

JEGYZŐKÖNYV

Adatbázisrendszerek I.

Féléves feladat

Készítette: **Horváth Ákos Zsigmond**

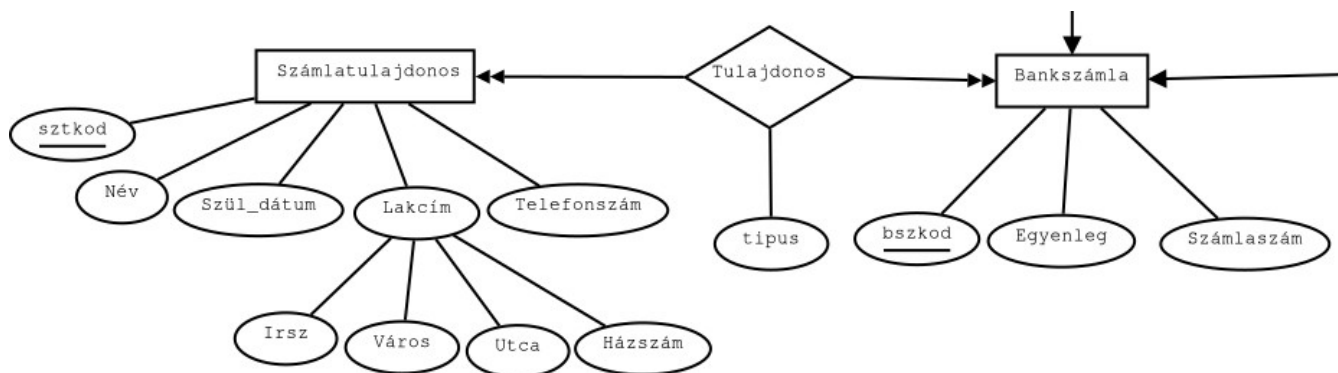
Neptun kód: **R3SZY2**

1a) Az adatbázis ER modell készítése

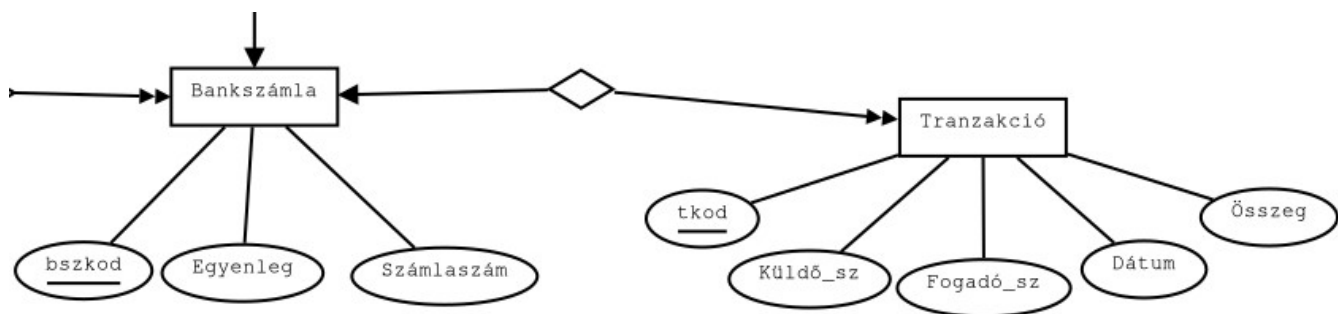
Feladat leírása:

Ebben a feladatban létrehoztam az adatbázis tábláinak az Entity-Relationship modelljét. A diagram létrehozásához a Dia Diagram Editor szoftvert használtam. A .dia forrásfájl megtalálható a Github repositoryban.

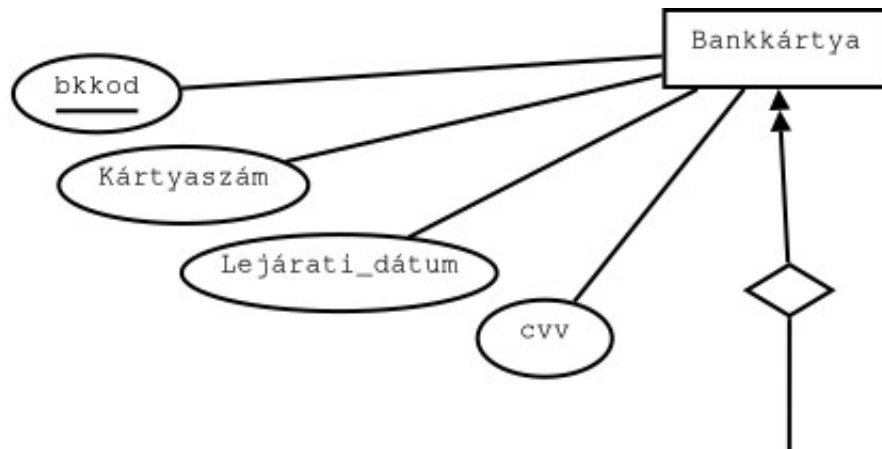
Az adatbázis egy bank adatait tárolja.



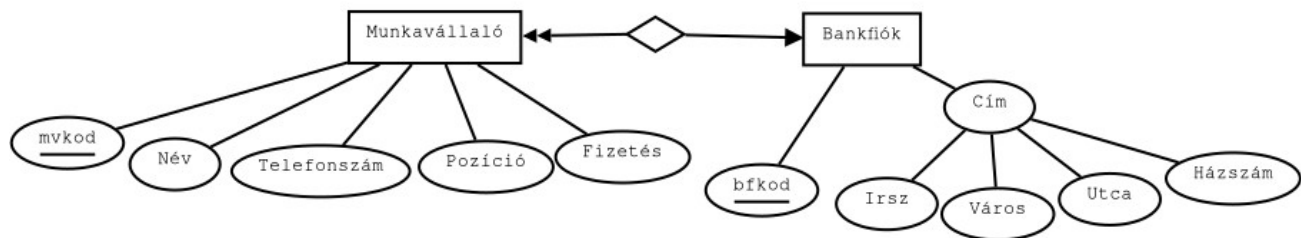
Látható a Számlatulajdonos és Bankszámla tábla, melyek között N:M kapcsolat jön létre, egy közbülső “Tulajdonos” tábla használatával.



A Tranzakció tábla számlák közötti utalásokat tárolhat, a Bankszámla és Tranzakció táblák között 1:N kapcsolat van.



A Bankszámla és Bankkártya tábla között 1:N kapcsolat van.



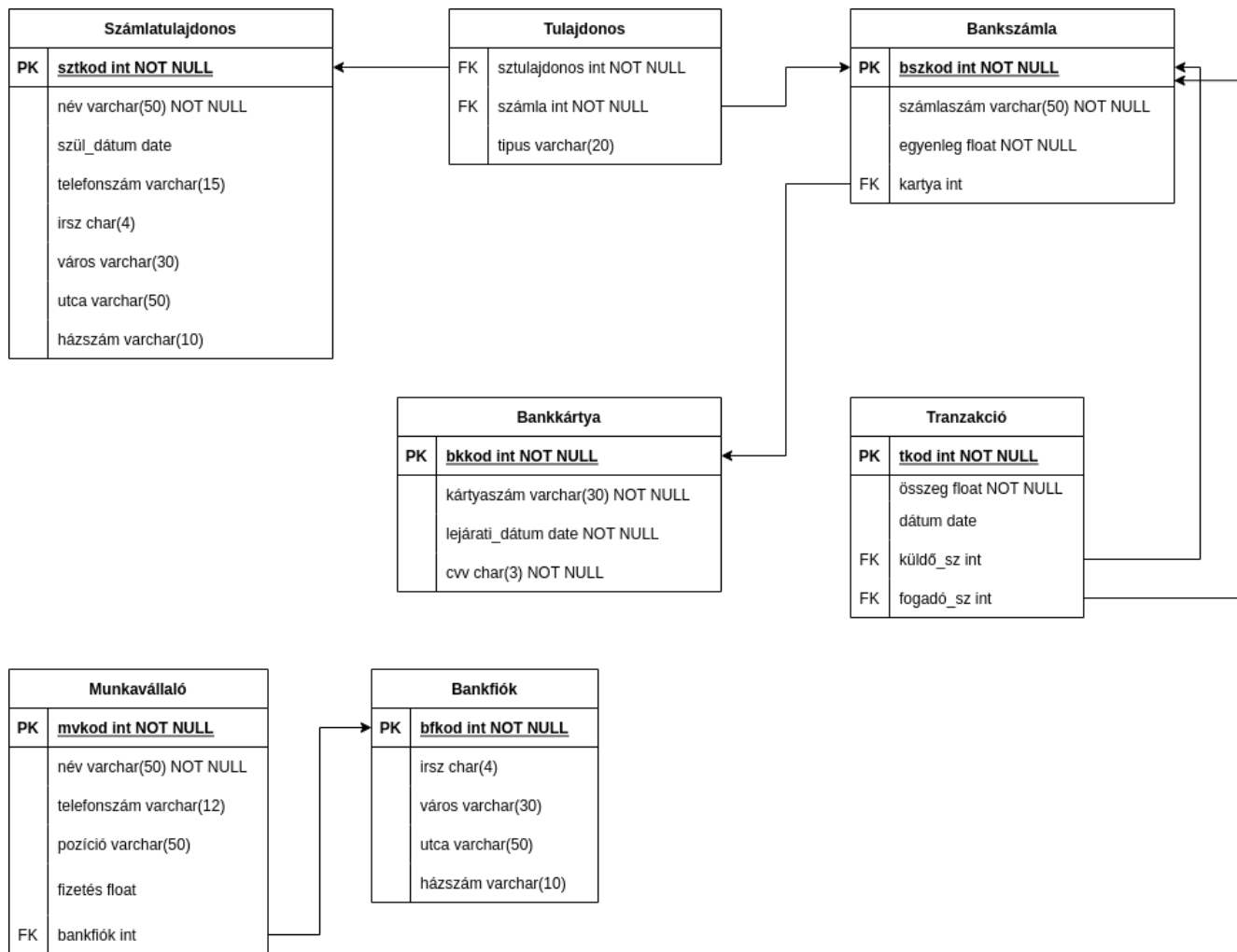
Létrehoztam két táblát a bank Bankfiók épületeinek az adatainak az eltárolására, illetve azoknak a Munkavállalók számára is, akik az adott Bankfiókban dolgoznak.

A Bankfiók és Munkavállaló táblák között 1:N kapcsolat van, aminek a jelentése az, hogy egy Bankfiók rekordhoz több Munkavállaló is tartozhat.

1b) Az ER modell konvertálása relációs modellre

A feladat leírása:

A meglévő ER modellben szereplő táblákat és kapcsolatokat elkészítettem relációs modellben is. A relációs modell diagramot a <https://app.diagrams.net/> weboldalon készítettem. Ennek a diagramnak is elérhető a forrásfájlja a repositoryban.



A relációs modellben a táblák mezőinek a típusait is megszabtam.

Az ER modellben szereplő Tulajdonos kapcsolatból tábla lett, amelyben a Bankszámla és Számlatulajdonos közötti kapcsolatot két idegen kulcs tarja fenn. Egy “típus” mező is van a Tulajdonos táblában, ami akkor lesz használatban, ha több tulajdonos tartozik egy bankszámlához. Leírja azt, hogy milyen szerepe van a többi tulajdonosnak, pl. meghatalmazott, társtulajdonos, stb.

Pár fontosabb mezőben is láthatók a “NOT NULL” kikötések, melyek azt jelentik, hogy ezeknek a mezőknek kötelezően kell tartalmazniuk valamilyen értéket.

1c) Az adatbázis relációs séma készítése

A feladat leírása:

Bankszámla[bszkod, számlaszám, egyenleg, kártya]

Számlatulajdonos[sztkod, név, szül_dátum, telefonszám, irsz, város, utca, házszám]

Tulajdonos[sztulajdonos, számla, típus]

Tranzakció[tkod, összeg, dátum, küldő_sz, fogadó_sz]

Bankkártya[bkkod, kártyaszám, lejárat_i dátum, cvv]

Bankfiók[bfkod, irsz, város, utca, házszám]

Munkavállaló[mvkod, név, pozíció, fizetés, bankfiók]

1d) A táblák elkészítése

A feladat leírása:

Az adatbázis elkészítéséhez MariaDB szoftvert használtam, ami a MySQL forkja, és teljes mértékben kompatibilis az Oracle MySQL-el.

A táblákat a terminálban, a DBMS szöveges kezelőfelülete segítségével készítettem.

Az .sql fájl elérhető a github repositoryban.

Számlatulajdonos tábla:

```
create table Számlatulajdonos(  
sztkod int NOT NULL,  
név varchar(50) NOT NULL,  
szül_dátum date,  
telefonszám varchar(15),  
irsz char(4),  
város varchar(30),  
utca varchar(50),  
házszám varchar(10),  
primary key(sztkod)  
);
```

Bankkártya:

```
create table Bankkártya(  
bkkod int NOT NULL,  
kártyaszám varchar(30) NOT NULL,  
lejárat_i_dátum date NOT NULL,  
cvv char(3) NOT NULL,  
primary key(bkkod)  
);
```

Bankszámla:

```
create table Bankszámla(  
bszkod int NOT NULL,  
számlaszám varchar(50) NOT NULL,  
egyenleg float NOT NULL,  
kártya int,  
primary key(bszkod),  
foreign key(kártya) references Bankkártya(bkkod)  
);
```

Tranzakció:

```
create table Tranzakció(  
tkod int NOT NULL,  
float összeg,  
dátum date,  
küldő_sz int,  
fogadó_sz int,  
primary key(tkod),  
foreign key(küldő_sz) references Bankszámla(bszkod),  
foreign key (fogadó_sz) references Bankszámla(bszkod)  
);
```

Bankfiók:

```
create table Bankfiók(  
bfkod int NOT NULL,  
irsz char(4),  
város varchar(30),  
utca varchar(50),  
házszám varchar(10),  
primary key(bfkod)  
);
```

Munkavállaló:

```
create table Munkavállaló(  
mvkod int NOT NULL,  
név varchar(50) NOT NULL,  
telefonszám varchar(12),  
pozíció varchar(50),  
fizetés float,  
bankfiók int,  
primary key(mvkod),  
foreign key(bankfiók) references Bankfiók(bfkod)  
);
```

Tulajdonos:

```
create table Tulajdonos(  
sztulajdonos int NOT NULL,  
számla int NOT NULL,  
típus varchar(20)  
);  
  
alter table Tulajdonos add constraint szamla_fk foreign key(számla)  
references Bankszámla(bszkod)  
  
alter table Tulajdonos add constraint sztulaj_fk foreign  
key(sztulajdonos) references Számlatulajdonos(sztkod)
```

Elkészült táblák:

```
MariaDB [db1_bank]> describe Bankszámla;
```

Field	Type	Null	Key	Default	Extra
bszkod	int(11)	NO	PRI	NULL	
számlaszám	varchar(50)	NO		NULL	
egyenleg	float	NO		NULL	
kártya	int(11)	YES	MUL	NULL	

4 rows in set (0.000 sec)

```
MariaDB [db1_bank]> describe Számlatulajdonos;
```

Field	Type	Null	Key	Default	Extra
sztkod	int(11)	NO	PRI	NULL	
név	varchar(50)	NO		NULL	
szül_dátum	date	YES		NULL	
telefonszám	varchar(15)	YES		NULL	
irsz	char(4)	YES		NULL	
város	varchar(30)	YES		NULL	
utca	varchar(50)	YES		NULL	
házsza	varchar(10)	YES		NULL	

8 rows in set (0.000 sec)

```
MariaDB [db1_bank]> describe Tulajdonos;
```

Field	Type	Null	Key	Default	Extra
sztulajdonos	int(11)	NO	MUL	NULL	
számla	int(11)	NO	MUL	NULL	
típus	varchar(20)	YES		NULL	

3 rows in set (0.000 sec)

MariaDB [db1_bank]> describe Tranzakció;

Field	Type	Null	Key	Default	Extra
tkod	int(11)	NO	PRI	NULL	
összeg	float	YES		NULL	
dátum	date	YES		NULL	
küldő_sz	int(11)	YES	MUL	NULL	
fogadó_sz	int(11)	YES	MUL	NULL	

5 rows in set (0.000 sec)

MariaDB [db1_bank]> describe Bankkártya;

Field	Type	Null	Key	Default	Extra
bkkod	int(11)	NO	PRI	NULL	
kártyaszám	varchar(30)	NO		NULL	
lejárati_dátum	date	NO		NULL	
cvv	char(3)	NO		NULL	

4 rows in set (0.000 sec)

MariaDB [db1_bank]> describe Munkavállaló;

Field	Type	Null	Key	Default	Extra
mvkod	int(11)	NO	PRI	NULL	
név	varchar(50)	NO		NULL	
telefonszám	varchar(12)	YES		NULL	
pozíció	varchar(50)	YES		NULL	
fizetés	float	YES		NULL	
bankfiók	int(11)	YES	MUL	NULL	

6 rows in set (0.000 sec)

```
MariaDB [db1_bank]> describe Bankfiók;
```

Field	Type	Null	Key	Default	Extra
bfkod	int(11)	NO	PRI	NULL	
irsz	char(4)	YES		NULL	
város	varchar(30)	YES		NULL	
utca	varchar(50)	YES		NULL	
házzsám	varchar(10)	YES		NULL	

5 rows in set (0.000 sec)

1e) A táblák feltöltése

A feladat leírása:

Feltöltöttem a táblákat kitalált adatokkal, úgy, hogy minden táblában legyen 4-5 rekord:

Bankkártya tábla:

```
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(1,"1111-2222-3333-444", '2023-10-01', '123');
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(2,"1111-2222-3333-555", '2023-08-01', '456');
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(3,"1111-1234-4567-5555", '2023-07-01', '222');
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(4,"1122-2233-4455-6677", '2023-11-01', '777');
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(5,"7777-8888-9999-0000", '2020-11-01', '133');
insert into Bankkártya(bkkod, kártyaszám, lejáratidátum, cvv) values
(6,"7777-6666-5555-4444", '2021-12-01', '111');
```

Számlatulajdonos tábla:

```
insert into Számlatulajdonos (sztkod, név, szüldátum, telefonszám,
irsz, város, utca, házzsám) values(1, "Kiss István", '1980-5-11',
"06201122345", 1033, "Budapest", "Petőfi utca", 1);
insert into Számlatulajdonos (sztkod, név, szüldátum, telefonszám,
irsz, város, utca, házzsám) values(2, "Nagy Béla", '1990-7-20',
"06203322111", 3525, "Miskolc", "Fa utca", 15);
insert into Számlatulajdonos (sztkod, név, szüldátum, telefonszám,
irsz, város, utca, házzsám) values(3, "Kovács Erzsébet", '1975-1-15',
"06304455333", 4000, "Debrecen", "Erzsébet utca", 3);
```

```

insert into Számlatulajdonos (sztkod, név, szül_dátum, telefonszám,
irsz, város, utca, házszám) values(4, "Tóth János", '2000-1-2',
"0630111111", 7600, "Pécs", "Pécs utca", 10);
insert into Számlatulajdonos (sztkod, név, szül_dátum, telefonszám,
irsz, város, utca, házszám) values(5, "Szabó Mária", '1982-7-8',
"06307777888", 1097, "Budapest", "Pest utca", 1);
insert into Számlatulajdonos (sztkod, név, szül_dátum, telefonszám,
irsz, város, utca, házszám) values(6, "Kovács István", '1977-3-4',
"06302222333", 4000, "Debrecen", "Erzsébet utca", 3);

```

Bankszámla tábla:

```

insert into Bankszámla(bszkod, számlaszám, egyenleg, kártya) values
(1, "111111-222222", 120000, 1);
insert into Bankszámla(bszkod, számlaszám, egyenleg, kártya) values
(2, "333-444", 200000, 3);
insert into Bankszámla(bszkod, számlaszám, egyenleg, kártya) values
(3, "123-456", 35000.2, 2);
insert into Bankszámla(bszkod, számlaszám, egyenleg, kártya) values
(4, "678-912", 1000300, 4);
insert into Bankszámla(bszkod, számlaszám, egyenleg, kártya) values
(5, "555-444-333", 678320.8, 5);

```

Tulajdonos tábla:

```

insert into Tulajdonos(sztulajdonos, számla) values(1, 1);
insert into Tulajdonos(sztulajdonos, számla) values(2, 2);
insert into Tulajdonos(sztulajdonos, számla) values(3, 3);
insert into Tulajdonos(sztulajdonos, számla) values(4, 4);
insert into Tulajdonos(sztulajdonos, számla) values(5, 5);
insert into Tulajdonos(sztulajdonos, számla, típus) values(6, 5,
"Társtulajdonos");

```

Tranzakció tábla:

```

insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (1, 10000, '2021-01-11', 1, 2)
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (2, 25000, '2021-5-23', 3, 5);
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (3, 1000, '2019-6-12', 4, 5);
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (4, 6700, '2018-9-20', 5, 1);
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (5, 7800, '2021-4-6', 5, 3);
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (6, 300, '2019-12-7', 1, 4);
insert into Tranzakció(tkod, összeg, dátum, küldő_sz, fogadó_sz)
values (7, 300, '2019-12-7', 1, 4);

```

Bankfiók tábla:

```
insert into Bankfiók(bfkod, irsz, város, utca, házszám) values(1, 1234, "Budapest", "Pest utca", 12);
insert into Bankfiók(bfkod, irsz, város, utca, házszám) values(2, 3500, "Miskolc", "Miskolc utca", 25);
```

Munkavállaló tábla:

```
insert into Munkavállaló(mvkod, név, telefonszám, pozíció, fizetés, bankfiók) values(1, "Lantos Ferenc", 0612342423, "Ügyfélszolgálat", "250000", 1);
insert into Munkavállaló(mvkod, név, telefonszám, pozíció, fizetés, bankfiók) values(2, "Tóth Gyula", 06305555777, "Biztonsági őr", "200000", 1);
insert into Munkavállaló(mvkod, név, telefonszám, pozíció, fizetés, bankfiók) values(3, "Tóth Gyula", 06305555777, "Biztonsági őr", "200000", 1);
insert into Munkavállaló(mvkod, név, telefonszám, pozíció, fizetés, bankfiók) values(4, "Nagy István", 06304444333, "Ügyfélszolgálat", "250000", 2);
```

1f) Lekérdezések (SQL és relációs algebra) min. 10 lekérdezés

Feladat leírása:

Városonkénti számlatulajdonosok száma:

```
select város,count(bszkod) from Bankszámla inner join Tulajdonos on Tulajdonos.számla=Bankszámla.bszkod inner join Számlatulajdonos on Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos group by város;
```

```
MariaDB [db1_bank]> select város,count(bszkod) from Bankszámla inner join Tulajdonos on Tulajdonos.számla=Bankszámla.bszkod inner join Számlatulajdonos on Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos group by város;
+-----+-----+
| város | count(bszkod) |
+-----+-----+
| Budapest | 2 |
| Debrecen | 2 |
| Miskolc | 1 |
| Pécs | 1 |
+-----+-----+
4 rows in set (0.001 sec)
```

Mennyi bankszámla van, amelynek az egyenlege több, mint 200000 forint:

```
select count(*) from Bankszámla where egyenleg>200000;
```

```
MariaDB [db1_bank]> select count(*) from Bankszámla where egyenleg>200000;
+-----+
| count(*) |
+-----+
|         2 |
+-----+
1 row in set (0.000 sec)
```

Egy évnél régebbi tranzakciók:

```
select * from Tranzakció where dátum < curdate() - interval 1 year;
```

```
MariaDB [db1_bank]> select * from Tranzakció where dátum < curdate() - interval 1 year;
+-----+-----+-----+-----+-----+
| tkod | összeg | dátum      | küldő_sz | fogadó_sz |
+-----+-----+-----+-----+-----+
| 3    | 25000 | 2020-06-12 | 2        | 1         |
| 4    | 1000  | 2019-06-12 | 4        | 5         |
| 5    | 6700  | 2018-09-20 | 5        | 1         |
| 7    | 300   | 2019-12-07 | 1        | 4         |
+-----+-----+-----+-----+-----+
4 rows in set (0.000 sec)
```

Budapesti bankfiókban dolgozók neve és fizetése:

```
select név,fizetés from Munkavállaló inner join Bankfiók on
Bankfiók.bfkod=Munkavállaló.bankfiók where város="Budapest";
```

```
MariaDB [db1_bank]> select név,fizetés from Munkavállaló inner join Bankfiók on Bankfiók.bfkod=Munkavállaló.bankfiók where város="Budapest";
+-----+-----+
| név      | fizetés |
+-----+-----+
| Lantos Ferenc | 250000 |
| Pintér János  | 350000 |
| Tóth Gyula    | 200000 |
+-----+-----+
3 rows in set (0.000 sec)
```

Bankkártyák amelyek lejártak, vagy egy év múlva le fognak járni:

```
select * from Bankkártya where lejáratidátum < curdate() + interval 1 year;
```

```
MariaDB [db1_bank]> select * from Bankkártya where lejáratidátum < curdate() + interval 1 year;
+-----+-----+-----+-----+
| bkkod | kártyaszám      | lejáratidátum | cvv |
+-----+-----+-----+-----+
|      5 | 4444-3333-2222-1111 | 2020-11-01    | 133 |
|      6 | 3333-3333-3333-9999 | 2021-12-01    | 111 |
+-----+-----+-----+-----+
2 rows in set (0.000 sec)
```

Összes pénz, amely a bankszámlákon van:

```
select sum(egyenleg) from Bankszámla;
```

```
MariaDB [db1_bank]> select sum(egyenleg) from Bankszámla;
+-----+
| sum(egyenleg) |
+-----+
| 2033621.01171875 |
+-----+
1 row in set (0.000 sec)
```

Kiss István nevű számlatulajdonos számlájáról indított tranzakciók összege és dátuma:

```
select összeg,dátum from Tranzakció inner join Bankszámla on
Bankszámla.bszkod=küldő_sz inner join Tulajdonos on
Tulajdonos.számla=Bankszámla.bszkod inner join Számlatulajdonos on
Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos where
Számlatulajdonos.név="Kiss István";
```

```
MariaDB [db1_bank]> select összeg,dátum from Tranzakció inner join Bankszámla on Bankszámla.bszkod=küldő_sz inner join Tulajdonos on Tulajdonos.számla=Bankszámla.bszkod inner join Számlatulajdonos on Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos where Számlatulajdonos.név="Kiss István";
+-----+-----+
| összeg | dátum      |
+-----+-----+
| 10000  | 2021-01-11 |
| 300    | 2019-12-07 |
+-----+-----+
2 rows in set (0.000 sec)
```

Számlák átlagegyenlege, városonként:

```
select város, avg(egyenleg) from Bankszámla inner join Tulajdonos on
Tulajdonos.számla=Bankszámla.bszkod inner join Számlatulajdonos on
Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos group by város;
```

```
MariaDB [db1_bank]> select város, avg(egyenleg) from Bankszámla inner join Tulajdonos on Tulajdonos.számlla=Bankszámla.bszkod inner join Számlatulajdonos on Számlatulajdonos.sztkod=Tulajdonos.sztulajdonos group by város;
```

város	avg(egyenleg)
Budapest	399160.40625
Debrecen	356660.505859375
Miskolc	200000
Pécs	1000300

```
4 rows in set (0.000 sec)
```

Munkavállalók átlagfizetése, pozíció szerint:

```
select pozíció,avg(fizetés) from Munkavállaló group by pozíció;
```

```
MariaDB [db1_bank]> select pozíció,avg(fizetés) from Munkavállaló group by pozíció;
```

pozíció	avg(fizetés)
Biztonsági őr	200000
Tanácsadó	350000
Ügyfélszolgálat	250000

```
3 rows in set (0.000 sec)
```

2021-ben végzett tranzakciók átlag értéke:

```
select avg(összeg) from Tranzakció where year(dátum)=2021;
```

```
MariaDB [db1_bank]> select avg(összeg) from Tranzakció where year(dátum)=2021;
```

avg(összeg)
14266.666666666666

```
1 row in set (0.000 sec)
```

30 évnél idősebb számlatulajdonosok száma:

```
select count(*) from Számlatulajdonos where timestampdiff(year,
szül_dátum, curdate()) > 30;
```

```
MariaDB [db1_bank]> select count(*) from Számlatulajdonos where timestampdiff(year, szül_dátum, c
urdate()) > 30;
+-----+
| count(*) |
+-----+
|         5 |
+-----+
1 row in set (0.000 sec)
```