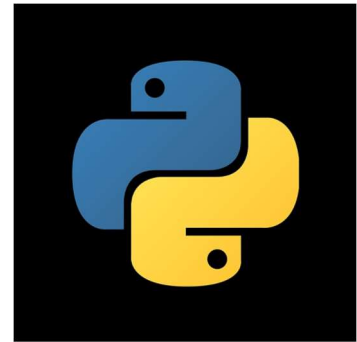


Laboratorio de Organización de Lenguajes y Compiladores 2.



PROYECTO 1 – XSQL IDE

MANUAL USUARIO

FECHA: 29/12/2023

Juan Josue Zuleta BeB

Carné: 202006353

Gerson Sebastian Quintana Berganza

Carné: 201908686

Objetivos del Sistema

El presente sistema se dirige a un público general interesado en interactuar con un Sistema de Gestión de Bases de Datos (DBSM) inteligente, diseñado para responder a comandos específicos. Estos comandos facilitarán la interacción del usuario con un DBMS tradicional capaz de gestionar bases de datos, realizar consultas y proporcionar resultados de manera similar a un DBMS convencional. Aunque el programa no abarca todos los aspectos de un DBMS actual, representa una sólida plataforma inicial para el desarrollo de futuros sistemas.

Además, se busca que el sistema sea una herramienta útil para comprender de manera más profunda el funcionamiento del procesamiento de lenguajes basados en sintaxis. Este programa ha sido desarrollado utilizando PLY (Python Lex-Yacc), incorporando cada uno de los elementos esenciales de un analizador. Cabe destacar que se incluyó la traducción a código de tres direcciones para lograr un programa más completo y versátil.

El objetivo principal es plasmar los conocimientos adquiridos durante el curso de Organización de Lenguajes y Compiladores 2, especialmente en los aspectos globales relacionados con analizadores y traductores de lenguajes. Este proyecto refleja el compromiso con la aplicación práctica de los conceptos aprendidos en el ámbito académico, contribuyendo al avance y comprensión de tecnologías clave en el desarrollo de sistemas de gestión de bases de datos.

Información del Sistema

El Sistema de Gestión de Bases de Datos (DBSM) propuesto tiene como objetivo principal proporcionar una interfaz inteligente y eficiente para usuarios que buscan interactuar con bases de datos de manera intuitiva. Desarrollado como parte del curso de Organización de Lenguajes y Compiladores 2, este sistema se centra en los siguientes aspectos clave:

- Interactividad Avanzada:

Ofrece una interfaz interactiva que permite a los usuarios comunicarse con el DBSM mediante comandos específicos.

Facilita la interacción con un DBMS tradicional para la gestión de bases de datos y la ejecución de consultas.

- Desarrollo como Base:

Sirve como una base sólida para futuros desarrollos en el ámbito de los Sistemas de Gestión de Bases de Datos (DBMS).

Aunque no abarca todos los aspectos de un DBMS completo, proporciona una plataforma inicial para el desarrollo continuo.

- Procesamiento de Lenguajes:

Utiliza PLY (Python, Lex, Yacc) para la implementación del analizador, cubriendo aspectos esenciales del procesamiento de lenguajes basados en sintaxis.

Incluye la traducción a código de tres direcciones, añadiendo versatilidad y complejidad al programa.

- Aplicación de Conocimientos Académicos:

Refleja los conocimientos adquiridos durante el curso de Organización de Lenguajes y Compiladores 2.

Contribuye a la comprensión práctica de analizadores y traductores de lenguajes en un entorno de desarrollo real.

Requisitos Mínimos del Sistema

Sistema operativo 64 bits

- Microsoft Windows 10/8/7/Vista/2003/XP (incl.64-bit)
- macOS 10.5 o superior
- Linux GNOME o KDE desktop
- Procesador a 1.6 GHz o superior
- 1 GB (32 bits) o 2 GB (64 bits) de RAM (agregue 512 MB al host si se ejecuta en una máquina virtual)
- 3 GB de espacio disponible en el disco duro
- Disco duro de 5400 RPM
- Tarjeta de vídeo compatible con DirectX 9 con resolución de pantalla de 1024 x 768 o más.
- Navegador web (Recomendado: Google Chrome)

Software (Indispensable tenerlo instalado)

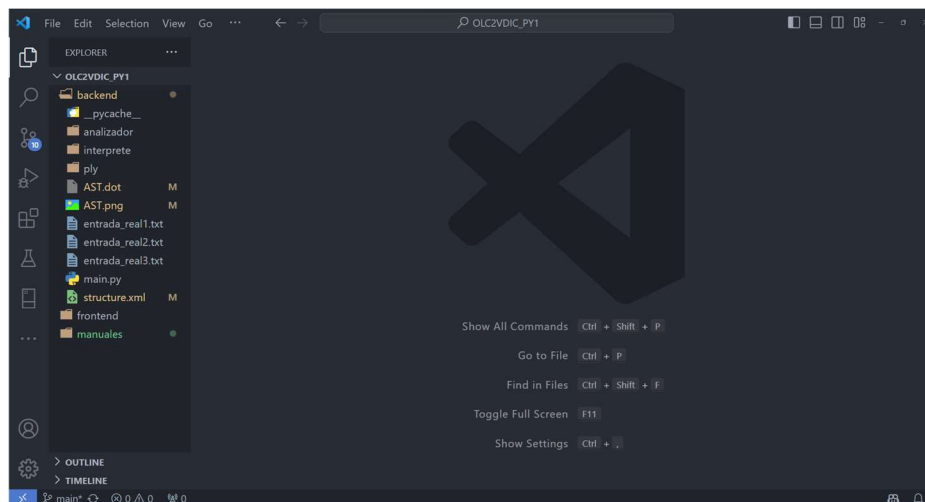
- IDE: Visual Studio Code
Puede conseguirse en:
<https://code.visualstudio.com/>
- Python 3.12 o posterior
Puede conseguirse en:
<https://www.python.org/downloads/>
- Node JS 21 o posterior
Puede conseguirse en:
<https://nodejs.org/en/download/current>

BIENVENIDO A XSQL IDE

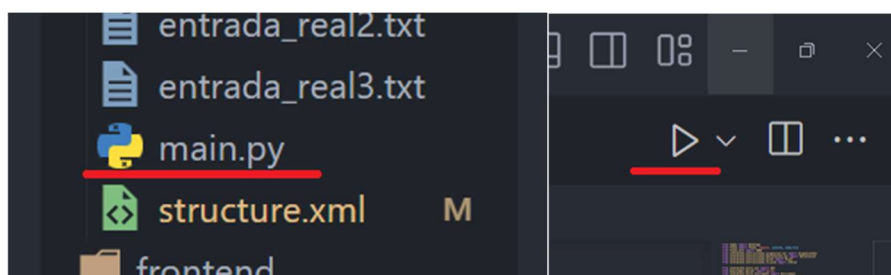
Inicialización del programa

Para poder acceder a la inicialización del programa se debe acceder a la carpeta de origen la cual contiene el código fuente. Par lo cual deberá seguir los siguientes pasos.

- Paso 1: Cumplir con los requisitos del sistema.
- Paso 2: Ejecutar Visual Studio Code, entorno en el cual fue desarrollado el programa.
- Paso 3: Descomprimir el archivo con extensión (zip).
- Paso 4: Arrastrar la carpeta dentro de Visual Studio Code, en caso requiera autorización o consentimiento, deberá aceptarlo. Tendrá una vista similar:




- Paso 5: Ejecutar el programa haciendo click izquierdo sobre el archivo con nombre main.py, paso seguido dar click al botón “play” ubicado en la esquina superior derecha de Visual Studio Code:



Nota Importante: Evite realizar cambios en el código fuente.

Una vez desplegado el servidor vera un mensaje como este:

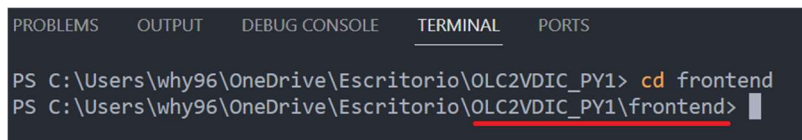


```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\why96\OneDrive\Escritorio\OLC2VDIC_PY1> & C:/Users/why96/AppData/Local/Programs/Python/Python312/python.exe c:/Users/why96/OneDrive/Escritorio/OLC2VDIC_PY1/backend/main.py
* Serving Flask app 'main'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 795-529-205
```

Esto significa que el servidor Flask se ha ejecutado con éxito, de no verse desplegado este mensaje revisar que los archivos no hayan sido alterados o revisar que tenga instaladas todas las dependencias del proyecto, librerías etc.

Una vez desplegado el Backend debe ingresar a la carpeta de frontend, por medio de la terminal, con el siguiente comando. Una vez ubicado en la carpeta raíz, escribir el comando 'cd frontend' esto lo ubicara en la carpeta frontend:

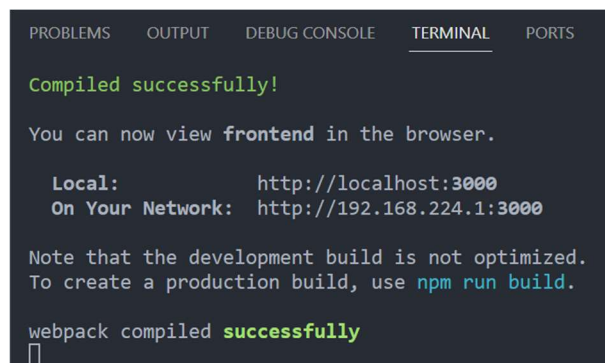


```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\why96\OneDrive\Escritorio\OLC2VDIC_PY1> cd frontend
PS C:\Users\why96\OneDrive\Escritorio\OLC2VDIC_PY1\frontend>
```

Una vez que se encuentre en esta carpeta deberá ejecutar el comando: **npm start** este comando desplegara el servidor de node en el cual se encuentra alojado nuestro frontend basado en React.js

Una vez desplegado el servidor, vera un mensaje como este:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Compiled successfully!

You can now view frontend in the browser.

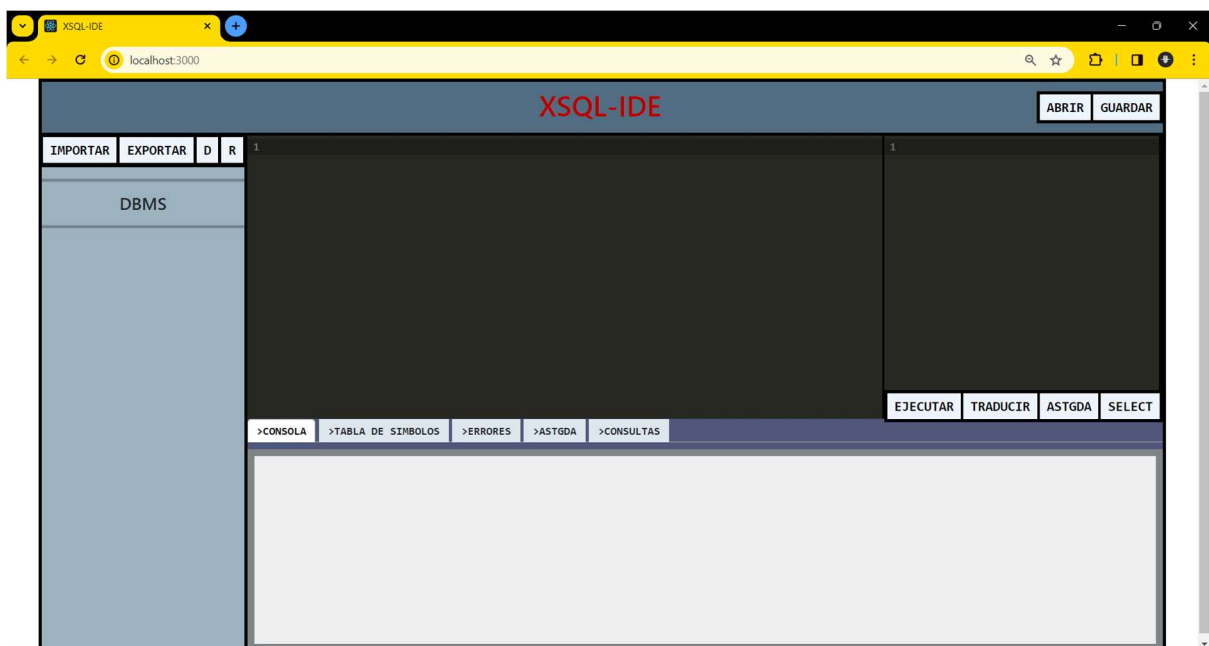
Local:      http://localhost:3000
On Your Network:  http://192.168.224.1:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Esto significa que el servidor Node.js se ha ejecutado con éxito, de no verse desplegado este mensaje revisar que los archivos no hayan sido alterados o revisar que tenga instaladas todas las dependencias del proyecto, librerías etc. Esto a su vez desplegará en su navegador por defecto la página web que es la interfaz de usuario, mediante la cual, podrá interactuar con la aplicación.

Vea una pantalla como esta:



Cada uno de los componentes de la aplicación se detallan a continuación, se recomienda seguir los pasos recomendados para obtener los resultados esperados.

Panel del DBMS:

Iniciaremos explicando el panel del lado izquierdo el cual se centra en el administrador de las bases de datos este panel se encarga de mostrar, modificar, importar, exportar todo lo relacionado con las bases de datos almacenadas.



Importar:

Con este botón podremos importar una base de datos existente de un archivo. SQL el cual contenga los inserts de registros de tablas.

Exportar:

Con este botón podremos exportar una base de datos existente en el DBMS hacia un archivo. SQL en el cual almacenaremos los inserts de registros de las tablas actuales de la base de datos seleccionada.

D *(Dump)

Con este botón podremos exportar una base de datos existente en el DBMS hacia un archivo. SQL en el cual almacenaremos las tablas actuales de la base de datos seleccionada.

R *(Recargar)

Con este botón recargaremos la interfaz, de modo que cualquier cambio dentro de las bases de datos almacenadas se verán reflejadas en el panel del DBMS, mostrando la base de datos actual y también todas las tablas, vistas, funciones y procedimientos.



Botones de función básica:



Abrir:

Con este botón podremos abrir un archivo, con extensión .SQL e importarlo directamente al área de texto del editor.

Guardar:

Con este botón podremos guardar el contenido actual del editor de texto en un archivo con extensión .SQL y de esta manera poder utilizarlo en futuras consultas.

Editor de texto:

Esta es la parte mas importante del IDE, en este panel podremos escribir sintaxis del lenguaje XSQL la cual podremos procesar de diferentes maneras la cuales se especificarán posteriormente.



El editor de texto cuenta con el resaltado de sintaxis lo cual facilitara reconocer palabras clave, caracteres especiales o diferenciar distintos parámetros dentro de funciones o procedimientos.

Panel de Ejecución:

En este panel encontraremos las funciones específicas del IDE las cuales nos permitirán obtener los resultados de nuestras consultas, los detalles de cada uno se especifican a continuación:



Ejecutar:

Para ejecutar el código se debe presionar el botón 'Ejecutar' mediante este botón se ejecutarán todas las instrucciones de este IDE, por lo cual este botón siempre debe ser presionado antes que cualquier otra petición, en orden siempre ira primero.

Traducir:

Este botón nos entregara la traducción a condigo de 3 direcciones sobre el panel de traducción, este botón generara el código de tres direcciones de las instrucciones especificas para las cuales esta preparado el IDE, estas son, asignaciones, declaraciones, if, while, case, funciones y procedimientos.

ASTGDA:

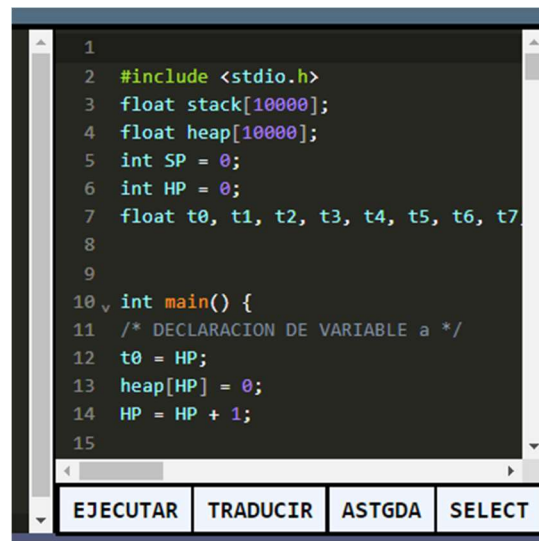
Para generar el Árbol de Sintaxis Abstracta, utilizamos este botón para generar una imagen que podemos ver detalladamente en el panel de reportes.

Select:

Este es quizá el botón mas importante dentro de las funcionalidades del IDE ya que nos permite visualizar de manera grafica las consultar realizadas dentro del IDE, estas consultas serán generadas en formato de tablas, estas consultas son generadas en el panel de reportes, el orden de los botones es, primero ejecutar, luego select, y por ultimo el panel de reportes, de no hacerlo en el orden especificado, no se obtendrán los resultados esperados.

Panel de traducción:

En este panel se genera el código de tres direcciones de las instrucciones previamente especificadas en la descripción del botón traducir:



```
1
2 #include <stdio.h>
3 float stack[10000];
4 float heap[10000];
5 int SP = 0;
6 int HP = 0;
7 float t0, t1, t2, t3, t4, t5, t6, t7;
8
9
10 int main() {
11     /* DECLARACION DE VARIABLE a */
12     t0 = HP;
13     heap[HP] = 0;
14     HP = HP + 1;
15 }
```

EJECUTAR TRADUCIR ASTGDA SELECT

Panel de reportes:

En este panel encontraremos todos los reportes generados en tiempo real por el IDE, los cuales se especifican a continuación:



Consola:

En este panel se muestran todas las respuestas por parte de IDE sobre cualquier comando ingresado, tanto como si se ejecuta con éxito o si resulta en un error, también se detallan las respuestas del analizador, de operaciones o funciones básicas del editor:

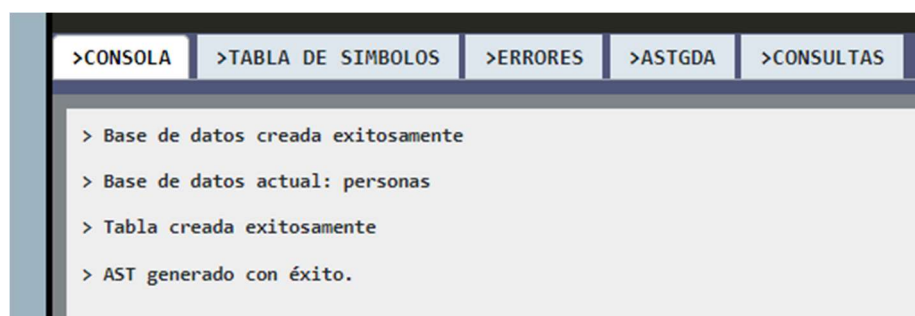


Tabla de símbolos:

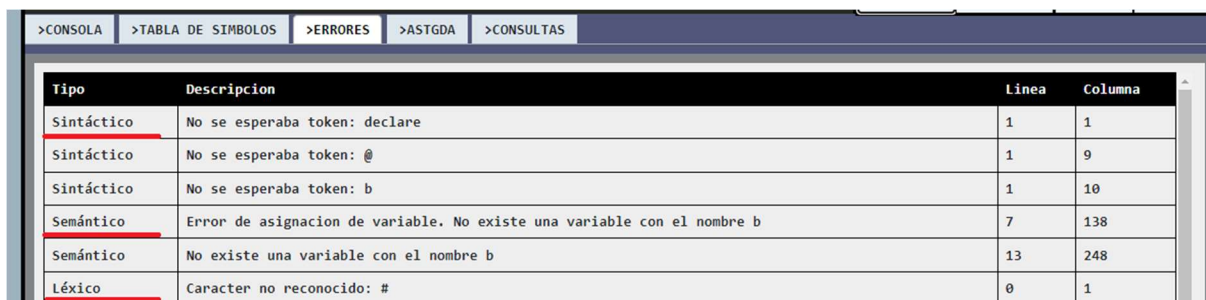
En este panel se muestran los símbolos actuales en el analizador, los símbolos reconocidos y sus características, estos son mostrados en formato de tablas:



| Id | Tipo | Valor | Entorno | Parametros | Simbolo |
|----|----------|------------------|---------|------------|----------|
| a | NVARCHAR | hola-cmoo | Global | | VARIABLE |
| b | INT | 12 | Global | | VARIABLE |
| c | DECIMAL | 44.4 | Global | | VARIABLE |
| d | BIT | 1 | Global | | VARIABLE |
| e | DATE | 12-12-2012 | Global | | VARIABLE |
| f | DATETIME | 12-12-2012 23:48 | Global | | VARIABLE |

Errores:

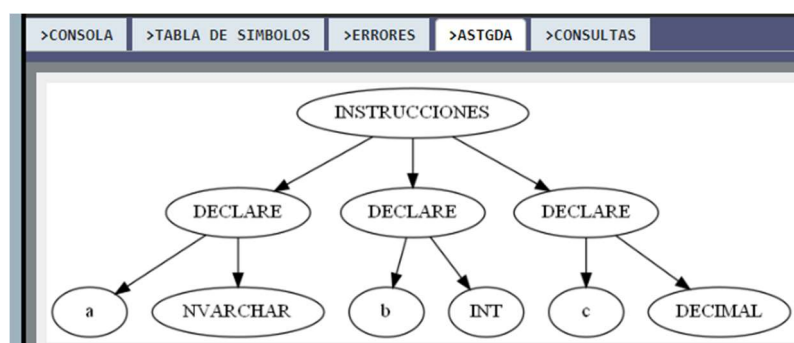
En este panel se muestran los errores detectados por el analizador, estos incluyen, errores léxicos, sintácticos y semánticos que puedan llegar a ocurrir a nivel de ejecución, estos serán almacenados en formato de tablas y serán tendrán estas características:



| Tipo | Descripcion | Linea | Columna |
|------------|---|-------|---------|
| Sintáctico | No se esperaba token: declare | 1 | 1 |
| Sintáctico | No se esperaba token: @ | 1 | 9 |
| Sintáctico | No se esperaba token: b | 1 | 10 |
| Semántico | Error de asignacion de variable. No existe una variable con el nombre b | 7 | 138 |
| Semántico | No existe una variable con el nombre b | 13 | 248 |
| Léxico | Caracter no reconocido: # | 0 | 1 |

ASTGDA:

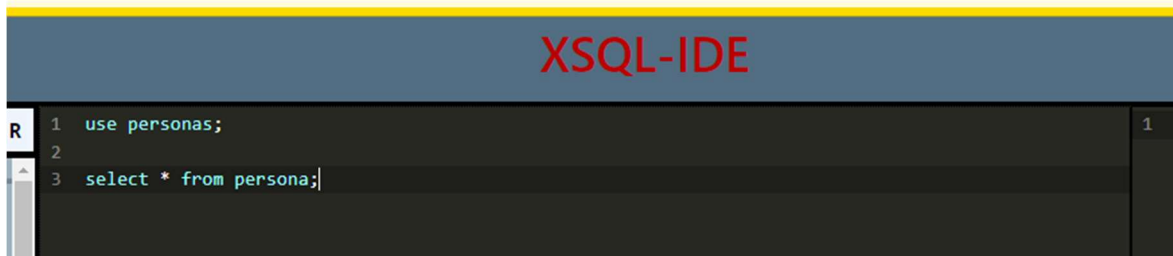
En este panel se muestra el Árbol de Sintaxis Abstracta generada por el analizador después de cada ejecución:



Consultas:

En este panel se muestran las consultas generadas a las bases de datos por medio del comando select, que es el comando mas poderoso dentro del editor, por medio del cual podemos consultar todo todo tipo de datos especificos o generales de las bases de datos, sobre tablas en especifico.

Ejemplo de consulta:

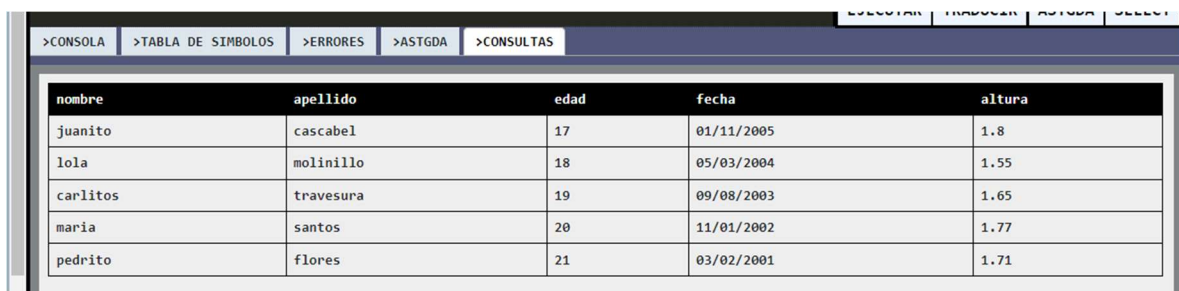


The screenshot shows the XSQL-IDE interface. The title bar is blue with 'XSQL-IDE' in red. Below it is a dark editor area with a light blue line number column on the left. The editor contains the following SQL code:

```
1 use personas;  
2  
3 select * from persona;
```

On the right side of the editor, there is a small vertical panel with the number '1'.

Ejemplo de respuesta:



The screenshot shows the XSQL-IDE interface with the 'CONSULTAS' tab selected. It displays a table with the results of a SQL query. The table has five columns: nombre, apellido, edad, fecha, and altura. The data is as follows:

| nombre | apellido | edad | fecha | altura |
|----------|-----------|------|------------|--------|
| juanito | cascabel | 17 | 01/11/2005 | 1.8 |
| lola | molinillo | 18 | 05/03/2004 | 1.55 |
| carlitos | travesura | 19 | 09/08/2003 | 1.65 |
| maria | santos | 20 | 11/01/2002 | 1.77 |
| pedrito | flores | 21 | 03/02/2001 | 1.71 |

Detener Cliente:

Para detener el cliente, en la consola que esta ejecutando el cliente React.js, se debe presionar 'ctrl+c' le preguntara si desea detener el servicio para lo cual debe presionar 'Y' si su editor esta en ingles o 'S' si su editor esta en español.

Detener el Servidor:

Para detener el servicio del servidor Flask, se debe ubicar en la consola que lo esta ejecutando y luego presionar 'ctrl+c' esto detendrá la ejecución del servidor.

Enlace a repositorio:

https://github.com/why-ego/OLC2VDIC_PY1