

2.5.5 ExpandableListView(可折叠列表)的基本使用

分类 [Android 基础入门教程](#)

Android 基础入门教程(Q群号 : 153836263)

本节引言：

本节要讲解的Adapter类控件是ExpandableListView，就是可折叠的列表，它是ListView的子类，在ListView的基础上它把应用中的列表项分为几组，每组里又可包含多个列表项。至于样子，类似于QQ联系人列表，他的用法与ListView非常相似，只是ExpandableListVivew显示的列表项 需由ExpandableAdapter提供。下面我们来学习这个控件的基本使用！官方API：[ExpandableListView](#)

1.相关属性

android:childDivider：指定各组内子类表项之间的分隔条，图片不会完全显示，分离子列表项的是一条直线

android:childIndicator：显示在子列表旁边的Drawable对象，可以是一个图像

android:childIndicatorEnd：子列表项指示符的结束约束位置

android:childIndicatorLeft：子列表项指示符的左边约束位置

android:childIndicatorRight：子列表项指示符的右边约束位置

android:childIndicatorStart：子列表项指示符的开始约束位置

android:groupIndicator：显示在组列表旁边的Drawable对象，可以是一个图像

android:indicatorEnd：组列表项指示器的结束约束位置

android:indicatorLeft：组列表项指示器的左边约束位置

android:indicatorRight：组列表项指示器的右边约束位置

android:indicatorStart：组列表项指示器的开始约束位置



反馈

2.实现ExpandableAdapter的三种方式

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)

1. 扩展**BaseExpandableListAdapter**实现ExpandableAdapter。
2. 使用**SimpleExpandableListAdapter**将两个List集合包装成ExpandableAdapter
3. 使用**SimpleCursorTreeAdapter**将Cursor中的数据包装成SimpleCuroTreeAdapter 本节示例使用的是第一个，扩展BaseExpandableListAdapter，我们需要重写该类中的相关方法，下面我们通过一个代码示例来体验下！

3.代码示例

我们来看下实现的效果图：



下面我们就来实现上图的这个效果：

核心是重写**BaseExpandableListAdapter**，其实和之前写的普通的BaseAdapter是类似的，但是BaseExpandableListAdapter则分成了两部分：组和子列表，具体看代码你就知道了！

另外，有一点要注意的是，重写**isChildSelectable()**方法需要返回true，不然不会触发子Item的点击事件！下面我们来写写：

首先是组和子列表的布局：

item_exlist_group.xml：

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res
/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="5dp">

    <TextView
```

- 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 ViewPager(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

```

        android:id="@+id/tv_group_name"
        android:layout_width="match_parent"
        android:layout_height="56dp"
        android:gravity="center_vertical"
        android:paddingLeft="30dp"
        android:text="AP"
        android:textStyle="bold"
        android:textSize="20sp" />

```

```
</LinearLayout>
```

item_exlist_item.xml :

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res
/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="5dp"
    android:background="#6BBA79">

    <ImageView
        android:id="@+id/img_icon"
        android:layout_width="48dp"
        android:layout_height="48dp"
        android:src="@mipmap/iv_lol_icon1"
        android:focusable="false"/>

    <TextView
        android:id="@+id/tv_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
        android:layout_marginTop="15dp"
        android:focusable="false"
        android:text="提莫"
        android:textSize="18sp" />

</LinearLayout>

```

然后是自定义的Adapter类：

MyBaseExpandableListAdapter.java :

```

/**
 * Created by Jay on 2015/9/25 0025.
 */
public class MyBaseExpandableListAdapter extends BaseExpandable
ListAdapter {

    private ArrayList<Group> gData;
    private ArrayList<ArrayList<Item>> iData;

```

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JavaScip...

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

```
private Context mContext;

public MyBaseExpandableListAdapter(ArrayList<Group> gData, ArrayList<ArrayList<Item>> iData, Context mContext) {
    this.gData = gData;
    this.iData = iData;
    this.mContext = mContext;
}

@Override
public int getGroupCount() {
    return gData.size();
}

@Override
public int getChildrenCount(int groupPosition) {
    return iData.get(groupPosition).size();
}

@Override
public Group getGroup(int groupPosition) {
    return gData.get(groupPosition);
}

@Override
public Item getChild(int groupPosition, int childPosition) {
    return iData.get(groupPosition).get(childPosition);
}

@Override
public long getGroupId(int groupPosition) {
    return groupPosition;
}

@Override
public long getChildId(int groupPosition, int childPosition) {
    return childPosition;
}

@Override
public boolean hasStableIds() {
    return false;
}

//取得用于显示给定分组的视图。这个方法仅返回分组的视图对象
@Override
public View getGroupView(int groupPosition, boolean isExpanded, View convertView, ViewGroup parent) {

    ViewHolderGroup groupHolder;
    if(convertView == null){
        convertView = LayoutInflater.from(mContext).inflate
```

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...

```

(
    R.layout.item_exlist_group, parent, false);
    groupHolder = new ViewHolderGroup();
    groupHolder.tv_group_name = (TextView) convertView.
findViewById(R.id.tv_group_name);
    convertView.setTag(groupHolder);
} else {
    groupHolder = (ViewHolderGroup) convertView.getTag(
);
}
    groupHolder.tv_group_name.setText(gData.get(groupPositi
on).getGroupName());
    return convertView;
}

//取得显示给定分组给定子位置的数据用的视图
@Override
public View getChildView(int groupPosition, int childPositi
on, boolean isLastChild, View convertView, ViewGroup parent) {
    ViewHolderItem itemHolder;
    if (convertView == null) {
        convertView = LayoutInflater.from(mContext).inflate
(
            R.layout.item_exlist_item, parent, false);
        itemHolder = new ViewHolderItem();
        itemHolder.img_icon = (ImageView) convertView.findV
iewById(R.id.img_icon);
        itemHolder.tv_name = (TextView) convertView.findVie
wById(R.id.tv_name);
        convertView.setTag(itemHolder);
    } else {
        itemHolder = (ViewHolderItem) convertView.getTag();
    }
    itemHolder.img_icon.setImageResource(iData.get(groupPos
ition).get(childPosition).getiId());
    itemHolder.tv_name.setText(iData.get(groupPosition).get
(childPosition).getiName());
    return convertView;
}

//设置子列表是否可选中
@Override
public boolean isChildSelectable(int groupPosition, int chi
ldPosition) {
    return true;
}

private static class ViewHolderGroup{
    private TextView tv_group_name;
}

private static class ViewHolderItem{
    private ImageView img_icon;

```

```

        private TextView tv_name;

    }

}

```

PS：存储子列表的数据不一定要用ArrayList<ArrayList>这种，根据自己的需求定义~

最后是MainActivity的布局以及Java代码：

布局文件:activity_main.xml：

```

<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:padding="5dp"
    tools:context=".MainActivity">

    <ExpandableListView
        android:id="@+id/exlist_lol"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:childDivider="#E02D2F"/>

</RelativeLayout>

```

MainActivity.java：

```

public class MainActivity extends AppCompatActivity {

    private ArrayList<Group> gData = null;
    private ArrayList<ArrayList<Item>> iData = null;
    private ArrayList<Item> lData = null;
    private Context mContext;
    private ExpandableListView exlist_lol;
    private MyBaseExpandableListAdapter myAdapter = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mContext = MainActivity.this;
        exlist_lol = (ExpandableListView) findViewById(R.id.exlist_lol);

        //数据准备
        gData = new ArrayList<Group>();
        iData = new ArrayList<ArrayList<Item>>();
        gData.add(new Group("AD"));
    }
}

```

```

gData.add(new Group("AP"));
gData.add(new Group("TANK"));

lData = new ArrayList<Item>();

//AD组
lData.add(new Item(R.mipmap.iv_lol_icon3, "剑圣"));
lData.add(new Item(R.mipmap.iv_lol_icon4, "德莱文"));
lData.add(new Item(R.mipmap.iv_lol_icon13, "男枪"));
lData.add(new Item(R.mipmap.iv_lol_icon14, "韦鲁斯"));
iData.add(lData);
//AP组
lData = new ArrayList<Item>();
lData.add(new Item(R.mipmap.iv_lol_icon1, "提莫"));
lData.add(new Item(R.mipmap.iv_lol_icon7, "安妮"));
lData.add(new Item(R.mipmap.iv_lol_icon8, "天使"));
lData.add(new Item(R.mipmap.iv_lol_icon9, "泽拉斯"));
lData.add(new Item(R.mipmap.iv_lol_icon11, "狐狸"));
iData.add(lData);
//TANK组
lData = new ArrayList<Item>();
lData.add(new Item(R.mipmap.iv_lol_icon2, "诺手"));
lData.add(new Item(R.mipmap.iv_lol_icon5, "德邦"));
lData.add(new Item(R.mipmap.iv_lol_icon6, "奥拉夫"));
lData.add(new Item(R.mipmap.iv_lol_icon10, "龙女"));
lData.add(new Item(R.mipmap.iv_lol_icon12, "狗熊"));
iData.add(lData);

myAdapter = new MyBaseExpandableListAdapter(gData, iData, mContext);
exlist_lol.setAdapter(myAdapter);

//为列表设置点击事件
exlist_lol.setOnChildClickListener(new ExpandableListVi
ew.OnChildClickListener() {
    @Override
    public boolean onChildClick(ExpandableListView pare
nt, View v, int groupPosition, int childPosition, long id) {
        Toast.makeText(mContext, "你点击了: " + iData.ge
t(groupPosition).get(childPosition).getName(), Toast.LENGTH_SH
ORT).show();

        return true;
    }
});
}
}

```

4. 代码下载：

[ExpandableListViewDemo.zip](#)

本节小结：

好的，本节给大家介绍了ExpandableListView的基本使用，嘿嘿，有点意思~ 这里只是一个示例，其他的根据自己的需求自行扩展~谢谢

← 2.5.4 AutoCompleteTextView(自动完成文本框)的基本使用
2.5.6 ViewFlipper(翻转视图)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...
- Foundation 块状...
- Foundation 网...

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