**CDACL - 006 - Market Analysis**



**Team ID:** PTID-CDA-DEC-24-283

**Project ID:** CDACL-006

**Project Name:** Market Analysis

**By**

***Team member 1: -***

**Name: Raasi P**

**Email:** [**rashireddy981212@gmail.com**](mailto:rashireddy981212@gmail.com)

***Team member 2: -***

**Name: Neetu Sharma**

**Email:** [**neets.2012@gmail.com**](mailto:neets.2012@gmail.com)

**Perform the Problem Queries:**

1. What are the top 10 aisles with the highest number of products?
2. How many unique departments are there in the dataset?
3. What is the distribution of products across departments?
4. What are the top 10 products with the highest reorder rates?
5. How many unique users have placed orders in the dataset?
6. What is the average number of days between orders for each user?
7. What are the peak hours of order placement during the day?
8. How does order volume vary by day of the week?
9. What are the top 10 most ordered products?
10. How many users have placed orders in each department?
11. What is the average number of products per order?
12. What are the most reordered products in each department?
13. How many products have been reordered more than once?
14. What is the average number of products added to the cart per order?
15. How does the number of orders vary by hour of the day?
16. What is the distribution of order sizes (number of products per order)?
17. What is the average reorder rate for products in each aisle?
18. How does the average order size vary by day of the week?
19. What are the top 10 users with the highest number of orders?
20. How many products belong to each aisle and department?

Also, analyse customer purchasing behaviour and product performance to optimize marketing strategies and improve customer satisfaction.

**Attributes information:**

1. **Aisles Dataset:**

* aisle\_id: Unique identifier for the aisle. *(****Primary Keys****)*
* aisle: Name of the aisle.

1. **Departments Dataset:**

* department\_id: Unique identifier for the department. *(****Primary Keys****)*
* department: Name of the department.

1. **Order Products Prior Dataset:**

* order\_id: Unique identifier for the order. *(****Primary Keys****)*
* product\_id: Unique identifier for the product.
* add\_to\_cart\_order: Order in which the product was added to the cart.
* reordered: Indicates whether the product was reordered in this order (1 for reordered, 0 for not reordered).

1. **Orders Dataset:**

* order\_id: Unique identifier for the order. *(****Primary Keys****)*
* user\_id: Unique identifier for the user.
* eval\_set: Evaluation set (prior, train, test).
* order\_number: Order sequence number for the user.
* order\_dow: Day of the week the order was placed.
* order\_hour\_of\_day: Hour of the day the order was placed.
* days\_since\_prior\_order: Number of days since the last order.

1. **Products Dataset:**

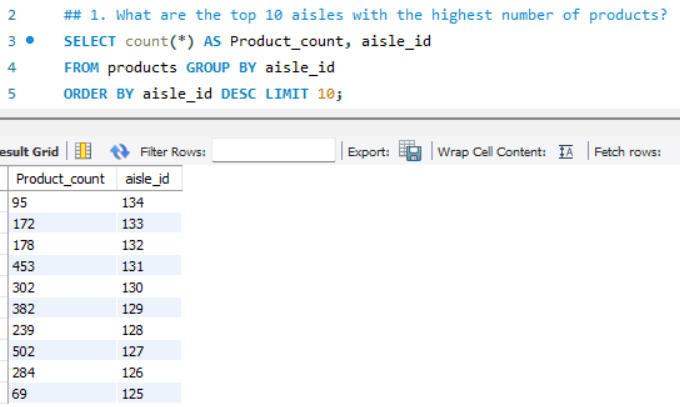
* product\_id: Unique identifier for the product.
* product\_name: Name of the product.
* aisle\_id: Identifier for the aisle the product belongs to. *(****Primary Keys****)*
* department\_id: Identifier for the department the product belongs to.

**QUERIES WITH ANSWERS**

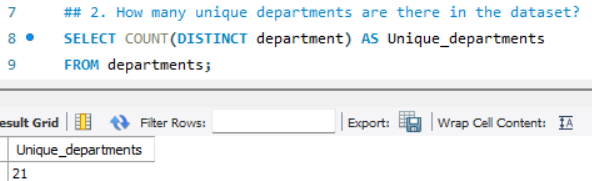
**PERFORM THE PROBLEM QUERIES**

|  |
| --- |
| *1) What are the top 10 aisles with the highest number of products?*  SELECT count(\*) AS Product\_count, aisle\_id  FROM products GROUP BY aisle\_id  ORDER BY aisle\_id DESC LIMIT 10; |
| *2) How many unique departments are there in the dataset?*  SELECT COUNT(DISTINCT department) AS Unique\_departments  FROM departments; |
| *3) What is the distribution of products across departments?*  SELECT department\_id, count(\*) AS Product\_count  FROM products GROUP BY department\_id  ORDER BY department\_id; |
| *4) What are the top 10 products with the highest reorder rates?*  SELECT reordered, product\_id  FROM order\_products\_train  ORDER BY reordered  DESC LIMIT 10; |
| *5) How many unique users have placed orders in the dataset?*  SELECT COUNT(DISTINCT user\_id) AS Unique\_users  FROM orders; |
| *6) What is the average number of days between orders for each user?*  SELECT user\_id, AVG(days\_since\_prior\_order)  AS Avg\_days\_between\_orders  FROM orders GROUP BY user\_id; |
| *7) What are the peak hours of order placement during the day?*  SELECT order\_hour\_of\_day, count(\*) as order\_count  FROM orders  GROUP BY order\_hour\_of\_day  ORDER BY order\_count DESC; |
| *8) How does order volume vary by day of the week?*  SELECT order\_dow, count(\*) AS order\_count  FROM orders  GROUP BY order\_dow  ORDER BY order\_count DESC; |
| *9) What are the top 10 most ordered products?*  SELECT product\_id, COUNT(\*) AS product\_count  FROM order\_products\_train  GROUP BY product\_id  ORDER BY product\_count DESC LIMIT 10; |
| *10) How many users have placed orders in each department?*  SELECT p.department\_id, count(DISTINCT o.user\_id) AS user\_count  FROM products p JOIN order\_products\_train opt  ON p.product\_id = opt.product\_id  JOIN orders o ON opt.order\_id = o.order\_id  GROUP BY p.department\_id; |
| *11) What is the average number of products per order?*  SELECT order\_id, AVG(product\_id) AS avg\_products\_per\_order  FROM order\_products\_train  GROUP BY order\_id; |
| *12) What are the most reordered products in each department?*  SELECT p.department\_id, opt.product\_id,  SUM(opt.reordered) AS reordered\_products  FROM products p JOIN order\_products\_train opt  ON p.product\_id = opt.product\_id  GROUP BY opt.product\_id  ORDER BY reordered\_products DESC; |
| *13) How many products have been reordered more than once?*  SELECT COUNT(\*) AS products\_reordered\_more\_than\_once  FROM  (SELECT product\_id, SUM(reordered) AS reorder\_count  FROM order\_products\_train  GROUP BY product\_id HAVING reorder\_count > 1)  AS subquery; |
| *14) What is the average number of products added to the cart per order?*  SELECT order\_id, AVG(product\_id) AS avg\_product\_count  FROM order\_products\_train  GROUP BY order\_id; |
| *15) What is the average number of products added to the cart per order?*  SELECT order\_hour\_of\_day, COUNT(\*) AS order\_number\_count  FROM orders  GROUP BY order\_hour\_of\_day  ORDER BY order\_number\_count DESC; |
| *16) What is the distribution of order sizes (number of products per order)?*  SELECT order\_id, COUNT(\*) AS no\_of\_products\_per\_order  FROM order\_products\_train  GROUP BY order\_id; |
| *17) What is the average reorder rate for products in each aisle?*  SELECT p.aisle\_id, AVG(opt.reordered) AS avg\_reorder\_rate  FROM order\_products\_train opt  JOIN products p ON opt.product\_id = p.product\_id  GROUP BY p.aisle\_id; |
| *18) How does the average order size vary by day of the week?*  SELECT o.order\_dow AS order\_day, AVG(order\_number) AS avg\_order\_size  FROM order\_products\_train opt  JOIN orders o ON opt.order\_id = o.order\_id  GROUP BY o.order\_dow; |
| *19) What are the top 10 users with the highest number of orders?*  SELECT user\_id, COUNT(order\_id) AS order\_count  FROM orders  GROUP BY user\_id  ORDER BY order\_count DESC  LIMIT 10; |
| *20) How many products belong to each aisle and department?*  SELECT aisle\_id, department\_id, COUNT(\*) AS product\_count  FROM products  GROUP BY aisle\_id, department\_id; |

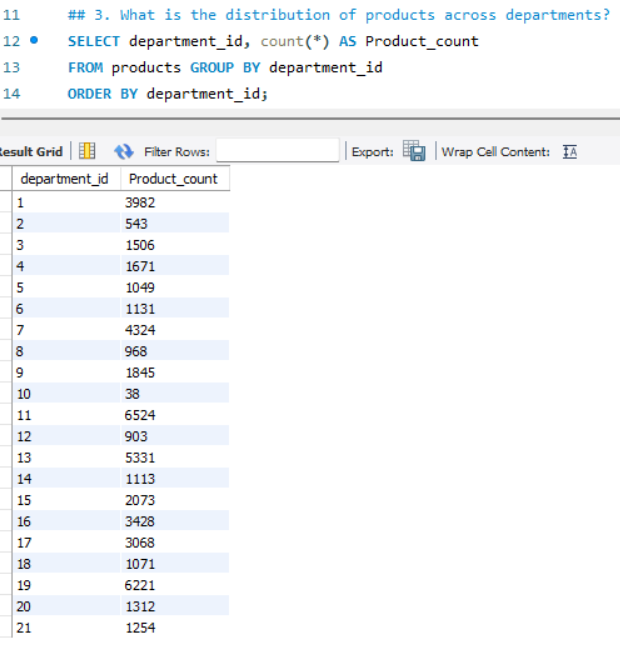
**MySQL Execution with Explanation:**

****

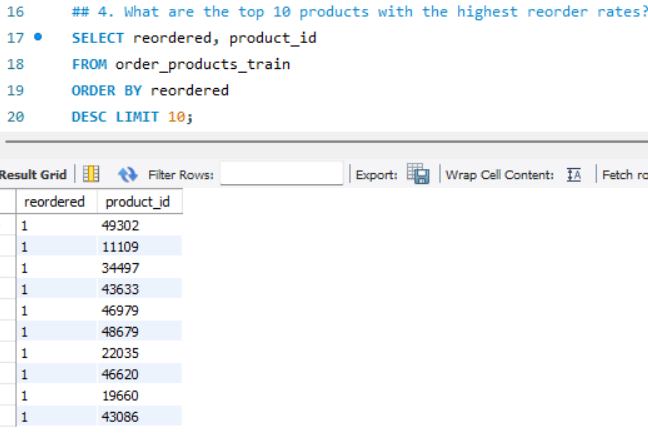
**Statement Explanation:** The above SELECT Statement retrieves count products (COUNT(\*)) grouped by aisle\_id. Sort by product count (ORDER BY Product\_count DESC) and limit to the top 10 aisles (LIMIT 10) as shown in the result.

****

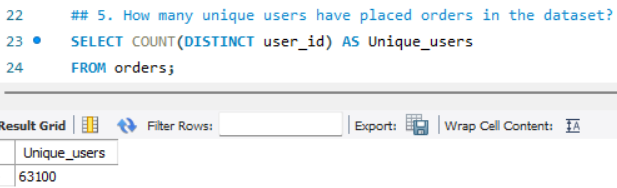
**Statement Explanation:** The above SELECT Statement retrieves COUNT(DISTINCT department\_id) to count unique department IDs in the departments table as shown in the result.

****

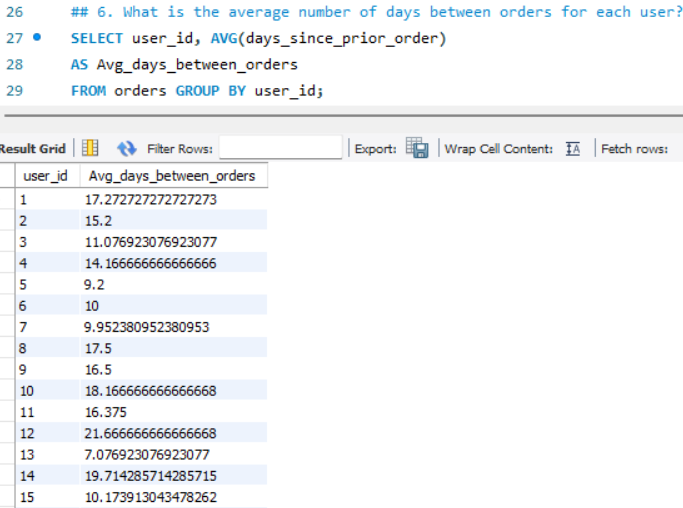
**Statement Explanation:** The above SELECT Statement retrieves count products grouped by department\_id. Sort by department\_id as shown in the result.

****

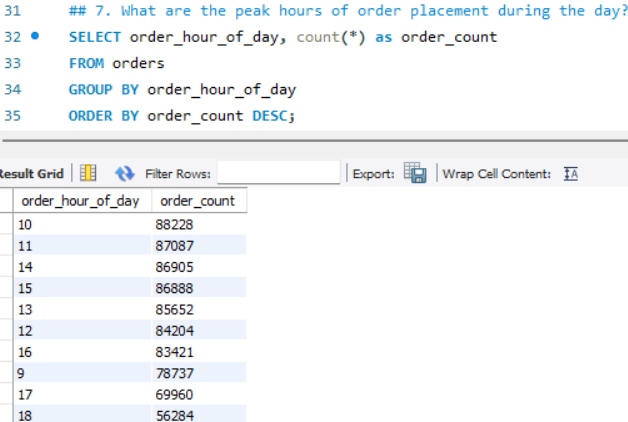
**Statement Explanation:** The above SELECT Statement retrieves calculated average reorder rate (AVG(reordered)) for each product\_id. Sort by reorder rate and take the top 10 as shown in the result.

****

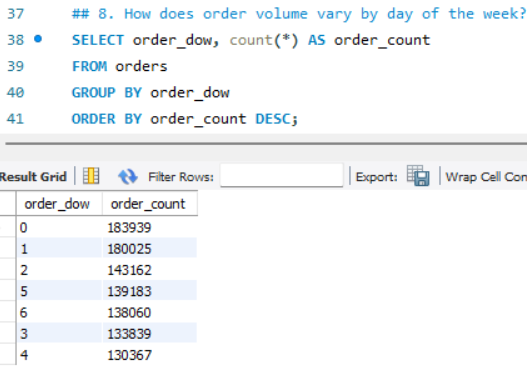
**Statement Explanation:** The above SELECT Statement retrieves COUNT(DISTINCT user\_id) to count unique users in the orders table as shown in the result.

****

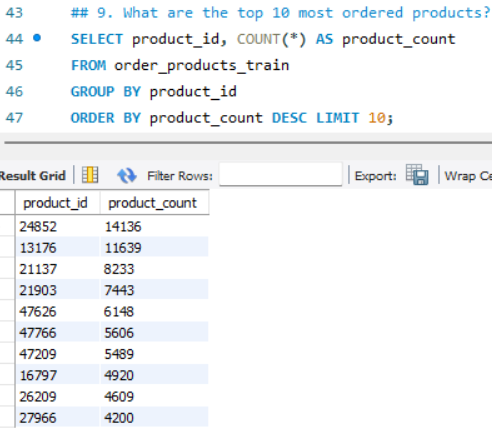
**Statement Explanation:** The above SELECT Statement retrieves group by user\_id and compute the average (AVG(days\_since\_prior\_order)) as shown in the result.

****

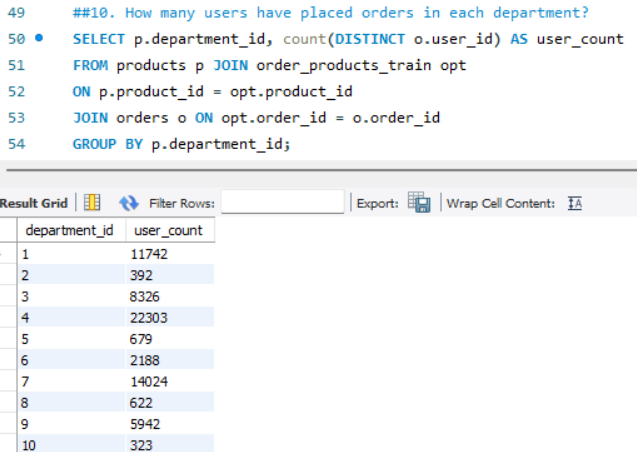
**Statement Explanation:** The above SELECT Statement retrieves count orders for each hour (order\_hour\_of\_day). Sort by order count in descending order as shown in the result.

****

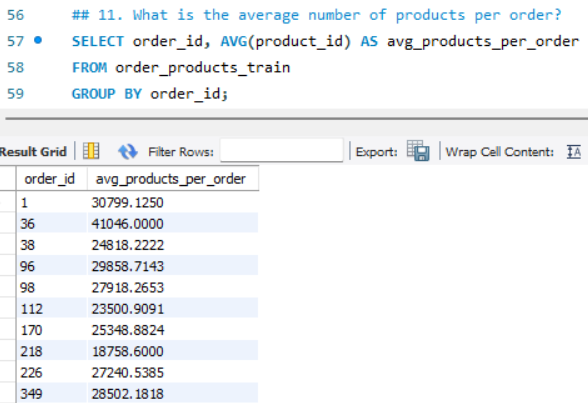
**Statement Explanation:** The above SELECT Statement retrieves count orders for each order\_dow. Sort by order count in descending order as shown in the result.

****

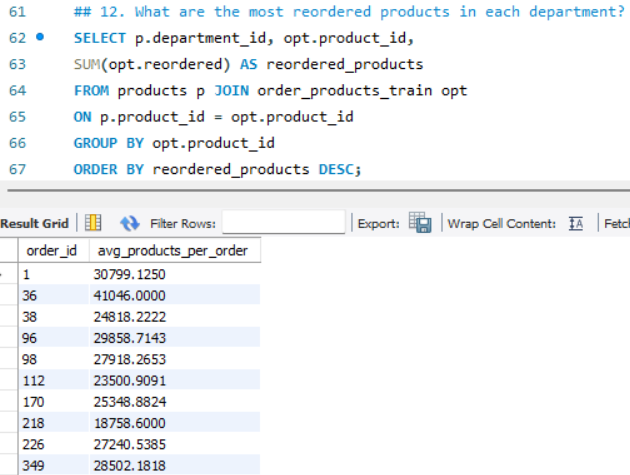
**Statement Explanation:** The above SELECT Statement retrieves count orders for each product\_id. Sort by order count and take the top 10 as shown in the result.

****

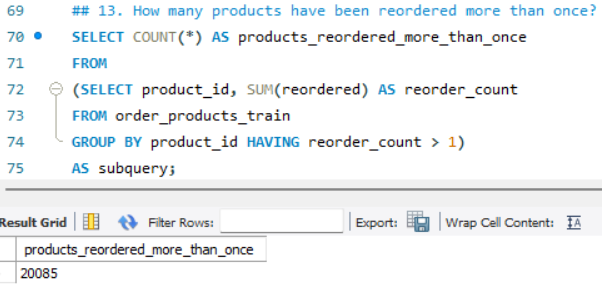
**Statement Explanation:** The above SELECT Statement retrieves join of products from order\_products\_train, and orders tables. Group by department\_id and count unique user\_id as shown in the result.

****

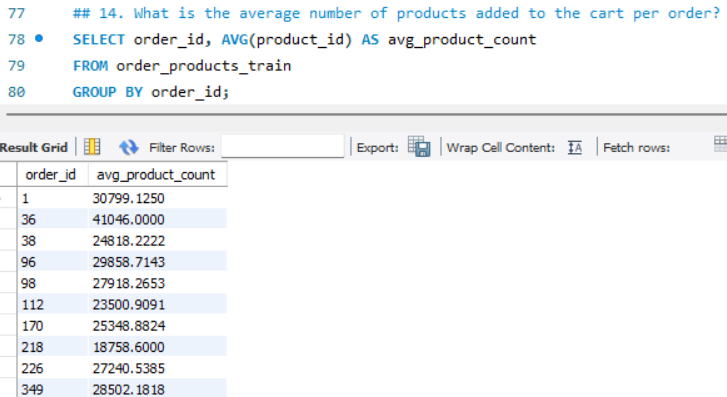
**Statement Explanation:** The above SELECT Statement retrieves group by order\_id and compute the average number of products as shown in the result.

****

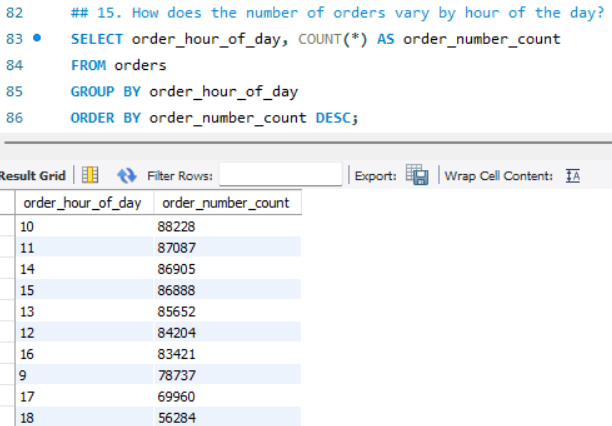
**Statement Explanation:** The above SELECT Statement retrieves join of products and order\_products\_train, group by department\_id and product\_id, and sum reorders (SUM(opt.reordered)) as shown in the result.

****

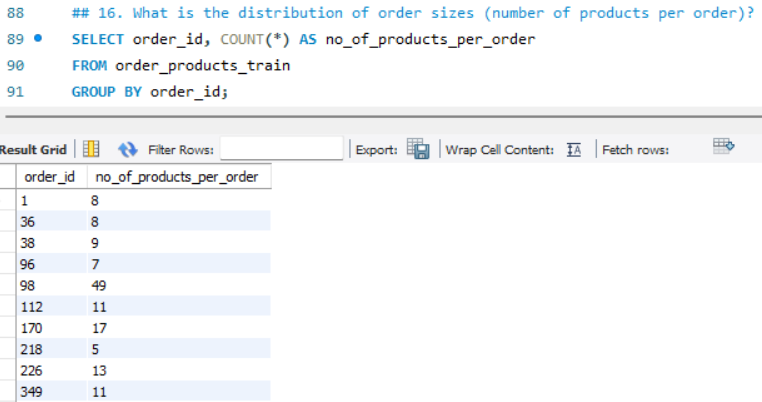
**Statement Explanation:** The above SELECT Statement retrieves filtered products with SUM(reordered) > 1 and count them as shown in the result.

****

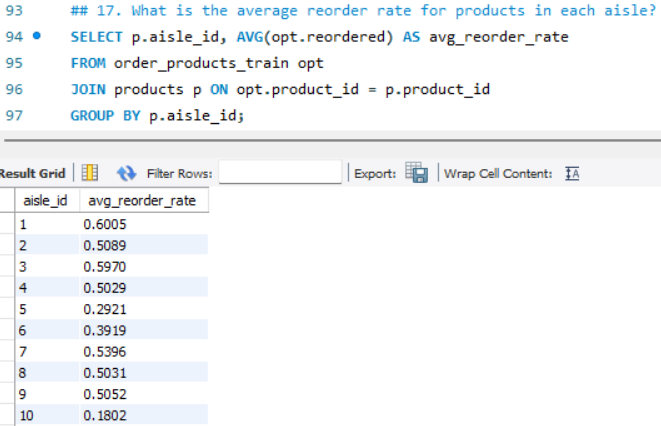
**Statement Explanation:** The above SELECT Statement retrieves group by order\_id and compute the average number of products as shown in the result.

****

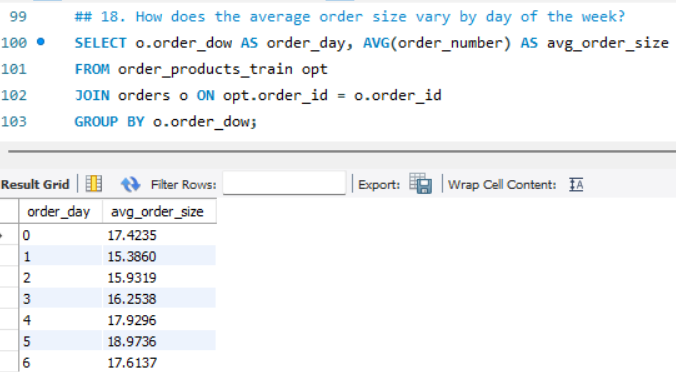
**Statement Explanation:** The above SELECT Statement retrieves count orders grouped by order\_hour\_of\_day as shown in the result.

****

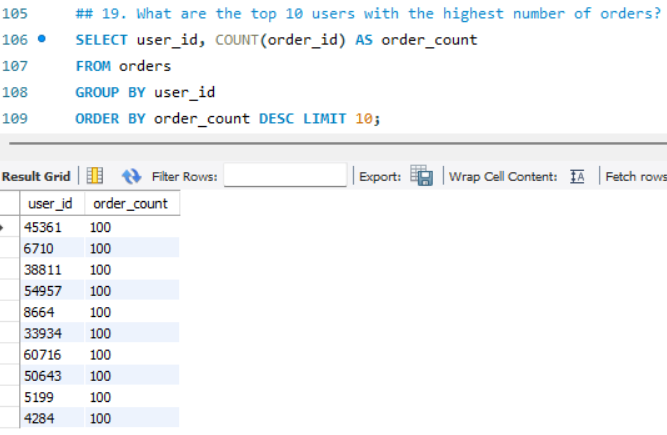
**Statement Explanation:** The above SELECT Statement retrieves count products grouped by order\_id as shown in the result.

****

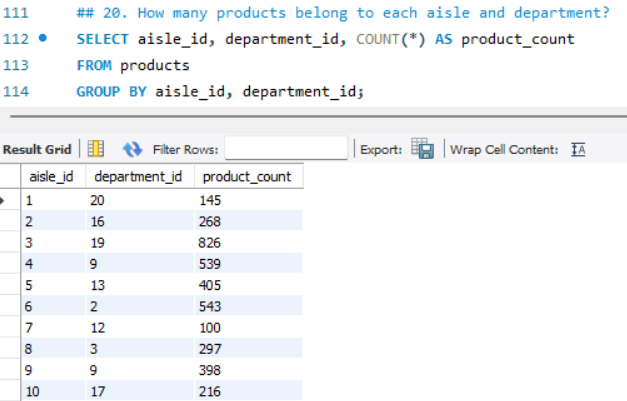
**Statement Explanation:** The above SELECT Statement retrieves join of products and order\_products\_train. Group by aisle\_id and compute AVG(reordered) as shown in the result.

****

**Statement Explanation:** The above SELECT Statement retrieves join of orders and order\_products\_train, group by order\_dow, and compute the average order size as shown in the result.

****

**Statement Explanation:** The above SELECT Statement retrieves group by user\_id and count orders. Sort and take the top 10 as shown in the result.

****

**Statement Explanation:** The above SELECT Statement retrieves group by aisle\_id and department\_id and count products as shown in the result.