**南京大学智能软件与工程学院**

**《数据管理基础》课程 实验指导手册**

**实验一：用SQL进行数据操作**

**截止日期：2024年05月31日**

**实验要求：**

严禁抄袭！若发现抄袭SQL代码或虚构实验报告，本次实验成绩为零分。

请认真完成所有实验内容，并于截止日期之前提交实验报告。请按照实验报告模板的要求，将实验结果组织成两个文件进行提交，所提交的文件应简洁明了。

文件1：实验结果报告（请按照‘模板’要求进行组织，以PDF格式文档提交）

文件2：SQL脚本（可批处理执行的SQL脚本文件；或者剔除文件1实验结果报告中的所有说明文字和结果贴图，只保留“实验过程”中的所有SQL命令，可供复制、粘贴、执行。请尽量使用标准的SQL语句。）

如果你参考了他人的实验成果，请在实验报告中注明并致谢。

**提交方式：**

以文件方式提交到院本科教学支撑平台 https://selearning.nju.edu.cn

**评分标准：**按时提交（40%）+ 报告内容（60%）

**实验内容：**

完成下列SQL基本操作，并将所有SQL语句和要求的实验截图放在实验报告中。

1. **使用SQL语句建立基本表(Customers, Agents, Products, Orders)**

1.1 Customers表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 非空 | 主键 | 说明 |
| cid | Char(4) | 是 | 是 | 编号 |
| cname | Char(20) | 是 |  | 姓名 |
| city | Char(20) |  |  | 城市 |
| discnt | real |  |  | 折扣 |

1.2 Agents表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 非空 | 主键 | 说明 |
| aid | Char(3) | 是 | 是 | 编号 |
| aname | Char(20) | 是 |  | 名称 |
| city | Char(20) |  |  | 城市 |
| perc | smallint |  |  | 佣金比例 |

1.3 Products表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 非空 | 主键 | 说明 |
| pid | Char(3) | 是 | 是 | 编号 |
| pname | Char(20) | 是 |  | 名称 |
| city | Char(20) |  |  | 城市 |
| quantity | int | 是 |  | 库存数量 |
| price | real | 是 |  | 单价 |

1.4 Orders表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 非空 | 主键 | 说明 |
| ordno | int | 是 | 是 | 订单编号 |
| orddate | Date | 是 |  | 订单日期 |
| cid | Char(4) | 是 |  | 顾客编号 |
| aid | Char(3) | 是 |  | 经销商编号 |
| pid | Char(3) | 是 |  | 商品编号 |
| qty | int |  |  | 数量 |
| dols | real |  |  | 金额 |

1. **使用SQL语句插入数据**
   1. 向Customers表插入下列数据：

('c001', 'Tiptop', 'Duluth', 10.00);

('c002', 'Basics', 'Dallas', 12.00);

('c003', 'Allied', 'Dallas', 8.00);

('c004', 'ACME', 'Duluth', 8.00);

('c006', 'ACME', 'Kyoto', 0.00);

* 1. 向Agents表插入下列数据：

('a01', 'Smith', 'New York', 6);

('a02', 'Jones', 'Newark', 6);

('a03', 'Brown', 'Tokyo', 7);

('a04', 'Gray', 'New York', 6);

('a05', 'Otasi', 'Duluth', 5);

('a06', 'Smith', 'Dallas', 5);

* 1. 向Products表插入下列数据：

('p01', 'comb', 'Dallas', 111400, 0.50);

('p02', 'brush', 'Newark', 203000, 0.50);

('p03', 'razor', 'Duluth', 150600, 1.00);

('p04', 'pen', 'Duluth', 125300, 1.00);

('p05', 'pencil', 'Dallas', 221400, 1.00);

('p06', 'folder', 'Dallas', 123100, 2.00);

('p07', 'case', 'Newark', 100500, 1.00);

* 1. 向Orders表插入下列数据：

(1011, '2016-01-08', 'c001', 'a01', 'p01', 1000, 450.00);

(1012, '2016-01-12', 'c001', 'a01', 'p01', 1000, 450.00);

(1019, '2016-02-24', 'c001', 'a02', 'p02', 400, 180.00);

(1017, '2016-02-10', 'c001', 'a06', 'p03', 600, 540.00);

(1018, '2016-02-16', 'c001', 'a03', 'p04', 600, 540.00);

(1023, '2016-03-12', 'c001', 'a04', 'p05', 500, 450.00);

(1022, '2016-03-08', 'c001', 'a05', 'p06', 400, 720.00);

(1025, '2016-04-07', 'c001', 'a05', 'p07', 800, 720.00);

(1013, '2016-01-13', 'c002', 'a03', 'p03', 1000, 880.00);

(1026, '2016-05-20', 'c002', 'a05', 'p03', 800, 704.00);

(1015, '2016-01-23', 'c003', 'a03', 'p05', 1200, 1104.00);

(1014, '2016-01-18', 'c003', 'a03', 'p05', 1200, 1104.00);

(1021, '2016-02-28', 'c004', 'a06', 'p01', 1000, 460.00);

(1016, '2016-01-25', 'c006', 'a01', 'p01', 1000, 500.00);

(1020, '2016-02-05', 'c006', 'a03', 'p07', 600, 600.00);

(1024, '2016-03-12', 'c006', 'a06', 'p01', 800, 400.00);

1. **请用SQL语言写出下述查询，并将查询结果截图保留在实验报告中。**
   1. 查询没有为居住在Duluth的任何顾客订购过任何商品的经销商的编号；
   2. 查询城市名称中含有字符串’New’的经销商的编号、名称和所在城市；
   3. 查询仅通过a03和a05两个经销商订购过商品的顾客编号；
   4. 查询符合下述条件的商品的编号：至少有一个顾客通过与该顾客位于同一个城市的经销商订购过该商品；
   5. 返回每一个顾客的第一份订单和最后一份订单，结果返回顾客编号、第一份订单及最后一份订单的订单编号和订购日期；（按照订单编号的大小区分订单的先后；如果一个顾客只有一份订单，那么该订单作为该顾客的第一份订单，他的最后一份订单属性返回空值；不需要考虑没有订单的顾客。）
   6. 在所有有顾客的城市中都被销售过的商品的编号；
   7. 查询居住在Dallas的所有顾客都订购过的商品编号；
   8. 查询享有最高销售提成比例的经销商；（请写出不少于三种的不同表示方法）
   9. 查询仅仅通过a04号经销商订购过商品的顾客编号，并给出每个顾客的购买总金额；
   10. 查询符合下述要求的经销商的销售统计结果（经销商编号，订单条数，订单的平均销售金额）：订单的平均销售金额达到或超过500，结果按照订单平均销售金额的降序输出查询结果；
2. **删除基本表**
   1. 删除Orders表
   2. 删除Customers表
   3. 删除Agents表
   4. 删除Products表