# CodeGen帮助文档

# 介绍

codeGen(Code Generate)是一款基于velocity模版引擎的代码生成工具。通过读取pdm文件(PowerDesigner的physical data model)中数据库表结构，根据代码结构配置好模版，就能根据模版生成代码。本工具可以生成增，删，改，查从前端到后端所有文件，不用编写一行代码（只要你的模版写的够好），至少能给您的项目节省30％到40％的开发时间。大家要知道如果一个表结构有30个以上的字段，前端页面至少有30个以上的控件代码要写，开发人员可能要花费大量的时间编写和调试这些控件。这样的重复劳动会给开发人员产生厌恶感和疲劳感，长时间的重复劳动会使开发人员工作效率下降。开发人员应该把时间花在更有创造性，更有价值代码上。本工具不仅可以根据数据定义生成代码，也可以根据你所定义的模板文件生成任何文本文件。所以本工具的宗旨是："能用机器完成的代码，就应该由机器来完成"。

# 运行环境

运行本工具前先安装jdk1.6或以上版本（如果你本机没有安装） ，JDK1.6官方下载\_JDK6官方下载地址:<http://www.java.net/download/jdk6/6u10/promoted/b32/binaries/jdk-6u10-rc2-bin-b32-windows-i586-p-12_sep_2008.exe>

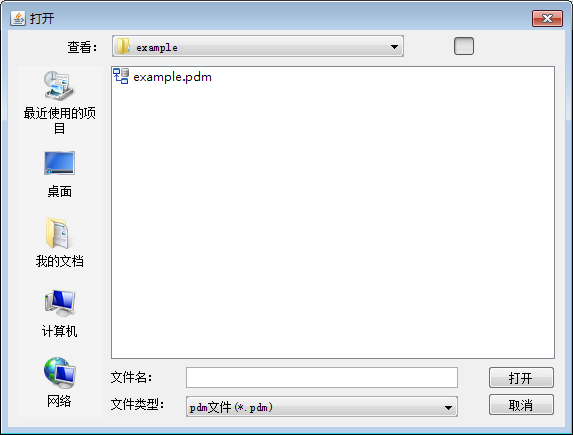
# 功能介绍

## 3.1选择pdm文件

界面左上角菜单区 点击 文件-->打开pdm 选择你要生成代码的pdm文件如图



点击 【打开pdm】如下图



选择pdm文件后在界面左边会已树型结构来展示数据库表结构如下图：



过滤文本框输入想要查找表名或则中文名中部分字符就可以过滤出包含输入字符的表

## 3.2选择表结构

主界面中左边树型表结构中可以通过过滤找到需要生成代码的表，点击选择的表节点后，主界面右侧【表选择区】会显示表结构信息如图



【表选择区】会展示pdm中会展示主表（选择的表），子表（子表中字段的外键对应表为主表）的信息，主表对子表的数据是1对多的关系，目前只支持一个子表展示。展示信息中还包括主子表的字段明细信息（根据pdm设置展示出来），展示信息表格如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表信息** | | | | |
| 项目 | 对应pdm项 | 是否可修改 | 备注 |
| 表名 | 表Code | 否 |  |
| 名称 | 表Name | 是 |  |
| 扩展1 |  | 是 | 默认为空 |
| 扩展2 |  | 是 | 默认为空 |
| **字段信息** | | | |
| 列代码 | 字段Code | 否 |  |
| 列名称 | 字段Name | 是 |  |
| 主键 | 字段是否主键 | 否 |  |
| 数据类型 | 字段Data Type | 否 |  |
| 长度 | 字段Length | 否 |  |
| 精度 | 字段Precision | 否 |  |
| 备注 | 字段Comment | 是 |  |
| 外键 | 字段是否外键 | 是 | 可以修改，点击 |
| 外键对应表 | 外键对应表 | 是 | 点击可以选择表 |
| 是否隐藏 |  | 是 | 如果是主键默认为隐藏，其他字段不隐藏 |
| 界面控件 |  | 是 |  |
| 扩展1 |  |  |  |
| 扩展2 |  |  |  |

【表选择区】tab页有如下几个功能：

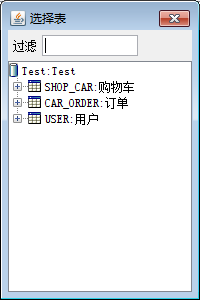
### 1.调整字段顺序

【表选择区】tab页中选择字段行，点击【上移】或【下移】按钮来向上或则向下调整字段顺序。

### 2.子表设置

【表选择区】tab页中下 子表区域可根据需要点击【增加子表】和【删除子表】按钮来增加和删除子表，目前只支持一个子表信息。

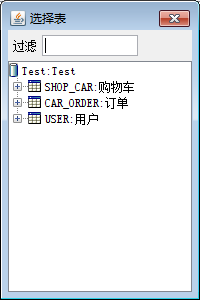
点击【增加子表】按钮会弹出表选择对话框如下图，在对话框中可以选择子表，双击子表后，列表自动显示子表信息，【增加子表】按钮会变成【删除子表】按钮；



点击【删除子表】按钮，子表列表信息会清空，【删会除子表】按钮变成【增加子表】按钮。

### 3.外键表设置

【表选择区】tab页列表中的外键对应表列可以修改，点击该单元格会弹出表选择对话框如下图。在对话框中可以选择修改字段对应的外键表，双击表节点后，该单元格会自动设置成选择表的表名。如果不选择表推出则自动清空该单元格。

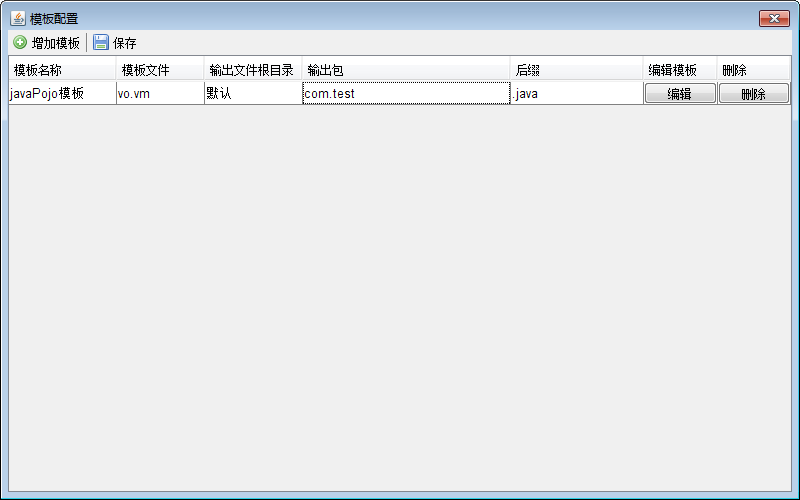


## 3.2系统配置

### **3.2.1模板配置**

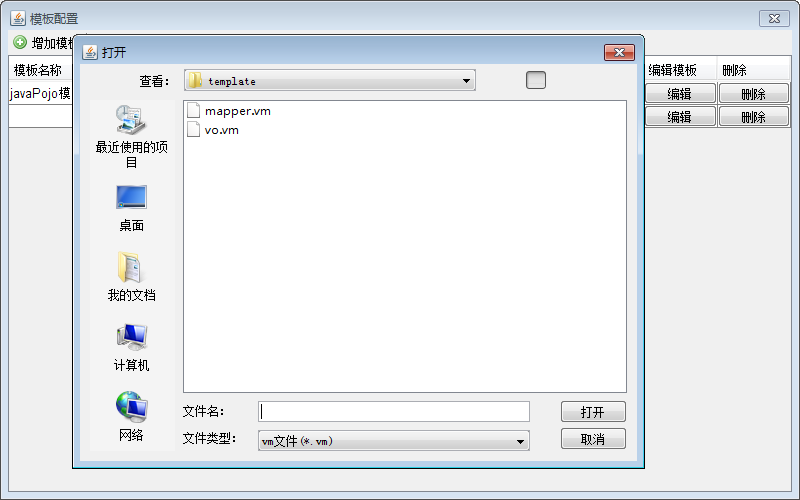
第3.3章 代码生成区要生成代码则需要选择模板，模板则在本章配置，具体配置步骤如下：

界面左上角菜单区 点击 系统配置-->模板配置 弹出模板配置界面如图



* 增加模板

点击如上图中【增加模板】按钮，模板列表会自动增加一行，分别输入模板名称点击模板文件列则弹出选择模板文件对话框如下图，选择编写好的模板（如何编写模板请参考第4章）后，模板的文件名会自动设置到模板名称列。再输入[输出文件根目录]，[文件后缀]列

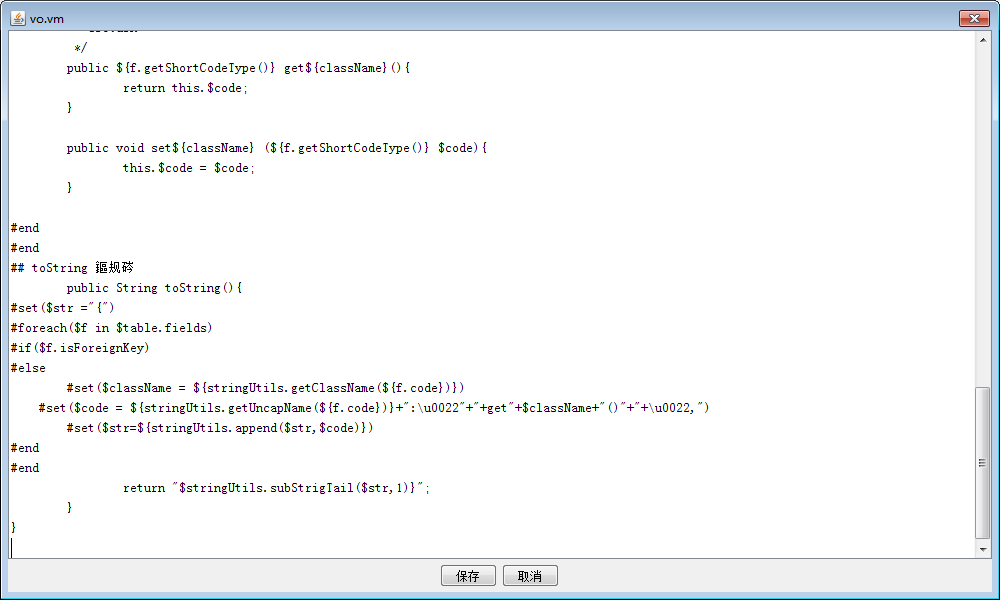


* 保存

增加模板，或则修改模板后点击【保存】按钮，可以保存该模板配

* 编辑

点击【编辑】按钮，弹出选择行的模板文件内容编辑窗口，如下图：可以编辑模板内容，点击编辑窗口下方的【保存】按钮则可以保存修改内容，点击编辑窗口下方的【取消】则操作取消。



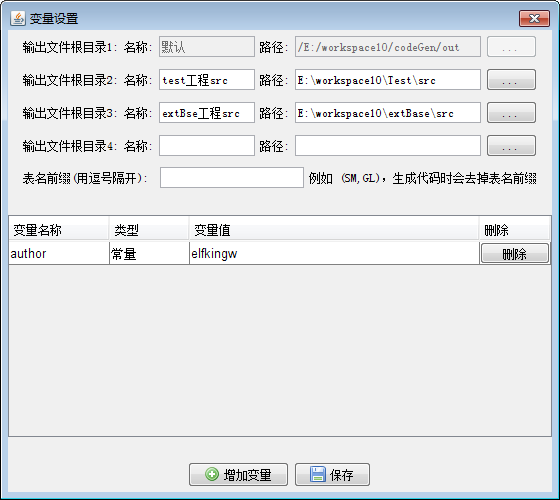
* 删除

点击【删除】按钮，则可以删除选择行的模板信息，注意删除操作后，需要点击【保存】按钮才能生效。

### 3.2.3 变量配置

变量配置主要设置【输出文件根目录】（3.2.1中会用到），【表名前缀】，【变量设置】。

界面主要设置界面左上角菜单区 点击 系统配置-->变量配置 弹出变量配置界面



* 增加变量

如上图点击【增加变量】变量列表会自动增加一行，输入[变量名称],[类型],[变量值]，点击【保存】则可保存设置变量

* 保存

点击【保存】按钮保存修改的记录

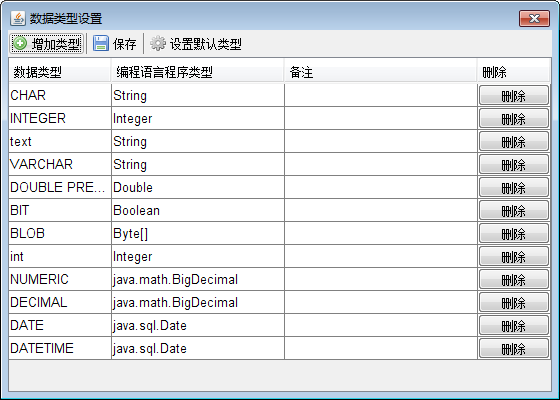
* 删除

点击【删除】按钮删除行，点击【保存】按钮删除才会生效

### 3.2.4 数据类型配置

数据类型设置主要是设置数据类型和编程语言类型对应关系设置，生成代码时根据类型对照设置将数据类型转化成编程语言程序类型。

界面左上角菜单区 点击 系统配置-->数据类型配置 弹出模板配置界面如图



* 增加类型

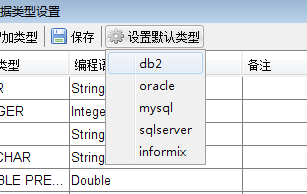
点击【增加类型】按钮自动增加一行数据类型设置，输入【数据类型】，【编程语言程序类型】，【设置默认类型】

* 保存

点击【保存】按钮保存修改的记录

* 设置默认类型

设置默认类型有一下几个数据库类型对照，如下图，选择数据库后自动设置成默认该数据库默认对照类型，点击【保存】按钮保存

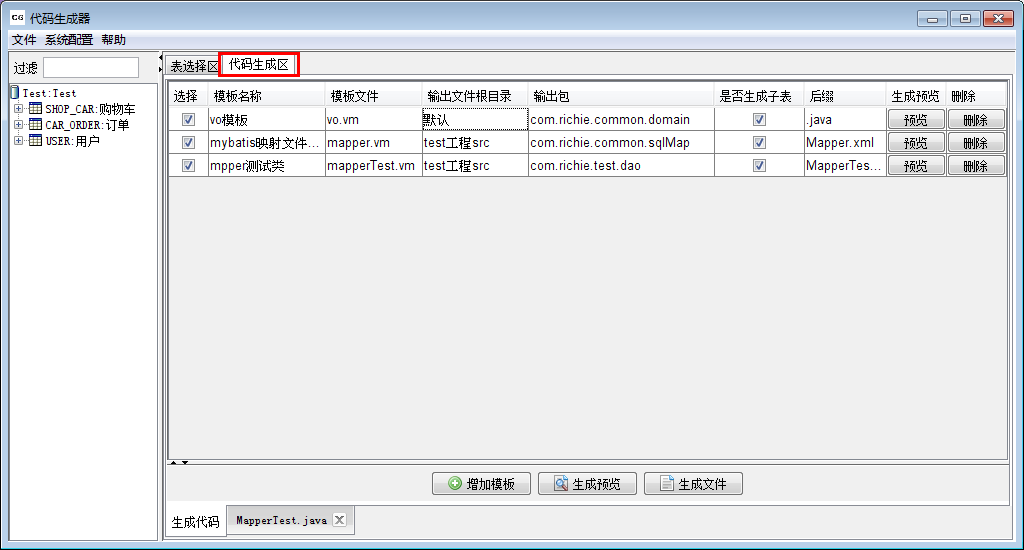


* 删除

点击【删除】按钮删除选择的记录，点击【保存】后才生效

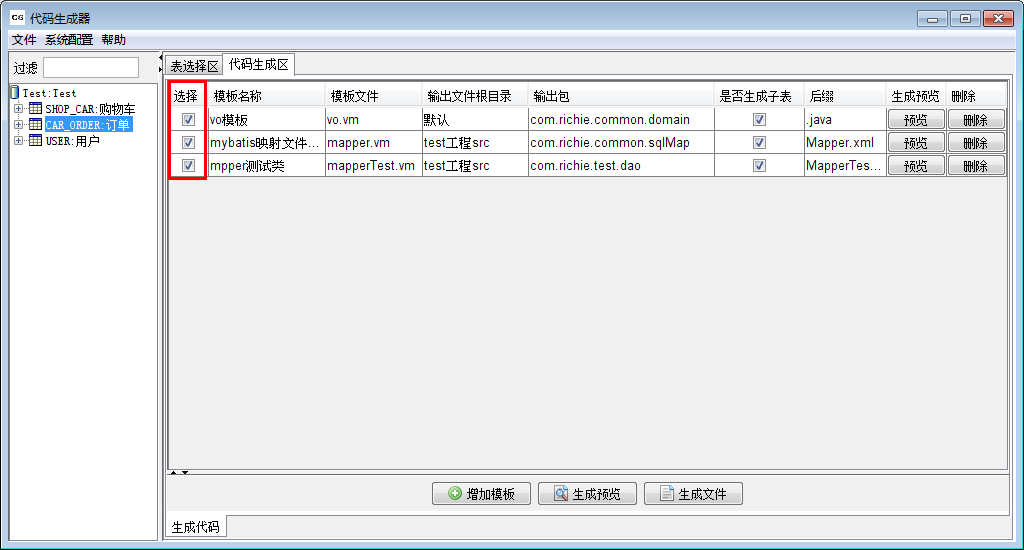
## 3.3代码生成

第3.2章选择需要生成代码的表后，进入【代码生成区】tab页如下图，增加模板后，选择生成代码的模板，生成代码。



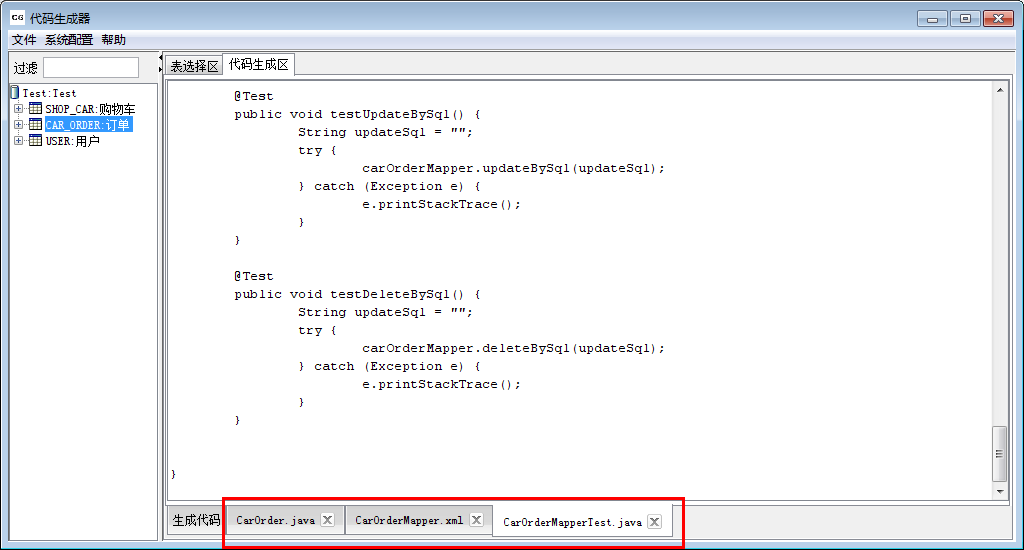
* 选择模板

点击增加【增加模板】按钮，如上界面列表会自动增加一行，【模板名称】列为下拉框（下拉框的值为第3.2.1章中设置的模板）可以选择，选择一个模板后，【模板文件】，【输出文件根目录】，【输出包】，【后缀】，自动根据模板设置里的值联动。可以增加多行模板。选择要生成模板如下图

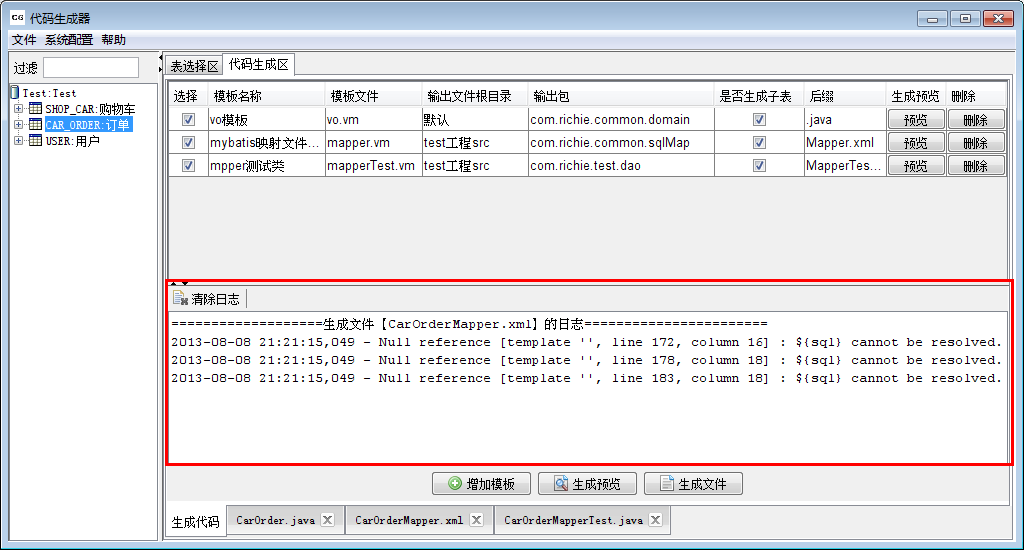


* 生成预览

增加模板后，可以选择一个或则多少模板，点击【生成预览】按钮，根据勾选的模板，生成预览tab页如下图，可以预览生成的代码



如果哪个文件生成有问题可以查看日志如下图



点击【清除日志】按钮，清空并隐藏日志面板

* 生成文件

增加模板后，可以选择一个或则多少模板，点击【生成文件】按钮，根据勾选的模板，自动生成文件，如果【是否生成子表】勾选了，子表也按模板一起生成，生成文件规则：

|  |  |  |
| --- | --- | --- |
| 文件属性 | 值 | 备注 |
| 文件名 | 选择表的表名转化成类名+后缀名 |  |
| 文件绝对路径 | 生成文件根目录+报名 |  |
| 文件内容 | 按模板生成 |  |

例如：选择表为SHOP\_CAR 模板为如下：



生成文件名为ShopCar.java 文件路径为：默认路径/com/richie/common/domain

# 操作步骤

生成文件操作步骤如下：

* 选择pdm文件，导入文件
* 选择要生成代码的表
* 设置模板，选择指定模板
* 预览或生成代码

# 4.如何编写模板

本工具是通过velocity模版引擎来生成代码，模版文件扩展名为.vm结尾。Velocity模版语法简单易用，包括基本的 for ，if else ，case等语法，velocaity模版基本语法可参照文档《Velocity模版使用指南中文版.pdf》

如果你读过此文档，相信你对模版中如何定义变量和调用方法会有些了解，下章会仔细介绍如何该工具如何定义变量和调用方法。

# 5 .模版定义变量和调用接口

如果你读过上一章提及到的文档《Velocity模版使用指南中文版.pdf》，本工具默认设置了调用方法接口，也支持自定义变量和调用借口的扩展。

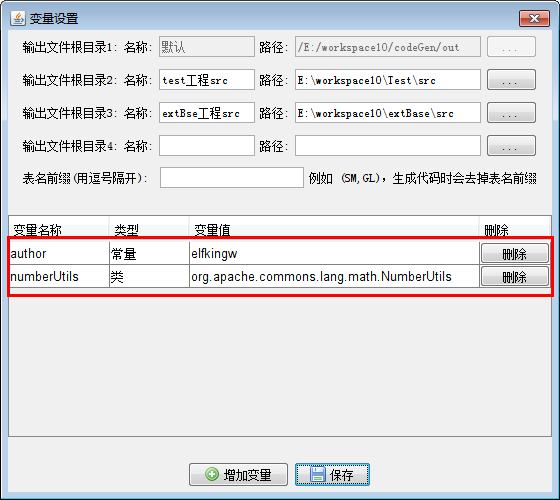
* **默认定义方法接口如下表格**

|  |  |  |  |
| --- | --- | --- | --- |
| 方法key | 对应类 | 描述 | 用法 |
| table | org.richie.codeGen.core.model.Table | 选择生成代码的表信息： | 和java调用方法一样  例如：  $table.code  $table.getCode()  都是获取表名 |
| stringUtils | org.richie.codeGen.core.util.StringUtil | 字符处理工具类 | $stringUtils.getClassName(\*\*)  获取驼峰式类名 |
| dateUtils | org.richie.codeGen.core.util.DateUtil | 日期处理工具类 | $dateUtils.getNow() 当前日期 |

备注：如上对应类的方法调用结构可以详细参考api

* **自定义方法接口**

在【3.2.3变量配置】中可以自定义增加，增加类型为【类】和【常量】，定义类型为【类】，则在模板中按方法接口一样调用，定义为【常量】则在模板中按变量一样调用，例如：



在模板中掉用 $author 生成代码则为 elfkingw，

调用 $numberUtils.max(\*\*)则返回数字数组中最大的数

备注：增加自定义的类只能在jdk和如下jar包的类才行：

