

****

**实 验 报 告**

实验课程 Android应用程序开发

实验项目 实验2 布局模式（一）

实验地点 2-721

指导教师 郑向阳

班 级 14计本二

学生姓名 吴高航

学 号 14219266216

教师评分

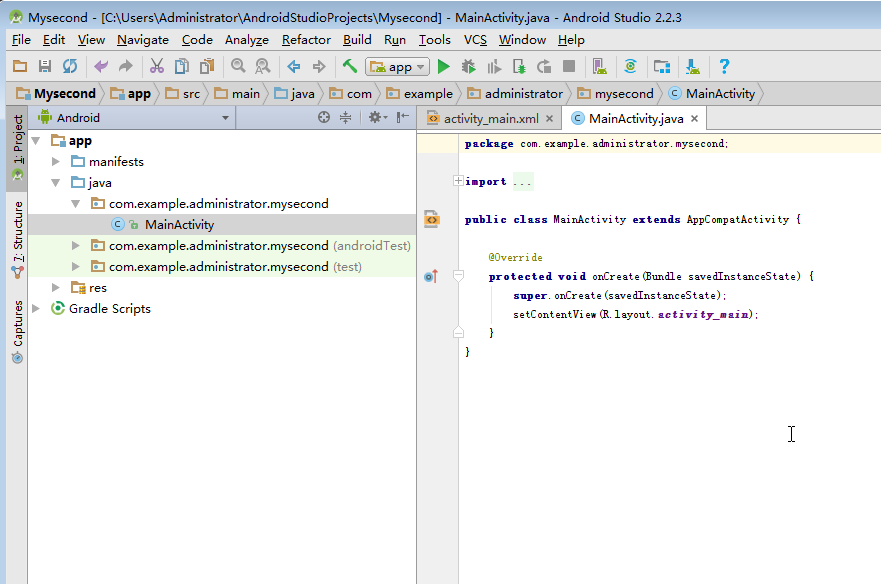
日 期 2017-02-21

实验2 布局模式（一）

1. 实验目的
   1. 理解android开发的一般过程
   2. 掌握XML文件的一般使用方法
   3. 掌握在XML中设置TEXTVIEW、BUTTON等控件的属性
2. 准备

JDK、SDK、AndroidStudio

1. 实验内容
   1. 在AndroidStudio中新建项目BMI，并说明步骤：



* 1. 说明activity\_main.xml文件的作用

MainActivity一般为创建一个Android工程时默认的主页面,主页面的内容为一个TextView控件

* 1. 说明在android中布局配置（layout）的种类？

1.LinearLayout(线性布局) 提供了控件水平垂直排列的模型，同时可以通过设置子控件的weight布局参数控制各个控件在布局中的相对大小。水平（vertical）垂直（horizontal） fill-parent:占满整个屏幕，wrap-content：刚好适合控件内容的大小

2.AbsoluteLayout（坐标布局） 可以让子元素指定准确的x/y坐标值，并显示在屏幕上。(0, 0)为左上角，当向下或向右移动时，坐标值将变大。AbsoluteLayout没有页边框，允许元素之间互相重叠（尽管不推荐）。我们通常不推荐使用AbsoluteLayout，除非你有正当理由要使用它，因为它使界面代码太过刚性，以至于在不同的设备上可能不能很好地工作。

3.RelativeLayout（相对布局） 允许子元素指定他们相对于其它元素或父元素的位置（通过ID指定）。因此，你可以以右对齐，或上下，或置于屏幕中央的形式来排列两个元素。元素按顺序排列，因此如果第一个元素在屏幕的中央，那么相对于这个元素的其它元素将以屏幕中央的相对位置来排列。如果使用XML来指定这个layout，在你定义它之前，被关联的元素必须定义。

4. FrameLayout(单帧布局) 是最简单的一个布局对象。它被定制为你屏幕上的一个空白备用区域，之后你可以在其中填充一个单一对象 — 比如，一张你要发布的图片。所有的子元素将会固定在屏幕的左上角；你不能为FrameLayout中的一个子元素指定一个位置。后一个子元素将会直接在前一个子元素之上进行覆盖填充，把它们部份或全部挡住（除非后一个子元素是透明的）。 Android:src=”@drawable/”属性指定所需图片的文件位置，用ImageView显示图片时，也应当用android：src指定要显示的图片

5.TableLayout（表格布局） 以行列的形式管理子控件，每一行为一个TableRow的对象，TableRow也可以添加子控件

* 1. 在目录layout中新建linearlayout.xml并设计以下界面，（请参考后 面的例子），请截图你的界面设计并复制界面设计的代码：



图 1 linearlayout.xml设计界面

截图：



代码：

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/activity\_main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 tools:context="com.example.administrator.mysecond.MainActivity"**>  
 <**LinearLayout  
 android:layout\_height="wrap\_content"  
 android:layout\_width="match\_parent"  
 android:layout\_weight="1"  
 android:orientation="vertical"** >  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:textSize="18sp"  
 android:textColor="#FFFFFF"  
 android:gravity="center"  
 android:background="#aa0000"  
 android:text="第一行 -01"** />  
  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:textSize="18sp"  
 android:textColor="#FFFFFF"  
 android:background="#bb1"  
 android:gravity="center"  
 android:text="第二行 -02"  
 android:id="@+id/textView"** />  
  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:textSize="18sp"  
 android:textColor="#FFFFFF"  
 android:background="#7FFF00"  
 android:gravity="center"  
 android:text="第三行 -03"** />  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:textSize="18sp"  
 android:textColor="#FFFFFF"  
 android:background="#aa21"  
 android:gravity="center"  
 android:text="第四行 -04"** />  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:textSize="18sp"  
 android:textColor="#FFFFFF"  
 android:background="#145"  
 android:gravity="center"  
 android:text="第五行 -05"** />  
 </**LinearLayout**>  
</**RelativeLayout**>

请参考以下例子的设计及相关xml文件的代码：



图 2 参考案例的设计



1. 实验心得（50字）

通过这次课程我了解了安卓界面的布局配置学习了怎么通过代码来实现我需要的配置要求，接下来的课程我会更加认真，学习安卓的程序设计。