

Button

知识解析

Button 表示一个按钮。用户点击后会作出响应。具体的响应行为需要我们来定义（一般通过监听器来处理）。

Button 是 TextView 的子类, 因此, 原则上, TextView 的属性设置均可以运用于 Button。例如: Button 的 layout 方面会有两种属性:

wrap_content: layout_width 为 wrap_content 时, 这个 button 会依据 button 上的 text 长度为基准

match_parent: layout_width 为 match_parent 时, 则会以 parent 最宽的长度为主

另外可以给 Button 设置一个背景图片:

`android:background="@drawable/pic"`

或者给 Button 设置图片:

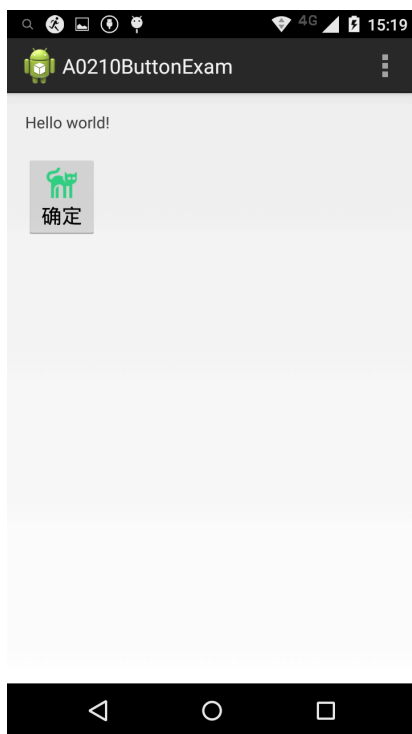
`android:drawableTop/drawableBottom/drawableLeft/drawableRight`, 分别可以在按钮的上下左右加上一张图片。

`android:onClick="clickEvent"`: 可以通过 `onClick` 属性, 设置一个 Activity 中的对应这个按钮的处理方法, 这个处理方法的原型是:

```
public void clickEvent(View view){  
    }  
}
```

其中, `clickEvent` 和 `onClick` 属性中定义的方法名称应该一致, 并且有一个 `View` 参数。方法必须为 `public` 且无返回值。

功能演示



实战操作

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.test.a0210buttonexam.MainActivity" >
    <TextView
        android:id="@+id/textView1"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView1"
    android:layout_below="@+id/textView1"
    android:layout_marginTop="20dp"
    android:drawableTop="@drawable/cat"
    android:text="确定" />

</RelativeLayout>
```

职业素质

Button 继承了 TextView，它主要是在 UI 界面上生成一个按钮，该按钮可以提供用户点击，当用户点击按钮时，按钮会触发一个监听事件 onClick。

按钮使用起来比较容易，舍已设置不用的背景颜色，还可以设置不同的样式让按钮更具有风格，更好看。