

# Informatika 2 Zárthelyi dolgozat

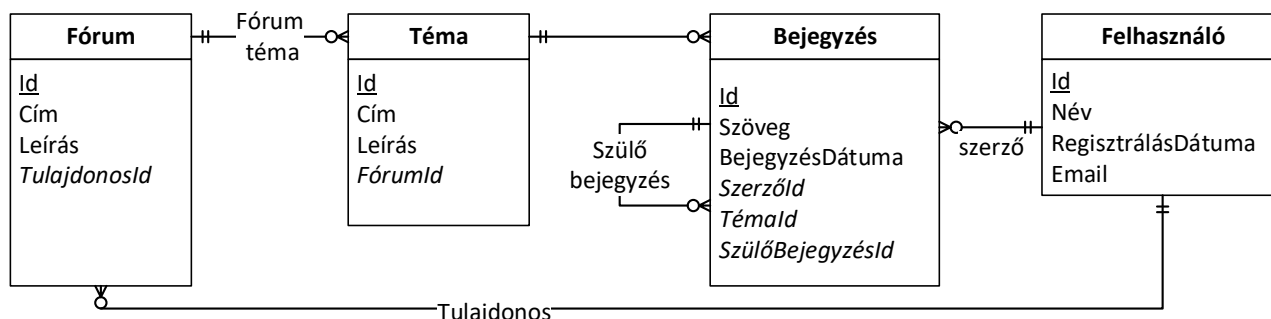
2018.04.20

## A csoport

A rendelkezésre álló idő 50 perc.  Elégséges szint: 45% (18 pont).	NÉV (olvasható név, nem aláírás!!!)	
	NEPTUN:	

Feladat	Pontszám
1 [ 6 ]	
2 [ 6 ]	
3 [ 6 ]	
4 [ 10 ]	
5 [ 6 ]	
6 [ 6 ]	
Σ [ 40 ]	

Adott az alábbi ábrán látható adatbázis séma, amely fórumok, azon belül témák és a témákhoz kapcsolódó bejegyzések adatait tárolja. A bejegyzések és a fórumok felhasználóhoz tartoznak (TulajdonosId, ill. SzerzőId oszlopok). A bejegyzéseket hierarchiába lehet rendezni, bármely bejegyzésnek lehet szülőbejegyzése (SzülőBejegyzésId). A kulcs oszlopok minden esetben AUTO\_INCREMENT opcióval vannak ellátva. (A dőlt betűs attribútumok a külső kulcsokat jelölik.)



**1. Feladat** Írjon SQL lekérdezést, amely kilistázza, hogy az egyes témákban mikor írták az utolsó bejegyzést! Az eredményben a téma címe, a kapcsolódó fórum címe, a fórum tulajdonosának neve és az utolsó bejegyzés dátuma szerepeljen! [6 pont]

```

select f.Cím, t.Cím, f.Név, max(b.BejegyzésDátum) -- 2 pont
from Fórum f
    inner join Téma t on t.FórumId = f.Id
    inner join Bejegyzés b on b.témaId = t.Id
    inner join Felhasználó fh on fh.Id = f.TulajdonosId -- 3 pont
group by f.Id -- 1 pont
    
```

**2. Feladat** Írjon SQL lekérdezést, amely kiszámítja, hogy a gmail-es (...@gmail.com) címmel rendelkező felhasználók eddig összesen hány bejegyzést írtak! [6 pont]

```

select count(b.id) -- 2 pont
from Bejegyzés b
    inner join Felhasználó f on f.Id = b.SzerzőId -- 2 pont
where LOWER(f.Email) LIKE '%@gmail.com' -- 2 pont
    
```

**3. Feladat** Tegyük fel, hogy az adatbázisban pontosan egy bejegyzés található, amelynek szövege „Info2 zárthelyi tudnivalók”. Írjon SQL utasítást, amely lekérdezi, hogy ehhez a bejegyzéshez hány gyerek lekérdezés található. [6 pont]

```

select count(b.Szöveg) -- 2 pont
from Bejegyzés b
    where b.SzülőBejegyzésId = -- 2 pont
        (select b.Id from Bejegyzés b
            where b.Szöveg = 'info2 zárthelyi tudnivalók') -- 2 pont
    
```

**4. feladat** A 3-as azonosítójú fórum címe „Informatika 2”. Hozzon létre egy új témát, amely ebben a fórumban található, címe pedig „2017/18/2”! [2 pont]

```
insert into Téma(Cím, Leírás, FórumId)
values ('2017/18/2', '2017/18 tavaszi félév', 3);          -- 2 pont (leírás opcionális)
```

Tegyük fel, hogy az új téma azonosítója 6 lett, tegyük fel továbbá, hogy létezik egy 4-es azonosítójú téma ugyanezen fórumon belül, amelynek címe „2016/17/2”! Írjon SQL utasításokat, amelyek ez utóbbi témából kitörlik az összes olyan bejegyzést, amely 2018. február 1 előtti, a többit pedig átsorolják az új csoportba! [8 pont]

```
delete from Bejegyzés                                     -- 2 pont
where BejegyzésDátuma < '2018-02-01' and TémaId = 4     -- 2 pont
```

```
update Bejegyzés                                        -- 1 pont
set TémaId = 6                                         -- 1 pont
where TémaId = 4 and BejegyzésDátuma >= '2018-02-01' -- 2 pont
```

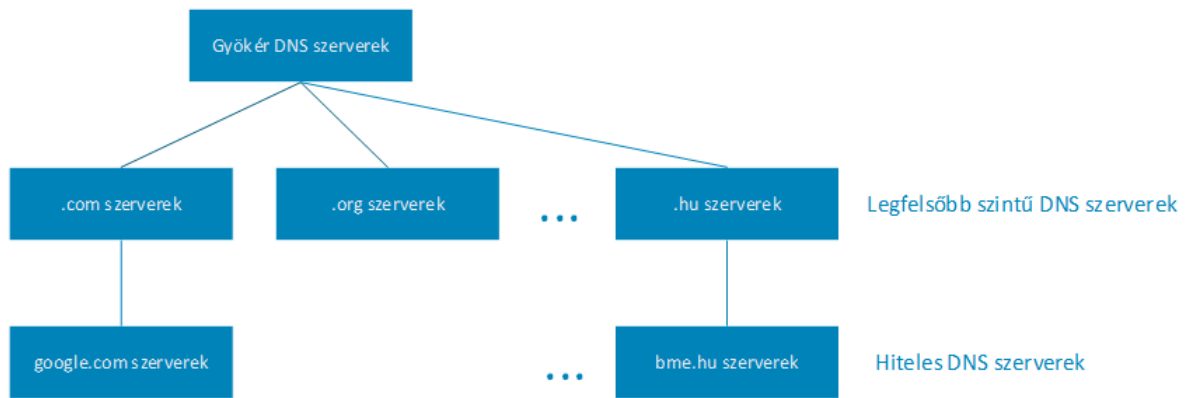
**5. Feladat** Mit jelent a csomagkapcsolás fogalma, hogyan működik egy csomagkapcsolt hálózat? Milyen problémák merülnek fel csomagkapcsolt hálózatokban? Miben különbözik a vonalkapcsolt hálózattól? [6 pont]

- Az elküldendő üzenet bitsorozatát csomagokra bontjuk, és ezeket egyenként küldjük el.
- Csomagkapcsolt hálózatban a csomópontok közötti összeköttetéseken teljes sávszélességgel küldjük az adatokat a szomszédos csomópontok között.
- A csomópontok tárol és továbbít elven működnek: bevárják a teljes bejövő csomagot és csak utána továbbítják a következő csomópontnak.
- A csomópontokban feltorlódhatnak a csomagok, illetve különböző csomagok más útvonalon juthatnak el a címzethez, ezért csomagvesztés, vagy sorrendhiba léphet fel.
- A vonalkapcsolt hálózatokban egy virtuális csatornát adott sávszélességgel lefoglalunk a küldő és a fogadó között, ennek kiépítése érinti az összes közbülső csomópontot.
  - Ebben az esetben az átvitel minőségére (sávszélesség, késleltetés) a hálózat könnyen biztosítékot tud adni.

**6. Feladat** Ismertesse, hogy milyen DNS szervereket különböztetünk meg, melyiknek mi a feladata, és milyen hierarchiába rendezve biztosítják a DNS szolgáltatást! [6 pont]

A DNS rendszert hierarchiába rendezett adatbázisszerverek alkotják.

- A **gyökér** (Root-Level) DNS szerverekből 13 darab van, amelyet 12 különböző önálló szervezet működtet. Ezek IP címei mindenki által ismertek és állandóak. [1 pont]
- A gyökér szerverek a **legmagasabb szintű** (Top-Level – TLD) DNS szerverek elérhetőségét tárolják. A TLD szerverek felelősek a legfelsőbb szintű domainekért (pl. .com, .org, .eu, .hu). [1 pont]
- A következő szinten található az egyes szervezetek saját **hiteles DNS szerverei**, amelyek a szervezeten belüli DNS leképezéseket tárolják. Minden, publikus IP címet kezelő szervezet kötelezően meg kell adjon egy hiteles DNS szervert, amely mindig a legfrissebb név-cím leképezéseket kell tárolja az adott szervezeten belül. [1 pont]



[2 pont]

- Bár a hierarchiának nem képezik részét, nagyon fontos elemei a DNS rendszernek a **lokális (helyi) DNS szerverek**. A legtöbb Internet szolgáltató kezel ilyen szervert, amelynek a feladata alapvetően a gyorsítótárazás. [1 pont]

Informatika 2 Zárthelyi dolgozat  
2018.04.20  
**B csoport**

Feladat	Pontszám
1 [ 7 ]	
2 [ 9 ]	
3 [ 7 ]	
4 [ 3 ]	
5 [ 6 ]	
6 [ 8 ]	
Σ [ 40 ]	

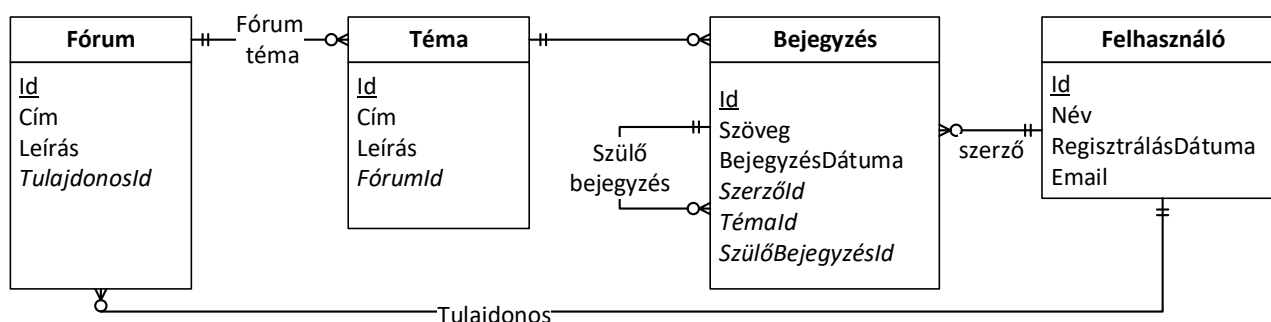
A rendelkezésre álló idő 50 perc.

NÉV (olvasható név, nem aláírás!!!)

Elégséges szint: 45% (18 pont).

NEPTUN:

Adott az alábbi ábrán látható adatbázis séma, amely fórumok, azon belül témák és a témákhoz kapcsolódó bejegyzések adatait tárolja. A bejegyzések és a fórumok felhasználóhoz tartoznak (TulajdonosId, ill. SzerzőId oszlopok). A bejegyzéseket hierarchiába lehet rendezni, bármely bejegyzésnek lehet szülőbejegyzése (SzülőBejegyzésId). A kulcs oszlopok minden esetben AUTO\_INCREMENT opcióval vannak ellátva. (A dőlt betűs attribútumok a külső kulcsokat jelölik.)



**1. Feladat** Írjon SQL lekérdezést, amely kiszámítja, hogy a gmail-es (...@gmail.com) címmel rendelkező felhasználók eddig összesen hány bejegyzést írtak idén! [7 pont]

```
select count(b.id) -- 2 pont
from Bejegyzés b
     inner join Felhasználó f on f.Id = b.SzerzőId -- 2 pont
where LOWER(f.Email) LIKE '%@gmail.com' and -- 2 pont - lower nem kötelező
     b.BejegyzésDátuma >= '2018.01.01' -- 1 pont
```

**2. Feladat** Írjon SQL lekérdezést, amely kilistázza, hogy az egyes fórumokhoz hány idei bejegyzés tartozik! Az eredmény egy táblázat, amely tartalmazza a fórum azonosítóját, címét és megfelelő bejegyzések számát. [9 pont]

```
select f.Id, f.Cím, count(b.id) -- 3 pont
from Fórum f
     inner join Téma t on t.FórumId = f.Id
     inner join Bejegyzés b on b.TémaId = t.Id -- 3 pont
where b.BejegyzésDátuma >= '2018-01-01' -- 2 pont
group by f.Id -- 1 pont
```

**3. Feladat** Írjon SQL utasítást, amely kilistázza azokat a témákat (azonosító és cím), amelyekhez legalább 50 bejegyzés található! [7 pont]

```
select t.Id, t.Cím -- 2 pont
from Téma t
     inner join Bejegyzés b on b.TémaId = t.Id -- 2 pont
group by t.Id -- 1 pont
having count(b.id) >= 50 -- 2 pont
```

**4. Feladat** Írjon SQL utasításokat, amely kitörli az adatbázisból az 1-es azonosítójú felhasználót! Figyeljen arra, hogy a külső kulcs kényszerek ne sérüljenek (ha kell törölje a kapcsolódó rekordokat)! [3 pont]

```
delete from Bejegyzés where SzerzőId = 1;           -- 1 pont
delete from Fórum where TulajdonosId = 1;         -- 1 pont
delete from Felhasználó where id = 1;             -- 1 pont
```

**5. Feladat** Ismertesse az adatbázis tranzakciók 4 alaptulajdonságát (ACID) (név és a tulajdonság jelentése) [6 pont]

1. Atomicitás: egy tranzakció utasításai oszthatatlan egységet képeznek. A végrehajtás során vagy mindegyik teljesül, vagy egyik sem.
2. Kozisztencia: az adatbázis védett a soft crash elleni védelemre. A tranzakciók határai jelölik a konzisztens állapotokat. Soft crash után is konzisztens állapotba tér vissza a rendszer.
3. Izoláció: több tranzakció párhuzamos futtatása esetén a tranzakciók utasításainak eredménye ugyanaz, mintha a tranzakciók egymástól független rendszerekben lettek volna futtatva, azaz a tranzakciók elkülönülnek egymástól.
4. Tartósság: sikeres a tranzakciók eredménye tartós, azaz a háttértárra kiírása kerül.

**6. Feladat** Ismertesse, hogy a DNS bejegyzések (resource record-ok) milyen mezőkből állnak és ezeknek mi a jelentése az egyes típusok esetén! [8 pont]

Mezők: [név, érték, típus, TTL] (1 pont)

Típusok: (6 pont)

- A típusú bejegyzés – egy IP címhez tartozó név: név = hosztnév, érték = IP cím
- CNAME típusú bejegyzés – álnévhez tartozó kanonikus név bejegyzése: név = álnév, érték = kanonikus név
- NS – egy domainhez tartozó hiteles névszerver bejegyzése: név = domain, érték = hiteles névszerver
- MX bejegyzés – egy levelezési címtartományhoz tartozó névszerver bejegyzése: név = levelezési domain, érték = névszerver neve

TTL: Time To Live – az adott bejegyzés érvényességi ideje (1 pont)