

## Ajax

笔记本: JavaScript

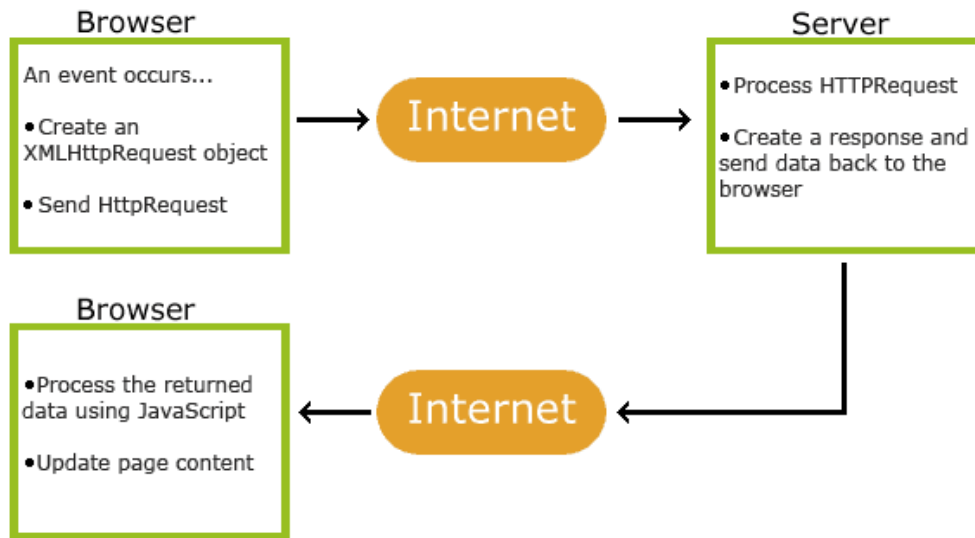
创建时间: 2019-06-05 10:07

更新时间: 2019-06-05 12:02

作者: whjthinking@163.com

URL: <http://www.runoob.com/ajax/ajax-xmlhttprequest-send.html>

### • Work Principle



### • XMLHttpRequest Object

- All Modern browsers (IE7+, Firefox, Chrome, Safari and Opera) support XMLHttpRequest Object (IE5 and IE6 use ActiveXObject)
- XMLHttpRequest is used to transfer data between backstage and server, which means it can update some part of the web pages without reloading the whole website.
- To create an XMLHttpRequest Object, you should use:

```
variable=new XMLHttpRequest(); //Old version Internet Explorer should use  
ActiveX object: variable=new ActiveXObject ("Microsoft.XMLHTTP")
```

```
var xmlhttp;  
if (window.XMLHttpRequest)  
{  
    xmlhttp=new XMLHttpRequest();  
}  
else  
{  
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");  
}
```

### • Send Request to servers

- we use the methods **open()** and **send()** of XMLHttpRequest Object to send request to servers.

```
xmlhttp.open("GET", "ajax_info.txt", true);
```

xmlhttp.send();	
method	describe
open(method, url, async)	set the type of the request, URL and whether it is a asynchronous processing of requests. method: the type of the request: GET or POST url: the directory of the file async: true (asynchronous) or false (synchronous)
send(string)	send the request to servers string: only used in POST request

### • GET or POST

- Compared with POST, GET is simpler and faster, and can be used in most cases.
- However, use POST request in the following cases:
  - unable to use cache files (update the files or databases in servers)
  - send a flood of data to servers (POST has no data limit)
  - when sending user input which includes unknown chars, POST is more stable and reliable than GET
- *GET Request*

```
xmlhttp.open("GET", "/try/ajax/demo_get2.php?
fname=Henry&lname=Ford", true);
xmlhttp.send();
```

### - POST Request

```
xmlhttp.open("POST", "/try/ajax/demo_post2.php", true);
xmlhttp.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");
xmlhttp.send("fname=Henry&lname=Ford");
```

### • Async=true

```
xmlhttp.onreadystatechange=function()
{
    if(xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.getElementById("myDiv").innerHTML=xmlhttp.responseText;
    }
}
xmlhttp.open("GET", "/try/ajax/ajax_info.text", true);
xmlhttp.send();
```

### • Onreadystatechange Event

attribute	describe
onreadystatechange	store function (or function name). The function will be called once the attribute readyState changes.
readyState	store XMLHttpRequest status. changes from 0 to 4: <ul style="list-style-type: none"> <li>• 0: didn't initialize request</li> <li>• 1: server connection established</li> </ul>

	<ul style="list-style-type: none"> <li>• 2: request is received</li> <li>• 3: request is processing</li> <li>• 4: request is finished, and response is ready</li> </ul>
status	200: "OK" 404: cannot find the page

- When readyState equals to 4 and status is 200, the response is ready:

```
xmlhttp.onreadystatechange=function()
{
    if(xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.getElementById("myDiv").innerHTML=xmlhttp.responseText;
    }
}
```

- **Server Response**

- use attributes responseText or responseXML to get XML HttpRequest Object

attribute	describe	example
responseText	get response data in string form	document.getElementById("myDiv").innerHTML=xmlhttp.responseText;
responseXML	get response data in XML form	xmlDoc=xmlhttp.responseXML; txt=""; x=xmlDoc.getElementsByTagName("ARTIST"); for (i=0; i<x.length; i++) { txt=txt + x[i].childNodes[0].nodeValue + " "; } document.getElementById("myDiv").innerHTML=txt;

