

Output

Action Output			
#	Time	Action	Message
			Duration / Fetch
⚠	1 18:12:20	CREATE DATABASE IF NOT EXISTS ecommerce	1 row(s) affected, 1 warning(s): 1007 Can't create database 'ecommerce'; database exists
✓	2 18:12:20	USE ecommerce	0 row(s) affected
✗	3 18:12:20	CREATE TABLE customers (customer_id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(100))	Error Code: 1050. Table 'customers' already exists
✓	4 18:12:33	USE ecommerce	0 row(s) affected
✓	5 18:12:33	SELECT product_id, name, price FROM products WHERE price > 100 ORDER BY price DESC LIMIT 0, 1000	2 row(s) returned
✓	6 18:12:33	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.product_id GROUP BY p.name	3 row(s) returned
✓	7 18:12:33	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT 0, 1000	2 row(s) returned
✓	8 18:12:33	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT 0, 1000	3 row(s) returned
✓	9 18:12:33	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT 0, 1000	2 row(s) returned
✓	10 18:12:33	SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products) LIMIT 0, 1000	1 row(s) returned
✓	11 18:12:33	SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id)	0 row(s) returned
✓	12 18:12:33	SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id	1 row(s) returned
✓	13 18:12:33	SELECT AVG(price) AS avg_price FROM products LIMIT 0, 1000	1 row(s) returned

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

Administration Schemas Information

Schema: ecommerce

products 1 Result 2 Result 3 Result 4 Result 5 products 6 customers 7 Result 8 × Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.p...	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	name	total_sold
▶	Headphones	2
▶	Laptop	1
▶	Phone	1

Administration Schemas

Information

Schema: ecommerce

products 1 Result 2 × Result 3 Result 4 Result 5 products 6 customers 7 Result 8 Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.p...	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	name	order_id
▶	Alice	1
	Bob	2

Administration Schemas

Information Schema: ecommerce

products 1 Result 2 Result 3 × Result 4 Result 5 products 6 customers 7 Result 8 Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.product_id;	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	name	order_id
▶	Alice	1
▶	Bob	2
▶	Charlie	NULL

Administration Schemas

Information

Schema: ecommerce

products 1 Result 2 Result 3 Result 4 × Result 5 products 6 customers 7 Result 8 Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.p...	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	name	price
▶	Laptop	999.99

Administration Schemas

Information

Schema: ecommerce

products 1 Result 2 Result 3 Result 4 Result 5 products 6 × customers 7 Result 8 Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.p...	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries*

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Context Help | Snippets

Administration Schemas Information Schema: ecommerce

product_id	name	price
1	Laptop	999.99
2	Phone	499.49
NULL	NULL	NULL

products 1 × Result 2 Result 3 Result 4 Result 5 products 6 customers 7 Result 8 Result 9

Output

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.product_id;	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	name	order_id
▶	Alice	1
	Bob	2

Administration Schemas

Information Schema: ecommerce

products 1 Result 2 Result 3 Result 4 Result 5 × products 6 customers 7 Result 8 Result 9

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.product_id;	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT 1;	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator ecommerce_dataset SQL File 4* ecommerce_queries* SQLAdditions

SCHEMAS

Filter objects

aryadb
ecommerce
Tables
Views
Stored Procedures
Functions
sys

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

```
7
8 -- b. INNER, LEFT, RIGHT JOIN
9 • SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id;
10 • SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id;
11 • SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id;
12
13 -- c. Subqueries
14 • SELECT name, price FROM products WHERE price > (SELECT AVG(price) FROM products);
15 • SELECT name FROM customers WHERE customer_id IN (SELECT customer_id FROM orders GROUP BY customer_id HAVING COUNT(*) > 1);
16
17 -- d. Aggregates: SUM, AVG
18 • SELECT SUM(p.price * oi.quantity) AS total_revenue FROM order_items oi JOIN products p ON oi.product_id = p.product_id;
19 • SELECT AVG(price) AS avg_price FROM products;
20
21 -- e. Create View
22 CREATE VIEW
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg_price
526.490000

Administration Schemas

Information

Schema: ecommerce

products 1 Result 2 Result 3 Result 4 Result 5 products 6 customers 7 Result 8 Result 9 ×

Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
19	17:36:29	SELECT p.name, SUM(oi.quantity) AS total_sold FROM order_items oi JOIN products p ON oi.product_id = p.p...	3 row(s) returned	0.000 sec / 0.000 sec
20	17:36:29	SELECT c.name, o.order_id FROM customers c INNER JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.016 sec / 0.000 sec
21	17:36:29	SELECT c.name, o.order_id FROM customers c LEFT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	3 row(s) returned	0.000 sec / 0.000 sec
22	17:36:29	SELECT c.name, o.order_id FROM customers c RIGHT JOIN orders o ON c.customer_id = o.customer_id LIMIT...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session