```sql
--NUMBER 5 Write down the correct query that will return a list of employees
--whose age is 30 years old and above, and its respective department.
--FN=Finance, CS=Customer Service, HR=Human Resource, MKT=Marketing.

SELECT
    employee_id,
    country,
    CASE SPLIT_PART(employee_id, '-', 1) --SPLIT_PART(string, delimiter,
position)
        WHEN 'FIN' THEN 'Finance'
        WHEN 'CS' THEN 'Customer Service'
        WHEN 'HR' THEN 'Human Resources'
        WHEN 'MKT' THEN 'Marketing'
        ELSE 'Unknown'
    END AS department --name of the new column
FROM
    employee_table
WHERE
    age >= 30;

--NUMBER 6. Query that will get a list of seller_id whose sum
--of returned items is greater than 5% of their total items sold.

SELECT
    seller_id,
    COALESCE(SUM(item_purchased) FILTER (WHERE status = 'RETURNED'),0) AS
returned_items,
    COALESCE(SUM(item_purchased) FILTER (WHERE status = 'DELIVERED'),0) AS
delivered_items
FROM
    order_table
GROUP BY
    seller_id
HAVING
    COALESCE(SUM(item_purchased) FILTER (WHERE status = 'RETURNED'),0) >
    (COALESCE(SUM(item_purchased) FILTER (WHERE status = 'DELIVERED'),0) * 0.05)
```

```sql
--NUMBER 7. Write down the correct query that will yield the following expected
output,
--where c_cost is the cumulative sum of cost by order_date.

SELECT o.create_date, d.buyer_id, SUM(o.cost) OVER (ORDER BY o.create_date) AS
c_cost
FROM order_table o
JOIN delivery_table d ON o.order_id = d.order_id
```