AJAX

by Mykola Sotula

CAN REACT DO THIS?

React itself doesn't have any allegiance to any particular way of fetching data.

In fact, as far as React is concerned, it doesn't even know there's a "server" in the picture at all.

React simply renders components, using data from only two places: **props** and **state**.

So therefore, to use some data from the server, you need to get that data into your components' props or state.

You can complicate this process with services and data models (er, "build abstractions") as much as you desire, but ultimately it's just components rendering props and state.

soft**serve**

HTTP LIBRARIES

- Axios
- **Superagent**
- **Fetch**
- **Etc.**

softserve

AXIOS

Axios is a very popular JavaScript library you can use to perform HTTP requests, that works in both Browser and Node.js platforms.

Using Axios has quite a few advantages over the native Fetch API:

- supports older browsers (Fetch needs a polyfill)
- has a way to abort a request
- has a way to set a response timeout
- has built-in CSRF protection
- supports upload progress
- performs automatic JSON data transformation
- works in Node.js

INSTALLATION

```
$ npm install axios
```
Or

Using jsDelivr CDN:

```
<script src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js"></script>
```

Using unpkg CDN:

```
<script src="https://unpkg.com/axios/dist/axios.min.js"></script>
```

API

You can start an HTTP request from the axios object:

This returns a promise. You can use async/await to resolve that promise to the response object:

```javascript
(async () => {
    const resp = await axios({
        url : 'https://swapi.dev/api/people/3/',
        method : 'get'
    });

    console.log(resp);
})();
```

softserve

API METHODS

For convenience, you will
generally use the methods

- axios.get()
- axios.post()
- axios.delete()
- axios.put()
- axios.patch()
- axios.options()

```
(async () => {
  const resp = await axios.get('https://swapi.dev/api/people/3/');
  console.log(resp);
})();
```

softserve

API PROMISE SYNTAX

If you don't want to use async/await you can use the
Promises syntax:

```
(async () => {
  axios.get('https://swapi.dev/api/people/3/')
    .then(resp => console.log(resp.data));
})();
```

softserve

API GET PARAMETERS

With Axios you can perform this by using that URL

```javascript
(async () => {
  axios.get('https://randomuser.me/api/?results=5000')
    .then(resp => console.log(resp.data));
})();
```

or you can use a params property in the options:

```javascript
axios.get('https://randomuser.me/api/', {
  params : {
    results : 5000
  }
})
  then(resp => console.log(resp.data));
```

softserve

API POST REQUESTS

Performing a POST request is just like doing a GET request, but instead of axios.get, you use axios.post:

```javascript
(async () => {
  axios.post('https://somesite.com', {
    title : 'Title of data',
    someData : [1, 2, 3, 4, 5]
  })
    .then(resp => console.log(resp.data));
})();
```

softserve

API ERROR HANDLING

Work with errors:

```javascript
(async () => {
  axios.get('https://randomuser.me/11api/', {
    params : {
      results : 5000
    }
  })
    .then(resp => console.log(resp.data))
    .catch(error => console.error('Look here', error));
})();
```

softserve