

IDEA使用小窍门 提高开发效率

一. 关于版本管理

- 在团队开发中,目前大部分公司还是使用svn为主,有一个问题是使用svn的时候,如果没有设置忽略提交的配置,每次提交都会包括很大一部分的中间临时文件,而我们则需要从中过滤出不要提交的和不要提交的内容。甚至有时候在心情烦躁的时候,误操作把*.iml这种idea的项目配置文件也给提交到svn上。
- 通过这个教程<http://www.jianshu.com/p/0f9624043636>,把不需要的过滤掉,就能让你的svn提交变的很清晰。

二. toString的自定义

- 开发中经常会想,字符串与对象之间能互相转换,那会不会方便很多呢,存储变量就保存个字符串就可以了,然后toString就是我想到的方式,实现字符串与对象之间的转换,自己去写toString来实现对象的json话,那实现太麻烦了,用第三方的json库倒是也可以,不过最轻量级的当然是通过自定义toString的模版啦。给出模版代码:

```

public java.lang.String toString() {
    final java.lang.StringBuilder sb = new java.lang.StringBuilder("{}");
    #set ($i = 0)
    #foreach ($field in $fields)
    #set ($nullable = $field.object || $field.array)
    #if ($nullable)
    if ($field.accessor != null) {
    #end
    #set ($quotation = $field.string || $field.date)
    #if ($i == 0)
    sb.append("####")
    #else
    sb.append(",###")
    #end
    #if ($quotation)
    \"$field.name\": \"\"")
    #else
    \"$field.name\": \"")
    #end
    #if ($field.array)
    .append(java.util.Arrays.toString($field.name))
    #elseif ($field.list)
    .append(java.util.Arrays.toString( $field.accessor .toArray()) )
    #elseif ($field.string || $field.date)
    .append($field.accessor)
    #else
    .append($field.accessor)
    #end
    #if($quotation)
    .append("\"");
    #else
    .append("");
    #end
    #if ($nullable)
    }
    #end
    #set ($i = $i + 1)
    #end
    sb.append('}');
    return sb.toString().replace("null","");
}

```

模版中内置的变量名对应的含义,参照IntelliJ IDEA的官网定义说的很清楚

<https://www.jetbrains.com/help/idea/2017.1/generate-tostring-settings-dialog.html?search=primitiveArray>

三. 提高效率的开发插件

- GsonFormat : 第二条说的是怎么实现对象转换成字符串, 这里这个就厉害了, 转换json字符串到实体对象, 方便创建对象, toString使用第二条定义的模版, 实现对象与字符串之间的通过BeanFactory实现无缝转换。**说到对象的定义, 以后定义的实体类都实现Parcelable接口, 在定义完成成员变量后, 再进行实现调用idea提供的功能, 自动生成需要的方法与变量**
- FindViewById : 这个也很厉害啦, 这个是我从tv端里看到华哥推荐的, 然后用了一下, 确实不错, 使用ButterKnife确实也挺便捷的, 但是我还是喜欢用原生的方式, 毕竟ButterKnife会让idea加一些额外插件, 时不时的注解可能也会变。**这个的依赖是xml中命名更规范, 这个要注意**
- Android Styler: 这个也厉害了, 就是我们在布局的时候, 对可能会重用的属性会提取出来, 定义一个样式, 程序员就是为了偷懒而生的(应该理解成用更高效的方式实现相同的事, 别理解错了), 那么, 我不想一条条自己写, 然后又看到了这个插件, 贼jb爽。拷贝xml定义的属性, 拷贝到style.xml文件夹, 选中之后, 按下快捷键(我的是梅花键+shift+D), 输入名字, 就好啦。

“

当然, 还有很多很好的插件, 自己也可以根据需求自己做插件 脑洞有多大, 效果有多高, google 一下“idea 插件开发”, 就有很多教程, 由于时间关系, 我没有去深入。

LiveTemplates

- 页面模版: 还记得我们怎么创建app模块的吗? 是不是有很多内置的activity可以给你选择, 在项目中, 右键也可以创建activity, 选择模版, 这些其实也是可以自定义的, 在项目重构的时候, 想要配置一个下拉刷新的activity的模版, 限于时间原因, 没有弄, 使用的另外一种方式, 下面会提到。
<http://blog.csdn.net/lmj623565791/article/details/51635533>
- 还有一种是我们常用的, 就是按几个字母就出来的, 比如按foreach会出来的循环一样, 这个也是可以自定义的, 可以把你觉得自己觉得频繁用到, 但是觉得繁琐的命令, 在这里定义, 位置在Preference->Live Templates
- 关于最开始说的下拉刷新的代码, 很多类其实大同小异, 如果每次都拷贝写一遍, 着实有点麻烦, 那有用过创建单例的那个java类吗? 是不是感觉写单例方便了很多, 其实这也是可以自定义的。我的刷新加载的fragment与presenter就是通过创建模版来实现的。

正则表达式的使用

在Android Studio中, 其实很多搜索, 替换都支持正则表达式, 如果用的好, 会让你查找与替换方便很多, 比如在之前从触手录移动代码到触手tv中, 经常会需要重命名很多id的值, 而替换也很简单, 就是开头加入csrec_, 但是手动就很麻烦, 这时候就可以正则表达式配合替换来简化, 打开替换搜索栏, 勾选Regex, Match Case, 比如我想搜索R.id/drawable/layout.XXX相关的所有, 那我在第一栏, 也就是搜索栏, 输入匹配符, \bR.(\w+).*, 在替换栏输入R.\$1.csrec, 然后我只需要点按钮就可以实现, 我想替换的内容, 注意这里替换的就按照普通的想替换的文本就

可以了，不要加任何转义，比如有次我把替换写成`bR.$1.csrec`，就报错`u+0008`,多加了个退格键，但是肉眼是看不到的，可是idea会报错。

正则表达式不需要死记硬背，理解之后，用的时候回头去看一下就可以了。

<https://deerchao.net/tutorials/regex/regex.htm>