

SQL TASK 5

```
--REVENUE AS PER PRODUCT ID  
SELECT  
    product_id,  
    SUM(amount) AS total_revenue  
FROM online_sales  
GROUP BY product_id  
ORDER BY total_revenue DESC;
```

	product_id	total_revenue
1	117	33773
2	118	32569
3	116	32565
4	106	30916
5	103	30711
6	105	28508
7	102	28176
8	119	27725
9	109	26854
10	101	26483
11	104	26366
12	112	25399
13	110	25208
14	114	25041
15	115	23718
16	111	23606

```
--TOTAL REVENUE
```

```
SELECT SUM(AMOUNT) AS TOTAL_REVENUE FROM online_sales;
```

	TOTAL_REVENUE
1	530155

```
--FOR VOLUME
```

```
SELECT  
COUNT(DISTINCT order_id) AS order_volume FROM online_sales;
```

	order_volume
1	1000

```
--monthly sales trend
```

```
SELECT
    YEAR(order_date) AS order_year,
    DATENAME(MONTH, order_date) AS order_month_name,
    COUNT(DISTINCT order_id) AS order_volume,
    SUM(amount) AS total_revenue
FROM online_sales
GROUP BY YEAR(order_date), DATENAME(MONTH, order_date), MONTH(order_date)
ORDER BY order_year, MONTH(order_date);
```

Results		Messages		
	order_year	order_month_name	order_volume	total_revenue
1	2023	January	46	24905
2	2023	February	58	29410
3	2023	March	47	26079
4	2023	April	37	20659
5	2023	May	42	22337
6	2023	June	42	20692
7	2023	July	50	24724
8	2023	August	36	18611
9	2023	September	33	16529
10	2023	October	41	19636
11	2023	November	33	17425
12	2023	December	44	22777
13	2024	January	44	23083
14	2024	February	41	20117
15	2024	March	46	26475
16	2024	April	42	21252

```
--product_id with the highest number of orders
```

```
SELECT TOP 1
    product_id,
    COUNT(*) AS order_count
FROM online_sales
GROUP BY product_id
ORDER BY order_count DESC;
```

Results		Messages	
	product_id	order_count	
1	118	62	

--Top 3 orders with the highest amount value:

```
SELECT TOP 3
    order_id,
    product_id,
    amount,
    order_date
FROM online_sales
ORDER BY amount DESC;
```

Results		Messages		
	order_id	product_id	amount	order_date
1	199	111	1000	2024-07-24
2	734	108	999	2023-01-18
3	960	105	996	2023-04-05