**兼容Android 7.0 App升级**

|  |  |  |
| --- | --- | --- |
| 时间 | 版本 | 作者 |
| 2017-3-30 | V 1.0.0 | hardy |

1. 在AndroidManifest.xml清单文件中注册Provider

<provider  
 android:name="android.support.v4.content.FileProvider"  
 android:authorities="com.qk365.qkonline.fileprovider"  
 android:exported="false"  
 android:grantUriPermissions="true">  
 <meta-data  
 android:name="android.support.FILE\_PROVIDER\_PATHS"  
 android:resource="@xml/qk\_file\_paths"  
 tools:replace="android:resource"/>  
</provider>

authorities: 常用命名为包名+fileprovider

exported要求必须为false,为true会报安全异常

grantUriPermissions表示URI临时访问权限，true为需要临时访问权限

resource指当前组件引用res文件夹中qk\_file\_paths文件，这个文件名可以随便写，只要在res文件夹中有这个文件即可。

在res文件夹中新建xml文件夹，再新建qk\_file\_paths.xml文件，内容如下

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <paths>  
 <external-path  
 name="files\_root"  
 path="Android/data/com.qk365.qkonline/" />  
 <external-path  
 name="external\_storage\_root"  
 path="." />  
 <!-- yan notice: do not edit -->  
 <root-path path="" name="camera\_photos" />  
 </paths>  
</resources>

path填上Android/data/包名

1. 安装Apk时注意兼容高低版本

Intent intent = new Intent(Intent.*ACTION\_VIEW*);  
//判断是否是AndroidN以及更高的版本  
 if (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*N*) {  
 intent.setFlags(Intent.*FLAG\_GRANT\_READ\_URI\_PERMISSION*);  
 Uri contentUri = FileProvider.*getUriForFile*(context, BuildConfig.*APPLICATION\_ID* + ".fileprovider", apkFile);  
 intent.setDataAndType(contentUri, "application/vnd.android.package-archive");  
 } else {  
 intent.setDataAndType(Uri.*fromFile*(apkFile), "application/vnd.android.package-archive");  
 intent.setFlags(Intent.*FLAG\_ACTIVITY\_NEW\_TASK*);  
 }  
 context.startActivity(intent);

主要是注意getUriForFile方法中的参数，第二个参数为BuildConfig.APPLICATION\_ID + “.fileprovider”

即包名+.fileprovider，这个和AndroidManifest中的Provider中的属性authorities一致