

18. Anropa REST-APIer

HTTP och JSON



JSON (JavaScript Object Notation)

Ett textbaserat sätt att beskriva objekt i JavaScript.

Används ofta för att på ett effektivt sätt skicka data mellan webbklient och -server.

Ett objekt definieras med `{}`.

JSON (JavaScript Object Notation)

Ett textbaserat sätt att beskriva objekt i JavaScript.

Används ofta för att på ett effektivt sätt skicka data mellan webbklient och -server.

Ett objekt definieras med {}.

Objekt har egenskaper i name/value-par

```
{  
  "name": "value"  
}
```

JSON (JavaScript Object Notation)

Egenskaperna kan ha värden som är t.ex. objekt, strängar, datum, tal, boolean, arrays.

```
{  
  "firstName": "Boaty",  
  "lastName": "McBoatface",  
  "length": 25,  
  "isBoat": true,  
  "crew": [  
    { name: "Phillips", title: "Captain" }  
  ]  
}
```

HTTP (Hyper Text Transfer Protocol)

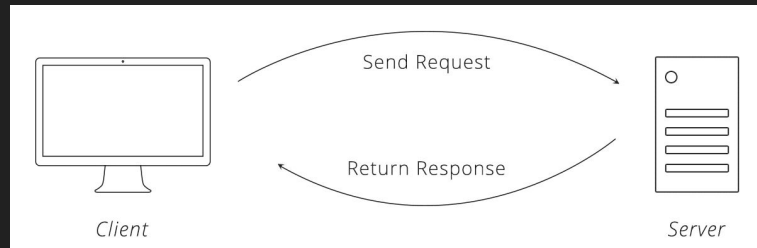
Kommunikationsprotokoll för kommunikation mellan klient och server.

Används av alla webbläsare.

Klienten gör en *request* till servern. Servern svarar med en *response*.

HTTPS är krypterad trafik över HTTP.

Ett **REST-API** är en server som svarar på requests.



HTTP (Hyper Text Transfer Protocol)

HTTP har metoder för klienten att anropa servern med (*request*):

- GET - för att hämta information
- POST - för att skicka information

Vanliga statuskoder i *response*

- 200: Ok
- 401: Unauthorized
- 404: Not found

HTTP Status Codes		
Level 200 (Success) 200 : OK 201 : Created 203 : Non-Authoritative Information 204 : No Content	Level 400 400 : Bad Request 401 : Unauthorized 403 : Forbidden 404 : Not Found 409 : Conflict	Level 500 500 : Internal Server Error 503 : Service Unavailable 501 : Not Implemented 504 : Gateway Timeout 599 : Network timeout 502 : Bad Gateway

REST-APIer

REST = REpresentational State Transfer

API = Application Programming Interface

REST-APIer använder **HTTP** för kommunikation och **JSON** för datatransport.

REST-APIer

Anrop till REST API görs till så kallade **endpoints**.

Ex.

GET `http://api.ava.se/teachers/`

Skulle kunna returnera **en lista** med alla lärare på Åva.

GET `http://api.ava.se/teachers/martinloman`

Skulle kunna returnera **ett objekt** med information om Martin.

REST-APIer - skicka parametrar

Parametrar kan skickas som query-parametrar.

Query-parametrar är på formen key/value. `parameternamn=värde`

Queryn läggs på URLen:

- Inleds med `?`
- Delas upp med `&`

Ex.

`GET http://api.ava.se/teachers/search?name=martin&course=WEBWEB01`

Anropa REST-API med JavaScript

1. Ett anrop till ett API tar lite tid.
2. JavaScript väntar inte på resultat innan det kör vidare.
3. Vi måste instruera JavaScript att vänta på svaret.
4. Detta kallas **asynkront**.
5. Vi kommer använda nyckelorden **async** och **await**.

Anropa REST-API med JavaScript

En funktion som deklareraras som asynkron

```
async function getTeachers() {  
  
}
```

Ett anrop som väntar på en asynkron funktion.

```
let teachers = await getTeachers()
```

Anropa REST-API med JavaScript

En funktion som deklarerar som asynkron

```
async function getTeachers() {  
  }  
}
```

Ett anrop som väntar på en asynkron funktion.

```
let teachers = await getTeachers()
```

Viktigt!

För att det här ska fungera måste man BÅDE **deklarera** sin funktion som asynkron OCH **vänta** på den.

Anropa REST-API med JavaScript

En bra inbyggd funktion för att anropa REST APIer

fetch()

https://www.w3schools.com/js/js_api_fetch.asp

Använd gärna async/await. Se länken ovan.

API-nyckel

De flesta API:er kräver att man registrerar sig för att kunna hämta eller skriva data

Ofta handlar det om en lång textsträng som klistras in i API:ets URL (adress), typiskt som en del av query-strängen

Exempel:

https://api.themoviedb.org/3/search/movie?query=Favoritfilmer&api_key=5f06e2549a07ff2t46205c70a08g1d3e

Om du registrerar dig (användarnamn+lösenord) på t.ex. The Movie Database så kan du logga in och hämta din API-nyckel och klistra in den i URL:en enligt ovan

På så sätt vet API:ets administratör vem som utför anropet och kan vid behov förhindra missbruk (t.ex. överbelastningsattacker)

Veckans uppgifter (två lektioner)

Utgå från kod i detta repo: <https://github.com/martinloman/js-call-api-template>

I koden finns det kommentarer som hjälper er att komma igång.

Uppgiften är att:

- Läsa text ur ett textfält i sidan.
- Använda texten för att skicka till ett REST API.
- När APIet svarat ska resultatet visas i sidan.
- Snygga till er sida lite med CSS.
- När det är klart kan ni fortsätta att utforska ert API och se om ni kan utveckla er sida. Kan ni göra en sida som visar mer detaljer om något som ni hämtat från APIet.

Exempel på gratis APler

Filmer: <https://www.themoviedb.org/documentation/api>

Anime: <https://ghibli.rest> (tillåter GET-requests utan registrering/nyckel)

Väder: <https://developer.accuweather.com/apis>

Recept: <https://spoonacular.com/food-api/docs>

Giphy: <https://developers.giphy.com/docs/api>

Discord: <https://discord.com/developers/docs/intro>

Massor av andra APler: <https://free-apis.github.io/#/>