

```
SELECT * FROM emp;
SELECT * FROM dept;
```

1.1. Feladat

Listázza ki a 20-as részleg dolgozóinak nevét, belépési idejét, foglalkozását a nevek szerint csökkenően rendezve.

```
SELECT ename, hiredate, job
FROM emp
WHERE deptno=20
ORDER BY ename DESC;
```

1.2. Feladat

Készítsen két listát, melyek a dolgozók adatait tartalmazzák. Az egyiket a fizetés szerint növekvően, a másikat a fizetés szerint csökkenően rendezze.

```
SELECT *
FROM emp
ORDER BY sal ASC;
```

```
SELECT *
FROM emp
ORDER BY sal DESC;
```

1.3. Feladat

Listázza ki a dolgozók nevét, fizetést, jövedelmét a jövedelmük szerint csökkenően rendezve. (A jövedelem a fizetés és a jutalék összege.)

```
SELECT ename, sal, sal+NVL(comm,0) AS jov
FROM emp
ORDER BY jov ASC;
```

1.4. Feladat

Listázza ki a dolgozók nevét, részlegüket, jövedelmüket, és az adójukat (a jövedelmük 20%-a), az adójuk szerint csökkenően, a nevük szerint pedig növekvő módon rendezve.

```
SELECT ename, deptno, sal, sal*0.2 AS tax
FROM emp
ORDER BY tax DESC, ename ASC;
```

1.5. Feladat

Írassa ki azon alkalmazottak nevét, munkakörét és fizetését, akiknek fizetése nincs az 1500-2850 USD tartományban. A lista fejléce legyen "Név", "Munkakör", "Fizetés".

```
SELECT ename AS nev, job AS munkakor, sal AS fizetes
FROM emp
WHERE sal<1500 OR sal>2850;
```

1.6. Feladat

Írassa ki azon dolgozók nevét, munkakörét, fizetését, jutalékát és részleg-azonosítóját, akik 1000 USD-nál többet keresnek, és 1981. március 1. és szeptember 30. között léptek be a vállalathoz.

```
SELECT ename, job, sal, comm, deptno
FROM emp
WHERE sal>1000
AND hiredate BETWEEN '1981-03-01' AND '1981-09-30';
```

1.7. Feladat

Írassa ki minden jutalékkal rendelkező alkalmazott nevét, jutalékát, főnökének azonosítóját. Legyen a lista rendezett a főnök azonosítója, és az alkalmazottak neve szerint.

```
SELECT ename, comm, mgr
       FROM emp
      WHERE comm IS NOT NULL
      ORDER BY mgr, ename;
```

1.8. Feladat

Írassa ki azon alkalmazottak azonosítóját, nevét, foglalkozását, fizetését és jutalékát, akiknek jutaléka meghaladja a fizetésük 50%-át.

```
SELECT empno, ename, job, sal, comm
       FROM emp
      WHERE comm>sal/2;
```

1.9. Feladat

Írja ki azon dolgozók nevét, foglalkozását, fizetését és belépési dátumát, akik 1981-ben léptek be a vállalathoz. A lista legyen a belépési dátum szerint rendezve.

```
SELECT ename, job, sal, hiredate
       FROM emp
      WHERE hiredate BETWEEN '1981-01-01' AND '1981-12-31'
      ORDER BY hiredate;
```

1.10. Feladat

Listázza azon alkalmazottak nevét, foglalkozását, jövedelmét, akiknek a nevében két "L" betű szerepel, továbbá vagy a 30-as részlegen dolgozik, vagy a főnökének azonosítója 7782.

```
SELECT ename, job, sal
       FROM emp
      WHERE ename LIKE '%L%L%'
            AND (deptno=30 OR mgr=7782);
```

1.11. Feladat

Listázza ki részlegazonosító szerint rendezve a clerk és a salesman munkakörű dolgozók éves fizetését a részleg szerint rendezve.

```
SELECT sal*12
       FROM emp
      WHERE job='CLERK' OR job='SALESMAN'
      ORDER BY deptno ASC;
```

1.12. Feladat

Listázza ki az összes dolgozót oly módon, hogy azoknál, akik nem kapnak jutalékot, az a szöveg jelenjen meg, hogy "Nincs jutalék". A lista fejléce legyen azonosító, belépési dátum, név, foglalkozás, jutalék.

```
SELECT empno AS azonosito, hiredate AS "belepesi datum", ename AS nev, job AS foglalkozas,
       CASE
          WHEN comm IS NULL THEN 'Nincs jutalek'
          ELSE TO_CHAR(comm)
       END AS jutalek
       FROM emp;
```

1.13. Feladat

Listázza ki a man karaktersorozatot tartalmazó munkakörben dolgozók nevét és munkakörét, a munkakör és a név szerint rendezve.

```
SELECT ename, job
      FROM emp
     WHERE job LIKE '%MAN%'
     ORDER BY ename, job;
```

1.14. Feladat

Listázza foglalkozás szerint csoportosítva azon dolgozók nevét, foglalkozását, jövedelmét és részlegét, akiknek jövedelme kisebb 2500 USD-nál, valamint 1981 és 1982 között léptek be. A keletkezett lista elsődlegesen a foglalkozás, másodlagosan a dolgozó neve szerint legyen rendezve.

```
SELECT ename, job, sal, deptno
      FROM emp
     WHERE sal < 2500
           AND hiredate LIKE '1981%'
     ORDER BY job, ename;
```

1.15. Feladat

Listázza ki azoknak az alkalmazottaknak a nevét, éves fizetését és a munkában eltöltött hónapjainak számát, akik 1981.07.01. előtt léptek be a vállalathoz. A lista legyen a hónapszámok szerint csökkenően rendezve.

```
SELECT ename, sal*12, TRUNC(MONTHS_BETWEEN(CURRENT_DATE, hiredate))
      FROM emp
     WHERE hiredate < '1981-07-01'
     ORDER BY hiredate DESC;
```

1.16. Feladat

Listázza ki a C és az M betűvel kezdődő foglalkozású alkalmazottak nevét (nevüket nagy betűvel kezdve és kisbetűvel folytatva), valamint nevük hosszát. Rendezze a listát a foglalkozás szerint.

```
SELECT CONCAT(SUBSTR(ename,0,1), LOWER(SUBSTR(ename,2,256))) AS nev, LENGTH(ename)
      FROM emp
     WHERE job LIKE 'C%' OR job LIKE 'M%';
```

1.17. Feladat

A belépési dátum napjai szerint csoportosítva listázza azon dolgozók azonosítóját, nevét, jövedelmét, munkába állásuk napját, részlegét, akiknek jövedelme 1300 és 5500 USD közötti érték. A keletkezett lista elsődlegesen a napok sorszáma szerint, másodlagosan a dolgozó neve szerint legyen rendezve. A hét első napja legyen a vasárnap.

```
SELECT empno, ename, sal, hiredate, deptno
      FROM emp
     WHERE sal BETWEEN 1300 AND 5500
     ORDER BY hiredate, ename;
```

1.18. Feladat

A vállalatnál hűségjutalmat adnak, és ehhez szükséges azon dolgozók azonosítója, neve, fizetése, munkában eltöltött éve, akik legalább 15 éve álltak munkába. Rendezze a listát a munkában eltöltött évek szerint csökkenően, valamint az azonosító szerint növekvően.

```
SELECT empno, ename, sal, years
      FROM
        (SELECT empno, ename, sal,
                TRUNC(MONTHS_BETWEEN(CURRENT_DATE, hiredate)/12) AS years
          FROM emp)
     WHERE years >= 15
     ORDER BY years DESC, empno ASC;
```

1.19. Feladat

Listázza a dolgozók nevét, munkakörét, fizetését és a fizetési kategóriáját, mely 1000 USD alatt 1, 2000 USD alatt 2, stb., ez utóbbi szerint csökkenően rendezve. (A fizetéstartomány 1..6000 USD.)

```
SELECT ename, job, sal,
       CASE
         WHEN sal<1000 THEN '1'
         WHEN sal<2000 THEN '2'
         WHEN sal<3000 THEN '3'
         WHEN sal<4000 THEN '4'
         WHEN sal<5000 THEN '5'
         WHEN sal<6000 THEN '6'
       END AS cat
FROM emp
ORDER BY cat DESC;
```

1.20. Feladat

Listázza a dolgozók nevét, azonosítóját és beosztását a név szerint rendezve. Egy dolgozó beosztása "Főnök", ha van beosztottja, egyébként NULL érték.

```
SELECT ename, empno,
       CASE
         WHEN empno IN (SELECT mgr FROM emp) THEN 'FŐNÖK'
         ELSE NULL
       END AS beoszt
FROM emp
ORDER BY ename;
```

2.1. Feladat

Listázza munkakörönként az átlagfizetéseket két tizedesre kerekítve. Rendezze átlagfizetések szerint csökkenően.

```
SELECT job, ROUND(AVG(sal), 2) AS avgsal
FROM emp
GROUP BY job
ORDER BY avgsal DESC;
```

2.2. Feladat

Listázza csökkenően rendezve a főnökök átlagfizetését egész értékre kerekítve. (Főnök az a dolgozó, akinek azonosítója szerepel az mgr oszlopban.)

```
SELECT ROUND(AVG(sal))
FROM emp
WHERE empno IN (SELECT mgr FROM emp);
```

2.3. Feladat

Listázza részlegenként a legnagyobb és legkisebb havi jövedelmeket.

```
SELECT deptno, MIN(sal), MAX(sal)
FROM emp
GROUP BY deptno;
```

2.4. Feladat

Listázza a legalább egy dolgozójú részlegeket a dolgozószám szerint csökkenően rendezve.

```
SELECT deptno, COUNT(deptno) AS szam
FROM emp
GROUP BY deptno
ORDER BY szam DESC;
```

2.5. Feladat

Listázza ki a főnökök azonosítóit, valamint azt, hogy hány beosztottjuk van. Rendezze a listát a beosztottak száma szerint csökkenően. Akinek nincs főnöke, oda írjon valamilyen megjegyzést (tulajdonos vagy elnök stb.).

```
SELECT
  CASE
    WHEN mgr IS NULL THEN 'Boss'
    ELSE TO_CHAR(mgr,'9999')
  END AS mgr,
  COUNT(mgr) AS szolga
FROM emp
GROUP BY mgr
ORDER BY szolga DESC;
```

2.6. Feladat

Listázza az azonosítójuk hárommal való oszthatósága alapján a dolgozók átlagjövedelmét, dolgozók számát, és legkisebb fizetését.

```
SELECT ROUND(AVG(sal),2),
  CASE
    WHEN MOD(empno,3)=0 THEN '0'
    WHEN MOD(empno,3)=1 THEN '1'
    WHEN MOD(empno,3)=2 THEN '2'
  END AS m,
  MIN(sal)
FROM emp
GROUP BY MOD(empno,3);
```

2.7. Feladat

Listázza a 2000 USD-nál nagyobb átlagjövedelmeket egész értékre kerekítve a foglalkozás szerint csoportosítva. A lista a foglalkozás szerint legyen rendezett.

```
SELECT job, ROUND(AVG(sal),0) AS atlag
FROM emp
GROUP BY job HAVING ROUND(AVG(sal),0)>=2000
ORDER BY job;
```

2.8. Feladat

Listázza azokat a részlegeket, ahol a fizetésátlag nagyobb 1500 USD-nál. Rendezze fizetésátlag szerint csökkenően.

```
SELECT deptno, ROUND(AVG(sal),0) AS atlag
FROM emp
GROUP BY deptno HAVING ROUND(AVG(sal),0)>=1500
ORDER BY atlag DESC;
```

2.9. Feladat

Listázza foglalkozásonként a legnagyobb jövedelmeket, jövedelem szerint rendezve.

```
SELECT job, MAX(sal) AS max
FROM emp
GROUP BY job
ORDER BY max;
```

2.10. Feladat

Listázza ki, hogy az egyes foglalkozási csoportokon belül hányan dolgoznak. A lista a létszám szerint legyen rendezett.

```
SELECT job, COUNT(job) AS letszam
FROM emp
GROUP BY job
ORDER BY letszam;
```

2.11. Feladat

Listázza ki a főnökök azonosítóit és a főnökökhöz tartozó beosztottak számát, ez utóbbi adat szerint rendezve.

```
SELECT mgr, COUNT(mgr) AS szolga
      FROM emp
      GROUP BY mgr
      ORDER BY szolga;
```

2.12. Feladat

Listázza azon foglalkozások átlagjövedelmét, amelyek nevében a man alsztring megtalálható. A listát rendezze az átlagjövedelem szerint csökkenő sorrendben.

```
SELECT job, ROUND(AVG(sal),0) AS atlag
      FROM emp
      WHERE job LIKE '%MAN%'
      GROUP BY job
      ORDER BY atlag DESC;
```

2.13. Feladat

Listázza rendezve azon foglalkozási csoportok átlagfizetését, ahol kettő, vagy ennél több alkalmazott dolgozik.

```
SELECT job, ROUND(AVG(sal),0) AS atlag
      FROM emp
      GROUP BY job HAVING COUNT(job)>=2;
```

2.14. Feladat

Írjon utasítást azon részlegek azonosítójának, dolgozói számának és azok legnagyobb és legkisebb jövedelmének lekérdezésére, ahol a részlegszám páros. A lista a részleg azonosító szerint legyen rendezve.

```
SELECT deptno, COUNT(empno), MAX(sal), MIN(sal)
      FROM emp
      WHERE MOD(deptno,2)=0
      GROUP BY deptno
      ORDER BY deptno;
```

2.15. Feladat

Listázza ki az azonosító paritása szerint csoportosítva a dolgozókat. Hagyja ki azon dolgozókat, akik 1981 után léptek be a vállalathoz. Rendezze elsődlegesen paritás szerint, másodlagosan a dolgozó neve szerint.

```
SELECT ename,
      CASE
        WHEN MOD(empno,2)=0 THEN 'Paros'
        ELSE 'Paratlan'
      END AS paritas
      FROM emp
      WHERE hiredate<'1982-01-01'
      ORDER BY paritas, ename;
```

2.16. Feladat

Számítsa ki az átlagos jutalékot.

```
SELECT AVG(NVL(comm,0)) FROM emp;
```

2.17. Feladat

Készítsen listát a páros és páratlan azonosítójú dolgozók számáról.

```
SELECT
      SUM(CASE WHEN MOD(empno,2)=0 THEN 1 ELSE 0 END) AS paros,
      SUM(CASE WHEN MOD(empno,2)=1 THEN 1 ELSE 0 END) AS paratlan
      FROM emp;
```

2.18. Feladat

Listázza fizetési kategóriák szerint a dolgozók számát. (A fizetési kategóriákat vagy Ön definiálja, vagy vegye a salgrade táblából.)

```
SELECT
    SUM(CASE WHEN sal<1000 THEN 1 ELSE 0 END) AS "<1000",
    SUM(CASE WHEN sal BETWEEN 1000 AND 3000 THEN 1 ELSE 0 END) AS "1000<x<3000",
    SUM(CASE WHEN sal>3000 THEN 1 ELSE 0 END) AS ">3000"
FROM emp;
```

2.19. Feladat

Listázza főnökönként (mgr) a főnökhöz tartozó legkisebb dolgozói fizetéseket. Hagyja ki azon dolgozók fizetését, akiknek nincs főnökük, valamint azokat a csoportokat, ahol a legkisebb fizetés nagyobb 2000 USD-nál. Rendezze a listát a legkisebb fizetések szerint növekvően.

```
SELECT mgr, MIN(sal)
FROM emp
WHERE mgr IS NOT NULL
GROUP BY mgr HAVING MIN(sal)<2000
ORDER BY MIN(sal) ASC;
```

2.20. Feladat

Listázza főnökönként (mgr) a főnökhöz tartozó dolgozói átlagfizetéseket. Hagyja ki azon dolgozók fizetését, akiknek nincs főnökük, valamint azokat a csoportokat, ahol az átlagfizetés nagyobb 3000 USD-nál. Rendezze a listát az átlagfizetések szerint csökkenően.

```
SELECT mgr, AVG(sal)
FROM emp
WHERE mgr IS NOT NULL
GROUP BY mgr HAVING AVG(sal)<3000
ORDER BY AVG(sal) DESC;
```

2.21. Feladat

Listázza főnökönként a főnökhöz tartozó dolgozók jövedelme közül a legnagyobbat. Hagyja ki a listakészítésből azon dolgozókat, akiknek nincs jutalékuk, valamint azokat a (legnagyobb) jövedelemeket, melyek nagyobbak 3500 USD-nál. Rendezze a listát a legnagyobb jövedelem szerint csökkenően.

```
SELECT mgr, MAX(sal)
FROM emp
WHERE comm IS NOT NULL
GROUP BY mgr HAVING MAX(sal)<3500
ORDER BY MAX(sal) DESC;
```

2.22. Feladat

Listázza részlegenként az egy tizedesre kerekített átlagfizetéseket. Hagyja ki az átlag meghatározásból az 1981. január. 1-e előtt belépett dolgozókat, valamint azon részlegek átlagfizetését, melyekben a legkisebb fizetés kisebb 1000 USD-nál. Rendezze a listát az átlagfizetések szerint növekvően.

```
SELECT deptno, ROUND(AVG(sal),1)
FROM emp
WHERE hiredate>'1981-01-01'
GROUP BY deptno HAVING MIN(sal)>1000
ORDER BY AVG(sal) ASC;
```

2.23. Feladat

Listázza munkakörönként a dolgozók számát és az egész értékre kerekített átlagfizetésüket numerikusan és grafikusán is. Ez utóbbit csillag (*) karakterek sorozataként balra igazítva jelenítse meg oly módon, hogy e sorozatban 200 USD-onként egy csillag karakter álljon. Rendezze a listát az átlagfizetések szerint csökkenően.

```
SELECT job, COUNT(empno), ROUND(AVG(sal)), LPAD('*', TRUNC(AVG(sal))/200, '*')
      FROM emp
      GROUP BY job
      ORDER BY ROUND(AVG(sal)) DESC;
```

2.24. Feladat

Listázza főnökönként a legrégebb óta munkaviszonyban álló dolgozóknak a mai napig munkában töltött éveinek számát numerikusan és grafikusán is. Ez utóbbit kettőskereszt (#) karakterek sorozataként balra igazítva jelenítse meg oly módon, hogy e sorozatban 5 évenként egy kettőskereszt karakter álljon. Rendezze a listát az évek száma szerint növekvően.

```
SELECT mgr,
      EXTRACT(YEAR FROM SYSDATE)-EXTRACT(YEAR FROM MIN(hiredate)) AS evekszama,
      LPAD('#', (EXTRACT(YEAR FROM SYSDATE)-EXTRACT(YEAR FROM MIN(hiredate)))/5, '#') AS graf
      FROM emp
      GROUP BY mgr
      ORDER BY evekszama ASC;
```

3.1. Feladat

Listázza azon dolgozók nevét és részlegük nevét, akiknek nevében az A betű szerepel.

```
SELECT ename, dname
      FROM emp INNER JOIN dept USING(deptno)
      WHERE ename LIKE '%A%';
```

3.2. Feladat

Listázza ki a Dallas-i telephely minden dolgozójának nevét, munkakörét, fizetését és részlegének azonosítóját.

```
SELECT ename, job, sal, deptno
      FROM emp INNER JOIN dept USING(deptno)
      WHERE loc='DALLAS';
```

3.3. Feladat

Listázza ki a clerk munkakörű dolgozókat foglalkoztató részleg azonosítóját, nevét és telephelyét. A lista legyen rendezve a részlegnév szerint.

```
SELECT deptno, dname, loc
      FROM emp INNER JOIN dept USING(deptno)
      WHERE job='CLERK'
      ORDER BY dname;
```

3.4. Feladat

Listázza ki a Dallas-ban és a Chicago-ban dolgozók nevét, munkakörét és telephelyét. A lista telephely szerint legyen rendezett.

```
SELECT ename, job, loc
      FROM emp INNER JOIN dept USING(deptno)
      WHERE loc='DALLAS' OR loc='CHICAGO'
      ORDER BY loc;
```


3.5. Feladat

Listázza ki az egyes részlegek nevét, telephelyük címét, dolgozóik átlagfizetését a részlegnevek szerint rendezve.

```
SELECT dname, loc, AVG(sal)
      FROM emp INNER JOIN dept USING(deptno)
      GROUP BY dname, loc
      ORDER BY dname;
```

3.6. Feladat

Listázza ki a 20-as és a 30-as részleg legnagyobb fizetésű dolgozóinak azonosítóját, nevét, foglalkozását, jutalékát és belépési dátumát.

```
SELECT empno, ename, job, NVL(comm,0), hiredate
      FROM dept INNER JOIN emp USING(deptno),
      (SELECT MAX(sal) AS msal
       FROM dept INNER JOIN emp USING(deptno)
       GROUP BY deptno)
      WHERE (deptno=20 OR deptno=30)
      AND sal=msal;
```

3.7. Feladat

Listázza ki minden részleg legkisebb jövedelmű dolgozójának azonosítóját, nevét, foglalkozását, jutalékát és belépési dátumát.

```
SELECT empno, ename, job, NVL(comm,0), hiredate
      FROM dept INNER JOIN emp USING(deptno),
      (SELECT MIN(sal) AS msal
       FROM emp INNER JOIN dept USING(deptno)
       GROUP BY deptno)
      WHERE sal=msal;
```

3.8. Feladat

Listázza ki azon részlegek nevét és telephelyét, ahol a dolgozók átlagjövedelme kisebb, mint 2200 USD.

```
SELECT dname, loc
      FROM dept INNER JOIN emp USING(deptno)
      GROUP BY deptno, dname, loc HAVING AVG(sal)<2200;
```

3.9. Feladat

Írjon olyan lekérdezést, ami megadja az összes jutalékkal rendelkező alkalmazott nevét, részlegének nevét és helyét.

```
SELECT ename, dname, loc
      FROM emp INNER JOIN dept USING(deptno)
      WHERE comm IS NOT NULL;
```

3.10. Feladat

Listázza a dolgozók nevét és azonosítóját a főnökük (mgr) nevével és azonosítójával együtt úgy, hogy akinek nincs főnöke, annak a NULL érték helyére a "Legfőbb" karaktersorozatot írja.

```
SELECT x.ename, x.empno,
      CASE
        WHEN y.ename IS NULL THEN 'Ajm do bossz'
        ELSE y.ename
      END AS mgrname,
      y.empno AS mgrno
      FROM emp x, emp y
      WHERE x.mgr=y.empno(+);
```

3.11. Feladat

Listázza ki a NEW YORK telephely minden dolgozójának nevét, azonosítóját, jövedelmét és főnökének nevét, telephelyét.

```
SELECT x.ename, x.empno, x.sal, y.ename, y.loc
FROM
    (SELECT *
     FROM emp INNER JOIN dept USING(deptno)) x,
    (SELECT *
     FROM emp INNER JOIN dept USING(deptno)) y
WHERE x.loc='NEW YORK'
AND x.mgr=y.empno(+);
```

3.12. Feladat

Listázza mindazon alkalmazott nevét, részlegének nevét és fizetését, akiknek fizetése megegyezik valamelyik Dallasban dolgozó alkalmazottéval. Legyen a lista fejléce név, részlegnév, fizetés, és a lista legyen a fizetés és a részlegnév szerint rendezett.

```
SELECT DISTINCT ename AS nev, dname AS reszlegnev, sal AS fizetes
FROM emp INNER JOIN dept USING(deptno),
    (SELECT sal AS dsal
     FROM emp INNER JOIN dept USING(deptno)
     WHERE loc='DALLAS')
WHERE sal IN dsal
ORDER BY fizetes, reszlegnev;
```

3.13. Feladat

Listázza azon dolgozókat, akiknek neve *hasonlít* egy munkakör nevéhez.

```
SELECT DISTINCT x.ename
FROM emp x, emp y
WHERE SOUNDEX(x.ename) IN SOUNDEX(y.job);
```

3.14. Feladat

Listázza azon főnökök azonosítóját, akik nem menedzser foglalkozásúak. A lista legyen a főnök azonosító (mgr) szerint rendezett.

```
SELECT x.empno, x.ename
FROM emp x, emp y
WHERE x.empno IN y.mgr
AND x.job <> 'MANAGER'
GROUP BY x.empno, x.ename
ORDER BY x.empno;
```

3.15. Feladat

Hány olyan főnök van, aki nem menedzser foglalkozású?

```
SELECT COUNT(z.empno)
FROM
    (SELECT x.empno
     FROM emp x, emp y
     WHERE x.empno IN y.mgr
     AND x.job <> 'MANAGER'
     GROUP BY x.empno) z;
```

3.16. Feladat

Listázza a főnökeik szerint csoportosítva a legkisebb jövedelmű dolgozókat. Hagyja ki azon dolgozókat, akiknek nincs főnökük, valamint azokat a csoportokat, ahol a legkisebb jövedelem nagyobb 3000 USD-nál. Rendezze a listát a legkisebb jövedelmek szerint növekvően.

```
SELECT MIN(sal), mgr
FROM emp
WHERE mgr IS NOT NULL
GROUP BY mgr HAVING MIN(sal)<3000
ORDER BY MIN(sal) ASC;
```

3.17. Feladat

Listázza minden olyan dolgozó azonosítóját és nevét, akik olyan részlegben dolgoznak, melyen található nevében T betűt tartalmazó dolgozó. Legyen a lista fejléce azonosító, név, részleg helye, és a lista legyen a részleg helye és a név szerint rendezett.

```
SELECT DISTINCT x.empno AS azonosito, x.ename as nev, x.loc as "részleg helye"
FROM
    (SELECT *
     FROM emp INNER JOIN dept USING(deptno)) x,
    (SELECT deptno
     FROM emp INNER JOIN dept USING(deptno)
     WHERE ename LIKE '%T%') y
WHERE x.deptno IN y.deptno
ORDER BY "részleg helye", nev;
```

3.18. Feladat

Listázza a főnökeik (mgr) szerint csoportosítva azokat a dolgozókat, akiknek fizetése e csoportosítás szerint a legkisebb, de nagyobb 1000 USD-nál. A lista a fizetés növekvő értéke szerint legyen rendezett. Legyen a lista fejléce: főnök kódja, dolgozónév, fizetése.

```
SELECT DISTINCT y.mgr, x.ename, y.msalsal
FROM emp x,
    (SELECT mgr, MIN(sal) AS msalsal
     FROM emp
     GROUP BY mgr HAVING MIN(sal)>1000
     ORDER BY msalsal) y
WHERE y.mgr=x.mgr;
```

3.19. Feladat

Listázza azon főnököknél (mgr) a legkisebb és legnagyobb fizetéseket, melyeknél a legkisebb fizetések 3000 USD-nál alacsonyabbak. A listát a legkisebb fizetés szerint rendezze, a fejléc pedig legyen "Főnök kódja", "Legkisebb Fizetés" és "Legnagyobb Fizetés".

```
SELECT DISTINCT y.mgr AS "fonok kodja", min AS "legkisebb fizetes", max AS "legnagyobb fizetes"
FROM emp x,
    (SELECT mgr, MIN(sal) AS min, MAX(sal) AS max
     FROM emp
     GROUP BY mgr HAVING MIN(sal)<3000) y
WHERE y.mgr=x.mgr
ORDER BY min;
```

3.20. Feladat

Listázza a fizetés szerint csökkenően rendezve az eladók (salesman) és a hivatalnokok (clerk) főnökeinek nevét és fizetését, a saját nevét, munkakörét, fizetését, valamint a saját fizetés – főnök fizetés arányukat a főnök neve szerint elsődlegesen, a fizetés arány szerint másodlagosan rendezve.

```
SELECT x.ename AS mgrname, x.sal AS mgrsal, y.ename AS name, y.job AS job, y.sal AS sal, y.sal/x.sal AS arany
FROM emp x, emp y
WHERE (y.job='SALESMAN' OR y.job='CLERK')
      AND y.mgr=x.empno
ORDER BY mgrname, arany;
```