**STEP 1: Create Database and Table**

**1. Create Database**

CREATE DATABASE ecommerce\_analysis;



USE ecommerce\_analysis;



**2. Create Table (ecommerce\_data)**

CREATE TABLE ecommerce\_data (

Order\_Date DATE,

Time TIME,

Aging INT,

Customer\_Id INT,

Gender VARCHAR(10),

Device\_Type VARCHAR(20),

Customer\_Login\_type VARCHAR(20),

Product\_Category VARCHAR(100),

Product VARCHAR(100),

Sales FLOAT,

Quantity FLOAT,

Discount FLOAT,

Profit FLOAT,

Shipping\_Cost FLOAT,

Order\_Priority VARCHAR(20),

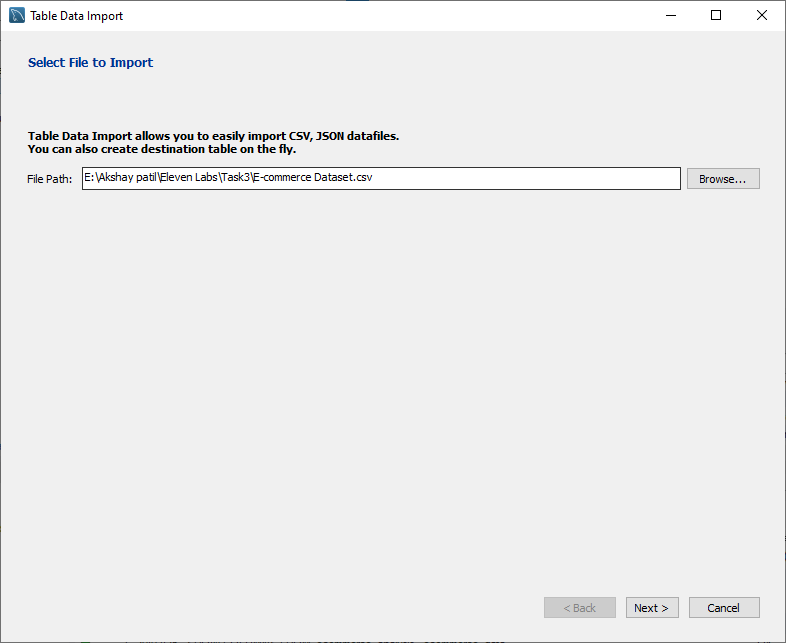
Payment\_method VARCHAR(20)

);

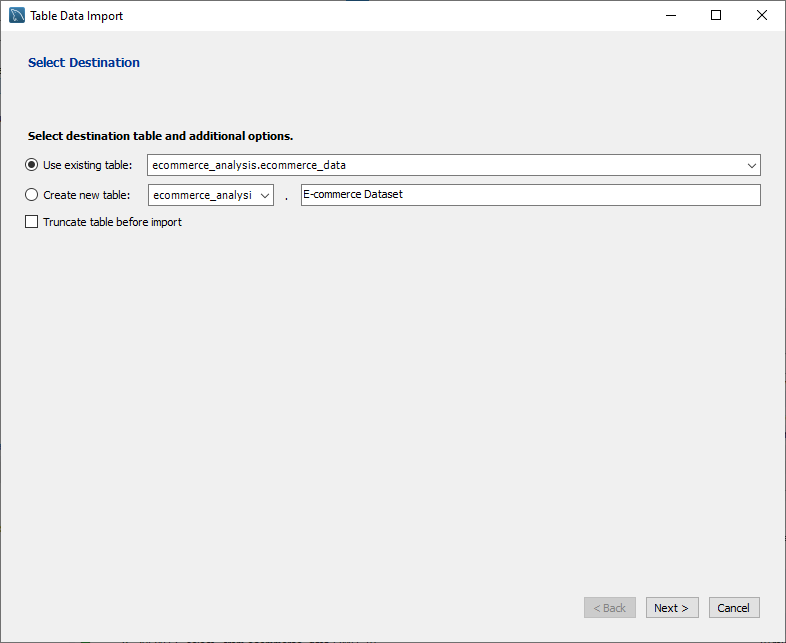


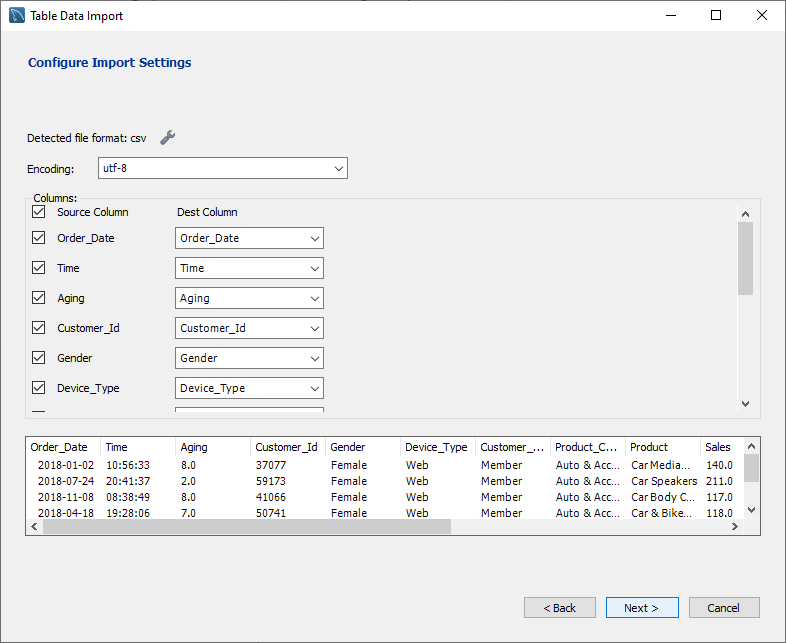
**3. Import CSV**

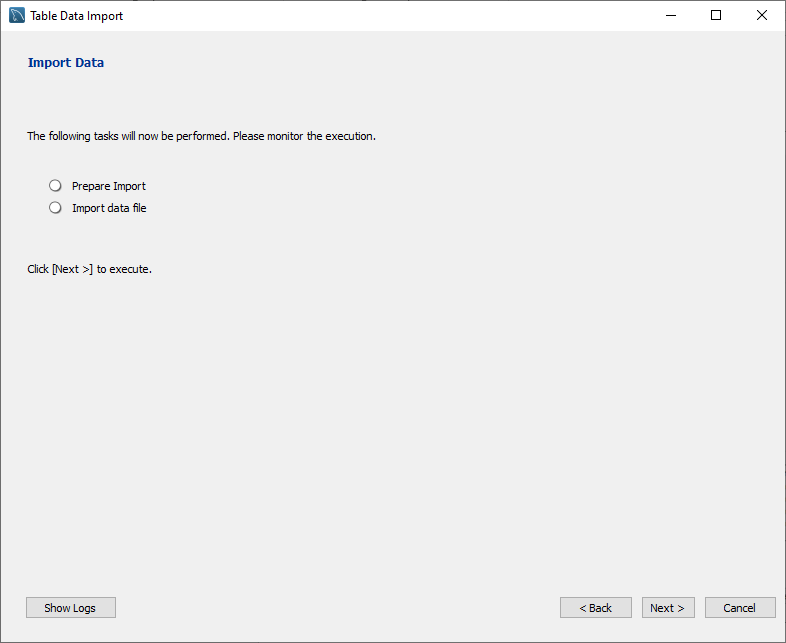
Right click on Tables > Table Data import wizard > Choose dataset and click on Next

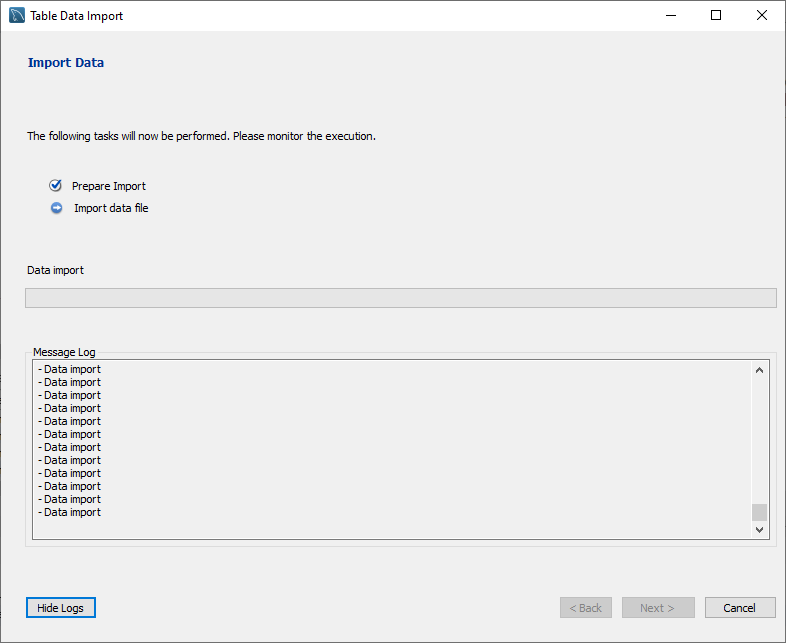


Use existing table







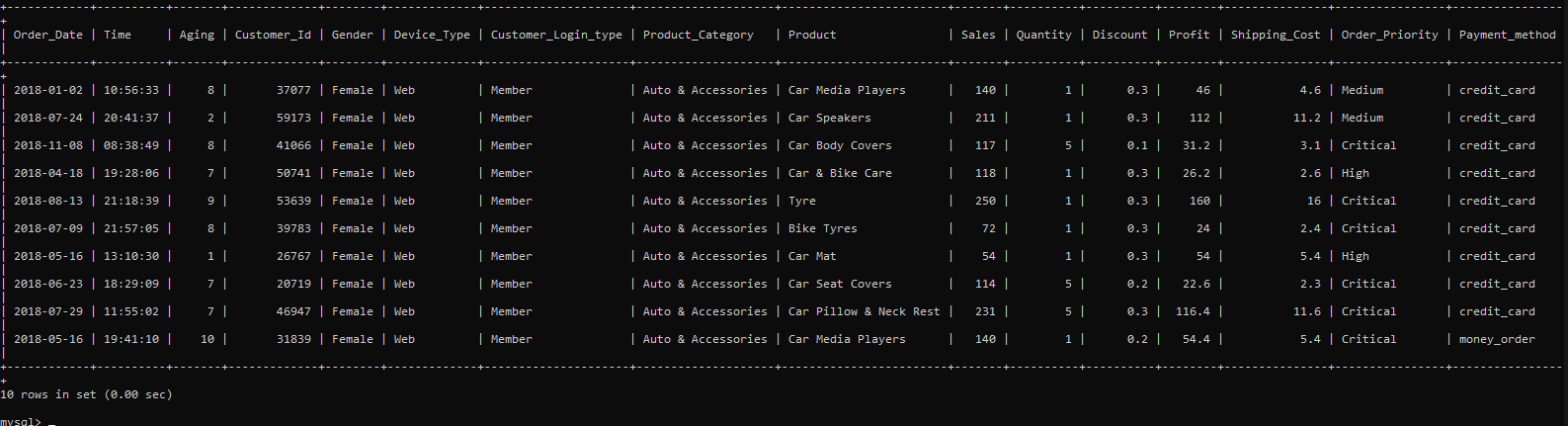


**Basic Queries**

**1. View first 10 rows**

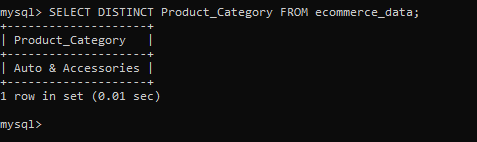
SELECT \* FROM ecommerce\_data

LIMIT 10;

****

**2. Columns overview: Show distinct product categories**

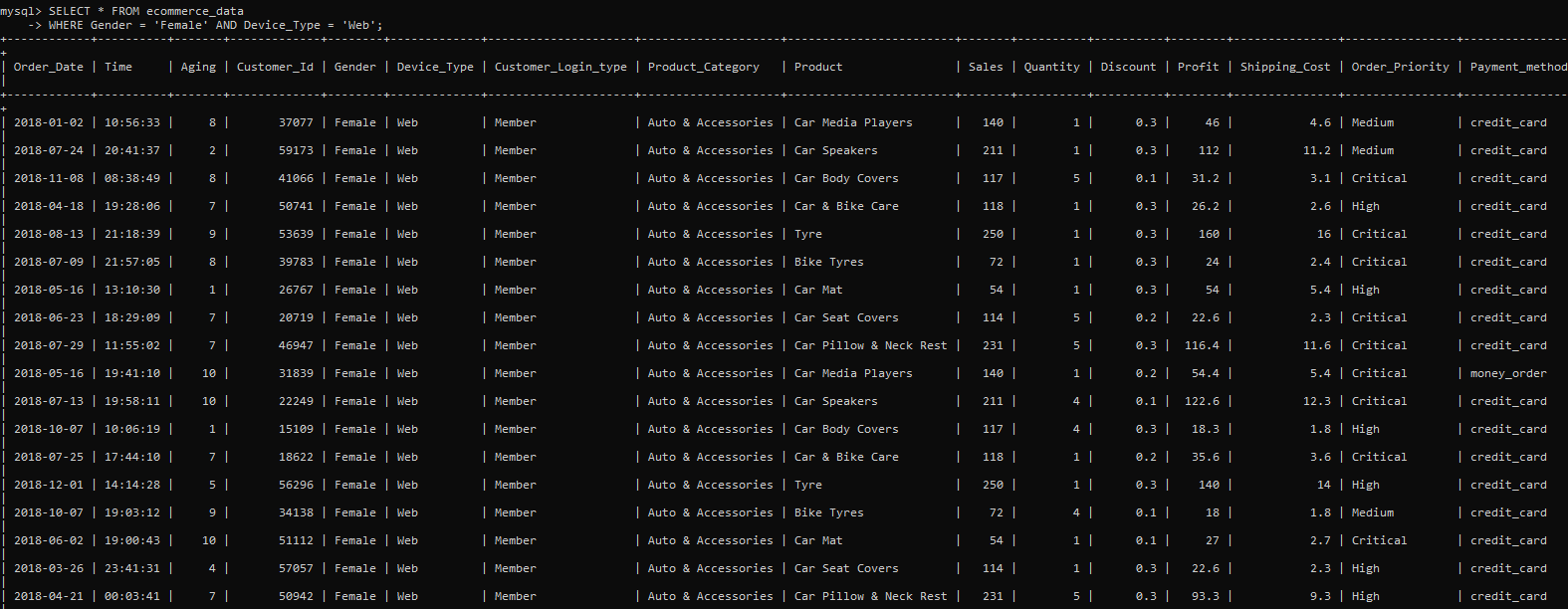
SELECT DISTINCT Product\_Category FROM ecommerce\_data;



**3. Orders from 'Female' customers using Web**

SELECT \* FROM ecommerce\_data

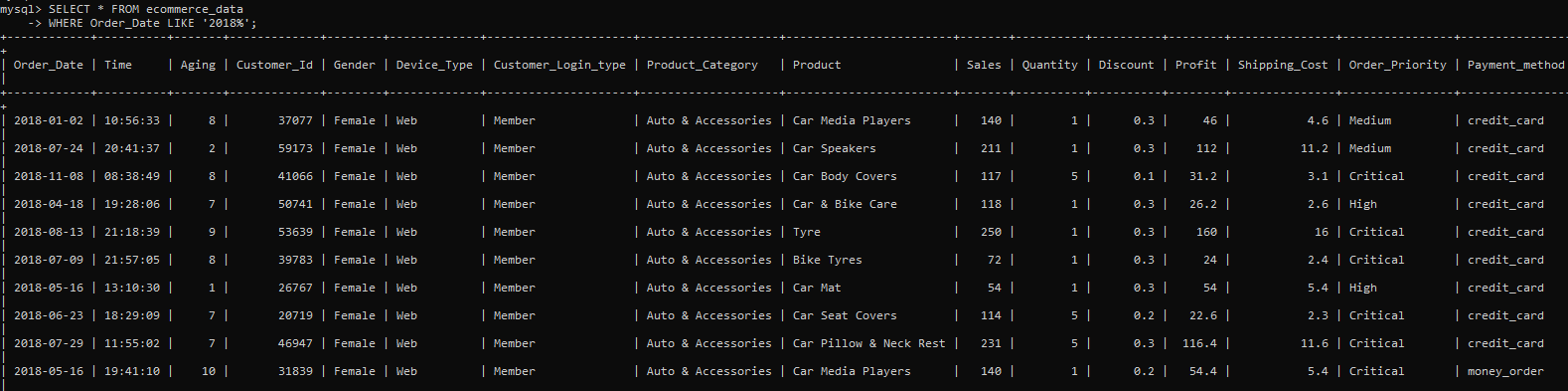
WHERE Gender = 'Female' AND Device\_Type = 'Web';



**4. Orders placed in 2018**

SELECT \* FROM ecommerce\_data

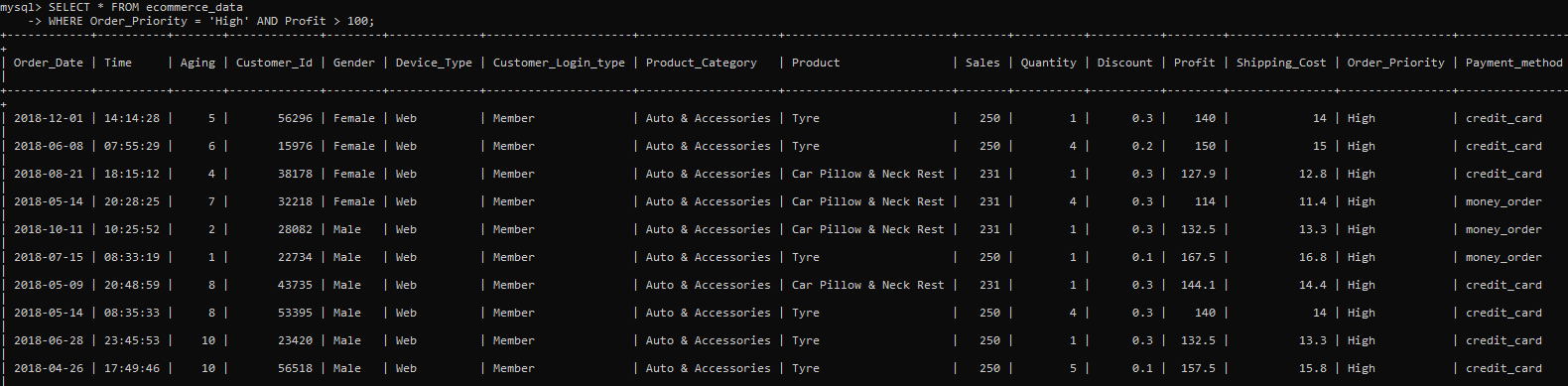
WHERE Order\_Date LIKE '2018%';

****

**5. Orders with high priority and profit > 100**

SELECT \* FROM ecommerce\_data

WHERE Order\_Priority = 'High' AND Profit > 100;

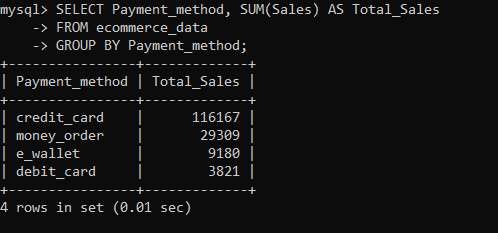
****

**6. Group sales by payment method**

SELECT Payment\_method, SUM(Sales) AS Total\_Sales

FROM ecommerce\_data

GROUP BY Payment\_method;

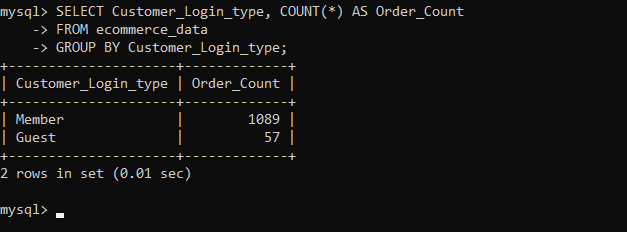


**7. Count orders per login type**

SELECT Customer\_Login\_type, COUNT(\*) AS Order\_Count

FROM ecommerce\_data

GROUP BY Customer\_Login\_type;

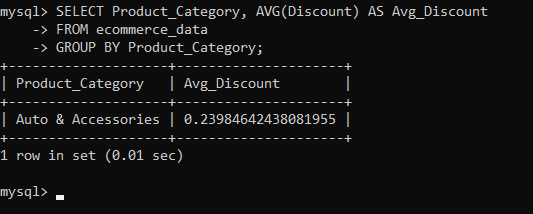


**8. Average discount by product category**

SELECT Product\_Category, AVG(Discount) AS Avg\_Discount

FROM ecommerce\_data

GROUP BY Product\_Category;

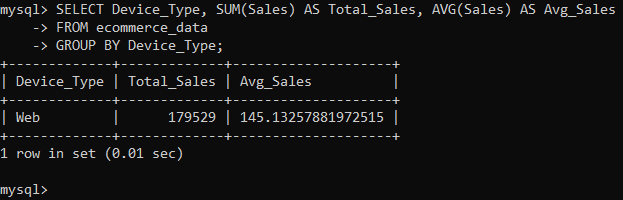


**9. Total and average sales by device**

SELECT Device\_Type, SUM(Sales) AS Total\_Sales, AVG(Sales) AS Avg\_Sales

FROM ecommerce\_data

GROUP BY Device\_Type;

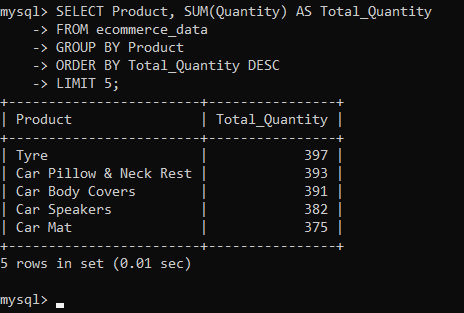


**10. Quantity sold per product (Top 5)**

SELECT Product, SUM(Quantity) AS Total\_Quantity

FROM ecommerce\_data GROUP BY Product

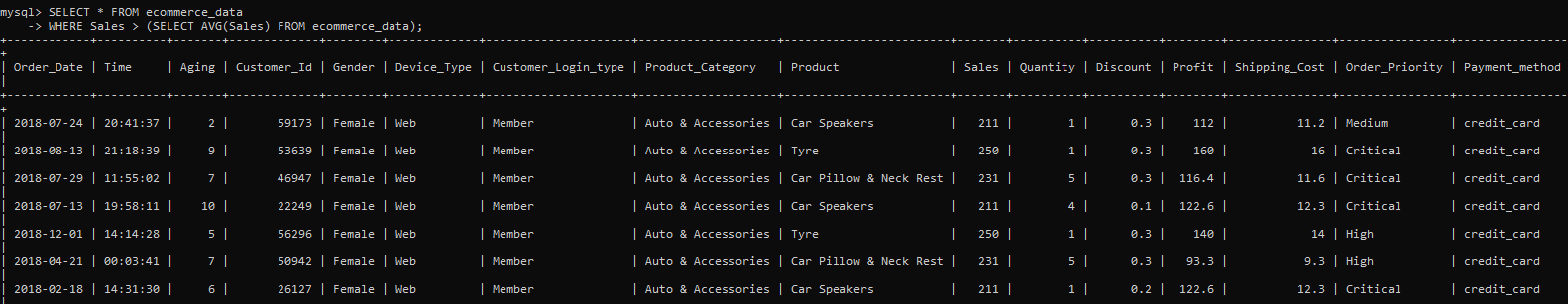
ORDER BY Total\_Quantity DESC LIMIT 5;



**11. Subquery: Orders where sales > average sales**

SELECT \* FROM ecommerce\_data

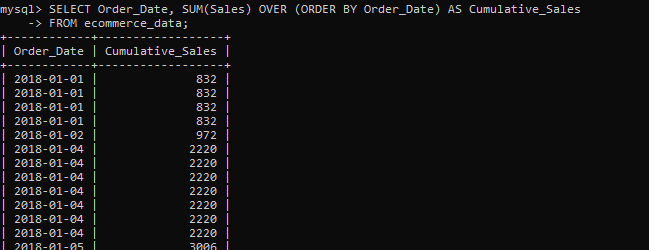
WHERE Sales > (SELECT AVG(Sales) FROM ecommerce\_data);

****

**12. Cumulative sales by date (Window function for PostgreSQL)**

SELECT Order\_Date, SUM(Sales) OVER (ORDER BY Order\_Date) AS Cumulative\_Sales

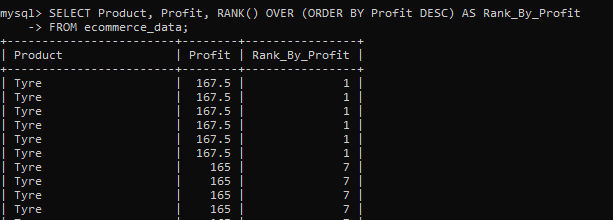
FROM ecommerce\_data;

****

**13. RANK products by profit**

SELECT Product, Profit, RANK() OVER (ORDER BY Profit DESC) AS Rank\_By\_Profit

FROM ecommerce\_data;

****

**14. Average Revenue Per User (ARPU)**

SELECT AVG(user\_sales) AS ARPU

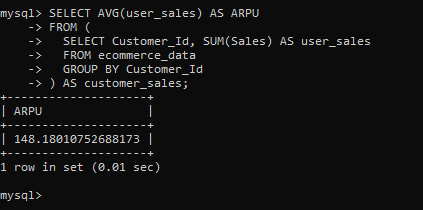
FROM (

SELECT Customer\_Id, SUM(Sales) AS user\_sales

FROM ecommerce\_data

GROUP BY Customer\_Id

) AS customer\_sales;



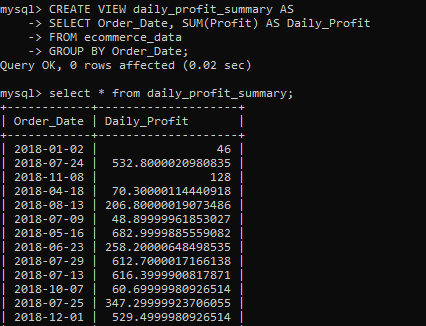
**15. Create a view: Daily profit summary**

CREATE VIEW daily\_profit\_summary AS

SELECT Order\_Date, SUM(Profit) AS Daily\_Profit

FROM ecommerce\_data

GROUP BY Order\_Date;



**16. Orders with NULL shipping cost**

SELECT \* FROM ecommerce\_data

WHERE Shipping\_Cost IS NULL;

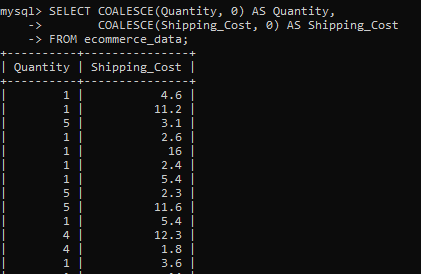


**17. Replace NULL with default values**

SELECT COALESCE(Quantity, 0) AS Quantity,

COALESCE(Shipping\_Cost, 0) AS Shipping\_Cost

FROM ecommerce\_data;

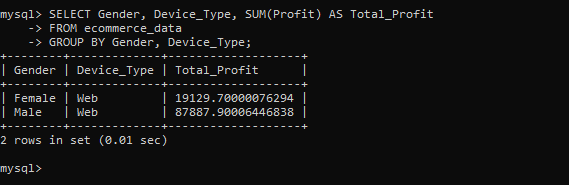


**19. Total profit by gender and device type**

SELECT Gender, Device\_Type, SUM(Profit) AS Total\_Profit

FROM ecommerce\_data

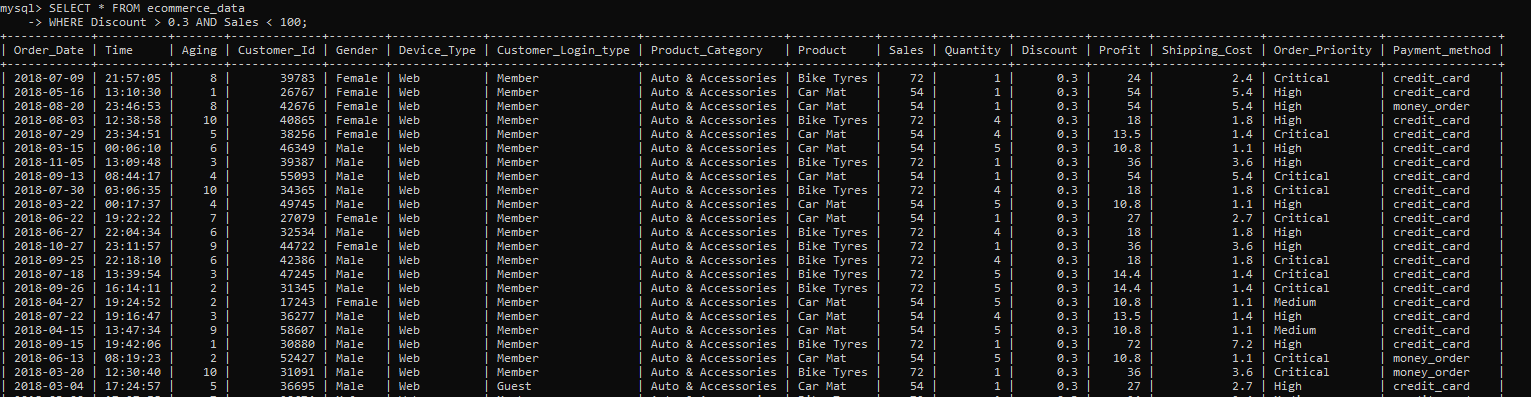
GROUP BY Gender, Device\_Type;



**20. High discount but low sales**

SELECT \* FROM ecommerce\_data

WHERE Discount > 0.3 AND Sales < 100;

****