

# JEGYZŐKÖNYV

Adatbázisrendszerek I.

Féléves feladat

Készítette: Orosz Dániel  
Neptunkód: C5S7FM

## A feladat leírása:

A féléves feladatom témája egy olyan adatbázis, amely edzőtermek adatbázisát mutatja be. A tagok adataira, bérlet tulajdonságokra, személyi edzők adataira lehet rákeresni. Öt darab egyed tulajdonságait fogjuk vizsgálni.

## ER modell leírása:

### Edzőterem egyed:

- Ekód: Az Edzőterem egyed elsődleges kulcsa
- Név: Az edzőterem nevét tárolja
- Cím: Összetett tulajdonság, ami az edzőterem címét tárolja

### Személyi edző:

- SzKód: A Személyi edző egyed elsődleges kulcsa.
- Név: A személyi edző nevét tárolja
- Telefonszám: A személyi edző telefonszámát tárolja
- Típus : Az edzések típusát tárolja, többértékű tulajdonság.
- Lakcím: Összetett tulajdonság, az edző címét tárolja

### Tag egyed:

- TKód: A Tag egyed elsődleges kulcsa
- Név: A tagok neveit tárolja
- Telefonszám: A tagok telefonszámait tárolja
- Születési idő: A tagok születési dátumát tárolja
- Lakcím: A tagok lakcímét tárolja
- Email cím: A tagok email címét tárolja
- Kor: Származtatott tulajdonság, a születési időből kilehett számolni. A tagok kora

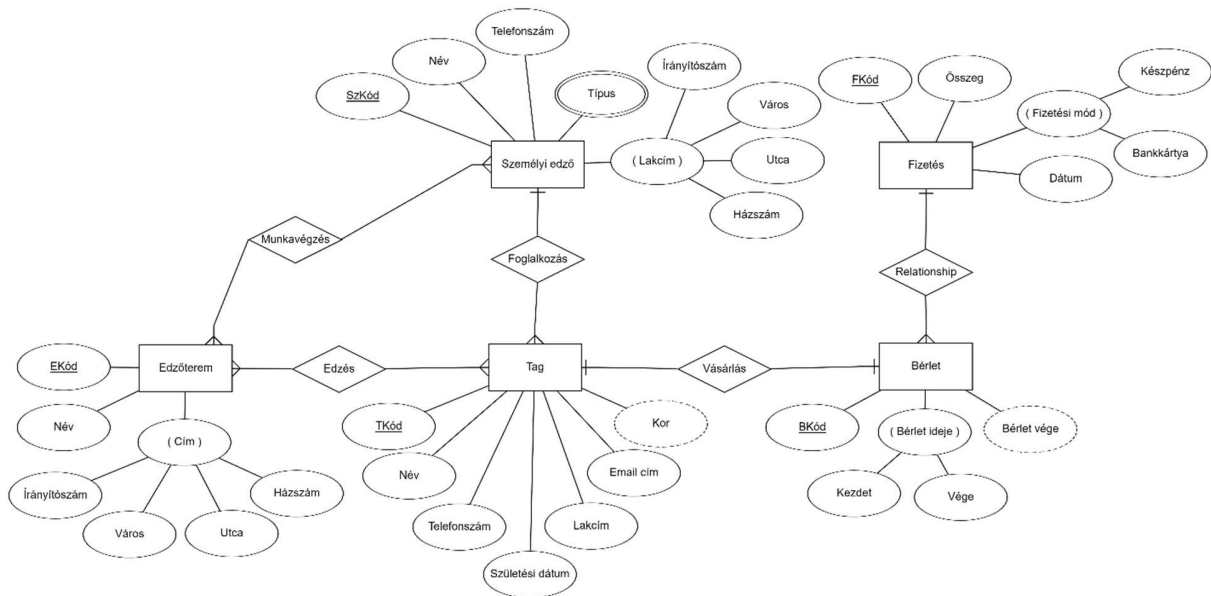
### Bérlet egyed:

- BKód: A Bérlet egyed elsődleges kulcsa
- Bérlet ideje: Összetett tulajdonság, a bérlet kezdetét és végét tárolja
- Bérlet vége: Származtatott tulajdonság, a bérlet idejéből kiszámítható. A bérlet lejárat

### Fizetés:

- FKód: A Fizetés egyed elsődleges kulcsa
- Összeg: A fizetés végösszegét tárolja
- Fizetési mód: Összetett tulajdonság, készpénzes vagy bankkártyás fizetést tárolja
- Dátum: A fizetés dátumát tárolja

## ER Modell:



## Kapcsolatok magyarázata:

Edzőterem - Tag: Több a többhöz kapcsolat van a két egyed között, mert egy edzőterembe több ember vagy tag is járhat, illetve egy ember több edzőterembe is járhat.

Edzőterem – Személyi edző: Több a többhöz kapcsolat van a két egyed között, mert több személyi edző is lehet egy edzőteremben, illetve egy személyi edző több edzőteremben is dolgozhat.

Személyi edző – Tag: Egy a többhöz kapcsolat van a két egyed között, mert egy tagnak egy személyi edzője lesz, de egy személyi edző több taggal is foglalkozhat.

Tag – Bérlet: Egy az egyhez kapcsolat van a két egyed között, mert egy ember csak egy bérletet vesz csak és egy bérletnek csak egy tulajdonosa lehet.

Bérlet – Fizetés: Egy a többhöz kapcsolat van a két egyed között, mert csak egyszer kell fizetni a bérletért, de egyetlen fizetéssel akár több bérletet is vásárolhat.

## Relációs modell:

A **Személyi edző** egyedből tábla lesz, elsődleges kulcsa az SzKód lesz. Tulajdonságai a *Név*, *Telefonszám*, *Írányítószám*, *Város*, *Utca*, *Házzszám*. Három idegen kulcs is ideje mutat.

A **Tag** egyedből tábla lesz, amelynek esődleges kulcsa a TKód. Tulajdonságai a *Név*, *Telefonszám*, *Lakcím*, *Születési dátum*, *Email cím*. Kettő idegen kulcsa van, az egyik az SzKód, ami a **Személyi edző** táblára mutat, és a BKód, ami a **Bérlet** táblára mutat.

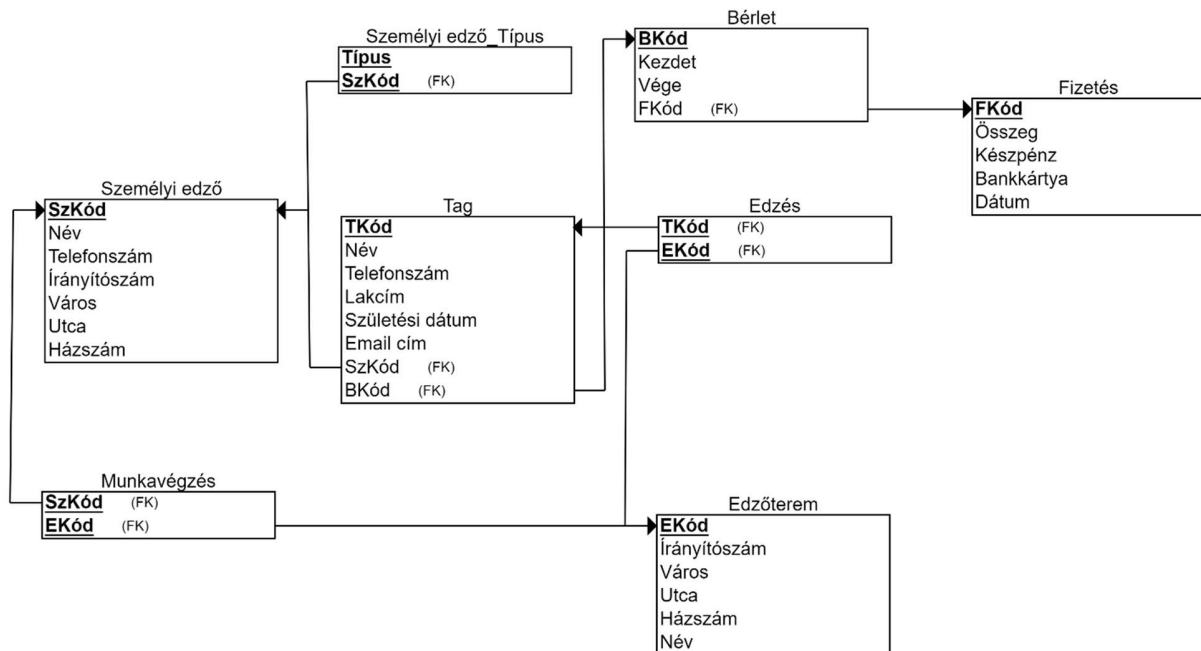
A **Bérlet** egyedből tábla lesz, elsődleges kulcsa a BKód. Tulajdonságai a *Bérlet kezdete* és a *Bérlet vége*. Egy idegen kulcsa van, ami a **Fizetés** táblára mutat.

Az **Edzőterem** egyedből tábla lesz, elsődleges kulcsa az EKód. Tulajdonságai az *Írányítószám*, *Város*, *Utca*, *Házzszám* és *Név*.

A **Fizetés** egyedből is tábla lesz, ennek elsődleges kulcsa az FKód. Tulajdonságai az *Összeg*, *Készpénz*, *Bankkártya* és a *Dátum*.

Az **Edzés** kapcsolatból egy kapcsolattábla lesz, ami kettő idegenkulcsot tartalmaz, a TKód, ami a **Tag** táblát, és az EKód, ami az **Edzőterem** táblát kapcsolja össze.

A **Munkavégzés** kapcsolatból egy kapcsolattábla lesz amiben kettő idegenkulcs lesz, az SzKód és az EKód, amik a **Személyi edző** és **Edzőterem** táblát köti össze.



## Az adatbázis relációs sémái:

Személyi\_edző [SzKód, Név, Telefonszám, Írányítószám, Város, Utca, Házszám]  
Tag [TKód, Név, Telefonszám, Lakcím, Születési dátum, Email\_cím, SzKód, BKód]  
Bérlet [BKód, Kezdet, Vége, FKód]  
Fizetés [FKód, Összeg, Készpénz, Bankkártya, Dátum]  
Edzőterem [EKód, Írányítószám, Város, Utca, Házszám, Név]  
Személyi\_edző\_Típus [Típus, SzKód]  
Munkavégzés [SzKód, EKód]  
Edzés [TKód, EKód]

## A táblák létrehozása:

```
create table Személyi_edző(  
    SzKód int primary key,  
    Név varchar(30) not null,  
    Telefonszám varchar(14),  
    Írányítószám number,  
    Város varchar(30),  
    Utca varchar(30)  
    Házszám number  
);
```

```
create table Fizetés(  
    FKód int primary key,  
    Összeg number,  
    Készpénz number(1)  
    Bankkártya number(1)  
    Dátum date  
);
```

```
create table Edzőterem(  
    EKód int primary key,  
    Írányítószám number,  
    Város varchar(30),  
    Utca varchar(30)  
    Házszám number  
    Név varchar(30)  
);
```

```
create table Munkavégzés(  
    SzKód int,  
    EKód int,  
    primary key (SzKód, EKód),  
    foreign key (SzKód) references Személyi_edző(SzKód),  
    foreign key (EKód) references Edzőterem(EKód)  
);
```

```
create table Személyi_edző_Típus(  
    Típus varchar(30) not null,  
    SzKód int,  
    primary key (Típus, SzKód)  
    foreign key(SzKód) references Személyi_edző(SzKód)  
);
```

```
create table Bérlet(  
    BKód int primary key,  
    Kezdet date,  
    Vége date,  
    FKód int  
    foreign key (FKód) references Fizetés(FKód)  
);
```

```
create table Tag(  
    TKód int primary key,  
    Név varchar(30) not null,  
    Telefonszám varchar(14),  
    Lakcím varchar(60),  
    Születési_dátum date,  
    Email_cím varchar(25),  
    SzKód int,  
    BKód int,  
    foreign key (SzKód) references Személyi_edző(SzKód),  
    foreign key (BKód) references Bérlet(BKód),  
);
```

```
create table Edzés(  
    TKód int,  
    EKód int,  
    primary key (TKód, EKód),  
    foreign key (TKód) references Tag(TKód)  
    foreign key (EKód) references Edzőterem(EKód)  
);
```

## A táblák feltöltése:

### Fizetés tábla:

```
begin
insert into Fizetés values (1, '1', '0', '06/12/2021', '8000');
insert into Fizetés values (2, '0', '1', '09/06/2021', '4000');
insert into Fizetés values (3, '0', '1', '03/27/2020', '10000');
insert into Fizetés values (4, '0', '1', '09/09/2020', '7000');
insert into Fizetés values (5, '1', '0', '11/14/2021', '13000');
insert into Fizetés values (6, '0', '1', '10/17/2021', '5000');
insert into Fizetés values (7, '0', '1', '05/24/2021', '1000');
insert into Fizetés values (8, '1', '0', '12/29/2021', '3500');
insert into Fizetés values (9, '0', '1', '03/07/2021', '12000');
insert into Fizetés values (10, '0', '1', '02/27/2021', '6000');
end;
```

### Edzőterem tábla:

```
begin
insert into Edzőterem values(1, 'King Gym', '3700', 'Kazincbarcika', 'Béke tér', '3');
insert into Edzőterem values(2, 'BB Gym', '3500', 'Miskolc', 'Szeles utca', '71');
insert into Edzőterem values(3, 'Not Jam Fitness', '3500', 'Miskolc', 'Széchenyi István út', '3');
insert into Edzőterem values(4, 'Kingsize Gym', '3500', 'Miskolc', 'Csokonai Vitéz Mihály utca', '37');
insert into Edzőterem values(5, 'Wellnesss Fitness', '3433', 'Nyékládháza', 'Erkel utca', '22');
insert into Edzőterem values(6, 'Planet Fitness', '3580', 'Tiszaújváros', 'Rózsa tér', '18');
insert into Edzőterem values(7, 'Metropoll Fitness', '4090', 'Polgár', 'Hajdú utca', '5');
insert into Edzőterem values(8, 'M1 Fitness', '3434', 'Mályi', 'Akácos utca', '1');
insert into Edzőterem values(9, 'Cutler Fitness', '3770', 'Sajószentpéter', 'Fecske út', '4');
insert into Edzőterem values(10, 'Barcika Fitness', '3700', 'Kazincbarcika', 'Pollack Mihály utca', '48');
end;
```

### Személyi edző tábla:

```
begin
insert into Személyi_edző values(1, '3500 Miskolc Árok utca 19', 'Molnár Levente', '+36708832584');
insert into Személyi_edző values(2, '3434 Mályi Pillangó utca 2', 'Bagi Dávid', '+36305445549');
insert into Személyi_edző values(3, '4090 Polgár Toldi utca 40', 'Fazekas Mátyás', '+36301232587');
insert into Személyi_edző values(4, '3580 Tiszaújváros Templom út 11', 'Molnár Andrea', '+36202346843');
insert into Személyi_edző values(5, '3500 Miskolc Gyár utca 4', 'Kovács Erik',
```

```

'+36202678300');
insert into Személyi_edző values(6, '3500 Miskolc Irma utca 26', 'Juhász Anna',
'+36202223579');
insert into Személyi_edző values(7, '4090 Polgár Szőlő utca 1', 'Boldog Zsolt',
'+36705274367');
insert into Személyi_edző values(8, '3433 Nyékládháza Boglárka utca 16', 'Fazekas Dániel',
'+36302234562');
insert into Személyi_edző values(9, '3434 Mályi Tavasz utca 6', 'Balogh Olivér',
'+36302235575');
insert into Személyi_edző values(10, '3500 Miskolc Napfény utca 30', 'Tompai János',
'+36702986577');
end;

```

### **Munkavégzés tábla:**

```

begin
insert into Dolgozik values(1, 6);
insert into Dolgozik values(2, 3);
insert into Dolgozik values(3, 8);
insert into Dolgozik values(4, 2);
insert into Dolgozik values(5, 1);
insert into Dolgozik values(6, 9);
insert into Dolgozik values(7, 10);
insert into Dolgozik values(8, 4);
insert into Dolgozik values(9, 5);
insert into Dolgozik values(10, 7);
end;

```

### **Személyi\_edző\_Típus tábla:**

```

begin
insert into Személyi_Edző_Típus values('Spinracing', 6);
insert into Személyi_Edző_Típus values('Aerob', 5);
insert into Személyi_Edző_Típus values('TRX', 3);
insert into Személyi_Edző_Típus values('Cardio', 7);
insert into Személyi_Edző_Típus values('Aerob', 10);
insert into Személyi_Edző_Típus values('Cross training', 1);
insert into Személyi_Edző_Típus values('Súlyemelés', 4);
insert into Személyi_Edző_Típus values('Aerobic', 8);
insert into Személyi_Edző_Típus values('Cross training', 2);
insert into Személyi_Edző_Típus values('Súlyemelés', 9);
end;

```



### **Bérlet tábla:**

```
begin
insert into Bérlet values(1, '01/14/2021', '02/14/2021', 4);
insert into Bérlet values(2, '07/25/2021', '08/25/2021', 2);
insert into Bérlet values(3, '11/03/2020', '11/15/2020', 8);
insert into Bérlet values(4, '05/05/2021', '05/17/2021', 1);
insert into Bérlet values(5, '05/21/2020', '06/21/2020', 3);
insert into Bérlet values(6, '08/14/2021', '09/14/2021', 5);
insert into Bérlet values(7, '04/22/2021', '05/12/2021', 6);
insert into Bérlet values(8, '05/09/2021', '06/09/2021', 7);
insert into Bérlet values(9, '02/20/2021', '03/20/2020', 10);
insert into Bérlet values(10, '01/04/2020', '01/16/2020', 9);
end;
```

### **Tag tábla:**

```
begin
insert into Tag values(1, 'Jakab József', '+36305658423', '3433 Mályi Park utca 1',
'06/23/1997', 7, 5);
insert into Tag values(2, 'Horváth Cintia', '+36302646759', '3433 Nyékládháza Iskola utca 7',
'03/05/2001', 2, 1);
insert into Tag values(3, 'Madách Dániel', '+36205234532', '3580 Tiszaújváros Dohány út 17',
'09/23/2005', 3, 2);
insert into Tag values(4, 'Kiss Béla', '+36700343675', '3770 Sajószentpéter Harmat utca 30',
'08/23/1999', 6, 10);
insert into Tag values(5, 'Nagy Marcell', '+36302549301', '3433 Nyékládháza Orgona utca 21',
'09/18/2001', 5, 8);
insert into Tag values(6, 'Smidt Gábor', '+36701067496', '3434 Mályi Szeder utca 3',
'01/26/1989', 8, 9);
insert into Tag values(7, 'Balogh András', '+36207234769', '4090 Polgár Hévíz utca 15',
'02/14/1994', 1, 4);
insert into Tag values(8, 'Kovács Anna', '+36203016437', '3500 Miskolc Szeles utca 11',
'05/26/2002', 9, 7);
insert into Tag values(9, 'Halász Géza', '+36703463760', '3580 Tiszaújváros Tisza út 5',
'03/29/2005', 10, 6);
insert into Tag values(10, 'Lakatos Antal', '+36300523797', '3770 Sajószentpéter Kálvin tér 4',
'04/08/1982', 4, 3);
end;
```

### **Edzés tábla:**

```
begin
insert into Edzés values(1, 4);
insert into Edzés values(2, 6);
insert into Edzés values(3, 7);
```

```
insert into Edzés values(4, 9);
insert into Edzés values(5, 2);
insert into Edzés values(6, 4);
insert into Edzés values(7, 3);
insert into Edzés values(8, 5);
insert into Edzés values(9, 1);
insert into Edzés values(10, 10);
end;
```

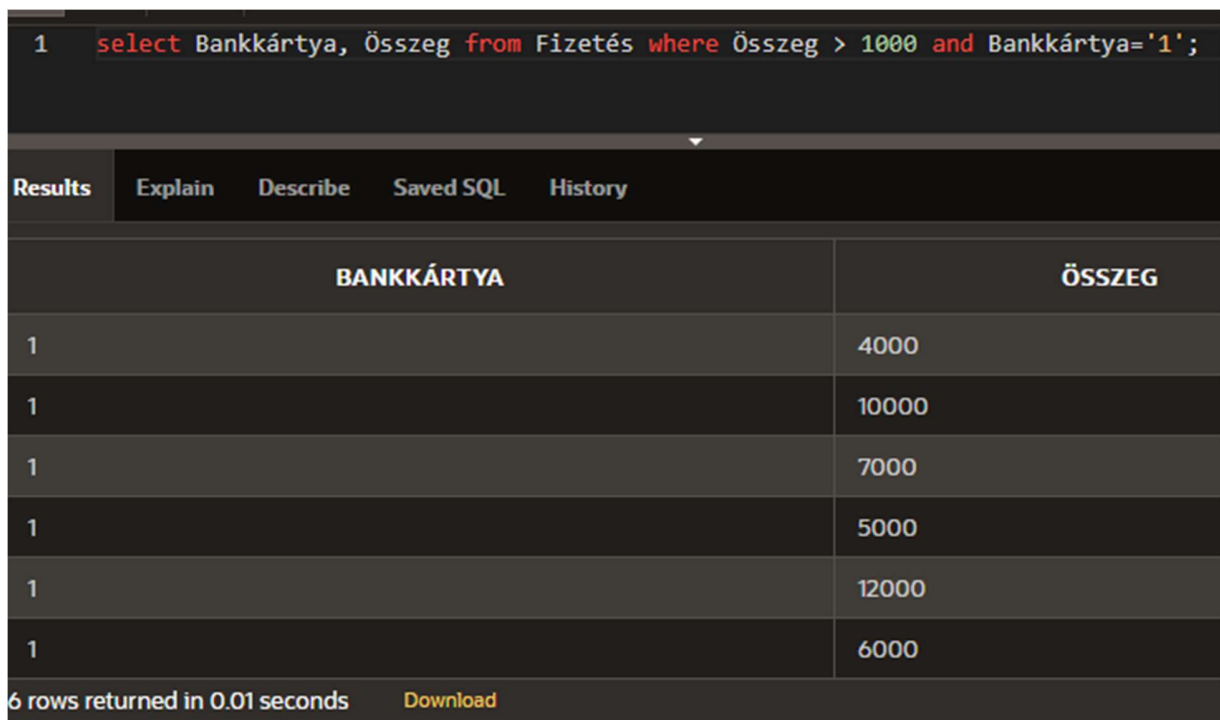
## Lekérdezések:

1. Egyszerű select: Azok összegek és fizetésmód számának kiírása, amelyek 1000-nél többet fizettek és bankkártyával.

`select Bankkártya, Összeg from Fizetés where Összeg > 1000 and Bankkártya='1';`

$\pi_{bankkártya, összeg}$

$(\sigma_{összeg > 1000 \text{ AND } bankkártya = "1"} \text{ fizetés})$



The screenshot shows a SQL query execution interface. At the top, a text area contains the query: `1 select Bankkártya, Összeg from Fizetés where Összeg > 1000 and Bankkártya='1';`. Below the query area, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected, displaying a table with two columns: 'BANKKÁRTYA' and 'ÖSSZEG'. The table contains six rows of data. At the bottom of the interface, it states '6 rows returned in 0.01 seconds' and provides a 'Download' link.

BANKKÁRTYA	ÖSSZEG
1	4000
1	10000
1	7000
1	5000
1	12000
1	6000

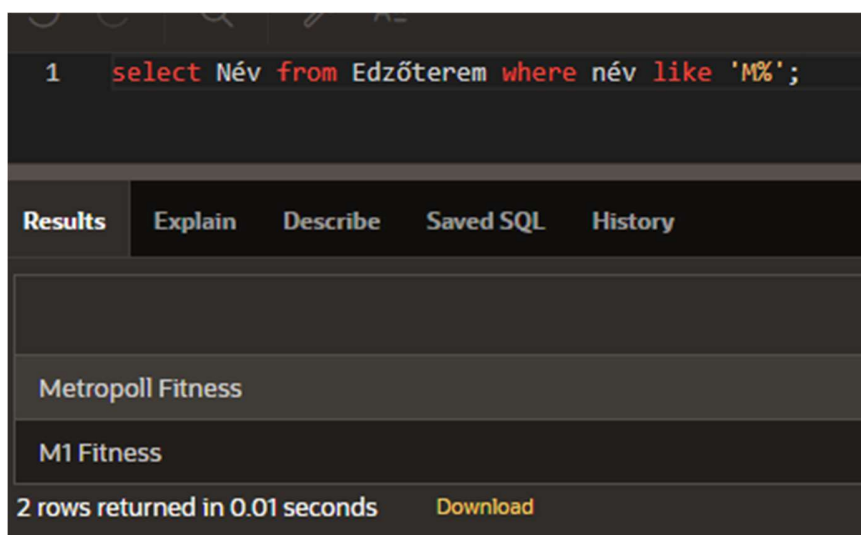
6 rows returned in 0.01 seconds [Download](#)

2. Egyszerű select: Azok az edzőtermek nevének kiírása, amik „M”-betűvel kezdődnek.

`select Név from Edzőterem where név like 'M%';`

$\pi_{név}$

$(\sigma_{név \text{ LIKE "M\%"}} edzőterem)$



```
1 select Név from Edzőterem where név like 'M%';
```

Results	Explain	Describe	Saved SQL	History
Metropoll Fitness				
M1 Fitness				
2 rows returned in 0.01 seconds <a href="#">Download</a>				

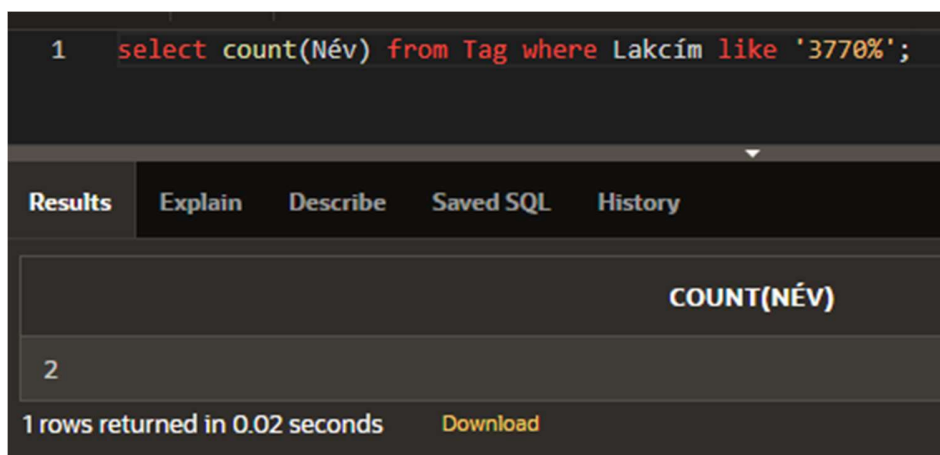
3. Összesítő művelet: Számolja meg azoknak a Tagoknak a számát, akik „3770” alatti lakcímen laknak.

`select count(Név) from Tag where Lakcím like '3770%';`

$\pi_{COUNT(név)}$

$\gamma_{COUNT(név)}$

$\sigma_{lalcím \text{ LIKE "3770\%"}} tag$



```
1 select count(Név) from Tag where Lakcím like '3770%';
```

Results	Explain	Describe	Saved SQL	History
COUNT(NÉV)				
2				
1 rows returned in 0.02 seconds <a href="#">Download</a>				

```
select avg(Összeg) from Fizetés where Készpénz=1;
```

 $\gamma_{AVG}$  (összeg)

```
1 select avg(Összeg) from Fizetés where Készpénz=1;
```

AVG(ÖSSZEG)
8166.666666666666666666666666666666666667

1 rows returned in 0.00 seconds [Download](#)

```
select Fizetés.Készpénz, Bérlet.Kezdet from Fizetés full outer join Bérlet on Fizetés.FKód =
Bérlet.BKód where Készpénz=1;
```

$$\sigma_{k\acute{e}szp\acute{e}nz = 1} (\text{f\acute{ı}zet\acute{e}s} \bowtie^o_{f\acute{ı}zet\acute{e}s} .fk\acute{o}d = b\acute{e}rlet .bk\acute{o}d \text{ } b\acute{e}rlet)$$

```
1 select Fizetés.Készpénz, Bérlet.Kezdet from Fizetés full outer join Bérlet on Fizetés.FKód = Bérlet.BKód where Készpénz=1;
```

Results Explain Describe Saved SQL History

KÉSZPÉNZ	KEZDET
1	01/14/2021
1	05/21/2020
1	05/09/2021

3 rows returned in 0.03 seconds [Download](#)

6. Inner join: A bérlet tulajdonosát köti össze a taggal.

`select Bérlet.BKód, Tag.TKód, Tag.Név from Bérlet inner join Tag on Bérlet.BKód = Tag.TKód;`

$\pi_{bérlet . bkód, tag . tkód, tag . név} (bérlet \bowtie_{bérlet . bkód = tag . tkód} tag)$

```
1 select Bérlet.BKód, Tag.TKód, Tag.Név from Bérlet inner join Tag on Bérlet.BKód = Tag.TKód;
```

Results	Explain	Describe	Saved SQL	History
BKÓD	TKÓD	NÉV		
1	1	Jakab József		
2	2	Horváth Cintia		
3	3	Madách Dániel		
4	4	Kiss Béla		
5	5	Nagy Marcell		
6	6	Smidt Gábor		
7	7	Balogh András		
8	8	Kovács Anna		
9	9	Halász Géza		
10	10	Lakatos Antal		

10 rows returned in 0.01 seconds [Download](#)

7. Rendezés select: Azoknak az edzőtermek nevének kiírása, amiknek a házszáma nagyobb mint 10 csökkenő sorrendben a házsám szerint

`select Név, Házsám from Edzőterem where Házsám < 10 order by Házsám desc;`

$\tau_{házsám \downarrow}$

$\pi_{név, házsám}$

$\sigma_{házsám < 10} edzőterem$

```
1 select Név, Házsám from Edzőterem where Házsám < 10 order by Házsám desc;
```

NÉV	HÁZSZÁM
Metropoli Fitness	5
Cutler Fitness	4
Not Jam Fitness	3
King Gym	3
M1 Fitness	1

5 rows returned in 0.01 seconds [Download](#)

8. Al-lekérdezés: Az átlagfizetéskor kártyával kevesebbet fizetők.

`select Bankkártya, Összeg from Fizetés where Összeg < (select avg(Összeg) from Fizetés) and Bankkártya=1;`

$\pi_{bankkártya, összeg}$

$\sigma_{összeg < \pi_{AVG(összeg)}}$

$\gamma_{AVG(összeg)} fizetés$

AND bankkártya = "1" *fizetés*

```
1 select Bankkártya, Összeg from Fizetés where Összeg < (select avg(Összeg) from Fizetés) and Bankkártya=1;
```

Results	Explain	Describe	Saved SQL	History
BANKKÁRTYA	ÖSSZEG			
1	4000			
1	5000			
1	1000			
1	6000			

4 rows returned in 0.00 seconds [Download](#)

9. Csoportképzés: Az adott napon a bankkártyás fizetések számát írja ki.

`select count(*) "Bankkártyás fizetés", Dátum from Fizetés where Bankkártya=1 group by Dátum;`

$\gamma_{dátum, COUNT(*) \rightarrow Bankkártyás\ Fizetés}$

$\sigma_{bankkártya = 1} fizetés$

```
1 select count(*) "Bankkártyás fizetés", Dátum from Fizetés where Bankkártya=1 group by Dátum;
```

Bankkártyás fizetés	DÁTUM
1	05/24/2021
1	10/17/2021
1	03/27/2020
1	03/07/2021
1	09/09/2020
1	09/06/2021
1	02/27/2021

7 rows returned in 0.01 seconds [Download](#)

10. Al-lekérdezés: A legalacsonyabb házszám fölötti edzőtermet kéri le.

select Név, Házszám from Edzőterem where Házszám = (select min(Házszám) from Edzőterem);

$\pi_{név, házszám}$

$\sigma_{házszám = " } \pi_{MIN(házszám)}$

$\gamma_{MIN(házszám) } edzoterem$

```
1 select Név, Házszám from Edzőterem where Házszám = (select min(Házszám) from Edzőterem);
```

NÉV	HÁZSZÁM
M1 Fitness	1

1 rows returned in 0.01 seconds [Download](#)

11. Származtatott tulajdonság: A Tagok Kora, amely származtatott tulajdonság.

select Név, floor((sysdate-Születési\_dátum)/365) as Kor from Tag;

$\pi_{név, floor((sysdate - születési\_dátum)/ 365) \rightarrow kor} tag$

```
1 select Név, floor((sysdate-Születési_dátum)/365) as Kor from Tag;
```

NÉV	KOR
Jakab József	24
Horváth Cintia	20
Madách Dániel	16
Kiss Béla	22
Nagy Marcell	20
Smidt Gábor	32
Balogh András	27
Kovács Anna	19
Halász Géza	16
Lakatos Antal	39

10 rows returned in 0.01 seconds [Download](#)