



PIZZA SALES PROJECT USING SQL



1.RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
select count(order_id) as total_orders from orders;
```

Result Grid	
	total_orders
>	21350

2.CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```
select round(sum(order_details.quantity * pizzas.price),2) as total_sales  
from order_details join pizzas  
on pizzas.pizza_id = order_details.pizza_id;
```

Result Grid	
	total_sales
▶	817860.05

3. IDENTIFY THE HIGHEST-PRICED PIZZA.

```
select pizza_types.name, pizzas.price  
from pizza_types join pizzas  
on pizza_types.pizza_type_id = pizzas.pizza_type_id  
order by pizzas.price desc limit 1;
```

Result Grid		Filter Row
	name	price
▶	The Greek Pizza	35.95

4.IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select pizzas.size, count(order_details.order_details_id) as order_count  
from pizzas join order_details  
on pizzas.pizza_id = order_details.pizza_id  
group by pizzas.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 pizza_types.name,  
sum(order_details.quantity) as quantity  
from pizza_types join pizzas  
on pizza_types.pizza_type_id = pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.name order by quantity desc limit 5;
```

	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

6.JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED

```
select pizza_types.category,  
sum(order_details.quantity) as quantity  
from pizza_types join pizzas  
on pizza_types.pizza_type_id = pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.category order by quantity desc;
```

	category	quantity
>	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

7.DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
select hour(order_time) as hour , count(order_id) as order_count from orders  
group by hour(order_time);
```

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642

8.JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
select category , count(name) from pizza_types group by category;
```

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

9.GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

```
select round(avg(quantity),0) as avg_pizza_ordered_per_day from  
(select orders.order_date,sum(order_details.quantity) as quantity from orders join  
order_details on orders.order_id = order_details.order_id group by orders.order_date) as order_quantity ;
```

Result Grid | Filter Rows:

	avg_pizza_ordered_per_day
▶	138

10.DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
select pizza_types.name,  
sum(order_details.quantity * pizzas.price) as revenue  
from pizza_types join pizzas  
on pizzas.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.name order by revenue desc limit 3 ;
```

Result Grid | Filter Rows:

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

11. CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```
select pizza_types.category, round(sum(order_details.quantity*pizzas.price) /  
    (select round(sum(order_details.quantity*pizzas.price),2) AS total_sales  
     from order_details join pizzas on pizzas.pizza_id = order_details.pizza_id) * 100,2)  
    as revenue |from pizza_types join pizzas  
on pizza_types.pizza_type_id = pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza types.category order by revenue desc;
```

Result Grid |

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

12. ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
select order_date,sum(revenue) over(order by order_date) as cum_revenue from
(select orders.order_date,
sum(order_details.quantity * pizzas.price) as revenue
from order_details join pizzas on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

	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
	2015-01-08	19399.05
	2015-01-09	21526.4
	2015-01-10	23990.350000000002

13.DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASEED ON REVENUE FOR EACH PIZZA CATEGORY

```
select name, revenue from (select category, name, revenue,
rank() over(partition by category order by revenue desc) as rn
from (select pizza_types.category, pizza_types.name,
sum((order_details.quantity)* pizzas.price) as revenue
from pizza_types join pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id
join order_details on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where rn<= 3;
```

	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 Italian Supreme Pizza	33476.75
	The Sicilian Pizza	30940.5
	The Four Cheese Pizza	32265.70000000065

THANK YOU!

