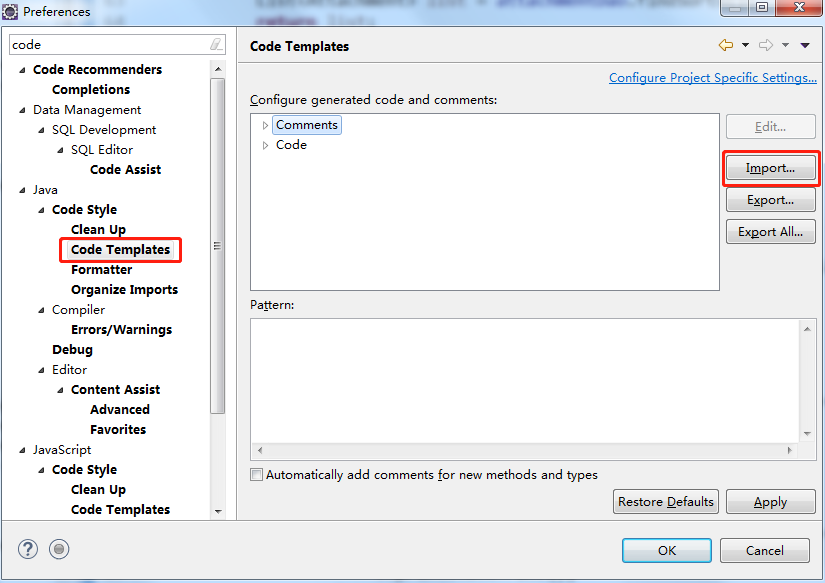
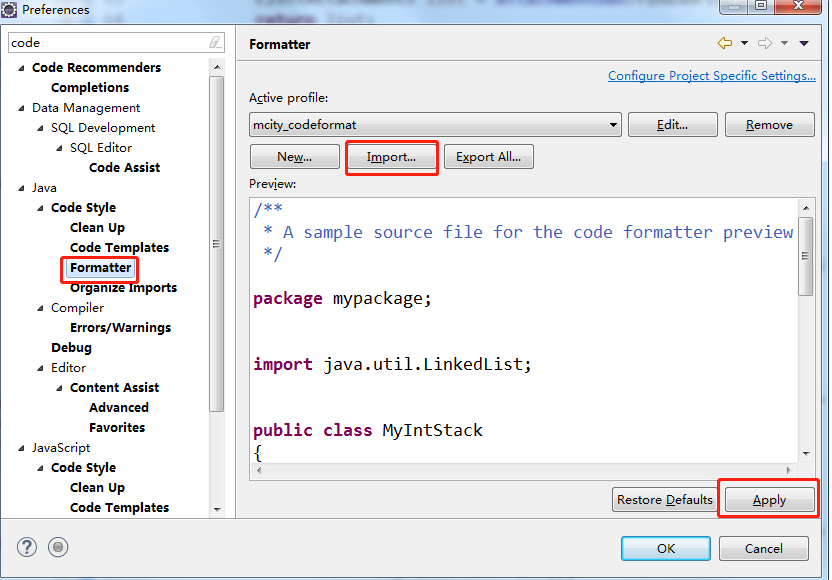
为了统一大家的编码风格，这里给提供了格式化的formatter.xml文件和注释模版codeTemplates.xml。这里简单说明一下怎么使用：

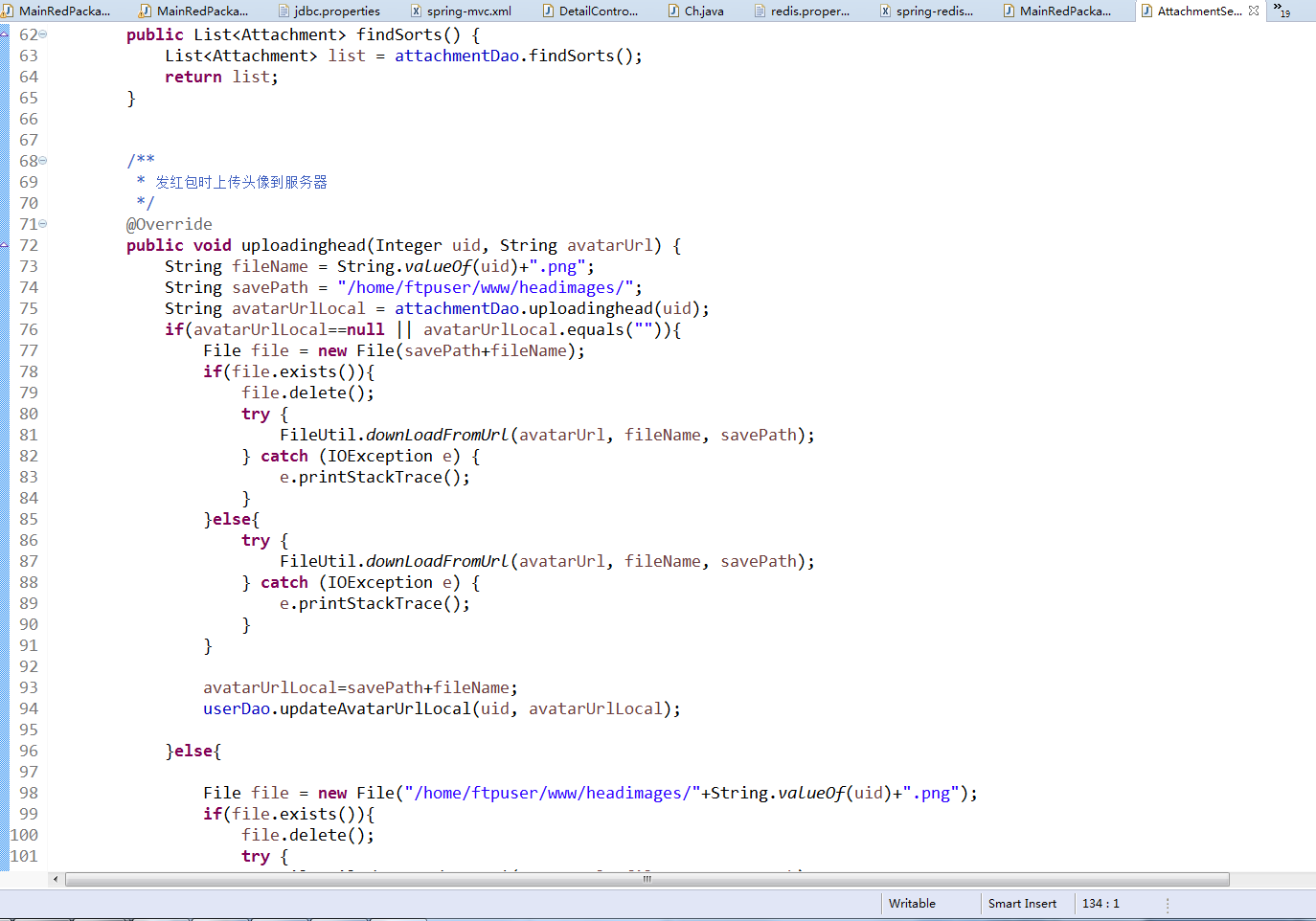
Eclipse中选择Window—Preference—Java—Code Template—Import，选择codeTemplates.xml文件导入，然后Apply即可；

同样地，Eclipse中选择Window—Preference—Java—Formatter—Import，选择我们的formatter.xml文件，然后Apply即完成。

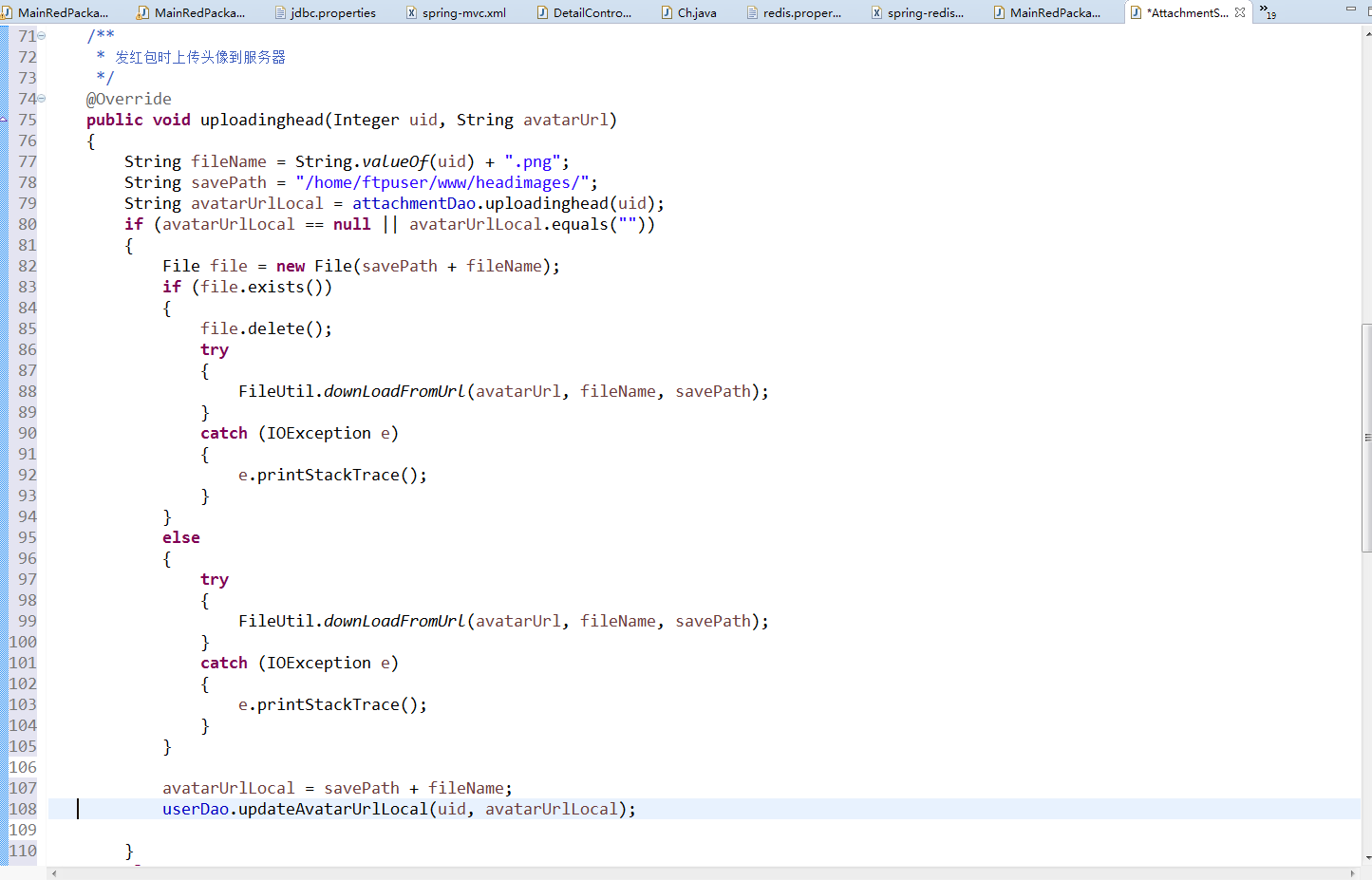




格式化之前的代码：



格式化（Ctrl+shift+f）之后的代码，是不是清晰了不少：

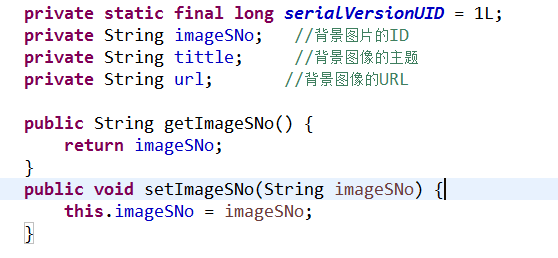


新建一个类会自动添加文件注释，类注释，方法注释等：



按模板为某个变量、方法添加注释（选择要添加注释的对象，Alt+shift+j）：、

添加注释格式化之前：

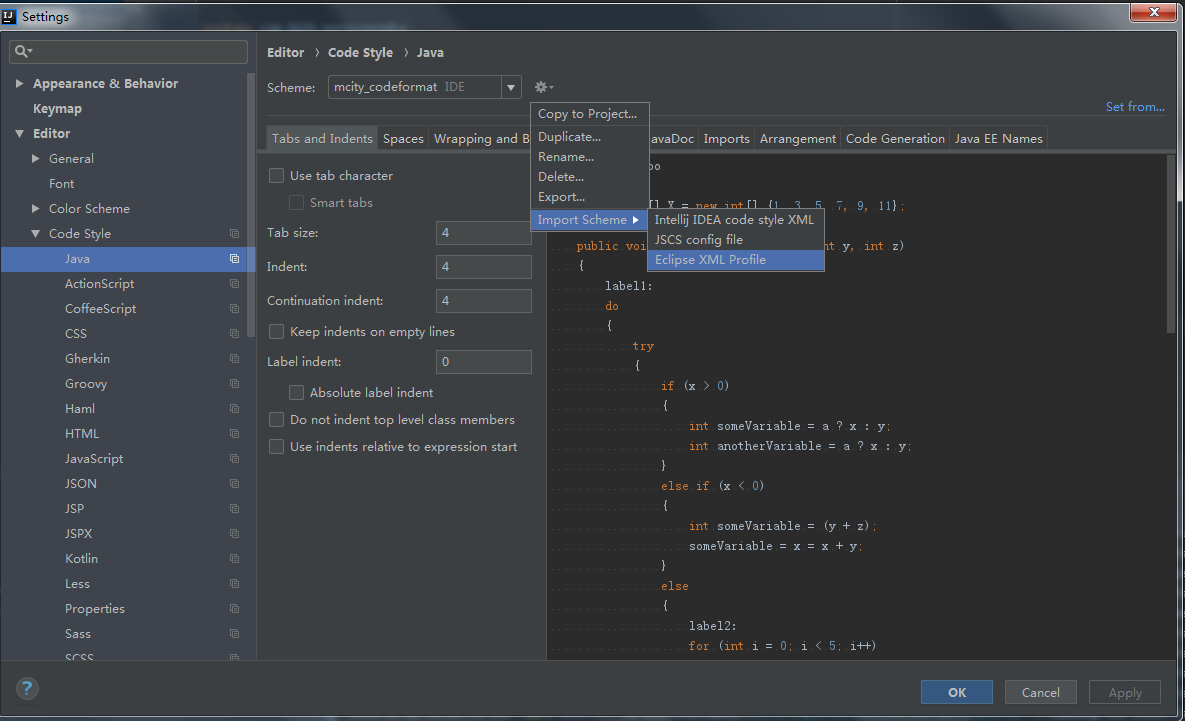


添加注释并格式化之后：



IDEA中格式化模板的使用：

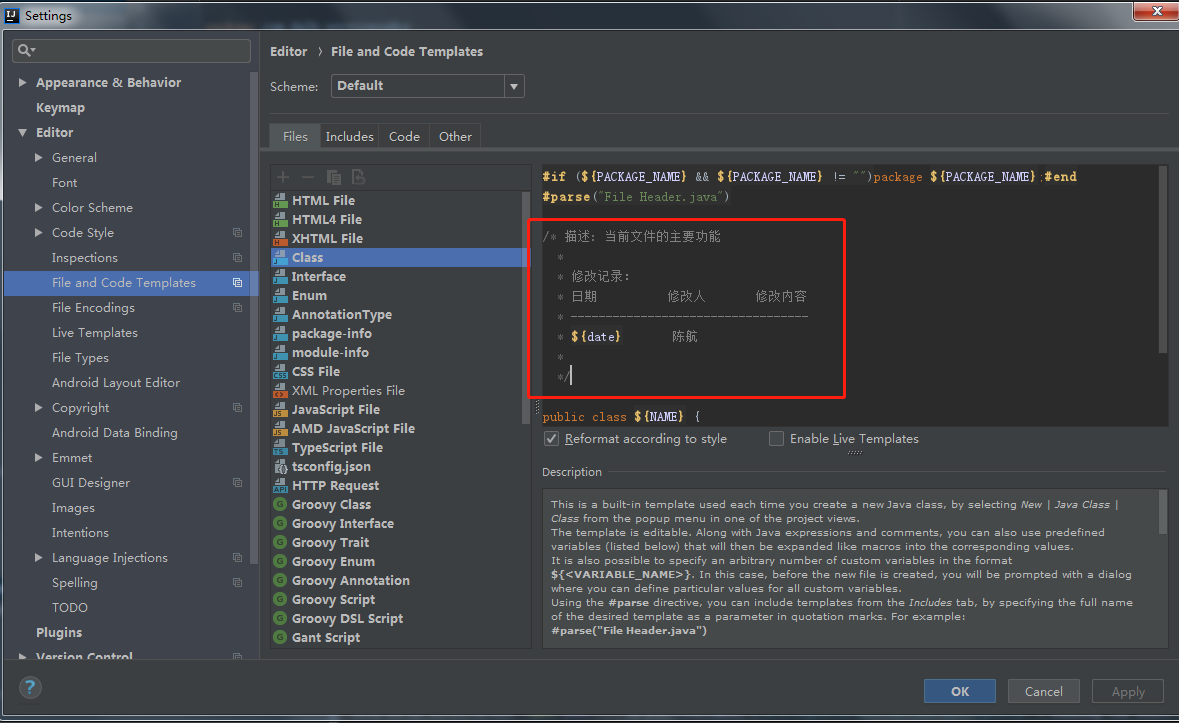
格式化模板导入：IDEA中选择File—Settings—Editor—CodeStyle—java，然后选择导入formatter.xml文件



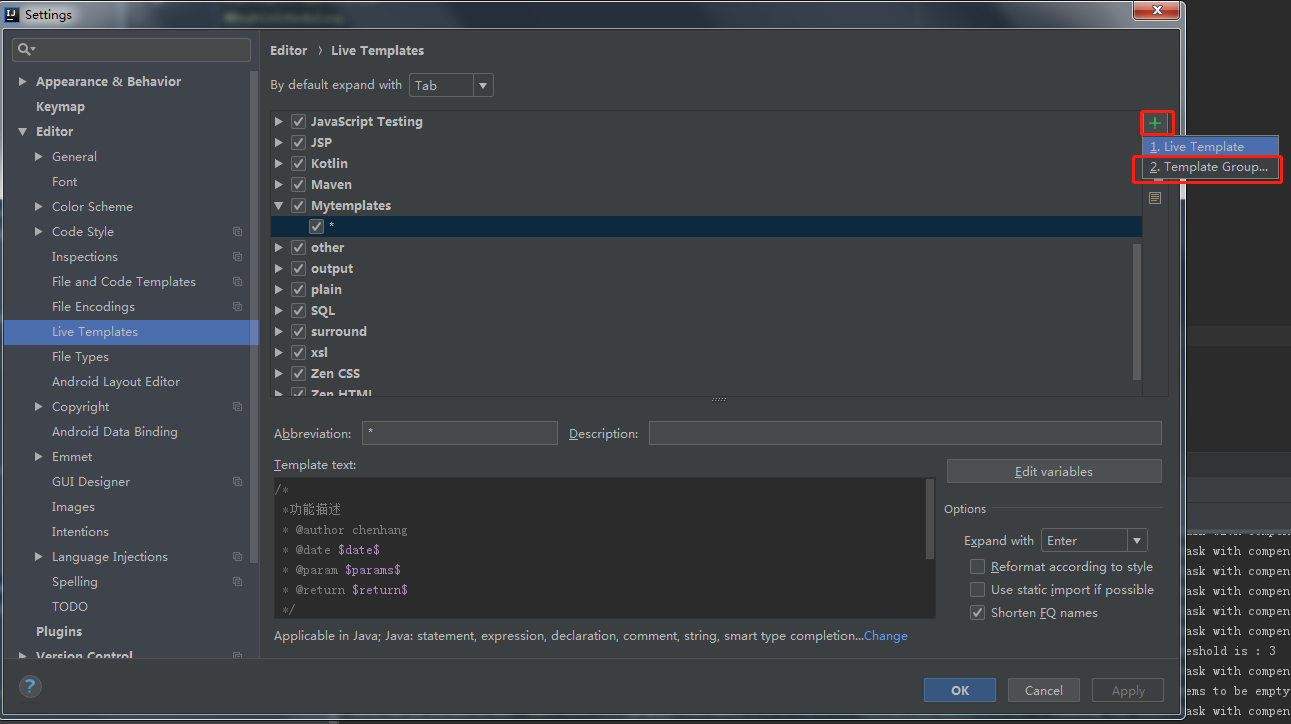
配置文件注释：IDEA中选择File—Settings—Editor—File and Code Templates，选择class文件添加一下内容：

/\* 描述: 当前文件的主要功能  
 \*  
 \* 修改记录：  
 \* 日期 修改人 修改内容  
 \* ----------------------------------  
 \* **${**date**}** 陈航   
 \*  
 \*/

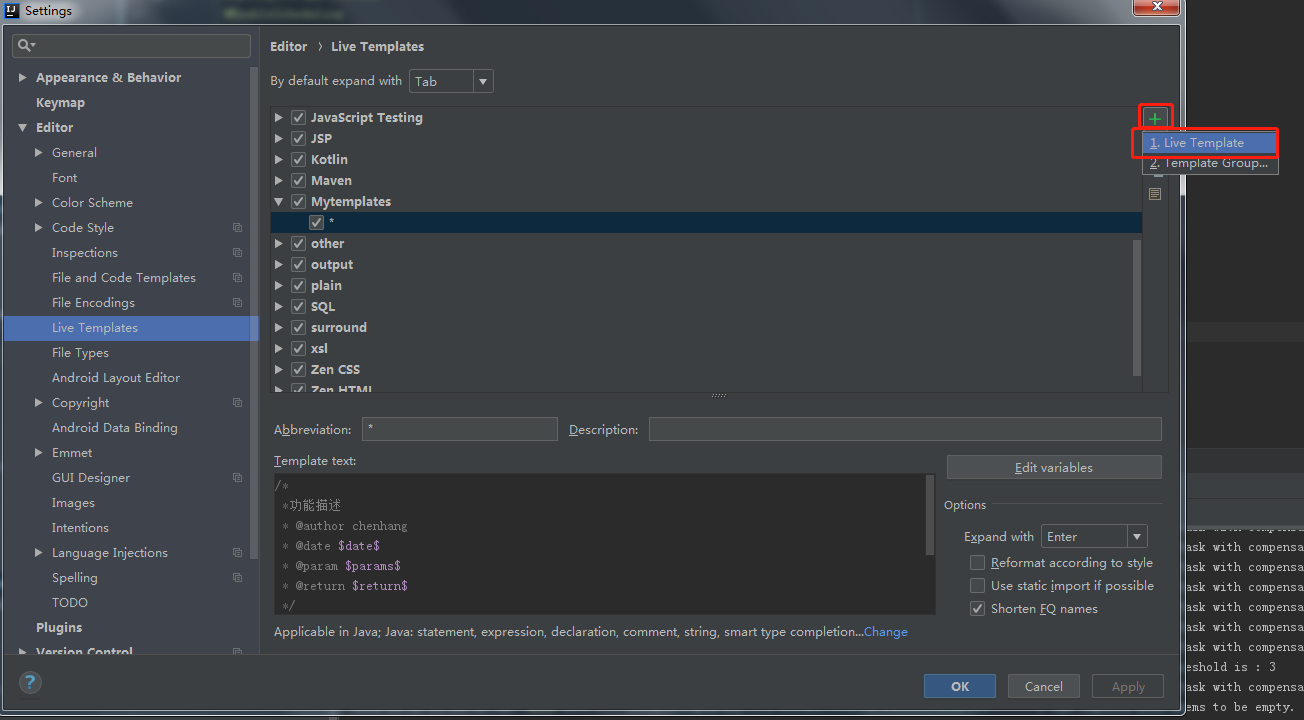
（注意修改人改成自己）



添加方法注释模板：IDEA中选择File—Settings—Editor—Live Templates如下添加MyTemplates Template Group，



然后选择MyTemplates，再从右边添加LiveTemplate，此时可以在下方完成配置工作

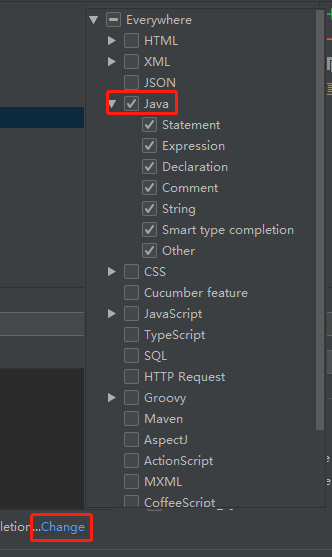


分为以下7个配置步骤：

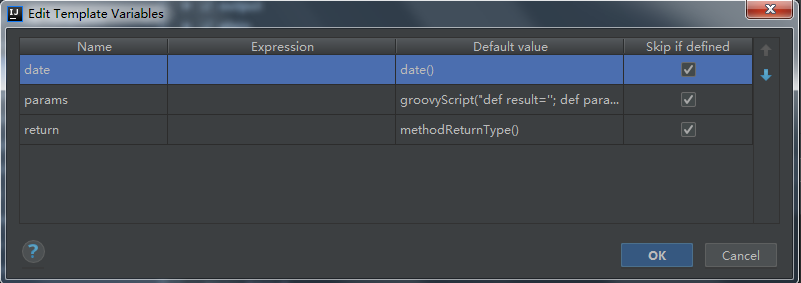
1. 填写缩写，即生成下方注释的快捷标识，这里我设置的是\*
2. 描述
3. 注释主体

/\*  
 \*功能描述   
 \* @author chenhang  
 \* @date $date$  
 \* @param $params$   
 \* @return $return$  
 \*/

1. 应用的位置，我们选择java就可以了



1. 注释里面变量编辑



多个参数分割显示：groovyScript("def result=''; def params=\"${\_1}\".replaceAll('[\\\\[|\\\\]|\\\\s]', '').split(',').toList(); for(i = 0; i < params.size(); i++) {result+=' \* @param ' + params[i] + ((i < params.size() - 1) ? '\\n':'')}; return result", methodParameters())

1. 注释确定生成键，我这里选择Enter（生成注释快捷键组合\*+Enter）
2. Apply

