

## 2.5.3 Spinner(列表选项框)的基本使用

分类 [Android 基础入门教程](#)

[Android 基础入门教程\(Q群号 : 153836263\)](#)

[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\(滚动条\)](#)

### 本节引言：

本来本节是想给大家介绍一个Gallery(画廊)的一个控件的，后来想想还是算了，因为在Android 4.1后就已经被弃用了，尽管我们可以通过兼容不来使用Gallery，不过想想 还是算了，因为Gallery在每次切换图片的时候，都需要重新创建视图，这样无疑会造成 很大资源浪费！我们可以通过其他方法来实现Gallery效果，比如通过HorizontalScrollView 来实现水平滚动效果，或者编写一个水平方向的ListView~有兴趣自己谷歌！

本节学习的是一个叫做Spinner的Adapter控件！应用场景：当我们的app需要用户输入数据时，除了让用户自己打字以外，还有一种比较贴心的设计：列出一组选项让用户从中挑选，从而方便了我们的用户！话不多说，开始学习Spinner的基本用法~

### 1.相关属性

**android:dropDownHorizontalOffset**：设置列表框的水平偏移距离

**android:dropDownVerticalOffset**：设置列表框的水平垂直距离

**android:dropDownSelector**：列表框被选中时的背景

**android:dropDownWidth**：设置下拉列表框的宽度

**android:gravity**：设置里面组件的对其方式

**android:popupBackground**：设置列表框的背景

**android:prompt**：设置对话框模式的列表框的提示信息(标题)，只能引用string.xml 中的资源id,而不能直接写字符串

**android:spinnerMode**：列表框的模式，有两个可选值：**dialog**：对话框风格的窗口 **dropdown**：下拉菜单风格的窗口(默认)

可选属性：**android:entries**：使用数组资源设置下拉列表框的列表项目

### 2.使用示例：

对了，Spinner会默认选中第一个值，就是默认调用 `spinner.setSelection(0)`，你可以通过这个设置默认的选中值，另外，会触发一次 `OnItemSelectedListener` 事件，暂时没找到解决方法，下面折衷的处理是：添加一个 `boolean` 值，然后设置 为 `false`，在 `onItemSelectedListener` 时进行判断，`false` 说明是默认触发的，不做任何操作 将 `boolean` 值设置为 `true`；`true` 的话则正常触发事件！示例中写了两个不同的Spinner，从数据源，列表框风格等进行对比~ 接下来我们来看下

效果图：



代码实现：

这里依然使用的我们前面的可复用 `BaseAdapter`：

第一个Spinner的数据源编写：

在 `res/values` 下编写一个：`myarrays.xml` 的文件，内容如下：

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="data">
        <item>英勇青铜</item>
        <item>不屈白银</item>
        <item>荣耀黄金</item>
        <item>华贵铂金</item>
        <item>璀璨钻石</item>
        <item>超凡大师</item>
        <item>最强王者</item>
    </string-array>
</resources>
```

接着是第二个Spinner的布局：`item_spin_hero.xml`：

- 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

```
<?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">

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

    <TextView
        android:id="@+id/txt_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="15dp"
        android:text="德玛西亚"
        android:textSize="16sp" />

</LinearLayout>
```



反馈

[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](#)

再接着编写一个Entitiy实体类：**Hero.java**：

```
/**
 * Created by Jay on 2015/9/24 0024.
 */
public class Hero {

    private int hIcon;
    private String hName;

    public Hero() {
    }

    public Hero(int hIcon, String hName) {
        this.hIcon = hIcon;
        this.hName = hName;
    }

    public int gethIcon() {
        return hIcon;
    }

    public String gethName() {
        return hName;
    }

    public void sethIcon(int hIcon) {
        this.hIcon = hIcon;
    }
}
```

```

    }

    public void sethName(String hName) {
        this.hName = hName;
    }
}

```

最后是MainActivity的布局与Java代码部分：

布局文件：**activity\_main.xml**：

```

<LinearLayout 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:orientation="vertical"
    android:padding="5dp"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="选择您的排位分段"
        android:textColor="#44BDED"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/spin_one"
        android:layout_width="100dp"
        android:layout_height="64dp"
        android:entries="@array/data"
        android:prompt="@string/spin_title"
        android:spinnerMode="dialog" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="选择你的拿手英雄~"
        android:textColor="#F5684A"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/spin_two"
        android:layout_width="wrap_content"
        android:layout_height="64dp" />

</LinearLayout>

```

**MainActivity.java**：

```

public class MainActivity extends AppCompatActivity implements

```

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...

```

AdapterView.OnItemClickListener {

    private Spinner spin_one;
    private Spinner spin_two;
    private Context mContext;
    //判断是否为刚进去时触发onItemSelected的标志
    private boolean one_selected = false;
    private boolean two_selected = false;
    private ArrayList<Hero> mData = null;
    private BaseAdapter myAdappter = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mContext = MainActivity.this;
        mData = new ArrayList<Hero>();
        bindViews();
    }

    private void bindViews() {
        spin_one = (Spinner) findViewById(R.id.spin_one);
        spin_two = (Spinner) findViewById(R.id.spin_two);

        mData.add(new Hero(R.mipmap.iv_lol_icon1, "迅捷斥候: 提莫 (Teemo) "));
        mData.add(new Hero(R.mipmap.iv_lol_icon2, "诺克萨斯之手: 德莱厄斯 (Darius) "));
        mData.add(new Hero(R.mipmap.iv_lol_icon3, "无极剑圣: 易 (Yi) "));
        mData.add(new Hero(R.mipmap.iv_lol_icon4, "德莱厄斯: 德莱文 (Draven) "));
        mData.add(new Hero(R.mipmap.iv_lol_icon5, "德邦总管: 赵信 (XinZhao) "));
        mData.add(new Hero(R.mipmap.iv_lol_icon6, "狂战士: 奥拉夫 (Olaf) "));

        myAdappter = new MyAdapter<Hero>(mData, R.layout.item_spin_hero) {
            @Override
            public void bindView(ViewHolder holder, Hero obj) {
                holder.setImageResource(R.id.img_icon, obj.getImageResource());
                holder.setText(R.id.txt_name, obj.getName());
            }
        };
        spin_two.setAdapter(myAdappter);
        spin_one.setOnItemClickListener(this);
        spin_two.setOnItemClickListener(this);
    }
}

```

```
@Override
public void onItemSelected(AdapterView<?> parent, View view
, int position, long id) {
    switch (parent.getId()){
        case R.id.spin_one:
            if(one_selected){
                Toast.makeText(mContext,"您的分段是~: " + parent.getItemAtPosition(position).toString(),
                    Toast.LENGTH_SHORT).show();
            }else one_selected = true;
            break;
        case R.id.spin_two:
            if(two_selected){
                TextView txt_name = (TextView) view.findViewById(R.id.txt_name);
                Toast.makeText(mContext,"您选择的英雄是~: " + txt_name.getText().toString(),
                    Toast.LENGTH_SHORT).show();
            }else two_selected = true;
            break;
    }
}

@Override
public void onNothingSelected(AdapterView<?> parent) {
}
}
```

另外关于Spinner的OnItemSelectedListener，以及如何获得选中项的值，就自己看上面的 代码啦~

### 3.代码示例下载：

[SpinnerDemo.zip](#)

### 本节小结

好的，本节给大家介绍了Spinner(下拉选项框)的使用，例子还是蛮有趣的，哈哈~！别问我哪个区什么段位，我可是人机小王子，可惜一



直在青铜分段苦苦挣扎~ 你知道为什么的，好吧，本节就到这里~

← 2.5.2 GridView(网格视图)的基本使用

2.5.4 AutoCompleteTextView(自动完成文本框)的基本使用 →



	<div>在线实例</div> <ul style="list-style-type: none"><li>· HTML 实例</li><li>· CSS 实例</li><li>· JavaScript 实例</li><li>· Ajax 实例</li><li>· jQuery 实例</li><li>· XML 实例</li><li>· Java 实例</li></ul>	<div>字符集&amp;工具</div> <ul style="list-style-type: none"><li>· HTML 字符集设置</li><li>· HTML ASCII 字符集</li><li>· HTML ISO-8859-1</li><li>· HTML 实体符号</li><li>· HTML 拾色器</li><li>· JSON 格式化工具</li></ul>	<div>最新更新</div> <ul style="list-style-type: none"><li>· PHP接收 json , 并...</li><li>· Foundation CSS ...</li><li>· Foundation CSS ...</li><li>· Foundation 图标...</li><li>· Foundation 网格...</li><li>· Foundation 块状...</li><li>· Foundation 网格...</li></ul>	<div>站点信息</div> <ul style="list-style-type: none"><li>· 意见反馈</li><li>· 免责声明</li><li>· 关于我们</li><li>· 文章归档</li></ul>
				<div>关注微信</div> <div></div> <div>Copyright © 2013-2015 菜鸟教程 runoob.com All Rights Reserved. 备案号：闽ICP备15012807号-1</div>