**Imagen que contiene Texto

Descripción generada automáticamente10.TALLER – Lenguaje de Consulta Relacional - DDL**

Apellido y Nombre del estudiante: Andres camilo barco roa

***Nota: cargar el archivo en formato word, recordar la estructura para guardar el archivo mencionada en clase.***

El estudiante deberá realizar en el siguiente taller con las capturas de pantallas del query con los resultados de las consultas, aplicando los conceptos vistos en clase:

Una empresa de gestión de eventos llamada **"EventMaster"** necesita una base de datos para almacenar información sobre eventos, clientes y reservas.

**Objetivo**

1. Crear la base de datos.

2. Crear y registrar en las tablas Eventos (id\_evento, nombre, fecha, capacidad, precio); Clientes (id\_cliente, nombre, correo, teléfono); y Reservas (id\_reserva, id\_evento, id\_cliente, cantidad\_boletos , total\_pago); con 10 registros en cada una.

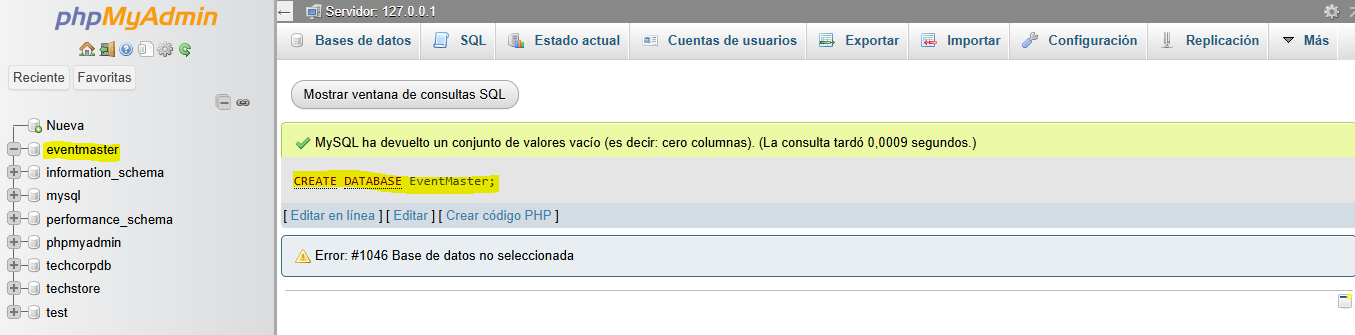
3. Realizar consultas SQL con respuestas esperadas.

1. Mostrar todos los eventos
2. Obtener los nombres y precios de los eventos más costosos (mayores a $1.000.000)
3. Contar la cantidad total de clientes registrados.
4. Mostrar las reservas realizadas por el cliente "Carlos Pérez"
5. Obtener la fecha y el nombre del evento con menor capacidad.
6. Calcular el ingreso total de todas las reservas.
7. Obtener el evento más reservado (mayor cantidad de boletos vendidos)
8. Mostrar clientes cuyo nombre inicie con "C"
9. Eliminar todas las reservas para el evento "Cena de Gala"

**SOLUCIÓN**

1. Crear la base de datos.

**CREATE DATABASE EventMaster;**



2. Crear y registrar en las tablas Eventos (id\_evento, nombre, fecha, capacidad, precio); Clientes (id\_cliente, nombre, correo, teléfono); y Reservas (id\_reserva, id\_evento, id\_cliente, cantidad\_boletos , total\_pago); con 10 registros en cada una.

**CREATE TABLE Eventos (**

**id\_evento INT AUTO\_INCREMENT PRIMARY KEY,**

**nombre VARCHAR(100) NOT NULL,**

**fecha DATE NOT NULL,**

**capacidad INT NOT NULL,**

**precio DECIMAL(10, 2) NOT NULL**

**);**

**INSERT INTO Eventos (nombre, fecha, capacidad, precio)**

**VALUES**

**('Concierto Rock', '2025-05-20', 500, 150000.00),**

**('Feria Tecnológica', '2025-06-10', 300, 80000.00),**

**('Obra de Teatro', '2025-04-25', 200, 100000.00),**

**('Festival de Jazz', '2025-07-01', 400, 120000.00),**

**('Torneo de eSports', '2025-05-05', 250, 90000.00),**

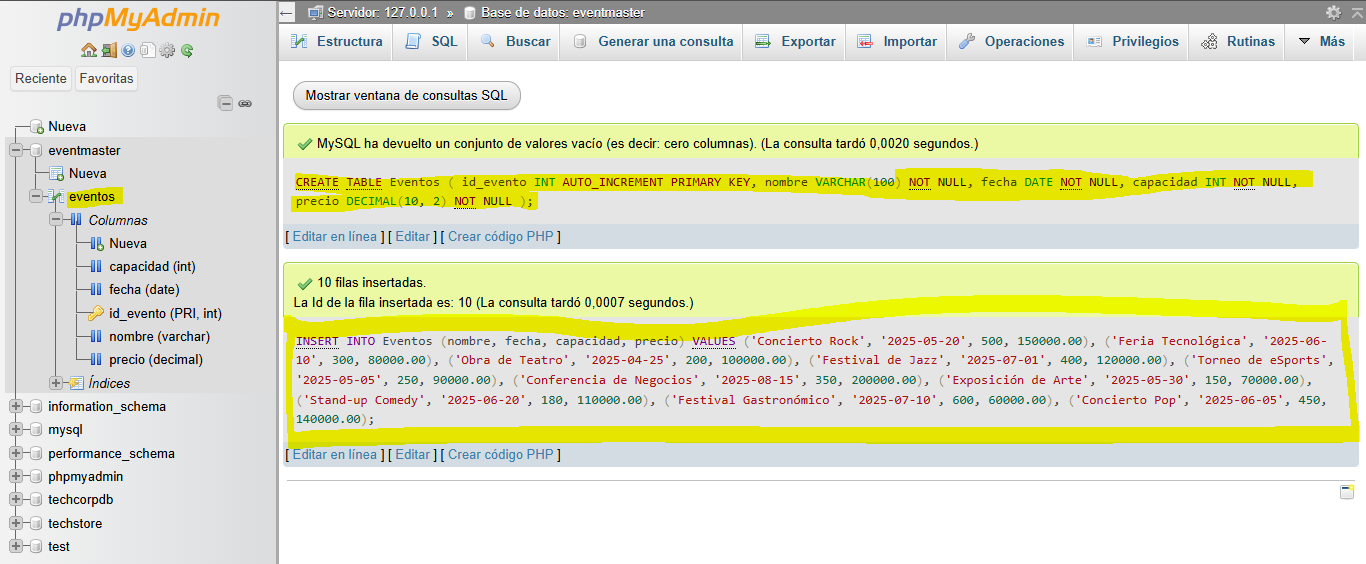
**('Conferencia de Negocios', '2025-08-15', 350, 200000.00),**

**('Exposición de Arte', '2025-05-30', 150, 70000.00),**

**('Stand-up Comedy', '2025-06-20', 180, 110000.00),**

**('Festival Gastronómico', '2025-07-10', 600, 60000.00),**

**('Concierto Pop', '2025-06-05', 450, 140000.00);**



**CREATE TABLE Clientes (**

**id\_cliente INT AUTO\_INCREMENT PRIMARY KEY,**

**nombre VARCHAR(100) NOT NULL,**

**correo VARCHAR(100) NOT NULL,**

**telefono VARCHAR(20) NOT NULL**

**);**

**INSERT INTO Clientes (nombre, correo, telefono) VALUES**

**('Ana López', 'ana@gmail.com', '3214567890'),**

**('Carlos Pérez', 'carlos@gmail.com', '3001234567'),**

**('María García', 'maria@gmail.com', '3127894561'),**

**('Luis Rodríguez', 'luis@gmail.com', '3156547891'),**

**('Sofía Ramírez', 'sofia@gmail.com', '3182223344'),**

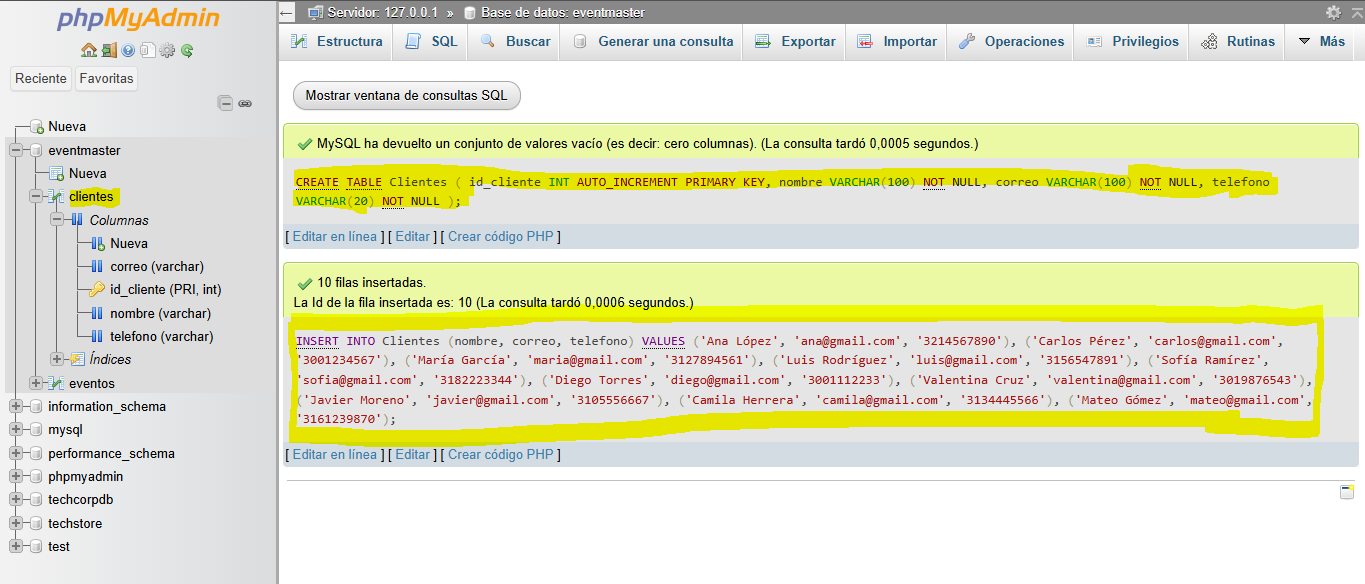
**('Diego Torres', 'diego@gmail.com', '3001112233'),**

**('Valentina Cruz', 'valentina@gmail.com', '3019876543'),**

**('Javier Moreno', 'javier@gmail.com', '3105556667'),**

**('Camila Herrera', 'camila@gmail.com', '3134445566'),**

**('Mateo Gómez', 'mateo@gmail.com', '3161239870');**



**CREATE TABLE Reservas (**

**id\_reserva INT AUTO\_INCREMENT PRIMARY KEY,**

**id\_evento INT NOT NULL,**

**id\_cliente INT NOT NULL,**

**cantidad\_boletos INT NOT NULL,**

**total\_pago DECIMAL(12, 2) NOT NULL,**

**FOREIGN KEY (id\_evento) REFERENCES Eventos(id\_evento),**

**FOREIGN KEY (id\_cliente) REFERENCES Clientes(id\_cliente)**

**);**

**INSERT INTO Reservas (id\_evento, id\_cliente, cantidad\_boletos, total\_pago)**

**VALUES**

**(1, 1, 2, 300000.00),**

**(2, 2, 1, 80000.00),**

**(3, 3, 3, 300000.00),**

**(4, 4, 4, 240000.00),**

**(5, 5, 5, 360000.00),**

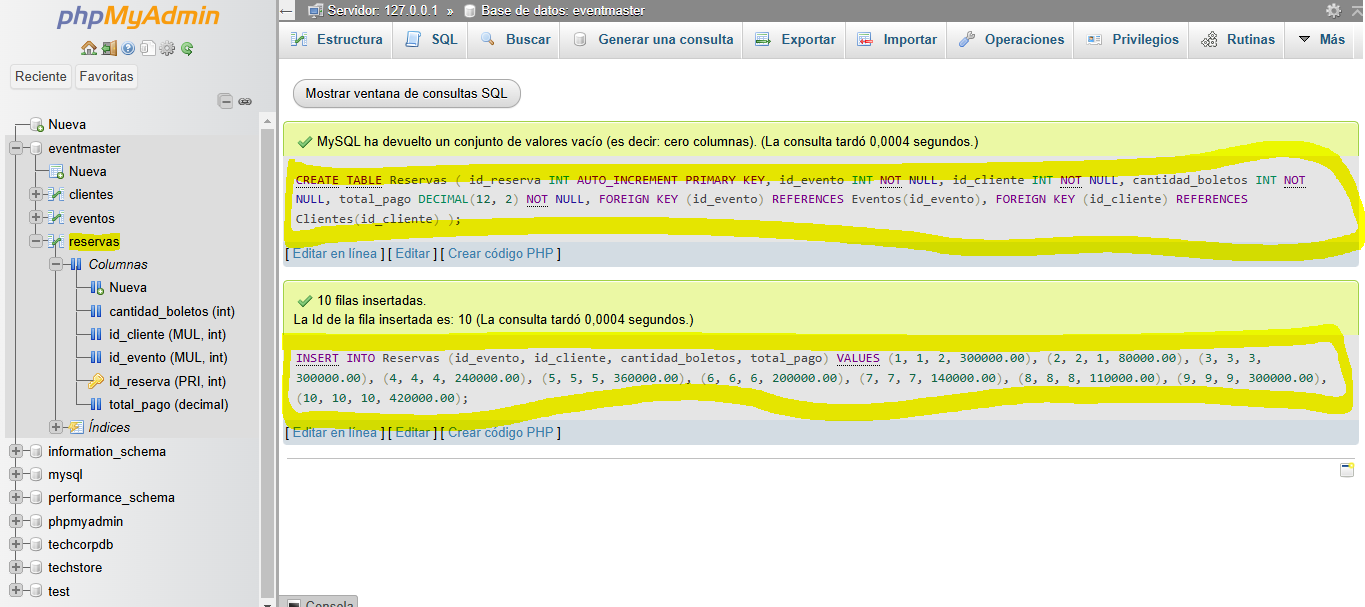
**(6, 6, 6, 200000.00),**

**(7, 7, 7, 140000.00),**

**(8, 8, 8, 110000.00),**

**(9, 9, 9, 300000.00),**

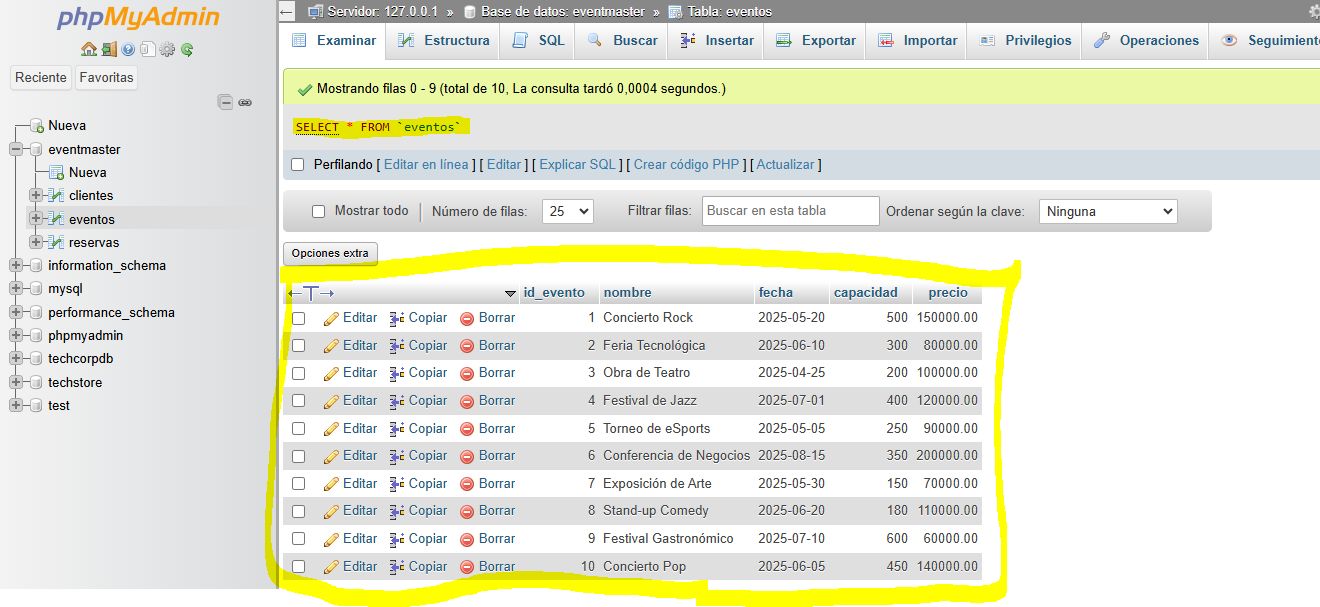
**(10, 10, 10, 420000.00);**



3. Realizar consultas SQL con respuestas esperadas.

1. Mostrar todos los eventos

**SELECT \* FROM Eventos;**

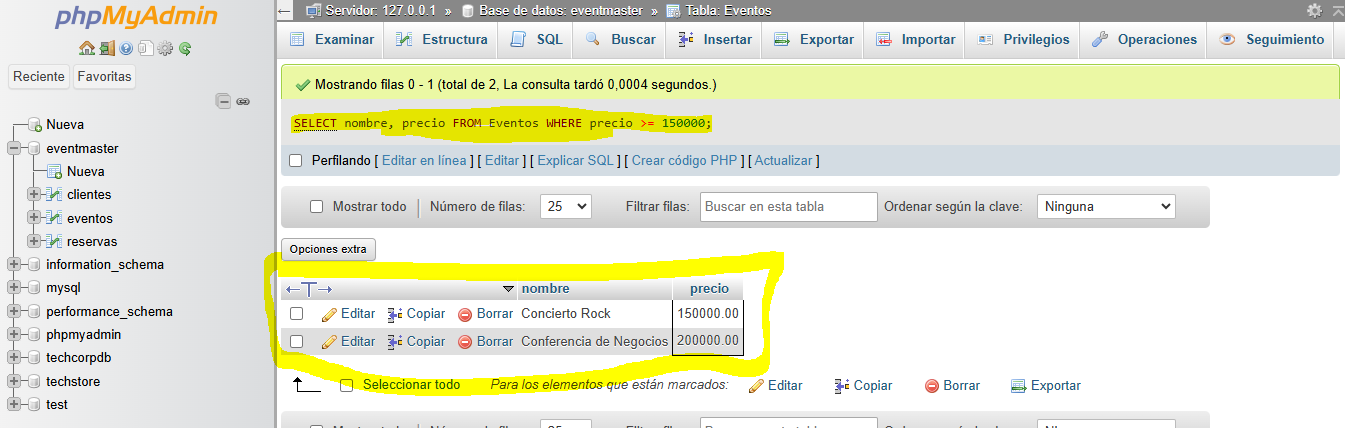


1. Obtener los nombres y precios de los eventos más costosos (mayores a $1.000.000) nota(se ajusto precio a $150.000, en la DB no hay precios de $1.000.000)

**SELECT nombre, precio**

**FROM Eventos**

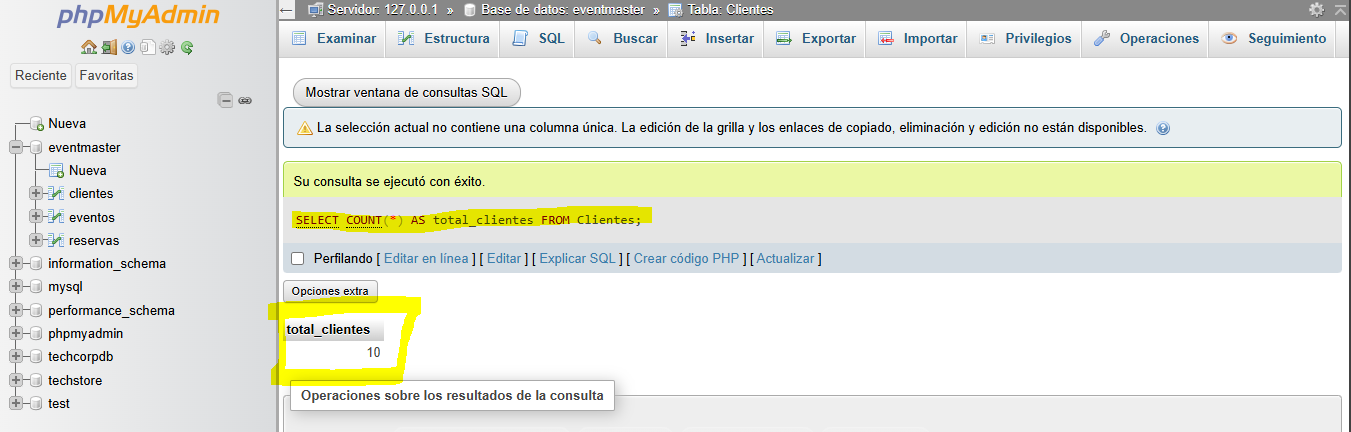
**WHERE precio >= 150000;**



1. Contar la cantidad total de clientes registrados.

**SELECT COUNT(\*) AS total\_clientes**

**FROM Clientes;**



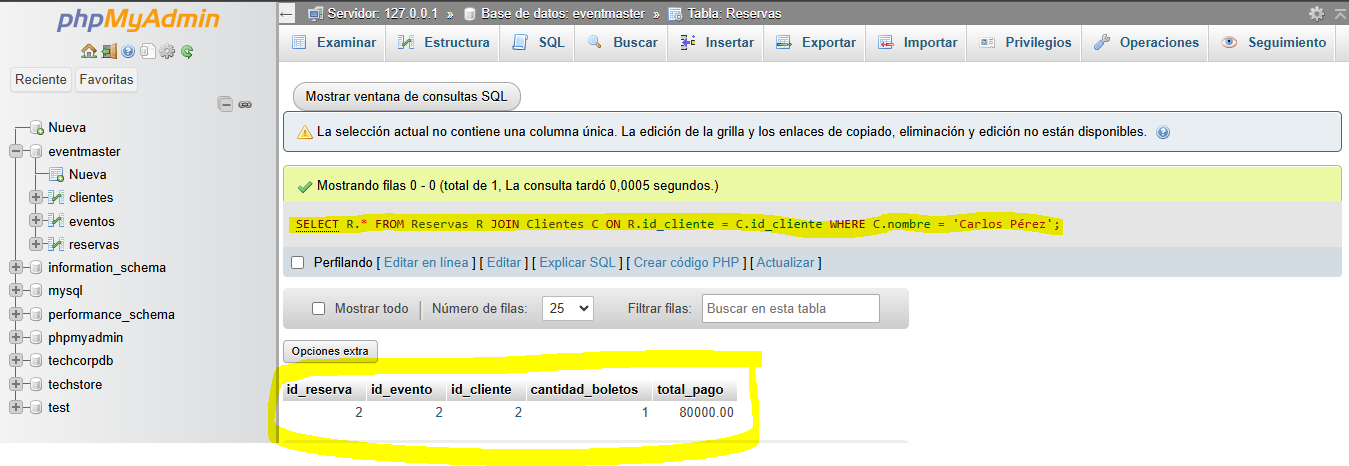
1. Mostrar las reservas realizadas por el cliente "Carlos Pérez"

**SELECT R.\***

**FROM Reservas R**

**JOIN Clientes C ON R.id\_cliente = C.id\_cliente**

**WHERE C.nombre = 'Carlos Pérez';**



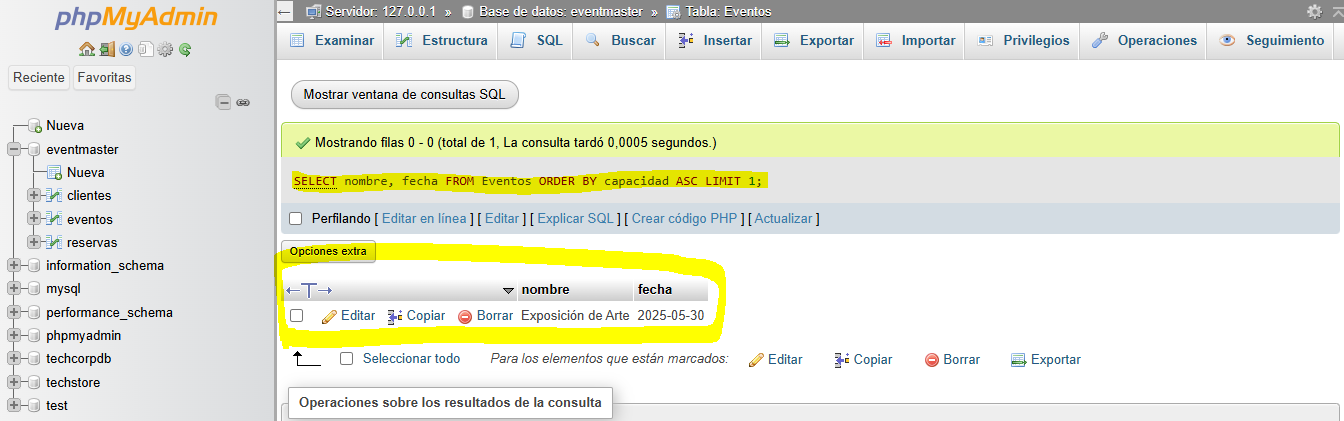
1. Obtener la fecha y el nombre del evento con menor capacidad.

**SELECT nombre, fecha**

**FROM Eventos**

**ORDER BY capacidad ASC**

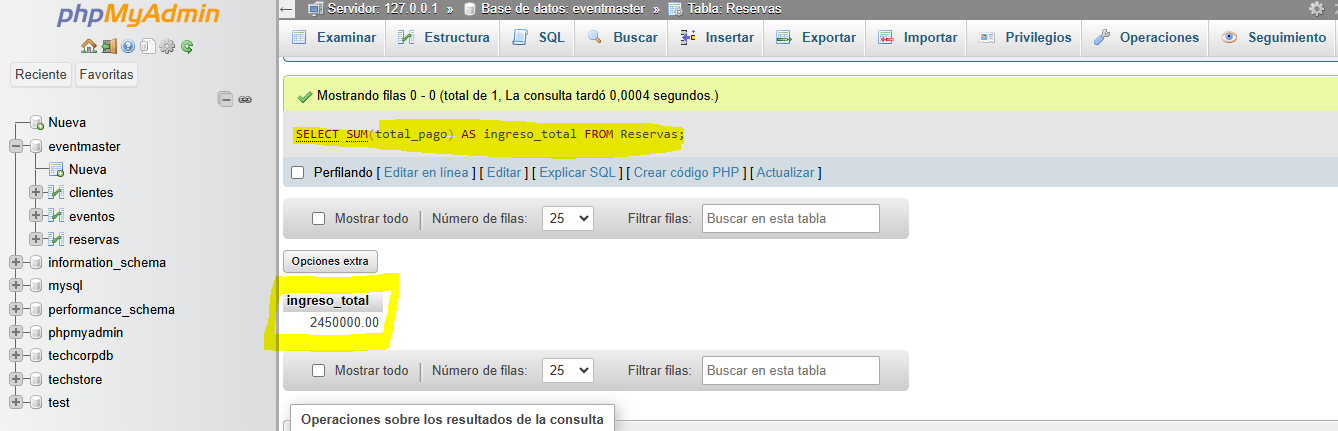
**LIMIT 1;**



1. Calcular el ingreso total de todas las reservas.

**SELECT SUM(total\_pago) AS ingreso\_total**

**FROM Reservas;**



1. Obtener el evento más reservado (mayor cantidad de boletos vendidos)

**SELECT E.nombre, SUM(R.cantidad\_boletos) AS total\_boletos**

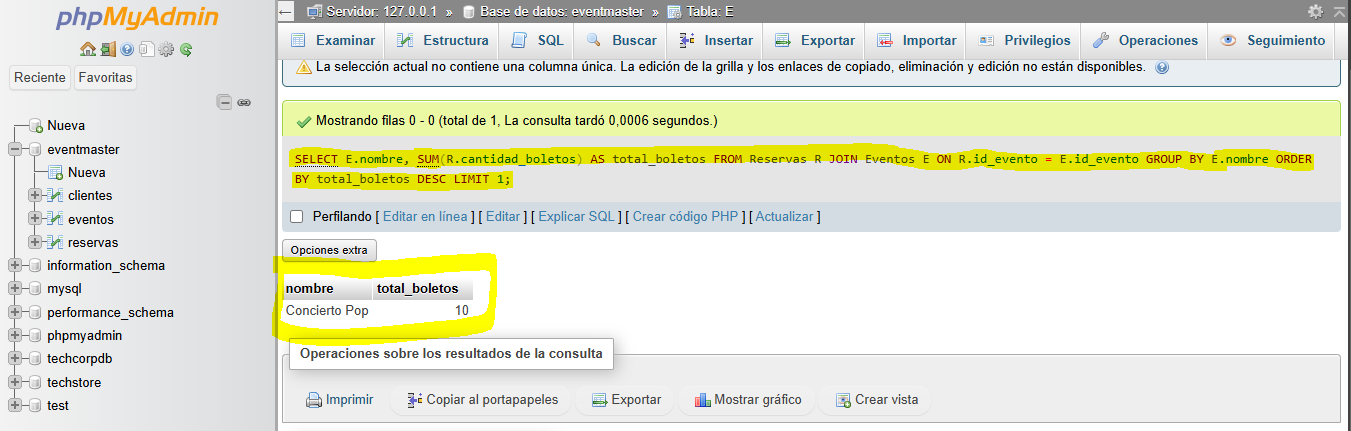
**FROM Reservas R**

**JOIN Eventos E ON R.id\_evento = E.id\_evento**

**GROUP BY E.nombre**

**ORDER BY total\_boletos DESC**

**LIMIT 1;**

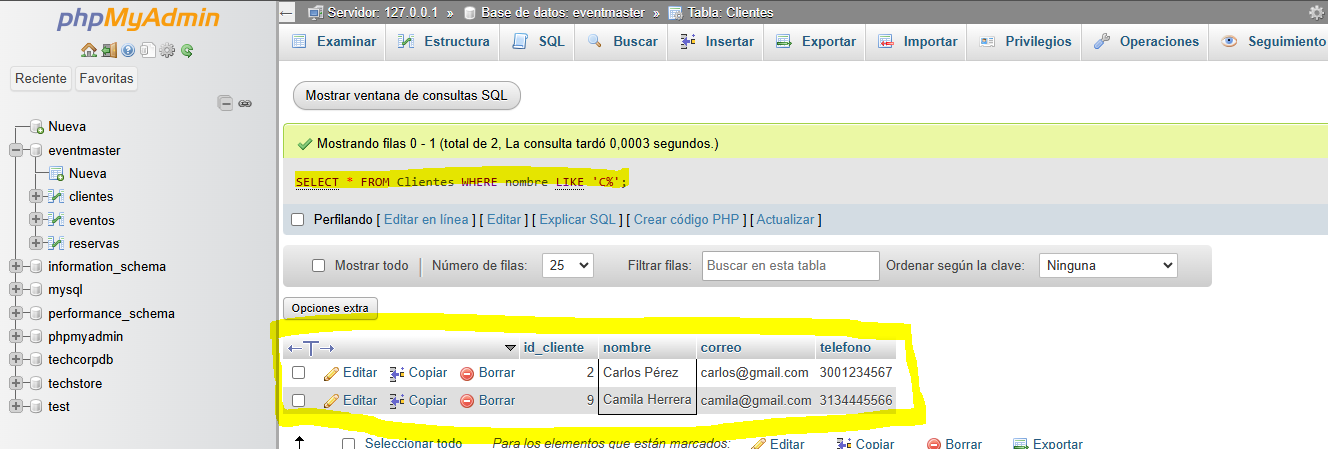


1. Mostrar clientes cuyo nombre inicie con "C"

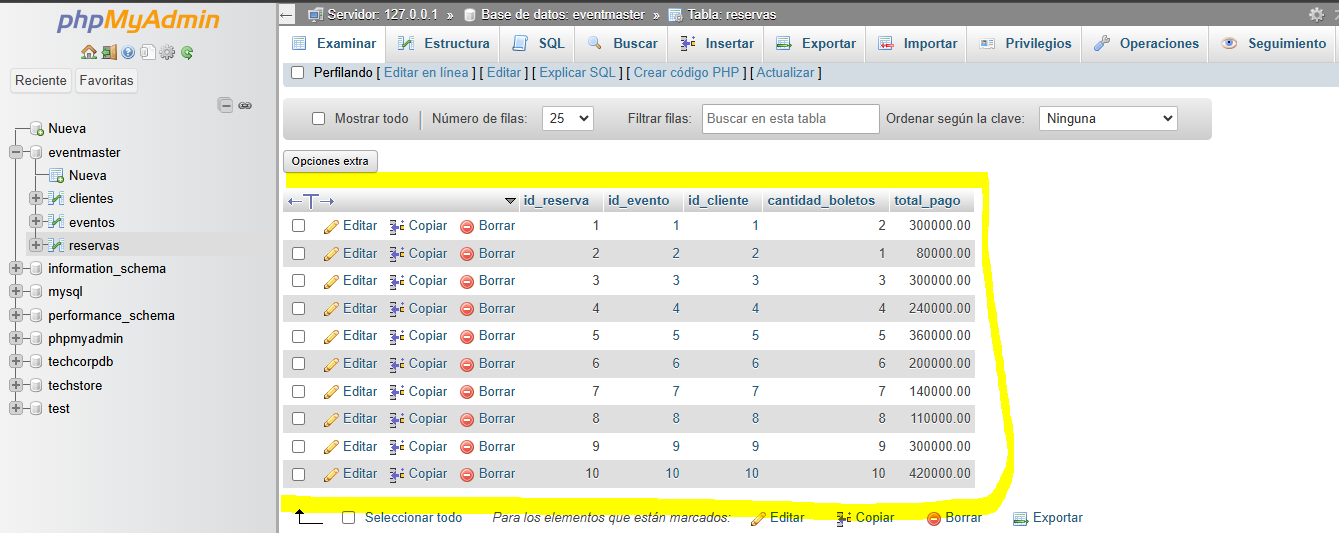
**SELECT \***

**FROM Clientes**

**WHERE nombre LIKE 'C%';**



1. Eliminar todas las reservas para el evento " Festival Gastronómico"



**DELETE FROM Reservas**

**WHERE id\_evento = (**

**SELECT id\_evento**

**FROM Eventos**

**WHERE nombre = 'Festival Gastronómico'**

**);** 