

```
1 function _Ajax(options) {
2     // 解构options
3     let {
4         method='get',
5         url='',
6         params=null,
7         requestHeader={},
8         timeout=null,
9         successCB=()=>{},
10        errorCB=()=>{}
11    } = options
12
13    let xhr = null
14    try {
15        xhr = new XMLHttpRequest()
16    } catch (e) {
17        xhr = new ActiveXObject('Microsoft.XMLHttp')
18    }
19
20    // 将{a: 1, b: 2}转换为a=1&b=2格式
21    function _transParams(params) {
22        let keys = Object.keys(params)
23        return keys.reduce((pre, cur, index) => {
24            const flag = index === keys.length - 1 ? '' : '&'
25            // get请求的value值需要用encodeURIComponent()方法进行编码
26            const value = encodeURIComponent(params[cur])
27            return `${pre}${cur}=${value}${flag}`
28        }, '')
29    }
30
31    // 如果传入timeout说明在该时间结束后要中断请求
32    let timer = null
33    if (timeout) {
34        timer = setTimeout(() => {
35            xhr.abort()
36            clearTimeout(timer)
37        }, timeout)
38    }
39
40    if (/^(get|head|option|delete)$/i.test(method)) { // 需要将params接在url后面
41        xhr.open(method, `${url}?${_transParams(params)}${new Date().getTime()}`, true)
42    } else { // 将data以请求体的形式发送
43        xhr.open(method, `${url}`, true)
44    }
45
46    xhr.onreadystatechange = function() {
47        if (xhr.readyState === 4 ) {
48            // 此处的try-catch是为了捕获请求超时，请求超时时readyState还是会变成4，但是status会报错
49            try {
50                if (xhr.status >= 200 && xhr.status < 300 || xhr.status === 304) {
51                    // 响应成功后可回调,且将timeout设置的定时器取消掉
52                    timer && clearTimeout(timer)
53                    successCB.call(this, xhr)
54                }
55            } catch(e) {
56                // 响应失败的回调
57                errorCB.call(this, xhr)
58            }
59        }
60    }
61
62    // 发送额外的请求头，请求头需要一个一个设置 是set set set! 不是send!
63    Object.keys(requestHeader).length && Object.keys(requestHeader).forEach(item => {
64        xhr.setRequestHeader(item, requestHeader[item])
65    })
66
67    // post类请求需要发送Content-type: application/x-www-form-urlencoded
68    if (/^(post|put)$/i.test(method)) {
69        xhr.setRequestHeader('Content-type', 'application/x-www-form-urlencoded')
70    }
71
72    // 此处的data初始值是null,如果是get类的请求，可以直接send(null),如果是post类的请求， send(params)
73    xhr.send(_transParams(params))
74 }
75
76 // 使用
77 _Ajax({
78     method: 'get',
79     url: 'http://localhost:1124/getInfo',
80     params: {name: 'lidan', age: 27},
81     // 由于设置简单请求以外的请求头，所以此次请求会先发送一次预检请求
82     requestHeader: {'x-csrf-token': '111dddaannn'},
83     timeout: 3000,
84     successCB: (xhr) => {
85         console.log(xhr, '=====' )
86     },
87     errorCB: (errXHR) => {}
88 })
```