

1. Mostrar los empleados de toda la empresa que ganan más que la media salarial de los empleados cuyo jefe es Nancy Greenberg. *Show all the employees that earn more than the average salary of the employees whose boss is Nancy Greenberg.*

```
select *
from EMPLOYEES
where SALARY > (select avg(e.salary)
                from employees e, employees j
                where j.employee_id = e.manager_id
                and j.first_name = 'Nancy' and j.last_name = 'Greenberg')
```

2. Mostrar todos los empleados de la empresa que ganan más que el jefe de Shelli Baida. *Show all the employees that earn more than the boss of Shelli Baida.*

```
select first_name, salary
from EMPLOYEES
where salary > (select jef.salary
                from employees emp, EMPLOYEES jef
                where jef.EMPLOYEE_ID = emp.MANAGER_ID
                and emp.first_name = 'Shelli' and emp.LAST_NAME = 'Baida')
```

3. Mostrar todos los empleados (con el trabajo de cada uno) del departamento en el que trabaja el empleado John Chen. *Show all the employees of the department where John Chen works.*

```
select employees.first_name, employees.last_name, jobs.job_title, department_name
from departments, employees, jobs
where employees.job_id= jobs.job_id
    and departments.department_id= employees.department_id
    and departments.department_id= (select department_id
                                    from employees
                                    where first_name='John' and last_name='Chen')
```

4. Mostrar todos los empleados con el departamento, ciudad y país de cada uno de ellos, que tienen el mismo trabajo que el empleado cuyo email es VPATABAL. *Show all the employees and their department, city and country with the same job as the employee whose email is VPATABA*

```
select first_name, last_name, department_name, city, country_name
from COUNTRIES C, LOCATIONS L, DEPARTMENTS D, EMPLOYEES E
where C.COUNTRY_ID = L.COUNTRY_ID
    and L.LOCATION_ID = D.LOCATION_ID
    and D.DEPARTMENT_ID = E.DEPARTMENT_ID
    and E.JOB_ID = (select E.JOB_ID
                    from EMPLOYEES E
                    where E.EMAIL = 'VPATABAL')
```

Mostrar todos los empleados (con el departamento y país al que pertenecen) cuya comisión es mayor que la comisión del empleado 151 y cuya antigüedad sea menor o igual que la del jefe del departamento 10.

```
select first_name, last_name, commission_pct, round((sysdate - hire_date)/365,2) as
Antigüedad, department_name, city
from LOCATIONS L, DEPARTMENTS D, EMPLOYEES E
where L.LOCATION_ID = D.LOCATION_ID
and D.DEPARTMENT_ID = E.DEPARTMENT_ID
and E.COMMISSION_PCT > (select E.COMMISSION_PCT
                        from EMPLOYEES E
                        where E.EMPLOYEE_ID = 151)
and round((sysdate - hire_date)/365,2) <= (select round((sysdate-E.HIRE_DATE)/365,2)
from EMPLOYEES E, DEPARTMENTS D
where E.EMPLOYEE_ID = D.MANAGER_ID
and D.DEPARTMENT_ID = 10)
```

5. Mostrar todas las relaciones jerárquicas entre departamentos apoyándose en las relaciones jerárquicas entre los jefes de departamento.

```
select dep_inf.department_name, dep_sup.department_name
from (select department_name, manager_id
      from departments) dep_sup,
      (select department_name, manager_id
      from departments) dep_inf,
      employees emp
where emp.employee_id=dep_inf.manager_id
and emp.manager_id=dep_sup.manager_id
```

6. Mostrar todos los departamentos (con el país al que pertenecen) en el que haya algún empleado que cobre comisiones.

```
select department_name, country_name
from COUNTRIES C, LOCATIONS L, DEPARTMENTS D
where C.COUNTRY_ID = L.COUNTRY_ID
and L.LOCATION_ID = D.LOCATION_ID
and D.DEPARTMENT_ID in ( select distinct department_id
                        from EMPLOYEES
                        where COMMISSION_PCT is not null)
```

7. Mostrar todos los empleados que tengan actualmente el mismo trabajo que el que tenía el empleado 102 el día 20 de Mayo de 1995.

```
SELECT *
from EMPLOYEES
where JOB_ID in (select JH.job_id
                from JOB_HISTORY JH,
                where JH.EMPLOYEE_ID = 102
                and JH.START_DATE>'20/05/95' and JH.END_DATE<'20/05/95')
```

8. Mostrar todos los empleados que desarrollen actualmente cualquiera de los trabajos que ha desarrollado anteriormente el empleado 176.

```
select first_name, last_name
from employees
where job_id in (select job_id
                 from job_history
                 where employee_id=176)
```

9. Mostrar todos los empleados que sólo han desarrollado en la empresa el trabajo que desarrollan actualmente.

```
select e.*
from EMPLOYEES e
where e.EMPLOYEE_ID not in (select EMPLOYEE_ID from JOB_HISTORY)
```

```
(select e.*
from employees e
where e.job_id in (select job_id
                  from job_history jh
                  where e.employee_id = jh.employee_id)
and (select count(distinct job_id)
     from job_history jh
     where e.employee_id=jh.employee_id) =1)
union
(select e.*
from employees e
where employee_id not in (select employee_id from job_history))
```

10. Mostrar la lista de todos los empleados con su salario, la diferencia salarial de cada empleado respecto al salario más alto de la empresa y la diferencia salarial de cada empleado respecto al salario más bajo de la empresa.

```
select first_name, last_name, salary ,
salary - (select max(salary) from EMPLOYEES) as DiferenciaAlta ,
salary - (select min(salary) from EMPLOYEES) DiferenciaBaja
from EMPLOYEES
```

11. Mostrar los empleados que ganan más del salario medio de la empresa, mostrando la desviación de cada uno de ellos respecto al salario medio (sin decimales).

```
select first_name, last_name, salary, round(salary-(select avg(salary) from employees))
from employees
where salary>(select avg(salary) from employees)
```

12. Mostrar una lista de todos los empleados de América con los siguientes datos por cada empleado: país, estado/provincia, ciudad, departamento, nombre, apellido, salario, nombre y apellido del jefe del empleado, nombre y apellido del jefe del departamento del empleado.

```
select country_name, state_province, city, department_name, emp.first_name Empleado,
emp.last_name, emp.salary, jef.first_name Jefe, jef.last_name, jefdep.first_name JefeDep,
jefdep.LAST_NAME
from COUNTRIES C, LOCATIONS L, DEPARTMENTS D, EMPLOYEES emp, EMPLOYEES
jef, EMPLOYEES jefdep
where C.COUNTRY_ID = L.COUNTRY_ID
and L.LOCATION_ID = D.LOCATION_ID
and D.DEPARTMENT_ID = emp.DEPARTMENT_ID
and emp.MANAGER_ID = jef.EMPLOYEE_ID
and jefdep.EMPLOYEE_ID = D.MANAGER_ID
```

13. Mostrar todos los departamentos de America, indicando para cada uno de ellos quien (nombre y apellido) es el empleado que más cobra, así como el salario de dicho empleado.

```
select department_name, first_name, last_name, salary
from COUNTRIES C, LOCATIONS L, DEPARTMENTS D,
(select emp.*
from EMPLOYEES emp
where salary = (select max(salary)
from EMPLOYEES empsal
where empsal.DEPARTMENT_ID = emp.DEPARTMENT_ID)) e
where C.COUNTRY_ID = L.COUNTRY_ID
and L.LOCATION_ID = D.LOCATION_ID
and D.DEPARTMENT_ID = e.DEPARTMENT_ID
and C.REGION_ID = 2
```

14. Mostrar una lista de todos los empleados cuyo salario es superior a la media de salarios de su país, indicando para cada uno de ellos el país al que pertenecen. Ordenar por país.

```
select first_name, last_name, salary, country_name
from COUNTRIES C, LOCATIONS L, DEPARTMENTS D, EMPLOYEES E
where C.COUNTRY_ID = L.COUNTRY_ID
and L.LOCATION_ID = D.LOCATION_ID
and D.DEPARTMENT_ID = E.DEPARTMENT_ID
and salary > (select avg(salary)
from COUNTRIES co, LOCATIONS lo, DEPARTMENTS de, EMPLOYEES emp
where co.COUNTRY_ID = lo.COUNTRY_ID
and lo.LOCATION_ID = de.LOCATION_ID
and de.DEPARTMENT_ID = emp.DEPARTMENT_ID
and c.country_id=co.country_id)
ORDER BY country_name
```

15. Mostrar todos los departamentos que tengan más de 4 empleados.

```
select *  
from DEPARTMENTS D  
where (select count(EMPLOYEE_ID)  
       from EMPLOYEES E  
       where D.DEPARTMENT_ID = E.DEPARTMENT_ID) > 4
```

16. Mostrar todos los empleados de Seattle que hayan tenido 2 trabajos en el pasado (en el histórico).

```
select E.*  
from LOCATIONS L, DEPARTMENTS D, EMPLOYEES E  
where L.LOCATION_ID = D.LOCATION_ID  
and D.DEPARTMENT_ID = E.DEPARTMENT_ID  
and L.CITY = 'Seattle'  
and (select count(distinct JH.JOB_ID)  
     from JOB_HISTORY JH  
     where E.EMPLOYEE_ID = JH.EMPLOYEE_ID)=2
```

17. Mostrar todos los empleados que cobran la misma comisión que su jefe de departamento. Si el jefe de departamento no cobra comisiones, se deberán mostrar también todos los empleados de dicho departamento que no cobren comisión.

```
select emp.First_name, emp.last_name, emp.commission_pct  
from DEPARTMENTS D, EMPLOYEES emp  
where D.DEPARTMENT_ID = emp.DEPARTMENT_ID  
and NVL(emp.COMMISSION_PCT,0) = (select nvl(commission_pct,0)  
                                 from EMPLOYEES jef  
                                 where jef.EMPLOYEE_ID = D.MANAGER_ID)
```

18. Mostrar la cantidad de países que empiezan por 'A' que hay en cada continente.

```
select regions.region_name "Continente", count(*) "Países en el continente"  
from regions, countries  
where regions.region_id= countries.region_id  
and countries.country_name like 'A%'  
group by regions.region_name
```

19. Mostrar todos los nombres de departamento, indicando por cada uno de ellos el salario máximo de su departamento, ordenando la salida de forma descendente por salario máximo.

```
select department_name, max(salary)  
from departments d, employees e  
where d.department_id= e.department_id  
group by department_name  
order by max(salary) DESC
```

20. Mostrar todos los nombres de departamento de America, indicando por cada uno de ellos el salario mínimo de su departamento. Mostrar sólo los departamentos cuyo salario mínimo se encuentre entre 6000 y 9000.

```
select d.department_name, min(e.salary)  
from regions r, countries c ,locations l, departments d, employees e  
where r.region_id= c.region_id
```

```
and c.country_id = l.country_id
and l.location_id= d.location_id
and d.department_id= e.department_id
and r.region_name='Americas'
group by d.department_name
having min(e.salary) BETWEEN 6000 AND 9000
```

21. Mostrar todos los nombres de departamento, indicando por cada uno de ellos la media salarial de su departamento, excluyendo los departamentos en los que se haya algún empleado que cobre comisión.

```
select d.department_name, round(AVG(salary),2)
from departments d, employees e
where d.department_id= e.department_id
and d.department_id not in(select DISTINCT e.department_id
                           from employees e
                           where e.commission_pct is not null
                           and e.department_id is not null)
group by d.department_name
```

22. Mostrar todos los nombres de departamento, indicando por cada uno de ellos la media salarial, el salario máximo y el salario mínimo de cada departamento.

```
select departments.department_name, round(avg(salary),0), max(salary), min(salary)
from departments, employees
where departments.department_id= employees.department_id
group by departments.department_name
```

23. Mostrar todos los departamentos de America que tienen más de 2 empleados, indicando para cada uno de ellos la ciudad en la que se encuentra, el nombre de departamento y la cantidad de empleados que tiene. Ordenar la salida por el número de empleados.

```
select d.department_name, l.city, count(e.employee_id)
from regions r, countries c, locations l, departments d, employees e
where r.region_id= c.region_id
and c.country_id= l.country_id
and l.location_id= d.location_id
and d.department_id= e.department_id
and r.region_name='Americas'
group by d.department_name, l.city
having count(e.employee_id)>2
order by count(e.employee_id)
```

24. Mostrar la cantidad de empleos que ha tenido en el pasado cada uno de los empleados de la empresa. Si un empleado no ha tenido antes ningún otro empleo, debe aparecer como 0.

```
select e.first_name, e.last_name, count(distinct jh.job_id)
from employees e, job_history jh
where e.employee_id= jh.employee_id (+)
and jh.job_id (+) <> e.job_id
GROUP by e.first_name, e.last_name
```

25. Mostrar todos los continentes, regiones, países y departamentos. Sin un país no tiene departamentos, dicho país también deberá aparecer.

```
select r.region_name, c.country_name, l.state_province, l.city, d.department_name
```

Human Resources – Subqueries

Ejercicios planteados por Ignacio Fontecha Hernández

```
from regions r, countries c, locations l, departments d
where r.region_id= c.region_id
and c.country_id= l.country_id
and l.location_id= d.location_id (+)
```

26. Mostrar todos los empleados de America, indicando para cada uno de ellos el departamento del que dicho empleado es jefe. Si un empleado no es jefe de departamento, también deberá aparecer en la lista.

```
select first_name, last_name, department_name
from (select e.employee_id ,e.first_name , e.last_name
      from employees e, departments d, locations l, countries c
      where e.department_id = d.department_id
      and d.location_id= l.location_id
      and l.country_id= c.country_id
      and c.region_id=2) empamerica,
departments
where employee_id=manager_id (+)
```

27. Mostrar todos los jefes de departamento, y por cada uno de ellos todos los empleos que han tenido a lo largo de la historia. Si un jefe de departamento no ha tenido otros empleos en el pasado, también deberá aparecer en la lista.

```
select e.first_name , e.last_name ,d.department_name, e.employee_id , h.job_id
from employees e, departments d, job_history h
where e.employee_id = d.manager_id
      and d.manager_id= h.employee_id (+)
```

28. Mostrar todos los empleados (nombre y apellido), indicando para cada uno de ellos su jefe (nombre y apellido). Los empleados que no tengan jefe también deben aparecer en la lista.

```
select e.first_name "nombre emp", e.last_name "apellido emp", j.first_name "nombre jefe",
j.last_name "apellido jefe"
from employees e, employees j
where j.employee_id(+)= e.manager_id
```