

阿东入门系列《三》网络请求



前文讲配置的时候我们说过，开发小程序时可以选择有 APPID 和无 APPID，两种方式。

1. 当有 APPID 的情况下，进行网络通信，只能和指定的域名进行通信。如果没有进行配置，编译时会报如下错误：

▼ Tue Nov 15 2016 13:57:42 GMT+0800 (中国标准时间) 合法域名校验出错

✖ ▶ http://192.168.1.1:8080/app/guangao 不在以下合法域名列表中，请参考文档：<https://mp.weixin.qq.com/debug/wxadoc/dev/api/network-request.html>

▶ Array[0]

pages/pageshu/pageshu asShow have been invoked

配置方法：

设置域名

需要在微信公众平台的小程序中设置域名。在微信小程序的设置界面可以看到设置选项：



微信公众平台 | 小程序



首页



开发管理



用户身份



数据分析



模板消息

选择开发设置：



可以看到服务器设置:

服务器配置	说明	操作
request合法域名	https://www.example.com	修改
socket合法域名	https://www.example.com	
uploadFile合法域名	https://www.example.com	
downloadFile合法域名	https://www.example.com	

这里我们可以设置我们这个 APPID 可以访问的域名，每种最多可以设置两个。(注意，此处只能使用 **https** 的域名，这个申请过程需要花费一定的时间)

2. 当没有 APPID 的情况下，就方便多了，可以随意进行网络请求，不限制域名，但是，这种情况下不能进行发布，也不能在手机上预览。如果想正式做小程序开发，还是需要有一个 https 的域名，但是学习的情况下 http 的就足够了。

在小程序中网络请求大致分为四种类型。

- 普通 HTTPS 请求(wx.request)
- 上传文件(wx.uploadFile)
- 下载文件(wx.downloadFile)
- WebSocket 通信(wx.connectSocket)

这里主要讲 wx.request：

使用 wx.request 可以发起一个 http 请求，一个微信小程序被限制为同时只有 5 个网络请求。注意是同时。

```
wx.request({  
  url: 'http://192.168.1.137:80/app/guanggao',  
  method: 'POST',  
  data: {  
    type: "1"  
  }  
})
```

```
    },
    header: {
        'Accept': 'application/json'
    },
    success: function (res) {
        that.setData({
            images: res.data.data.guanggao
        })
    }
    fail: function(err){
        console.log(err)
    }
    })
```

• 1

• 2

• 3

• 4

• 5

• 6

• 7

• 8

- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18

上面的代码会发送一个 http get 请求，其中的参数也比较容易理解。

- **url** 服务器的 url 地址
- **data** 请求的参数可以采用 String data: " xxx=xxx&xxx=xxx" 的形式或者 Object data:{ "userId" :1}的形式
- **header** 设置请求的 header
- **method** http 的方法,默认为 GET 请求

- **success** 接口成功的回调
- **fail** 接口失败的回调

另外还有一个参数没有在代码里:

- **complete** 调用接口结束之后的回调，无论成功或者失败该接口都会被调用

超时的设置

在前文中已经提到了在 app.js 中设置 networkTimeout 可以设置四种类型网络访问的超时时间：

```
"networkTimeout": {  
  "request": 10000,  
  "connectSocket": 10000,  
  "uploadFile": 10000,  
  "downloadFile": 10000  
}
```