数据库第三次作业

141250106

申彬

## 完成情况

完成了sql部分的数据生成和余票查询功能.无座票和每站预留票只完成了表结构,初始化数据和解决方案,没有实现.

## 运行

1. 在本地新建数据库
2. 导入项目,打开src\main\resources\application.properties
3. 修改前三项

**spring.datasource.url**=jdbc:mysql://localhost:3306/{yourdb}  
**spring.datasource.username**={username}  
**spring.datasource.password**={password}

1. maven install并运行.

第一次运行会初始化数据,请保持网络连接,因为创建线路时会爬取一些数据.

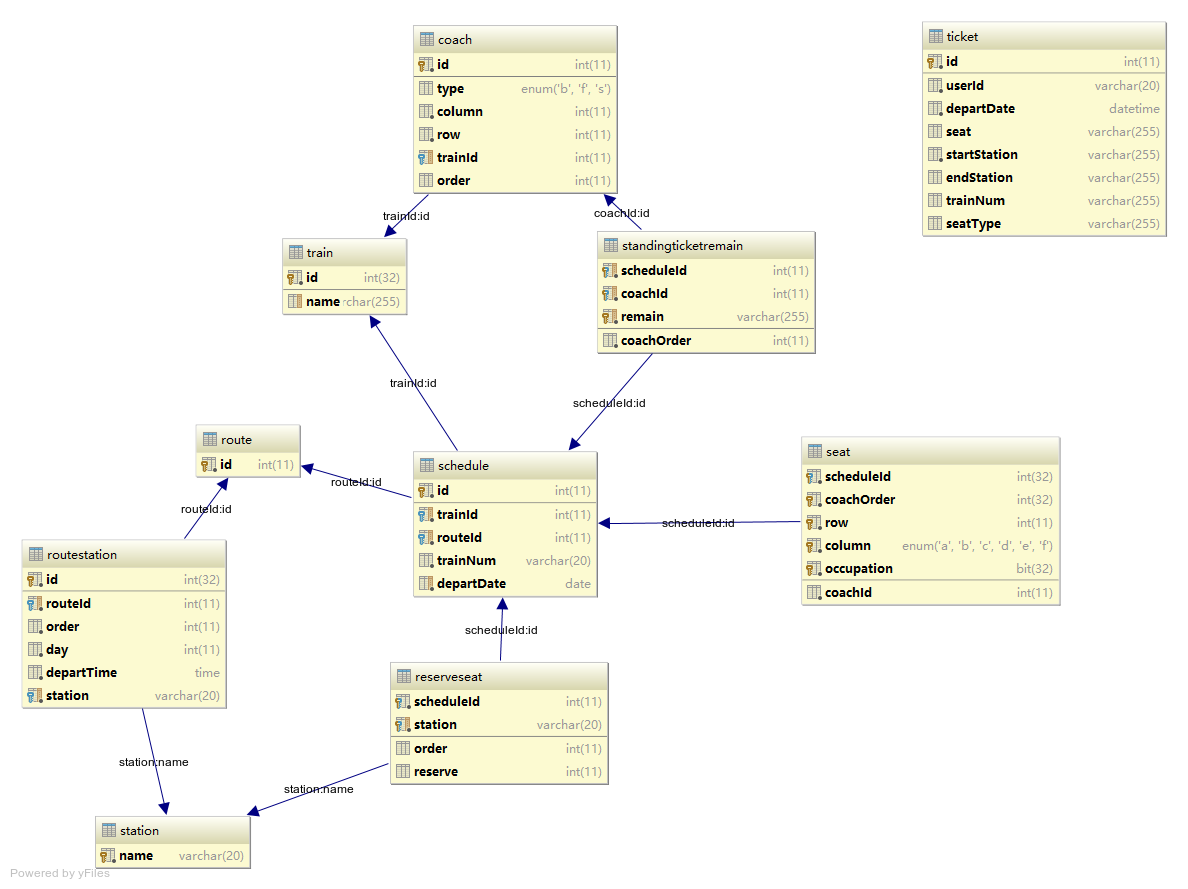
src\main\resources\schema.sql为创建表结构的脚本.初始化数据大约需要40s.

## 使用

src\main\java\edu\nju\dbhomework\DbHomeworkApplication.java为main class,main方法中可以修改这三个变量,分别代表出发日期,出发地和到达地.

//下边三个变量可以修改  
Date date = TimeUtil.*parse*("yyyy-MM-dd","2016-11-13");  
String startStation = "北京南";  
String endStation = "宜兴";

## 设计



表分为两类.

**第一类**是基础数据,包括:

route表和routeStation表,分别记录了线路的id和线路经过的车站.

Train表和cocah表,分别记录了火车信息和火车的车厢信息,包括车厢类型等.

Station表,车站名称,如南京南,北京东.

**第二类**是列车时刻信息,参照基础信息生成,包括:

Schedule表,列车时刻表,记录了7天的列车时刻信息,可以描述为,G15在11月13日行驶线路A,实际列车为B.

Seat表,是每趟车的座位占用情况,每一行记录一个座位,约有70万行.

reserveSeat表,记录每趟车每站预留多少票.

standingTicketRemain表,记录每趟车每个车厢剩余多少站票.

Ticket表,记录卖出的票.

### 数据生成

首先生成基本表数据,先生成station,再route和train.它们之间关系非常清晰.

再生成与时刻信息相关的表.首先是schedule表,然后是预留票,坐票,站票等.

### 规则

1. 坐票

Seat中,每个座位使用一个二进制串来表示占用情况,记为occupation,如该车次有6站,某位旅客已经买了从站0到站3的票,那occupation为

000111

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 1 | 1 | 1 |
| 5号站(终点) | 4号站 | 3号站 | 2号站 | 1号站 | 0号站(起点) |

因为该旅客在站3已经下车,因此3号站未被占座.

这样,假如一位旅客想从站1坐到站5,那么他期望的e\_occupation为111110.

若 occupation&e\_occupation为0,他就可以买票. &为按位与运算.

1. 预留票

为了照顾长途旅客,各个站设置预留票,优先出售非预留票,以便将预留票留给后来的买票者.

卖出一张预留票,该站预留票数减一.

某站的可购买座位 = 总座位 – 之前站的预留票数和 – 已被购买座位.某站的预留票就相当于在总座位中,从这一站开始,到终点站为止,一直占用该座位.而当一站预留票卖出,占用解除,之后站的旅客即可在前一个旅客下车后使用该座位.

1. 站票

规定二等舱有站票,数量为二等舱座位数/10. standingTicketRemain中存了一个字符串remain,形如 8,8,8,8,8,8,8.

表示的是该车厢在各个站剩余多少站票.若一个人经历的站(除到达站)中站票数最小值即为他查到的站票数.

1. 售票

坐票的出售使用最先匹配,及买给他能占用的第一个座位.此时无法保证连买的两张票是连坐票.

采用最先匹配而不是最优匹配(即占用他可占用的空余数最少的座位)的原因是效率,因为最优匹配会遍历所有座位.

不分配连坐票的原因一方面是效率,另一方面是增加座位利用率.

### 反范式

1. Seat表和standingTicketRemain表在一列内存多值,这样可以显著压缩表的行数.
2. 冗余.
3. routeStation表中,既记录了出发时间,又记录了该站是第几站.这样做是为了在计算occupation时减少查询的复杂度.
4. Seat表和standingTicketRemain是冗余的,因为它的数据可以从ticket表中推出.
5. standingTicketRemain中车厢冗余了一份coach order,即车厢号.这样做是为了方便排序.

## 补充

由于生病,比较虚弱,所以只完成sql部分的余票查询功能,并且去掉了无座票和每站预留票的功能.