

Adatbázisrendszerek I. beadandó

Sindely Richárd-P1UV07

2023

Tartalomjegyzék

| | |
|--------------------------------------|----------|
| 1. Bevezetés | 3 |
| 2. ER diagram | 5 |
| 3. RS diagram | 6 |
| 4. SQL | 7 |
| 4.1. Adatbázis létrehozása | 7 |
| 4.2. Adatbázis feltöltése | 9 |
| 4.3. UPDATE parancsok | 9 |
| 4.4. Select parancsok | 10 |

1. Bevezetés

Az adatbázisomhoz a volánbusz rendszere adott inspirációt, amelyben van egy **Utas** egyed, amely a következő tulajdonságokkal rendelkezik:

- uid (PK - Elsődleges kulcs)
- név (Az utas neve)
- szülev (Utas születési éve)
- személyi ig. szám (Az utas személyi ig. száma)
- telefonszám (Utas telefonszáma, többértékű)
- kor (Leszármaztatott érték)
- jegy/bérlet típusa

A következő egyed a **Busz** ami egy adott buszról tartalmaz információkat.

- bid (PK - Elsődleges kulcs)
- rendszám (Busz rendszáma)
- típusszám (Busz típus/gyártási száma)
- márká (Busz gyártója)

Van még egy **Söfőr** egyed ami a céghez tartozó buszsöfőröket tartja számon, melynek a tulajdonságai a következők:

- sid (PK - Elsődleges kulcs)
- név (Söfőr neve)
- kezdés éve (Az év amikor a söfőr elkezdett dolgozni a cégnél)
- ledolgozott évek száma (Leszármaztatott érték)

Következő egyed az a **Járatszám** ami a következő tulajdonságokkal rendelkezik:

- jid (PK - Elsődleges kulcs)
- szám (Járatnak a száma)

Az **Utas** és a **Járatszám** egyedek között van egy kapcsolat ami tartalmaz egy tulajdonságot ami a:

- ülés (Ülés száma ahova az utas ül)

A **Megállók** egyed a következő tulajdonságokkal bír:

- mid (PK - Elsődleges kulcs)
- koordináta (Megálló helyzetének koordinátája)
- név (Megálló neve)

A **Járatszám** és a **Megállók** egyedek között van egy kapcsolat ami kettő tulajdonságot is tartalmaz:

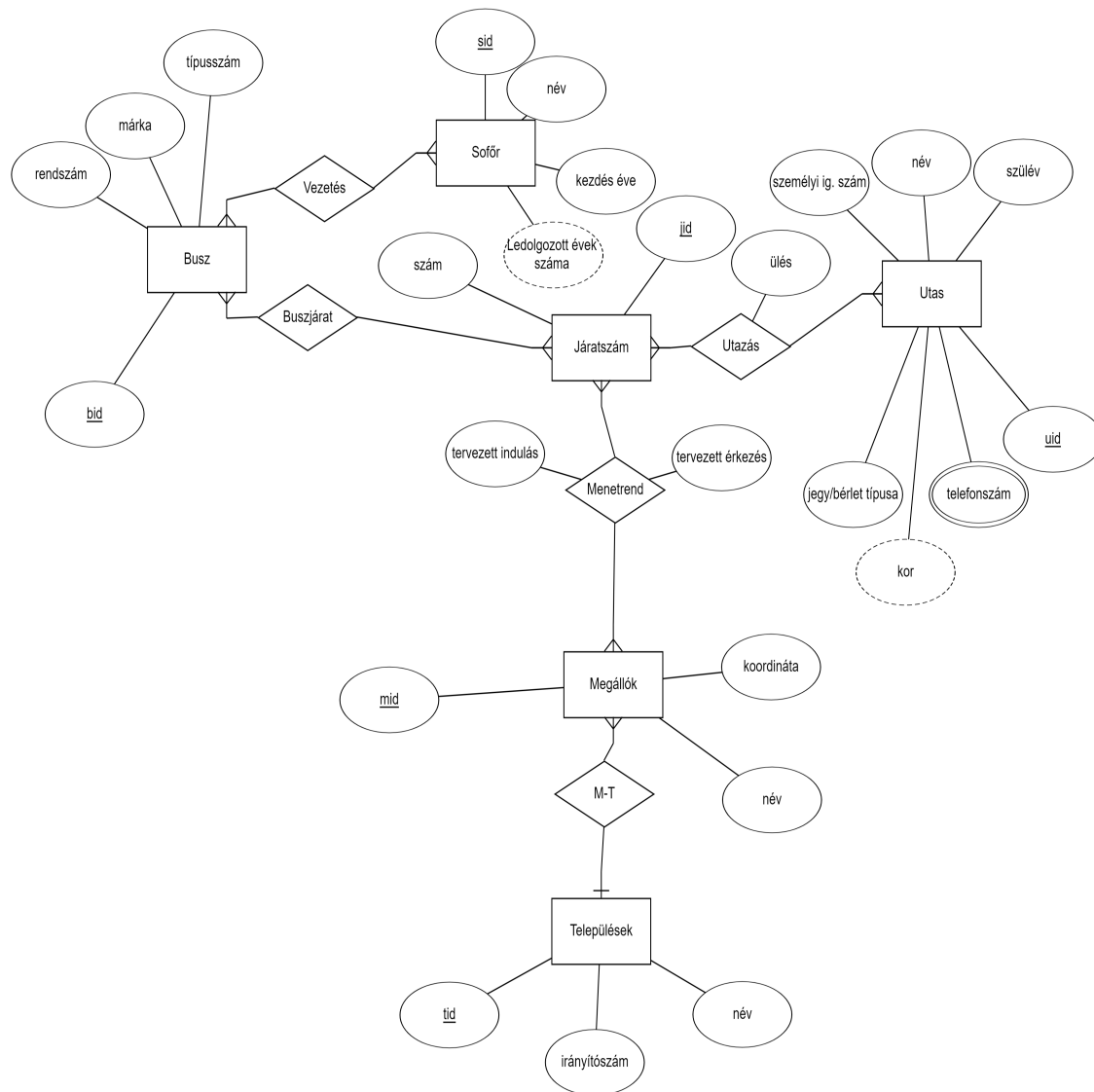
- tervezett indulás
- tervezett érkezés

Végül a **Települések** egyed tartalmazza a települések tulajdonságait amelyek a következők:

- tid (PK - Elsődleges kulcs)
- irányítószám (Települések irányítószáma)
- név (Települések neve)

2. ER diagram

Előzőleg bemutatott egyedek, tulajdonságok ER diagrammja:



4. SQL

4.1. Adatbázis létrehozása

```
CREATE TABLE Utas(  
    utid NUMBER(1) NOT NULL PRIMARY KEY,  
    név varchar(30) NOT NULL,  
    személyi_szám varchar(30) NOT NULL,  
    jegy_bérlet_típus varchar(30) NOT NULL,  
    szülév int NOT NULL  
);  
CREATE TABLE Utas_telefonszám(  
    t_id NUMBER(1) NOT NULL PRIMARY KEY,  
    telefonszám varchar(30),  
    u_id NUMBER(1) NOT NULL,  
    FOREIGN KEY (u_id) REFERENCES Utas(utid)  
);  
CREATE TABLE Járatszám(  
    jid NUMBER(1) NOT NULL PRIMARY KEY,  
    szám int NOT NULL  
);  
CREATE TABLE Utazás(  
    ülés int,  
    ut_id NUMBER(1) NOT NULL,  
    FOREIGN KEY (ut_id) REFERENCES Utas(utid)  
);  
CREATE TABLE Sofőr(  
    soid NUMBER(1) NOT NULL PRIMARY KEY,  
    név varchar(30) NOT NULL,  
    kezdés_éve int NOT NULL  
);  
CREATE TABLE Busz(  
    bid NUMBER(1) NOT NULL PRIMARY KEY,  
    márka varchar(30) NOT NULL,  
    típusszám int NOT NULL,  
    rendszám varchar(30) NOT NULL  
);  
CREATE TABLE Vezetés(  

```

```
s_id NUMBER(1) NOT NULL,
b_id NUMBER(1) NOT NULL,
FOREIGN KEY (s_id) REFERENCES Sofőr(soid),
FOREIGN KEY (b_id) REFERENCES Busz(bid)
);
CREATE TABLE Települések(
    tid NUMBER(1) NOT NULL PRIMARY KEY,
    név varchar(30) NOT NULL,
    irányítószám int NOT NULL
);
CREATE TABLE Megállók(
    mid NUMBER(1) NOT NULL PRIMARY KEY,
    koordináta varchar(100) NOT NULL,
    név varchar(30) NOT NULL,
    t_id NUMBER(1) NOT NULL,
    FOREIGN KEY (t_id) REFERENCES Települések(tid)
);
CREATE TABLE Menetrend(
    tervezett_indulás_óra int,
    tervezett_indulás_perc int,
    tervezett_érkezés_óra int,
    tervezett_érkezés_perc int,
    j_id NUMBER(1) NOT NULL,
    m_id NUMBER(1) NOT NULL,
    FOREIGN KEY (j_id) REFERENCES Járatszám(jid),
    FOREIGN KEY (m_id) REFERENCES Megállók(mid)
);
CREATE TABLE Buszjárat(
    b_id NUMBER(1) NOT NULL,
    j_id NUMBER(1) NOT NULL,
    FOREIGN KEY (b_id) REFERENCES Busz(bid),
    FOREIGN KEY (j_id) REFERENCES Járatszám(jid)
);
```


4.2. Adatbázis feltöltése

```
INSERT INTO Utas VALUES(1,'Kiss Tamás','1356EA','Diák',2004);
INSERT INTO Utas_telefonszám VALUES(1,'06703459823',1);
INSERT INTO Járatszám VALUES(1,1359);
INSERT INTO Utazás VALUES(22,1);
INSERT INTO Sofőr VALUES(1,'Benedek Ferenc',2013);
INSERT INTO Busz VALUES(1,'Volvo',17346,'RGA-578');
INSERT INTO Vezetés VALUES(1,1);
INSERT INTO Települések VALUES(1,'Dunaharaszti',2330);
INSERT INTO Megállók VALUES(1,'12735E-2372737N','Dunaharaszti, Kisbolt',1);
INSERT INTO Menetrend VALUES(12,45,12,43,1,1);
INSERT INTO Buszjárat VALUES(1,1);
```

4.3. UPDATE parancsok

Busz táblában a két rendszám megegyezik. Két busz van a táblában. A Volvo rendszáma hibás. Megoldás:

```
UPDATE Busz SET rendszám='DBA-972' WHERE rendszám like 'RGA-578' AND márka like 'Volvo';
```

Járatszámnál 1-2 ID-vel rendelkezőknek ugyanaz a járatszáma, illetve 3-4-nek is. A 2-es ID-vel rendelkező legyen 980, a 4-es ID-vel rendelkező pedig legyen 1501.

```
UPDATE Járatszám SET szám=980 WHERE jid=2
UPDATE Járatszám SET szám=1501 WHERE jid=4
```

Az Utas táblában el van írva a

jegy_bérlet_típus

Megoldás:

```
ALTER TABLE Utas RENAME COLUMN jegy_bérlet_típus TO jegy_bérlet_típus
```

Utazás táblából hiányzik egy FK ami a Járatszámhoz kapcsolódik. Megoldás:

```
ALTER TABLE Utazás ADD ja_id int;
ALTER TABLE Utazás ADD FOREIGN KEY (ja_id) REFERENCES Járatszám(jid);
```

Most hozzáadjuk a megfelelő értékeket:

```
UPDATE Utazás SET ja_id=1 WHERE ut_id=1;
UPDATE Utazás SET ja_id=2 WHERE ut_id=2;
UPDATE Utazás SET ja_id=3 WHERE ut_id=3;
```

4.4. Select parancsok

2000 előtt született Utasok:

```
SELECT név from Utas WHERE szülev<2000;
```

2000 után született utasok száma:

```
SELECT Count(név) from Utas WHERE szülev>2000
```

Taksony településnek az irányítószáma:

```
SELECT irányítószám from Települések WHERE név like 'Taksony';
```

Az összes sofőr lekérdezése:

```
SELECT név from Sofőr;
```

Összes sofőr lekérdezése aki 2010 előtt kezdett el dolgozni:

```
SELECT név from Sofőr WHERE kezdés_éve<2010;
```

Teljes árú jegy/bérletet vett utasok neve:

```
SELECT név from Utas WHERE jegy_bérlet_típus like 'Teljes árú';
```

Volvo típusú buszok rendszámának lekérdezése:

```
SELECT rendszám from Busz WHERE márka like 'Volvo';
```

Az utasok között mennyi a diák?

```
SELECT Count(*) from Utas WHERE jegy_bérlet_típus like 'Diák';
```

Az 1359-es járatszámú busznak hány megállója van?

```
SELECT Count(*) from Járatszám JOIN Menetrend on jid=j_id JOIN Megállók  
ON m_id=mid WHERE szám=1359;
```

Az 1359-es járatszámú busznak megállóinak a nevei:

```
SELECT Megállók.név from Járatszám JOIN Menetrend on jid=j_id JOIN Megállók  
ON m_id=mid WHERE szám=1359;
```

Dunaharaszti átmenő buszok járatszáma:

```
SELECT szám from Járatszám JOIN Menetrend on jid=j_id JOIN Megállók ON  
m_id=mid JOIN Települések ON tid=t_id WHERE Települések.név like 'Dunaharaszti';
```

Hány busz megy át Taksony településen?

```
SELECT Count(szám) from Járatszám JOIN Menetrend on jid=j_id JOIN Megállók  
ON m_id=mid JOIN Települések ON tid=t_id WHERE Települések.név like 'Taksony';  
1359-es járatszámú buszon utazók száma:
```

```
SELECT Count(*) from Járatszám JOIN Utazás ON jid=ja_id JOIN Utas on utid=ut_id  
WHERE szám=1359;
```

Ha tudjuk hogy egy járatszámú busz megy csak Taksonyba, akkor hány utas van rajta?

```
SELECT Count(*) from Járatszám JOIN Utazás ON jid=ja_id JOIN Utas on utid=ut_id  
WHERE szám=(SELECT szám from Járatszám JOIN Menetrend on jid=j_id JOIN  
Megállók ON m_id=mid JOIN Települések ON tid=t_id WHERE Települések.név  
like 'Taksony');
```