# AJAX

## 什么是 AJAX ？

AJAX = Async JavaScript and XML。

AJAX 是一种用于创建快速动态网页的技术。

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，必需重载整个网页面。

# AJAX XHR

## XHR过程

XMLHttpRequest 是 AJAX 的基础。

XMLHttpRequest对象是一个javascript对象，它类似于一个小线程，在不刷新网页的前提下运行并于后端交互。

XML: 一种可以扩展的文本标记语言，可以扩展自定义的语义标签。很早以前 xml 常用于从服务端返回数据结构.

目前主要用json格式返回数据（后来居上，更简便）

### 创建XHR

var xhrName = new XMLHttpRequest();//支持IE7+

### 设置XHR

xhrName.open("GET","test1.txt",true);

**open(method,url,async,username,userpassword)**

//**规定请求的类型**、URL 以及是否异步处理请求。method：请求的类型；主要是GET 或 POST，url：文件在服务器上的位置(后端提供定义)；async：true（异步）或 false（同步）。

当async==true时，将该任务加入宏任务

**当async==false时，同步执行，立即执行且send之后readystate==4**

### 发送XHR

xhrName.send();

**xhrName .send(string)**

**将请求发送到**服务器。string：只有在用"POST"，并且需要发送参数的时候，string才有意义

xmlhttp.send("user="+username+"&password="+password)

### XHR的事件监听

xhrName. onreadystatechange = function(){

//设置如何处理接收的response

}

## XHR 响应

response：响应的主体内容

responseText 获得字符串形式的响应数据。

responseXML 获得 XML 形式的响应数据。

### onreadystatechange 事件

当请求被发送到服务器时，我们需要执行一些基于响应的任务。每当 readyState 改变时，就会触发 onreadystatechange 事件。readyState 属性存有 XMLHttpRequest 的状态信息。

下面是 XMLHttpRequest 对象的三个重要的属性：

**onreadystatechange** 存储函数（或函数名），每当 readyState 属性改变时，就会调用该函数。

**readyState** 存有 XMLHttpRequest 的状态。从 0 到 4 发生变化。

状态 0：unsent，刚创建的 XMLHttpRequest 实例，还没有发送。

状态 1：（载入）已调用 send() 方法，正在发送请求。

**状态 2：（载入完成）send() 方法执行完成，已经接收到全部响应内容**

状态 3：loading，（交互）正在解析响应内容

状态 4：done，表示响应的主体内容解析完成，可以在客户端调用了

**status**

http状态码，如200: "OK"；404: 未找到页面。

当 readyState 等于 4 且状态为 200 时，表示响应已就绪,已接收到服务端返回的响应。

xmlhttp.onreadystatechange=function()

{

if (xmlhttp.readyState==4 && xmlhttp.status==200)

{

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;

}

}

onabort: 表示请求中断后要处理的事。和 xhr.abort() 一起使用。

ontimeout: 表示请求的超时，执行的方法。和timeout设定的事件一起使用。

response: 响应的主体内容。

responseText: 响应的具体内容是字符串，一般是 json 字符串

responseXML: 响应的具体内容是文档。

status: http 的状态码。

statusText: 状态码描述

withCredentials：表示是否允许跨域。

getAllResponseHeaders：获取所有响应头信息。

xhr.open()：打开URL请求。

xhr.send()：表示发送 ajax。

setRequestHeader(): 设置请求头。这个属性在必须在xhr.open()后面。

# axios

axios 是使用 promise 封装的 ajax。axios 不是一个类是一个方法。

axios.get().then().catch()的方式，也可axios({//set axios }).then()

# fetch

fetch是ES8中新增的api，是 http 的数据请求方式，是 XMLHttpRequest 的一种代替方案，