

JEGYZŐKÖNYV

Adatbázisrendszerek I.

Féléves feladat

Étterem adatbázis

Készítette: **Fekete Máté**

Neptunkód: **JR9KY7**

A feladat leírása:

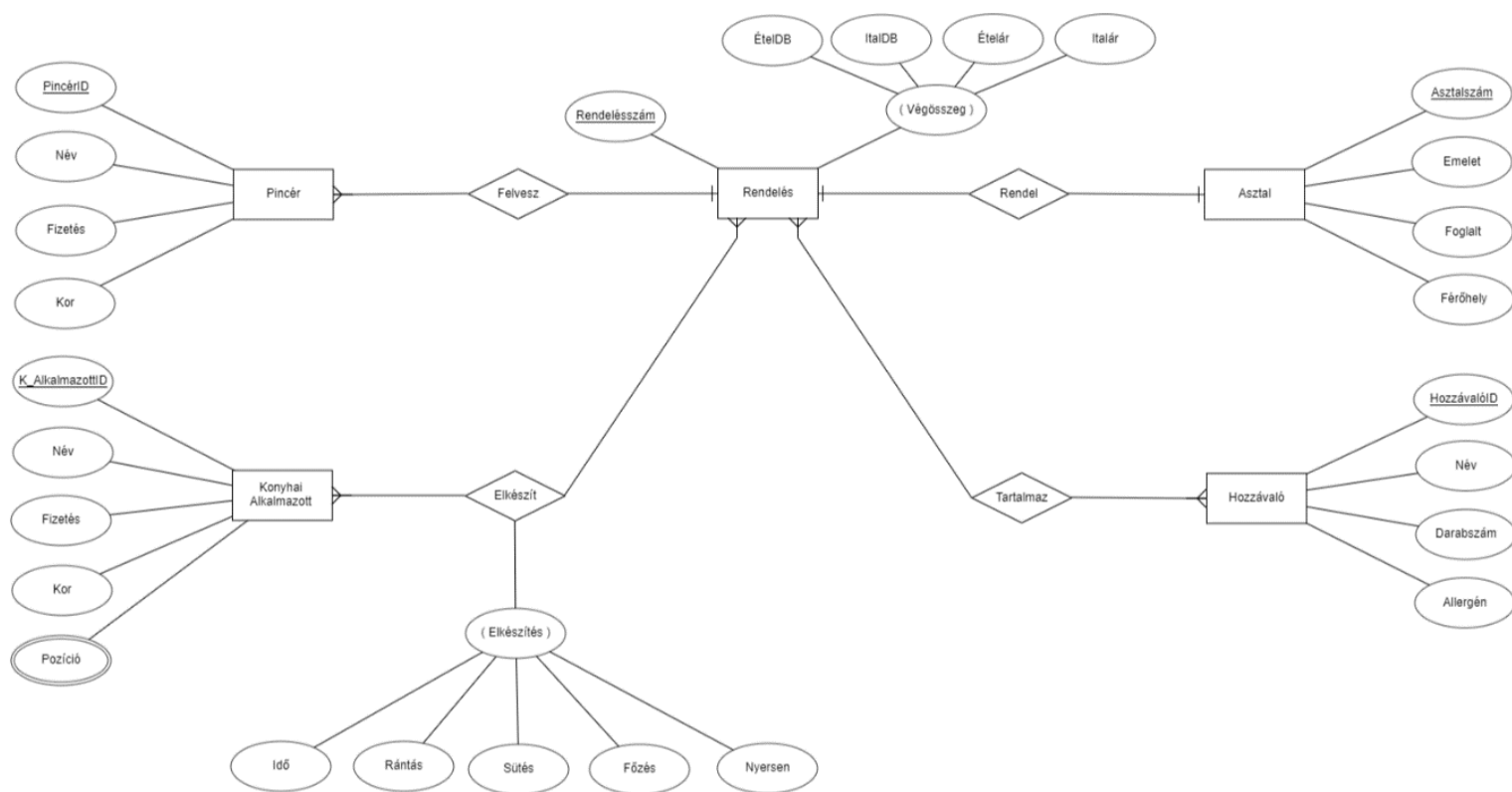
A feladatban létre kellett hozni egy adatbázist, mely megfelel a követelményeknek. Ilyen kritériumok közé tartoznak, hogy legyen legalább 5 egyed és mindegyiknek legyen legalább 4 tulajdonsága, ezeket kapcsolatok kössék össze, melyek között legyen egy-egy (1:1), egy-több (1:N) és több-több (N:M) kapcsolat is. A több-több kapcsolatnak pedig legyenek további tulajdonságai. Először egy ER modellt kell kidolgozni, ahol feltüntetjük az egyedeket, azok tulajdonságait, a kapcsolatokat és ha van akkor annak tulajdonságait. Itt többféle tulajdonság is szerepel. Miután elkészült az ER modell átkonvertáltam relációs modellé és készítettem róla egy relációs sémát is. A konvertálás befejeztével létrehoztam egy MySQL adatbázist és elkezdtem SQL parancsokkal kialakítani a táblákat, azokat feltölteni és lekérdezéseket készíteni.

ER modell leírása:

Itt egy reális ER adatbázist hozok létre, melynek jellemzői a következők:

- RENDELÉS EGYED:
 - **Rendelésszám:** egyedi tulajdonság és azonosító.
 - **Végösszeg** származtatott tulajdonság, ami az **ÉtelDB**, **ItalDB**, **Ételár**, **Italár** tulajdonságokból áll össze ($\text{ÉtelDB} * \text{Ételár} + \text{ItalDB} * \text{Italár}$).
- ASZTAL EGYED:
 - **Asztalszám:** egyedi tulajdonság és azonosító.
 - **Emelet** tulajdonság: az asztal hányadik emeleten található az étteremben.
 - **Foglalt** tulajdonság: azt mutatja, hogy éppen foglalt-e az asztal vagy sem.
 - **Férőhely** tulajdonság: az asztalnál egyszerre elférő emberek száma.
- PINCÉR EGYED:
 - **PincérID** tulajdonság: egyedi tulajdonság és azonosító.
 - **Név** tulajdonság: a pincér neve.
 - **Fizetés** tulajdonság: a pincér fizetése.
 - **Kor** tulajdonság: a pincér életkora.

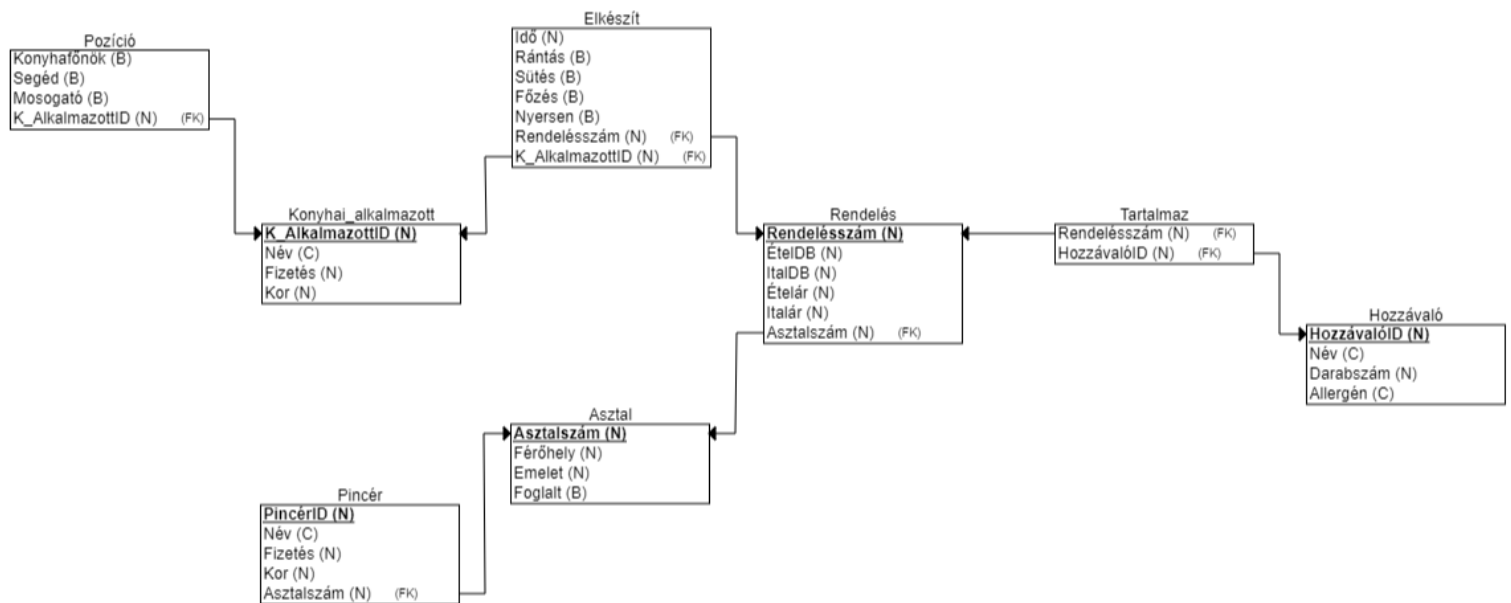
- KONYHAI ALKALMAZOTT EGYED:
 - **K_AlkalmazottID** tulajdonság: egyedi tulajdonság és azonosító.
 - **Név** tulajdonság: a konyhai alkalmazott neve.
 - **Fizetés** tulajdonság: a konyhai alkalmazott fizetése.
 - **Kor** tulajdonság: a konyhai alkalmazott életkora.
 - **Pozíció** tulajdonság: többértékű tulajdonság, a konyhai alkalmazott pozícióját jelzi.
- HOZZÁVALÓ EGYED:
 - **HozzávalóID** tulajdonság: egyedi tulajdonság és azonosító.
 - **Név** tulajdonság: a hozzávaló neve.
 - **Darabszám** tulajdonság: hány darab van az adott hozzávalóból raktáron.
 - **Allergén** tulajdonság: a hozzávalóban található allergén neve, lehet NULL érték is, akkor nem tartalmaz allergént.
- RENDEL KAPCSOLAT:
 - Egy-egy (1:1) kapcsolat, ami a Rendelés és az Asztal egyedeket köti össze.
- FELVESZ KAPCSOLAT:
 - Egy-több (1:N) kapcsolat, ami a Pincér (N) és Rendelés (1) egyedeket köti össze.
- TARTALMAZ KAPCSOLAT:
 - Több-több (N:M) kapcsolat, a Rendelés és a Hozzávaló egyedeket köti össze.
- ELKÉSZÍT KAPCSOLAT:
 - Több-több (N:M) kapcsolat, ami a Rendelés és a Konyhai Alkalmazott egyedeket köti össze.
 - Elkészítés összetett tulajdonsága, melyhez kapcsolódnak a következő tulajdonságok:
 - **Idő:** a rendelés elkészítési ideje percben
 - **Rántás:** tartalmaz-e a rendelés rántott ételt.
 - **Sütés:** tartalmaz-e a rendelés sült ételt.
 - **Főzés:** tartalmaz-e a rendelés főtt ételt.
 - **Nyersen:** tartalmaz-e a rendelés nyers ételt.



Adatbázis átkonvertálása relációs modellre:

- Az Asztal egyedből egy táblázat lesz, amibe belekerül az összes tulajdonsága.
- A Rendelés egyedből egy táblázat lesz, amibe bekerül a Rendelészám, ÉteldB, ItalDB, Ételár, Italár és az Asztalszám, mely egy idegen kulcs.
- A Pincér egyedből egy táblázat lesz, amibe belekerül az összes tulajdonsága és az Asztalszám, mit idegen kulcs.
- A Hozzávaló egyedből egy tábla lesz, amibe belekerül az összes tulajdonsága.
- A Konyhai Alkalmazott egyedből egy táblázat lesz, amibe belekerül az összes tulajdonsága kivétel a Pozíció, ami egy összetett tulajdonság.
- A Pozíció összetett tulajdonságból egy tábla lesz, amibe a következő elemek kerülnek: Konyhafőnök, Segéd, Mosogató, melyek boolean típusúak. Ezeken kívül egy K_AlkalmazottiD is, mint idegen kulcs.

- Az Elkészít (N:M) kapcsolatból egy tábla lesz, melybe bekerülnek az Elkészítés összetett tulajdonság összes tulajdonsága, valamint ezek mellett a Rendelésszám és K_AlkalmazottiID, melyek idegen kulcsok.
- A Tartalmaz (N:M) kapcsolatból egy tábla lesz, amibe a Rendelésszám és a HozzávalóID kerül. Ezek mind idegen kulcsok.



Adatbázis relációs sémája:

- Asztal [Asztalszám, Férőhely, Emelet, Foglalt]
- Rendelés [Rendelésszám, ÉtelDB, ItalDB, Ételár, Italár, Asztalszám]
- Pincér [PincérID, Név, Fizetés, Kor, Asztalszám]
- Hozzávaló [HozzávalóID, Név, Darabszám, Allergén]
- Konyhai_alkalmazott [K_AlkalmazottiID, Név, Fizetés, Kor]
- Pozíció [Konyhafőnök, Segéd, Mosogató, K_AlkalmazottiID]
- Elkészít [Idő, Rántás, Sütés, Főzés, Nyersen, Rendelésszám, K_AlkalmazottiID]
- Tartalmaz [Rendelésszám, HozzávalóID]

Táblák létrehozása:

A táblák létrehozásánál figyelembe kell venni a sorrendet az idegen kulcsok létrehozása miatt. A következő SQL pancsokkal hoztam létre a táblákat sorrendben:

- `CREATE TABLE Asztal (asztalszam int primary key, ferohely int check(ferohely > 1), emelet int check(emelet >= 0), foglalt boolean default FALSE);`
- `CREATE TABLE Rendeles (rendelesszam int primary key, etelID int check(etelID >= 0), italID int check(italID >= 0), etelar int check(etelar >= 0), italar int check(italar >= 0), asztalszam int, foreign key (asztalszam) references Asztal(asztalszam));`
- `CREATE TABLE Pincer (pincerID int primary key, nev varchar(35), fizetes int check(fizetes > 200000), kor int check(kor >= 18), asztalszam int, foreign key (asztalszam) references Asztal(asztalszam));`
- `CREATE TABLE Hozzavalo (hozzavaloID int primary key, nev varchar(30), darabszam int check(darabszam >= 0), allergen varchar(20));`
- `CREATE TABLE Tartalmaz (rendelesszam int, hozzavaloID int, foreign key (rendelesszam) references Rendeles(rendelesszam), foreign key (hozzavaloID) references Hozzavalo(hozzavaloID));`
- `CREATE TABLE Konyhai_alkalmazott (k_alkalmazottID int primary key, nev varchar(35), fizetes int check(fizetes > 200000), kor int check(kor >= 18));`
- `CREATE TABLE Elkeszit (ido int check(ido > 10), rantas boolean default FALSE, sutas boolean default FALSE, fozes boolean default FALSE, nyersen boolean default FALSE, rendelesszam int, k_alkalmazottID int, foreign key (rendelesszam) references Rendeles(rendelesszam), foreign key (k_alkalmazottID) references Konyhai_alkalmazott(k_alkalmazottID));`

- CREATE TABLE Pozicio (konyhafonok boolean default FALSE, seged boolean default FALSE, mosogato boolean default FALSE, k_alkalmazottiID int, foreign key (k_alkalmazottiID) references Konyhai_alkalmazott(k_alkalmazottiID));

Táblák feltöltése:

Asztal tábla:

- INSERT INTO Asztal VALUES (1, 6, 0, FALSE);
- INSERT INTO Asztal VALUES (2, 2, 1, TRUE);
- INSERT INTO Asztal VALUES (3, 2, 1, TRUE);
- INSERT INTO Asztal VALUES (4, 10, 0, FALSE);
- INSERT INTO Asztal VALUES (5, 4, 0, TRUE);
- INSERT INTO Asztal VALUES (6, 4, 1, TRUE);
- INSERT INTO Asztal VALUES (7, 8, 0, FALSE);
- INSERT INTO Asztal VALUES (8, 4, 0, FALSE);
- INSERT INTO Asztal VALUES (9, 2, 1, TRUE);
- INSERT INTO Asztal VALUES (10, 2, 1, TRUE);
- INSERT INTO Asztal VALUES (11, 8, 0, TRUE);

Rendelés tábla:

- INSERT INTO Rendeles VALUES (1, 2, 4, 3000, 500, 2);
- INSERT INTO Rendeles VALUES (2, 2, 2, 5000, 600, 3);
- INSERT INTO Rendeles VALUES (3, 0, 4, 0, 800, 5);
- INSERT INTO Rendeles VALUES (4, 4, 0, 2800, 0, 6);
- INSERT INTO Rendeles VALUES (5, 0, 4, 0, 1300, 9);
- INSERT INTO Rendeles VALUES (6, 2, 6, 10000, 750, 10);
- INSERT INTO Rendeles VALUES (7, 8, 12, 4600, 750, 11);

Pincér tábla:

- INSERT INTO Pincer VALUES (1, 'Szabo Laszlo', 320000, 36, 2);
- INSERT INTO Pincer VALUES (2, 'Szabo Laszlo', 320000, 36, 3);
- INSERT INTO Pincer VALUES (3, 'Kovacs Istvan', 230000, 27, 5);
- INSERT INTO Pincer VALUES (4, 'Kovacs Istvan', 230000, 27, 6);
- INSERT INTO Pincer VALUES (5, 'Lukacs Balazs', 260000, 21, 9);
- INSERT INTO Pincer VALUES (6, 'Lukacs Balazs', 260000, 21, 10);

Hozzávalók tábla:

- INSERT INTO Hozzavalo VALUES (1, 'teszta', 20, 'gluten');
- INSERT INTO Hozzavalo VALUES (2, 'tej', 20, 'laktoz');
- INSERT INTO Hozzavalo VALUES (3, 'vaj', 32, 'laktoz');
- INSERT INTO Hozzavalo VALUES (4, 'liszt', 32, 'gluten');
- INSERT INTO Hozzavalo VALUES (5, 'liszt', 15, NULL);
- INSERT INTO Hozzavalo VALUES (6, 'marhahus', 23, NULL);
- INSERT INTO Hozzavalo VALUES (7, 'csirkehus', 35, NULL);
- INSERT INTO Hozzavalo VALUES (8, 'vadhus', 27, NULL);
- INSERT INTO Hozzavalo VALUES (9, 'rizs', 36, NULL);
- INSERT INTO Hozzavalo VALUES (10, 'burgonya', 21, NULL);
- INSERT INTO Hozzavalo VALUES (11, 'alkohol', 70, NULL);
- INSERT INTO Hozzavalo VALUES (12, 'alkohol mentes', 70, NULL);

Tartalmaz tábla:

- INSERT INTO Tartalmaz VALUES (1, 1);
- INSERT INTO Tartalmaz VALUES (1, 3);
- INSERT INTO Tartalmaz VALUES (1, 4);
- INSERT INTO Tartalmaz VALUES (1, 7);
- INSERT INTO Tartalmaz VALUES (1, 9);
- INSERT INTO Tartalmaz VALUES (1, 12);
- INSERT INTO Tartalmaz VALUES (2, 8);
- INSERT INTO Tartalmaz VALUES (2, 10);
- INSERT INTO Tartalmaz VALUES (2, 11);
- INSERT INTO Tartalmaz VALUES (3, 11);
- INSERT INTO Tartalmaz VALUES (3, 12);
- INSERT INTO Tartalmaz VALUES (4, 1);
- INSERT INTO Tartalmaz VALUES (4, 3);
- INSERT INTO Tartalmaz VALUES (4, 7);
- INSERT INTO Tartalmaz VALUES (4, 9);
- INSERT INTO Tartalmaz VALUES (4, 2);
- INSERT INTO Tartalmaz VALUES (5, 11);
- INSERT INTO Tartalmaz VALUES (6, 12);
- INSERT INTO Tartalmaz VALUES (6, 11);
- INSERT INTO Tartalmaz VALUES (6, 10);
- INSERT INTO Tartalmaz VALUES (6, 8);
- INSERT INTO Tartalmaz VALUES (6, 5);
- INSERT INTO Tartalmaz VALUES (6, 9);
- INSERT INTO Tartalmaz VALUES (6, 7);
- INSERT INTO Tartalmaz VALUES (7, 3);
- INSERT INTO Tartalmaz VALUES (7, 4);
- INSERT INTO Tartalmaz VALUES (7, 6);

- INSERT INTO Tartalmaz VALUES (7, 7);
- INSERT INTO Tartalmaz VALUES (7, 8);
- INSERT INTO Tartalmaz VALUES (7, 9);
- INSERT INTO Tartalmaz VALUES (7, 10);
- INSERT INTO Tartalmaz VALUES (7, 11);
- INSERT INTO Tartalmaz VALUES (7, 12);

Konyhai Alkalmazott tábla:

- INSERT INTO Konyhai_Alkalmazott VALUES (1, 'Dobos Norbert', 870000, 38);
- INSERT INTO Konyhai_Alkalmazott VALUES (2, 'Szoboszlai Dominik', 340000, 29);
- INSERT INTO Konyhai_Alkalmazott VALUES (3, 'Vallai Balint', 310000, 22);
- INSERT INTO Konyhai_Alkalmazott VALUES (4, 'Balogh Geza', 326000, 25);
- INSERT INTO Konyhai_Alkalmazott VALUES (5, 'Kovacs Noemi', 378000, 24);
- INSERT INTO Konyhai_Alkalmazott VALUES (6, 'Foldi Anita', 248000, 20);
- INSERT INTO Konyhai_Alkalmazott VALUES (7, 'Kiss Bela', 248000, 19);
- INSERT INTO Konyhai_Alkalmazott VALUES (8, 'Nagy Andras', 248000, 21);

Pozíció tábla:

- INSERT INTO Pozicio VALUES (TRUE, FALSE, FALSE, 1);
- INSERT INTO Pozicio VALUES (FALSE, TRUE, FALSE, 2);
- INSERT INTO Pozicio VALUES (FALSE, TRUE, FALSE, 3);
- INSERT INTO Pozicio VALUES (FALSE, TRUE, FALSE, 4);
- INSERT INTO Pozicio VALUES (FALSE, TRUE, FALSE, 5);
- INSERT INTO Pozicio VALUES (FALSE, FALSE, TRUE, 6);
- INSERT INTO Pozicio VALUES (FALSE, FALSE, TRUE, 7);
- INSERT INTO Pozicio VALUES (FALSE, FALSE, TRUE, 8);

Elkészít tábla:

- INSERT INTO Elkeszit VALUES (45, FALSE, TRUE, TRUE, FALSE, 1, 2);
- INSERT INTO Elkeszit VALUES (47, FALSE, FALSE, TRUE, TRUE, 2, 3);
- INSERT INTO Elkeszit VALUES (47, FALSE, FALSE, TRUE, TRUE, 2, 1);
- INSERT INTO Elkeszit VALUES (12, FALSE, FALSE, FALSE, FALSE, 3, 4);
- INSERT INTO Elkeszit VALUES (60, TRUE, TRUE, TRUE, TRUE, 4, 5);
- INSERT INTO Elkeszit VALUES (15, FALSE, FALSE, FALSE, FALSE, 5, 5);
- INSERT INTO Elkeszit VALUES (50, FALSE, TRUE, TRUE, TRUE, 6, 1);
- INSERT INTO Elkeszit VALUES (70, TRUE, TRUE, TRUE, FALSE, 7, 1);

A rendelés táblába bevitettem egy végösszeg oszlopot, ami a rendelés teljes árát tükrözi és ezt ki is számítottam a táblázatok feltöltése után. A használt SQL parancsok:

- ALTER TABLE Rendeles ADD vegosszeg int;
- UPDATE Rendeles SET vegosszeg = (etelDB * etelar) + (italDB * italar);

Lekérdezések:

1. Kérdezzük le az összes pincér nevét és fizetésüket!

SELECT nev, fizetes FROM Pincer GROUP BY nev;

$\pi_{nev, fizetes}$

$\gamma_{nev, pincer}$

```
MariaDB [feleves]> SELECT nev, fizetes FROM Pincer GROUP BY nev;
+-----+-----+
| nev          | fizetes |
+-----+-----+
| Kovacs Istvan | 230000  |
| Lukacs Balazs | 260000  |
| Szabo Laszlo  | 320000  |
+-----+-----+
3 rows in set (0.001 sec)
```

2. Kérdezzük le azoknak a konyhai alkalmazottaknak a nevét, korát és fizetését, akiknek a fizetése több, mint 350000 Ft!

SELECT nev, kor, fizetes FROM Konyhai_Alkalmazott WHERE fizetes > 350000;

$\pi_{nev, kor, fizetes}$

$\sigma_{fizetes > 350000} konyhai_alkalmazott$

```
MariaDB [feleves]> SELECT nev, kor, fizetes FROM Konyhai_Alkalmazott WHERE fizetes > 350000;
+-----+-----+-----+
| nev          | kor  | fizetes |
+-----+-----+-----+
| Dobos Norbert | 38   | 870000  |
| Kovacs Noemi  | 24   | 378000  |
+-----+-----+-----+
2 rows in set (0.000 sec)
```

3. Kérdezzük le azon konyhai alkalmazottak neveit, akik 22 évnél idősebbek!

SELECT nev, kor FROM Konyhai_Alkalmazott WHERE kor < 22;

$\pi_{nev, kor}$

$\sigma_{kor < 22}$ *konyhai_alkalmazott*

```
MariaDB [feleves]> SELECT nev, kor FROM Konyhai_Alkalmazott WHERE kor < 22;
+-----+-----+
| nev      | kor  |
+-----+-----+
| Foldi Anita | 20  |
| Kiss Bela  | 19  |
| Nagy Andras | 21  |
+-----+-----+
3 rows in set (0.000 sec)
```

4. Kérdezzük le a rendelések végösszegeinek átlagárát!

SELECT AVG(vegosszeg) FROM Rendeles;

$\pi_{AVG(vegosszeg)}$

$\gamma_{AVG(vegosszeg)}$ *rendeles*

```
MariaDB [feleves]> SELECT AVG(vegosszeg) FROM Rendeles;
+-----+
| AVG(vegosszeg) |
+-----+
| 15585.7143     |
+-----+
1 row in set (0.001 sec)
```

5. Kérdezzük le minden asztalhoz a hozzá tartozó rendelés végösszegét!

SELECT Asztal.asztalszam, Rendeles.vegosszeg FROM Asztal INNER JOIN Rendeles ON Asztal.asztalszam = Rendeles.asztalszam;

$\pi_{asztal.asztalszam, rendeles.vegosszeg}$ (*asztal*

$\bowtie_{asztal.asztalszam = rendeles.asztalszam}$ *rendeles*)

```
MariaDB [feleves]> SELECT Asztal.asztalszam, Rendeles.vegosszeg FROM Asztal INNER JOIN Rendeles ON Asztal.asztalszam = Rendeles.asztalszam;
+-----+-----+
| asztalszam | vegosszeg |
+-----+-----+
| 2          | 8000      |
| 3          | 11200     |
| 5          | 3200      |
| 6          | 11200     |
| 9          | 5200      |
| 10         | 24500     |
| 11         | 45800     |
+-----+-----+
7 rows in set (0.040 sec)
```

6. Kérdezzük le azokat a rendelésszámokat és a hozzá tartozó végösszeget, melyekhez tejet használtak fel!

SELECT rendeleesszam, vegosszeg **FROM** Rendeles **WHERE** rendeleesszam **IN** (SELECT rendeleesszam **FROM** Tartalmaz **WHERE** hozzavaloID = (SELECT hozzavaloID **FROM** Hozzavalo **WHERE** nev = 'tej'));

```
MariaDB [feleves]> SELECT rendeleesszam, vegosszeg FROM Rendeles WHERE rendeleesszam IN (SELECT rendeleesszam FROM Tartalmaz WHERE hozzavaloID = (SELECT hozzavaloID FROM Hozzavalo WHERE nev = 'tej'));
+-----+-----+
| rendeleesszam | vegosszeg |
+-----+-----+
|          4    |    11200  |
+-----+-----+
1 row in set (0.034 sec)
```

7. Kérdezzük le azokat az asztalszámokat és hozzá tartozó férőhelyeket, emeletet, rendelésszámot és végösszeget, ahol a legmagasabb fizetésű pincér szolgál fel!

SELECT Asztal.asztalszam, Asztal.ferohely, Asztal.emelet, Rendeles.rendelesszam, Rendeles.vegosszeg **FROM** Asztal **INNER JOIN** Rendeles **ON** Asztal.asztalszam = Rendeles.asztalszam **WHERE** Asztal.asztalszam **IN** (SELECT Pincer.asztalszam **FROM** Pincer **WHERE** Pincer.fizetes = (SELECT MAX(fizetes) **FROM** Pincer));

```
MariaDB [feleves]> SELECT Asztal.asztalszam, Asztal.ferohely, Asztal.emelet, Rendeles.rendelesszam, Rendeles.vegosszeg FROM Asztal INNER JOIN Rendeles ON Asztal.asztalszam = Rendeles.asztalszam WHERE Asztal.asztalszam IN (SELECT Pincer.asztalszam FROM Pincer WHERE Pincer.fizetes = (SELECT MAX(fizetes) FROM Pincer));
+-----+-----+-----+-----+-----+
| asztalszam | ferohely | emelet | rendeleesszam | vegosszeg |
+-----+-----+-----+-----+-----+
|          2 |        2 |      1 |          1    |    8000    |
|          3 |        2 |      1 |          2    |   11200    |
+-----+-----+-----+-----+-----+
2 rows in set (0.016 sec)
```

8. Kérdezzük le azokat a rendelésszámokat és a hozzá tartozó végösszeget, melyeket a konyhafőnök készített el!

SELECT rendeleesszam, vegosszeg **FROM** Rendeles **WHERE** rendeleesszam **IN** (SELECT rendeleesszam **FROM** Elkeszit **WHERE** k_alkalmazottiID = (SELECT k_alkalmazottiID **FROM** Konyhai_Alkalmazott **WHERE** k_alkalmazottiID = (SELECT k_alkalmazottiID **FROM** Pozicio **WHERE** konyhafonok = 1)));

```
MariaDB [feleves]> SELECT rendeleesszam, vegosszeg FROM Rendeles WHERE rendeleesszam IN (SELECT rendeleesszam FROM Elkeszit WHERE k_alkalmazottiID = (SELECT k_alkalmazottiID FROM Konyhai_Alkalmazotti WHERE k_alkalmazottiID = (SELECT k_alkalmazottiID FROM Pozicio WHERE konyhafonok = 1)));
```

| rendeleesszam | vegosszeg |
|---------------|-----------|
| 2 | 11200 |
| 6 | 24500 |
| 7 | 45800 |

```
3 rows in set (0.001 sec)
```

9. Kérdezzük le, hogy hány darab konyhai segéd használt fel vadhúst!

```
SELECT COUNT(k_alkalmazottiID) AS Vadhust_keszito_segit_DB FROM Pozicio WHERE k_alkalmazottiID IN (SELECT k_alkalmazottiID FROM Elkeszit WHERE Elkeszit.rendelesszam IN (SELECT rendeleesszam FROM Tartalmaz WHERE hozzavaloid = (SELECT hozzavaloid FROM Hozzavaloid WHERE nev = 'vadhust')) GROUP BY k_alkalmazottiID) AND seged = 1;
```

```
MariaDB [feleves]> SELECT COUNT(k_alkalmazottiID) AS Vadhust_keszito_segit_DB FROM Pozicio WHERE k_alkalmazottiID IN (SELECT k_alkalmazottiID FROM Elkeszit WHERE Elkeszit.rendelesszam IN (SELECT rendeleesszam FROM Tartalmaz WHERE hozzavaloid = (SELECT hozzavaloid FROM Hozzavaloid WHERE nev = 'vadhust')) GROUP BY k_alkalmazottiID) AND seged = 1;
```

| Vadhust_keszito_segit_DB |
|--------------------------|
| 1 |

```
1 row in set (0.001 sec)
```

10. Kérdezzük le azon pincérek nevét és korát, akik a konyhafőnök által elkészített ételt szolgálják fel, illetve az asztalszámot, az asztal férőhelyét és azt, hogy melyik emeleten található az asztal!

```
SELECT Pincer.nev, Pincer.kor, Asztal.asztalszam, Asztal.ferohely, Asztal.emelet FROM Asztal INNER JOIN Pincer ON Asztal.asztalszam = Pincer.asztalszam WHERE Asztal.asztalszam IN (SELECT rendeleesszam FROM Elkeszit WHERE k_alkalmazottiID = (SELECT k_alkalmazottiID FROM Pozicio WHERE konyhafonok = 1));
```

```
MariaDB [feleves]> SELECT Pincer.nev, Pincer.kor, Asztal.asztalszam, Asztal.ferohely, Asztal.emelet FROM Asztal INNER JOIN Pincer ON Asztal.asztalszam = Pincer.asztalszam WHERE Asztal.asztalszam IN (SELECT rendeleesszam FROM Elkeszit WHERE k_alkalmazottiID = (SELECT k_alkalmazottiID FROM Pozicio WHERE konyhafonok = 1));
```

| nev | kor | asztalszam | ferohely | emelet |
|---------------|-----|------------|----------|--------|
| Szabo Laszlo | 36 | 2 | 2 | 1 |
| Kovacs Istvan | 27 | 6 | 4 | 1 |

```
2 rows in set (0.001 sec)
```