

Product Analysis

Log onto [Mode Analytics](#) and from the home page, create a new report by clicking on the green plus sign button in the upper right-hand corner. Enter the starter code where provided for each exercise. You may want to create a new tab for each exercise.

Please use the discussion forum for any questions and/or comments you might have. Once you have tried the exercises, feel free to watch the solutions video. Good luck with your practice!

Note: When querying a table, remember to prepend `dsv1069`, which is the schema, or folder that contains the course data.

Exercise 0: Count how many users we have

Starter Code:

```
SELECT * FROM dsv1069.users
LIMIT 100
```

Exercise 1: Find out how many users have ever ordered

Starter Code:

```
SELECT *
FROM
  dsv1069.users
```

Exercise 2:

--Goal find how many users have reordered the same item

Starter Code:

```
SELECT * FROM dsv1069.orders
```

Exercise 3:

--Do users even order more than once?

Starter Code:

```
SELECT * FROM dsv1069.orders
```

Exercise 4:

--Orders per item

Starter Code:

```
SELECT * FROM dsv1069.orders
```

Exercise 5:

--Orders per category

Starter Code:

```
SELECT * FROM dsv1069.orders
```

Exercise 6:

--Goal: Do user order multiple things from the same category?

Starter Code:

```
SELECT
  user_id,
  item_id,
  COUNT(line_item_id) as items_ordered
FROM
  dsv1069.orders
GROUP BY
  user_id,
  item_id
```

Exercise 7:

--Goal: Find the average time between orders

--Decide if this analysis is necessary

Starter Code:

```

SELECT
    first_orders.user_id,
    date(first_orders.paid_at) AS first_order_date,
    date(second_orders.paid_at) AS second_order_date,
    date(second_orders.paid_at) - date(first_orders.paid_at) AS date_diff
    -- ^ THIS WILL VARY DEPENDING ON YOUR FLAVOR OF SQL
FROM
    (
    SELECT
        user_id,
        invoice_id,
        paid_at,
        DENSE_RANK( ) OVER (PARTITION BY user_id ORDER BY paid_at ASC)
        AS order_num
    FROM
        dsv1069.orders
    ) first_orders
JOIN
    (
    SELECT
        user_id,
        invoice_id,
        paid_at,
        DENSE_RANK( ) OVER (PARTITION BY user_id ORDER BY paid_at ASC)
        AS order_num
    FROM
        dsv1069.orders
    ) second_orders
ON
    first_orders.user_id = second_orders.user_id
WHERE
    first_orders.order_num = 1
AND
    second_orders.order_num = 2

```