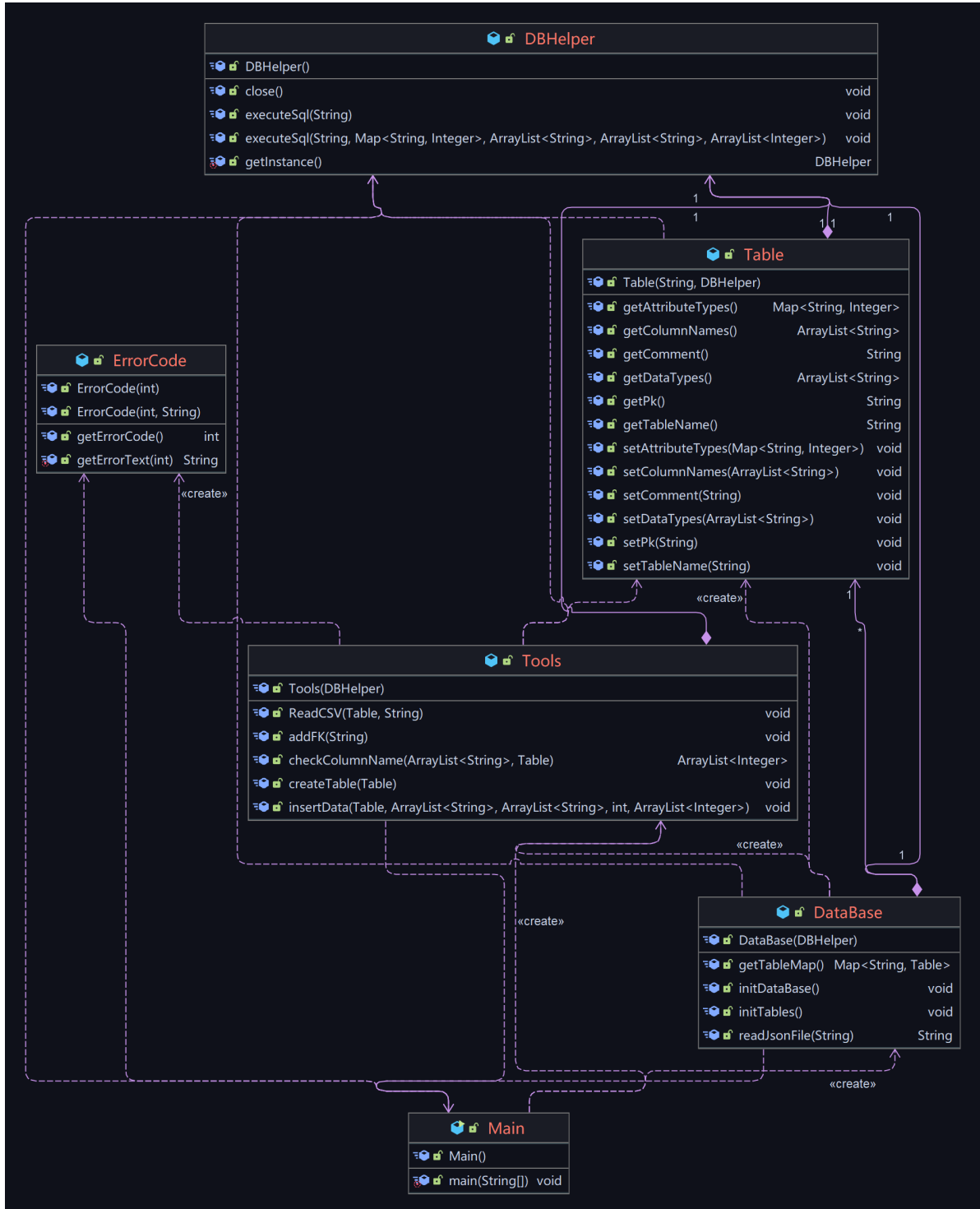


## 项目代码结构



## 1. Table:

数据库表对象，包括属性、属性类型、表名等基本信息

## 2. Tools:

- `addFK(String fkSql)`  
给数据库表添加外键
- `insertData(Table table, ArrayList<String> csvColumnNames, ArrayList<String> csvRowData, int lines, ArrayList<Integer> validCsvColumnNameIndex)`  
向指定的表中插入数据
- `readCSV(Table table, String csvName)`  
根据CSV路径读取数据并按行将数据插入对应的数据库表
- `createTable(Table table)`  
创建数据库表
- `checkColumnName(ArrayList<String> csvColumnName, Table table)`  
检查csv文件表头是否符合导入要求

## 3. DBHelper:

- `executeSql(String sql, Map<String, Integer> attributeTypes, ArrayList<String> csvColumnNames, ArrayList<String> csvRowData, ArrayList<Integer> validCsvColumnNameIndex)`  
执行带参数的sql语句
- `executeSql(String sql)`  
执行不带参数的sql语句
- `DBHelper()`  
建立数据库连接
- `close()`  
关闭数据库连接

## 4. DataBase:

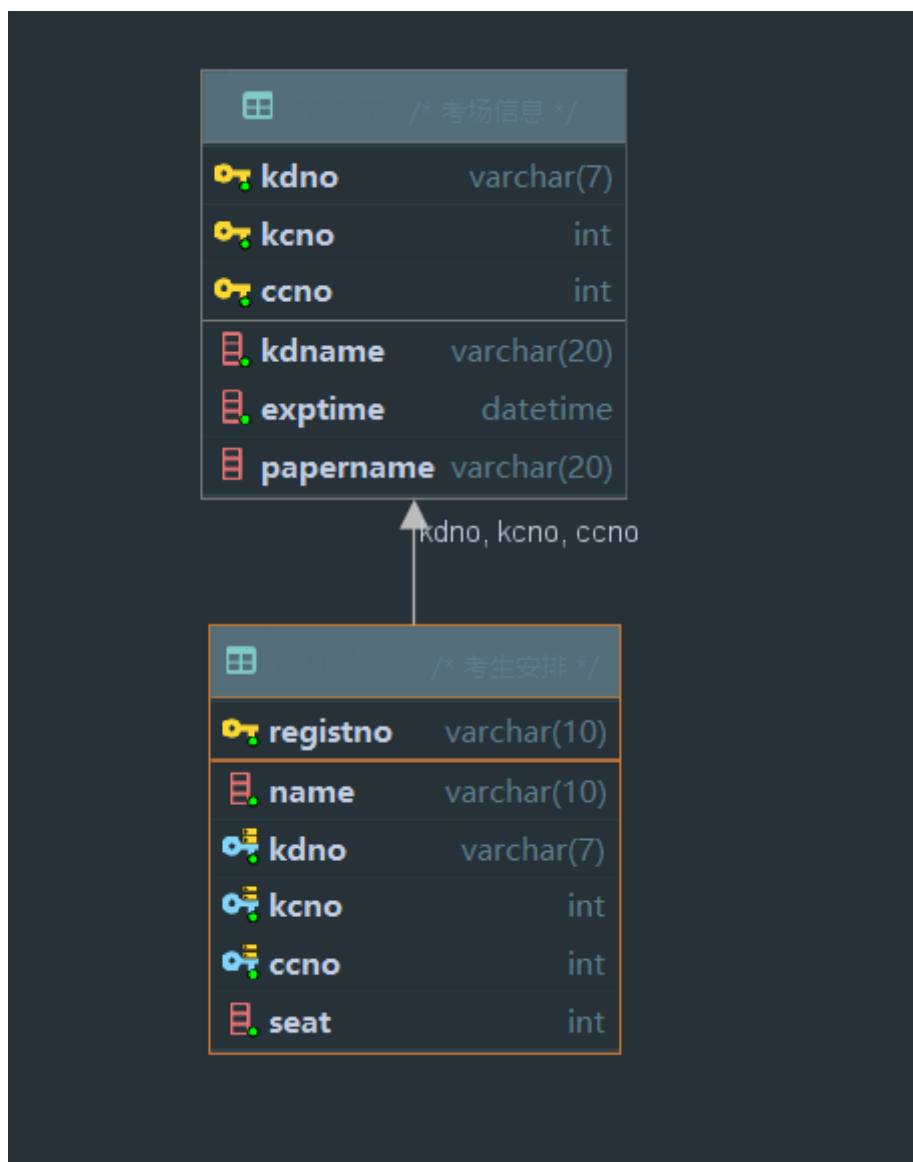
- `initDataBase()`  
初始化数据库，连接数据库
- `initTables()`  
读取表结构
- `readJsonFile(String fileName)`  
读取创建数据库表的json文件

## 5. ErrorCode:

异常处理

IO_EXCEPTION = 1	IO异常
CSV_LOSS = 2	找不到指定导入的csv文件
DIFFERENT_COLUMN_SCALE = 3	csv文件表结构与数据库表属性规模不一致
COLUMN_ERROR = 4	数据库某属性不在csv中
DATA_ROW_ERROR = 5	某一行数据无法插入，主键或外键限制
DATA_NOT_MATCH = 6	csv某行数据与表头规模不一致

## 数据库架构



共两个数据表

## 1. student:

考生安排表

- 主键: registno
- 外键: room表的主键 (kdno, kcno, ccno)

## 2. room:

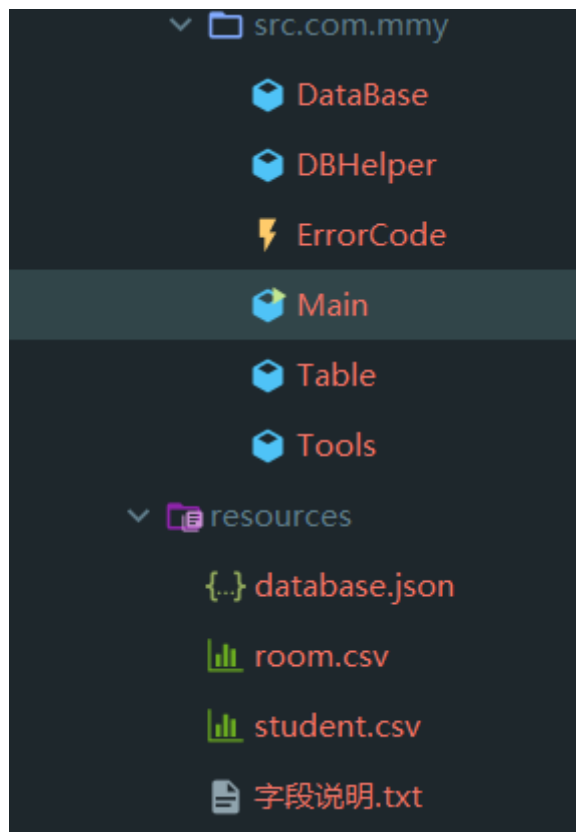
考场信息表

- 主键: (kdno, kcno, ccno)

# 文件架构、运行、功能实现

---

## 文件架构



1. 需要导入的csv文件必须要放在resources文件夹下
2. database.json文件存放了数据库的表结构

```
{
  "exam": {
    "tables": [
      {
        "tableName": "student",
        "columnName": ["registno", "name", "kdno", "kcno", "ccno", "seat"],
        "dataType": ["varchar(10) not null", "varchar(10) not null", "varchar(7) not null", "int not null", "int not null", "int not null"],
        "attributeType": [1, 1, 1, 2, 2, 2],
        "pk": "\tconstraint student_pk\t\tprimary key (registno)\n",
        "comment": "考生安排"
      },
      {
        "tableName": "room",
        "columnName": ["kdno", "kcno", "ccno", "kdname", "exptime", "papername"],
        "dataType": ["varchar(7) not null", "int not null", "int not null", "varchar(20) not null", "datetime not null", "varchar(20) null"],
        "attributeType": [1, 2, 2, 1, 3, 1],
        "pk": "\tconstraint room_pk\t\tprimary key (kdno, kcno, ccno)\n",
        "comment": "考场信息"
      }
    ],
    "fk": "alter table student\tadd constraint student_room_kdno_kcno_ccno_fk\t\tforeign key (kdno, kcno, ccno) references room (kdno, kcno, ccno);"
  }
}
```

"tableName": 表名  
 "columnName": 属性名  
 "dataType": 属性类型和约束  
 "attributeType": 属性类型 1-->String 2-->int 3-->datetime  
 "pk": 主键sql  
 "comment": 数据库表的描述

## 运行

根据命令行提示运行，项目一旦运行就会自动在数据库中创建表，数据库名为 exam，需要注意先导入 room.csv，因为 student 中含有来自 room 的外键，可以从其他 csv 文件导入数据到这两个数据库表，需要导入的 csv 文件编码为 GBK，不然中文会乱码

## 功能实现

1. 能够导入给出的两个 csv 文件
2. 当 csv 表头不完全包含数据库表的表头时无法导入
3. 当 csv 表头大小与数据库表头不同，但是又完全包含数据库表头，自己选择是否要导入
4. csv 表头属性顺序与数据库表属性不一致也能导入
5. 当 csv 某一行数据与其表头规模不一致，无法导入

## 思考

如果外部数据（原始数据表）数据不完整（例如某个不应该为空的字段缺失数据）或不一致（例如本应有外键关系的数据并没有保持引用完整性），有哪些方法可以处理？

1. 如果导入的外部数据不完整，比如某个不应该为空的字段空了或者字段的类型不符合数据库表的对应属性的要求
  - 在执行 sql 语句之前就进行检测，避免这样的错误发生，可丢弃的数据就直接丢弃，不可丢弃的就设置默认值，然后执行 sql 语句
  - 设置异常处理，当抛出对应的异常时要么丢弃记录，要么补上默认值继续执行
  - 在创建数据库表的时候就为某些属性设置默认值，当传入的参数为空时，直接设置为默认值
2. 外部数据不一致，比如主键冲突或者外键引用不完整之类的
  - 像主键冲突，抛出异常时要么删除这条记录，要么就更改这条记录的主键，或者更新记录

- 外键引用不完整，可以根据外键属性设置为外键默认值或者删除这条记录，一般而言不应该更改被引用表的数据或者结构，只能在含有外键的表内更改数据

## 处理原始数据的原则

---

1. 在读取原始数据的时候会有可能属性名的名字与表中对应属性的名字不一致，需要事先保证能相互对上号，统一属性名
2. 如果数据中存在中文，声明数据库的编码避免乱码
3. 在原始表中可能会有重复数据，可以导入前删除数据或者根据主键约束不导入
4. 如果一张表中存在外键，则要首先保证外键所在的表已经存在在数据库中
5. 如果一张表中存在外键，还要保证要插入的tuple所对应的外键值在外键所在的表中已经存在
6. 数据库表属性与对应导入属性的值类型要保持一致