Local instance MySQL80 ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

Query 1   ×

SQL Additions

Limit to 1000 rows

```sql
1    -- Top 5 products with highest total sales amount
2  • SELECT p.Product, SUM(s.Amount) AS TotalSales
3    FROM sales s
4    JOIN products p ON s.PID = p.PID
5    GROUP BY p.Product
6    ORDER BY TotalSales DESC
7    LIMIT 5;
8
```

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

Navigator

SCHEMAS

Filter objects

▼ ⬛ awesome chocolates
  ▼ 🗂 Tables
    ▶ ▦ geo
    ▶ ▦ people
    ▶ ▦ products
    ▶ ▦ sales
  🗂 Views
  🗂 Stored Procedures
  🗂 Functions
▶ ⬛ deepanshdb
▶ ⬛ ecommerce1db
▶ ⬛ ecommerce2
▶ ⬛ ecommerce3
▶ ⬛ ecommerce4
▼ ⬛ ecommerce_dataset
  ▶ 🗂 Tables
  🗂 Views
  🗂 Stored Procedures
  🗂 Functions
▶ ⬛ ecommercedb
▶ ⬛ newschema
▶ ⬛ sys

Administration   Schemas

Information

Schema: awesome
chocolates

Result Grid | 🔳   Filter Rows: | Export: | Wrap Cell Contents: | Fetch rows:

Product	TotalSales
After Nines	2112502
Raspberry Choco	2090242
Almond Choco	2055627
99% Dark & Pure	2023070
Organic Choco Syrup	2016707

Result
Grid

Form
Editor

Field
Types

Query
Stats

Execution
Plan

Result 5   ×

ⓘ Read Only   Context Help   Snippets

Object Info   Session

Output

Query Completed

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

▼ ▤ awesome chocolates
  ▼ ▤ Tables
    ▶ ▤ geo
    ▶ ▤ people
    ▶ ▤ products
    ▶ ▤ sales
  ▤ Views
  ▤ Stored Procedures
  ▤ Functions
▶ ▤ deepanshdb
▶ ▤ ecommerce1db
▶ ▤ ecommerce2
▶ ▤ ecommerce3
▶ ▤ ecommerce4
▼ ▤ ecommerce_dataset
  ▶ ▤ Tables
  ▤ Views
  ▤ Stored Procedures
  ▤ Functions
▶ ▤ ecommercedb
▶ ▤ newschema
▶ ▤ sys

Administration Schemas

Information

Schema: awesome
chocolates

Object Info Session

Query 1 ×

```
 8  •  SELECT s.*, pe.Salesperson
 9     FROM sales s
10     LEFT JOIN people pe ON s.SPID = pe.SPID;
11
12     -- RIGHT JOIN: All salespersons with their sales (if any)
13  •  SELECT pe.Salesperson, s.Amount
14     FROM sales s
15     RIGHT JOIN people pe ON s.SPID = pe.SPID;
16
```

Result Grid   Filter Rows: Export: Wrap Cell Contents: Fetch rows:

Salesperson	Amount
Barr Faughny	6503
Barr Faughny	6636
Barr Faughny	280
Barr Faughny	14238
Barr Faughny	5278
Barr Faughny	5159
Barr Faughny	2163
Barr Faughny	7567
Barr Faughny	8673
Barr Faughny	2653
Barr Faughny	5684
Barr Faughny	7049
Barr Faughny	308
Barr Faughny	7203
Barr Faughny	6048
Barr Faughny	413
Barr Faughny	2310
Barr Faughny	1057
Barr Faughny	3493

Result 6   Result 7   Result 8 ×

Output

Query Completed

SQL Additions

Jump to

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

Read Only   Context Help   Snippets

```sql
1    -- Products that have sales above average sales amount
2  • SELECT p.Product, SUM(s.Amount) AS TotalSales
3    FROM sales s
4    JOIN products p ON s.PID = p.PID
5    GROUP BY p.Product
6    HAVING TotalSales > (
7        SELECT AVG(Amount) FROM sales
8    );
9
```

Product	TotalSales
Raspberry Choco	2090242
White Choc	1913296
99% Dark & Pure	2023070
Baker's Choco Chips	1906583
Manuka Honey Choco	1965089
85% Dark Bars	1953063
Organic Choco Syrup	2016707
Caramel Stuffed Bars	2010407
Milk Bars	1989078
Mint Chip Choco	1871751
Orange Choco	1899212
Eclairs	1955975
After Nines	2112502
Spicy Special Slims	2004415
Drinking Coco	1941562
Peanut Butter Cubes	1992060
Choco Coated Almonds	1980566

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

awesome_choclate* ×

Limit to 1000 rows

```sql
1    -- Average number of boxes sold per region
2    SELECT g.Region, AVG(s.Boxes) AS AvgBoxes
3    FROM sales s
4    JOIN geo g ON s.GeoID = g.GeoID
5    GROUP BY g.Region;
6
7    -- Total amount and customers per product category
8  • SELECT pr.Category, SUM(s.Amount) AS TotalAmount, SUM(s.Customers) AS TotalCustomers
9    FROM sales s
10   JOIN products pr ON s.PID = pr.PID
11   GROUP BY pr.Category;
12
```

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

Category	TotalAmount	TotalCustomers
Bars	21746501	621656
Other	7836654	224875
Bites	13978391	386416

Result 21   Result 22 ×

Read Only

Output

Query Completed

Local instance MySQL80 ×

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

Navigator

SCHEMAS

Filter objects

- ▼ **awesome chocolates**
  - ▼ Tables
    - ► geo
    - ► people
    - ► products
    - ► sales
  - Views
  - Stored Procedures
  - Functions
- ► deepanshdb
- ► ecommerce1db
- ► ecommerce2
- ► ecommerce3
- ► ecommerce4
- ▼ ecommerce_dataset
  - ► Tables
  - Views
  - Stored Procedures
  - Functions
- ► ecommercedb
- ► newschema
- ► sys

Administration   Schemas

Information

Schema: awesome chocolates

Object Info   Session

awesome_chocolate* ×

Limit to 1000 rows ▾

```
1    -- View to analyze total sales by product and region
2    CREATE VIEW ProductSalesByRegion AS
3    SELECT p.Product, g.Region, SUM(s.Amount) AS TotalSales
4    FROM sales s
5    JOIN products p ON s.PID = p.PID
6    JOIN geo g ON s.GeoID = g.GeoID
7    GROUP BY p.Product, g.Region;
8
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	18:43:16	show tables	4 row(s) returned	0.062 sec / 0.000 sec
2	18:43:27	desc geo	3 row(s) returned	0.000 sec / 0.000 sec
3	18:43:44	select * from geo LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
4	18:44:18	select * from sales LIMIT 0, 1000	1000 row(s) returned	0.032 sec / 0.000 sec
5	18:48:02	SELECT p.Product, SUM(s.Amount) AS TotalSales FROM sales s JOIN products p ON s.PID = p.PID GROUP ...	5 row(s) returned	0.047 sec / 0.000 sec
6	18:50:54	SELECT pe.Salesperson, g.Geo, s.Amount FROM sales s INNER JOIN people pe ON s.SPID = pe.SPID INNE...	1000 row(s) returned	0.125 sec / 0.000 sec
7	18:50:54	SELECT s.*, pe.Salesperson FROM sales s LEFT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
8	18:50:54	SELECT pe.Salesperson, s.Amount FROM sales s RIGHT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
9	18:51:59	SELECT pe.Salesperson, g.Geo, s.Amount FROM sales s INNER JOIN people pe ON s.SPID = pe.SPID INNE...	1000 row(s) returned	0.000 sec / 0.000 sec
10	18:51:59	SELECT s.*, pe.Salesperson FROM sales s LEFT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
11	18:51:59	SELECT pe.Salesperson, s.Amount FROM sales s RIGHT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
12	18:51:59	SELECT p.Product, SUM(s.Amount) AS TotalSales FROM sales s JOIN products p ON s.PID = p.PID GROUP ...	5 row(s) returned	0.063 sec / 0.000 sec

Query interrupted

SQL Additions

Jump to

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

Context Help   Snippets

File   Edit   View   Query   Database   Server   Tools   Scripting   Help

Navigator

SCHEMAS

Filter objects

▼ 🔲 awesome chocolates
  ▼ 🗐 Tables
    ▶ 🔳 geo
    ▶ 🔳 people
    ▶ 🔳 products
    ▶ 🔳 sales
  🗐 Views
  🗐 Stored Procedures
  🗐 Functions
▶ 🔲 deepanshdb
▶ 🔲 ecommerce1db
▶ 🔲 ecommerce2
▶ 🔲 ecommerce3
▶ 🔲 ecommerce4
▼ 🔲 ecommerce_dataset
  ▶ 🗐 Tables
  🗐 Views
  🗐 Stored Procedures
  🗐 Functions
▶ 🔲 ecommercedb
▶ 🔲 newschema
▶ 🔲 sys

Administration   Schemas

Information

Schema: awesome
chocolates

Object Info   Session

awesome_choclate* ×

Limit to 1000 rows

```
1    -- Add indexes for faster JOINs and WHERE clause performance
2    CREATE INDEX idx_sales_spid ON sales(SPID);
3  • CREATE INDEX idx_sales_geoid ON sales(GeoID);
4  • CREATE INDEX idx_sales_pid ON sales(PID);
5  • CREATE INDEX idx_sales_saledate ON sales(SaleDate);
6
```

SQL Additions

◄   ►   Jump to

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

Context Help   Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓	1	18:43:16	show tables	4 row(s) returned	0.062 sec / 0.000 sec
✓	2	18:43:27	desc geo	3 row(s) returned	0.000 sec / 0.000 sec
✓	3	18:43:44	select * from geo LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
✓	4	18:44:18	select * from sales LIMIT 0, 1000	1000 row(s) returned	0.032 sec / 0.000 sec
✓	5	18:48:02	SELECT p.Product, SUM(s.Amount) AS TotalSales FROM sales s JOIN products p ON s.PID = p.PID GROUP ...	5 row(s) returned	0.047 sec / 0.000 sec
✓	6	18:50:54	SELECT pe.Salesperson, g.Geo, s.Amount FROM sales s INNER JOIN people pe ON s.SPID = pe.SPID INNE...	1000 row(s) returned	0.125 sec / 0.000 sec
✓	7	18:50:54	SELECT s.*, pe.Salesperson FROM sales s LEFT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
✓	8	18:50:54	SELECT pe.Salesperson, s.Amount FROM sales s RIGHT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
✓	9	18:51:59	SELECT pe.Salesperson, g.Geo, s.Amount FROM sales s INNER JOIN people pe ON s.SPID = pe.SPID INNE...	1000 row(s) returned	0.000 sec / 0.000 sec
✓	10	18:51:59	SELECT s.*, pe.Salesperson FROM sales s LEFT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
✓	11	18:51:59	SELECT pe.Salesperson, s.Amount FROM sales s RIGHT JOIN people pe ON s.SPID = pe.SPID LIMIT 0, 1000	1000 row(s) returned	0.016 sec / 0.000 sec
✓	12	18:51:59	SELECT p.Product, SUM(s.Amount) AS TotalSales FROM sales s JOIN products p ON s.PID = p.PID GROUP ...	5 row(s) returned	0.063 sec / 0.000 sec

Query interrupted