

Jegyzőkönyv
Adatbázis rendszerek I.
Féléves feladat

Készítette: **Földesi Dominik**

Neptunkód: **G616TL**

Gyak: **Kedd 8-10**

Gyakvezető: **Dr. Bednarik László**

A feladat leírása:

A feladatom egy olyan adatbázis megtervezése, amely Magyarország egyes pékségeit kezeli. Az adatbázisban rá tudunk keresni pékségekre, az árult termékekre, pékekre. A pékségekben kiszállítás is üzemel, így a futárokat és a vevőket is tároljuk az adatbázisban.

Az ER modell egyedei és tulajdonságai:

- **A termék egyed tulajdonságai:**
 - TermékID: A termék egyed elsődleges kulcsa
 - Név: A termék neve
 - Ár: A termék ára.
- **A pékség egyed tulajdonságai:**
 - PékségID: A pékség egyed elsődleges kulcsa.
 - Név: Egyes pékségek megnevezései.
 - Nyitvatartás: A pékségek nyitvatartási ideje.
 - Elérhetőség: Üzletek elérhetőségei. Összetett tulajdonság.
- **A vevő egyed tulajdonságai**
 - VevőID: A vevő egyed elsődleges kulcsa.
 - Név: A vevő neve.
 - Cím: A vevő címe. Összetett tulajdonság.
 - Telefonszám: A vevő telefonszáma.
- **A futár egyed tulajdonságai**
 - FutárID: A futár egyed elsődleges kulcsa.
 - Név: A futár neve.
 - Telefonszám: A futár telefonszáma.
- **A pékek egyed tulajdonságai:**
 - PékID: a pék elsődleges kulcsa.
 - Név: A pék neve.
 - Beosztás: Milyen időintervallumban dolgozik

Egyedek közötti kapcsolat:

- **Pékség és Termék:**

A pékség és termék egyedek között 1:N kapcsolat, mivel egy termék csak egy pékséghez tartozhat, de egy pékségnek lehet több terméke is.

- **Termék és Vevő:**

A termék és vevő egyedek között N:M kapcsolat van, mivel egy vevő rendelhet többfajta terméket is, és egy termékekből rendelhet több vevő is.

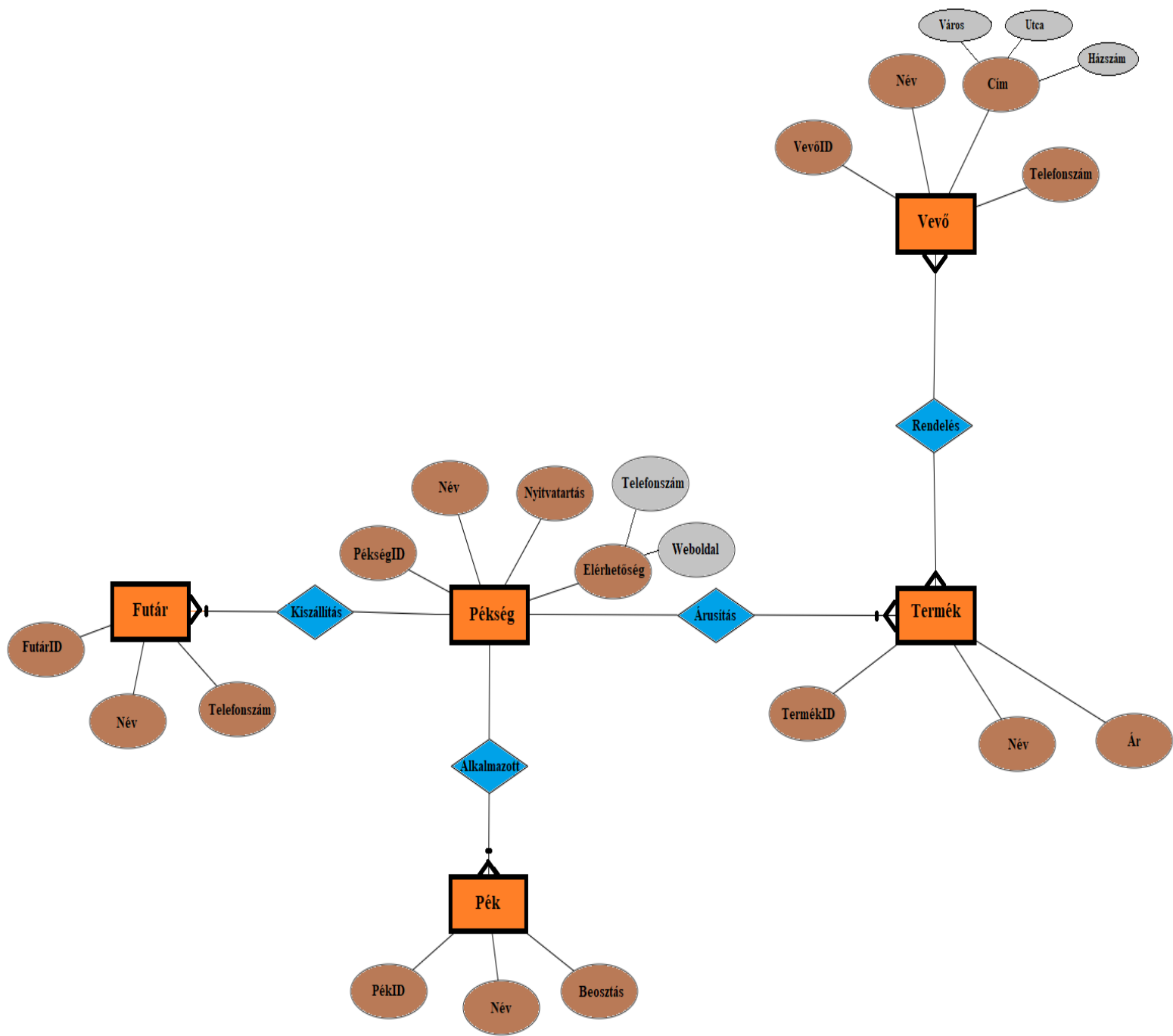
- **Pékség és Futár:**

A pékség és futár egyedek között 1:N kapcsolat van, mivel egy pékség alkalmazhat több futárt, de egy futár csak egy pékségnél dolgozhat.

- **Pékség és Pék:**

A Pék és a Pékség egyedek között N:M kapcsolat van, mivel egy pékségben dolgozhat több pék is, és egy pék is dolgozhat több pékségben, amennyiben a pékségek ugyan azon franchise-ba tartoznak.

Az adatbázis ER-modellje:



Konvertálás relációs modellé:

A **pékség** egyedből egy tábla lesz. A tulajdonságaiból és az *Elérhetőség* komponenseiből lesznek a mezők. A PékségID lesz az elsődleges kulcs.

A **Termék** egyedből egy tábla lesz. Az attribútumaiból lesznek a mezők, A TermékID lesz az elsődleges kulcsa, valamint lesz egy idegen kulcsa a PékségID, mely a **Pékség** tábla elsődleges kulcsához kapcsolódik.

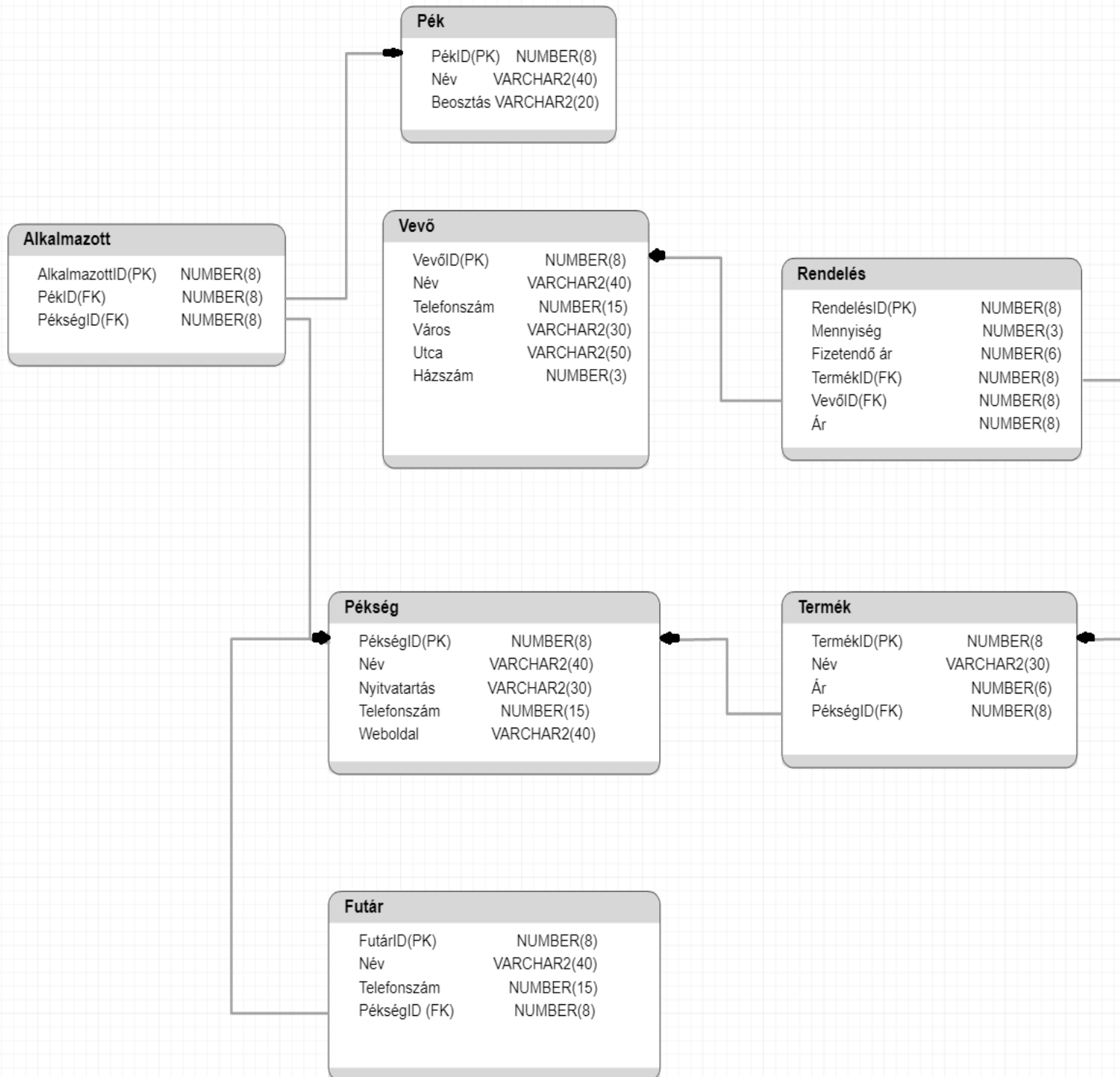
A **Vevő** egyedből egy tábla lesz. A tulajdonságaiból és a *Cím* attribútumaiból lesznek a mezők. A VevőID lesz az elsődleges kulcs.

A **Rendelés** kapcsolat egy külön tábla lesz, mivel N:M kapcsolatról van szó. A mezőit a kapcsolat tulajdonságai és idegen kulcsok alkotják. A RendelésID lesz az elsődleges kulcsa, valamint lesz két idegen kulcsa, a TermékID mely a **Termék** tábla elsődleges kulcsával áll kapcsolatban, illetve a VevőID, amely a **Vevő** tábla elsődleges kulcsával lesz kapcsolatban.

Az **alkalmazott** kapcsolat is egy külön tábla lesz, mivel itt is N:M kapcsolat létezik. Az idegen kulcsok alkotják. AlkalmazottID lesz az elsődleges kulcsa, valamint ennek is lesz két idegen kulcsa, a PékID, mely a **pék** tábla elsődleges kulcsával áll kapcsolatban, illetve a PékségID, mely a **pékség** tábla elsődleges kulcsával áll kapcsolatban.

A **Futár** egyedből egy tábla lesz. A tulajdonságaiból lesznek a mezők. A FutárID lesz az elsődleges kulcsa, illetve lesz egy idegen kulcsa a PékségID, mely a **Pékség** tábla elsődleges kulcsával lesz kapcsolatban.

Az adatbázis relációs modellje:



A táblák létrehozása:

```
CREATE TABLE Pekseg
```

```
(  
    PeksegID INT(8) NOT NULL,  
    Nev CHAR(40),  
    Nyitvatartas CHAR(30),  
    Telefonszam INT(15),  
    Weboldal CHAR(40),  
    PRIMARY KEY (PeksegID)  
);
```

```
CREATE TABLE Termek
```

```
(  
    TermekID INT(8) NOT NULL,  
    Nev CHAR(30),  
    Ar INT(6),  
    PRIMARY KEY (TermekID)  
    FOREIGN KEY (PeksegID) REFERENCES Pekseg(PeksegID)  
);
```

```
CREATE TABLE Vevo (
```

```
    VevoID INT(8) NOT NULL,  
    Nev CHAR(40),  
    Telefonszam INT(15),  
    Varos CHAR(30),  
    Utca CHAR(50),  
    Hazszam INT(3),  
    PRIMARY KEY (VevoID)  
);
```

```
CREATE TABLE Futar
```

```
(  
    FutarID INT(8) NOT NULL,  
    Nev CHAR(40),  
    Telefonszam INT(15),  
    PeksegID INT(8),  
    PRIMARY KEY (FutarID)  
    FOREIGN KEY (PeksegID) REFERENCES Pekseg(PeksegID)  
);
```

```
CREATE TABLE Rendeles
```

```
(  
    RendelesID INT(8) NOT NULL,  
    Mennyiseg INT(3),  
    Ar INT(6),  
    Fizetendo_ar AS (Mennyiseg * Ar),  
    PRIMARY KEY (RendelesID)  
    FOREIGN KEY (TermekID) REFERENCES Termek(TermekID)
```

```

        FOREIGN KEY (VevoID) REFERENCES Vevo(VevoID)
    );

CREATE TABLE Pek
(
    PekID INT(8) NOT NULL,
    Nev CHAR(40),
    Beosztas CHAR(20),
    PRIMARY KEY (PekID)
);

CREATE TABLE Alkalmazott
(
    AlkalmazottID INT(8)
    PRIMARY KEY (AlkalmazottID)
    FOREIGN KEY (PekID) REFERENCES Pek(PekID)
    FOREIGN KEY (PeksegID) REFERENCES Pekseg(PeksegID)
);

```

A táblák feltöltése adatokkal

```

BEGIN

INSERT INTO Pekseg (PeksegID,Nev,Nyitvatartas,Telefonszam,Weboldal) VALUES
(1,Jacques liszt pékség','10:00-18:00',365145013,'Jacquespekseg.hu');
INSERT INTO Pekseg (PeksegID,Nev,Nyitvatartas,Telefonszam,Weboldal) VALUES
(2, 'Pékműhely és szerelmeslevél','11:00-19:00',365143333,'pekmuhely.hu');
INSERT INTO Pekseg (PeksegID,Nev,Nyitvatartas,Telefonszam,Weboldal) VALUES
(3, 'Nagyanyó pékség','9:00-19:00',3635576843,'nagyanyoka.hu');
INSERT INTO Pekseg (PeksegID,Nev,Nyitvatartas,Telefonszam,Weboldal) VALUES
(4,'Numann Pékség','10:00-19:00',3636517666,'Numann.hu');
INSERT INTO Pekseg (PeksegID,Nev,Nyitvatartas,Telefonszam,Weboldal) VALUES
(5,'RádiPékség','12:00-19:00',3636416598,'radipekseg.hu');

```

END;

BEGIN

```

INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(1,"Croissant",1500,1);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(2,"Sajtoskifli",137,1);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(3,'Kakaóscsiga',250,1);

```



```

INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(4,'Croissant',788,2);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(5,'Sajtoskifli',300,2);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(6,'Kakaóscsiga',150,2);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(7,'Croissant',210,3);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(8,'Sajtoskifli',333,3);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(9, 'Kakaóscsiga',250,3);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(10,'Croissant',274,4);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(11,'Sajtoskifli',800,4);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(12,'Kakaóscsiga',440,4);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(13,'Croissant',700,5);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(14,'Sajtoskifli',300,5);
INSERT INTO Termek (TermekID,Nev,Ar,PeksegID) VALUES
(15,'Kakaóscsiga',310,5);

```

BEGIN

```

INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(1,'Végh Béla,36305145033,3300,'Eger','Egri csillagok',33);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(2,'Nagy József,3625556666,3300,'Eger','Mátyás', 48);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(3,'Kovács Kristóf',36703534434,3300,'Eger','Rákóczi',203);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(4,'Rácz Kincső',36301234567,3399,'Miskolc','Egri',15);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(5,'Fodrász Kristóf',36207777776,3397,'Füzesabony','Fő',53);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(6,'Orr Tamás',36453463344,3300,'Füzesabony','Szécheny',71);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(7,'Vessző Hajnal',36302554433,3300,'Eger','Hadnagy',2);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(8,'Lakatos Kristóf',36309755467,3300,'Eger','Merengő',3);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(9,'Kis Dominik',36307474745,3300,'Eger','Szécheny',55);
INSERT INTO Vevo (VevoID,Nev,Telefonszam,Varos,Utca,Hazszam) VALUES
(10, 'Molnár Győző',36309981104,3300,'Eger','Rákóczi',112);

```

END;

BEGIN

```
INSERT INTO Futar (FutarID,Nev,Telefonszam, PeksegID) VALUES
(1,'Rácz Csaba',36304626123,1);
INSERT INTO Futar (FutarID,Nev,Telefonszam, PeksegID) VALUES
(2,'Nagy Gigi',36208951453,2);
INSERT INTO Futar (FutarID,Nev,Telefonszam, PeksegID) VALUES
(3,'Kovács József',36705345832,3);
INSERT INTO Futar (FutarID,Nev,Telefonszam, PeksegID) VALUES
(4,'Kocsis Péter',36302367474,4);
INSERT INTO Futar (FutarID,Nev,Telefonszam, PeksegID) VALUES
(5,'Kovács András Kristóf',36302135656,5);
```

END;

BEGIN

```
INSERT INTO Pek (PekID,Nev,Beosztas) VALUES
(1,'Fehér József','8:00-17:00');
INSERT INTO Pek (PekID,Nev,Beosztas) VALUES
(2,'Nagy Csaba','9:00-17:00');
INSERT INTO Pek (PekID,Nev,Beosztas) VALUES
(3,'Kovács József','8:00-16:00');
INSERT INTO Pek (PekID,Nev,Beosztas) VALUES
(4,'Háborús János','3:00-22:00');
INSERT INTO Pek (PekID,Nev,Beosztas) VALUES
(5,'Farkas Bence','12:00-14:00');
```

END;

BEGIN

```
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(1,4,300,5,1);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(2,3,700,13,2);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(3,1,300,14,3);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(4,5,250,3,4);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(5,1,150,6,5);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(6,1,440,12,6);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(7,4,333,8,7);
INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
```

(8,5,250,9,8);

INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(9,3,150,6,9);

INSERT INTO Rendeles (RendelesID,Mennyiseg,Ar,TermekID,VevoID) VALUES
(10,6,800,11,10);

BEGIN

INSERT INTO Alkalmazott (AlkalmazottID, PekID, PeksegID) VALUES
(1,2,3);

INSERT INTO Alkalmazott (AlkalmazottID, PekID, PeksegID) VALUES
(2,3,4);

INSERT INTO Alkalmazott (AlkalmazottID, PekID, PeksegID) VALUES
(3,4,5);

INSERT INTO Alkalmazott (AlkalmazottID, PekID, PeksegID) VALUES
(4 5,1);

INSERT INTO Alkalmazott (AlkalmazottID, PekID, PeksegID) VALUES
(5,1,2);

END;

MÓDOSÍTÁSOK

1. Annak az embernek aki a Szécheny utcában lakik, beállítja a város-t Egernek

UPDATE vevo
SET Varos="Eger"
WHERE Utca= "Szécheny"

2. A Nagy Gigi Nevű futár telefonszámát átírja az alábbi telefonszámra

UPDATE futar
SET Telefonszam="36405145"
WHERE Nev="Nagy Gigi"

3. A 3.pék beosztását megváltoztatjuk éjszakai beosztásra

UPDATE pek
SET Beosztas ="17:00-23:00"
WHERE PekID=3

4. A József nevű pékeknek névnapjuk van, így nem kell bejönniük dolgozni, Beosztástalatt felmentettre frissítjük

UPDATE pek
SET Beosztas ="Felmentett"
WHERE Nev LIKE ="%József%"

5. Az összes Croissantot elvitték, így helyettesítenünk kell Sonkás táskával.

```
UPDATE termék  
SET Nev="Sonkás Táska"  
WHERE Nev="Croissant"
```

LEKÉRDEZÉSEK

1.Lekérdezzük az Egri vásárlók nevét.

```
SELECT Nev  
FROM Vevo  
WHERE Varos="Eger";
```

$\pi_{\text{nev}}(\sigma_{\text{Varos}=\text{"Eger"}}(\text{VEVO}))$

2. Lekérdezzük a Kristóf nevű vásárlókat.

```
SELECT Nev  
FROM Vevo  
WHERE Nev LIKE "%Kristóf%";
```

$\pi_{\text{Nev}}(\sigma_{\text{Nev LIKE \%Kristóf\%}}(\text{VEVO}))$

3. Lekérjük Mennyibe került a legkisebb vásárlás.

```
SELECT MIN(Fizetendo_ar) AS "legkisebb vásárlás"  
FROM Rendeles;
```

$\pi_{\text{MIN(Fizetendo_ar)}}(\text{RENDELES})$

4. Lekérdezzük milyen pékárukat lehet vásárolni a Numann Pékségben.

```
SELECT Termek.Nev  
FROM Termek  
JOIN Pekseg ON Pekseg.PeksegID=Termek.PeksegID  
WHERE Pekseg.Nev= "Numann Pékség";
```

$\pi_{\text{Termek.nev}}(\sigma_{\text{Pekseg.nev="Numann Pékség"}}(\text{TERMEK}) \bowtie \text{Pekseg.PeksegID=Termek.PeksegID}(\text{PEKSEG}))$

5. Lekérdezzük a legkevesebbet vett süteményt.

```
SELECT COUNT(rendeles.TermekID) AS "db", Termek.Nev
FROM Rendeles
JOIN Termek ON Termek.TermekID = Rendeles.RendelesID
GROUP BY Termek.Nev
ORDER BY 1 ASC
Limit 1;
```

$\pi_{\text{count}(\text{rendeles.termekID}), \text{Termek Nev}}(\text{RENDELES}) \bowtie \text{Termek.TermekID}=\text{Rendeles.RendelesID}(\text{TERMEK})$

6. Lekérdezzük a Rákóczi utcában élő vevők nevét és a rendeléseit.

```
SELECT Termek.Nev, Vevo.Nev, Rendeles.RendelesID
FROM Termek
JOIN Rendeles ON Rendeles.TermekID = Termek.TermekID
JOIN Vevo ON Rendeles.VevoID = Vevo.VevoID
WHERE Vevo.Utca="Rákóczi";
```

$\pi_{\text{termek.nev, vevo.nev, rendeles.rendelesID}}(\sigma_{\text{Vevo.Utca}=\text{"Rákóczi"}}(\text{TERMEK}) \bowtie \text{rendeles.TermekID}=\text{Termek.TermekID}(\text{RENDELES}) \bowtie \text{Rendeles.vevoID}=\text{Vevo.VevoID}(\text{VEVO}))$

7. Lekérdezzük azokat a vevőket akik az átlagnál kevesebbet költöttek

```
SELECT Vevo.Nev
From Vevo
Join Rendeles ON Vevo.VevoID=Rendeles.VevoID
WHERE (Rendeles.Fizetendo_ar <=(SELECT AVG (Rendeles.Fizetendo_ar) FROM
Rendeles));
```

$\pi_{\text{Vevo.Nev}}(\sigma_{(\text{Rendeles.Fizetendo_ar} \leq (\text{SELECT AVG (Rendeles.Fizetendo_ar) FROM Rendeles}))}(\text{VEVO}) \bowtie \text{Vevo.VevoID}=\text{Rendeles.VevoID}(\text{RENDELES}))$

8. Ha valaki 1000 FT felett vásárol, kap 30% os kedvezményt. Lekérdezzük, hogy az adott rendelés jogosult-e a kedvezményre.

```
SELECT Vevo.Nev, Rendeles.RendelesID, CASE WHEN Rendeles.Fizetendo_ar>1000
THEN "Igen" ELSE "Nem" END AS "Jogosult-e"
FROM Rendeles
JOIN Vevo ON Vevo.VevoID=Rendeles.VevoID;
```

$\pi_{\text{Vevo.Nev, Rendeles.RendelesID, CASE WHEN Rendeles.Fizetendo_ar} > 1000 \text{ THEN "Igen" ELSE "Nem" END AS "Jogosult-e"}}(\text{RENDELES}) \bowtie \text{Vevo.VevoID}=\text{Rendeles.VevoID}(\text{VEVO})$

9. Ha igen mekkora összegre jogosult?

```
SELECT Vevo.Nev, Rendeles.RendelesID,  
CASE WHEN Rendeles.Fizetendo_ar>1000 THEN "Igen" ELSE "Nem" END AS "Jogosult-  
e",  
CASE WHEN Rendeles.Fizetendo_ar>1000 THEN Fizetendo_ar*0.1 ELSE 0 END AS  
"Kedvezmeny"  
FROM Rendeles  
JOIN Vevo ON Vevo.VevoID=Rendeles.VevoID
```

π Vevo.Nev, Rendeles.RendelesID, CASE WHEN Rendeles.Fizetendo_ar>1000 THEN 'Igen' ELSE 'Nem' END AS „Jogosult-e”, CASE
WHEN Rendeles.Fizetendo_ar>1000 THEN Fizetendo_ar*0.1 ELSE 0 END AS „Kedvezmeny” (RENDELES) ⋈
Vevo.VevoID=Rendeles.VevoID (VEVO)

10. A Rádi pékség beruházni szeretne új pékséggel bővíteni a franchise láncolatát. Ennek az összege 100Millió forint lesz. A kakaós csigából érkező bevételt 100%-ban erre fogják felhasználni. Hány Kakaós Csigát kell eladni erre beruházni?

```
SELECT 100000000/(  
SELECT Ar  
FROM Termek  
JOIN Pekseg ON Pekseg.PeksegID = Termek.PeksegID  
WHERE termek.Nev="Kakaóscsiga" AND Pekseg.nev="Rádi Pékség"  
) AS "Eladandó Kakaós Csig Darab"
```

$\pi_{100000000/\pi_{ar}}(\sigma_{termek.Nev='Kakaóscsiga' \text{ AND } Pekseg.nev='Rádi Pékség'}$
) AS „Eladandó Kakaós Csig Darab”(TERMEK) ⋈ Pekseg.PeksegID=Termek.PeksegID (PEKSEG))