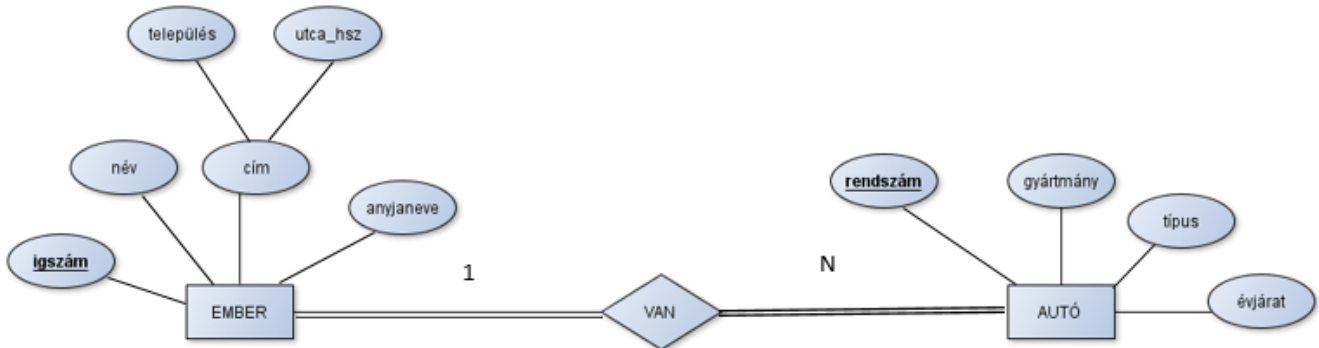


## GYAKORLÓ FELADATOK

### 1. Készítsük el az alábbi feladat EK-modelljét!

Emberek és autók adatait szeretnénk tárolni. Egy embernek több autója is lehet. A feladatban szereplő autók mindegyikének van tulajdonosa, valamint minden tárolt személynek van legalább egy autója. Az embereknek tároljuk az igazolvány számát, a nevét, a címét, anyja nevét. Az autóknak tároljuk a rendszámát, színét, gyártmányát, típusát, évjáratát.



### 2. EK-modell alapján készítsük el a relációs modellt!

Leképezési szabály: 1-N kapcsolat, teljes részvételű N oldal

*Mindkét egyed típusnak külön relációt hozunk létre, melyek kulcsai a kulcsattribútumok lesznek. Az N-oldali relációt kiegészítjük az 1-oldal elsődleges kulcsával, mint idegen kulccsal. Ha a kapcsolatnak vannak attribútumai, azok az N-oldali relációba kerülnek.*

EMBER (igszám, név, település, utca, anyjaneve)

AUTÓ (rendszám, gyártmány, típus, évjárat, igszám)

### 3. Hozzuk létre a táblákat, és töltsük fel adatokkal!

a) Táblák létrehozása:

```
CREATE TABLE ember(
  igszam varchar(8) primary key,
  nev varchar(40) not null,
  telepules varchar(50),
  utca varchar(30),
  anyjaneve varchar(40));
```

```
CREATE TABLE auto(
  rendszam varchar(6) primary key,
  gyartmany varchar(30) not null,
  tipus varchar(30),
  evjarat year,
  igszam varchar(8),
  foreign key(igszam) references ember(igszam));
```

b) Adatfeltöltés:

```
INSERT INTO ember VALUES('AA111222','Kis Ede','Paks','Kossuth u.10.','Nagy Emma');
```

c) Legyen olyan személy, akinek nem minden adatát töltjük ki. Pl.: anyja nevét nem adjuk meg

```
INSERT INTO táblanév (A1, ..., An) VALUES (v1, ..., vn);
```

```
INSERT INTO ember (igszam,nev,telepules, utca) VALUES('AA111333','Kis Emma','Pápa','Kossuth u.20.');
```

d) Vegyen fel egy új mezőt az ember táblába település tulajdonság elé az irányítószám tárolására.

```
ALTER TABLE ember ADD irszam smallint unsigned AFTER nev; (előjel nélküli egész típus: 0 .. 65535)
```

e) Töltse fel az irányítószámot legalább az első rekord esetében!

```
UPDATE ember SET irszam=7040 WHERE igszam='AA111222';
```

f) Módosítsa az utca mező típusát karakteres változó 30 hosszúságról 50-re!

```
ALTER TABLE ember MODIFY utca varchar(50);
```

g) Módosítsa az auto tábla „evjarat” mezőjének elnevezését! Röviden legyen „ev” az új tulajdonságnév!

```
ALTER TABLE táblanév CHANGE régi_attr_név új_attr_név típus;
```

```
|ALTER TABLE auto CHANGE evjarat ev year;
```

h) Vegyen fel egy új mezőt az auto táblába a szín (szin) tárolására.

```
ALTER TABLE auto ADD szin varchar(15);
```

i) Vegyen fel egy új mezőt az auto táblába a vételár (ar) tárolására.

```
ALTER TABLE auto ADD ar int;
```

j) Töltse fel a hiányzó adatokkal a két táblázatot!

#### 4. Lekérdezések

Projekció (vetítés, itt oszlopválogatás):

```
SELECT oszlopnév(ek) FROM táblanév ;
```

Szelekció (sorválogatás):

```
SELECT oszlopnév(ek) FROM táblanév WHERE feltétel;
```

a) Jelenítsük meg a tulajdonosok igazolványszámát, nevét és a települést, ahol élnek!

```
SELECT igszam, nev, telepules FROM ember;
```

b) Szűrjük ki a paksi lakosok minden adatát!

```
SELECT * FROM ember WHERE telepules='Paks';
```

c) Paksi lakosok igazolványszáma és neve!

```
SELECT igszam, nev FROM ember WHERE telepules='Paks';
```

d) Jelenítse meg a „K” betővel kezdődő vezetéknevvvel rendelkező embereket!

```
|SELECT * FROM ember WHERE nev like 'K%';
```

e) Listázza ki a „K” betővel kezdődő vezetéknevvvel rendelkező embereket, akik Pakson élnek!

```
|SELECT * FROM ember WHERE nev like 'K%' and telepules='Paks';
```

f) Az ember tábla adatait jelenítse meg, de úgy, hogy a cím egy mezőben legyen kiírva!

```
|SELECT igszam, nev, concat(irszam,' ',telepules,' ',utca) as cim ,anyjaneve FROM ember;
```

g) Írassa ki az autókat évjárat szerint növekvő sorrendben!

```
SELECT * FROM auto order by ev;
```

h) Írassa ki az autókat évjárat szerint csökkenő sorrendben!

```
SELECT * FROM auto order by ev desc;
```

i) Írassa ki az autók össz árát, átlagárát két tizedesjeggyel, a legolcsóbb autót, a legdrágább autót!

(Az eredményeket tartalmazó mezőket nevezze is el!)

```
SELECT sum(ar) as 'Össz ár', round(avg(ar),2) as 'átlagár', min(ar) as 'min', max(ar) as 'max' FROM auto;
```

j) Hány fő lakik Pakson a tulajdonosok közül?

```
SELECT concat(count(*),' fő') as 'paksiak' FROM ember WHERE telepules='Paks';
```

k) *Hány fő lakik Pakson és hány fő Pápán?*

telepules	Hány fő?
Paks	3 fő
Pápa	1 fő

**Csoportosítás:** A csoportosításkor az eredmény táblában minden csoporthoz egy-egy sor keletkezik.

**Csoportválogatás:** A GROUP BY által visszaadott eredménytáblából kiválasztja azokat a sorokat, amelyek eleget tesznek a HAVING utáni feltételnek.

**SELECT** oszlopnév, összesítő függvény **FROM** táblanév  
**GROUP BY** oszlopnév(ek) **HAVING** feltétel

```
SELECT telepules, concat(count(*),' fő') as 'Hány fő?' FROM ember
GROUP BY telepules
HAVING telepules='Paks' or telepules='Pápa';
```

l) *Írassuk ki kinek milyen autója van (minden tábla minden adatát jelenítsük meg)?*

```
SELECT * FROM ember, auto WHERE ember.igszam=auto.igszam;
```

vagy

```
SELECT * FROM ember natural join auto ;
```

m) *Írassuk ki kinek milyen autója van (amire kíváncsiak vagyunk: igazolványszám, név, rendszám, gyártmány, típus)?*

```
SELECT ember.igszam, nev, rendszam, gyartmany, tipus FROM ember, auto
WHERE ember.igszam=auto.igszam;
```

vagy

```
SELECT ember.igszam, nev, rendszam, gyartmany, tipus FROM ember natural join auto;
```

n) *Írassuk ki kinek milyen autója van (amire kíváncsiak vagyunk: igazolványszám, név, rendszám, gyártmány, típus), de csak a paksi lakhellyel rendelkezőket jelenítsük meg az eredménytáblában!*

```
SELECT ember.igszam, nev, rendszam, gyartmany, tipus FROM ember, auto
WHERE ember.igszam=auto.igszam and telepules='Paks';
```

vagy

```
SELECT ember.igszam, nev, rendszam, gyartmany, tipus FROM ember natural join auto
WHERE telepules='Paks';
```

o) *Keressük ki a legolcsóbb autó árát!*

```
SELECT min(ar) FROM auto;
```

p) *Keressük ki a legolcsóbb autó adatait! (lekérdezéssel előállított tábla használata lekérdezésben /a WHERE / záradékban)*

```
SELECT * FROM auto WHERE ar=(SELECT min(ar) FROM auto);
```

q) *Keressük ki a legolcsóbb autó tulajdonosának az adatait!*

```
SELECT ember.igszam, nev, rendszam, gyartmany, ar FROM auto natural join ember
WHERE ar=(SELECT min(ar) FROM auto);
```

vagy

```
SELECT ember.igszam, nev, rendszam, gyartmany, ar FROM auto , ember
WHERE ar=(SELECT min(ar) FROM auto) and ember.igszam=auto.igszam;
```

r) *Kinek hány autója van?*

```
SELECT igszam, concat(count(*),' db') as 'darab' FROM auto natural join ember GROUP BY igszam;
```

s) Kinek van a legtöbb autója?

```
SELECT * FROM
  (SELECT igszam, concat(count(*),' db') as 'darab' FROM auto natural join ember GROUP BY igszam) as y
  WHERE darab= (SELECT max(darab) FROM
  (SELECT igszam, concat(count(*),' db') as 'darab' FROM auto natural join ember GROUP BY igszam)as x);
```

vagy nézetábrával rövidebb:

**Gyorsításra, vagy egyszerűsítésre:** Nézetábrának nevezzük az olyan táblákat, amelyek egy elkérés (SELECT) eredményeként jönnek létre. Tényleges tárolásra nem kerülnek, csak az őket előállító utasítást tárolja az adatbázisrendszer. Amikor szükség van rá, akkor a tárolt utasítást végrehajtva létrehozza az adatbázisrendszer a nézetábrát az adatbázis pillanatnyi tartalmának megfelelően. A nézetábrák lekérésre ugyanúgy használhatók, mint a táblák.

**Nézetábra definiálása:** CREATE VIEW Nézetábranev (attribútumok neve ) AS SELECT...;

```
CREATE VIEW legtobb as
  SELECT igszam, concat(count(*),' db') as 'darab' FROM auto natural join ember GROUP BY igszam;
SELECT * FROM legtobb WHERE darab= (SELECT max(darab) FROM legtobb);
```

t) Melyek azok az autópárok, amelyeknek ugyanaz a színe? Azaz színenként párokba rendezve írassuk ki az autók adatait!

```
SELECT * FROM auto as a1, auto as a2
  WHERE a1.szín=a2.szín and a1.gyártmány<a2.gyártmány;
```

vagy, ha csak a rendszámra, gyártmányra, és a színre vagyunk kíváncsiak:

```
SELECT a1.rendszer, a1.gyártmány, a1.szín, a2.rendszer, a2.gyártmány, a2.szín FROM auto as a1, auto as a2
  WHERE a1.szín=a2.szín and a1.gyártmány<a2.gyártmány;
```

vagy

```
SELECT * FROM
  (SELECT rendszer as rendszer1, gyártmány as gyártmány1, szín FROM auto) as x
  natural join
  (SELECT rendszer as rendszer2, gyártmány as gyártmány2, szín FROM auto) as y
  WHERE gyártmány1<gyártmány2;
```