

Week6 作业

一、题目一

1.在数据库中创建一个关系，并将数据导入该关系
代码如下：

```
CREATE TABLE product (  
    product_no INT PRIMARY KEY,  
    name VARCHAR(255),  
    price DECIMAL(10, 2)  
);  
COPY product (product_no, name, price) FROM 'D:/本科计算机/数据库原理/week6 作业/product.txt' DELIMITER ',';  
COPY product TO 'D:/本科计算机/数据库原理/week6 作业/product.csv'  
DELIMITER ','  
CSV HEADER;
```

结果如下：

	product_no ▾	name ▾	price ▾
1	111	hair spray	20.00
2	222	hair splint	102.00

二、题目二

1.添加一个新的商品，编号为 666，名字为 cake，价格不详
代码如下：

```
INSERT INTO product(product_no,name,price)  
VALUES ( product_no 666, name 'cake', price NULL);
```

更新后的关系如下：

	product_no ▾	name ▾	price ▾
1	111	hair spray	20.00
2	222	hair splint	102.00
3	666	cake	<null>

2.使用一条 SQL 语句同时添加 3 个商品，内容自拟
代码如下：

```
INSERT INTO product(product_no,name,price)  
VALUES ( product_no 555, name 'fruit', price 10.8),  
      ( product_no 444, name 'bread', price 15),  
      ( product_no 333, name 'badminton racket', price 800);
```

更新后的关系如下：

	product_no ▾	name ▾	price ▾
1	111	hair spray	20.00
2	222	hair splint	102.00
3	666	cake	<null>
4	555	fruit	10.80
5	444	bread	15.00
6	333	badminton racket	800.00

3.将商品价格统一打 8 折

代码如下：

```
UPDATE product
SET price = price * 0.8;
```

更新后关系如下：

	product_no ▼	name ▼	price ▼
1	111	hair spray	16.00
2	222	hair splint	81.60
3	666	cake	<null>
4	555	fruit	8.64
5	444	bread	12.00
6	333	badminton racket	640.00

4.将价格大于 100 的商品上涨 2%，其余上涨 4%

代码如下：

```
UPDATE product
SET price = CASE
    WHEN price > 100 THEN price * 1.02
    ELSE price * 1.04
END;

DELETE FROM product
WHERE name LIKE "%cake%";
```

更新后关系如下：

	product_no ▼	name ▼	price ▼
1	111	hair spray	16.64
2	222	hair splint	84.86
3	666	cake	<null>
4	555	fruit	8.99
5	444	bread	12.48
6	333	badminton racket	652.80

5.将名字包含 cake 的商品删除

代码如下：

```
DELETE FROM product
WHERE name LIKE '%cake%';
```

更新后关系如下：

	product_no ▼	name ▼	price ▼
1	111	hair spray	16.64
2	222	hair splint	84.86
3	555	fruit	8.99
4	444	bread	12.48
5	333	badminton racket	652.80

6.将价格高于平均价格的商品删除

代码如下：

```
DELETE FROM product
WHERE price > (SELECT AVG(price) FROM product);
```

更新后关系如下：

	product_no ▼	name ▼	price ▼
1	111	hair spray	16.64
2	222	hair splint	84.86
3	555	fruit	8.99
4	444	bread	12.48

三、题目三

1.在关系插入 10 万条数据

代码如下：

```
INSERT INTO product (product_no,name, price)
SELECT
    product_no gs,
    name 'Product' || gs,
    price ROUND((random() * 1000)::numeric, 2)
FROM generate_series(1, 100000) AS gs;
```

更新后关系如下：

	product_no ▼	name ▼	price ▼
1	1	Product1	884.73
2	2	Product2	800.84
3	3	Product3	426.78
4	4	Product4	838.08
5	5	Product5	355.99
6	6	Product6	480.00
7	7	Product7	606.24

2.比较 DELETE 和 TRUNCATE 的差别

DELETE 是数据操作语言（DML），会对表中的数据一行一行地进行删除，效率较低；而 TRUNCATE 是数据定义语言（DDL），直接清空整张表的数据，效率较高。下两张图分别展示了两种操作的所用时间，DELETE 操作用时 104ms，TRUNCATE 操作用时 14ms。

```
university.public> TRUNCATE TABLE product
[2025-04-20 21:43:42] 在 14 ms 内完成
```

```
QUERY PLAN
1 Delete on product (cost=0.00..1637.00 rows=0 width=0) (actual time=104.089..104.089 rows=0 loops=1)
2   -> Seq Scan on product (cost=0.00..1637.00 rows=100000 width=6) (actual time=0.007..7.271 rows=100000 loops=1)
3 Planning Time: 0.927 ms
4 Execution Time: 104.118 ms
```