|  |
| --- |
| Axios Tutorial |

**Axios Tutorial**

**React Course**

[My React Course](https://www.udemy.com/course/react-tutorial-and-projects-course/?referralCode=FEE6A921AF07E2563CEF)

**Support**

Find the App Useful? [You can always buy me a coffee](https://www.buymeacoffee.com/johnsmilga)

**Docs**

[Axios Docs](https://axios-http.com/docs/intro)

**Install**

npm install axios

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

**First Request**

* import axios
* axios.get(url)
* axios.post(url)
* axios.patch/put(url)
* axios.delete(ulr)
* default get axios(url)
* returns a promise
* response data located in data property
* error in error.response

import axios from 'axios';

const fetchData = async () => {

try {

// axios.get(), axios.post(),axios.put(), axios.delete()

const response = await axios(url);

console.log(response);

} catch (error) {

console.log(error.response);

}

};

**Headers**

* second argument
* axios.get(url,{})
* third argument in requests with data
* axios.post(url,{data},{})

const fetchDadJoke = async () => {

try {

const { data } = await axios(url, {

headers: {

Accept: 'application/json',

},

});

// console.log(data);

setJoke(data.joke);

} catch (error) {

console.log(error.response);

}

};

**Post Request**

* send data to the server
* axios.post(url, { data })
* more options (auth header) - axios.post(url, { data },{})

try {

const resp = await axios.post(url, { data });

} catch (error) {

console.log(error.response.data);

}

**Global Defaults**

axios.defaults.headers.common['Accept'] = 'application/json';

axios.defaults.baseURL = 'https://api.example.com';

axios.defaults.headers.common['Authorization'] = AUTH\_TOKEN;

axios.defaults.headers.post['Content-Type'] =

'application/x-www-form-urlencoded';

**Custom Instance**

const authFetch = axios.create({

baseURL: 'https://course-api.com',

headers: {

Accept: 'application/json',

},

});

**Interceptors**

* global and custom

authFetch.interceptors.request.use(

(request) => {

request.headers.common['Accept'] = `application/json`;

console.log('request sent');

// must return request

return request;

},

(error) => {

return Promise.reject(error);

}

);

authFetch.interceptors.response.use(

(response) => {

console.log('got response');

return response;

},

(error) => {

console.log(error.response);

if (error.response.status === 404) {

// do something

console.log('NOT FOUND');

}

return Promise.reject(error);

}

);