



Playing with HTTP APIs

Übung 2

Einleitung

- Wir werden in der Vorlesung noch genauer auf das Thema eingehen
- Wir wissen: Komponenten haben Schnittstellen (Interfaces)
- Heute kommt es oft vor, dass diese per HTTP zu Verfügung gestellt wird (ob gut oder schlecht → Vorlesung)
- Schnittstellen nutzen dabei HTTP Verben – die wichtigsten
 - GET: Gibt mir eine Ressource e.g.
`http://localhost:80/api/persons/12`
 - PUT: Mach ein Update einer Person e.g.
`http://localhost:80/api/persons/12`
 - POST: Erzeuge eine Person
e.g. `http://localhost:80/api/persons`
 - DELETE: Lösche eine Person e.g.
`http://localhost:80/api/persons/12`

JSON

- Oft wird die Antwort des Services als JSON (JavaScript Object Notation) zurückgegeben

```
{
  "Herausgeber": "Xema",
  "Nummer": "1234-5678-9012-3456",
  "Deckung": 2e+6,
  "Waehrung": "EURO",
  "Inhaber":
  {
    "Name": "Mustermann",
    "Vorname": "Max",
    "maennlich": true,
    "Hobbys": ["Reiten", "Golfen", "Lesen"],
    "Alter": 42,
    "Kinder": [],
    "Partner": null
  }
}
```

SQLite

- <https://sqlitebrowser.org/dl/>
- Nur bedingt für parallelen Zugriff geeignet

Swagger

- Rufen Sie <https://localhost:44309/swagger> auf - was sehen Sie hier?

swagger

Select a spec

Directory API 1.0.0

Directory API ^{1.0.0}

/swagger/v1/swagger.json

Directory

POST /api/entries

Models

```
Person {  
  id           integer($int32)  
  firstname    string  
  lastname     string  
}
```

Übung 2

- Unter AnotherWebProject finden sie die Angabe
- Erweitern Sie die Schnittstelle um:
 - GET api/entries: Liefert alle Entries zurück
 - DELETE api/entries/{id}: Löscht das Entry mit der Id = {id}
- Derzeit wird die Email Adresse nicht gespeichert – erweitern sie die Personen Entität um die Email
- Sie werden oft mit Legacy Code in Berührung kommen. Suchen sie alle Stellen zum Erweitern. Hint:
 - wwwroot / js / main.js
 - Nutzen Sie den Entwickler-Modus des Browsers (F12)

Übung 2

- Fügen Sie einen „Bearbeiten“ Button neben „Delete“ ein
- Erweitern sie das Interface um:
 - GET `api/entries/{id}`: Abrufen des zu ändernden Datensatzes
 - PUT `api/entries/{id}`: Macht ein Update der Entität (erweitern sie auch die UI entsprechend: in der Liste neben dem „Delete“ Button einen „Bearbeiten“ Button)
- Mit Submit wird der geänderte Datensatz wieder gespeichert

Achtung: Browser-Caching

- Browser Cachen – im Entwickler Modus kann der Cache zur Gänze geleert werden
- In der Praxis: Haben Sie Skripte wie Gulp oder Webpack, welche das Javascript versionieren → daher: wenn Sie bei der Übung die Welt nicht mehr verstehen → zuerst Cache leeren – eventuell altes Skript noch im Cache

