1. **项目简介**

本项目名称为简单的SQL查询分析器，使用Java语言来实现，使用javafx编写界面。本查询分析器支持用户自行配置Spark SQL连接信息、支持SQL语句查询等功能。

1. **总体设计**

**1. 查询分析器的功能如下：**

1. 提供连接界面，用户可以通过输入端口号、用户名、密码信息来连接指定数据库；
2. 支持SQL语句的基本功能（建表、删表、查询等）；
3. 展示当前连接库下的表、字段，方便编写SQL脚本；
4. 列表化展示查询结果。

**2. 系统软硬件平台**

1. **系统开发平台**

系统：WIN10系统

软件：idealC – 2020.1.1,

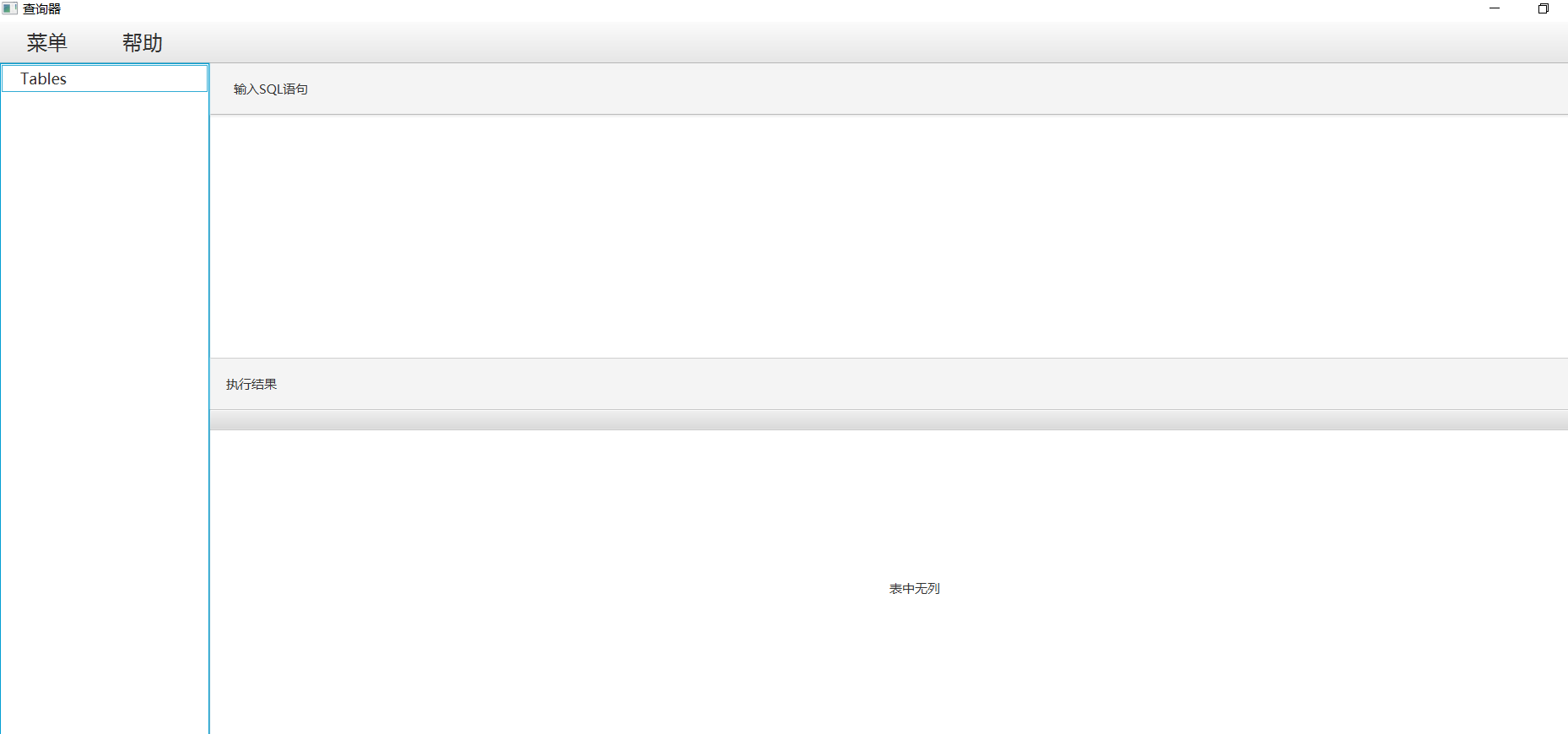
1. **系统运行平台**

一台联网的电脑

1. **详细设计说明**
2. **系统结构：**

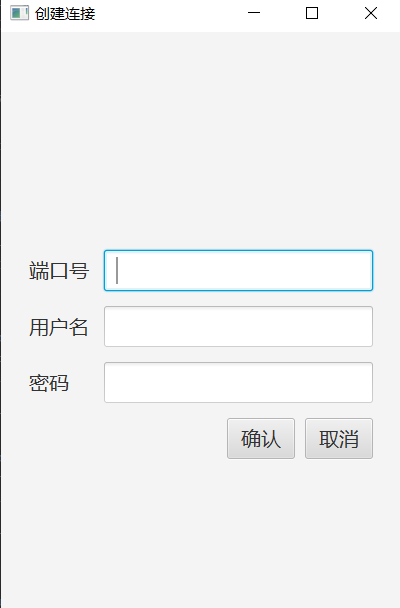
本文件同步器由三个功能代码（ChartData.java, Function.java,UI.java）文件以及一系列依赖包构成。ChartData.java文件负责接收查询的结果，并将其与展示结果的tableview绑定；Function.java文件负责查询器的连接、SQL语句执行、展示链接下的表和字段等功能；UI.java中编写了该查询器的各个页面的UI。

1. **界面设计：**
2. **主页面**



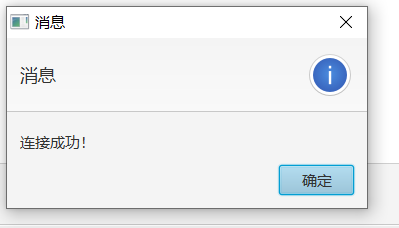
主页面布局参考了Navicat的布局方式。最上方是菜单栏（菜单中有连接、执行、刷新、退出四个功能选项），左侧能展示当前连接下数据库中的表和表内的字段；右上角是SQL语句输入框，右下角是查询结果输出框。

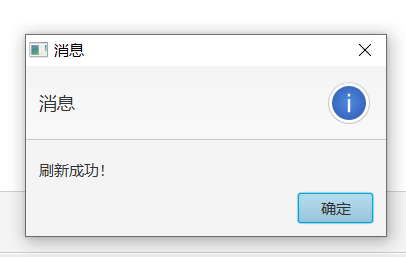
1. **连接界面**

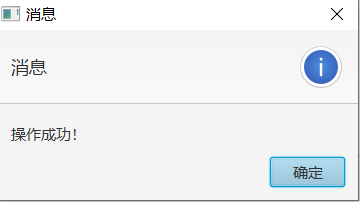


1. **信息提示界面**

提示界面是利用调用java函数来实现，页面展示如下：

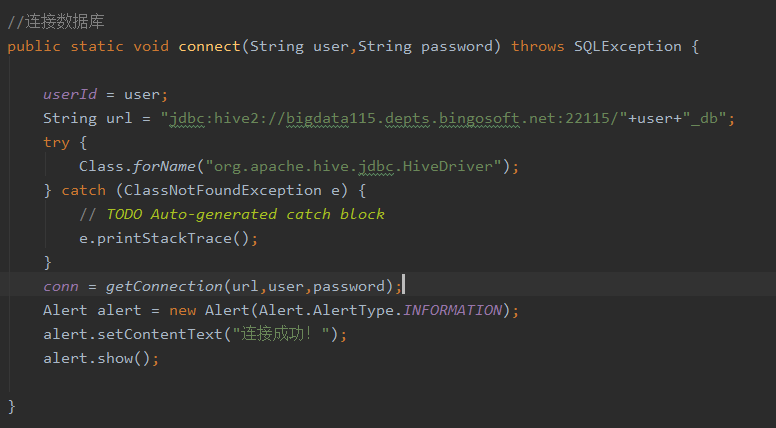






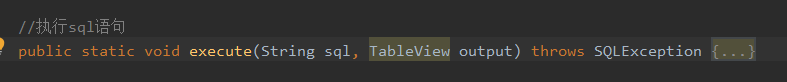
1. **详细功能设计：**
2. **查询器的连接功能：**

首先编写连接页面的UI，用户在其上面输入信息后，传至Function.java中的connect函数中，由该函数进行数据库的连接：

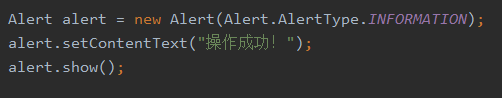


1. **SQL语句执行功能：**

当用户在SQL语句输入框输入语句时，将该语句存为String变量sql，再传入Function.java的execute（）函数中：



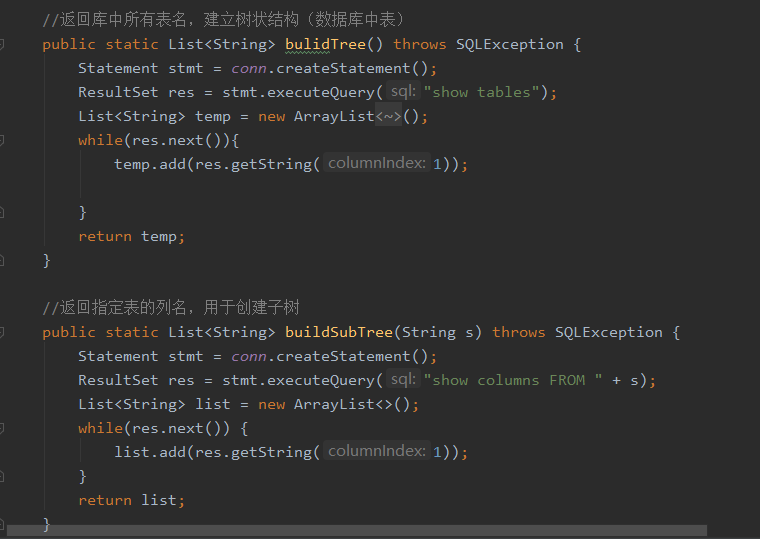
在execute函数中，我们利用split()函数，对sql字符串进行分割，获取当前语句执行对象的表名（用于获取字段），再利用ChartData.java，将查询结果和用于输出的tableview绑定。另外，在执行成功后，我们还有提示信息框，提示用户操作成功。



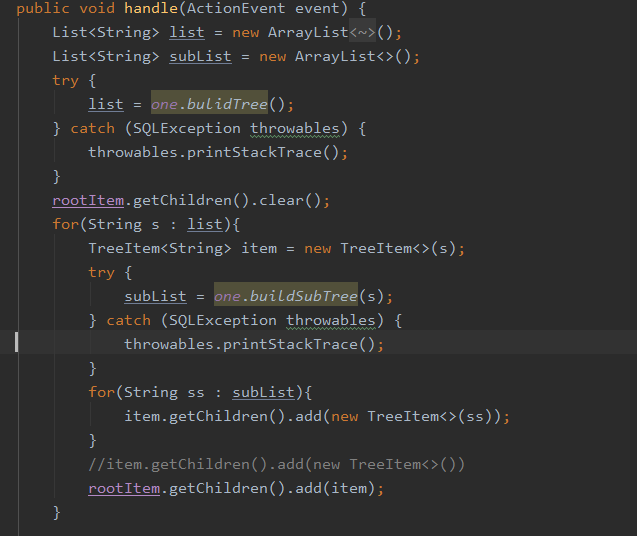
附：对SQL语句进行分类，如果是查询语句，则会展示列表化输出；如果是其他语句，再输出框无任何显示，建表、删表等改变表的操作，需要再点击“刷新”按钮，才会更新当前表和字段的结构。

1. **列表化展示链接下的表、字段：**

使用TreeView作为展示的前端组件。利用Function.java中的bulidTree()和buildSubTree()来分别查询表、字段，并返回动态数组：



在UI.java中对返回的数组与TreeView进行绑定：



1. **总结与反思**

在开发前，确定好使用什么框架、语言至关重要，因为它关乎你的程序开发能否成功。在上网查询了不少资料以及结合自身能力考虑，放弃了一开始采用flask框架的打算，决定使用java来编写界面和逻辑。

在开发过程中，阅读软件需求，并转化为todolist是一个不可省略的过程。在一开始设计SQL语句执行功能时，没考虑分情况考虑，导致写出来的代码一直出错；然后将查询语句的情况分离出来，便解决了该问题。

此外，和同学交流、上网查询资料有助于解决程序中出现的问题。