SQL Queries Related To The Orders Of Pizzas

A. Key Performance Indicators :

1. Overall Earnings:

```
SELECT SUM(grand_total) AS Overall_Earnings from pizza_order_records;
```

	Overall_Earnings
1	817860.05083847

2. Mean Order Value:

```
SELECT SUM (grand_total) / COUNT(DISTINCT purchase_id) AS Mean_Order_Value FROM pizza_order_records;
```

	Mean_Order_Value
1	38.3072623343546

3. Cumulative Pizza Sold:

```
SELECT * FROM pizza_order_records;
SELECT SUM(quantity) AS Cumulative_Pizza_Sold from pizza_order_records;
```

	Cumulative_Pizza_Sold
1	49574

4. Total Number of Orders:

```
SELECT COUNT(DISTINCT purchase_id) AS Total_Number_of_Orders from pizza_order_records;
```

	Total_Number_of_Orders
1	21350

5. Average Pizzas Per Order:

```
SELECT * FROM pizza_order_records;
SELECT CAST(CAST(SUM(quantity) AS DECIMAL(10,5)) / CAST(COUNT(DISTINCT purchase_id) AS DECIMAL(10,5))
AS DECIMAL(10,2)) AS Average_Pizzas_per_order FROM pizza_order_records;
```

	Average_Pizzas_per_order
1	2.32

B. Hourly pattern of overall order count:

```
SELECT * FROM pizza_order_records;
SELECT DATEPART(HOUR, purchase_time) AS order_hours, COUNT(DISTINCT purchase_id) AS total_orders FROM pizza_order_records
GROUP BY DATEPART(HOUR, purchase_time) ORDER BY DATEPART(HOUR, purchase_time);
```

	order_hours	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

C. Daily pattern of overall order count:

```
SELECT * FROM pizza_order_records;
SELECT DATENAME(DW, purchase_date) AS order_day, COUNT(DISTINCT purchase_id)
AS total_orders FROM pizza_order_records GROUP BY DATENAME(DW, purchase_date);
```

	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

D. Percentage of Sales by Pizza Type:

```
SELECT * FROM pizza_order_records;
SELECT pizza_type, CAST(SUM(grand_total) AS DECIMAL(10,2)) as total_revenue,
CAST(SUM(grand_total) * 100 / (SELECT SUM(grand_total) from pizza_order_records) AS DECIMAL(10,2))
AS percentage_of_sale_by_pizza_type FROM pizza_order_records GROUP BY pizza_type;
```

	pizza_type	total_revenue	percentage_of_sale_by_pizza_type
1	Classic	220053.10	26.91
2	Chicken	195919.50	23.96
3	Veggie	193690.45	23.68
4	Supreme	208197.00	25.46

E. Percentage of Sales by Crust Size:

```
SELECT * FROM pizza_order_records;
SELECT crust_size, CAST(SUM(grand_total) AS DECIMAL(10,2)) as total_revenue,
CAST(SUM(grand_total) * 100 / (SELECT SUM(grand_total) from pizza_order_records) AS DECIMAL(10,2))
AS percentage_of_sale_by_crust_size FROM pizza_order_records GROUP BY crust_size ORDER BY crust_size;
```

	crust_size	total_revenue	percentage_of_sale_by_crust_size
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

F. Total Units Sold by Pizza Type:

```
SELECT * FROM pizza_order_records;
SELECT pizza_type, SUM(quantity) as Total_units_Sold FROM pizza_order_records
WHERE MONTH(purchase_date) = 2 GROUP BY pizza_type ORDER BY Total_units_Sold DESC;
```

	pizza_type	Total_units_Sold
1	Classic	1178
2	Supreme	964
3	Veggie	944
4	Chicken	875

G. Top 5 Best Selling Pizzas by Total Quantity Sold:

```
SELECT * FROM pizza_order_records;
SELECT TOP 5 pizza_name, SUM(quantity) AS Total_Units_Sold FROM pizza_order_records
GROUP BY pizza_name ORDER BY Total_Units_Sold ASC;
```

	pizza_name	Total_Units_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

H. Bottom 5 Worst Selling Pizzas by Total Quantity Sold:

```
SELECT * FROM pizza_order_records;
SELECT TOP 5 pizza_name, SUM(quantity) AS Total_Units_Sold FROM pizza_order_records
GROUP BY pizza_name ORDER BY Total_Units_Sold ASC;
```

	pizza_name	Total_Units_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

I. To Check the Dashboard Month Filter for February:

```
SELECT * FROM pizza_order_records;
SELECT DATENAME(DW, purchase_date) AS order_day, COUNT(DISTINCT purchase_id) AS total_orders
FROM pizza_order_records WHERE MONTH(purchase_date) = 2 GROUP BY DATENAME(DW, purchase_date);
```

	order_day	total_orders
1	Friday	281
2	Monday	220
3	Saturday	233
4	Sunday	232
5	Thursday	229
6	Tuesday	240
7	Wednesday	250

J. To Check the Dashboard Quarter Filter for Quarter-2 (April-June):

SQLQuery2.sql - DE...NB5JU9\krish (77))*

```
SELECT * FROM pizza_order_records;
SELECT DATENAME(DW, purchase_date) AS order_day, COUNT(DISTINCT purchase_id) AS total_orders
FROM pizza_order_records WHERE DATEPART(QUARTER, purchase_date) = 2 GROUP BY DATENAME(DW, purchase_date);
```

100 %

Results | Messages

	order_day	total_orders
1	Wednesday	764
2	Saturday	800
3	Monday	802
4	Sunday	670
5	Friday	905
6	Thursday	768
7	Tuesday	716