Ajax与Axios

1. 认识Ajax

#### ajax简介

Ajax 即“Asynchronous Javascript And XML”（异步 JavaScript 和 XML），是指一种创建交互式网页应用的网页开发技术。

Ajax = 异步 JavaScript 和 XML（标准通用标记语言的子集）。

Ajax 是一种用于创建快速动态网页的技术。

Ajax 是一种在无需重新加载整个网页的情况下，能够更新部分网页的技术。

通过在后台与服务器进行少量数据交换，Ajax 可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。

传统的网页（不使用 Ajax）如果需要更新内容，必须重载整个网页页面。

$.ajax({

url: 'http://jsonplaceholder.typicode.com/users',

type: 'get',

dataType: 'json',

data: {

//'a': 1,

//'b': 2,

},

success: function (response) {

console.log(response)；

}

})

1. axios

用于浏览器和node.js的基于Promise的HTTP客户端

1. 从浏览器制作XMLHttpRequests

2. 让HTTP从node.js的请求

3. 支持Promise API

4. 拦截请求和响应

5. 转换请求和响应数据

6. 取消请求

7. 自动转换为JSON数据

8. 客户端支持防止XSRF

axios({

url: 'http://jsonplaceholder.typicode.com/users',

method: 'get',

responseType: 'json', // 默认的

data: {

//'a': 1,

//'b': 2,

}

}).then(function (response) {

console.log(response);

console.log(response.data);

}).catch(function (error) {

console.log(error);

})

1. 原生js实现ajax请求

#### Get请求

// 1.创建一个XMLHttpRequest的对象.

var xml=null; //初始值设为空

if(XMLHttpRequest){

xml=new XMLHttpRequest; //兼容非IE6

}else{

xml=new ActiveXObject('Microsoft.XMLHTTP'); //兼容IE6浏览器

}

//2.通过open与服务器建立连接 open(method,url,async) ;

//method包含 GET、POST方式。

//第三个参数同步(false)或者异步(true)

xml.open('GET',url,true);

//3.发送请求 send(string)

//string(参数) 仅用于post请求，GET请求不需要传参数，如果有参数直接拼接到url中。

xml.send();

//4.状态和响应

xml.onreadystatechange=function(){

//当readyState==4并且status==200时请求成功,否则请求失败

if(xml.readyState==4&&xml.status==200){

//请求成功

}else{

//请求失败

}

}

#### Post请求

var xml=null;

var data={a:1,b:2};

if(XMLHttpRequest){

xml=new XMLHttpRequest;

}else{

xml=new ActiveXObject('Microsoft.XMLHTTP')

}

xml.open('POST',url,true);

xml.send(data); //这里的data是要传递的参数

xml.onreadystatechange=function(){

if(xml.readyState==4&&xml.status==200){

//请求成功

}else{

//请求失败

}

}

1. Axios的使用

首先安装axios,

方法一：npm install axios

方法二： CDN引入 <script src="https://unpkg.com/axios/dist/axios.min.js"></script>

get请求：

//1.get请求(无参数)

axios.get('http://www.xxx')

.then(function(response){

//请求成功

}).catch(function(erroe){

//请求失败

});

//2.get请求(有参数)

axios.get('http://www.xxx?a=1&b=2')

.then(function(response){

//请求成功

}).catch(function(erroe){

//请求失败

});

　　post请求：

//必须引入qs对data进行stringify 安装axios时已经安装了qs,无需再安装，引入即可用。

// import Qs from 'qs'

let data=Qs.stringify({a:1,b:2});

axios.post('http://www.xxx',data)

.then(function(response){

//请求成功

}).catch(function(error){

//请求失败

})

//4.多个请求同时发送

function axiosOne(){

return axios.get('http://www.url.one')

};

function axiosTwo(){

return axios.get('http://www.url.two')

};

axios.all([axiosOne(),axiosTwo()])

.then(axios.spread(function(acct, perms){

console.log(acct);//请求一的结果；

console.log(perms);//请求二的结果

}))