

TALLER

CONSULTAS BASE DE DATOS SUMERCA

NOMBRE: JULIO EDUARDO LARA PUPO

MATERIA: PROCEDIMIENTOS ALMACENADOS

PROFESOR(A): MARY LUZ RUBIANO

PROGRAMA: TECNOLOGIA EN DESARROLLO DE SOFTWARE

Contenido

EJERCICIOS 4

1-SELECCIONAR LA SUCURSAL QUE MÁS CANTIDAD DE PRODUCTOS DE UN TIPO DETERMINADO SE HA VENDIDO EN EL ÚLTIMO MES	4
2-SELECCIONAR EL TIPO Y PRODUCTO MÁS VENDIDO DEL AÑO 2016 EN CUANTO A CANTIDADES SE REFIERE.	4
3-SELECCIONAR EL VENDEDOR QUE MÁS VENTAS HA REALIZADO EN UN MES, RETORNANDO EL NOMBRE DEL VENDEDOR, EL MES Y LA CANTIDAD VENDIDA.....	5
6-SELECCIONAR EL PRODUCTO POR TIPO QUE MÁS SE HA VENDIDO EN CUANTO AL VALOR SE REFIERE.	6
8-CREAR LA CONSULTA PARA IDENTIFICAR LOS PRODUCTOS MÁS VENDIDOS DURANTE EL ÚLTIMO AÑO, DONDE SE DEBE RETORNAR EL NOMBRE DEL PRODUCTO, LA CANTIDAD Y EL AÑO.	6
10-CONSTRUIR LA CONSULTA QUE PERMITA CONOCER LAS VENTAS TOTALES DE CADA VENDEDOR EN CADA SUCURSAL, RETORNANDO EL NOMBRE DEL VENDEDOR, SUCURSAL Y VALOR TOTAL DE VENTAS POR CADA VENDEDOR ORDENADO DESCENDENTEMENTE POR VALOR.....	7
12-CONSTRUIR UNA CONSULTA QUE LISTE EL ID DEL VENDEDOR, EL NOMBRE DEL VENDEDOR CON SUS APELLIDOS EN EL MISMO CAMPO, SU IDENTIFICACIÓN, SU SUCURSAL ORDENADOS POR ALFABÉTICAMENTE DE MENOR A MAYOR.	8
15-CUÁLES ES EL VALOR PROMEDIO DE LAS VENTAS DE CADA SUCURSAL, DEBE RETORNAR EL NOMBRE DE LA SUCURSAL Y EL VALOR PROMEDIO DE LAS VENTAS.	8
16-LISTAR LA CANTIDAD DE PRODUCTOS QUE HAY EN EL INVENTARIO, RETORNANDO EL ID DEL PRODUCTO, NOMBRE DE PRODUCTO Y CANTIDAD DE PRODUCTO EXISTENTES ASCENDENTEMENTE.	9

EJERCICIOS QUE REQUIRIERON MODIFICACION E INSERCIÓN DE DATOS ADICIONALES (5)... 10

5-SELECCIONAR LOS PRODUCTOS QUE SE HAN VENDIDO MÁS DE 10 VECES Y ORDENAR EL RESULTADO DE FORMA DESCENDENTE.	10
9-CREAR UNA CONSULTA QUE MUESTRE EL CÓDIGO Y NOMBRE DEL PRODUCTO DE TODOS LOS PRODUCTOS CUYO VALOR DE COMPRA ES SUPERIOR A \$50.000 MIL PESOS.....	11
13-CUÁLES SON LOS PRODUCTOS VENDIDOS DE CADA VENDEDOR, TRAYENDO EL CÓDIGO DEL VENDEDOR, NOMBRE DEL VENDEDOR, LA CANTIDAD TOTAL DE PRODUCTOS Y EL VALOR TOTAL DE LAS VENTAS POR DICHOS PRODUCTOS.	11
14-CONSTRUIR UNA CONSULTA QUE PERMITA CONOCER EL SALARIO TOTAL A PAGAR A CADA VENDEDOR TENIENDO EN CUENTA QUE POR VENTAS SUPERIORES A: a-\$50.000 mil pesos obtendrá una comisión del 2% del total de la venta, b-\$100.000 mil pesos obtendrá una comisión del 5% del total de la venta, c-\$150.000 millón de pesos obtendrá una comisión del 10% del total de la venta Y d-Para todas las ventas menores a \$50.000 la	

comisión es del 1% del total de la venta Nota: El salario básico de todos los vendedores es de \$1.200.000 mil pesos: 12

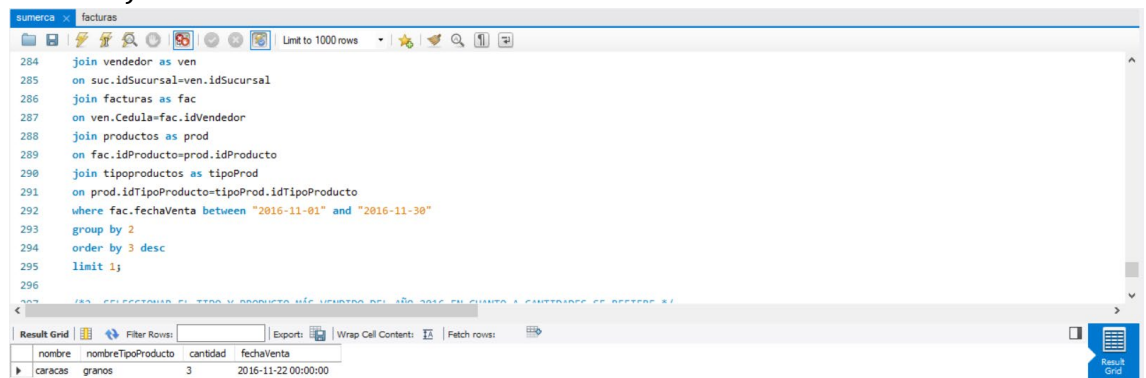
17-Obtener el listado de clientes que no hayan realizado compras durante el último año pero sus compras totales son iguales o superiores a 1.000.000, debe retornar el id del cliente, nombre-apellido, la fecha y el valor de la última compra realizada. 14

EJERCICIOS

1-SELECCIONAR LA SUCURSAL QUE MÁS CANTIDAD DE PRODUCTOS DE UN TIPO DETERMINADO SE HA VENDIDO EN EL ÚLTIMO MES

/*1. SELECCIONAR LA SUCURSAL QUE MÁS CANTIDAD DE PRODUCTOS DE UN TIPO DETERMINADO SE HA VENDIDO EN EL ÚLTIMO MES.*/*

```
select suc.nombre, tipoProd.nombreTipoProducto,  
sum(fac.cantidad) as cantidad, fac.fechaVenta from sucursal as  
suc  
join vendedor as ven  
on suc.idSucursal=ven.idSucursal  
join facturas as fac  
on ven.Cedula=fac.idVendedor  
join productos as prod  
on fac.idProducto=prod.idProducto  
join tipoproductos as tipoProd  
on prod.idTipoProducto=tipoProd.idTipoProducto  
where fac.fechaVenta between "2016-06-01" and "2016-11-30"  
group by 2  
order by 3 desc  
limit 1;
```



2-SELECCIONAR EL TIPO Y PRODUCTO MÁS VENDIDO DEL AÑO 2016 EN CUANTO A CANTIDADES SE REFIERE.

/*2. SELECCIONAR EL TIPO Y PRODUCTO MÁS VENDIDO DEL AÑO 2016 EN CUANTO A CANTIDADES SE REFIERE.*/*

/*Se piden solo los campos: [TIPO_PROD], [NOMBRE_PROD], [CANT], AÑO*/

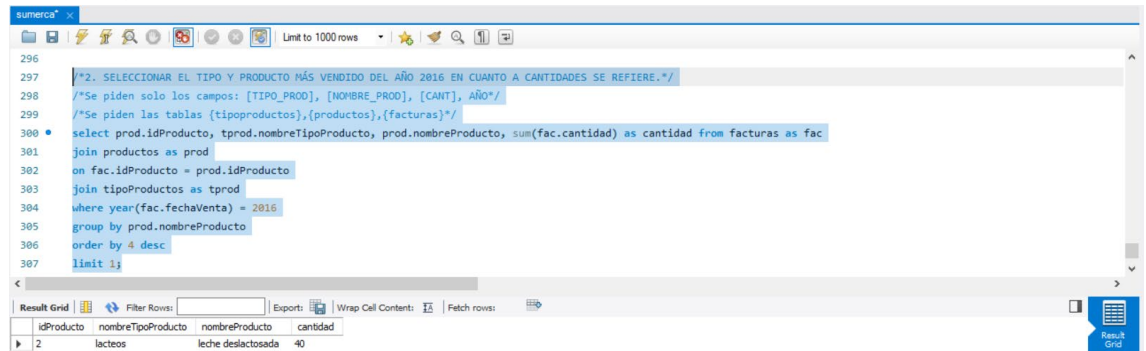
/*Se piden las tablas {tipoproductos},{productos},{facturas}*/

```
select prod.idProducto, tprod.nombreTipoProducto,  
prod.nombreProducto, sum(fac.cantidad) as cantidad from facturas  
as fac  
join productos as prod
```

```

on fac.idProducto = prod.idProducto
join tipoProductos as tprod
where year(fac.fechaVenta) = 2016
group by prod.nombreProducto
order by 4 desc
limit 1;

```



3-SELECCIONAR EL VENDEDOR QUE MÁS VENTAS HA REALIZADO EN UN MES, RETORNANDO EL NOMBRE DEL VENDEDOR, EL MES Y LA CANTIDAD VENDIDA.

```

/*3. SELECCIONAR EL VENDEDOR QUE MÁS VENTAS HA REALIZADO EN UN MES,
RETORNANDO EL NOMBRE DEL VENDEDOR, EL MES Y LA CANTIDAD VENDIDA.*/

```

```

/*Se piden solo los campos [NOMBREVENDEDOR],[MES],[CANTIDADVENDIDA]*/

```

```

/*Se piden solo las tablas {usuarios},{vendedor},{facturas}*/

```

```

select ven.Cedula, usar.nombre, month(fac.fechaVenta) as mes, fac.cantidad from
usuarios as usar

```

```

join vendedor as ven

```

```

on usar.idUsuarios=ven.Cedula

```

```

join facturas as fac

```

```

on ven.Cedula=idVendedor

```

```

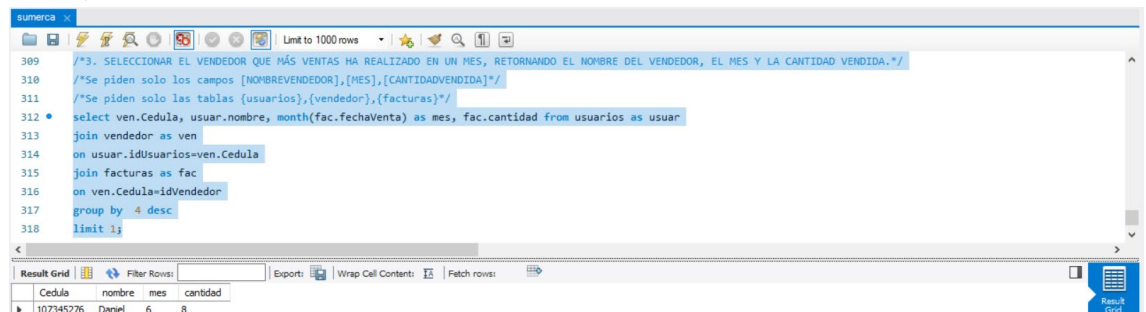
group by 4 desc

```

```

limit 1;

```



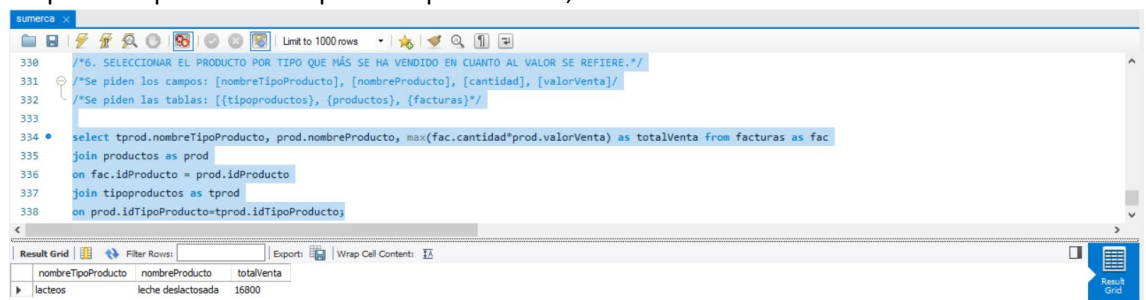
6-SELECCIONAR EL PRODUCTO POR TIPO QUE MÁS SE HA VENDIDO EN CUANTO AL VALOR SE REFIERE.

/*6. SELECCIONAR EL PRODUCTO POR TIPO QUE MÁS SE HA VENDIDO EN CUANTO AL VALOR SE REFIERE.*/

/*Se piden los campos: [nombreTipoProducto], [nombreProducto], [cantidad], [valorVenta]/

/*Se piden las tablas: [{tipoproductos}, {productos}, {facturas}*/

```
select tprod.nombreTipoProducto, prod.nombreProducto,
max(fac.cantidad*prod.valorVenta) as totalVenta from facturas as fac
join productos as prod
on fac.idProducto = prod.idProducto
join tipoproductos as tprod
on prod.idTipoProducto=tprod.idTipoProducto;
```



8-CREAR LA CONSULTA PARA IDENTIFICAR LOS PRODUCTOS MÁS VENDIDOS DURANTE EL ÚLTIMO AÑO, DONDE SE DEBE RETORNAR EL NOMBRE DEL PRODUCTO, LA CANTIDAD Y EL AÑO.

/*8. CREAR LA CONSULTA PARA IDENTIFICAR LOS PRODUCTOS MÁS VENDIDOS DURANTE EL ÚLTIMO AÑO, DONDE SE DEBE RETORNAR EL NOMBRE DEL PRODUCTO, LA CANTIDAD Y EL AÑO. */

/*Se piden los campos: [nombreProducto], [cant], [año]*/

/*Se piden las tablas: {vendedor}, {usuarios}, {sucursal}*/

```
select prod.nombreProducto, fac.cantidad, fac.fechaVenta from
facturas as fac
join productos as prod
on fac.idProducto=prod.idProducto
where year(fac.fechaVenta) = 2016
order by 2 desc;
```

The screenshot shows the Sumercá SQL editor with a query to find the top-selling products in 2016. The query is as follows:

```

350 /*8. CREAR LA CONSULTA PARA IDENTIFICAR LOS PRODUCTOS MÁS VENDIDOS DURANTE EL ÚLTIMO AÑO,
351 DONDE SE DEBE RETORNAR EL NOMBRE DEL PRODUCTO, LA CANTIDAD Y EL AÑO. */
352 /*Se piden los campos: [nombreProducto], [cant], [año]*/
353 /*Se piden las tablas: {vendedor}, {usuarios}, {sucursal}*/
354 select prod.nombreProducto, fac.cantidad, fac.fechaVenta from facturas as fac
355 join productos as prod
356 on fac.idProducto=prod.idProducto
357 where year(fac.fechaVenta) = 2016
358 order by 2 desc;
359
360

```

Below the query, the 'Result Grid' displays the following data:

nombreProducto	cantidad	fechaVenta
leche deslactosada	8	2016-06-20 00:00:00
gaseosa	6	2016-06-20 00:00:00
arveja	4	2016-06-20 00:00:00
lenteja	3	2016-11-22 00:00:00
jabon	2	2016-06-20 00:00:00
jabon en polvo	2	2016-06-20 00:00:00

10-CONSTRUIR LA CONSULTA QUE PERMITA CONOCER LAS VENTAS TOTALES DE CADA VENDEDOR EN CADA SUCURSAL, RETORNANDO EL NOMBRE DEL VENDEDOR, SUCURSAL Y VALOR TOTAL DE VENTAS POR CADA VENDEDOR ORDENADO DESCENDENTEMENTE POR VALOR.

/*10. CONSTRUIR LA CONSULTA QUE PERMITA CONOCER LAS VENTAS TOTALES DE CADA VENDEDOR EN CADA SUCURSAL, RETORNANDO EL NOMBRE DEL VENDEDOR, SUCURSAL Y VALOR TOTAL DE VENTAS POR CADA VENDEDOR

ORDENADO DESCENDENTEMENTE POR VALOR. */

/*Se piden los campos: [nombreSucursal], [nombreVendedor],

[[cantidad]*[valorVenta]]*/

/*Se piden las tablas: {sucursal}, , {usuarios}, {vendedor}, {facturas}, {productos} */

select suc.nombre as 'sucursal', ven.Cedula, usar.nombre as 'Vendedor',

sum(fac.idProducto*prod.idProducto) as 'ventaTotal' from sucursal as suc

join vendedor as ven

on suc.idSucursal = ven.idSucursal

join usuarios as usar

on ven.Cedula=usuar.idUsuarios

join facturas as fac

on usar.idUsuarios=fac.idVendedor

join productos as prod

on fac.idProducto=prod.idProducto

group by 3

order by 1;

The screenshot shows a SQL IDE window titled 'sumerca.v3'. The query is as follows:

```

441
442
443 /*10. CONSTRUIR LA CONSULTA QUE PERMITA CONOCER LAS VENTAS TOTALES DE CADA VENDEDOR EN CADA
444 SUCURSAL, RETORNANDO EL NOMBRE DEL VENDEDOR, SUCURSAL Y VALOR TOTAL DE VENTAS POR CADA VENDEDOR
445 ORDENADO DESCENDIENTEMENTE POR VALOR. */
446 /*Se piden los campos: [nombreSucursal], [nombreVendedor], [[cantidad]*[valorVenta]]*/
447 /*Se piden las tablas: {sucursal}, {usuarios}, {vendedor}, {facturas}, {productos} */
448
449 select suc.nombre as 'sucursal', ven.Cedula, usuar.nombre as 'Vendedor', sum(fac.idProducto*prod.idProducto) as 'ventaTotal' from sucursal as suc
450 join vendedor as ven
451 on suc.idSucursal = ven.idSucursal
452 join usuarios as usuar
453 on ven.Cedula=usuar.idusuarios
454 join facturas as fac
455 on usuar.idusuarios=fac.idvendedor
456 join productos as prod
457 on fac.idProducto=prod.idProducto
458 group by 3
459 order by 1;

```

The result grid shows the following data:

sucursal	Cedula	Vendedor	ventaTotal
caracas	107345276	Daniel	110
Tintal	1089345821	Kevin	74

12-CONSTRUIR UNA CONSULTA QUE LISTE EL ID DEL VENDEDOR, EL NOMBRE DEL VENDEDOR CON SUS APELLIDOS EN EL MISMO CAMPO, SU IDENTIFICACIÓN, SU SUCURSAL ORDENADOS POR ALFABÉTICAMENTE DE MENOR A MAYOR.

/*12. CONSTRUIR UNA CONSULTA QUE LISTE EL ID DEL VENDEDOR, EL NOMBRE DEL VENDEDOR CON SUS APELLIDOS EN EL MISMO CAMPO, SU IDENTIFICACIÓN, SU SUCURSAL ORDENADOS POR ALFABÉTICAMENTE DE MENOR A MAYOR. */

/*Se piden los campos: [idVend+nombreVen+apellidosVen], [idVend], [sucursalVen]*/

/*Se piden las tablas: {vendedor}, {usuarios}, {sucursal}*/

```

select concat_ws(' ', ven.Cedula, usuar.nombre, usuar.apellido) as datosVendedores,
ven.Cedula, suc.nombre as sucursal from sucursal as suc
join vendedor as ven
on suc.idSucursal=ven.idSucursal
join usuarios as usuar
on usuar.idusuarios=ven.Cedula
order by 3;

```

The screenshot shows a SQL IDE window titled 'sumerca'. The query is as follows:

```

362
363
364 /*12. CONSTRUIR UNA CONSULTA QUE LISTE EL ID DEL VENDEDOR, EL NOMBRE DEL VENDEDOR CON SUS APELLIDOS
365 EN EL MISMO CAMPO, SU IDENTIFICACIÓN, SU SUCURSAL ORDENADOS POR ALFABÉTICAMENTE DE MENOR A MAYOR. */
366 /*Se piden los campos: [idVend+nombreVen+apellidosVen], [idVend], [sucursalVen]*/
367 /*Se piden las tablas: {vendedor}, {usuarios}, {sucursal}*/
368
369 select concat_ws(' ', ven.Cedula, usuar.nombre, usuar.apellido) as datosVendedores, ven.Cedula, suc.nombre as sucursal from sucursal as suc
370 join vendedor as ven
371 on suc.idSucursal=ven.idSucursal
372 join usuarios as usuar
373 on usuar.idusuarios=ven.Cedula
374 order by 3;

```

The result grid shows the following data:

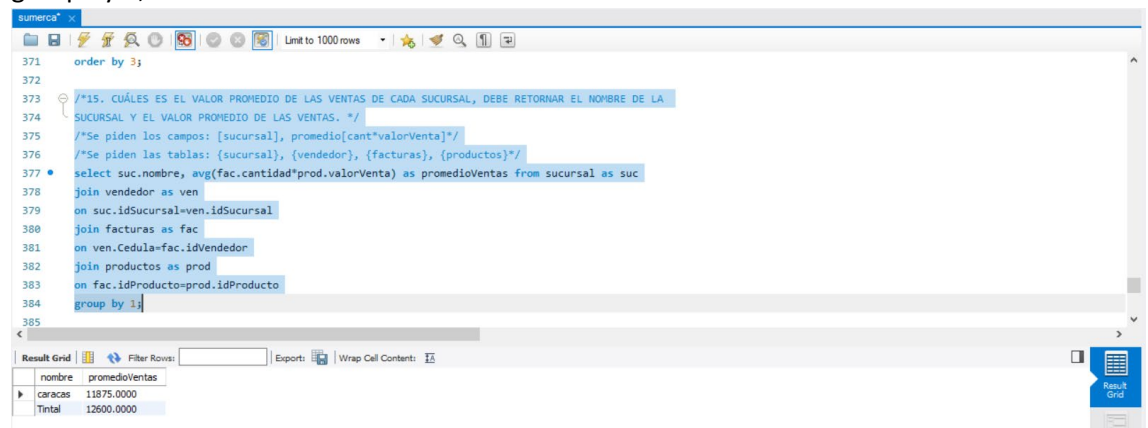
datosVendedores	Cedula	sucursal
107345276 Daniel Cifuentes	107345276	caracas
1089345821 Kevin Gomez	1089345821	Tintal

15-CUÁLES ES EL VALOR PROMEDIO DE LAS VENTAS DE CADA SUCURSAL, DEBE RETORNAR EL NOMBRE DE LA SUCURSAL Y EL VALOR PROMEDIO DE LAS VENTAS.


```

/*15. CUÁLES ES EL VALOR PROMEDIO DE LAS VENTAS DE CADA SUCURSAL, DEBE
RETORNAR EL NOMBRE DE LA
SUCURSAL Y EL VALOR PROMEDIO DE LAS VENTAS. */
/*Se piden los campos: [sucursal], promedio[cant*valorVenta]*/
/*Se piden las tablas: {sucursal}, {vendedor}, {facturas}, {productos}*/
select suc.nombre, avg(fac.cantidad*prod.valorVenta) as promedioVentas from
sucursal as suc
join vendedor as ven
on suc.idSucursal=ven.idSucursal
join facturas as fac
on ven.Cedula=fac.idVendedor
join productos as prod
on fac.idProducto=prod.idProducto
group by 1;

```



16-LISTAR LA CANTIDAD DE PRODUCTOS QUE HAY EN EL INVENTARIO, RETORNANDO EL ID DEL PRODUCTO, NOMBRE DE PRODUCTO Y CANTIDAD DE PRODUCTO EXISTENTES ASCENDENTEMENTE.

```

/*16. LISTAR LA CANTIDAD DE PRODUCTOS QUE HAY EN EL INVENTARIO,
RETORNANDO EL ID DEL PRODUCTO,
NOMBRE DE PRODUCTO Y CANTIDAD DE PRODUCTO EXISTENTES
ASCENDENTEMENTE. */
/*Se piden los campos: [producto], [cantidad]*/
/*Se piden las tablas: {inventarios}, {productos}*/
select prod.idProducto, prod.nombreProducto, sum(inv.cantidad)
as cantidad from inventarios as inv
right join productos as prod
on inv.idProducto=prod.idProducto
where inv.cantidad > 0
group by inv.idProducto
order by cantidad;

```

The screenshot shows a database query editor window titled 'sumerca.v3' with a tab for 'inventarios'. The SQL query is as follows:

```

383
384 /*16. LISTAR LA CANTIDAD DE PRODUCTOS QUE HAY EN EL INVENTARIO, RETORNANDO EL ID DEL PRODUCTO,
385 NOMBRE DE PRODUCTO Y CANTIDAD DE PRODUCTO EXISTENTES ASCENDENTE. */
386 /*Se piden los campos: [producto], [cantidad]*/
387 /*Se piden las tablas: {inventarios}, {productos}*/
388 • select prod.idProducto, prod.nombreProducto, sum(inv.cantidad) as cantidad from inventarios as inv
389 right join productos as prod
390 on inv.idProducto=prod.idProducto
391 where inv.cantidad > 0
392 group by inv.idProducto
393 order by cantidad;
394

```

The results are displayed in a table with the following data:

idProducto	nombreProducto	cantidad
9	jabon	3
6	soflan	5
8	jugo	5
7	gaseosa	8
1	leche entera	8
4	arveja	10
2	leche deslactosada	12
5	jabon en polvo	12
3	lenteja	13

EJERCICIOS QUE REQUIRIERON MODIFICACION E INSERCIÓN DE DATOS ADICIONALES (5)

5-SELECCIONAR LOS PRODUCTOS QUE SE HAN VENDIDO MÁS DE 10 VECES Y ORDENAR EL RESULTADO DE FORMA DESCENDENTE.

/*5. SELECCIONAR LOS PRODUCTOS QUE SE HAN VENDIDO MÁS DE 10 VECES Y ORDENAR EL RESULTADO DE FORMA DESCENDENTE. */
/*Se piden los campos: [NOMBREPRODUCTO],[CANT]*/
/*Se piden las tablas: {productos},{facturas}*/

```

select prod.nombreProducto, fac.cantidad from facturas as fac
join productos as prod
on fac.idProducto = prod.idProducto
where fac.cantidad > 10
order by 2 desc;

```

The screenshot shows a database query editor window titled 'sumerca.v3' with a tab for 'facturas'. The SQL query is as follows:

```

319
320 /*5. SELECCIONAR LOS PRODUCTOS QUE SE HAN VENDIDO MÁS DE 10 VECES Y ORDENAR EL RESULTADO DE FORMA DESCENDENTE. */
321 /*Se piden los campos: [NOMBREPRODUCTO],[CANT]*/
322 /*Se piden las tablas: {productos},{facturas}*/
323
324 • select prod.nombreProducto, fac.cantidad from facturas as fac
325 join productos as prod
326 on fac.idProducto = prod.idProducto
327 where fac.cantidad > 10
328 order by 2 desc;

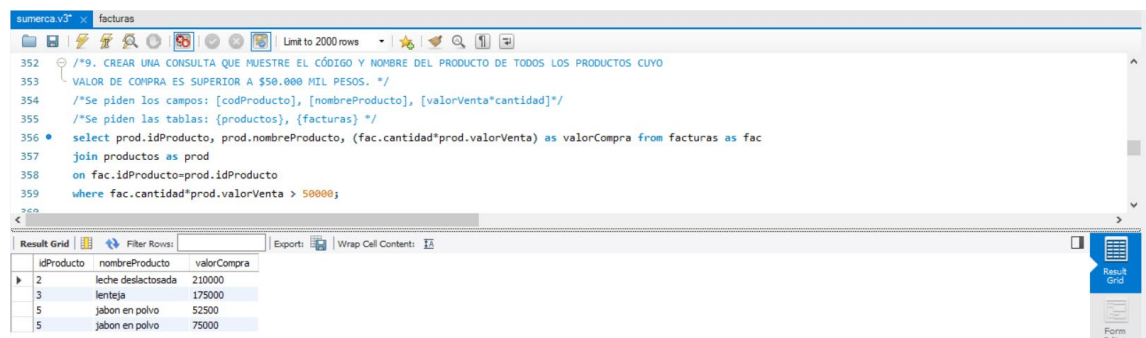
```

The results are displayed in a table with the following data:

nombreProducto	cantidad
leche deslactosada	100
lenteja	50

9-CREAR UNA CONSULTA QUE MUESTRE EL CÓDIGO Y NOMBRE DEL PRODUCTO DE TODOS LOS PRODUCTOS CUYO VALOR DE COMPRA ES SUPERIOR A \$50.000 MIL PESOS.

```
/*9. CREAR UNA CONSULTA QUE MUESTRE EL CÓDIGO Y NOMBRE DEL
PRODUCTO DE TODOS LOS PRODUCTOS CUYO
VALOR DE COMPRA ES SUPERIOR A $50.000 MIL PESOS. */
/*Se piden los campos: [codProducto], [nombreProducto],
[valorVenta*cantidad]*/
/*Se piden las tablas: {productos}, {facturas} */
select prod.idProducto, prod.nombreProducto,
(fac.cantidad*prod.valorVenta) as valorCompra from facturas as
fac
join productos as prod
on fac.idProducto=prod.idProducto
where fac.cantidad*prod.valorVenta > 50000;
```



The screenshot shows a database query editor window titled 'facturas'. The SQL query is as follows:

```
/*9. CREAR UNA CONSULTA QUE MUESTRE EL CÓDIGO Y NOMBRE DEL PRODUCTO DE TODOS LOS PRODUCTOS CUYO
VALOR DE COMPRA ES SUPERIOR A $50.000 MIL PESOS. */
/*Se piden los campos: [codProducto], [nombreProducto], [valorVenta*cantidad]*/
/*Se piden las tablas: {productos}, {facturas} */
select prod.idProducto, prod.nombreProducto, (fac.cantidad*prod.valorVenta) as valorCompra from facturas as fac
join productos as prod
on fac.idProducto=prod.idProducto
where fac.cantidad*prod.valorVenta > 50000;
```

Below the query editor, a 'Result Grid' is displayed with the following data:

	idProducto	nombreProducto	valorCompra
2		leche deslactosada	210000
3		lenteja	175000
5		jabon en polvo	52500
5		jabon en polvo	75000

13-CUÁLES SON LOS PRODUCTOS VENDIDOS DE CADA VENDEDOR, TRAYENDO EL CÓDIGO DEL VENDEDOR, NOMBRE DEL VENDEDOR, LA CANTIDAD TOTAL DE PRODUCTOS Y EL VALOR TOTAL DE LAS VENTAS POR DICHS PRODUCTOS.

```
/*13-CUÁLES SON LOS PRODUCTOS VENDIDOS DE CADA VENDEDOR,
TRAYENDO EL CÓDIGO DEL VENDEDOR,
NOMBRE DEL VENDEDOR, LA CANTIDAD TOTAL DE PRODUCTOS Y EL VALOR
TOTAL DE LAS VENTAS POR
DICHS PRODUCTOS. */
```

```
select ven.Cedula, usar.nombre as vendedor, max(fac.cantidad),
prod.nombreProducto, max(fac.cantidad*prod.valorVenta)
from usuarios as usar
join vendedor as ven
on usar.idUsuarios=ven.cedula
join facturas as fac
on ven.cedula=fac.idVendedor
join productos as prod
on fac.idProducto=prod.idProducto
group by 2;
```

sumerca.v3 facturas

Limit to 2000 rows

```

432
433 /*13-CUÁLES SON LOS PRODUCTOS VENDIDOS DE CADA VENDEDOR, TRAYENDO EL CÓDIGO DEL VENDEDOR,
434 NOMBRE DEL VENDEDOR, LA CANTIDAD TOTAL DE PRODUCTOS Y EL VALOR TOTAL DE LAS VENTAS POR
435 DICHOS PRODUCTOS. */
436
437 select ven.Cedula, usar.nombre as vendedor, max(fac.cantidad), prod.nombreProducto, max(fac.cantidad*prod.valorVenta)
438 from usuarios as usar
439 join vendedor as ven
440 on usar.idUsuarios=ven.cedula
441 join facturas as fac
442 on ven.cedula=fac.idVendedor
443 join productos as prod
444 on fac.idProducto=prod.idProducto
445 group by 2;
446

```

Result Grid

Cedula	vendedor	max(fac.cantidad)	nombreProducto	max(fac.cantidad*prod.valorVenta)
10010010	anderson	7	leche en polvo	52500
11111111	casimiro	4	leche entera	6800
107345276	Daniel	8	arveja	16800
53160832	eliza	10	arveja	75000
9069966	julio	50	lenteja	175000
1089345821	Kevin	6	gaseosa	15000
10101010	milller	4	leche entera	6800
64552398	yudy	100	leche deslactosada	210000

14-CONSTRUIR UNA CONSULTA QUE PERMITA CONOCER EL SALARIO TOTAL A PAGAR A CADA VENDEDOR TENIENDO EN CUENTA QUE POR VENTAS SUPERIORES A: a-\$50.000 mil pesos obtendrá una comisión del 2% del total de la venta, b-\$100.000 mil pesos obtendrá una comisión del 5% del total de la venta, c-\$150.000 millón de pesos obtendrá una comisión del 10% del total de la venta Y d-Para todas las ventas menores a \$50.000 la comisión es del 1% del total de la venta Nota: El salario básico de todos los vendedores es de \$1.200.000 mil pesos:

/*14-CONSTRUIR UNA CONSULTA QUE PERMITA CONOCER EL SALARIO TOTAL A PAGAR A CADA VENDEDOR TENIENDO EN CUENTA QUE POR VENTAS SUPERIORES A:
a-\$50.000 mil pesos obtendrá una comisión del 2% del total de la venta,
b-\$100.000 mil pesos obtendrá una comisión del 5% del total de la venta,
c-\$150.000 mil de pesos obtendrá una comisión del 10% del total de la venta Y
d-Para todas las ventas menores a \$50.000 la comisión es del 1% del total de la venta
Nota: El salario básico de todos los vendedores es de \$1.200.000 mil pesos:*/

INSERT INTO usuarios VALUES

(53160832,'eliza','rodriguez','1977-08-21','Calle 21 #17-41','3102591602'),
(64552398,'yudy','pupo','1977-08-22','Calle 22 #18-42','3102591601'),
(9069966,'julio','jimeno','1977-08-23','Calle 23 #19-43','3102591603'),
(11111111,'casimiro','Hernandez','1977-08-24','Calle 24 #20-44','3102591604'),
(10101010,'milller','ochoa','1977-08-25','Calle 25 #21-45','3102591605'),
(10010010,'anderson','ardila','1977-08-26','Calle 26 #22-46','3102591606');

INSERT INTO vendedor VALUES

```
(53160832,3,1200000),  
(64552398,3,1200000),  
(9069966,4,1200000),  
(11111111,5,1200000),  
(10101010,5,1200000),  
(10010010,5,1200000);
```

```
SET SQL_SAFE_UPDATES=0;  
update vendedor set salario = 1200000  
where salario = 796000;
```

```
INSERT INTO facturas VALUES  
(10024,4,53160832,17657700,4,'2016-07-20 00:00:00'),  
(10025,5,53160832,17657700,10,'2016-07-20 00:00:00'),  
(10018,4,53160832,17657700,4,'2016-08-20 00:00:00'),  
(10019,2,64552398,17657700,100,'2016-06-20 00:00:00'),  
(10020,3,9069966,17657700,50,'2016-07-20 00:00:00'),  
(10021,1,11111111,17657700,4,'2016-06-20 00:00:00'),  
(10022,1,10101010,17657700,4,'2016-07-20 00:00:00'),  
(10023,5,10010010,17657700,7,'2016-06-20 00:00:00');
```

```
select ven.Cedula, concat(usua.nombre, ' ', usua.apellido) as  
'vendedor', month(fac.fechaVenta) as 'mes de la comision'  
, ven.salario, sum(fac.cantidad*prod.valorVenta) as 'ventas',  
if(sum(fac.cantidad*prod.valorVenta)<50000,(sum(fac.cantidad*prod.valorVenta)*0.0  
1)+ven.salario,  
if(sum(fac.cantidad*prod.valorVenta)>=50000 and  
sum(fac.cantidad*prod.valorVenta)<100000,(sum(fac.cantidad*prod.valorVenta)*0.02  
) +ven.salario,  
if(sum(fac.cantidad*prod.valorVenta)>=100000 and  
sum(fac.cantidad*prod.valorVenta)<150000,(sum(fac.cantidad*prod.valorVenta)*0.05  
) , (sum(fac.cantidad*prod.valorVenta)*0.1)))) as 'SalarioTotal a Pagar'  
from vendedor as ven  
join usuarios as usua  
on ven.Cedula=usua.idUsuarios  
join facturas as fac  
on usua.idUsuarios=fac.idVendedor  
join productos as prod  
on fac.idProducto=prod.idProducto  
group by 2,3;
```

sumerca v37 facturas

```

442 • INSERT INTO usuarios VALUES
443 (53160832,'eliza','rodriguez','1977-08-21','Calle 21 #17-41','3182591682'),
444 (64552398,'yudy','pupo','1977-08-22','Calle 22 #18-42','3182591681'),
445 (9069966,'julio','jimeno','1977-08-23','Calle 23 #19-43','3182591683'),
446 (11111111,'casimiro','Hernandez','1977-08-24','Calle 24 #20-44','3182591684'),
447 (10101010,'miller','ochoa','1977-08-25','Calle 25 #21-45','3182591685'),
448 (10010010,'anderson','ardila','1977-08-26','Calle 26 #22-46','3182591686');
449
450 • INSERT INTO vendedor VALUES
451 (53160832,3,1200000),
452 (64552398,3,1200000),
453 (9069966,4,1200000),
454 (11111111,5,1200000),
455 (10101010,5,1200000),
456 (10010010,5,1200000);
457
458 • SET SQL_SAFE_UPDATES=0;
459 • update vendedor set salario = 1200000
460 where salario = 796000;
461
462 • INSERT INTO facturas VALUES
463 (10024,4,53160832,17657700,4,'2016-07-20 00:00:00'),
464 (10025,5,53160832,17657700,10,'2016-07-20 00:00:00'),
465 (10019,4,53160832,17657700,4,'2016-08-20 00:00:00'),
466 (10019,2,64552398,17657700,100,'2016-06-20 00:00:00'),
467 (10020,3,9069966,17657700,50,'2016-07-20 00:00:00'),
468 (10021,1,11111111,17657700,4,'2016-06-20 00:00:00'),
469 (10022,1,10101010,17657700,4,'2016-07-20 00:00:00'),
470 (10023,5,10010010,17657700,7,'2016-06-20 00:00:00');
471
472 • select ven.Cedula, concat(usua.nombre, ' ',usua.apellido) as 'vendedor',month(fac.fechaVenta) as 'mes de la comision'
473 ,ven.salario, sum(fac.cantidad*prod.valorVenta) as 'ventas',
474 if(sum(fac.cantidad*prod.valorVenta)<50000,(sum(fac.cantidad*prod.valorVenta)*0.01)+ven.salario,
475 if(sum(fac.cantidad*prod.valorVenta)>=50000 and sum(fac.cantidad*prod.valorVenta)<100000,(sum(fac.cantidad*prod.valorVenta)*0.02)+ven.salario,
476 if(sum(fac.cantidad*prod.valorVenta)>=100000 and sum(fac.cantidad*prod.valorVenta)<150000,(sum(fac.cantidad*prod.valorVenta)*0.05),(sum(fac.cantidad*prod.valorVenta)*0.1)))) as 'SalarioTotal a Pagar'
477 from vendedor as ven
478 join usuarios as usua
479 on ven.Cedula=usua.idusuarios
480 join facturas as fac
481 on usua.idusuarios=fac.idvendedor
482 join productos as prod
483 on fac.idProducto=prod.idProducto
484 group by 2,3;

```

sumerca v37 facturas

```

471 • select ven.Cedula, concat(usua.nombre, ' ',usua.apellido) as 'vendedor',month(fac.fechaVenta) as 'mes de la comision'
472 ,ven.salario, sum(fac.cantidad*prod.valorVenta) as 'ventas',
473 if(sum(fac.cantidad*prod.valorVenta)<50000,(sum(fac.cantidad*prod.valorVenta)*0.01)+ven.salario,
474 if(sum(fac.cantidad*prod.valorVenta)>=50000 and sum(fac.cantidad*prod.valorVenta)<100000,(sum(fac.cantidad*prod.valorVenta)*0.02)+ven.salario,
475 if(sum(fac.cantidad*prod.valorVenta)>=100000 and sum(fac.cantidad*prod.valorVenta)<150000,(sum(fac.cantidad*prod.valorVenta)*0.05),(sum(fac.cantidad*prod.valorVenta)*0.1)))) as 'SalarioTotal a Pagar'
476 from vendedor as ven
477 join usuarios as usua
478 on ven.Cedula=usua.idusuarios

```

Cedula	vendedor	mes de la comision	salario	ventas	SalarioTotal a Pagar
10010010	anderson ardila	6	1200000	52500	1201050.00
11111111	casimiro Hernandez	6	1200000	6800	1200068.00
107345276	Daniel Cifuentes	6	1200000	37000	1200370.00
107345276	Daniel Cifuentes	11	1200000	10500	1200105.00
53160832	eliza rodriguez	7	1200000	90600	1201812.00
53160832	eliza rodriguez	8	1200000	15600	1200156.00
9069966	julio jimeno	7	1200000	175000	17500.00
1089345821	Kevin Gomez	6	1200000	25200	1200252.00
10101010	miller ochoa	7	1200000	6800	1200068.00
64552398	yudy pupo	6	1200000	210000	210000.00

17-Obtener el listado de clientes que no hayan realizado compras durante el último año pero sus compras totales son iguales o superiores a 1.000.000, debe retornar el id del cliente, nombre-apellido, la fecha y el valor de la última compra realizada.

/*17. Obtener el listado de clientes que no hayan realizado compras durante el último año pero sus compras totales son iguales o superiores a 1.000.000, debe retornar el id del cliente, nombre-apellido, la fecha y el valor de la última compra realizada.*/

```

insert into usuarios values (457865456,'Julio','Lara','2022-07-23','Av
circunvalar','3108532882');
insert into clientes values (457865456,1234,'pagina.com');
/*FORMA 1*/
select clientes.cedula, concat(usuarios.nombre, usuarios.apellido) as 'nombreCliente',
facturas.fechaVenta, sum(facturas.cantidad*productos.valorVenta) as 'valorVenta'
from clientes
left join usuarios
on clientes.cedula=usuarios.idUsuarios
left join facturas
on facturas.idCliente = clientes.cedula
left join productos
on productos.idProducto=facturas.idProducto
where facturas.fechaVenta is null
group by clientes.cedula;

```

