickle eine Angular Anwendung (verwende Angular Material UI

<https://material.angular.io>) die mittels Apollo Client auf deinen Apollo GraphQL Server zugreift.

Erstelle eine Komponente die eine GraphQL Query verwendet, um die Cars abzufragen und diese tabellarisch oder per Cards darzustellen.

Schreibe HTML / CSS so, dass das Design / UI in deinen Augen ansprechend ist.

4 Extra

4.1 GraphQL Anwendung

Statt hard-codiertem Array schreibe Resolver die Daten aus einer lokalen MongoDB lesen und in diese schreiben.

4.2 Angular Anwendung

Schreibe die gleiche Anwendung mithilfe eines anderen Frameworks / Library, z. B. react oder Vue.js.