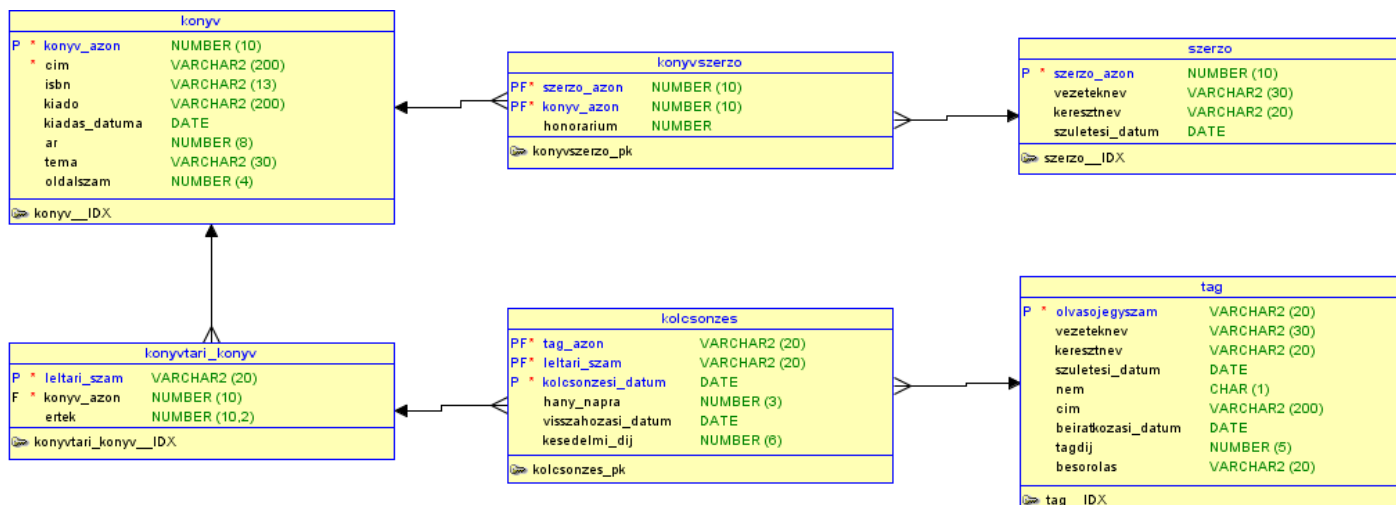


SQL feladatok és megoldások a könyvtár sémára

Készítette: Vágner Anikó



Az első select utasítások

1. Listázzuk ki a tagokat!

```
select *
from konyvtar.tag;
```

2. Listázzuk ki a tagok nevét!

```
select vezeteknev, keresztnév
from konyvtar.tag;
```

Select utasítás order by utasításrészsel

3. Listázzuk ki a tagok nevét! A listát rendezzük keresztnév szerint csökkenően, majd vezetéknév szerint növekvően!

```
select vezeteknev || ' ' || keresztnév
from konyvtar.tag
order by keresztnév desc, vezeteknev asc;
```

4. Listázzuk ki a tagok nevét! A listát rendezzük név szerint!

```
select vezeteknev || ' ' || keresztnév
from konyvtar.tag
order by név;
```

5. Listázzuk ki a könyveket. A lista legyen ár szerint rendezett, és a null értékek elől szerepeljenek.

```
select *
from konyvtar.konyv
order by ár nulls first;
```

6. Listázzuk ki a könyveket. A lista legyen ár szerint csökkenően rendezett, és a null értékek a végén szerepeljenek.

```
select *
from konyvtar.konyv
order by ár desc nulls last;
```

Select utasítás where utasításrészsel

7. Listázzuk ki a női tagokat! A listát rendezzük név szerint!

```
select *
from konyvtar.tag
where nem='n'
order by vezeteknev, keresztnév;
```

8. Listázzuk ki Luc Erna nevű tag minden adatát!

```
select *
from konyvtar.tag
where vezeteknev='Luc' and keresztnév='Erna';
```

9. Listázzuk ki a diák besorolású tagok minden adatát! A lista legyen név szerint rendezett.

```
select *
from konyvtar.tag
```

```
where besorolas='diák'
order by vezeteknev, keresztnev;
```

10. Listázzuk ki azokat a tagokat, akiknek a tagdíja több, mint 1000 és a besorolása felnőtt!

```
select *
from konyvtar.tag
where tagdij>1000 and besorolas='felnőtt';
```

11. Listázzuk ki azokat a könyveket, amelyek krimi témájúak, és amelyeknek az ára több, mint 3000! A lista legyen cím szerint rendezett.

```
select *
from konyvtar.konyv
where tema='krimi' and ar>3000
order by cim;
```

12. Listázzuk ki azokat a könyveket, amelyek krimi témájúak, vagy amelyeknek az ára több, mint 3000! A lista legyen cím szerint rendezett.

```
select *
from konyvtar.konyv
where tema='krimi' or ar>3000
order by cim;
```

13. Írjunk megjegyzést!

- o /*többsoros
megjegyzés*/
- o --megjegyzés

14. Listázzuk ki azokat a tagokat, akiknek a tagdíja több, mint 1000 vagy a besorolása felnőtt!

```
select *
from konyvtar.tag
where tagdij>1000 or besorolas='felnőtt';
```

15. Listázzuk ki azokat a tagokat, akikre nem igaz, hogy a tagdíja több, mint 1000 vagy a besorolása felnőtt!

```
select *
from konyvtar.tag
where not (tagdij>1000 or besorolas='felnőtt');
```

16. Keressük azokat a tagokat, akiknek a besorolása diák, nyugdíjas vagy gyerek.

- o select *
from konyvtar.tag
where besorolas='diák' or besorolas='nyugdíjas'
or besorolas='gyerek'
order by besorolas, vezeteknev, keresztnev;
- o select *
from konyvtar.tag
where besorolas in ('diák', 'nyugdíjas', 'gyerek')
order by besorolas, vezeteknev, keresztnev;

17. Keressük azoknak a könyveknek a címét és árát, amelyeknek az ára 1000 és 3000 között van. A listát rendezzük ár, azon belül cím szerint.

- o select ar, cim
from konyvtar.konyv
where ar>=1000 and ar<=3000
order by ar, cim;
- o select ar, cim
from konyvtar.konyv
where ar between 1000 and 3000
order by ar, cim;

18. Keressük azoknak a könyveknek a címét és árát, amelyeknek az ára nem nagyobb vagy egyenlő, mint ezer. A listát rendezzük ár, azon belül cím szerint.

```
select ar, cim
from konyvtar.konyv
where not (ar >=1000)
order by ar, cim;
```

19. Keressük azoknak a könyveknek a címét és árát, amelyeknek az ára nem ezer és háromezer között van. A listát rendezzük ár, azon belül cím szerint.

```
select ar, cim
from konyvtar.konyv
where ar not between 1000 and 3000
order by ar, cim;
```

20. Keressük azokat a könyveket, amelyek nem krimi és sci-fi témájúak. A listát rendezzük ár, azon belül cím szerint.

```
select *
from konyvtar.konyv
where tema not in ('krimi', 'sci-fi')
order by ar, cim;
```

21. Keressük azokat a könyveket, amelyek sci-fi témájúak vagy olcsóbbak 1000-nél és oldalszámuk több, mint 200 vagy a Gondolat kiadó a kiadójuk.

```
select *
from konyvtar.konyv
where (tema='sci-fi' or ar<1000)
and (oldalszam>200 or kiado='Gondolat');
```

22. Keressük azokat a könyveket, amelyeknek nincs megadva a kiadójuk.

```
select *
from konyvtar.konyv
where kiado is null;
```

23. Keressük azokat a könyveket, amelyeknek meg van adva a kiadójuk.

```
select *
from konyvtar.konyv
where kiado is not null;
```

24. Keressük azokat a könyveket, amelyeknek a címe a 'Re' sztringgel kezdődik.

```
select *
from konyvtar.konyv
where cim like 'Re%'
```

25. Keressük azokat a könyveket, amelyeknek a címe nem a 'Re' sztringgel kezdődik.

```
select *
from konyvtar.konyv
where cim not like 'Re%';
```

26. Keressük azokat a könyveket, amelyeknek a címének a 2. karaktere 'a'.

```
select *
from konyvtar.konyv
where cim like '_a%';
```

27. Melyek azok a könyvek, amelyeknek az oldalankénti ára több, mint 20? Listázzuk a könyv címét, azonosítóját, oldalankénti árát. A lista legyen az oldalankénti ár szerint rendezett.

```
select cim, konyv_azon, ar/oldalszam
from konyvtar.konyv
where ar/oldalszam>20
order by ar/oldalszam;
```

Select utasítás függvényekkel

28. Listázzuk ki a könyvek címét, és oldalankénti árát. Ez utóbbi értékkel próbáljuk ki a trunc és a round függvények működését.

```
select cim,ar/oldalszam, trunc(ar/oldalszam), trunc(ar/oldalszam,2),
       round(ar/oldalszam), round(ar/oldalszam,2)
from konyvtar.konyv;
```

29. Listázzuk ki -5 abszolút értékét!

```
select abs(-5)
from dual;
```

30. Mennyi 12*50 és 25-nek a gyöke?

```
select 12*50, sqrt(25)
from dual;
```

31. Listázzuk ki az 'almafa' sztring hosszát!

```
select length('almafa')
from dual;
```

32. Listázzuk ki, hogy az 'almafa' sztringben az 'af' sztring hányadik karaktertől kezdődik!

```
select instr('almafa','af')
from dual;
```

33. Listázzuk ki a tagok nevét és nemét. Az utolsó oszlopon szerepeljen a 'férfi' karaktersorozat, ha a nem 'f', a 'nő', ha a nem 'n', és '?' egyébként.

```
select vezeteknev, keresztnév, nem, decode(nem,'f','férfi','n','nő','?')
from konyvtar.tag;
```

34. Listázzuk ki a könyvek oldalszámát! A 2. oszlopban is a könyvek oldalszáma szerepeljen, azonban ha az nincs megadva, akkor -1 jelenjen meg.

```
select oldalszam, nvl(oldalszam, -1)
from konyvtar.konyv;
```

35. Listázzuk ki a könyvek címét és témáját. A témát még egyszer is jelenítsük meg úgy, hogy ha null érték, akkor helyette 3 db '*' -ot írjunk.

```
select cim,tema, nvl(tema, '***')
from konyvtar.konyv;
```

36. Listázzuk ki a felhasználói nevünket!

```
select user
from dual;
```

37. Füzzük össze az 'alma' és a 'fa' szavakat kétféle megoldással!

```
select concat('alma','fa'),'alma' || 'fa' from dual;
```

38. Listázzuk ki az 'almafa' szót nagy kezdőbetűvel!

```
select initcap('almafa')
from dual;
```

39. Listázzuk ki a tagok vezetéknévét, majd a tagok nevét csupa kisbetűvel, és csupa nagybetűvel!

```
select vezeteknev, lower(vezeteknev), upper(vezeteknev)
from konyvtar.tag;
```

40. Listázzuk ki a tagok vezetéknévét! Próbáljuk ki a substr függvény működését, a 3. karaktertől kezdődően vegyünk 4 karaktert a vezetéknévből.

```
select vezeteknev, substr(vezeteknev,3,4)
from konyvtar.tag;
```

41. Listázzuk ki a tagok vezetéknéveit! Próbáljuk ki a replace függvényt. Ha a vezetéknévben szerepel az 'er' karaktersorozat, cseréljük le 3 db *-ra.

```
select vezeteknev, replace(vezeteknev, 'er', '***' )
from konyvtar.tag;
```

42. Listázzuk ki azokat a tagokat, akinek a vezetéknévében legalább két 'a' betű szerepel, mindegy, hogy kicsi vagy nagy.

```
select *
from konyvtar.tag
where lower(vezeteknev) like '%a%a%';
```

43. Listázzuk ki a tagok születési dátumát!

```
select to_char(születesi_datum, 'yyyy.mm.dd hh24:mi:ss')
from konyvtar.tag;
```

44. Írjuk ki a mai dátumot, a holnapi dátumot, az egy héttel ezelőtti dátumot és a 12 órával ezelőtti dátumot. A dátumok mellett az idő is szerepeljen.

```
select to_char(sysdate, 'yyyy.mon dd hh24:mi:ss'),
       to_char(sysdate+1, 'yyyy.mon dd hh24:mi:ss'),
       to_char(sysdate-7, 'yyyy.mon dd hh24:mi:ss'),
       to_char(sysdate-1/2, 'yyyy.mon dd hh24:mi:ss')
from dual;
```

45. Listázzuk ki a mai dátumot, a két hónappal későbbi dátumot és a két hónappal korábbi dátumot.

```
select to_char(sysdate, 'yyyy.mon dd hh24:mi:ss'),
       to_char(add_months(sysdate,2), 'yyyy.mon dd hh24:mi:ss'),
       to_char(add_months(sysdate,-2), 'yyyy.mon dd hh24:mi:ss')
from dual;
```

46. Listázzuk ki a mai dátumot, az egy évvel későbbi dátumot és az egy évvel korábbi dátumot.

```
select to_char(sysdate, 'yyyy.mon dd hh24:mi:ss'),
       to_char(add_months(sysdate,12), 'yyyy.mon dd hh24:mi:ss'),
       to_char(add_months(sysdate,-12), 'yyyy.mon dd hh24:mi:ss')
from dual;
```

47. Hány nap telt el 2000 január 1 óta?

```
select sysdate-to_date('2000.01.01', 'yyyy.mm.dd')
from dual;
```

48. Hány év telt el 2000 január 1 óta?

```
select months_between(sysdate, to_date('2000.01.01', 'yyyy.mm.dd'))/12
from dual;
```

49. Kik a 30 évnél fiatalabb tagok?

```
o select vezeteknev, keresztnév, to_char(születesi_datum, 'yyyy.mm.dd')
   from konyvtar.tag
  where months_between(sysdate, születesi_datum)/12 < 30;

o select vezeteknev, keresztnév, to_char(születesi_datum, 'yyyy.mm.dd')
   from konyvtar.tag
  where add_months(születesi_datum, 12*30) > sysdate;
```

50. Melyek azok a tagok, akik 2000.01.01 előtt születtek és akiknek a besorolása nyugdíjas, vagy felnőtt?

```
select *
from konyvtar.tag
where születesi_datum < to_date('2000.01.01', 'yyyy.mm.dd')
and besorolas in ('nyugdíjas', 'felnőtt');
```

51. Melyek azok a tagok, akiknek a címében szerepel az 'út' szó, vagy a nevükben pontosan két 'e' betű szerepel? A lista legyen név szerint rendezve.

```
select *
from konyvtar.tag
where lower(cim) like '%út%' or
       (lower(vezeteknev)||keresztnév) like '%e%' and
       lower(vezeteknev)||keresztnév not like '%e%e%')
order by vezeteknev, keresztnév;
```

52. Listázzuk ki az 1980.03.02 előtt született férfi tagok nevét és születési dátumát.

```
select vezeteknev, keresztnév, to_char(születesi_datum, 'yyyy.mm.dd')
from konyvtar.tag
where nem='f' and születesi_datum < to_date('1980.03.02', 'yyyy.mm.dd');
```

53. Azokat a tagokat keressük, akinek a nevében legalább kettő 'e' betű szerepel és igaz rájuk, hogy 40 évnél fiatalabbak vagy besorolásuk gyerek.

```
select vezeteknev, keresztnév,
       to_char(SZULETESI_DATUM, 'yyyy.mm.dd'), besorolas
from konyvtar.tag
where lower(vezeteknev||keresztnév) like '%e%e%'
       and (besorolas='gyerek' or months_between(sysdate, SZULETESI_DATUM)/12<40);
```

54. Listázzuk ki a tagok nevét, születési dátumát, életkorát és besorolását. A lista legyen besorolás szerint csökkenően, azon belül név növekvően rendezett.

```
select vezeteknev, keresztnév, besorolas,
       to_char(születesi_datum, 'yyyy.mm.dd'),
       months_between(sysdate, szuletesi_datum)/12
from konyvtar.tag
order by besorolas desc, vezeteknev, keresztnév;
```

Select utasítás csoportosító függvényekkel, group by és having utasításrészekkel

55. Listázzuk ki a könyvek minimum árát, maximum árát, összárát, átlagárát és darabszámát. Vizsgáljuk meg, hogy az átlagárát hogyan számolja a rendszer.

```
select min(ar), max(ar), sum(ar), avg(ar), count(konyv_azon), sum(ar)/count(konyv_azon), sum(ar)/count(ar)
from konyvtar.konyv;
```

56. Számoljuk meg, hogy hány könyv azonosító, hány ár, és hány sor van a könyv táblába. Figyeljük meg és magyarázzuk meg a különbséget.

```
select count(konyv_azon), count(ar), count(*)
from konyvtar.konyv;
```

57. Hány téma van?

```
select count(tema)
from konyvtar.konyv;
```

58. Hány darab 70 évnél idősebb szerző van?

```
select count(*)
from konyvtar.szerzo
where months_between(sysdate, szuletesi_datum)/12>70;
```

59. Mi a legidősebb szerző születési dátuma és életkora?

```
select to_char(min(születesi_datum), 'yyyy.mm.dd'),
       months_between(sysdate, min(születesi_datum))/12
from konyvtar.szerzo;
```

60. ABC sorrendben melyik az első és az utolsó téma?

```
select min(tema), max(tema)
from konyvtar.konyv;
```

61. Listázzuk ki a témákat! Mindegyik csak egyszer szerepeljen.

```
select distinct tema
from konyvtar.konyv;
```

62. Hány különböző téma van?

```
select count(distinct tema)
from konyvtar.konyv;
```

63. Hány darab olyan könyv van, amelyiknek a címe pontosan kettő 'a' betűt (mindegy, hogy kicsi vagy nagy) tartalmaz?

```
select count(*)
from konyvtar.konyv
where lower(cim) like '%a%a%'
       and lower(cim) not like '%a%a%a%';
```

64. Mi a legidősebb szerző születési dátuma?

```
select to_char(min(születesi_datum), 'yyyy.mm.dd')
from konyvtar.szerzo;
```

65. Mi a témája azoknak a könyveknek, amelyeknek az oldalankénti ára 10 és 150 között van. Minden téma csak egyszer szerepeljen, és legyen a lista rendezett.

```
select distinct tema
from konyvtar.konyv
where ar/oldalszam between 10 and 150
order by tema;
```

66. Mi a horror, sci-fi vagy krimi témájú könyvek átlagára?

```
select avg(ar)
from konyvtar.konyv
where tema in ('horror', 'sci-fi', 'krimi');
```

67. Mi a horror, sci-fi, krimi témájú könyvek közül a legdrágábbnak az ára?

```
select max(ar)
from konyvtar.konyv
where tema in ('horror', 'sci-fi', 'krimi');
```

68. Mennyi a legkisebb oldalszámú könyv oldalszáma azok között a könyvek között, amelyeknek az ára 1000 és 5000 között van.

```
select min(oldalszam)
from konyvtar.konyv
```

```
where ar between 1000 and 5000;
```

69. Listázzuk ki az 1940 és 1960 között született női tagok besorolását. Minden besorolás csak egyszer szerepeljen. Rendezzük a listát.

```
select distinct besorolas
from konyvtar.tag
where nem='n' and
      szulesi_datum between to_date('1940.01.01','yyyy.mm.dd')
      and to_date('1960.12.31','yyyy.mm.dd')
order by besorolas;
```

70. Hány női, diák besorolású tag van?

```
select count(*)
from konyvtar.tag
where nem='n' and besorolas='diák';
```

71. Hány különböző városnév szerepel a tagok címeiben?

```
select count(distinct substr(cim,6,instr(cim,',')-6))
from konyvtar.tag;
```

72. Hány olyan tag van, aki 40 évnél fiatalabb vagy a nevében pontosan kettő darab 'e' betű szerepel?

```
select count(*)
from konyvtar.tag
where months_between(sysdate, szulesi_datum)/12<40
or (lower(vezeteknev)||' '||keresztnev) like '%e%e%'
and lower(vezeteknev)||' '||keresztnev) not like '%e%e%e%';
```

73. Hány olyan könyv van, amelyeknek az oldalankénti ára kevesebb, mint 10?

```
select count(*)
from konyvtar.konyv
where ar/oldalszam<10
```

74. Mi a legutolsó kiadás dátuma?

```
select to_char(max(kiadas_datuma),'yyyy.mm.dd')
from konyvtar.konyv;
```

75. Mi az összértéke azoknak a könyvpéldányoknak, amelyek az 1116152201 azonosítójú könyvekhez tartoznak.

```
select sum(ertek)
from konyvtar.konyvtari_konyv
where konyv_azon=1116152201;
```

76. Mi a legfiatalabb női, diák besorolású tag születési dátuma?

```
select to_char(max(szulesi_datum),'yyyy.mm.dd')
from konyvtar.tag
where nem='n' and besorolas='diák';
```

77. Témánként mennyi a minimum ár, a maximum ár, az árak összege, az átlagár, és a könyvek száma? A lista legyen téma szerint rendezett.

```
select tema, min(ar), max(ar), sum(ar),
avg(ar), count(*), count(KONYV_AZON)
from konyvtar.konyv
group by tema
order by tema;
```

78. Az egyes témákhoz hány könyv tartozik?

```
select tema, min(ar)
from konyvtar.konyv
group by tema;
```

79. Melyek azok a témák, amelyekhez 7-nél több könyv tartozik?

```
select tema
from konyvtar.konyv
group by tema
having count(*)>7;
```

80. Városonként hány olyan tag van, aki 1980.03.01 előtt született?

```
select substr(cim, 6,instr(cim,',')-6) varos, count(*)
from konyvtar.tag
where szulesi_datum<to_date('1980.03.01','yyyy.mm.dd')
group by substr(cim, 6,instr(cim,',')-6)
order by varos;
```

81. Melyik szerző (csak szerzo_azon) összhonorárium a nagyobb, mint 2000000?

```
select szerzo_azon, sum(honorarium)
from konyvtar.konyvszerzo
group by szerzo_azon
having sum(honorarium)>2000000;
```

82. Témánként mennyi az átlagos oldalszám?

```
select tema, avg(oldalszam)
from konyvtar.konyv
group by tema;
```

83. Melyik az a téma, amelynek az átlagos oldalszáma kevesebb, mint 400?

```
select tema, avg(oldalszam)
from konyvtar.konyv
group by tema
having avg(oldalszam)<400;
```

84. Momogrammonként hány tag van?

```
select substr(vezeteknev,1,1)||substr(keresztnev,1,1), count(*)
from konyvtar.tag
group by substr(vezeteknev,1,1)||substr(keresztnev,1,1);
```

85. Melyik az a születési év és hónap, amelyikben 10-nél kevesebb tag született?

```
select to_char(születesi_datum,'yyyy.mm'), count(*)
from konyvtar.tag
group by to_char(születesi_datum,'yyyy.mm')
having count(*)<10;
```

86. Besorolásonként mennyi az átlagos tagdíj?

```
select besorolas, avg(tagdij)
from konyvtar.tag
group by besorolas;
```

87. Könyvenként (csak konyv_azon kell) mennyi az összérték?

```
select konyv_azon, sum(ertek)
from konyvtar.konyvtari_konyv
group by konyv_azon;
```

88. Kiadónként mi a legutolsó kiadás dátuma?

```
select kiado, to_char(max(kiadas_datuma),'yyyy.mm.dd')
from konyvtar.konyv
group by kiado;
```

89. Melyek azok a témák, amelyekben 3-nál több olyan könyvet adtak ki, amelyeknek az ára 1000 és 3000 között van?

```
select tema, count(konyv_azon)
from konyvtar.konyv
where ar between 1000 and 3000
group by tema
having count(konyv_azon)>3;
```

90. Besorolásonként és nemenként mennyi az átlagos tagdíj és a tagok száma?

```
select besorolas, nem, avg(tagdij), count(*)
from konyvtar.tag
group by besorolas, nem;
```

91. A keresztnév első két betűjeként csoportosítva hány női tag van? Rendezzük a listát darabszám szerint csökkenően.

```
select substr(keresztnev,1,2), count(*) db
from konyvtar.tag
where nem='n'
group by substr(keresztnev,1,2)
order by db desc;
```

92. Könyvazonosítónként mi a legnagyobb értéke a példányoknak?

```
select konyv_azon, max(ertek)
from konyvtar.konyvtari_konyv
group by konyv_azon;
```

93. Melyik kiadó adott ki 10 ezer feletti összértékben könyvet?

```
select kiado,sum(ar)
from konyvtar.konyv
group by kiado
having sum(ar)>10000;
```

94. Melyek azok a könyvek (csak konyv_azon kell), amelyekhez 5-nél több példány tartozik?

```
select konyv_azon, count(*)
from konyvtar.konyvtari_konyv
group by konyv_azon
having count(*)>5;
```

95. Melyik témának kevesebb az összára 10000-nél?

```
select tema, sum(ar)
from konyvtar.konyv
group by tema
having sum(ar)<10000;
```

96. Témánként mennyi a legdrágább árú könyv ára?

```
select tema, max(ar)
from konyvtar.konyv
group by tema;
```

97. A 400 oldalnál kevesebb oldalszámú könyvek közül témánként mennyi a legolcsóbb könyvek ára.

```
select tema, min(ar)
from konyvtar.konyv
where oldalszam<400
group by tema;
```

98. Melyik kiadó adott ki annyi könyvet, amelynek az összoldalszáma több, mint 500?

```
select kiado
from konyvtar.konyv
group by kiado
having sum(oldalszam)>500;
```

99. Melyik az a hónap (év nélkül), amikor 1-nél több szerző született?

```
select to_char(születesi_datum,'mm'), count(*)
from konyvtar.szerzo
group by to_char(születesi_datum,'mm')
having count(*)>1;
```

100. A besorolásonkénti átlagszámmak mennyi az átlaga?

```
select avg(count(olvasojegyszam))
from konyvtar.tag
group by besorolas;
```

Select utasítás joinnal

101. Listázzuk ki a könyvek azonosítóit, a könyvek címeit és a könyvekhez kapcsolódó példányok leltári számait. (Csak azokat a könyveket és példányokat listázzuk, amelyeknek van a másik táblában megfelelője.)

- ```
select ko.konyv_azon, ko.cim, kk.konyv_azon, kk.leltari_szam
from konyvtar.konyv ko inner join konyvtar.konyvtari_konyv kk
on ko.konyv_azon=kk.konyv_azon;
```
- ```
select konyv_azon, ko.cim, kk.leltari_szam
from konyvtar.konyv ko inner join konyvtar.konyvtari_konyv kk
using (konyv_azon);
```
- ```
select konyv_azon, ko.cim, kk.leltari_szam
from konyvtar.konyv ko join konyvtar.konyvtari_konyv kk
using (konyv_azon);
```
- ```
select konyv_azon, ko.cim, kk.leltari_szam
from konyvtar.konyv ko, konyvtar.konyvtari_konyv kk
where ko.konyv_azon=kk.konyv_azon;
```

102. Mi a leltári száma az Ácsi Milán nevű tag aktuálisan kikölcsönzött könyveinek?

```
select leltari_szam
from konyvtar.tag tag inner join konyvtar.kolcsonzes k1
on tag.olvasojegyszam=k1.tag_azon
where vezeteknev='Ácsi' and keresztnév='Milán'
and visszahozasi_datum is null;
```

103. Milyen könyveket (azonosító és cím) kölcsönzött Ácsi Milán?

```
select kv.konyv_azon, kv.cim, to_char(ko.kolcsonzesi_datum,'yyyy.mm.dd')
from konyvtar.tag tg inner join konyvtar.kolcsonzes ko
on tg.olvasojegyszam=ko.tag_azon
inner join konyvtar.konyvtari_konyv kk
on ko.leltari_szam=kk.leltari_szam
inner join konyvtar.konyv kv
on kv.konyv_azon=kv.konyv_azon
where vezeteknev='Ácsi'
and keresztnév='Milán';
```

104. Listázzuk a horror témájú könyvekért kapott honoráriumokat.

```
select kv.konyv_azon, cim, honorarium
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
where tema='horror';
```

105. Ki írta a Hasznos holmik című könyvet?

```
select vezeteknev, keresztnév
from konyvtar.konyv kv inner join
konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
inner join konyvtar.szerzo sz
on ksz.szerzo_azon=sz.szerzo_azon
where cim='Hasznos holmik';
```

106. Mik a leltári számai a Tíz kicsi néger című könyvhöz tartozó példányoknak?

```
select leltari_szam
from konyvtar.konyv ko inner join konyvtar.konyvtari_konyv kk
on ko.konyv_azon=kk.konyv_azon
where cim='Tíz kicsi néger';
```

107. Mi a sci-fi, krimi, horror témájú könyvek címe és szerzőinek a neve?

```
select cim, vezeteknev||' '||keresztnév, tema
from konyvtar.konyv ko inner join konyvtar.konyvszerzo ksz
on ko.konyv_azon=ksz.konyv_azon
```



```

inner join konyvtar.szerzo sz
on ksz.szerzo_azon=sz.szerzo_azon
where tema in ('sci-fi', 'krimi', 'horror');

```

108. Mi a szerző azonosítója a sci-fi, krimi, horror témájú könyvek szerzőinek? Minden azonosítót csak egyszer listázzunk, a lista legyen rendezett.

```

select distinct szerzo_azon
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
where tema in ('sci-fi', 'krimi', 'horror')
order by szerzo_azon;

```

109. Mi a 40 évesnél fiatalabb olvasók által kikölcsönzött könyvek leltári száma?

```

select leltari_szam
from konyvtar.kolcsonzes kcs inner join konyvtar.tag tg
on kcs.tag_azon=tg.olvasojegyszam
where months_between(sysdate, tg.szuletési_datum)/12<40;

```

110. Mi a szerző azonosítója a Tíz kicsi néger szerzőjének?

```

select szerzo_azon
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
where cim='Tíz kicsi néger';

```

111. Mi az Agatha Christie által írt könyvek címe?

```

select cim
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon
where vezeteknev='Christie' and keresztnév='Agatha';

```

112. Keressünk olyan tagokat, akik hamarabb születtek, mint Agyalá Gyula.

```

o select tobbi.vezeteknev, tobbi.keresztnév,
       to_char(agygy.szuletési_datum, 'yyyy.mm.dd'),
       to_char(tobbi.szuletési_datum, 'yyyy.mm.dd')
from konyvtar.tag agygy, konyvtar.tag tobbi
where agygy.Vezeteknev='Agyalá' and agygy.keresztnév='Gyula'
and tobbi.szuletési_datum<agygy.szuletési_datum;

o select tobbi.vezeteknev, tobbi.keresztnév,
       to_char(agygy.szuletési_datum, 'yyyy.mm.dd'),
       to_char(tobbi.szuletési_datum, 'yyyy.mm.dd')
from konyvtar.tag agygy inner join konyvtar.tag tobbi
on tobbi.szuletési_datum<agygy.szuletési_datum
where agygy.Vezeteknev='Agyalá' and agygy.keresztnév='Gyula';

```

113. Melyik olvasó fiatalabb Agatha Christie írótól?

```

o select to_char(sz.szuletési_datum, 'yyyy.mm.dd'), to_char(t.szuletési_datum, 'yyyy.mm.dd'),
       t.vezeteknev, t.keresztnév, t.olvasojegyszam
from konyvtar.szerzo sz, konyvtar.tag t
where sz.vezeteknev='Christie'
and sz.keresztnév='Agatha'
and sz.szuletési_datum<t.szuletési_datum;

o select to_char(sz.szuletési_datum, 'yyyy.mm.dd'), to_char(t.szuletési_datum, 'yyyy.mm.dd'),
       t.vezeteknev, t.keresztnév, t.olvasojegyszam
from konyvtar.szerzo sz inner join konyvtar.tag t
on sz.szuletési_datum<t.szuletési_datum
where sz.vezeteknev='Christie'
and sz.keresztnév='Agatha';

```

114. Kik azok a tagok, akik ugyanabban a városban születtek, mint Agyalá Gyula?

```

o select tobbi.vezeteknev, tobbi.keresztnév, tobbi.OLVASOJEGYSZAM
from konyvtar.tag agygy, konyvtar.tag tobbi
where agygy.vezeteknev='Agyalá' and agygy.keresztnév='Gyula'
and substr(agygy.cim,6, instr(agygy.cim, ',')-6)=substr(tobbi.cim,6, instr(tobbi.cim, ',')-6);

o select tobbi.vezeteknev, tobbi.keresztnév, tobbi.OLVASOJEGYSZAM
from konyvtar.tag agygy inner join konyvtar.tag tobbi
on substr(agygy.cim,6, instr(agygy.cim, ',')-6)=substr(tobbi.cim,6, instr(tobbi.cim, ',')-6)
where agygy.vezeteknev='Agyalá' and agygy.keresztnév='Gyula';

```

115. Hogy hívják Neena Kochhar főnökét? Használjuk a HR sémát.

```

select fonok.first_name, fonok.last_name
from hr.employees nk inner join hr.employees fonok
on nk.manager_id=fonok.employee_id
where nk.first_name='Neena' and nk.last_name='Kochhar';

```

116. Írjuk ki az IT_PROG job_id-jú dolgozók főnökének a nevét. Használjuk a HR sémát.

```

select distinct fonok.first_name, fonok.last_name
from hr.employees it inner join hr.employees fonok
on it.manager_id=fonok.employee_id
where it.job_id='IT_PROG';

```

117. Az 5000-nél olcsóbb (árú) könyveknek hány különböző szerzője van?

```
select count(distinct szerzo_azon)
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
where ar<5000;
```

118. Listázzuk ki azokat a kiadókat, amelyek 1000000 kevesebb összhonoráriumot osztottak ki. A lista legyen rendezett.

```
select kiado, sum(honorarium)
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
group by kiado
having sum(nvl(honorarium,0))<1000000
order by kiado;
```

119. Listázzuk ki azokat a kiadókat, amelyek 1000000 kevesebb összhonoráriumot osztottak ki azoknak a szerzőknek, akik 1950 előtt születtek. A lista legyen rendezett.

```
select kiado, sum(honorarium)
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
inner join konyvtar.szerzo sz
on ksz.szerzo_azon=sz.szerzo_azon
where sz.születesi_datum<to_date('1950','yyyy')
group by kiado
having sum(nvl(honorarium,0))<1000000
order by kiado;
```

120. Keressük azokat a szerzőket, akik krimi, sci-fi, horror témájú könyvek megírásáért 2 milliónál több összhonoráriumot kaptak.

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnev, sum(honorarium)
from konyvtar.szerzo sz inner join
konyvtar.konyvszerzo ksz on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv kv on ksz.konyv_azon=kv.konyv_azon
where tema in ('krimi', 'sci-fi', 'horror')
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnev
having sum(honorarium)>2000000;
```

121. Mely szerzőknek nagyobb az összhonoráriumuk 1 milliónál?

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnev, sum(honorarium)
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnev
having sum(honorarium)>1000000;
```

122. Hány példány tartozik az egyes könyvekhez (azon, cím)? Csak azokat a könyveket listázzuk, amelyekhez tartozik példány.

```
select kv.cim, count(kk.leltari_szam), kv.konyv_azon
from konyvtar.konyv kv inner join konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
group by kv.cim, kv.konyv_azon
order by cim;
```

123. Az egyes szerzők hány könyvet írtak? Csak azokat a szerzőket listázzuk, akik írtak könyvet.

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnev, count(*)
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnev;
```

124. Listázzuk ki, hogy az egyes tagok hány könyvtári könyvet kölcsönöztek ki. Csak azokat a tagokat listázzuk, akik kölcsönöztek könyvet.

```
select vezeteknev, keresztnev, count(*), tg.olvasojegyszam
from konyvtar.tag tg inner join konyvtar.kolcsonzes ko
on tg.olvasojegyszam=ko.tag_azon
group by vezeteknev, keresztnev, tg.olvasojegyszam;
```

125. Kik (tag_azon) kölcsönözték ki azokat a példányokat, amelyek 3000-nél kevesebb értékűek?

```
select tag_azon
from konyvtar.kolcsonzes ko inner join
konyvtar.konyvtari_konyv kk
on ko.leltari_szam=kk.leltari_szam
where ertekek<3000;
```

126. Az egyes könyveknek hány szerzője van?

```
select ko.konyv_azon, cim, count(*), kiado, tema
from konyvtar.konyv ko inner join konyvtar.konyvszerzo ksz
on ko.konyv_azon=ksz.konyv_azon
group by ko.konyv_azon, cim, kiado, tema
order by cim;
```

127. Melyik könyvekre fizettek 1 milliónál több összhonoráriumot?

```
select cim, kv.konyv_azon, sum(honorarium)
from konyvtar.konyv kv inner join konyvtar.konyvszerzo ksz
on kv.konyv_azon=ksz.konyv_azon
group by cim, kv.konyv_azon
having sum(honorarium)>1000000
order by cim;
```

128. Kik azok a debreceni tagok, akik 1-nél több példányt kölcsönöztek ki?

```
select vezeteknev, keresztnev, olvasojegyszam, count(leltari_szam)
from konyvtar.tag tg inner join konyvtar.kolcsonzes kcs
on tg.olvasojegyszam=kcs.tag_azon
```

```
where substr(cim, 6, instr(cim, ',')-6)='Debrecen'
group by vezeteknev, keresztnév, olvasojegyszam
having count(legtari_szam)>1;
```

129. Kik azok a tagok (tag_azon), akik egy példányt legalább kétszer kölcsönöztek ki?

```
select tag_azon
from konyvtar.kolcsonzes
group by tag_azon, legtari_szam
having count(kolcsonzesi_datum)>1;
```

130. Kik azok a tagok, akik egy példányt legalább kétszer kölcsönöztek ki?

```
select tag_azon, tg.vezeteknev, tg.keresztnév
from konyvtar.kolcsonzes ko inner join konyvtar.tag tg
on ko.tag_azon=tg.olvasojegyszam
group by tag_azon, legtari_szam, tg.vezeteknev, tg.keresztnév
having count(kolcsonzesi_datum)>1
```

131. Ki írta a Napóleon című könyvet?

```
select vezeteknev, keresztnév
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon
where cim='Napóleon';
```

132. Listázzuk ki azokat a könyveket, amelyeknek az ára kevesebb, mint 5000, és van olyan szerzője, aki 20 évvel ezelőtt született, és 5000-nél több honoráriumot kapott a könyvírásért.

```
select cim, ko.konyv_azon
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon
where ar<5000
and add_months(sysdate, -20*12)>szuletési_datum
and honorarium>5000;
```

133. Melyik az a könyv, amely nem sci-fi, krimi és horror témájú, és amelyhez több, mint 3 példány tartozik?

```
select ko.konyv_azon, cim, count(kk.legtari_szam)
from konyvtar.konyv ko inner join konyvtar.konyvtari_konyv kk
on ko.konyv_azon=kk.konyv_azon
where tema not in ('sci-fi', 'krimi', 'horror')
group by ko.konyv_azon, cim
having count(kk.legtari_szam)>3;
```

134. Melyik szerző írt 3-nál kevesebb könyvet?

```
select sz.szerzo_azon, vezeteknev, keresztnév, count(konyv_azon)
from konyvtar.szerzo sz left outer join
konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, vezeteknev, keresztnév
having count(konyv_azon)<3;
```

135. Listázzuk ki az összes tagot, és ha kölcsönzött könyvet, akkor tegyük mellé az általa kölcsönzött könyv azonosítóját!

```
select tag.vezeteknev, tag.keresztnév, tag.olvasojegyszam, ko.legtari_szam, kk.konyv_azon
from konyvtar.tag tag left outer join konyvtar.kolcsonzes ko
on tag.olvasojegyszam=ko.tag_azon
left outer join konyvtar.konyvtari_konyv kk
on ko.legtari_szam=kk.legtari_szam;
```

136. Az egyes szerzők hány könyvet írtak?

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnév, count(konyv_azon)
from konyvtar.szerzo sz left outer join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnév;
```

137. Kik írtak 3-nál több könyvet?

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnév
from konyvtar.szerzo sz inner join
konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnév
having count(konyv_azon)>3;
```

138. Listázzuk témánként a horror, sci-fi, krimi témájú könyvekért kapott összhonoráriumot!

```
select tema, sum(nvl(honorarium,0))
from konyvtar.konyv ko left outer join konyvtar.konyvszerzo ksz
on ko.konyv_azon=ksz.konyv_azon
where tema in ('horror', 'sci-fi', 'krimi')
group by tema;
```

139. Kik írtak 3-nál kevesebb könyvet?

```
select sz.szerzo_azon, sz.vezeteknev, sz.keresztnév, count(konyv_azon)
from konyvtar.szerzo sz left outer join
konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
```

```
group by sz.szerzo_azon, sz.vezeteknev, sz.keresztnev
having count(konyv_azon)<3;
```

140. Listázzuk a PARK KÖNYVKIADÓ KFT. és a EURÓPA KÖNYVKIADÓ KFT. kiadójú könyvek összes könyvének azonosítóját, és ha van hozzá példány, akkor annak a leltári számát.

```
select kv.konyv_azon, cim, kk.leltari_szam
from konyvtar.konyv kv left outer join
    konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
where kiado in ('PARK KÖNYVKIADÓ KFT.', 'EURÓPA KÖNYVKIADÓ KFT.');
```

141. Az egyes könyvekhez hány könyvtári könyv tartozik?

```
select kv.konyv_azon, cim, count(kk.leltari_szam), count(*)
from konyvtar.konyv kv left outer join
    konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
group by kv.konyv_azon, cim;
```

142. Listázzuk ki az összes krimi témájú könyvet, és ha tartozik hozzájuk példány, akkor tegyük mellé a leltári számát.

```
select kv.konyv_azon, cim, kk.leltari_szam
from konyvtar.konyv kv left outer join konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
where kv.tema='krimi';
```

143. Melyik könyvtári könyveket kölcsönözték ki 3-nál kevesebbszer?

```
select kk.leltari_szam, count(kolcsonzesi_datum)
from konyvtar.konyvtari_konyv kk left outer join
    konyvtar.kolcsonzes ko
on kk.leltari_szam=ko.leltari_szam
group by kk.leltari_szam
having count(kolcsonzesi_datum)<3;
```

144. Mely könyvek valamely példányát kölcsönözték ki 3-nál kevesebbszer?

```
select kv.konyv_azon, cim, count(kolcsonzesi_datum)
from konyvtar.konyv kv inner join konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
left outer join
    konyvtar.kolcsonzes ko
on kk.leltari_szam=ko.leltari_szam
group by kv.konyv_azon, cim
having count(kolcsonzesi_datum)<3;
```

Select utasítás beágyazott lekérdezéssel

145. Melyek azok a szerzők, akik nem szereztek könyvet?

```
o select *
  from konyvtar.szerzo
 where szerzo_azon not in (select szerzo_azon
                           from konyvtar.konyvszerzo);

o select sz.szerzo_azon, sz.vezeteknev, sz.keresztnev
  from konyvtar.szerzo sz left outer join
    konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
 where ksz.konyv_azon is null;
```

146. Ki a legidősebb szerző?

```
select *
from konyvtar.szerzo
where szuletesi_datum=(select min(szuletesi_datum)
                        from konyvtar.szerzo);
```

147. Kérdezzük le a Napóleon című könyvek leltári számát!

```
o select leltari_szam
  from konyvtar.konyv kv inner join konyvtar.konyvtari_konyv kk
on kv.konyv_azon=kk.konyv_azon
 where cim='Napóleon';

o select leltari_szam
  from konyvtar.konyvtari_konyv
 where konyv_azon in (select konyv_azon
                       from konyvtar.konyv
                       where cim='Napóleon');

o select leltari_szam
  from konyvtar.konyvtari_konyv kk inner join (select konyv_azon
                                                from konyvtar.konyv
                                                where cim='Napóleon') kv
on kk.konyv_azon=kv.konyv_azon;
```

148. A diák besorolású tagok között ki a legidősebb?

```
select *
from konyvtar.tag
where szuletesi_datum=(select min(szuletesi_datum)
                        from konyvtar.tag);
```

```

        where besorolas='diák')
and besorolas='diák';

```

149. A női tagok között mi a legfiatalabb tagnak a neve?

```

select *
from konyvtar.tag
where szuletesi_datum=(select max(szuletesi_datum)
                        from konyvtar.tag
                        where nem='n')
and nem='n';

```

150. Témánként mi a legdrágább árú könyv címe?

```

o select ko.tema, cim
   from konyvtar.konyv ko inner join (select tema, max(ar) mar
                                     from konyvtar.konyv
                                     group by tema) m
   on ko.tema=m.tema and ko.ar=m.mar;

o select tema, cim
   from konyvtar.konyv
  where (tema, ar) in (select tema, max(ar)
                     from konyvtar.konyv
                     group by tema);

o select tema, cim
   from konyvtar.konyv ko
  where ar=(select max(ar)
            from konyvtar.konyv belso
            where belso.tema=ko.tema)

```

151. Mi a legdrágább értékű könyv címe?

```

select cim
from konyvtar.konyv
where konyv_azon in (select konyv_azon
                    from konyvtar.konyvtari_konyv
                    where erteke=(select max(erteke)
                                   from konyvtar.konyvtari_konyv));

```

152. Melyik könyvből nincs példány?

```

select *
from konyvtar.konyv
where konyv_azon not in (select konyv_azon from konyvtar.konyvtari_konyv);

```

153. A krimi témájú könyvekből melyik a legdrágább?

```

select *
from konyvtar.konyv
where tema='krimi' and ar=(select max(ar)
                           from konyvtar.konyv
                           where tema='krimi');

```

154. Melyik szerző kapta a legnagyobb összhonoráriumot?

```

select sz.szerzo_azon, vezeteknev, keresztnév, sum(nvl(honorarium,0))
from konyvtar.szerzo sz left outer join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, vezeteknev, keresztnév
having sum(nvl(honorarium,0)) =(select max(sum(nvl(honorarium,0)))
                              from konyvtar.szerzo sz left outer join konyvtar.konyvszerzo ksz
                              on sz.szerzo_azon=ksz.szerzo_azon
                              group by sz.szerzo_azon)

```

155. Melyik könyvhöz tartozik a legkevesebb példány?

```

select ko.konyv_azon, ko.cim, count(legtari_szam)
from konyvtar.konyv ko left outer join konyvtar.konyvtari_konyv kk
on ko.konyv_azon=kk.konyv_azon
group by ko.konyv_azon, ko.cim
having count(legtari_szam)=(select min(count(legtari_szam))
                           from konyvtar.konyv ko left outer join konyvtar.konyvtari_konyv kk
                           on ko.konyv_azon=kk.konyv_azon
                           group by ko.konyv_azon);

```

156. Keressük azokat a tagokat, akiknek van kölcsönzésük.

```

select *
from konyvtar.tag ta
where exists (select 1
             from konyvtar.kolcsonzes ko
             where ta.olvasojegyszam =ko.tag_azon);

```

157. Keressük azokat a tagokat, akik nem kölcsönöztek.

```

o select *
   from konyvtar.tag ta
  where not exists (select 1
                   from konyvtar.kolcsonzes ko
                   where ta.olvasojegyszam =ko.tag_azon);

o select *
   from konyvtar.tag t
  where t.olvasojegyszam not in (select tag_azon

```

```
from konyvtar.kolcsonzes);
```

```
o select t.olvasojegyszam, t.vezeteknev, t.keresztnev
from konyvtar.tag t left outer join
    konyvtar.kolcsonzes ko
    on t.olvasojegyszam=ko.tag_azon
where ko.leltari_szam is null;
```

158. Melyik szerző írta a legkevesebb könyvet?

```
o select sz.szerzo_azon, vezeteknev, keresztnev
from konyvtar.szerzo sz left outer join
    konyvtar.konyvszerzo ksz
    on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, vezeteknev, keresztnev
having count(konyv_azon)=(select min(count(konyv_azon))
                        from konyvtar.szerzo sz left outer join
                            konyvtar.konyvszerzo ksz
                            on sz.szerzo_azon=ksz.szerzo_azon
                        group by sz.szerzo_azon);

o with seged as (select sz.szerzo_azon, count(konyv_azon) db,
                        vezeteknev, keresztnev
from konyvtar.szerzo sz left outer join
    konyvtar.konyvszerzo ksz
    on sz.szerzo_azon=ksz.szerzo_azon
group by sz.szerzo_azon, vezeteknev, keresztnev)
select *
from seged
where db=(select min(db) from seged)
```

159. Témánként ki az utoljára kiadott könyv szerzője?

```
select ko.konyv_azon, tema, vezeteknev, keresztnev, kiadas_datuma
from konyvtar.konyv ko left outer join konyvtar.konyvszerzo ksz
on ko.konyv_azon=ksz.konyv_azon
left outer join konyvtar.szerzo sz
on ksz.szerzo_azon=sz.szerzo_azon
where (tema, kiadas_datuma) in (select tema, max(kiadas_datuma)
                                from konyvtar.konyv
                                group by tema);
```

160. Az egyes szerzők melyik könyvért kapták a legnagyobb honoráriumot?

```
select sz.vezeteknev, sz.keresztnev, honorarium, ko.cim
from konyvtar.szerzo sz left outer join
    konyvtar.konyvszerzo ksz
    on sz.szerzo_azon=ksz.szerzo_azon
left outer join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon
where (sz.szerzo_azon, nvl(honorarium,0)) in
    (select sz2.szerzo_azon, max(nvl(honorarium,0))
    from konyvtar.szerzo sz2 left outer join
        konyvtar.konyvszerzo ksz2
        on sz2.szerzo_azon=ksz2.szerzo_azon
    group by sz2.szerzo_azon)
order by vezeteknev, keresztnev;
```

161. Melyek azok a könyvek, amelyek az átlagos árú könyvektől olcsóbbak és amelyeket Móra Ferenc szerző írt?

```
select cim, ar
from konyvtar.szerzo sz inner join
    konyvtar.konyvszerzo ksz
    on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon
where ar<(select avg(ar) from konyvtar.konyv)
and vezeteknev='Móra'
and keresztnev='Ferenc';
```

162. Melyik az a séma, amely a legtöbb táblát tartalmazza (amelyekre van valamilyen jogunk)?

```
select owner, count(*)
from all_tables
group by owner
having count(*)=(select max(count(*))
                from all_tables
                group by owner);
```

Select utasítás rownummal

163. Listázzuk ki az első 10 legidősebb tag nevét!

```
select b.*, rownum
from (select vezeteknev, keresztnev, to_char(születesi_datum,'yyyy.mm.dd')
      from konyvtar.tag
      order by születesi_datum) b
where rownum<11;
```

164. Listázzuk ki az első 10 legdrágább könyvet.

```
select *
from (select *
      from konyvtar.konyv
```

```
order by ar desc nulls last)
where rownum<11;
```

165. Kérdezzük le azoknak a szerzőknek a nevét, akik benne vannak abban a listában, akik a 10 legnagyobb honoráriumot kapták.

```
select vezeteknev, keresztnév
from konyvtar.szerzo
where szerzo_azon in (select szerzo_azon
                      from (select *
                            from konyvtar.konyvszerzo
                            order by honorarium desc nulls last)
                      where rownum<11);
```

Select utasítás halmazművelettel

166. Listázzuk egy listában a szerzők keresztnévét és születési hónapját illetve a tagok keresztnévét és születési hónapját. Minden név, hónap páros annyiszor szerepeljen, ahányszor a táblákban előfordul.

```
select keresztnév, to_char(születesi_datum,'mm')
from konyvtar.szerzo
union all
select keresztnév, to_char(születesi_datum,'mm')
from konyvtar.tag;
```

167. Listázzuk egy listában a szerzők keresztnévét és születési hónapját illetve a tagok keresztnévét és születési hónapját. Minden név, hónap páros csak egyszer szerepeljen.

```
select keresztnév, to_char(születesi_datum,'mm')
from konyvtar.szerzo
union
select keresztnév, to_char(születesi_datum,'mm')
from konyvtar.tag;
```

168. Melyek azok a keresztnévek, amelyek szerző keresztnéve is és egyben valamelyik tagé is.

```
o select keresztnév
  from konyvtar.szerzo
 intersect
 select keresztnév
  from konyvtar.tag;

o select sz.keresztnév
  from konyvtar.szerzo sz inner join konyvtar.tag t
    on sz.keresztnév=t.keresztnév;
```

169. Melyek azok a keresztnévek, amelyek valamelyik szerző keresztnéve, de egyetlen tagé sem?

```
select keresztnév
from konyvtar.szerzo
minus
select keresztnév
from konyvtar.tag;
```

170. Tegyük egy listába a könyvek témáit és árait illetve a könyvek kiadóit és árait.

```
select tema, ar
from konyvtar.konyv
union
select kiado, ar
from konyvtar.konyv;
```

171. Tegyük egy listába a szerzők keresztnéveit, születési dátumait, besorolásukként null értéket feltüntetve és a tagok keresztnéveit, születési dátumait és besorolásait.

```
select keresztnév, születesi_datum, null besorolas
from KONYVTAR.SZERZO
union all
select keresztnév, születesi_datum, besorolas
from konyvtar.tag;
```

172. Tegyük egy listába a szerzők keresztnéveit, születési dátumait, besorolásukként 'szerző' értéket feltüntetve és a tagok keresztnéveit, születési dátumait és besorolásait.

```
select keresztnév, születesi_datum, 'szerző' be
from KONYVTAR.SZERZO
union all
select keresztnév, születesi_datum, besorolas
from konyvtar.tag;
```

DDL és DML utasítások

173. Hozzunk létre táblát személy néven a következő oszlopokkal:

- o azon: max. 5 számjegyű egész szám,
- o nev: maximum 30 hosszú változó hosszúságú karaktersorozat, amelyet ki kell tölteni,
- o szul_dat: dátum típusú,
- o irsz: pontosan 4 karakter hosszú sztring,
- o cim: maximum 40 hosszú változó hosszúságú karaktersorozat,
- o zsebpenn: szám, amelynek maximum 12 számjegye lehet, amelyből az utolsó kettő a tizedesvessző után áll.

A tábla elsődleges kulcsa az azon oszlop legyen. A nev, szul_dat, cim együtt legyen egyedi. A zsebpenn, ha ki van töltve, legyen nagyobb, mint 100.

```
create table szemely
(azon number(5),
nev varchar2(30) not null,
szul_dat date,
irsz char(4),
cim varchar2(40),
zsebpenn number(12,2),
constraint sz_pk primary key (azon),
constraint sz_uq unique (nev, szul_dat, cim),
constraint cs_ch check (zsebpenn>100));
```

174. Szűrjünk be a személy táblába három sort.

```
insert into szemely (azon, nev, szul_dat, irsz, cim, zsebpenn)
values (10,'Kiss Béla',
to_date('20100101 20:01','yyyymmdd hh24:mi'), '4028',
'Debrecen',null);

insert into szemely (azon, nev, szul_dat, irsz, cim, zsebpenn)
values (20,'Kiss Eszter',
to_date('20100101 20:01','yyyymmdd hh24:mi'), '4028',
'Debrecen',null);
commit;

insert into szemely (azon, nev, szul_dat, irsz, cim, zsebpenn)
values (30,'Nagy Tibor',
to_date('20100101 20:01','yyyymmdd hh24:mi'), '4028',
'Debrecen',120.32);
commit;
```

175. Hozzunk létre táblát bicikli néven a következő oszlopokkal:

- o azon: max. 5 számjegyű egész szám,
- o szin: maximum 20 hosszú változó hosszúságú karaktersorozat,
- o tulaj_azon: max. 5 számjegyű egész szám.

A tábla elsődleges kulcsa az azon oszlop legyen. A tulaj_azon hivatkozzon a személy tábla elsődleges kulcsára.

```
create table bicikli
(azon number(5),
szin varchar2(20),
tulaj_azon number(5),
constraint bi_pk primary key (azon),
constraint bi_fk foreign key (tulaj_azon) references szemely (azon));
```

176. Szűrjünk be három sort a bicikli táblába!

```
insert into bicikli (azon, szin, tulaj_azon)
values (100,'piros',5);

insert into bicikli (azon, szin, tulaj_azon)
values (110,'piros',null);

insert into bicikli (azon, szin, tulaj_azon)
values (200,'kék',20);
commit;
```

177. Listázzuk ki a személyek biciklijeit. A lista tartalmazza azokat a személyeket is, akiknek nincs biciklijük, és azokat a bicikliket is, amelyeknek nincs tulajdonosuk.

```
select *
from bicikli bi full outer join szemely sz
on bi.tulaj_azon=sz.azon;
```

178. Hozzunk létre táblát eldobom néven egy nev nevű oszloppal, amely maximum 30 hosszú változó hosszúságú karaktersorozat típusú.

```
create table eldobom
(nev varchar2(30));
```

179. Dobjuk el az eldobom táblát!

```
drop table eldobom;
```

180. Töröljük a piros színű bicikliket!

```
delete from bicikli
where szin='piros';
commit;
```

181. Módosítsuk a bicikli tábla szín oszlopában lévő értéket: fűzzünk hozzá egy '2'-es karaktert, annál a sornál, ahol az azon értéke 200!

```
update bicikli
set szin=szin||'2'
where azon=200;
commit;
```

182. Adjuk a bicikli táblához új oszlopot típus névvel és maximum 30 hosszú változó hosszúságú karaktersorozattal.

```
alter table bicikli
add (tipus varchar2(30));
```

183. Dobjuk el a bicikli tábla típus oszlopát!

```
alter table bicikli
drop column tipus;
```

184. Nevezzük át a bicikli tábla tulaj_azon oszlopát t_azon-ra.


```
alter table bicikli
rename column tulaj_azon to t_azon;
```

185. Módosítsuk a bicikli tábla szín oszlopának típusát varchar2(30)-ra!

```
alter table bicikli
modify (szin varchar2(30));
```

186. Nevezzük át a személy tábla cs_ch nevű megszorítását sz_ck-ra!

```
alter table személy
rename constraint cs_ch to sz_ck;
```

187. Nevezzük át a személy táblát személy2-re.

```
rename személy to személy2;
```

188. Dobjuk el a sz_uq nevű megszorítást a személy2 tábláról.

```
alter table személy2
drop constraint sz_uq;
```

189. Adjunk megszorítást a személy2 táblához: a név, születési dátum és a cím együtt legyen egyedi.

```
alter table személy2
add constraint sz_uq unique (nev,szul_dat,cim);
```

190. Dobjuk el a bicikli tábla elsődleges kulcsát!

```
alter table bicikli
drop primary key;
```

191. Legyen az azon oszlop a bicikli tábla elsődleges kulcsa!

```
alter table bicikli
add constraint bi_pk primary key (azon);
```

192. Hozzunk létre táblát, amely azt tartalmazza, hogy melyik szerző milyen című könyveket írt. A tábla tartalmazza a könyvek oldalankénti árát is.

```
create table szerzok_konyvek as
select cim, vezeteknev, keresztnév, ar/oldalszam ar_per_oldalszam
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon;
```

193. Hozzunk létre nézetet v_szerzo_konyv néven, amelyben azt listázzuk, hogy az egyes könyveknek kik a szerzői. A lista csak azokat a könyveket tartalmazza, amelyeknek van szerzője. A lista tartalmazza a könyvek oldalankénti árát is.

```
create view v_szerzo_konyv as
select cim, vezeteknev, keresztnév, ar/oldalszam ar_per_oldalszam
from konyvtar.szerzo sz inner join konyvtar.konyvszerzo ksz
on sz.szerzo_azon=ksz.szerzo_azon
inner join konyvtar.konyv ko
on ksz.konyv_azon=ko.konyv_azon;
```

194. Hozzunk létre nézete, amely a horror, sci-fi, krimi témájú könyvek címét, leltári számát és oldalankénti árát listázza.

```
create view v_felos_konyvek as
select cim, leltari_szam, ar/oldalszam ar_per_oldalszam
from konyvtar.konyv ko inner join
konyvtar.konyvtari_konyv kk
on ko.konyv_azon=kk.konyv_azon
where tema in ('krimi','sci-fi','horror');
```

195. Hozzunk létre nézetet legidosebb_szerzo néven, amely a legidősebb szerző nevét és születési dátumát listázza.

```
create view legidosebb_szerzo as
select vezeteknev||' '||keresztnév nev, to_char(születesi_datum,'yyyy.mm.dd') szul_dat
from konyvtar.szerzo
where születesi_datum=(select min(születesi_datum)
from konyvtar.szerzo);
```

DML utasítások

196. Vegyünk fel a könyvszerző táblába egy sort: Agatha Christie megírta a legdrágább könyvet, amiért 5000 ft honoráriumot kapott.

```
insert into konyvszerzo (szerzo_azon , konyv_azon, honorarium)
select szerzo_azon, konyv_azon, 5000
from konyvtar.szerzo, konyvtar.konyv
where vezeteknev='Christie'
and keresztnév='Agatha'
and ar=(select max(ar) from konyvtar.konyv);
commit;
```

197. Növeljük meg azon szerzők honoráriumát az általuk írt könyv árának a 10-szeresével, akik 1900 után születtek.

```
update konyvszerzo ksz
set honorarium=honorarium+(select ar*10
from konyvtar.konyv ko
where ksz.konyv_azon=ko.konyv_azon)
where szerzo_azon in (select szerzo_azon
from konyvtar.szerzo
where születesi_datum > '1900-01-01');
```

```

        where szuletesi_datum>to_date('1900','yyyy'));
commit;

```

198. Töröljük azokat a könyveket, amelyekhez nincs példány.

```

delete
from konyv
where konyv_azon not in (select konyv_azon
                        from konyvtar.konyvtari_konyv);
commit;

```

199. Töröljük azokat a konyvtari_konyveket, amelyeket nem kölcsönöztek ki.

```

delete
from konyvtari_konyv
where leltari_szam not in (select leltari_szam
                        from konyvtar.kolcsonzes);

commit;

```

200. Azokhoz könyvekhez kapcsolódó honoráriumot növeljük meg 10%-kal, amelyeknek az oldalankénti ára több, mint 20.

```

update konyvszerzo
set honorarium =honorarium*1.1
where konyv_azon in (select konyv_azon
                    from konyvtar.konyv
                    where ar/oldalszam>20);
commit;

```

201. Növeljük meg a honoráriumot a könyv árának a felével azon írásk esetén, ahol a szerző 1900 után születtek és a könyv kiadásának a nevében szerepel a KIADÓ szó.

```

update konyvszerzo ksz
set honorarium=honorarium+(select ar/2
                        from konyvtar.konyv ko
                        where ko.konyv_azon=ksz.konyv_azon)
where konyv_azon in (select konyv_azon
                    from konyvtar.konyv
                    where kiado like '%KIADÓ%')
and szerzo_azon in (select szerzo_azon
                    from konyvtar.szerzo
                    where szuletesi_datum>to_date('1900','yyyy'));

commit;

```

202. Növeljük meg azon könyvtári könyvek értékét 10%-kal, amelyeket 2000 után kölcsönözték ki, és amelyeknek a címében pontosan 2 db 'a' betű (mindegy, hogy kicsi vagy nagy) szerepel.

```

update konyvtari_konyv
set ertekek=ertekek*1.1
where leltari_szam in
(select leltari_szam
 from konyvtar.kolcsonzes
 where kolcsonzesi_datum>to_date('2000','yyyy'))
and konyv_azon in (select konyv_azon
                    from konyvtar.konyv
                    where lower(cim) like '%a%a%'
                    and lower(cim) not like '%a%a%a%');

commit;

```

203. Agatha Christie által írt összes könyvből vegyünk fel egy-egy új példányt a könyvtári könyv táblába. A példányok értéke egyezzen meg a könyvek árával. A könyvek leltári számát generáljuk a szerző monogramjából, a könyv_azonosítójából és a mai dátumból.

```

insert into konyvtari_konyv (konyv_azon, ertekek, leltari_szam)
select ko.konyv_azon, ar,
       substr(vezeteknev, 1,1)||substr(keresztnev,1,1)||ko.konyv_azon||to_char(sysdate,'yyyymmdd')
from konyvtar.konyv ko, konyvtar.konyvszerzo ksz, konyvtar.szerzo sz
where ko.konyv_azon=ksz.konyv_azon
and ksz.szerzo_azon=sz.szerzo_azon
and keresztnev='Agatha' and vezeteknev='Christie';
commit;

```

204. Módosítsuk azoknak a könyvtári könyveknek az értékét, amelyeknek az értéke nagyobb, mint a hozzátartozó könyv árának a fele. Az eredeti értéket csökkentjük azzal az értékkel, amelyet úgy számolunk ki, hogy a könyv árát elosztjuk a kiadás dátuma óta eltelt évek számával.

```

update konyvtari_konyv kk
set ertekek=ertekek-(select ar/(months_between(sysdate, kiadas_datuma)/12)
                    from konyvtar.konyv ko
                    where kk.konyv_azon=ko.konyv_azon)
where ertekek>(select ar*0.5 from konyvtar.konyv ko
               where kk.konyv_azon=ko.konyv_azon);

rollback;

```

205. Módosítsuk azon tagok beiratkozási dátumát, akiknek a beiratkozási dátuma későbbre esik, mint a legelső kölcsönzési dátuma. Az új beiratkozási dátuma legyen a legelső kölcsönzési dátuma.

```

update tag
set beiratkozas_i_datum=(select min(kolcsonzesi_datum)
                        from konyvtar.kolcsonzes kol
                        where tag.olvasojegyszam=kol.tag_azon)
where beiratkozas_i_datum>(select min(kolcsonzesi_datum)
                        from konyvtar.kolcsonzes kol
                        where tag.olvasojegyszam=kol.tag_azon);

```

206. Módosítsuk a könyvszerző táblát: az 1900 után született szerzők és azon könyvek esetén, amelyek ára több, mint 5000 növeljük meg a honorariumot a könyv árának a 70%-ával.

```
update konyvszerzo ksz
set honorarium=honorarium+(select ar*0.7
                             from konyvtar.konyv ko
                             where ksz.KONYV_AZON=ko.konyv_azon)
where szerzo_azon in (select szerzo_azon
                      from konyvtar.szerzo
                      where szuletesi_datum>to_date('1900','yyyy'))
and konyv_azon in (select konyv_azon
                  from konyvtar.konyv
                  where ar>5000);

commit;
```

207. Töröljük azokat a kölcsönzéseket, amelyben horror, sci-fi, vagy krimi témájú könyveket olyan női olvasók kölcsönöztek, akik Debrecenben laknak.

```
delete
from kolcsonzes
where leltari_szam in (select leltari_szam
                        from konyvtar.konyvtari_konyv
                        where konyv_azon in (select konyv_azon
                                              from konyvtar.konyv
                                              where tema in ('horror', 'sci-fi', 'krimi'))))
and tag_azon in (select olvasojegyszam
                from konyvtar.tag
                where nem='n'
                and substr(cim,6,instr(cim,',')-6)='Debrecen');

commit;
```

208. Töröljük azokat a könyvszerzőbeli sorokat, amelyek esetén a honorárium kisebb, mint a hozzá kapcsolódó könyv árának a 10-szerese.

```
delete
from konyvszerzo ksz
where honorarium<(select ar*10
                  from konyvtar.konyv ko
                  where ksz.konyv_azon=ko.konyv_azon);

commit;
```

209. Töröljük azokat a könyveket, amelyekhez nem tartozik konyvszerzo, és nem tartozik konyvtari_konyv, és 5000-nél olcsóbbak vagy nincs áruk.

```
delete
from konyv
where konyv_azon not in (select konyv_azon
                        from konyvtar.konyvszerzo)
and konyv_azon not in (select konyv_azon
                      from konyvtar.konyvtari_konyv)
and (ar is null or ar<5000);

commit;
```

210. A legidősebb tag ma kikölcsönözte a Napóleon című könyvhöz tartozó legdrágább példányt (mivel két Napóleon van, ezért mindkettőhöz egy-egy példányt). Vegyük fel a megfelelő sort a kölcsönzés táblába.

```
insert into kolcsonzes(kolcsonzesi_datum, tag_azon, leltari_szam)
select sysdate, olvasojegyszam, leltari_szam
from konyvtar.tag, konyvtar.konyvtari_konyv
where szuletesi_datum=(select min(szuletesi_datum)
                      from konyvtar.tag)
and konyv_azon in (select konyv_azon
                  from konyvtar.konyv
                  and (ertek, konyv_azon) in (select max(ertek), konyv_azon
                                              from konyvtar.konyvtari_konyv
                                              where konyv_azon in (select konyv_azon
                                                                    from konyvtar.konyv
                                                                    where cim='Napóleon')
                                              group by konyv_azon) );

commit;
```

211. A 2000 után kiadott könyvek árát növeljük a duplájára, az oldalszámát felezzük el.

```
update konyv
set ar=ar*2, oldalszam=oldalszam/2
where kiadas_datuma>=to_date('2000','yyyy');

commit;
```

212. Módosítsuk a kölcsönzés táblát a diákok esetén: a visszahozási dátum legyen kölcsönzési dátum megnövelve annyi nappal ahány évesek, a késedelmi díjat növeljük a könyv értékének a felével (ha null érték volt, akkor könyv értékének a fele legyen)!

```
update kolcsonzes ko
set visszahozasi_datum=kolcsonzesi_datum+
  (select sysdate-szuletesi_datum
   from konyvtar.tag
   where ko.tag_azon=olvasojegyszam),
kesedelmi_dij=nvl(kesedelmi_dij,0)+(select nvl(ertek,0)/2
                                   from konyvtar.konyvtari_konyv kk
                                   where ko.leltari_szam=kk.leltari_szam)
where tag_azon in (select olvasojegyszam
                  from konyvtar.tag
                  where besorolas='diák');

commit;
```

213. Móra Ferenc megírja az a könyvet, amelyikhez a legtöbb példány van. Szúrjunk be ennek megfelelően egy sort a könyvszerző táblába.

```
insert into konyvszerzo (szerzo_azon, konyv_azon)
select szerzo_azon, ko.konyv_azon
from konyvtar.szerzo,
     konyvtar.konyv ko left outer join
```

```
konyvtar.konyvtari_konyv kk  
on ko.konyv_azon=kk.konyv_azon  
where vezetekneve='Móra' and keresztnev='Ferenc'  
group by ko.konyv_azon, szerzo_azon  
having count(kk.leltari_szam)=(select max(count(leltari_szam))  
from konyvtar.konyv ko left outer join  
konyvtar.konyvtari_konyv kk  
on ko.konyv_azon=kk.konyv_azon  
group by ko.konyv_azon);
```

commit;