PIZZA-PREPA

PIZZA-SALES ANALYSIS USING (SQL)

ABOUT PROJECT

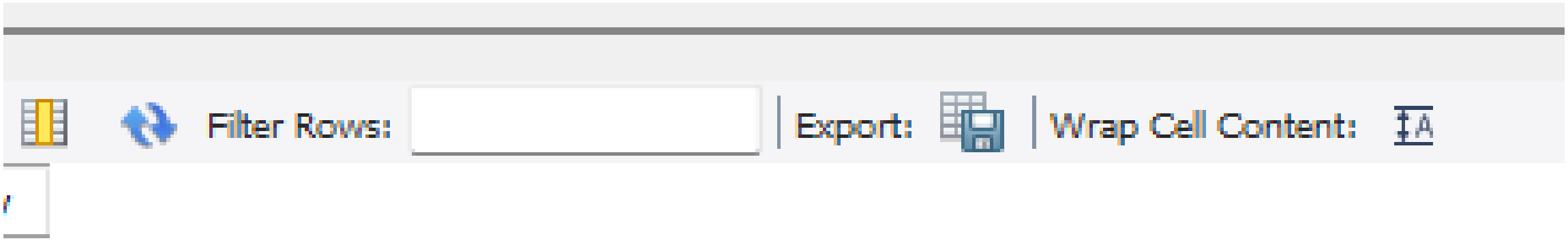
- This SQL Project is designed to analyze and retrieve various insights from a pizza prepa database. The database consists of multiple tables, including orders, order_details, pizzas, and pizza_types. The project involves writing queries to extract and analyze data at different levels of complexity

```sql
-- retrive the total no of ordr placed.
select count(order_id)as total_order from orders;
```

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

total_order
21350

```
-- calculate the total revenue generated fr
select sum(price)as total_rev from pizzas;
```



```
32    -- identify the heighest price of pizza
33
34 •  SELECT
35        *
36    FROM
37        pizzas
38    ORDER BY price DESC
39    LIMIT 1;
```

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

pizza_id	pizza_type_id	size	price
the_greek_xxl	the_greek	XXL	35.95

```sql
-- identify most common pizza size orderd
select count(size)as cont_most ,size from pizzas group by size limit 1;
```

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

cont_most	size
32	S

```sql
-- list the top 5 most order pizza_types along with the quantity
select * from order_details;

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

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

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

```sql
-- join the ness tables to find the total quantity of each pizza cat orders.
SELECT
    SUM(quantity) AS total_quantity, category AS pizza_category
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY category;
```

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

total_quantity	pizza_category
14888	Classic
11649	Veggie
11987	Supreme
11050	Chicken

```sql
-- determine the distribution of orders by hour of the day

select sum(quantity) as quant, hour(time) as hrs from order_details
join orders on order_details.order_id= orders.order_id
group by hrs order by hrs asc;

-- group the orders by date and cal Avg no of pizzas per day
```

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

quant	hrs
4	9
18	10
2728	11
6776	12
6413	13
3613	14
3216	15
4239	16
5211	17
5417	18
4406	19
3534	20
2545	21
1386	22
68	23

```sql
77    -- group the orders by date and cal Avg no of pizzas per day
78 •  select sum(quantity) as quant, date(date) as date from order_details
79    join orders on order_details.order_id= orders.order_id
80    group by date order by date asc;
```

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

quant	date
162	2015-01-01
165	2015-01-02
158	2015-01-03
106	2015-01-04
125	2015-01-05
147	2015-01-06
138	2015-01-07
173	2015-01-08
127	2015-01-09
146	2015-01-10
116	2015-01-11
119	2015-01-12
120	2015-01-13
150	2015-01-14
123	2015-01-15
158	2015-01-16
125	2015-01-17

Result 10

```sql
-- top 3 pizza based on revenue
select name, sum(price) as rev from pizzas
join pizza_types on pizzas.pizza_type_id=pizza_types.pizza_type_id
group by name order by rev desc limit 3;
```

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

name	rev
The Greek Pizza	109.95
The Italian Vegetables Pizza	50.5
The Barbecue Chicken Pizza	50.25

```sql
-- Calculate the percentage contribution of each pizza type to total revenue.
select pizza_types.category as category,
Round((sum(pizzas.price *order_details.quantity)/(select sum(pizzas.price * order_details.quantity)
from
pizzas
join
order_details on order_details.pizza_id = pizzas.pizza_id))*100,2)as percentage
from
pizza_types
join
pizzas on pizzas.pizza_type_id = pizza_types.pizza_type_id
join
order_details on pizzas.pizza_id = order_details.pizza_id
group by category

```

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

category	percentage
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96

```sql
-- analyze the cumulative revenue generated over time.
select order_date, round(sum(revenue)
over(order by order_date),1) as cmu_revenue from (select
orders.order_date,
sum(Pizzas.price*order_details.quantity) as revenue
from
pizzas
join
order_details on order_details.pizza_id = pizzas.pizza_id
join
orders on orders.order_id= order_details.order_id
group by order_date) as sales;
```

order_date	cmu_revenue
2015-01-01	2713.9
2015-01-02	5445.8
2015-01-03	8108.2
2015-01-04	9863.6
2015-01-05	11929.6

```sql
(select
 pizza_types.name as name,
 pizza_types.category as category,
 sum(pizzas.price*order_details.quantity) as revenue
 from
 pizza_types
 join pizzas on pizzas.pizza_type_id=pizza_types.pizza_type_id
 join order_details on pizzas.pizza_id=order_details.pizza_id
 group by name,category
 )as a) as b
 where rannk <=3;
```

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

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

THANK YOU