Pusula Talent Academy 2025 - SQL & DBA Case Study

Question 1: Performance & Scalability Analysis in Hospital Data

Scenario:
A table below has been used for 5 years in a Hospital Information Management System (HBYS). Each day, about 25,000 rows are inserted.
Recently, queries on this table have become slower and users have reported difficulty accessing past records.

```
CREATE TABLE HastaIslemLog (
    Id INT IDENTITY(1,1) PRIMARY KEY,
    HastaId INT,
    IslemTarihi DATETIME,
    IslemKodu NVARCHAR(20),
    Aciklama NVARCHAR(500)
);
```

Question:
1. What could be the reasons for the performance degradation?
2. What improvements would you suggest for better sustainability?
3. Do you think using the table in this way for 5 years was the correct approach? Why or why not?

Note: Open-ended. Focus on reasoning, not just query writing.

Question 2: Index Strategy & Query Optimization Thinking

Scenario:
The following query is frequently used by end users:

```
SELECT *
FROM HastaKayit
WHERE LOWER(AdSoyad) LIKE '%ahmet%' AND YEAR(KayitTarihi) = 2024
```

Question:
1. What performance problems might arise from this query?
2. How would you optimize this query and/or the table structure?
3. Are there any improvements that could be made on the application side?

Note: Expect analysis and optimization suggestions.

Question 3: T-SQL Query Challenge (Hospital Sales Example)

Scenario:

Pusula Talent Academy 2025 - SQL & DBA Case Study

In the HBYS system, the hospital pharmacy sells products. Sales and product details are stored in the following tables:

```
CREATE TABLE Urun (
    UrunID INT PRIMARY KEY,
    UrunAdi NVARCHAR(100),
    Fiyat DECIMAL(10,2)
);

CREATE TABLE Satis (
    SatisID INT PRIMARY KEY,
    UrunID INT FOREIGN KEY REFERENCES Urun(UrunID),
    Adet INT,
    SatisTarihi DATETIME
);
```

Sample Data:

```
-- Urun
(1, 'Laptop', 15000.00), (2, 'Mouse', 250.00), (3, 'Klavye', 450.00)
-- Satis
(1, 1, 2, '2024-01-10'), (2, 2, 5, '2024-01-15'), (3, 1, 1, '2024-02-20'),
(4, 3, 3, '2024-03-05'), (5, 2, 7, '2024-03-25'), (6, 3, 2, '2024-04-12')
```

Tasks:

1. Write a query that returns, per year and per product, the total sales amount (Fiyat * Adet) and total quantity.
2. For each year, identify the product with the highest sales amount.
3. Write a query to list products that were never sold.

Note: Use MSSQL syntax.