**APIStoreSDK(Android)**

**使用说明文档**

**V1.0**

# APP集成APIStoreSDK

# APIStoreSDK概述

APIStoreSDK为开发者提供调用APIStore接口的方法，是一个轻量级的库，其中封装了异步网络请求和响应结束的回调方法，使用非常便捷。

# 集成APIStoreSDK

Eclipse/ADT将APIStoreSDK.jar添加到Android工程中libs文件夹下即可，调用时需import对应的类。Android Studio请在放入libs目录后，在project structure-> Dependencies-> File Dependency中添加jar包依赖。

在AndroidManifest.xml中,加入所需权限，这些权限将帮助sdk判断网络状态:

<uses-permission android:name=*"android.permission.READ\_PHONE\_STATE"* />

<uses-permission android:name=*"android.permission.INTERNET"* />

<uses-permission android:name=*"android.permission.ACCESS\_NETWORK\_STATE"* />

<uses-permission android:name=*"android.permission.ACCESS\_WIFI\_STATE"* />

# 初始化APIStoreSDK

在调用APIStoreSDK的接口前需要进行初始化, 这一步必须在Application中完成。

您需要在项目中建立一个继承自Application的类，并在项目配置文件AndroidManifest.xml中application标签下android:name中指定该类：

您需要在Application中添加的代码：

|  |
| --- |
| **package** com.baidu.apistore.sdk;  **import** android.app.Application;  // 请在AndroidManifest.xml中application标签下android:name中指定该类  **public** **class** MyApplication **extends** Application {    @Override  **public** **void** onCreate() {  // **TODO** 您的其他初始化流程  ApiStoreSDK.*init*(**this**, "您的apikey");  **super**.onCreate();  }  } |

上面的apikey可在apistore登录后，查看个人中心或者任何一个服务的详情页点击调试获得。

您需要在AndroidManifest.xml中添加的代码：

|  |
| --- |
| <!-- 权限配置 -->  <uses-permission  android:name=*"android.permission.READ\_PHONE\_STATE"* />  <uses-permission  android:name=*"android.permission.INTERNET"* />  <uses-permission  android:name=*"android.permission.ACCESS\_NETWORK\_STATE"* />  <uses-permission  android:name=*"android.permission.ACCESS\_WIFI\_STATE"* />    <!-- 指定mApplication，在mApplication中调用init方法 -->  <application  android:icon=*"@drawable/ic\_launcher"*  android:label=*"@string/app\_name"*  android:theme=*"@style/AppTheme"*  android:name=*"com.baidu.apistore.sdk.mApplication"*>  <!-- 请将mApplication替换为自己的类 -->  …… |

# 查找调用接口的参数值

在http://apistore.baidu.com上找出需要调用的接口，记录下接口地址、请求方法、请求参数。

如**天气查询接口apistore.baidu.com/apiworks/servicedetail/112.html:**



点击API标签，即可获得这些信息。（注：使用SDK不需要在调用API时额外再添加apikey参数）。

具体在调用付费服务时，需要购买，否者会提示余额不足。



# 接口调用

APIStoreSDK封装了完整的异步网络请求和回调逻辑，回调请继承ApiCallBack类并改写其中的onSuccess()、onError()、onComplete()方法（如不需要，可不用都写）。onSuccess会在请求成功时调用，onError会在请求失败时调用，onComplete总是会调用。responseString是api接口返回的响应字符串，或者空（请求失败时）。

调用的示例如下：

|  |
| --- |
| Parameters para = **new** Parameters();  para.put("city", "beijing");  ApiStoreSDK.*execute*("http://apis.baidu.com/heweather/weather/free",  ApiStoreSDK.*GET*,  para,  **new** ApiCallBack() {  @Override  **public** **void** onSuccess(**int** status, String responseString) {  Log.*i*("sdkdemo", "onSuccess");  mTextView.setText(responseString);  }    @Override  **public** **void** onComplete() {  Log.*i*("sdkdemo", "onComplete");  }    @Override  **public** **void** onError(**int** status, String responseString, Exception e) {  Log.*i*("sdkdemo", "onError, status: " + status);  Log.*i*("sdkdemo", "errMsg: " + (e == **null** ? "" : e.getMessage()));  mTextView.setText(getStackTrace(e));  }  }); |

接口的参数含义如下

ApiStoreSDK.execute(

String url, // 接口地址

String method, // 接口方法

Parameters params, // 接口参数

ApiCallBack callBack) { // 接口回调方法

其中方法为”get”或”post”，推荐使用sdk提供的变量ApiStoreSDK.*GET*和ApiStoreSDK.*POST*.

回调方法在UI线程中执行，可以直接修改UI。

onError 调用时，statusCode参数值对应如下：

-1 没有检测到当前网络；

-3 没有进行初始化;

-4 传参错误

其他数值是http状态码，或服务器响应的errNum，请查阅响应字符串responseString

响应数据保存在responseString中，样例如下：

|  |
| --- |
| {"HeWeather data service 3.0":[{"aqi":{"city":{"aqi":"105","co":"1","no2":"48","o3":"3","pm10":"89","pm25":"78","qlty":"轻度污染","so2":"4"}},"basic":{"city":"北京","cnty":"中国","id":"CN101010100","lat":"39.904000","lon":"116.391000","update":{"loc":"2015-11-18 19:50","utc":"2015-11-18 11:50"}},"status":"ok","suggestion":{"comf":{"brf":"较不舒适","txt":"白天会有降雪发生，您会感觉偏冷，不很舒适，请注意添加衣物。"}}}]} |