

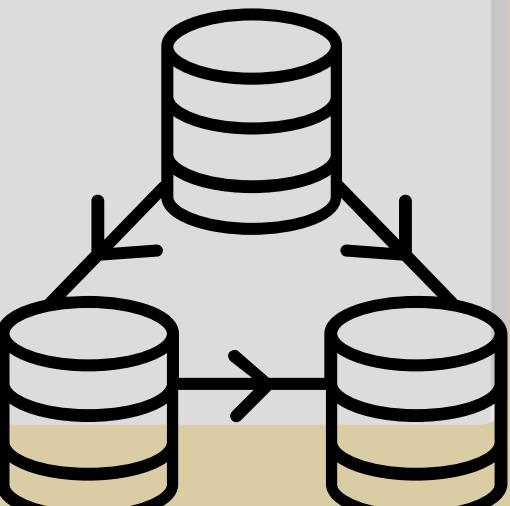
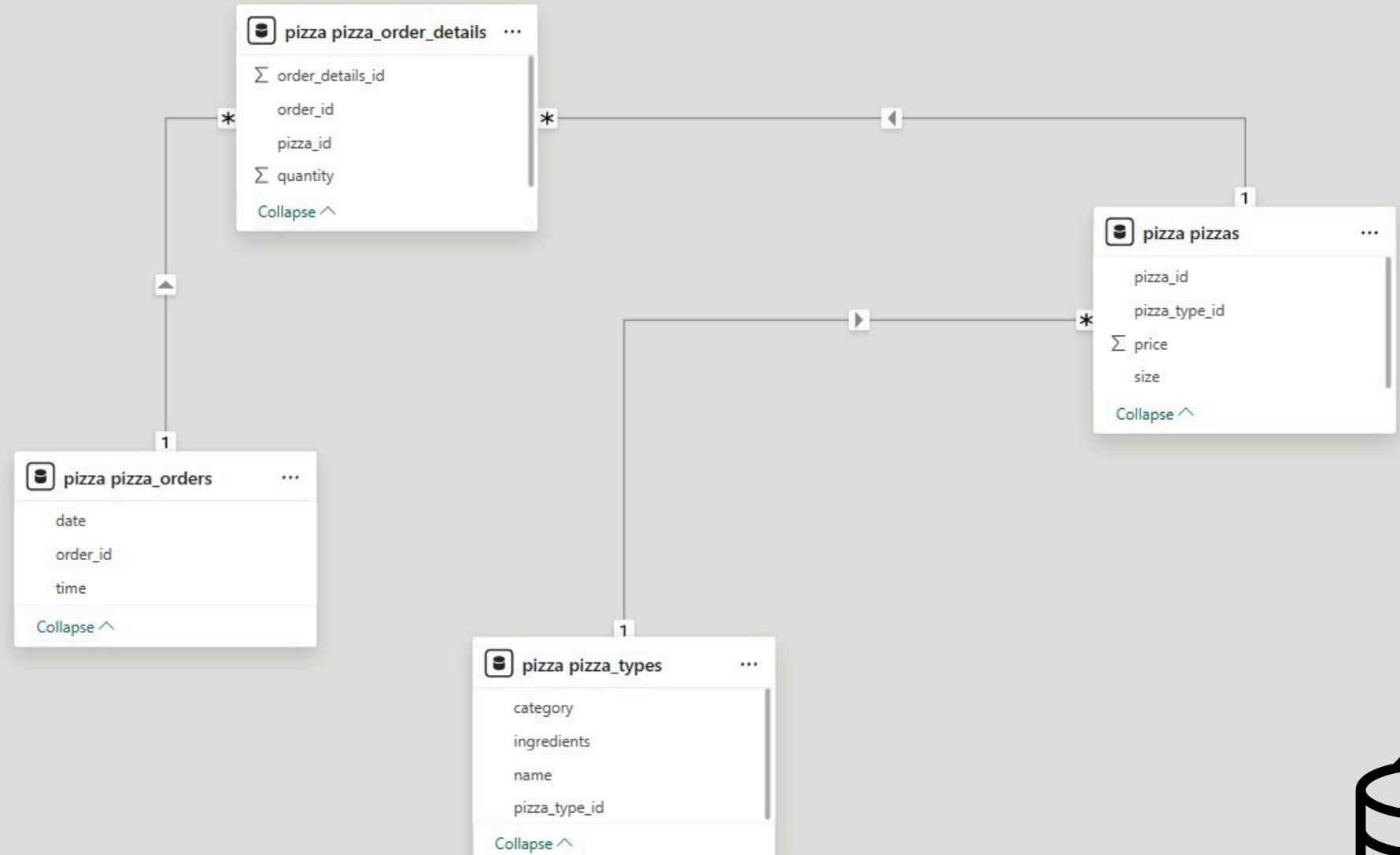


Pizza Sales Analysis



Subhapriya Sanki

Schema



#1 Query The Total Number Of Orders Placed?

#2 Query The Total Number Of Pizza Id Types?

#3 Query The Total Number Of Pizza Types?

#4 Query Total Order Quantity Placed?

#5 Query The First Date Of Transaction And Last Date Of Transaction?

#6 Query The Total Revenue Generated ?

#7 Query The Highest Priced Pizza? #Display Pizza Name, Price

#8 Query The Lowest Priced Pizza? #Display Pizza Name, Price

#9 Query The Pizza Ordered Qty By Size?

#10 Query The Pizza Ordered Qty By Category?

#11 Query The Pizza Ordered Qty By Pizza Name?

#12 Query The Top 7 Types Based Orders Qty? Display Pizza Name, Order Qty?

#13 Query The Distribution Of Orders By Hour Of Day?
#14 Query The Pizza Order Qty By Date And Calculate The Average Numbers Of Pizzas Ordered Per Day?

#14 Query The Pizza Order Qty By Date And Calculate The Average Numbers Of Pizzas Ordered Per Day?

#15 Query The Top 7 Pizza Names By Revenue?
#Display Pizza Name, Revenue , Orderqty

#16 Query The Percentage Contribution Of Each Pizza Category To Total Revenue?

17 Analyse the cumulative revenue generated over time?

#18 Determine The Top 3 Most Ordered Pizza Types Based On Revenue For Each Category?

pizza

▼ C +



Filter



ENTITIES 4



pizza_order_details

order_details_id	int
order_id	int
pizza_id	text
quantity	int

pizza_orders

order_id	int
date	text
time	text

pizza_types

pizza_type_id	text
name	text
category	text
ingredients	text

pizzas

pizza_id	text
pizza_type_id	text
size	text
price	double

```
1 #1 Query The Total Number Of Orders Placed? #49574
2 SELECT COUNT(order_id) TTL_ORD from pizza_order_details ;
3
4 #2 Query The Total Number Of Pizza Id Types? # 32
5 SELECT COUNT( DISTINCT pizza_type_id ) from pizzas ;
6 #3 Query The Total Number Of Pizza Types? #32
7 SELECT COUNT ( DISTINCT NAME ) FROM pizza_types;
8 #4 Query Total Order Quantity Placed? #49574
9 SELECT SUM(quantity) TTL_QTY from pizza_order_details;
10 #5 Query The First Date Of Transaction And Last Date Of Transaction?
11
12 SELECT MAX(date) F_DATE , MIN(date) L_DATE from pizza_orders;
```

	F_DATE	L_DATE
1	2015-12-31	2015-01-01

Save

pizza

▼ C +



Filter



ENTITIES 4



pizza_order_details

order_details_id	int
order_id	int
pizza_id	text
quantity	int

pizza_orders

order_id	int
date	text
time	text

pizza_types

pizza_type_id	text
name	text
category	text
ingredients	text

pizzas

pizza_id	text
pizza_type_id	text
size	text
price	double

```
1 #6 Query The Total Revenue Generated ? #817860
2 SELECT ROUND(SUM(A.PRICE * B.quantity)) as TOTAL_REV
3   FROM Pizzas A JOIN PIZZA_ORDER_DETAILS B
4     ON A.PIZZA_ID = B.PIZZA_ID;
5
6 #7 Query The Highest Priced Pizza? #Display Pizza Name, Price
7 SELECT A.NAME , MAX(B.PRICE) as P_PRICE
8       FROM pizzas JOIN pizza_types A JOIN pizzas B
9         ON A.pizza_type_id = B.pizza_type_id
10        GROUP BY NAME ;
11
12 #8 Query The Lowest Priced Pizza? #Display Pizza Name, Price
13
14 SELECT A.NAME , MIN(B.PRICE) as P_PRICE
15       FROM pizzas JOIN pizza_types A JOIN pizzas B
16         ON A.pizza_type_id = B.pizza_type_id
17        GROUP BY NAME ;
```

Save

TOTAL_REV

1 817860

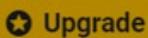
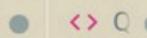
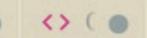


File Edit View Tools Help

localhost:3306/pizza - Beekeeper Studio - Free Version



pizza



Upgrade

Filter



ENTITIES 4



pizza_order_details

order_details_id	int
order_id	int
pizza_id	text
quantity	int

pizza_orders

order_id	int
date	text
time	text

pizza_types

pizza_type_id	text
name	text
category	text
ingredients	text

pizzas

pizza_id	text
pizza_type_id	text
size	text
price	double

```
1 #9 Query The Pizza Ordered Qty By Size?
2
3
4
5 WITH CTE1 as ( SELECT A.SIZE , SUM(B.quantity) as TTL_QTY
6                      FROM pizzas A JOIN pizza_order_details B
7                      ON A.PIZZA_ID = B.PIZZA_ID
8                      GROUP BY SIZE )
9
10          SELECT * FROM CTE1;
```

	SIZE	TTL_QTY
1	M	15635
2	L	18956
3	S	14403
4	XL	552
5	XXL	28

Save

Run

localhost:3306/pizza

mysql

5

0 affected

211ms

Download



File Edit View Tools Help localhost:3306/pizza - Beekeeper Studio - Free Version

Filter

ENTITIES 4

- pizza_order_details
 - order_details_id int
 - order_id int
 - pizza_id text
 - quantity int
- pizza_orders
 - order_id int
 - date text
 - time text
- pizza_types
 - pizza_type_id text
 - name text
 - category text
 - ingredients text
- pizzas
 - pizza_id text
 - pizza_type_id text
 - size text
 - price double

#10 Query The Pizza Ordered Qty By Category?

```
1 #10 Query The Pizza Ordered Qty By Category?
2
3
4
5 WITH CTE1 as ( SELECT A.CATEGORY, B.PIZZA_ID
6                 FROM pizza_types A JOIN pizzas B
7                   ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID ),
8          CTE2 as ( SELECT CTE1.CATEGORY , SUM(C.quantity) TTL_QTY
9                 FROM CTE1 JOIN PIZZA_ORDER_DETAILS C
10                ON CTE1.PIZZA_ID = C.PIZZA_ID
11               GROUP BY CATEGORY )
12      SELECT * FROM CTE2
```

Save Run ▾

	CATEGORY	TTL_QTY
1	Classic	14888
2	Veggie	11649
3	Supreme	11987
4	Chicken	11050

localhost:3306/pizza mysql 4 0 affected 233ms Download ▾



pizza

Filter



ENTITIES 4



pizza_order_details

order_details_id
order_id
pizza_id
quantityint
int
text
int

pizza_orders

order_id
date
timeint
text
text

pizza_types

pizza_type_id
name
category
ingredientstext
text
text
text

pizzas

pizza_id
pizza_type_id
size
pricetext
text
text
double

```
1 #11 Query The Pizza Ordered Qty By Pizza Name?
2
3
4
5 WITH CTE1 as ( SELECT A.NAME , B.PIZZA_ID
6                 FROM pizza_types A JOIN pizzas B
7                   ON A.PIZZA_TYPE_ID = B.pizza_type_id ),
8
9             CTE2 as ( SELECT CTE1.NAME , SUM(C.quantity) TTL_ORD_QTY
10                  FROM CTE1 JOIN PIZZA_ORDER_DETAILS C
11                    ON CTE1.PIZZA_ID = C.PIZZA_ID
12                  GROUP BY NAME
13
14                  ORDER BY TTL_ORD_QTY DESC )
15
16
17 select * from CTE2
```

	NAME	TTL_ORD_QTY
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371
6	The California Chicken Pizza	2370

File Edit View Tools Help

localhost:3306/pizza - Beekeeper Studio - Free Version

pizza C + Filter ENTITIES 4 +

#12 Query The Top 7 Types Based Orders Qty? Display Pizza Name, Order Qty?

```

4 WITH CTE1 as ( SELECT A.PIZZA_TYPE_ID , A.NAME , B.PIZZA_ID
5      FROM pizza_types A JOIN pizzas B
6      ON A.pizza_type_id = B.pizza_type_id ),
7
8     CTE2 as ( SELECT CTE1.PIZZA_TYPE_ID,CTE1.NAME, SUM(C.quantity) as TTL_QTY
9            FROM CTE1 JOIN PIZZA_ORDER_DETAILS C
10           ON CTE1.PIZZA_ID = C.PIZZA_ID
11           GROUP BY PIZZA_TYPE_ID,NAME ),
12
13    CTE3 as ( SELECT * , RANK() OVER(ORDER BY TTL_QTY DESC ) as RNK FROM CTE2)
14
15   SELECT * FROM CTE3 WHERE RNK <=7;
16
17

```

Save Run ▾

	PIZZA_TYPE_ID	NAME	TTL_QTY	RNK
1	classic_dlx	The Classic Deluxe Pizza	2453	1
2	bbq_ckn	The Barbecue Chicken Pizza	2432	2
3	hawaiian	The Hawaiian Pizza	2422	3
4	pepperoni	The Pepperoni Pizza	2418	4
5	thai_ckn	The Thai Chicken Pizza	2371	5
6	cali_ckn	The California Chicken Pizza	2370	6

localhost:3306/pizza mysql 7 0 affected 384ms Download ▾

pizza

C +

<> C ● <> Q ● <> Q ● <> Q ● <> Q ●

+ Upgrade

Filter

▼

ENTITIES 4

+

pizza_order_details

order_details_id	int
order_id	int
pizza_id	text
quantity	int

pizza_orders

order_id	int
date	text
time	text

pizza_types

pizza_type_id	text
name	text
category	text
ingredients	text

pizzas

pizza_id	text
pizza_type_id	text
size	text
price	double

```
1 #13 Query The Distribution Of Orders By Hour Of Day?
2
3
4 WITH CTE as (SELECT HOUR(A.TIME) HH , SUM(B.quantity) as TOTAL_ORDERS_DISTRIBUTION
5     FROM pizza_orders A JOIN pizza_order_details B
6     ON A.order_id = B.order_id
7     GROUP BY HH
8     ORDER BY TOTAL_ORDERS_DISTRIBUTION DESC  )
9
10    SELECT * FROM CTE;
```

	HH	TOTAL_ORDERS_DISTRIBUTION
1	12	6776
2	13	6413
3	18	5417
4	17	5211
5	19	4406
6	16	4239

Save

Run ▾

File Edit View Tools Help

localhost:3306/pizza - Beekeeper Studio - Free Version

pizza C + Filter ENTITIES 4 +

#14 Query The Pizza Order Qty By Date And Calculate The Average Numbers Of Pizzas Ordered Per Day?

```

1 WITH CTE as ( SELECT A.DATE as DT , SUM(B.quantity) TOTAL_ORD_QTY
2   FROM PIZZA_ORDERS A JOIN PIZZA_ORDER_DETAILS B
3     ON A.ORDER_ID = B.ORDER_ID
4     GROUP BY DT ),
5
6   CTE2 as ( SELECT CTE.DT , AVG(TOTAL_ORD_QTY) as AVG_QTY
7     FROM CTE
8     GROUP BY DT )
9
10 Select CTE2.DT, CTE2.AVG_QTY,CTE.TOTAL_ORD_QTY From CTE2 ,CTE;
11
12
13
14
15
16
17

```

Save Run ▾

	DT	AVG_QTY	TOTAL_ORD_QTY
1	2015-12-31	178.0000	162
2	2015-12-30	82.0000	162
3	2015-12-29	80.0000	162
4	2015-12-28	102.0000	162
5	2015-12-27	89.0000	162
6	2015-12-26	95.0000	162

localhost:3306/pizza mysql 50000 / 128164 0 affected 827ms Download ▾

pizza

C +



Upgrade

Filter



ENTITIES 4



pizza_order_details

order_details_id	int
order_id	int
pizza_id	text
quantity	int

pizza_orders

order_id	int
date	text
time	text

pizza_types

pizza_type_id	text
name	text
category	text
ingredients	text

pizzas

pizza_id	text
pizza_type_id	text
size	text
price	double

```
1 #15 Query The Top 7 Pizza Names By Revenue? #Display Pizza Name, Revenue , Orderqty
2
3 WITH CTE1 as ( SELECT A.NAME , B.PIZZA_ID, B.PRICE
4                 FROM pizza_types A JOIN pizzas B
5                   ON A.PIZZA_TYPE_ID = B.PIZZA_TYPE_ID ),
6
7
8 CTE2 as ( SELECT CTE1.NAME , SUM(C.quantity) as TOTAL_ORD_QTY , SUM(CTE1.PRICE * C.quantity) as REVENUE
9                 FROM CTE1 JOIN PIZZA_ORDER_DETAILS C
10                ON CTE1.PIZZA_ID = C.PIZZA_ID
11              GROUP BY CTE1.NAME ),
12
13 CTE3 as (SELECT NAME, TOTAL_ORD_QTY , RANK() OVER(ORDER BY REVENUE DESC ) as RNK
14                 FROM CTE2)
15
16
17     SELECT * from CTE3 where rnk <=7;
```

Save

Run ▾

	NAME	TOTAL_ORD_QTY	RNK
1	The Thai Chicken Pizza	2371	1
2	The Barbecue Chicken Pizza	2432	2
3	The California Chicken Pizza	2370	3
4	The Classic Deluxe Pizza	2453	4
5	The Spicy Italian Pizza	1924	5
6	The Southwest Chicken Pizza	1917	6

pizza

C +



Upgrade

Filter



ENTITIES 4



pizza_order_details

order_details_id
order_id
pizza_id
quantityint
int
text
int

pizza_orders

order_id
date
timeint
text
text

pizza_types

pizza_type_id
name
category
ingredientstext
text
text
text

pizzas

pizza_id
pizza_type_id
size
pricetext
text
text
double

```
1 #16 Query The Percentage Contribution Of Each Pizza Category To Total Revenue?
2
3 WITH CTE1 as ( SELECT A.PIZZA_TYPE_ID , SUM(A.PRICE * B.quantity ) as Revenue
4                 FROM pizzas A JOIN pizza_order_details B
5                 ON A.pizza_id = B.pizza_id
6                 GROUP BY PIZZA_TYPE_ID ),
7
8     CTE2 as ( SELECT C.CATEGORY , SUM(CTE1.Revenue) as PER_DAY_REV
9                 FROM CTE1 JOIN pizza_types C
10                ON CTE1.PIZZA_TYPE_ID = C.PIZZA_TYPE_ID
11                GROUP BY C.CATEGORY ),
12
13    CTE3 as (SELECT ROUND(SUM(PER_DAY_REV),2) as TOTAL_REV FROM CTE2)
14
15    SELECT CTE2.CATEGORY, (PER_DAY_REV/TOTAL_REV)*100 as REVENUEPERCENTAGE FROM CTE3,CTE2 ;
```

Save

Run ▾

	CATEGORY	REVENUEPERCENTAGE
1	Classic	26.905960255669655
2	Veggie	23.682590927384293
3	Supreme	25.45631126009883
4	Chicken	23.955137556847287

File Edit View Tools Help

localhost:3306/pizza - Beekeeper Studio - Free Version

pizza C + Filter ENTITIES 4 +

#17 Analyse the cumulative revenue generated over time?

```

1 WITH CTE1 as ( select A.ORDER_ID , SUM(A.quantity * B.PRICE) as REV
2             FROM pizza_order_details A JOIN pizzas B
3             ON A.pizza_id = B.pizza_id
4             GROUP BY A.ORDER_ID ),
5
6             CTE2 as ( SELECT C.DATE , SUM(CTE1.REV) as TOTAL_REV
7             FROM CTE1 JOIN pizza_orders C
8             ON CTE1.ORDER_ID = C.order_id
9             GROUP BY C.DATE ),
10
11             CTE3 as ( SELECT CTE2.DATE, SUM(CTE2.TOTAL_REV) OVER(ORDER BY CTE2.DATE) as CUMM_REV
12             FROM CTE2 )
13
14             SELECT * FROM CTE3;
15
16
17

```

Save Run ▾

	DATE	CUMM_REV
1	2015-01-01	2713.8500000000004
2	2015-01-02	5445.75
3	2015-01-03	8108.15
4	2015-01-04	9863.6
5	2015-01-05	11929.55
6	2015-01-06	14358.5

localhost:3306/pizza mysql 358 0 affected 289ms Download ▾

File Edit View Tools Help

localhost:3306/pizza - Beekeeper Studio - Free Version

pizza

Filter

ENTITIES 4

- pizza_order_details**
 - order_details_id int
 - order_id int
 - pizza_id text
 - quantity int
- pizza_orders**
 - order_id int
 - date text
 - time text
- pizza_types**
 - pizza_type_id text
 - name text
 - category text
 - ingredients text
- pizzas**
 - pizza_id text
 - pizza_type_id text
 - size text
 - price double

#18 Determine The Top 3 Most Ordered Pizza Types Based On Revenue For Each Category?

```

1 With CTE1 as ( SELECT A.PIZZA_ID ,A.PIZZA_TYPE_ID , ROUND(SUM(A.PRICE * B.quantity )) REVENUE
2   FROM Pizzas A JOIN pizza_order_details B
3     ON A.PIZZA_ID = B.PIZZA_ID
4       GROUP BY PIZZA_ID ,PIZZA_TYPE_ID  ),
5
6   CTE2 as ( SELECT CTE1.PIZZA_ID ,CTE1.PIZZA_TYPE_ID, CTE1.REVENUE , C.CATEGORY
7     FROM CTE1 JOIN PIZZA_TYPES C
8       ON CTE1.PIZZA_TYPE_ID = C.PIZZA_TYPE_ID ),
9
10  CTE3 as (select CTE2.PIZZA_ID , CTE2.PIZZA_TYPE_ID ,CTE2.REVENUE ,CTE2.CATEGORY ,
11    RANK() over(PARTITION BY CATEGORY ORDER BY REVENUE DESC) as RNK
12      from cte2)
13
14  (select * from cte3 where RNK <=3);
15
16
17

```

Save Run ▾

	PIZZA_ID	PIZZA_TYPE_ID	REVENUE	CATEGORY	RNK
1	thai_ckn_l	thai_ckn	29258	Chicken	1
2	southw_ckn_l	southw_ckn	21082	Chicken	2
3	bbq_ckn_l	bbq_ckn	20584	Chicken	3
4	big_meat_s	big_meat	22968	Classic	1
5	classic_dlx_m	classic_dlx	18896	Classic	2
6	hawaiian_l	hawaiian	15164	Classic	3

localhost:3306/pizza mysql 12 0 affected 254ms Download ▾

Thanks For Watching

"If you find this post helpful, I'd appreciate it if you could like, share, and follow me for more updates and insights"



LIKE & SHARE



www.linkedin.com/in/subhapriya-sanki