TASK-6

show databases;

use e_commerce_db;

#Extract monthname

- ➔ select monthname(order_date) as MonthsName from orders group by monthname(order_date);
- ➔ Alter table orders add column Month_Name varchar(25);
- ➔ update orders set Month_Name = monthname(order_date);
- ➔ select * from orders;

Output:

order_id	customer_id	order_date	Month_Name
1	51	2023-04-12	April
2	13	2023-10-24	October
3	58	2023-12-18	December
4	42	2023-09-19	September
5	22	2023-03-17	March
6	21	2023-04-20	April
7	2	2023-08-06	August
8	98	2023-01-31	January

By month()

- ➔ select month(o.order_date) as months,sum(p.amount) as Revneue from orders o inner join payments p on o.order_id=p.order_id group by month(order_date) order by months;

Output:

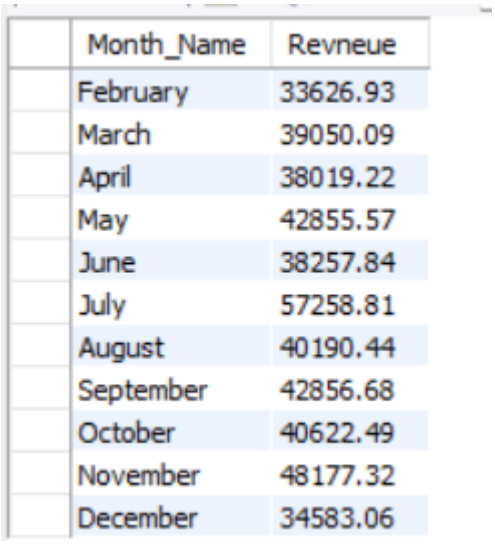
months	Revneue
1	51548.16
2	33626.93
3	39050.09
4	38019.22
5	42855.57
6	38257.84
7	57258.81
8	40190.44

By month Name

#code 1: by monthname()

➔ select monthname(o.order_date) as Month_Name,sum(p.amount) as Revneue from orders o inner join payments p on o.order_id=p.order_id group by monthname(order_date) ,month(order_date) order by month(order_date);

Output:

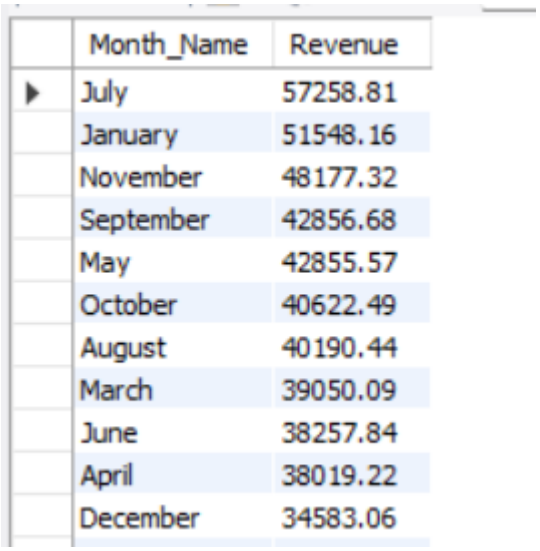


Month_Name	Revneue
February	33626.93
March	39050.09
April	38019.22
May	42855.57
June	38257.84
July	57258.81
August	40190.44
September	42856.68
October	40622.49
November	48177.32
December	34583.06

#Code 2 by Month_Name Column

➔ select o.Month_Name ,sum(p.amount) as Revenue from orders o inner join payments p on o.order_id=p.order_id group by o.Month_Name
order by Revenue desc;

Output:



Month_Name	Revenue
July	57258.81
January	51548.16
November	48177.32
September	42856.68
May	42855.57
October	40622.49
August	40190.44
March	39050.09
June	38257.84
April	38019.22
December	34583.06

By Years

➔ select year(o.order_date) as Years,sum(p.amount) as Revneue from orders o inner join payments p on o.order_id=p.order_id group by Year(order_date) order by Years;

Output:

	Years	Revneue
▶	2023	505079.10
	2024	1967.51

#By Weekdays Note: 0 = Monday, 1 = Tuesday, 2 = Wednesday, 3 = Thursday, 4 = Friday, 5 = Saturday, 6 = Sunday.

➔ select weekday(o.order_date) as Weaks,sum(p.amount) as Revneue from orders o inner join payments p on o.order_id=p.order_id
group by weekday(order_date)
order by Weaks;

Output:

	Weaks	Revneue
▶	0	77872.90
	1	78602.39
	2	68559.75
	3	68209.30
	4	74682.74
	5	70017.93
	6	69101.60

Using Extract()

➔ SELECT
EXTRACT(YEAR FROM o.order_date) AS order_year,
EXTRACT(MONTH FROM o.order_date) AS order_month,
SUM(p.amount) AS total_revenue,
COUNT(DISTINCT o.order_id) AS total_orders
FROM orders o
inner join payments p
on p.order_id=o.order_id
GROUP BY order_year, order_month
ORDER BY order_year, order_month;

Output:

order_year	order_month	total_revenue	total_orders
2023	1	49580.65	92
2023	2	33626.93	65
2023	3	39050.09	76
2023	4	38019.22	74
2023	5	42855.57	88
2023	6	38257.84	78
2023	7	57258.81	105
2023	8	40190.44	75
2023	9	42856.68	96
2023	10	40622.49	84
2023	11	48177.32	96
2023	12	34583.06	68
2024	1	1967.51	3

#without using Extract()

➔ select year(o.order_date) as Years, monthname(o.order_date) as Month_Name,sum(p.amount) as Revneue, count(distinct(o.order_id)) Total_order from orders o
inner join payments p
on o.order_id=p.order_id
group by Year(o.order_date),monthname(o.order_date)
order by Years, Month_Name;

Output:

Years	Month_Name	Revneue	Total_order
2023	April	38019.22	74
2023	August	40190.44	75
2023	December	34583.06	68
2023	February	33626.93	65
2023	January	49580.65	92
2023	July	57258.81	105
2023	June	38257.84	78
2023	March	39050.09	76
2023	May	42855.57	88
2023	November	48177.32	96
2023	October	40622.49	84
2023	September	42856.68	96
2024	January	1967.51	3

105