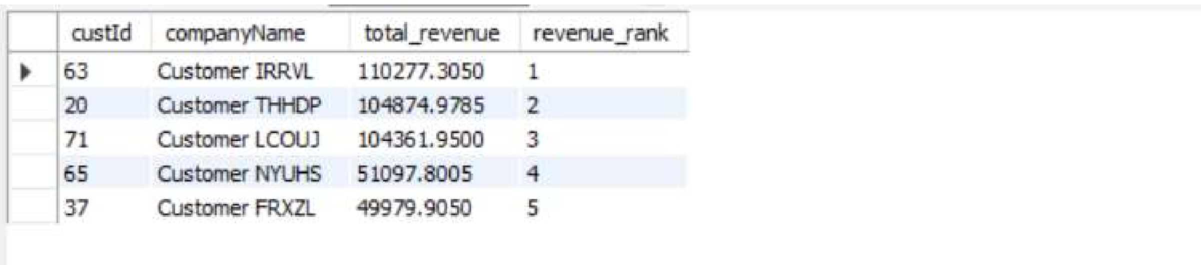
==

Module 1: Foundational Analysis

==

Task 1: Top 5 customers by revenue

Identify VIP customers who have spent the most overall. Useful for prioritization, targeted promotions, and designing loyalty programs

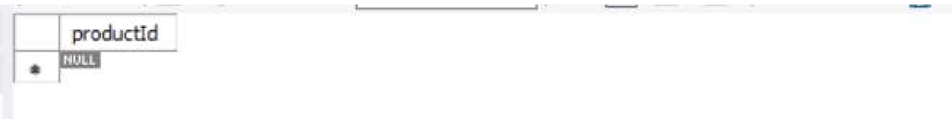


custId	companyName	total_revenue	revenue_rank
63	Customer IRRVL	110277.3050	1
20	Customer THHDP	104874.9785	2
71	Customer LCOUJ	104361.9500	3
65	Customer NYUHS	51097.8005	4
37	Customer FRXZL	49979.9050	5

Comment: A VIP list identified in this way helps you decide who to give early shipping, additional discounts, or a dedicated account manager. It's a quick basis for RFM segmentation. You can base retention campaigns and priority support programs on it.

Task 2: Products never ordered

Find products that were never ordered. Useful for clearance, renaming, pricing adjustments, or removing legacy items

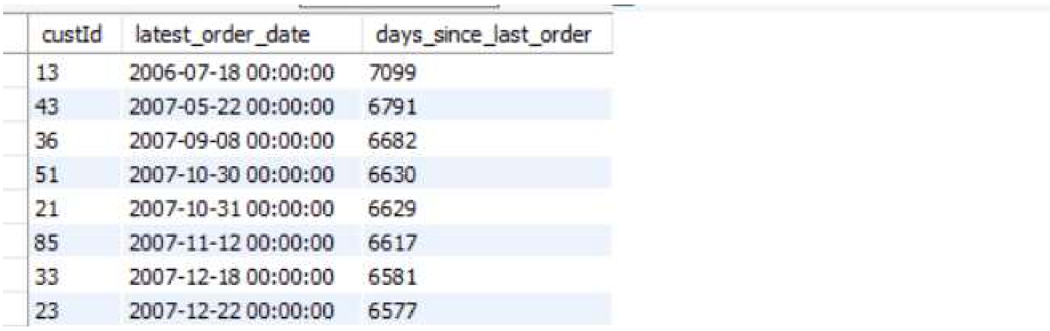


productId
NULL

Comment: The resulting list can be used for stock clearance and to free up tied capital in inventory.

Task 3: Customers with last order > 120 days ago

Identify customers who haven't purchased for a long period (120+ days) — a starting point for win-back campaigns

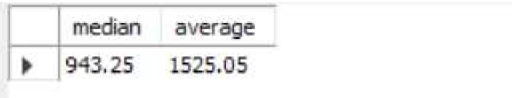


custId	latest_order_date	days_since_last_order
13	2006-07-18 00:00:00	7099
43	2007-05-22 00:00:00	6791
36	2007-09-08 00:00:00	6682
51	2007-10-30 00:00:00	6630
21	2007-10-31 00:00:00	6629
85	2007-11-12 00:00:00	6617
33	2007-12-18 00:00:00	6581
23	2007-12-22 00:00:00	6577

Comment: Such a list is a ready-made input for reactivation campaigns (emails, discounts).

Task 4: Average vs median order value

Compare average order value to median to understand the influence of outliers and set thresholds (e.g., free shipping).

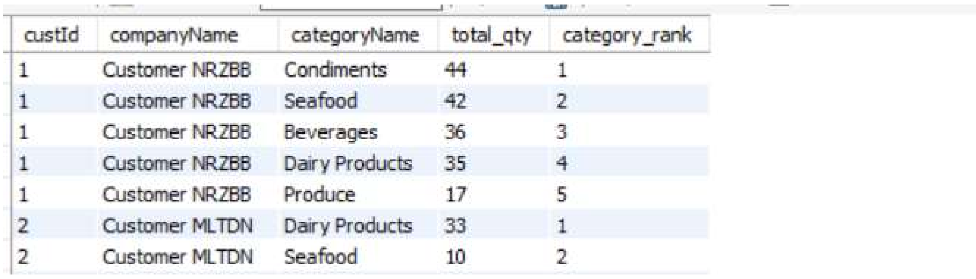


median	average
943.25	1525.05

Comment: Median provides a robust central tendency measure; using both median and mean helps set realistic KPIs and thresholds.

Task 5: Top product category per customer

Determine a customer's favorite product category by quantity purchased—valuable for personalization and recommendations



custId	companyName	categoryName	total_qty	category_rank
1	Customer NRZBB	Condiments	44	1
1	Customer NRZBB	Seafood	42	2
1	Customer NRZBB	Beverages	36	3
1	Customer NRZBB	Dairy Products	35	4
1	Customer NRZBB	Produce	17	5
2	Customer MLTDN	Dairy Products	33	1
2	Customer MLTDN	Seafood	10	2

Comment: Knowing customers' preferred categories helps improve CTR and conversion for recommendations and targeted campaigns.

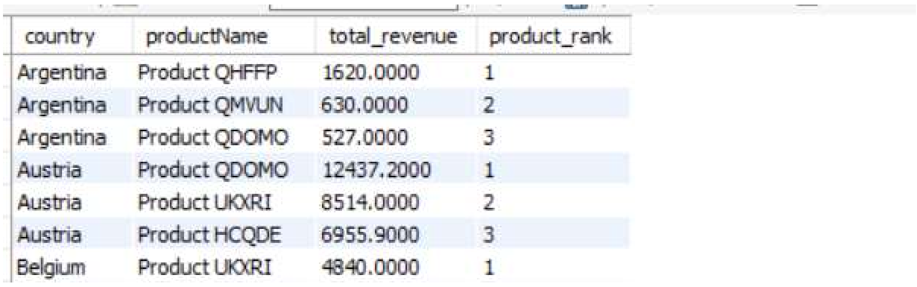
==

Module 2: Temporal & Statistical Analysis

==

Task 1: Top 3 products by revenue in each country

Identify local bestsellers to plan inventory and marketing per market

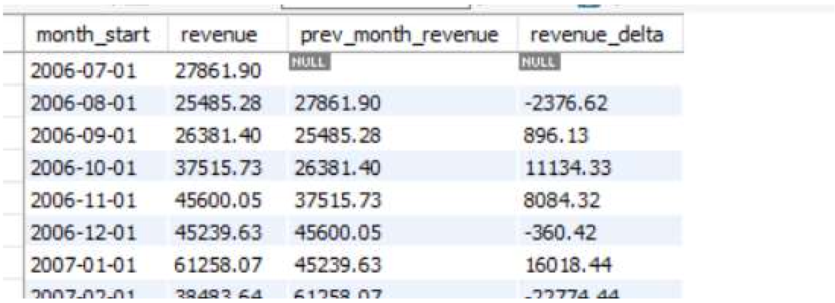


country	productName	total_revenue	product_rank
Argentina	Product QHFFP	1620.0000	1
Argentina	Product QMVUN	630.0000	2
Argentina	Product QDOMO	527.0000	3
Austria	Product QDOMO	12437.2000	1
Austria	Product UKXRI	8514.0000	2
Austria	Product HCQDE	6955.9000	3
Belgium	Product UKXRI	4840.0000	1

Comment: Use to identify and promote market leaders and secure stock for top products.

Task 2: Month-over-month revenue change

Observe month-on-month increases or decreases to react operationally

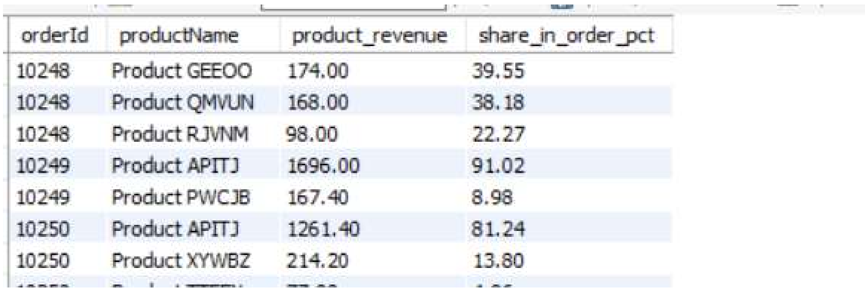


month_start	revenue	prev_month_revenue	revenue_delta
2006-07-01	27861.90	NULL	NULL
2006-08-01	25485.28	27861.90	-2376.62
2006-09-01	26381.40	25485.28	896.13
2006-10-01	37515.73	26381.40	11134.33
2006-11-01	45600.05	37515.73	8084.32
2006-12-01	45239.63	45600.05	-360.42
2007-01-01	61258.07	45239.63	16018.44
2007-02-01	38483.64	61258.07	-22774.44

Comment: Shows velocity of change to manage stock and advertising budgets.

Task 3: Product share in the basket

Determine what contributes to the basket value—key for cross-selling strategies

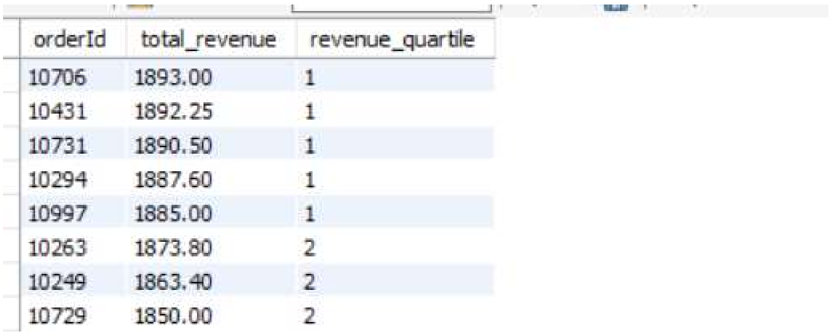


orderId	productName	product_revenue	share_in_order_pct
10248	Product GEEOO	174.00	39.55
10248	Product QMVUN	168.00	38.18
10248	Product RJVNM	98.00	22.27
10249	Product APITJ	1696.00	91.02
10249	Product PWCJB	167.40	8.98
10250	Product APITJ	1261.40	81.24
10250	Product XYWBZ	214.20	13.80
[illegible]	[illegible]	[illegible]	[illegible]

Comment: Useful for prioritizing cross-sell offers within the checkout flow.

Task 4: Order value quartiles (Quartile baskets)

Divide orders into quartiles to apply different operational or marketing policies

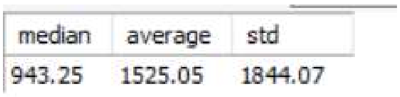


orderId	total_revenue	revenue_quartile
10706	1893.00	1
10431	1892.25	1
10731	1890.50	1
10294	1887.60	1
10997	1885.00	1
10263	1873.80	2
10249	1863.40	2
10729	1850.00	2

Comment: Example use—set free shipping thresholds differently per quartile.

Task 5: Mean, median, and standard deviation of order values

Understand typical order value and variability to segment customers and personalize offers

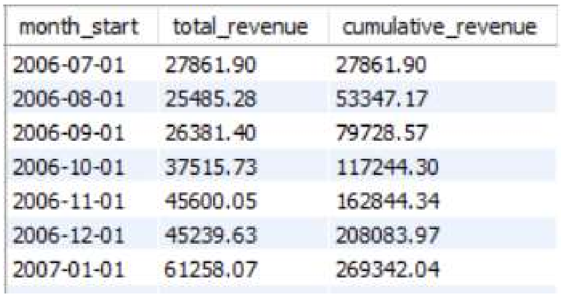


median	average	std
943.25	1525.05	1844.07

Comment: High variance suggests heterogeneous customer behavior—some have small baskets, others large. Use to tailor marketing and segmentation.

Task 6: Cumulative revenue over time

See how revenue accumulates over time to track progress towards goals

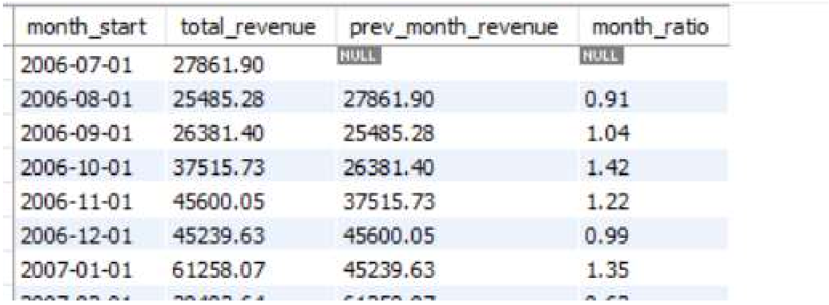


month_start	total_revenue	cumulative_revenue
2006-07-01	27861.90	27861.90
2006-08-01	25485.28	53347.17
2006-09-01	26381.40	79728.57
2006-10-01	37515.73	117244.30
2006-11-01	45600.05	162844.34
2006-12-01	45239.63	208083.97
2007-01-01	61258.07	269342.04

Comment: Useful to monitor when cumulative KPIs are met or exceeded.

Task 7: Month-over-month seasonality ratio

Detect seasonal patterns to predict demand and manage inventory

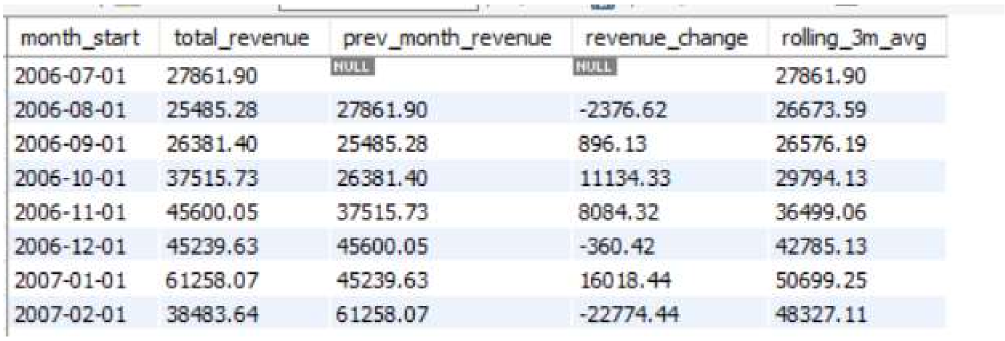


month_start	total_revenue	prev_month_revenue	month_ratio
2006-07-01	27861.90	NULL	NULL
2006-08-01	25485.28	27861.90	0.91
2006-09-01	26381.40	25485.28	1.04
2006-10-01	37515.73	26381.40	1.42
2006-11-01	45600.05	37515.73	1.22
2006-12-01	45239.63	45600.05	0.99
2007-01-01	61258.07	45239.63	1.35

Comment: Repeating patterns year-over-year confirm seasonality and inform stocking decisions.

Task 8: Revenue trend approximation (rolling average)

Approximate time trend to quickly understand if revenue is increasing, decreasing, or stable



month_start	total_revenue	prev_month_revenue	revenue_change	rolling_3m_avg
2006-07-01	27861.90	NULL	NULL	27861.90
2006-08-01	25485.28	27861.90	-2376.62	26673.59
2006-09-01	26381.40	25485.28	896.13	26576.19
2006-10-01	37515.73	26381.40	11134.33	29794.13
2006-11-01	45600.05	37515.73	8084.32	36499.06
2006-12-01	45239.63	45600.05	-360.42	42785.13
2007-01-01	61258.07	45239.63	16018.44	50699.25
2007-02-01	38483.64	61258.07	-22774.44	48327.11

Comment: Simple moving averages and differences provide quick trend signals for operational decisions.

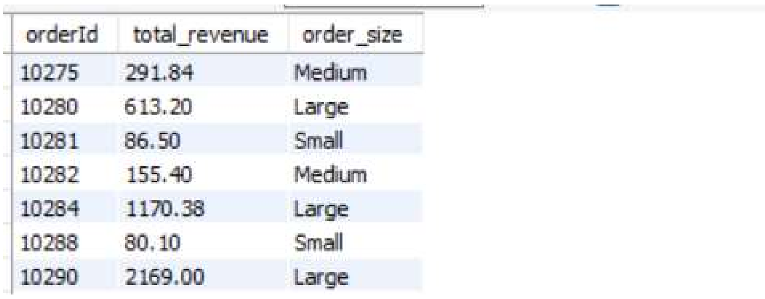
==

Module 3: Conditional Logic & Pivoting

==

Task 1: CASE Conditional Logic – Order Classification

Label orders as Small/Medium/Large by value to build segments for KPI reporting and marketing rules

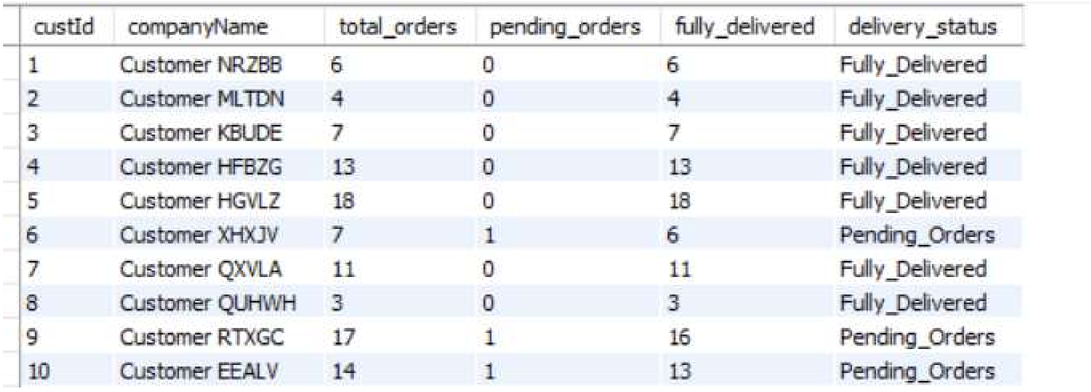


orderId	total_revenue	order_size
10275	291.84	Medium
10280	613.20	Large
10281	86.50	Small
10282	155.40	Medium
10284	1170.38	Large
10288	80.10	Small
10290	2169.00	Large

Comment: Classification allows building segment-specific KPIs and marketing actions (e.g., targeted campaigns for 'Large' orders).

Task 2: Order status / operational labeling with CASE

Monitor customers by delivery status to ensure SLAs and follow-up for pending shipments.

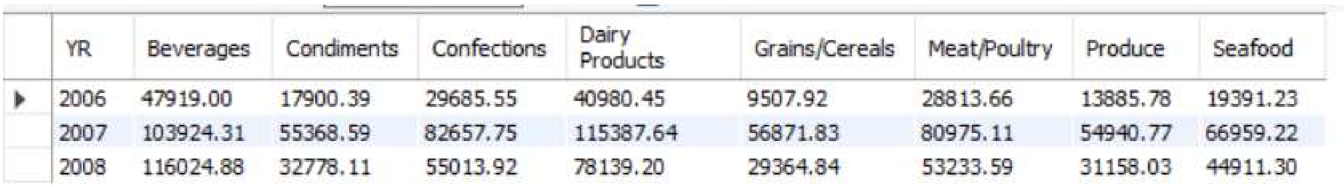


custId	companyName	total_orders	pending_orders	fully_delivered	delivery_status
1	Customer NRZBB	6	0	6	Fully_Delivered
2	Customer MLTDN	4	0	4	Fully_Delivered
3	Customer KBUDE	7	0	7	Fully_Delivered
4	Customer HFBZG	13	0	13	Fully_Delivered
5	Customer HGVLZ	18	0	18	Fully_Delivered
6	Customer XHXJV	7	1	6	Pending_Orders
7	Customer QXVLA	11	0	11	Fully_Delivered
8	Customer QUHWH	3	0	3	Fully_Delivered
9	Customer RTXGC	17	1	16	Pending_Orders
10	Customer EEALV	14	1	13	Pending_Orders

Comment: Useful for logistics dashboards and customer support prioritization.

Task 3: Pivot: revenue by category (conditional aggregation)

Build a pivot-like report showing category revenues as columns per year

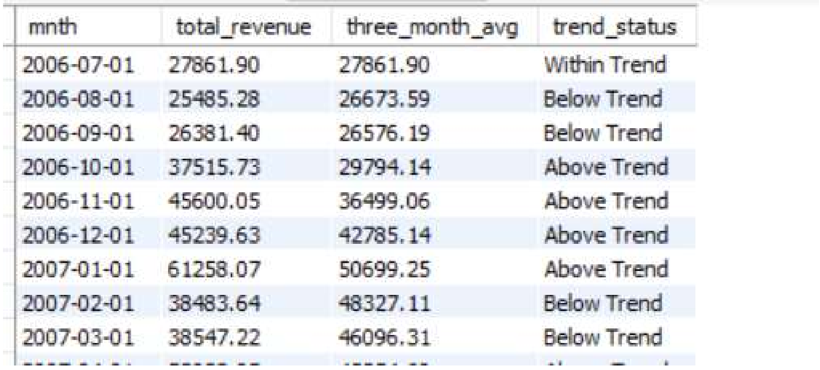


	YR	Beverages	Condiments	Confections	Dairy Products	Grains/Cereals	Meat/Poultry	Produce	Seafood
▶	2006	47919.00	17900.39	29685.55	40980.45	9507.92	28813.66	13885.78	19391.23
	2007	103924.31	55368.59	82657.75	115387.64	56871.83	80975.11	54940.77	66959.22
	2008	116024.88	32778.11	55013.92	78139.20	29364.84	53233.59	31158.03	44911.30

Comment: This pivot is useful when native pivot features are not available; great for export to CSV/Excel.

Task 4: Rolling Average + CASE – Trend Analysis

Mark months above/below 3-month average to identify trend shifts



mnth	total_revenue	three_month_avg	trend_status
2006-07-01	27861.90	27861.90	Within Trend
2006-08-01	25485.28	26673.59	Below Trend
2006-09-01	26381.40	26576.19	Below Trend
2006-10-01	37515.73	29794.14	Above Trend
2006-11-01	45600.05	36499.06	Above Trend
2006-12-01	45239.63	42785.14	Above Trend
2007-01-01	61258.07	50699.25	Above Trend
2007-02-01	38483.64	48327.11	Below Trend
2007-03-01	38547.22	46096.31	Below Trend

Comment: Helps highlight underperforming months for operational attention.

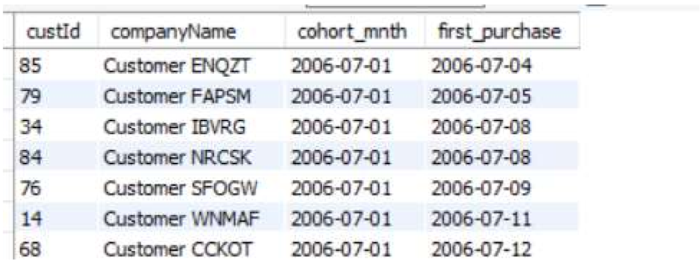
==

Module 4: Cohorts & Retention Analysis

==

Task 1: Identifying Date of First Purchase

Assign each customer to the month of their first purchase to enable cohort tracking

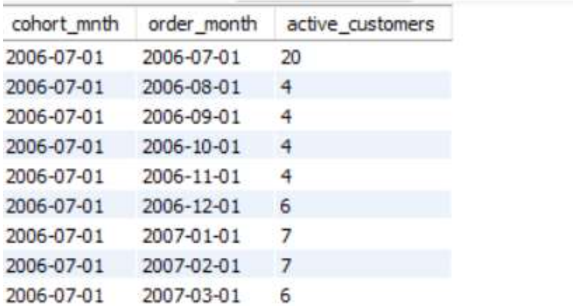


custId	companyName	cohort_mnth	first_purchase
85	Customer ENQZT	2006-07-01	2006-07-04
79	Customer FAPSM	2006-07-01	2006-07-05
34	Customer IBVRG	2006-07-01	2006-07-08
84	Customer NRCSK	2006-07-01	2006-07-08
76	Customer SFOGW	2006-07-01	2006-07-09
14	Customer WNMAF	2006-07-01	2006-07-11
68	Customer CCKOT	2006-07-01	2006-07-12

Comment: Cohort assignment is the basis for retention curves and lifecycle analysis.

Task 2: Number of Returning Customers Month-by-Month

Track how many customers from each cohort return in subsequent months to measure loyalty

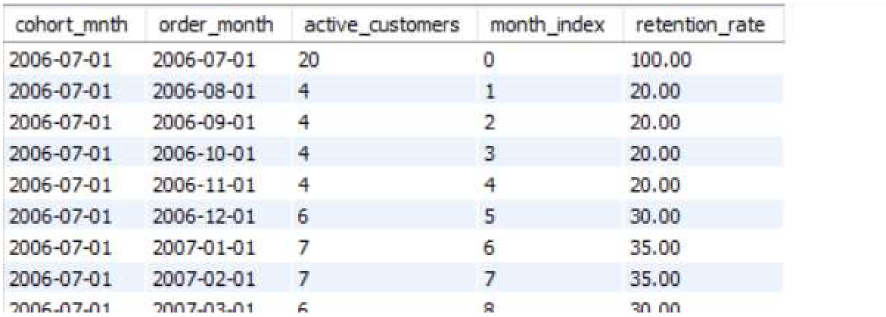


cohort_mnth	order_month	active_customers
2006-07-01	2006-07-01	20
2006-07-01	2006-08-01	4
2006-07-01	2006-09-01	4
2006-07-01	2006-10-01	4
2006-07-01	2006-11-01	4
2006-07-01	2006-12-01	6
2006-07-01	2007-01-01	7
2006-07-01	2007-02-01	7
2006-07-01	2007-03-01	6

Comment: Shows raw retention in counts — a step toward retention rates and cohort health assessment.

Task 3: Calculating Month-on-Month Retention Rate

Express retention as a percentage of the cohort to normalize across cohort sizes

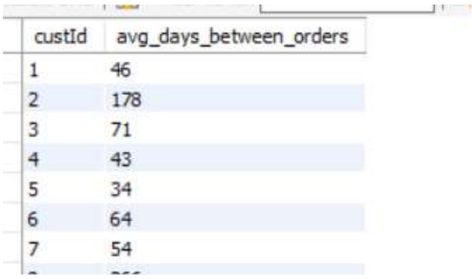


cohort_mnth	order_month	active_customers	month_index	retention_rate
2006-07-01	2006-07-01	20	0	100.00
2006-07-01	2006-08-01	4	1	20.00
2006-07-01	2006-09-01	4	2	20.00
2006-07-01	2006-10-01	4	3	20.00
2006-07-01	2006-11-01	4	4	20.00
2006-07-01	2006-12-01	6	5	30.00
2006-07-01	2007-01-01	7	6	35.00
2006-07-01	2007-02-01	7	7	35.00
2006-07-01	2007-03-01	6	8	30.00

Comment: Retention rates allow comparison across cohorts and to measure retention program effectiveness.

Task 4: Return Analysis – Time to Repurchase (Days Between Orders)

Understand typical time between purchases to time marketing and replenishment triggers.



custId	avg_days_between_orders
1	46
2	178
3	71
4	43
5	34
6	64
7	54

Comment: Use avg_days_between_orders to schedule outreach and predict repurchase windows.

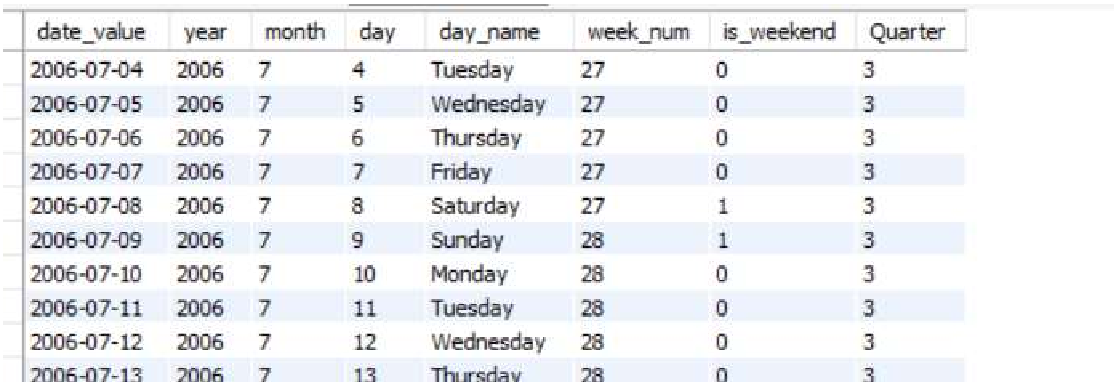
==

Module 5: Date Dimension & Daily Activity

==

Task 1: Create dim_date (date table without gaps)

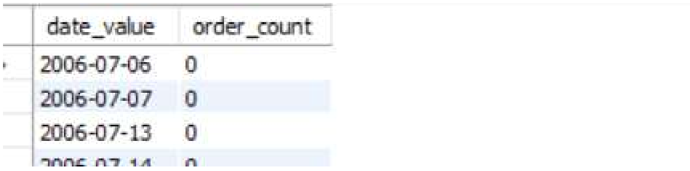
A robust dim_date allows consistent joins and prevents missing-date issues in aggregations



date_value	year	month	day	day_name	week_num	is_weekend	Quarter
2006-07-04	2006	7	4	Tuesday	27	0	3
2006-07-05	2006	7	5	Wednesday	27	0	3
2006-07-06	2006	7	6	Thursday	27	0	3
2006-07-07	2006	7	7	Friday	27	0	3
2006-07-08	2006	7	8	Saturday	27	1	3
2006-07-09	2006	7	9	Sunday	28	1	3
2006-07-10	2006	7	10	Monday	28	0	3
2006-07-11	2006	7	11	Tuesday	28	0	3
2006-07-12	2006	7	12	Wednesday	28	0	3
2006-07-13	2006	7	13	Thursday	28	0	3

Task 2. Days without orders (daily activity)

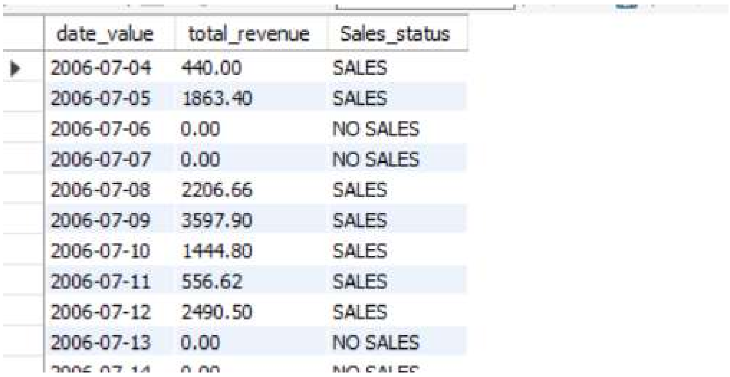
Identify days with zero orders to spot anomalies or expected closures.



date_value	order_count
2006-07-06	0
2006-07-07	0
2006-07-13	0
2006-07-14	0

Task 3: Days without sales per month

Use to monitor operational days missed per month and to investigate causes.



date_value	total_revenue	Sales_status
2006-07-04	440.00	SALES
2006-07-05	1863.40	SALES
2006-07-06	0.00	NO SALES
2006-07-07	0.00	NO SALES
2006-07-08	2206.66	SALES
2006-07-09	3597.90	SALES
2006-07-10	1444.80	SALES
2006-07-11	556.62	SALES
2006-07-12	2490.50	SALES
2006-07-13	0.00	NO SALES
2006-07-14	0.00	NO SALES

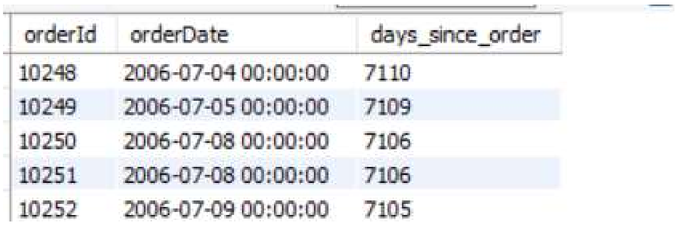
==

Module 6: User-Defined Functions (UDF)

==

Task 1: order_age(order_date) – days since order

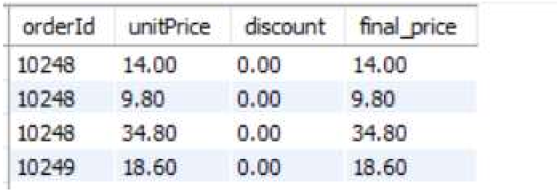
Handy for computing order age directly in queries and for SLA checks



orderId	orderDate	days_since_order
10248	2006-07-04 00:00:00	7110
10249	2006-07-05 00:00:00	7109
10250	2006-07-08 00:00:00	7106
10251	2006-07-08 00:00:00	7106
10252	2006-07-09 00:00:00	7105

Task 2: discount(price, discount) – compute discounted price

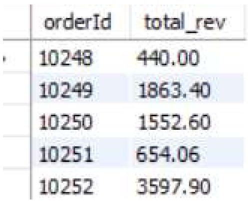
Encapsulating discount calculation ensures consistency across reports



orderId	unitPrice	discount	final_price
10248	14.00	0.00	14.00
10248	9.80	0.00	9.80
10248	34.80	0.00	34.80
10249	18.60	0.00	18.60

Task 3: revenue(unit_price, quantity, discount) – revenue after discount

Simplifies revenue calculations in queries and improves readability



orderId	total_rev
10248	440.00
10249	1863.40
10250	1552.60
10251	654.06
10252	3597.90

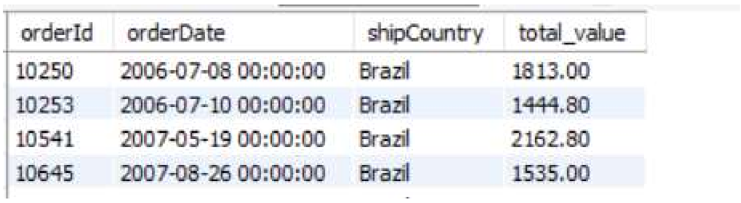
==

Module 7: Stored Procedures

==

Task 1: GetCustomerOrders(cust_code) – list orders for a customer

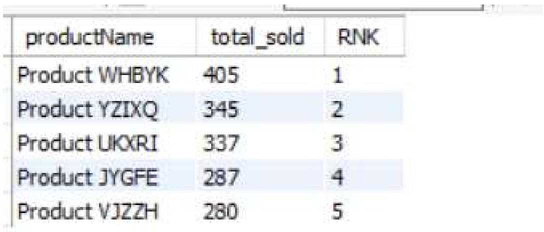
Useful for customer-service portals and account summaries



orderId	orderDate	shipCountry	total_value
10250	2006-07-08 00:00:00	Brazil	1813.00
10253	2006-07-10 00:00:00	Brazil	1444.80
10541	2007-05-19 00:00:00	Brazil	2162.80
10645	2007-08-26 00:00:00	Brazil	1535.00

Task 2: Product_Country(country) – top products by country

A simple stored procedure for quick top-product lookups per market



productName	total_sold	RNK
Product WHBYK	405	1
Product YZIXQ	345	2
Product UKXRI	337	3
Product JYGFE	287	4
Product VJZZH	280	5