

Fork me on GitHub

JohnTsai

Android开发 邮箱:JohnTsai.Work@gmail.com

博客园

首页

博问

闪存

新随笔

联系

订阅

管理

随笔-80 文章-0 评论-138

公告

我的邮箱:JohnTsai.Work@gmail.com

我的Github:<https://github.com/JohnTsai>

昵称: JohnTsai

园龄: 1年8个月

粉丝: 86

关注: 64

+加关注

< 2016年1月 >						
日	一	二	三	四	五	六
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

我的标签

Android(46)
Java(16)
技术(8)
Git(6)
GitHub(5)
Python(4)
Activity(3)
Android Studio(3)
Layout(3)
Markdown(3)
更多

随笔档案(80)

2015年11月 (2)
2015年10月 (3)
2015年9月 (1)
2015年8月 (1)
2015年7月 (5)
2015年6月 (6)
2015年5月 (2)
2015年3月 (2)
2015年2月 (3)
2015年1月 (6)
2014年12月 (8)
2014年11月 (11)
2014年10月 (12)
2014年9月 (8)
2014年8月 (8)
2014年7月 (2)

积分与排名

Android四大组件之一——Activity的生命周期（图文详解）

转载请在文章开头处注明本博客网址: <http://www.cnblogs.com/JohnTsai>联系方式: JohnTsai.Work@gmail.com

[Android四大组件学习系列Activity篇]

1.Android四大组件之一——Activity(一)定义、状态和后退栈(图文详解)

2.Android四大组件之一——Activity的生命周期（图文详解）

上一篇文章讲了Activity的定义、状态和后退栈，现在讲讲Activity的重点和难点——Activity的生命周期。

• Activity生命周期的回调方法：



```
//创建Activity
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
}

//Activity快要变成可见的
@Override
protected void onStart() {
    super.onStart();
}

//Activity变成可见的，处于运行状态
@Override
protected void onResume() {
    super.onResume();
}

//其他Activity获得用户焦点，（Activity快要暂停了）
@Override
protected void onPause() {
    super.onPause();
}

//Activity不再可见，处于停止状态
@Override
protected void onStop() {
    super.onStop();
}

//Activity快要被销毁了
@Override
protected void onDestroy() {
    super.onDestroy();
}
```



这些方法定义了Activity完整的生命周期，实现这些方法，我们能监控Activity生命周期中的三个嵌套循环(monitor three nested activity lifecycle)

1.Activity的**entire lifetime**（全部的生命期）发生在调用**onCreate()**和调用**onDestory()**之间。
在**onCreate()**方法中执行全局状态的建立(例如定义布局)，在**onDestory()**方法中释放所有保存的资源。

2.Activity的**visible lifetime**(可见的生命期)发生在调用**onStart()**和**onStop()**之间。
在这个期间，用户能在屏幕上看见Activity，和它进行交互。
系统在Activity的完整寿命中可能多次调用**onStart()**和**onStop()**，正如Activity交替地对用户可见或隐藏。

最新评论

1. Re:Markdown技巧：两种方式实现页内跳转
@Life is a life....
--JohnTsai
2. Re:Markdown技巧：两种方式实现页内跳转
@Life is a life....
--JohnTsai
3. Re:Markdown技巧：两种方式实现页内跳转
<h2 id='1'>1. 单反入门</h2>

我实际使用时，使用单引号才能跳转，求解。
--Life is a life.
4. Re:Android技巧分享——Android开发超好用工具吐血推荐mark

--伊甸一点
5. Re:《Thinking in Java》习题——吸血鬼数字
向前辈致敬
- naadp

阅读排行榜

1. Android四大组件之——Activity的生命周期（图文详解）(2867)
2. Android技巧分享——让官方模拟器和genymotion虚拟机飞起来(2052)
3. Android Studio小技巧(2006)
4. Markdown技巧：两种方式实现页内跳转(1936)
5. Android开发工具——Android studio1.0正式版使用技巧(1908)

评论排行榜

1. Android开发工具——Android studio1.0正式版使用技巧(18)
2. Android 开源库——侧滑菜单栏（SlidingMenu）的导入和使用(16)
3. Android技巧分享——Android开发超好用工具吐血推荐(16)
4. 《Thinking in Java》习题——吸血鬼数字(9)
5. Android学习之——ListView下拉刷新(8)

推荐排行榜

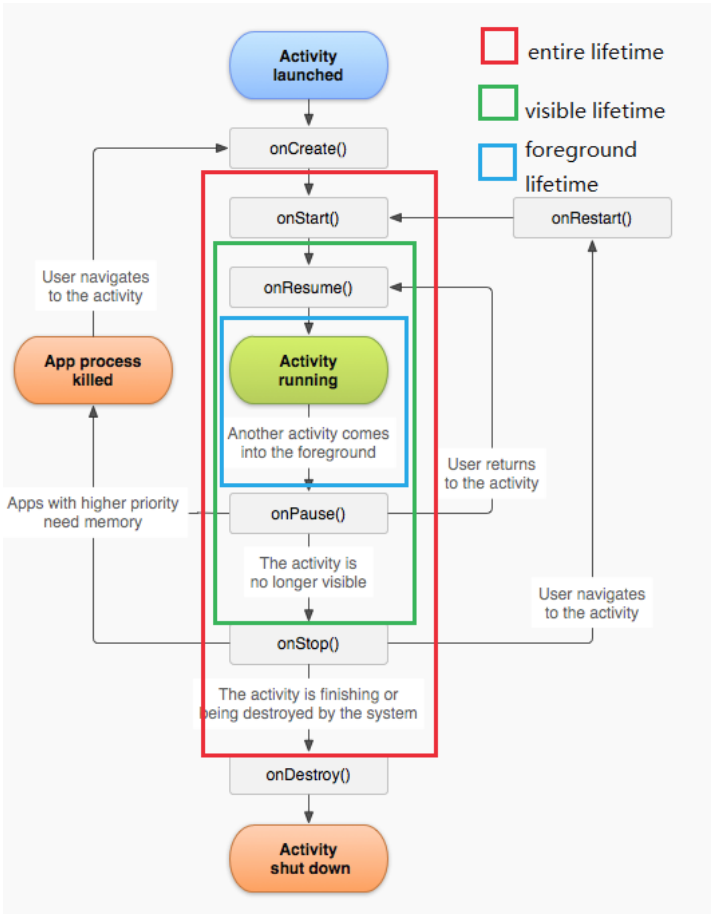
1. Android开发工具——Android studio1.0正式版使用技巧(10)
2. Android学习之——ListView下拉刷新(7)
3. Android Studio小技巧(7)
4. Android技巧分享——Android开发超好用工具吐血推荐(5)
5. 最好的Java和Android开发IDE---IntelliJ IDEA使用技巧(5)

Android四大组件之——Activity的生命周期（图文详解） - JohnTsai - 博客园

3Activity的foreground lifetime (前台的生命期)发生在调用onResume()和onPause()之间。

在这期间，Activity在屏幕上所有其他Activity的前面，有用户输入焦点。一个Activity能频繁的在前台进入和出去之间转变。

下面这张图很好的讲解了Activity的生命周期和上面说的三种生命期：



下面通过一个Demo来学习以下Activity的生命周期:

Demo很简单，就只有两个Activity，MainActivity和SecondActivity和一个按钮。点击按钮，由MainActivity跳转到SecondActivity

MainActivity：

```
1 package com.example.activitytest;
2
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.util.Log;
7 import android.view.View;
8 import android.view.View.OnClickListener;
9 import android.widget.Button;
10
11 public class MainActivity extends Activity {
12
13     private Button btn;
14     private static final String TAG = "ActivityTest";
15     @Override
16     protected void onCreate(Bundle savedInstanceState) {
17         super.onCreate(savedInstanceState);
18         Log.d(TAG, "MainActivity onCreate");
19         setContentView(R.layout.activity_main);
20         btn = (Button) findViewById(R.id.btn);
21         btn.setOnClickListener(new OnClickListener() {
22
23             @Override
24             public void onClick(View v) {
25                 Intent intent = new Intent(MainActivity.this, SecondActivity.class);
26                 startActivity(intent);
27             }
28         });
29     }
30 }
```

```
31     @Override
32     protected void onPause() {
33         Log.d(TAG, "MainActivity onPause ");
34         super.onPause();
35     }
36     @Override
37     protected void onResume() {
38         Log.d(TAG, "MainActivity onResume ");
39         super.onResume();
40     }
41     @Override
42     protected void onStart() {
43         super.onStart();
44         Log.d(TAG, "MainActivity onStart ");
45     }
46     @Override
47     protected void onStop() {
48         super.onStop();
49         Log.d(TAG, "MainActivity onStop ");
50     }
51     @Override
52     protected void onDestroy() {
53         super.onDestroy();
54         Log.d(TAG, "MainActivity onDestroy ");
55     }
56     @Override
57     protected void onRestart() {
58         super.onRestart();
59         Log.d(TAG, "MainActivity onRestart ");
60     }
61 }
```



SecondActivity:



```
1 package com.example.activitytest;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.util.Log;
6
7 public class SecondActivity extends Activity {
8     private static final String TAG = "ActivityTest";
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        setContentView(R.layout.second_layout);
12        super.onCreate(savedInstanceState);
13        Log.d(TAG, "SecondActivity onCreate");
14    }
15
16    @Override
17    protected void onStart() {
18        super.onStart();
19        Log.d(TAG, "SecondActivity onStart");
20    }
21
22    @Override
23    protected void onResume() {
24        super.onResume();
25        Log.d(TAG, "SecondActivity onResume");
26    }
27
28    @Override
29    protected void onPause() {
30        super.onPause();
31        Log.d(TAG, "SecondActivity onPause");
32    }
33
34    @Override
35    protected void onStop() {
36        super.onStop();
37        Log.d(TAG, "SecondActivity onStop");
38    }
39
40    @Override
41    protected void onDestroy() {
42        super.onDestroy();
43        Log.d(TAG, "SecondActivity onDestroy");
44    }
45
46    @Override
47    protected void onRestart() {
48        super.onRestart();
49        Log.d(TAG, "SecondActivity onRestart");
50    }
51 }
```

47 }



使用Activity需要在Manifest.xml文件中注册：



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.activitytest"
4     android:versionCode="1"
5     android:versionName="1.0" >
6
7     <uses-sdk
8         android:minSdkVersion="8"
9         android:targetSdkVersion="21" />
10
11     <application
12         android:allowBackup="true"
13         android:icon="@drawable/ic_launcher"
14         android:label="@string/app_name"
15         android:theme="@style/AppTheme" >
16         <activity
17             android:name=".MainActivity"
18             android:label="@string/app_name" >
19             <intent-filter>
20                 <action android:name="android.intent.action.MAIN" />
21                 <category android:name="android.intent.category.LAUNCHER" />
22             </intent-filter>
23         </activity>
24         <activity
25             android:name=".SecondActivity"
26             android:label="SecondActivity"
27             android:theme="@android:style/Theme.Dialog"
28             >
29         </activity>
30     </application>
```



1. 运行应用，MainActivity运行。

可以看出一个Activity运行调用的回调方法是: onCreate()->onStart()->onResume()

PID	TID	Application	Tag	Text
1582	1582	com.example.activitytest	ActivityTest	MainActivity onCreate
1582	1582	com.example.activitytest	ActivityTest	MainActivity onStart
1582	1582	com.example.activitytest	ActivityTest	MainActivity onResume

2. 点击跳转按钮，由MainActivity跳转到SecondActivity:

1>SecondActivity是Dialog。

1582	1582	com.example.activitytest	ActivityTest	MainActivity onPause
1582	1582	com.example.activitytest	ActivityTest	SecondActivity onCreate
1582	1582	com.example.activitytest	ActivityTest	SecondActivity onStart
1582	1582	com.example.activitytest	ActivityTest	SecondActivity onResume



由于SecondActivity是Dialog，所以MainActivity依然可见，所以只调用了它的onPause()方法。

2>若将SecondActivity改为不是Dialog（只需在Manifest.xml文件中删掉 android:theme="@android:style/Theme.Dialog"这一行即可）再点击跳转按钮：

1637	1637	com.example.activitytest	ActivityTest	MainActivity onPause
1637	1637	com.example.activitytest	ActivityTest	SecondActivity onCreate
1637	1637	com.example.activitytest	ActivityTest	SecondActivity onStart
1637	1637	com.example.activitytest	ActivityTest	SecondActivity onResume
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStop

对于MainActivity,由于不可见，状态由运行变为停止。

依次调用了它的回调方法:onPause()->onStop()

3.Activity处于运行状态时，点击返回按钮。

1637	1637	com.example.activitytest	ActivityTest	MainActivity onPause
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStop
1637	1637	com.example.activitytest	ActivityTest	MainActivity onDestroy

点击返回按钮，Activity由运行状态变为死亡状态，依次调用它的回调方法:onPause()->onStop()->onDestroy()

4.Activity处于运行状态，点击主页按钮，返回桌面。

1637	1637	com.example.activitytest	ActivityTest	MainActivity onPause
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStop

Activity由运行状态变为停止状态，依次调用它的回调方法:onPause()->onStop()

5.Activity处于停止状态，即由桌面返回到Activity。

1637	1637	com.example.activitytest	ActivityTest	MainActivity onRestart
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStart
1637	1637	com.example.activitytest	ActivityTest	MainActivity onResume

停止->运行，依次调用回调方法：onRestart()->onStart()->onResume()

6.Activity处于运行状态，旋转屏幕：

1637	1637	com.example.activitytest	ActivityTest	MainActivity onPause
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStop
1637	1637	com.example.activitytest	ActivityTest	MainActivity onDestroy
1637	1637	com.example.activitytest	ActivityTest	MainActivity onCreate
1637	1637	com.example.activitytest	ActivityTest	MainActivity onStart
1637	1637	com.example.activitytest	ActivityTest	MainActivity onResume

可以看出，旋转屏幕是一个销毁Activity然后重新创建Activity的过程。

运行->暂停->停止->死亡->运行

依次调用回调方法: onPause()->onStop()->onDestroy()->onCreate()->onStart()->onResume()

7.Activity处于运行状态，手机此时锁屏。

```
10-11 03:33:18.387 9649-9649/com.ijtsai.recyclerviewdemo D/MainActivity: onPause
10-11 03:33:18.387 9649-9649/com.ijtsai.recyclerviewdemo D/MainActivity: onStop
```

可以看出，手机锁屏，此时由于Activity不可见，会进入停止状态。

依次调用回调方法：onPause()->onStop()

8.当应用正在运行时，手机锁屏，然后解锁回到应用界面。

```
10-11 03:42:48.603 9649-9649/com.ijtsai.recyclerviewdemo D/MainActivity: onRestart
10-11 03:42:48.603 9649-9649/com.ijtsai.recyclerviewdemo D/MainActivity: onStart
10-11 03:42:48.619 9649-9649/com.ijtsai.recyclerviewdemo D/MainActivity: onResume
```

可以看出应用从停止状态恢复到运行状态。

依次调用回调方法：onRestart()->onStart()->onResume()

本人邮箱：JohnTsai.Work@gmail.com, 欢迎交流讨论。

欢迎转载，转载请在文章开头处注明本博客网址：http://www.cnblogs.com/JohnTsai

如果喜欢，请点推荐！谢谢~

标签: Android, Activity

好文要顶

关注我

收藏该文



JohnTsai

关注 - 64

粉丝 - 86

+ 加关注

« 上一篇: Android四大组件之一——Activity(一)定义、状态和后退栈(图文详解)

» 下一篇: Android技巧分享——如何用电脑下载在Google play中应用的apk文件

posted @ 2014-10-26 23:19 JohnTsai 阅读(2867) 评论

评论列表

#1楼 2014-10-27 15:57 *深海

好文!

支:

#2楼[楼主] 2014-10-27 18:50 JohnTsai