

LA PINO'Z SQL

PROJECT



INTRODUCTION



In my SQL project, I applied a range of queries to tackle diverse data-related inquiries. Leveraging SELECT, JOIN, and WHERE clauses, I extracted targeted information from relational databases. Utilizing GROUP BY and HAVING, I aggregated data and performed insightful analyses, uncovering patterns and trends within the dataset. Subqueries were employed to nest queries, enabling complex data manipulations and comparisons. I optimized performance by indexing key columns and fine-tuning query structures. Through the implementation of SQL functions and expressions, I calculated derived metrics and transformed raw data into actionable insights. My project demonstrated proficiency in SQL fundamentals, showcasing the ability to navigate relational databases and extract meaningful information through strategic query formulation.

BASIC QUESTIONS

- 01** Retrieve the total number of orders placed.
- 02** Calculate the total revenue generated from pizza sales.
- 03** Identify the highest-priced pizza.
- 04** Identify the most common pizza size ordered.
- 05** List the top 5 most ordered pizza types along with their quantities.

INTERMEDIATE QUESTIONS

- 01** Join the necessary tables to find the total quantity of each pizza category ordered.
- 02** Determine the distribution of orders by hour of the day.
- 03** Join relevant tables to find the category-wise distribution of pizzas.
- 04** Group the orders by date and calculate the average number of pizzas ordered per day.
- 05** Determine the top 3 most ordered pizza types based on revenue.

ADVANCED QUESTIONS

- 01** Calculate the percentage contribution of each pizza type to total revenue.
- 02** Analyze the cumulative revenue generated over time.
- 03** Determine the top 3 most ordered pizza types based on revenue for each pizza category.

File Edit View Query Database Server Tools Scripting Help

Navigator orders SQL File 7*

SCHEMAS

Filter objects

future

lapinoz

Tables

orders

orders_details

pizza_types

pizzas

Views

Stored Procedures

Functions

laptop

motorola

srajal

sys

Administration Schemas

Information

Schema: lapinoz

-- Retrieve the total number of orders placed.

1

2

3 • SELECT

4 COUNT(order_id) AS total_orders

5 FROM

6 orders;

7

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

	total_orders
	21350

Result 2 ×

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
60	13:48:33	SELECT * FROM lapinoz.orders_details	48620 row(s) returned	0.016 sec / 0.140 sec
61	13:48:37	SELECT * FROM lapinoz.pizzas	96 row(s) returned	0.000 sec / 0.000 sec
62	13:49:23	select (order_details.quantity * pizzas.price) as total_sales from order_details join pizzas on pizzas.pizza_id = ord...	Error Code: 1146. Table 'lapinoz.order_details' doesn't exist	0.000 sec
63	13:49:25	select (order_details.quantity * pizzas.price) as total_sales from order_details join pizzas on pizzas.pizza_id = ord...	Error Code: 1146. Table 'lapinoz.order_details' doesn't exist	0.000 sec
64	13:57:01	select 'from orders'	21350 row(s) returned	0.000 sec / 0.032 sec
65	13:58:26	select count(order_id) as total_orders from orders	1 row(s) returned	0.016 sec / 0.000 sec

Object Info Session

SQLAdditions

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

Navigator orders SQL File 7* orders_details pizzas pizzas orders_details

SCHEMAS

Filter objects

future lapinoz pizza_types pizzas Views Stored Procedures Functions

laptop motorola srajal sys

-- Calculate the total revenue generated from pizza sales.

10 • SELECT
11 ROUND(SUM(orders_details.quantity * pizzas.price),
12 2) AS total_sales
13 FROM
14 orders_details
15 JOIN
16 pizzas ON pizzas.pizza_id = orders_details.pizza_id;

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

total_sales
817860.05

Administration Schemas

Information

Schema: lapinoz

Result 6 x Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
77	14:11:58	SELECT round(sum(orders_details.qunatity*pizzas.price),2) as total_sales from orders_details join pizzas on pizz...	Error Code: 1054. Unknown column 'orders_details.qunatity' in 'field list'	0.000 sec
78	14:14:19	Select (orders_details.qunatity*pizzas.price)from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	Error Code: 1054. Unknown column 'orders_details.qunatity' in 'field list'	0.000 sec
79	14:14:35	Select (orders_details*qunatity*pizzas.price)from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	Error Code: 1054. Unknown column 'orders_details' in 'field list'	0.000 sec
80	14:14:46	SELECT * FROM lapinoz.orders_details	48620 row(s) returned	0.000 sec / 0.156 sec
81	14:16:00	Select (orders_details.quantity*pizzas.price)from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	48620 row(s) returned	0.015 sec / 0.250 sec

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

File Edit View Query Database Server Tools Scripting Help

Navigator orders SQL File 7*

SCHEMAS

Filter objects

future
lapinoz (selected)
Tables
orders
orders_details
pizza_types
pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

18
19 -- Identify the highest-priced pizza.
20 • SELECT
21 pizza_types.name, pizzas.price
22 FROM
23 pizza_types
24 JOIN
25 pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
26 ORDER BY pizzas.price DESC
27 LIMIT 1;
28
29
30

SQLAdditions

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

Administration Schemas

Information Schema: lapinoz

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

	name	price
▶	The Greek Pizza	35.95

Result 8 ×

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✖ 79	14:14:35	Select (orders_details*qunatity*pizzas.price) from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	Error Code: 1054. Unknown column 'orders_details' in 'field list'	0.000 sec
✓ 80	14:14:46	SELECT * FROM lapinoz.orders_details	48620 row(s) returned	0.000 sec / 0.156 sec
✓ 81	14:16:00	Select (orders_details.quantity*pizzas.price) from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	48620 row(s) returned	0.015 sec / 0.250 sec
✓ 82	14:16:50	Select round(sum(orders_details.quantity*pizzas.price),2) as total_sales from orders_details join pizzas on pizzas...	1 row(s) returned	0.203 sec / 0.000 sec
✓ 83	14:24:29	SELECT ROUND(SUM(orders_details.quantity * pizzas.price), 2) AS total_sales FROM orders_det...	1 row(s) returned	0.141 sec / 0.000 sec
✓ 84	14:24:46	select pizza_types.name,pizzas.price from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pi...	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* pizzas

SCHEMAS

Filter objects

future
lapinoz
Tables: orders, orders_details, pizza_types, pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

-- Identify the most common pizza size ordered.

• SELECT pizzas.size, COUNT(orders_details.order_details_id) AS order_count FROM pizzas JOIN orders_details ON pizzas.pizza_id = orders_details.pizza_id GROUP BY pizzas.size ORDER BY order_count DESC;

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

Administration Schemas

Information

Table: pizzas

Columns:

	size	order_count
L	18526	
M	15385	
S	14137	
XL	544	
XXL	28	

Result 9 ×

Result Grid | Filter Rows: Export: Wrap Cell Content: Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
81	14:16:00	Select (orders_details.quantity*pizzas.price) from orders_details join pizzas on pizzas.pizza_id=orders_details.piz...	48620 row(s) returned	0.015 sec / 0.250 sec
82	14:16:50	Select round(sum(orders_details.quantity*pizzas.price),2) as total_sales from orders_details join pizzas on pizzas...	1 row(s) returned	0.203 sec / 0.000 sec
83	14:24:29	SELECT ROUND(SUM(orders_details.quantity * pizzas.price), 2) AS total_sales FROM orders_det...	1 row(s) returned	0.141 sec / 0.000 sec
84	14:24:46	select pizza_types.name,pizzas.price from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pi...	1 row(s) returned	0.000 sec / 0.000 sec
85	14:30:40	SELECT * FROM lapinoz.pizzas	96 row(s) returned	0.000 sec / 0.000 sec
86	14:34:20	select pizzas.size,count(orders_details.order_details_id) as order_count from pizzas join orders_details on pizzas...	5 row(s) returned	0.250 sec / 0.000 sec

Object Info Session

File Edit View Query Database Server Tools Scripting Help

Navigator orders SQL File 7* pizza_types

SCHEMAS

Filter objects

future
lapinoz
Tables
orders
orders_details
pizza_types
pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

-- List the top 5 most ordered pizza types along with their quantities.

• SELECT pizza_types.name, SUM(orders_details.quantity) AS quantity

FROM pizza_types

JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id

JOIN orders_details ON orders_details.pizza_id = pizzas.pizza_id

GROUP BY pizza_types.name

ORDER BY quantity DESC

LIMIT 5;

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

Administration Schemas

Information

Table: pizzas

Columns:

	name	quantity
▶	The Classic Deluxe Pizza	2453
▶	The Barbecue Chicken Pizza	2432
▶	The Hawaiian Pizza	2422
▶	The Pepperoni Pizza	2418
▶	The Thai Chicken Pizza	2371

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

Result 11 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
✓	86 14:34:20	select pizzas.size,COUNT(orders_details.order_details_id) as order_count from pizzas join orders_details on pizzas...	5 row(s) returned	0.250 sec / 0.000 sec
✓	87 14:35:25	SELECT pizzas.size, COUNT(orders_details.order_details_id) AS order_count FROM pizzas JOIN ...	1 row(s) returned	0.250 sec / 0.000 sec
✓	88 14:39:06	SELECT * FROM lapinoz.pizza_types	32 row(s) returned	0.000 sec / 0.000 sec
✗	89 14:43:25	select pizza_types.name, sum(orders_details.quantity) from pizza_types join pizzas on pizza_types.pizza_type_i...	Error Code: 1055. Expression #1 of ORDER BY clause is not in GROUP BY clause and contains nonaggregate...	0.016 sec
✗	90 14:44:28	select pizza_types.name, sum(orders_details.quantity) from pizza_types join pizzas on pizza_types.pizza_type_i...	Error Code: 1055. Expression #1 of ORDER BY clause is not in GROUP BY clause and contains nonaggregate...	0.000 sec
✓	91 14:45:00	select pizza_types.name, sum(orders_details.quantity) as quantity from pizza_types join pizzas on pizza_types.pi...	5 row(s) returned	0.328 sec / 0.000 sec

Object Info Session

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* pizza_types pizzas pizza_types

SCHEMAS

- future
- lapinoz**
 - Tables: orders, orders_details, pizza_types, pizzas
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

Column	Type
pizza_id	text
pizza_type_id	text
size	text
price	double

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

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Result 12 x

Output:

Action Output

#	Time	Action	Message
89	14:43:25	select pizza_types.name, sum(orders_details.quantity) from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id group by pizza_types.name;	Error Code: 1055
90	14:44:28	select pizza_types.name, sum(orders_details.quantity) from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id group by pizza_types.name;	Error Code: 1055
91	14:45:00	select pizza_types.name, sum(orders_details.quantity) as quantity from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id group by pizza_types.name;	5 row(s) returned
92	14:48:20	SELECT * FROM lapinoz.pizzas	96 row(s) returned
93	14:48:24	SELECT * FROM lapinoz.pizza_types	32 row(s) returned
94	14:51:23	SELECT pizza_types.category, SUM(orders_details.quantity) AS quantity FROM pizza_types JOIN ...	4 row(s) returned

Object Info Session

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders

SCHEMAS

- future
- lapinoz**
 - Tables: orders, orders_details, pizza_types, pizzas
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

Column	Type
pizza_id	text
pizza_type_id	text
size	text
price	double

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

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

Result 14 x

Output:

File Edit View Query Database Server Tools Scripting Help

Navigator SQL File 7* orders

SCHEMAS

Filter objects

future
lapinoz
Tables
orders
orders_details
pizza_types
pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

SQLAdditions

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

orders

```
68 FROM
69     orders
70 GROUP BY HOUR(order_time);
71 -- Join relevant tables to find the category-wise distribution of pizzas.
72 • SELECT
73     category, COUNT(name)
74 FROM
75     pizza_types
76 GROUP BY category;
77
78
79
80
```

Administration Schemas

Information

Table: pizzas

Columns:

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

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

Result 15 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
93	14:48:24	SELECT * FROM lapinoz.pizza_types	32 row(s) returned	0.000 sec / 0.000 sec
94	14:51:23	SELECT pizza_types.category, SUM(orders_details.quantity) AS quantity FROM pizza_types JOIN ...	4 row(s) returned	0.312 sec / 0.000 sec
95	14:52:23	SELECT * FROM lapinoz.orders	21350 row(s) returned	0.000 sec / 0.031 sec
96	14:56:49	select hour(order_time),count(order_id) from orders group by hour(order_time)	15 row(s) returned	0.016 sec / 0.000 sec
97	14:57:22	select hour(order_time)as hour,count(order_id)as order_count from orders group by hour(order_time)	15 row(s) returned	0.032 sec / 0.000 sec
98	15:01:46	select category,count(name)from pizza_types group by category	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7 orders

SCHEMAS

Filter objects

future
lapinoz
Tables
orders
orders_details
pizza_types
pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

-- Group the orders by date and calculate the average number of pizzas ordered per day.

77 -- Group the orders by date and calculate the average number of pizzas ordered per day.
78 • SELECT
79 ROUND(AVG(quantity), 0)
80 FROM
81 (SELECT
82 orders.order_date, SUM(orders_details.quantity) AS quantity
83 FROM
84 orders
85 JOIN orders_details ON orders.order_id = orders_details.order_id
86 GROUP BY orders.order_date) AS order_quantity;
87
88
89

Administration Schemas

Information:

Table: pizzas

Columns:

pizza_id	text
pizza_type_id	text
size	text
price	double

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

round(avg(quantity),0)
138

Result 17 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
97	14:57:22	select hour(order_time)as hour,count(order_id)as order_count from orders group by hour(order_time)	15 row(s) returned	0.032 sec / 0.000 sec
98	15:01:46	select category,count(name) from pizza_types group by category	4 row(s) returned	0.000 sec / 0.000 sec
99	15:06:48	select orders.order_date,sum(orders_details.quantity) from orders join orders_details on orders.order_id=orders... 358 row(s) returned	0.156 sec / 0.000 sec	
100	15:09:49	select avg(quantity) from select orders.order_date,sum(orders_details.quantity) as quantity from orders join order... Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec	
101	15:12:03	select round(avg(quantity),0) from (select orders.order_date,sum(orders_details.quantity) as quantity from orders ... Error Code: 1248. Every derived table must have its own alias	0.016 sec	
102	15:12:36	select round(avg(quantity),0) from (select orders.order_date,sum(orders_details.quantity) as quantity from orders ... 1 row(s) returned	0.187 sec / 0.000 sec	

Object Info Session

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

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7 orders

SCHEMAS

Filter objects

future
lapinoz
Tables
orders
orders_details
pizza_types
pizzas
Views
Stored Procedures
Functions
laptop
motorola
srajal
sys

-- Group the orders by date and calculate the average number of pizzas ordered per day.

77 -- Group the orders by date and calculate the average number of pizzas ordered per day.
78 • SELECT
79 ROUND(AVG(quantity), 0)
80 FROM
81 (SELECT
82 orders.order_date, SUM(orders_details.quantity) AS quantity
83 FROM
84 orders
85 JOIN orders_details ON orders.order_id = orders_details.order_id
86 GROUP BY orders.order_date) AS order_quantity;
87
88
89

Administration Schemas

Information:

Table: pizzas

Columns:

pizza_id	text
pizza_type_id	text
size	text
price	double

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

round(avg(quantity),0)
138

Result 17 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
97	14:57:22	select hour(order_time)as hour,count(order_id)as order_count from orders group by hour(order_time)	15 row(s) returned	0.032 sec / 0.000 sec
98	15:01:46	select category,count(name) from pizza_types group by category	4 row(s) returned	0.000 sec / 0.000 sec
99	15:06:48	select orders.order_date,sum(orders_details.quantity) from orders join orders_details on orders.order_id=orders... 358 row(s) returned	0.156 sec / 0.000 sec	
100	15:09:49	select avg(quantity) from select orders.order_date,sum(orders_details.quantity) as quantity from orders join order... Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec	
101	15:12:03	select round(avg(quantity),0) from (select orders.order_date,sum(orders_details.quantity) as quantity from orders ... Error Code: 1248. Every derived table must have its own alias	0.016 sec	
102	15:12:36	select round(avg(quantity),0) from (select orders.order_date,sum(orders_details.quantity) as quantity from orders ... 1 row(s) returned	0.187 sec / 0.000 sec	

Object Info Session

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

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders pizza_types

SCHEMAS

- future
- lapinoz
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

-- Determine the top 3 most ordered pizza types based on revenue.

```
88 • SELECT
89     pizza_types.name,
90     SUM(orders_details.quantity * pizzas.price) AS revenue
91 FROM
92     pizza_types
93     JOIN
94     pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
95     JOIN
96     orders_details ON orders_details.pizza_id = pizzas.pizza_id
97 GROUP BY pizza_types.name
98 ORDER BY revenue DESC
99 LIMIT 3;
```

Administration Schemas

Information

Table: pizzas

Columns:

pizza_id	text
pizza_type_id	text
size	text
price	double

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

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Result 25 x Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
106	15:22:29	select pizza_types.name, orders_details.quantity*pizzas.price from pizza_types join pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id	0 row(s) returned	0.078 sec / 0.000 sec
107	15:23:08	select pizza_types.name, orders_details.quantity*pizzas.price as revenue from pizza_types join pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id	0 row(s) returned	0.078 sec / 0.000 sec
108	15:24:43	SELECT ROUND(AVG(quantity), 0) FROM (SELECT orders.order_date, SUM(orders_details.quantity) as quantity FROM orders_details JOIN pizzas ON orders_details.pizza_id = pizzas.pizza_id GROUP BY orders.order_date)	1 row(s) returned	0.140 sec / 0.000 sec
109	15:24:55	select pizza_types.name, orders_details.quantity*pizzas.price as revenue from pizza_types join pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id	48620 row(s) returned	0.000 sec / 0.219 sec
110	15:25:43	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id	32 row(s) returned	0.296 sec / 0.000 sec
111	15:26:13	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id	3 row(s) returned	0.313 sec / 0.000 sec

Object Info Session

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

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders pizza_types

SCHEMAS

Filter objects

- future
- lapinoz**
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas**
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

	category	revenue
▶	Classic	26.90596025566967
▶	Supreme	25.45631126009862
▶	Chicken	23.955137556847287
▶	Veggie	23.682590927384577

Result 28 X

Output

Action Output

#	Time	Action	Message
109	15:24:55	select pizza_types.name, orders_details.quantity*pizzas.price as revenue from pizza_types join pizzas o...	48620 row(s) returned
110	15:25:43	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas... 32 row(s) returned	
111	15:26:13	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas... 3 row(s) returned	
112	15:35:53	SELECT pizza_types.category, (SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(SUM(o... 4 row(s) returned	
113	15:37:20	SELECT pizza_types.category, round(SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(... 4 row(s) returned	
114	15:38:01	SELECT pizza_types.category, (SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(SUM(o... 4 row(s) returned	

Object Info Session

MySQL Workbench

Local instance MySQL80 X

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders pizza_types

SCHEMAS

Filter objects

- future
- lapinoz**
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas**
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

	category	revenue
▶	Classic	26.90596025566967
▶	Supreme	25.45631126009862
▶	Chicken	23.955137556847287
▶	Veggie	23.682590927384577

Result 28 X

Output

Action Output

#	Time	Action	Message
109	15:24:55	select pizza_types.name, orders_details.quantity*pizzas.price as revenue from pizza_types join pizzas o...	48620 row(s) returned
110	15:25:43	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas... 32 row(s) returned	
111	15:26:13	select pizza_types.name, sum(orders_details.quantity*pizzas.price) as revenue from pizza_types join pizzas... 3 row(s) returned	
112	15:35:53	SELECT pizza_types.category, (SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(SUM(o... 4 row(s) returned	
113	15:37:20	SELECT pizza_types.category, round(SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(... 4 row(s) returned	
114	15:38:01	SELECT pizza_types.category, (SUM(orders_details.quantity * pizzas.price)/(SELECT ROUND(SUM(o... 4 row(s) returned	

Object Info Session

The screenshot shows the SSMS interface with the following details:

- File, Edit, View, Query, Database, Server, Tools, Scripting, Help** menu bar.
- Schemas** pane on the left showing databases: future, lapinoz, laptop, motorola, srajal, sys. The **lapinoz** database is selected.
- SQL File 7*** tab in the top center.
- Script toolbar** with various icons for script operations.
- Text Editor** containing the following T-SQL script:

```
120    -- Analyze the cumulative revenue generated over time.
121 •  select order_date,
122      sum(revenue)over(order by order_date)as cum_revenue
123      from
124      (select orders.order_date,
125      sum(orders_details.quantity*pizzas.price)as revenue
126      from orders_details join pizzas
127      on orders_details.pizza_id=pizzas.pizza_id
128      join orders
129      on orders.order_id=orders_details.order_id
130      group by orders.order_date) as sales;
```

- Result Grid** pane showing the results of the query:

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7

- Action Output** pane at the bottom showing the execution history:

#	Time	Action	Message	Duration / Fetch
113	15:37:20	SELECT pizza_types.category, round(SUM(orders_details.quantity * pizzas.price)/(SELECT COUNT(*) FROM orders)) AS Revenue	ROUND(... 4 row(s) returned)	0.422 sec / 0.000 sec
114	15:38:01	SELECT pizza_types.category, (SUM(orders_details.quantity * pizzas.price)/(SELECT COUNT(*) FROM orders)) AS Revenue	ROUND(SUM(o... 4 row(s) returned)	0.422 sec / 0.000 sec
115	15:40:11	SELECT pizza_types.category, round(SUM(orders_details.quantity * pizzas.price) / (SELECT COUNT(*) FROM orders)) AS Revenue	RO... 4 row(s) returned	0.422 sec / 0.000 sec
116	15:40:56	SELECT pizza_types.category, round(SUM(orders_details.quantity * pizzas.price) / (SELECT COUNT(*) FROM orders)) AS Revenue	RO... 4 row(s) returned	0.407 sec / 0.000 sec
117	15:50:04	select orders.order_date, sum(orders_details.quantity*pizzas.price)as revenue from orders_details join pizzas on ...	358 row(s) returned	0.250 sec / 0.000 sec
118	15:52:40	select order_date, sum(revenue)over(order by order_date)as cum_revenue from (select orders.order_date, sum(...	358 row(s) returned	0.282 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* × orders pizza_types

SCHEMAS

- future
- lapinoz
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

SQL Editor:

```
133  (select category,name,revenue,
134   rank()over (partition by category order by revenue desc)as rn
135   from
136  (select pizza_types.name,pizza_types.category,
137   sum((orders_details.quantity)*pizzas.price) as revenue
138   from pizza_types join pizzas
139   on pizza_types.pizza_type_id=pizzas.pizza_type_id
140   join orders_details
141   on orders_details.pizza_id=pizzas.pizza_id
142   group by pizza_types.category,pizza_types.name)as a) as b
143  where rn <=3;
```

Result Grid:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Indian Supreme Pizza	22176.75

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
117	15:50:04	select orders.order_date, sum(orders_details.quantity*pizzas.price)as revenue from orders_details join pizzas on ...	358 row(s) returned	0.250 sec / 0.000 sec
118	15:52:40	select order_date, sum(revenue)over(order by order_date)as cum_revenue from (select orders.order_date, sum(...	358 row(s) returned	0.282 sec / 0.000 sec
119	16:03:40	select pizza_types.name,pizza_types.category, sum((orders_details.quantity)*pizzas.price) as revenue from pizz...	32 row(s) returned	0.359 sec / 0.000 sec
120	16:06:41	select category.name,revenue, rank()over (partition by category order by revenue desc)as m from (select pizza...	32 row(s) returned	0.375 sec / 0.000 sec
121	16:06:58	select category.name,revenue, rank()over (partition by category order by revenue desc)as m from (select pizza...	32 row(s) returned	0.375 sec / 0.000 sec
122	16:10:48	select name,revenue from (select category.name,revenue, rank()over (partition by category order by revenue d...	12 row(s) returned	0.375 sec / 0.000 sec

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders pizza_types

SCHEMAS

Filter objects

- future
- lapinoz**
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas**
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

Column Name	Type
pizza_id	text
pizza_type_id	text
size	text
price	double

Result Grid | Filter Rows: Export: Wrap Cell Content: Result 36

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Tomato Supreme Pizza	22176.75

Action Output

#	Time	Action	Message
117	15:50:04	select orders.order_date, sum(orders_details.quantity*pizzas.price)as revenue from orders_details join pizzas on ...	358 row(s) returned
118	15:52:40	select order_date, sum(revenue)over(order by order_date)as cum_revenue from (select orders.order_date, sum(...	358 row(s) returned
119	16:03:40	select pizza_types.name,pizza_types.category, sum((orders_details.quantity)*pizzas.price) as revenue from pizz...	32 row(s) returned
120	16:06:41	select category.name,revenue, rank()over (partition by category order by revenue desc)as rn from (select pizza...	32 row(s) returned
121	16:06:58	select category.name,revenue, rank()over (partition by category order by revenue desc)as rn from (select pizza...	32 row(s) returned
122	16:10:48	select name,revenue from (select category.name,revenue, rank()over (partition by category order by revenue d...	12 row(s) returned

Object Info Session

MySQL Workbench Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: orders SQL File 7* orders pizza_types

SCHEMAS

Filter objects

- future
- lapinoz**
 - Tables
 - orders
 - orders_details
 - pizza_types
 - pizzas**
 - Views
 - Stored Procedures
 - Functions
- laptop
- motorola
- srajal
- sys

Administration Schemas

Information

Table: pizzas

Columns:

Column Name	Type
pizza_id	text
pizza_type_id	text
size	text
price	double

Result Grid | Filter Rows: Export: Wrap Cell Content: Result 36

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Tomato Supreme Pizza	22176.75

Action Output

#	Time	Action	Message
117	15:50:04	select orders.order_date, sum(orders_details.quantity*pizzas.price)as revenue from orders_details join pizzas on ...	358 row(s) returned
118	15:52:40	select order_date, sum(revenue)over(order by order_date)as cum_revenue from (select orders.order_date, sum(...	358 row(s) returned
119	16:03:40	select pizza_types.name,pizza_types.category, sum((orders_details.quantity)*pizzas.price) as revenue from pizz...	32 row(s) returned
120	16:06:41	select category.name,revenue, rank()over (partition by category order by revenue desc)as rn from (select pizza...	32 row(s) returned
121	16:06:58	select category.name,revenue, rank()over (partition by category order by revenue desc)as rn from (select pizza...	32 row(s) returned
122	16:10:48	select name,revenue from (select category.name,revenue, rank()over (partition by category order by revenue d...	12 row(s) returned

Object Info Session

THANK YOU

20 April, 2024