

030609

CHRISTIAN MILLÁN SORIA

1º DAW TARDE

Crear en cada una de las tablas un disparador que registre los movimientos en la tabla para cada una de las siguientes circunstancias:

- Antes de un INSERT.
- Antes de un UPDATE.
- Antes de un DELETE.

Para ello, se deberá insertar, actualizar y borrar datos para demostrar su correcto funcionamiento.

Creo una tabla secundaria para cada una de las tablas de la base de datos, donde se almacenarán los movimientos de su tabla correspondiente.

```
create table `tcentr_triggers`(  
  `id` int not null auto_increment,  
  `tipo` varchar(45) null,  
  `fecha` datetime null default current_timestamp,  
  primary key (`id`)  
);  
  
/*****/  
  
create table `tdepto_triggers`(  
  `id` int not null auto_increment,  
  `tipo` varchar(45) null,  
  `fecha` datetime null default current_timestamp,  
  primary key (`id`)  
);  
  
/*****/  
  
create table `temple_triggers`(  
  `id` int not null auto_increment,  
  `tipo` varchar(45) null,  
  `fecha` datetime null default current_timestamp,  
  primary key (`id`)  
);
```



Ahora creo los triggers de cada tipo para cada tabla.

```

delimiter //
create trigger registro1_tcentr after insert on tcentr
for each row
begin
    insert into tcentr_triggers(tipo) value('after insert');
end//
delimiter ;

/*****/

delimiter //
create trigger registro2_tcentr before insert on tcentr
for each row
begin
    insert into tcentr_triggers(tipo) value('before insert');
end//
delimiter ;

/*****/

delimiter //
create trigger registro3_tcentr after update on tcentr
for each row
begin
    insert into tcentr_triggers(tipo) value('after update');
end//
delimiter ;

/*****/

delimiter //
create trigger registro4_tcentr before update on tcentr
for each row
begin
    insert into tcentr_triggers(tipo) value('before update');
end//
delimiter ;

/*****/

delimiter //
create trigger registro5_tcentr after delete on tcentr

```

```

for each row
begin
    insert into tcentr_triggers(tipo) value('after delete');
end//
delimiter ;

/*****

delimiter //
create trigger registro6_tcentr before delete on tcentr
for each row
begin
    insert into tcentr_triggers(tipo) value('before delete');
end//
delimiter ;

/*****

-- igual para las tablas "tdepto" y "temple"...

```

Prueba de comportamiento correcto:

```
insert into tcentr values(30, 'test', 'test');
```

	id	tipo	fecha
▶	1	before insert	2023-05-09 02:04:56
	2	after insert	2023-05-09 02:04:56
*	NULL	NULL	NULL

```
update tcentr set numce=40 where numce=30;
```

	id	tipo	fecha
▶	1	before insert	2023-05-09 02:04:56
	2	after insert	2023-05-09 02:04:56
	3	before update	2023-05-09 02:07:18
	4	after update	2023-05-09 02:07:18
*	NULL	NULL	NULL

```
delete from tcentr where numce=40;
```

	id	tipo	fecha
▶	1	before insert	2023-05-09 02:04:56
	2	after insert	2023-05-09 02:04:56
	3	before update	2023-05-09 02:07:18
	4	after update	2023-05-09 02:07:18
	5	before delete	2023-05-09 02:09:18
	6	after delete	2023-05-09 02:09:18
	NULL	NULL	NULL

Además, para cada una de las preguntas se deberá hacer lo siguiente (siempre que sea posible):

- Crear una vista.
- Crear un procedimiento.
- Crear una función.

*Utilizar cursores y bucles siempre que sea posible.

BLOQUE 1

1. Hallar, por orden alfabético, los nombres de los departamentos cuyo director lo es en funciones y no en propiedad.

```
drop view if exists ej1b1;
create view ej1b1 as
select nomde from tdepto where tidir='f' order by 1;

/*****/

select * from empresa.ej1b1;
```

	nomde
▶	ORGANIZACION
	SECTOR INDUSTRIAL

```
drop procedure if exists procedure_ej1b1;
delimiter //
create procedure procedure_ej1b1()
begin
    select nomde from tdepto where tidir='f' order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej1b1();
```

	nomde
►	ORGANIZACION
	SECTOR INDUSTRIAL

2. Obtener un listín telefónico de los empleados del departamento 121, incluyendo nombre del empleado, número del empleado y extensión telefónica. Por orden alfabético descendente.

```
drop view if exists ej2b1;
create view ej2b1 as
select nomem, numem, extel from temple where numde=121 order by 1 desc;

/*****/

select * from empresa.ej2b1;
```

	nomem	numem	extel
►	RUIZ, FABIOLA	370	360
	PONS, CESAR	110	350
	PEREZ, JULIO	150	340

```
drop procedure if exists procedure_ej2b1;
delimiter //
create procedure procedure_ej2b1(in numde_param int)
begin
    select nomem, numem, extel from temple where numde=numde_param order by nomem
    desc;
end //
delimiter ;

/*****/

call empresa.procedure_ej2b1(121);
```

	nomem	numem	extel
►	RUIZ, FABIOLA	370	360
	PONS, CESAR	110	350
	PEREZ, JULIO	150	340

3. Obtener por orden creciente una relación de todos los números de extensiones telefónicas de los empleados. Elimina las repeticiones.

```
drop view if exists ej3b1;
create view ej3b1 as
select distinct extel from temple order by 1;

/*****/
```

```
select * from empresa.ej3b1;
```

	extel
▶	220
	340
	350
	360
	410
	508
	750
	760
	800
	810
	850
	880
	910

```
drop procedure if exists procedure_ej3b1;
delimiter //
create procedure procedure_ej3b1()
begin
    select distinct extel from temple order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej3b1();
```

	extel
▶	220
	340
	350
	360
	410
	508
	750
	760
	800
	810
	850
	880
	910

4. Hallar la comisión, nombre y salario de los empleados con más de un hijo, clasificados por comisión, y dentro de la comisión por orden alfabético. El listado debe incluir también los empleados con más de un hijo aunque no tengan comisión.

```

drop view if exists ej4b1;
create view ej4b1 as
select ifnull(comis, 0), nomem, salar from temple where numhi>1 order by 1, 2;

/*****/

select * from empresa.ej4b1;

```

	ifnull(comis, 0)	nomem	salar
►	0.00	FLOR, DOROTEA	1500.00
	0.00	GIL, GLORIA	2200.00
	0.00	LOPEZ, ANTONIO	2200.00
	0.00	PONS, CESAR	1000.00
	0.00	VEIGA, JULIANA	1500.00
	500.00	PEREZ, MARCOS	1800.00
	800.00	GARCIA, OCTAVIO	2200.00
	1000.00	LARA, LUCRECIA	1800.00
	1000.00	SANZ, LAVINIA	2200.00
	1100.00	TEROL, LUCIANO	1200.00

```

drop procedure if exists procedure_ej4b1;
delimiter //
create procedure procedure_ej4b1()
begin
    declare finalizado int default false;
    declare comision decimal(10, 2);
    declare nombre_empleado varchar(50);
    declare salario decimal(10, 2);

    declare cursor_datos_temple cursor for
        select ifnull(comis, 0), nomem, salar
        from temple
        where numhi>1
        order by 1, 2;

    declare continue handler for not found set finalizado = true;

    open cursor_datos_temple;

    while not finalizado do
        fetch cursor_datos_temple into comision, nombre_empleado, salario;
        if not finalizado then
            select comision, nombre_empleado, salario;
        end if;
    end while;

    close cursor_datos_temple;
end //
delimiter ;

```

```

/*****

```

```

call empresa.procedure_ej4b1();

```

	comision	nombre_empleado	salario
▶	1100.00	TEROL, LUCIANO	1200.00

Result 1
Result 2
Result 3
Result 4
Result 5
Result 6
Result 7
Result 8
Result 9
Result 10 ×



5. Obtener salario y nombre de los empleados con dos hijos por orden decreciente de salario y por orden alfabético dentro del salario.

```

drop view if exists ej5b1;
create view ej5b1 as
select salar, nomem from temple where numhi=2 order by 1 desc, 2;

```

```

/*****

```

```

select * from empresa.ej5b1;

```

	salar	nomem
▶	1800.00	LARA, LUCRECIA
	1800.00	PEREZ, MARCOS
	1200.00	TEROL, LUCIANO

```

drop procedure if exists procedure_ej5b1;
delimiter //
create procedure procedure_ej5b1()
begin
    declare salario decimal(10,2);
    declare nombre_empleado varchar(50);
    declare var_count int;

    declare cur_datos cursor for
        select salar, nomem
        from temple

```



```

where numhi=2
order by salar desc, nomem;

select count(*) into var_count from temple where numhi=2;

open cur_datos;
fetch cur_datos into salario, nombre_empleado;

while var_count>0 do
  select salario, nombre_empleado;
  fetch cur_datos into salario, nombre_empleado;
  set var_count=var_count-1;
end while;

close cur_datos;
end //
delimiter ;

/*****/

call empresa.procedure_ej5b1();

```

	salario	nombre_empleado
▶	1200.00	TEROL, LUCIANO

Result 1 Result 2 Result 3 x



6. Obtener el nombre de los empleados cuya comisión es superior o igual al 50% de su salario, por orden alfabético.

```

drop view if exists ej6b1;
create view ej6b1 as
select nomem from temple where comis>=salar*0.5 order by 1;

/*****/

select * from empresa.ej6b1;

```

	nomem
▶	DIEZ, AMELIA
	LARA, LUCRECIA
	TEROL, LUCIANO

```
drop procedure if exists procedure_ej6b1;
delimiter //
create procedure procedure_ej6b1()
begin
    select nomem from temple where comis>=salar*0.5 order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej6b1();
```

	nomem
▶	DIEZ, AMELIA
	LARA, LUCRECIA
	TEROL, LUCIANO

7. En una campaña de ayuda familiar se ha decidido dar a los empleados una paga extra de 30 euros por hijo a partir del tercero inclusive. Obtener por orden alfabético para estos empleados: nombre y salario total que van a cobrar incluyendo esta paga extra.

```
drop view if exists ej7b1;
create view ej7b1 as
select nomem, salar+30*(numhi-2) from temple where numhi>2 order by 1;

/*****/

select * from empresa.ej7b1;
```

	nomem	salar+30*(numhi-2)
▶	FLOR, DOROTEA	1590.00
	GARCIA, OCTAVIO	2230.00
	GIL, GLORIA	2230.00
	LOPEZ, ANTONIO	2320.00
	PONS, CESAR	1030.00
	SANZ, LAVINIA	2230.00
	VEIGA, JULIANA	1560.00

```
drop procedure if exists procedure_ej7b1;
delimiter //
create procedure procedure_ej7b1()
```

```

begin
  select nomem, salar+30*(numhi-2) from temple where numhi>2 order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej7b1();

```

	nomem	salar+30*(numhi-2)
►	FLOR, DOROTEA	1590.00
	GARCIA, OCTAVIO	2230.00
	GIL, GLORIA	2230.00
	LOPEZ, ANTONIO	2320.00
	PONS, CESAR	1030.00
	SANZ, LAVINIA	2230.00
	VEIGA, JULIANA	1560.00

7.1. En una campaña de ayuda familiar se ha decidido dar a los empleados una paga extra de 30 euros por hijo a partir del tercero inclusive. Obtener por orden alfabético para estos empleados: nombre y salario total que van a cobrar incluyendo esta paga extra. Para el resto de los empleados se debe mostrar el nombre y el salario que reciben siempre. Hacer este ejercicio de dos formas diferentes: con el operador "union" y con la expresión "case".

```

drop view if exists ej71b1;
create view ej71b1 as
select nomem, salar+30*(numhi-2) from temple where numhi>2 union select nomem,
salar from temple where numhi<=2 order by 1;

/*****/

select * from empresa.ej71b1;

```

	nomem	salar+30*(numhi-2)
►	DIEZ, AMELIA	1500.00
	FLOR, DOROTEA	1590.00
	GARCIA, OCTAVIO	2230.00
	GIL, GLORIA	2230.00
	LARA, LUCRECIA	1800.00
	LOPEZ, ANTONIO	2320.00
	MARTÍN, MICHAELA	1800.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	1800.00
	PONS, CESAR	1030.00
	RUIZ, FABIOLA	1800.00
	SANZ, LAVINIA	2230.00
	TEROL, LUCIANO	1200.00
	VEIGA, JULIANA	1560.00

```
drop procedure if exists procedure_ej71b1;
delimiter //
create procedure procedure_ej71b1()
begin
    declare nom_em varchar(50);
    declare salario decimal(10,2);
    declare num_em int;

    declare cur_empleados cursor for select nomem, salar, numhi from temple;

    declare continue handler for not found set @fin=1;

    set @fin=0;

    create temporary table empleados_temp(
        nom_empleado varchar(50),
        salario_nuevo decimal(10,2)
    );

    open cur_empleados;

    fetch cur_empleados into nom_em, salario, num_em;

    while(@fin=0) do
        if(num_em>2) then
            set salario=salario+(30*(num_em-2));
        end if;

        insert into empleados_temp (nom_empleado, salario_nuevo)
        values (nom_em, salario);

        fetch cur_empleados into nom_em, salario, num_em;
    end while;

    close cur_empleados;

    select nom_empleado, salario_nuevo
    from empleados_temp
    where salario_nuevo is not null
    order by 1;

    drop temporary table if exists empleados_temp;
end //
delimiter ;

/*****/

call empresa.procedure_ej71b1();
```

	nom_empleado	salario_nuevo
►	DIEZ, AMELIA	1500.00
	FLOR, DOROTEA	1590.00
	GARCIA, OCTAVIO	2230.00
	GIL, GLORIA	2230.00
	LARA, LUCRECIA	1800.00
	LOPEZ, ANTONIO	2320.00
	MARTÍN, MICAELA	1800.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	1800.00
	PONS, CESAR	1030.00
	RUIZ, FABIOLA	1800.00
	SANZ, LAVINIA	2230.00
	TEROL, LUCIANO	1200.00
	VEIGA, JULIANA	1560.00

```

drop view if exists ej72b1;
create view ej72b1 as
select nomem, case when numhi>2 then salar+30*(numhi-2) else salar end as
nuevo_salario from temple order by nomem;

/*****/

select * from empresa.ej72b1;

```

	nomem	nuevo_salario
►	DIEZ, AMELIA	1500.00
	FLOR, DOROTEA	1590.00
	GARCIA, OCTAVIO	2230.00
	GIL, GLORIA	2230.00
	LARA, LUCRECIA	1800.00
	LOPEZ, ANTONIO	2320.00
	MARTÍN, MICAELA	1800.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	1800.00
	PONS, CESAR	1030.00
	RUIZ, FABIOLA	1800.00
	SANZ, LAVINIA	2230.00
	TEROL, LUCIANO	1200.00
	VEIGA, JULIANA	1560.00

```

drop procedure if exists procedure_ej72b1;
delimiter //
create procedure procedure_ej72b1()
begin
    declare nombre_empleado varchar(50);
    declare salario_actual float;
    declare horas_extras int;
    declare nuevo_salario float;

    declare cur_empleados cursor for select nomem, salar, numhi from temple;

```

```

open cur_empleados;

fetch cur_empleados into nombre_empleado, salario_actual, horas_extras;

while(select count(*)) do
  if horas_extras>2 then
    set nuevo_salario=salario_actual+(30*(horas_extras-2));
  else
    set nuevo_salario=salario_actual;
  end if;

  select nombre_empleado, nuevo_salario;

  fetch cur_empleados into nombre_empleado, salario_actual, horas_extras;
end while;

close cur_empleados;
end //
delimiter ;

/*****/

call empresa.procedure_ej72b1();

```

nombre_empleado	nuevo_salario
MARTÍN, MICAELA	1800



Result 1 Result 2 Result 3 Result 4 Result 5 Result 6 Result 7 Result 8 Result 9 Result 10 Result 11 Result 12 Result 13 Result 14 x

8. Hallar por orden alfabético los nombres de los empleados tales que, si se les da una gratificación de 60 euros por hijo, el total de esta gratificación no supera a la décima parte de su salario.

```

drop view if exists ej8b1;
create view ej8b1 as
select nomem from temple where 60*numhi<=salar/10 order by 1;

/*****/

select * from empresa.ej8b1;

```

	nomem
►	DIEZ, AMELIA
	GARCIA, OCTAVIO
	GIL, GLORIA
	LARA, LUCRECIA
	MARTÍN, MICAELA
	PEREZ, JULIO
	PEREZ, MARCOS
	RUIZ, FABIOLA
	SANZ, LAVINIA
	TEROL, LUCIANO

```
drop procedure if exists procedure_ej8b1;
delimiter //
create procedure procedure_ej8b1()
begin
    select nomem from temple where 60*numhi<=salar/10 order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej8b1();
```

	nomem
►	DIEZ, AMELIA
	GARCIA, OCTAVIO
	GIL, GLORIA
	LARA, LUCRECIA
	MARTÍN, MICAELA
	PEREZ, JULIO
	PEREZ, MARCOS
	RUIZ, FABIOLA
	SANZ, LAVINIA
	TEROL, LUCIANO

9. Obtener para los departamentos con un presupuesto superior a 5000 euros, su nombre junto con el nombre del centro donde está ubicado.

```
drop view if exists ej9b1;
create view ej91b1 as
select d.nomde, c.nomce from tcentr c, tdepto d where c.numce=d.numce and
d.presu>5000;

/*****/

select * from empresa.ej91b1;
```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES

```

drop procedure if exists procedure_ej9b1;
delimiter //
create procedure procedure_ej9b1()
begin
    select d.nomde, c.nomce from tcentr c, tdepto d where c.numce=d.numce and
d.presu>5000;
end //
delimiter ;

/*****/

call empresa.procedure_ej9b1();

```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES

```

drop view if exists ej92b1;
create view ej92b1 as
select d.nomde, c.nomce from tcentr c, tdepto d where c.numce=d.numce and
d.presu>5000
union
select d.nomde, c.nomce from tcentr c join tdepto d on c.numce=d.numce where
d.presu>5000 and c.nomce not in (select c.nomce from tcentr c, tdepto d where
c.numce=d.numce and d.presu>5000);

/*****/

select * from empresa.ej92b1;

```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES

```

drop procedure if exists procedure_ej92b1;
delimiter //
create procedure procedure_ej92b1()
begin
    select d.nomde, c.nomce from tcentr c, tdepto d where c.numce=d.numce and

```



```

d.presu>5000
union
select d.nomde, c.nomce from tcentr c join tdepto d on c.numce=d.numce where
d.presu>5000 and c.nomce not in (select c.nomce from tcentr c, tdepto d where
c.numce=d.numce and d.presu>5000);
end //
delimiter ;

/*****/

call empresa.procedure_ej92b1();

```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES

10. Para los empleados del departamento de Nóminas, obtener el nombre, salario y número de hijos. Hacer el ejercicio de dos formas: utilizando un producto cartesiano y con la cláusula "join".

```

drop view if exists ej101b1;
create view ej101b1 as
select e.nomem, e.salar, e.numhi from temple e, tdepto d where e.numde=d.numde
and d.nomde='nominas';

/*****/

select * from empresa.ej101b1;

```

	nomem	salar	numhi
►	PEREZ, MARCOS	1800.00	2
	VEIGA, JULIANA	1500.00	4

```

drop procedure if exists procedure_ej10b1;
delimiter //
create procedure procedure_ej101b1()
begin
select e.nomem, e.salar, e.numhi from temple e, tdepto d where e.numde=d.numde
and d.nomde='nominas';
end //
delimiter ;

/*****/

call empresa.procedure_ej101b1();

```

	nomem	salar	numhi
▶	PEREZ, MARCOS	1800.00	2
	VEIGA, JULIANA	1500.00	4

```
drop view if exists ej102b1;
create view ej102b1 as
select e.nomem, e.salar, e.numhi from temple e join tdepto d on e.numde=d.numde
where d.nomde='nominas';

/*****/

select * from empresa.ej102b1;
```

	nomem	salar	numhi
▶	PEREZ, MARCOS	1800.00	2
	VEIGA, JULIANA	1500.00	4

```
drop procedure if exists procedure_ej102b1;
delimiter //
create procedure procedure_ej102b1()
begin
    select e.nomem, e.salar, e.numhi from temple e join tdepto d on e.numde=d.numde
    where d.nomde='nominas';
end //
delimiter ;

/*****/

call empresa.procedure_ej102b1();
```

	nomem	salar	numhi
▶	PEREZ, MARCOS	1800.00	2
	VEIGA, JULIANA	1500.00	4

BLOQUE 2

1. Hallar el nombre de los empleados que no tienen comisión, clasificados de manera que aparezcan primero aquellos nombres que son más cortos.

```
drop view if exists ej1b2;
create view ej1b2 as
select nomem as 'Empleados sin comisión' from temple where comis is null order by
length(nomem), 1;

/*****/
```

```
select * from empresa.ej1b2;
```

	Empleados sin comisión
▶	GIL, GLORIA
	PONS, CESAR
	PEREZ, JULIO
	FLOR, DOROTEA
	RUIZ, FABIOLA
	LOPEZ, ANTONIO
	VEIGA, JULIANA
	MARTÍN, MICAELA

```
drop procedure if exists procedure_ej1b2;
delimiter //
create procedure procedure_ej1b2()
begin
    select nomem as 'Empleados sin comisión' from temple where comis is null order
    by length(nomem), 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej1b2();
```

	Empleados sin comisión
▶	GIL, GLORIA
	PONS, CESAR
	PEREZ, JULIO
	FLOR, DOROTEA
	RUIZ, FABIOLA
	LOPEZ, ANTONIO
	VEIGA, JULIANA
	MARTÍN, MICAELA

2. Hallar, por orden alfabético, los nombres de los empleados suprimiendo las dos últimas letras.

```
drop view if exists ej2b2;
create view ej2b2 as
select substring(nomem, 1, length(nomem)-2) as 'Nombre recortado' from temple
order by 1;

/*****/

select * from empresa.ej2b2;
```

	Nombre recortado
►	DIEZ, AMEL
	FLOR, DOROT
	GARCIA, OCTAV
	GIL, GLOR
	LARA, LUCREC
	LOPEZ, ANTON
	MARTÍN, MICAEL
	PEREZ, JUL
	PEREZ, MARC
	PONS, CES
	RUIZ, FABIO
	SANZ, LAVIN
	TEROL, LUCIA
	VEIGA, JULIA

```
drop procedure if exists procedure_ej2b2;
delimiter //
create procedure procedure_ej2b2()
begin
    select substring(nomem, 1, length(nomem)-2) as 'Nombre recortado' from temple
order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej2b2();
```

	Nombre recortado
►	DIEZ, AMEL
	FLOR, DOROT
	GARCIA, OCTAV
	GIL, GLOR
	LARA, LUCREC
	LOPEZ, ANTON
	MARTÍN, MICAEL
	PEREZ, JUL
	PEREZ, MARC
	PONS, CES
	RUIZ, FABIO
	SANZ, LAVIN
	TEROL, LUCIA
	VEIGA, JULIA

3. Hallar cuántos departamentos hay y el presupuesto anual medio de ellos.

```
drop view if exists ej3b2;
create view ej3b2 as
select count(distinct nomde) as 'Nº de departamentos',
```

```
round(sum(presu)/count(distinct nomde), 2) as 'Presupuesto anual medio' from
tdepto;
```

```
/*****
```

```
select * from empresa.ej3b2;
```

	Nº de departamentos	Presupuesto anual medio
▶	6	7000.00

```
drop procedure if exists procedure_ej3b2;
delimiter //
create procedure procedure_ej3b2()
begin
    select count(distinct nomde) as 'Nº de departamentos',
    round(sum(presu)/count(distinct nomde), 2) as 'Presupuesto anual medio' from
tdepto;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej3b2();
```

	Nº de departamentos	Presupuesto anual medio
▶	6	7000.00

4. Hallar la masa salarial anual (salario más comisión) de la empresa (se suponen 14 pagas anuales).

```
drop view if exists ej4b2;
create view ej4b2 as
select sum((salar+comis)*14) as 'Masa salarial' from temple;
```

```
/*****
```

```
select * from empresa.ej4b2;
```

	Masa salarial
▶	224000.00

```
drop procedure if exists procedure_ej4b2;
delimiter //
```

```

create procedure procedure_ej4b2()
begin
    select sum((salar+comis)*14) as 'Masa salarial' from temple;
end //
delimiter ;

/*****/

call empresa.procedure_ej4b2();

```

	Masa salarial
▶	224000.00

5. Hallar la masa salarial anual (salario más comisión) de cada empleado (se suponen 14 pagas anuales). Hacer el ejercicio de diferentes maneras: con el operador "union", con la función "ifnull", con la función "if" y con la expresión "case".

```

drop view if exists ej5b2;
create view ej5b2 as
select nomem as 'Nombre', masa as 'Masa salarial' from (select nomem, salar as
'masa' from temple union select nomem, comis as 'masa' from temple) as
temp_salarios group by nomem;

/*****/

select * from empresa.ej5b2;

```

	Nombre	Masa salarial
▶	PONS, CESAR	1000.00
	TEROL, LUCIANO	1200.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	1800.00
	VEIGA, JULIANA	1500.00
	SANZ, LAVINIA	2200.00
	LOPEZ, ANTONIO	2200.00
	GARCIA, OCTAVIO	2200.00
	FLOR, DOROTEA	1500.00
	GIL, GLORIA	2200.00
	DIEZ, AMELIA	1500.00
	LARA, LUCRECIA	1800.00
	RUIZ, FABIOLA	1800.00
	MARTÍN, MICAELA	1800.00

```

drop procedure if exists procedure_ej5b2;
delimiter //
create procedure procedure_ej5b2()
begin
    select nomem as 'Nombre', masa as 'Masa salarial' from (select nomem, salar as

```

```
'masa' from temple union select nomem, comis as 'masa' from temple) as
temp_salarios group by nomem;
end //
delimiter ;

/*****/

call empresa.procedure_ej5b2();
```

	Nombre	Masa salarial
►	PONS, CESAR	1000.00
	TEROL, LUCIANO	1200.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	1800.00
	VEIGA, JULIANA	1500.00
	SANZ, LAVINIA	2200.00
	LOPEZ, ANTONIO	2200.00
	GARCIA, OCTAVIO	2200.00
	FLOR, DOROTEA	1500.00
	GIL, GLORIA	2200.00
	DIEZ, AMELIA	1500.00
	LARA, LUCRECIA	1800.00
	RUIZ, FABIOLA	1800.00
	MARTÍN, MICAELA	1800.00

"ifnull":

```
drop view if exists ej51b2;
create view ej51b2 as
select nomem as 'Nombre', (salar+ifnull(comis, 0)*14) as 'Masa salarial' from
temple order by 1;

/*****/

select * from empresa.ej51b2;
```

	Nombre	Masa salarial
►	DIEZ, AMELIA	14100.00
	FLOR, DOROTEA	1500.00
	GARCIA, OCTAVIO	13400.00
	GIL, GLORIA	2200.00
	LARA, LUCRECIA	15800.00
	LOPEZ, ANTONIO	2200.00
	MARTÍN, MICAELA	1800.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	8800.00
	PONS, CESAR	1000.00
	RUIZ, FABIOLA	1800.00
	SANZ, LAVINIA	16200.00
	TEROL, LUCIANO	16600.00
	VEIGA, JULIANA	1500.00

```

drop procedure if exists procedure_ej51b2;
delimiter //
create procedure procedure_ej51b2()
begin
    select nomem as 'Nombre', (salar+ifnull(comis, 0)*14) as 'Masa salarial' from
temple order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej51b2();

```

	Nombre	Masa salarial
►	DIEZ, AMELIA	14100.00
	FLOR, DOROTEA	1500.00
	GARCIA, OCTAVIO	13400.00
	GIL, GLORIA	2200.00
	LARA, LUCRECIA	15800.00
	LOPEZ, ANTONIO	2200.00
	MARTÍN, MICAELA	1800.00
	PEREZ, JULIO	2200.00
	PEREZ, MARCOS	8800.00
	PONS, CESAR	1000.00
	RUIZ, FABIOLA	1800.00
	SANZ, LAVINIA	16200.00
	TEROL, LUCIANO	16600.00
	VEIGA, JULIANA	1500.00

"if":

```

drop view if exists ej52b2;
create view ej52b2 as
select nomem as 'Nombre', (salar+if(comis is null, 0, comis))*14 as 'Masa
salarial' from temple order by 1;

/*****/

select * from empresa.ej52b2;

```


	Nombre	Masa salarial
►	DIEZ, AMELIA	33600.00
	FLOR, DOROTEA	21000.00
	GARCIA, OCTAVIO	42000.00
	GIL, GLORIA	30800.00
	LARA, LUCRECIA	39200.00
	LOPEZ, ANTONIO	30800.00
	MARTÍN, MICAELA	25200.00
	PEREZ, JULIO	30800.00
	PEREZ, MARCOS	32200.00
	PONS, CESAR	14000.00
	RUIZ, FABIOLA	25200.00
	SANZ, LAVINIA	44800.00
	TEROL, LUCIANO	32200.00
	VEIGA, JULIANA	21000.00

```

drop procedure if exists procedure_ej52b2;
delimiter //
create procedure procedure_ej52b2()
begin
    select nomem as 'Nombre', (salar+if(comis is null, 0, comis))*14 as 'Masa
salarial' from temple order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej52b2();

```

	Nombre	Masa salarial
►	DIEZ, AMELIA	33600.00
	FLOR, DOROTEA	21000.00
	GARCIA, OCTAVIO	42000.00
	GIL, GLORIA	30800.00
	LARA, LUCRECIA	39200.00
	LOPEZ, ANTONIO	30800.00
	MARTÍN, MICAELA	25200.00
	PEREZ, JULIO	30800.00
	PEREZ, MARCOS	32200.00
	PONS, CESAR	14000.00
	RUIZ, FABIOLA	25200.00
	SANZ, LAVINIA	44800.00
	TEROL, LUCIANO	32200.00
	VEIGA, JULIANA	21000.00

"case":

```

drop view if exists ej53b2;
create view ej53b2 as
select nomem as 'Nombre', case when comis is null then salar*14

```

```

else(salar+comis)*14 end as 'Masa salarial' from temple order by 1;

/*****/

select * from empresa.ej53b2;

```

	Nombre	Masa salarial
►	DIEZ, AMELIA	33600.00
	FLOR, DOROTEA	21000.00
	GARCIA, OCTAVIO	42000.00
	GIL, GLORIA	30800.00
	LARA, LUCRECIA	39200.00
	LOPEZ, ANTONIO	30800.00
	MARTÍN, MICAELA	25200.00
	PEREZ, JULIO	30800.00
	PEREZ, MARCOS	32200.00
	PONS, CESAR	14000.00
	RUIZ, FABIOLA	25200.00
	SANZ, LAVINIA	44800.00
	TEROL, LUCIANO	32200.00
	VEIGA, JULIANA	21000.00

```

drop procedure if exists procedure_ej53b2;
delimiter //
create procedure procedure_ej53b2()
begin
    select nomem as 'Nombre', case when comis is null then salar*14
else(salar+comis)*14 end as 'Masa salarial' from temple order by 1;
end //
delimiter ;

/*****/

call empresa.procedure_ej53b2();

```

	Nombre	Masa salarial
►	DIEZ, AMELIA	33600.00
	FLOR, DOROTEA	21000.00
	GARCIA, OCTAVIO	42000.00
	GIL, GLORIA	30800.00
	LARA, LUCRECIA	39200.00
	LOPEZ, ANTONIO	30800.00
	MARTÍN, MICAELA	25200.00
	PEREZ, JULIO	30800.00
	PEREZ, MARCOS	32200.00
	PONS, CESAR	14000.00
	RUIZ, FABIOLA	25200.00
	SANZ, LAVINIA	44800.00
	TEROL, LUCIANO	32200.00
	VEIGA, JULIANA	21000.00

6. Hallar cuántos empleados han ingresado en el año actual. Utiliza la función "year".

```
drop view if exists ej6b2;
create view ej6b2 as
select count(distinct nomem) as 'Nº de empleados este año' from temple where
year(fecin)=2022;
```

```
/* **** */
```

```
select * from empresa.ej6b2;
```

	Nº de empleados este año
▶	2

```
drop procedure if exists procedure_ej6b2;
delimiter //
create procedure procedure_ej6b2()
begin
    select count(distinct nomem) as 'Nº de empleados este año' from temple where
    year(fecin)=2022;
end //
delimiter ;
```

```
/* **** */
```

```
call empresa.procedure_ej6b2();
```

	Nº de empleados este año
▶	2

También se puede obtener el año actual según la fecha del sistema donde se ejecuta la base de datos, en este caso mi propio ordenador, por lo que la fecha actual es 2023. El resultado es este:

```
drop view if exists ej61b2;
create view ej61b2 as
select count(distinct nomem) as 'Nº de empleados este año' from temple where
year(fecin)=year(now());
```

```
/* **** */
```

```
select * from empresa.ej61b2;
```

	Nº de empleados este año
▶	0

```
drop procedure if exists procedure_ej61b2;
delimiter //
create procedure procedure_ej61b2()
begin
    select count(distinct nomem) as 'Nº de empleados este año' from temple where
    year(fecin)=year(now());
end //
delimiter ;

/*****/

call empresa.procedure_ej61b2();
```

	Nº de empleados este año
▶	0

7. Hallar la diferencia entre el salario más alto y el salario más bajo.

```
drop view if exists ej7b2;
create view ej7b2 as
select max(salar)-min(salar) as 'Diferencia' from temple;

/*****/

select * from empresa.ej7b2;
```

	Diferencia
▶	1200.00

```
drop procedure if exists procedure_ej7b2;
delimiter //
create procedure procedure_ej7b2()
begin
    select max(salar)-min(salar) as 'Diferencia' from temple;
end //
delimiter ;

/*****/

call empresa.procedure_ej7b2();
```

	Diferencia
▶	1200.00

BLOQUE 3

1. Obtener, por orden alfabético, los nombres de los empleados cuyos salarios superen la mitad del salario del empleado 180.

```
drop view if exists ej1b3;
create view ej1b3 as
select nomem from temple where salar>(select salar/2 from temple where numem=180)
order by nomem asc;
```

/*****

```
select * from empresa.ej1b3;
```

	nomem
▶	DIEZ, AMELIA
	FLOR, DOROTEA
	GARCIA, OCTAVIO
	GIL, GLORIA
	LARA, LUCRECIA
	LOPEZ, ANTONIO
	MARTÍN, MICAELA
	PEREZ, JULIO
	PEREZ, MARCOS
	PONS, CESAR
	RUIZ, FABIOLA
	SANZ, LAVINIA
	TEROL, LUCIANO
	VEIGA, JULIANA

```
drop procedure if exists procedure_ej1b3;
delimiter //
create procedure procedure_ej1b3(in numem_intr int)
begin
    select nomem from temple where salar>(select salar/2 from temple where
numem=numem_intr) order by nomem asc;
end //
delimiter ;
```

/*****

```
call empresa.procedure_ej1b3(180);
```

	nomem
►	DIEZ, AMELIA
	FLOR, DOROTEA
	GARCIA, OCTAVIO
	GIL, GLORIA
	LARA, LUCRECIA
	LOPEZ, ANTONIO
	MARTÍN, MICAELA
	PEREZ, JULIO
	PEREZ, MARCOS
	PONS, CESAR
	RUIZ, FABIOLA
	SANZ, LAVINIA
	TEROL, LUCIANO
	VEIGA, JULIANA

2. Obtener, por orden alfabético, los nombres de los empleados cuyos salarios superen dos veces al mínimo salario de los empleados del departamento 121.

```
drop view if exists ej2b3;
create view ej2b3 as
select e.nomem from temple e inner join tdepto d on e.numde=d.numde where
d.numde=121 and e.salar>(2*(select min(salar) from temple where numde=121)) order
by e.nomem asc;

/*****/

select * from empresa.ej2b3;
```

	nomem
►	PEREZ, JULIO

```
drop procedure if exists procedure_ej2b3;
delimiter //
create procedure procedure_ej2b3(in numde_intr int)
begin
select e.nomem from temple e inner join tdepto d on e.numde=d.numde where
d.numde=numde_intr and e.salar>(2*(select min(salar) from temple where
numde=numde_intr)) order by e.nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej2b3(121);
```

	nomem
►	PEREZ, JULIO

3. Obtener, por orden alfabético, los nombres y los salarios de los empleados cuyo salario es inferior a tres veces la comisión más baja existente distinta de null.

```
drop view if exists ej3b3;
create view ej3b3 as
select nomem, salar from temple where comis is not null and salar<(select
min(comis)*3 from temple where comis is not null)order by nomem asc;

/*****/

select * from empresa.ej3b3;
```

	nomem	salar
▶	TEROL, LUCIANO	1200.00

```
drop procedure if exists procedure_ej3b3;
delimiter //
create procedure procedure_ej3b3()
begin
    select nomem, salar from temple where comis is not null and salar<(select
min(comis)*3 from temple where comis is not null)order by nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej3b3();
```

	nomem	salar
▶	TEROL, LUCIANO	1200.00

4. Obtener, utilizando el predicado "between", por orden alfabético, los nombres y los salarios de los empleados con hijos cuyo salario dividido por su número de hijos cumpla una, o ambas, de las dos condiciones siguientes:

- Que sea inferior a 1200 euros.
- Que sea superior al doble de su comisión.

```
drop view if exists ej4b3;
create view ej4b3 as
select nomem, salar from temple where numhi>0 and salar/numhi between 0 and 1200
or salar/numhi>comis*2 order by nomem asc;

/*****/

select * from empresa.ej4b3;
```

	nomem	salar
►	FLOR, DOROTEA	1500.00
	GARCIA, OCTAVIO	2200.00
	GIL, GLORIA	2200.00
	LARA, LUCRECIA	1800.00
	LOPEZ, ANTONIO	2200.00
	PEREZ, MARCOS	1800.00
	PONS, CESAR	1000.00
	SANZ, LAVINIA	2200.00
	TEROL, LUCIANO	1200.00
	VEIGA, JULIANA	1500.00

```

drop procedure if exists procedure_ej4b3;
delimiter //
create procedure procedure_ej4b3()
begin
    select nomem, salar from temple where numhi>0 and salar/numhi between 0 and 1200
or salar/numhi>comis*2 order by nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej4b3();

```

	nomem	salar
►	FLOR, DOROTEA	1500.00
	GARCIA, OCTAVIO	2200.00
	GIL, GLORIA	2200.00
	LARA, LUCRECIA	1800.00
	LOPEZ, ANTONIO	2200.00
	PEREZ, MARCOS	1800.00
	PONS, CESAR	1000.00
	SANZ, LAVINIA	2200.00
	TEROL, LUCIANO	1200.00
	VEIGA, JULIANA	1500.00

5. Obtener, por orden alfabético, los nombres de los empleados cuyo primer apellido es "Mora" o empieza por "Mora".

```

drop view if exists ej5b3;
create view ej5b3 as
select nomem from temple where nomem like 'mora%' or nomem like '% mora%' order by
nomem asc;

/*****/

select * from empresa.ej5b3;

```


	nomem
--	-------

```
drop procedure if exists procedure_ej5b3;
delimiter //
create procedure procedure_ej5b3()
begin
    select nomem from temple where nomem like 'mora%' or nomem like '% mora%' order
    by nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej5b3();
```

	nomem
--	-------

6. Obtener, por orden alfabético, los nombres de los empleados cuyo primer apellido termina en "ez" y su nombre de pila termina en "o" y tiene al menos tres letras.

```
insert into temple (numem, nomem, fecna, fecin, numhi, salar, comis, extel, numde)
values(1, 'Diego Perez Rodriguez', '1990-01-01', '2019-01-01', 2, 2500.00, NULL,
'1234', 1);

insert into temple (numem, nomem, fecna, fecin, numhi, salar, comis, extel, numde)
values(2, 'Hugo Sanchez Martinez', '1995-05-12', '2020-01-01', 1, 3000.00, 200.00,
'5678', 2);

insert into temple (numem, nomem, fecna, fecin, numhi, salar, comis, extel, numde)
values(3, 'Pedro Jimenez Perez', '1993-07-20', '2018-01-01', 0, 2000.00, NULL,
'4321', 2);

insert into temple (numem, nomem, fecna, fecin, numhi, salar, comis, extel, numde)
values(4, 'Fernando Ramirez Velez', '1989-02-28', '2015-01-01', 3, 4000.00,
500.00, '8765', 3);
```

```
drop view if exists ej6b3;
create view ej6b3 as
select nomem from temple where substring_index(nomem, ' ', -1) like '%o' and
substring_index(nomem, ' ', 1) like '%ez%' and char_length(substring_index(nomem, ' ', 1))>=3 order by nomem asc;
```

```

/*****

```

```

select * from empresa.ej6b3;

```

	nomem
▶	LOPEZ, ANTONIO
	PEREZ, JULIO

```

drop procedure if exists procedure_ej6b3;
delimiter //
create procedure procedure_ej6b3()
begin
    select nomem from temple where substring_index(nomem, ' ', -1) like '%o' and
    substring_index(nomem, ' ', 1) like '%ez%' and char_length(substring_index(nomem, '
    ', 1))>=3 order by nomem asc;
end //
delimiter ;

```

```

/*****

```

```

call empresa.procedure_ej6b3();

```

	nomem
▶	LOPEZ, ANTONIO
	PEREZ, JULIO

7. Obtener, utilizando el predicado "in", por orden alfabético, los nombres de los empleados del departamento 111 cuyo salario es igual a alguno de los salarios del departamento 112. ¿Cómo lo obtendrías con el predicado "any"?

"in":

```

drop view if exists ej7b3;
create view ej7b3 as
select nomem from temple where numde=111 and salar in(select salar from temple
where numde=112) order by nomem asc;

```

```

/*****

```

```

select * from empresa.ej7b3;

```

	nomem
▶	LARA, LUCRECIA
	SANZ, LAVINIA

```

drop procedure if exists procedure_ej7b3;
delimiter //
create procedure procedure_ej7b3()
begin
    select nomem from temple where numde=111 and salar in(select salar from temple
where numde=112) order by nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej7b3();

```

	nomem
▶	LARA, LUCRECIA
	SANZ, LAVINIA

"any":

```

drop view if exists ej71b3;
create view ej71b3 as
select nomem from temple where numde=111 and salar=any(select salar from temple
where numde = 112) order by nomem asc;

/*****/

select * from empresa.ej71b3;

```

	nomem
▶	LARA, LUCRECIA
	SANZ, LAVINIA

```

drop procedure if exists procedure_ej71b3;
delimiter //
create procedure procedure_ej71b3()
begin
    select nomem from temple where numde=111 and salar=any(select salar from temple
where numde = 112) order by nomem asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej71b3();

```

	nomem
▶	LARA, LUCRECIA
	SANZ, LAVINIA

8.a. Obtener, por orden alfabético, los nombres y comisiones de los empleados del departamento 110 si hay en él algún empleado que tenga comisión.

```
drop view if exists ej81b3;
create view ej81b3 as
select nomem, comis from temple where numde=110 and comis is not null order by
nomem asc;
```

```
/*****
```

```
select * from empresa.ej81b3;
```

	nomem	comis
▶	PEREZ, MARCOS	500.00

```
drop procedure if exists procedure_ej81b3;
delimiter //
create procedure procedure_ej81b3(in numde_intr int)
begin
    select nomem, comis from temple where numde=numde_intr and comis is not null
    order by nomem asc;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej81b3(110);
```

	nomem	comis
▶	PEREZ, MARCOS	500.00

8.b. Obtener, por orden alfabético, los nombres de los departamentos que tienen algún empleado con comisión.

```
drop view if exists ej82b3;
create view ej82b3 as
select distinct tdepto.nomde from tdepto join temple on temple.numde=tdepto.numde
where temple.comis is not null order by tdepto.nomde asc;
```

```
/*****
```

```
select * from empresa.ej82b3;
```

nomde
NOMINAS
SECTOR INDUSTRIAL
SECTOR SERVICIOS

```
drop procedure if exists procedure_ej82b3;
delimiter //
create procedure procedure_ej82b3()
begin
    select distinct tdepto.nomde from tdepto join temple on
    temple.numde=tdepto.numde where temple.comis is not null order by tdepto.nomde
    asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej82b3();
```

nomde
NOMINAS
SECTOR INDUSTRIAL
SECTOR SERVICIOS

9. Obtener el nombre de cada departamento, junto con el nombre del centro al que pertenece.

```
drop view if exists ej9b3;
create view ej9b3 as
select tdepto.nomde, tcentr.nomce from tdepto join tcentr on
tdepto.numce=tcentr.numce;

/*****/

select * from empresa.ej9b3;
```

nomde	nomce
DIRECCIÓN GENERAL	SEDE CENTRAL
ORGANIZACION	SEDE CENTRAL
PERSONAL	SEDE CENTRAL
NOMINAS	RELACION CON CLIENTES
SECTOR INDUSTRIAL	RELACION CON CLIENTES
SECTOR SERVICIOS	RELACION CON CLIENTES

```

drop procedure if exists procedure_ej9b3;
delimiter //
create procedure procedure_ej9b3()
begin
    select tdepto.nomde, tcentr.nomce from tdepto join tcentr on
tdepto.numce=tcentr.numce;
end //
delimiter ;

/*****/

call empresa.procedure_ej9b3();

```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	ORGANIZACION	SEDE CENTRAL
	PERSONAL	SEDE CENTRAL
	NOMINAS	RELACION CON CLIENTES
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES

10. Obtener el nombre de cada empleado, el número y el nombre del departamento al que pertenece y el nombre del correspondiente centro.

```

drop view if exists ej10b3;
create view ej10b3 as
select
    temple.nomem as empleado,
    temple.numem as numero_empleado,
    tdepto.nomde as departamento,
    tdepto.numde as numero_departamento,
    tcentr.nomce as centro
from temple join tdepto on temple.numde=tdepto.numde join tcentr on
tdepto.numce=tcentr.numce;

/*****/

select * from empresa.ej10b3;

```

	empleado	numero_empleado	departamento	numero_departamento	centro
►	LOPEZ, ANTONIO	260	DIRECCIÓN GENERAL	100	SEDE CENTRAL
	FLOR, DOROTEA	280	ORGANIZACION	120	SEDE CENTRAL
	GIL, GLORIA	290	ORGANIZACION	120	SEDE CENTRAL
	PONS, CESAR	110	PERSONAL	121	SEDE CENTRAL
	PEREZ, JULIO	150	PERSONAL	121	SEDE CENTRAL
	RUIZ, FABIOLA	370	PERSONAL	121	SEDE CENTRAL
	PEREZ, MARCOS	180	NOMINAS	110	RELACION CON CLIENTES
	VEIGA, JULIANA	190	NOMINAS	110	RELACION CON CLIENTES
	SANZ, LAVINIA	240	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	LARA, LUCRECIA	360	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	TEROL, LUCIANO	130	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	GARCIA, OCTAVIO	270	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	DIEZ, AMELIA	330	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	MARTÍN, MICAELA	380	SECTOR SERVICIOS	112	RELACION CON CLIENTES

```

drop procedure if exists procedure_ej10b3;
delimiter //
create procedure procedure_ej10b3()
begin
    select
        temple.nomem as empleado,
        temple.numem as numero_empleado,
        tdepto.nomde as departamento,
        tdepto.numde as numero_departamento,
        tcentr.nomce as centro
    from temple join tdepto on temple.numde=tdepto.numde join tcentr on
tdepto.numce=tcentr.numce;
end //
delimiter ;

/*****/

call empresa.procedure_ej10b3();

```

	empleado	numero_empleado	departamento	numero_departamento	centro
►	LOPEZ, ANTONIO	260	DIRECCIÓN GENERAL	100	SEDE CENTRAL
	FLOR, DOROTEA	280	ORGANIZACION	120	SEDE CENTRAL
	GIL, GLORIA	290	ORGANIZACION	120	SEDE CENTRAL
	PONS, CESAR	110	PERSONAL	121	SEDE CENTRAL
	PEREZ, JULIO	150	PERSONAL	121	SEDE CENTRAL
	RUIZ, FABIOLA	370	PERSONAL	121	SEDE CENTRAL
	PEREZ, MARCOS	180	NOMINAS	110	RELACION CON CLIENTES
	VEIGA, JULIANA	190	NOMINAS	110	RELACION CON CLIENTES
	SANZ, LAVINIA	240	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	LARA, LUCRECIA	360	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	TEROL, LUCIANO	130	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	GARCIA, OCTAVIO	270	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	DIEZ, AMELIA	330	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	MARTÍN, MICAELA	380	SECTOR SERVICIOS	112	RELACION CON CLIENTES

11. Obtener, para el departamento 110, su nombre, nombre del centro y dirección, junto con el nombre del empleado que sea el director.

```
drop view if exists ej11b3;
create view ej11b3 as
select d.nomde, c.nomce, c.señas, e.nomem from tdepto d join tcentr c on
d.numce=c.numce join temple e on d.direc=e.numem where d.numde=110;

/*****/

select * from empresa.ej11b3;
```

	nomde	nomce	señas	nomem
▶	NOMINAS	RELACION CON CLIENTES	C. ATOCHA, 405, MADRID	GIL, GLORIA

```
drop procedure if exists procedure_ej11b3;
delimiter //
create procedure procedure_ej11b3(in numde_intr int)
begin
    select d.nomde, c.nomce, c.señas, e.nomem from tdepto d join tcentr c on
d.numce=c.numce join temple e on d.direc=e.numem where d.numde=numde_intr;
end //
delimiter ;

/*****/

call empresa.procedure_ej11b3(110);
```

	nomde	nomce	señas	nomem
▶	NOMINAS	RELACION CON CLIENTES	C. ATOCHA, 405, MADRID	GIL, GLORIA

BLOQUE 4

1. Hallar, por departamentos, la edad en años cumplidos del empleado más mayor, así como la edad media del mismo (el empleado debe tener comisión). Ordenar el resultado por edades.

```
drop view if exists ej1b4;
create view ej1b4 as
select
    tdepto.nomde,
    max(datediff(current_date, temple.fecna)/365) as edad_mas_viejo,
    avg(datediff(current_date, temple.fecna)/365) as edad_media
from tdepto join temple on tdepto.numde=temple.numde where temple.comis is not
null group by tdepto.nomde order by edad_mas_viejo desc;

/*****/

select * from empresa.ej1b4;
```


	nomde	edad_mas_viejo	edad_media
►	SECTOR SERVICIOS	56.7041	46.81733333
	NOMINAS	43.3370	43.33700000
	SECTOR INDUSTRIAL	42.5562	40.89180000

Alternativa (creo que más correcta) para que los números salgan redondeados:

```
drop view if exists ej102b4;
create view ej102b4 as
select
  tdepto.nomde,
  round(max(datediff(current_date, temple.fecna)/365),0) as edad_mas_viejo,
  round(avg(datediff(current_date, temple.fecna)/365),0) as edad_media
from tdepto join temple on tdepto.numde=temple.numde where temple.comis is not
null group by tdepto.nomde order by edad_mas_viejo desc;
```

/*****

```
select * from empresa.ej102b4;
```

	nomde	edad_mas_viejo	edad_media
►	SECTOR SERVICIOS	57	47
	NOMINAS	43	43
	SECTOR INDUSTRIAL	43	41

```
drop procedure if exists procedure_ej1b4;
delimiter //
create procedure procedure_ej1b4()
begin
  select
    tdepto.nomde,
    round(max(datediff(current_date, temple.fecna)/365),0) as edad_mas_viejo,
    round(avg(datediff(current_date, temple.fecna)/365),0) as edad_media
  from tdepto join temple on tdepto.numde=temple.numde where temple.comis is not
  null group by tdepto.nomde order by edad_mas_viejo desc;
end //
delimiter ;
```

/*****

```
call empresa.procedure_ej1b4();
```

	nomde	edad_mas_viejo	edad_media
►	SECTOR SERVICIOS	57	47
	NOMINAS	43	43
	SECTOR INDUSTRIAL	43	41

2. Agrupando por departamento y número de hijos, hallar cuántos empleados hay en cada grupo.

```
drop view if exists ej2b4;
create view ej2b4 as
select tdepto.nomde, temple.numhi, count(*) as numempleados from temple inner join
tdepto on temple.numde=tdepto.numde group by tdepto.nomde, temple.numhi;

/*****/

select * from empresa.ej2b4;
```

	nomde	numhi	numempleados
►	DIRECCIÓN GENERAL	6	1
	NOMINAS	2	1
	NOMINAS	4	1
	SECTOR INDUSTRIAL	3	1
	SECTOR INDUSTRIAL	2	1
	SECTOR SERVICIOS	2	1
	SECTOR SERVICIOS	3	1
	SECTOR SERVICIOS	0	2
	ORGANIZACION	5	1
	ORGANIZACION	3	1
	PERSONAL	3	1
	PERSONAL	1	2

```
drop procedure if exists procedure_ej2b4;
delimiter //
create procedure procedure_ej2b4()
begin
select tdepto.nomde, temple.numhi, count(*) as numempleados from temple inner
join tdepto on temple.numde=tdepto.numde group by tdepto.nomde, temple.numhi;
end //
delimiter ;

/*****/

call empresa.procedure_ej2b4();
```

	nomde	numhi	numempleados
►	DIRECCIÓN GENERAL	6	1
	NOMINAS	2	1
	NOMINAS	4	1
	SECTOR INDUSTRIAL	3	1
	SECTOR INDUSTRIAL	2	1
	SECTOR SERVICIOS	2	1
	SECTOR SERVICIOS	3	1
	SECTOR SERVICIOS	0	2
	ORGANIZACION	5	1
	ORGANIZACION	3	1
	PERSONAL	3	1
	PERSONAL	1	2

3. Para cada extensión telefónica, hallar cuántos empleados la usan y el salario medio de estos.

```
drop view if exists ej3b4;
create view ej3b4 as
select extel, count(*) as numempleados, avg(salar) as salariomedio from temple
group by extel;
```

```
/*****
```

```
select * from empresa.ej3b4;
```

	extel	numempleados	salariomedio
►	350	2	1250.000000
	810	1	1200.000000
	340	1	2200.000000
	508	1	1800.000000
	760	1	2200.000000
	220	1	2200.000000
	800	1	2200.000000
	410	1	1500.000000
	910	1	2200.000000
	850	1	1500.000000
	750	1	1800.000000
	360	1	1800.000000
	880	1	1800.000000
	8765	1	4000.000000

```
drop procedure if exists procedure_ej3b4;
delimiter //
create procedure procedure_ej3b4()
begin
    select extel, count(*) as numempleados, avg(salar) as salariomedio from temple
    group by extel;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej3b4();
```

	extel	numempleados	salariomedio
▶	350	2	1250.000000
	810	1	1200.000000
	340	1	2200.000000
	508	1	1800.000000
	760	1	2200.000000
	220	1	2200.000000
	800	1	2200.000000
	410	1	1500.000000
	910	1	2200.000000
	850	1	1500.000000
	750	1	1800.000000
	360	1	1800.000000
	880	1	1800.000000
	8765	1	4000.000000

4. Para los departamentos cuyo salario medio supera al de la empresa, hallar cuántas extensiones telefónicas tienen. Se debe mostrar el número de departamento ("numde") y el número de extensiones telefónicas distintas que tiene cada uno de ellos.

```
drop view if exists ej4b4;
create view ej4b4 as
select tdepto.numde, count(distinct temple.extel) as numextensiones from tdepto
join temple on tdepto.numde=temple.numde where temple.salar>(select avg(salar)
from temple) group by tdepto.numde having avg(temple.salar)>(select avg(salar)
from temple);
```

```
/*****
```

```
select * from empresa.ej4b4;
```

	numde	numextensiones
▶	100	1
	111	1
	112	1
	120	1
	121	1

```
drop procedure if exists procedure_ej4b4;
delimiter //
create procedure procedure_ej4b4()
begin
    select tdepto.numde, count(distinct temple.extel) as numextensiones from tdepto
join temple on tdepto.numde=temple.numde where temple.salar>(select avg(salar)
from temple) group by tdepto.numde having avg(temple.salar)>(select avg(salar)
from temple);
```

```

end //
delimiter ;

/*****

call empresa.procedure_ej4b4();

```

	numde	numextensiones
▶	100	1
	111	1
	112	1
	120	1
	121	1

5. Hallar el máximo valor de la suma de los salarios de los departamentos. Queremos obtener el número de departamentos ("numde") y la suma de sus salarios, pero del departamento cuya suma de salarios es la mayor de todas.

```

drop view if exists ej5b4;
create view ej5b4 as
select numde, sum(salar) as suma_salarios from temple where numde=(select numde
from temple group by numde order by sum(salar) desc limit 1) group by numde;

/*****

select * from empresa.ej5b4;

```

	numde	suma_salarios
▶	112	6700.00

```

drop procedure if exists procedure_ej5b4;
delimiter //
create procedure procedure_ej5b4()
begin
    select numde, sum(salar) as suma_salarios from temple where numde=(select numde
from temple group by numde order by sum(salar) desc limit 1) group by numde;
end //
delimiter ;

/*****

call empresa.procedure_ej5b4();

```

	numde	suma_salarios
▶	112	6700.00

6. Para cada departamento con presupuesto inferior a 10000 euros, hallar el nombre del centro donde está ubicado y el máximo salario de sus empleados, sólo si este excede 1500 euros. Clasificar alfabéticamente por nombre de departamento. Hacer el ejercicio de dos maneras, con producto cartesiano y con "join".

Producto cartesiano:

```
drop view if exists ej61b4;
create view ej61b4 as
select tc.nomce, td.nomde, max(te.salar) from tcentr tc, tdepto td, temple te
where tc.numce=td.numce and td.numde=te.numde and td.presu<10000 and te.salar>1500
group by tc.nomce, td.nomde order by td.nomde asc;

/*****/

select * from empresa.ej61b4;
```

	nomce	nomde	max(te.salar)
►	RELACION CON CLIENTES	NOMINAS	1800.00
	SEDE CENTRAL	ORGANIZACION	2200.00
	SEDE CENTRAL	PERSONAL	2200.00
	RELACION CON CLIENTES	SECTOR SERVICIOS	2200.00

```
drop procedure if exists procedure_ej61b4;
delimiter //
create procedure procedure_ej61b4()
begin
    select tc.nomce, td.nomde, max(te.salar) from tcentr tc, tdepto td, temple te
    where tc.numce=td.numce and td.numde=te.numde and td.presu<10000 and te.salar>1500
    group by tc.nomce, td.nomde order by td.nomde asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej61b4();
```

	nomce	nomde	max(te.salar)
►	RELACION CON CLIENTES	NOMINAS	1800.00
	SEDE CENTRAL	ORGANIZACION	2200.00
	SEDE CENTRAL	PERSONAL	2200.00
	RELACION CON CLIENTES	SECTOR SERVICIOS	2200.00

"join":

```
drop view if exists ej62b4;
create view ej62b4 as
```

```
select tc.nomce, td.nomde, max(te.salar) from tcentr tc join tdepto td on
tc.numce=td.numce join temple te on td.numde=te.numde where td.presu<10000 and
te.salar>1500 group by tc.nomce, td.nomde order by td.nomde asc;
```

```
/*****
```

```
select * from empresa.ej62b4;
```

	nomce	nomde	max(te.salar)
►	RELACION CON CLIENTES	NOMINAS	1800.00
	SEDE CENTRAL	ORGANIZACION	2200.00
	SEDE CENTRAL	PERSONAL	2200.00
	RELACION CON CLIENTES	SECTOR SERVICIOS	2200.00

```
drop procedure if exists procedure_ej62b4;
delimiter //
create procedure procedure_ej62b4()
begin
    select tc.nomce, td.nomde, max(te.salar) from tcentr tc join tdepto td on
    tc.numce=td.numce join temple te on td.numde=te.numde where td.presu<10000 and
    te.salar>1500 group by tc.nomce, td.nomde order by td.nomde asc;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej62b4();
```

	nomce	nomde	max(te.salar)
►	RELACION CON CLIENTES	NOMINAS	1800.00
	SEDE CENTRAL	ORGANIZACION	2200.00
	SEDE CENTRAL	PERSONAL	2200.00
	RELACION CON CLIENTES	SECTOR SERVICIOS	2200.00

7. Hallar, por orden alfabético, los nombres de los departamentos que dependen de los que tienen un presupuesto inferior a 10000 euros. Realizar la consulta de cuatro formas distintas: con predicado "in", con predicado "any", con producto cartesiano y con "join".

"in":

```
drop view if exists ej71b4;
create view ej71b4 as
select nomde from tdepto where presu<10000 and depde in(select numde from tdepto)
order by nomde asc;
```

```
/*****
```

```
select * from empresa.ej71b4;
```

	nomde
►	NOMINAS
	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

```

drop procedure if exists ej71b4;
delimiter //
create procedure procedure_ej71b4()
begin
    select nomde from tdepto where presu<10000 and depde in(select numde from
tdepto) order by nomde asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej71b4();

```

	nomde
►	NOMINAS
	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

"any":

```

drop view if exists ej72b4;
create view ej72b4 as
select nomde from tdepto where presu<10000 and depde=any(select numde from tdepto)
order by nomde asc;

/*****/

select * from empresa.ej72b4;

```

	nomde
►	NOMINAS
	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

```

drop procedure if exists ej72b4;
delimiter //
create procedure procedure_ej72b4()

```



```

begin
  select nomde from tdepto where presu<10000 and depde=any(select numde from
tdepto) order by nomde asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej72b4();

```

	nomde
►	NOMINAS
	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

Producto cartesiano:

```

drop view if exists ej73b4;
create view ej73b4 as
select distinct t2.nomde from tdepto t1, tdepto t2 where t1.presu<10000 and
t1.numde=t2.depde order by t2.nomde asc;

/*****/

select * from empresa.ej73b4;

```

	nomde
►	NOMINAS
	PERSONAL
	SECTOR INDUSTRIAL
	SECTOR SERVICIOS

```

drop procedure if exists ej73b4;
delimiter //
create procedure procedure_ej73b4()
begin
  select distinct t2.nomde from tdepto t1, tdepto t2 where t1.presu<10000 and
t1.numde=t2.depde order by t2.nomde asc;
end //
delimiter ;

/*****/

call empresa.procedure_ej73b4();

```

	nomde
►	NOMINAS
	PERSONAL
	SECTOR INDUSTRIAL
	SECTOR SERVICIOS

"join":

```
drop view if exists ej74b4;
create view ej74b4 as
select distinct t2.nomde from tdepto t1 join tdepto t2 on t1.numde=t2.depde where
t1.presu<10000 order by t2.nomde asc;
```

```
/*****
```

```
select * from empresa.ej74b4;
```

	nomde
►	NOMINAS
	PERSONAL
	SECTOR INDUSTRIAL
	SECTOR SERVICIOS

```
drop procedure if exists ej74b4;
delimiter //
create procedure procedure_ej74b4()
begin
    select distinct t2.nomde from tdepto t1 join tdepto t2 on t1.numde=t2.depde
where t1.presu<10000 order by t2.nomde asc;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej74b4();
```

	nomde
►	NOMINAS
	PERSONAL
	SECTOR INDUSTRIAL
	SECTOR SERVICIOS

8. Obtener, por orden alfabético, los nombres de los departamentos cuyo presupuesto es inferior al 10% de la suma de los salarios anuales (14 pagas) de sus empleados.

```
drop view if exists ej8b4;
create view ej8b4 as
```

```
select tdepto.nomde from tdepto inner join(select numde, sum(salar*14) as
sum_salar from temple group by numde) as salarios on tdepto.numde=salarios.numde
where tdepto.presu<salarios.sum_salar*0.1 order by tdepto.nomde asc;
```

```
/*****
```

```
select * from empresa.ej8b4;
```

	nomde
►	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

```
drop procedure if exists procedure_ej8b4;
delimiter //
create procedure procedure_ej8b4()
begin
    select tdepto.nomde from tdepto inner join(select numde, sum(salar*14) as
sum_salar from temple group by numde) as salarios on tdepto.numde=salarios.numde
where tdepto.presu<salarios.sum_salar*0.1 order by tdepto.nomde asc;
end //
delimiter ;
```

```
/*****
```

```
call empresa.procedure_ej8b4();
```

	nomde
►	ORGANIZACION
	PERSONAL
	SECTOR SERVICIOS

***Los ejercicios 9, 10 y 11 son los mismos que los del bloque 3, pero ahora debemos utilizar bien las cláusulas "right" y "left" en el "join" para obtener un resultado válido en todos los casos.**

9. Obtener el nombre de cada departamento, junto con el nombre del centro al que pertenece.

```
drop view if exists ej9b4;
create view ej9b4 as
select tdepto.nomde, tcentr.nomce from tdepto left join tcentr on
tdepto.numce=tcentr.numce;
```

```
/*****
```

```
select * from empresa.ej9b4;
```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	NOMINAS	RELACION CON CLIENTES
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES
	ORGANIZACION	SEDE CENTRAL
	PERSONAL	SEDE CENTRAL

```

drop procedure if exists procedure_ej9b4;
delimiter //
create procedure procedure_ej9b4()
begin
    select tdepto.nomde, tcentr.nomce from tdepto left join tcentr on
tdepto.numce=tcentr.numce;
end //
delimiter ;

/*****/

call empresa.procedure_ej9b4();

```

	nomde	nomce
►	DIRECCIÓN GENERAL	SEDE CENTRAL
	NOMINAS	RELACION CON CLIENTES
	SECTOR INDUSTRIAL	RELACION CON CLIENTES
	SECTOR SERVICIOS	RELACION CON CLIENTES
	ORGANIZACION	SEDE CENTRAL
	PERSONAL	SEDE CENTRAL

10. Obtener el nombre de cada empleado, el número y el nombre del departamento al que pertenece y el nombre del correspondiente centro.

```

drop view if exists ej10b4;
create view ej10b4 as
select
    temple.nomem as empleado,
    temple.numem as numero_empleado,
    tdepto.nomde as departamento,
    tdepto.numde as numero_departamento,
    tcentr.nomce as centro
from temple left join tdepto on temple.numde=tdepto.numde left join tcentr on
tdepto.numce=tcentr.numce;

/*****/

select * from empresa.ej10b4;

```

	empleado	numero_empleado	departamento	numero_departamento	centro
►	PONS, CESAR	110	PERSONAL	121	SEDE CENTRAL
	TEROL, LUCIANO	130	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	PEREZ, JULIO	150	PERSONAL	121	SEDE CENTRAL
	PEREZ, MARCOS	180	NOMINAS	110	RELACION CON CLIENTES
	VEIGA, JULIANA	190	NOMINAS	110	RELACION CON CLIENTES
	SANZ, LAVINIA	240	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	LOPEZ, ANTONIO	260	DIRECCIÓN GENERAL	100	SEDE CENTRAL
	GARCIA, OCTAVIO	270	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	FLOR, DOROTEA	280	ORGANIZACION	120	SEDE CENTRAL
	GIL, GLORIA	290	ORGANIZACION	120	SEDE CENTRAL
	DIEZ, AMELIA	330	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	LARA, LUCRECIA	360	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	RUIZ, FABIOLA	370	PERSONAL	121	SEDE CENTRAL
	MARTÍN, MICAELA	380	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	Fernando Ramirez...	4	NULL	NULL	NULL

```

drop procedure if exists procedure_ej10b4;
delimiter //
create procedure procedure_ej10b4()
begin
    select
        temple.nomem as empleado,
        temple.numem as numero_empleado,
        tdepto.nomde as departamento,
        tdepto.numde as numero_departamento,
        tcentr.nomce as centro
    from temple left join tdepto on temple.numde=tdepto.numde left join tcentr on
tdepto.numce=tcentr.numce;
end //
delimiter ;

/*****/

call empresa.procedure_ej10b4();

```

	empleado	numero_empleado	departamento	numero_departamento	centro
►	PONS, CESAR	110	PERSONAL	121	SEDE CENTRAL
	TEROL, LUCIANO	130	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	PEREZ, JULIO	150	PERSONAL	121	SEDE CENTRAL
	PEREZ, MARCOS	180	NOMINAS	110	RELACION CON CLIENTES
	VEIGA, JULIANA	190	NOMINAS	110	RELACION CON CLIENTES
	SANZ, LAVINIA	240	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	LOPEZ, ANTONIO	260	DIRECCIÓN GENERAL	100	SEDE CENTRAL
	GARCIA, OCTAVIO	270	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	FLOR, DOROTEA	280	ORGANIZACION	120	SEDE CENTRAL
	GIL, GLORIA	290	ORGANIZACION	120	SEDE CENTRAL
	DIEZ, AMELIA	330	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	LARA, LUCRECIA	360	SECTOR INDUSTRIAL	111	RELACION CON CLIENTES
	RUIZ, FABIOLA	370	PERSONAL	121	SEDE CENTRAL
	MARTÍN, MICAELA	380	SECTOR SERVICIOS	112	RELACION CON CLIENTES
	Fernando Ramirez...	4	NULL	NULL	NULL

11. Obtener, para el departamento 110, su nombre, nombre del centro y dirección, junto con el nombre del empleado que sea el director.

```
drop view if exists ej11b4;
create view ej11b4 as
select d.nomde, c.nomce, c.señas, e.nomem from tdepto d join tcentr c on
d.numce=c.numce left join temple e on d.direc=e.numem where d.numde=110;

/*****/

select * from empresa.ej11b4;
```

	nomde	nomce	señas	nomem
►	NOMINAS	RELACION CON CLIENTES	C. ATOCHA, 405, MADRID	GIL, GLORIA

```
drop procedure if exists procedure_ej11b4;
delimiter //
create procedure procedure_ej11b4(in numde_intr int)
begin
select d.nomde, c.nomce, c.señas, e.nomem from tdepto d join tcentr c on
d.numce=c.numce left join temple e on d.direc=e.numem where d.numde=numde_intr;
end //
delimiter ;

/*****/

call empresa.procedure_ej11b4(110);
```

	nomde	nomce	señas	nomem
►	NOMINAS	RELACION CON CLIENTES	C. ATOCHA, 405, MADRID	GIL, GLORIA

12. Utilizando las tablas "tcentr" y "tdepto", plantear un enunciado, de modo que para obtener la correspondiente consulta sea necesario utilizar la cláusula "right join". Añadir en modo visual las filas necesarias a las tablas para así poder comprobar el resultado. La solución del ejercicio consta de un enunciado y una consulta.

Posible enunciado:

Consulta:

```
drop view if exists ej12b4;
create view ej12b4 as

/*****/
```

```
select * from empresa.ej12b4;
```



```
drop procedure if exists procedure_ej12b4;
delimiter //
create procedure procedure_ej12b4()
begin

end //
delimiter ;

/*****/

call empresa.procedure_ej12b4();
```



13. Utilizando las tablas "tdepto" y "temple", plantear un enunciado, de modo que para obtener la correspondiente consulta sea necesario utilizar la cláusula "full join". Añadir en modo visual las filas necesarias a las tablas para así poder comprobar el resultado. La solución del ejercicio consta de un enunciado y una consulta.

Posible enunciado:

Consulta:

```
drop view if exists ej13b4;
create view ej13b4 as

/*****/

select * from empresa.ej13b4;
```



```
drop procedure if exists procedure_ej13b4;
delimiter //
create procedure procedure_ej13b4()
begin

end //
delimiter ;
```

```
/*  
*****  
*/
```

```
call empresa.procedure_ej13b4();
```

