**24.Axios 跨域和封装**

学习要点：

1. axios 跨域
2. axios 封装

本节课我们来开始了解下在 Vue 引入 Axios 后，跨域和封装的问题。**一．axios 跨域**

# 注意：课程中用极简的方式引出问题和解决，并不去扩展更多细节；

1. 而更多的细节，将会在实例和项目中衍生，这样 ajax 基础不足同学能抗住；
2. 首先，mock/axios 有一套 7 课时的微课程，推荐在 vue 核心篇后学；
3. 所以，这里就讲跨域和封装，不去探讨 axios 本身了，先在脚手架安装；

npm install axios

# 我这里提供两个远程的 json 数据，一个在服务器端设置可跨域，一个不可；

https://cdn.liyanhui.com/data.json (可跨域，设置过)

https://cdn.ycku.com/data.json (不可跨域，默认)

1. 可跨域，那处理起来就非常简单了，直接加载数据即可；

import ***axios*** from 'axios'

getName() {

***axios***.get('https://cdn.liyanhui.com/data.json').then(res => {

***console***.log(res.data)

this.name = res.data[**0**].username

})

}

PS：如果把 liyanhui 改成ycku，则无法获取数据，不可跨域；

# 创建 vue.config.js 文件，设置一下跨域代理；

module.exports = { devServer : {

proxy : 'https://cdn.ycku.com/'

}

}

***axios***.get('/data.json').then(res => {...

**二．axios 封装**

# 首先，为何要封装？因为直接使用 axios，每个组件都要重复配置各种参数；

1. 虽然，我们的例子中一个参数都没配置，你可以想象一下，重复会很多；
2. 第二，方便后期维护，比如要更改某个参数，甚至换掉 axios，只要改一次；
3. 在根目录创建 http/api.js，用于封装 axios；

import ***axios*** from 'axios'

//或者 export function get(url) export const *get* = (url) => {

return new ***Promise***((resolve, reject) => {

***axios***.get(url).then(res => { resolve(res.data);

}).catch(err => { reject(err.data)

})

})

}

import {get} from '@/http/api.js'

getName() { get('/data.json').then(data => {

this.name = data[**0**].username

})

}