

数据库概论课程实习

2017 年 11 月 22 日

实习目的

本次课程实习的目的在于让同学们通过具体应用来理解和掌握数据库设计的整个过程，加深对课程中所学基本概念的理解，学习和使用数据库应用系统的开发工具，并可以对设计过程中可能碰到的问题提出自己的解决方案。

实习题目：共享单车管理系统

随着移动互联网技术的不断进步，各种各样的应用应运而生。这些应用便利了人们的娱乐、购物、出行、餐饮等方方面面，提供了更加方便透明的信息获取平台，减少了人们在日常生活中所需要付出的人力成本。比如以 oFo、摩拜为代表的共享单车应用迅速崛起，掀起了人们单车出行的热潮。如今，大街小巷五颜六色的共享单车随处可见，人们可以随时随地找到一辆共享单车，解决自己短程出行的需求。而共享单车随放随停的模式，也极大节约了人们的时间成本。在共享单车应用中，一个完善的共享单车管理系统尤为重要。一方面，它需要管理共享单车数据，为共享单车的调度、维护等提供支持；另一方面，它还需要管理用户数据，为用户的用车、支付等操作提供方便。

本实习需要设计的共享单车管理系统可以参考 oFo 应用。为了简化模型，可以认为每辆共享单车拥有唯一的编号，用户在每次对共享单车进行操作的时候会提供单车的编号以及自己的位置信息。每个用户在使用共享单车之前需要进行注册并完善个人信息，如账户名、性别、职业等。用户能够查询特定位置附近可以使用的共享单车。下面对共享单车管理系统的组成要素进行介绍，同学们可以根据自己的理解来进一步完善这个设计：

1. **共享单车管理系统名称：**需要为你的共享单车管理系统起一个名称，这个名称可作为数据库的名称。
2. **用户：**每名用户都具有自己的基本信息（主要指用户的账户名、密码、性别、职业等信息）以及一些账户信息（主要指余额等信息），用户会在注册时设定基本信息，在使用时频繁地更新账户信息。
3. **共享单车：**每辆共享单车具有基本信息（主要指共享单车的编号、型号、使用价格等等）以及一些附加信息（主要指密码、位置、状态等等），共享单车在投入使用时会设定基本信息，在投入后会频繁修改附加信息。
4. **订单信息：**用户每次使用共享单车会形成一份订单。订单信息包括用户租车时间、租车地点、还车时间、还车地点、共享单车编号、花费金额等等。一名用户可能在不同时间提交多份订单，每份订单只能由一名用户提交。
5. **数据库管理员：**数据库管理员拥有所有权限，可以修改数据库中的所有信息。管理员可以增加、删除、修改共享单车信息和用户信息。管理员可以根据共享单车编号查找共享单车信息，可以查找指定位置附近的共享单车信息。
6. **XML应用：**在用户使用共享单车应用时，需要在地图上实时显示用户的位置以及用户附近可以使用的共享单车，在这个过程中可以使用XML文件来进行数据的传输。要求根据数据库中的数据，动态地生成用于显示用户附近可使用的共享单车的XML

文件。文件的样例如下：

<XML>

```
<User> //单个用户信息
  <UserName> Li Hua </UserName> //用户名
  <Location> //用户的位置信息
    <District> Hai Dian</District> //行政区
    <Street> Yiheyuan Street</Street> //街道
    <Longitude>116.000000</Longitude> //经度，精确到小数点后 6 位
    <Latitude>39.000000</Latitude> //纬度，精确到小数点后 6 位
  </Location>
  <Order> //用户最近一次的订单信息
    <StartTime>2017.11.01 15:00</StartTime> //订单开始时间
    <EndTime>2017.11.01 15:05</EndTime> //订单结束时间
    <SerialNumber>100002</SerialNumber> //共享单车编号
    <Cost>1.50</Cost> //订单花费
    <Status>Finished</Status> //订单状态
  </Order>
  <BikeList> //可用的共享单车列表
    <Bike> //共享单车 1
      <SerialNumber>100001</SerialNumber> //共享单车编号
      <Type>Small Size</Type> //共享单车类型
      <Location> //共享单车的位置信息
        <District> Hai Dian</District> //行政区
        <Street> Yiheyuan Street</Street> //街道
        <Longitude>116.000010</Longitude> //经度，精确到小数
点后 6 位
        <Latitude>39.000010</Latitude> //纬度，精确到小数点后 6
位
      </Location>
    </Bike>
    <Bike> //其它共享单车
      ...
    </Bike>
  </BikeList>
</User>
</XML>
```

DTD 描述

```
<!DOCTYPE Download [
  <!ELEMENT User(UserName, Location, Order)>
  <!ELEMENT Location(District, Street, Longitude, Latitude)>
  <!ELEMENT Order(StartTime, EndTime, SerialNumber, Cost, Status)>
  <!ELEMENT StartTime (#PCDATA)>
  <!ELEMENT EndTime (#PCDATA)>
```

```
<!ELEMENT Cost (#PCDATA)>
<!ELEMENT Status(#PCDATA)>
<!ELEMENT BikeList(Bike+)>
<!ELEMENT Bike(SerialNumber, Type, Location)>
<!ELEMENT UserName (#PCDATA)>
<!ELEMENT District (#PCDATA)>
<!ELEMENT Street (#PCDATA)>
<!ELEMENT Longitude (#PCDATA)>
<!ELEMENT Latitude (#PCDATA)>
<!ELEMENT SerialNumber (#PCDATA)>
<!ELEMENT Type (#PCDATA)>
]>
```

实习要求

1.E-R模型

- 建立一个E-R图来表示这个数据库的概念设计
- 要注意给出实体的主码
- 满足必要的约束条件

2.关系模型

- 根据E-R图创建关系模式，并对关系模式进行规范化，根据数据维护和查询的实际需求，在信息冗余和查询效率之间取得权衡
- 创建必要的索引、视图和完整性约束等
- 要求关系模型必须反映E-R图的基本内容

3.创建必要的数据库

- 创建一定量的合理的数据，使得查询结果可理解并有意义。要求用户人数不少于50人，共享单车总数不少于100辆，每个用户和共享单车都有基本信息。（提示：可以用程序生成不低于上述要求的数据。）

4.实现查询、触发器及用户权限管理

对创建的数据库进行查询得出以下结果。包括：

- 找出每个行政区内，共享单车数低于区域内街道平均共享单车数的街道。
- 找出某段时间内，完成订单数量最多的用户（如有并列，请列出所有这样的用户）。
- 对于在某个街道内有骑行记录的用户，找出这些用户的总消费金额。
- 查询故障率大于所有单车平均故障率的单车类型，并分别按照其平均故障率以及订单数量排序
- 查询故障单车数量比可使用单车数量更大的所有街道。

触发器设计：

- 设计一个触发器来监测用户的骑行过程，当用户在某个位置停留较长时间时（阈值自行设定）便报警，该用户在界面上看到提示，提醒用户结束骑行并及时结算。

用户权限管理：

- 系统管理员可以增加、删除或者修改共享单车和用户的信息。

5.XML应用

- 要求输入一个用户账户名可以获得该用户周围可用的共享单车信息：首先利用SQL查询关系数据库中对应的用户信息，根据其位置查找符合要求的共享单车信息，然后将这些信息封装在一份XML文档中，该XML文档遵循上述DTD描述。

6.用户程序

- 登录界面提示输入用户名和密码。按“确定”按钮时即进行用户名和密码的验证。同时验证类型（即系统管理员和普通用户）。可以提供注册功能，通过注册可以增加普通用户。
- 支持租车界面。用户可以通过该界面租借指定编号的共享单车，要求每个用户同时只能够租借一辆单车。
- 支持结算页面。用户可以通过该界面结束租借并完成支付。
- 支持查找共享单车页面。用户可以通过该界面查找附近可以使用的共享单车列表（不要在地图上显示）。
- 支持订单信息页面。能够显示用户历史完成过的订单信息。
- 支持用户信息页面。能够显示用户的基本信息和账户信息。
- 支持搜索页面。实现4中提到的各种查询操作，合理即可。
- 支持管理员页面。管理员可以增加或者删除用户。可以增加、删除、修改共享单车信息。可以查找指定共享单车的信息。
- 其它自由发挥，或参考oFo、摩拜，界面美观友好者，可以加分。

7.其它

- 具体实现平台不限，但要求可以演示。界面简洁实用即可，注重数据库原理与方法的运用。数据库设计做到思路清晰、合理、简洁、高效。以上给出的系统基本功能必须实现，同学们可以进行功能上的扩充。
- 数据库管理工具可以使用 MySQL、Oracle 或者 Microsoft SQL Server 等。
- 系统架构可以选用下述两种结构中的一种：
 1. B/S 结构。即 Browser/Server 结构，使用 PHP（需要安装 Apache + PHP）或者 JSP（需要安装 Java + Apache + Tomcat）进行开发。
 2. C/S 结构。使用 Java 或者 Visual C++进行开发。

需要提交的材料

1. E-R图
2. 关系模式表
3. 实例数据（不必提交所有数据，给出一些典型的实例数据即可，不超过1MB）
4. SQL程序（包括查询、触发器及用户权限管理所用的SQL程序）
5. 用户程序的设计文档及代码（包括程序使用说明）
6. 小组分工情况，包括组长和各个组员完成工作的具体描述
7. 一个ReadMe.txt文档，在最顶层的文件夹

以上各项打包为一个RAR文件提交给助教邮箱: chenypeng@pku.edu.cn。

评分主要标准

系统演示没有错误，基本功能完整。数据库设计合理、简洁、高效，设计思路清晰，代码结构良好、简洁规范，系统框架合理高效。

作业要求

1. 请同学们按照自愿的方式组成课程小组，**每组 1~3 人**完成课程作业。每个小组必须在 2017 年 12 月 1 日之前将小组成员名单、组长联系方式发送至助教邮箱（chenyipeng@pku.edu.cn），邮件标题数据库课程实习分组。
2. 各小组需要在 2018 年 1 月 1 日之前提交课程实习作业。并在提交之前应当同助教约定好时间当面演示。

注意事项

1. 请各小组务必在截止日期之前按时提交课程实习作业，逾期按没有完成课程实习处理。
2. 在完成课程实习的过程中，如果存在任何的疑问或者建议，请及时与助教联系。助教的联系方式为：陈逸鹏（电话 18810521589，[邮箱 chenyipeng@pku.edu.cn](mailto:chenyipeng@pku.edu.cn)），周雨熙（电话 18800193635，[邮箱 joy_yuxi@pku.edu.com](mailto:joy_yuxi@pku.edu.com)），实验室为理科二号楼 2216S。
3. 本次课程实习的分数占课程总成绩的 20%。
4. **请注意所有发送的邮件以收到助教回复为准。**