

# <数据库概论>实验: SQL操作实习 实验报告

姓名:周宇翔 学号:181860152 联系方式:18851172365

## <数据库概论>实验: SQL操作实习 实验报告

姓名:周宇翔 学号:181860152 联系方式:18851172365

实验环境

实验内容

- 一.使用SQL语句建立基本表
- 二.使用SQL语句插入数据
- 三.SQL语言的查询
- 四.视图的创建与访问
- 五.数据完整性约束及事务处理
- 六.删除基本表

实验中遇到的困难及解决办法

参考文献及致谢

## 实验环境

windows10 × MySQL 8.0.22 × MySQL workbench图形界面

## 实验内容

### 一.使用SQL语句建立基本表

代码:

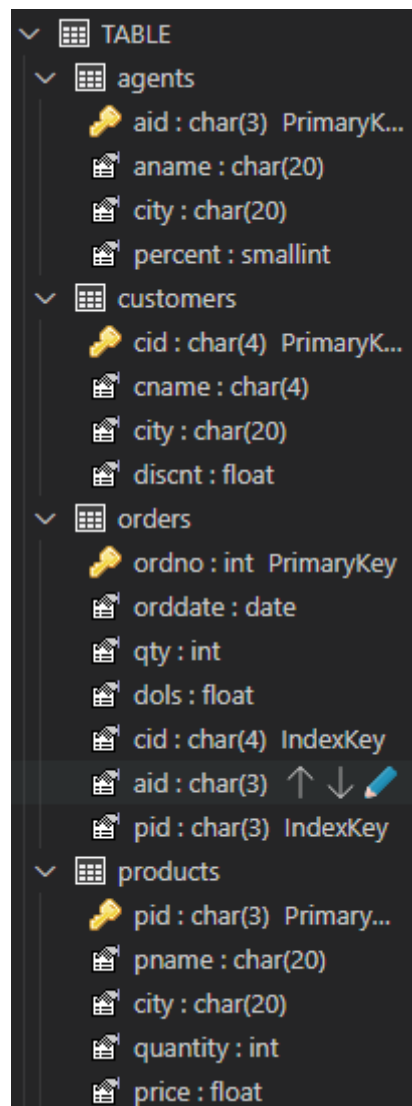
```
1 CREATE TABLE Customers(  
2     cid CHAR(4) PRIMARY KEY,  
3     cname CHAR(4) NOT NULL,  
4     city CHAR(20),  
5     discnt FLOAT  
6 );  
7  
8 CREATE TABLE Agents(  
9     aid CHAR(3) PRIMARY KEY,  
10    aname CHAR(20) NOT NULL,  
11    city CHAR(20),  
12    percent SMALLINT  
13 );  
14  
15 CREATE TABLE Products(  
16     pid CHAR(3) PRIMARY KEY,  
17     pname CHAR(20) NOT NULL,  
18     city CHAR(20),  
19     quantity INT NOT NULL,  
20     price FLOAT NOT NULL  
21 );  
22  
23 CREATE TABLE Orders(  
24     ordno INT PRIMARY KEY,  
25     orddate DATE NOT NULL,
```

```

26     qty INT,
27     dols FLOAT,
28     cid CHAR(4),
29     aid CHAR(3),
30     pid CHAR(3),
31     CONSTRAINT key1
32     FOREIGN KEY (cid) REFERENCES Customers(cid)
33         ON UPDATE CASCADE
34         ON DELETE SET NULL,
35
36     CONSTRAINT key2
37     FOREIGN KEY (cid) REFERENCES Agents(aid)
38         ON UPDATE RESTRICT
39         ON DELETE SET NULL,
40
41     CONSTRAINT key3
42     FOREIGN KEY (pid) REFERENCES Products(pid)
43         ON UPDATE SET NULL
44         ON DELETE SET NULL
45 );

```

实验截图:



## 二.使用SQL语句插入数据

代码:

由于插入的语句类似,每一张表的插入只选取一条SQL语句

```
1  INSERT INTO Customers VALUES('c001', 'Tiptop', 'Duluth', 10.00);
2
3  INSERT INTO Agents VALUES('a01', 'Smith', 'New York', 6);
4
5  INSERT INTO Products VALUES('p01', 'comb', 'Dallas', 111400, 0.50);
6
7  INSERT INTO Orders VALUES(1011, '2016-01-08', 'c001', 'a01', 'p01', 1000,
8  450.00);
```

实验截图:

	aid	aname	city	percent
▶	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
*	NULL	NULL	NULL	NULL

Agents

	cid	cname	city	disct
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
*	NULL	NULL	NULL	NULL

Customers

	pid	pname	city	quantity	price
▶	p01	comb	Dallas	111400	0.5
	p02	brush	Newark	203000	0.5
	p03	razor	Duluth	150600	1
	p04	pen	Duluth	125300	1
	p05	pencil	Dallas	221400	1
	p06	folder	Dallas	123100	2
	p07	case	Newark	100500	1
*	NULL	NULL	NULL	NULL	NULL

Products

	ordno	orddate	cid	aid	pid	qty	dols
▶	1011	2016-01-08	c001	a01	p01	1000	450
	1012	2016-01-12	c001	a06	p01	1000	450
	1013	2016-01-13	c002	a03	p03	1000	880
	1014	2016-01-18	c003	a03	p05	1200	1104
	1015	2016-01-23	c003	a03	p05	1200	1104
	1016	2016-01-25	c006	a01	p01	1000	500
	1017	2016-02-10	c001	a06	p03	600	540
	1018	2016-02-16	c001	a03	p04	600	540
	1019	2016-02-24	c002	a02	p02	400	180
	1020	2016-02-05	c006	a03	p07	600	600
	1021	2016-02-28	c004	a06	p01	1000	460
	1022	2016-03-08	c001	a05	p06	400	720
	1023	2016-03-12	c001	a04	p05	500	450
	1024	2016-03-12	c006	a06	p01	800	400
	1025	2016-04-07	c001	a05	p07	800	720
	1026	2016-05-20	c002	a05	p03	800	704
	1027	2016-05-20	c006	a03	p05	500	450
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Orders

### 三.SQL语言的查询

1.

```

1  SELECT a.aid
2  FROM agents a
3  WHERE NOT EXISTS (
4      SELECT *
5      FROM orders o,customers c
6      WHERE o.aid = a.aid AND o.cid = c.cid AND c.city = 'Duluth'
7  );

```

查询结果:

	aid
▶	a06
*	NULL

2.

```

1  SELECT a.aid
2  FROM agents a
3  WHERE EXISTS(
4      SELECT p.pid
5      FROM products p
6      WHERE NOT EXISTS(
7          SELECT c.cid
8          FROM customers c
9          WHERE (c.city = 'Duluth' OR c.city = 'Kyoto') AND NOT EXISTS(
10             SELECT *
11             FROM orders o
12             WHERE o.cid = c.cid AND o.pid = p.pid AND o.aid =a.aid)))

```

查询结果:

	aid
▶	a06
*	NULL

3.

```
1 SELECT p.pid
2 FROM products p
3 WHERE EXISTS(
4     SELECT *
5     FROM orders o,agents a,customers c
6     WHERE o.pid = p.pid AND o.aid = a.aid AND o.cid = c.cid AND a.city =
7     c.city
8 );
```

查询结果:

	pid
▶	p06
	p07
*	NULL

4.

```
1 SELECT p.pid
2 FROM products p
3 WHERE NOT EXISTS(
4     SELECT c1.city
5     FROM customers c1
6     WHERE NOT EXISTS(
7         SELECT *
8         FROM orders o,customers c2
9         WHERE c1.city = c2.city AND c2.cid = o.cid AND p.pid = o.pid));
```

查询结果:

	pid
▶	p05
*	NULL

5.

```
1 SELECT a.aid,a.percent
2 FROM agents a
3 WHERE NOT EXISTS(
4     SELECT c.cid
5     FROM customers c
6     WHERE c.city = 'Duluth' AND NOT EXISTS(
7         SELECT *
8         FROM orders o
9         WHERE o.cid = c.cid AND a.aid = o.aid));
```

查询结果:

	aid	percent
▶	a06	5
*	NULL	NULL

6.

不使用统计函数:

```
1 SELECT a1.aid
2 FROM agents a1
3 WHERE a1.percent >= all(
4     SELECT a2.percent
5     FROM agents a2
6 );
```

使用统计函数:

```
1 SELECT a1.aid
2 FROM agents a1
3 WHERE a1.percent = ANY(
4     SELECT MAX(a2.percent)
5     FROM agents a2
6 );
```

查询结果:

	aid
▶	a03
*	NULL

7.

```
1 SELECT o1.cid,o1.ordno,o1.orddate,o2.ordno,o2.orddate
2 FROM orders o1,orders o2
3 WHERE o1.cid = o2.cid AND o1.ordno = ANY(
4     SELECT min(o3.ordno)
5     FROM orders o3
6     WHERE o3.cid = o1.cid)
7     AND o2.ordno = ANY(
8     SELECT max(o4.ordno)
9     FROM orders o4
10    WHERE o4.cid = o1.cid
11 );
```

查询结果:

	cid	ordno	orddate	ordno	orddate
▶	c001	1011	2016-01-08	1025	2016-04-07
	c002	1013	2016-01-13	1026	2016-05-20
	c003	1014	2016-01-18	1015	2016-01-23
	c006	1016	2016-01-25	1027	2016-05-20
	c004	1021	2016-02-28	1021	2016-02-28

8.

```
1 SELECT *
2 FROM agents a
3 WHERE a.city LIKE 'N%o%';
```

查询结果:

	aid	aname	city	percent
▶	a01	Smith	New York	6
	a04	Gray	New York	6
*	NULL	NULL	NULL	NULL

9.

```
1 SELECT o.cid,COUNT(*),SUM(dols),AVG(dols)
2 FROM orders o
3 GROUP BY o.cid
4 ORDER BY SUM(dols) DESC;
```

查询结果:

	cid	COUNT(*)	SUM(dols)	AVG(dols)
▶	c001	7	3870	552.8571428571429
	c003	2	2208	1104
	c006	4	1950	487.5
	c002	3	1764	588
	c004	1	460	460

10.

```
1 SELECT o1.pid,o1.aid
2 FROM orders o1
3 GROUP BY o1.pid,o1.aid
4 HAVING SUM(o1.qty) >= ALL(SELECT SUM(o2.qty)
5                             FROM orders o2
6                             WHERE o2.pid = o1.pid
7                             GROUP BY o2.aid
8                             );
```

查询结果:

	pid	aid
▶	p01	a06
	p03	a03
	p05	a03
	p04	a03
	p02	a02
	p06	a05
	p07	a05

## 四.视图的创建与访问

1.带WITH CHECK OPTION选项的视图的创建和访问

```
1 #1_1
2 CREATE VIEW vip
3 AS SELECT *
4     FROM customers c
5     WHERE c.discnt >=10
6     WITH CHECK OPTION;
7 #1_2
8 INSERT INTO vip VALUES('c100','Naz','Istanbul',15);
9 #1_3
10 INSERT INTO vip VALUES('c101','Glass','Houston',5);
11 #1_4
12 INSERT INTO customers VALUES('c101','Glass','Houston',5);
13 #1_5
```

```

14 SELECT *
15 FROM vip;
16 #1_6
17 DELETE FROM customers c
18 WHERE c.cid = 'c100' or c.cid = 'c101';
19 #1_7
20 DROP VIEW vip;

```

(1) 创建视图后:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12

(2) 插入第一条元组(插入成功):

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
	c100	Naz	Istanbul	15
*	NULL	NULL	NULL	NULL

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c100	Naz	Istanbul	15

(3) 插入第二条元组(插入失败):

182 11:08:16 INSERT INTO vip VALUES('c101','Glass','Houston',5) Error Code: 1369. CHECK OPTION failed 'sales.vip'

(4) 直接在customers内插入第二条元组:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
	c100	Naz	Istanbul	15
	c101	Glass	Houston	5
*	NULL	NULL	NULL	NULL

(5) 查询视图:

	cid	cname	city	discnt
	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c100	Naz	Istanbul	15

(6) 删除上述两条元组:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
*	NULL	NULL	NULL	NULL

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12

(7) 删除视图:略

## 2. 嵌套视图的创建和访问



```

1 #2_1
2 CREATE VIEW ordview(ordno,ordyear,ordmonth,cid,aid,pid,qty,dols)
3 AS SELECT ordno,YEAR(orddate),MONTH(orddate),cid,aid,pid,qty,dols
4 FROM orders;
5 #2_2
6 CREATE VIEW monthview(pid,ordmonth,ordnum,dols)
7 AS SELECT pid,ordmonth,COUNT(*),SUM(dols)
8 FROM ordview
9 GROUP BY pid,ordmonth;
10 #2_3
11 SELECT o1.pid,o1.ordmonth,o1.dols
12 FROM monthview o1
13 WHERE o1.dols >= ALL(
14     SELECT max(o2.dols)
15     FROM monthview o2
16     WHERE o2.pid = o1.pid
17 );
18 #2_4
19 DROP VIEW monthview;
20 DROP VIEW ordview;

```

(1)创建ordview:

	ordno	ordyear	ordmonth	cid	aid	pid	qty	dols
▶	1011	2016	1	c001	a01	p01	1000	450
	1012	2016	1	c001	a06	p01	1000	450
	1013	2016	1	c002	a03	p03	1000	880
	1014	2016	1	c003	a03	p05	1200	1104
	1015	2016	1	c003	a03	p05	1200	1104
	1016	2016	1	c006	a01	p01	1000	500
	1017	2016	2	c001	a06	p03	600	540
	1018	2016	2	c001	a03	p04	600	540
	1019	2016	2	c002	a02	p02	400	180
	1020	2016	2	c006	a03	p07	600	600
	1021	2016	2	c004	a06	p01	1000	460
	1022	2016	3	c001	a05	p06	400	720
	1023	2016	3	c001	a04	p05	500	450
	1024	2016	3	c006	a06	p01	800	400
	1025	2016	4	c001	a05	p07	800	720
	1026	2016	5	c002	a05	p03	800	704
	1027	2016	5	c006	a03	p05	500	450

(2)创建monthview:

	pid	ordmonth	ordnum	dols
▶	p01	1	3	1400
	p03	1	1	880
	p05	1	2	2208
	p03	2	1	540
	p04	2	1	540
	p02	2	1	180
	p07	2	1	600
	p01	2	1	460
	p06	3	1	720
	p05	3	1	450
	p01	3	1	400
	p07	4	1	720
	p03	5	1	704
	p05	5	1	450

(3)查询:

	pid	ordmonth	dols
▶	p01	1	1400
	p03	1	880
	p05	1	2208
	p04	2	540
	p02	2	180
	p06	3	720
	p07	4	720

(4)删除:略

## 五.数据完整性约束及事务处理

### 1.事务的提交与回滚

```
1  #1_1
2  SET SQL_SAFE_UPDATES = 0;
3  SET AUTOCOMMIT = 0;
4  #1_2
5  BEGIN;
6  SELECT *
7  FROM Customers c
8  ORDER BY c.cid ASC;
9  #1_3
10 DELETE FROM customers c
11 WHERE c.city = 'Kyoto';
12 #1_4
13 UPDATE customers c
14 SET c.discnt = 15
15 WHERE c.cid = 'c004';
16 #1_5
17 SELECT *
18 FROM Customers c
19 ORDER BY c.cid ASC;
20 #1_6
21 ROLLBACK;
```

(2)查询结果:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
*	NULL	NULL	NULL	NULL

(5)查询结果:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	15
*	NULL	NULL	NULL	NULL

(6)回滚后的结果:

	cid	cname	city	discnt
▶	c001	Tiptop	Duluth	10
	c002	Basics	Dallas	12
	c003	Allied	Dallas	8
	c004	ACME	Duluth	8
	c006	ACME	Kyoto	0
*	NULL	NULL	NULL	NULL

## 2.外关键字的约束管理

```

1  #2_1
2  SET AUTOCOMMIT = 0;
3  #2_2
4  BEGIN;
5  SELECT *
6  FROM orders o
7  ORDER BY o.cid ASC;
8  #2_3
9  UPDATE customers c
10 SET c.cid = 'c009'
11 WHERE c.cid = 'c002';
12 #2_4
13 DELETE FROM customers c
14 WHERE c.cid = 'c004';
15 #2_5
16 SELECT *
17 FROM orders o
18 ORDER BY o.cid ASC;
19 #2_6
20 ROLLBACK;

```

(2)查询结果:

[illegible]

(5)查询结果:

	ordno	orddate	cid	aid	pid	qty	dols
▶	1021	2016-02-28	NULL	a06	p01	1000	460
	1011	2016-01-08	c001	a01	p01	1000	450
	1012	2016-01-12	c001	a06	p01	1000	450
	1017	2016-02-10	c001	a06	p03	600	540
	1018	2016-02-16	c001	a03	p04	600	540
	1022	2016-03-08	c001	a05	p06	400	720
	1023	2016-03-12	c001	a04	p05	500	450
	1025	2016-04-07	c001	a05	p07	800	720
	1014	2016-01-18	c003	a03	p05	1200	1104
	1015	2016-01-23	c003	a03	p05	1200	1104
	1016	2016-01-25	c006	a01	p01	1000	500
	1020	2016-02-05	c006	a03	p07	600	600
	1024	2016-03-12	c006	a06	p01	800	400
	1027	2016-05-20	c006	a03	p05	500	450
	1013	2016-01-13	c009	a03	p03	1000	880
	1019	2016-02-24	c009	a02	p02	400	180
	1026	2016-05-20	c009	a05	p03	800	704
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

(6)回滚后的结果:

	ordno	orddate	cid	aid	pid	qty	dols
▶	1011	2016-01-08	c001	a01	p01	1000	450
	1012	2016-01-12	c001	a06	p01	1000	450
	1013	2016-01-13	c002	a03	p03	1000	880
	1014	2016-01-18	c003	a03	p05	1200	1104
	1015	2016-01-23	c003	a03	p05	1200	1104
	1016	2016-01-25	c006	a01	p01	1000	500
	1017	2016-02-10	c001	a06	p03	600	540
	1018	2016-02-16	c001	a03	p04	600	540
	1019	2016-02-24	c002	a02	p02	400	180
	1020	2016-02-05	c006	a03	p07	600	600
	1021	2016-02-28	c004	a06	p01	1000	460
	1022	2016-03-08	c001	a05	p06	400	720
	1023	2016-03-12	c001	a04	p05	500	450
	1024	2016-03-12	c006	a06	p01	800	400
	1025	2016-04-07	c001	a05	p07	800	720
	1026	2016-05-20	c002	a05	p03	800	704
	1027	2016-05-20	c006	a03	p05	500	450
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

### 3.触发器的创建与应用

```

1  #3_1
2  SET AUTOCOMMIT = 0;
3  BEGIN;
4  CREATE TRIGGER update_orders
5  BEFORE INSERT ON orders FOR EACH ROW
6      UPDATE products p
7      SET p.quantity = p.quantity - NEW.qty
8      WHERE p.pid = NEW.pid;
9  #3_2
10 INSERT INTO orders
11     VALUES(1028, '2020-12-09', 'c001', 'a01', 'p06', 10000, 10000.00);
12 #3_3
13 DROP TRIGGER update_orders;
14

```

(2)插入前:

	pid	pname	city	quantity	price
▶	p01	comb	Dallas	111400	0.5
	p02	brush	Newark	203000	0.5
	p03	razor	Duluth	150600	1
	p04	pen	Duluth	125300	1
	p05	pencil	Dallas	221400	1
	p06	folder	Dallas	123100	2
	p07	case	Newark	100500	1
✱	NULL	NULL	NULL	NULL	NULL

插入后:

	pid	pname	city	quantity	price
▶	p01	comb	Dallas	111400	0.5
	p02	brush	Newark	203000	0.5
	p03	razor	Duluth	150600	1
	p04	pen	Duluth	125300	1
	p05	pencil	Dallas	221400	1
	p06	folder	Dallas	113100	2
	p07	case	Newark	100500	1
✱	NULL	NULL	NULL	NULL	NULL

## 六.删除基本表

```
1 DROP TABLE orders;  
2 DROP TABLE agents;  
3 DROP TABLE customers;  
4 DROP TABLE products;
```

## 实验中遇到的困难及解决办法

一开始图形界面用的Vscode插件很难用,后面改用自带的workbench,workbench yyds!

其它好像没了..

## 参考文献及致谢

菜鸟教程[[MySQL 教程](#) | [菜鸟教程\(runoob.com\)](#)]: