



Proyecto Final

EMPRESA “ROMBO PISOS”

CURSO: SQL

ESTUDIANTE: MARÍA SOL GARCÍA RIVEROS

COMISIÓN: 72705

TUTOR: NICOLÁS MAUGERI

PROFESOR: ANDERSON MICHAEL TORRES

FECHA ENTREGA FINAL: 30/01/25



ÍNDICE

| | |
|--|----|
| Introducción | 3 |
| Situación Problemática | 3 |
| Modelo de Negocio | 3 |
| Diagrama Entidad-Relación | 4 |
| Listado de Tablas y Descripción | 4 |
| TABLA Venta y Control de los Movimientos (Tabla de Hecho) | 4 |
| TABLA Productos de los Pedidos (Tabla Intermedia) | 4 |
| TABLA Vendedor (Tabla Dimensional) | 5 |
| TABLA Repartidor (Tabla Dimensional) | 5 |
| TABLA Transporte (Tabla Dimensional) | 5 |
| TABLA Pedido (Tabla de Hecho) | 5 |
| TABLA Cliente (Tabla Dimensional) | 5 |
| TABLA Encuesta de Satisfacción (Tabla Dimensional) | 6 |
| TABLA Proveedor (Tabla Dimensional) | 6 |
| TABLA Producto Baldosas (Tabla Dimensional) | 6 |
| TABLA Tipo de Color de las Baldosas (Tabla Dimensional) | 6 |
| TABLA Tipo de Size de las Baldosas (Tabla Dimensional) | 6 |
| TABLA Tipo de Material de las Baldosas (Tabla Dimensional) | 6 |
| Objetos de la Base de datos | 7 |
| Objetos de la Base de datos: Views | 7 |
| 1 VIEW top10_clientes | 7 |
| 2 VIEW cliente_producto_comprado | 7 |
| 3 VIEW tareas_pendientes_repartidor | 8 |
| 4 VIEW productos_vendidos | 8 |
| 5 VIEW pedido_pago_status | 9 |
| Objetos de la Base de datos: Functions | 9 |
| 1 Funcion calculo_monto_total_pedido | 9 |
| 2 Funcion calculo_promedio_satisfaccion | 9 |
| 3 Funcion calculo_pedido_pendiente | 10 |

| | |
|---|----|
| 4 Funcion calculo_baldosas_vendidas_por_producto..... | 10 |
| Objetos de la Base de datos: Procedure | 11 |
| 1 Procedure update_status_pedido..... | 11 |
| 2 Procedure update_status_pedido..... | 11 |
| Objetos de la Base de datos: Trigger..... | 11 |
| 1 Trigger validar_pedido | 11 |
| 2 Trigger eliminar_pedidos_cancelados | 12 |
| Inserción de datos | 12 |
| Usuarios..... | 13 |
| Transaction Control Language | 13 |
| CASO 1..... | 13 |
| CASO 2..... | 13 |
| Informe..... | 14 |
| USO: Carpeta DB_SCRIPTS | 16 |
| Back Up..... | 16 |
| Herramientas Utilizadas | 16 |

Introducción

La empresa planea expandir las ventas a nuevos clientes de Chile y Uruguay. Para ello, deberá contar con una administración y distribución local adecuada que le permita cumplir sus objetivos.

Situación Problemática

La empresa “Rombo Pisos”, con clientes distribuidos por todo el país, enfrenta la necesidad de un mayor control sobre los pedidos y la información diaria para optimizar su desempeño y realizar más ventas. En este contexto, se ha solicitado la implementación de una base de datos relacional que mejore la eficiencia en la recepción, registro y entrega de pedidos en las actividades rutinarias.

Para esta primera fase de prueba, se cuenta con registros de dos semanas de trabajo. Si esta solución resulta efectiva, la empresa podrá extender su uso a toda su información.

Modelo de Negocio

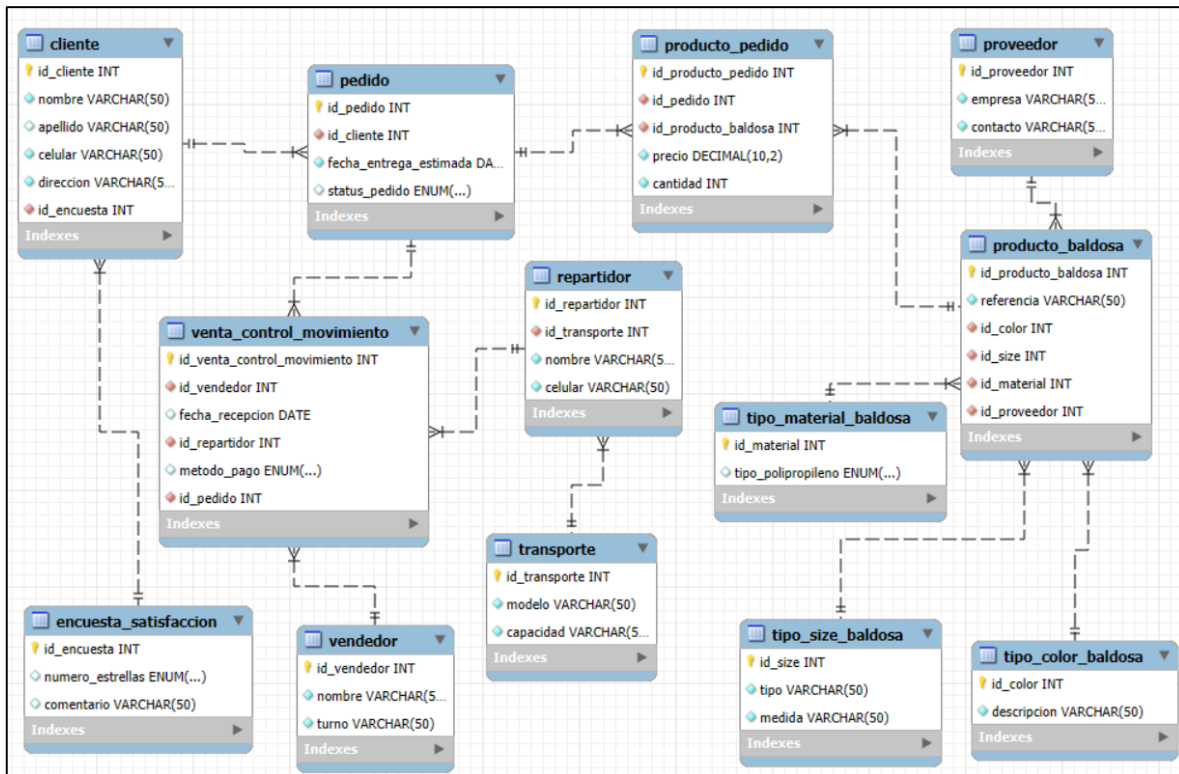
“Rombo Pisos” es una empresa -establecida en 2021- dedicada a la reventa de pisos, especializándose en baldosas de polipropileno encastrables para uso industrial y hogareño. Su propuesta de valor se centra en ofrecer una solución de pavimentación que transforme los espacios de trabajo en entornos más confortables y funcionales, mediante el proceso 3F: fácil colocación, fácil limpieza y fácil desarme.

La empresa opera principalmente en la Zona Oeste de Buenos Aires y atiende a clientes de toda la República Argentina, abarcando tanto el sector residencial como el industrial. Rombo Pisos se enfoca en la satisfacción del cliente mediante baldosas de calidad y un servicio ágil.

La implementación de una base de datos relacional permitirá a Rombo Pisos optimizar la gestión de pedidos y mejorar la experiencia del cliente, apoyando su crecimiento y expansión en el mercado.

Diagrama Entidad-Relación

El DER propuesto es el siguiente:



Listado de Tablas y Descripción

La creación de las tablas se realizará mediante el script 1_DB_Tablas.sql ubicado en la carpeta DB_SCRIPTS.

A continuación, se presentarán las tablas generadas.

TABLA Venta y Control de los Movimientos (Tabla de Hecho)

Permitir registrar las ventas con sus particularidades como fecha de recepción.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|--------------------------|-----------------------------|---|-------------|
| venta_control_movimiento | id_venta_control_movimiento | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | id_vendedor | INT NOT NULL | FOREIGN KEY |
| | fecha_recepcion | DATE DEFAULT (CURRENT_TIMESTAMP) | |
| | id_repartidor | INT NOT NULL | FOREIGN KEY |
| | metodo_pago | ENUM ('efectivo','transferencia','debito') DEFAULT 'efectivo' | |
| | id_pedido | INT NOT NULL | FOREIGN KEY |

TABLA Productos de los Pedidos (Tabla Intermedia)

Permite asociar los productos (baldosas) que conforman cada pedido.

| Tabla | Columna | Columna1 | Tipo KEY |
|-----------------|---------------------|-----------------------------|-------------|
| producto_pedido | id_producto_pedido | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | id_pedido | INT NOT NULL | FOREIGN KEY |
| | id_producto_baldosa | INT NOT NULL | FOREIGN KEY |
| | precio | DECIMAL(10,2) NOT NULL | |
| | calidad | INT NOT NULL | |

TABLA Vendedor (Tabla Dimensional)

Contiene los datos de los vendedores que toman el pedido a los clientes.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|----------|-------------|-----------------------------|-------------|
| vendedor | id_vendedor | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | nombre | VARCHAR(50) NOT NULL | |
| | turno | VARCHAR(50) NOT NULL | |

TABLA Repartidor (Tabla Dimensional)

Posee información sobre los repartidores que realizan las entregas.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|------------|---------------|-----------------------------|-------------|
| repartidor | id_repartidor | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | id_transporte | INT NOT NULL | FOREIGN KEY |
| | nombre | VARCHAR(50) NOT NULL | |
| | celular | VARCHAR(50) NOT NULL | |

TABLA Transporte (Tabla Dimensional)

Posee información sobre los transportes utilizados para la distribución de pedidos.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|------------|---------------|-----------------------------|-------------|
| transporte | id_transporte | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | modelo | VARCHAR(50) NOT NULL | |
| | capacidad | VARCHAR(50) NOT NULL | |

TABLA Pedido (Tabla de Hecho)

Permite registrar los pedidos desde su fecha de entrega hasta quien realizo el encargo.

| Tabla | Columna | Columna1 | Tipo KEY |
|--------|------------------------|--|-------------|
| pedido | id_pedido | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | id_cliente | INT NOT NULL | FOREIGN KEY |
| | fecha_entrega_estimada | DATE NOT NULL | |
| | status_pedido | ENUM ('pendiente','en proceso','entregado','cancelado') DEFAULT 'pendiente' | |

TABLA Cliente (Tabla Dimensional)

Contiene los datos personales de los clientes como su dirección.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|---------|-------------|-----------------------------|-------------|
| cliente | id_cliente | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | nombre | VARCHAR (50) NOT NULL | |
| | apellido | VARCHAR (50) DEFAULT NULL | |
| | celular | VARCHAR (50) NOT NULL | |
| | direccion | VARCHAR(50) NOT NULL | |
| | id_encuesta | INT NOT NULL | FOREIGN KEY |

TABLA Encuesta de Satisfacción (Tabla Dimensional)

Contiene las calificaciones que los clientes realizan del servicio.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|-----------------------|------------------|--------------------------------|-------------|
| encuesta_satisfaccion | id_encuesta | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | numero_estrellas | ENUM ('0','1','2','3','4','5') | |
| | comentario | VARCHAR (50) DEFAULT NULL | |

TABLA Proveedor (Tabla Dimensional)

Contiene los datos de los proveedores de las baldosas que se comercializan.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|-----------|--------------|-----------------------------|-------------|
| proveedor | id_proveedor | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | empresa | VARCHAR(50) NOT NULL | |
| | contacto | VARCHAR(50) NOT NULL | |

TABLA Producto Baldosas (Tabla Dimensional)

Contiene los productos disponibles y sus características.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|------------------|---------------------|-----------------------------|-------------|
| producto_baldosa | id_producto_baldosa | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | referencia | VARCHAR(50) NOT NULL | |
| | id_color | INT NOT NULL | FOREIGN KEY |
| | id_size | INT NOT NULL | FOREIGN KEY |
| | id_material | INT NOT NULL | FOREIGN KEY |
| | id_proveedor | INT NOT NULL | FOREIGN KEY |

TABLA Tipo de Color de las Baldosas (Tabla Dimensional)

Detalla la característica “color” de los productos.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|--------------------|-------------|-----------------------------|-------------|
| tipo_color_baldosa | id_color | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | descripcion | VARCHAR(50) NOT NULL | |

TABLA Tipo de Size de las Baldosas (Tabla Dimensional)

Detalla la característica “medida” de los productos.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|-------------------|---------|-----------------------------|-------------|
| tipo_size_baldosa | id_size | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | tipo | VARCHAR(50) NOT NULL | |
| | medida | VARCHAR(50) NOT NULL | |

TABLA Tipo de Material de las Baldosas (Tabla Dimensional)

Detalla la característica “material” de los productos.

| Tabla | Columna | Tipo de Datos | Tipo KEY |
|-----------------------|--------------------|--|-------------|
| tipo_material_baldosa | id_material | INT NOT NULL AUTO_INCREMENT | PRIMARY KEY |
| | tipo_polipropileno | ENUM ('virgen','reciclado') DEFAULT 'virgen' | |

Objetos de la Base de datos

La creación de los objetos se realizará mediante el script 2_DB_Datos.sql ubicado en la carpeta DB_SCRIPTS.

Objetos de la Base de datos: Views

1 VIEW top10_clientes

Descripción: Muestra el top 10 de los clientes que realizaron más pedidos.

Objetivo: Poder observar los clientes más relevantes con mayor adquisición de baldosas.

Columnas:

id_cliente: Id del cliente.

nombre_cliente: Nombre del cliente.

cantidad_pedidos: Cantidad de pedidos.

total_venta: Monto total de los pedidos.

Tablas:

- cliente
- pedido
- producto_pedido

2 VIEW cliente_producto_comprado

Descripción: Muestra las cantidades y monto total por cada producto comprado por cliente.

Objetivo: Poder observar las baldosas adquiridas por cada cliente.

Columnas:

id_cliente: Id del cliente.

id_baldosa: Id del producto.

cantidad_baldosa: Cantidad de baldosas adquiridas.

monto_total: Monto total de las baldosas adquiridas.

Tablas:

- pedido
- producto_pedido

3 VIEW tareas_pendientes_repartidor

Descripción: Muestra aquellos pedidos tomados por los vendedores, pero aun no entregados por los repartidores, es decir, en estado pendiente.

Objetivo: Poder observar los pedidos pendientes de forma sencilla.

Columnas:

id_pedido: Id del pedido.

fecha_recepcion: Fecha de recepción del pedido.

id_vendedor: Id del vendedor.

nombre_vendedor: Nombre del vendedor.

id_repartidor: Id del repartidor.

nombre_repartidor: Nombre del repartidor.

fecha_entrega: Fecha de entrega estimada.

status_pedido: Status del pedido.

Tablas:

- venta_control_movimiento
- vendedor
- pedido
- repartidor

4 VIEW productos_vendidos

Descripción: Muestra las cantidades vendidas de cada modelo de baldosa.

Objetivo: Poder observar la cantidad de unidades vendidas por baldosa.

Columnas:

empresa_proveedor: Nombre de la empresa proveedora.

nombre_modelo_baldosa: Nombre del modelo de la baldosa.

color_baldosa: Color de la baldosa.

material_baldosa: Material de la baldosa.

medida_baldosa: Medida de la baldosa.

cantidad_vendida: Cantidades vendidas.

Tablas:

- proveedor

- producto_baldosa
- tipo_color_baldosa
- tipo_material_baldosa
- tipo_size_baldosa
- producto_pedido

5 VIEW pedido_pago_status

Descripción: Muestra cuando los pedidos fueron recibidos junto con su estado y método de pago.

Columnas:

fecha_recepcion: Fecha de recepción del pedido.

id_pedido: Id del pedido.

estado: Status del pedido.

forma_de_pago: Metodo de pago.

Tablas:

- pedido
- venta_control_movimiento

Objetos de la Base de datos: Functions

1 Funcion calculo_monto_total_pedido

Descripción: Permite calcular el monto monetario total del pedido.

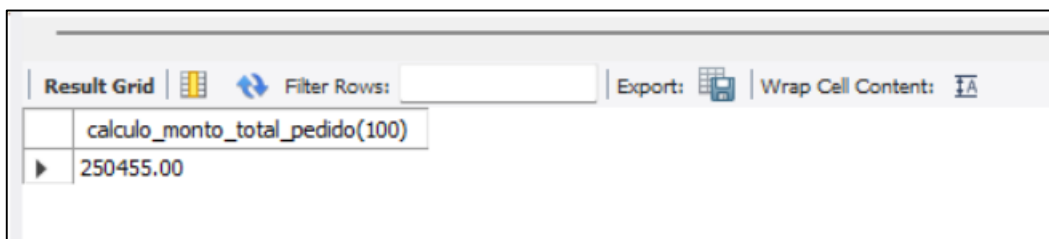
Parámetro: id_pedido INT

Retorno: DECIMAL(10,2) El valor total del pedido.

Ejemplo de uso:

```
SELECT calculo_monto_total_pedido(100);
```

Retorno del ejemplo de uso:



| Result Grid | | Filter Rows: | Export: | Wrap Cell Content: |
|-------------|---------------------------------|--------------|---------|--------------------|
| | calculo_monto_total_pedido(100) | | | |
| ▶ | 250455.00 | | | |

2 Funcion calculo_promedio_satisfaccion

Descripción: Devuelve el promedio de calificación de todas las encuestas de satisfacción.

Parámetro: No es necesario.

Retorno: DECIMAL (10,1) Promedio de satisfacción de todas las encuestas cargadas.

Ejemplo de uso:

```
SELECT calculo_promedio_satisfaccion();
```

Retorno del ejemplo de uso:

| Result Grid | | Filter Rows: | Export: | Wrap Cell Content: |
|-------------|---------------------------------|--------------|---------|--------------------|
| | calculo_promedio_satisfaccion() | | | |
| ▶ | 3.0 | | | |

3 Funcion calculo_pedido_pendiente

Descripción: Devuelve la cantidad de pedidos pendiente.

Parámetro: No es necesario.

Retorno: INT Cantidad de pedidos pendientes a la fecha.

Ejemplo de uso:

```
SELECT calculo_pedido_pendiente ();
```

Retorno del ejemplo de uso:

| Result Grid | | Filter Rows: | Export: | Wrap Cell Content: |
|-------------|-----------------------------|--------------|---------|--------------------|
| | calculo_pedido_pendiente () | | | |
| ▶ | 15 | | | |

4 Funcion calculo_baldosas_vendidas_por_producto

Descripción: Devuelve la cantidad de baldosas vendidas por producto.

Parámetro: id_producto_baldosa INT

Retorno: INT Cantidad de baldosas vendidas por producto.

Ejemplo de uso:

```
SELECT calculo_baldosas_vendidas_por_producto (1);
```

Retorno del ejemplo de uso:

| | |
|---|--------------|
| Result Grid | Filter Rows: |
| calculo_baldosas_vendidas_por_producto(1) | |
| 8949 | |

Objetos de la Base de datos: Procedure

1 Procedure update_status_pedido

Descripción: Actualiza el estado del pedido

Parametros:

parametro_id_pedido (INT)

parametro_status ENUM ('pendiente','en proceso','entregado','cancelado')

Retorno: Actualiza un registro en una tabla.

Ejemplo de uso:

```
CALL update_status_pedido (2,'entregado');
```

2 Procedure update_status_pedido

Descripción: Actualiza el estado del pedido

Parametros:

parametro_tabla VARCHAR(50)

parametro_campo VARCHAR(50)

parametro_orden ENUM('ASC','DESC')

Retorno: Ordena los registros de una tabla en particular a partir de un campo especifico de forma ascendente o descendente.

Ejemplo de uso:

```
CALL ordenamiento_tabla ('vendedor','nombre','DESC');
```

Objetos de la Base de datos: Trigger

1 Trigger validar_pedido

Tabla: venta_control_movimiento

Evento: BEFORE INSERT

Acción: Verifica que antes de agregar un registro nuevo en la tabla venta_control_movimiento exista el id_pedido en la tabla pedido.

Script:

```
DELIMITER $$
CREATE TRIGGER validar_pedido
BEFORE INSERT ON venta_control_movimiento
FOR EACH ROW
BEGIN
    IF NOT EXISTS (SELECT id_pedido FROM pedido WHERE id_pedido= NEW.id_pedido) THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Id_pedido no existe';
    END IF;
END$$
DELIMITER ;
```

2 Trigger eliminar_pedidos_cancelados

Tabla: venta_control_movimiento

Evento: AFTER UPDATE ON

Acción: Elimina de la tabla venta control movimientos aquellos registros cuyo status en la tabla pedido sea actualizado a status "cancelado".

Script:

```
DELIMITER $$
CREATE TRIGGER eliminar_pedidos_cancelados
AFTER UPDATE ON pedido
FOR EACH ROW
BEGIN
    IF NEW.status_pedido = 'cancelado' AND OLD.status_pedido != 'cancelado' THEN
        DELETE FROM venta_control_movimiento a WHERE a.id_pedido = OLD.id_pedido
        LIMIT 1;
    END IF;
END $$
DELIMITER ;
```

Inserción de datos

La inserción de datos se realizará mediante el script 3_DB_Datos.sql ubicado en la carpeta DB_SCRIPTS.

Usuarios

La creación y otorgamiento de permisos de los usuarios se realizará mediante el script 4_DB_Usuarios.sql ubicado en la carpeta DB_SCRIPTS.

El relevamiento de usuarios es el siguiente:

| Usuario | Contraseña | Permisos |
|---------------------------------------|-------------|--|
| userlectura@localhost | User1JDBC1* | Permiso de solo lectura de todas las tablas |
| userdatos@localhost | User2JDBC2* | Permisos de lectura, inserción y modificación de datos |

Transaction Control Language

Adicionalmente, se realizó a modo de ejemplo dos casos de la modificación de registros controlado por transacciones los cuales se podrán ejecutar mediante el script 5_DB_Transaction.sql ubicado en la carpeta DB_SCRIPTS.

En el mismo, se encontrarán comentadas ciertas líneas el cual permitirán reutilizar el script y de esta forma, deshacer y /o confirmar el cambio.

CASO 1

Tabla: venta_control_movimiento

Uso: Eliminación de un registro controlado por transacciones.

```
/*CASO 1*/
/*Verificar la existencia del registro*/
SELECT * FROM venta_control_movimiento
WHERE id_venta_control_movimiento=1;

/*Iniciar Transaccion*/
START TRANSACTION;

/*Eliminar una transaccion de una tabla*/
DELETE FROM venta_control_movimiento
WHERE id_venta_control_movimiento=1;

/*En caso de deshacer el cambio*/
/*ROLLBACK;*/

/*Confirmar transaccion*/
/*COMMIT;*/
```

CASO 2

Tabla: vendedor

Uso: Insertar registros y guardar puntos controlado por transacciones.

```

/*CASO 2*/
/*Verificar la existencia de los registros*/
SELECT * FROM vendedor;

/*Iniciar Transaccion*/
START TRANSACTION;

/*Agregar los primeros 4 registros*/
INSERT INTO vendedor (nombre, turno) VALUES
('Javier','Turno A 9.00hs a 13.30hs') ,
('Santiago','Turno A 9.00hs a 13.30hs'),
('Jonathan','Turno C 15.00hs a 20.00hs'),
('Melanie','Turno C 15.00hs a 20.00hs');

/*Guardar*/
SAVEPOINT punto4registro;

/*Agregar los ultimos 4 registros*/
INSERT INTO vendedor (nombre, turno) VALUES
('Sebastian','Turno A 9.00hs a 13.30hs') ,
('Marcela','Turno A 9.00hs a 13.30hs'),
('Sol','Turno B 13.30hs a 18.00hs'),
('Ezequiel','Turno B 13.30hs a 18.00hs');

/*Guardar*/
SAVEPOINT punto8registro;

/*Eliminar savepoint*/
/*RELEASE SAVEPOINT punto4registro;*/

/*Deshacer el cambio*/
/*ROLLBACK;*/

/*Confirmar transaccion*/
/*COMMIT;*/

```

Informe

En base a la información de muestra almacenada en las tablas, se generó un informe principal nutrido por las tablas de clientes, pedido y producto_pedido.

El mismo permite visualizar información como:

- ☒ cuantos pedidos corresponden a cada estado (como pendiente, cancelado y entregado)

- ☒ aquellos pedidos que no fueron señados por los clientes (también llamados “Sin Estado”)
- ☒ pedidos totales
- ☒ ingresos totales
- ☒ la evolución diaria de los ingresos
- ☒ unidades vendidas totales
- ☒ unidades vendidas por modelo de baldosa
- ☒ el modelo de baldosa estrella (es decir, más vendida)

El informe generado es el siguiente:



USO: Carpeta DB_SCRIPTS

Esta carpeta contiene por separado los scripts para la creación de la base de datos junto con sus tablas, objetos, usuarios y datos, entre otros.

Procedimiento

- 1- Abrir MySQLWorkbench y conectarse a la base de datos.
- 2- Abrir cada sql script desde la opción "File" -> "Open SQL Script" y por consiguiente, ejecutarlos según el orden numérico de cada archivo. sql.

Back Up

El backup de la base de datos se encuentra en la carpeta DB_BACKUP.

Pasos para su ejecución:

- 1- Abrir MySQLWorkbench y conectarse a la base de datos. Si no está configurada, crear una nueva.
- 2- Ir a la opción Server ->Data Import
- 3- Elegir como método de Importación: Import from Self-Contained File y seleccionar el archivo .sql de BackUp.
- 4- Elegir como opción de importación: Dump Structure and Dat y seleccionar el botón Start Import.

Herramientas Utilizadas

A lo largo del curso se han utilizado:

- MySQL Workbench como sistema de gestión de base de datos relacional.
- Zoom para asistir a las clases.
- Lenguaje SQL para la realización de las consultas y scripts.
- Mockaroo para la generación de datos aleatorios.
- Excel y Power BI para la generación del informe.