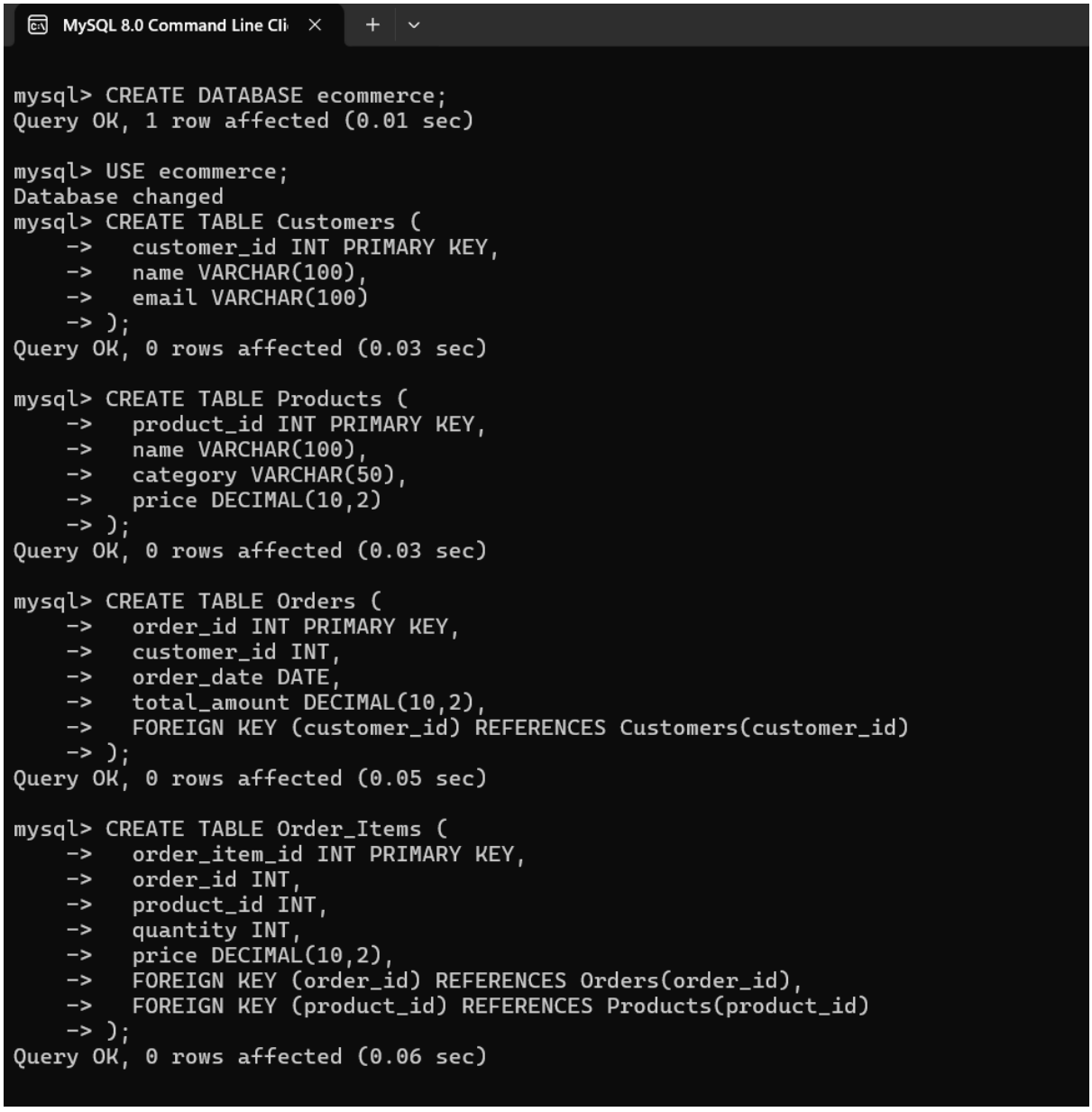
SQL Data Analysis using MySQL

Sql code:

```
CREATE DATABASE ecommerce;
USE ecommerce;
CREATE TABLE Customers (
  customer_id INT PRIMARY KEY,
  name VARCHAR(100),
  email VARCHAR(100)
);

CREATE TABLE Products (
  product_id INT PRIMARY KEY,
  name VARCHAR(100),
  category VARCHAR(50),
  price DECIMAL(10,2)
);
CREATE TABLE Orders (
  order_id INT PRIMARY KEY,
  customer_id INT,
  order_date DATE,
  total_amount DECIMAL(10,2),
  FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)
);

CREATE TABLE Order_Items (
  order_item_id INT PRIMARY KEY,
  order_id INT,
  product_id INT,
  quantity INT,
```

```
  price DECIMAL(10,2),
  FOREIGN KEY (order_id) REFERENCES Orders(order_id),
  FOREIGN KEY (product_id) REFERENCES Products(product_id)
);
```



insert data manually:

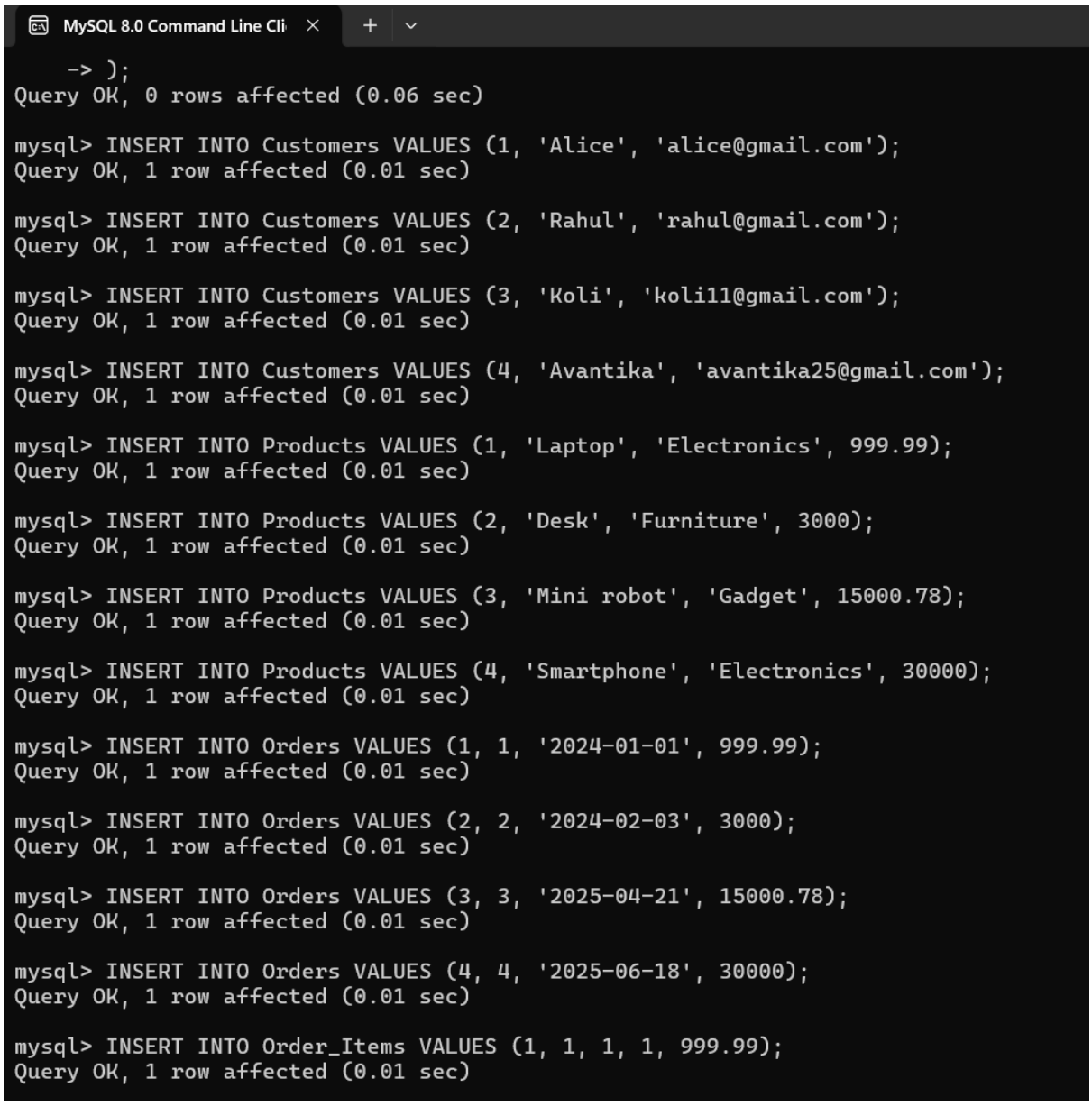
sql code:

INSERT INTO Customers VALUES (1, 'Alice', 'alice@gmail.com'); ..(like this more)

INSERT INTO Products VALUES (1, 'Laptop', 'Electronics', 999.99); ..(like this more)

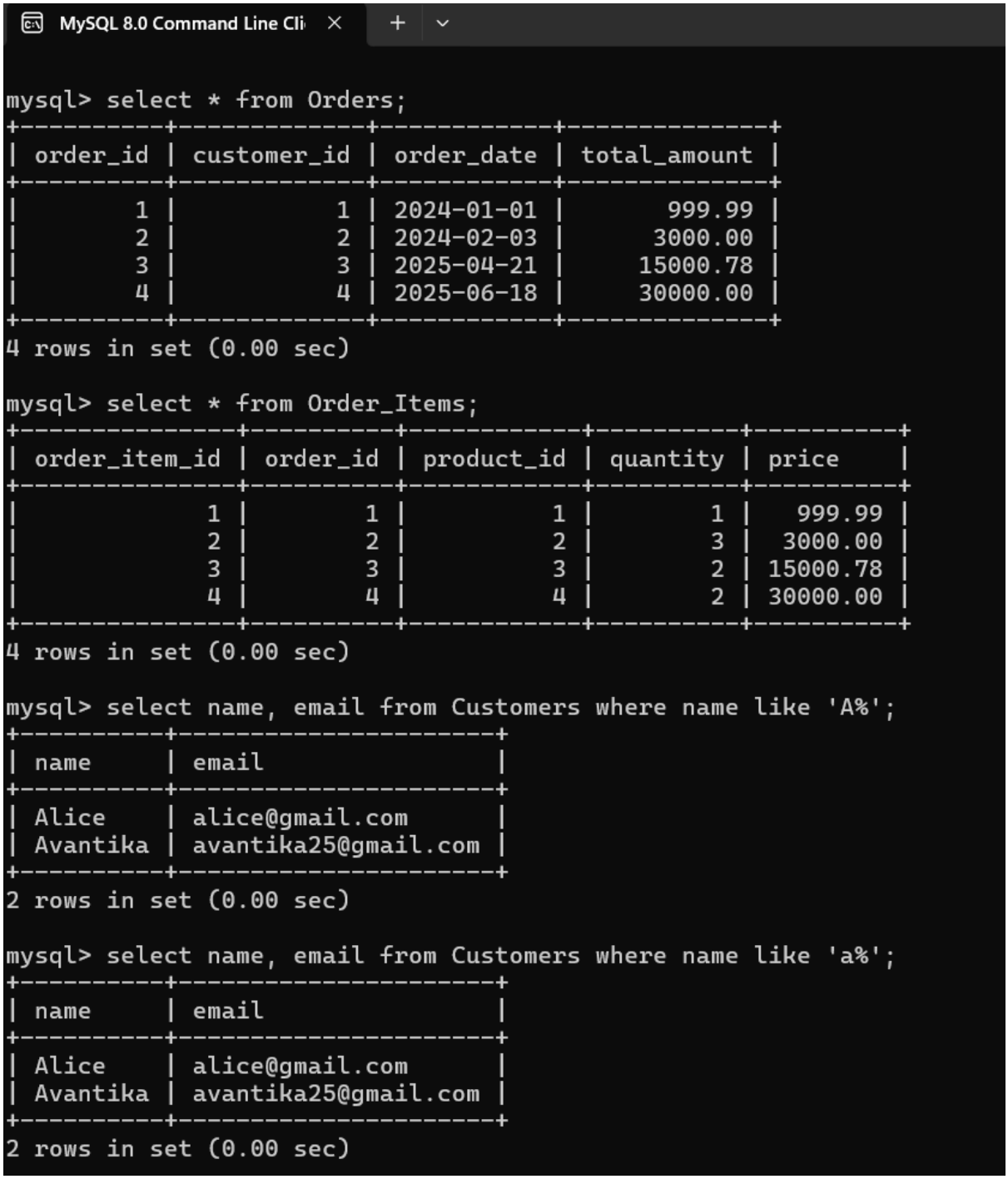
INSERT INTO Orders VALUES (1, 1, '2024-01-01', 999.99); ..(like this more)

INSERT INTO Order_Items VALUES (1, 1, 1, 1, 999.99); ..(like this more)



MySQL 8.0 Command Line Cli

```
mysql> INSERT INTO Order_Items VALUES (2, 2, 2, 3, 3000);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Order_Items VALUES (3, 3, 3, 2, 15000.78);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Order_Items VALUES (4, 4, 4, 2, 30000);
Query OK, 1 row affected (0.01 sec)

mysql> select * from Customers;
+-------------+----------+----------------------+
| customer_id | name     | email                |
+-------------+----------+----------------------+
|           1 | Alice    | alice@gmail.com      |
|           2 | Rahul    | rahul@gmail.com      |
|           3 | Koli     | koli11@gmail.com     |
|           4 | Avantika | avantika25@gmail.com |
+-------------+----------+----------------------+
4 rows in set (0.00 sec)

mysql> select * from Products;
+------------+------------+-------------+----------+
| product_id | name       | category    | price    |
+------------+------------+-------------+----------+
|          1 | Laptop     | Electronics |   999.99 |
|          2 | Desk       | Furniture   |  3000.00 |
|          3 | Mini robot | Gadget      | 15000.78 |
|          4 | Smartphone | Electronics | 30000.00 |
+------------+------------+-------------+----------+
4 rows in set (0.00 sec)

mysql> select * from Orders;
+----------+-------------+------------+--------------+
| order_id | customer_id | order_date | total_amount |
+----------+-------------+------------+--------------+
|        1 |           1 | 2024-01-01 |       999.99 |
|        2 |           2 | 2024-02-03 |      3000.00 |
|        3 |           3 | 2025-04-21 |     15000.78 |
|        4 |           4 | 2025-06-18 |     30000.00 |
+----------+-------------+------------+--------------+
```



a. Basic Query

Sql code:

SELECT name, email FROM CustomersWHERE name LIKE 'A%';

b. JOIN + GROUP BY + Aggregates

Sql code

SELECT c.name, SUM(o.total_amount) AS total_spent

FROM Customers c

```
JOIN Orders o ON c.customer_id = o.customer_id
GROUP BY c.name;
```

c. Subquery

Sql code:

```
SELECT name FROM Customers
WHERE customer_id IN (
  SELECT customer_id FROM Orders
  GROUP BY customer_id
  HAVING SUM(total_amount) > (
    SELECT AVG(total_amount) FROM Orders
  )
);
```

d. Create a View

Sql code:

```
CREATE VIEW customer_summary AS
SELECT c.name, COUNT(o.order_id) AS total_orders, SUM(o.total_amount) AS total_spent
FROM Customers c
JOIN Orders o ON c.customer_id = o.customer_id
GROUP BY c.name;
```

e. Optimize with Indexes

sql code:

```
CREATE INDEX idx_customer_id ON Orders(customer_id);
CREATE INDEX idx_order_date ON Orders(order_date);
```

MySQL 8.0 Command Line Cli

```
mysql> INSERT INTO Order_Items VALUES (2, 2, 2, 3, 3000);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Order_Items VALUES (3, 3, 3, 2, 15000.78);
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Order_Items VALUES (4, 4, 4, 2, 30000);
Query OK, 1 row affected (0.01 sec)

mysql> select * from Customers;
+-------------+----------+----------------------+
| customer_id | name     | email                |
+-------------+----------+----------------------+
|           1 | Alice    | alice@gmail.com      |
|           2 | Rahul    | rahul@gmail.com      |
|           3 | Koli     | koli11@gmail.com     |
|           4 | Avantika | avantika25@gmail.com |
+-------------+----------+----------------------+
4 rows in set (0.00 sec)

mysql> select * from Products;
+------------+------------+-------------+----------+
| product_id | name       | category    | price    |
+------------+------------+-------------+----------+
|          1 | Laptop     | Electronics |   999.99 |
|          2 | Desk       | Furniture   |  3000.00 |
|          3 | Mini robot | Gadget      | 15000.78 |
|          4 | Smartphone | Electronics | 30000.00 |
+------------+------------+-------------+----------+
4 rows in set (0.00 sec)

mysql> select * from Orders;
+----------+-------------+------------+--------------+
| order_id | customer_id | order_date | total_amount |
+----------+-------------+------------+--------------+
|        1 |           1 | 2024-01-01 |       999.99 |
|        2 |           2 | 2024-02-03 |      3000.00 |
|        3 |           3 | 2025-04-21 |     15000.78 |
|        4 |           4 | 2025-06-18 |     30000.00 |
+----------+-------------+------------+--------------+
```

MySQL 8.0 Command Line Cli

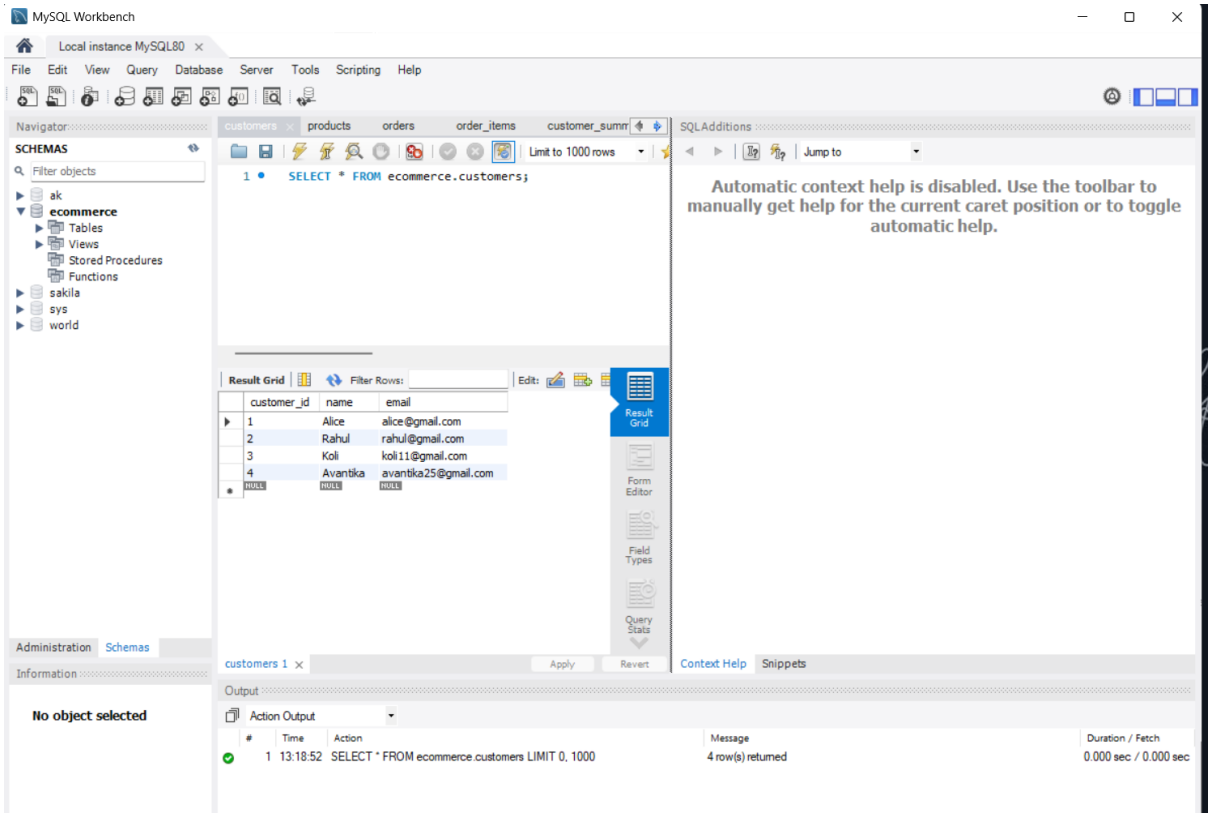
```
mysql> select * from Orders;
+----------+-------------+------------+--------------+
| order_id | customer_id | order_date | total_amount |
+----------+-------------+------------+--------------+
|        1 |           1 | 2024-01-01 |       999.99 |
|        2 |           2 | 2024-02-03 |      3000.00 |
|        3 |           3 | 2025-04-21 |     15000.78 |
|        4 |           4 | 2025-06-18 |     30000.00 |
+----------+-------------+------------+--------------+
4 rows in set (0.00 sec)

mysql> select * from Order_Items;
+---------------+----------+------------+----------+----------+
| order_item_id | order_id | product_id | quantity | price    |
+---------------+----------+------------+----------+----------+
|             1 |        1 |          1 |        1 |   999.99 |
|             2 |        2 |          2 |        3 |  3000.00 |
|             3 |        3 |          3 |        2 | 15000.78 |
|             4 |        4 |          4 |        2 | 30000.00 |
+---------------+----------+------------+----------+----------+
4 rows in set (0.00 sec)

mysql> select name, email from Customers where name like 'A%';
+----------+---------------------+
| name     | email               |
+----------+---------------------+
| Alice    | alice@gmail.com     |
| Avantika | avantika25@gmail.com |
+----------+---------------------+
2 rows in set (0.00 sec)

mysql> select name, email from Customers where name like 'a%';
+----------+---------------------+
| name     | email               |
+----------+---------------------+
| Alice    | alice@gmail.com     |
| Avantika | avantika25@gmail.com |
+----------+---------------------+
2 rows in set (0.00 sec)
```

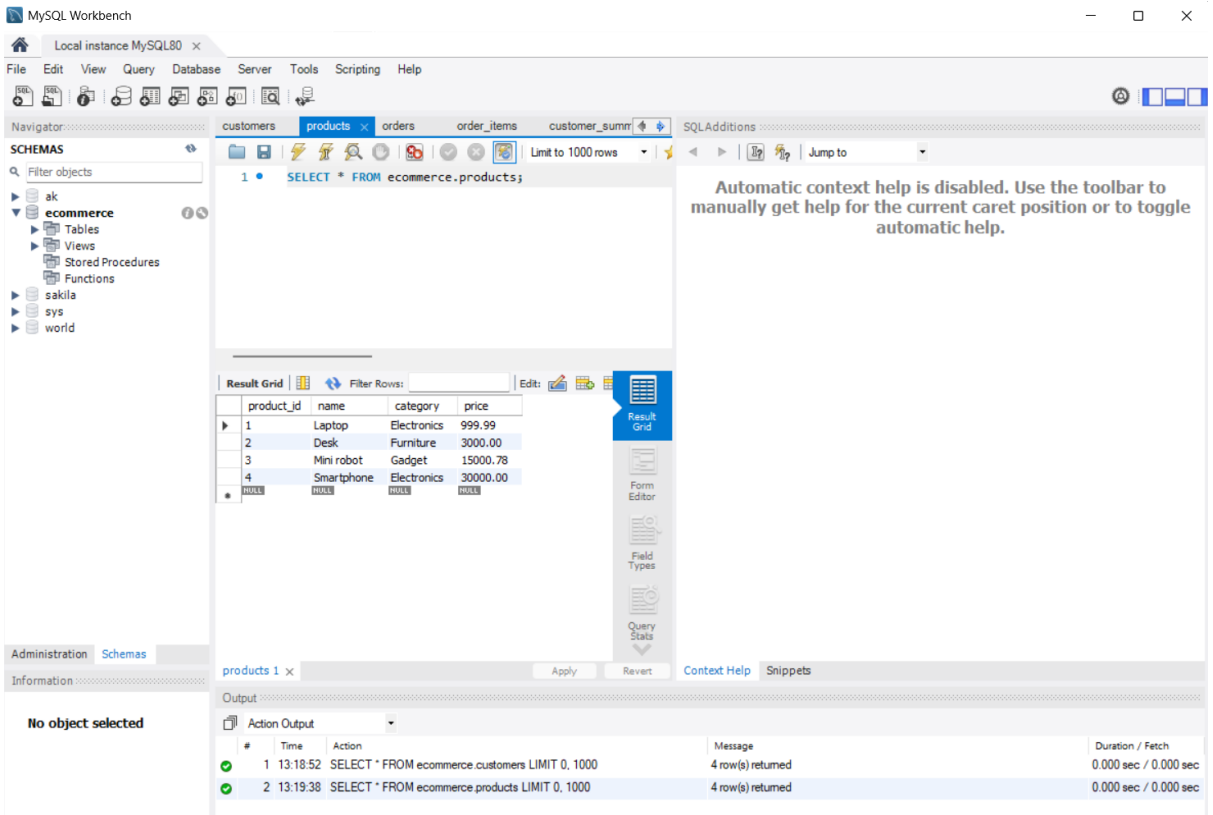
```
MySQL 8.0 Command Line Cli

mysql> select c.name, sum(o.total_amount) as total_spent from Customers c;
ERROR 1054 (42S22): Unknown column 'o.total_amount' in 'field list'
mysql> select c.name, sum(o.total_amount) as total_spent from Customers c join Orders o on c.customer_id=o.customer_id group by c.name;
+----------+-------------+
| name     | total_spent |
+----------+-------------+
| Alice    |      999.99 |
| Rahul    |     3000.00 |
| Koli     |    15000.78 |
| Avantika |    30000.00 |
+----------+-------------+
4 rows in set (0.00 sec)

mysql> select name from Customers where customer_id in ( select customer_id from Orders group by customer_id having sum(total_amount) > ( select avg(total_a
mount) from Orders) );
+----------+
| name     |
+----------+
| Koli     |
| Avantika |
+----------+
2 rows in set (0.01 sec)

mysql> create view customer_summary as select c.name, count(o.order_id_ as total_orders, sum(o.total_amount) as total_spent from Customers c join Orders o o
n c.customer_id = o.customer_id groupby c.name;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'a
s total_orders, sum(o.total_amount) as total_spent from Customers c join Orders' at line 1
mysql> create view customer_summary as select c.name, count(o.order_id) as total_orders, sum(o.total_amount) as total_spent from Customers c join Orders o o
n c.customer_id = o.customer_id group by c.name;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from customer_summary;
+----------+--------------+-------------+
| name     | total_orders | total_spent |
+----------+--------------+-------------+
| Alice    |            1 |      999.99 |
| Rahul    |            1 |     3000.00 |
| Koli     |            1 |    15000.78 |
| Avantika |            1 |    30000.00 |
+----------+--------------+-------------+
```

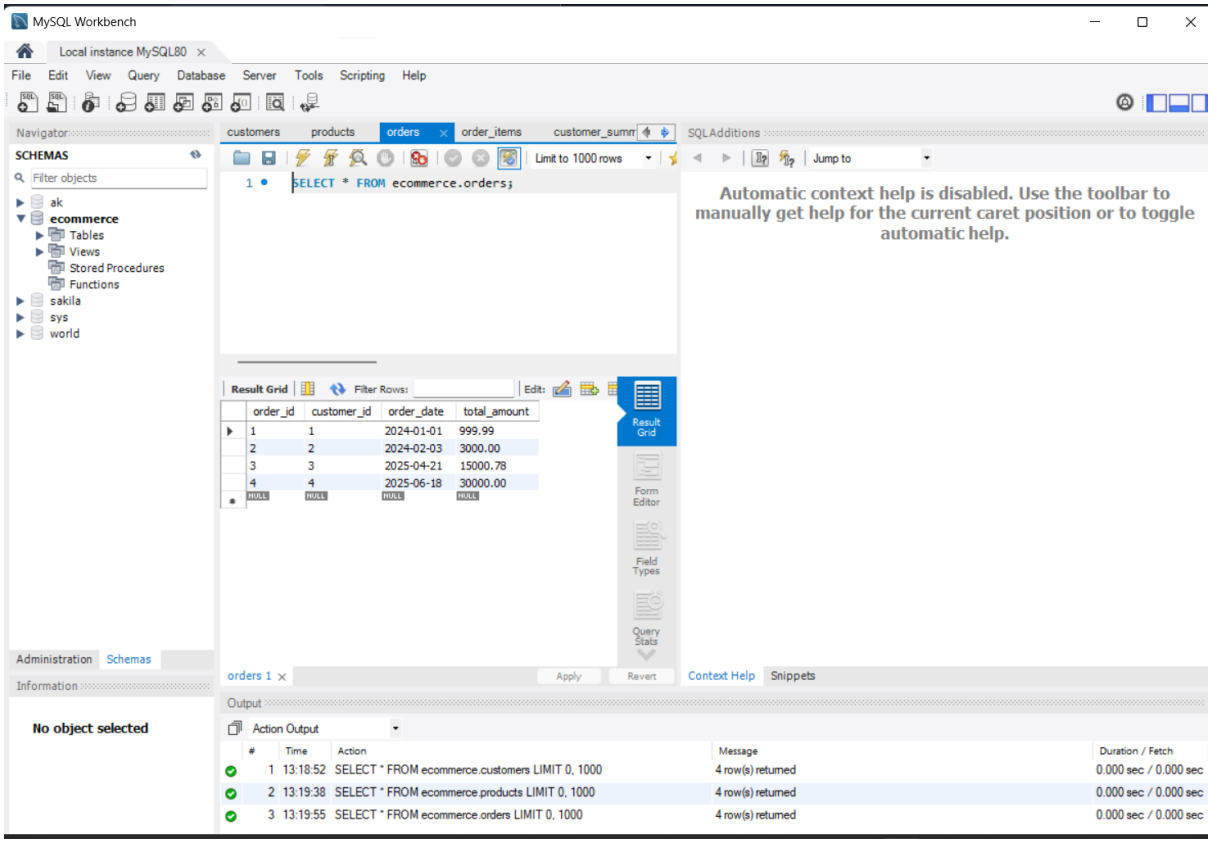
```
MySQL 8.0 Command Line Cli

mysql> CREATE INDEX idx_customer_id ON Orders(customer_id);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE INDEX idx_order_date ON Orders(order_date);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

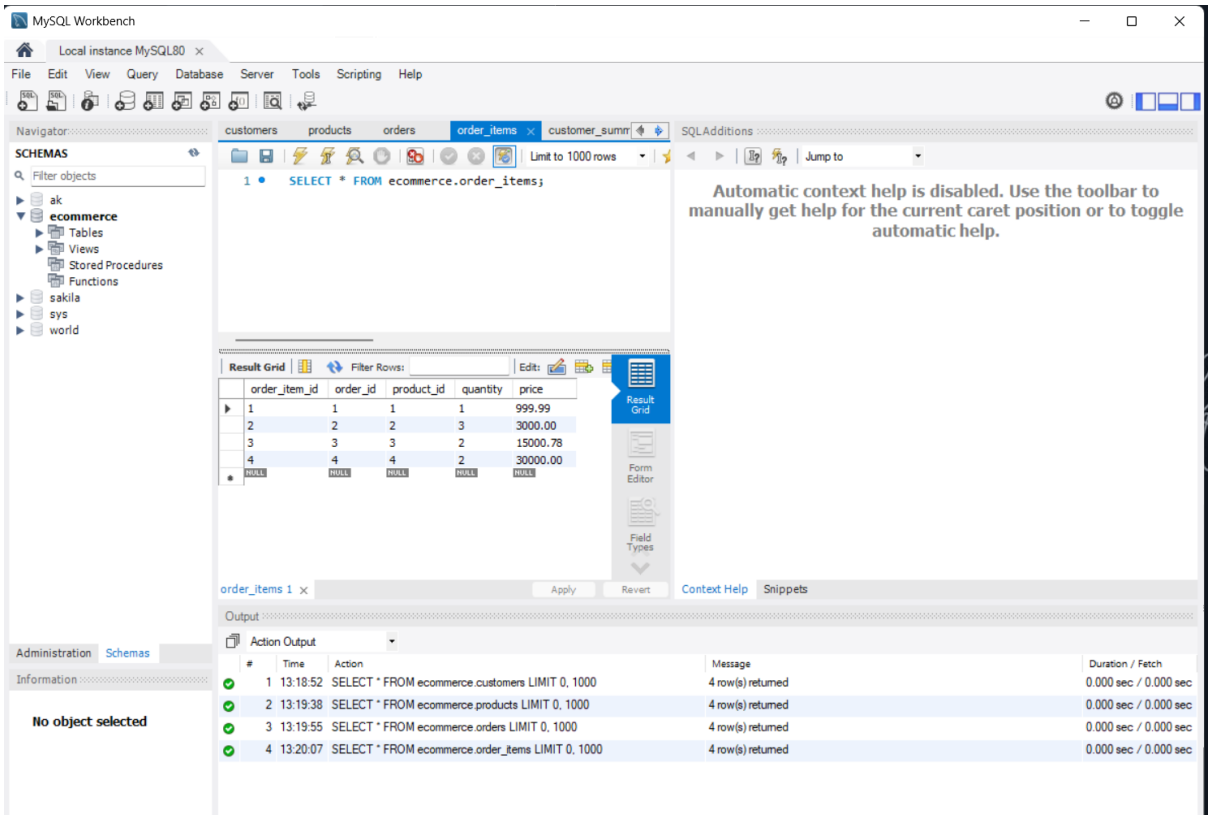
```
mysql> SHOW INDEX FROM Orders;
+--------+------------+----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+----------
-----+---------+------------+
| Table  | Non_unique | Key_name       | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_com
ment | Visible | Expression |
+--------+------------+----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+----------
-----+---------+------------+
| orders |          0 | PRIMARY        |            1 | order_id    | A         |           4 |     NULL |   NULL |      | BTREE      |         |
     | YES     | NULL       |
| orders |          1 | idx_customer_id |            1 | customer_id | A         |           4 |     NULL |   NULL | YES  | BTREE      |         |
     | YES     | NULL       |
| orders |          1 | idx_order_date  |            1 | order_date  | A         |           4 |     NULL |   NULL | YES  | BTREE      |         |
     | YES     | NULL       |
+--------+------------+----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+----------
-----+---------+------------+
```

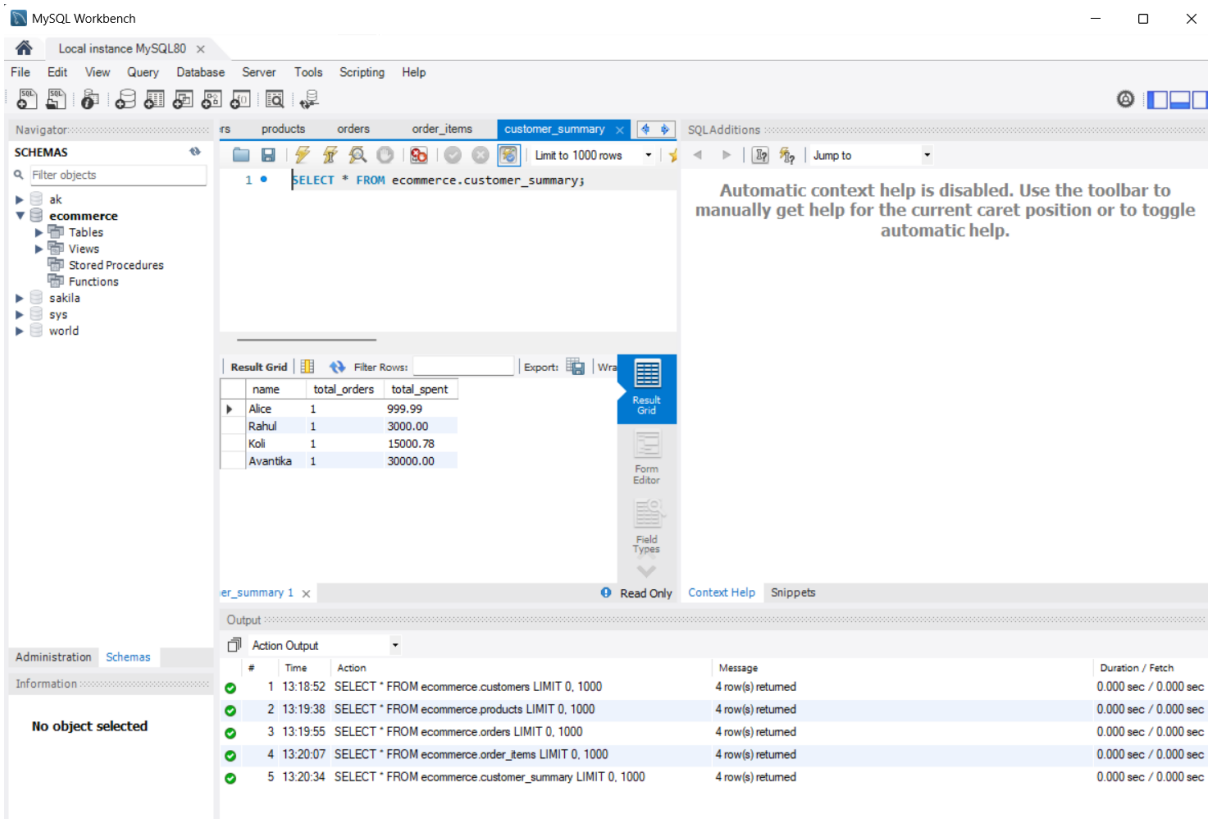
```
mysql> EXPLAIN SELECT * FROM Orders WHERE customer_id = 1;
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
| id | select_type | table  | partitions | type | possible_keys   | key             | key_len | ref   | rows | filtered | Extra |
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
|  1 | SIMPLE      | Orders | NULL       | ref  | idx_customer_id | idx_customer_id | 5       | const |    1 |   100.00 | NULL  |
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
1 row in set, 1 warning (0.00 sec)

mysql> SELECT * FROM Customers;
+-------------+----------+----------------------+
| customer_id | name     | email                |
+-------------+----------+----------------------+
|           1 | Alice    | alice@gmail.com      |
|           2 | Rahul    | rahul@gmail.com      |
|           3 | Koli     | koli11@gmail.com     |
|           4 | Avantika | avantika25@gmail.com |
+-------------+----------+----------------------+
4 rows in set (0.00 sec)

mysql> SELECT * FROM Orders WHERE total_amount > 500;
+----------+-------------+------------+--------------+
| order_id | customer_id | order_date | total_amount |
+----------+-------------+------------+--------------+
|        1 |           1 | 2024-01-01 |       999.99 |
|        2 |           2 | 2024-02-03 |      3000.00 |
|        3 |           3 | 2025-04-21 |     15000.78 |
|        4 |           4 | 2025-06-18 |     30000.00 |
+----------+-------------+------------+--------------+
4 rows in set (0.00 sec)

mysql>
```

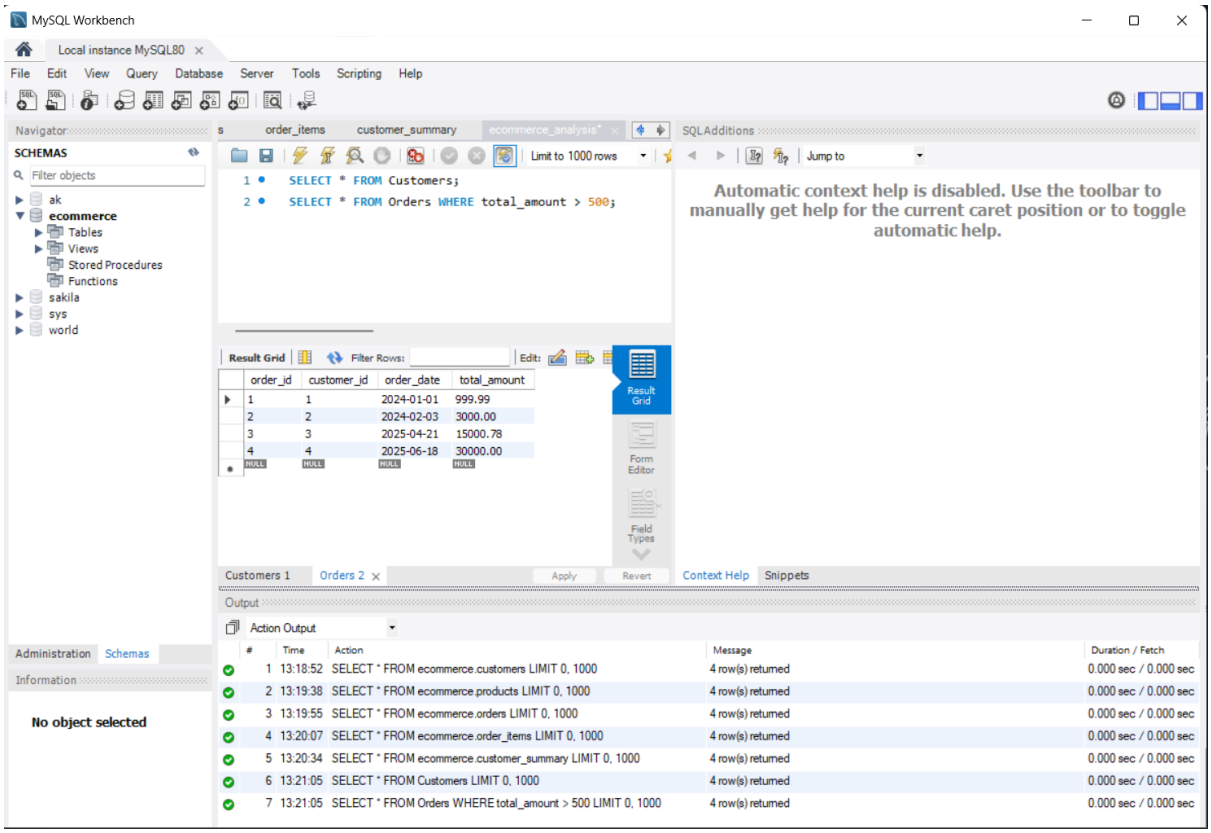

MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
customers products orders order_items customer_summ
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM ecommerce.customers;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Edit:
customer_id name email
1 Alice alice@gmail.com
2 Rahul rahul@gmail.com
3 Koli koli11@gmail.com
4 Avantika avantika25@gmail.com
Result Grid
Form Editor
Field Types
Query Stats
Administration Schemas
Information
No object selected
customers 1
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
customers products orders order_items customer_summ
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM ecommerce.products;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Edit:
product_id name category price
1 Laptop Electronics 999.99
2 Desk Furniture 3000.00
3 Mini robot Gadget 15000.78
4 Smartphone Electronics 30000.00
Result Grid
Form Editor
Field Types
Query Stats
Administration Schemas
Information
No object selected
products 1
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
2 13:19:38 SELECT * FROM ecommerce.products LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
customers products orders order_items customer_summ
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM ecommerce.orders;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Edit:
order_id customer_id order_date total_amount
1 1 2024-01-01 999.99
2 2 2024-02-03 3000.00
3 3 2025-04-21 15000.78
4 4 2025-06-18 30000.00
Result Grid
Form Editor
Field Types
Query Stats
Administration Schemas
Information
No object selected
orders 1
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
2 13:19:38 SELECT * FROM ecommerce.products LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
3 13:19:55 SELECT * FROM ecommerce.orders LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
customers products orders order_items customer_summ
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM ecommerce.order_items;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Edit:
order_item_id order_id product_id quantity price
1 1 1 1 999.99
2 2 2 3 3000.00
3 3 3 2 15000.78
4 4 4 2 30000.00
Result Grid
Form Editor
Field Types
Administration Schemas
Information
No object selected
order_items 1
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
2 13:19:38 SELECT * FROM ecommerce.products LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
3 13:19:55 SELECT * FROM ecommerce.orders LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
4 13:20:07 SELECT * FROM ecommerce.order_items LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
products orders order_items customer_summary
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM ecommerce.customer_summary;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Export:
name total_orders total_spent
Alice 1 999.99
Rahul 1 3000.00
Koli 1 15000.78
Avantika 1 30000.00
Result Grid
Form Editor
Field Types
Read Only
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
2 13:19:38 SELECT * FROM ecommerce.products LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
3 13:19:55 SELECT * FROM ecommerce.orders LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
4 13:20:07 SELECT * FROM ecommerce.order_items LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
5 13:20:34 SELECT * FROM ecommerce.customer_summary LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
Administration Schemas
Information
No object selected


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
ak
ecommerce
Tables
Views
Stored Procedures
Functions
sakila
sys
world
order_items customer_summary ecommerce_analysis*
SQLAdditions
Limit to 1000 rows
Jump to
1 SELECT * FROM Customers;
2 SELECT * FROM Orders WHERE total_amount > 500;
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Result Grid
Filter Rows:
Edit:
order_id customer_id order_date total_amount
1 1 2024-01-01 999.99
2 2 2024-02-03 3000.00
3 3 2025-04-21 15000.78
4 4 2025-06-18 30000.00
Result Grid
Form Editor
Field Types
Customers 1
Orders 2
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 13:18:52 SELECT * FROM ecommerce.customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
2 13:19:38 SELECT * FROM ecommerce.products LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
3 13:19:55 SELECT * FROM ecommerce.orders LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
4 13:20:07 SELECT * FROM ecommerce.order_items LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
5 13:20:34 SELECT * FROM ecommerce.customer_summary LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
6 13:21:05 SELECT * FROM Customers LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
7 13:21:05 SELECT * FROM Orders WHERE total_amount > 500 LIMIT 0, 1000 4 row(s) returned 0.000 sec / 0.000 sec
Administration Schemas
Information
No object selected