JEISSON DAVID GUABAVE VARGAS

DANIEL JAVIER MARIN

OSMAN ENRIQUE CAÑON GONZALEZ

FELIPE MARTINEZ

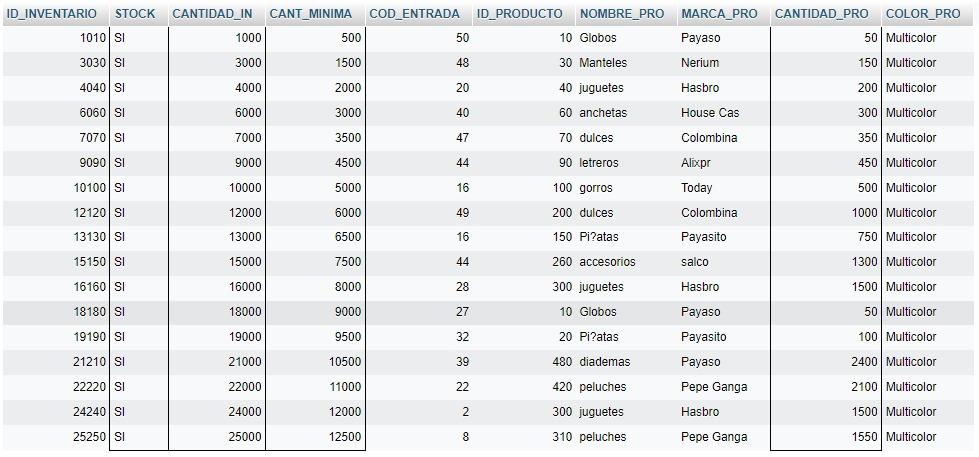
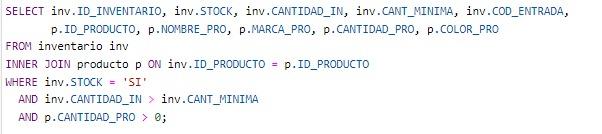
2558723

ADSO

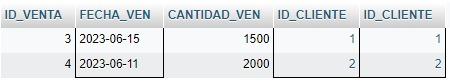
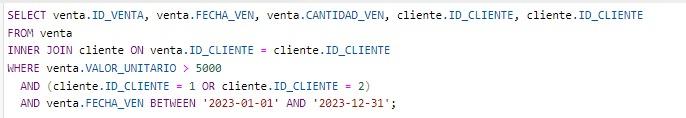
GAES Nro.2

CONSULTAS INNER JOIN

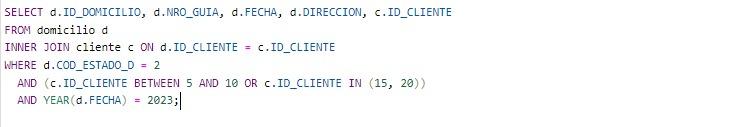
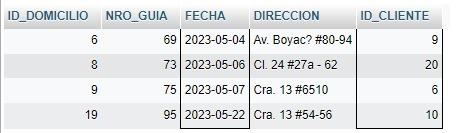
INVENTARIO:



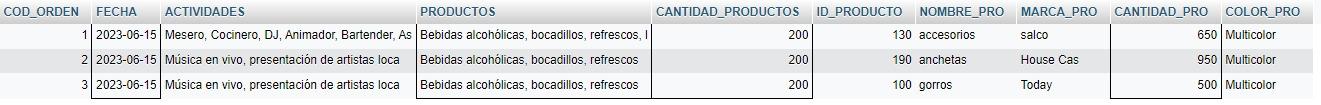
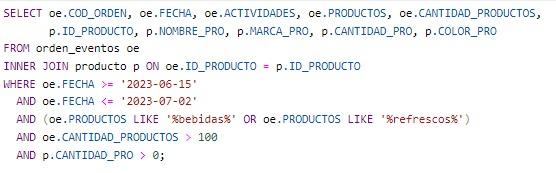
VENTAS:



DOMICILIOS:



EVENTOS:



PROCEDIMIENTOS:

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `InsertarEvento`(

IN p\_COD\_ORDEN INT,

IN p\_FECHA DATE,

IN p\_ACTIVIDADES VARCHAR(45),

IN p\_PRODUCTOS VARCHAR(45),

IN p\_CANTIDAD\_PRODUCTOS INT,

IN p\_DESCRIPCION\_EVENTO VARCHAR(45),

IN p ID\_PRODUCTO INT,

IN p\_ID\_VENTA INT,

IN p\_COD\_EVENTO INT

)

BEGIN

INSERT INTO `orden\_eventos` (

`COD\_ORDEN`,

`FECHA`,

`ACTIVIDADES`,

`PRODUCTOS`,

`CANTIDAD\_PRODUCTOS`,

`DESCRIPCION\_EVENTO`,

`ID\_PRODUCTO`,

`ID\_VENTA`,

`COD\_EVENTO`

) VALUES (

p\_COD\_ORDEN,

p\_FECHA,

p\_ACTIVIDADES,

p\_PRODUCTOS,

p\_CANTIDAD\_PRODUCTOS,

p\_DESCRIPCION\_EVENTO,

p\_ID\_PRODUCTO,

p\_ID\_VENTA,

p\_COD\_EVENTO

);

END$$

DELIMITER ;

LLAMAR: CALL InsertarEvento(8, '2023-09-01', 'Presentación musical', 'Bebidas alcohólicas, bocadillos', 150, 'Concierto', 250, 10, 4);

—------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `ObtenerDomiciliosPorEstado`(IN estado INT)

BEGIN

SELECT \* FROM domicilio WHERE COD\_ESTADO\_D = estado;

END$$

DELIMITER ;

LLAMAR: CALL ObtenerDomiciliosPorEstado(1)

—------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `ObtenerProductosPorMarca`(IN marca VARCHAR(45))

BEGIN

SELECT \*

FROM producto

WHERE MARCA\_PRO = marca;

END$$

DELIMITER ;

LLAMAR: CALL ObetenerProductosPorMarca (‘Payaso’)

—------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `RealizarVentaCompleja`(

IN p\_ID\_VENTA INT,

IN p\_FECHA\_VEN DATE,

IN p\_CANTIDAD\_VEN INT,

IN p\_ENVIO\_VEN VARCHAR(45),

IN p\_VALOR\_UNITARIO INT,

IN p\_DESCUENTO INT,

IN p\_IVA INT,

IN p\_ID\_PRODUCTO INT,

IN p\_ID\_CLIENTE INT

)

BEGIN

-- Variables locales

DECLARE v\_SUBTOTAL INT;

DECLARE v\_TOTAL INT;

-- Calcular subtotal

SET v\_SUBTOTAL = p\_CANTIDAD\_VEN \* p\_VALOR\_UNITARIO;

-- Calcular descuento

IF p\_DESCUENTO IS NOT NULL THEN

SET v\_SUBTOTAL = v\_SUBTOTAL - p\_DESCUENTO;

END IF;

-- Calcular IVA

SET v\_TOTAL = v\_SUBTOTAL + (v\_SUBTOTAL \* p\_IVA / 100);

-- Insertar venta

INSERT INTO `venta` (

`ID\_VENTA`,

`FECHA\_VEN`,

`CANTIDAD\_VEN`,

`ENVIO\_VEN`,

`VALOR\_UNITARIO`,

`DESCUENTO`,

`IVA`,

`SUBTOTAL`,

`TOTAL`,

`ID\_PRODUCTO`,

`ID\_CLIENTE`

) VALUES (

p\_ID\_VENTA,

p\_FECHA\_VEN,

p\_CANTIDAD\_VEN,

p\_ENVIO\_VEN,

p\_VALOR\_UNITARIO,

p\_DESCUENTO,

p\_IVA,

v\_SUBTOTAL,

v\_TOTAL,

p\_ID\_PRODUCTO,

p\_ID\_CLIENTE

);

SELECT 'Venta realizada correctamente.' AS Resultado;

-- Realizar otras acciones complejas aquí...

END$$

DELIMITER ;

LLAMAR: CALL RealizarVentaCompleja(1, '2023-06-01', 10, 'Envío rápido', 100, 10, 16, 1, 1);

—------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `obtenerActividades`()

BEGIN

SELECT \* FROM actividades;

END$$

DELIMITER ;



—------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `obtenerDireccionCliente`(IN idCliente INT)

BEGIN

DECLARE direccionCliente VARCHAR(45);

-- Obtener la dirección del cliente según su ID

SELECT DIRECCION INTO direccionCliente

FROM domicilio

WHERE ID\_CLIENTE = idCliente

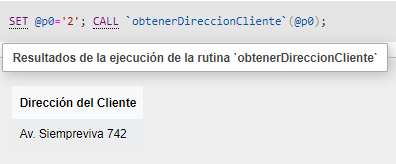
LIMIT 1;

-- Imprimir la dirección del cliente

SELECT direccionCliente AS 'Dirección del Cliente';

END$$

DELIMITER ;



—-----------------------------------------------------------------------------------------------------------------------

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `obtenerEventos`()

BEGIN

SELECT \* FROM eventos;

END$$

DELIMITER ;



—-----------------------------------------------------------------------------------------------------------------------

DELIMITER $$

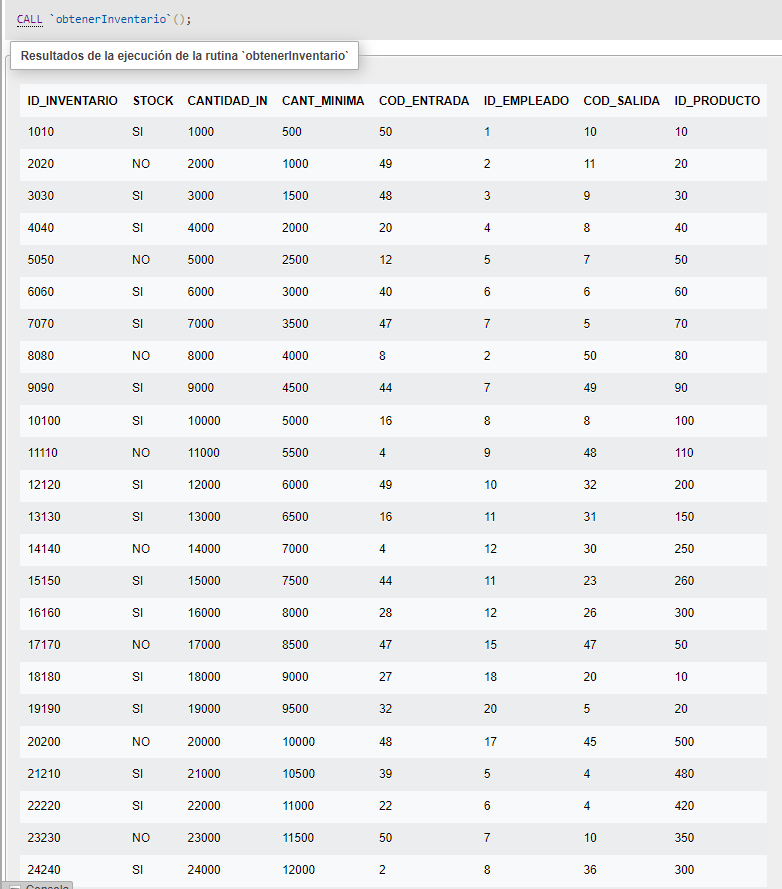
CREATE DEFINER=`root`@`localhost` PROCEDURE `obtenerInventario`()

BEGIN

SELECT \* FROM inventario;

END$$

DELIMITER ;



TRIGGERS:

CREATE TRIGGER `actividades\_trigger` AFTER INSERT ON `actividades`

FOR EACH ROW BEGIN

– Acciones a realizar después de insertar un nuevo registro en la tabla

-- Ejemplo: INSERT INTO log\_actividades (COD\_ACTIVIDAD, ACCION) VALUES (NEW.COD\_ACTIVIDAD, 'Insertado');

-- Aquí puedes realizar la lógica personalizada que necesites

END

—-----------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `actividades\_trigger\_after\_update` AFTER UPDATE ON `actividades`

FOR EACH ROW BEGIN

-- Acciones a realizar después de actualizar un registro en la tabla

-- Ejemplo: INSERT INTO log\_actividades (COD\_ACTIVIDAD, ACCION) VALUES (NEW.COD\_ACTIVIDAD, 'Actualizado');

-- Aquí puedes realizar la lógica personalizada que necesites

END

—-----------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `actualizarStock` AFTER INSERT ON `venta`

FOR EACH ROW BEGIN

DECLARE stockActual int;

DECLARE cantidadVenta int;

DECLARE nuevoStock int;

SELECT CANTIDAD\_IN INTO stockActual FROM inventario WHERE ID\_INVENTARIO = NEW.ID\_PRODUCTO;

SELECT CANTIDAD\_VEN INTO cantidadVenta FROM venta WHERE ID\_VENTA = NEW.ID\_VENTA;

SET nuevoStock = stockActual - cantidadVenta;

UPDATE inventario SET CANTIDAD\_IN = nuevoStock WHERE ID\_INVENTARIO = NEW.ID\_PRODUCTO;

END

—-----------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `afterInsertDomicilio` AFTER INSERT ON `domicilio`

FOR EACH ROW BEGIN

-- Obtener el ID de venta del nuevo registro de domicilio

DECLARE idVenta INT;

SET idVenta = NEW.ID\_VENTA;

-- Actualizar el campo "ENVIO\_VEN" en la tabla "venta" según el ID de venta

UPDATE venta

SET ENVIO\_VEN = 'Enviado'

WHERE ID\_VENTA = idVenta;

END

—-----------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `calcularTotal` BEFORE INSERT ON `venta`

FOR EACH ROW BEGIN

DECLARE subtotal INT;

DECLARE total INT;

SET subtotal = NEW.CANTIDAD\_VEN \* NEW.VALOR\_UNITARIO;

SET total = subtotal - NEW.DESCUENTO + NEW.IVA;

SET NEW.SUBTOTAL = subtotal;

SET NEW.TOTAL = total;

END

—------------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `eventos\_trigger` AFTER INSERT ON `eventos`

FOR EACH ROW BEGIN

-- Acciones a realizar después de insertar un nuevo registro en la tabla

-- Ejemplo: INSERT INTO log\_eventos (COD\_EVENTO, ACCION) VALUES (NEW.COD\_EVENTO, 'Insertado');

-- Aquí puedes realizar la lógica personalizada que necesites

END

—------------------------------------------------------------------------------------------------------------------------

CREATE TRIGGER `eventos\_trigger\_after\_update` AFTER UPDATE ON `eventos`

FOR EACH ROW BEGIN

-- Acciones a realizar después de actualizar un registro en la tabla

-- Ejemplo: INSERT INTO log\_eventos (ID\_EVENTO, ACCION) VALUES (NEW.COD\_EVENTO, 'Actualizado');

-- Aquí puedes realizar la lógica personalizada que necesites

END