

Национална програма "Обучение за ИТ умения и кариера" https://it-kariera.mon.bg/e-learning/



AJAX

Asynchronous JavaScript and XML

Съдържание

- 1. Какво e AJAX?
 - 1. XMLHttpRequest
- 2. Какво e Fetch?
 - 1. GET заявка
 - 2. POST заявка



Какво е АЈАХ?

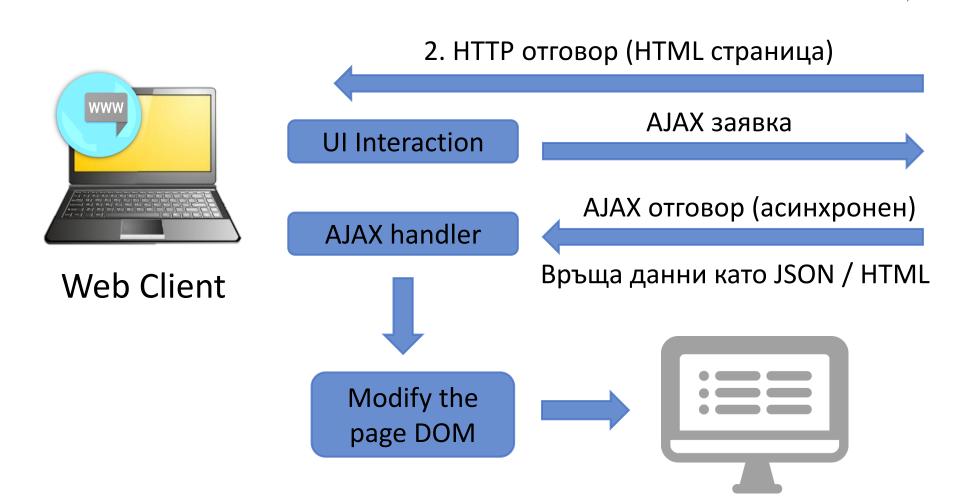
Asynchronous JavaScript And XML

Какво е АЈАХ?

- Зареждане на заден план на динамично съдържание / данни
- Зарежда данни от уеб сървъра и ги визуализира
- Два вида АЈАХ
 - Частично изобразяване на страница
 - Зарежда HTML фрагмент + показва го в <div>
 - JSON yczysa
 - Зарежда JSON обект и го показва с JS / jQuery

АЈАХ: Работен процес

1. НТТР заявка (първоначално зареждане на страницата)





XMLHttpRequest стандартен API за AJAX

```
<button onclick="loadRepos()">Load Repos</button>
<div id="res"></div>
```

```
function loadRepos() {
  const req = new XMLHttpRequest();
  req.onreadystatechange = function() {
   if (this.readyState == 4 && this.status == 200)
      document.getElementById("res").textContent =
        this.responseText;
  req.open("GET",
    "https://api.github.com/users/testnakov/repos", true);
 req.send();
```

Kaкво e Fetch? Fetch API

Какво e Fetch?

- Методът fetch() позволява извикване на уеб заявки
- Той е подобен на XMLHttpRequest (XHR). Основната разлика е, че Fetch API:
 - Използва обещания (Promises)
 - E no-npocm и чист API

```
fetch('./api/some.json')
   .then(function(response) {...})
   .catch(function(err) {...})
```

Проста Request Заявка

- Отговорът на заявката от fetch() e Stream обект
- Четенето на потока се случва асинхронно
- Когато се извика методът json(), се връща обещание
 - Статуса на отговора се проверява (трябва да е 200), преди да се преобрази отговора в JSON

```
if (response.status !== 200) {
   // handle error
}
response.json()
   .then(function(data) { console.log(data)})
```

GET Заявка

• GET заявката с Fetch API изглежда така:

```
fetch('https://swapi.co/api/people/4')
  .then((response) => response.json())
  .then((data) => console.log(JSON.stringify(data)))
  .catch((error) => console.error(error))
```

POST Заявка

• За да направим POST заявка, можем да зададем параметри на метода и тялото в опциите на fetch()

```
fetch('/url', {
   method: 'post',
   headers: { 'Content-type': 'application/json' },
   body: JSON.stringify(data),
})
```

Методи на Тялото

- clone() създаване на копие на отговора
- json() разрешава обещания с JSON
- redirect() създаване на ново обещание, но с различен URL адрес
- text() решава обещанието със низ

Задача: Load GitHub Repos with AJAX

```
GitHub username:
<input type="text" id="username" value="dimitarminchev" />
<button onclick="loadRepos()">Load Repos</button>
```

GitHub username: dimitarminchev Load Repos

- dimitarminchev/ambie
- dimitarminchev/BFU-ROVER
- <u>dimitarminchev/codingburgas.bg</u>
- dimitarminchev/CPC
- dimitarminchev/CS104
- <u>dimitarminchev/DCPA</u>
- dimitarminchev/dimitarminchev
- dimitarminchev/HAS8266
- dimitarminchev/Hexamite
- dimitarminchev/HEXRO
- dimitarminchev/ITCareer
- dimitarminchev/LSP
- dimitarminchev/MIRROR
- dimitarminchev/ROBO-FABRIC
- dimitarminchev/SPOJ0

Peшение: Load GitHub Repos with AJAX [1/2]

```
function loadRepos() {
      const repos = document.getElementById("repos");
      repos.innerHTML = '';
      const username = document.getElementById("username").value;
      const url = `https://api.github.com/users/${username}/repos`;
      fetch(url)
          .then((response) => response.json())
          .then((data) => displayRepos(data))
          .catch((err) => displayError(err))
      // Continues on the next slide...
```

Peшение: Load GitHub Repos with AJAX [2/2]

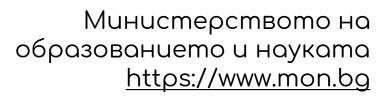
```
function createRepo(name, url) {
  // TODO: Implement
function displayRepos(repoItems) {
  repoItems.forEach(repo => {
    const { full_name, html_url } = repo;
    const repoItem = createRepo(full_name, html_url);
    repos.appendChild(repoItem)}
  });
function displayError(err) {
  const listItem = document.createElement('li');
  listItem.textContent = err;
  repos.appendChild(listItem); }}
```

Обобщение

- 1. Какво e AJAX?
 - 1. XMLHttpRequest
- 2. Какво e Fetch?
 - 1. GET заявка
 - 2. POST заявка



Национална програма "Обучение за ИТ умения и кариера" https://it-kariera.mon.bg







Документът е разработен за нуждите на Национална програма "Обучение за ИТ умения и кариера" на Министерството на образованието и науката (МОН) и се разпространява под свободен лиценз СС-ВҮ-NС-SA (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).