

SENTENCIAS SQL

Empleados (id, dni, nombre, apellido, fechaNacimiento, salario, ciudad)

Asignaciones (id, nombre, idEmpleado, idProyecto, idSupervisor, horasDedicadas)

Proyectos (id, nombre, empresa, ciudad)

- A. Mostrar el nombre y apellidos de los empleados que tienen algún proyecto en su ciudad. No debe haber duplicados. La salida debe estar ordenada por apellido, y en caso de mismo apellido, por nombre.

```
SELECT DISTINCT emp.nombre, emp.apellido
FROM empleados emp
INNER JOIN asignaciones asg ON emp.id = asg.idEmpleado
INNER JOIN proyectos pro ON asg.idEmpleado = pro.id
WHERE emp.ciudad = pro.ciudad
ORDER BY emp.apellido, emp.nombre;
```

- B. Mostrar el nombre y apellidos de los empleados que ganan más que algún supervisor suyo que viva en su ciudad.

```
SELECT DISTINCT emp.nombre, emp.apellido
FROM empleados emp
JOIN asignaciones asg ON emp.id = asg.idEmpleado
JOIN empleados sup ON asg.idSupervisor = sup.id
WHERE emp.salario > sup.salario AND emp.ciudad = sup.ciudad;
```

- C. Listar el nombre y apellido de todos los empleados cuyo nombre empieza por 'A' con el número de proyectos asignados y suma total de horas asignadas a proyectos.

```
SELECT emp.nombre, emp.apellido, COUNT(asg.idProyecto),  
SUM(asg.horasDedicadas)  
FROM empleados emp  
INNER JOIN asignaciones asg ON emp.id = asg.idEmpleado  
WHERE emp.nombre LIKE 'A%'  
GROUP BY emp.id;
```

- D. Mostrar las ciudades que sólo tienen un proyecto y ciudades de empleados con fecha de nacimiento desconocida

```
SELECT pro.ciudad  
FROM proyectos pro  
GROUP BY pro.ciudad  
HAVING COUNT(DISTINCT pro.id) = 1;  
UNION  
SELECT DISTINCT emp.ciudad  
FROM empleados emp  
WHERE emp.fechaNacimientos IS NULL;
```