

**-- 1 Obtener los nombres y salarios de los empleados con más de 1000 euros de salario por orden alfabético.**

```
SELECT
    nombre,
    ape1,
    ape2,
    salario
FROM
    empleado
WHERE
    salario > 1000
ORDER BY
    ape1,
    nombre;
```

**-- 2 Obtener el nombre de los empleados cuya comisión es superior al 20% de su salario.**

```
SELECT
    nombre,
    ape1,
    ape2,
    comision,
    salario
FROM
    empleado
WHERE
    comision > (0.2 * salario) ;
```

**-- 3 Obtener el código de empleado, código de departamento, nombre y sueldo total en pesetas de aquellos empleados cuyo sueldo total (salario + comisión) supera los 1800 euros. Presentarlos ordenados por su código de departamento y dentro de estos, por orden alfabético**

```
SELECT
    codemple,
    coddpto,
    nombre,
    ape1,
    ape2,
    ( salario + nvl(comision, 0) ) * 166.386 AS sueldoopts
FROM
    empleado
WHERE
    salario + nvl(comision, 0) > 1800
ORDER BY
    coddpto,
    ape1,
    ape2,
    nombre;
```

**-- 4 Obtener por orden alfabético los nombres de empleados cuyo salario igualen o superen en más de un 5% al salario de la empleada "MARIA JAZMIN"**

```
SELECT
    nombre,
    ape1,
    salario
FROM
    empleado
WHERE
    salario >= (
        SELECT
            salario * 1.05
        FROM
            empleado
        WHERE
            ape1 = 'JAZMIN'
            AND nombre = 'MARIA'
    )
ORDER BY
    ape1,
    nombre;
```

**-- 5 Obtener una listado ordenado por años en la empresa con los nombres, y apellidos de los empleados y los años de antigüedad en la empresa.**

```
SELECT
    nombre,
    ape1,
    ape2,
    months_between(sysdate, fechaingreso) / 12 AS antigüedad
FROM
    empleado
ORDER BY
    antigüedad;
```

**-- 6 Obtener el nombre de los empleados que trabajan en un departamento con presupuesto superior a 50.000 euros. Hay que usar predicado cuantificado**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    coddpto = SOME (
        SELECT
            coddpto
        FROM
            dpto
        WHERE
            presupuesto > 50000
    );
```

**-- 7 Obtener los nombres y apellidos de empleados que más cobran en la empresa.Considerar el salario más la comisión.**

```
SELECT
    nombre,
    ape1,
    ape2,
    nvl(comision, 0) + salario AS salario_total
FROM
    empleado
WHERE
    nvl(comision, 0) + salario >= ALL (
        SELECT
            nvl(comision, 0) + salario
        FROM
            empleado
    )
ORDER BY
    salario_total DESC;
```

**-- 8 Obtener en orden alfabético los nombres de empleado cuyo salario es inferior al mínimo de los empleados del departamento 1.**

```
SELECT
    nombre,
    ape1,
    ape2,
    salario
FROM
    empleado
WHERE
    salario < ALL (
        SELECT
            salario
        FROM
            empleado
        WHERE
            coddpto = 1
    )
ORDER BY
    ape1,
    ape2,
    nombre;
```

**-- 9 Obtener los nombre de empleados que trabajan en el departamento del cual es jefe el empleado con código 1.**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    coddpto IN (
        SELECT
            coddpto
        FROM
            dpto
        WHERE
            codemplejefe = 1
    );
```

**-- 10 Obtener los nombres de los empleados cuyo primer apellido empiece por las letras p, q, r, s.**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    substr(ape1, 1, 1) IN ( 'P', 'Q', 'R', 'S' )
ORDER BY
    ape1,
    ape2,
    nombre;
```

**-- 11 Obtener los empleados cuyo nombre de pila contenga el nombre JUAN.**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    nombre LIKE '%JUAN%'
ORDER BY
    nombre,
    ape1,
    ape2;
```

**-- 12 Obtener los nombres de los empleados que viven en ciudades en las que hay algún centro de trabajo**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    upper(localidad) IN (
        SELECT
            upper(localidad)
        FROM
            centro
    );
```

**-- 13 Obtener el nombre del jefe de departamento que tiene mayor salario de entre los jefes de departamento.**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    codemple IN (
        SELECT
            codemplejefe
        FROM
            dpto
        )
    AND salario = (
        SELECT
            MAX(salario)
        FROM
            empleado
        WHERE
            codemple IN (
                SELECT
                    codemplejefe
                FROM
                    dpto
            )
    );
```

**-- 14 Obtener en orden alfabético los salarios y nombres de los empleados cuyo salario sea superior al 60% del máximo salario de la empresa.**

```
SELECT
    nombre,
    ape1,
    ape2,
    salario
FROM
    empleado
WHERE
    salario > 0.6 * (
        SELECT
            MAX(salario)
        FROM
            empleado
    )
ORDER BY
    ape1,
    ape2,
    nombre;
```

**-- 15 Obtener en cuántas ciudades distintas viven los empleados.**

```
SELECT
    COUNT(DISTINCT localidad)
FROM
    empleado;
```

**-- 16 El nombre y apellidos del empleado que más salario cobra**

```
SELECT
    nombre,
    ape1,
    ape2,
    salario
FROM
    empleado
WHERE
    salario = (
        SELECT
            MAX(salario)
        FROM
            empleado
    );
```

**-- 17 Obtener las localidades y número de empleados de aquellas en las que viven más de 3 empleados**

```
SELECT
    localidad,
    COUNT(*)
FROM
    empleado
GROUP BY
    localidad
HAVING
    COUNT(*) > 3
ORDER BY
    localidad;
```

**-- 18 Obtener para cada departamento cuántos empleados trabajan, la suma de sus salarios y la suma de sus comisiones para aquellos departamentos en los que hay algún empleado cuyo salario es superior a 1700 euros.**

```
SELECT
    coddpto,
    SUM(salario)          AS salariototal,
    SUM(nvl(comision, 0)) AS comisiontotal,
    COUNT(codemple)      AS numempleados
FROM
    empleado
WHERE
    coddpto IN (
        SELECT DISTINCT
            coddpto
        FROM
            empleado
        WHERE
            salario > 1700
    )
GROUP BY
    coddpto;
```

**-- 19 Obtener el departamento que más empleados tiene**

```
SELECT
    denominacion
FROM
    dpto,
    empleado
WHERE
    empleado.coddpto = dpto.coddpto
GROUP BY
    dpto.coddpto,
    denominacion
HAVING
    COUNT(empleado.codemple) >= ALL (
        SELECT
            COUNT(codemple)
        FROM
            empleado
        GROUP BY
            coddpto
    );
```



**-- 20 Obtener los nombres de todos los centros y los departamentos que se ubican en cada uno, así como aquellos centros que no tienen departamentos.**

```
SELECT
    direccion      AS nomcentro,
    denominacion   AS dpto
FROM
    centro tc
    LEFT JOIN dpto td ON tc.codcentro = td.codcentro
ORDER BY
    1,
    2;
```

**-- 21 Obtener el nombre del departamento de más alto nivel, es decir, aquel que no depende de ningún otro.**

```
SELECT
    denominacion AS nomdpto
FROM
    dpto
WHERE
    coddptodepende IS NULL;
```

**-- 22 Obtener todos los departamentos existentes en la empresa y los empleados (si los tiene) que pertenecen a él.**

```
SELECT
    denominacion,
    nombre,
    ape1,
    ape2
FROM
    dpto td
    LEFT JOIN empleado te ON td.coddpto = te.coddpto
ORDER BY
    denominacion;
```

**-- 23 Obtener un listado en el que aparezcan todos los departamentos existentes y el departamento del cual depende, si depende de alguno.**

```
SELECT
    dpt1.denominacion      AS nomdpto,
    nvl(dptdelqdepende.denominacion, ' ') AS nomdptodelqdepende
FROM
    dpto dpt1
    LEFT JOIN dpto dptdelqdepende ON dpt1.coddptodepende = dptdelqdepende.coddpto;
```

**-- 24 Obtener un listado ordenado alfabéticamente donde aparezcan los nombres de los empleados y a continuación el literal "tiene comisión" si la tiene,y "no tiene comisión" si no la tiene.**

```
SELECT
    nombre,
    ape1,
    ape2,
    'tiene comision'
FROM
    empleado
WHERE
    comision IS NOT NULL
UNION
SELECT
    nombre,
    ape1,
    ape2,
    'no tiene comision'
FROM
    empleado
WHERE
    comision IS NULL
ORDER BY
    4,
    2;
```

**-- 25 Obtener un listado de las localidades en las que hay centros y no vive ningún empleado ordenado alfabéticamente.**

```
SELECT
    upper(tc.localidad)
FROM
    centro tc
WHERE
    rtrim(ltrim(upper(tc.localidad))) NOT IN (
    SELECT DISTINCT
    rtrim(ltrim(upper(te.localidad)))
    FROM
    empleado te
    );
```

**-- 26 Obtener un listado de las localidades en las que hay centros y además vive al menos un empleado ordenado alfabéticamente.**

```
SELECT
    upper(tc.localidad)
FROM
    centro tc
WHERE
    rtrim(ltrim(upper(tc.localidad))) IN (
        SELECT DISTINCT
            rtrim(ltrim(upper(te.localidad)))
        FROM
            empleado te
    );
```

-- 27 Esta cuestión puntúa por 2. Se desea dar una gratificación por navidades en función de la antigüedad en la empresa siguiendo estas pautas:

- Si lleva entre 1 y 5 años, se le dará 100 euros
- Si lleva entre 6 y 10 años, se le dará 50 euros por año
- Si lleva entre 11 y 20 años, se le dará 70 euros por año
- Si lleva más de 21 años, se le dará 100 euros por año

Obtener un listado de los empleados,ordenado alfabéticamente,indicando cuánto le corresponde de gratificación

```
SELECT
    nombre,
    ape1,
    ape2,
    100 AS gratificacion
FROM
    empleado
WHERE
    trunc((sysdate - fechaingreso) / 365) BETWEEN 1 AND 5
UNION
SELECT
    nombre,
    ape1,
    ape2,
    50 * trunc((sysdate - fechaingreso) / 365)
FROM
    empleado
WHERE
    trunc((sysdate - fechaingreso) / 365) BETWEEN 6 AND 10
UNION
SELECT
    nombre,
    ape1,
    ape2,
    70 * trunc((sysdate - fechaingreso) / 365)
FROM
    empleado
WHERE
    trunc((sysdate - fechaingreso) / 365) BETWEEN 11 AND 20
UNION
SELECT
    nombre,
    ape1,
    ape2,
    100 * trunc((sysdate - fechaingreso) / 365)
FROM
    empleado
WHERE
    ( sysdate - fechaingreso ) / 365 >= 21
ORDER BY
    1,
    2,
    3;
```

**-- 29 Obtener los nombres, apellidos de los empleados que no son jefes de departamento.**

```
SELECT
    nombre,
    ape1,
    ape2
FROM
    empleado
WHERE
    codemple NOT IN (
        SELECT
            codemplejefe
        FROM
            dpto
    );
```