EXERCISE 3 : SQL CASE STATEMENTS

Question 1

Classify each product by price
- Expensive IF Price > 1000
- Mid_Range IF Price between 100 & 1000
- Budget IF Price < 100

```sql
SELECT Product_name, Price,
        CASE
                WHEN Price > 1000 THEN `Expensive'
                WHEN Price Between 100 AND 1000
                THEN `Mid_Range'
                WHEN Price < 100 THEN `Budget'
        END AS Price_Category
FROM Sales. Public? Price;
```

Product_Name	Price	Price_Category
Laptop	1200	Expensive
Phone	800	Mid_Range
Keyboard	45	Budget
Monitor	300	Mid_Range
Mouse	25	Budget

Question 2

Label each Order
- High value for Order ≥ 1000
- Medium Value for 500 & 999
- Low Value for Order < 500

```sql
SELECT Customer_name, Amount,
    CASE
        WHEN Amount >= 1000 THEN `High_Value`
        WHEN Amount Between 500 AND 999,99
        THEN `Medium_range`
        ELSE `<500_Low_range`
    END AS Order value Category
FROM ORDER;
```

Customer Name	Amount	Order Value Category
Alice	150	Low Value
Bob	560	Medium Value
Charlie	999,99	Medium Value
Diana	45,50	Low Value
Ethan	1200	High Value

Question 3

```sql
SELECT Emp_name, Department,
    Salary,
    CASE
        WHEN Department = `IT`
        AND Salary > 80 000 THEN `Senior IT`
        WHEN Department = `HR`
        AND Salary > 55 000 THEN `Experienced`
        ELSE STAFF
    END AS Position_Level
FROM Employees;
```

Emp_Name	Department	Salary	Position_Level
John	IT	85000	Senior IT
Sara	HR	60 000	Experienced HR
Mark	IT	75 000	STAFF
Lucy	Finance	95 000	STAFF
Tom	HR	55 000	STAFF

Question 4

```
SELECT Student_name, Score,
    CASE
        WHEN Score => 90 THEN `A´
        WHEN Score BETWEEN 80 AND 89
        THEN `B´
        WHEN Score BETWEEN 70 AND 79
        THEN `C´
        WHEN Score BETWEEN 60 AND 69
        THEN `D´
        WHEN Score <60 THEN `F´
    END AS GRADE
FROM Students;
```

Student_name	Score	Grade
Anna	92	A
Ben	76	C
Cara	59	F
David	83	B
Ella	88	D

Question 5

```sql
SELECT Delivery_id, delivery_Time_Minutes
    CASE
        WHEN Delivery_Time_Minutes <= 30 THEN `Fast`
        WHEN Delivery_Time_Minutes BETWEEN 31
        AND 60 THEN `On_Time`
        WHEN Delivery_Time_Minutes = 60 THEN `Late`
    END AS Performance
From Delivery;
```

Delivery_id	delivery_Time_Minutes	Performance
1	45	On time
2	80	Late
3	30	FAST
4	65	Late
5	100	Late

Question 6

```sql
SELECT Issue_Type, Priority,
    CASE
        WHEN Priority = 3 THEN `High`
        WHEN Priority = 2 THEN `Medium`
        WHEN Priority = 1 THEN `Low`
    END AS Priority_babel
FROM Tickets;
```

Issue type	Priority	Priority_labels
login issue	1	Low
Server down	3	High
Slow System	2	Medium
Email error	2	Medium
Password reset	1	Low

Question 7

```
SELECT Student_ID, (Days_Present / Total_days)
          * 100 AS Attendance_Percentage,
   CASE
          WHEN Attendance_Percentage >= 90
          THEN `Excellent'
          WHEN Attendance_Percentage Between 75
          AND 89 THEN `Good'
          WHEN Attendance_Percentage < 75    THEN
          `Needs_Improvement'
          END As Attendance_Status
FROM Attendance;
```

Student_ID	Attendance_Percentage	Attendance Status
1	90 %	Excellent
2	60 %	Needs Improvement
3	96%	Excellent
4	50%	Needs Improvement
5	100%	Excellent

Question 8

```sql
SELECT Product_id, Stock_qty;
    CASE
        WHEN Stock_qty = 0 THEN `Out_of_Stock`
        WHEN Stock_qty BETWEEN 1 AND 5
        THEN `Low_Stock`,
        ELSE `>5_In_Stock`
    END AS Stock_Status
FROM Product_Inventory
```

Product_id	Stock_Qty	Product Inventory Stock_Status
1	5	In_Stock
2	0	Out_of_Stock
3	25	In_Stock
4	10	IN_Stock
5	3	Low_Stock

Question 9

```sql
SELECT Subject, enrolled_students,
    CASE
        WHEN Enrolled_students >= 25 THEN `Large`
        WHEN Enrolled_Students BETWEEN 10
        AND 24 THEN `Medium`
        WHEN Enrolled_Students < 10 THEN `Small`
    END AS Class_Size_Category
FROM Classes;
```

Subject	Enrolled_Students	Class_Size_Category
Maths	30	Large
English	25	Large
Science	15	Medium
Art	5	Small
History	20	Medium

Question 10

```sql
SELECT Payment_id, Payment_method,
        Amount,
    CASE.
        WHEN Amount >= 200 THEN `Eligible`
    ELSE `Not_Eligible`
    ENDAS Discount_Eligibility
FROM Payments
WHERE Payment_Method = `CASH`;
```

Payment_id	Payment method	Amount	Discount_Eligibility
1	CASH	50	Not Eligible
2	CASH	200	Eligible
3	CASH	150	Not_Eligible
5	CASH	300	Eligible