

Android 命名规范 (提高代码可读性)

作者: [hiquanta](#)

原文链接: <https://ld246.com/article/1491033741093>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

刚接触 [Android](https://ld246.com/forward?goto=http%3A%2F%2Flib.csdn.net%2Fbase%2Fandroid "Android知识库") 的时候命名都是按照拼音来，所以有的时候想看懂命名的那个控件是什么用的，就要读一遍甚至好几遍知道，这样的话，在代码的审查和修改过程中就会浪费不少不必要的时间。如果就是我一个人开发，个人维护的话还好，可是如果一个项目是团队分工合作，这样让你的同事去看你的代码就更加吃力了因为大家之间的编程方式不一样，所以，在开发过程中，命名规范统一尤为重要，最好是团队中统一大家命名方法，这样对于日后的工作会轻松很多。

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> 在面试的时候，审核一个程序员的编程水平的时候，命名规范也是一大标准，所以，下面我就结下我从网上别人的经验和自己的感悟中收集总结的android命名规范，希望对大家有所帮助。</span></span><span class="highlight-line"><span class="highlight-cl"></span></span><span class="highlight-line"><span class="highlight-cl"> 提醒：在命名规的统一下，在加上注释，这样日后维护会高效快捷好多。</span></span></code></pre>
```

先初略介绍下当前主要的标识符命名法和英文缩写规则：（这段别人那边借鉴的）

在讲解命名规范前，先初略介绍下当前主要的标识符命名法和英文缩写规则。

标识符命名法

标识符命名法最要有四种：

1 驼峰(Camel)命名法:又称小驼峰命名法，除首单词外，其余所有单词的第一个字母大写。

2 帕斯卡(pascal)命名法:又称大驼峰命名法，所有单词的第一个字母大写

3 下划线命名法:单词与单词间用下划线做间隔。

4 匈牙利命名法:广泛应用于微软编程环境中，在以 Pascal 命名法的变量前附加小写序列说明该量的类型。量的取名方式为：+ + 范围前缀，类型前缀，限定词。

个人觉得标识符命名原则：**尽可能的用最少的字符而又能完整的表达标识符的含义。**

英文缩写原则：

1 较短的单词可通过去掉“元音”形成缩写

2 较长的单词可取单词的头几个字母形成缩写

3 此外还有一些约定成俗的英文单词缩写。

下面为常见的英文单词缩写：

|

名称

|

缩写

|

icon

|

ic（主要用在 app 的图标）

|

color

|

cl（主要用于颜色值）

|

divider

|

di（主要用于分隔线，不仅包括 Listview 中的 divider，还包括普通布局中的线）

|

selector

|

|

<p>sl (主要用于某一 view 多种状态, 不仅包括 Listview 中的 selector, 还包括按钮的 selector) <p>
<p>|

|</p>
<p>average</p>
<p>|</p>
<p>avg</p>
<p>|

|</p>
<p>background</p>
<p>|</p>
<p>Bg (主要用于布局和子布局的背景) </p>
<p>|

|</p>
<p>buffer</p>
<p>|</p>
<p>buf</p>
<p>|

|</p>
<p>control</p>
<p>|</p>
<p>ctrl</p>
<p>|

|</p>
<p>delete</p>
<p>|</p>
<p>del</p>
<p>|

|</p>
<p>document</p>
<p>|</p>
<p>doc</p>
<p>|

|</p>
<p>error</p>
<p>|</p>
<p>err</p>
<p>|

|</p>
<p>escape</p>
<p>|</p>
<p>esc</p>
<p>|

|</p>
<p>increment</p>
<p>|</p>
<p>inc</p>
<p>|

|</p>
<p>infomation</p>
<p>|</p>
<p>info</p>
<p>|

|</p>

<p>initial</p>
<p>|</p>
<p>init</p>
<p>|

|</p>
<p>image</p>
<p>|</p>
<p>img</p>
<p>|

|</p>
<p>Internationalization</p>
<p>|</p>
<p>l18N</p>
<p>|

|</p>
<p>length</p>
<p>|</p>
<p>len</p>
<p>|

|</p>
<p>library</p>
<p>|</p>
<p>lib</p>
<p>|

|</p>
<p>message</p>
<p>|</p>
<p>msg</p>
<p>|

|</p>
<p>password-</p>
<p>|</p>
<p>pwd</p>
<p>|

|</p>
<p>position</p>
<p>|</p>
<p>pos</p>
<p>|

|</p>
<p>server</p>
<p>|</p>
<p>srv</p>
<p>|

|</p>
<p>string</p>
<p>|</p>
<p>str</p>
<p>|

|</p>
<p>temp</p>
<p>|</p>
<p>tmp</p>
<p>|

|</p>
<p>window</p>
<p>|</p>
<p>wnd(win)</p>
<p>|</p>
<p>程序中使用单词缩写原则：不要用缩写，除非该缩写是约定俗成的。</p>
<p>命名规范：</p>
<p>1 包 (packages) : 采用反域名命名规则，全部使用小写字母。一级包名为 com，二级包名为 x
(可以是公司或则个人的随便)，三级包名根据应用进行命名，四级包名为模块名或层级名</p>
<p>|</p>
<p>包名</p>
<p>|</p>
<p>此包中包含</p>
<p>|

</p>
<p>com.xx.应用名称缩写.activities</p>
<p>|</p>
<p>页面用到的 Activity 类 (activities 层级名用户界面层)</p>
<p>|

</p>
<p>com.xx.应用名称缩写.base</p>
<p>|</p>
<p>页面中每个 Activity 类共享的可以写成一个 i 额 BaseActivity 类 (基础共享的类)</p>
<p>|

</p>
<p>com.xx.应用名称缩写.adapter</p>
<p>|</p>
<p>页面用到的 Adapter 类 (适配器的类)</p>
<p>|

</p>
<p>com.xx.应用名称缩写.tools</p>
<p>|</p>
<p>此包中包含：公共工具方法类 (tools 模块名) </p>
<p>|

</p>
<p>com.xx.应用名称缩写.bean</p>
<p>(或则 com.xx.应用名称缩写.unity)</p>
<p>|</p>
<p>此包中包含：元素类</p>
<p>|

</p>
<p>com.xx.应用名称缩写.db</p>
<p>|</p>
<p>数据库操作类</p>
<p>|

</p>
<p>com.xx.应用名称缩写.view</p>
<p>(或则 com.xx.应用名称缩写.ui)</p>
<p>|</p>
<p>自定义的 View 类等</p>
<p>|

</p>
<p>com.xx.应用名称缩写.service</p>

<p>|</p>
<p>Service 服务</p>
<p>|

</p>
<p>com.xx.应用名称缩写.broadcast</p>
<p>|</p>
<p>Broadcast 服务</p>
<p>|</p>
<p>2 类 (classes) : 名词, 采用大驼峰命名法, 尽量避免缩写, 除非该缩写是众所周知的, 比如 TML,URL, 如果类名称中包含单词缩写, 则单词缩写的每个字母均应大写。</p>
<p>|</p>
<p>类</p>
<p>|</p>
<p>描述</p>
<p>|</p>
<p>例如</p>
<p>|

</p>
<p>activity 类</p>
<p>|</p>
<p>Aty 或者 Activity 为后缀标识</p>
<p>|</p>
<p>欢迎页面类 WelcomeAty.或者 WelcomeActivity</p>
<p>|

</p>
<p>Adapter 类</p>
<p>|</p>
<p>Adp 或者 Adapte 为后缀标识</p>
<p>|</p>
<p>新闻详情适配器 NewtDetailAdp 或则直接 NewDetailAdapter</p>
<p>|

</p>
<p>解析类</p>
<p>|</p>
<p>Hlr 为后缀标识</p>
<p>|</p>
<p>首页解析类 HomePosterHlr</p>
<p>|

</p>
<p>公共方法类</p>
<p>|</p>
<p>Tools 或 Manager 为后缀标识</p>
<p>|</p>
<p>线程池管理类: ThreadPoolManager

日志工具类: LogTools</p>
<p>|

</p>
<p>数据库类</p>
<p>|</p>
<p>以 DBHelper 后缀标识</p>
<p>|</p>
<p>新闻数据库: NewDBHelper</p>
<p>|

</p>

<p>Service 类</p>
<p>|</p>
<p>以 Service 为后缀标识</p>
<p>|</p>
<p>时间服务 TimeService</p>
<p>|

</p>
<p>BroadcastReceive 类</p>
<p>|</p>
<p>以 Broadcast 为后缀标识</p>
<p>|</p>
<p>时间通知 TimeBroadcast</p>
<p>|

</p>
<p>ContentProvider</p>
<p>|</p>
<p>以 Provider 为后缀标识</p>
<p>| |

</p>
<p>直接写的共享基础类</p>
<p>|</p>
<p>以 Base 开头</p>
<p>|</p>
<p>BaseActivity,BaseFragment</p>
<p>|</p>
<p>3 接口 (interface) : 命名规则与类一样采用大驼峰命名法, 多以 able 或 ible 结尾, 如 interface Runnable ;

interface Accessible 。</p>
<p>4 方法 (methods) : 动词或动名词, 采用小驼峰命名法例如: onCreate(),run()</p>
<p>|</p>
<p>方法</p>
<p>|</p>
<p>说明</p>
<p>|

</p>
<p>initXX()</p>
<p>|</p>
<p>初始化相关方法,使用 init 为前缀标识, 如初始化布局 initView()</p>
<p>|

</p>
<p>isXX()</p>
<p>|</p>
<p>checkXX()方法返回值为 boolean 型的请使用 is 或 check 为前缀标识</p>
<p>|

</p>
<p>getXX()</p>
<p>|</p>
<p>返回某个值的方法, 使用 get 为前缀标识</p>
<p>|

</p>
<p>processXX()</p>
<p>|</p>
<p>对数据进行处理的方法, 尽量使用 process 为前缀标识</p>
<p>|

|</p>
<p>displayXX()</p>
<p>|</p>
<p>弹出提示框和提示信息，使用 display 为前缀标识</p>
<p>|

</p>
<p>saveXX()</p>
<p>|</p>
<p>与保存数据相关的，使用 sav 为 e 前缀标识</p>
<p>|

</p>
<p>resetXX()</p>
<p>|</p>
<p>对数据重组的，使用 reset 前缀标识</p>
<p>|

</p>
<p>clearXX()</p>
<p>|</p>
<p>清除数据相关的</p>
<p>|

</p>
<p>removeXXX()</p>
<p>|</p>
<p>清除数据相关的</p>
<p>|

</p>
<p>drawXXX()</p>
<p>|</p>
<p>绘制数据或效果相关的，使用 draw 前缀标识</p>
<p>|</p>
<p>5 变量 (variables) 采用小驼峰命名法。类中控件名称必须与 xml 布局 id 保持一致。</p>
<p>用统一的量词通过在结尾处放置一个量词，就可创建更加统一的变量，它们更容易理解，也更容易搜索。例如，请使用 strCustomerFirst 和 strCustomerLast，而不要使用 strFirstCustomer 和 strLastCustomer。

<p>量词列表：量词后缀说明

First 一组变量中的第一个

Last 一组变量中的最后一个

Next 一组变量中的下一个变量

Prev 一组变量中的上一个

Cur 一组变量中的当前变量</p>
<p>6 常量 (Constants) 全部大写,采用下划线命名法.例如: MIN_WIDTH</p>
<p>7 资源文件 (图片 drawable 文件夹下) : 全部小写, 采用下划线命名法, 加前缀区分</p>
<p>命名模式: activity 名称_逻辑名称/common_逻辑名称</p>
<p>如果有多种形态如按钮等除外如 btn_xx.xml (selector) </p>
<p>|</p>
<p>名称 </p>
<p>|</p>
<p>功能 </p>
<p>|

</p>
<p>btn_xx</p>
<p>|</p>
<p>按钮图片使用 btn_整体效果** (selector) **</p>
<p>|


```
|</p>
<p>btn_xx_normal</p>
<p>|</p>
<p>按钮图片使用 btn_正常情况效果</p>
<p>|<br>
|</p>
<p>btn_xx_press</p>
<p>|</p>
<p>按钮图片使用 btn_点击时候效果</p>
<p>|<br>
|</p>
<p>bg_head</p>
<p>|</p>
<p>背景图片使用 bg_功能_说明</p>
<p>|<br>
|</p>
<p>def_search_cell</p>
<p>|</p>
<p>默认图片使用 def_功能_说明</p>
<p>|<br>
|</p>
<p>icon_more_help</p>
<p>|</p>
<p>图标图片使用 icon_功能_说明</p>
<p>|<br>
|</p>
<p>seg_list_line</p>
<p>|</p>
<p>具有分隔特征的图片使用 seg_功能_说明</p>
<p>|<br>
|</p>
<p>sel_ok</p>
<p>|</p>
<p>选择图标使用 sel_功能_说明</p>
<p>|</p>
<p>命名后缀: </p>
<p>|</p>
<p><strong>后缀</strong> </p>
<p>|</p>
<p><strong>说明</strong> </p>
<p>|<br>
|</p>
<p>unit</p>
<p>|</p>
<p>在使用 xml 的 tilemode 来配图片时, element 图片使用此后缀</p>
<p>|<br>
|</p>
<p>nor</p>
<p>|</p>
<p>图片的状态, 代表普通状态</p>
<p>|<br>
|</p>
<p>hl</p>
<p>|</p>
```

<p>图片的状态, 代表高亮状态</p>

<p>|

|</p>

<p>press</p>

<p>|</p>

<p>图片的状态, 代表按下状态</p>

<p>|

|</p>

<p>select</p>

<p>|</p>

<p>图片的状态, 代表其所占的 view 被选中</p>

<p>|

|</p>

<p>unselect</p>

<p>|</p>

<p>图片的状态, 代表其所占的 view 没有被选中</p>

<p>|</p>

<hr>

<p>8 资源布局文件 (XML 文件 (layout 布局文件)) : </p>

<p>全部小写, 采用下划线命名法</p>

<p>1). contentview 命名, Activity 默认布局, 以去掉后缀的 Activity 类进行命名。不加后缀: </p>

<p>** 功能模块.xml

**</p>

<pre><code class="highlight-chroma"> 例如: main.xml、more.xml、settings.xml

 或则: activity_

能模块.xml

 例如: **activity

main.xml、activity_more.xml**

</code></pre>

<p>2). Dialog 命名: dialog_描述.xml</p>

<pre><code class="highlight-chroma"> 例如: **dlg_hint.xml**

</code></pre>

<p>2). PopupWindow 命名: ppw_描述.xml</p>

<pre><code class="highlight-chroma"> 例如: **ppw_info.xml**

</code></pre>

<p>3). 列表项命名 listitem_描述.xml</p>

<pre><code class="highlight-chroma"> 例如: **listitem_city.xml**

</code></pre>

<p>4). 包含项: include_模块.xml</p>

<pre><code class="highlight-chroma"> 例如: **include_head.xml、include_bottom.xml**

</code></pre>

<p>5). adapter 的子布局: 功能模块_item.xml</p>

<pre><code class="highlight-chroma"> 例如: **main_item.xml、**

</code></pre>

<p>9 动画文件 (anim 文件夹下) : 全部小写, 采用下划线命名法, 加前缀区分。</p>

<p>//前面为动画的类型，后面为方向</p>
<p>|</p>
<p>动画命名例子</p>
<p>|</p>
<p>规范写法</p>
<p>|</p>
<p>备注</p>
<p>|

|</p>
<p>fade_in</p>
<p>|</p>
<p>淡入</p>
<p>|</p>
<p>|

|</p>
<p>fade_out</p>
<p>|</p>
<p>淡出</p>
<p>|</p>
<p>|

|</p>
<p>push_down_in</p>
<p>|</p>
<p>从下方推入</p>
<p>|</p>
<p>|

|</p>
<p>push_down_out</p>
<p>|</p>
<p>从下方推出</p>
<p>|</p>
<p>|

|</p>
<p>push_left</p>
<p>|</p>
<p>推像左方</p>
<p>|</p>
<p>|

|</p>
<p>slide_in_from_top</p>
<p>|</p>
<p>从头部滑动进入</p>
<p>|</p>
<p>|

|</p>
<p>zoom_enter</p>
<p>|</p>
<p>变形进入</p>
<p>|</p>
<p>|

|</p>
<p>slide_in</p>
<p>|</p>
<p>滑动进入</p>

<p>|

|</p>
<p>shrink_to_middle</p>
<p>|</p>
<p>中间缩小</p>
<p>| </p>
<p>10 资源 ID (resourcesid) : 大小写规范与方法名一致, 采用小驼峰命名法。命名规范为“资
控件的缩写 名” + “变量名”。注意: 页面控件名称应该和控件 id 名保持一致
</p>
<p>** strings.xml, colors.xml 等中的 id 命名: **</p>
<p>命名模式: activity 名称_功能模块名称_逻辑名称/activity 名称_逻辑名称/common_逻辑名称</
>
<p>strings.xml 中, 使用 activity 名称注释, 将文件内容区分开来</p>
<p>11 layout 中的 id 命名</p>
<p>命名模式为: view 缩写_模块名称_view 的逻辑名称</p>
<p>view 的缩写详情如下: </p>
<p>|</p>
<p>控件 </p>
<p>|</p>
<p>缩写 </p>
<p>|

|</p>
<p>LayoutView</p>
<p>|</p>
<p>lv</p>
<p>|

|</p>
<p>RelativeLayout</p>
<p>|</p>
<p>rv</p>
<p>|

|</p>
<p>TextView</p>
<p>|</p>
<p>tv</p>
<p>|

|</p>
<p>Button</p>
<p>|</p>
<p>btn</p>
<p>|

|</p>
<p>ImageButton</p>
<p>|</p>
<p>imgBtn</p>
<p>|

|</p>
<p>ImageView</p>
<p>|</p>
<p>mgView 或则 iv</p>
<p>|

|</p>
<p>CheckBox</p>
<p>|</p>

<p>chk</p>
<p>|

|</p>
<p>RadioButton</p>
<p>|</p>
<p>rdoBtn</p>
<p>|

|</p>
<p>analogClock</p>
<p>|</p>
<p>anaClk</p>
<p>|

|</p>
<p>DigitalClock</p>
<p>|</p>
<p>dgtClk</p>
<p>|

|</p>
<p>DatePicker</p>
<p>|</p>
<p>dtPk</p>
<p>|

|</p>
<p>EditText</p>
<p>|</p>
<p>edtTxt</p>
<p>|

|</p>
<p>TimePicker</p>
<p>|</p>
<p>tmPk</p>
<p>|

|</p>
<p>toggleButton</p>
<p>|</p>
<p>tglBtn</p>
<p>|

|</p>
<p>ProgressBar</p>
<p>|</p>
<p>proBar</p>
<p>|

|</p>
<p>SeekBar</p>
<p>|</p>
<p>skBar</p>
<p>|

|</p>
<p>AutoCompleteTextView</p>
<p>|</p>
<p>autoTxt</p>
<p>|

|</p>
<p>ZoomControls</p>

<p>|</p>
<p>zmCtl</p>
<p>|

|</p>
<p>VideoView</p>
<p>|</p>
<p>vdoVi</p>
<p>|

|</p>
<p>WdbView</p>
<p>|</p>
<p>webVi</p>
<p>|

|</p>
<p>RantingBar</p>
<p>|</p>
<p>ratBar</p>
<p>|

|</p>
<p>Tab</p>
<p>|</p>
<p>tab</p>
<p>|

|</p>
<p>Spinner</p>
<p>|</p>
<p>spn</p>
<p>|

|</p>
<p>Chronometer</p>
<p>|</p>
<p>cmt</p>
<p>|

|</p>
<p>ScollView</p>
<p>|</p>
<p>sclVi</p>
<p>|

|</p>
<p>TextSwitch</p>
<p>|</p>
<p>txtSwt</p>
<p>|

|</p>
<p>ImageSwitch</p>
<p>|</p>
<p>imgSwt</p>
<p>|

|</p>
<p>listView</p>
<p>|</p>
<p>lVi 或则 lv</p>
<p>|

|</p>

<p>ExpandableList</p>
<p>|</p>
<p>epdLt</p>
<p>|

</p>
<p>MapView</p>
<p>|</p>
<p>mapVi</p>
<p>|</p>
<p>12.activity 中的 view 变量命名</p>
<p>命名模式为：逻辑名称 +view 缩写</p>
<p>建议：如果 layout 文件很复杂，建议将 layout 分成多个模块，每个模块定义一个 moduleView
older，其成员变量包含所属 view</p>
<p>13.styles.xml：将 layout 中不断重现的 style 提炼出通用的 style 通用组件，放到 styles.xml
；</p>
<p>14.使用 layer-list 和 selector</p>
<p>Android 编码规范建议（别人弄的觉得蛮有道理）</p>
<p>1.<a href="https://ld246.com/forward?goto=http%3A%2F%2Flib.csdn.net%2Fbase%2Fja
ase" title="Java SE知识库" target="_blank" rel="nofollow ugc">Java 代码中不出现中文，
多注释中可以出现中文</p>
<p>2.局部变量命名、静态成员变量命名</p>
<p>只能包含字母，单词首字母除第一个外，都为大写，其他字母都为小写</p>
<p>3.常量命名</p>
<p>只能包含字母和_，字母全部大写，单词之间用_隔开</p>
<p>4.图片尽量分拆成多个可重用的图片</p>
<p>5.服务端可以实现的，就不要放在客户端</p>
<p>6.引用第三方库要慎重，避免应用大容量的第三方库，导致客户端包非常大</p>
<p>7.处理应用全局异常和错误，将错误以邮件的形式发送给服务端</p>
<p>8.图片的.9 处理</p>
<p>9.使用静态变量方式实现界面间共享要慎重</p>
<p>10.Log(系统名称模块名称接口名称，详细描述)</p>
<p>11.单元<a href="https://ld246.com/forward?goto=http%3A%2F%2Flib.csdn.net%2Fbase
2Fsoftwaretest" title="软件测试知识库" target="_blank" rel="nofollow ugc">测试（逻辑
测试、界面测试）</p>
<p>12.不要重用父类的 handler，对应一个类的 handler 也不应该让其子类用到，否则会导致 mess
ge.what 冲突</p>
<p>13.activity 中在一个 View.OnClickListener 中处理所有的逻辑</p>
<p>14.strings.xml 中使用 %1\$s 实现字符串的通配</p>
<p>15.如果多个 Activity 中包含共同的 UI 处理，那么可以提炼一个 CommonActivity，把通用部
分由它来处理，其他 activity 只要继承它即可</p>
<p>16.使用 button+activitygroup 实现 tab 效果时，使用 Button.setSelected(true)，确保按钮处
选择状态，并使 activitygroup 的当前 activity 与该 button 对应</p>
<p>17.如果所开发的为通用组件，为避免冲突，将 drawable/layout/menu/values 目录下的文件
增加前缀</p>
<p>18.数据一定要效验，例如</p>
<p>字符型转数字型，如果转换失败一定要有缺省值；</p>
<p>服务端响应数据是否有效判断</p>
<p>** 最后，这些都是些个人总结和网络上看到的其他网友的建议，**</p>
<p>** 仅供参考，只要形成一个团体上的统一即可，简单知道他的意思就可以了。**</p>
<p>** 程序员养成良好的编码习惯和命名规范，对于自己和整个团队都有好处。 **</p>