

[首页](#) [ANDROID](#) [互联网](#) [杂乱无章](#) [科技资讯](#) [程序员人生](#) [程序员笑话](#) [编程技术](#) [网址导航](#)

## 2.5.1 ListView Item多布局的实现

分类 **Android 基础入门教程**

**Android 基础入门教程(Q群号 : 153836263)**

### 本节引言：

本节是ListView这个小节的最后一节，给大家带来的是ListView多布局Item的实现，何为ListView Item多布局，打个比方，QQ这种聊天列表：



假如他是用一个ListView做的，那么一个ListView上不就就有两种不同的Item咯！一左一右，嘿嘿，本节就来教大家如何实现ListView的多布局！

- 1.0 Android基础入门教程
  - 1.0.1 2015年最新Android基...
- 1.1 背景相关与系统架构分析
- 1.2 开发环境搭建
  - 1.2.1 使用Eclipse + ADT + S...
  - 1.2.2 使用Android Studio开...
- 1.3 SDK更新不了问题解决
- 1.4 Genymotion模拟器安装
  - 1.5.1 Git使用教程之本地仓...
  - 1.5.2 Git之使用GitHub搭建...
- 1.6 .9(九妹)图片怎么玩
- 1.7 界面原型设计
- 1.8 工程相关解析(各种文件...
- 1.9 Android程序签名打包
- 1.11 反编译APK获取代码&...
- 2.1 View与ViewGroup的概念
  - 2.2.1 LinearLayout(线性布局)
  - 2.2.2 RelativeLayout(相对布...
  - 2.2.3 TableLayout(表格布局)
  - 2.2.4 FrameLayout(帧布局)
  - 2.2.5 GridLayout(网格布局)
  - 2.2.6 AbsoluteLayout(绝对...
- 2.3.1 TextView(文本框)详解
- 2.3.2 EditText(输入框)详解
- 2.3.3 Button(按钮)与ImageB...
- 2.3.4 ImageView(图像视图)
- 2.3.5.RadioButton(单选按钮...
- 2.3.6 开关按钮ToggleButton...
- 2.3.7 ProgressBar(进度条)
- 2.3.8 SeekBar(拖动条)
- 2.3.9 RatingBar(星级评分条)
- 2.4.1 ScrollView(滚动条)



反馈

## 1.要点讲解：

重写getItemViewType()方法对应View是哪个类别，以及getViewTypeCount()方法返回 总共多少个类别！然后再getView那里调用getItemViewType获得对应类别，再加载对应的View！

## 2.代码实现：

这里的话直接用上一节的两个布局，然后另外写一个Adapter重写要点中的几个地方：

**MutiLayoutAdapter.java：**

```
/**
 * Created by Jay on 2015/9/23 0023.
 */
public class MutiLayoutAdapter extends BaseAdapter{

    //定义两个类别标志
    private static final int TYPE_BOOK = 0;
    private static final int TYPE_APP = 1;
    private Context mContext;
    private ArrayList<Object> mData = null;

    public MutiLayoutAdapter(Context mContext,ArrayList<Object>
mData) {
        this.mContext = mContext;
        this.mData = mData;
    }

    @Override
    public int getCount() {
        return mData.size();
    }

    @Override
    public Object getItem(int position) {
        return mData.get(position);
    }

    @Override
    public long getItemId(int position) {
        return position;
    }

    //多布局的核心，通过这个判断类别
    @Override
    public int getItemViewType(int position) {
        if (mData.get(position) instanceof App) {
```

2.4.2 Date & Time组件(上)

2.4.3 Date & Time组件(下)

2.4.4 Adapter基础讲解

2.4.5 ListView简单实用

2.4.6 BaseAdapter优化

2.4.7 ListView的焦点问题

2.4.8 ListView之checkbox错...

2.4.9 ListView的数据更新问题

2.5.0 构建一个可复用的自定...

2.5.1 ListView Item多布局的...

2.5.2 GridView(网格视图)的...

2.5.3 Spinner(列表选项框)...

2.5.4 AutoCompleteTextVie...

2.5.5 ExpandableListView(...

2.5.6 ViewFlipper(翻转视图)...

2.5.7 Toast(吐司)的基本使用

2.5.8 Notification(状态栏通...

2.5.9 AlertDialog(对话框)详解

2.6.0 其他几种常用对话框基...

2.6.1 PopupWindow(悬浮框...

2.6.2 菜单(Menu)

2.6.3 ViewPager的简单使用

2.6.4 DrawerLayout(官方侧...

3.1.1 基于监听的事件处理机制

3.2 基于回调的事件处理机制

3.3 Handler消息传递机制浅析

3.4 TouchListener PK OnTo...

3.5 监听EditText的内容变化

3.6 响应系统设置的事件(Co...

3.7 AsyncTask异步任务

3.8 Gestures(手势)

4.1.1 Activity初学乍练

4.1.2 Activity初窥门径

4.1.3 Activity登堂入室

4.2.1 Service初涉

4.2.2 Service进阶

4.2.3 Service精通

4.3.1 BroadcastReceiver牛...

4.3.2 BroadcastReceiver庖...

4.4.1 ContentProvider初探

```

        return TYPE_APP;
    } else if (mData.get(position) instanceof Book) {
        return TYPE_BOOK;
    } else {
        return super.getItemViewType(position);
    }
}

//类别数目
@Override
public int getViewTypeCount() {
    return 2;
}

@Override
public View getView(int position, View convertView, ViewGroup parent) {
    int type = getItemViewType(position);
    ViewHolder1 holder1 = null;
    ViewHolder2 holder2 = null;
    if (convertView == null) {
        switch (type) {
            case TYPE_APP:
                holder1 = new ViewHolder1();
                convertView = LayoutInflater.from(mContext).
inflate(R.layout.item_one, parent, false);
                holder1.img_icon = (ImageView) convertView.f
indViewById(R.id.img_icon);
                holder1.txt_aname = (TextView) convertView.f
indViewById(R.id.txt_aname);
                convertView.setTag(R.id.Tag_APP, holder1);
                break;
            case TYPE_BOOK:
                holder2 = new ViewHolder2();
                convertView = LayoutInflater.from(mContext).
inflate(R.layout.item_two, parent, false);
                holder2.txt_bname = (TextView) convertView.f
indViewById(R.id.txt_bname);
                holder2.txt_bauthor = (TextView) convertView
.findViewById(R.id.txt_bauthor);
                convertView.setTag(R.id.Tag_Book, holder2);
                break;
        }
    } else {
        switch (type) {
            case TYPE_APP:
                holder1 = (ViewHolder1) convertView.getTag(
R.id.Tag_APP);
                break;
            case TYPE_BOOK:
                holder2 = (ViewHolder2) convertView.getTag(
R.id.Tag_Book);

```

[4.4.2 ContentProvider再探...](#)[4.5.1 Intent的基本使用](#)[4.5.2 Intent之复杂数据的传递](#)[5.1 Fragment基本概述](#)[5.2.1 Fragment实例精讲—...](#)[5.2.2 Fragment实例精讲—...](#)[5.2.3 Fragment实例精讲—...](#)[5.2.4 Fragment实例精讲—...](#)[5.2.5 Fragment实例精讲—...](#)[6.1 数据存储与访问之——文...](#)[6.2 数据存储与访问之——S...](#)[6.3.1 数据存储与访问之——...](#)[6.3.2 数据存储与访问之——...](#)[7.1.1 Android网络编程要学...](#)[7.1.2 Android Http请求头与...](#)[7.1.3 Android HTTP请求方...](#)[7.1.4 Android HTTP请求方...](#)[7.2.1 Android XML数据解析](#)[7.2.2 Android JSON数据解析](#)[7.3.1 Android 文件上传](#)[7.3.2 Android 文件下载 \( 1 \)](#)[7.3.3 Android 文件下载 \( 2 \)](#)[7.4 Android 调用 Webservice](#)[7.5.1 WebView\(网页视图\)基...](#)[7.5.2 WebView和JavaScrip...](#)[7.5.3 Android 4.4后WebVie...](#)[7.5.4 WebView文件下载](#)[7.5.5 WebView缓存问题](#)[7.5.6 WebView处理网页返...](#)[7.6.1 Socket学习网络基础准备](#)[7.6.2 基于TCP协议的Socket...](#)[7.6.3 基于TCP协议的Socket...](#)[7.6.4 基于UDP协议的Socke...](#)[8.1.1 Android中的13种Draw...](#)[8.1.2 Android中的13种Draw...](#)[8.1.3 Android中的13种Draw...](#)[8.2.1 Bitmap\(位图\)全解析 P...](#)[8.2.2 Bitmap引起的OOM问题](#)[8.3.1 三个绘图工具类详解](#)[8.3.2 绘图类实战示例](#)

```

        break;
    }
}

Object obj = mData.get(position);
//设置下控件的值
switch (type){
    case TYPE_APP:
        App app = (App) obj;
        if(app != null){
            holder1.img_icon.setImageResource(app.getaI
con());

            holder1.txt_aname.setText(app.getaName());
        }
        break;
    case TYPE_BOOK:
        Book book = (Book) obj;
        if(book != null){
            holder2.txt_bname.setText(book.getbName());
            holder2.txt_bauthor.setText(book.getbAuthor
());
        }
        break;
}

return convertView;
}

//两个不同的ViewHolder
private static class ViewHolder1{
    ImageView img_icon;
    TextView txt_aname;
}

private static class ViewHolder2{
    TextView txt_bname;
    TextView txt_bauthor;
}
}

```

这里有个地方要注意的，convertView.setTag(R.id.Tag\_APP,holder1);我们平时都直接setTag(Object)的，这个是setTag的重载方法，参数是一个唯一的key以及后面的一个对象！唯一！！！我一开始直接把TYPE\_BOOK作为第一个参数，然后就报下面这个错误：

```

23 14:09:21.280 30716-30716/? E/AndroidRuntime: FATAL EXCEPTION: main
Process: jay.com.listviewdemo6, PID: 30716
java.lang.IllegalArgumentException: The key must be an application-specific resource id.

```

**The key must be an application-specific resource id** 就是前面这个要唯一，定义一个final类型的int变量和硬编码一个值的方式都是行不通的 这里的做法是直接在strings.xml中添加：

8.3.3 Paint API之—— Mask...

8.3.4 Paint API之—— Xferm...

8.3.5 Paint API之—— Xferm...

8.3.6 Paint API之—— Xferm...

8.3.7 Paint API之—— Xferm...

8.3.8 Paint API之—— Xferm...

8.3.9 Paint API之—— Color...

8.3.10 Paint API之—— Colo...

8.3.11 Paint API之—— Colo...

8.3.12 Paint API之—— Path...

8.3.13 Paint API之—— Sha...

8.3.14 Paint几个枚举/常量值...

8.3.15 Paint API之——Type...

8.3.16 Canvas API详解(Part 1)

8.3.17 Canvas API详解(Part...

8.3.18 Canvas API详解(Part...

8.4.1 Android动画合集之帧...

8.4.2 Android动画合集之补...

8.4.3 Android动画合集之属...

8.4.4 Android动画合集之属...

9.1 使用SoundPool播放音...

9.2 MediaPlayer播放音频与...

9.3 使用Camera拍照

9.4 使用MediaRecord录音

10.1 TelephonyManager(电...

10.2 SmsManager(短信管理...

10.3 AudioManager(音频管...

10.4 Vibrator(振动器)

10.5 AlarmManager(闹钟服务)

10.6 PowerManager(电源服...

10.7 WindowManager(窗口...

10.8 LayoutInflater(布局服务)

10.9 WallpaperManager(壁...

10.10 传感器专题(1)——相...

10.11 传感器专题(2)——方...

10.12 传感器专题(3)——加...

10.12 传感器专题(4)——其...

10.14 Android GPS初涉

11.0 《2015最新Android基...

```
<item name="Tag_APP" type="id"></item>
<item name="Tag_Book" type="id"></item>
```

当然你也可以在res/values/下另外创建一个ids.xml文件，把上面这段代码贴上去！除了这个还有一个要注意的地方，就是这个区分类别的标志要从0开始算，不然会报下面这样的错误：

```
Process: jay.com.listviewdemo6, PID: 529
java.lang.ArrayIndexOutOfBoundsException: length=2; index=2
    at android.widget.AbsListView$RecycleBin.addScrapView(AbsListView.java:6563)
    at android.widget.AbsListView.trackMotionScroll(AbsListView.java:4916)
    at android.widget.AbsListView.scrollIfNeeded(AbsListView.java:3398)
```

### MainActivity.java :

```
public class MainActivity extends AppCompatActivity {

    private static final int TYPE_BOOK = 0;
    private static final int TYPE_APP = 1;
    private ListView list_content;
    private ArrayList<Object> mData = null;
    private MuliLayoutAdapter myAdapter = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //数据准备:
        mData = new ArrayList<Object>();
        for(int i = 0; i < 20; i++){
            switch ((int) (Math.random() * 2)) {
                case TYPE_BOOK:
                    mData.add(new Book("《第一行代码》", "郭霖"));
                    break;
                case TYPE_APP:
                    mData.add(new App(R.mipmap.iv_icon_baidu, "百度"));
                    break;
            }
        }

        list_content = (ListView) findViewById(R.id.list_content);
        myAdapter = new MuliLayoutAdapter(MainActivity.this, mData);
        list_content.setAdapter(myAdapter);
    }
}
```

上面随机生成0和1，0就往集合中添加一个Book的对象，1的话就添加一个App的对象！

### 3.代码下载：

[ListViewDemo6.zip](#)

### 本节小结：

好的，本节给大家讲解了ListView Item多布局的实现，就是两个方法的重写，然后getView()做下判断，设置不同的布局而已~代码非常简单~

关于ListView的知识就告一段落吧，当然ListView的知识并不止这些，异步加载，优化等等，这些我们都会在进阶部分进行学习~就说这么多，谢谢~

← 2.5.0 构建一个可复用的自定义BaseAdapter      2.5.2 GridView(网格视图)的基本使用 →



#### 在线实例

- HTML 实例
- CSS 实例
- JavaScript 实例
- Ajax 实例
- jQuery 实例
- XML 实例
- Java 实例

#### 字符集&工具

- HTML 字符集设置
- HTML ASCII 字符集
- HTML ISO-8859-1
- HTML 实体符号
- HTML 拾色器
- JSON 格式化工具

#### 最新更新

- PHP接收 json，并...
- Foundation CSS ...
- Foundation CSS ...
- Foundation 图标...
- Foundation 网格...
- Foundation 块状...
- Foundation 网格...

#### 站点信息

- 意见反馈
- 免责声明
- 关于我们
- 文章归档

#### 关注微信



Copyright © 2013-2015 菜鸟教程 [runoob.com](#) All Rights Reserved. 备案号：闽ICP备15012807号-1