StrutsImprovement : rapid-generator

This page last changed on Jul 21, 2008 by badqiu.

通常代码生成器存在的问题.

- 二次开发困难
- 过于智能,自动插入我们的项目中,程序员还需考虑旧的代码会不会被覆盖的问题
- 没有将存放模板的目录名称及文件名称利用起来,导致还需配置每个模板文件生成的文件名,目录结构
  具体请查看我写的文章:为何代码生成器都要这么智能呢?

本代码生成器的特性

- 基于FreeMarker模板语言
- 基于数据库,并支持多种数据库(mysql,sql server,oracle测试通过)
- 半手工方式,生成的代码放在某个目录,再手工拷贝回来工作区
- 易于做二次开发,整个生成器本身就是java源代码,源代码核心十分精简,并且鼓励你做修改代码,也可以作为任何语言的代码生成器
- 配置简单,只有一个配置文件generator.properties
- 以application方式运行生成器,生成不同的table直接修改相关java代码即可
- 将文件系统的目录名称及文件名称作为生成器的一部分,模板文件的的名称与目录名称可以直接引用相关变量,如
  ${basepackage}/${className}.java (${className}=Blog,则会生成Blog.java)
- 以@testExpression结尾的模板文件为有条件忽略,如果testExpression的值在数据模型为true则生成该文件,生成的文件不会包含@te
- 支持文件插入操作,如模板输出生成的地方已经有该同名的文件存在,并且文件中有包含"webapp-generator-insert-location"标记,则模

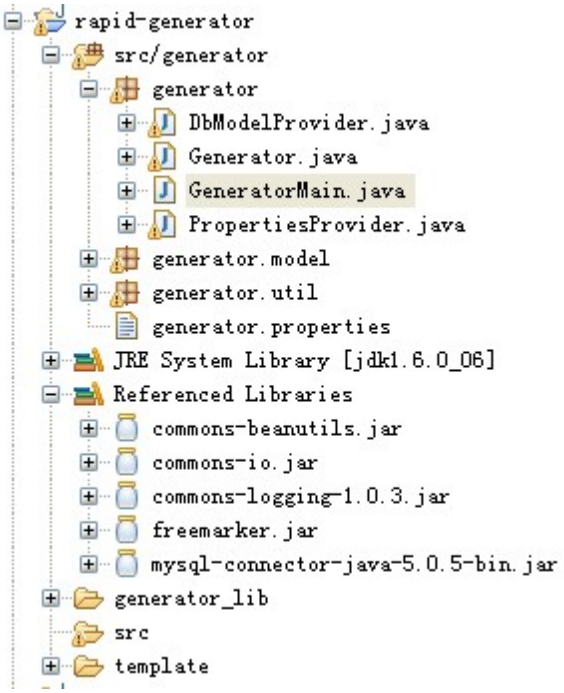
代码生成器的运行

- 将generator_lib中的jar加入classpath,generator_lib自带了几种常用jdbc驱动
- 修改generator.properties的数据库连接属性及其它属性
- 以application的方式运行GeneratorMain类,要生成不同的table,直接修改代码即可

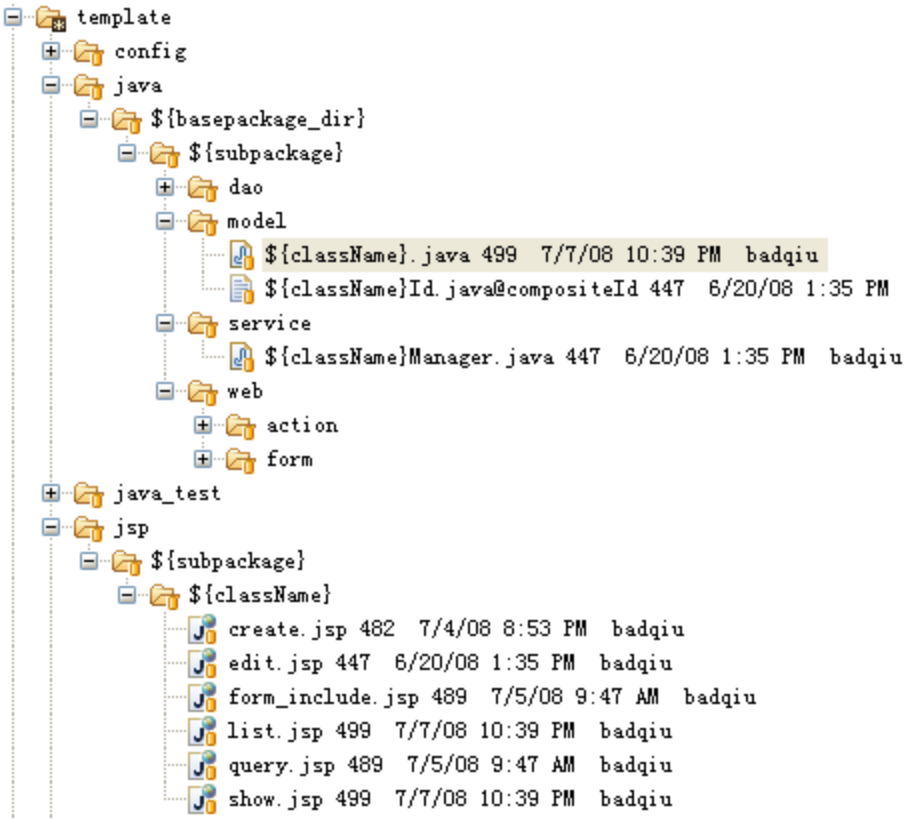
```
public class GeneratorMain {
    public static void main(String[] args) throws Exception {
        Generator g = new Generator();

        g.clean();
        g.generateTable("blog");
//      g.generateAllTable();
    }
}
```

eclipse配置环境截图



代码template目录结构



author:badqiu

project home