

SQL Data Analysis for Pizza Sales

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return all records in the pizza table

*/

SELECT *

FROM

pizza;

Query		Query History		
1	/*			
2	Created By: Loaone Seleka			
3	Created Date: 21/02/2023			
4	Query description: Return all records in the pizza table			
5	*/			
6	SELECT *			
7	FROM			
8	pizza;			
9	/*			
Data output		Messages		
	pizza_id [PK] character varying (100)	pizza_type_id character varying (250)	pizza_size character varying (10)	pizza_price numeric
1	bbq_ckn_s	bbq_ckn	S	12.75
2	bbq_ckn_m	bbq_ckn	M	16.75
3	bbq_ckn_l	bbq_ckn	L	20.75
4	cali_ckn_s	cali_ckn	S	12.75
5	cali_ckn_m	cali_ckn	M	16.75
6	cali_ckn_l	cali_ckn	L	20.75
7	ckn_alfredo_s	ckn_alfredo	S	12.75
8	ckn_alfredo_m	ckn_alfredo	M	16.75
9	ckn_alfredo_l	ckn_alfredo	L	20.75
10	ckn_pesto_s	ckn_pesto	S	12.75
11	ckn_pesto_m	ckn_pesto	M	16.75
12	ckn_pesto_l	ckn_pesto	L	20.75
13	southw_ckn_s	southw_ckn	S	12.75
Total rows: 96 of 96		Query complete 00:00:00.196		

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the number of records in the pizza table

*/

SELECT

COUNT(*) as count_of_pizza

FROM

pizza;

Query

Query History

11

Created Date: 21/02/2023

12

Query description: Return the number of records in the pizza table

13

*/

14

SELECT

15

COUNT(*) as count_of_pizza

16

FROM

17

pizza;

18

/*

19

Created By: Loaone Seleka

Data output

Messages

Notifications

count_of_pizza

bigint

1

96

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return unique records for pizza size as well as the count distinct of the pizza size

*/

SELECT

DISTINCT(pizza_size) AS unique_pizza_size

FROM

pizza;

Query

Query History

19

Created By: Loaone Seleka

20

Created Date: 21/02/2023

21

Query description: Return unique records for pizza size as well as the count distinct of the pizza size

22

*/

23

SELECT

24

DISTINCT(pizza_size) AS unique_pizza_size

25

FROM

26

pizza;

27

28

SELECT

29

COUNT(DISTINCT pizza_size) AS count_of_unique_pizza_size

30

FROM

31

pizza;

Data output

Messages

Notifications

SELECT

COUNT(DISTINCT pizza_size) AS count_of_unique_pizza_size

FROM

pizza;

Query

Query History

19

Created By: Loaone Seleka

20

Created Date: 21/02/2023

21

Query description: Return unique records for pizza size as well as the count distinct of the pizza size

22

*/

23

SELECT

24

DISTINCT(pizza_size) AS unique_pizza_size

25

FROM

26

pizza;

27

28

SELECT

29

COUNT(DISTINCT pizza_size) AS count_of_unique_pizza_size

30

FROM

31

pizza;

Data output

Messages

Notifications

+

📄

▼

📋

🗑️

🗄️

⬇️

📈

count_of_unique_pizza_size

bigint

🔒

1

5

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return unique records for pizza size

*/

SELECT

 pizza_size,

 COUNT(pizza_size) as number_of_pizza_size

FROM

 pizza

GROUP BY

 pizza_size

ORDER BY

 number_of_pizza_size DESC;

Query	Query History
31	pizza;
32	/*
33	Created By: Loaone Seleka
34	Created Date: 21/02/2023
35	Query description: Return unique records for pizza size
36	*/
37	SELECT
38	pizza_size,
39	COUNT(pizza_size) as number_of_pizza_size
40	FROM
41	pizza
42	GROUP BY
43	pizza_size
44	ORDER BY
45	number_of_pizza_size DESC;
46	/*

Data output	Messages	Notifications
<div><div>+</div><div>File</div><div>Dropdown</div><div>Copy</div><div>Delete</div><div>Database</div><div>Download</div><div>Refresh</div></div>		
	<div><div>pizza_size</div><div>character varying (10)</div><div>lock</div></div>	<div><div>number_of_pizza_size</div><div>bigint</div><div>lock</div></div>
1	S	32
2	M	31
3	L	31
4	XXL	1
5	XL	1

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the aggregation pf pizza price for each pizza size

*/

SELECT

 pizza_size,
 SUM(pizza_price) AS total_pizza_rprice,
 ROUND(AVG(pizza_price),2) AS average_pizza_price,
 MIN(pizza_price) AS minimum_pizza_price,
 MAX(pizza_price) AS maximum_pizza_price,
 COUNT(pizza_price) AS count_of_pizza_price

FROM

 pizza

GROUP BY

 pizza_size

ORDER BY

 pizza_size ASC;

Query

Query History

46

/*

47

Created By: Loaone Seleka

48

Created Date: 21/02/2023

49

Query description: Return the aggregation pf pizza price for each pizza size

50

*/

51

SELECT

52

pizza_size,

53

SUM(pizza_price) AS total_pizza_rprice,

54

ROUND(AVG(pizza_price),2) AS average_pizza_price,

55

MIN(pizza_price) AS minimum_pizza_price,

56

MAX(pizza_price) AS maximum_pizza_price,

57

COUNT(pizza_price) AS count_of_pizza_price

58

FROM

59

pizza

60

GROUP BY

61

pizza_size

62

ORDER BY

63

pizza_size ASC;

Data output

Messages

Notifications

	pizza_size character varying (10)	total_pizza_rprice numeric	average_pizza_price numeric	minimum_pizza_price numeric	maximum_pizza_price numeric	count_of_pizza_price bigint
1	L	620.70	20.02	15.25	21	31
2	M	495.75	15.99	12.5	16.75	31
3	S	400.40	12.51	9.75	23.65	32
4	XL	25.5	25.50	25.5	25.5	1
5	XXL	35.95	35.95	35.95	35.95	1

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the pizza name and ingredients where ingredeients does not include tomatoes and category is chicken

*/

SELECT

 pizza_name,

 pizza_ingredients,

 COUNT(pizza_category) AS number_of_pizza_category

FROM pizza_type

WHERE

 pizza_category = 'Chicken'

AND

 pizza_ingredients NOT LIKE '%Tomatoes%'

GROUP BY pizza_name, pizza_ingredients

ORDER BY number_of_pizza_category DESC;

Query Query History

```
62 ORDER BY
63     pizza_size ASC;
64 /*
65 Created By: Loaone Seleka
66 Created Date: 21/02/2023
67 Query description: Return the pizza name and ingredients where ingredeients does not include tomatoes and category is chicken
68 */
69 SELECT
70     pizza_name,
71     pizza_ingredients,
72     COUNT(pizza_category) AS number_of_pizza_category
73 FROM pizza_type
74 WHERE
75     pizza_category = 'Chicken'
76 AND
77     pizza_ingredients NOT LIKE '%Tomatoes%'
78 GROUP BY pizza_name, pizza_ingredients
79 ORDER BY number_of_pizza_category DESC;
```

Data output Messages Notifications

	pizza_name text	pizza_ingredients text	number_of_pizza_category bigint
1	The Californ...	Chicken, Artichok...	1
2	The Chicken...	Chicken, Red Oni...	1

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Combining the date and time columns to be one column in the orders table

*/

SELECT

order_id,

order_date,

order_time,

CONCAT(order_date,' ',order_time) AS order_date_time

FROM orders

LIMIT 10;

Query

Query History

81

Created By: Loaone Seleka

82

Created Date: 21/02/2023

83

Query description: Combining the date and time columns to be one column in the orders table

84

*/

85

SELECT

86

order_id,

87

order_date,

88

order_time,

89

CONCAT(order_date,' ',order_time) AS order_date_time

90

FROM orders

91

LIMIT 10;

Data output

M Loading...fications

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the pizza name. size and price of pizza ordering them by the highest pizza

*/

SELECT

pt.pizza_name,

pi.pizza_size,

pi.pizza_price

FROM

pizza pi

LEFT JOIN pizza_type pt

ON pt.pizza_type_id = pi.pizza_type_id

ORDER BY

pi.pizza_price DESC;

Query

Query History

93

/*

94

Created By: Loaone Seleka

95

Created Date: 21/02/2023

96

Query description: Return the pizza name. size and price of pizza ordering them by the highest pizza

97

*/

98

SELECT

99

pt.pizza_name,

100

pi.pizza_size,

101

pi.pizza_price

102

FROM

103

pizza pi

104

LEFT JOIN pizza_type pt

105

ON pt.pizza_type_id = pi.pizza_type_id

106

ORDER BY

107

pi.pizza_price DESC;

108

Data output

Messages

Notifications

	pizza_name text	pizza_size character varying (10)	pizza_price numeric
1	The Greek Pizza	XXL	35.95
2	The Greek Pizza	XL	25.5
3	The Brie Carre Pizza	S	23.65
4	The Italian Vegetables Pizza	L	21
5	The Spicy Italian Pizza	L	20.75
6	The Chicken Alfredo Pizza	L	20.75
7	The Soppressata Pizza	L	20.75
8	The Chicken Pesto Pizza	L	20.75

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Checking if we have duplicate orders in the order table

*/

SELECT

order_id,

COUNT(order_id) AS count_orders

FROM orders

GROUP BY order_id

HAVING

COUNT(order_id) > 1;

The screenshot shows a SQL query editor with a query history pane on the left and a data output pane at the bottom. The query history pane lists the following queries:

- 109 /*
- 110 Created By: Loaone Seleka
- 111 Created Date: 21/02/2023
- 112 Query description: Checking if we have duplicate orders in the order table
- 113 */
- 114 SELECT
- 115 order_id,
- 116 COUNT(order_id) AS count_orders
- 117 FROM orders
- 118 GROUP BY order_id
- 119 HAVING
- 120 COUNT(order_id) > 1;

The data output pane shows the following columns:

order_id	count_orders
[PK] integer	bigint

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Returning no null values from the pizza ingredients field

*/

SELECT

 COUNT(pizza_ingredients) AS count_of_not_null_ingredients

FROM pizza_type

WHERE

 pizza_ingredients IS NOT NULL;

The screenshot shows a database query editor interface. At the top, there are tabs for 'Query' and 'Query History'. The 'Query' tab is active, displaying a SQL query. The query is as follows:

```
122  /*
123  Created By: Loaone Seleka
124  Created Date: 21/02/2023
125  Query description: Returning no null values from the pizza ingredients field
126  */
127  SELECT
128      COUNT(pizza_ingredients) AS count_of_not_null_ingredients
129  FROM pizza_type
130  WHERE
131      pizza_ingredients IS NOT NULL;
```

Below the query editor, there are tabs for 'Data output', 'Messages', and 'Notifications'. The 'Data output' tab is active, showing a table with the results of the query. The table has two columns: 'count_of_not_null_ingredients' (type: bigint) and a value of 32.

count_of_not_null_ingredients
32

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return top 10 pizza which were ordered the most

*/

SELECT

 pt.pizza_name,

 pi.pizza_size,

 COUNT(od.quantity) AS number_of_orders

FROM

 pizza pi

INNER JOIN pizza_type pt

ON pt.pizza_type_id = pi.pizza_type_id

INNER JOIN order_details od

ON od.pizza_id = pi.pizza_id

GROUP BY

pt.pizza_name,pi.pizza_size

ORDER BY

number_of_orders DESC

LIMIT 10;

Query

Query History

132

Created by: Loane Seteka

133

Created Date: 21/02/2023

134

Query description: Return top 10 pizza which were ordered the most

135

*/

136

SELECT

137

pt.pizza_name,

138

pi.pizza_size,

139

COUNT(od.quantity) AS number_of_orders

140

FROM

141

pizza pi

142

INNER JOIN pizza_type pt

143

ON pt.pizza_type_id = pi.pizza_type_id

144

INNER JOIN order_details od

145

ON od.pizza_id = pi.pizza_id

146

GROUP BY

147

pt.pizza_name,pi.pizza_size

148

ORDER BY

Data output

Messages

Notifications



	<div>pizza_name</div> <div>text</div>	<div>pizza_size</div> <div>character varying (10)</div>	<div>number_of_orders</div> <div>bigint</div>
1	The Big Meat Pizza	S	1811
2	The Thai Chicken Pizza	L	1365
3	The Five Cheese Pizza	L	1359
4	The Four Cheese Pizza	L	1273
5	The Classic Deluxe Pizza	M	1159
6	The Spicy Italian Pizza	L	1088
7	The Hawaiian Pizza	S	1001

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return chicken pizza category total price between 2015-01-15 and 2015-01-31 for

pizza price greater

than 100

*/

SELECT

pt.pizza_name,
pt.pizza_category,
pi.pizza_size,
ord.order_date,
SUM(pi.pizza_price) AS total_pizza_price

FROM pizza pi

LEFT JOIN order_details od

ON od.pizza_id = pi.pizza_id

LEFT JOIN orders ord

ON ord.order_id = od.order_id

LEFT JOIN pizza_type pt

ON pt.pizza_type_id = pi.pizza_type_id

WHERE

pt.pizza_category = 'Chicken'

AND

(ord.order_date BETWEEN '2015-01-15' AND '2015-01-31')

GROUP BY pt.pizza_name,pt.pizza_category,pi.pizza_size,ord.order_date

HAVING SUM(pi.pizza_price) >= 100

ORDER BY total_pizza_price DESC;

Query

Query History

```

151  /*
152  Created By: Loaone Seleka
153  Created Date: 21/02/2023
154  Query description: Return chicken pizza category total price between 2015-01-15 and 2015-01-31 for pizza price greater
155  than 100
156  */
157  SELECT
158      pt.pizza_name,
159      pt.pizza_category,
160      pi.pizza_size,
161      ord.order_date,
162      SUM(pi.pizza_price) AS total_pizza_price
163  FROM pizza pi
164  LEFT JOIN order_details od
165  ON od.pizza_id = pi.pizza_id
166  LEFT JOIN orders ord

```

Data output

Messages

Notifications

	pizza_name text	pizza_category character varying (200)	pizza_size character varying (10)	order_date date	total_pizza_price numeric
1	The Barbecue Chicken Pizza	Chicken	L	2015-01-28	166.00
2	The Barbecue Chicken Pizza	Chicken	L	2015-01-17	145.25
3	The Thai Chicken Pizza	Chicken	L	2015-01-17	124.50
4	The Thai Chicken Pizza	Chicken	L	2015-01-21	124.50
5	The California Chicken Pizza	Chicken	L	2015-01-17	103.75
6	The California Chicken Pizza	Chicken	L	2015-01-20	103.75
7	The Southwest Chicken Piz...	Chicken	L	2015-01-16	103.75

Total rows: 13 of 13

Query complete 00:00:00 097

Query

Query History

```

178 Created By: Loaone Seleka
179 Created Date: 21/02/2023
180 Query description: Categorizing quantity by Low, Medium and High based on the orders
181 than 100
182 */
183 SELECT
184     order_id,
185     quantity,
186     CASE
187         WHEN quantity = 1 THEN 'Low'
188         WHEN quantity = 2 THEN 'Medium'
189         WHEN quantity >= 3 THEN 'High'
190     END AS quantity_level
191 FROM order_details
192 ORDER BY order_details DESC;
193 /*
  
```

Data output

Messages

Notifications

+

📄

▼

📋

🗑️

🔄

⬇️

📈

	order_id integer	quantity integer	quantity_level text
1	21350	1	Low
2	21349	1	Low
3	21348	1	Low
4	21348	1	Low
5	21348	1	Low
6	21347	1	Low
7	21347	1	Low

Total rows: 1000 of 48620

Query complete 00:00:00.340

```
/*
```

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the average price of pizza size less than the actual pizza price than 100

```
*/
```

```
SELECT
```

```
    pizza_id,
```

```
    pizza_size,
```

```
    pizza_price
```

```
FROM pizza
```

```
WHERE
```

```
    pizza_price <
```

```
(SELECT ROUND(AVG(pizza_price),2)FROM pizza)
```

```
ORDER BY
```

```
    pizza_price DESC;
```

Query Query History

```
193  /*
194  Created By: Loaone Seleka
195  Created Date: 21/02/2023
196  Query description: Return the average price of pizza size less than the actual pizza price
197  than 100
198  */
199  SELECT
200      pizza_id,
201      pizza_size,
202      pizza_price
203  FROM pizza
204  WHERE
205      pizza_price <
206      (SELECT ROUND(AVG(pizza_price),2) FROM pizza)
207  ORDER BY
208      pizza_price DESC;
```

Data output Messages Notifications



	pizza_id [PK] character varying (100)	pizza_size character varying (10)	pizza_price numeric
1	sicilian_m	M	16.25
2	calabrese_m	M	16.25
3	mediterraneo_m	M	16
4	veggie_veg_m	M	16
5	big_meat_m	M	16
6	spinach_fet_m	M	16
7	classic_dlx_m	M	16

Total rows: 49 of 49 Query complete 00:00:00.087

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return the average price of pizza size less than the actual pizza price

*/

SELECT

 pizza_size,

 AVG(pizza_price) AS average_pizza_price,

 (SELECT ROUND(AVG(pizza_price),2) FROM pizza) AS global_average_pizza_price

FROM pizza

GROUP BY pizza_size

ORDER BY

 average_pizza_price DESC;

Query	Query History
208	pizza_price DESC;
209	/*
210	Created By: Loaone Seleka
211	Created Date: 21/02/2023
212	Query description: Return the average price of pizza size less than the actual pizza price
213	*/
214	SELECT
215	pizza_size,
216	AVG(pizza_price) AS average_pizza_price,
217	(SELECT ROUND(AVG(pizza_price),2) FROM pizza) AS global_average_pizza_price
218	FROM pizza
219	GROUP BY pizza_size
220	ORDER BY
221	average_pizza_price DESC;

Data output

Messages

Notifications

	<div>pizza_size</div> <div>character varying (10)</div>	<div>average_pizza_price</div> <div>numeric</div>	<div>global_average_pizza_price</div> <div>numeric</div>
1	XXL	35.9500000000000000	16.44
2	XL	25.5000000000000000	16.44
3	L	20.0225806451612903	16.44
4	M	15.9919354838709677	16.44
5	S	12.5125000000000000	16.44

/*

Created By: Loaone Seleka

Created Date: 21/02/2023

Query description: Return order IDs and date where the quantity is greater or equal to 3

*/

SELECT

 order_id,

 order_date

FROM orders

WHERE

 order_id IN

 (SELECT order_id FROM order_details WHERE quantity >= 3)

ORDER BY

 order_date DESC;

Query

Query History

222

/*

223

Created By: Loaone Seleka

224

Created Date: 21/02/2023

225

Query description: Return order IDs and date where the quantity is greater or equal to 3

226

*/

227

SELECT

228

order_id,

229

order_date

230

FROM orders

231

WHERE

232

order_id IN

233

(SELECT order_id FROM order_details WHERE quantity >= 3)

234

ORDER BY

235

order_date DESC;

Data output

Messages

Notifications

	order_id [PK] integer	order_date date
1	21288	2015-12-31
2	19411	2015-11-27
3	18845	2015-11-18
4	18286	2015-11-08
5	17171	2015-10-18
6	17112	2015-10-17
7	16619	2015-10-08
8	16429	2015-10-03
9	15898	2015-09-23

Total rows: 24 of 24

Query complete 00:00:00.069