# 生成excel文件

com.qtxu.db.excuter.ExcelCollectionParser用于生成excel文件

|  |
| --- |
| ExcelCollectionParser ecp = **new** ExcelCollectionParser();  ecp.parseXML();  ecp.excute(); |

使用时注意：

1. 数据源的配置：src/datasource.xml。
2. Excel配置文件：exceldocument.xml

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <excel-collection>  <excel fileName="D:/123.xls">  <sheet name="客户信息" sql="select \* from myclass" dataSourceName="mysql" />  <sheet name="客户信息(中)" sql="select id 编号, name 姓名, sex 性别, degree 分数 from myclass" dataSourceName="mysql" />  </excel>  <excel fileName="D:/测试.xlsx">  <sheet name="客户信息" sql="select id 编号, name 姓名, sex 性别, degree 分数 from myclass" dataSourceName="mysql" />  </excel>  <excel fileName=*"D:/测试.xlsx"*>  <sheet name=*"客户信息"* sql=*"select SortNo,ItemName,ItemNo from CODE\_LIBRARY where CodeNo = 'VouchType' and IsInUse = '1' ORDER BY ITEMNO"* dataSourceName=*"jsals6kf"* />  </excel>  </excel-collection> |

1每个<excel>标签为一个excel文件，属性fileName为文件名

2每个<sheet>标签为excel中的页签，属性name为页签的名称，sql为页签数据的来源查询语句，sql中的列名为页签中标题，属性dataSourceName为数据源配置的数据源名。

# 生成insert语句文件

com.qtxu.db.excuter.SelectToInsert用于生成insert语句文件

## 根据sql语句生成insert语句文件

|  |
| --- |
| SelectToInsert sti = **new** SelectToInsert(sql, datasourcename);  //设置insert的输出文件，如果不设置默认当前路径下insert.sql  sti.setInsertFile(insertFile);  sti.excute(); |

## 根据文件生成insert语句文件

|  |
| --- |
| SelectToInsert sti = **new** SelectToInsert(sqlFile, datasourcename);  //设置insert的输出文件，如果不设置默认当前路径下insert.sql  sti.setInsertFile(insertFile);  sti.excute(); |

其中，sqlFile中的内容为每行一条完整的select查询语句。