

Adatbázis rendszerek I.

Féléves feladat

Készítette: **Csalavári Kristóf**

Neptunkód: **CZTNL6**

Gyakorlatvezető : **Bednarik László**

A Feladat leírása:

A feladat adatbázis leírást ad egy kórház belső adatbázisáról, ahol számon vannak tartva az igazgatók, kórházak és a kórházak osztályai, orvosok, nővérek és betegek. Az adatbázisban megtalálható az adott kórház dolgozóinak adatai és kórházon belüli azonosítói. Az adatbázisban megtalálható még a betegek személyes információi is.

Az ER modell egyedei és tulajdonságai:

Egyedek: Igazgató, Kórház, Osztály, Orvos, Nővér, Beteg

- **Kórház egyed tulajdonságai:**
 - **KorhazID** : A kórház egyed elsődleges kulcsa.
 - **KorhazNev** : A kórház neve.
 - **Cím** : A kórház címe.
- **Igazgató egyed tulajdonságai:**
 - **IgazgatoID** : Az igazgató azonosítója
 - **Nev** : Az igazgató neve
 - **SzulDatum**: Az igazgató születési dátuma.
 - **Cím** : Az igazgató címe
- **Osztály egyed tulajdonságai:**
 - **OsztalyID** : Az osztály egyedi azonosítója

- **OsztalyNev** : Az adott osztaly neve
- **Orvos egyed tulajdonságai:**
 - **OrvosID** : Az orvos egyedi kórház specifikus azonosítója.
 - **OrvosNEV** : Az orvos neve.
 - **SzulDatum** : Az orvos születés dátuma.
- **Nővér egyed tulajdonságai:**
 - **NoverID** : A nővér egyedi kórház specifikus azonosítója.
 - **NoverNEV** : A nővér neve.
- **Beteg egyed tulajdonságai:**
 - **BetegID**: A beteg saját kórházon belüli azonosítója amit akkor kap meg amikor becheckel a kórházba.
 - **BetegNEV**: A beteg teljes neve.
 - **BetegSzulDatum**: A beteg születési dátuma.

Az egyedek közötti kapcsolatok:

Az **Igazgató** és **Kórház** egyed között egy az egyhez kapcsolat van , ugyanis egy kórháznak egy igazgatója van és egy ember csak egy kórháznak az igazgatója.

A **Kórház** és **Osztály** egyed között egy a többhöz kapcsolat van mert egy kórháznak több osztálya is van de egy osztálynak csak egy kórháza lehet.

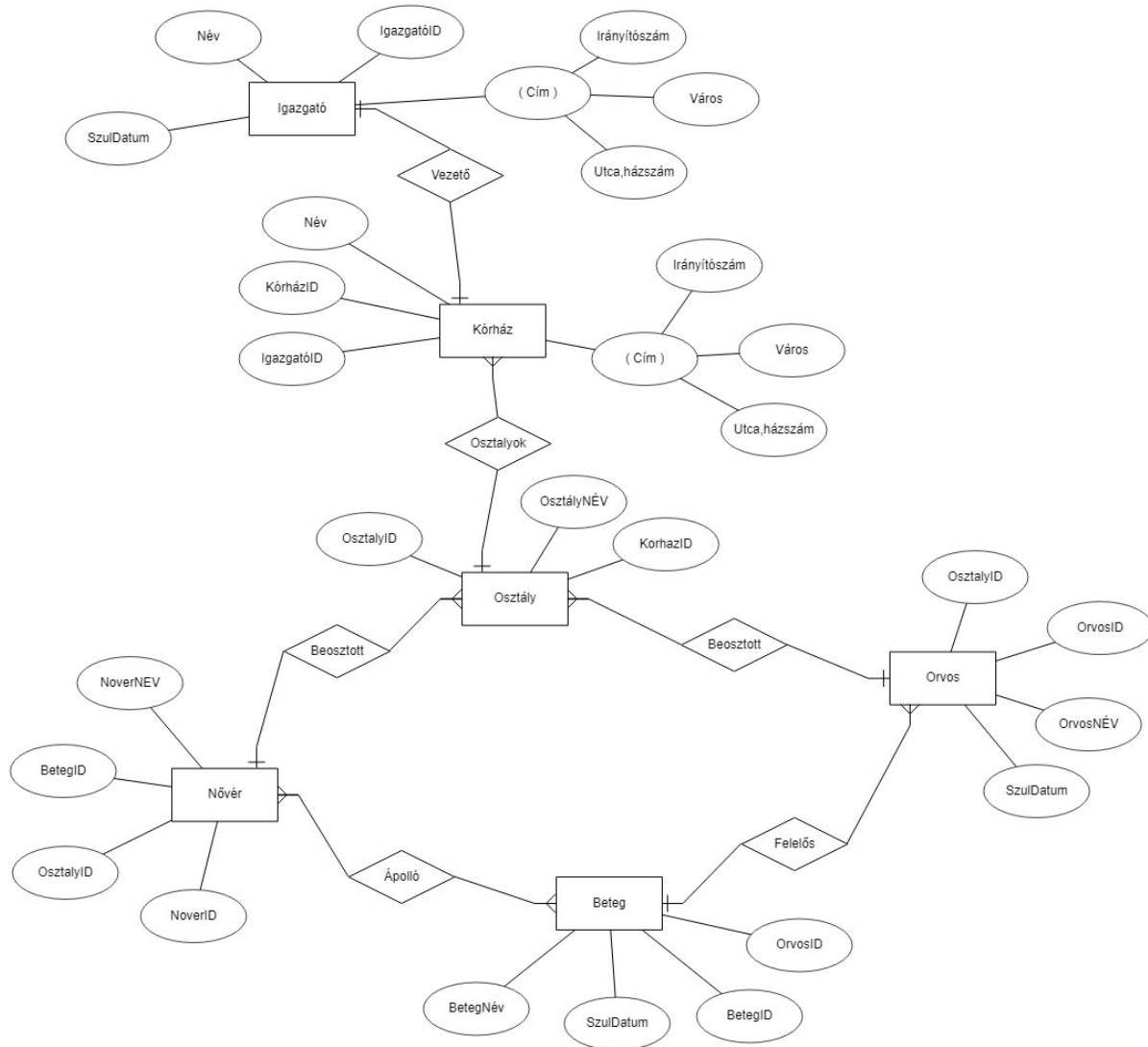
Az **Osztály** és **Orvos** egyed között is egy a többhöz kapcsolat van jelen, ugyanis egy osztálynak több orvosa is lehet egyszerre de egy orvos csak egy osztályhoz lehet kirendelve.

Az **Orvos** és **Beteg** egyed között is egy a többhöz kapcsolat van jelen.

A **Nővér** és **Beteg** egyed között több a többhöz kapcsolat van mert egy beteghez több nővér is ki van rendelve és egy nővérhez több beteg is ki van osztva.

Az **Osztály** és **Nővér** között egy a többhöz kapcsolat van.

Az adatbázis ER modellje:



Az adatbázis relációs modellje:

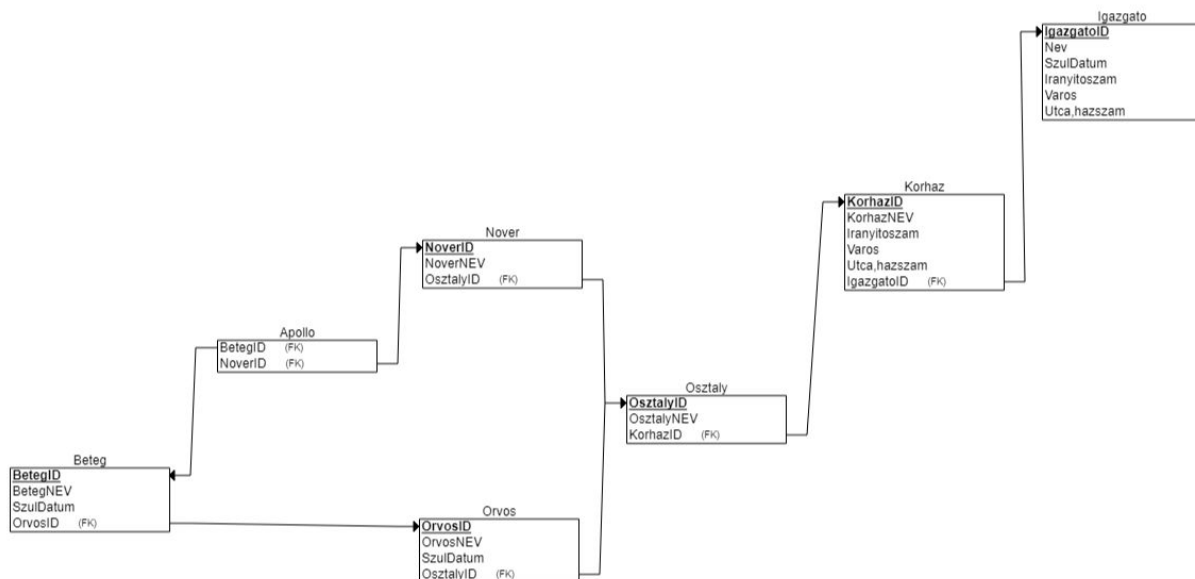
Az **Igazgató** egyedből tábla lesz, tulajdonságaiból adódnak a mezők. Az **IgazgatóID** adja az elsődleges kulcsot.

A **Kórház** egyedből reláció lesz, tulajdonságaiból a mezők. A **KórházID** fogja adni az elsődleges kulcsot.

Az **Osztály** egyedből is reláció lesz , a tulajdonságaiból mezők.Az **OsztályID** fogja adni az elsődleges kulcsot.

Az **Orvos**,**Beteg** és **Nővér** egyedből is külön-külön relációk lesznek amelyek tulajdonságaiból adódnak majd a mezők és mindegyiknek a saját ID-je fogja adni az elsődleges mezőt.

A **Nővér** és **Beteg** egy **Ápoló** nevű táblával kapcsolódnak melynek mezői két idegen kulcs lesz . **BetegID** és **NővérID** lesznek ezek a kulcsok.



A táblák létrehozása:

```
CREATE TABLE IGAZGATO(  
    IgazgatoID int(5) PRIMARY KEY,  
    Nev char(20),  
    SzulDatum int(10),  
    Iranyitoszam int(5),  
    Varos char(10),  
    Utca char(30),
```

```
Hazszam int(2)
);
CREATE TABLE KORHAZ(
    KorhazID int(5) PRIMARY KEY,
    KorhazNev char(30),
    Iranyitoszam int(4),
    Varos char(10),
    Utca char(30),
    Hazszam int(2),
    IgazgatoID int(5),
    FOREIGN KEY(IgazgatoID) REFERENCES IGAZGATO(IgazgatoID)
);
```

```
CREATE TABLE OSZTALY(
    OszталyID int(5) PRIMARY KEY,
    OszталyNev char(20),
    KorhazID int(5), FOREIGN KEY(KorhazID) REFERENCES KORHAZ(KorhazID)
);
```

```
CREATE TABLE ORVOS(
    OrvosID int(5) PRIMARY KEY,
    OrvosNev char(20),
    SzulDatum int(10),
    OszталyID int (5) , FOREIGN KEY (OszталyID) REFERENCES
OSZTALY(OszталyID)
);
```

```
CREATE TABLE NOVER(
    NoverID int(5) PRIMARY KEY,
    NoverNev char(20),
```

```

        OsztalyID int(5), FOREIGN KEY (OsztalyID) REFERENCES
OSZTALY(OsztalyID)
);

CREATE TABLE BETEG(
    BetegID int(5) PRIMARY KEY,
    BetegNev char(20),
    BetegNEM char(1),
    SzulDatum int(10),
    Betegseg char(20),
    OrvosID int(5),FOREIGN KEY (OrvosID) REFERENCES ORVOS(OrvosID));

CREATE TABLE APOLO(
    BetegID int(5),FOREIGN KEY (BetegID) REFERENCES BETEG(BetegID),
    NoverID int(5),FOREIGN KEY(NoverID) REFERENCES NOVER(NoverID));

```

A táblák feltöltése:

```

INSERT INTO IGAZGATO VALUES(1,"Kiss Péter",19990905,3900,"Szerencs","Petőfi
Sándor ut",10);
INSERT INTO IGAZGATO VALUES(2,"Nagy Ádám",19890605,3500,"Miskolc","Fürt
Sándor utca",5);

INSERT INTO KORHAZ VALUES(10,"Borsod Megyei Korhaz",3500,"Miskolc","Kossuth
utca",11,1);
INSERT INTO KORHAZ VALUES(20,"Simmelweis",3529,"Miskolc","Csabai
kapu",11,2);

INSERT INTO OSZTALY VALUES(11,"Sebészet",10);
INSERT INTO OSZTALY VALUES(12,"Belgyógyászat",10);
INSERT INTO OSZTALY VALUES(13,"Fül-Orr-Gége",10);

```

INSERT INTO OSZTALY VALUES(21,"Sebészet",20);
INSERT INTO OSZTALY VALUES(22,"Belgyógyászat",20);
INSERT INTO OSZTALY VALUES(23,"Fül-Orr-Gége",20);
INSERT INTO OSZTALY VALUES(24,"Ideggyógyászat",20);

INSERT INTO ORVOS VALUES(1,"Kelemen Richárd",19950810,11);
INSERT INTO ORVOS VALUES(2,"Deák Benjamin",19930121,11);
INSERT INTO ORVOS VALUES(3,"Major Ármin",19940211,12);
INSERT INTO ORVOS VALUES(4,"Kende Csanád",1991046,13);

INSERT INTO ORVOS VALUES(5,"Pataki Gergely",19950810,21);
INSERT INTO ORVOS VALUES(6,"Vászoly Áron",19930121,22);
INSERT INTO ORVOS VALUES(7,"Csonka Renátó",19940211,23);
INSERT INTO ORVOS VALUES(8,"Balog Zsolt",1991046,24);

INSERT INTO NOVER VALUES(1,"Dobos Daniella",11);
INSERT INTO NOVER VALUES(2,"Biró Zsanett",11);
INSERT INTO NOVER VALUES(3,"Kelemen Borbála",12);
INSERT INTO NOVER VALUES(4,"Miksa Regina",13);
INSERT INTO NOVER VALUES(5,"Simon Lili",21);
INSERT INTO NOVER VALUES(6,"Kis Vanessza",21);
INSERT INTO NOVER VALUES(7,"Szücs Tünde",22);
INSERT INTO NOVER VALUES(8,"Balázs Flóra",23);
INSERT INTO NOVER VALUES(9,"Király Laura",24);
INSERT INTO NOVER VALUES(10,"Katona Viktória",24);

INSERT INTO BETEG VALUES(1,"Márton
Alexander","M",19910108,"Ideggyógyászat",8);
INSERT INTO BETEG VALUES(2,"Jónás Gergő","M",19990305,"Fül-Orr-Gége",7);
INSERT INTO BETEG VALUES(3,"Szabó Imre","M",20010105,"Sebészet",1);

```
INSERT INTO BETEG VALUES(4,"Szilágyi Gábor","M",20080501,"Belgyógyászat",3);
INSERT INTO BETEG VALUES(5,"Csonka Ábel","M",20100911,"Fül-Orr-Gége",4);
INSERT INTO BETEG VALUES(6,"Lukács Rebeka","F",19851102,"Sebészet",2);
INSERT INTO BETEG VALUES(7,"Illés Adrienn","F",19940901,"Belgyógyászat",6);
INSERT INTO BETEG VALUES(8,"Pintér Liliána","F",19910201,"Fül-Orr-Gége",4);
INSERT INTO BETEG VALUES(9,"Molnár Beatrix","F",20030404,"Sebészet",5);
INSERT INTO BETEG VALUES(10,"Jónás Regina","F",20040109,"Belgyógyászat",6);
```

```
INSERT INTO APOLO VALUES(1,10);
INSERT INTO APOLO VALUES(2,9);
INSERT INTO APOLO VALUES(3,8);
INSERT INTO APOLO VALUES(4,7);
INSERT INTO APOLO VALUES(5,6);
INSERT INTO APOLO VALUES(6,5);
INSERT INTO APOLO VALUES(7,4);
INSERT INTO APOLO VALUES(8,3);
INSERT INTO APOLO VALUES(9,2);
INSERT INTO APOLO VALUES(10,10);
```

SQL Lekérdezése:

Sebészetre beosztott nővérek a semmelweis kórházban:

```
SELECT NoverNev,KorhazNev FROM NOVER, OSZTALY, KORHAZ WHERE
NOVER.OsztalyID = OSZTALY.OsztalyID AND KORHAZ.KorhazID = osztaly.KorhazID
AND OSZTALY.OsztalyNev = "Sebészet" AND KORHAZ.Iranyitoszam = 3529
```

Rendezzük az orvosokat betegszám alapján:

```
SELECT * FROM (SELECT COUNT(BETEG.OrvosID) CNT,OrvosNev FROM
ORVOS,BETEG WHERE ORVOS.OrvosID = BETEG.OrvosID
```


GROUP BY (BETEG.OrvosID)
ORDER BY COUNT(BETEG.OrvosID)desc) ADAT

Az összes női beteg:

SELECT BetegNev FROM BETEG WHERE BETEG.BetegNEM LIKE "F"

Az összes fül-orr-gége beteg:

SELECT BetegNev FROM BETEG WHERE BETEG.Betegseg LIKE "Fül-Orr-Gége"

Az összes női belgyógyászati beteg sebészetre való átrakása:

SELECT * FROM BETEG;
UPDATE BETEG SET BETEG.Betegseg = "Sebészet" WHERE BETEG.BetegNEM =
"F" AND BETEG.Betegseg = "Belgyógyászat";
SELECT * FROM BETEG;

Melyik beteget melyik nővér ápolja majd rendezve lett nővér alapján:

SELECT BetegNev,NoverNev FROM BETEG,NOVER,APOLO
WHERE BETEG.BetegID = APOLO.BetegID AND NOVER.NoverID = APOLO.NoverID
ORDER BY NoverNev

Melyik beteghez nem tartozik ápoló:

SELECT BETEGNEV FROM BETEG
WHERE BETEGID NOT IN (SELECT BETEGID FROM APOLO)