

# Jegyzőkönyv

## Adatbázis rendszerek I.

### Féléves feladat

**Készítette:** Barta Balázs

**Szak:** Programtervező  
Informatikus Bsc

**Neptunkód:** S90NXX

## A feladat leírása:

A feladatban egy elképzelt adatbázist hozok létre ami egy fiktív ország általános iskolának adatait tartalmazza, szimulált körülményekkel, az alábbi jellemzőkkel:

A **diák egyed** név és születési dátum tulajdonsága magától értetődő.

A kor egy születési dátumból számolt származtatott tulajdonság.

A diák megnyert versenyei egy többértékű tulajdonság, ami a megnyert versenyek nevét tartalmazza. (Egy diák értelemszerűen több versenyt is nyerhet, ezért többértékű.)

A D\_id egy sorszám, ami a diák egyértelmű azonosítására szolgál iskolán belül.

Egy diák egyszerre csak egy iskola jár be.

Egy diák több tárgyat is felvehet, és egy tárgyat is felvehet több diák.

Minden diáknak csak egy osztályfőnöke lehet.

Az **iskola egyed** tulajdonságai egy évszám és egy logikai tulajdonság, ami azt jelenti, hogy 8 vagy négy osztályos iskoláról van-e szó.

Az S\_id egy sorszám, ami az iskola egyértelmű azonosítására szolgál.

Egy iskolába több diák is járhat egyszerre, viszont feltételezzük, hogy tanárok is csak egy iskolában dolgoznak.

A **tárgy egyed** három tulajdonsággal rendelkezik:

O\_id, ami egy sorszám, ami a tárgy egyed kulcs tulajdonsága.

A név a tárgy nevét tartalmazza, míg a hossz az óra hosszát percben megadva.

Egy diák több tárgyat is felvehet, és egy tárgyat is felvehet több diák.

Feltételezzük, hogy egy tárgyat egyszerre csak egy tanár oktat.

A **tanár egyed** tulajdonságai a következők:

T\_id, ami a tanár egyed kulcs tulajdonsága (Szintén egy sorszám)

A tanár neve.

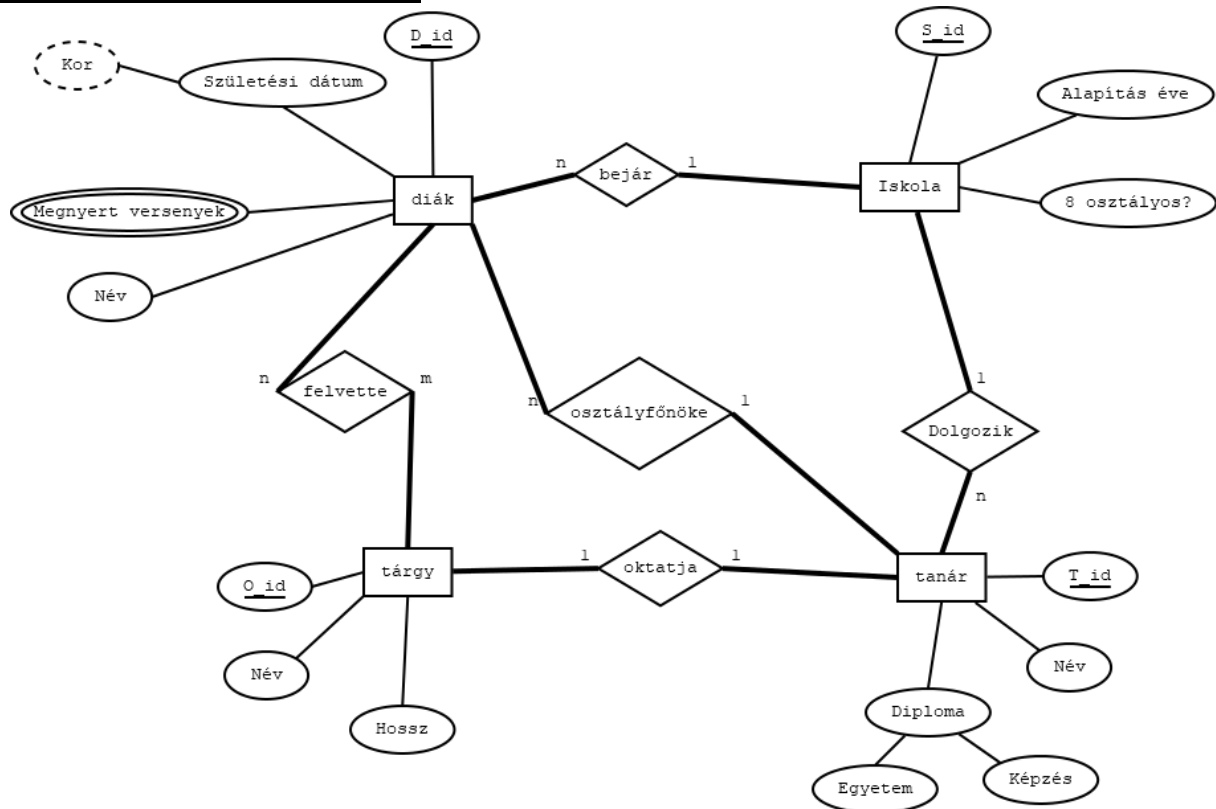
Valamint a tanár diplomája, ami egy összetett tulajdonság, ami felépül a következő tulajdonságokból: Melyik egyetemen szerezte, és milyen képzésen.

Feltételezzük, hogy egy tárgyat egyszerre csak egy tanár oktat.

Feltételezzük, hogy tanárok csak egy iskolában dolgoznak egyszerre.

Minden diáknak csak egy osztályfőnöke lehet.

## Az adatbázis ER modellje:



## Az adatbázis konvertálása relációs modellre:

A diák egyedből két tábla készül, hiszen a megnyert versenyek egy többértékű tulajdonság, valamint a kor származtatott tulajdonság kimarad, számítása a következő módon történik: jelenlegi dátum – születési dátum.

A maradék tulajdonságok (név, szüldat) bekerülnek a táblába.

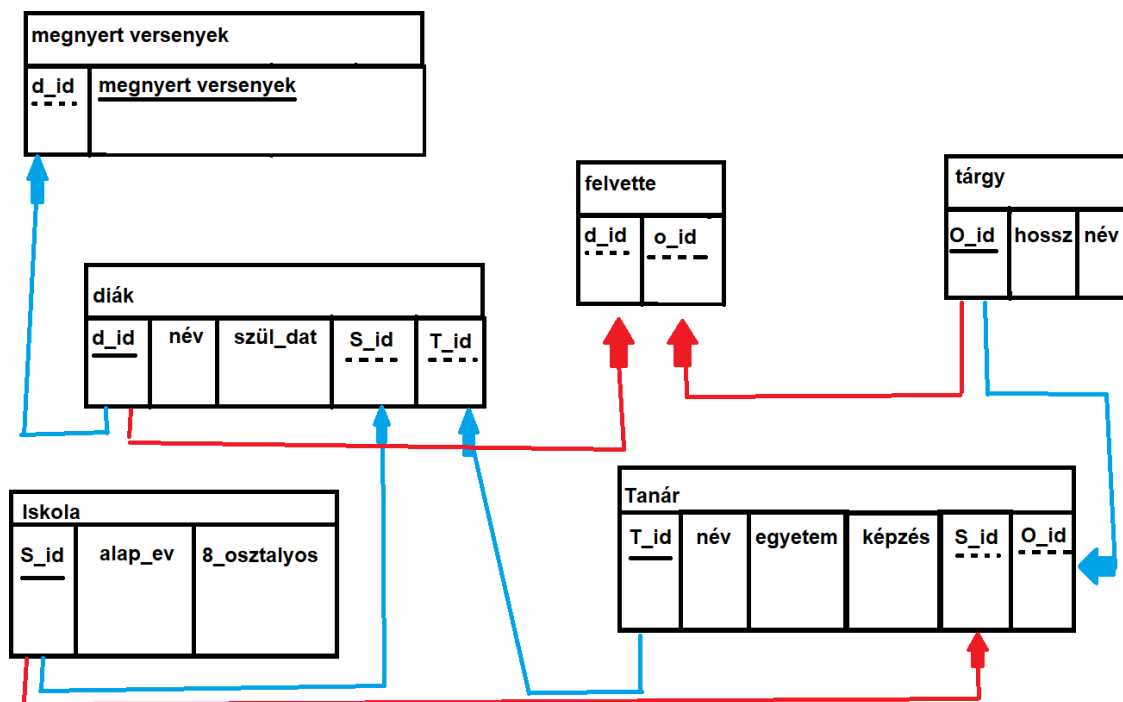
Kulcsmező továbbra is a D\_id, viszont kapcsolókulcsként ide kerül a T\_id és az S\_id kulcsok a Tanár és Iskola egyedekből.

A tárgy egyedből egy tábla készül, O\_id, név, hossz tulajdonságokkal

A tárgy és diák egyedekből közösen készülni fog egy kapcsolótábla, hiszen több a többhöz kapcsolat áll fent mögöttük.

Az iskola egyedből egy tábla készül, S\_id, alapítás éve, 8 osztályos adattagokkal.

A tanár egyedből egy tábla készül, viszont mivel a diploma összetett tulajdonság, annak altulajdonságai kerülnek bele majd a táblába. Kapcsolókulcsként megjelenik itt az O\_id és az S\_id



### Az adatbázis relációs sémái:

diák(d\_id, név, szül\_dat, S\_id, T\_id)

megnyert versenyek(d\_id, megnyert versenyek)

tárgy(o\_id, hossz, név)

felvette(d\_id, o\_id)

Iskola(S\_id, alap\_ev, 8\_osztalyos)

Tanár(T\_id, név, egyetem, képzés, S\_id, O\_id)

## A táblák létrehozása:

A létrehozásnál ügyelni kell a sorrendre, először azokat a táblákat kell létrehozni, amelyekben nincs idegen kulcs, és ezután azokat, amelyekben van, hiszen az idegen kulcsnak a már létrehozott táblára kell mutatnia. Az idegen kulcsot tartalmazó mezők típusának meg kell egyeznie a referenciaként szolgáló, másik táblában található kulcsmező típusával.

Amelyik tábla nem helyes adatszerkezettel kerül létrehozásra, később módosítva lesz (hisz a feladat előír módosító parancsokat).

```
create table Targy(  
    O_id int primary key,  
    hossz int,  
    nev varchar(20)  
);  
  
create table Iskola(  
    S_id int primary key,  
    alap_ev int,  
    nyosztalyos int  
);  
  
create table Tanar(  
    T_id int primary key,  
    nev varchar(30),  
    egyetem varchar(20),  
    kepzes varchar(20),  
    O_id int,  
    S_id int,  
    foreign key (O_id) references Targy(O_id),  
    foreign key (S_id) references Iskola(S_id)  
);  
  
create table diak(  
    D_id int primary key,  
    nev varchar(30),  
    szuldat date,  
    S_id int,  
    T_id int,  
    foreign key(S_id) references Iskola(S_id),  
    foreign key (T_id) references Tanar(T_id)  
);  
  
create table versenyek(  
    V_id int primary key,  
    V_nev varchar(30)  
);  
  
create table versenyek2(  
    D_id int,  
    V_id int,  
    primary key(D_id, V_id),  
    foreign key(D_id) references Diak(D_id),
```

```

    foreign key(V_id) references versenyek(V_id)
);

create Table felvette(
    D_id int,
    O_id int,
    primary key(D_id, O_id),
    foreign key (D_id) references Diak(D_id),
    foreign key (O_id) references Targy(O_id)
)

```

## A táblák módosítása:

1.)

```

alter table Iskola
add constraint igazhamis
check(ny_osztalyos = 1 or ny_osztalyos = 0)

```

2.)

```

alter table Tanar
add constraint egyatobbhoz
unique(O_id);

```

3.)

```

rename versenyek2 to megnyert_versenyek

```

4.)

```

alter table iskola add nev varchar(30);

```

(megjegyzés: az eredeti modellben nincs az iskoláknak neve, ezt most vettem észre, így javítom)

## A táblák feltöltése:

1.) Targy

```

begin
insert into Targy values(1, 45, 'matematika');
insert into Targy values(2, 45, 'irodalom');
insert into Targy values(3, 90, 'informatika');
insert into Targy values(4, 30, 'testnevelés');
insert into Targy values(5, 45, 'technika');
end

```

## 2.) Iskola

```
begin
insert into Iskola values(1, 1806, 1, 'Táncsics Mihály Gimnázium');
insert into Iskola values(2, 1963, 1, 'Építőipari Iskola');
insert into Iskola values(3, 2002, 0, 'Akadémia');
insert into Iskola values(4, 2004, 1, 'Teleki Pál Iskola');
insert into Iskola values(5, 1992, 0, 'Kaposszekcsői Ált. Iskola');
end
```

## 3.) Tanár

```
begin
insert into Tanar values(1, 'Kő Kata', 'ELTE', 'Bölcsész', 2 , 3);
insert into Tanar values(2, 'Pál Péter', 'ELTE', 'Testnevelő', 4 , 4);
insert into Tanar values(3, 'Wincz Eszter', 'SZTE', 'Informatikus', 3 , 3);
insert into Tanar values(4, 'Mekk Elek', 'PTE', 'Matematikus', 1 , 4);
insert into Tanar values(5, 'Pécs Antal', 'ME', 'Mérnök', 5 , 3);
end
```

## 4.) Diák

```
begin
insert into Diak values(1, 'Dékány Péter', '06/23/2009', 3, 5);
insert into Diak values(2, 'Kovács Antal', '02/17/2009', 3, 5);
insert into Diak values(3, 'Nádházy Gergely', '04/11/2010', 4, 4);
insert into Diak values(4, 'Bencze Zsombor', '06/01/2008', 4, 4);
insert into Diak values(5, null, '05/02/2001', 3, 1);
insert into Diak values(6, 'Tánczos Gergely', '09/02/2010', 3, 3);
insert into Diak values(7, 'Szakács Dániel', '12/11/2008', 4, 2);
insert into Diak values(8, 'Zilai Gergely', '12/28/2010', 4, 2);
insert into Diak values(9, 'Szalontai Panna', '01/05/2009', 3, 5);
insert into Diak values(10, 'Perényi Kitti', '07/11/2008', 4, 2);
insert into Diak values(11, 'Prof. Dr. Lenkei', '09/09/1999', 3, 5);
end
```

## 5.) Versenyek

```
begin
insert into Versenyek values(1, 'Ki mit tud?!');
insert into Versenyek values(2, 'Nyelv-ÉSZ');
insert into Versenyek values(3, 'Nemes Tihamér');
insert into Versenyek values(4, 'E-Hód');
insert into Versenyek values(5, 'Dusza Árpád');
end
```

## 5.) Megnyert versenyek

```
begin
insert into Megnyert_versenyek values(1, 1);
insert into Megnyert_versenyek values(1, 4);
insert into Megnyert_versenyek values(2, 3);
insert into Megnyert_versenyek values(8, 5);
insert into Megnyert_versenyek values(10, 1);
insert into Megnyert_versenyek values(1, 2);
end
```

## 6.) Felvette

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

## 5 db módosítás:

1.)

```
update Iskola set Nev = 'Selmecsi Akadémia'
where nev='Akadémia'
```

2.)

```
delete from Diak
where nev like 'Prof.%'
```

3.)

```
update Diak set Nev = 'Tóth Tóbiás'
where nev is null
```

4.)

```
update Diak set szuldat = '11/25/2010'
where nev like '%Gergely'
```

5.)

```
delete from Megnyert_versenyek
where D_id=8
```

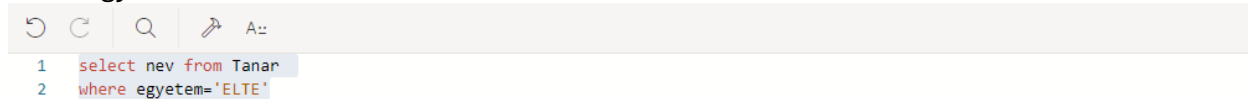


## Lekérdezések:

### 2 db egyszerű:

1-Melyik tanárok tanultak az ELTE-n?

```
select nev from Tanar  
where egyetem='ELTE'
```

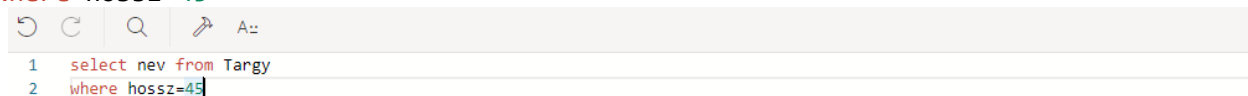


```
1 select nev from Tanar  
2 where egyetem='ELTE'
```

Results		Explain	Describe	Saved SQL	History
		NEV			
Kő Kata					
Pál Péter					
2 rows returned in 0.01 seconds		<a href="#">Download</a>			

2-Mely tárgyak tartanak 45 percig?

```
select nev from Targy  
where hossz=45
```



```
1 select nev from Targy  
2 where hossz=45
```

Results		Explain	Describe	Saved SQL	History
		NEV			
matematika					
irodalom					
technika					
3 rows returned in 0.02 seconds		<a href="#">Download</a>			

## 2db összesítő:

1-Hány darab gyereknek osztályfőnöke Péch Antal?

```
select count(*) from diak join tanar using(T_ID)
where Tanar.nev='Péch Antal'
```

```
1 select count(*) from diak join tanar using(T_ID)
2 where Tanar.nev='Péch Antal'
```

Results	Explain	Describe	Saved SQL	History
				COUNT(*)
				3
1 rows returned in 0.02 seconds				<a href="#">Download</a>

2 – Az összes óra hosszának az átlaga:

```
select avg(hossz) from Targy
```

```
1 select avg(hossz) from Targy
```

Results	Explain	Describe	Saved SQL	History
				AVG(HOSSZ)
				51
1 rows returned in 0.01 seconds				<a href="#">Download</a>

## 2 db inner join:

1 - Tánczos Gergely felvett tárgyainak a neve:

```
select Targy.nev from Felvette
inner join Diak using(D_id) inner join Targy using(O_ID)
where Diak.nev='Tánczos Gergely'
```

```
1 select Targy.nev from Felvette inner join Diak using(D_id) inner join Targy using(O_ID)
2 where Diak.nev='Tánczos Gergely'
```

Results		Explain	Describe	Saved SQL	History
					NEV
irodalom					
informatika					
technika					
3 rows returned in 0.01 seconds					<a href="#">Download</a>

2 – Kik vették fel a matematikát?

```
select Diak.nev from Felvette
inner join Diak using(D_id) inner join Targy using(O_ID)
where Targy.nev='matematika'
```

```
1 select Diak.nev from Felvette inner join Diak using(D_id) inner join Targy using(O_ID)
2 where Targy.nev='matematika'
```

<div>Results</div> <div>Explain</div> <div>Describe</div> <div>Saved SQL</div> <div>History</div>				
NEV				
Nádházy Gergely				
Bencze Zsombor				
Zilai Gergely				
Perényi Kitti				
4 rows returned in 0.01 seconds <a href="#">Download</a>				

## 2 db outer join:

1 – Melyik diák hány versenyt nyert meg?(Akkor is, ha 0 – csökkenő sorrendben)

```
select nev, count(V_id) from diak
left join megnyert_versenyekek using(D_id)
group by nev order by count(V_id) DESC
```

↶ ↷ 🔍 📌 A..
1 select nev, count(V_id) from diak
2 left join megnyert_versenyekek using(D_id)
3 group by nev order by count(V_id) DESC

Results	Explain	Describe	Saved SQL	History
NEV				
Dékány Péter		3		
Kovács Antal		1		
Perényi Kitti		1		
Szakács Dániel		0		
Nádházy Gergely		0		
Tánczos Gergely		0		
Tóth Tóbiás		0		
Szalontai Panna		0		
Bencze Zsombor		0		
Zilai Gergely		0		

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

2- Melyik tanárnak hány diákja van(növekvő sorrendben)?

```
select Tanar.nev, count(D_id) from diak
left join Tanar using(T_id)
group by Tanar.nev order by count(D_id)
```

↶ ↷ 🔍 📌 A..
1 select Tanar.nev, count(D_id) from diak
2 left join Tanar using(T_id)
3 group by Tanar.nev order by count(D_id)

Results	Explain	Describe	Saved SQL	History
NEV				
Kő Kata		1		
Wincs Eszter		1		
Mekk Elek		2		
Pécs Antal		3		
Pál Péter		3		

5 rows returned in 0.04 seconds [Download](#)

## 2 db group by:

1 – Melyik tárgyat hányan vették fel?

```
select Targy.nev, count(O_id) from Felvette
join Targy using(O_id)
group by Targy.nev
```

↶	↷	🔍	🔗	Az
1	select	Targy.nev,	count(O_id)	from Felvette
2	join	Targy	using(O_id)	
3	group by	Targy.nev		

Results	Explain	Describe	Saved SQL	History
NEV				
informatika		5		
testnevelés		5		
irodalom		2		
technika		3		
matematika		4		
5 rows returned in 0.01 seconds <a href="#">Download</a>				

2 – Hány tanító van az iskolában, ahol a legtöbbet tanítanak?

```
select max(count(S_id)) from Tanar
group by S_id
```

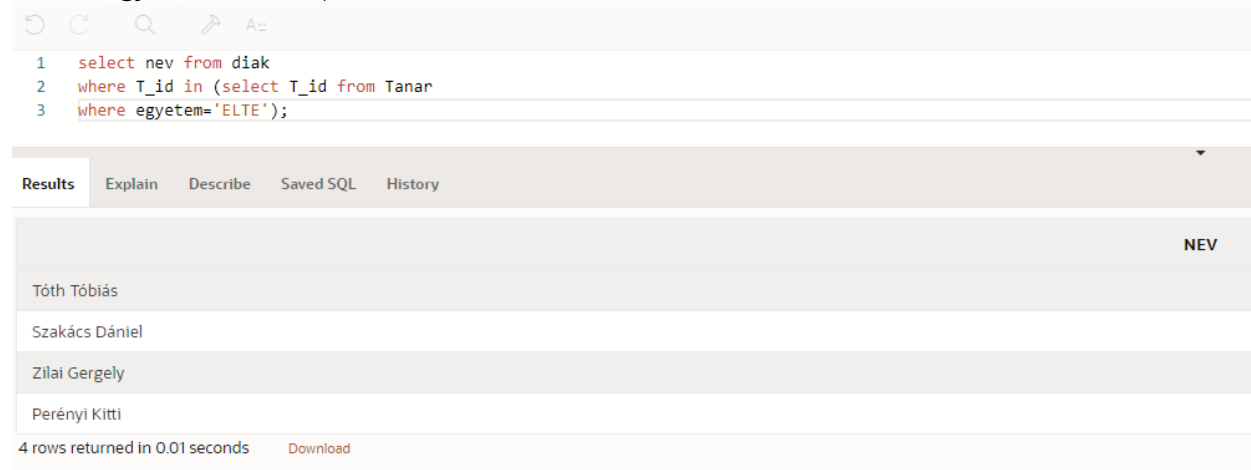
↶	↷	🔍	🔗	Az
1	select	max(count(S_id))	from Tanar	
2	group by	S_id		

Results	Explain	Describe	Saved SQL	History
MAX(COUNT(S_ID))				
3				
1 rows returned in 0.00 seconds <a href="#">Download</a>				

## 2db al-lekérdezést használó:

1 - ELTE-n tanult tanítók diákjainak a névsora:

```
select nev from diak
where T_id in
(select T_id from Tanar
where egyetem='ELTE');
```



The screenshot shows a SQL query editor with a toolbar containing icons for undo, redo, search, and a keyboard shortcut 'A:'. Below the toolbar, the SQL query is entered in a text area. The results section below the query shows a table with one column named 'NEV' and four rows of student names. At the bottom of the results section, it indicates '4 rows returned in 0.01 seconds' and provides a 'Download' link.

```
1 select nev from diak
2 where T_id in (select T_id from Tanar
3 where egyetem='ELTE');
```

NEV
Tóth Tóbiás
Szakács Dániel
Zilai Gergely
Perényi Kitty

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

2 – „Ki mit tud?!” versenyt megnyert hallgatók osztályfőnökei:

```
select Tanar.nev from Diak join Tanar using(T_id)
where D_id in
(select D_id from Megnyert_versenyek join Versenyek using(V_id)
where V_nev = 'Ki mit tud?!')
```



The screenshot shows a SQL query editor with a toolbar containing icons for undo, redo, search, and a keyboard shortcut 'A:'. Below the toolbar, the SQL query is entered in a text area. The results section below the query shows a table with one column named 'NEV' and two rows of teacher names. At the bottom of the results section, it indicates '2 rows returned in 0.03 seconds' and provides a 'Download' link.

```
1 select Tanar.nev from Diak join Tanar using(T_id)
2 where D_id in
3 (select D_id from Megnyert_versenyek join Versenyek using(V_id)
4 where V_nev = 'Ki mit tud?!')
```

NEV
Pécs Antal
Pál Péter

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

Melyik diák hány éves?

```
select nev, floor((sysdate-szüldat)/365) from diak
```

```
1 select nev, floor((sysdate-szüldat)/365) from diak
```

Results	Explain	Describe	Saved SQL	History
NEV		FLOOR((SYSDATE-SZÜLDAT)/365)		
Dékány Péter	11			
Kovács Antal	11			
Nádházy Gergely	10			
Bencze Zsombor	12			
Tóth Tóbiás	19			
Tánczos Gergely	10			
Szakács Dániel	11			
Zilai Gergely	10			
Szalontai Panna	11			
Perényi Kitti	12			
0 rows returned in 0.00 seconds		<a href="#">Download</a>		

(Megjegyzés: Igen, Tóbiás 2001-gyel lett beleírva a táblába véletlen..)

**1 db összetett:**

A Selmeci Akadémián tanuló, Péch Antal osztályfőnökü, informatikát hallgató tanulók megnyert versenyei:

```
select V_nev from Megnyert_versenyekek join versenyekek using(V_id)
where D_id in
(select D_id from Diak left join Iskola using(S_id) join Felvette using(D_ID)
where Iskola.nev='Selmeci Akadémia'
and T_id in
(select T_id from Tanar
where nev='Pécs Antal')
and O_id in
(select O_id from Targy
where nev='informatika'))
```

```
1 select V_nev from Megnyert_versenyek join versenyek using(V_id)
2 where D_id in
3 ((select D_id from Diak left join Iskola using(S_id) join Felvette using(D_ID)
4   where Iskola.nev='Selmeci Akadémia'
5   and T_id in
6     (select T_id from Tanar
7     where nev='Pécs Antal')
8   and O_id in
9     (select O_id from Targy
10    where nev='informatika'))
11
```

Results

Explain

Describe

Saved SQL

History

V_NEV
Ki mit tud?!
Nyelv-ÉSZ
Nemes Tihámér
E-Hód

4 rows returned in 0.22 seconds
[Download](#)