Adatbázis – kezelés féléves feladat

Főző Árpád – FKAVXF

# Környezet:

Egy ország hadseregeinek kezdetleges adatbázisát szeretném elkészíteni. Az adatbázisban eltárolásra kerülnek a raktárak, ahonnan az ellátás érkezik, a hadseregek, a tábornokok/vezetők illetve a szóban forgó hadseregben szolgáló hősök. A hadseregekhez az ellátmány, raktárakból fog érkezni. Ehhez meg kell határozni, melyik sereghez, mit és mennyit visznek. Minden sereghez, tartozik pontosan egy tábornok, hősöknél nincs ilyen kikötés.

Raktár: A seregben használatos eszközök itt vannak eltárolva. Minden raktárnak van egy egyedi azonosítója és helye, emellett minden eszköz neve és mennyisége is itt található.

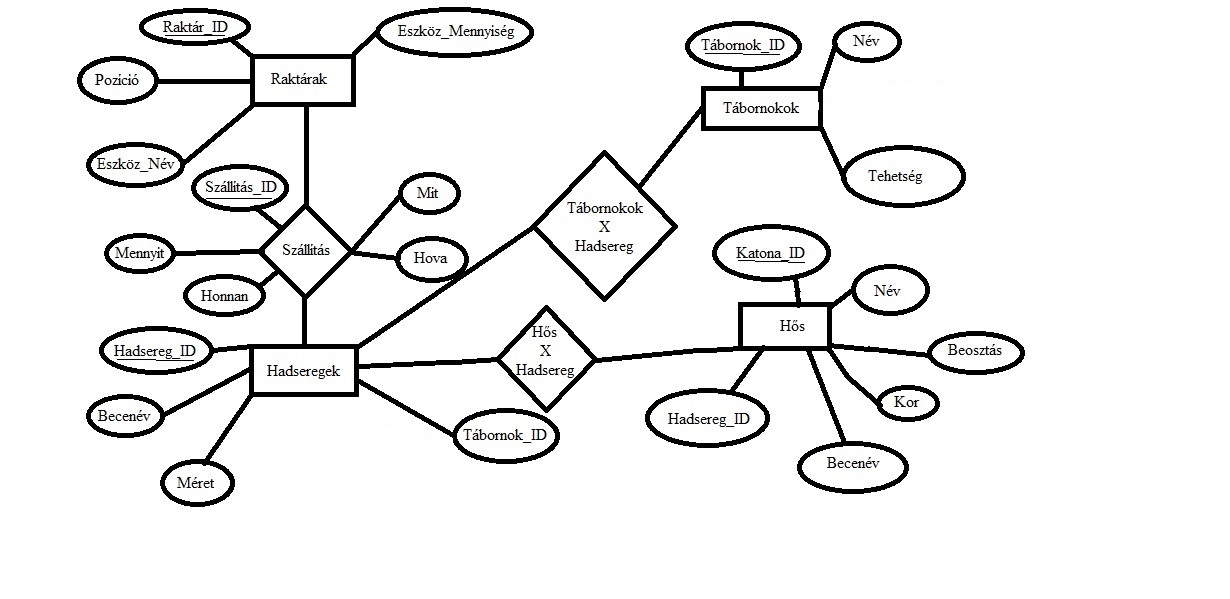
Hadseregek: Minden seregnek van egy egyedi azonosítója, beceneve. Itt látjuk még a sereg méretét, a felszerelések nevét és számát, plusz a sereghez tartozó tábornok azonosítóját.

Tábornokok: A tábornokok azonosítója, neve és a tehetségének szubjektív számértéke ( 1 – 10 ) kerül itt eltárolásra.

Hősök: A hősök neve, azonosítója, kora, beosztása, beceneve és a sereg azonosítója, amelyben szolgálatot végeznek, található itt.

Szállírás: Tartalmazza a szállítás azonosítóját, illetve, hogy honnan, hova visznek, mit és mennyit.

# E/K Modell



Raktárak(Raktár\_ID, Pozíció, Eszköz\_Név, Eszköz\_Mennyiség)

Szállítás(Szállítás\_ID, Mit, Honnan, Hova, Mennyit)

Honnan (FK): Idegen kulcs, hivatkozik a Raktárak táblában, Raktár\_ID elsődleges kulcsra

Hova (FK): Idegen kulcs, hivatkozik a Hadseregek táblában, Hadsereg\_ID elsődleges kulcsra

Hadseregek(Hadsereg\_ID, Tábornok\_ID, Becenév, Méret)

Tábornok\_ID (FK): Idegen kulcs, hivatkozik a Tábornokok táblában, Tábornok\_ID elsődleges kulcsra (PK)

Tábornokok(Tábornok\_ID, Név, Tehetség)

Hadsereg\_ID (FK): Idegen kulcs, hivatkozik a Hadseregek táblában, Hadsereg\_ID elsődleges kulcsra (PK)

Hős(Katona\_ID, Hadsereg\_ID, Név, Becenév, Beosztás, Kor)

Hadsereg\_ID (FK): Idegen kulcs, hivatkozik a Hadseregek táblában, Hadsereg\_ID elsődleges kulcsra (PK)

# Normalizálás

Raktárak(Raktár\_ID, Pozíció, Eszköz\_Név ,Eszköz\_ Mennyiség)

1NF: teljesül, nincs benne összetett attribútum

2NF: teljesül, a kulcshalmaz egy elemű

3NF: teljesül, csak kulcsfüggőség van

BCNF: teljesül

Szállítás(Szállítás\_ID, Mit, Honnan, Hova, Mennyit)

1NF: teljesül, nincs benne összetett attribútum

2NF: teljesül, a kulcshalmaz egy elemű

3NF: teljesül, csak kulcsfüggőség van

BCNF: teljesül

Hadseregek(Hadsereg\_ID, Tábornok\_ID, Becenév, Méret)

1NF: teljesül, nincs benne összetett attribútum

2NF: teljesül, a kulcshalmaz egy elemű

3NF: teljesül, csak kulcsfüggőség van

BCNF: teljesül

Tábornokok(Tábornok\_ID, Név, Tehetség)

1NF: teljesül, nincs benne összetett attribútum

2NF: teljesül, a kulcshalmaz egy elemű

3NF: teljesül, csak kulcsfüggőség van

BCNF: teljesül

Hős(Katona\_ID, Hadsereg\_ID, Név, Becenév, Beosztás, Kor)

1NF: teljesül, nincs benne összetett attribútum

2NF: teljesül, a kulcshalmaz egy elemű

3NF: teljesül, csak kulcsfüggőség van

BCNF: teljesül

# Táblák létrehozása és megszorítások

create table Raktárak

(

Raktár\_ID number(10) not null,

Pozíció varchar(20),

Eszköz\_Név varchar(20),

Eszköz\_Mennyiség number(10),

constraint r\_pk primary key(Raktár\_ID)

);

create table Tábornokok

(

Tábornok\_ID number(10) not null,

Név varchar(20),

Tehetség number(2),

constraint t\_pk primary key(Tábornok\_ID)

);

create table Hadseregek

(

Hadsereg\_ID number(10) not null,

Tábornok\_ID number(10) not null,

Becenév varchar(20),

Méret number(10),

constraint h\_pk primary key(Hadsereg\_ID),

constraint h\_fk foreign key (Tábornok\_ID) references Tábornokok(Tábornok\_ID)

);

create table Szállítás

(

Szállítás\_ID number(10) not null,

Honnan number(10) not null,

Hova number(10) not null,

Mennyit number(10),

Mit varchar(20),

constraint sz\_pk primary key(Szállítás\_ID),

constraint sz\_fk1 foreign key(Honnan) references Raktárak(Raktár\_ID),

constraint sz\_fk2 foreign key(Hova) references Hadseregek(Hadsereg\_ID)

);

create table Hős

(

Katona\_ID number(10) not null,

Hadsereg\_ID number(10) not null,

Név varchar(20),

Becenév varchar(20),

Beosztás varchar(20),

Kor number(2),

constraint ho\_pk primary key(Katona\_ID),

constraint ho\_fk foreign key(Hadsereg\_ID) references Hadseregek(Hadsereg\_ID)

);

# Táblák feltöltése

**insert into Raktárak**

values (000001, 'Győr', 'Turán I', '10000');

insert into Raktárak

values (000002, 'Győr', 'Turán II', '6500');

insert into Raktárak

values (000003, 'Budapest', 'Turán III', '100');

insert into Raktárak

values (000004, 'Budapest', 'Danuvia 43M', '1250000');

insert into Raktárak

values (000005, 'Budapest', 'Tas', '980');

insert into Raktárak

values (000006, 'Budapest', 'Panzer IV', '4500');

insert into Raktárak

values (000007, 'Debrecen', 'Csepel bicikli', '25000');

insert into Raktárak

values (000008, 'Debrecen', 'Lapát', '2000000');

insert into Raktárak

values (000009, 'Debrecen', 'Sisak', '1550000');

insert into Raktárak

values (000010, 'Miskolc', 'Torta', '60000');

insert into Raktárak

values (000011, 'Miskolc', 'Bakancs', '1550000');

insert into Raktárak

values (000012, 'Miskolc', 'Botond', '100000');

insert into Raktárak

values (000013, 'Miskolc', 'MÁVAG Héja', '2000');

insert into Raktárak

values (000014, 'Miskolc', 'MÁVAG BF109', '200');

insert into Raktárak

values (000015, 'Miskolc', 'Panzer VI', '3');

**insert into Tábornokok**

values (00100, 'Lakatos Miklós', 5);

insert into Tábornokok

values (00200, 'Kiss Jenő', 7);

insert into Tábornokok

values (00300, 'Kipi Pál', 1);

insert into Tábornokok

values (00400, 'Bödön Ödön', 10);

insert into Tábornokok

values (00500, 'Rontó Rommel', 8);

insert into Tábornokok

values (00600, 'Vas András', 4);

insert into Tábornokok

values (00700, 'Illés Gergely', 9);

insert into Tábornokok

values (00800, 'Bem József', 7);

insert into Tábornokok

values (00900, 'Vitéz Anna', 8);

insert into Tábornokok

values (00110, 'Mérnök Arnold', 3);

insert into Tábornokok

values (00120, 'Depi Dolorez', 2);

insert into Tábornokok

values (00130, 'Sajtos Stangli', 10);

insert into Tábornokok

values (00140, 'Topalidisz Károly', 5);

insert into Tábornokok

values (00150, 'Király Károly', 10);

insert into Tábornokok

values (00160, 'Huszár Áron', 4);

**insert into Hadseregek**

values(10000,00160,'Huszárok',85000);

insert into Hadseregek

values(20000,00100,'Leventék',40000);

insert into Hadseregek

values(30000,00120,'Alföld Betyárai',160000);

insert into Hadseregek

values(40000,00300,'Fekete Sereg',85000);

insert into Hadseregek

values(50000,00400,'Trabant Alakulat',182000);

insert into Hadseregek

values(60000,00500,'Görgey Gránátosok',12350);

insert into Hadseregek

values(70000,00200,'Pogányok',250000);

insert into Hadseregek

values(80000,00600,'Medvék',98000);

insert into Hadseregek

values(90000,00700,'Farkasok',155000);

insert into Hadseregek

values(11000,00150,'Poloskák',85000000);

**insert into Hős**

values(6254683,10000,'Kiss Ernő','Vakond','Géppuskás',32);

insert into Hős

values(8766331,10000,'Zimány Bálint','Sólyom','Vadászpilóta',24);

insert into Hős

values(1269875,60000,'Szaracén Dávid','Piros Arany','Lángszórós',36);

insert into Hős

values(9856150,10000,'Bakai Zoltán','Albatrosz','Vadászpilóta',19);

insert into Hős

values(1658796,11000,'Mortár Mária','Füstölgő Rózsa','Mesterlövész',29);

insert into Hős

values(5648987,20000,'Orosz Franciska','Bunkerbontó','Tiszt',32);

insert into Hős

values(1232468,10000,'Romhányi József','Napóleon','Tüzér',32);

insert into Hős

values(4887632,30000,'Nagy Nándor','Vipera','Géppuskás',30);

insert into Hős

values(6546879,30000,'Nagy Sándor','Kígyó','Géppuskás',29);

insert into Hős

values(8979844,30000,'Nagy Károly','Aligátor','Mérnök',26);

insert into Hős

values(7894652,50000,'Varkaly Ármin','Süllő','Kommandós',32);

insert into Hős

values(6451648,50000,'Kapai József','Rafkós Róka','Pancélosparancsnok',41);

**insert into Szállítás**

values(00001,000011,30000, 500,'Bakancs');

insert into Szállítás

values(00002,000006,10000, 600,'Panzer IV');

insert into Szállítás

values(00003,000010,30000, 50,'Torta');

insert into Szállítás

values(00004,000009,11000, 1500,'Sisak');

insert into Szállítás

values(00005,000001,60000, 200,'Turán I');

insert into Szállítás

values(00006,000001,60000, 1500,'Turán I');

insert into Szállítás

values(00007,000011,30000, 500,'Bakancs');

insert into Szállítás

values(00008,000015,70000, 1,'Panzer IV');

insert into Szállítás

values(00009,000008,90000, 50000,'Lapát');

insert into Szállítás

values(00010,000008,30000, 40000,'Lapát');

insert into Szállítás

values(00011,000008,10000, 125000,'Lapát');

select \* from Raktárak;

select \* from Hadseregek;

select \* from Hős;

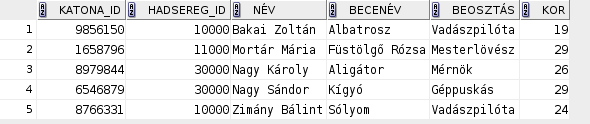
select \* from Tábornokok;

select \* from Szállítás;

# Egyszerű lekérdezések

**Listázzuk ki azokat a hősöket, akik 30 éves kor alatt vannak abc sorrendben.**

select \* from Hős where kor < 30 order by név;



**Mennyi torta van szállítás alatt?**

select mennyit from szállítás where mit like 'Torta';



**Hány fajta Turán tank van raktáron?**

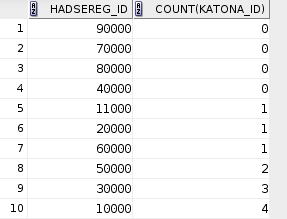
select count(eszköz\_név) "Turán tankok" from raktárak where eszköz\_név like '%Turán%';



# Csoportosító lekérdezések

**Listázzuk ki, a hadseregek azonosítóját, növekvő sorrendben a bennük szolgáló hősök alapján.**

select ha.hadsereg\_id,count(katona\_id) from hadseregek ha full join hős hő on ha.hadsereg\_id = hő.hadsereg\_id group by ha.hadsereg\_id order by count(katona\_id) asc;



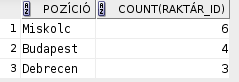
**Listázzuk ki azoknak a hadseregeknek a becenevét ahová legalább 3 féle hadianyagot szállítanak.**

select count(sz.szállítás\_id),h.becenév from szállítás sz full join hadseregek h on h.hadsereg\_id = sz.hova group by h.becenév having count(szállítás\_id) >= 3;



**Listázzuk ki azokat a városokat ahol legalább 3 raktár található.**

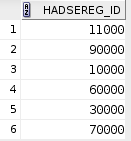
select pozíció,count(raktár\_id) from raktárak group by pozíció having count(raktár\_id) >= 3;



# Többtáblás lekérdezések

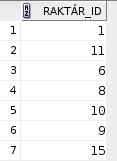
**Melyek azok a hadseregek, ahova szállítmány érkezik?**

select hadsereg\_id from hadseregek h inner join szállítás sz on h.hadsereg\_id = sz.hova group by hadsereg\_id;



**Melyek, azok a raktárak ahonnan éppen szállítanak?**

select raktár\_id from raktárak r inner join szállítás sz on r.raktár\_id = sz.honnan group by raktár\_id;



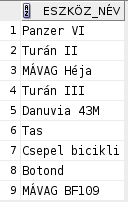
**Melyek, azok az eszközök, amelyeket éppen szállítanak a hadseregekhez, de még maradt belőlük raktáron?**

select eszköz\_név from raktárak r inner join szállítás sz on r.eszköz\_név = sz.mit group by eszköz\_név;



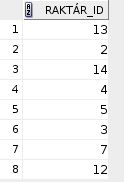
**Mely eszközöket nem szállítják éppen egyetlen hadsereghez sem?**

select eszköz\_név from raktárak r full outer join szállítás sz on r.eszköz\_név = sz.mit where sz.mit is null group by eszköz\_név;



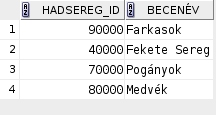
**Mely raktárból nem szállítanak sehova sem?**

select raktár\_id from raktárak r full outer join szállítás sz on r.raktár\_id = sz.honnan where sz.honnan is null group by raktár\_id;



**Mely hadseregekben nincsenek hősök?**

select h.hadsereg\_id, h.becenév from hadseregek h full outer join hős hő on h.hadsereg\_id = hő.hadsereg\_id where hő.hadsereg\_id is null group by h.hadsereg\_id,h.becenév ;



# Allekérdezések

**Szeretnénk tudni, hogy Kiss Ernő melyik hadseregben van és mekkora annak a hadseregnek a mérete.**

select hadsereg\_id,becenév,méret from hadseregek where hadsereg\_id in (select hadsereg\_id from hős where név = 'Kiss Ernő');



**Listázzunk ki bármely szállítás alatt lévő hadianyagot, amellyel fel lehetne szerelni egy hadsereg összes katonáját.**

select mit from szállítás where mennyit > any(select méret from hadseregek) group by mit;



**Listázzuk ki azokat a hadseregeket, amelyeket nem tudnánk teljesen felszerelni a raktárban található egyik felszereléssel sem.**

select hadsereg\_id, becenév, méret from hadseregek where méret>all(select eszköz\_mennyiség from raktárak);



**Listázzuk ki azokat a hadseregeket ahol van géppuskás hős.**

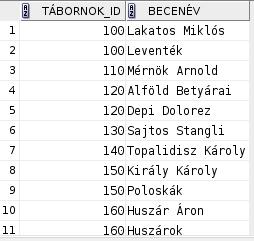
select hadsereg\_id, becenév from hadseregek where exists(select hadsereg\_id from hős where beosztás = 'Géppuskás' and hadseregek.hadsereg\_id = hős.hadsereg\_id);



# Halmaz Operátorok

**Listázzuk ki melyik tábornok, melyik becenevű hadsereget irányítja.**

select tábornok\_id,becenév from hadseregek r union select tábornok\_id,név from tábornokok;



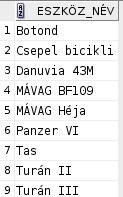
**Listázzuk ki azokat az eszközöket, amelyek raktáron is vannak és szállítanak is.**

select eszköz\_név from raktárak intersect select mit from szállítás;



**Listázzuk ki azokat az eszközöket, amelyek raktáron vannak, de éppen nem szállítják.**

select eszköz\_név from raktárak minus select mit from szállítás;



# Nézetek

**Készítsünk nézetet, amelyben benne van minden szállítmány, honnan jönnek ezek a szállítmányok és hova mennek;**

create view Mozgatások as select \* from raktárak r join szállítás sz on r.raktár\_id = sz.honnan join hadseregek h on sz.hova = h.hadsereg\_id;

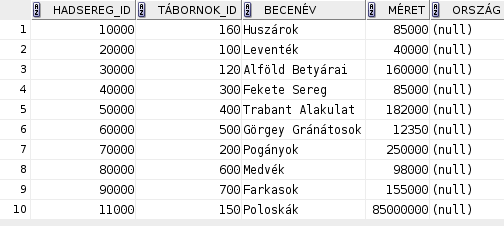
**Készítsünk nézetet, amelyben benne vannak a tábornokok, hadseregek és hozzájuk tartozó hősök adatai.**

create view Hadak as select t.tábornok\_id,t.név,t.tehetség,h.hadsereg\_id,h.méret, hős.katona\_id,hős.becenév,hős.beosztás,hős.kor from tábornokok t join hadseregek h on t.tábornok\_id = h.tábornok\_id join hős on hős.hadsereg\_id = h.hadsereg\_id;

# DDL utasítások

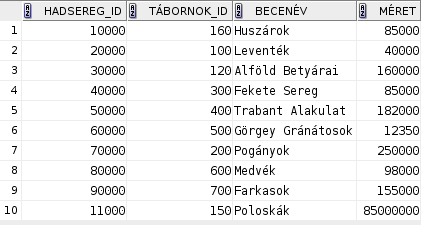
**Szeretnénk látni, hogy melyik országban állomásoznak éppen a seregek, adjunk hozzá egy új oszlopot erre a célra.**

alter table hadseregek add ország varchar(100);



**Mégse kell.**

alter table hadseregek drop column ország;



**Egy bombázás során felrobbantották az összes raktárak ezért ki kellene törölnünk a Raktárak táblát. Ahhoz, hogy ezt megtehessük, meg kell szüntetni a Szállítás táblában lévő sz\_fk1 kikötést.**

alter table szállítás drop constraint sz\_fk1;

drop table raktárak;

**Nevezzük át a Tábornokok táblát Szervezőkre a Név oszlopot meg Szervező\_név-re.**

alter table tábornokok rename to szervezők;

alter table szervezők rename column név to szervező\_név;

select \* from szervezők;



# DML utasítások

**Huszár Áron tábornokunk, sajnálatos módon meghalt, töröljük ki a tábornokok táblából.**

delete from tábornokok where upper(név) = upper('huszár áron');

**A katonák rosszak voltak, ezért nem kapnak tortát. Töröljük az összes torta szállítást.**

delete from szállítás where upper(mit) = upper('torta');

**Szaracén Dávid jelezte, hogy megérkeztek hozzájuk a szállítmányok, töröljük a seregéhez menő szállítmányt.**

delete from szállítás where hova in(select h.hadsereg\_id from hadseregek h inner join hős on h.hadsereg\_id = hős.hadsereg\_id where upper(név) = upper('szaracén dávid'));

**A 40000 azonosítójú hadsereghez csatlakozott 2000 helyi lakos, frissítsük a sereg méretét, viszont nem biztos, hogy sokáig maradnak, ezért hozzunk létre egy mentési pontot és teszteljük is le.**

savepoint helyi\_elott;

update hadseregek set hadseregek.méret = hadseregek.méret + 2000 where hadsereg\_id = 40000;

rollback helyi\_elott;





**2 év telt el, növeljük meg a hősök korát.**

update hős set kor = kor + 2;

commit;

**Felezzük meg azokat a seregeket, amik akármelyik raktárban tárolt felszerelésnél kevesebben vannak.**

update hadseregek set méret = méret/2 where méret < any( select eszköz\_mennyiség from raktárak);

# Jogosultsáskezelés

create role hadügyminiszter;

GRANT create session,   
create table, create view   
TO hadügyminiszter;

create role tiszt;

GRANT create session, create view to tiszt;

create user józsi identified by józsi123;

create user pista identified by pista123;

grant hadügyminiszter to józsi;

grant tiszt to pista;

# Triggerek

create trigger Jelzo1

after insert on tábornokok

begin

dbms\_output.put\_line('Új tábornok beszúrva!');

end;

create trigger Jelzo2

after insert on hős

begin

dbms\_output.put\_line('Új hős beszúrva!');

end;