

FETCH API

@ANDREICEK

WHAT'S FETCH?

- » a new way of doing network requests
 - » so you can stop using third party libds for network
 - » so you can stop using XMLHttpRequest
- » it supports streaming APIs

```
fetch(url, {  
  method: 'GET', // POST, PUT, DELETE, ...  
  body: 'string',  
  headers: {  
    'x-clacks-overhead': 'GNU Terry Pratchett'  
  },  
}); // returns a promise that will resolve with a Response object
```

Full documentation: <https://goo.gl/5WN11P>

USAGE: GET REQUEST

```
fetch('https://httpbin.org/get')  
  .then((res) => res.json()) // json() method returns a promise  
  .then((data) => console.log(data));
```

USAGE: POST REQUEST

```
fetch('https://httpbin.org/post', {  
  method: 'POST',  
  body: JSON.stringify({msg: 'hello'}),  
}).then(() => console.log('Done! 🎉'));
```

ERRORS

- » `fetch` will not throw on HTTP errors (404, 500, etc.)
- » it only throws on network errors, so check the status on Response object

```
fetch('https://httpbin.org/404')
  .then((res) => {
    if (!res.ok) { // also res.status for code
      throw new Error(res.statusText);
    }
    return res.json();
  })
  .catch((e) => console.log('😞', e.message));
```

WHY?

- » so you don't include jQuery on projects where you don't need it
- » so you don't use promise based HTTP clients (axios)
- » faster requests
- » resuability (with react-native e.g.)
- » better control over caching

SUPPORT

- » Chrome >42
- » Edge >14 (see note)
- » Firefox >34
- » IE no support (see note)
- » Opera >29
- » Safari >10.1

NOTE

For some browsers you will need a polyfill. The good thing is that the polyfill is `small` (7 KB), and it only activates when a browser doesn't have `fetch`.

» <https://github.com/github/fetch>

» <https://github.com/matthew-andrews/isomorphic-fetch>

» <https://github.com/bitinn/node-fetch>

TNXX



ANDREI.IM <-- SEE FOR SLIDES
@ANDREICEK