Programming Applications with Databases

Exercise Set 6

1. Consider the following table:

ID	Patient	Patient Address	Appointment time and location	Price	Physician	Appointment cause
1	Jan Kot	6 Dolna Street, 44-444 Bór	2029-02-01 12:30, room 12	300 zł	Oleg Wyrwiząb	Dental: Denture fitting in (. . .)
2	Maria Mysz	9 Górna Street, 55-555 Las	2030-01-04 11:45, room 7	150 zł	Ewa Ciarka	Dermatology: Birthmark inspection (. . .)

   Design alternative schemas for the same data that conform with 1NF, 2NF, and 3NF. There is no need to create an SQL script, it is enough to present appropriate schemas.
   [4p]

2. Consider the following query:

```
SELECT DISTINCT c.PESEL, c.Nazwisko
FROM Egzemplarz e
JOIN Ksiazka k ON e.Ksiazka_ID=k.Ksiazka_ID
JOIN Wypozyczenie w ON e.Egzemplarz_ID=w.Egzemplarz_ID
JOIN Czytelnik c ON c.Czytelnik_ID = w.Czytelnik_ID;
```

   The following query that contains a subquery returns the same result:

```
SELECT c.PESEL, c.Nazwisko
FROM Czytelnik c WHERE c.Czytelnik_ID IN
(SELECT w.Czytelnik_ID FROM Wypozyczenie w
JOIN Egzemplarz e ON e.Egzemplarz_ID=w.Egzemplarz_ID
JOIN Ksiazka k ON e.Ksiazka_ID=k.Ksiazka_ID)
```

   Compare and briefly analyze query plans. Generate more data and try to compare execution times. Create another variant of this query (i.e. guaranteed to return the same result) with a one more subquery. Analyze its query plan and compare the execution time with the others.
   [3p]

3. Consider a query that joins the tables *Książka* (book) and *Egzemplarz* (book copy, specimen). Create clustered and non-clustered indexes, and at least one covering index to support the query. Analyze the query plans.
   [3p]

Paweł Rajba