



JavaScript 2

Lektion => 5

Utbildare: Mahmud Al Hakim

NACKADEMIN

Lektionstillfällets mål

Mål med lektionen

- Ajax via jQuery
- Fetch API
- Praktiska övningar
- **Att läsa:** sid. 388–408
- **Arbetsmetod**
 - Teori och praktik varvas under lektionen

NACKADEMIN

Kort summering av föregående lektion

- Vi har gått igenom
 - Ajax
 - XML
 - JSON
 - Introduktion till API

Without jQuery, AJAX coding can be a bit tricky!

Writing regular AJAX code can be a bit tricky, because different browsers have different syntax for AJAX implementation. This means that you will have to write extra code to test for different browsers. However, the jQuery team has taken care of this for us, so that we can write AJAX functionality with only one single line of code.

https://www.w3schools.com/jquery/jquery_ajax_intro.asp

Ajax via jQuery-metoden load()

```
$ (document) . ready ( function () {
```

```
    $ ("#demo") . load ("demo.txt");
```

```
});
```

OBS!

load() är en metod
som finns i jQuery

Mer om metoden load()

```
$("#demo").load("demo.txt", function(response,status,xhr){  
  
    // response - contains the result data from the request  
    console.log(response);  
  
    // status - contains the status of the request  
    console.log(status);  
  
    // xhr - contains the XMLHttpRequest object  
    console.log(xhr);  
});
```

Filen demo.json

```
{  
  "id": 1,  
  "name": "Mahmud Al Hakim",  
  "username": "mahmud",  
  "email": "mahmud@example.com",  
  "phone": "076-1659999",  
  "company": {  
    "name": "Web Academy AB",  
    "address": "Hemfridsvägen, 192 67 Sollentuna",  
    "website": "http://webacademy.se"  
  }  
}
```

Hämta JSON via metoden getJSON()

```
$ .getJSON("demo.json", function(response){  
    console.log(response);  
  
    $("#demo").html(response.name + '<br>' +  
        response.phone );  
});
```

Dog API

<https://dog.ceo/dog-api>



Dog API

Documentation

Breeds list

About

ENDPOINTS

[List all breeds](#)

[Random image](#)

[By breed](#)

LIST ALL BREEDS

<https://dog.ceo/api/breeds/list/all>

Dog API - LIST ALL BREEDS

<h2>LIST ALL BREEDS</h2>

<ol id="breeds">

```
$(document).ready(function() {
  $.getJSON("https://dog.ceo/api/breeds/list/all",
    function(data) {
      $.each(data.message, function(index, array) {
        $("#breeds").append("<li>" + index + "</li>");
      });
    });
});
```

Övning

- Skapa en webbsida som visar en slumpmässig hund från <https://dog.ceo/dog-api>

Random image from Dog API



Metoden ajax()

```
$ajax({  
    url: "demo.json",  
    dataType: "json",  
    success: function(data) {  
        console.log(data);  
    }  
});
```

https://www.w3schools.com/jquery/ajax_ajax.asp

RANDOM USER GENERATOR

[HTTPS://RANDOMUSER.ME](https://randomuser.me)

RANDOM USER GENERATOR

A free, [open-source](#) API for generating random user data. Like Lorem Ipsum, but for people.

 Follow us @randomapi



Hi, My name is
Milton Hunter



Övning

- Skapa en webbsida som visar en slumpmässig person från
<https://randomuser.me>
- Visa bild, namn, ålder och city.
- Skapa en knapp som länkar till e-postadressen.



Bérénice Brun

Age: 37

City: AULNAY-SOUS-BOIS

Contact Me

Fetch API

“The Fetch API provides an interface for fetching resources (including across the network). It will seem familiar to anyone who has used XMLHttpRequest, but the new API provides a more powerful and flexible feature set.”

[https://developer.mozilla.org/en-US/docs/Web/API/Fetch API](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API)

Fetch – Exempel

```
fetch('demo.json')
  .then(resp => resp.json())
  .then(data => console.log(data))
  .catch(err => console.error(err))
```

- Fetch returnerar en Promise (överkurs)
- Tips: <https://javascript.info/promise-basics>

Arrow functions

```
resp => resp.json()
```

- Arrow-operatorn används för att skapa funktioner i JavaScript.
- Arrow-syntaxen är enklare och kortare jämfört med standard syntax.

```
function( resp){  
    return resp.json();  
}
```

Arbeta med REST API

<https://jsonplaceholder.typicode.com>

JSONPlaceholder

Fake Online REST API for Testing and Prototyping

Serving ~200M requests per month

Resources (endpoints)

- <https://jsonplaceholder.typicode.com/posts>
- <https://jsonplaceholder.typicode.com/comments>
- <https://jsonplaceholder.typicode.com/albums>
- <https://jsonplaceholder.typicode.com/photos>
- <https://jsonplaceholder.typicode.com/todos>
- <https://jsonplaceholder.typicode.com/users>

Hämta alla användare från JSONPlaceholder

```
fetch('https://jsonplaceholder.typicode.com/users')
  .then(resp => resp.json())
  .then(json => console.log(json))
  .catch(err => console.error(err))
```

Övning 1 – Skriv ut en lista över alla användare från JSONPlaceholder med hjälp av fetch()

Leanne Graham

Ervin Howell

Clementine Bauch

Patricia Lebsack

Chelsey Dietrich

Mrs. Dennis Schulist

Kurtis Weissnat

Nicholas Runolfsdottir V

Glenna Reichert

Clementina DuBuque

Övning 2 – Skriv ut en tabell över namn och e-post

Name	Email
Leanne Graham	Sincere@april.biz
Ervin Howell	Shanna@melissa.tv
Clementine Bauch	Nathan@yesenia.net
Patricia Lebsack	Julianne.OConner@kory.org
Chelsey Dietrich	Lucio_Hettinger@annie.ca
Mrs. Dennis Schulist	Karley_Dach@jasper.info
Kurtis Weissnat	Telly.Hoeger@billy.biz
Nicholas Runolfsdottir V	Sherwood@rosamond.me
Glenna Reichert	Chaim_McDermott@dana.io
Clementina DuBuque	Rey.Padberg@karina.biz

Övning 3 – Skriv ut adressetiketter

Leanne Graham

Kulas Light
Apt. 556
92998-3874
Gwenborough

Ervin Howell

Victor Plains
Suite 879
90566-7771
Wisokyburgh

Clementine Bauch

Douglas Extension
Suite 847
59590-4157
McKenziehaven

Patricia Lebsack

Hoeger Mall
Apt. 692
53919-4257
South Elvis

Chelsey Dietrich

Skiles Walks
Suite 351
33263
Roscoeview

Mrs. Dennis Schulist

Norberto Crossing
Apt. 950
23505-1337
South Christy

Övning 4 – Skapa en kopia av följande webbsida

<https://dog.ceo/dog-api/breeds-list>

https://dog.ceo/api/breed/ /images/random Fetch!



Summering av dagens lektion

- Ajax via jQuery
- Fetch API
- Praktiska övningar
- **Att läsa:** 388–408
- Reflektioner kring dagens lektion?

Framåtblick inför nästa lektion

- Under nästa lektion kommer vi att jobba med några JavaScript API:er
 - Feature Detection med Modernizr
 - Geolocation API
 - Web Storage API