Question 1:

1. **Classify students by performance Query:**

```
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 50 AND 74 THEN 'Average'
    ELSE 'Poor'
  END AS performance
FROM students;
```

Output:

Student_name	Test_score	Performance
Alice	85	Good
Bob	45	Poor
Charlie	73	Average
David	66	Average
Eva	92	Excellent
Frank	58	Average
Grace	77	Good
Henry	33	Poor
Ivy	100	Excellent
Jack	0	Poor

Question 2:

1. **Sales category (Low < 500, Medium <1000, High >= 1000) Query:**

```
SELECT *,
 CASE
  WHEN total_amount < 500 THEN 'Low'
  WHEN total_amount < 1000 THEN 'Medium'
  ELSE 'High'
 END AS sales_category
FROM sales;
```

Output:

Sale_ID	Sale_date	Customer_id	Product ID	Quantity	Total_amount	Sales_category
1	2025-01-05	101	501	2	95	Low
2	2025-01-10	102	502	15	1200	High
3	2025-01-11	103	503	1	350	Low
4	2025-01-15	101	504	3	480	Low

2. **Bulk vs Regular Sale (Bulk if quantity >= 5) Query:**

```
SELECT *,
 CASE
  WHEN quantity >= 5 THEN 'Bulk Sale'
  ELSE 'Regular Sale'
 END AS sale_type
FROM sales;
```

Output:

Sale_ID	Quantity	Sales_type
1	2	Regular sale
2	15	Bulk sale
3	1	Regular sale
4	3	Regular sale

3. **Count of sales category Query:**

```
SELECT
 CASE
  WHEN total_amount < 500 THEN 'Low'
  WHEN total_amount < 1000 THEN 'Medium'
  ELSE 'High'
 END AS sales_category,
 COUNT(*) AS sale_count
FROM sales
GROUP BY sales_category;
```

Output:

Sales_category	Total_sales
Low	3
High	1

4. **Frequent vs Occasional Buyers (Frequent = >= 2 sales) Query:**

```
SELECT
 customer_id,
 COUNT(*) AS sales_count,
 CASE
  WHEN COUNT(*) >= 2 THEN 'Frequent Buyer'
  ELSE 'Occasional Buyer'
```

```
 END AS buyer_type
FROM sales
GROUP BY customer_id;
```

Output:

Customer_id	Sales_count	Customer_type
101	2	Frequent buyer
102	1	Occasional buyer
103	1	Occasional buyer

Question 3: Employees Table

5. **Salary bucket Query:**

```
SELECT *,
 CASE
  WHEN salary < 30000 THEN 'Low'
  WHEN salary <= 50000 THEN 'Medium'
  ELSE 'High'
 END AS salary_bracket
FROM employees;
```

Output:

Employee_id	Salary	Salary_bucket
201	28000	Low
202	55000	High
203	72000	High
204	35000	Medium

6. **New hire vs Experienced Query:**

```
SELECT *,
 CASE
  WHEN hire_date >= '2024-01-01' THEN 'New Hire'
  ELSE 'Experienced'
 END AS hire_type
FROM employees;
```

Output:

Employee_id	Hire_date	Hire_type
201	2024-11-01	New hire
202	2022-03-15	Experienced
203	2020-07-10	Experienced
204	2025-02-20	New hire

7. **Count by salary bucket Query:**

```
SELECT
 CASE
  WHEN salary < 30000 THEN 'Low'
  WHEN salary <= 50000 THEN 'Medium'
  ELSE 'High'
 END AS salary_bracket,
 COUNT(*) AS employee_count
FROM employees
GROUP BY salary_bracket;
```

Output:

Salary_bucket	Employee_count
Low	1
Medium	1
High	2

8. **Average salary per department + classification Query:**

```
SELECT
 department,
 AVG(salary) AS average_salary,
 CASE
  WHEN AVG(salary) < 40000 THEN 'Low'
  WHEN AVG(salary) <= 60000 THEN 'Average'
  ELSE 'High'
 END AS salary_classification
FROM employees
GROUP BY department;
```

Output:

Department	Average_salary	Salary_classification
HR	31500	Low
Finance	55000	Average
IT	72000	High

Question 4: Products table

9. **Stock status Query:**

```
SELECT *,
 CASE
  WHEN stock_quantity = 0 THEN 'Out of Stock'
  ELSE 'In Stock'
 END AS stock_status
FROM products;
```

Output:

Product_id	Product_name	Category	Price	Stock_quantity	Stock_status
301	Office chair	Furniture	899.99	5	In stock
302	Pen pack	Stationery	45	100	In stock
303	Monitor 24	Electronics	1850	0	Out of stock
304	stapler	stationery	55	10	In stock

10. **Price category Query:**

```
SELECT *,
 CASE
  WHEN price < 100 THEN 'Cheap'
  WHEN price <= 1000 THEN 'Mid-Range'
  ELSE 'Expensive'
 END AS price_category
FROM products;
```

Output:

Product_name	Price	Price_category
Office chair	899.99	Mid-range
Pen pack	45	Cheap
Monitor 24	1850	Expensive
stapler	55	Cheap

11. **Count by price category Query:**

```
SELECT
 CASE
  WHEN price < 100 THEN 'Cheap'
  WHEN price <= 1000 THEN 'Mid-Range'
  ELSE 'Expensive'
 END AS price_category,
 COUNT(*) AS product_count
FROM products
GROUP BY price_category;
```

Output:

Price_category	Product_count
Cheap	2
Mid-range	1
Expensive	1

12. **Reorder status Query:**

```
SELECT *,
 CASE
  WHEN stock_quantity < 10 THEN 'Reorder Needed'
  ELSE 'Sufficient Stock'
 END AS reorder_status
FROM products;
```

Output:

Product_name	Stock_quantity	Reorder_status
Office chair	5	Reorder needed
Pen pack	100	Sufficient stock
Monitor 24	0	Reorder needed
stapler	10	Sufficient stock

Question 5: Student table

13. **Grade column Query:**

```
SELECT *,
 CASE
  WHEN marks >= 75 THEN 'A'
  WHEN marks >= 60 THEN 'B'
  WHEN marks >= 50 THEN 'C'
  ELSE 'D'
 END AS grade
FROM students;
```

Output:

Full_name	Marks	Grade
Lerato Nkosi	78	A
Daniel Mthembu	62	B
Nthabiseng Molefe	49	D
John Mashaba	53	C

14. **Student level Query:**

```
SELECT *,
 CASE
  WHEN enrollment_year = 2024 THEN 'First Year'
  WHEN enrollment_year = 2023 THEN 'Second Year'
  ELSE 'Senior'
 END AS student_level
FROM students;
```

Output:

Full_name	Enrolment_year	Student_level
Lerato Nkosi	2023	Second year
Daniel Mthembu	2022	Senior
Nthabiseng Molefe	2024	First year
John Mashaba	2023	Second year

15. **Count per grade Query:**

```
SELECT
  CASE
    WHEN marks >= 75 THEN 'A'
    WHEN marks >= 60 THEN 'B'
    WHEN marks >= 50 THEN 'C'
    ELSE 'D'
  END AS grade,
  COUNT(*) AS student_count
FROM students
GROUP BY grade;
```

Output:

Grade	Student_count
A	1
B	1
C	1
D	1

16. **Scholarship eligibility (>= 75) Query:**

```
SELECT *,
 CASE
  WHEN marks >= 75 THEN 'Eligible'
  ELSE 'Not Eligible'
 END AS scholarship_eligibility
FROM students;
```

Output:

Full_name	Marks	Scholarship_egibility
Lerato Nkosi	78	Eligible
Daniel Mthembu	62	Not eligible
Nthabiseng Molefe	49	Not eligible
John Mashaba	53	Not eligible

Question 6: Website traffic table:

17. **Session type classification Query:**

```
SELECT *,
 CASE
  WHEN session_duration >= 300 THEN 'Long'
  WHEN session_duration >= 100 THEN 'Medium'
  ELSE 'Short'
 END AS session_type
FROM website_traffic;
```

Output:

Visit_id	Session_duration	Session_type
601	45	Short
602	200	Medium
603	360	Long
604	90	Short

18. **Frequent vs occasional visitor Query:**

```
SELECT
 user_id,
 COUNT(*) AS visit_count,
 CASE
  WHEN COUNT(*) >= 2 THEN 'Frequent Visitor'
  ELSE 'Occasional Visitor'
 END AS visitor_type
FROM website_traffic
GROUP BY user_id;
```

Output:

User_id	Visit_count	Visitor_type
901	2	Frequent visitor
902	1	Occasional visitor
903	1	Occasional visitor

19. **Weekday vs weekend Query:**

```
SELECT *,
 CASE
  WHEN EXTRACT(DOW FROM visit_date) IN (0, 6) THEN 'Weekend'
  ELSE 'Weekday'
 END AS day_type
FROM website_traffic;
```

Output:

Visit_id	Visit_date	Day_type
601	2025-03-18	Weekday
602	2025-03-18	Weekday
603	2025-03-19	Weekday
604	2025-03-20	Weekday

20. **Count by session type Query:**

```
SELECT
 CASE
  WHEN session_duration >= 300 THEN 'Long'
  WHEN session_duration >= 100 THEN 'Medium'
  ELSE 'Short'
 END AS session_type,
 COUNT(*) AS visit_count
FROM website_traffic
GROUP BY session_type;
```

Output:

Session_type	Visit_count
Short	2
Medium	1
Long	1