

Jegyzőkönyv

Adatbázisrendszerek I.

Féléves feladat: Futárszolgálat

Készítette: **Balázs Tamás**
Neptunkód: **HM23GB**

A feladat leírása:

A beadandó témája egy olyan adatbázis ami egy futárszolgálat/csomagszállító cég adatait kezeli. A rendszer leírja a vevő adatait, az általa megrendelt termékeket és azt is hogy honnan rendelt a vevő.

Az ER modell egyedei és tulajdonságai:

- **A rendelés egyed tulajdonságai**
 - RendelésID: A rendelés egyed azonosítja, elsődleges kulcs.
 - Tétel: Többszörös tulajdonság. A rendelésben szereplő tétel.
 - Teljes ár: A rendelt tétel teljes ára. Származtatott tulajdonság.
 - Dátum: A rendelés dátuma
- **A vevő egyed tulajdonságai**
 - VevőID: A Vevő egyed elsődleges kulcsa.
 - Név: A vevő neve.
 - Telefonszám: A vevő telefonszáma.
 - Cím: Összetett tulajdonság. A vevő címe (irányítószám, város, utca és házszám).
 - E-mail cím: A vevő e-mail címe.
- **A bankkártya egyed tulajdonságai**
 - BankkártyaID: A Bankkártya egyed elsődleges kulcsa, a bankkártya száma.
 - Tulajdonos: A bankkártya tulajdonos neve.
 - Lejárat dátum: A kártya lejárat dátuma.
- **A futár egyed tulajdonságai**
 - FutárID: A Futár egyed azonosítója. Elsődleges kulcs.
 - Futár név: A futár neve.
 - Telefonszám: A futár telefonszáma.
- **Az eladó/üzlet egyed tulajdonságai**
 - EladóID: Az eladó egyed elsődleges kulcsa.
 - Üzletnév: Az üzlet vagy eladó neve akitől rendeltek.
 - Telefonszám: Az eladó telefonszáma.

Kapcsolatok:

- **Rendelés és Futár:**

Ez egy 1:N típusú kapcsolat hiszen egy rendelést csak 1 futár szállíthat ki viszont egy futárnak lehet több rendelése is amit teljesítenie kell.
- **Rendelés és Vevő:**

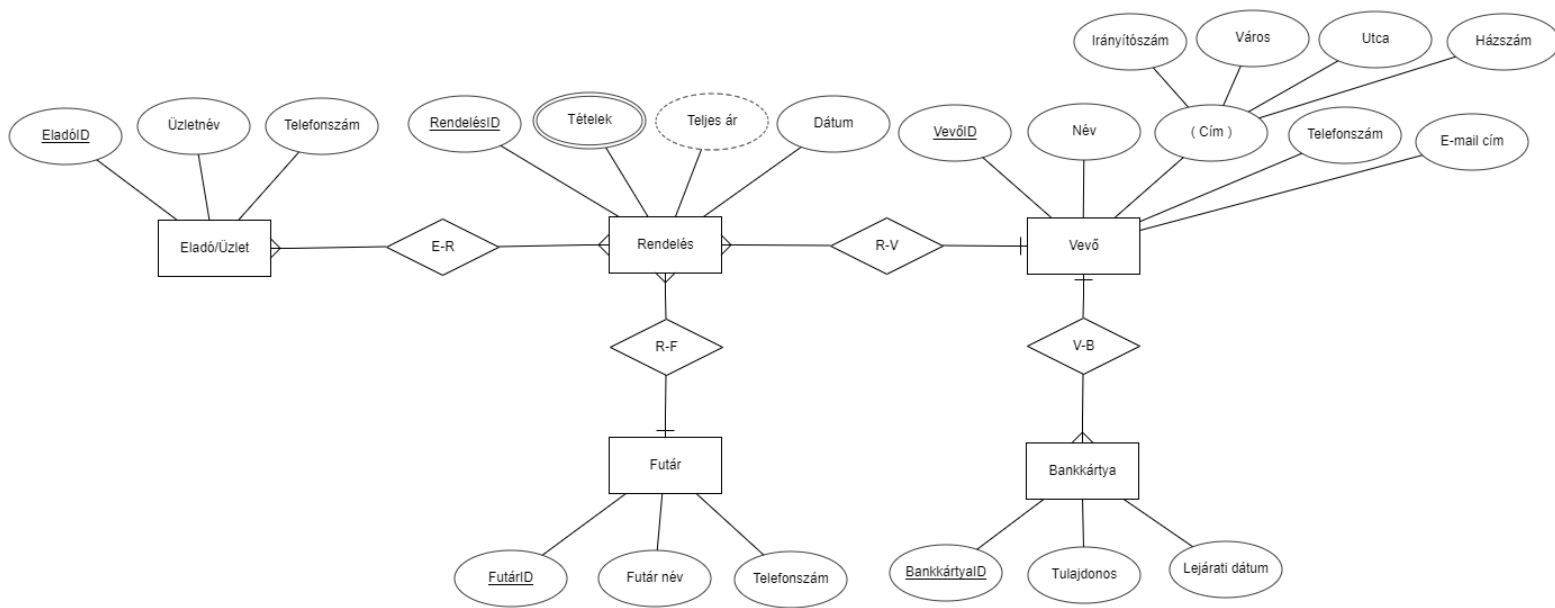
A két egyed között a kapcsolatot a kiszállítás teremti meg. Egy vevőnek lehet több megrendelése, míg egy megrendelés csak egy vevőhöz kapcsolódhat, így a kapcsolat 1:N típusú.

- **Vevő és Bankkártya:**

Legjobban a birtoklás írja le a kapcsolatot, melynél ugyancsak 1:N kapcsolatot realizálhatunk, hiszen ugyanaz a kártya nem tarthat több emberhez, de egy embernek lehet több bankkártyája is. Ezt az életben látott példák is igazolják.

- **Eladó/üzlet és Rendelés:**

Itt több-több kapcsolatot láthatunk, mivel egy eladótól lehet több rendelést is leadni, valamint egy rendelésen belül is lehet több eladótól/üzletből rendelni.



Az adatbázis konvertálása relációs modellre:

A **Rendelés** egyedből egy tábla lesz mely elsődleges kulcsa a RendelésID lesz. Mezői a teljes ár és a dátum. A VevőID egy idegen kulcs mely a **Vevő** tábla elsődleges kulcsához kapcsolódik. A FutárID szintúgy idegen kulcs mely a **Futár** tábla elsődleges kulcsához kapcsolódik.

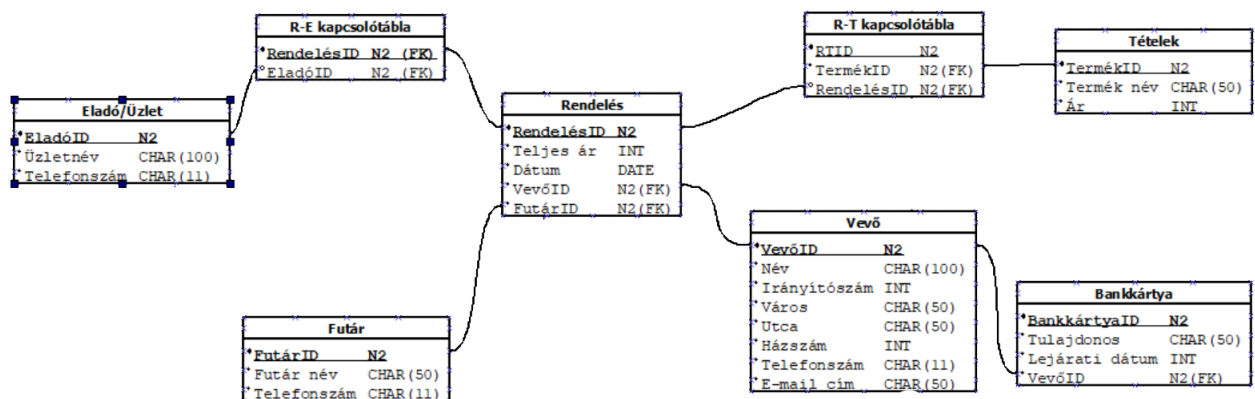
Egy rendeléshez több tétel tartozik alapesetben. Ehhez is szükséges egy kapcsolótábla mivel ugyanaz a termék megnevezés több rendeléshez is kapcsolódhat.

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

A **Bankkártya** is egy táblát fog jelenteni, ahol a kártya száma lesz az ID és a többi tulajdonsága is mezőket jelent majd. A VevőID egy idegen kulcs ami a **Vevő** tábla elsődleges kulcsához kapcsolódik.

A **Futár** egyed is egy tábla lesz. A FutárID az elsődleges kulcs, a tulajdonságaiból lesznek a mezők.

Az E-R kapcsolatból a **Eladó/üzlet** és a **Futár** között egy kapcsolótábla lesz, hiszen a N:M típusú. Ebben a az esetben a EladóID az üzletet, míg a RendelésID a rendelést azonosítja.



Az adatbázis relációs sémái:

Rendelés(**RendelésID**, Teljes ár, Dátum, VevőID, EladóID, FutárID)

Futár(**FutárID**, Futár név, Telefonszám)

Vevő(**VevőID**, Név, Irányítószám, Város, Utca, Házszám, Telefonszám, E-mail cím)

Bankkártya(**BankkártyaID**, Tulajdonos, Lejárat dátum, VevőID)

R-T kapcsolótábla(**RTID**, TermékID, RendelésID)

Tételek(**TermékID**, Termék név, Ár)

R-E kapcsolótábla(**EladóID**, RendelésID)

Eladó/üzlet(**EladóID**, Üzletnév, Telefonszám)

Táblák Létrehozása:

Elsőként az idegen kulcs nélküli táblákat érdemes létrehozni, ezt követően kell azokat, melyekben idegenkulcs található. Az idegenkulcsot tartalmazó mezők típusának meg kell egyeznie a másik táblában lévő mező típusával, amire mutat.

```
CREATE TABLE vevo(  
  VevoID INT PRIMARY KEY NOT NULL,  
  Nev VARCHAR(50),  
  Iranyitoszam INT,  
  Varos VARCHAR(50),  
  Utca VARCHAR(50),  
  Hazszam INT,  
  Telefonszam VARCHAR(11),  
  Emailcim VARCHAR(50)  
);
```

```
CREATE TABLE futar(  
  FutarID INT PRIMARY KEY NOT NULL,  
  Futarnev VARCHAR(50),  
  Telefonszam INT  
);
```

```
CREATE TABLE elado(  
  EladoID INT PRIMARY KEY NOT NULL,  
  Eladonev VARCHAR(50),  
  Telefonszam INT  
);
```

```
CREATE TABLE tetelek(  
  TermekID INT PRIMARY KEY NOT NULL,  
  Termeknev VARCHAR(50),  
  Ar INT  
);
```

```
CREATE TABLE bankkartya(  
BankkartyaID INT PRIMARY KEY NOT NULL,  
Tulajdonos VARCHAR(50),  
Lejarati_datum INT,  
VevoID INT,  
FOREIGN KEY (VevoID) REFERENCES vevo(VevoID)  
);  
  
CREATE TABLE rendeles(  
RendelesID INT PRIMARY KEY NOT NULL,  
Teljesar INT,  
Datum DATE,  
VevoID INT,  
FutarID INT,  
FOREIGN KEY (VevoID) REFERENCES vevo(VevoID),  
FOREIGN KEY (FutarID) REFERENCES futar(FutarID)  
);  
  
CREATE TABLE rt_kapcsolat(  
RTID INT PRIMARY KEY NOT NULL,  
TermekID INT,  
RendelesID INT,  
FOREIGN KEY (TermekID) REFERENCES tetelek(TermekID),  
FOREIGN KEY (RendelesID) REFERENCES rendeles(RendelesID)  
);  
  
CREATE TABLE re_kapcsolat(  
RendelesID INT PRIMARY KEY NOT NULL,  
EladoID INT,  
FOREIGN KEY (EladoID) REFERENCES elado(EladoID)  
);
```

A táblák feltöltése:

vevo tábla:

INSERT INTO vevo VALUES (1, 'Kiss Zoltán', '3500', 'Miskolc', 'Zoltán utca', 16, '06201234567', 'zoli@gmail.com');

INSERT INTO vevo VALUES (2, 'Tóth Lajos', '3500', 'Miskolc', 'Lajos utca', 35, '06307654321', 'lajos@citromail.com');

INSERT INTO vevo VALUES (3, 'Nagy Endre', '3500', 'Miskolc', 'Endre utca', 23, '06701234567', 'enre@gmail.com');

INSERT INTO vevo VALUES (4, 'Olasz Renátó', '3500', 'Miskolc', 'Halász utca', 3, '06207172635', 'renato@gmail.com');

INSERT INTO vevo VALUES (5, 'Lakatos Tamás', '3500', 'Miskolc', 'Mogyoródi utca', 10, '06704443335', 'tomi@gmail.com');

INSERT INTO vevo VALUES (6, 'Szabó Bence', '3500', 'Miskolc', 'Petőfi utca', 67, '06309876543', 'bence@gmail.com');

futar tábla:

INSERT INTO futar VALUES(1, 'Kiss Rebeka', "06709182736");

INSERT INTO futar VALUES(2, 'Nagy József', "06207465329");

INSERT INTO futar VALUES(3, 'Alföldi Béla', "06301234567");

INSERT INTO futar VALUES(4, 'Veres János', "06204444444");

INSERT INTO futar VALUES(5, 'Tóth Lajos', "06206665358");

INSERT INTO futar VALUES(6, 'Kiss Lóránt', "06709991113");

elado tábla:

INSERT INTO elado VALUES(1,'H&M', '06208753499');

INSERT INTO elado VALUES(2,'Tesco', '06701298349');

INSERT INTO elado VALUES(3,'Emag', '06208889123');

INSERT INTO elado VALUES(4,'Mediamarkt', '06701853533');

INSERT INTO elado VALUES(5,'Mobilfox', '06206378111');

INSERT INTO elado VALUES(6,'Footshop', '06308888812');

INSERT INTO elado VALUES(7,'Euronics', '06330225554');

INSERT INTO elado VALUES(8,'Konzolvilág', '06309124665');

INSERT INTO elado VALUES(9,'Notino', '06700924003');

INSERT INTO elado VALUES(10,'Nike', '06209770047');

tetelek tábla:

INSERT INTO tetelek VALUES(1, 'farmer', 7000);

INSERT INTO tetelek VALUES(2, 'melegítő', 15000);

INSERT INTO tetelek VALUES(3, 'kontroller', 20000);

INSERT INTO tetelek VALUES(4, 'univerzális távirányító', 5000);

INSERT INTO tetelek VALUES(5, 'telefonok', 5000);

INSERT INTO tetelek VALUES(6, 'cipő', 30000);

INSERT INTO tetelek VALUES(7, 'játék szoftver', 21000);

INSERT INTO tetelek VALUES(8, 'parfüm', 30000);

INSERT INTO tetelek VALUES(9, 'cipő', 28000);

INSERT INTO tetelek VALUES(10, 'póló', 4000);

INSERT INTO tetelek VALUES(11, 'tv', 80000);

INSERT INTO tetelek VALUES(12, 'játékkonzol', 150000);

INSERT INTO tetelek VALUES(13, 'pulcsi', 6000);

INSERT INTO tetelek VALUES(14, 'monitor', 35000);

INSERT INTO tetelek VALUES(15, 'kabát', 27000);

bankkartya tábla:

INSERT INTO bankkartya VALUES(1,'Kiss Zoltán',0123,1);

INSERT INTO bankkartya VALUES(2,'Tóth Lajos',1024,2);

INSERT INTO bankkartya VALUES(3,'Nagy Endre',0426,3);

INSERT INTO bankkartya VALUES(4,'Nagy Jánosné',1222,3);

INSERT INTO bankkartya VALUES(5,'Olasz Renátó',0525,4);

INSERT INTO bankkartya VALUES(6,'Lakatos Tamás',0323 ,5);

INSERT INTO bankkartya VALUES(7,'Szabó Bence',0724,6);

rendeles tábla:

```
INSERT INTO rendeles VALUES(1,80000, '2021-11-24',4,2);
INSERT INTO rendeles VALUES(2,150000, '2021-11-26',3,6);
INSERT INTO rendeles VALUES(3,13000, '2021-11-28',1,5);
INSERT INTO rendeles VALUES(4,30000, '2021-11-29',2,1);
INSERT INTO rendeles VALUES(5,45000,'2021-11-27',6,3);
INSERT INTO rendeles VALUES(6,30000, '2021-11-29',5,4);
INSERT INTO rendeles VALUES(7,5000,'2021-11-28',1,1);
INSERT INTO rendeles VALUES(8,171000,'2021-11-27',4,6);
INSERT INTO rendeles VALUES(9,27000,'2021-11-29',6,2);
INSERT INTO rendeles VALUES(10,4000,'2021-11-25',3,3);
```

rt_kapcsolo tábla:

```
INSERT INTO rt_kapcsolo VALUES(1,11,1);
INSERT INTO rt_kapcsolo VALUES(2,12,2);
INSERT INTO rt_kapcsolo VALUES(3,1,3);
INSERT INTO rt_kapcsolo VALUES(4,13,3);
INSERT INTO rt_kapcsolo VALUES(5,8,4);
INSERT INTO rt_kapcsolo VALUES(6,2,5);
INSERT INTO rt_kapcsolo VALUES(7,6,5);
INSERT INTO rt_kapcsolo VALUES(8,6,6);
INSERT INTO rt_kapcsolo VALUES(9,5,7);
INSERT INTO rt_kapcsolo VALUES(10,12,8);
INSERT INTO rt_kapcsolo VALUES(11,7,8);
INSERT INTO rt_kapcsolo VALUES(12,15,9);
INSERT INTO rt_kapcsolo VALUES(13,10,10);
INSERT INTO rt_kapcsolo VALUES(14,1,11);
INSERT INTO rt_kapcsolo VALUES(15,2,11);
INSERT INTO rt_kapcsolo VALUES(16,6,11);
INSERT INTO rt_kapcsolo VALUES(17,10,11);
```

INSERT INTO rt_kapcsolo VALUES(18,13,11);

INSERT INTO rt_kapcsolo VALUES(19,4,12);

INSERT INTO rt_kapcsolo VALUES(20,14,13);

INSERT INTO rt_kapcsolo VALUES(21,13,13);

re_kapcsolo tábla:

INSERT INTO re_kapcsolo VALUES(1,7);

INSERT INTO re_kapcsolo VALUES(2,4);

INSERT INTO re_kapcsolo VALUES(3,1);

INSERT INTO re_kapcsolo VALUES(4,9);

INSERT INTO re_kapcsolo VALUES(5,2);

INSERT INTO re_kapcsolo VALUES(6,6);

INSERT INTO re_kapcsolo VALUES(7,5);

INSERT INTO re_kapcsolo VALUES(8,8);

INSERT INTO re_kapcsolo VALUES(9,10);

INSERT INTO re_kapcsolo VALUES(10,2);

INSERT INTO re_kapcsolo VALUES(11,1);

INSERT INTO re_kapcsolo VALUES(12,3);

INSERT INTO re_kapcsolo VALUES(13,8);

Lekérdezések:

1. Listázzuk ki azokat a rendeléseket melyeket 2021.11.29-én adtak le.

SELECT * FROM rendeles WHERE Datum = '2021-11-29';

RendelesID	Teljesar	Datum	VevoID	FutarID
4	30000	2021-11-29	2	1
6	30000	2021-11-29	5	4
9	27000	2021-11-29	6	2

$\Pi_{\text{RendelesID, Teljesar, Datum, VevoID, FutarID}} [\sigma_{\text{Datum} = '2021-11-29'}(\text{rendeles})]$

2. Listázzuk azokat a tételeket melyek 25000 forintnál többbe kerülnek.

`SELECT TermekID, Termeknev, Ar FROM tetelek WHERE Ar > 25000;`

TermekID	Termeknev	Ar
6	cipő	30000
8	parfüm	30000
9	cipő	28000
11	tv	80000
12	játékkonzol	150000
14	monitor	35000
15	kabát	27000

$\Pi_{\text{TermekID, Termeknev, Ar}} [\sigma_{\text{Ar} > 25000}(\text{termekek})]$

3. Számolja meg hány rendelés került az átlagnál többbe.

`SELECT count(RendelesID), AVG(Teljesar) FROM rendeles WHERE Teljesar > (SELECT avg(Teljesar) FROM rendeles);`

count(RendelesID)	AVG(Teljesar)
5	103600.0000

$\Pi_{\text{count(RendelesID), AVG(Teljesar)}} [\sigma_{\text{Teljesar} > (\pi \text{ AVG(Teljesar)}(\text{rendeles}))}(\text{rendeles})]$

4. A tételek árak szerint növekvő sorrendbe rakása.

`SELECT Termeknev, Ar FROM tetelek ORDER BY Ar;`

Termeknev	Ar
farmer	7000
melegítő együttes	15000
kontroller	20000
játék szoftver	21000
kabát	27000
cipő	28000
cipő	30000
parfüm	30000
monitor	35000
tv	80000
játékkonzol	150000

$\Pi_{\text{Termeknev, Ar}(\text{tetelek})}$

5. Listázzuk ki a vevők telefonszámát és e-mail címét.

`SELECT Nev,Telefonszam,Emailcim FROM vevo;`

	Nev	Telefonszám	Emailcim
►	Kiss Zoltán	06201234567	zoli@gmail.com
	Tóth Lajos	06307654321	lajos@citromail.com
	Nagy Endre	06701234567	enre@gmail.com
	Olasz Renátó	06207172635	renato@gmail.com
	Lakatos Tamás	06704443335	tomi@gmail.com
	Szabó Bence	06309876543	bence@gmail.com

$\Pi_{\text{Nev,Telefonszam,Emailcim}} \text{VEVO}$

6. Listázza ki, hogy a vevőknek hány kártyája van regisztrálva.

`SELECT Nev,count(BankkartyaID) FROM bankkartya INNER JOIN
vevo ON bankkartya.VevoID = vevo.VevoID GROUP BY nev;`

Nev	count(BankkartyaID)
Kiss Zoltán	1
Tóth Lajos	1
Nagy Endre	2
Olasz Renátó	1
Lakatos Tamás	1
Szabó Bence	1

$\Pi_{\text{Nev,count(BankkartyaID)}} \Gamma_{\text{count(BankkartyaID),Nev}} (\text{Bankkartya} \bowtie_{\text{Bankkartya.VevoID=vevo.VevoID}} \text{vevo})$

7. Listázza ki mikor rendeltek farmert.

`SELECT TermekID,Datum FROM rendeles INNER JOIN rt_kapcsoló ON rendeles.RendelesID = rt_kapcsoló.RendelesID WHERE TermekID='1';`

TermekID	Datum
1	2021-11-28
1	2021-11-25

$\Pi_{\text{TermekID,Datum}} [\sigma_{\text{TermekID}='1'}(\text{rendeles} \bowtie_{\text{RENDELES.RendelesID} = \text{rt_kapcsoló.RendelesID}} \text{rt_kapcsoló})]$

8. Kiestázza a vevő kártyájának kártyaszámát és a lejáratí dátumát, ha a vevő a 10-es házszámán lakí.

`SELECT BankkartyaID, Lejartat_datum FROM bankkartya INNER JOIN vevo ON bankkartya.VevoID=vevo.VevoID WHERE Hazszám=10;`

BankkartyaID	Lejartat_datum
6	323

$\Pi_{\text{Bankkartya, Lejartat_datum}} [\sigma_{\text{Hazszám}=10}(\text{bankkartya} \bowtie_{\text{bankkartya.VevoID} = \text{vevo.VevoID}} \text{vevo})]$

9. Kiestázza bármely 30000-nél drágább rendelés azonosítóját.

`SELECT RendelesID FROM rendeles WHERE Teljesar > ANY (SELECT Teljesar FROM rendeles WHERE Teljesar=30000);`

RendelesID
1
2
5
8
11
13

$\Pi_{\text{RendelesID}} \sigma_{\text{Teljesar} > (\Pi_{\text{Teljesar}} \sigma_{\text{Teljesar}=30} \text{rendeles})} \text{rendeles}$

10. Kilstázza azokat a vevőket akiknek a házszaa nagyobb mint 20.

SELECT VevoID,Nev,Iranyitoszam,Varos,Utca,Telefonszam,Emailcim
FROM vevo **WHERE** hazszam>20;

VevoID	Nev	Iranyitoszam	Varos	Utca	Telefonszam	Emailcim
2	Tóth Lajos	3500	Miskolc	Lajos utca	06307654321	lajos@citromail.com
3	Nagy Endre	3500	Miskolc	Endre utca	06701234567	enre@gmail.com
6	Szabó Bence	3500	Miskolc	Petőfi utca	06309876543	bence@gmail.com

$\Pi_{\text{VevoID,Nev,Iranyitoszam,Varos,Utca,Telefonszam,Emailcim}} [\sigma_{\text{hazszam} > 20}(\text{vevo})]$