



Proyecto Electricidad Alsina “Base de datos”



Alejo Ariel
Giaimo

Llevamos la empresa
al siguiente nivel

CODERHOUSE



Sobre nosotros

Líderes en la distribución de materiales eléctricos e iluminación

Entre usuarios particulares, comercios de electricidad, industrias, empresas constructoras, empresas de instalación y montaje, grandes empresas, instituciones y organismos públicos con quienes trabajamos constantemente.

CODERHOUSE



Introducción



Una creciente necesidad de Mejorar la calidad de la base de datos.

En el trascurso de esta presentación, estaremos abordados diferentes puntos relacionados con la creación de la base de datos y sus respectivos componentes de las mismas, además estaremos tratando diferentes puntos como:

- Objetivos.
- La situación problemática.
- Las soluciones.
- El modelo de negocio.

Por otro lado, destacaré la importancia de implementar la base datos para la empresa Electricidad Alsina.

CODERHOUSE

ELECTRICIDAD
ALSINA

Objetivos



Objetivos Claros

En primer lugar, es fundamental comprender los objetivos que nos propusimos alcanzar con este proyecto, implementar una base datos relacional que pueda contener toda la información que dispone la empresa para la venta de insumos eléctricos.

CODERHOUSE

ELECTRICIDAD
ALSINA

El problema



Una necesidad de digitalizar los procesos que se realizan de forma manual en el comercio

Es importante comprender la situación problemática que motivó a Electricidad Alsina a querer modificar su estructura de datos, actualmente no cuentan con un sistema que esté desarrollado en la empresa, solamente lo hacen de forma manual, es por eso que se está llevando a cabo una serie de requerimientos para entender los flujos que componen la operatoria de la venta de insumos eléctricos, con el fin de poder implementar una base de datos que le de soporte a la operatoria de la empresa.

CODERHOUSE

ELECTRICIDAD
ALSINA

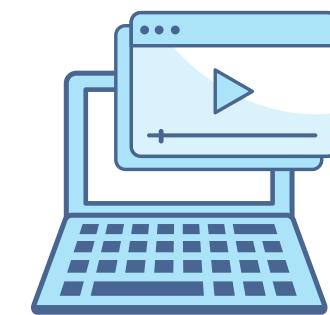


Las soluciones



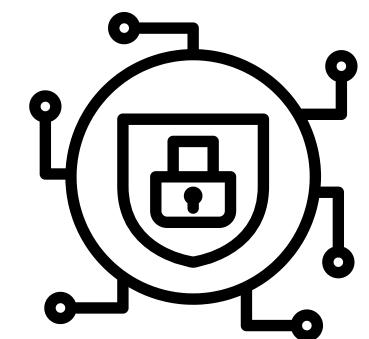
Optimizar la gestión de datos

Disponer de los datos de forma eficaz y eficiente.



Mejorar la eficiencia operativa

Implementar una base de datos que de soporte a la operativa de las ventas de la empresa.



Asegurar la integridad y seguridad de los datos

Mayor seguridad en los datos.

CODERHOUSE

ELECTRICIDAD
ALSINA



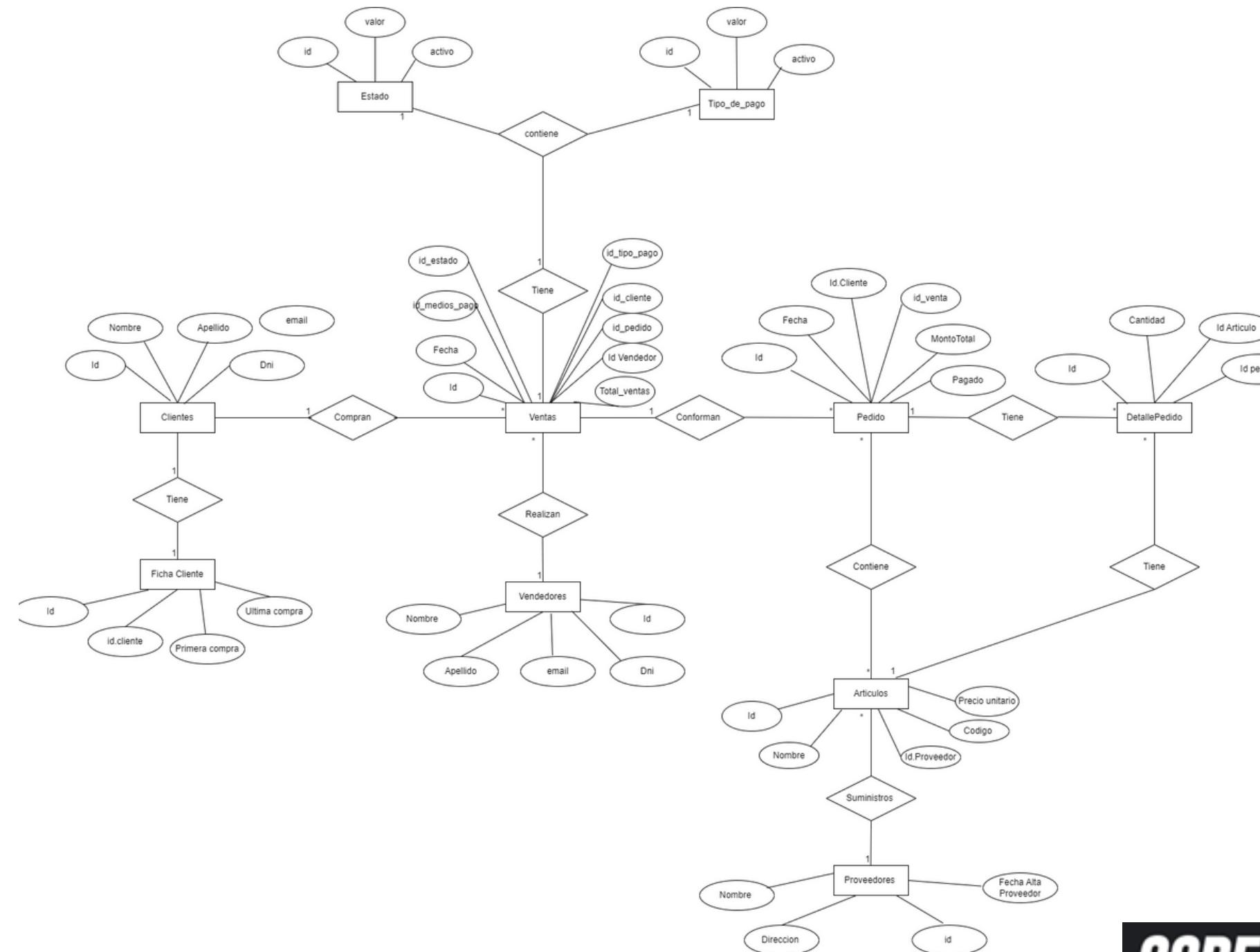
Modelo de Negocios

Un aspecto fundamental de nuestro proyecto es el modelo de negocio que hemos desarrollado para abordar la problemática de nuestro cliente, donde se ve afectado por el volumen de información que está manejando y no tiene un sustento digital que pueda almacenar dicha información.

El modelo se basa en Implementar una base de datos que de soporte a toda la operatoria de las ventas de la empresa.

Almacenar los datos de los clientes, proveedores, artículos y vendedores de la empresa, Por último poder desarrollar diferentes reportes que se pueda utilizar para la toma de decisiones.

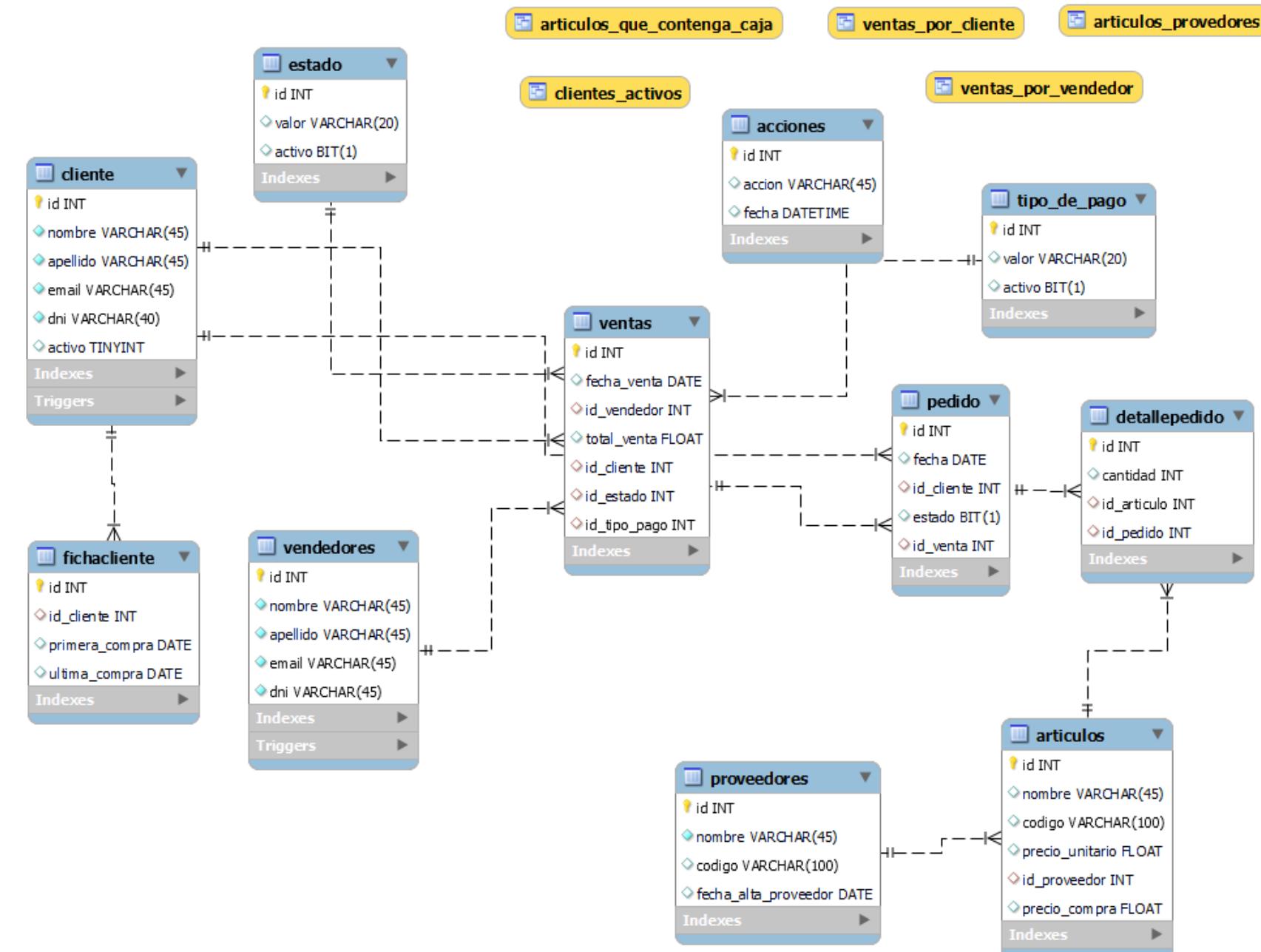
Diagrama de Entidad Relación



CODERHOUSE



Diagrama de Entidad Relación inversa



CODERHOUSE



*Ingresar al repositorio en gitlab del proyecto ProyectoDB-ElectricidadAlsina y buscar el archivo llamado EER para su mejor visualización

Data Base Electricidad Alsina

*Ingresar al enlace para visualizar el armado de la base de datos “Electricidad Alsina”

Crear las tablas de la Base de datos

Ingresar al repositorio del proyecto electricidad Alsina en GitLab y buscar el archivo “[ScriptCreateTableDBAlsina.sql](#)”, una vez encontrado el archivo descargar y ejecutar el script en workbench para la creación de las tablas



Inserción de datos



Secuencia para poder importar los datos a la base Electricidad Alsina.

En primer lugar, se deberá descargar desde el repositorio del proyecto que está alojado en Gitlab con el nombre [script “Script Insert Electricidad Alsina”](#), posteriormente se deberá ejecutar el script en Workbench

A screenshot of the MySQL Workbench interface. On the left, the 'SCHEMAS' tree shows two databases: 'alsina' and 'gammers_model'. The 'alsina' database is expanded, showing tables like 'articulos', 'cliente', 'detallepedido', 'estado', 'fichadiente', 'pedido', 'proveedores', 'tipo_de_pago', 'vendedores', and 'ventas'. The 'gammers_model' database is also expanded, showing tables 'class', 'comment', and 'commentary'. On the right, the main window displays the SQL editor with the following code:

```
1
2      #Script db electricidad alsina#
3      #Tabla Estado#
4      INSERT INTO estado( valor,activo) VALUES
5      ('Rechazado',1),
6      ('EnProceso',1),
7      ('Pagado',1);
8      ##Select * from estado;
9
10     #tabla tipo de pago#
11    INSERT INTO tipo_de_pago( valor,activo) VALUES
12    ('Efectivo',1),
13    ('Tarjeta',1),
14    ('Cheque',1);
15    ##select * from tipo_de_pago;
16
```

Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

Luego ejecutar los archivos CVS compartidos en el repositorio del proyecto en Gitlab

<https://github.com/agteam32/ProyectoDB-ElectricidadAlsina/blob/main/Script-insert-Electricidad-alsina.sql>

Ejecutar los archivos de la siguiente manera de forma consecutiva, como se manifestó en este documento:

- Tabla Articulos
- Tabla Clientes
- Tabla FichaCliente
- Tabla Venta
- Tabla Pedido
- Tabla DetallePedido

Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

Para poder importar los datos de los archivos y que se inserten en las tablas detalladas anteriormente se deberá ejecutar un select * from de la tabla, luego hacer clic en el icono importar archivo

The screenshot shows the MySQL Workbench interface. On the left, there's a tree view of schemas and tables, with 'Schemas' selected. In the center, a query editor window displays the following SQL code:

```
15
16 *  select * from articulos
17 *  select * from cliente;
18 *  select * from estado
19 *  select * from fichacliente
20  select * from ventas
21  select * from pedido
22  select * from tipo_de_pago
23  select * from detallepedido;
24 *  select * from proveedores;
25 *  select * from vendedores;
```

Below the query editor is a result grid titled 'Result Grid'. It has columns: id, nombre, codigo, predio_unitario, id_proveedor, and predio_compra. There are two rows of data:

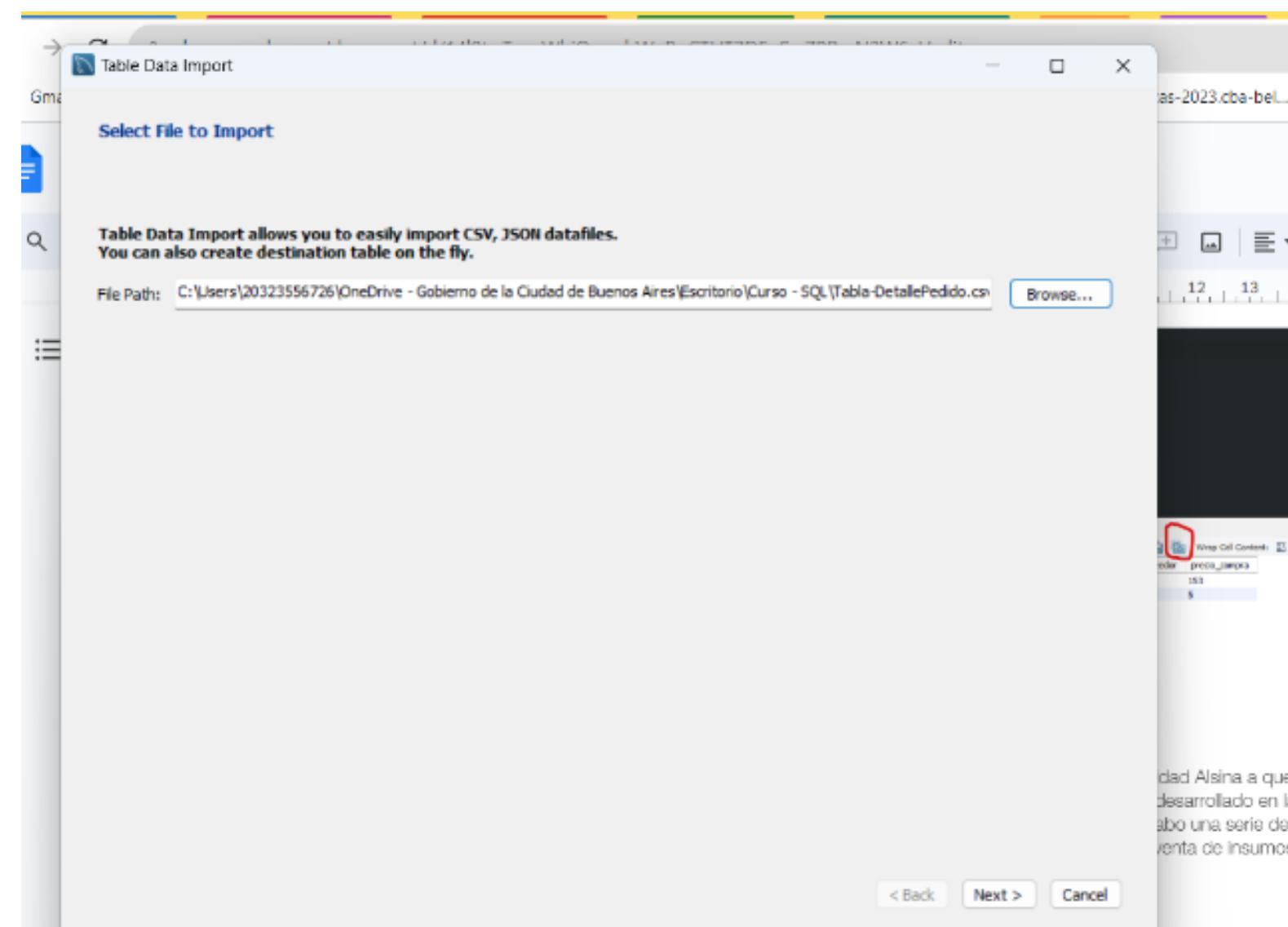
	id	nombre	codigo	predio_unitario	id_proveedor	predio_compra
▶	1	Cinta tela blanca 9.14 mts. ajec	000310210M	110027	3	153
	2	Cinta aisladora esp 0.015mm 10m negra	0003208-1510	1002.25	3	5

A red circle highlights the 'Import' button in the toolbar above the result grid.

Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

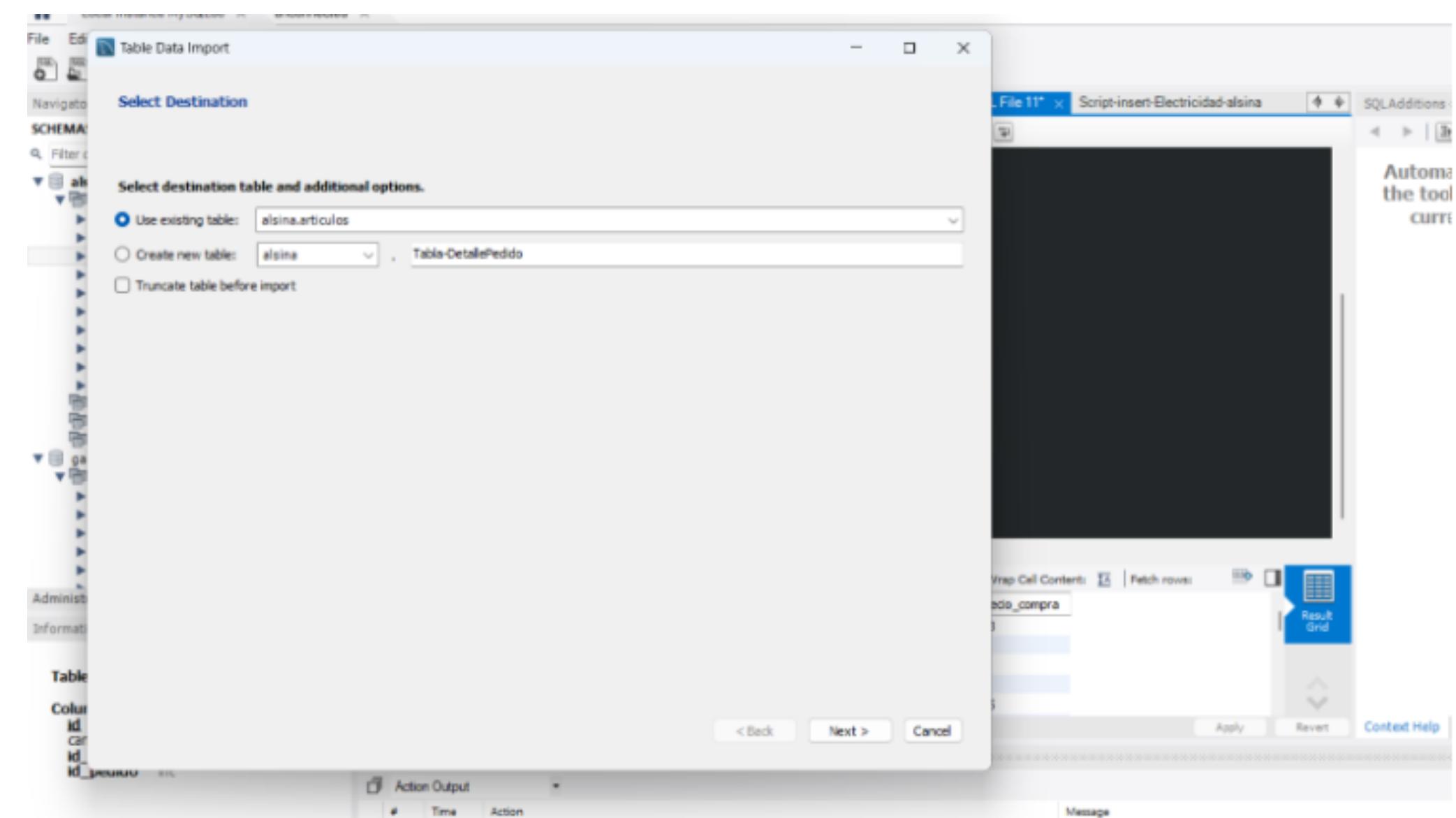
Seleccionar la ubicacion donde esta guardado los archivos que contiene la informacion de las tablas, apretar en el boton Next.



Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

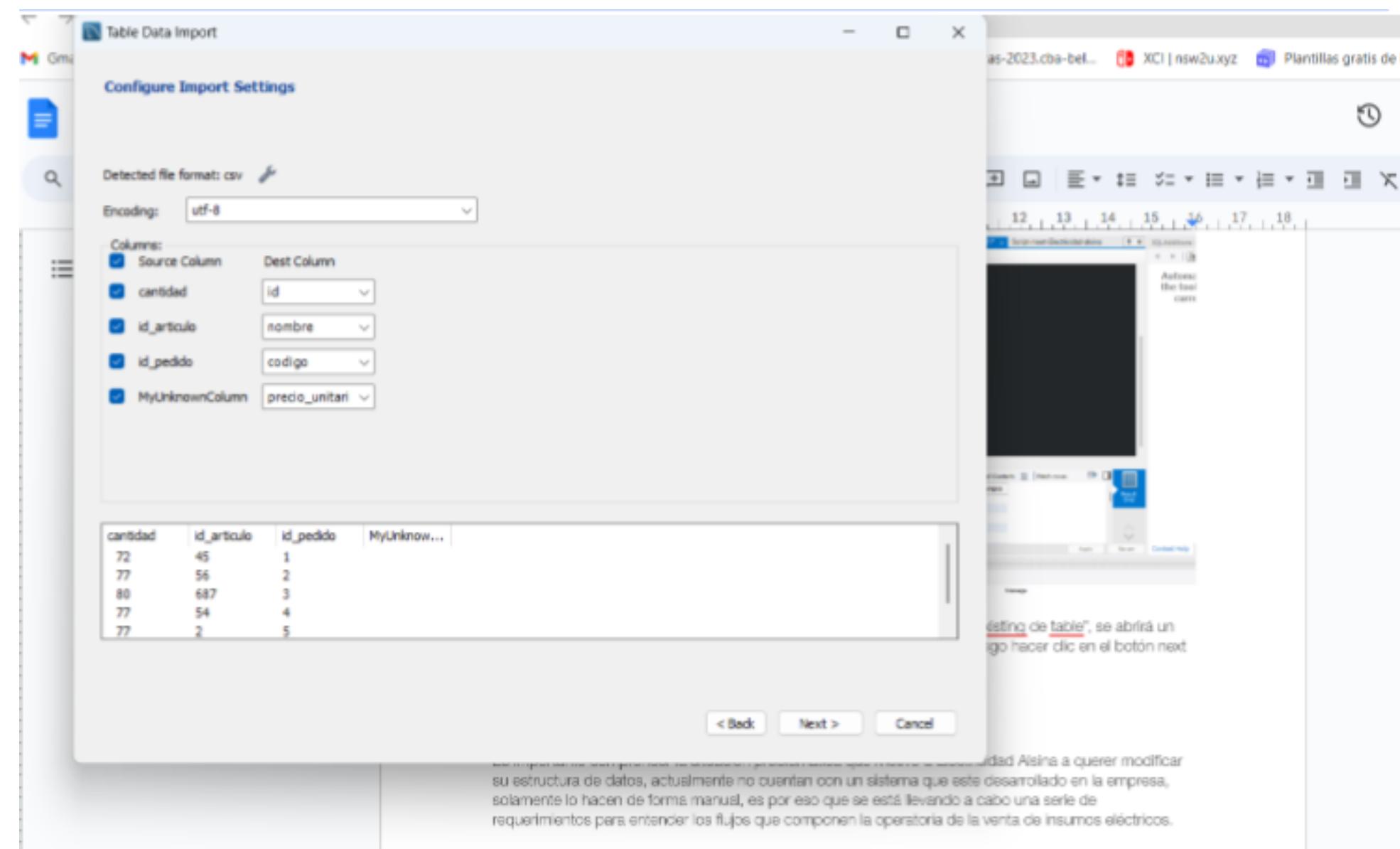
Seleccionar el destino, en este caso debemos seleccionar la opción de “existing de table”, luego hacer clic en el botón Next



Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

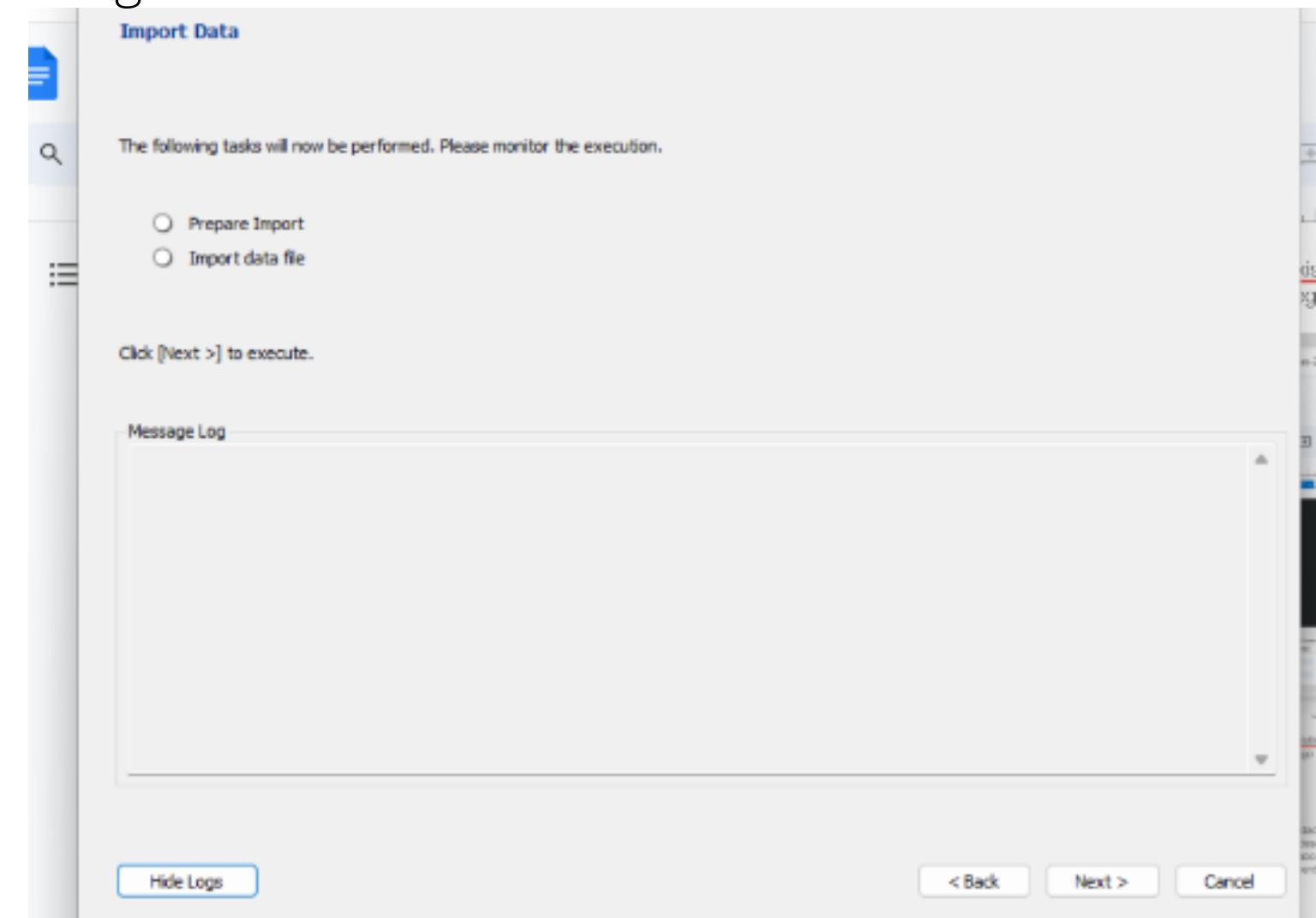
Se abrirá un dropdown con las tablas disponibles, seleccionar la tabla en cuestión y luego hacer clic en el botón next



Inserción de datos

Secuencia para poder importar los datos a la base Electricidad Alsina.

Por último hacer clic en el botón next, para que comience el proceso de importación de los datos, una vez finalizado la importación en el apartado de logs podremos visualizar el total de los archivos importados, si llega a encontrar algún error nos mostrará en la visualización de los logs.



Vistas Electricidad alsina

Vistas creadas para poder ver información parcial de la base Electricidad Alsina.

Ingresar al Gitlab del proyecto Electricidad Alsina y descargar el script [viewDbAlsina](#) para su ejecucion.

Vista 1- Ventas_por_vendedor

Descripción: visualizar las ventas por vendedor

Objetivo Esta vista nos permite ver las ventas totales por vendedor

Tablas/datos: Vendedores/ventas

```
CREATE VIEW Ventas_por_Vendedor AS  
  (SELECT nombre, apellido,v.id,v.fecha_venta,v.total_venta  
   FROM vendedores ve  
   JOIN ventas v  
   ON ve.id= v.id_vendedor);  
  SELECT * FROM Ventas_por_Vendedor ;
```

Vista 2 Clientes_activos

Descripción: visualizar clientes activos

Objetivo Esta vista nos permite ver todos los clientes que esten activos

Tablas/datos: Vendedores/ventas

```
CREATE VIEW clientes_activos AS  
  (SELECT nombre,apellido,activo  
   FROM cliente  
   WHERE activo=1);  
  SELECT * FROM clientes_activos;
```

Vista 3 Articulos_que_contenga_caja

Descripción: Artículos que contenga la palabra caja

Objetivo Visualizar los artículos que contengan la palabra caja

```
CREATE VIEW articulos_que_contenga_caja AS  
  (SELECT nombre, codigo,precio_compra  
   FROM articulos  
   WHERE nombre like '%CAJA%');  
  SELECT * FROM articulos que contenga caja;
```

Vista 4 Articulos_Proveedores

Descripción: Artículos discriminados por proveedores

Objetivo Visualizar los artículos separados por proveedores

Tablas/datos: artículos/proveedores

```
CREATE VIEW articulos_proveedores AS  
  (SELECT a.nombre,a.precio_compra,p.nombre as proveedor  
   FROM articulos a  
   JOIN proveedores p  
   ON a.id= p.id);  
  SELECT * FROM articulos_proveedores;
```

Vista 5 Ventas_por_cliente

Descripción: ventas discriminadas por cliente

Objetivo Visualizar las ventas totalizadas y agrupadas por cliente

Tablas/datos: ventas/clientes

```
(SELECT c.nombre, c.apellido,v.total_venta  
  FROM cliente c  
  JOIN ventas v  
  ON c.id = v.id_cliente  
  GROUP BY c.nombre, c.apellido,v.total_venta  
  ORDER BY total_venta);
```

Funciones Electricidad alsina

Funciones para poder gestionar operaciones concretas que reciben un parametro y a su vez retornan un resultado específico

Ingresar al Gitlab del proyecto Electricidad Alsina y descargar el script [funciones-giaimoalejo](#) para su ejecucion.

Función 1- Precio de venta

Descripción: Calcula el precio de venta de un producto con el margen deseado.

Objetivo: Calcular el precio de venta de un producto.

```
#Funcion para sacar el precio de venta de un producto#
DELIMITER $$ 
CREATE FUNCTION `calcular_precio_de_venta` (costo_producto INT, margen float) RETURNS float
    NO SQL
BEGIN
    DECLARE resultado FLOAT;
    DECLARE pv int;
    SET pv = 1;
    SET resultado = costo_producto / (pv-margen);
    RETURN resultado;
END
$$

SELECT calcular_precio_de_venta(50,0.30) AS Precio_de_venta;
```

Función 2- Artículos por rango de precios

Descripción: Agrupar los artículos por rango de precios en tres grandes grupos.

Objetivo: Discriminar los artículos por rango de precios y separar en tres grupos definidos en la función.

```
20 #Funcion que nos permite describir por rango de precios de los articulos#
21 delimiter //
22 • create function A_por_rango_precio(
23     precio_unitario float)
24     returns varchar(30)
25     deterministic
26 begin
27     case
28         when precio_unitario<3000 then
29             return 'producto precio bajo';
30         when precio_unitario>=3000 and precio_unitario<5000 then
31             return 'Producto precio medio';
32         when precio_unitario>=5000 then
33             return 'Producto precio alto';
34     end case;
35 end //
36 delimiter ;
```

Stored procedure Electricidad alsina

Stored procedure ejecutan un comando de instrucciones entre sí que afectan la información con el fin de lograr un objetivo dentro del sistema

Ingresar al Gitlab del proyecto Electricidad Alsina y descargar el script sp-giaimoalejo para su ejecución.

Stored procedure que nos permite Mostar todas las ventas de un cierto monto que pasamos por parámetro.

```
91  DELIMITER $$  
92 * CREATE PROCEDURE `Sp_ventas` (  
93     IN v_ventas int (255))  
94 BEGIN  
95     select fecha_venta,id_vendedor,total_venta,id_cliente  
96         from ventas  
97         where total_venta>= v_ventas;  
98 END  
99 $$
```

Stored procedure que nos permite poder insertar un cliente que realizo una primera compra y no esta en la base de datos de electricidad alsina

```
MITER //  
CREATE PROCEDURE sp_add_CLIENTE(IN CLIENTE_NOMBRE VARCHAR(45), IN CLIENTE_APELLIDO VARCHAR(45), CLIENTE_EMAIL VARCHAR(45)  
IN  
DECLARE CLIENTE INT DEFAULT 0;  
DECLARE EXIT HANDLER FOR SQLEXCEPTION, NOT FOUND, SQLWARNING  
BEGIN  
    ROLLBACK;  
    SELECT 'Error';  
END;  
START TRANSACTION;
```

Trigger Electricidad alsina

Trigger es una funcionalidad o un disparador que se ejecuta de forma automatica en la base de datos al momento de realizar un insert, update o delete.

Ingresar al Gitlab del [proyecto Electricidad Alsina](#) y descargar el [script trigger-giaimoalejo](#) para su ejecución.

- Antes de ejecutar el script de los trigger , se deberá crear la tabla “Acciones”, esta misma va a contener los logs de los trigger.

Trigger para cuando se inserta un cliente nuevo a la base de datos.

```
delimiter //
create trigger tr_log_tabla_cliente after insert on cliente
for each row
begin
  insert into acciones(accion) value ('Se creo un registro en cliente');
end//
delimiter ;
```

Trigger para cuando se inserta un vendedor nuevo a la base de datos.

```
#trigger nos da un log de cada insert que haya en la tabla vendedores#
delimiter //
CREATE TRIGGER tr_alerta_nuevo_vendedor
BEFORE INSERT
ON vendedores
FOR EACH ROW
BEGIN
  INSERT INTO acciones(accion) VALUE ('Se creo un registro en vendedores');
END;
```

Informes Electricidad Alsina

Informes para Electricidad Alsina

- El Primer informe nos muestra las ventas de un mes determinado.

```
Select fecha_venta as Fecha, total_venta as Total from venta  
where fecha_venta BETWEEN '2021-05-01' and '2021-05-31'  
order by fecha_venta asc;
```

1-Ventas Por Dia Mes mayo Año 2021	
2021-05-16	\$92,069.00
2021-05-16	\$97,378.00
2021-05-17	\$84,981.00
2021-05-18	\$44,105.00
2021-05-18	\$55,854.00
2021-05-18	\$9,217.00
2021-05-25	\$52,834.00
2021-05-26	\$51,550.00
2021-05-28	\$80,828.00
2021-05-29	\$23,801.00

- El segundo informe nos muestra el agrupado de ventas por fecha y a su vez nos totaliza la venta del mes.

```
#2-Agrupar las ventas por dia y sumarlas#  
Select fecha_venta ,sum(total_venta) as Total from ventas  
where fecha_venta BETWEEN '2021-05-01' and '2021-05-31'  
group by fecha_venta with rollup;
```

2-Ventas agrupadas por dias Mes Mayo año 2021 Total	
2021-05-16	\$189,447.00
2021-05-17	\$84,981.00
2021-05-18	\$109,176.00
2021-05-25	\$52,834.00
2021-05-26	\$51,550.00
2021-05-28	\$80,828.00
2021-05-29	\$23,801.00
Total	\$592,617.00



Informes Electricidad alsina

Informes para Electricidad Alsina

- El tercer informe nos muestra el promedio de ventas de un mes

```
#3-Promedio de ventas de un mes específico #
Select avg(total_venta) as promedio_venta from ventas
where fecha_venta BETWEEN '2021-05-01'and '2021-05-31';
```

3-Promedio de ventas de mes mayo año 2021

\$59,261.70

- El cuarto informe nos muestra el agrupado de ventas por fecha y a su vez nos totaliza la venta del mes.

```
#4 Ventas discriminadas por vendedor#
Select u.fecha_venta as fecha , u.total_venta as total, v.nombre as Nombre_Vendedor,v.apellido as Apellido_Vendedor
from ventas u
left join vendedores v
on u.id_vendedor = v.id
where fecha_venta
BETWEEN '2021-05-01'and '2021-05-31'
order by nombre asc;
```

4-Ventas Discriminadas Por Vendedor

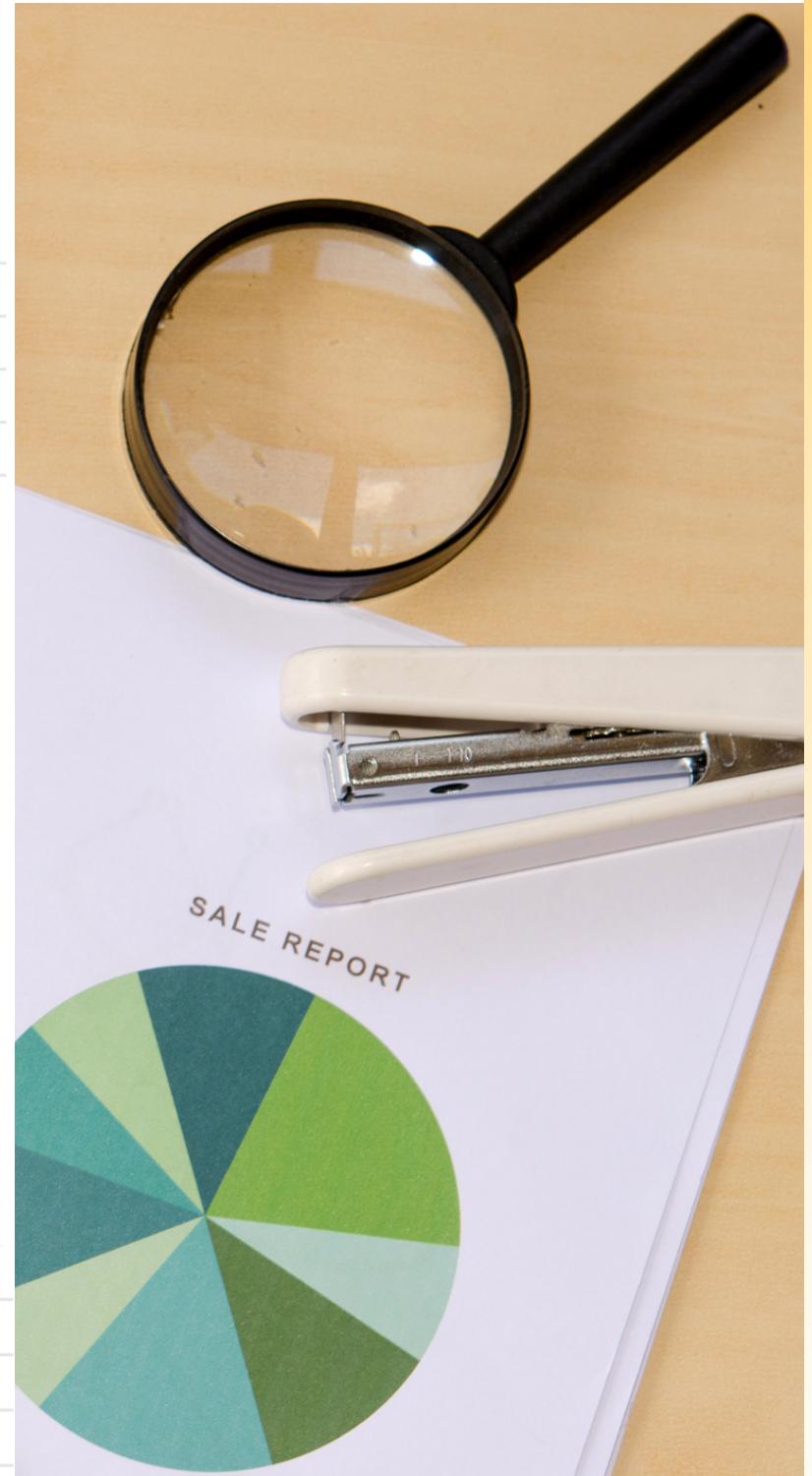
2021-05-16	\$92,069.00	Alberto	Nuñez
2021-05-28	\$80,828.00	Alberto	Nuñez
2021-05-16	\$97,378.00	Carla	Ramos
2021-05-18	\$55,854.00	Carla	Ramos
2021-05-17	\$84,981.00	Florencia	Arletaz
2021-05-25	\$52,834.00	Florencia	Arletaz
2021-05-18	\$44,105.00	Florencia	Arletaz
2021-05-26	\$51,550.00	Maria	Paz
2021-05-29	\$23,801.00	Maria	Paz
2021-05-18	\$9,217.00	Rodrigo	Forclaz

- El quinto informe realiza un conteo de las ventas y las agrupa por vendedor y totaliza el total de ventas de un mes determinado.

```
#5conteo de las ventas de un mes y agrupado por vendedor#
select vendedores, count(*) as VentasMes
from
(SELECT v.nombre as Vendedores,u.total_venta
from ventas u
left join vendedores v
on u.id_vendedor = v.id
where month (u.fecha_venta)=5 and year (u.fecha_venta)='2021') as mytable
group by vendedores with rollup;
```

5-Cantidad de ventas por vendedor

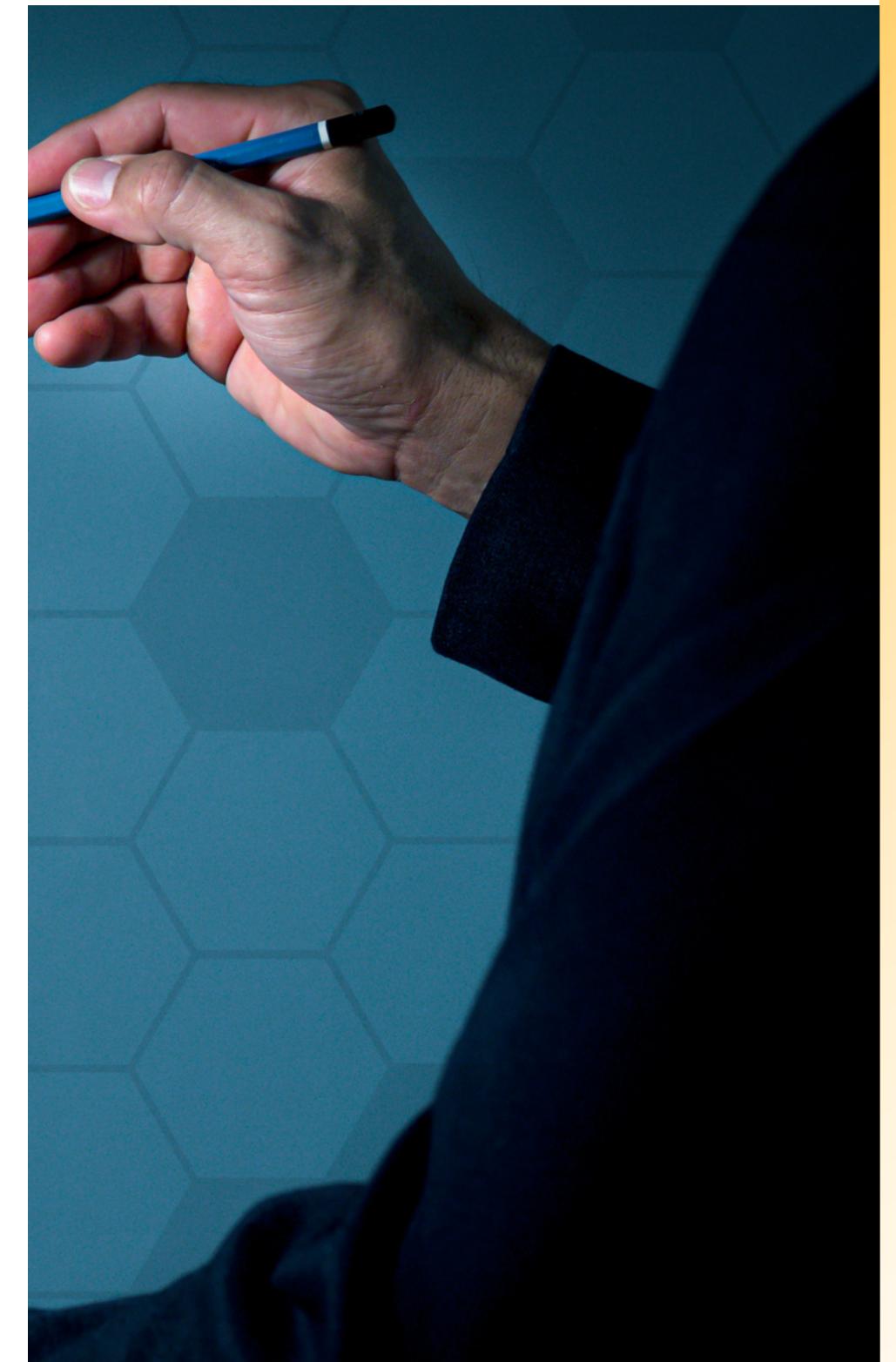
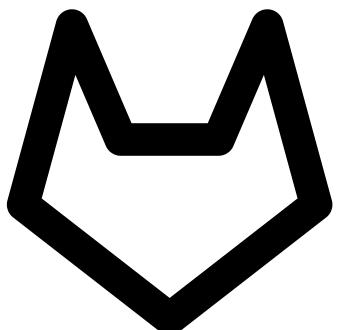
Alberto	2
Carla	2
Florencia	3
Maria	2
Rodrigo	1
Total	10



Herramientas utilizada en el proyecto Electricidad alsina



- My Sql Workbench se utilizo para el armado de la base de datos del proyecto Electricidad Alsina
- Diagrams.net se utilizó para realizar el diagrama de relación de entidades del proyecto Electricidad Alsina.
- Gitlab fue la herramienta para almacenar todo el repositorio del proyecto Electricidad Alsina.
- Canva fue la herramienta para diseñar y documentar el proyecto Electricidad Alsina.



Lineas Finales

En Conclusión:

Al llegar al final de este proyecto, podemos mirar hacia atrás con orgullo y satisfacción por los logros alcanzados. Cada paso, cada desafío, y cada esfuerzo ha contribuido a la realización de un trabajo excepcional. Este proyecto no solo representa un hito en mi camino profesional, sino también un testimonio de nuestra dedicación y colaboración.

Quisiera expresar mi sincero agradecimiento a todos los involucrados: desde el equipo de trabajo de corder como a la profesora. Sin su apoyo inquebrantable, no habría llegado tan lejos ni alcanzado tales alturas. Cada aporte ha sido invaluable y ha dejado una marca indeleble en este proyecto.

Agradezco sinceramente a todos los que han contribuido a este proyecto y espero que sea de su agrado.

Gracias nuevamente a todos.

Atentamente,

Giaimo Alejo Ariel

Actualmente Analista Funcional.