# android界面开发

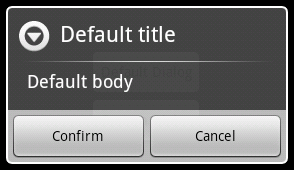
1. 界面元素
   1. 窗口：Activity

应用程序中每个屏幕显示都通过继承和扩展基类Activity实现

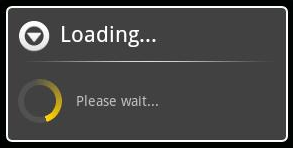
* 1. 对话框：Dialog

一种显示于Activity之上的界面元素，由Activity管理，用于显示一些临时信息和功能，它比Activity开销小

* + 1. AlertDialog



* + 1. ProgressDialog



* 1. 可视化控件：View

Button,CheckBox,RadioButton,EditText等

* 1. 其它

标题栏、菜单等

1. 实现布局
   1. LinearLayout 线性布局

垂直或水平一个接着一个排列控件

* 1. RelativeLayout相对坐标布局

例子：

<Button

android:id=*"@+id/button2"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

android:layout\_alignBaseline=*"@+id/button1"*

android:layout\_alignParentRight=*"true"*

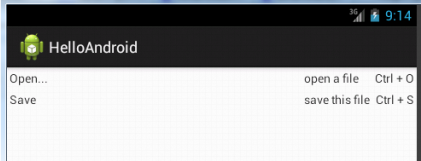
android:layout\_marginRight=*"18dp"*

android:text=*"linearHorizontal"* />

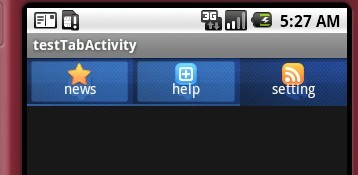
还有其它很多的属性，可参考官方文档：

http://developer.android.com/reference/android/widget/RelativeLayout.LayoutParams.html

* 1. TableLayout 表单布局



* 1. TabWidget 切换卡



1. 布局中需要注意的一些属性
   1. layout\_width 和 layout\_height

wrap\_content：根据控件的内容决定控件大小

fill\_parent：占满整个父窗口

* 1. layout\_weight

用于给一个线性布局中的诸多视图的重要度赋值。 所有的视图都有一个layout\_weight值，默认为零，意思是需要显示多大的视图就占据多大的屏幕空 间。若赋一个高于零的值，则将父视图中的可用空间分割，分割大小具体取决于每一个视图的layout\_weight 值以及该值在当前屏幕布局的整体 layout\_weight值和在其它视图屏幕布局的layout\_weight值中所占的比率而定。