

# 课程作业

课程名称 : 数据库系统原理  
作业次数 : 作业#2  
学号 : 21281280  
姓名 : 柯劲帆  
班级 : 物联网2101班  
指导老师 : 郝爽  
修改日期 : 2024年3月29日

## 1. 题目1

尝试标识该业务场景中的数据模式，即列出该业务涉及到的所有实体型（包括实体名及属性，可以适当简化）。采用如下形式列出：

实体型名1 (属性1, ..., 属性n)

尝试在你选定的数据库平台中建立实体型所对应的表，并想办法往其中插入数据，验证你所设计的模型的合理性。

该业务中涉及到的所有实体型及其属性如下：

实体型名1 (属性1, ..., 属性n)	含义
<b>Passengers</b> (ID, Name, Phone_number)	乘车乘客
<b>Users</b> (ID, Password, Phone_number, CitizenID)	购票平台用户
<b>Stations</b> (ID, Name, Province, City)	车站
<b>Trains</b> (ID, Property, StationID_start, StationID_arrive)	列车车次
<b>Seats</b> (ID, Type, TrainID)	座位
<b>Tickets</b> (ID, Price, Time_start, Time_arrive, StationID_start, StationID_arrive, SeatID)	相邻两站间的车票
<b>Transactions</b> (ID, Time, Paid, User, PassengerID, TrainID, StationID_start, StationID_arrive, SeatID)	订单
<b>Sold</b> (TransactionID, TicketID)	已售出的车票
<b>Passbys</b> (StationID, TrainID, arrive_time, leave_time)	列车过站表

## 2. 题目2

请自学了解Axure RP等快速原型设计工具的概念及功能，并尝试下载一个试用版。请你充当产品经理和产品界面模型设计工程师，模仿或创新设计两个用户界面UI，列出该用户界面对应的用户视图（用户所能操作的外模式）。请提交原始设计文档以证明你做了此项工作。



具体见原始文档。

## 3. 题目3

请通过你所设计的用户界面所对应的外模式，以及该外模式与各种模式间的映射关系，思考并叙述数据库系统从内模式、模式、外模式并最终到用户界面的映射过程，分析过程中可能会涉及到的数据加工环节。分别指出一般需要由程序员写程序实现的环节和由数据库管理系统实现的环节。

数据库系统通常遵循三级模式结构，包括内模式、模式（也称为概念模式）和外模式。这种结构旨在提供数据抽象和系统独立性。映射过程从存储在数据库中的数据（内模式）到最终用户界面的展示涉及多个步骤，每个步骤都可能需要数据加工和转换。下面是从内模式到用户界面的映射过程，以及过程中的数据加工环节和谁（程序员或数据库管理系统）负责实现这些环节的概述。

### 内模式到模式的映射

- **内模式**是数据库物理存储的低级表示，包括如何在存储介质（如磁盘）上存储数据（例如，记录格式、索引结构、压缩技术等）。
- **模式**或概念模式是数据库的整体逻辑视图，描述了数据库中所有数据的结构和关系，但不涉及数据如何存储。

**映射：**由数据库管理系统（DBMS）自动处理。这包括如何将概念模式中定义的实体、属性和关系映射到实际存储的数据上。数据加工通常不是必需的，因为这一层主要关注数据的逻辑结构而非其表示。

### 模式到外模式的映射

- **外模式**是数据库的用户视图，也称为子模式。一个数据库可以有多个外模式，每个外模式为一组用户或应用程序提供所需的数据视图。

**映射：**通常由DBMS自动处理，涉及将数据库的整体逻辑视图（模式）转换为特定用户或应用程序所需的视图（外模式）。这个过程可能涉及到数据的选择（仅展示部分数据）、投影（仅展示某些属性）和连接（整合多个表中的数据）操作。数据加工主要由DBMS执行的查询处理完成。

### 外模式到用户界面的映射

这是将用户界面所需的数据从数据库的外模式转换成最终用户可以理解和交互的形式。这个过程可能包括：

- **格式化**：将数据转换为用户界面要求的格式，如日期格式、货币格式等。
- **过滤**：根据用户请求或权限过滤数据。
- **聚合**：进行数据汇总，如计算总和、平均值等。
- **排序**：根据特定属性对数据进行排序。

**映射**：这个过程一般需要通过程序员通过编写应用程序代码来实现。虽然DBMS提供了从内模式到外模式的映射，但最终将数据呈现给用户，尤其是数据的加工和格式化，通常需要应用程序层面的逻辑来完成。

#### 数据加工环节

- **数据格式化和转换**：通常由程序员实现，以确保数据以用户友好的方式显示。
- **数据过滤和权限控制**：可能由程序员在应用层面实现，也可能通过数据库查询（由DBMS处理）实现，依据用户的权限展示数据。
- **数据聚合和排序**：通常在数据库查询中指定（由DBMS执行），但最终结果的展示和微调可能需要程序员的介入。

数据库管理系统负责从内模式到外模式的自动映射和数据处理，而将外模式数据加工成最终用户界面所需形式的任务，则通常需要程序员通过编程来完成。这种分工充分利用了DBMS的自动化处理能力，同时提供了灵活性，以适应不同用户界面的具体需求。

## 4. 题目4

请尝试学习掌握数据库系统数据规模（条或记录数）估算方法，估计所选业务场景的数据规模，针对这种规模，请说明这种规模的数据可能对内模式的实现或表示存在什么要求。

#### 数据规模估算

实体	数据规模
Passengers	假设每日新增乘客记录数为1,000条，年增长量为365,000条。五年后，总记录数大约为1,825,000条。
Users	如果80%的乘客注册成为用户，那么五年后，用户总数大约为1,460,000条记录。
Stations	考虑到火车站数量相对固定，假设有6,000个火车站。
Trains	约有4,000个不同的车次
Seats	假设每列火车平均有500个座位，共有2,000,000个座位
Tickets	每个座位约经过10个站，共有20,000,000张票，记录起售至发车共15天，则有300,000,000张票
Transactions	每天约600,000条记录，记录30天，共有18,000,000条
Sold	起售至发车共15天，平均满座率按60%计，12,000,000条记录
Passbys	6,000个车站预估共有100,000车次经过

#### 数据规模对内模式的要求

## 1. 存储空间

- **大容量存储**: 随着数据规模的增长, 需要大量的存储空间来存放记录, 尤其是对于Tickets和Transactions这样频繁增加的实体。

## 2. 性能优化

- **索引**: 为了提高查询效率, 对常用的查询字段 (如Passenger ID, Train ID, Station ID等) 建立索引是必要的。
- **分区**: 对于大量记录的表, 如Tickets和Transactions, 可以采用分区技术, 将数据分布在不同的物理区域, 以优化查询性能和数据管理。
- **压缩**: 对于历史数据, 可以考虑数据压缩技术来节省存储空间, 同时考虑对这部分数据的查询性能影响。

## 3. 数据安全和完整性

- **备份和恢复**: 随着数据规模的增长, 需要考虑高效的备份和恢复方案, 以应对数据丢失或损坏的情况。
- **事务管理**: 对于如Transactions这样的事务性数据, 需要强大的事务管理机制来确保数据的一致性和完整性。

## 4. 扩展性和可维护性

- **分布式数据库**: 对于极大规模的数据, 可能需要考虑分布式数据库解决方案, 以提供足够的存储空间和计算能力, 同时优化数据的访问速度。
- **数据清理**: 定期的数据清理和归档策略可以帮助保持数据库的性能, 移除不再需要的记录或将老旧数据迁移到更便宜的存储介质上。

随着数据规模的增长, 内模式的设计和实现需要考虑到存储管理、性能优化、数据安全和完整性以及系统的扩展性和可维护性等多方面的要求。通过合理的数据库设计和管理策略, 可以有效应对大规模数据带来的挑战。