

# Aufgabe GraphQL

---

Ziel dieser Aufgabe ist es die To-Do-Anwendung umzuschreiben und zwar so, dass statt REST die GraphQL API zum Einsatz kommt. Zur Einführung in das Thema GraphQL schauen Sie sich die Folien und dort referenzierte Tutorials an. Danach machen Sie sich eine Kopie der laufenden ToDo-App (oder noch besser eine Git-Branch) und versuchen diese so Schritt für Schritt abzuändern.

## Aufgabe 1

Arbeiten Sie den Basic-Tutorial durch: <https://www.howtographql.com/>. In vielen Teilen wiederholt dieser die Slides/Video aus der Vorlesung, aber in Englisch und etwas systematischer ... auf jeden Fall zu empfehlen, auch wenn Sie denken, alles ist klar 😊. Insbesondere empfehle ich die Kapitel: "Core Concepts".

## Aufgabe 2

Für die Serverseitige Implementierung empfiehlt sich als einfachste Möglichkeit das Express GraphQL Server. Arbeiten Sie sich ein mit Hilfe des folgenden Tutorials ein: <https://graphql.org/graphql-js/running-an-express-graphql-server/>

Die Installation läuft über **NPM**, Einbinden lässt sich das GraphQL-Server als reguläre Express-Middleware. Sie können gerne das **GraphiQL**-Tool verwenden, um GraphQL-Abfragen auszuprobieren. Wenn Sie in einem Webbrowser zu <http://localhost:4000/graphql> navigieren, sollte eine Oberfläche angezeigt werden, über die Sie Abfragen eingeben können.

## Aufgabe 3

Implementieren Sie eine weitere Route **/graphql** zu ihrer Implementierung von ToDo-App. Es sollen dabei zunächst alle CRUD-Funktionalitäten repliziert werden. Greifen Sie dabei nicht auf die REST-API sondern direkt auf die Mongoose-Modelle zu. Testen Sie die API mit einem der Werkzeuge Ihrer Wahl. GraphiQL wird wahrscheinlich die einfachste Variante sein.

Bemerkung: Sie können auch gerne auf die Generierung von GraphQL-Modellen ausgehend aus Mongoose-Modellen zurückgreifen (<https://www.npmjs.com/package/graphql-compose-mongoose>).

## Aufgabe 4

Implementieren sie ein Paar Queries, die mit REST ohne weiteres nicht möglich wären. Testen Sie auch diese mit dem Testwerkzeug Ihrer Wahl.

## Aufgabe 5

Vervollständigen Sie nun die App durch Anpassung von Ihrem Front-End. Dabei müssen Sie eigentlich nur die **fetch** - Aufrufe abändern, so dass jetzt GraphQL-Anfragen über POST verschickt werden. Beachten Sie dabei die Fehlermeldungen, die jetzt auf eine ganz andere Weise bekanntgegeben werden.

Bemerkung: Sie können auch gerne (insbesondere wenn Sie schon für die erste Teilaufgabe **React** verwendet haben) auf die Apollo-Client-Bibliothek zurückgreifen.