



## MANUAL TÉCNICO

Nombres:

Eduardo Alexander Reyes Gonzalez

Carnet: 202010904

Randy Marcelino Sum Coyoy

Carnet: 2253625030901

Fecha: 29 de diciembre de 2023



<b>Objetivo del manual técnico</b>	<b>2</b>
Alcance del sistema	2
Tecnologías utilizadas	2
Python:	3
PLY:	3
Graphviz:	3
Pillow:	3
<b>Entorno de Desarrollo</b>	<b>3</b>
Requisitos del sistema	3
Configuración del entorno de desarrollo	4
Descarga e instalación de dependencias	4
Python	4
Instalación en Windows:	4
Instalación en Manjaro Linux:	4
PLY	4
Graphviz	4
Instalación en Windows:	4
Instalación en Manjaro Linux:	4
Pillow	5
Instalación en Windows y Manjaro Linux:	5
Estructura del proyecto	6
ARCHIVOS ENTRADA	7
BASE_DATOS	7
DATOS EXPORTADOS	7
DUMPS	7
FUNCIONES	7
• ÁRBOL	7
• CREAR REPORTES	7
• DDL	7
• EXPORTAR IMPORTAR	7
• FUNCIONES SISTEMA	7
• OPERACIONES ARITMÉTICAS	7
• OPERACIONES LÓGICAS	7
• OPERACIONES RELACIONAL	8
• SSL	8
REPORTES	8
• ÁRBOL	8
• ERRORES	8
• TABLA	8
Gramatica.py	8
Interfaz.py	14
Despliegue	14



## Objetivo del manual técnico

El objetivo es proporcionar una guía detallada sobre la aplicación que es un intérprete para el lenguaje X-SQL mediante la traducción dirigida por la sintaxis. Facilitar la comprensión del sistema, implementación y mantenimiento. Así como servir de referencia para modificaciones, mejoras y actualizaciones en el futuro.

## Alcance del sistema

XSQL, una aplicación desarrollada como parte del curso de Organización de Lenguajes y Compiladores 2. Este IDE ha sido creado con el propósito de aplicar los conocimientos adquiridos en el curso, centrándose en el desarrollo de un intérprete para el lenguaje X-SQL. Para lograr este objetivo, hacemos uso de herramientas avanzadas para la generación de analizadores léxicos y sintácticos, en particular, la herramienta PLY (Python Lex-Yacc)

XSQL está diseñado para facilitar la creación, edición y ejecución de programas escritos en el lenguaje X-SQL. Este lenguaje, específicamente desarrollado para este proyecto, combina características de SQL con elementos adicionales, ofreciendo una solución versátil para el manejo de datos. Este sistema utiliza un formato de archivos XML para almacenar y organizar la estructura de las bases de datos, incluyendo tablas, funciones y procedimientos.

## Tecnologías utilizadas

En el desarrollo de nuestro proyecto, hemos adoptado un conjunto específico de tecnologías para garantizar la implementación exitosa de XSQL. A continuación, se presenta una descripción detallada de cada tecnología

### Python:

Python es un lenguaje de programación interpretado, versátil y de alto nivel. Su sintaxis clara y su amplia biblioteca estándar lo hacen ideal para el desarrollo de aplicaciones complejas.

### PLY:

PLY es una herramienta para construir analizadores léxicos y sintácticos en Python. Facilita la implementación de gramáticas complejas y análisis preciso.



Graphviz:

Graphviz es una herramienta para visualizar gráficamente estructuras de datos y grafos. En nuestro proyecto, utilizamos Graphviz para representar árboles de sintaxis.

Pillow:

Pillow es una biblioteca de imágenes para Python. En XSQL, la utilizamos para manipular gráficos e imágenes.

## Entorno de Desarrollo

### Requisitos del sistema

Antes de iniciar con el desarrollo o ejecución de la aplicación verifica que tu sistema Windows o Linux cumple con los siguientes requisitos

Python:

- Versión 3.6 o superior

Graphviz:

- Puedes instalarlo desde [graphviz.org](http://graphviz.org)

PLY:

- Mediante "pip install ply"

Pillow:

- Mediante "pip install pillow"

Conexión a internet para descargar e instalar las dependencias adicionales durante el proceso de instalación, un entorno correctamente configurado

### Configuración del entorno de desarrollo

- Clona el repositorio desde github con git clone  
<https://github.com/Eduardo1232000/OLC2-Proyecto1-3013805890101-2253625030901.git>
- Inicia con tu editor de texto preferido, recomendamos utilizar Visual Studio Code



---

## Descarga e instalación de dependencias

### Python

#### Instalación en Windows:

Descarga el instalador desde [python.org](https://python.org).

Sigue el asistente de instalación y asegúrate de marcar la casilla "Agregar Python al PATH" durante la instalación.

#### Instalación en Manjaro Linux:

Usa el administrador de paquetes Pacman: `sudo pacman -S python`.

### PLY

Usa pip, el gestor de paquetes de Python: `pip install ply`.

### Graphviz

#### Instalación en Windows:

Descarga el instalador desde [graphviz.org](https://graphviz.org).

Sigue el asistente de instalación.

#### Instalación en Manjaro Linux:

Usa Pacman: `sudo pacman -S graphviz`.

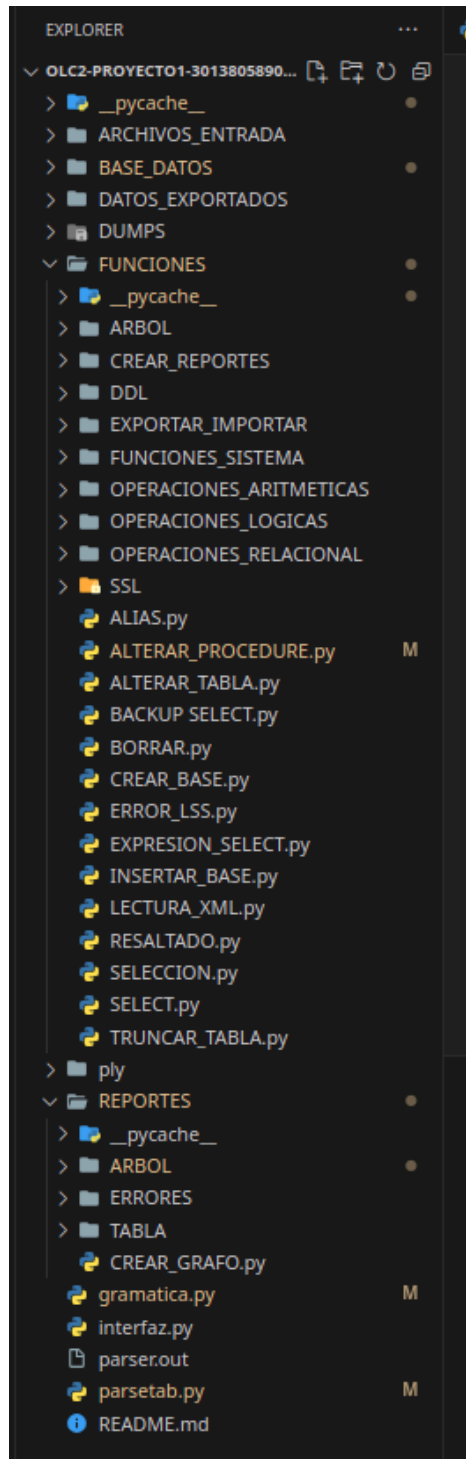
### Pillow

#### Instalación en Windows y Manjaro Linux:

Usa pip: `pip install pillow`.



## Estructura del proyecto





---

## ARCHIVOS ENTRADA

En este apartado tenemos documentos en formato .sql que nos sirven para hacer las pruebas de cada una de las instrucciones que puede realizar el IDE

## BASE\_DATOS

Aquí tenemos archivos en formato .xml en donde se guarda la información sobre las bases de datos creadas, tablas, funciones y procedimientos, aquí es donde se maneja y se manipula la información que se ingresa o que se solicita

## DATOS EXPORTADOS

Aquí se guardan archivos en formato .sql sobre la información que se tiene en base\_datos sobre alguna base, tabla, función o procedimiento

## DUMPS

Aquí se guardan los archivos con formato .sql que permiten crear un script de la base de datos, es un archivo con la creación de tablas, funciones y procedimientos. Solamente Estructura

## FUNCIONES

Aquí están los archivos que permiten la manipulación de funciones que retornan un valor escalar, hay funciones definidas por el usuario y funciones propias del sistema de Base de Datos.

- *ÁRBOL*
  - Aquí se encuentran los archivos que sirven para la creación de árboles ast, tablas de símbolos, tabla de funciones, variables actuales y globales, ejecución para instrucciones y expresiones, tipos y valores.
- *CREAR REPORTES*
  - Aquí se maneja la lógica para la creación de reportes de árbol, de errores, tabla de símbolos
- *DDL*
  - Aquí se maneja la lógica para la creación, eliminación, actualización de bases de datos, tablas, funciones, procedimientos
- *EXPORTAR IMPORTAR*
  - Para importar o exportar un archivo, así como la creación de DUMP
- *FUNCIONES SISTEMA*
  - funciones como CAS, CONCATENA, CONTAR, HOY.
- *OPERACIONES ARITMÉTICAS*
  - Funciones aritméticas como SUMA, RESTA, MULTIPLICACIÓN.
- *OPERACIONES LÓGICAS*
  - Funciones AND, OR, NOT



- *OPERACIONES RELACIONAL*
- *SSL*
  - Aquí se maneja la lógica del onjunto de sentencias propias del sistema XSQL, permite la escritura de sentencias de control y funciones propias utilizadas en el manejo de la información.

## REPORTES

Aquí se maneja la lógica para la creación de los distintos tipos de reportes

- *ÁRBOL*
- *ERRORES*
- *TABLA*

### Gramatica.py

Aquí se genera el analizador lexico, sintactico y semantico, tiene los tokens, las reglas de producción y la gramática para el sistema XSQL

La gramática es la siguiente:

```
"""instrucciones : instruccion instrucciones
                | instruccion """
```

```
"""instruccion : f_sistema
                | sent_create_database
                | sent_create_table
                | use_base
                | f_insert
                | f_delete
                | sent_alter_table
                | sent_drop
                | sent_truncate
                | declarar_var
                | asignacion_variable
                | declarar_procedure
                | exec_procedure
                | declarar_funcion
                | ins_if
                | ins_while
                | func_return
                """
```

```
"""ins_while : WHILE expresion BEGIN instrucciones END PTCOMA"""
```

```
"""ins_if : IF expresion THEN instrucciones ELSE instrucciones END IF PTCOMA """
```





""func\_return : RETURN expresion PTCOMA""

"" declarar\_funcion : CREATE FUNCTION name PARABRE variables\_procedure  
PARCIERRA RETURN \_dato AS BEGIN instrucciones END PTCOMA  
| CREATE FUNCTION name RETURN tipo\_dato AS BEGIN  
instrucciones END PTCOMA""

""llamada\_funcion : name PARABRE valores\_proc PARCIERRA  
| name PARABRE PARCIERRA""

"" declarar\_procedure : CREATE PROCEDURE name PARABRE variables\_procedure  
PARCIERRA AS BEGIN instrucciones END PTCOMA  
| CREATE PROCEDURE name AS BEGIN instrucciones END PTCOMA""

"" variables\_procedure : var\_procedure COMA variables\_procedure  
| var\_procedure""

"" var\_procedure : ARROBA name tipo\_dato  
| ARROBA name AS tipo\_dato""

"" exec\_procedure : EXEC name valores\_proc PTCOMA  
| EXEC name PTCOMA""

"" valores\_proc : valores\_procedure\_variables  
| valores\_procedure\_simple""

""valores\_procedure\_variables : valor\_procedure\_variable COMA valores\_proc  
| valor\_procedure\_variable""

""valor\_procedure\_variable : ARROBA name IGUAL expresion""

""valores\_procedure\_simple : valor\_procedure\_simple COMA valores\_proc  
| valor\_procedure\_simple""

""valor\_procedure\_simple : expresion""

"" declarar\_var : DECLARE ARROBA name tipo\_dato PTCOMA  
| DECLARE ARROBA name AS tipo\_dato PTCOMA""

"" asignacion\_variable : SET ARROBA name IGUAL expresion PTCOMA""

"" f\_sistema : SELECT expresiones\_select\_ini continuacion\_from""

""expresiones\_select\_ini : expresion\_select\_inicio COMA expresiones\_select\_ini



---

```
| expresion_select_inicio""

""expresion_select_inicio : expresion
    | MULTIPLICACION""

""expresion_select_final : expresion_select_final IGUAL expresion_select_final
    | expresion_select_final signo_relacional
expresion_select_final
    | expresion""

""alias : name PUNTO name""

""continuacion_from : PTCOMA
    | FROM columnas continuacion_where""

""continuacion_where : PTCOMA
    | WHERE expresion_select_final PTCOMA""

""operacion_sis : func_concatena
    | func_substraer
    | func_hoy
    | func_contar
    | func_suma
    | func_cas
    | exp_case
""

'func_concatena : CONCATENA PARABRE expresion COMA expresion PARCIERRA'

'func_substraer : SUBSTRAER PARABRE expresion COMA numero COMA numero
PARCIERRA'

'func_hoy : HOY PARABRE PARCIERRA'

""func_contar : CONTAR PARABRE MULTIPLICACION PARCIERRA ""

""signo_relacional : DIFERENTE
    | MENORQUE
    | MAYORQUE
    | MENORIGUAL
    | MAYORIGUAL""

""func_suma : SUMA PARABRE name PARCIERRA ""
```

---



---

'func\_cas : CAS PARABRE expresion AS tipo\_dato PARCIERRA '

""condiciones : expresion  
| expresion AND condiciones  
| expresion OR condiciones ""

""exp\_aritmetica : op\_suma  
| op\_resta  
| op\_multiplicacion  
| op\_division""

""op\_suma : expresion MAS expresion ""

""op\_resta : expresion RESTA expresion ""

""op\_multiplicacion : expresion MULTIPLICACION expresion ""

""op\_division : expresion DIVISION expresion ""

""exp\_relacional : exr\_igual  
| exr\_diferente  
| exr\_menorque  
| exr\_mayorque  
| exr\_menorigual  
| exr\_mayorigual""

""exr\_igual : expresion IGUALIGUAL expresion""

""exr\_diferente : expresion DIFERENTE expresion""

""exr\_menorque : expresion MENORQUE expresion""

""exr\_mayorque : expresion MAYORQUE expresion""

""exr\_menorigual : expresion MENORIGUAL expresion""

""exr\_mayorigual : expresion MAYORIGUAL expresion""

""exp\_logica : exl\_and  
| exl\_or  
| exl\_not""

""exl\_and : expresion AND expresion""



""exl\_or : expresion OR expresion""

""exl\_not : RNOT expresion %prec RNOT""

""sent\_create\_database : CREATE DATA BASE name PTCOMA ""

""sent\_create\_table : CREATE TABLE name PARABRE datos PARCIERRA PTCOMA""

""sent\_alter\_table : ALTER TABLE name alter\_action PTCOMA""

""alter\_action : alter\_add  
| alter\_drop  
| alter\_modify""

""alter\_add : ADD COLUMN name tipo\_dato  
| ADD CONSTRAINT name FOREIGN KEY PARABRE name PARCIERRA  
REFERENCES name PARABRE name PARCIERRA""

""alter\_drop : DROP COLUMN name  
| DROP CONSTRAINT name""

""alter\_modify : MODIFY COLUMN name tipo\_dato""

""sent\_drop : DROP TABLE name PTCOMA  
| DROP DATA BASE name PTCOMA""

""sent\_truncate : TRUNCATE TABLE name PTCOMA""

""datos : dato COMA datos  
| dato ""

""dato : dato\_con\_caract  
| dato\_sin\_caract  
| foreign\_key""

""dato\_sin\_caract : name tipo\_dato ""

""dato\_con\_caract : name tipo\_dato características""

""use\_base : USE name PTCOMA""

""foreign\_key : FOREIGN KEY PARABRE name PARCIERRA REFERENCE name  
PARABRE name PARCIERRA""



---

```
""caracteristicas : caracteristica caracteristicas
    | caracteristica""

""caracteristica : caract_nulo
    | caract_no_nulo
    | caract_primary_key
    | referencia_tabla""

"" referencia_tabla : REFERENCE name PARABRE name PARCIERRA""

""caract_nulo : NULL""

""caract_no_nulo : NOT NULL""

""caract_primary_key : PRIMARY KEY""

"" f_insert : INSERT INTO name PARABRE columnas PARCIERRA VALUES PARABRE
valores PARCIERRA PTCOMA""

"" f_delete : DELETE FROM name WHERE name IGUAL expresion PTCOMA""

"" columnas : name
    | name COMA columnas""

"" valores : expresion
    | expresion COMA valores""

""tipo_dato : INT
    | BIT
    | DECIMAL
    | DATE
    | DATETIME
    | dato_char
    | dato_varchar""

""dato_char : NCHAR PARABRE NINT PARCIERRA
    | NCHAR""

""dato_varchar : NVARCHAR PARABRE NINT PARCIERRA
    | NVARCHAR""

""expresion : exp_aritmetica
    | exp_logica
    | operacion_sis
```

---



```
| llamada_funcion
| exp_relacional
| variable
| op_between
| exp_if
| parentesis
| numero
| FECHA
| FECHAHORA
| CADENA
| expr_select
| name_columna
| alias ""

"" exp_case : CASE ca_whens ca_else END""

""ca_whens : ca_when ca_whens
  | ca_when""

""ca_when : WHEN expresion THEN expresion""

""ca_else : ELSE THEN expresion""

"" expr_select : expresion name_columna""

""variable : ARROBA name ""

""parentesis : PARABRE expresion PARCIERRA""

""numero : NINT
  | NBIT
  | NDECIMAL""

""name_columna : NOMBRE""

""name : NOMBRE""
```

Interfaz.py

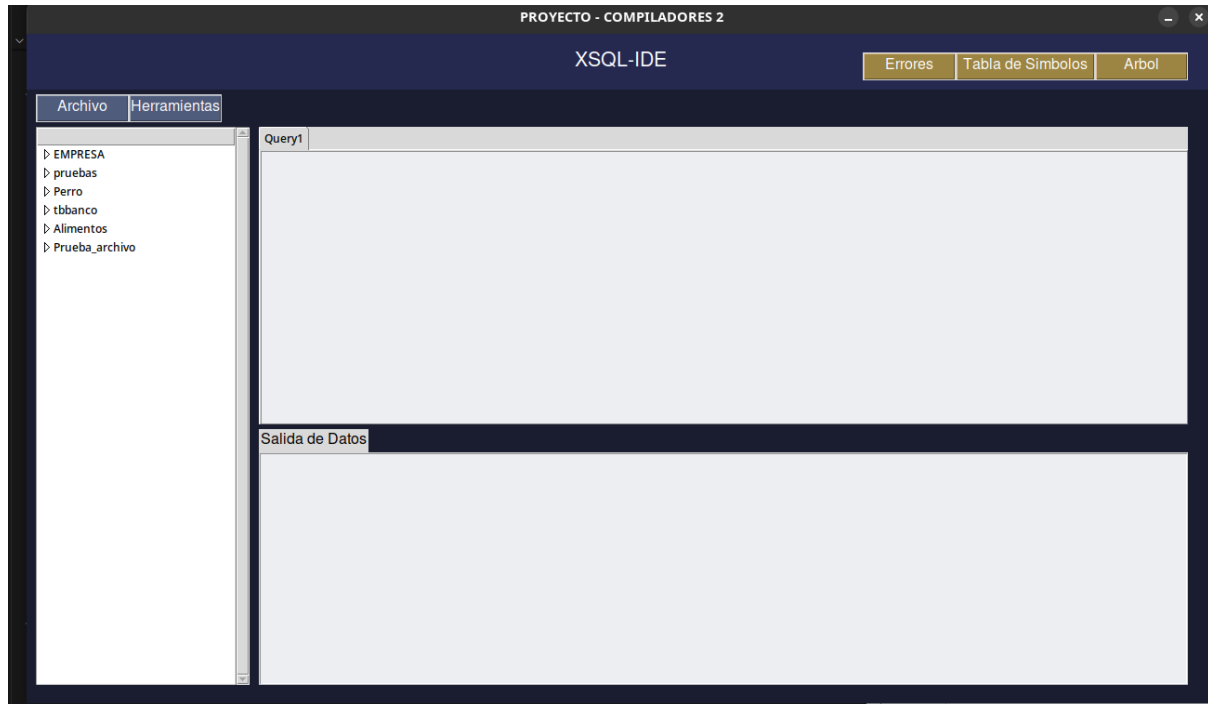
Aquí se maneja la lógica para el IDE que se utilizara, se usa tkinter para la interfaz gráfica

## Despliegue

- Navega al directorio donde está clonado el repositorio y abre tu editor de código favorito, recomiendo visual studio code



- Abre una nueva terminal y ejecuta los siguientes comandos:
  - `python interfaz.py`
- Se abrirá la interfaz gráfica de la siguiente manera:



Aquí podrás introducir las instrucciones XSQL en query, ejecutar un query desde la pestaña Herramientas, hacer imports, exports, ver la tabla de errores, símbolos y el árbol, podrás ver una salida de datos para hacer pruebas.