在某个页面中出于某种目的需要向其他非同一域下的页面发送某个AJAX请求，例如：www.example1.com域下的网页上使用ajax请求www.example2.com域上的服务，这就是**跨域请求**，处于安全性考虑，浏览器通常会阻止跨域请求。

为了能够实现跨域请求，这里有两种做法，不过前提都需要后端的配合。

方法一：后端返回头部加上 Access-Control-Allow-Origin: \*

方法二：JSONP，只适用于GET请求

JSONP的原理：

<script>元素并没有同源限制，也就意味着一个域上的页面可以引用其他域上的js文件。比如最常见的cdn方式引用js库文件。

将一个跨域的服务url放到<script>元素的src属性值中，那么浏览器将自动向此src指定的服务发送一个get请求，由于浏览器预期返回数据为一段js代码，所以，接到响应数据之后，浏览器会将它直接放到js引擎中执行。

注意

<Script>元素的src的请求只有get方式，所以目标服务必须支持get方法请求。有的服务器只允许post或者其他http请求方法，这种服务是无法使用jsonp的方式进行跨域请求的。

服务端响应的内容必须是一段可执行的js代码，否则使用jsonp将意义不大。这就是很多人会问为什么使用jsonp接收一个格式正确的json数据，但是浏览器会报语法错误的原因。

方法

方法1：手动构建<script>元素，添加到document节点中。

方法2：使用jquery的ajax方式实现jsonp请求

例如:

获取百度词条前10推荐

<https://sp0.baidu.com/5a1Fazu8AA54nxGko9WTAnF6hhy/su?wd=b&cb=jshow> url示例

返回数据为

jshow({q:"b",p:false,s:["baidu","bt","btchina","beyond","bbs","bbc","blog","bobo组合","bb霜"]});

function jshow(data){

console.log(data.s);

return data.s;

}

$.ajax({

url : “https://sp0.baidu.com/5a1Fazu8AA54nxGko9WTAnF6hhy/su”,

data: {wd: ‘a’ },

dataType: ‘jsonp’,

jsonp: ‘cb’,

jsonpCallback: ‘jshow’,

type: ‘GET’,

success: function(result){

alert(result);

}

})