



# Android 应用开发培训

王 雪明 .

mwongxming@gmail.com

---

[www.3gdc.com](http://www.3gdc.com)



- 在XML文件中申明UI 组件
  - Android提供了从xml的节点元素，这些节点元素对应代码中的UI组件。
  - 优点：直观简洁，可读性强；
  - 实现了UI界面和逻辑代码的分离
- 在代码中构造组件
  - 在java代码中构造组件。
  - 缺点：抽象模糊，可读性比较差；
  - 耦合性强，数据的表现和逻辑错杂很难



- 所有的View和ViewGroup都支持XML的属性。
- XML的属性是可以继承的。
- View和ViewGroup都支持下面两个属性。
  - ID属性
    - `android:id="@+id/my_button"`
    - `android:id="@android:id/empty"`
  - Layout Parameters



## Layout Parameters

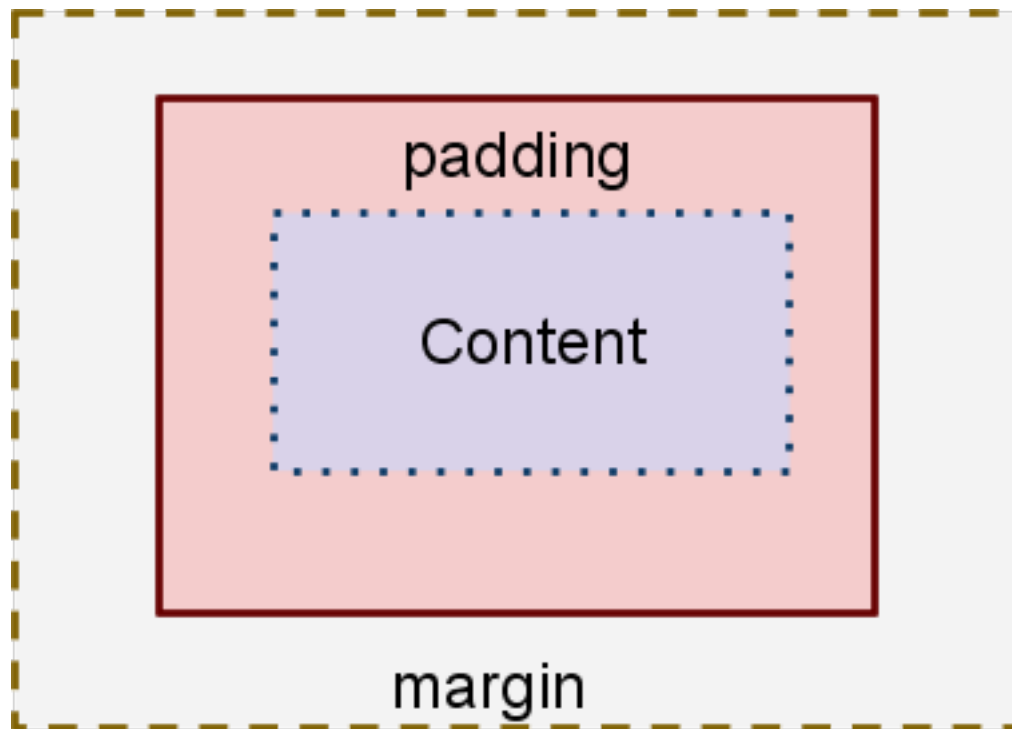
- **android:layout\_height**
- **android:layout\_width**

表达尺寸大小的三种方式

- 一个确定的数字(50px)
- **FILL\_PARENT**
- **WRAP\_CONTENT**



- View支持padding, 但是不支持margin。
- ViewGroup支持padding和margin



常用的一些方法  
`setPadding(int, int, int, int)`

`getPaddingLeft()`  
`getPaddingTop()`



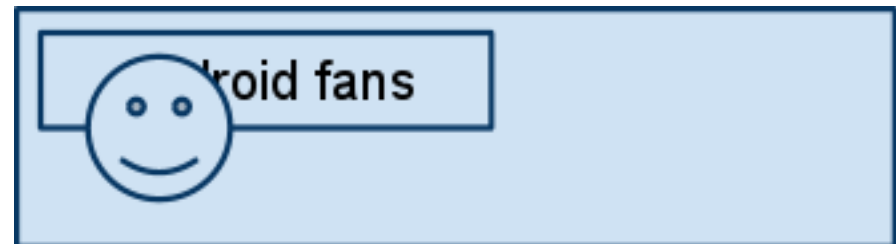
# FrameLayout 布局



什么是FrameLayout?

FrameLayout对象就好比一块在屏幕上提前预订好的空白区域，然后你可以填充一些元素到里边。

注意：所有的元素都被放置在FrameLayout区域**最左上**的区域。而且无法为这些元素指定一个确切的位置。如果一个FrameLayout里边有多个子元素，那么后边的子元素的显示会**重叠**在前一个元素上。





# LinearLayout 布局

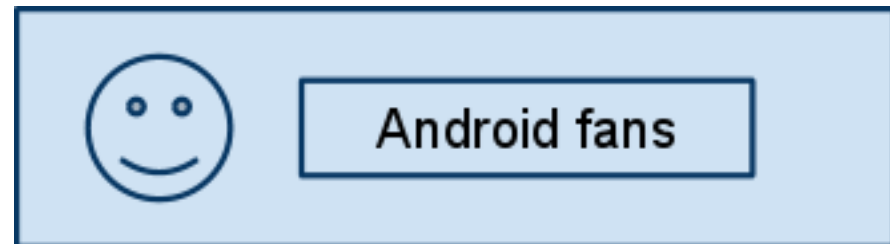


什么是LinearLayout ?

**LinearLayout**是一种Android中最常用的布局之一，它将自己包含的子元素按照一个方向进行排列。

方向有两种

- 水平 **Android:orientation="horizontal"**  
子元素 **从左到右** 一个一个的水平排列
- 竖直 **Android:orientation="vertical"**  
子元素 **从上到下** 一个接着一个的竖直排列





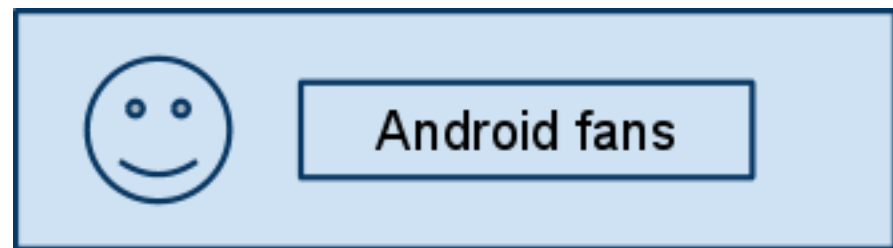
# RelativeLayout 布局



什么是RelativeLayout？

**RelativeLayout**是一个相对布局类。首先 RelativeLayout是一个容器，它里边的元素的位置是按照相对位置来计算的。

- 相对于其他组件的位置
- 相对于RelativeLayout的位置





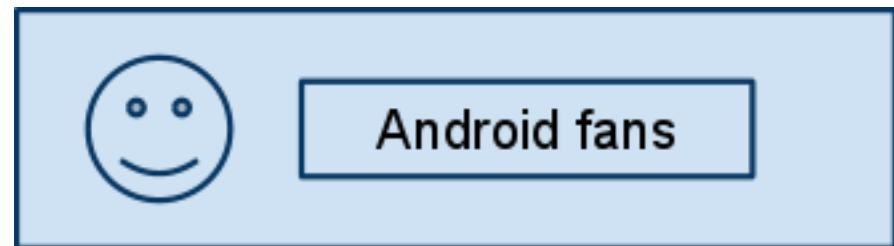


# TableLayout 布局



什么是TableLayout？

TableLayout 是一种表格式的布局。这种布局会把包含的元素以**行和列**的形式进行排列。表格的列数为每一行的最大列数。当然表格里边的单元格是可以为空。





- 如何用xml进行布局
- 布局的一些常见属性
- 组件的位置
- Android中的盒子模型
- 常见的布局类
  - FrameLayout
  - **LinearLayout**
  - **RelativeLayout**
  - TableLayout