

最近在做网站的时候遇到这样一个功能，在如图所示的页面中，需要用户点击链接的时候，能够以异步Ajax的方式判断服务器中是否存储有相应的Excel文件，如果没有的话就提示用户没有找到，如果有的话就下载到用户本地。

Travel cost/visit

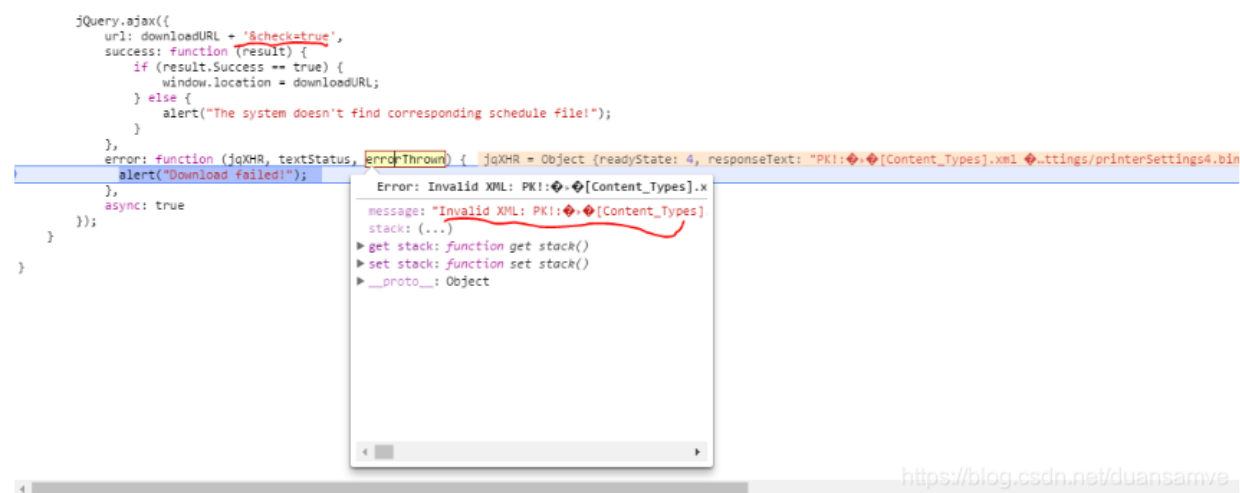
EUR

Parts discount

%

[Click to download the latest maintenance schedule file](#)

当然，这是很简单的一个问题，按照一般方式编写Ajax就可以了。但是当服务器端把文件内容以二进制的形式返回到浏览器端，浏览器的Ajax却抛出了错误。大致是ParseError, Invalid XML PK一类的错误信息。



造成这个问题的原因，并不是服务器端代码或者javascript代码有问题，而是通过Ajax下载文件的这种方式本来就是禁止的。出于安全因素的考虑，javascript是不能够保存文件到本地的，所以ajax考虑到了这点，只是接受xml, ajax, json格式的返回值，二进制的返回格式就会抛出这个异常。

因为response原因，一般请求浏览器是会处理服务器输出的response，例如生成png、文件下载等，然而ajax请求只是个“字符型”的请求，即请求的内容是以文本类型存放的。文件的下载是以二进制形式进行的，虽然可以读取到返回的response，但只是读取而已，是无法执行的，说白点就是js无法调用到浏览器的下载处理机制和程序。

文件下载的几种方式：

如何解决这个问题？

1、用window.location.href = url的方式：

有人会问，像上图这样的需求，在某个页面的时候点击下载链接，因为改变了window.location的值，岂不是当前页面就要被跳转了？事实是，我用的是Chrome浏览器，当点击那个link的时候，直接就弹出了文件保存的对话框，而且页面地址栏没有任何变化。这时候如果点击Save，文件就会保持，点击Cancel，操作就会取消，过程中当前页面会一直保持，不会跳转到其他页面。

如果一次执行多个location.href，实际后面的会将前面的替换掉，所以其实只能实现单个文件下载。

2、可以直接使用a标签实现浏览器文件下载：

```
<a href=" 下载地址" >点击下载</a>
```

或

```
var aLink = document.createElement('a');
aLink.download = "文件名";
aLink.href = "文件url地址";
document.body.appendChild(aLink);
aLink.click();
document.body.removeChild(aLink);
```

3、POST方式浏览器下载：可以使用jquery创建表单并提交实现文件下载：

```
var form = $("<form>");
form.attr("style","display:none");
form.attr("target","_blank");
form.attr("method","post");
form.attr("action",rootPath + "T_academic_essay/DownloadZipFile.do");
var input1 = $("<input>");
input1.attr("type","hidden");
input1.attr("name","strZipPath");
input1.attr("value",strZipPath);
$("body").append(form);
form.append(input1);
form.submit();
form.remove();
```

或

```
<button class="btn btn-default" id="download">下载</button>
```

```
function postExcelFile(params, url) {
    //params是post请求需要的参数，url是请求url地址
    const form = document.createElement("form");
    form.style.display = "none";
    form.action = url;
```

```

form.method = "post";
document.body.appendChild(form);
// 动态创建input并给value赋值
for (const key in params) {
    const input = document.createElement("input");
    input.type = "hidden";
    input.name = key;
    input.value = params[key];
    form.appendChild(input);
}

form.submit();
form.remove();
}

$("#download").click() => {
    postExcelFile({ currentPage: 2, pageSize: 20 },
        'url/xxxxxxx/');
})

```

4、get方式浏览器下载：使用隐藏iframe或新窗体解决：

```

export const downloadFile = (url) => {
    const iframe = document.createElement("iframe");
    iframe.style.display = "none"; // 防止影响页面
    iframe.style.height = 0; // 防止影响页面
    iframe.src = url;
    document.body.appendChild(iframe); // 这一行必须，iframe挂在到dom树上才会发请求
    // 5分钟之后删除（onload方法对于下载链接不起作用，就先抠脚一下吧）
    setTimeout(()=>{
        iframe.remove();
    }, 5 * 60 * 1000);
}

```

这个可以实现一次下载多个文件。

