

Universidad de San Carlos de Guatemala

Ingeniería en Ciencias y Sistemas

Laboratorio Organización de Lenguajes y Compiladores 2.



XSQL MANUAL TECNICO

Jhony Roel Fuentes López 202031288

Estuardo Israel Ramos Gómez 201830358

Introducción

En el vertiginoso mundo de la informática, el diseño y la implementación de compiladores desempeñan un papel fundamental en la creación de herramientas eficientes y potentes. En el contexto del curso de Compiladores 2, hemos abordado el desafío de desarrollar una aplicación que acepta el lenguaje XSQL, lo analiza y ejecuta operaciones básicas. Este proyecto no solo representa un ejercicio académico, sino también una oportunidad para explorar la convergencia de la teoría de compiladores con la práctica de la programación.

Objetivo del Proyecto:

El propósito central de nuestro proyecto es ofrecer una aplicación robusta y eficiente que pueda procesar sentencias XSQL, entendiendo su estructura gramatical y ejecutando las operaciones correspondientes. Para alcanzar este objetivo, hemos empleado la potencia de la biblioteca PLY (Python Lex-Yacc) junto con el lenguaje de programación Python. PLY proporciona las herramientas necesarias para construir analizadores léxicos y sintácticos de manera modular y fácil de entender, lo que resulta crucial en el desarrollo de compiladores.

Tecnologías utilizadas:

Python

Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML). Los desarrolladores utilizan Python porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas diferentes. El software Python se puede descargar gratis, se integra bien a todos los tipos de sistemas y aumenta la velocidad del desarrollo

PLY

PLY es una herramienta de análisis escrita exclusivamente en Python. Es, en esencia, una reimplementación de Lex y Yacc originalmente en lenguaje C. Fue escrito por David M. Beazley. PLY utiliza la misma técnica de análisis LALR que Lex y Yacc. También cuenta con amplias funciones de depuración y notificación de errores.

Requisitos de software.

Python versión 3.3

Dependencias:

PLY versión 4.0

graphviz 0.20.1

Pandas versión 3.0

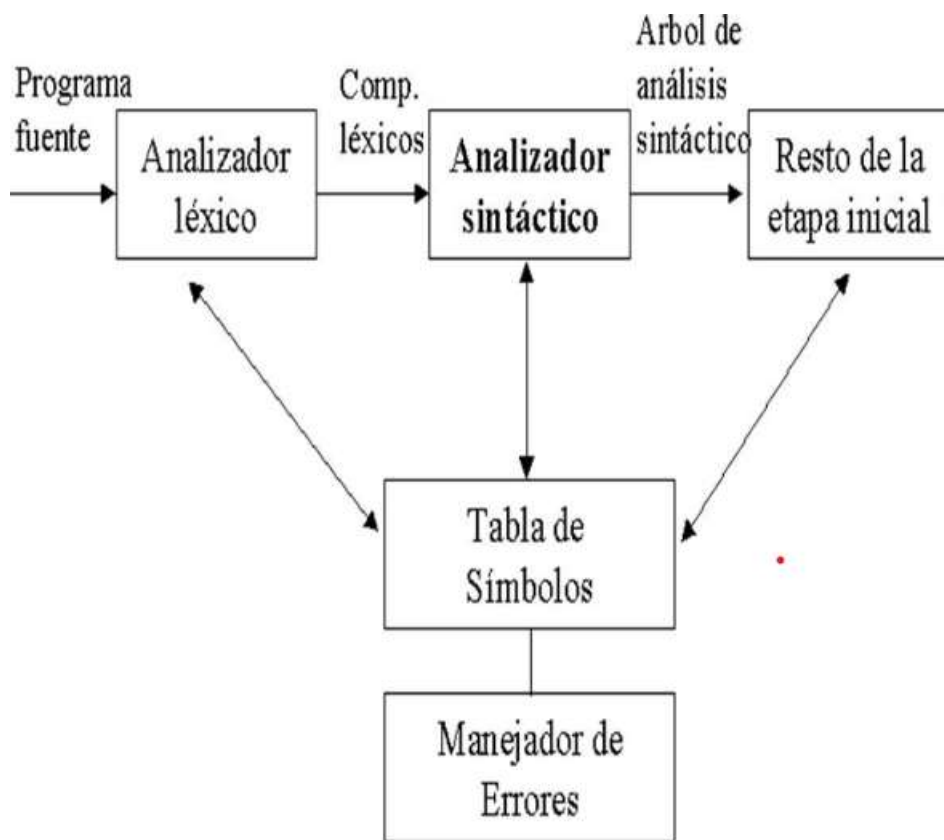
Requisitos mínimos de hardware.

4 GB de RAM

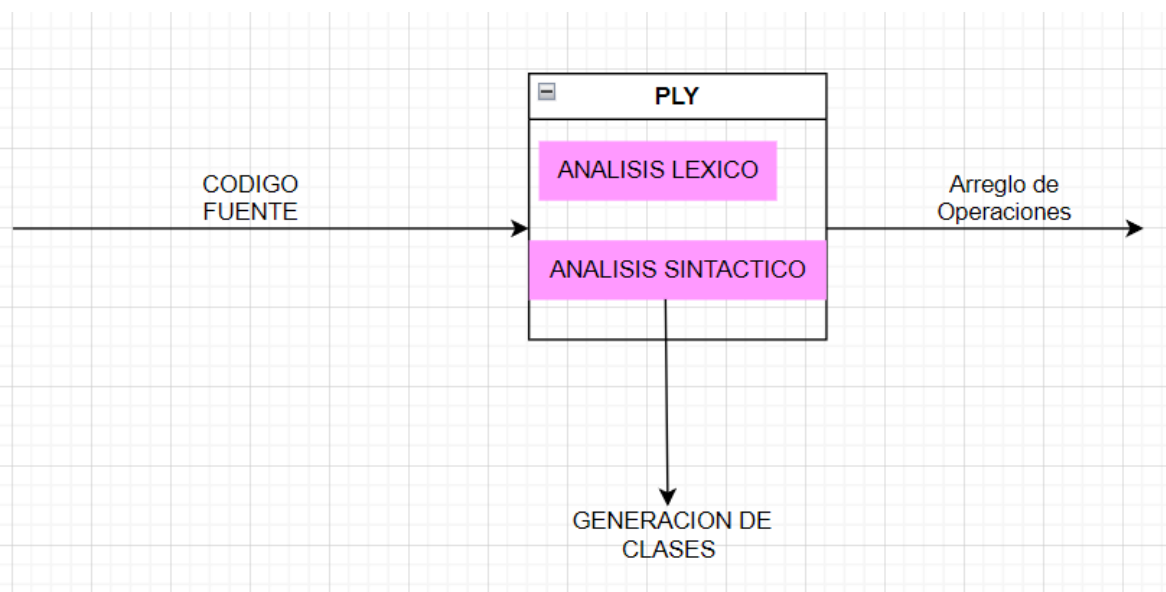
50 GB de espacio espacio disponible.

Sistema Operativo Windows 8 o superiores, o distribuciones Linux

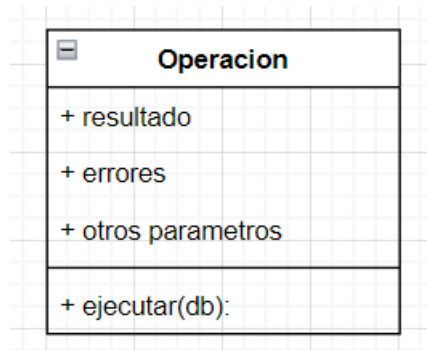
Arquitectura del compilador.



El código es introducido por la interfaz gráfica, este texto es enviado a ply para su análisis léxico y sintáctico en el análisis sintáctico dentro de la clase Parser.py, se crean las clases donde viene las instrucciones para ejecutar cada instrucción, para que después de completar el análisis, me devuelva un arreglo con todos los objetos creado según la operación que se necesita realizar.



Todas las clases tiene una estructura parecida: Tiene los parámetros para ejecutar la acción, además todas las clases tiene un método llamado **ejecutar** que recibe un parámetro, que es la base de datos que se tiene elegida actualmente. Además, se tienen 2 variables para guardar las salidas y errores que se pueden producir en la ejecución de cada operación.

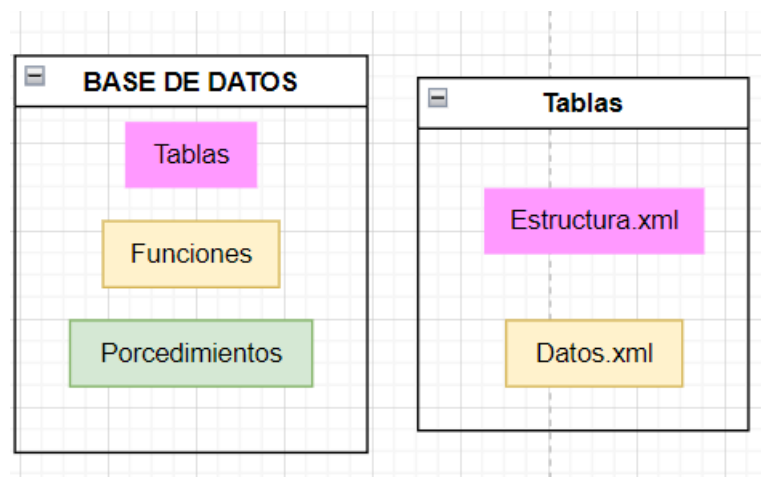


Estructura de las bases de datos.

Al momento de crear una base de datos, se crea un directorio con el nombre de esta, en la carpeta databases que se encuentra en el directorio principal, además se crea el directorio Tablas, donde se guardan la información de las tablas creadas.

Tablas:

Al momento de crear una tabla se crea un directorio con el nombre de la tabla creada, dentro de este directorio se crean 2 archivos xml, el primero es **Estructura.xml**, dentro de este se guarda los campos de la tabla, los tipos de estos y las llaves primarias y foráneas, para que en **Datos.xml** se guarden los registros ingresados por medio de la operación INSERT.



FUNCIONES Y PROCEDIMIENTOS

Para las funciones, al momento de guardarlas se crea un objeto de la clase Funcion, dentro de esta clase se verifica que no exista una función con el mismo nombre, si esto es correcto se procede a guardar la función en un arreglo donde se almacenan las funciones por base de datos, la misma clase se utiliza para los procedimientos, la diferencia es que los procedimientos no tienen ninguna salida.

Ejecutar:

Para ejecutar las funciones se crea la clase Ejecutar en este, se pasa el nombre de la función que se quiere ejecutar y los parámetros que son pasados a la tabla de símbolos de la función estos pueden ser nulos, se busca la función en el arreglo de funciones, si se encuentra se procede a iterar todos los elementos que tiene la función en Operaciones, utilizando el método ejecutar(db), descrito anteriormente.

