

五 计算多个指标

学习目标

- 掌握通过SQL在报表中计算一个业务对象的多个指标的方法
- 掌握如何通过SQL计算比率，百分比

1 计算一个业务对象的多个指标

- 我们在制作报表时，经常需要对同一业务对象计算多个指标
- 例如，我们要显示每个订单的产品数量和总价

```
SELECT
  order_id,
  COUNT(product_id) AS products,
  SUM(unit_price * quantity) AS total_price
FROM order_items
GROUP BY order_id;
```

- 在上面的SQL中，我们计算了每一个订单的产品数量 `COUNT ()` 和订单总价 `SUM ()`
- 数据通过 `order_items` 表查询，通过对 `order_id` 进行分组，使用 `COUNT(product_id)` 统计订单项的数量和，通过 `SUM(unit_price * quantity)` 计算每个订单的总价

练习53

- 需求：创建报表统计每个客户的订单数量以及他们的消费金额（折扣后）
- 报表显示三列：
 - 客户的ID (`customer_id`)
 - 订单数 (如 `order_count`)
 - 折扣后所有订单支付的总价 (`total_revenue_after_discount`)

```
SELECT
    o.customer_id,
    COUNT(DISTINCT o.order_id) AS order_count,
    SUM(unit_price * quantity * (1 - discount)) AS
total_revenue_after_discount
FROM orders o
JOIN order_items oi
    ON o.order_id = oi.order_id
GROUP BY o.customer_id;
```

查询结果

customer_id	order_count	total_revenue_after_discount
ALFKI	6	4273
ANATR	4	1402.95
ANTO	7	7023.98
AROUT	13	13390.65
BERGS	18	24927.58
BLAUS	7	3239.8
BLONP	11	18534.08
BOLID	3	4232.85
BONAP	17	21963.25
BOTTM	14	20801.6
BSBEV	10	6089.9
CACTU	6	1814.8
CENTC	1	100.8
CHOPS	8	12348.88
COMMI	5	3810.75
CONSH	3	1719.1
DRACD	6	3763.21
DUMO	4	1615.9
EASTC	8	14761.03
ERNSH	30	104874.98
FAMIA	7	4107.55
FOLIG	5	11666.9
FOLKO	19	29567.56
FRANK	15	26656.56

customer_id	order_count	total_revenue_after_discount
FRANR	3	3172.16
FRANS	6	1545.7
FURIB	8	6427.42
GALED	5	836.7
GODOS	10	11446.36
GOURL	9	8414.14
GREAL	11	18507.45
GROSR	2	1488.7
HANAR	14	32841.37
HILAA	18	22768.76
HUNGC	5	3063.2
HUNGO	19	49979.91
ISLAT	10	6146.3
KOENE	14	30908.38
LACOR	4	1992.05
LAMAI	14	9328.2
LAUGB	3	522.5
LAZYK	2	357
LEHMS	15	19261.41
LETSS	4	3076.47
LILAS	14	16076.6
LINOD	12	16476.56
LONEP	8	4258.6
MAGAA	10	7176.22

customer_id	order_count	total_revenue_after_discount
MAISD	7	9736.08
MEREP	13	28872.19
MORGK	5	5042.2
NORTS	3	649
OCEA	5	3460.2
OLDWO	10	15177.46
OTTIK	10	12496.2
PERIC	6	4242.2
PICCO	10	23128.86
PRINI	5	5044.94
QUEDE	9	6664.81
QUEE	13	25717.5
QUICK	28	110277.3
RANCH	5	2844.1
RATTC	18	51097.8
REGGC	12	7048.24
RICAR	11	12450.8
RICSU	10	19343.78
ROMEY	5	1467.29
SANTG	6	5735.15
SAVEA	31	104361.95
SEVES	9	16215.33
SIMOB	7	16817.1
SPECD	4	2423.35

customer_id	order_count	total_revenue_after_discount
SPLIR	9	11441.63
SUPRD	12	24088.78
THEBI	4	3361
THECR	3	1947.24
TOMSP	6	4778.14
TORTU	10	10812.15
TRADH	6	6850.66
TRAIH	3	1571.2
VAFFE	11	15843.93
VICTE	10	9182.43
VINET	5	1480
WANDK	10	9588.42
WARTH	15	15648.7
WELLI	9	6068.2
WHITC	14	27363.6
WILMK	7	3161.35
WOLZA	7	3531.95

练习54

- 需求：创建报表计算2016年的绩效，计算每位员工处理的订单的总数和总收入
- 报表包含如下字段：
 - `first_name` 员工的名字
 - `last_name` 员工姓氏
 - `order_count` 员工处理的订单总数
 - `order_revenue` 订单处理的订单总收入

```

SELECT
    first_name,
    last_name,
    COUNT(DISTINCT o.order_id) AS order_count,
    SUM(unit_price * quantity * (1 - discount)) AS order_revenue
FROM employees e
LEFT JOIN orders o
    ON e.employee_id = o.employee_id
LEFT JOIN order_items oi
    ON o.order_id = oi.order_id
WHERE order_date BETWEEN '2016-01-01' AND '2016-12-31'
GROUP BY e.employee_id,
    first_name,
    last_name;

```

查询结果

first_name	last_name	order_count	order_revenue
Nancy	Davolio	26	35764.52
Andrew	Fuller	16	21757.06
John	Smith	18	18223.96
Margaret	Peacock	30	49497.11
Steven	Buchanan	11	18383.92
Michael	Suyama	15	16642.6
Robert	King	11	15232.16
Laura	Callahan	19	22240.12
Anne	Dodsworth	5	9894.51
John	Smith	1	448

2 自定义指标

- 在制作报表的时候，经常需要在GROUP BY分组的基础上进一步对数据进行自定义分组，并统计自定义分组中其数量：

```

SELECT
    customer_id,
    COUNT(CASE
        WHEN shipped_date IS NOT NULL
            THEN order_id
    END) AS orders_shipped,
    COUNT(CASE
        WHEN shipped_date IS NULL
            THEN order_id
    END) AS orders_pending
FROM orders
GROUP BY customer_id;

```

- 上面的SQL中，我们按 `customer_id` 将订单数据分组，分组后将每个客户的订单划分成已发货和未发货两类，并统计已发货和未发货的订单数量
- 注意，`COUNT ()` 与 `CASE WHEN` 一起使用了两次，以计算两个不同组中的数据条目数量

练习55

- 创建报表：对于每个类别，请显示库存产品的数量（即“units_in_stock>0”的产品）和未库存的产品数量。该报告应包含三列：
 - `category_name`
 - `products_in_stock`
 - `products_not_in_stock`

```

SELECT
    category_name,
    SUM(CASE
        WHEN units_in_stock > 0 THEN 1
        ELSE 0
    END) AS products_in_stock,
    SUM(CASE
        WHEN units_in_stock = 0 THEN 1
        ELSE 0
    END) AS products_not_in_stock
FROM products p
JOIN categories c
    ON p.category_id = c.category_id

```



```
GROUP BY c.category_id,  
category_name;
```

查询结果

category_name	products_in_stock	products_not_in_stock
Beverages	12	0
Seafood	12	0
Dairy Products	9	1
Produce	5	0
Condiments	11	1
Confections	13	0
Meat/Poultry	3	3
Grains/Cereals	7	0

练习56

- 需求：统计每个订单的全价商品和打折商品数量（同一个ID的商品购买多件只计一次）
- 结果显示三列：
 - `order_id`，订单ID
 - `full_price_product_count` 未打折的订单项数
 - `discount_product_count` 已打折的订单项数

```
SELECT  
    o.order_id,  
    COUNT(CASE WHEN discount = 0 THEN product_id END) AS  
full_price_product_count,  
    COUNT(CASE WHEN discount > 0 THEN product_id END) AS  
discount_product_count  
FROM orders o  
JOIN order_items oi  
    ON o.order_id = oi.order_id  
GROUP BY o.order_id;
```

查询结果(部分)

order_id	full_price_product_count	discount_product_count
11038	2	1
10782	1	0
10725	3	0
10423	2	0
10518	3	0
10356	3	0
10963	0	1
10596	0	3
10282	2	0
10658	1	3
10283	4	0
10579	2	0
10693	1	3
10896	2	0
10660	1	0
10253	3	0
10425	0	2
10774	1	1
10615	1	0
10514	5	0

3 计算比率

在制作报表是，经常需要计算百分比，比率（退货率，好评率...）这样的指标，接下来看一下如何通过SQL实现

步骤1

- 需求：统计所有订单中已经出货的百分比。我们将通过几个步骤来编写此查询。步骤1：

```
SELECT
  COUNT(CASE
    WHEN shipped_date IS NOT NULL
    THEN order_id
  END) AS count_shipped,
  COUNT(order_id) AS count_all
FROM orders;
```

- 在上面的SQL中，我们计算了出货百分比的**分子**和**分母**。分子是发货的订单数，分母是订单的总数
- 我们将 `COUNT ()` 和 `CASE WHEN` 一起使用计算已发货的订单数量，使用 `COUNT ()` 计算订单总数

练习57

- 需求：统计所有折价商品的收入与所有商品的总收入之比。我们也将分步实现。
- 首先，计算两列：
 - `discounted_revenue` 所有订单中打折订单项的收入（折扣后）
 - `total_revenue` 所有订单总收入（折扣后）

```
SELECT
  SUM(CASE
    WHEN discount > 0
    THEN unit_price * quantity * (1 - discount)
  END) AS discounted_revenue,
  SUM(unit_price * quantity * (1 - discount)) AS total_revenue
FROM order_items;
```

查询结果

discounted_revenue	total_revenue
515094.43	1265793.04

步骤2

- 需求：统计所有订单中已经出货的百分比，在上一步中，我们已经计算了分子和分母，现在在查询中添加第三列：

```
SELECT
  COUNT(CASE
    WHEN shipped_date IS NOT NULL
    THEN order_id
  END) AS count_shipped,
  COUNT(order_id) AS count_all,
  COUNT(CASE
    WHEN shipped_date IS NOT NULL
    THEN order_id
  END) / COUNT(order_id) AS shipped_ratio
FROM orders;
```

- 在第三列中，我们通过将第一列的表达式除以第二列的表达式来计算“比率”。
- 计算比率时，分子分母都是整数，使用强制类型转换
`CAST(COUNT(order_id) AS decimal)` 保证计算的结果为小数

练习58

- 继续完成上面的需求：统计所有折价商品的收入与所有商品的总收入之比

```
SELECT
  SUM(CASE
    WHEN discount > 0
    THEN unit_price * quantity * (1 - discount)
  END) AS discounted_revenue,
  SUM(unit_price * quantity * (1 - discount)) AS total_revenue
FROM order_items;
```

- 在第一步的SQL中添加第三列：`discounted_ratio`。它应包含折扣订单项总收入（第1列）与所有订单项总收入（第2列）的比率。

```

SELECT
  SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) AS discounted_revenue,
  SUM(unit_price * quantity * (1 - discount)) AS
total_revenue,
  SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) / SUM(unit_price * quantity * (1 - discount)) AS
discounted_ratio
FROM order_items;

```

查询结果

discounted_revenue	total_revenue	discounted_ratio
515094.43	1265793.04	0.406934

步骤3

- 在步骤2中我们算出了比率，但通需要将结果保留指定位数的有效数字，可以使用 `ROUND(value, decimal_places)` 函数

```

SELECT
  COUNT(CASE
    WHEN shipped_date IS NOT NULL
      THEN order_id
    END) AS count_shipped,
  COUNT(order_id) AS count_all,
  ROUND(COUNT(CASE
    WHEN shipped_date IS NOT NULL
      THEN order_id
    END) / CAST(COUNT(order_id) AS decimal), 2) AS shipped_ratio
FROM orders;

```

- `ROUND(..., 2)` 将比率四舍五入到**小数点后两位**。

练习59

```
SELECT
  SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) AS discounted_revenue,
  SUM(unit_price * quantity * (1 - discount)) AS
total_revenue,
  SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) / SUM(unit_price * quantity * (1 - discount)) AS
discounted_ratio
FROM order_items;
```

- 继续完成之前的需求，修改上面的SQL，将“discounted_ratio”列中的比率四舍五入到小数点后三位。

```
SELECT
  SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) AS discounted_revenue,
  SUM(unit_price * quantity * (1 - discount)) AS
total_revenue,
  ROUND(SUM(CASE
    WHEN discount > 0
      THEN unit_price * quantity * (1 - discount)
    END) / SUM(unit_price * quantity * (1 - discount)), 2) AS
discounted_ratio
FROM order_items;
```

查询结果

discounted_revenue	total_revenue	discounted_ratio
515094.43	1265793.04	0.41

步骤4

- 我们最终的需求是要计算出一个百分比，所以对查询做进一步修改：

```
SELECT
    COUNT(CASE
        WHEN shipped_date IS NOT NULL
        THEN order_id
    END) AS count_shipped,
    COUNT(order_id) AS count_all,
    ROUND(COUNT(CASE
        WHEN shipped_date IS NOT NULL
        THEN order_id
    END) / COUNT(order_id) * 100, 2) AS shipped_ratio
FROM orders;
```

- 将ROUND () 函数的结果乘以100以获得百分比

练习60

- 需求：统计缺货商品的百分比
- 显示三列：count_discontinued, count_all 和 percentage_discontinued 保留两位有效数字

```
SELECT
    COUNT(CASE
        WHEN discontinued IS TRUE
        THEN product_id
    END) AS count_discontinued,
    COUNT(product_id) AS count_all,
    ROUND(COUNT(CASE
        WHEN discontinued IS TRUE
        THEN product_id
    END) / COUNT(product_id) * 100, 2) AS
percentage_discontinued
FROM products;
```

查询结果

count_discontinued	count_all	percentage_discontinued
8	77	10.39

4 分组计算百分比

- 我们还可以计算分组中的比率/百分比

```
SELECT
  ship_country,
  COUNT(CASE
    WHEN shipped_date IS NOT NULL
    THEN order_id
  END) AS count_shipped,
  COUNT(order_id) AS count_all,
  ROUND(COUNT(CASE
    WHEN shipped_date IS NOT NULL
    THEN order_id
  END) / COUNT(order_id) * 100, 2) AS shipped_ratio
FROM orders
GROUP BY ship_country;
```

- 在 **GROUP BY** 和 **SELECT** 子句中添加了 **ship_country** 列，可以统计每个国家/地区的发货订单百分比

练习61

```
SELECT
  COUNT(CASE
    WHEN discontinued IS TRUE
    THEN product_id
  END) AS count_discontinued,
  COUNT(product_id) AS count_all,
  ROUND(COUNT(CASE
    WHEN discontinued IS TRUE
    THEN product_id
  END) / COUNT(product_id) * 100, 2) AS
percentage_discontinued
FROM products;
```

- 修改上面的SQL，按产品类别分组统计每类商品的缺货比例


```

SELECT
    category_name,
    COUNT(CASE
        WHEN discontinued IS TRUE
        THEN product_id
    END) AS count_discontinued,
    COUNT(product_id) AS count_all,
    ROUND(COUNT(CASE
        WHEN discontinued IS TRUE
        THEN product_id
    END) / COUNT(product_id) * 100, 2) AS
percentage_discontinued
FROM products p
JOIN categories c
    ON p.category_id = c.category_id
GROUP BY c.category_id,
    category_name;

```

查询结果

category_name	count_discontinued	count_all	percentage_discontinued
Grains/Cereals	1	7	14.29
Dairy Products	0	10	0.00
Meat/Poultry	4	6	66.67
Condiments	1	12	8.33
Produce	1	5	20.00
Beverages	1	12	8.33
Seafood	0	12	0.00
Confections	0	13	0.00

练习62

- 需求：统计每位员工处理的订单中，法国客户下单的百分比
- 结果中包含五个字段：

- `first_name` , `last_name` , `count_france` , `count_all` 和 `percentage_france` (保留一位有效数字)
- 员工名字, 员工姓氏, 法国客户下单数量, 所有订单数量, 法国客户下单百分比
- 注意: 要查找客户所在的国家/地区, 请使用 `customers` 表中的 `country` 列

```
SELECT
    first_name,
    last_name,
    COUNT(CASE
        WHEN c.country = 'France'
        THEN order_id
    END) AS count_france,
    COUNT(order_id) AS count_all,
    ROUND(COUNT(CASE
        WHEN c.country = 'France'
        THEN order_id
    END) / COUNT(order_id) * 100, 1) AS percentage_france
FROM orders o
JOIN customers c
    ON o.customer_id = c.customer_id
JOIN employees e
    ON e.employee_id = o.employee_id
GROUP BY e.employee_id,
    first_name,
    last_name;
```

查询结果

first_name	last_name	count_france	count_all	percentage_france
Steven	Buchanan	5	41	12.2
Michael	Suyama	9	64	14.1
Margaret	Peacock	14	151	9.3
John	Smith	13	123	10.6
Anne	Dodsworth	3	43	7
Nancy	Davolio	8	121	6.6
John	Smith	0	1	0
Laura	Callahan	8	100	8
Andrew	Fuller	10	94	10.6
Robert	King	3	63	4.8

5 统计总量并计算占比

- 接下来我们要在报表中统计某个指标的总量并分组计算占比
- 需求：创建报表，统计2016年7月下订单的客户以及每个客户的消费金额占2016年7月总销售金额的占比

```

WITH total_sales AS (
  SELECT
    SUM(quantity * unit_price) AS july_sales
  FROM order_items oi
  JOIN orders o
    ON o.order_id = oi.order_id
  WHERE order_date >= '2016-07-01' AND order_date < '2016-08-01'
)
SELECT
  c.customer_id,
  SUM(quantity * unit_price) AS revenue,
  ROUND(SUM(quantity * unit_price) / total_sales.july_sales * 100, 2) AS revenue_percentage
FROM total_sales,
  customers c
JOIN orders o

```

```

    ON c.customer_id = o.customer_id
JOIN order_items oi
    ON oi.order_id = o.order_id
WHERE order_date >= '2016-07-01' AND order_date < '2016-08-01'
GROUP BY c.customer_id, total_sales.july_sales;

```

- 在CTE中，我们仅计算2016年7月的总销售收入
- 在外部查询中，我们查询了2016年7月有消费的客户的 `customer_id` 与该客户在2016年7月的消费金额
- 最后一列中，我们将客户的消费金额（来自上一列） / CTE中计算的2016年7月的总销售收入计算出每个客户贡献的销售收入占比
- 注意：
 - 1、我们必须在 `GROUP BY` 子句中添加 `july_sales` 列，因为它没有与任何聚合函数一起使用。
 - 2、我们通过以下方式连接“total_sales”和“customers”表：

```
FROM total_sales, customers c
```

将 `total_sales`（CTE的计算结果）与 `customers` 表中的所有行组合在一起

练习63

- 需求：创建报表，统计每个员工2017年处理的订单数量，及其在2017年所有订单中的占比
- 显示以下列：
 1. `employee_id`
 2. `first_name`
 3. `last_name`
 4. `order_count` 该员工在2017年处理的订单数
 5. `order_count_percentage` 该员工2017年处理的订单占2017年全部订单的百分比，将最后一列的值四舍五入到小数点后两位。

```

WITH total_count AS(
  SELECT
    COUNT(order_id) AS all_orders

```

```

FROM orders
WHERE order_date >= '2017-01-01' AND order_date < '2018-01-01'
)
SELECT
    e.employee_id,
    e.first_name,
    e.last_name,
    COUNT(order_id) AS order_count,
    ROUND(COUNT(order_id) / total_count.all_orders * 100, 2) AS
order_count_percentage
FROM total_count,
    employees e
JOIN orders o
    ON e.employee_id = o.employee_id
WHERE order_date >= '2017-01-01' AND order_date < '2018-01-01'
GROUP BY e.employee_id,
    e.first_name,
    e.last_name,
    total_count.all_orders;

```

查询结果

employee_id	first_name	last_name	order_count	order_count_percentage
1	Nancy	Davolio	55	13.48
2	Andrew	Fuller	41	10.05
3	John	Smith	71	17.40
4	Margaret	Peacock	81	19.85
5	Steven	Buchanan	18	4.41
6	Michael	Suyama	33	8.09
7	Robert	King	36	8.82
8	Laura	Callahan	54	13.24
9	Anne	Dodsworth	19	4.66

练习64

- 需求：创建报表，按不同国家/地区统计2018年发往该国家/地区的订单所产生的收入百分比。
- 显示三列：
 1. `ship_country`。
 2. `revenue` -2018年运往该国的所有订单产生的总收入。
 3. `revenue_percentage` -2018年发往该国的订单所产生的当年收入的百分比。

将百分比四舍五入到小数点后两位。按“收入”列按降序对结果进行排序。

```
WITH total_sales AS (  
    SELECT  
        SUM(quantity * unit_price) AS sales_2018  
    FROM order_items oi  
    JOIN orders o  
        ON o.order_id = oi.order_id  
    WHERE shipped_date >= '2018-01-01' AND shipped_date < '2019-  
01-01'  
)  
SELECT  
    o.ship_country,  
    SUM(quantity * unit_price) AS revenue,  
    ROUND(SUM(quantity * unit_price) / total_sales.sales_2018 *  
100, 2) AS revenue_percentage  
FROM total_sales,  
    orders o  
JOIN order_items oi  
    ON oi.order_id = o.order_id  
WHERE shipped_date >= '2018-01-01' AND shipped_date < '2019-  
01-01'  
GROUP BY o.ship_country,  
    total_sales.sales_2018  
ORDER BY revenue DESC;
```

查询结果

ship_country	revenue	revenue_percentage
USA	101311.88	21.65
Germany	85970.26	18.38
Brazil	44275.12	9.46
Austria	39714.40	8.49
UK	27690.16	5.92
Sweden	24084.40	5.15
Ireland	22796.34	4.87
Venezuela	21237.48	4.54
France	19433.16	4.15
Belgium	16609.08	3.55
Canada	11104.90	2.37
Switzerland	9341.30	2.00
Spain	8278.44	1.77
Denmark	8257.25	1.76
Italy	6762.40	1.45
Argentina	5921.50	1.27
Mexico	4244.90	0.91
Norway	3976.75	0.85
Portugal	2701.90	0.58
Finland	2287.00	0.49
Poland	1865.10	0.40

小结

- 回顾一下本小节介绍的概念：
 - 1.我们可以在单个查询中使用多个聚合函数来计算给定业务对象的多个指标。
 - 2.我们可以在单个查询中多次使用“COUNT (CASE WHEN ...)”来对多个组中的对象进行计数。
 - 3.计算百分比时，如果参与计算的分子分母都是整型，需要强转数据类型为decimal。可以使用 ROUND () 函数做四舍五入保留指定位数的有效数字，如下所示：

```
ROUND(count_shipped / count_all * 100, 2) AS shipped_ratio
```

练习65

- 需求：统计每个供应商提供的产品数量和每个供应商的平均单价
- 显示三列：“company_name”，“product_count”和“avg_unit_price”。

Find the number of products supplied and the average unit price for each supplier. Show three columns: `company_name`, `product_count`, and `avg_unit_price`.

```
SELECT
    company_name,
    COUNT(product_id) AS product_count,
    AVG(unit_price) AS avg_unit_price
FROM products p
JOIN suppliers s
    ON p.supplier_id = s.supplier_id
GROUP BY s.supplier_id,
    company_name;
```

查询结果

company_name	product_count	avg_unit_price
Exotic Liquids	3	15.666667
New Orleans Cajun Delights	4	20.35
Grandma Kelly's Homestead	3	31.666667
Tokyo Traders	3	46
Cooperativa de Quesos 'Las Cabras'	2	29.5
Mayumi's	3	14.916667
Pavlova, Ltd.	5	35.57
Specialty Biscuits, Ltd.	4	28.175
PB Knäckebröd AB	2	15
Refrescos Americanas LTDA	1	4.5
Heli Süßwaren GmbH & Co. KG	3	29.71
Plutzer Lebensmittelgroßmärkte AG	5	44.678
Nord-Ost-Fisch Handelsgesellschaft mbH	1	25.89
Formaggi Fortini s.r.l.	3	26.433333
Bigfoot Breweries	6	17.666667
Svensk Sjöföda AB	3	20
Aux joyeux ecclésiastiques	2	140.75
New England Seafood Cannery	2	14.025
Leka Trading	3	26.483333
Lyngbysild	2	10.75
Zaanse Snoepfabriek	2	11.125
Karkki Oy	3	18.083333
G'day, Mate	3	30.933333

company_name	product_count	avg_unit_price
Ma Maison	2	15.725
Pasta Buttini s.r.l.	2	28.75
Escargots Nouveaux	1	13.25
Gai pâturage	2	44.5
Forêts d'érables	2	38.9

练习66

- 需求：创建报表，分类统计商店中在售商品数量和已经停产商品数量
- 显示三列： `category_name` , `products_available` 和 `products_discontinued`
 - 类别名称，在售商品数量，停产商品数量

```

SELECT
  category_name,
  SUM(CASE
    WHEN discontinued IS FALSE
    THEN 1
    ELSE 0
  END) AS products_available,
  SUM(CASE
    WHEN discontinued IS TRUE
    THEN 1
    ELSE 0
  END) AS products_discontinued
FROM products p
JOIN categories c
  ON p.category_id = c.category_id
GROUP BY c.category_id,
  c.category_name;

```

查询结果

category_name	products_available	products_discontinued
Grains/Cereals	6	1
Dairy Products	10	0
Meat/Poultry	2	4
Condiments	11	1
Produce	4	1
Beverages	11	1
Seafood	12	0
Confections	13	0

练习67

- 需求：创建报表，统计所有产品中缺货（`units_in_stock = 0`）商品的百分比
- 显示三列：`count_unavailable`，`count_all` 和 `unavailable_percentage`。将百分比四舍五入到小数点后三位
 - 缺货商品（按 `product_id` 统计），商品总数量（按 `product_id` 统计），缺货商品占比

```
SELECT
  COUNT(CASE
    WHEN units_in_stock = 0 THEN product_id
  END) AS count_unavailable,
  COUNT(product_id) AS count_all,
  ROUND(COUNT(CASE
    WHEN units_in_stock = 0 THEN product_id
  END) / COUNT(product_id) * 100, 3) AS unavailable_percentage
FROM products;
```

查询结果

count_unavailable	count_all	unavailable_percentage
5	77	6.494

练习68

- 需求：创建报表，统计每位客户在2016年所下订单， 占有所有订单的百分比
- 结果显示三列：
 1. `customer_id` 。
 2. `order_count` -该客户在2016年下的订单数。
 3. `order_count_percentage` -订单的百分比。 将值四舍五入到小数点后两位。

```
WITH total_count AS (  
    SELECT  
        COUNT(order_id) AS all_orders  
    FROM orders  
    WHERE order_date BETWEEN '2016-01-01' AND '2016-12-31'  
)  
SELECT  
    customer_id,  
    COUNT(order_id) AS order_count,  
    ROUND(COUNT(order_id) /total_count.all_orders * 100, 2) AS  
order_count_percentage  
FROM total_count, orders  
WHERE order_date BETWEEN '2016-01-01' AND '2016-12-31'  
GROUP BY customer_id,  
    total_count.all_orders;
```

查询结果

customer_id	order_count	order_count_percentage
ANATR	1	0.66
ANTON	1	0.66
AROUT	2	1.32
BERGS	3	1.97
BLONP	3	1.97
BOLID	1	0.66
BONAP	3	1.97
BOTTM	1	0.66
BSBEV	1	0.66
CENTC	1	0.66
CHOPS	2	1.32
COMMI	1	0.66
DRACD	2	1.32
DUMON	1	0.66
EASTC	1	0.66
ERNSH	6	3.95
FAMIA	2	1.32
FOLKO	3	1.97
FRANK	4	2.63
FURIB	2	1.32