



Tiempo necesario para terminar: Puntos:  
115:38 8/10

1. Agregar un nuevo empleado de nombre "Juan Perez", sueldo 5000, legajo 200, domicilio "Saavedra 123" y localidad de código 12. Este empleado comenzó a trabajar el día 21 de Enero de 2014. 1 / 1 pto

```
insert into Empleado (legajo_emp, nombre, domicilio, cod_loc, sueldo,
fecha_ingreso)
values (200, "Juan Perez", "Saavedra 123", 12, 5000, "2014-01-21");
```

2. Imprimir un listado de ventas del mes de Enero de 2012, indicando: fecha, nombre de empleado, nombre de cliente, descripción y precio del electrodoméstico. Ordenar las ventas por fecha (más recientes a más antiguas) y, para misma fecha, por nombre de empleado alfabéticamente. 1 / 1 pto

```
select v.fecha, e.nombre, c.nombre, ele.descripcion, ele.precio
from Venta v join Cliente c on c.nro_cli=v.nro_cli
join Empleado e on e.legajo_emp=v.legajo_emp
join Electrodomestico ele on ele.cod_elect=v.cod_elect
where v.fecha between "2012-01-01" and "2012-01-31"
group by v.fecha desc, e.nombre asc;
```

3. Listar el nombre y sueldo de aquellos empleados que hayan efectuado más de 10 ventas. 1 / 1 pto

```
select e.nombre, e.sueldo
from Empleado e
where e.legajo_emp in (select v.legajo_emp
from Venta v
group by v.legajo_emp
having count(v.legajo_emp)>10);
```

4. Modificar a los empleados de la localidad "Ramos Mejía", cuyo nombre termina con la letra "S", de manera que su sueldo se incremente \$200. 1 / 1 pto

```
update Empleado e
set e.sueldo=e.sueldo +200
where e.nombre like "%S"
and e.cod_loc in (select l.cod_loc
from Localidad l
where l.descripcion="Ramos Mejia");
```

5. Indicar el sueldo total de los empleados por cada localidad (descripción). 1 / 1 pto

```
select lo.descripcion as 'Localidad' , sum(e.sueldo) as 'Sueldo Total'
from Empleado e join Localidad lo on e.cod_loc=lo.cod_loc
group by lo.cod_loc, lo.descripcion;
```

6. Eliminar a todos los empleados que no hayan realizado venta alguna en el año 2014. 1 / 1 pto

```
-- Si no existe al menos una venta de ese empleado en 2014 lo borra (Lo
formule así)
delete from Empleado e
where not exists (select 1
from Venta v
where e.legajo_emp=v.legajo_emp
and v.fecha between "2014-01-01" and "2014-12-31");
```

7. Listar el nombre de los empleados que hayan vendido todos los electrodomésticos. 0 / 1 pto

```
select e.nombre
from Empleado e
join Venta v on e.legajo_emp=v.legajo_emp
group by e.legajo_emp, e.nombre
having count(v.cod_elect)=(select count (el.cod_elect)
from Electrodomestico el);
```

El count(v.cod\_elect) debe tener un DISTINCT para que la consulta funcione

8. Mostrar el nombre y domicilio de aquellos empleados que realizaron alguna venta el mismo día que ingresaron a la empresa. 1 / 1 pto

```
select e.nombre, e.domicilio
from Empleado e
where exists ( select 1
from Venta v
where e.legajo_emp= v.legajo_emp
and v.fecha=e.fecha_ingreso);
```

9. Mostrar código y descripción de los electrodomésticos más caros (máximo precio entre todos los electrodomésticos) 1 / 1 pto

```
select e.cod_elect, e.descripcion
from Electrodomestico e
where e.precio = (select max(e2.precio)
from Electrodomestico e2 );
```

10. Cuántas ventas se realizaron en total, durante los años 2015 y 2020 únicamente, de aquellos electrodomésticos cuyo precio es mayor a \$20000? 0 / 1 pto

```
select count(*) as 'Ventas totales de mas de 20 mil'
from Venta v join Electrodomestico e on v.cod_elect = e.cod_elect
where e.precio>20000 and
v.fecha between "2015-01-01" and "2015-12-31"
or v.fecha between "2020-01-01" and "2020-12-31";
```

Al no establecer la precedencia del OR en las comparaciones de fechas con paréntesis, la lógica termina siendo incorrecta