# 准备工作（外面有）

## 1、数据库表

## 2、逆向生成代码

# 2.讲解代码

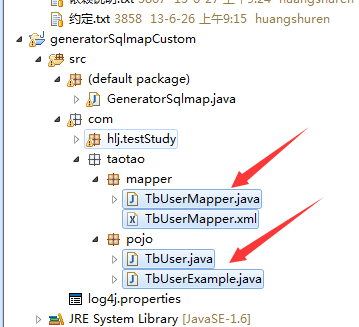
## 1、首先需要配置一个与数据库链接的配置文件

|  |
| --- |
| <generatorConfiguration>  <context id=*"testTables"* targetRuntime=*"MyBatis3"*>  <commentGenerator>  <!-- 是否去除自动生成的注释 true：是 ： false:否 -->  <property name=*"suppressAllComments"* value=*"true"* />  </commentGenerator>  <!--数据库连接的信息：驱动类、连接地址、用户名、密码 -->  <jdbcConnection driverClass=*"com.mysql.jdbc.Driver"*  connectionURL=*"jdbc:mysql://localhost:3306/taotao"* userId=*"root"*  password=*"147094"*>  </jdbcConnection>  <!-- 默认false，把JDBC DECIMAL 和 NUMERIC 类型解析为 Integer，为 true时把JDBC DECIMAL 和  NUMERIC 类型解析为java.math.BigDecimal -->  <javaTypeResolver>  <property name=*"forceBigDecimals"* value=*"false"* />  </javaTypeResolver>  <!-- targetProject:生成PO类的位置 -->  <javaModelGenerator targetPackage=*"com.taotao.pojo"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  <!-- 从数据库返回的值被清理前后的空格 -->  <property name=*"trimStrings"* value=*"true"* />  </javaModelGenerator>  <!-- targetProject:mapper映射文件生成的位置 -->  <sqlMapGenerator targetPackage=*"com.taotao.mapper"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  </sqlMapGenerator>  <!-- targetPackage：mapper接口生成的位置 -->  <javaClientGenerator type=*"XMLMAPPER"*  targetPackage=*"com.taotao.mapper"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  </javaClientGenerator>  <!-- 指定数据库表 -->  <table schema=*""* tableName=*"tb\_user"*></table>  </context>  </generatorConfiguration> |

## 2、Java类

|  |
| --- |
| **public** **class** GeneratorSqlmap {  **public** **void** generator() **throws** Exception{  List<String> warnings = **new** ArrayList<String>();  **boolean** overwrite = **true**;  //指定 逆向工程配置文件  File configFile = **new** File("generatorConfig.xml");  ConfigurationParser cp = **new** ConfigurationParser(warnings);  Configuration config = cp.parseConfiguration(configFile);  DefaultShellCallback callback = **new** DefaultShellCallback(overwrite);  MyBatisGenerator myBatisGenerator = **new** MyBatisGenerator(config,  callback, warnings);  myBatisGenerator.generate(**null**);  }  **public** **static** **void** main(String[] args) **throws** Exception {  **try** {  GeneratorSqlmap generatorSqlmap = **new** GeneratorSqlmap();  generatorSqlmap.generator();  } **catch** (Exception e) {  e.printStackTrace();  }    }  } |

## 3、点击运行即可



# 3、修改生成代码的类的名字

## 1、实体类的名字

|  |
| --- |
| <table tableName=*"TMP\_OA\_PLAN"* domainObjectName=*"TmpPlanOAVO"* schema=*""* >  <property name=*"ignoreQualifiersAtRuntime"* value=*"true"*/>  </table> |

## 2、实体类操作类变成*Criteria"*，默认是example

|  |
| --- |
| <!-- 替换条件类名后缀“Example”为“Criteria” -->  <plugin type=*"org.mybatis.generator.plugins.RenameExampleClassPlugin"*>  <property name=*"searchString"* value=*"Example$"*/>  <property name=*"replaceString"* value=*"Criteria"*/>  </plugin> |

## 3、配置文件如下

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <!DOCTYPE generatorConfiguration  PUBLIC "-//mybatis.org//DTD MyBatis Generator Configuration 1.0//EN"  "http://mybatis.org/dtd/mybatis-generator-config\_1\_0.dtd">  <generatorConfiguration>  <context id=*"testTables"* targetRuntime=*"MyBatis3"*>      <!-- 替换条件类名后缀“Example”为“Criteria” -->  <plugin type=*"org.mybatis.generator.plugins.RenameExampleClassPlugin"*>  <property name=*"searchString"* value=*"Example$"*/>  <property name=*"replaceString"* value=*"Criteria"*/>  </plugin>      <commentGenerator>  <!-- 是否去除自动生成的注释 true：是 ： false:否 -->  <property name=*"suppressAllComments"* value=*"true"* />  </commentGenerator>  <!--数据库连接的信息：驱动类、连接地址、用户名、密码 -->  <jdbcConnection driverClass=*"oracle.jdbc.driver.OracleDriver"*  connectionURL=*"jdbc:oracle:thin:@10.152.60.13:1521:zynjstkf"* userId=*"dicp"*  password=*"password"*>  </jdbcConnection>  <!-- 默认false，把JDBC DECIMAL 和 NUMERIC 类型解析为 Integer，为 true时把JDBC DECIMAL 和  NUMERIC 类型解析为java.math.BigDecimal -->  <javaTypeResolver>  <property name=*"forceBigDecimals"* value=*"false"* />  </javaTypeResolver>  <!-- targetProject:生成PO类的位置 -->  <javaModelGenerator targetPackage=*"com.hlj.sino.entry"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  <!-- 从数据库返回的值被清理前后的空格 -->  <property name=*"trimStrings"* value=*"true"* />  </javaModelGenerator>  <!-- targetProject:mapper映射文件生成的位置 -->  <sqlMapGenerator targetPackage=*"com.hlj.sino.mapper"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  </sqlMapGenerator>  <!-- targetPackage：mapper接口生成的位置 -->  <javaClientGenerator type=*"XMLMAPPER"*  targetPackage=*"com.hlj.sino.mapper"*  targetProject=*".\src"*>  <!-- enableSubPackages:是否让schema作为包的后缀 -->  <property name=*"enableSubPackages"* value=*"false"* />  </javaClientGenerator>  <!-- 指定数据库表 -->  <table tableName=*"TMP\_OA\_PLAN"* domainObjectName=*"TmpPlanOAVO"* schema=*""* >  <property name=*"ignoreQualifiersAtRuntime"* value=*"true"*/>  </table>  </context>  </generatorConfiguration> |

## 4、生成图片如下

