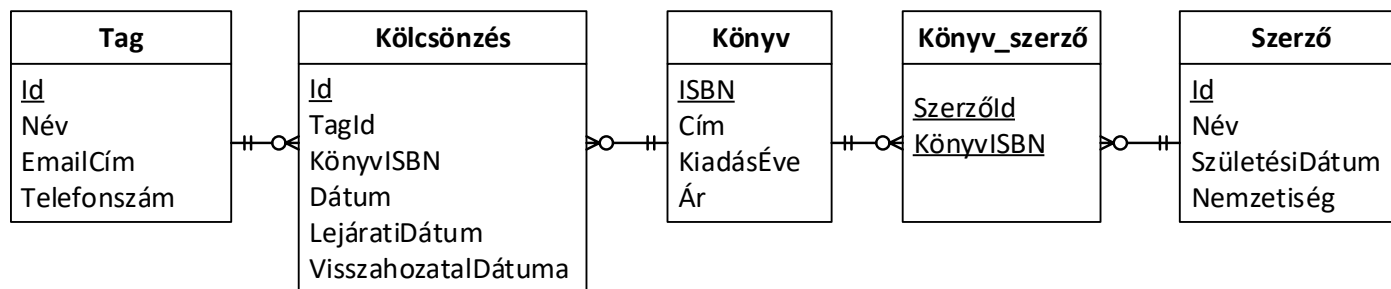


Informatika 2 Zárthelyi dolgozat 2017.04.10 A csoport		
A rendelkezésre álló idő 50 perc	NÉV (olvasható név, nem aláírás!!!):	
Elégséges szint: 45% (18 pont).	NEPTUN:	

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

Adott az alábbi ábrán látható adatbázis séma.



1. Feladat Írjon SQL lekérdezést, amely kilistázza az egyes tagok neveit és melléjük, hogy milyen szerzőtől kölcsönöztek könyvet! Egy tag-szerző pár csak egyszer szerepeljen az eredményben! **[6 pont]**

```

select distinct t.nev, sz.nev -- 3 pont
from tag t
inner join kolcsonzes kcs on t.id = kcs.tagid -- 1 pont
inner join konyv_szerzo ksz on ksz.konyvisbn = kcs.konyvisbn -- 1 pont
inner join szerzo sz on sz.id = ksz.szerzoid -- 1 pont

```

2. Feladat Listázza ki a 2010 előtt megjelent könyvek címeit és az egyes könyvek magyar nemzetiségű szerzőinek számát! *Feltehetjük, hogy minden könyvnek van legalább egy magyar nyelvű szerzője, tehát nem kell azokkal a könyvekkel foglalkozni, amelyeknek nincs szerzője, vagy nincs magyar nyelvű szerzője.* *** [7 pont]**

```

select k.cim, count(ksz.szerzoid) -- 2 pont
from konyv k
inner join konyv_szerzo ksz on ksz.konyvisbn = k.isbn -- 1 pont
where
    k.kiadaseve < 2010 -- 1 pont
    and -- 1 pont
    sz.nemzetiseg = 'magyar' -- 1 pont
group by k.id -- 1 pont

```

* (A ZH-ban ez a kiegészítés nem szerepelt, de a javításnál e szerint javítottunk.)

3. Feladat Írjon SQL utasításokat, amelyek a következő új adatokat szűrják be az adatbázisba: **[5 pont]**

- Vegyen fel egy új könyvet, amelynek címe: „Informatika 2 jegyzet”, a kiadás éve 2017, ISBN száma 1234567890, ára 2500 Ft.
- A könyvnek két szerzője van, akik már szerepelnek a **Szerző** táblában, azonosítójuk 1 és 2.
- Kölcsönözze ki a 3-as azonosítójú taggal a könyvet a mai dátummal. Feltételezheti, hogy az új kölcsönzésnek automatikusan generálunk azonosítót, mert az **Id** oszlop auto_increment módosítóval lett ellátva!

```
insert into konyv (isbn, cím, kiadaseve, ar)
values ('1234567890', 'Informatika 2 jegyzet', 2017, 2500);
```

-- 2 pont

```
insert into konyv_szerzo (szerzoid, konyvisbn)
values (1, '1234567890');
```

-- 1 pont

```
insert into konyv_szerzo (szerzoid, konyvisbn)
values (2, '1234567890');
```

-- 1 pont

```
insert into kolcsonzes (tagId, konyvisbn, datum)
values (3, '1234567890', curdate());
```

-- 1 pont
-- '2017-04-10' is jó a curdate() helyett

4. Feladat Írjon SQL utasítást, amely kitörli az adatbázisból az összes olyan kölcsönzést, amely „Horváth” vezetéknévű tagtól származik! (Vagyis olyan tagtól, amelynek a neve így kezdődik.) **[5 pont]**

```
delete from kolcsonzes
```

-- 1 pont

```
where tagid in
```

-- 1 pont

```
  (select id
```

-- 1 pont

```
   from tag
```



```
   where nev like 'Horvath%')
```

-- 2 pont

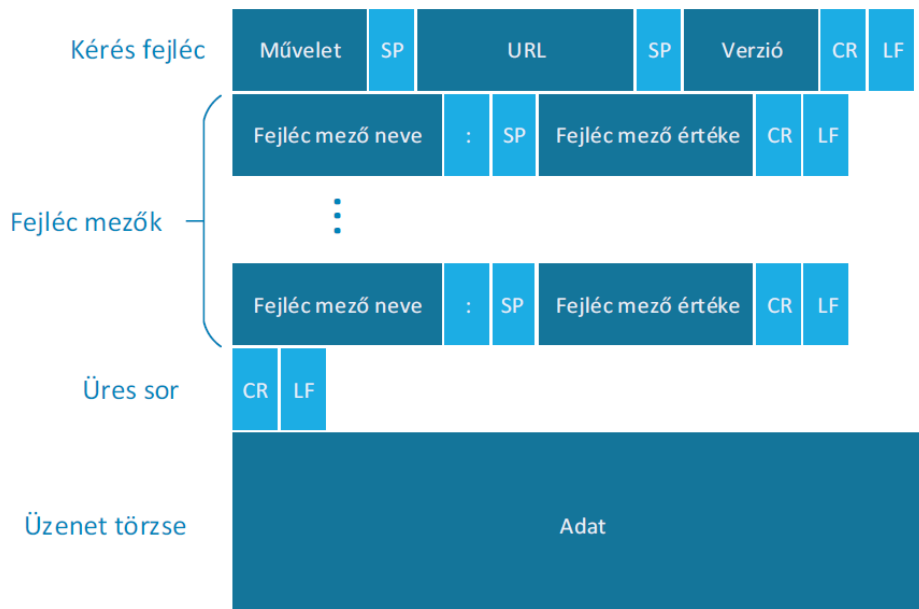
5. Feladat Ismertesse az Internet protokollrétegeit (sorrendben) és röviden ismertesse az egyes rétegek feladatait! **[7 pont]**

Réteg neve	Feladata
Alkalmazási réteg	Tartalmazza az elosztott alkalmazásokat, az elosztott alkalmazás komponensek egymásnak üzeneteket küldenek.
Szállítási réteg	Üzenetátvitel a hálózat két pontján lévő eszközökön futó alkalmazás folyamatok között.
Hálózati réteg	Üzenetátvitel a hálózat két pontján lévő eszközök között.
Adatkapcsolati réteg	Üzenetátvitel két szomszédos hálózati csomópont között.
Fizikai réteg	Az üzenetátvitel fizikai megvalósítása.

6. Feladat: Ismertesse a socketkezelés szerveroldali függvényeit és hogy melyik mire szolgál! [7 pont]

1. Szerver socket létrehozása (socket).
2. A socket hozzárendelése egy hálózati címhez és portszámhoz (bind). Itt a programozó feladata, hogy egy még nem használt portszámot megadjon. Ha a portszám foglalt, akkor nem jön létre a socket, hanem hibát kapunk.
3. Harmadik lépésben a szerver socketet passzív állapotba helyezzük, felkészítve arra, hogy később beérkező kapcsolatokat fogadjon (listen).
4. Várakozás kliensek kapcsolódására (accept). Amennyiben egy kliens csatlakozik, akkor létrehozunk egy új socketet, amelyen keresztül az újonnan csatlakozott klienssel tudunk kommunikálni. Erre azért van szükség, hogy az eredeti socketet fenntartsuk további csatlakozni akaró kliensek számára.
5. Adatok fogadása (recv).
6. Adatok küldése (send) a socketen keresztül.
7. A socket lezárása (closesocket), ha már nincsen rá szükség. Erre azért van szükség, hogy az operációs rendszer erőforrásait ne foglaljuk feleslegesen.

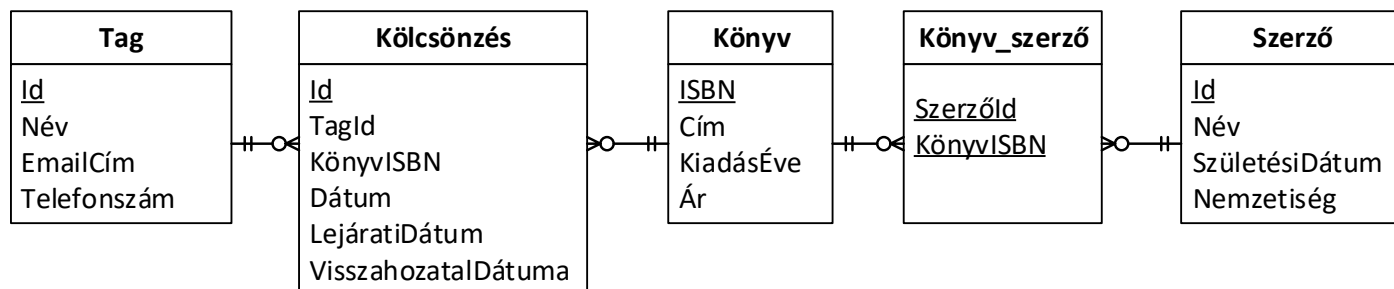
7. Feladat Ismertesse a HTTP kérés üzenetek felépítését! [3 pont]



Informatika 2 Zárthelyi dolgozat 2017.04.10 B csoport		
A rendelkezésre álló idő 50 perc. Elégséges szint: 45% (18 pont).	NÉV (olvasható név, nem aláírás!!!)	
	NEPTUN:	

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

Adott az alábbi ábrán látható adatbázis séma.



1. Feladat Írjon SQL lekérdezést, amely kilistázza az egyes könyvek címeit és melléjük azon tag neveit, akik kikölcsönözték a könyvet! Egy könyvcím-tagnév páros csak egyszer szerepeljen az eredményben! **[5 pont]**

```

select distinct k.cím, t.nevf
from konyv k
inner join kolcsonzes kcs on k.isbn = kcs.konyvisbn
inner join tag t on t.id = kcs.tagid
-- 2 pont
-- 1 pont
-- 1 pont
-- 1 pont

```

2. Feladat Írjon SQL lekérdezést, amely kilistázza a legdrágább könyvek címeit és kiadásuk évét! **[3 pont]**

```

select cím, kiadaseve
from konyv
where ar = (select max(ar) from konyv)
-- 1 pont
-- 2 pont

```

3. Feladat Írjon SQL lekérdezést, amely kilistázza, hogy az egyes szerzők műveit összesen hányszor kölcsönözték! Csak azok a szerzők jelenjenek meg az eredményben, akik magyar nemzetiségűek! **[7 pont]**

```

select sz.nevf, count(kcs.id)
from szerzo sz
inner join konyv_szerzo ksz on sz.id = ksz.szerzoid
inner join kolcsonzes kcs on kcs.konyvisbn = k.isbn
where sz.nemzetiseg = 'magyar'
group by sz.id
-- 2 pont
-- 1 pont
-- 1 pont
-- 1 pont
-- 1 pont
-- 1 pont

```

4. Feladat Az 1-es azonosítójú tag-nak van egy folyamatban lévő kölcsönzése az 1234567890 ISBN számú könyvre. (Onnan tudjuk, hogy melyik kölcsönzés van folyamatban, hogy nincs kitöltve a visszahozatal dátuma.) Írjon SQL nyelvű utasításokat, amelyek a következő műveleteket végzik el!

- A tag hozza vissza a kölcsönzött könyvet, vagyis a fenti kölcsönzésnek megfelelő bejegyzésben a visszahozatal dátumát állítsa az aktuális dátumra! **[5 pont]**
- Hozzon létre egy új kölcsönzést a 2-es azonosítójú taggal ugyanerre a könyvre a mai dátummal! Feltételezheti, hogy az új kölcsönzésnek automatikusan generálunk azonosítót, mert az **Id** oszlop **auto_increment** módosítóval lett ellátva! **[4 pont]**

```

update kolcsonzes
set visszahozataldatum = curdate() -- '2017-04-10' is jó -- 1 pont
where
  konyvisbn = '1234567890' -- 1 pont
  and
  visszahozataldatum is null -- 1 pont
  and
  tagid = 1 -- 1 pont

insert into kolcsonzes (tagid, konyvisbn, datum) -- 2 pont
values (2, '1234567890', curdate()) -- 2 pont
-- '2017-04-10' is jó a curdate() helyett

```

5. Feladat: Ismertesse a socketkezelés kliensoldali függvényeit és hogy melyik mire szolgál! [5 pont]

1. Új socket létrehozása (socket). A létrehozás során meg kell adni, hogy milyen szállítási rétegbeli protokollt szeretnénk használni (TCP vagy UDP).
2. A socket segítségével kapcsolódás egy szerverhez (connect). A paraméternél meg kell adni, a szerver IP címét és portszámát.
3. Adatok küldése (send) és
4. fogadása (recv).
5. A socket lezárása (closesocket), ha már nincsen rá szükség.

6. Feladat Válaszoljon a TCP-vel kapcsolatos következő kérdésekre:

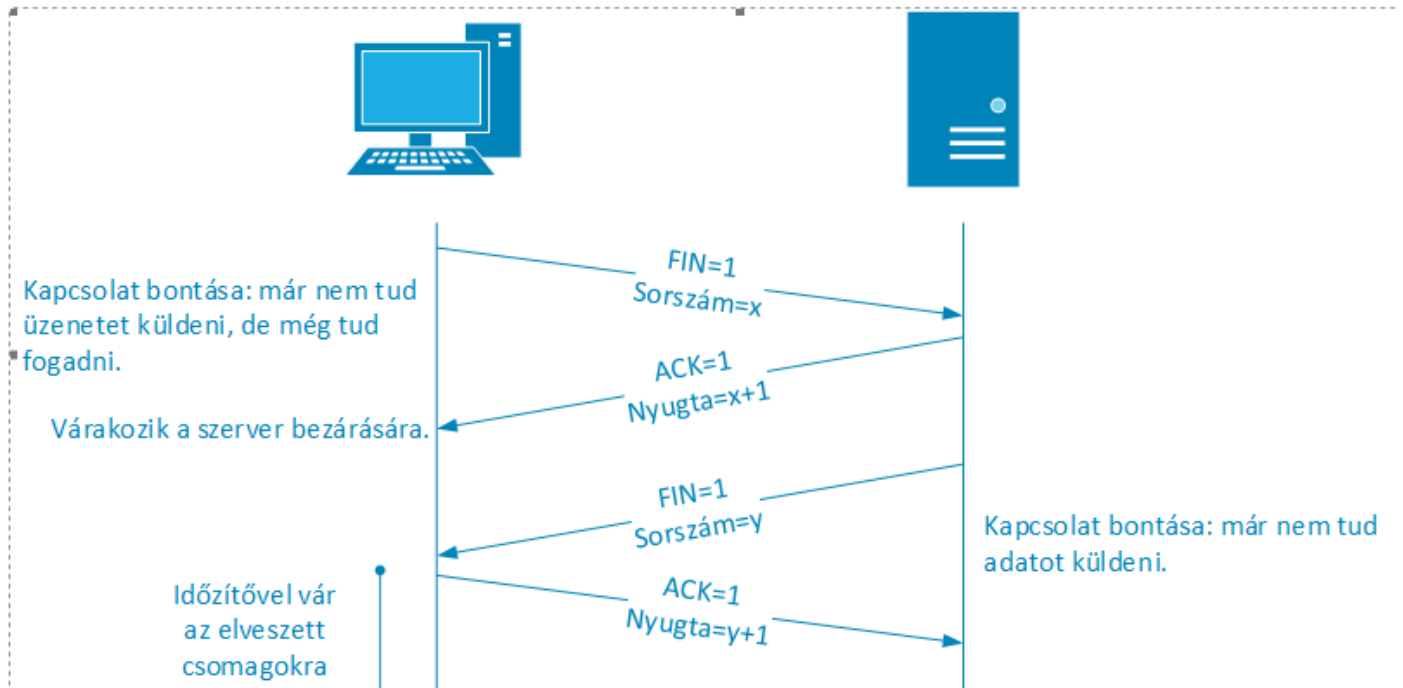
Mire szolgál a forgalomszabályozás? Hogyan valósítja meg a TCP a forgalomszabályozást? [2 pont]

- *A forgalomszabályozás feladata annak megoldása, hogy a küldő fél ne terhelje túl a címzettet. Ugyanis, ha a küldő gyorsabban küldi a csomagokat, mint ahogyan a címzett fogadni tudja, akkor túltelített pufferek miatt a címzett egy idő után eldobja az újonnan érkező csomagokat, miközben a hálózatot feleslegesen terheli a folyamatos adatfolyam. A forgalomszabályozás célja, hogy ilyenkor a küldő korlátozza a csomagküldést. [1 pont]*
- *A TCP szegmensben van egy vételi ablak mező. A vételi ablak jelzi, hogy még mennyi adat befogadására képes a vevő. Ennél többet nem érdemes küldeni, mert a tároló puffer hiányában csak eldobnánk az üzenetet, vagyis az elveszne. [1 pont]*

Milyen szabályok szerint képezzük az egyes csomagok sorszámát? [2 pont]

- *Az elküldendő adat a szállítási rétegben egy bájt sorozatként jelenik meg. Ezt a bájt sorozatot bontjuk fel szegmensekre. A hivatkozott sorszámok azonban nem a szegmensek sorszámát jelölik, hanem a szegmensek első bájtjának indexét ebben a sorozatban. [1 pont]*
- *Valójában az első index nem feltétlenül 0, hanem egy véletlenül generált szám, ezzel is csökkentve annak a valószínűségét, hogy ütközés legyen a hálózatban lévő korábbi csomagok sorszámával. [1 pont]*

7. Feladat: Ismertesse a TCP kapcsolatbontás forgatókönyvét! (Üzenetek sorrendje, jelzőbitek, sorszáмок.) [7 pont]



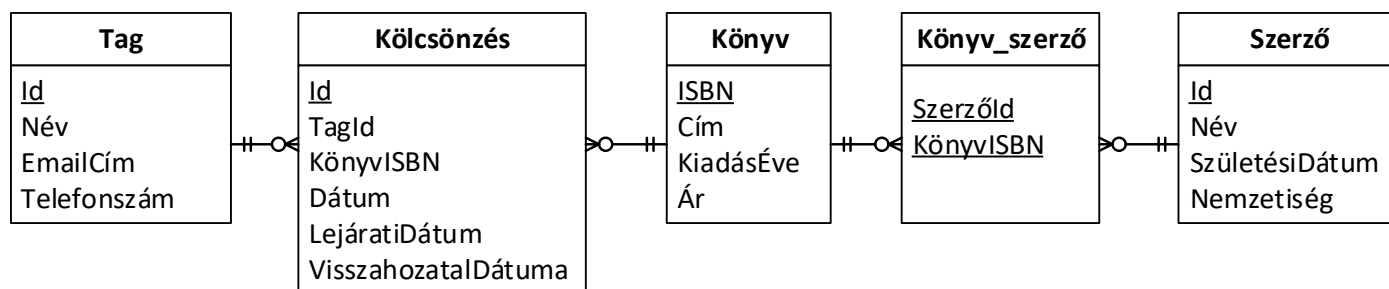
Informatika 2 Zárthelyi dolgozat

2017.04.10

iMSc feladat

NÉV (olvasható név, nem aláírás!!!):	
NEPTUN:	

Adott az alábbi ábrán látható adatbázis séma.



iMSc Feladat Írjon SQL lekérdezést, amely kilistázza az adatbázisban lévő összes könyv címét és ISBN számát, továbbá a könyv magyar szerzőinek a számát!

```

SELECT k.isbn, k.cim, count(sz.id) as MagyarSzerzokSzama
FROM konyv k
left outer join konyv_szerzo ksz on k.isbn = ksz.konyvisbn
left outer join (select id from szerzo where nemzetiseg = 'Magyar') sz on sz.id = ksz.szerzoid
group by k.isbn
    
```