

**UNIVERSIDAD RAFAEL LANDÍVAR DE
GUATEMALA**

**FACULTAD DE INGENIERIA EN
INFORMATICA Y SISTEMAS**

CURSO: ESTRUCTURA DE DATOS I
ING. PABLO GODOY



**Universidad
Rafael Landívar**
Tradición Jesuita en Guatemala

MANUAL DE USUARIO PROYECTO “MICROSQL”

INTEGRANTES

SEBASTIAN ORANTES CHANG	-	1085816
BRYAN MACARIO CORONADO	-	1283816

GUATEMALA, 05 DE MAYO DEL 2017

ÍNDICE

Introducción	3
Funcionalidad del software	
• Interfaz gráfica	4
• Carga diccionario	5
Funcionamiento interno	
• Creación de una nueva tabla	6
• Inserción de datos	7
• Selección de datos	8
• Eliminación de datos	9
• Eliminación de tablas	10
Conclusiones	11

MICRO SQL

El proyecto consistió en desarrollar un manejador de base de datos (DBMS) que recibe el nombre de “micro SQL”, y tiene como características principales:

- Gestionar estructuras de datos de forma personalizada.
- Permitir que el manejador se pueda acoplar a varios idiomas distintos del inglés.

Crear un programa que pueda atender las funcionalidades básicas y requeridas de un manejador de base de datos, además de poseer una interfaz gráfica amigable para que la experiencia del usuario sea agradable, para ello se utilizará el lenguaje de programación C# desarrollando el proyecto en el entorno de Visual Studio, además de aplicar las estructuras de datos desarrolladas durante el curso (árboles), para obtener como producto final un instalador del manejador “Micro SQL”.

Para el desarrollo de la aplicación, se utilizó el lenguaje de programación C#, porque es un lenguaje con el cual el desarrollador está familiarizado y además se utilizó un dll que se ha estado desarrollando a lo largo del curso, este posee diferentes estructuras de datos realizados en C#. Por otro lado se escogió C# por su paradigma que es POO y permite realizar una abstracción más fácil del problema.

FUNCIONALIDAD DEL SOFTWARE

1. Interfaz gráfica

El programa deberá de mostrar una interfaz gráfica con tres secciones:

1. Una primera sección donde se puede introducir código SQL para manipular los objetos de la base de datos.
2. La segunda sección incluye un “grid” donde pueda visualizarse el resultado de las instrucciones de recuperación de datos ejecutadas
3. La tercera sección tendrá la función de visualizar de forma atractiva el listado de tablas creadas y sus respectivas columnas.



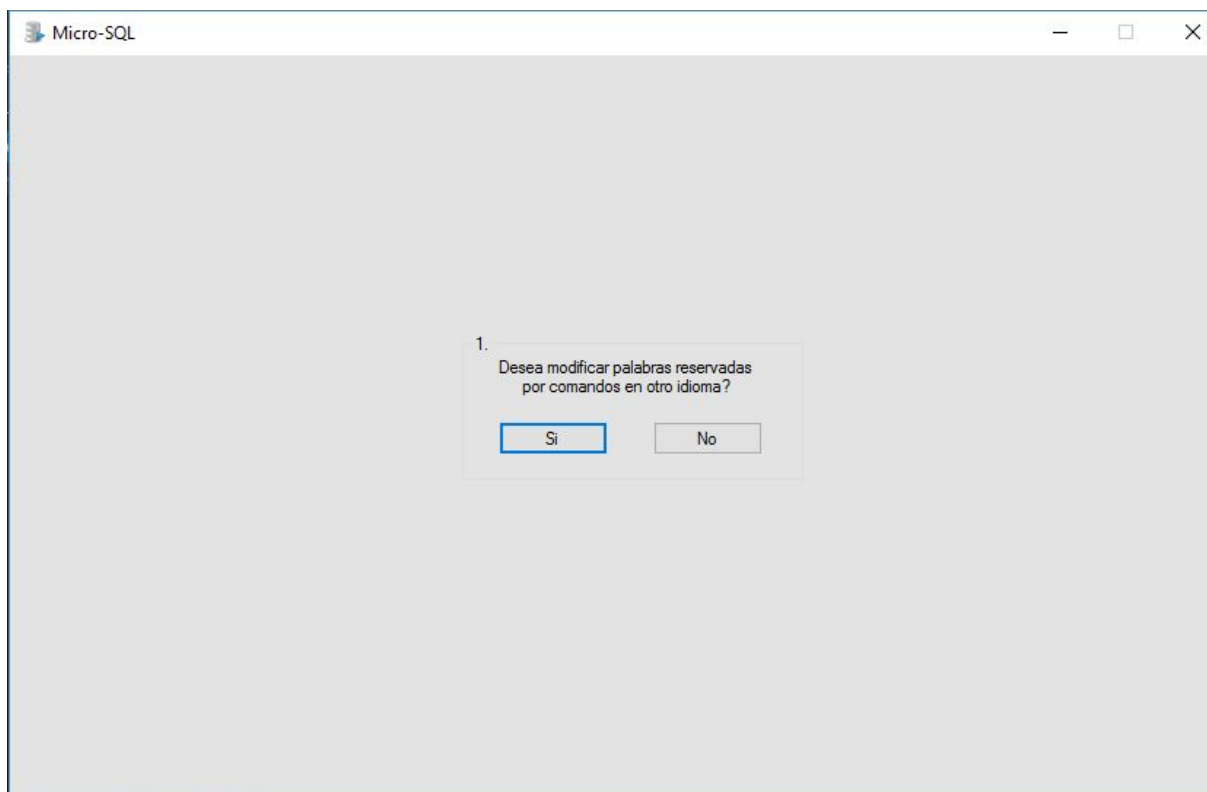
Carga de diccionario

Debido a que el programa debe aceptar comandos en varios idiomas, se pueden sustituir las palabras reservadas del lenguaje SQL, por lo que al iniciar el programa se puede cargar un diccionario de palabras reservadas desde el archivo C:/microSQL/microSQL.ini. Dicho archivo consiste en un archivo CSV que contiene por cada línea una pareja de palabras de la siguiente forma:

<Palabra Reservada>, <Comando en otro idioma>

El usuario deberá tener la oportunidad de personalizar este diccionario, puede hacerlo mediante la sustitución del archivo por uno nuevo que contenga la nueva definición. Si no existiera dicho archivo, el sistema genera uno nuevo colocando los valores por defecto.

Así mismo la configuración de la aplicación también debe permitir la carga de una nueva definición del diccionario desde la interfaz gráfica, así como una opción que permita restablecer a los valores predeterminados.

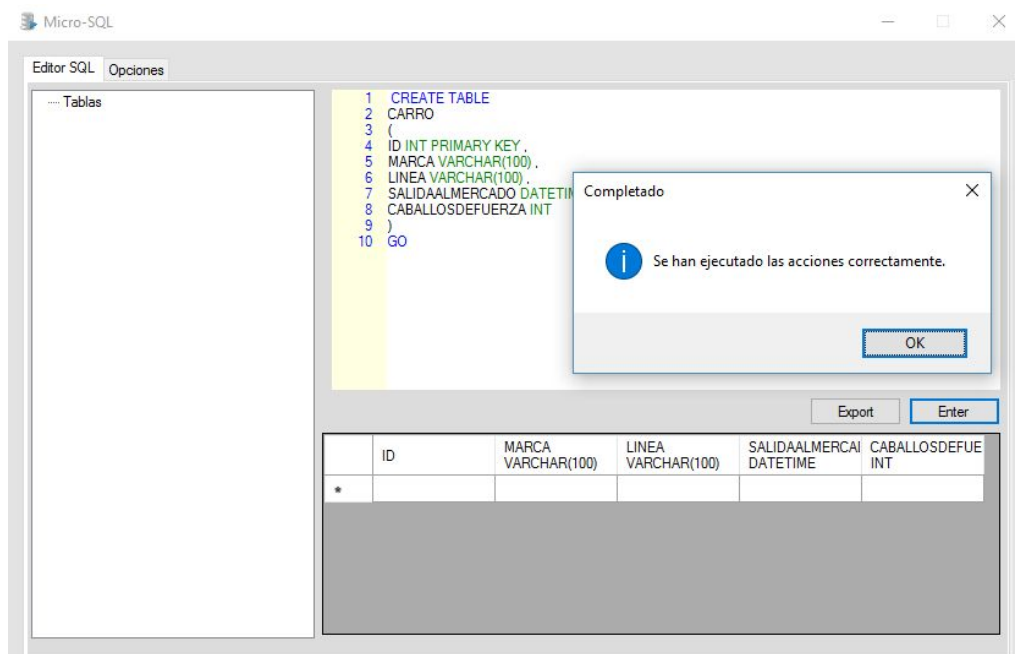
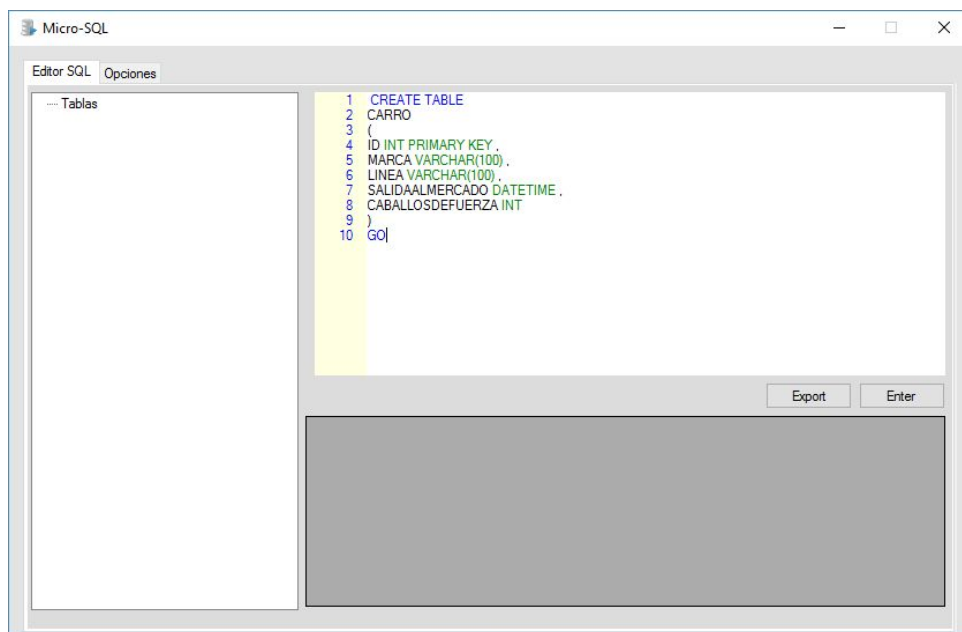


Funcionamiento interno

- **Creación de nueva tabla:**

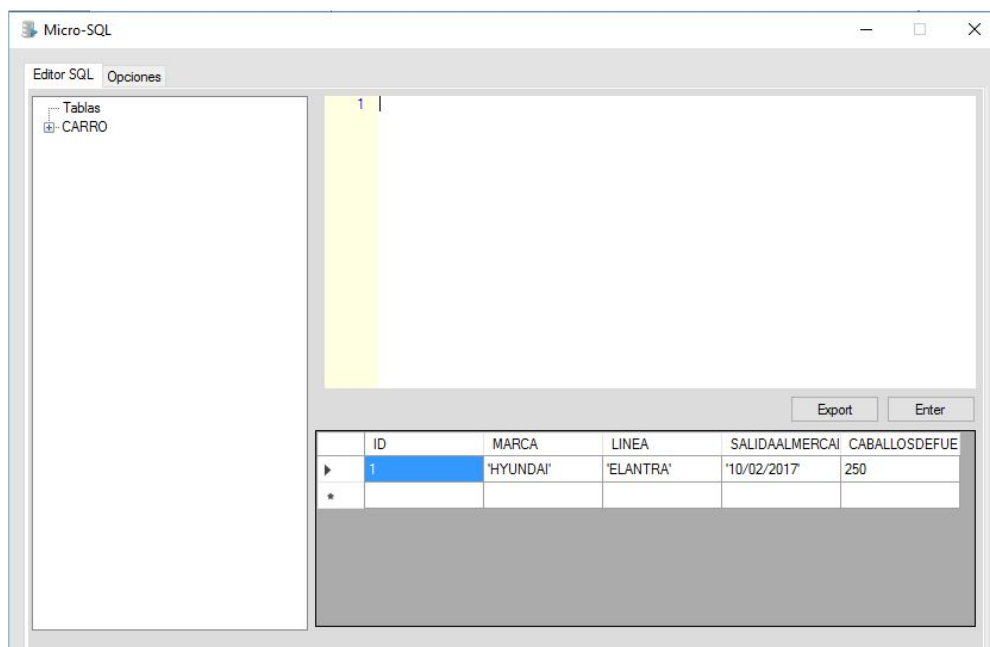
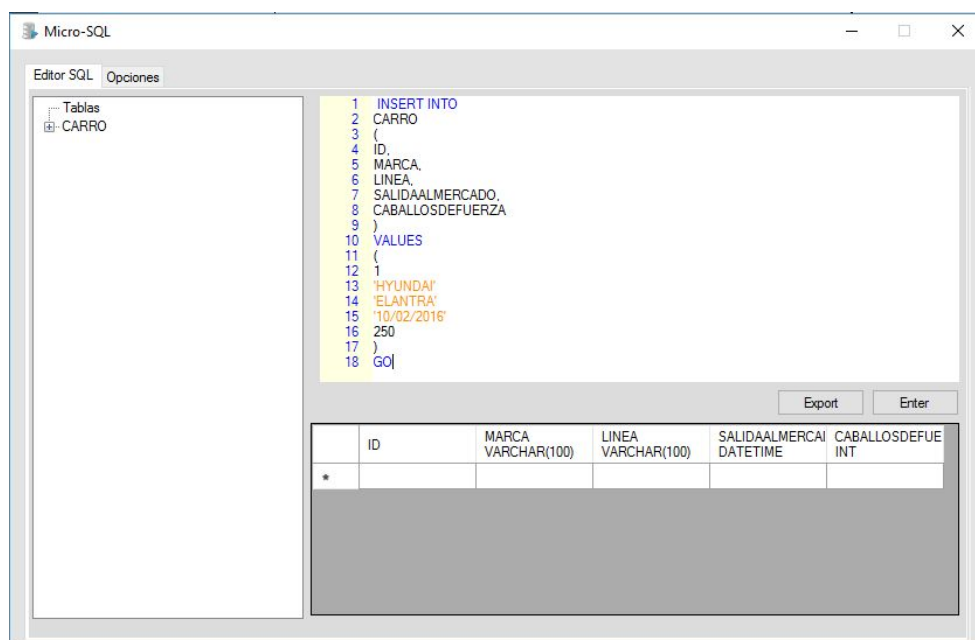
Permite la creación de una nueva tabla en la base de datos (internamente un árbol B).

- ❖ Los nombres de las columnas no pueden contener espacios.
- ❖ Los tipos de columnas admitidas son: INT, VARCHAR(100) y DATETIME.
- ❖ El máximo número de columnas por cada tipo de dato es 3.
- ❖ Se deberá agregar obligatoriamente una columna de nombre ID de tipo INT que actuará como llave primaria.



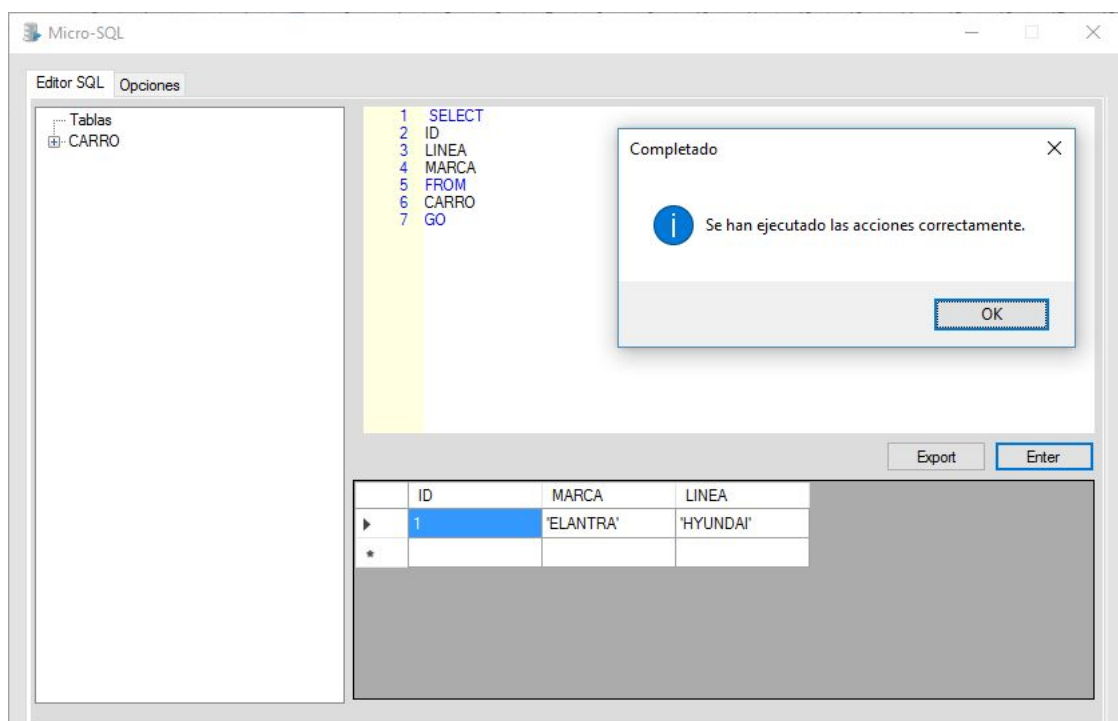
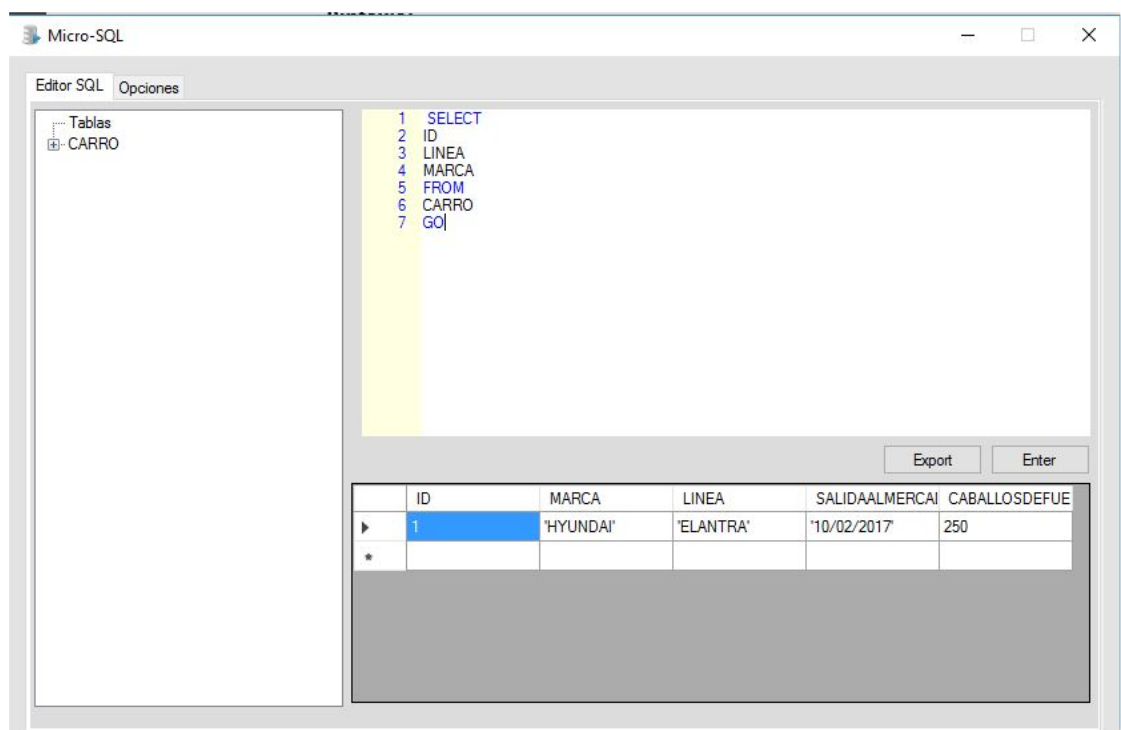
- **Inserción de datos:**

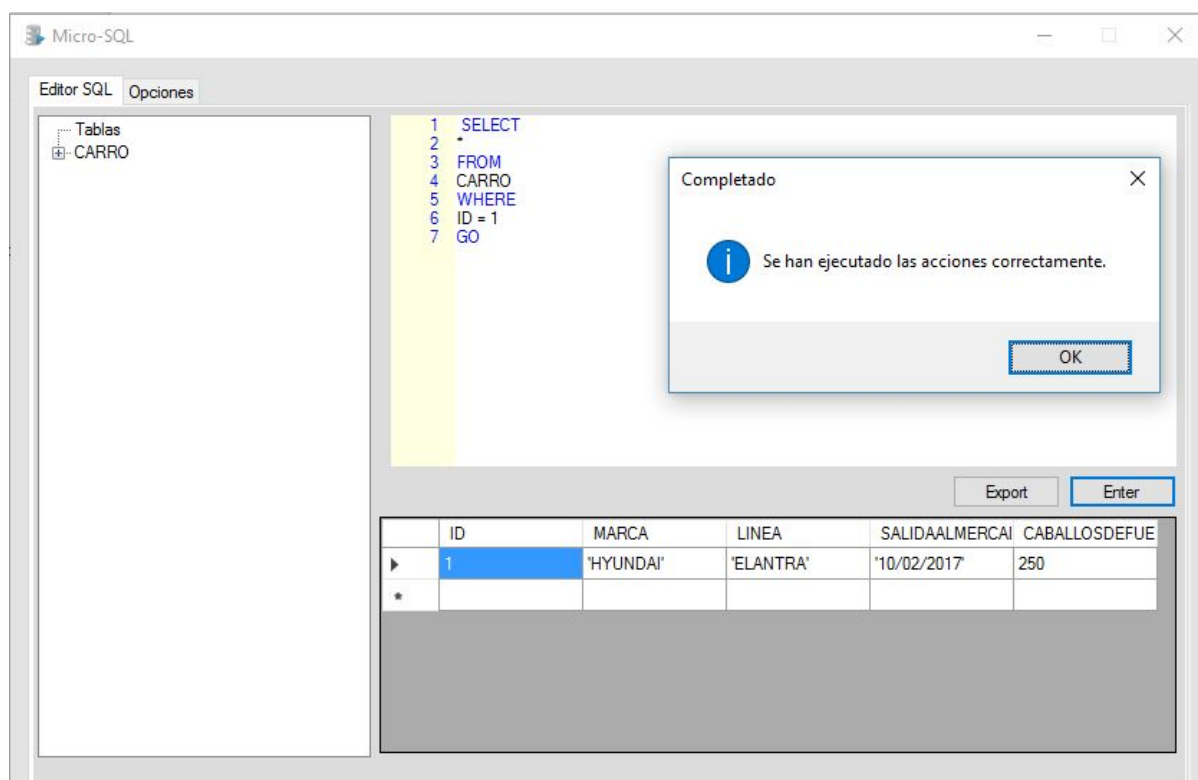
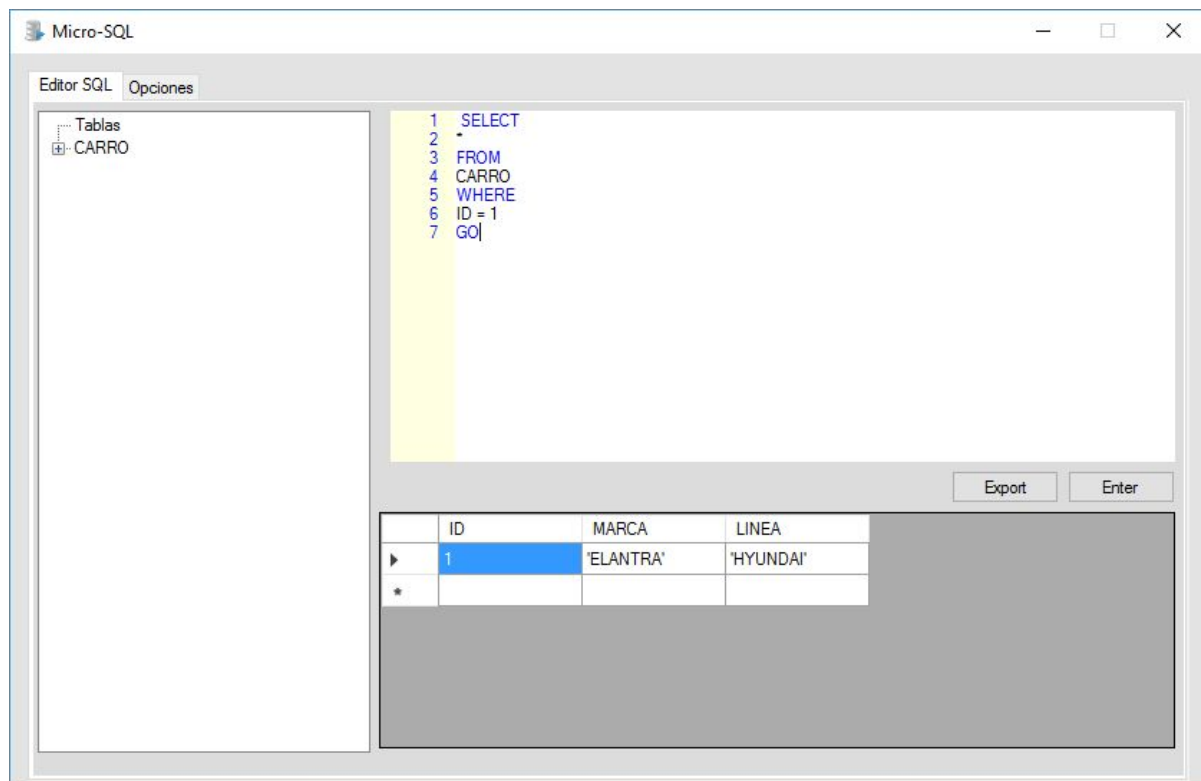
- ❖ Permite el ingreso de datos a una tabla específica de la base de datos.
- ❖ Se valida que la sintaxis cumpla con todas las reglas establecidas, en caso no se cumpla se genera error de sintaxis.
- ❖ Las columnas/datos para la inserción siempre se proporcionarán en el mismo orden en que fueron indicados en la creación de la tabla.
- ❖ Se proporcionarán valores para todos los campos ya que no se permiten campos “Nulos”.
- ❖ Los valores de campos VARCHAR deberán delimitarse por medio del símbolo ‘(apóstrofo o comilla simple)’.



- **Selección de datos:**

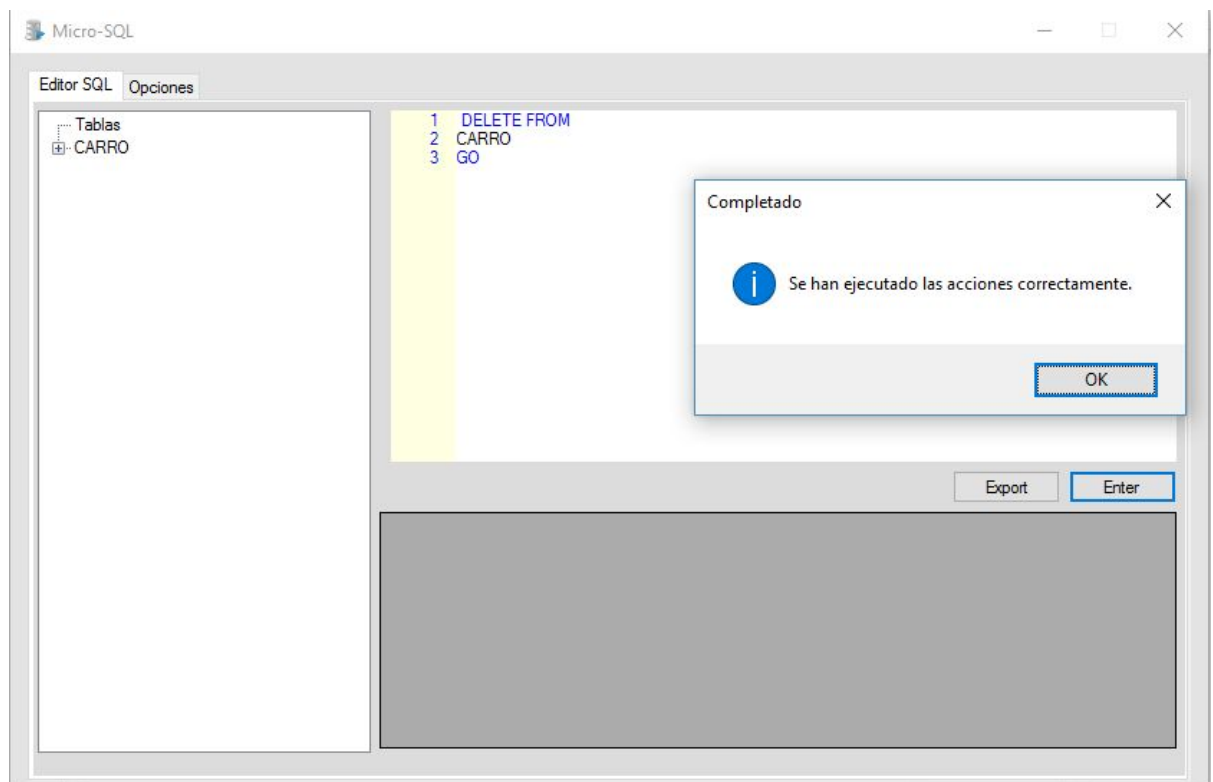
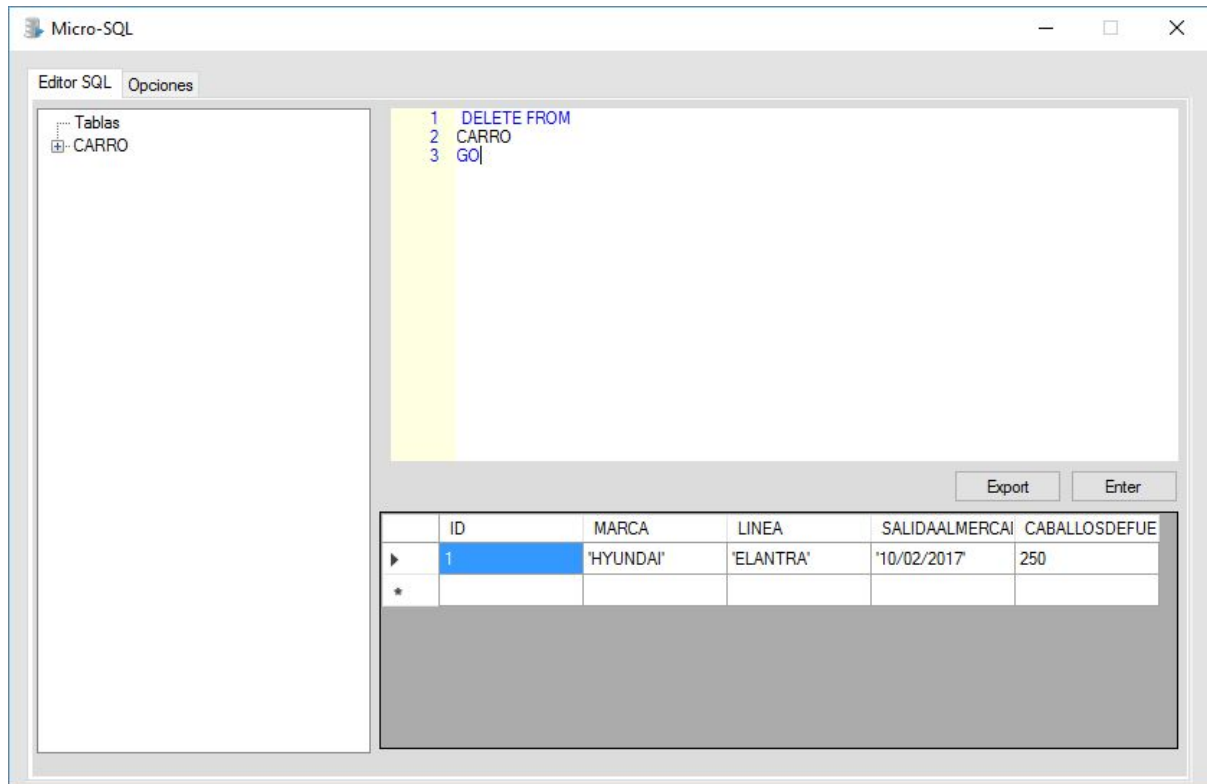
- ❖ Permite la extracción de datos de la Base de Datos, para una tabla específica.
- ❖ Los datos extraídos se muestran en un “grid”.
- ❖ El contenido del “grid” se deben poder exportar a un archivo CSV escogido por el usuario.
- ❖ Se valida que la sintaxis cumpla con todas las reglas establecidas, de lo contrario se muestra un error en pantalla.
- ❖ Si los campos indicados en la consulta no existen se muestra un error en pantalla indicando cuales son los campos inexistentes en la tabla.





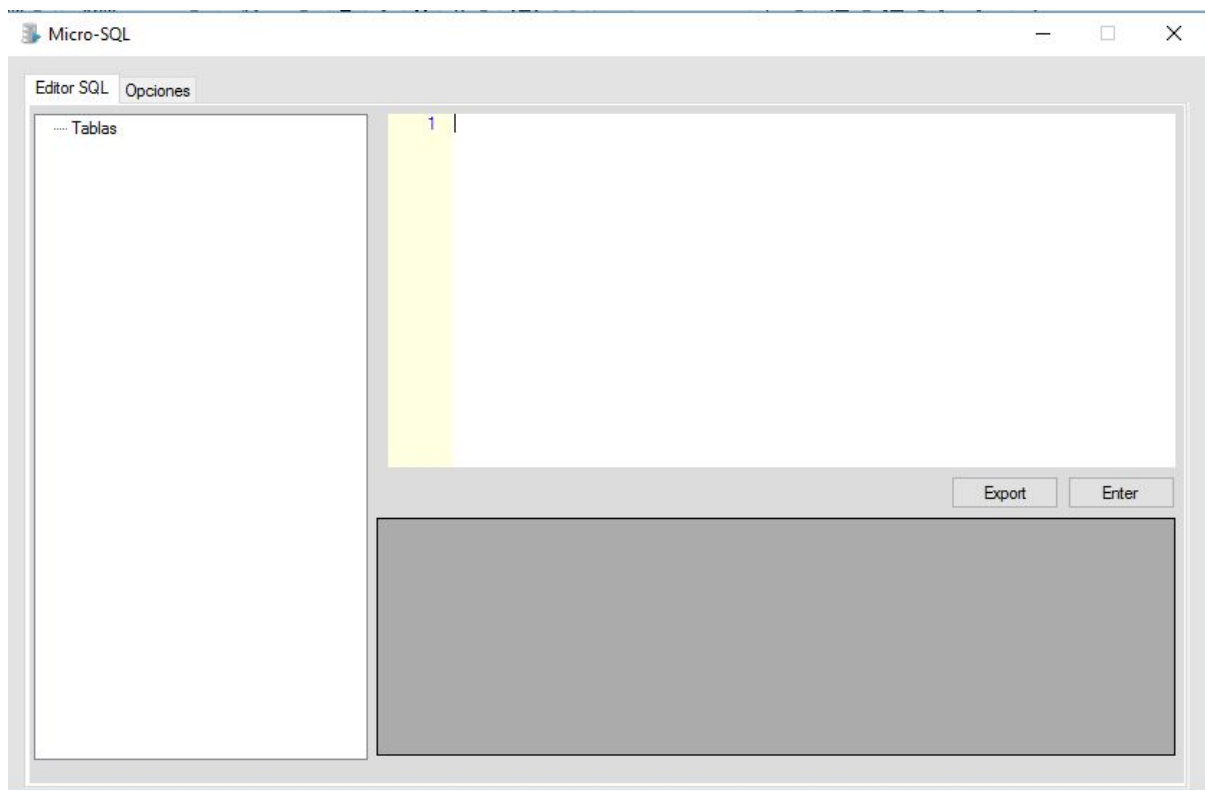
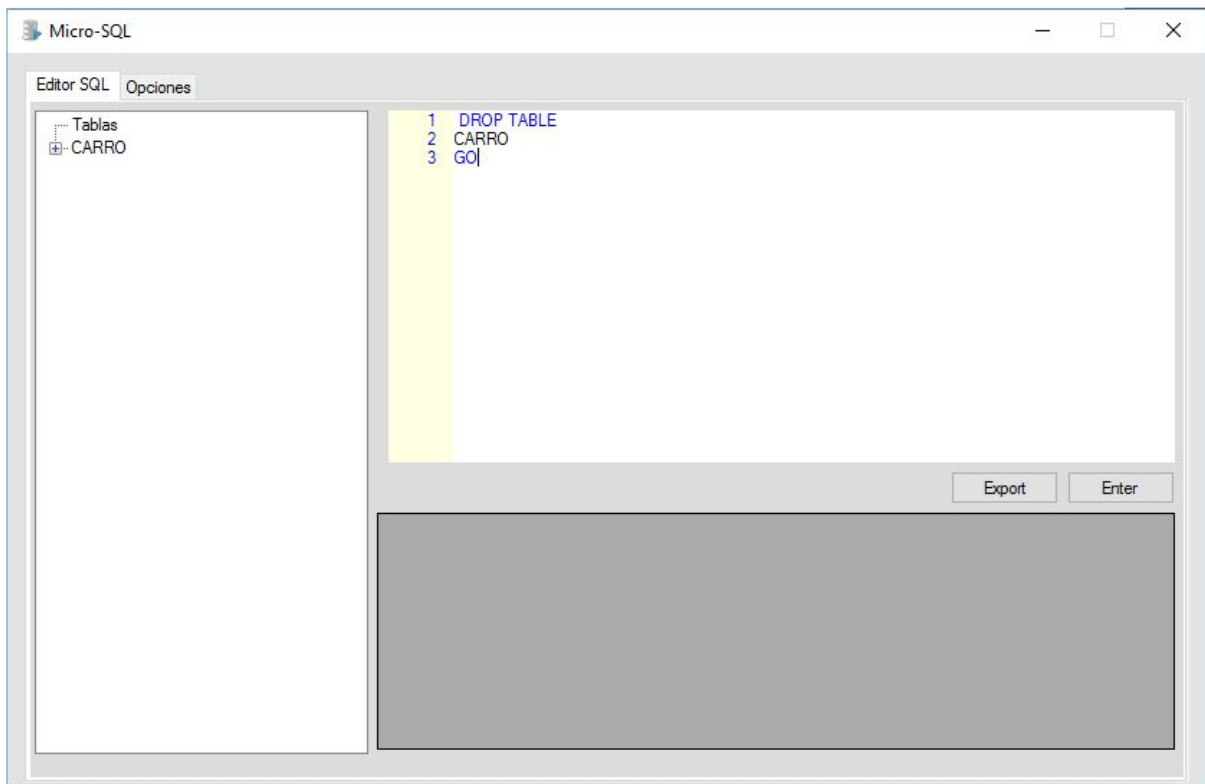
- **Eliminación de datos:**

- ❖ Permite eliminar datos almacenados en una tabla (eliminar filas de la tabla).
- ❖ Se valida que la sintaxis cumpla con todas las reglas establecidas, de lo contrario se muestra un error en pantalla.



- **Eliminación de tablas:**

- Elimina toda la tabla y sus datos, se borran todos los archivos asociados.
- Se valida que la sintaxis cumpla con todas las reglas establecidas, de lo contrario se muestra un error en pantalla.



Conclusiones

- El muy conocido C.R.U.D (Create, Read, Update, Delete) “crear, leer, actualizar o borrar”. Son las cuatro operaciones básicas que se realizan de forma más habitual.
- Algo importante para entender de una base de datos, además de almacenar información, es que todas las operaciones se realizan a través del lenguaje utilizado para ello que es S.Q.L (Structured Query Language) “lenguaje de consultas estructuradas”.
- MySQL es un motor que en comparación a Oracle DB es muy simple, su arquitectura es menos elaborada, pero es eficiente, fácil y rápido para algunos proyectos web, además de ser OpenSource.