

# WEB-API

Lerne mit einer einfachen Quiz-Web-API die grundlegenden Funktionen des Webs zu verstehen und wie die Kommunikation mit einem Server funktioniert.

Die Quiz-API soll Dir zeigen, wie Daten zwischen Nutzer:innen also Dir und der App ausgetauscht werden, das heißt, wie die Daten an den Server übermittelt werden und die Antwort vom Server an Dich zurückgeschickt wird. Bei Representational State Transfer (kurz REST) gibt es spezielle Operationen, um Anfragen durchzuführen, welche Du auch im Zuge dieses Kurses kennen lernen wirst.

Dieser Workshop deckt diesen Bereich der Cloud Computing Lab Matrix ab:

Dienste	Dienste im Web (Cloud Ressourcen im Web)	X	X	X
	Web-API (Representational State Transfer)			X

Besuche uns auf unserer Homepage

<https://cola.fh-joanneum.at/>



Viel Spaß wünscht das CoLa Team

Mathias Knoll, Harald Schwab, Stefanie Wieser, Priska Steininger,  
Michael Brickmann, Elmar Krainz, Petra Kletzenbauer, Sabine  
Proßnegg, Stefan Rapp

## Inhalt

<b>Spieleanleitung</b> .....	<b>3</b>
Funktionsweise RestAPI.....	3
CRUD-Operationen.....	3
Status Codes.....	5
Aufgaben.....	5

# Spieleanleitung

## Funktionsweise Rest-API

Eine REST-API ist eine Anwendungsprogrammierschnittstelle, die auf den Designprinzipien des REST-Architekturstils basiert und webbasierte Apps in der Kommunikation miteinander unterstützt. Grundsätzlich kannst du eine REST-API verwenden, um Daten zwischen dir und Apps auszutauschen. Die API übermittelt die Anfragen an den Server und leitet die Antwort des Servers an dich weiter (vgl. [www.talend.com](http://www.talend.com), [www.ibm.com](http://www.ibm.com)).

Auch die Quiz-App ist eine REST-API mit der du Anfragen an den Server schicken kannst, um Daten zu erstellen, zu lesen, zu ändern oder zu löschen. Somit sollst du einen ersten Einblick darüber bekommen, wie die Kommunikation zwischen dir und der REST-API stattfindet.

**CoLa2 - FragenAPI** 0.0.1 OAS 3.1  
/question-api/openapi.json

*Cloud Computing Lab*

**CoLa2 FragenAPI**

CoLa2 FragenAPI hilft dir mit einem einfachen Quiz die grundlegenden Funktionen des Web zu verstehen und wie die Kommunikation mit einem Server funktioniert. ↗

**Hauptseite**

- Du kannst dir den *HTML* Code der Hauptseite anzeigen lassen. [Gehe zu](#)

**Fragen**

- Du kannst dir alle **Fragen anzeigen** lassen, ...
  - oder du gibst im *zufall*-Query eine Zahl zwischen *1* und *100* ein, damit entsprechend viele Zufallsfragen abgerufen werden.
- Du kannst dir einzelne **Fragen** mit der **ID** anzeigen lassen. [Gehe zu](#)

**Antwort**

- Du kannst zu einer Frage eine Antwort zum Überprüfen an den Server senden. [Gehe zu](#)

🏠 Zurück zur Cloud Computing Lab | Coding Lab Homepage klick [hier](#)

**Weitere Infos und Kontakt**

CoLa2 - Website  
Send email to CoLa2  
GNU AFFERO GENERAL PUBLIC LICENSE

REST-APIs kommunizieren über HTTP (Hypertext Transfer Protocol). Mit der REST-API kannst du also Abfragen durchführen wie etwa Daten aus einer Datenbank lesen, Daten aktualisieren oder löschen. Das wird auch als CRUD-Operationen (**C**reate, **R**ead, **U**ppdate, **D**elete) bezeichnet. Die Informationen können dir in jedem beliebigen Format geliefert werden wie z.B. HTML, Python oder JavaScript Object Notation (JSON), was du in dieser Übung kennen lernen wirst (vgl. [www.ibm.com](http://www.ibm.com)).

## CRUD-Operationen

**CRUD** steht für **E**rstellen (Create), **L**esen (Read), **A**ktualisieren (Update) und **L**öschen (Delete). Dies sind die grundlegenden Operationen, die du mit der REST-API durchführen kannst. Im Rahmen der REST-API entsprechen sie den http-Methoden **POST**, **GET**, **PUT** und **DELETE** (vgl. [www.basedash.com](http://www.basedash.com)). Im Rahmen des Workshops lernst du die Operationen POST und GET genauer kennen.

### ERSTELLEN: HTTP POST

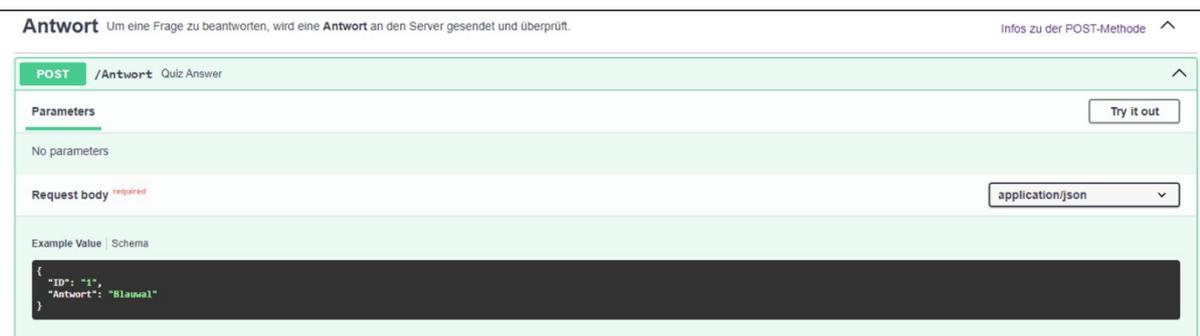
Um neue Daten in der API zu erstellen (wie in diesem Fall die Antwort hinzuzufügen), verwendest du die Methode HTTP POST. Zum Beispiel erstellst du die Antwort der Quizfrage mit POST und schickst diese an den Server.

### POST /Antwort

```
{  
  Id: "1",  
  Antwort: "Blauwahl"  
}
```

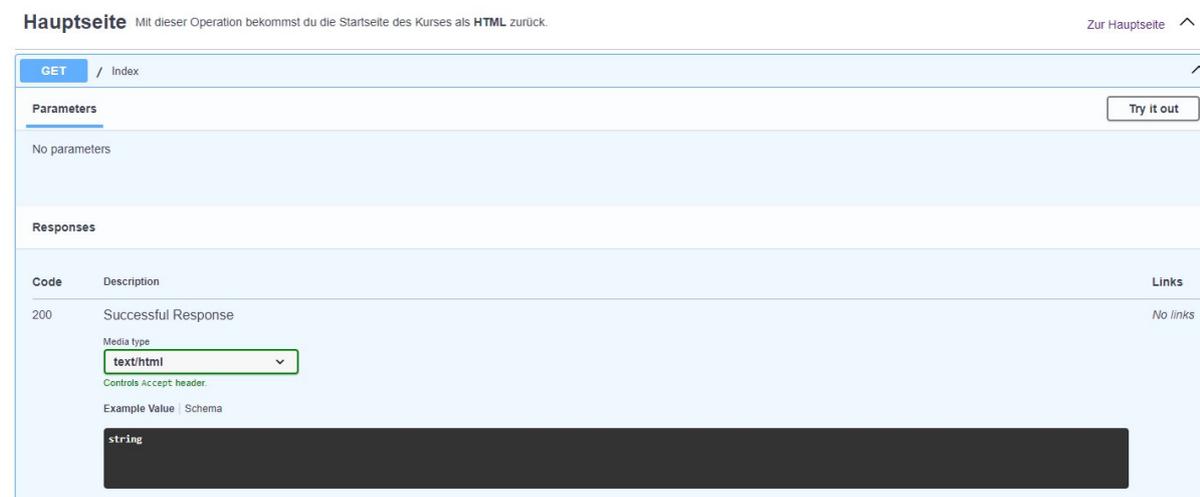
Das Format, das du in diesem Workshop verwendest, um die Antwort zu senden ist JavaScript Object Notation (JSON). Dies ist für Menschen leicht zu lesen und zu schreiben und für Maschinen leicht zu analysieren und zu generieren (vgl. [www.json.org](http://www.json.org)). JSON wird verwendet, um Daten zwischen Computern auszutauschen. Die Syntax ist von der JavaScript-Objektnotation abgeleitet, jedoch ist das JSON-Format nur Text ([www.w3schools.com](http://www.w3schools.com)).

Weitere Informationen über JSON findest du unter <https://www.json.org/json-en.html> und [https://www.w3schools.com/js/js\\_json\\_intro.asp](https://www.w3schools.com/js/js_json_intro.asp).



### LESEN: HTTP GET

Um Daten von der API abzurufen, kannst du HTTP GET verwenden. Mit GET / rufst du die Hauptseite der Quiz-API ab.



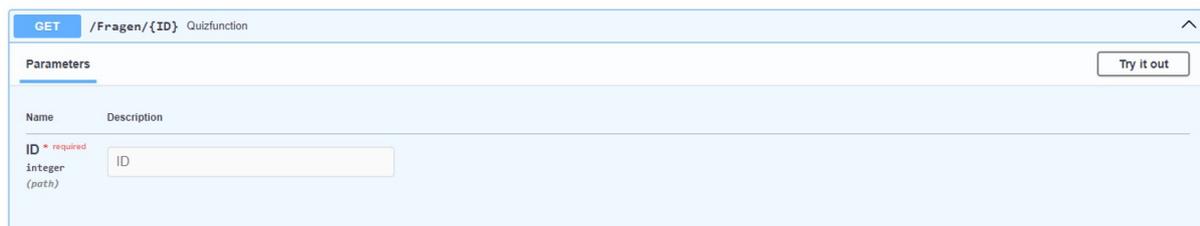
Mit **GET /Fragen** kannst du dir alle oder eine zufällige Frage vom Server abrufen.



Anfrage via GET /Fragen mit einer zufälligen FragenID [http:// https://cola.fh-joanneum.at/question-api/Fragen?zufall=2](http://cola.fh-joanneum.at/question-api/Fragen?zufall=2)

Wenn du über das HTTP-Protokoll Daten (Ressource) vom Server anforderst, wird z.B. die Zahl 2 als GET-Parameter in der URL mitgeschickt. Dies sind Name-Wert-Paare. GET-Parameter beginnen mit immer einem Fragezeichen **?**. Darauf folgt der Name der Variable mit dem Wert, getrennt durch ein **=** (vgl. [www.seobility.net](http://www.seobility.net)).

Mit **GET /Fragen/{id}** kannst du eine Frage mit einer speziellen FragenID abrufen.



## Status Codes

HTTP Status Codes geben dir Auskunft über das Ereignis einer Anfrage. Zu den gängigen Status-Codes gehören u.a. ([www.basedash.com](http://www.basedash.com)):

200 OK	Erfolgreiche GET- oder PUT-Anfrage.
201 Created	Erfolgreiche POST-Anfrage.
204 No Content	Erfolgreiche DELETE-Anfrage.
400 Bad Request	Die Anfrage war ungültig oder konnte nicht zugestellt werden.
404 Not Found	Die angeforderte Ressource konnte nicht gefunden werden.
405 Method Not Allowed	Die verwendete HTTP-Methode wird für diese Ressource nicht unterstützt.

## Aufgaben

1. Logge Dich bei der REST-API mit Namen ein.
2. Ruf alle Fragen oder nur eine zufällige anhand der GET-Anfrage ab.
3. Schicke Die Antwort via POST-Anfrage an den Server!
4. Probiere ein paar Fragen aus!



## Referenzen

GET Parameters (o. D.). Verfügbar unter: [https://www.seobility.net/en/wiki/GET\\_Parameters](https://www.seobility.net/en/wiki/GET_Parameters)

Introducing JSON (o. D.). Verfügbar unter: <https://www.json.org/json-en.html>

REST-API: Definition, Funktionen und Bedingungen (o. D.). Verfügbar unter:  
<https://www.talend.com/de/resources/was-ist-rest-api/>

Was ist eine REST-API? (o. D.). Verfügbar unter: <https://www.ibm.com/de-de/topics/rest-apis>

What are CRUD operations in a REST API? (2023). Verfügbar unter:  
<https://www.basedash.com/blog/what-are-crud-operations-in-a-rest-api>

JSON-Introduction. Verfügbar unter: [https://www.w3schools.com/js/js\\_json\\_intro.asp](https://www.w3schools.com/js/js_json_intro.asp)

