

PIZZA SALES SQL QUERIES

I. KPI's REQUIREMENT (Key performance indicator)

- a) **Total Revenue:** The sum of the total price of all pizza orders.

```
select sum(total_price) As total_revenue from pizza_sales;
```

	total_revenue
1	817860.05083847

- b) **Average order Value:** the average amount spent per order, calculated by dividing the total revenue by the total number of orders.

```
Select sum(total_price) / count(distinct order_id) as Avg_order_value from pizza_sales
```

100 %

Results Messages

	Avg_order_value
1	38.3072623343546

- c) **Total pizzas sold:** The sum of the quantities of all pizza sold

SQLQuery1.sql -...ASTER\masst (51))*

```
select sum(quantity) as total_pizzas_sold from pizza_sales
```

100 %

Results Messages

	total_pizzas_sold
1	49574

d) **Total orders:** The total number of orders placed

SQLQuery1.sql -...ASTER\masst (51))*

```

select * from pizza_sales
select count(distinct order_id) as Orders_Sold from pizza_sales

```

100 %

Results Messages

	pizza_id	order_id	pizza_name_id	quantity	order_date	order_time	unit_price	total_pric
1	1	1	hawaiian_m	1	2015-01-01	11:38:36.0000000	13.25	13.25
2	2	2	classic_dlx_m	1	2015-01-01	11:57:40.0000000	16	16

	Orders_Sold
1	21350

e) **Average pizzas per order:** The average number of pizzas sold per order, calculated by dividing the total number of pizzas sold by the total number of orders.

```

SELECT CAST(CAST(SUM(quantity) AS DECIMAL(10,2)) /
CAST(COUNT(DISTINCT order_id) AS DECIMAL(10,2)) AS
avg_pizzas_sold_per_order FROM pizza_sales;

```

	avg_pizzass_per_order
1	2.32

2. Daily Trend for Total Orders

```

SELECT DATENAME(DW,order_date) as order_day, count(distinct order_id) as
total_orders
from pizza_sales
group by datetime(DW, order_date)

```

Output:

	order_day	total_orders
1	Saturday	3158
2	Wednesday	3024
3	Monday	2794
4	Sunday	2624
5	Friday	3538
6	Thursday	3239
7	Tuesday	2973

3. Hourly Trend for Orders

```
select DATEPART(hour, order_time) as order_time, count(distinct order_id) as
total_orders
from pizza_sales
group by DATEPART(hour, order_time)
order by order_time
```

	order_time	total_orders
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28

```
select DATEPART(hour, order_time) as order_time, count(distinct order_id) as
total_orders
from pizza_sales
group by DATEPART(hour, order_time)
order by total_orders desc
```

	order_time	total_orders
1	12	2520
2	13	2455
3	18	2399
4	17	2336
5	19	2009
6	16	1920
7	20	1642
8	14	1472
9	15	1468
10	11	1231
11	21	1198
12	22	663
13	23	28
14	10	8
15	9	1

4. % of Sales by Pizza Category

```
select pizza_category, cast(sum(total_price) AS decimal(15,2)) as Total_sales,  
sum(total_price) * 100 / (select sum(total_price) from pizza_sales) as pct  
from pizza_sales  
group by pizza_category
```

	pizza_category	Total_sales	pct
1	Classic	220053.10	26.9059602306976
2	Chicken	195919.50	23.9551375322885
3	Veggie	193690.45	23.6825910258677
4	Supreme	208197.00	25.4563112111462

5. % of Sales by Pizza Size

```
select pizza_size, cast(sum(total_price) as decimal(15,2)) as Total_Sales,  
cast(sum(total_price) * 100 / (select cast(sum(total_price) as decimal(15,2)) from  
pizza_sales) as decimal(15,2)) as pct  
from pizza_sales  
group by pizza_size  
order by pizza_size
```

	pizza_size	Total_Sales	pct
1	L	375318.70	45.89
2	M	249382.25	30.49
3	S	178076.50	21.77
4	XL	14076.00	1.72
5	XXL	1006.60	0.12

6. Total Pizzas Sold by Pizza Category

```
select pizza_category, sum(quantity) as Total_Quantity
from pizza_sales
group by pizza_category
order by Total_Quantity desc
```

	pizza_category	Total_Quantity
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

7. Top 5 Best Sellers by Total Pizzas Sold

```
select pizza_size, cast(sum(total_price) as decimal(15,2)) as Total_Sales,
cast(sum(total_price) * 100 / (select cast(sum(total_price) as decimal(15,2)) from
pizza_sales) as decimal(15,2)) as pct
from pizza_sales
group by pizza_size
order by pizza_size
```

	pizza_category	Total_Quantity
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

8. Bottom 5 Best Sellers by Total Pizzas Sold

```
select top 5 pizza_name, sum(quantity) as Total_Pizzas_sold
from pizza_sales
group by pizza_name
order by Total_Pizzas_sold asc
```

	pizza_name	Total_Pizzas_sold
1	The Brie Carre Pizza	490
2	The Mediterranean Pizza	934
3	The Calabrese Pizza	937
4	The Spinach Supreme Pizza	950
5	The Soppressata Pizza	961

NOTE:

To apply the Month, Quarter, Week filters to the above queries you can use WHERE clause. Follow some of below examples

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders
FROM pizza_sales
WHERE MONTH(order_date) = 1
GROUP BY DATENAME(DW, order_date)
```

**Here MONTH(order_date) = 1 indicates that the output is for the month of January.
MONTH(order_date) = 4 indicates output for Month of April.*

```
SELECT DATENAME(DW, order_date) AS order_day, COUNT(DISTINCT order_id) AS total_orders
FROM pizza_sales
WHERE DATEPART(QUARTER, order_date) = 1
GROUP BY DATENAME(DW, order_date)
```

**Here DATEPART(QUARTER, order_date) = 1 indicates that the output is for the Quarter 1.
MONTH(order_date) = 3 indicates output for Quarter 3.*