

**2025.09.22.**

**Órai Feladat**

**Határidő:**

**75 perc**

# 1. DDL (Data Definition Language) feladatok

## Feladat 1.1: Táblák létrehozása

### Leírás:

Készíts egy `Employee` nevű táblát az alábbi oszlopokkal:

- `employee_id`: SERIAL, elsődleges kulcs
- `name`: VARCHAR(100), nem lehet NULL
- `position`: VARCHAR(50)
- `salary`: DECIMAL(10,2)
- `hire_date`: DATE

### További követelmény:

A táblát hozd létre úgy, hogy ha már létezik, akkor először töröld a `DROP TABLE IF EXISTS` parancs segítségével!

---

## Feladat 1.2: Táblák módosítása (ALTER TABLE)

### Leírás:

Módosítsd az előző feladatban létrehozott `Employee` táblát, hogy hozzáadd az alábbi oszlopot:

- `email`: VARCHAR(100)

Ezután használd az `UPDATE` parancsot, hogy minden rekordban az `email` oszlop értékét a következő módon állítsd be:

- Alakítsd kisbetűssé az `name` oszlop értékét, távolítsd el belőle a szóközőket, majd fűzd hozzá a `@company.com` karakterláncot.  
(Példa: *Frank Miller* → `frankmiller@company.com`)
- 

## Feladat 1.3: Táblák törlése

### Leírás:

Írj egy SQL utasítást, ami törli az alábbi táblákat, ha léteznek (használd a PostgreSQL `CASCADE` opcióját):

- `Orders`
  - `OrderItem`
-

## 2. DML (Data Manipulation Language) feladatok

### Feladat 2.1: Dummy adatok beszúrása

**Leírás:**

Hozz létre legalább három rekordot az `Employee` táblába. Gondolj arra, hogy különböző pozíciókat és fizetéseket adj meg (például Manager, Sales, Support stb.).

*Példa:*

```
INSERT INTO Employee (name, position, salary, hire_date) VALUES ('Alice Johnson', 'Manager', 3500.00, '2025-01-15');
```

---

### Feladat 2.2: Adatok frissítése

**Leírás:**

Írj egy SQL utasítást, amely 5%-kal növeli a fizetését azon alkalmazottnak, akiknek a pozíciója Sales.

*Példa:*

Ha egy Sales pozícióban dolgozó fizetése 2000.00, akkor az új fizetés legyen 2100.00.

---

### Feladat 2.3: Adatok törlése

**Leírás:**

Töröld ki minden olyan rekordot az `Orders` táblából, ahol a `total_amount` kisebb, mint 100.00.

*Példa:*

```
DELETE FROM Orders WHERE total_amount < 100.00;
```

---

## 3. DQL (Data Query Language) feladatok

### Feladat 3.1: INNER JOIN lekérdezés

**Leírás:**

Készíts egy lekérdezést, amely listázza az összes rendelést az alábbi adatokkal:

- Rendelés azonosító (`order_id`)
- Rendelés dátuma (`order_date`)

- Vásárló neve (feltételezd, hogy van egy `Customer` tábla, amely a `customer_id` mező alapján kapcsolódik az `Orders` táblához)
- Rendelés teljes összege (`total_amount`)

**Megjegyzés:**

Az `INNER JOIN` csak azokat a rekordokat adja vissza, ahol a kapcsolódó sorok léteznek mindkét táblában.

---

**Feladat 3.2: LEFT JOIN lekérdezés****Leírás:**

Készíts egy lekérdezést, amely listázza az összes terméket a `Product` táblából, valamint a hozzájuk tartozó üzlet nevét a `Shop` táblából.

- Ha egy termékhez nincs hozzárendelve üzlet (`NULL` érték), akkor is jelenjen meg a termék az eredményben.

**Megjegyzés:**

A `LEFT JOIN` az összes rekordot visszaadja a bal oldali táblából, még akkor is, ha nincs megfelelő egyezés a jobb oldali táblában.

---

**Feladat 3.3: RIGHT JOIN lekérdezés****Leírás:**

Készíts egy lekérdezést, amely listázza az összes üzletet a `Shop` táblából, valamint az adott üzlethez tartozó alkalmazottak nevét a `Employee` táblából.

- Az eredményben jelenjenek meg azok az üzletek is, amelyekhez nincs alkalmazott hozzárendelve.

**Megjegyzés:**

A `RIGHT JOIN` az összes rekordot visszaadja a jobb oldali táblából, még akkor is, ha nincs megfelelő egyezés a bal oldali táblában.

---

**Feladat 3.4: FULL OUTER JOIN lekérdezés****Leírás:**

Készíts egy lekérdezést, amely összekapcsolja a `Customer` és a `CustomerServiceTicket` táblákat, majd az eredményben jelenítse meg:

- Vásárló neve
- Ügyfélszolgálati jegy leírása (`issue_description`)
- Jegy státusza

- Az ügyet kezelő alkalmazott neve (ehhez kapcsolódik a `Employee` tábla az `employee_id` alapján)

**Megjegyzés:**

A FULL OUTER JOIN mindkét tábla összes rekordját visszaadja, függetlenül attól, hogy van-e egyezés a másik táblában.

---

**Feladat 3.5: \*CTE és több JOIN használata (\* = szorgalmi)****Leírás:**

Készíts egy lekérdezést, amely először kiszámolja, hogy melyik üzletben mennyi termék található összesen. Ehhez használd a Common Table Expression (CTE) technikát!

Majd a CTE eredményét kapcsold össze a `Shop` táblával, és jelenítsd meg az alábbi adatokat:

- Üzlet azonosító
- Üzlet neve
- Az adott üzletben található termékek teljes készlete (`total_stock`)