

Exercise 3 - Group Data analytics

Question 1

`SELECT Student_name, test-score
CASE`

`WHEN test-score BETWEEN 90 and 100 THEN 'Excellent',`

`WHEN test-score Between 75 and 89 THEN 'Good')`

`WHEN test-score BETWEEN 60 and 74 THEN 'Average')`

`ELSE 'Poor'`

`END AS Performance`

`WHERE test-score > 40`

`FROM students_performance`

Student_name	test-score	Performance
Alice	85	Good
Bob	45	Poor
Charlie	73	Average
Daniel	66	Average
Eva	92	Excellent
Frank	58	Average

2.1. `SELECT product_id, total_amount,
CASE quantity`

`WHEN total_amount >= 1000 THEN 'High',`

`WHEN total_amount BETWEEN 500 and 999 THEN 'Medium',`

`ELSE 'Low',`

`END AS sales_category,`

`FROM sales_summary;`

Product_id	total_amount	Sales_category	Quantity
S01	95	Low	2
S02	1200	High	15
S03	350	Low	1
S04	480	Low	3
S05	990	medium	7
S06	650	medium	6

2.2 SELECT product_id, quantity

CASE

WHERE quantity >= 5 THEN 'Bulk Sale'

ELSE 'Regular Sale'

END AS Sale-type

FROM Sales_summary

Product_id	Quantity	Sale-type
s01	2	Regular Sale
s02	15	Bulk Sale
s03	1	Regular Sale
s04	3	Regular Sale
s05	7	Bulk Sale
s06	6	Bulk Sale

2.3 SELECT Sales_category

FROM Sales_summary GROUP BY SUM(Quantity) AS Sum_of_Quantity

GROUP BY sales_category

FROM Sales_summary;

Sales Category	Sum_of_Quantity
Low	6
High	15
Medium	13

2.4

2.4 SELECT customer_id,
 count(sale_id) AS frequency
 GROUP BY customer_id
 CASE
 WHEN frequency > 1 THEN 'Frequent Buyer'
~~WHEN Frequency > 1 THEN 'Frequent Buyer'~~
 END AS Type_of_Buyer
 FROM sales_summary;

customer_id	frequency	Type_of_Buyer
101	2	Frequent
102	2	Frequent
103	1	
104	1	

Question 3

SELECT *

CASE

WHEN salary >= 60000 THEN 'High'

WHEN salary BETWEEN 40000 and 59999 THEN 'Mid'

~~WHEN salary ELSE 'Low'~~

END AS salary_Bracket

FROM employee_details

35

employee_id	first_name	last_name	department	salary	hiredate	Salary_Bracket
201	Thando	Mokoena	HR	28000	24-11-0	Low
202	Zanele	Khumalo	Finance	55000	22-03-15	Mid
203	Sipho	Dlamini	IT	72000	20-07-10 26-02-26	High
204	Lindwane	Sithole	HR	35000	25-02-20	Low
205	Tebogo	Nkuna	IT	47000	23-06-01	Mid
206	Samele	Zulu	Finance	63000	21-08-01	High

3.6 SELECT employee-id, hire-date

CASE

WHEN hire_date > 2024-01-01, THEN 'Newhire'

ELSE 'Furniture'

END AS hire-category

FROM employee_details;

employee_id	Hiredate	hire-category
201	24-11-01	New hire
202	22-03-15	Furniture
203	20-07-10	Furniture
204	25-02-20	New hire
205	23-06-01	Furniture
206	21-08-01	Furniture

3.7 SELECT salary_bracket,

Count (employee_id) AS number_of_employees

From employee_details

GROUPBY salary_bracket;

Salary-Bracket	number_of_employees
Low	2
Mid	2
High	2

3.8 SELECT department,

Avg(salary) AS avg-sal

CASE

WHEN ~~set~~ avg-sal >

THEN High

WHEN avg-sal BETWEEN

and Then Mid

ELSE WHEN avg-sal ELSE Low

FROM employee_det END AS Range

GroupBy department

Department	Avg-Sal	Classification
HR	62,31500	Low
Finance	59,000	Med
IT	89,500	High

Question 4

9. SELECT product-id, stock-quantity
CASE

WHEN stock-quantity > 0 THEN 'In Stock'
ELSE 'Out of Stock',
END AS stock-status
FROM product-inventory;

Product_id	Stock-quantity	stock-status
301	5	In Stock
302	100	In Stock
303	0	Out of Stock
304	10	In Stock
305	2	In Stock
306	8	In Stock

10. SELECT Product_id, price,

CASE

WHEN price >= 1000 THEN 'Premium'

WHEN price BETWEEN 300 and 999 THEN 'Standard'

ELSE 'Budget'

END AS ~~as~~ price-category,

FROM product-inventory;

Product_id	Price	Price_Category
301	899.99	Standard
302	45	Budget
303	1850	Premium
304	55	Budget
305	299.99	Star Budget
306	599	Standard

Q1) SELECT ~~Price~~ product_category
 Count (Product_id) AS count_of_products,
~~FROM~~ FROM product_inventory1,
 GROUPBY product_category;

Product category	Count_of_products
Budget	3
Standard	2
Premium	1

Q2) SELECT Product_id, stock_quantity

CASE

WHEN stock_quantity ≥ 10 THEN 'In-stock'

ELSE 'Re-order'

ENDAS REORDER_Status

FROM product_inventory

Product_id	Stock_quantity	Reorder_Status
301	9	Re-order
302	100	In-stock
303	0	Re-order
304	10	In-stock
305	2	Re-order
306	8	Re-order

3.13 SELECT Student-id, Marks

CASE

WHEN marks >= 80 THEN 'A'

WHEN marks BETWEEN 70 and 79 THEN 'B'

WHEN marks BETWEEN 60 and 69 THEN 'C'

WHEN marks BETWEEN 50 and 59 THEN 'D'

ELSE 'F'

END AS Grade,

FROM student_courses;

Student-id	Marks	Grade
401	78	B
402	62	C
403	49	F
404	53	D
405	84	A
406	90	A

3.14 SELECT Student-id, enrollment_year

CASE

WHEN enrollment_year > 2023 THEN 'Junior'

ELSE enrollment_year AS ELSE 'Senior'

END AS Student_level

FROM student_courses;

Student_id	Enrollment_year	Student_level
401	2023	Senior
402	2022	Senior
403	2024	Junior
404	2023	Senior
405	2022	Junior
406	2023	Senior

5.15 SELECT ~~student_id~~
~~Count (student_id)~~ AS No_of_Students
 FROM Student_courses
 GROUP BY Grade;

Grade	No-of-Students
A	2
B	1
C	1
D	1
F	1

5.16 SELECT student_id, course, marks

CASE

WHEN Marks >= 75 AND course IN ('Maths', 'Physics')

THEN Yes

~~ELSE~~ NO

END AS ~~Eligibility~~ Scholarship-eligibility

FROM Student_courses ;

Student-id	Marks	Courses	Scholarship-eligibil
401	78	Maths	Yes
402	62	Physics	No
403	49	Chemistry	No
404	53	Biology	No
405	84	Maths	Yes
406	90	Physics	Yes

6.17 `SELECT visit_id, session_duration
CASE
WHEN session_duration >= 3000 THEN 'Long'
WHEN session_duration BETWEEN 60 and 299 THEN 'Medium'
ELSE 'Short'
END AS session_type
FROM website_traffic_logs`

Visit_id	session_duration	session-type
601	45	short
602	200	medium
603	360	Long
604	90	medium
605	1460	medium
606	25	short

* 6.18 `SELECT User_idCount (user_id) AS 'Frequency',
Case
WHEN Frequency >1 THEN 'Frequent visitor'
ELSE '_'
End AS visitor_Frequency
FROM website_traffic_logs`

User Id	Frequency	visitor frequency
901	2	Frequent visitor
902	2	Frequent visitor
903	1	
904	1	

6.19 SELECT visit_id, visit_date,
CASE
WHEN ~~date~~ visit_date IN ('2025-03-12', '2025-03-23') THEN
'Weekend'
ELSE 'Weekday'
END AS day_type
FROM website_traffic_logs;

<u>Visit_id</u>	<u>visit_date</u>	<u>day-type</u>
601	2025-03-18	Weekday
602	2025-03-18	Weekday
603	2025-03-19	Weekday
604	2025-03-20	Weekday
605	2025-03-20	Weekday
606	2025-03-21	Weekday

6.20 SELECT session_type
Count (Visit_id) AS Frequency,
GROUP BY session_type,
FROM website_traffic_logs_2;

<u>Session_type</u>	<u>Frequency</u>
Short	2
Medium	3
Long	1

Question 7

7.21 SELECT feedback_id, feedback_score, product_category
CASE
WHEN feedback_score >= 9 THEN 'Excellent'
WHEN feedback BETWEEN 7 and 8 THEN 'Good'
WHEN feedback BETWEEN 5 and 6 THEN 'Average'
ELSE 'Poor'
END AS feedback_label
FROM customer_feedback;

feedback_id	feedback_score	feedback_label	Product-category
701	9	Excellent	Electronics
702	5	Average	Furniture
703	10	Excellent	Stationery
704	3	Poor	Furniture
705	7	Good	Electronics
706	4	Poor	Stationery

7.22 SELECT Product-category
Count(feedback_label) AS ~~not~~ Poor-feedback
GROUP BY product-category
Having ~~feedback_label~~ or Poor-feedback_label = Poor
AND feedback Count(feedback_label) > 1

Question

8.23 SELECT Employee_id, (days present / total days) * 100 AS Rate,

CASE

WHEN Rate >= 95 THEN 'Good' 'Excellent'

WHEN Rate BETWEEN 80 and 94 THEN 'Good'

WHEN Rate ELSE 'Needs Improvement'

END AS Attendance_rating

FROM employee_attendance;

employee_id	Rate	Attendance_rating
801	90%	Good
802	68%	Needs improvement
803	100%	Excellent
804	81%	Good
805	95%	Excellent
806	72.54, 84%	Needs improvement

8.24 SELECT Attendance rating

Count (Employee_id) AS no_of_EE

GROUP BY Attendance rating

From employee_attendance 1;

Attendance rating	NO-OF_EE
Excellent	2
Good	2
Needs Improvement	2

Question 9

9.25 SELECT Return_id, Return_amount

CASE

WHEN Return_amount > 300 THEN 'High Value'

WHEN ELSE 'Low Value'

END AS Return_type

FROM Order_returns;

Return_id	Return_amount	Return-type
901	450	High Value
902	100	Low Value
903	200	Low Value
904	350	High Value
905	150	Low Value
906	90	Low Value

9.26 SELECT ReturnReason

Count (Return_amount) AS no_of_returns

HAVING Return_amount > 100

GROUP BY ReturnReason

FROM order_returns

ReturnReason	no_of_returns
Defective	2
Changed Mind	0
Late delivery	1
Other	1

Question 10

10.27 SELECT campaign_id, engagement_score,
CASE
WHEN engagement_score >= 80 THEN 'High'
WHEN engagement_score BETWEEN 50 and 79 THEN 'Medium'
ELSE 'Low'
END AS engagement_level
FROM campaign_engagement

Campaign_Id	engagement_score	engagement_level
1001	85	High
1002	55	Medium
1003	20	Low
1004	75	Medium
1005	95	High
1006	60	Medium

to. 288 SELECT Region
Count (