

Ajax定义

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。（偷偷请求服务器）

Ajax作用

实现页面的无刷新请求服务器,提高用户体验

使用 JavaScript 向服务器提出请求并处理响应而不阻塞用户核心对象XMLHttpRequest。通过这个对象，您的 JavaScript 可在不重载页面的情况与 Web 服务器交换数据，即在不需要刷新页面的情况下，就可以产生局部刷新的效果。

Ajax 在浏览器与 Web 服务器之间使用异步数据传输（HTTP 请求），这样就可使网页从服务器请求少量的信息，而不是整个页面。

参考链接：

<https://developer.mozilla.org/zh-CN/docs/Web/API/XMLHttpRequest>

基本使用

1.创建ajax对象

```
new XMLHttpRequest()  
new ActiveXObject('Microsoft.XMLHTTP')
```

2.创建事件函数(处理服务器返回的结果)

```
onreadystatechange H5 onload  
readyState ajax对象的状态  
    0：请求未初始化  
    1：服务器连接已建立  
    2：请求已接收  
    3：请求处理中  
    4：请求已完成，且响应已就绪  
status 响应码  
    200："OK"  
    404：未找到页面  
    403：权限不足  
responseText 接受服务器返回的数据
```

```
var xhr = new XMLHttpRequest();  
console.log(xhr.readyState); //0  
xhr.open("GET", "./test.php");  
console.log(xhr.readyState); //1  
xhr.send();  
xhr.onreadystatechange = function(){  
    //console.log(this.readyState);  
    switch(this.readyState){  
        case 2:
```

```

        //console.log(this.getAllResponseHeaders());
        console.log(this.getResponseHeader("server"));
        break;
    case 3:
        console.log(this.responseText);
        break;
    case 4:
        console.log(this.responseText);
        break;
    }
};

```

结论：状态值为4时 执行响应的后续逻辑

```

xhr.onreadystatechange = function(){
    if (this.readyState === 4) {
        //响应后续逻辑
    }
};

```

3.初始化ajax请求

```

get
    xhr.open('get', '1.php?a=1&b=2&c=3', true);
post
    xhr.setRequestHeader('content-type', 'application/x-www-form-urlencoded');
    xhr.open('post', '1.php', true);

```

4.发送ajax请求

```

send()

```

服务器返回

```

echo 普通的字符串
数组 json_encode($arr)转成json字符串 JS接收时使用JSON.parse(this.responseText);
html文件
xml字符串
<?php
header('Content-Type: application/xml');
?>
<?xml version="1.1" encoding="utf-8"?>
<person>
    <uname>小明</uname>
    <age>16</age>
    <grade>88</grade>
</person>
js接收 document
responseXML

```

注意事项

同源策略

post发送ajax请求

```
setRequestHeader('content-type','application/x-www-form-urlencoded');
```

封装Ajax

由于 `readystatechange` 事件是在 `xhr` 对象状态变化时触发（不单是在得到响应时），也就意味着这个事件会被触发多次，所以我们有必要了解每一个状态值代表的含义：

readyState	状态描述	说明
0	UNSENT	代理（XHR）被创建，但尚未调用 <code>open()</code> 方法。
1	OPENED	<code>open()</code> 方法已经被调用，建立了连接。
2	HEADERS_RECEIVED	<code>send()</code> 方法已经被调用，并且已经可以获取状态行和响应头。
3	LOADING	响应体下载中， <code>responseText</code> 属性可能已经包含部分数据。
4	DONE	响应体下载完成，可以直接使用 <code>responseText</code> 。

参考：http响应码

http状态返回代码 1xx（临时响应）

表示临时响应并需要请求者继续执行操作的状态代码。

http状态返回代码 代码 说明

100 （继续） 请求者应当继续提出请求。 服务器返回此代码表示已收到请求的第一部分，正在等待其余部分。

101 （切换协议） 请求者已要求服务器切换协议，服务器已确认并准备切换。

http状态返回代码 2xx （成功）

表示成功处理了请求的状态代码。

http状态返回代码 代码 说明

200 （成功） 服务器已成功处理了请求。 通常，这表示服务器提供了请求的网页。

201 （已创建） 请求成功并且服务器创建了新的资源。

202 （已接受） 服务器已接受请求，但尚未处理。

203 （非授权信息） 服务器已成功处理了请求，但返回的信息可能来自另一来源。

204 （无内容） 服务器成功处理了请求，但没有返回任何内容。

205 （重置内容） 服务器成功处理了请求，但没有返回任何内容。

206 （部分内容） 服务器成功处理了部分 GET 请求。

http状态返回代码 3xx （重定向）

表示要完成请求，需要进一步操作。 通常，这些状态代码用来重定向。

http状态返回代码 代码 说明

300（多种选择）针对请求，服务器可执行多种操作。服务器可根据请求者 (user agent) 选择一项操作，或提供操作列表供请求者选择。

301（永久移动）请求的网页已永久移动到新位置。服务器返回此响应（对 GET 或 HEAD 请求的响应）时，会自动将请求者转到新位置。

302（临时移动）服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。

303（查看其他位置）请求者应当对不同的位置使用单独的 GET 请求来检索响应时，服务器返回此代码。

304（未修改）自从上次请求后，请求的网页未修改过。服务器返回此响应时，不会返回网页内容。

305（使用代理）请求者只能使用代理访问请求的网页。如果服务器返回此响应，还表示请求者应使用代理。

307（临时重定向）服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。

http状态返回代码 4xx（请求错误）

这些状态代码表示请求可能出错，妨碍了服务器的处理。

http状态返回代码 代码 说明

400（错误请求）服务器不理解请求的语法。

401（未授权）请求要求身份验证。对于需要登录的网页，服务器可能返回此响应。

403（禁止）服务器拒绝请求。

404（未找到）服务器找不到请求的网页。

405（方法禁用）禁用请求中指定的方法。

406（不接受）无法使用请求的内容特性响应请求的网页。

407（需要代理授权）此状态代码与 401（未授权）类似，但指定请求者应当授权使用代理。

408（请求超时）服务器等候请求时发生超时。

409（冲突）服务器在完成请求时发生冲突。服务器必须在响应中包含有关冲突的信息。

410（已删除）如果请求的资源已永久删除，服务器就会返回此响应。

411（需要有效长度）服务器不接受不含有效内容长度标头字段的请求。

412（未满足前提条件）服务器未满足请求者在请求中设置的其中一个前提条件。

413（请求实体过大）服务器无法处理请求，因为请求实体过大，超出服务器的处理能力。

414（请求的 URI 过长）请求的 URI（通常为网址）过长，服务器无法处理。

415（不支持的媒体类型）请求的格式不受请求页面的支持。

416（请求范围不符合要求）如果页面无法提供请求的范围，则服务器会返回此状态代码。

417（未满足期望值）服务器未满足"期望"请求标头字段的要求。

http状态返回代码 5xx（服务器错误）

这些状态代码表示服务器在尝试处理请求时发生内部错误。这些错误可能是服务器本身的错误，而不是请求出错。

http状态返回代码 代码 说明

500 （服务器内部错误） 服务器遇到错误，无法完成请求。

501 （尚未实施） 服务器不具备完成请求的功能。例如，服务器无法识别请求方法时可能会返回此代码。

502 （错误网关） 服务器作为网关或代理，从上游服务器收到无效响应。

503 （服务不可用） 服务器目前无法使用（由于超载或停机维护）。通常，这只是暂时状态。

504 （网关超时） 服务器作为网关或代理，但是没有及时从上游服务器收到请求。

505 （HTTP 版本不受支持） 服务器不支持请求中所用的 HTTP 协议版本。

一些常见的http状态返回代码为：

200 - 服务器成功返回网页

404 - 请求的网页不存在

503 - 服务不可用