



[<- Workshop 4](#)

Workshop 5

0. Instalar MySQL Workbench en Debian (anfitriona)

```
sudo apt-get install mysql-workbench-community
```

1. Crear vagrantfile en la carpeta database

```
vagrant init debian/bullseye64
```

2. Cambiar la IP de private_network a 192.168.56.11

3. Configurar maquina virtual y la base de datos

```
vagrant up
vagrant ssh
sudo apt-get update
sudo apt-get install mariadb-server mariadb-client vim vim-nox curl git

sudo mysql
create database northwind;
create user user_laravel identified by 'secret';
grant all privileges on northwind.* to user_laravel;
flush privileges;
use mysql;
select user, host from user;

sudo vim /etc/mysql/mariadb.conf.d/50-server.cnf      (COMENTAR BIND-
ADDRESS)
sudo systemctl restart mysql
```

4. Ir a la carpeta donde estan los backups y resturar backups de la base de datos

```
mysql northwind < northwind.sql -u user_laravel -p
mysql northwind < northwind-data.sql -u user_laravel -p
```

5. Challenge #1: Recupere el código (id) y la descripción (type_name) de los tipos de movimiento de inventario (inventory_transaction_types).

```
desc inventory_transaction_types;
select id `codigo`, type_name `description` from
inventory_transaction_types;
```

	codigo	description
1	1	Purchased
2	2	Sold
3	3	On Hold
4	4	Waste

6. Challenge #2: Recupere la cantidad de ordenes (orders) registradas por cada vendedor (employees).

```
desc orders;
desc employees;
select concat(e.first_name, ' ', e.last_name) as `Vendedor`,
       count(1) as `Cantidad`
  from orders o
 join employees e
    on e.id = o.employee_id
 group by `Vendedor`
 order by e.id;
```

	Vendedor	Cantidad
1	Nancy Freehafer	12
2	Andrew Cencini	4
3	Jan Kotas	6
4	Mariya Sergienko	8
5	Michael Neipper	4
6	Robert Zare	2
7	Laura Giussani	2
8	Anne Hellung-Larsen	10

7. Challenge #3: Recupere la lista de los 10 productos más ordenados (order_details), y la cantidad total de unidades ordenadas para cada uno de los productos. Deberá incluir como mínimo los campos de código (product_code), nombre del producto (product_name) y la cantidad de unidades.

```
desc order_details;
desc products;
select p.product_code as `Codigo`,
       p.product_name as `Producto`,
       round(sum(od.quantity),2) as `Cantidad`
from products p
join order_details od
  on od.product_id = p.id
group by p.id
order by `Cantidad` DESC
limit 10;
```

	Codigo	Producto	Cantidad
1	NWTB-43	Northwind Traders Coffee	650.00
2	NWTB-34	Northwind Traders Beer	487.00
3	NWTSO-41	Northwind Traders Clam Chowder	290.00
4	NWTB-81	Northwind Traders Green Tea	275.00
5	NWTCA-48	Northwind Traders Chocolate	200.00
6	NWTCM-40	Northwind Traders Crab Meat	120.00
7	NWTJP-6	Northwind Traders Boysenberry Spread	100.00
8	NWTP-57	Northwind Traders Ravioli	100.00
9	NWTD-72	Northwind Traders Mozzarella	90.00
10	NWTBGM-19	Northwind Traders Chocolate Biscuits Mix	85.00

8. Challenge #4: Recupere el monto total (invoices, orders, order_details, products) y la cantidad de facturas (invoices) por vendedor (employee). Debe considerar solamente las ordenes con estado diferente de 0 y solamente los detalles en estado 2 y 3, debe utilizar el precio unitario de las lineas de detalle de orden, no considere el descuento, no considere los impuestos, porque la comisi3n a los vendedores se paga sobre el precio base.

```
desc employees;
desc invoices;
desc orders;
desc order_details;
select count(1) as `Cantidad`,
       concat(e.first_name, ' ', e.last_name) as `Vendedor`,
       round(sum(od.unit_price * od.quantity),2) as `Monto`
from employees e
join orders o
  on o.employee_id = e.id
join order_details od
  on od.order_id = o.id
where o.status_id <> 0
     and od.status_id in (2,3)
group by e.id
order by `Cantidad` DESC, e.first_name ASC;
```

	Cantidad	Vendedor	Monto
1	10	Anne Hellung-Larsen	19974.25
2	10	Mariya Sergienko	6278.00
3	8	Nancy Freehafer	4972.50
4	7	Jan Kotas	5787.50
5	4	Michael Neipper	6378.00
6	3	Andrew Cencini	2617.50
7	3	Robert Zare	3786.50
8	1	Laura Giussani	680.00

9. Challenge #5: Recupere los movimientos de inventario del tipo ingreso. Tomando como base todos los movimientos de inventario (inventory_transactions), considere unicamente el tipo de movimiento 1 (transaction_type) como ingreso. Debe agrupar por producto (inventory_transactions.product_id) y deberá incluir como mínimo los campos de código (product_code), nombre del producto (product_name) y la cantidad de unidades ingresadas.

```
select p.product_code as `Codigo`,
       p.product_name as `Producto`,
       sum(it.quantity) as `Cantidad`
from products p
join inventory_transactions it
  on it.product_id = p.id
where it.transaction_type in (1)
group by p.id;
```

	Codigo	Producto	Cantidad
1	NWTB-1	Northwind Traders Chai	40
2	NWTCO-3	Northwind Traders Syrup	100
3	NWTCO-4	Northwind Traders Cajun Seasoning	40
4	NWTO-5	Northwind Traders Olive Oil	40
5	NWTJP-6	Northwind Traders Boysenberry Spread	100
6	NWTDFN-7	Northwind Traders Dried Pears	40
7	NWTS-8	Northwind Traders Curry Sauce	65
8	NWTDFN-14	Northwind Traders Walnuts	40
9	NWTCFV-17	Northwind Traders Fruit Cocktail	40
10	NWTBGM-19	Northwind Traders Chocolate Biscuits Mix	85

10. Challenge #6: Recupere los movimientos de inventario del tipo salida. Tomando como base todos los movimientos de inventario (inventory_transactions), considere unicamente los tipos de movimiento (transaction_type) 2, 3 y 4 como salidas. Debe agrupar por producto (products) y deberá incluir como mínimo los campos de código (product_code), nombre del producto (product_name) y la cantidad de unidades que salieron.

```
select p.product_code as `Codigo`,
       p.product_name as `Producto`,
       sum(it.quantity) as `Cantidad`
from products p
join inventory_transactions it
  on it.product_id = p.id
where it.transaction_type in (2,3,4)
group by p.id;
```

	Codigo	Producto	Cantidad
1	NWTB-1	Northwind Traders Chai	40
2	NWTCO-3	Northwind Traders Syrup	50
3	NWTCO-4	Northwind Traders Cajun Seasoning	40
4	NWTO-5	Northwind Traders Olive Oil	25
5	NWTJP-6	Northwind Traders Boysenberry Spread	100
6	NWTDFN-7	Northwind Traders Dried Pears	40
7	NWTS-8	Northwind Traders Curry Sauce	65
8	NWTCFV-17	Northwind Traders Fruit Cocktail	40
9	NWTBGM-19	Northwind Traders Chocolate Biscuits Mix	85
10	NWTJP-6	Northwind Traders Marmalade	40

11. Challenge #7: Genere un reporte de movimientos de inventario (inventory_transactions) por producto (products), tipo de transacción y fecha, entre las fechas 22/03/2006 y 24/03/2006 (incluyendo ambas fechas). Debe incluir como mínimo el código (product_code), el nombre del producto (product_name), la fecha truncada (transaction_created_date), la descripción del tipo de movimiento (type name) y la suma de cantidad (quantity) .

```
select p.product_code as `Codigo`,
       p.product_name as `Producto`,
       itt.type_name as `Tipo`,
       date_format(it.transaction_created_date, "%d/%m/%Y") as `Fecha`,
       sum(it.quantity) as `Cantidad`
from inventory_transactions it
join products p
  on it.product_id = p.id
join inventory_transaction_types itt
  on it.transaction_type = itt.id
where date(it.transaction_created_date) between '2006-03-22' and
'2006-03-24'
group by p.id, it.transaction_type
order by p.product_name asc;
```

	Codigo	Producto	Tipo	Fecha	Cantidad
1	NWTFN-74	Northwind Traders Almonds	Purchased	22/03/2006	20
2	NWTB-34	Northwind Traders Beer	Sold	24/03/2006	100
3	NWTB-34	Northwind Traders Beer	Purchased	22/03/2006	160
4	NWTJP-6	Northwind Traders Boysenberry Spread	Sold	24/03/2006	10
5	NWTJP-6	Northwind Traders Boysenberry Spread	Purchased	22/03/2006	100
6	NWTCO-4	Northwind Traders Cajun Seasoning	Purchased	22/03/2006	40
7	NWTCO-4	Northwind Traders Cajun Seasoning	Sold	24/03/2006	10
8	NWTB-1	Northwind Traders Chai	Sold	22/03/2006	15
9	NWTB-1	Northwind Traders Chai	On Hold	24/03/2006	25
10	NWTB-1	Northwind Traders Chai	Purchased	22/03/2006	40

12. Challenge #8: Genere la consulta SQL para un reporte de inventario, tomando como base todos los movimientos de inventario (inventory_transactions), considere los tipos de movimiento (transaction_type) 2, 3 y 4 como salidas y el tipo 1 como ingreso. Este reporte debe estar agrupado por producto (products) y deberá incluir como mínimo los campos de código (product_code), nombre del producto (product_name) y la sumariación de ingresos, salidas y la cantidad disponible en inventario (diferencia de ingresos - salidas).

```
select p.product_code as `Codigo`,
       p.product_name as `Producto`,
       sum(if(it.transaction_type = 1, it.quantity, 0)) as `Ingresos`,
       sum(if(it.transaction_type in (2,3,4), it.quantity, 0)) as `Salidas`,
       (sum(if(it.transaction_type = 1, it.quantity, 0)) -
        sum(if(it.transaction_type in (2,3,4), it.quantity, 0))) as `Disponible`
from products p
join inventory_transactions it
  on it.product_id = p.id
group by p.id;
```


	Codigo	Producto	Ingresos	Salidas	Disponible
1	NWTB-1	Northwind Traders Chai	40	40	0
2	NWTCO-3	Northwind Traders Syrup	100	50	50
3	NWTCO-4	Northwind Traders Cajun Seasoning	40	40	0
4	NWTO-5	Northwind Traders Olive Oil	40	25	15
5	NWTJP-6	Northwind Traders Boysenberry Spread	100	100	0
6	NWTDFN-7	Northwind Traders Dried Pears	40	40	0
7	NWTS-8	Northwind Traders Curry Sauce	65	65	0
8	NWTDFN-14	Northwind Traders Walnuts	40	0	40
9	NWTCFV-17	Northwind Traders Fruit Cocktail	40	40	0
10	NWTBGM-19	Northwind Traders Chocolate Biscuits Mix	85	85	0

13. Challenge #9: El mejor cliente


```
select concat(c.first_name, ' ', c.last_name) as `Cliente`,
       round(sum(od.quantity * od.unit_price), 2) as `Monto`
from customers c
join orders o
  on o.customer_id = c.id
join order_details od
  on od.order_id = o.id
join invoices i
  on i.order_id = o.id
group by c.id
order by `Monto` desc
limit 5;
```



Challenge #9

14. Challenge #10: El mejor mes

```
select date_format(i.invoice_date, "%Y") as `Ano`,
       date_format(i.invoice_date, "%m") as `Mes`,
       round(sum(od.quantity * od.unit_price), 2) as `Monto`
from invoices i
join orders o
  on o.id = i.order_id
join order_details od
  on od.order_id = o.id
group by `Ano`, `Mes`
order by `Monto` desc;
```

Challenge #10