TASK 6: Sales Trend Analysis Using Aggregations

Objective: Analyse monthly revenue and order volume.
Tools: MySQL

Query 1:

```sql
-- Monthly Revenue and Order Volume
SELECT
    EXTRACT(YEAR FROM order_date) AS year,
    EXTRACT(MONTH FROM order_date) AS month,
    SUM(amount) AS total_revenue,
    COUNT(DISTINCT order_id) AS total_orders
FROM
    online_sales
GROUP BY
    year, month
ORDER BY
    year, month;
```

Output:

Result Grid | Filter Rows: Search | Export:

year	month	total_revenue	total_orders
2022	12	393585.15	68
2023	1	316977.80	62
2023	2	333066.80	63
2023	3	292558.83	63
2023	4	330777.32	71
2023	5	278043.65	63
2023	6	288009.59	62
2023	7	331191.59	62
2023	8	348786.60	68
2023	9	315076.98	62
2023	10	300663.67	55
2023	11	250951.91	46
2023	12	337708.82	70
2024	1	11608.01	2

Query 2:

```sql
-- Top 3 Months by Revenue--
SELECT
    EXTRACT(YEAR FROM order_date) AS year,
    EXTRACT(MONTH FROM order_date) AS month,
    SUM(amount) AS total_revenue
FROM
    online_sales
GROUP BY
    year, month
ORDER BY
    total_revenue DESC
LIMIT 3;
```

Output:

Result Grid | Filter Rows: Search | Export: | Fetch rows:

year	month	total_revenue
2022	10	403745.48
2022	12	393585.15
2022	1	389424.62

Query 3:

```sql
-- City-wise Revenue
SELECT
    city,
    SUM(amount) AS total_revenue,
    COUNT(DISTINCT order_id) AS total_orders
FROM
    online_sales
GROUP BY
    city
ORDER BY
    total_revenue DESC;

```

Output:

Result Grid | Filter Rows: Search | Export:

city	total_revenue	total_orders
Mumbai	1395237.86	279
Bangalore	1291162.00	248
Delhi	1287980.66	252
Hyderabad	1248044.09	242
Kolkata	1244931.96	242
Chennai	1168217.72	237

Query 4:

```sql
-- Category-wise Revenue
SELECT
    category,
    SUM(amount) AS total_revenue,
    COUNT(DISTINCT order_id) AS total_orders
FROM
    online_sales
GROUP BY
    category
ORDER BY
    total_revenue DESC;

```

Output:

Result Grid | Filter Rows: Search | Export:

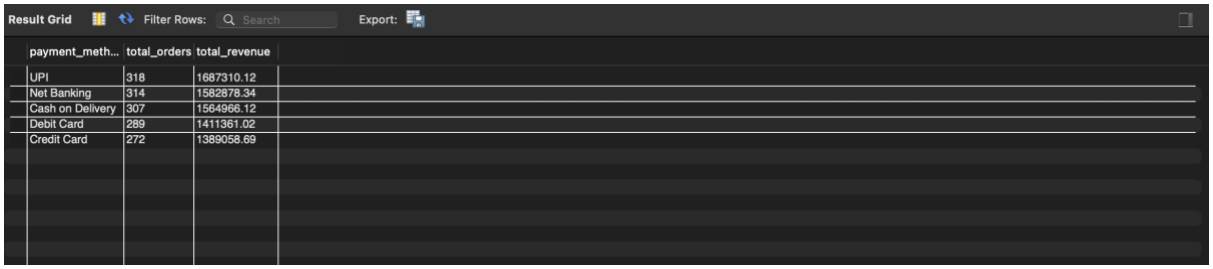
category	total_revenue	total_orders
Fashion	1644002.88	339
Grocery	1632149.42	314
Electronics	1507837.53	294
Home & Kitchen	1453990.67	268
Books	1397593.79	285

Query 5:

```sql
-- Payment Method Distribution
SELECT
    payment_method,
    COUNT(*) AS total_orders,
    SUM(amount) AS total_revenue
FROM
    online_sales
GROUP BY
    payment_method
ORDER BY
    total_revenue DESC;

```

Output:



payment_meth...	total_orders	total_revenue
UPI	318	1687310.12
Net Banking	314	1582878.34
Cash on Delivery	307	1564966.12
Debit Card	289	1411361.02
Credit Card	272	1389058.69

Query 6:

```sql
-- Customer Repeat Count
SELECT
    customer_id,
    COUNT(order_id) AS total_orders,
    SUM(amount) AS total_spent
FROM
    online_sales
GROUP BY
    customer_id
ORDER BY
    total_orders DESC
LIMIT 10;
```

Output:

customer_id	total_orders	total_spe...
10313	8	25779.60
10188	8	29010.07
10141	8	47915.31
10175	8	50752.70
10452	8	39549.91
10462	8	48372.41
10412	8	33822.51
10214	8	49538.38
10451	7	33830.37
10162	7	35206.24