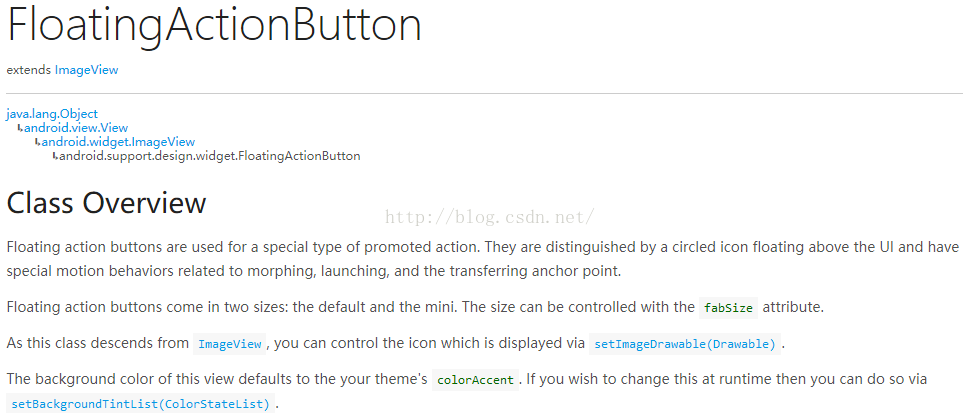
 首先看**FAB**是什么？



查看API文档我们大概就可以知道如何使用它了，那么有哪些属性呢？真的是多啊，这里就介绍几个常用的：

id:为FAB定义一个标识

alpha:FAB的透明度,作为一个值介于0(完全透明的)和1(完全不透明的)

clickable:FAB是否能响应点击事件

backgroundTint:设置FAB的背景颜色

rippleColor :设置FAB点击时的背景颜色  
elevation:默认状态下FAB的阴影大小  
pressedTranslationZ :点击时候FAB的阴影大小  
fabSize :设置FAB的大小，该属性有两个值，分别为normal和mini  
src :FAB的图标  
layout\_anchor:设置FAB的锚点，即以哪个控件为参照点设置位置  
layout\_anchorGravity:设置FAB相对锚点的位置

如何使用FAB?需要引入Support Design Library，AS中引入十分简单，只要在gradle中加入：

compile **'com.android.support:recyclerview-v7:23.3.0'**

下面在布局文件中如何使用吧！

**[html]** [view plain](http://blog.csdn.net/bat_os/article/details/51199043) [copy](http://blog.csdn.net/bat_os/article/details/51199043)

[print?](http://blog.csdn.net/bat_os/article/details/51199043)

1. **<android.support.design.widget.CoordinatorLayout**
2. xmlns:android="http://schemas.android.com/apk/res/android"
3. xmlns:wang="http://schemas.android.com/apk/res-auto"
4. android:id="@+id/coordin"
5. android:layout\_width="match\_parent"
6. android:layout\_height="match\_parent"
7. android:layout\_alignParentBottom="true"**>**
8. **<android.support.design.widget.FloatingActionButton**
9. android:id="@+id/fab"
10. android:layout\_width="wrap\_content"
11. android:layout\_height="wrap\_content"
12. android:clickable="true"
13. android:src="@mipmap/add\_circle\_black\_24dp.png"   //Google提供的material-design-icons，里面有大量的图标，根据实际选区
14. wang:backgroundTint="@color/blue"
15. android:layout\_marginRight="10dp"
16. android:layout\_marginBottom="10dp"
17. wang:fabSize="normal"
18. android:layout\_gravity="bottom|right"
19. android:elevation="30dp"
20. **/>**
21. ...
22. **</android.support.design.widget.CoordinatorLayout>**

先别管android.support.design.widget.CoordinatorLayout，有过自定义View的朋友，看见xmlns:wang="http://schemas

.android.com/apk/res-auto"是不是很熟悉，但是为什么要这样呢？这个就是来获取我们自定义属性的命名空间,然后

就可以在布局文件中使用我们自定义的属性。EC中需要加上完整的包名，而AS中只需要/res-auto就可以了。我们使用