

```

SELECT * FROM HR.EMPLOYEES;
SELECT * FROM HR.JOBS;
SELECT * FROM HR.JOB_HISTORY;
SELECT * FROM HR.COUNTRIES;
SELECT * FROM HR.DEPARTMENTS;
SELECT * FROM HR.LOCATIONS;
SELECT * FROM HR.REGIONES;

```

--2. Seleccionar el primer nombre, segundo nombre y salario de los empleados, ordenar ascendientemente por primer nombre.

```

SELECT FIRST_NAME, LAST_NAME, SALARY
FROM HR.EMPLOYEES
ORDER BY FIRST_NAME;

```

--3. Seleccionar los datos de los empleados cuyo identificador de tarea sea AD_VP.

```

SELECT * FROM HR.EMPLOYEES WHERE JOB_ID = 'AD_VP';

```

--4. Seleccionar los nombres, fecha de nacimiento, salario, número telefónico y email de los empleados con salario mayor o igual a 17000.

```

SELECT FIRST_NAME || ' ' || LAST_NAME NAME, HIRE_DATE, SALARY, PHONE_NUMBER, EMAIL
FROM HR.EMPLOYEES
WHERE SALARY >= 17000;

```

--5. Seleccionar first_name, last_name, department_name.

```

SELECT E.first_name, E.last_name, D.department_name
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID;

```

--6. Seleccionar la cantidad de empleados por departamento.

```

SELECT D.DEPARTMENT_ID, COUNT(E.EMPLOYEE_ID) CANT_EMP
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
GROUP BY D.DEPARTMENT_ID;

```

--7. Cantidad de empleados por ciudad.

```

SELECT L.CITY, COUNT(E.EMPLOYEE_ID) CANT_EMP
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
INNER JOIN HR.LOCATIONS L
ON D.LOCATION_ID = L.LOCATION_ID
GROUP BY L.CITY;

```

--8. Cantidad de departamentos por país:

```

SELECT C.COUNTRY_NAME, COUNT(D.DEPARTMENT_ID) CANT_DEPTOS
FROM HR.DEPARTMENTS D INNER JOIN HR.LOCATIONS L
ON D.LOCATION_ID = L.LOCATION_ID
INNER JOIN HR.COUNTRIES C
ON L.COUNTRY_ID = C.COUNTRY_ID
GROUP BY C.COUNTRY_NAME;

```

--9. Muestra la cantidad de empleados por región:

```

SELECT R.REGION_NAME, COUNT(E.EMPLOYEE_ID) CANT_EMP
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
INNER JOIN HR.LOCATIONS L ON D.LOCATION_ID = L.LOCATION_ID
INNER JOIN HR.COUNTRIES C ON L.COUNTRY_ID = C.COUNTRY_ID
INNER JOIN HR.REGIONES R ON C.REGION_ID = R.REGION_ID
GROUP BY R.REGION_NAME;

```

```

--10. Muestra la cantidad de empleados por país:
SELECT C.COUNTRY_NAME, COUNT(E.EMPLOYEE_ID) CANT_EMP
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
INNER JOIN HR.LOCATIONS L ON D.LOCATION_ID = L.LOCATION_ID
INNER JOIN HR.COUNTRIES C ON L.COUNTRY_ID = C.COUNTRY_ID
GROUP BY C.COUNTRY_NAME;

-- 11. Cantidad de empleados por país y por región:
SELECT C.COUNTRY_NAME, R.REGION_NAME, COUNT(E.EMPLOYEE_ID) CANT_EMP
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
INNER JOIN HR.LOCATIONS L ON D.LOCATION_ID = L.LOCATION_ID
INNER JOIN HR.COUNTRIES C ON L.COUNTRY_ID = C.COUNTRY_ID
INNER JOIN HR.REGIONS R ON C.REGION_ID = R.REGION_ID
GROUP BY C.COUNTRY_NAME, R.REGION_NAME;

--13. Seleccionar la cantidad de usuarios por país: ??????????????????

--14. Mostrar la historia de trabajos de cada empleado:
SELECT E.FIRST_NAME || ' ' || E.LAST_NAME
NAME, J.JOB_TITLE, H.START_DATE, H.END_DATE, H.JOB_ID
FROM HR.EMPLOYEES E INNER JOIN HR.JOBS J
ON E.JOB_ID = J.JOB_ID
INNER JOIN HR.JOB_HISTORY H ON J.JOB_ID = H.JOB_ID
ORDER BY NAME, START_DATE;

--15. Seleccionar todas las tareas (Jobs) CON EL SALARIO MÍNIMO:
SELECT JOB_TITLE, MIN_SALARY FROM HR.JOBS;

--16. Seleccionar los empleados menores a 20 años que el primer nombre empiece por
la letra A:
SELECT * FROM HR.EMPLOYEES
WHERE MONTHS_BETWEEN(SYSDATE, TO_DATE(HIRE_DATE, 'DD/MM/YYYY')) <= 240 --Si la
cantidad de meses que lleva en la empresa, no supera los 240 (20 años)
AND FIRST_NAME LIKE 'A%';

--17. Mostrar el jefe de un empleado (?????????????????)

--18. Muestre los apellidos y nombres ordenados alfabeticamente, de todos los
gerentes en los que haya estado un empleado
-- laborando en algun puesto MENOS DE TRES MESES, asuma mes comercial de 30
dias:

--19. Realice una consulta que muestre el nombre de la region y país, citando para
cada uno de los países, muestre el total
-- de empleados que laboran en cada país, y además el coeficiente de rotacion de
personal (total empleados entre total de
-- empleados despedidos o que cambiaron hacia otro puesto):
SELECT * FROM HR.EMPLOYEES;
SELECT * FROM HR.JOBS;
SELECT * FROM HR.JOB_HISTORY ORDER BY EMPLOYEE_ID;
SELECT * FROM HR.JOB_HISTORY;
SELECT * FROM HR.DEPARTMENTS;

--20. Realice una consulta que muestre el Department_id, el Department_name, el
total de empleados por departamento, la cantidad
-- de días promedio que tienen laborando los empleados en dicho depto, y el

```

nombre y el apellido del empleado que tiene mas dias
-- laborando en ese depto (asuma que solo hay un empleado con mas dias
laborando), muestre solo la informacion para deptos que
-- tienen mas de 3 empleados laborando:

--1. Mostrar el nombre (first_name y last_name) y el salario (salary) de todos los empleados.

--Cambie el nombre de la columna first_name a Nombre y last_name a Apellido y salary a Salario.

```
SELECT FIRST_NAME||' '||LAST_NAME NAME,SALARY  
FROM HR.EMPLOYEES;
```

--ALTER TABLE HR.EMPLOYEES RENAME COLUMN FIRST_NAME TO NOMBRE;

--ALTER TABLE HR.EMPLOYEES RENAME COLUMN LAST_NAME TO APELLIDO;

--ALTER TABLE HR.EMPLOYEES RENAME COLUMN SALARY TO SALARIO;

--2. Mostrar el nombre (first_name y last_name) y el salario (salary) de todos los empleados ordenado

--alfabéticamente por apellido (last_name).

```
SELECT FIRST_NAME||' '||LAST_NAME NAME,SALARY FROM HR.EMPLOYEES ORDER BY LAST_NAME  
ASC;
```

--3. Mostrar el apellido (last_name) y la Comisi n (commission_pct) que perciben los empleados cuya comisi n

--sea mayor a 0.25. Mostrarlos ordenado en forma descendente por last_name.

```
SELECT LAST_NAME,COMMISSION_PCT
```

```
FROM HR.EMPLOYEES
```

```
WHERE COMMISSION_PCT >= 0.25
```

```
ORDER BY LAST_NAME DESC;
```

--4. Mostrar la cantidad de empleados que trabajan en el departamento 100:

```
SELECT COUNT(*) CANT_DEPTO_100
```

```
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
```

```
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
```

```
WHERE E.DEPARTMENT_ID = 100;
```

--5. Mostrar todos los datos de los departamentos con identificador 10   70 (department_id):

```
SELECT * FROM HR.DEPARTMENTS WHERE DEPARTMENT_ID IN (10,70);
```

--6. Mostrar el nombre (last_name) de los empleados junto al nombre del departamento (department_name) donde trabajan:

```
SELECT E.LAST_NAME,D.DEPARTMENT_NAME
```

```
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
```

```
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID;
```

--7. Mostrar el nombre (last_name) de los empleados de los departamentos de Finanzas (Finance) y/o Transporte (Shipping):

```
SELECT E.LAST_NAME
```

```
FROM HR.EMPLOYEES E INNER JOIN HR.DEPARTMENTS D
```

```
ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
```

```
WHERE UPPER(D.DEPARTMENT_NAME) IN ('FINANCE','SHIPPING');
```

--8. Mostrar sin repetir los tipos de trabajos (jobs) que realizan en los departamentos los empleados.

-- Liste el el identificador del departamento y el nombre del trabajo:

```
SELECT * FROM HR.DEPARTMENTS;
```

```
SELECT * FROM HR.JOBS;
```

```
SELECT * FROM HR.JOB_HISTORY;  
SELECT * FROM HR.EMPLOYEES;
```

```
SELECT  
FROM HR.EMPLOYEES E INNER JOIN HR.JOB_HISTORY H  
ON
```

```
INNER JOIN HR.DEPARTMENTS D  
ON E.EMPLOYEE_ID = D.EMPLOYEE_ID  
INNER JOIN D.
```

--9. Mostrar los departamentos en los que los empleados realizan trabajos de Contabilidad (Accountant) y Asistente de Administración (Administration Assistant). Liste el identificador del departamento y el nombre del trabajo.
--NOTA: En este caso no hay tuplas que satisfagan la intersección.
--10. Mostrar el nombre de los departamentos que tienen al menos 3 empleados.
--11. Mostrar el nombre de los empleados (last_name) y el salario (salary) de aquellos empleados que tienen un salario mayor que el salario promedio. Ordene el listado en forma descendente por salario.
--12. Mostrar el salario máximo de los empleados del departamento 110. Ordene el listado en forma ascendente por salario.
--13. Mostrar el nombre del/de los empleados que tienen el sueldo máximo del departamento 110. Ordene el listado en forma ascendente por salario.
--14. Busque todos los empleados que no sean supervisores (o directores). Nota: En la tabla departments el atributo manager_id tiene la identificación de los supervisores de cada dpto.