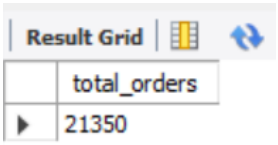
SQL PROJECT – PIZZA SALES ANALYSIS

Query with Output

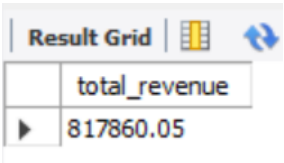
1) Retrieve the total number of orders placed.

```
SELECT
COUNT(*) AS total_orders
FROM
pizza_hut.orders;
```



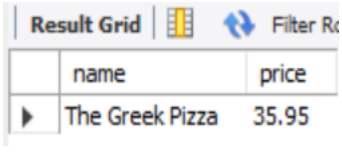
2) Calculate the total revenue generated from pizza sales.

```
SELECT
   ROUND(SUM(o.quantity * p.price), 2) AS total_revenue
FROM
   pizza_hut.pizzas p
      JOIN
   pizza_hut.order_details o ON p.pizza_id = o.pizza_id;
```



3) Identify the highest-priced pizza.

```
SELECT
   name, price
FROM
   pizza_hut.pizza_types p1
      JOIN
   pizza_hut.pizzas p2 ON p1.pizza_type_id = p2.pizza_type_id
```



```
WHERE
    price = (SELECT
            MAX(price)
        FROM
            pizza_hut.pizzas);
```

4) Identify the most common pizza size ordered.

```
SELECT
    size, COUNT(order_details_id) AS order_count
FROM
    pizza_hut.order_details o
        JOIN
    pizza_hut.pizzas p ON o.pizza_id = p.pizza_id
GROUP BY size
ORDER BY order_count DESC;
```

Result Grid

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

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

```
SELECT
    name, SUM(quantity) AS quantities
FROM
    pizza_hut.order_details o
        JOIN
    pizza_hut.pizzas p ON o.pizza_id =
p.pizza_id
        JOIN
    pizza_hut.pizza_types p1 ON p1.pizza_type_id = p.pizza_type_id
GROUP BY name
```

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

```
ORDER BY quantities DESC
LIMIT 5;
```

6) Join the necessary tables to find the total quantity of each pizza category ordered.

```
SELECT
    SUM(quantity) AS quantities, category
FROM
    pizza_hut.pizzas p
        JOIN
    pizza_hut.pizza_types p1 ON p.pizza_type_id = p1.pizza_type_id
        JOIN
    pizza_hut.order_details o ON o.pizza_id = p.pizza_id
GROUP BY category
ORDER BY quantities DESC;
```

quantities	category
14888	Classic
11987	Supreme
11649	Veggie
11050	Chicken

7) Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(order_time) AS hours, COUNT(order_id) AS
order_count
FROM
    pizza_hut.orders
GROUP BY HOUR(order_time);
```

hours	order_count
11	1231
12	2520
13	2455
14	1472
15	1468

8) Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT
    category, COUNT(name) AS pizza
```

category	pizza
Chicken	6
Classic	8
Supreme	9
Veggie	9

```
FROM
    pizza_hut.pizza_types
GROUP BY category;
```

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

avg_pizza_order_per_day
162
165
158
106
125

```
with temp as (select order_date,sum(quantity) as quantities
from pizza_hut.order_details o
join pizza_hut.orders o1 on o.order_id = o1.order_id
group by order_date)

select round(quantities,0) as avg_pizza_order_per_day from temp;
```

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

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

```
SELECT
    name, SUM(quantity * price) AS revenue
FROM
    pizza_hut.order_details o
        JOIN
    pizza_hut.pizzas p ON o.pizza_id = p.pizza_id
        JOIN
    pizza_hut.pizza_types p1 ON p1.pizza_type_id = p.pizza_type_id
GROUP BY name
ORDER BY revenue DESC
LIMIT 3;
```

11) Calculate the percentage contribution of each pizza type to total revenue.

```
SELECT
```

```
    category,
    ROUND(SUM(quantity * price) / (SELECT
                SUM(quantity * price)
            FROM
                pizza_hut.pizzas p
                    JOIN
                pizza_hut.order_details o ON p.pizza_id = o.pizza_id) * 100,
            2) as revenue
FROM
    pizza_hut.pizza_types p
        JOIN
    pizza_hut.pizzas p1 ON p.pizza_type_id = p1.pizza_type_id
        JOIN
    pizza_hut.order_details o ON o.pizza_id = p1.pizza_id
GROUP BY category
ORDER BY revenue DESC;
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

12) Analyze the cumulative revenue generated over time.

```
with temp as (select order_date,sum(quantity*price) as revenue
from pizza_hut.orders o
join pizza_hut.order_details o1 on o.order_id =
o1.order_id
join pizza_hut.pizzas p on p.pizza_id = o1.pizza_id
group by order_date)

select order_date,sum(revenue) over(order by order_date) as cum_revenue
```

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

from temp;

13) Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
with temp as (select category,name,sum(price*quantity) as revenue
from pizza_hut.pizza_types p
join pizza_hut.pizzas p1 on p.pizza_type_id = p1.pizza_type_id
join pizza_hut.order_details o on o.pizza_id = p1.pizza_id
group by category,name
),temp2 as

(select category,name,revenue,rank()over(partition by category order by revenue desc) as rn
from temp)

select * from temp2 where rn <4
```

category	name	revenue	rn
Chicken	The Thai Chicken Pizza	43434.25	1
Chicken	The Barbecue Chicken Pizza	42768	2
Chicken	The California Chicken Pizza	41409.5	3
Classic	The Classic Deluxe Pizza	38180.5	1
Classic	The Hawaiian Pizza	32273.25	2