

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

WHERE order_amount > 1500;

-- Total spending by each customer

SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;

Result Grid | Filter Rows: _____ | Exports: _____ | Wrap Cell Contents: _____

customer_id	total_spent
3	3499.00
1	3097.00
2	2499.00
4	1299.00

Result 12 X

Output

Action Output

#	Time	Action	Message	Duration / Fetch
48	09:24:21	select * from products LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
49	09:26:28	select * from orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Information

Schema: sampath

Result Grid

Read Only Context Help Snippets

5

ENG IN 09:32 18-11-2025

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

```
53 WHERE order_amount > 1500;
54
55 -- Total spending by each customer
56 • SELECT
57     customer_id,
58     SUM(order_amount) AS total_spent
59 FROM orders
60 GROUP BY customer_id
61 ORDER BY total_spent DESC;
62
```

Result Grid | Filter Rows: _____ | Exports: _____ | Wrap Cell Contents: _____

customer_id	total_spent
3	3499.00
1	3097.00
2	2499.00
4	1299.00

Result Grid

Administration Schemas

Information

Schema: sampath

Action Output

#	Time	Action	Message	Duration / Fetch
48	09:24:21	select * from products LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
49	09:26:28	select * from orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

5

ENG IN 09:32 18-11-2025

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Schema: sampath

Output

Action Output

#	Time	Action	Message	Duration / Fetch
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
54	09:33:14	SELECT AVG(order_amount) AS avg_amount, MIN(order_amount) AS min_amount, MAX(order_amount) AS max_amount FROM orders	1 row(s) returned	0.000 sec / 0.000 sec
55	09:34:38	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id	5 row(s) returned	0.000 sec / 0.000 sec

Result Grid | Filter Rows: | Exports | Wrap Cell Contents | Result Grid | Read Only | Context Help | Snippets

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

Limit to 1000 rows

SQLAdditions: Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
69
70    -- INNER JOIN: Orders with customer names
71 •  SELECT
72     o.order_id,
73     c.customer_name,
74     o.order_amount,
75     o.order_date
76   FROM orders o
77   INNER JOIN customers c
78     ON o.customer_id = c.customer_id;
```

order_id	customer_name	order_amount	order_date
1001	Arjun Reddy	1299.00	2023-07-01
1004	Arjun Reddy	1798.00	2023-07-07
1002	Sita Lakshmi	2499.00	2023-07-02
1003	Rahul Kumar	3499.00	2023-07-05
1005	Priya Sharma	1299.00	2023-07-09

Result 14 X

ENG IN 18-11-2025 09:35

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Schema: sampath

Output

Action Output

#	Time	Action	Message	Duration / Fetch
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
54	09:33:14	SELECT AVG(order_amount) AS avg_amount, MIN(order_amount) AS min_amount, MAX(order_amount) AS max_amount FROM orders	1 row(s) returned	0.000 sec / 0.000 sec
55	09:34:38	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id	5 row(s) returned	0.000 sec / 0.000 sec

Result Grid | Filter Rows: | Exports | Wrap Cell Contents | Result Grid | Read Only | Context Help | Snippets

SQL File 3* - sampath - Schema | SQL File 4* - customers - Table

Limit to 1000 rows

SQLAdditions: Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Object Info Session

2

ENG IN 09:35 18-11-2025

The screenshot shows the MySQL Workbench interface. In the central pane, a query editor displays the following SQL code:

```
69
70    -- INNER JOIN: Orders with customer names
71 •  SELECT
72      o.order_id,
73      c.customer_name,
74      o.order_amount,
75      o.order_date
76  FROM orders o
77  INNER JOIN customers c
78  ON o.customer_id = c.customer_id;
```

Below the code, a results grid shows the following data:

order_id	customer_name	order_amount	order_date
1001	Arjun Reddy	1299.00	2023-07-01
1004	Arjun Reddy	1798.00	2023-07-07
1002	Sita Lakshmi	2499.00	2023-07-02
1003	Rahul Kumar	3499.00	2023-07-05
1005	Priya Sharma	1299.00	2023-07-09

The bottom pane shows a history of actions with details like time, query, message, and duration. The status bar at the bottom right indicates the date and time as 18-11-2025 09:35.

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

```
102 WHERE order_amount > (
103     SELECT AVG(order_amount) FROM orders
104 );
105
106 -- 8.Products that have been ordered at least once
107 • SELECT product_name
108 FROM products
109 WHERE product_id IN (
110     SELECT product_id FROM orders
111 );
```

Result Grid | Filter Rows: | Exports: | Wrap Cell Contents: | Result Grid

customer_id	order_amount
2	2499.00
3	3499.00

Administration Schemas

Information

Schema: sampath

orders 20 X

Action Output

#	Time	Action	Message	Duration / Fetch
56	09:35:38	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
57	09:35:49	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date F...	5 row(s) returned	0.000 sec / 0.000 sec
58	09:36:54	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c...	5 row(s) returned	0.000 sec / 0.000 sec
59	09:38:28	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c...	5 row(s) returned	0.000 sec / 0.000 sec
60	09:39:20	SELECT customer_id, order_amount FROM orders WHERE order_amount > (SE...	2 row(s) returned	0.015 sec / 0.000 sec
61	09:40:24	SELECT customer_id, order_amount FROM orders WHERE order_amount > (SE...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

3

ENG IN 09:40 18-11-2025

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

94 FROM orders o
95 RIGHT JOIN customers c
96 ON o.customer_id = c.customer_id;
97
98 -- 7. Customers who placed orders above average order amount
99 • SELECT customer_id, order_amount
100 FROM orders
101 WHERE order_amount > (SELECT AVG(order_amount) FROM orders)
102
103 ;

Result Grid | Filter Rows: | Exports: | Wrap Cell Contents: | Result Grid

customer_id	order_amount
2	2499.00
3	3499.00

Administration Schemas

Information

Schema: sampath

orders 19 X

Action Output

#	Time	Action	Message	Duration / Fetch
55	09:34:38	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date FROM orders o RIGHT JOIN customers c ON o.customer_id = c.customer_id;	5 row(s) returned	0.000 sec / 0.000 sec
56	09:35:38	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
57	09:35:49	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date FROM orders o RIGHT JOIN customers c ON o.customer_id = c.customer_id;	5 row(s) returned	0.000 sec / 0.000 sec
58	09:36:54	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c, orders o WHERE c.customer_id = o.customer_id AND o.order_amount > 1500	5 row(s) returned	0.000 sec / 0.000 sec
59	09:38:20	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c, orders o WHERE c.customer_id = o.customer_id AND o.order_amount > 1500	5 row(s) returned	0.000 sec / 0.000 sec
60	09:39:20	SELECT customer_id, order_amount FROM orders WHERE order_amount > (SELECT AVG(order_amount) FROM orders)	2 row(s) returned	0.015 sec / 0.000 sec

Object Info Session

ENG IN 09:39 18-11-2025

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Schema: sampath

Output

Action Output

#	Time	Action	Message	Duration / Fetch
54	09:33:14	SELECT AVG(order_amount) AS avg_amount, MIN(order_amount) AS min_am...	1 row(s) returned	0.000 sec / 0.000 sec
55	09:34:38	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date F...	5 row(s) returned	0.000 sec / 0.000 sec
56	09:35:38	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
57	09:36:49	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date F...	5 row(s) returned	0.000 sec / 0.000 sec
58	09:36:54	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c...	5 row(s) returned	0.000 sec / 0.000 sec
59	09:38:28	SELECT c.customer_name, o.order_id, o.order_amount FROM customers c...	5 row(s) returned	0.000 sec / 0.000 sec

Result Grid | Filter Rows: | Exports: | Wrap Cell Contents: | Result Grid | Read Only | Context Help | Snippets

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

ON c.customer_id = o.customer_id;

-- 6.RIGHT JOIN (if supported): All orders + customers

SELECT c.customer_name,
o.order_id,
o.order_amount
FROM orders o
RIGHT JOIN customers c
ON o.customer_id = c.customer_id;

Result Grid

customer_name	order_id	order_amount
Arjun Reddy	1001	1299.00
Arjun Reddy	1004	1798.00
Sita Lakshmi	1002	2499.00
Rahul Kumar	1003	3499.00
Priya Sharma	1005	1299.00

SQLAdditions: Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

18-11-2025 09:38 ENG IN

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

```
53 WHERE order_amount > 1500;
54
55 -- Total spending by each customer
56 • SELECT
57     customer_id,
58     SUM(order_amount) AS total_spent
59 FROM orders
60 GROUP BY customer_id
61 ORDER BY total_spent DESC;
62
```

Result Grid | Filter Rows: _____ | Exports: _____ | Wrap Cell Contents: _____

customer_id	total_spent
3	3499.00
1	3097.00
2	2499.00
4	1299.00

Result Grid

Administration Schemas

Information

Schema: sampath

Action Output

#	Time	Action	Message	Duration / Fetch
48	09:24:21	select * from products LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
49	09:26:28	select * from orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id ORDER BY total_spent DESC;	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

5

ENG IN 09:32 18-11-2025

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sampath

- Tables
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Schema: sampath

Result Grid | Filter Rows: | Exports | Wrap Cell Contents: | Result Grid

Result 14 X

Output

Action Output

#	Time	Action	Message	Duration / Fetch
50	09:29:00	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
51	09:30:59	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
52	09:31:46	SELECT * FROM orders WHERE order_amount > 1500 LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
53	09:32:13	SELECT customer_id, SUM(order_amount) AS total_spent FROM orders GROUP BY customer_id	4 row(s) returned	0.000 sec / 0.000 sec
54	09:33:14	SELECT AVG(order_amount) AS avg_amount, MIN(order_amount) AS min_amount, MAX(order_amount) AS max_amount FROM orders	1 row(s) returned	0.000 sec / 0.000 sec
55	09:34:38	SELECT o.order_id, c.customer_name, o.order_amount, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

2

ENG IN 18-11-2025 09:35

Query 1 SQL File 3* sampath - Schema SQL File 4* customers - Table

Limit to 1000 rows

SQLAdditions: Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
69
70    -- INNER JOIN: Orders with customer names
71 •   SELECT
72     o.order_id,
73     c.customer_name,
74     o.order_amount,
75     o.order_date
76   FROM orders o
77   INNER JOIN customers c
78   ON o.customer_id = c.customer_id;
```