

首先先了解一下http请求的构成

请求行、请求头(head)、空行、请求体(body)

请求行：

格式：[Method Request-URI HTTP-Version 结尾符

method一般为POST GET等

Request-URI 为请求的uri，一般的跟在url后面的参数也位于这里

HTTP-Version 一般有http1.1 http2.0等 结尾符一般用\r\n

请求头

<https://www.cnblogs.com/ulysses-you/p/7837173.html>

空行

空行是用来分割请求头和请求体的

请求体

一般用于http的post method。通过实体报头规定消息主体的格式内容、例如Content-Type=text/plain,该实体报头规定了消息主体的数据是纯文本格式

常见的请求体格式 (content-type) :

常见的媒体格式类型如下:

text/html : HTML格式

text/plain : 纯文本格式

text/xml : XML格式

image/gif : gif图片格式

image/jpeg : jpg图片格式

image/png : png图片格式

以application开头的媒体格式类型:

application/xhtml+xml : XHTML格式

application/xml : XML数据格式

application/atom+xml : Atom XML聚合格式

application/json : JSON数据格式

application/pdf : pdf格式

application/msword : Word文档格式

application/octet-stream : 二进制流数据 (如常见的文件下载)

application/x-www-form-urlencoded : <form encType="">中默认的 encType, form表单数据被编码为key/value格式发送到服务器 (表单默认的提交数据的格式)

另外一种常见的媒体格式是上传文件之时使用的:

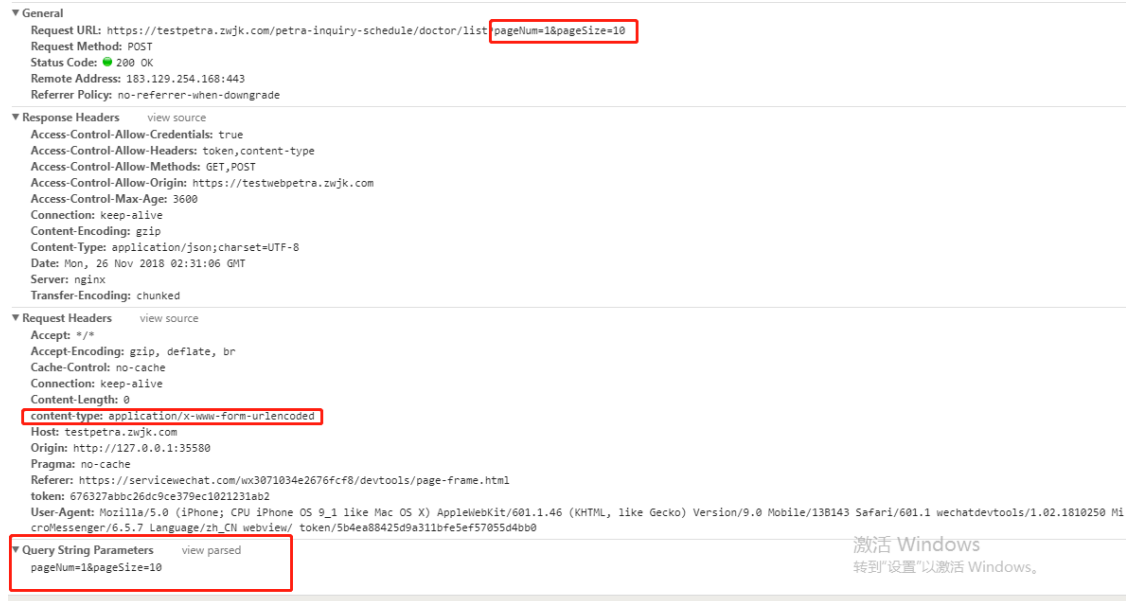
multipart/form-data : 需要在表单中进行文件上传时, 就需要使用该格式

下面开始介绍前后端请求的几种常见方式:

1. 参数跟在URL后面

这种情况参数是放在请求行里的，称为queryString.在控制台显示为 Query String Parameters.会被编码为key/value的格式跟在url的后面。GET请求也是这种方法。如果是POST请求，还把很长的参数放在url后面的话，虽然HTTP协议没有限制请求头的长度，但是很多浏览器都做了限制，所以，把content放在请求体里面，是不会有有限制的

(<https://www.cnblogs.com/ztiandan/archive/2013/04/16/3023758.html>)。



▼ General

Request URL: <https://testpetra.zwj.com/petra-inquiry-schedule/doctor/11st?pageNum=1&pageSize=10>

Request Method: POST

Status Code: 200 OK

Remote Address: 183.129.254.168:443

Referrer Policy: no-referrer-when-downgrade

▼ Response Headers

view source

Access-Control-Allow-Credentials: true

Access-Control-Allow-Headers: token,content-type

Access-Control-Allow-Methods: GET,POST

Access-Control-Allow-Origin: https://testwebpetra.zwj.com

Access-Control-Max-Age: 3600

Connection: keep-alive

Content-Encoding: gzip

Content-Type: application/json;charset=UTF-8

Date: Mon, 26 Nov 2018 02:31:06 GMT

Server: nginx

Transfer-Encoding: chunked

▼ Request Headers

view source

Accept: */*

Accept-Encoding: gzip, deflate, br

Cache-Control: no-cache

Connection: keep-alive

Content-Length: 0

Content-Type: application/x-www-form-urlencoded

Host: testpetra.zwj.com

Origin: http://127.0.0.1:35580

Pragma: no-cache

Referer: https://servicewechat.com/wx3071034e2676fcf8/devtools/page-frame.html

token: 676327abb26dc9ce379ec1021231ab2

User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13B143 Safari/601.1 wechatdevtools/1.02.1810250 MIcroMessenger/6.5.7 Language/zh-CN webview/ token/5b4ea88425d9a311bfe5ef57055d4bb0

▼ Query String Parameters

view parsed

pageNum=1&pageSize=10

激活 Windows
转到“设置”以激活 Windows。

2. content-type 为 application/x-www-form-urlencoded 格式

这种参数是放在请求体里面的。一般的默认请求就是这种情况，form表单如果没有指定encType，就会默认使用这种请求方式，数据会被编码为key/value的格式发送到服务器。在控制台显示的是Form Data



▼ Request Headers

view source

Accept: */*

Accept-Encoding: gzip, deflate, br

Cache-Control: no-cache

Connection: keep-alive

Content-Length: 35

Content-Type: application/x-www-form-urlencoded

Host: testpetra.zwj.com

Origin: http://127.0.0.1:35580

Pragma: no-cache

Referer: https://servicewechat.com/wx3071034e2676fcf8/devtools/page-frame.html

token: 676327abb26dc9ce379ec1021231ab2

User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13B143 Safari/601.1 wechatdevtools/1.02.1810250 MIcroMessenger/6.5.7 Language/zh-CN webview/ token/5b4ea88425d9a311bfe5ef57055d4bb0

▼ Form Data

view parsed

departmentId=8&pageNum=1&pageSize=10

激活 Windows
转到“设置”以激活 Windows。

3.content-type 为 multipart/form-data格式

这种参数是放在请求体里面的。form-data格式会有分割行（boundary）隔开，所以可以上传文件，如果form表单中涉及上传文件的操作，也需要吧encType设置为form-data的格式，同时form-data也可以用来传递key/value格式的数据。

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.  
▼ Request Payload  
-----WebKitFormBoundaryLdoHzEjBAvCvKJF  
Content-Disposition: form-data; name="__VIEWSTATE"  
  
/wEPDwUJNDc2MzMT0K3ZGT2rNXgStP0kFCU/TBR5KC8fx18qEs0k9b+1aAj297fcA==  
-----WebKitFormBoundaryLdoHzEjBAvCvKJF  
Content-Disposition: form-data; name="__VIEWSTATEGENERATOR"
```

然后到百度一查才发现一般需要提交文件就会用multipart/form-data这种形式，这种形式是以分界线来分割数据的，

```
▼ Request Headers view source  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8  
Accept-Encoding: gzip, deflate  
Accept-Language: zh-CN,zh;q=0.9  
Cache-Control: max-age=0  
Connection: keep-alive  
Content-Length: 9709  
Content-Type: multipart/form-data; boundary=-----WebKitFormBoundaryLdoHzEjBAvCvKJF
```

4.content-type 为 application/json格式

这种参数是放在请求体里面的。传递的参数是json格式的，相应的后台也需要以json格式来解析数据。在控制台显示的是Request Payload

```
▼ Request Headers view source  
Accept: /*  
Accept-Encoding: gzip, deflate, br  
Cache-Control: no-cache  
Connection: keep-alive  
Content-Length: 363  
Content-Type: application/json  
Host: testpetra.zwj.com  
Origin: http://127.0.0.1:35580  
Pragma: no-cache  
Referer: https://servicewechat.com/wx3071034e2676fcf8/devtools/page-frame.html  
token: 676327abbc26dc9ce379ec1021231ab2  
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13B143 Safari/601.1 wechatdevtools/1.02.1810250 MI  
croMessenger/6.5.7 Language/zh-CN webview/ token/5b4ea88425d9a311bfe5ef57055d4bb0  
▼ Request Payload view parsed  
{  
  "casePresentation": "啊啊啊啊",  
  "doctorId": "13",  
  "doctorDepartmentName": "心脑科",  
  "doctorPosition": "副主任医师",  
  "hospitalId": "123123",  
  "hospitalName": "河南省人民医院",  
  "age": "2",  
  "patientPhone": "15268109682",  
  "patientId": "91460ae0-05b3-403d-9dfe-0891fde549bb",  
  "patientIdCard": "",  
  "patientName": "林超测试",  
  "patientSex": "2",  
  "scheduleDate": "2018-11-26"  
}
```

一般来说response的格式也是json格式的。

```
▼ Response Headers view source  
Access-Control-Allow-Credentials: true  
Access-Control-Allow-Headers: token,content-type  
Access-Control-Allow-Methods: GET,POST  
Access-Control-Allow-Origin: https://testwebpetra.zwj.com  
Access-Control-Max-Age: 3600  
Connection: keep-alive  
Content-Type: application/json;charset=UTF-8  
Date: Mon, 26 Nov 2018 02:00:58 GMT  
Server: nginx  
Transfer-Encoding: chunked
```

5.放在header里面

目前为止遇到的情况是token会放在head里面，还有一些user_id之类的

▼ Request Headers

view source

Accept: */*
Accept-Encoding: gzip, deflate, br
Cache-Control: no-cache
Connection: keep-alive
Content-Length: 363
content-type: application/json
Host: testpetra.zwj.com
Origin: http://127.0.0.1:35580
Pragma: no-cache
Referer: https://servicewechat.com/wx3071034e2676fcf8/devtools/page-frame.html
token: 676327abbc26dc9ce379ec1021231ab2
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 9_1 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Version/9.0 Mobile/13B143 Safari/601.1 wechatdevtools/1.02.1810250 MicroMessenger/6.5.7 Language/zh_CN webview/ token/5b4ea88425d9a311bfe5ef57055d4bb0

▼ Response Headers

view source

Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: token,content-type
Access-Control-Allow-Methods: GET,POST
Access-Control-Allow-Origin: https://testwebpetra.zwj.com
Access-Control-Max-Age: 3600
Connection: keep-alive
Content-Encoding: gzip
Content-Type: application/json;charset=UTF-8
Date: Mon, 26 Nov 2018 02:31:06 GMT
Server: nginx
Transfer-Encoding: chunked