
LayoutInflater 的应用

1. LayoutInflater 简介

加载布局的任务通常都是在 Activity 中调用 setContentView()方法来完成的，其实 setContentView()方法的内部也是使用 LayoutInflater 来加载布局的，只不过这部分源码是 internal 的，不太容易查看到。那么接下来我们就来把 LayoutInflater 的工作流程仔细地剖析一遍，在特殊的场合 LayoutInflater 也会被经常用到。

2. LayoutInflater 的应用

先来看一下 LayoutInflater 的基本用法吧，它的用法非常简单，首先需要获取到 LayoutInflater 的实例，有两种方法可以获取到，第一种写法如下：

```
LayoutInflater inflater = LayoutInflater.from(context);
```

当然，还有另外一种写法也可以完成同样的效果：

```
LayoutInflater inflater = (LayoutInflater) context  
    .getSystemService(Context.LAYOUT_INFLATER_SERVICE);
```

其实第一种就是第二种的简单写法，只是 Android 给我们做了一下封装而已。得到了 LayoutInflater 的实例之后就可以调用它的 inflate()方法来加载布局了，如下所示：

```
inflater.inflate(resourceId, root);
```

inflate()方法一般接收两个参数，第一个参数就是要加载的布局 id，第二个参数是指给该布局的外部再嵌套一层父布局，如果不需要就直接传 null。这样就成功创建了一个布局的实例，之后再将它添加到指定的位置就可以显示出来了。

下面我们就通过一个非常简单的小例子，来更加直观地看一下 `LayoutInflater` 的用法。比如说当前有一个项目，其中 `MainActivity` 对应的布局文件叫做 `activity_main.xml`，代码如下所示：

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/main_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

</LinearLayout>
```

这个布局文件的内容非常简单，只有一个空的 `LinearLayout`，里面什么控件都没有，因此界面上应该不会显示任何东西。

那么接下来我们再定义一个布局文件，给它取名为 `button_layout.xml`，代码如下所示：

```
<Button xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" >

</Button>
```

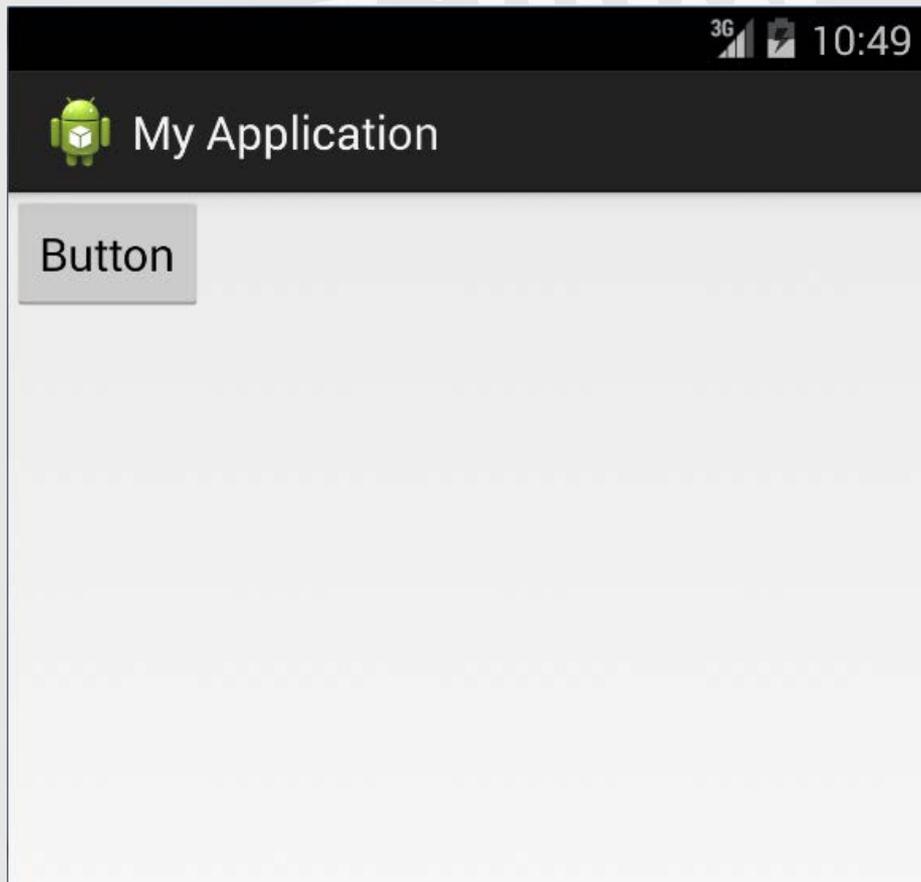
这个布局文件也非常简单，只有一个 `Button` 按钮而已。现在我们要想办法，如何通过 `LayoutInflater` 来将 `button_layout` 这个布局添加到主布局文件 `activity_main.xml` 的 `LinearLayout` 中。根据上面介绍的用法，修改 `MainActivity` 中的代码，如下所示：

```
public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        LinearLayout mainLayout = (LinearLayout) findViewById(R.id.main_layout);
        LayoutInflater inflater = LayoutInflater.from(this);
        View buttonLayout = inflater.inflate(R.layout.button_layout, null);
        mainLayout.addView(buttonLayout);
    }
}
```

可以看到，这里先是获取到了 `LayoutInflater` 的实例，然后调用它的 `inflate()` 方法来加载 `button_layout` 这个布局，最后调用 `LinearLayout` 的 `addView()` 方法将它添加到 `LinearLayout` 中。

现在可以运行一下程序，结果如下图所示：



`Button` 在界面上显示出来了！说明我们确实是借助 `LayoutInflater` 成功将 `button_layout` 这个布局添加到 `LinearLayout` 中了。`LayoutInflater` 技术广泛应用于需要动态添加 `View` 的时候，比如在 `ScrollView` 和 `ListView` 中，经常都可以看到 `LayoutInflater` 的身影。