SQL Fundamentals SOLUTIONS

Exercise 3: SQL CASE Statements

1. **Classify products by price**

SQL QUERY:

```
SELECT
  product_name,
  price,
  CASE
    WHEN price > 1000 THEN 'Expensive'
    WHEN price BETWEEN 100 AND 1000 THEN 'Mid-range'
    ELSE 'Budget'
  END AS price_category
FROM products;
```

EXPECTED OUTPUT

product_name	price	price_category
Laptop	1200	Expensive
Phone	800	Mid-range
Keyboard	45	Budget
Monitor	300	Mid-range
Mouse	25	Budget

2. **Label orders by value**

SQL QUERY:

```
SELECT
  customer_name,
  amount,
  CASE
    WHEN amount >= 1000 THEN 'High Value'
    WHEN amount BETWEEN 500 AND 999.99 THEN 'Medium Value'
    ELSE 'Low Value'
  END AS order_value_category
FROM orders;
```

EXPECTED OUTPUT

customer_name	amount	order_value_category
Alice	150.00	Low Value
Bob	560.00	Medium Value
Charlie	999.99	Medium Value
Diana	45.50	Low Value
Ethan	1200.00	High Value

3. **Categorize employee position**

SQL QUERY:

```
SELECT
  emp_name,
  department,
  salary,
  CASE
    WHEN department = 'IT' AND salary > 80000 THEN 'Senior IT'
    WHEN department = 'HR' AND salary > 55000 THEN 'Experienced HR'
    ELSE 'Staff'
  END AS position_level
FROM employees;
```

EXPECTED OUTPUT

emp_name	department	salary	position_level
John	IT	85000	Senior IT
Sara	HR	60000	Experienced HR
Mark	IT	75000	Staff
Lucy	Finance	95000	Staff
Tom	HR	55000	Staff

4. Assign letter grades

SQL QUERY:

```
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'
        ELSE 'F'
    END AS grade
FROM students;
```

EXPECTED OUTPUT

student_name	score	grade
Anna	92	A
Ben	76	C
Cara	59	F
David	83	B
Ella	68	D

5. Label delivery performance

SQL QUERY:

```
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'
    ELSE 'Late'
  END AS performance
FROM deliveries;
```

EXPECTED OUTPUT

delivery_id	delivery_time_minutes	performance

1	45	On Time
2	80	Late
3	30	Fast
4	65	Late
5	100	Late

6. Convert priority to labels

SQL QUERY

```
SELECT
  issue_type,
  priority,
  CASE
    WHEN priority = 3 THEN 'High'
    WHEN priority = 2 THEN 'Medium'
    WHEN priority = 1 THEN 'Low'
  END AS priority_label
FROM tickets;
```

EXPECTED OUTPUT

issue_type	priority	priority_label
Login issue	1	Low
Server down	3	High
Slow system	2	Medium
Email error	2	Medium
Password reset	1	Low

7. Attendance percentage and status

SQL QUERY:

```
SELECT
    student_id,
    (days_present * 100 / total_days) AS attendance_percentage,
    CASE
        WHEN (days_present * 100 / total_days) >= 90 THEN 'Excellent'
        WHEN (days_present * 100 / total_days) BETWEEN 75 AND 89 THEN 'Good'
        ELSE 'Needs Improvement'
    END AS attendance_status
FROM attendance;
```

EXPECTED OUTPUT

student_id	attendance_percentage	attendance_status
1	90.0	Excellent
2	60.0	Needs Improvement
3	96.0	Excellent
4	50.0	Needs Improvement
5	100.0	Excellent

8. Label stock status

SQL QUERY:

```
SELECT
    product_id,
    stock_qty,
    CASE
        WHEN stock_qty = 0 THEN 'Out of Stock'
        WHEN stock_qty BETWEEN 1 AND 4 THEN 'Low Stock'
        ELSE 'In Stock'
    END AS stock_status
FROM products_inventory;
```

EXPECTED OUTPUT

product_id	stock_qty	stock_status
1	5	In Stock
2	0	Out of Stock
3	25	In Stock
4	10	In Stock
5	3	Low Stock

9. Classify class size

SQL QUERY:

```
SELECT
    subject,
    enrolled_students,
    CASE
        WHEN enrolled_students >= 25 THEN 'Large'
        WHEN enrolled_students BETWEEN 10 AND 24 THEN 'Medium'
        ELSE 'Small'
    END AS class_size_category
FROM classes;
```

EXPECTED OUTPUT

subject	enrolled_students	class_size_category
Math	30	Large
English	25	Large
Science	15	Medium
Art	5	Small
History	20	Medium

10. Apply discount flag

SQL QUERY:

```
SELECT
  payment_id,
  payment_method,
  amount,
  CASE
    WHEN payment_method = 'Cash' AND amount >= 200 THEN 'Eligible for
Discount'
    ELSE 'Not Eligible'
  END AS discount_eligibility
FROM payments;
```

EXPECTED OUTPUT

payment_id	payment_method	amount	discount_eligibility
1	Card	50.00	Not Eligible
2	Cash	200.00	Eligible for Discount
3	Card	150.00	Not Eligible
4	PayPal	75.00	Not Eligible
5	Cash	300.00	Eligible for Discount