HTTP

1. Med callback
    - XMLHttpReques – i browser
    - request – på server (Node.js)
2. Med promise
    - fetch – ny standard i browser [1]
    - fetch – ej standard i Node.js
      brug npm modul 'node-fetch' for samme funktionalitet

[1] https://developer.mozilla.org/en-US/docs/Web/API/Fetch*API/Using*Fetch

Eksempel: `XMLHttpRequest.js` (browser)

```
let usersUrl = 'https://jsonplaceholder.typicode.com/users';
// let usersUrl = 'https://jsonplaceholder.typicode.com/usersX';
// let usersUrl = 'httpsX://jsonplaceholder.typicode.com/users';

let xhr = new XMLHttpRequest();
xhr.responseType = "json";
let async = true;
xhr.open('GET', usersUrl, async);
xhr.onreadystatechange = function () {
    if (xhr.readyState === XMLHttpRequest.DONE)
        if (xhr.status >= 400 || xhr.status === 0)
            console.log('Fejl: ' + xhr.status);
        else {
            let resultat = xhr.response;
            console.log('Resultat: ' + resultat[1].name);
        }
};
xhr.send();
```

Eksempel: `fetch.js (browser)

```
// const fetch = require('node-fetch');

let usersUrl = 'https://jsonplaceholder.typicode.com/users';
// let usersUrl = 'https://jsonplaceholder.typicode.com/usersX';
// let usersUrl = 'httpsX://jsonplaceholder.typicode.com/users';

fetch(usersUrl)
    .then(response => {
        if (response.status >= 400)
            throw new Error(response.status);
        else
            return response.json();
    })
    .then(resultat => console.log('Resultat: ' + resultat[1].name))
    .catch(fejl => console.log('Fejl: ' + fejl));
```

Eksempel: `fetch GET.js`

```
const fetch = require('node-fetch');

let url = 'https://jsonplaceholder.typicode.com/posts';

fetch(url)
    .then(response => {
        if (response.status >= 400)
            throw new Error(response.status);
        else
            return response.json();
    })
    .then(resultat => console.log(`Resultat: %o`, resultat))
    .catch(fejl => console.log('Fejl: ' + fejl));
```

Eksempel: fetch GET async.js

```
const fetch = require('node-fetch');

let url = 'https://jsonplaceholder.typicode.com/posts';

async function main() {
    try {
        let resultat = await fetch(url)
            .then(response => {
                if (response.status >= 400)
                    throw new Error(response.status);
                else
                    return response.json();
            });

        console.log(`Resultat: %o`, resultat);
    } catch (fejl) {
        console.log('Fejl: ' + fejl);
    }
}
main();
```

Eksempel: fetch POST.js

```
const fetch = require('node-fetch');

let url = 'https://jsonplaceholder.typicode.com/posts';
let data = { userId: 111, id: 101, title: 'titel', body: 'tekst' };

fetch(url, {
        method: "POST",
        body: JSON.stringify(data),
        headers: { 'Content-Type': 'application/json' }
    })
    .then(response => {
        if (response.status >= 400)
            throw new Error(response.status);
        else
            return response.json();
    })
    .then(resultat => console.log(`Resultat: %o`, resultat))
    .catch(fejl => console.log('Fejl: ' + fejl));
```

Eksempel: fetch PUT.js

```
const fetch = require('node-fetch');

let url = 'https://jsonplaceholder.typicode.com/posts/11';
let data = { userId: 111, title: 'titel', body: 'tekst' };

fetch(url, {
    method: "PUT",
    body: JSON.stringify(data),
    headers: { 'Content-Type': 'application/json' }
})
    .then(response => {
        if (response.status >= 400)
            throw new Error(response.status);
        else
            return response.json();
    })
    .then(resultat => console.log(`Resultat: %o`, resultat))
    .catch(fejl => console.log('Fejl: ' + fejl));
```

Eksempel: fetch DELETE.js

```
const fetch = require('node-fetch');

let url = 'https://jsonplaceholder.typicode.com/posts/11';

fetch(url, {
        method: "DELETE",
    })
    .then(response => {
        if (response.status >= 400)
            throw new Error(response.status);
        else
            return response.json();
    })
    .then(resultat => console.log(`Resultat: %o`, resultat))
    .catch(fejl => console.log('Fejl: ' + fejl));
```