



# Ratios

Calculating Ratios

Two Methods

Example: Subscription Rate

Example: Immediate Order

## ▼ Calculating Ratios

- The problem is to compute a ratio or a percentage given some data entries or system logs.
- For example:
  - Query the percentage of users who had some behavior from a table with user behavior logs.
  - Query the percentage of products that satisfy some criteria based on a purchase history table.
- Usually the numerator and the denominator are counts that come from the same table.

## ▼ Two Methods



There are 2 common ways to compute a ratio.

1. **Subquery**: Use a subquery to compute the **denominator** and the main query to compute the numerator and ratio.
2. **CASE WHEN** : Use **CASE WHEN** to compute the **numerator** and the main query to compute the denominator and ratio.

## ▼ Example: Subscription Rate

Table: Subscription

| user_id | premium |
|---------|---------|
|---------|---------|

- The **premium** column shows whether the user has opted in for the premium subscription.

| user_id | premium |
|---------|---------|
| 1       | TRUE    |
| 2       | FALSE   |
| 3       | TRUE    |

- Write a query to calculate the **premium subscription rate**: the count of premium subscribers over the total number of users.

#### ▼ Subquery method

- Use a subquery to get the denominator.

```
SELECT
  COUNT(user_id) * 1.0 / (SELECT COUNT(user_id) FROM subscription)
  AS ratio
FROM subscription
WHERE premium = 'TRUE';
```

- In MS SQL server, we need to multiply the numerator by 1.0 to avoid integer division, which will return 0.

#### ▼ CASE WHEN method

- Use `CASE WHEN` to return either 0 or 1 based on a certain condition.

##### ▼ SUM → numerator

- First use the `CASE WHEN` statement, then `SUM` over all the numbers returned → the count of rows that meet the condition we specified.

```
SUM(
  CASE
    WHEN condition THEN 1
    ELSE 0
  END
)
```

- We usually use this method to compute the **numerator**.

```
SELECT
  SUM(CASE WHEN premium = 'TRUE' THEN 1 ELSE 0 END) * 1.0 / COUNT(user_id)
  AS ratio
FROM subscription;
```

- The `SUM(CASE WHEN...)` statement will give us the count of premium subscribers.
- Multiply the numerator by 1.0 to avoid the integer division.

##### ▼ AVG

- ▼ An easier (i.e. better) way to get ratio by avoiding calculating the denominator.

```
SELECT
  AVG(CASE WHEN premium = 'TRUE' THEN 1.0 ELSE 0.0 END)
```

```
AS ratio
FROM subscription;
```

- This method only works when the denominator is the total count.
- **Note:** Change the return value to decimals, 1.0 and 0.0 to avoid integer division in the `AVG` function.

## ▼ Example: Immediate Order

Delivery Table Schema:

| column name        | type |
|--------------------|------|
| customer_id        | int  |
| order_date         | date |
| pref_delivery_date | date |

Table: Delivery

| customer_id | order_date | pref_delivery_date |
|-------------|------------|--------------------|
| 1           | 2019-08-01 | 2019-08-02         |
| 2           | 2019-08-02 | 2019-08-02         |
| 1           | 2019-09-02 | 2019-09-04         |
| 3           | 2019-10-12 | 2019-10-12         |
| 3           | 2019-10-09 | 2019-10-11         |
| 2           | 2019-08-11 | 2019-08-13         |
| 4           | 2019-01-09 | 2019-01-09         |

- Query **the percentage of users who placed their first order as an immediate order**.
  - The first order is the earliest order that a customer placed based on the order date.
  - The immediate order is defined as the same-day order; orders with the same customer preferred delivery date and the order date.
  - Get the result as a decimal named **immediate\_percentage**.

| customer_id | order_date | pref_delivery_date |
|-------------|------------|--------------------|
| 1           | 2019-08-01 | 2019-08-02         |
| 2           | 2019-08-02 | 2019-08-02         |
| 1           | 2019-09-02 | 2019-09-04         |
| 3           | 2019-10-12 | 2019-10-12         |
| 3           | 2019-10-09 | 2019-10-11         |
| 2           | 2019-08-11 | 2019-08-13         |
| 4           | 2019-01-09 | 2019-01-09         |

The immediate percentage is 50% - customers with id 2 and 4 satisfy the criteria, and the other 2 don't.

### Result:

| immediate_percentage |
|----------------------|
| 0.5                  |

### ▼ Subquery method

- Numerator: Customers whose first order is an immediate order.

```
SELECT
  customer_id
FROM delivery
```

```
GROUP BY customer_id
HAVING MIN (order_date) = MIN (pref_delivery_date)
```

- Use a **WITH** common table expression to store the result we just got.

```
WITH first_order AS (
  SELECT
    customer_id
  FROM delivery
  GROUP BY customer_id
  HAVING MIN(order_date) = MIN (pref_delivery_date)
)

SELECT
  COUNT(customer_id) * 1.0 /
  (SELECT COUNT(DISTINCT customer_id) FROM delivery)
  AS immediate_percentage
FROM first_order
```

#### ▼ **CASE WHEN** method

```
SELECT
  AVG(CASE
    WHEN first_order_date = pref_delivery_date THEN 1.0
    ELSE 0.0 END
  ) AS immediate_percentage
FROM ...
```

- Need a table which contains the **first order date for each customer** → get the rankings of the order dates and select the ranking = 1.

```
SELECT
  *,
  ROW_NUMBER() OVER (PARTITION BY customer_id ORDER BY order_date) AS
  order_rk
FROM delivery
```

#### **Result:**

| customer_id | order_date | pref_delivery_date | order_rk |
|-------------|------------|--------------------|----------|
| 1           | 2019-08-01 | 2019-08-02         | 1        |
| 2           | 2019-08-02 | 2019-08-02         | 1        |
| 1           | 2019-09-02 | 2019-09-04         | 2        |
| 3           | 2019-10-12 | 2019-10-12         | 1        |
| 3           | 2019-10-09 | 2019-10-11         | 2        |
| 2           | 2019-08-11 | 2019-08-13         | 2        |
| 4           | 2019-01-09 | 2019-01-09         | 1        |

- Put the query into a **WITH CTE** and call the table **ordered\_delivery**.

```

WITH ordered_delivery
AS (SELECT
    *,
    ROW_NUMBER() OVER (PARTITION BY customer_id ORDER BY order_date) AS
    order_rk
FROM delivery)

SELECT
    AVG(CASE
        WHEN order_date = pref_delivery_date THEN 1.0
        ELSE 0.0 END
    ) AS immediate_percentage
FROM ordered_delivery
WHERE order_rk = 1 # first order

```

- In the main query, select the first order for each customer.