

Ejercicio 1

```
use capgemini_arg;
select e.first_name, e.last_name, e.job_id, d.department_name
from employees e join departments d
on e.DEPARTMENT_ID = d.DEPARTMENT_ID
where e.COMMISSION_PCT = '0' and (salary between 7000 and 10000)
```

Ejercicio 2

```
use capgemini_arg;
select first_name, last_name, job_id
from employees
where job_id in ('it_prog', 'st_man', 'pr_rep')
order by JOB_ID
```

Ejercicio 3

```
use capgemini_arg;
-- select * from employees
select first_name, last_name, job_id, salary, commission_pct, salary + commission_pct
from employees
where first_name in ('Alberto')
-- alberto no existe
```

Ejercicio 4

```
use capgemini_arg;
-- select * from employees, de esta lista solo puedo obtener el department id
-- select * from departments, de esta lista obtengo department id y nombre del departament,
asumire que cogeremos solo department id
select distinct department_id from employees
-- order by department_id
-- modificacion para que me aparezca el ultimo como el primero
order by department_id desc
-- cuando veamos el update revisar como cambiamos un campo especifico
-- where department_id is null set -1
```

Ejercicio 5

```
use capgemini_arg;
select manager_id, last_name, job_id
from employees
where job_id not in ('SA_REP', 'AD_VP')
and last_name Like ('K%')
or (manager_id in ('100'))
and last_name like ('%tb%'))
order by first_name, manager_id desc
```

Ejercicio 6

```
-- use capgemini_arg;
-- select count(DEPARTMENT_ID) as emp, department_id
```

```

-- from employees
-- where DEPARTMENT_ID >= 40
-- group by DEPARTMENT_ID
-- order by emp desc
-- luego probando la consulta del ejercicio con un count en * en teoria me retornara mas por
los valores nulos
use capgemini_arg;
select count(*)
from employees
where DEPARTMENT_ID >= 40
group by DEPARTMENT_ID
order by 1 desc
-- es igual

```

Ejercicio 7

```

use capgemini_arg;
select count(*), country_id
from locations
group by COUNTRY_ID

```

Ejercicio 8

```

use capgemini_arg;
update employees set salary=0
where employee_id in (114,115);
commit
use capgemini_arg;
ALTER TABLE HR.EMPLOYEES MODIFY CONSTRAINT EMP_SALARY_MIN DISABLE VALIDATE;

```

Ejercicio 9

```

use capgemini_arg;
select min(salary), max(salary), max(salary)-min(salary) as difference from employees
group by job_id

```

Ejercicio 10

```

use capgemini_arg;
select job_title, salary
from jobs as job join employees as emp
where salary > 13000 and not job.JOB_ID like 'REP'

```

Ejercicio 11

```

use capgemini_arg;
select First_name
from employees
group by first_name
having count(*) > 1

```

Ejercicio 12

```

use capgemini_arg;

```

```
select DEPARTMENT_ID, avg(salary) as promedio
from employees
group by department_id
order by avg(salary) desc
limit 1
```