命名规范和编码规范

Revision Record 修订记录

| Date  日期 | Revision Version  修订版本 | CR ID  CR号 | Sec No.  修改章节 | Change Description  修改描述 | Author  作者 |
| --- | --- | --- | --- | --- | --- |
| 2017-05-10 | 0.0.1 |  |  | 添加命名规范和编码规范 | 魏群 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

[代码规范 2](#_Toc28277)

[一命名规范 2](#_Toc13469)

[Java类文件命名规范。 2](#_Toc21547)

[资源文件命名规范 2](#_Toc32226)

[java类中控件对象的命名规范。 2](#_Toc3143)

[strings.xml中常量的命名规范。 3](#_Toc4197)

[常量命名 3](#_Toc16469)

[二编码规范 3](#_Toc23833)

代码规范

## 一命名规范

### Java类文件命名规范。

1. Activity命名规范：以Acivity作为后缀. Eg: SettingActivity.
2. Adapter命名规范： 以Adapter作为后缀. Eg:SettingAdapter.
3. Entity 命名规范：大多数以Entity最为后缀. Eg:PersonEntity. 当作为全局变量时不算是实体，不受约束，如User。

### 资源文件命名规范

1. 页面布局文件。以act\_为前缀，已Activity的名称（去掉Activity后缀）作为后缀。（都是小写）。 Eg：act\_setting.xml
2. ListView中的item布局文件。以item\_作为固定前缀，列表项的名称为后缀。（都是小写）。Eg：某页面下有用户列表，空间名为lvUserList 则 item的layout应该都是：

Item\_lvUserList.xml.

1. Dialog布局文件。以dlg\_作为固定前缀，Dialog的功能名称为后缀（都是小写）Eg：dlg\_hint.xml
2. 部分xml文件，对于只在一个页面使用的资源，就以该页面的名称作为前缀。
3. 对于在各个模块，各个页面都有可能使用的资源，比如上下导航以common\_作为前缀。

### java类中控件对象的命名规范。

控件类型缩写+控件的逻辑名称（首字母大写）Eg：登录按钮，命名为btnLogin.

常用控件缩写

LayoutView lv

EditText et

RelativeView rv

TimePicker tp

TextView tv

Button btn

ImageButton img

ImageView iv

CheckButton cb

RadioButton rb

toggleButton tb

ProgressBar pb

WebView wv

RantingBar rb

CheckBox cb

Tab tab

RadioButton rb

ListView lv

RecycleView rv

CardView cv

### strings.xml中常量的命名规范。

1. 这些值大多在Layout中的控件上使用，所以以该常量所在的Activity名称作为前缀，后面跟控件名称。Eg 登录页面的登录按钮上的文字，loginActivity\_btnLogin\_text.
2. 如果涉及和公共模块和控件相关，就以common\_作为前缀。

### 常量命名

准守Java的命名规范，只能包含字母和下划线\_,字母全部大写,单词之间用下划线\_隔开.

## 二编码规范

1. 控件的声明要放在Activity级别。
2. Layout中的常量,要配置在strings.xml中。参见string.xml中的命名规范
3. Layout中的字体大小，都定义在dimens.xml中。参见string.xml中的命名规范
4. 页面之间传值，请使用Intent携带序列化实体数据的方式。
5. 为控件添加事件，请同意使用如下方式

btnLogin = (Button)findViewById(R.id.btnLogin);

btnLogin.setOnclickListener(

new View.OnClickListener{

@Override

Public void onClick(View v) {

login();

}

});

6.Activity中尽量不要嵌套内部类。

7.所有Adapter都放在adapter中，Aapter绑定的数据，一律为ArrayList<自定义可序列化实体>.在Adapter中创建适合于列表自身的ViewHolder实体类，请统一命名为ViewHolder。

8.实体类不要在不同模块间共享，但可以在同一模块的不用页面间共享。

9.为节省内存，请尽量使用ArrayList<自定义实体> 而不是HashMap

10.图片处理，请统一使用第三方组件（具体问题具体分析）。

11.在不涉及UI方面的时候请用ApplicationContext代替Context 避免内存泄露.

12.使用常量代替枚举.

13.在Activity中声明Handler时，请使用静态时+软引用 避免内存泄露.