UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERIA

ESCUELA DE CIENCIAS Y SISTEMAS

LABORATORIO DE LENGUAJES Y COMPILADORES 2

**MANUAL TECNICO**

NOMBRES: ROBIN OMAR BUEZO DIAZ

BRANDON ANDY JEFFERSON TEJAXUN PICHIYA

KEWIN MASLOVY PATZAN TZUN

CARNET: 201944994

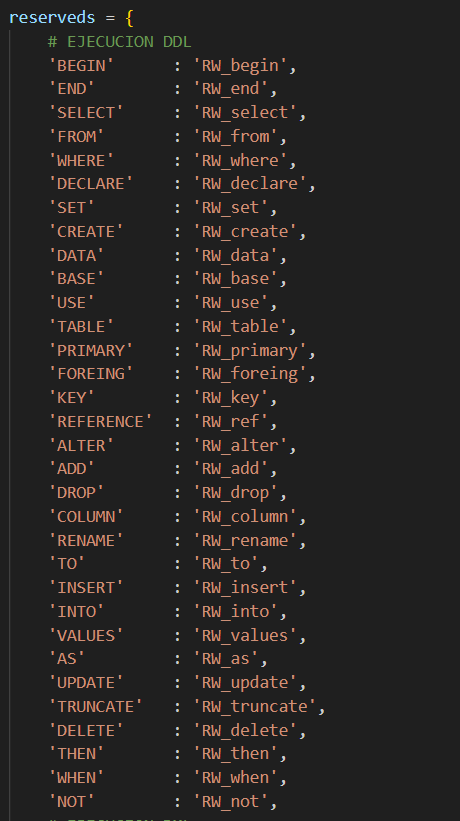
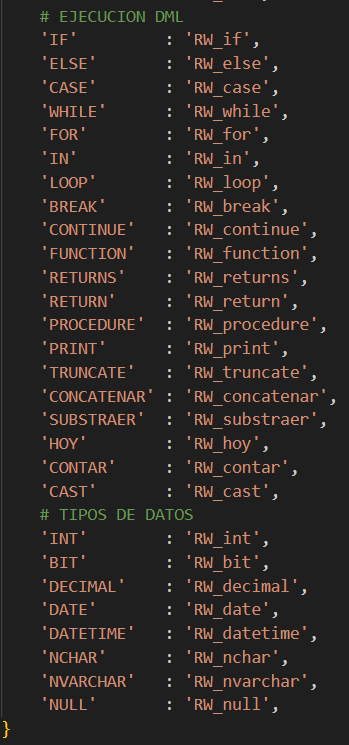
202112030

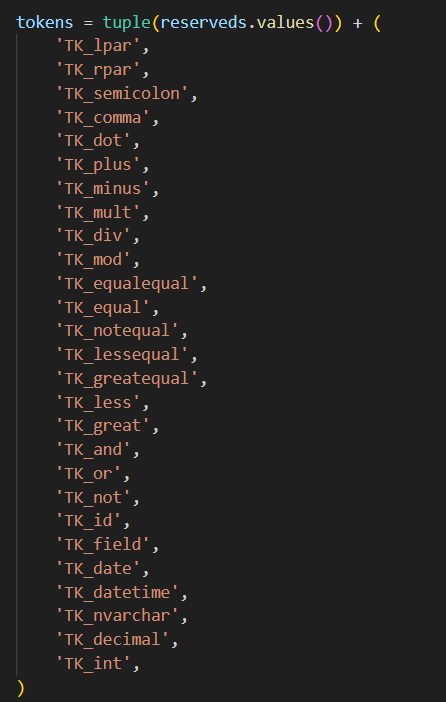
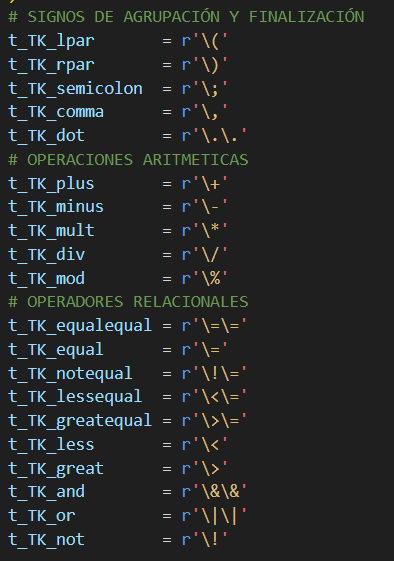
202103206

SECCION: A

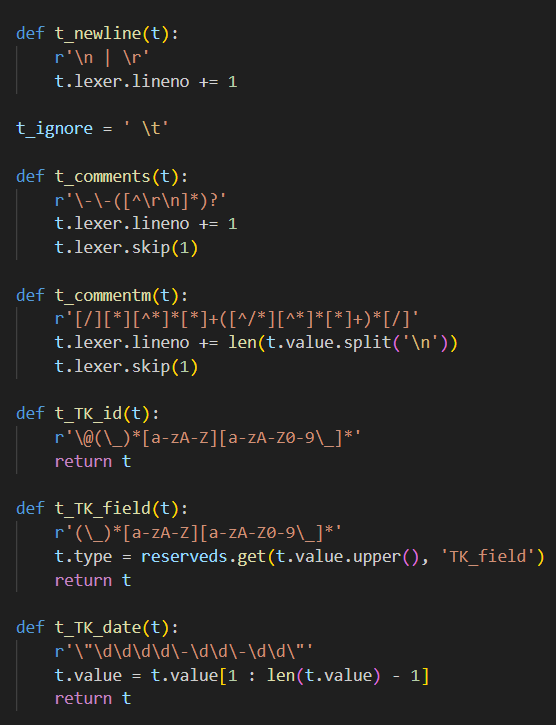
**ANALIZADOR LEXICO(SCANNER)**

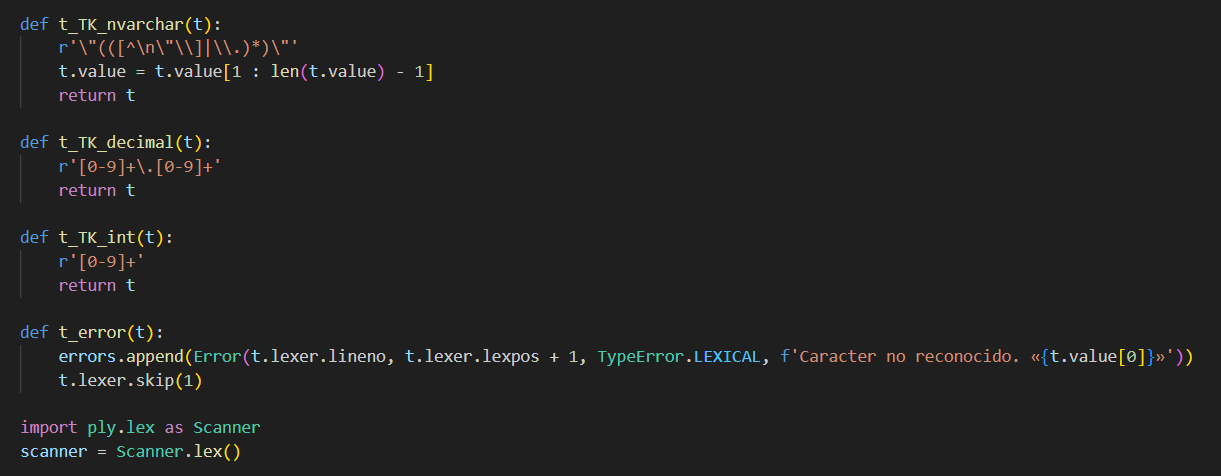
**Reglas** **Léxicas**: Creación de las reglas léxicas que el compilador utilizara durante la ejecución del programa.





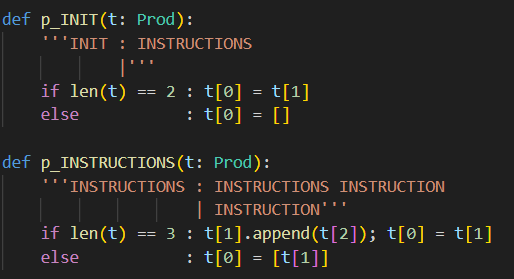
**Expresión Regulares:** Definición de las expresiones regulares que el programa podrá utilizar en la lectura de las entradas que esta recibe.

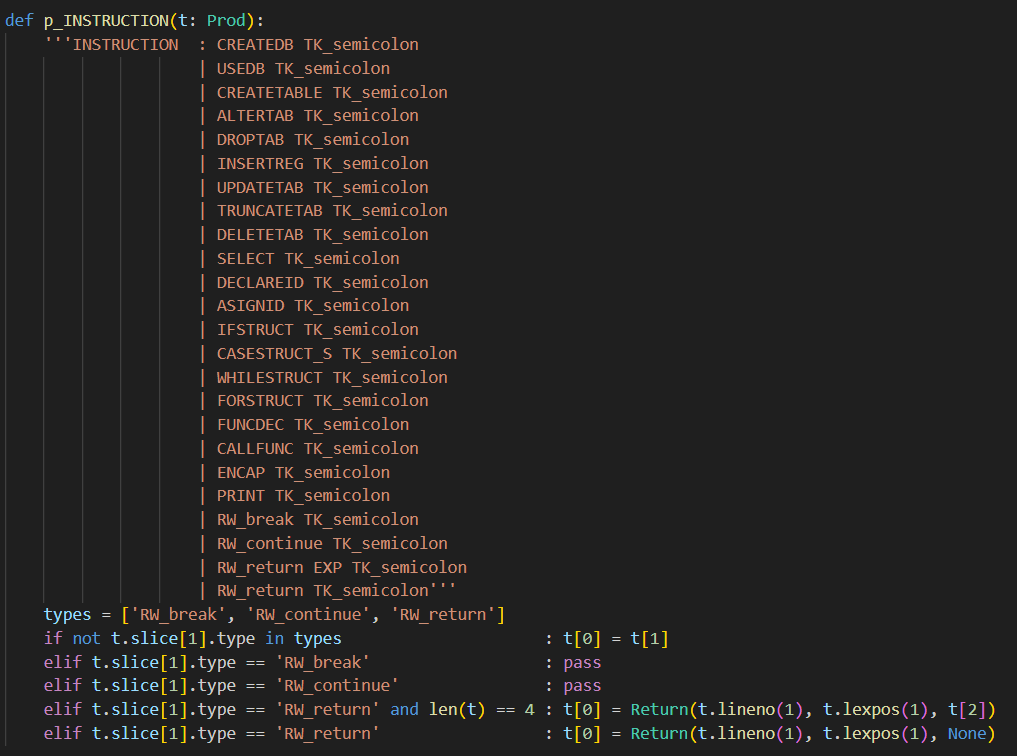


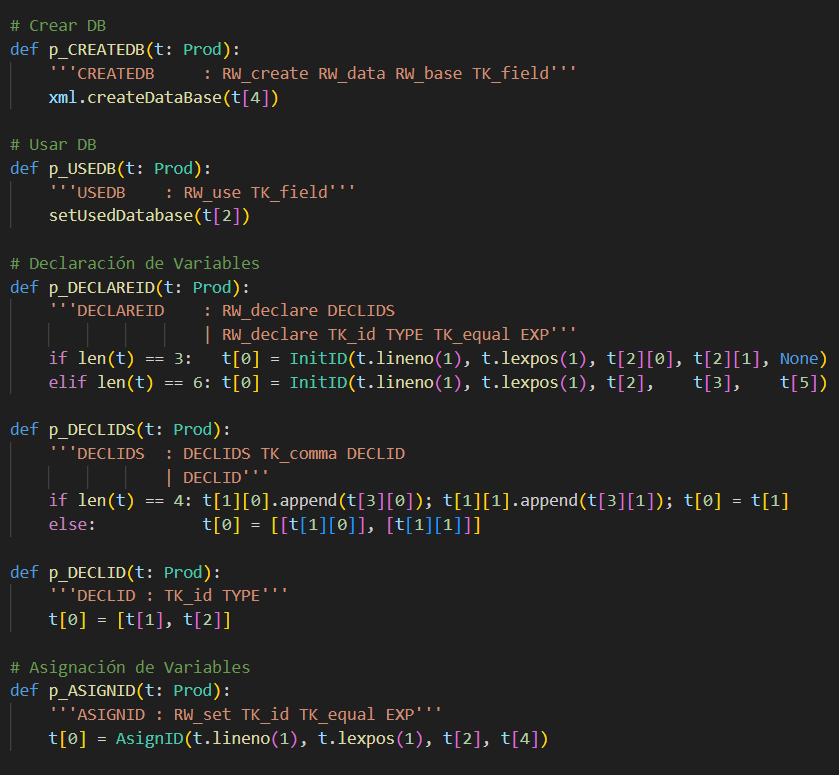


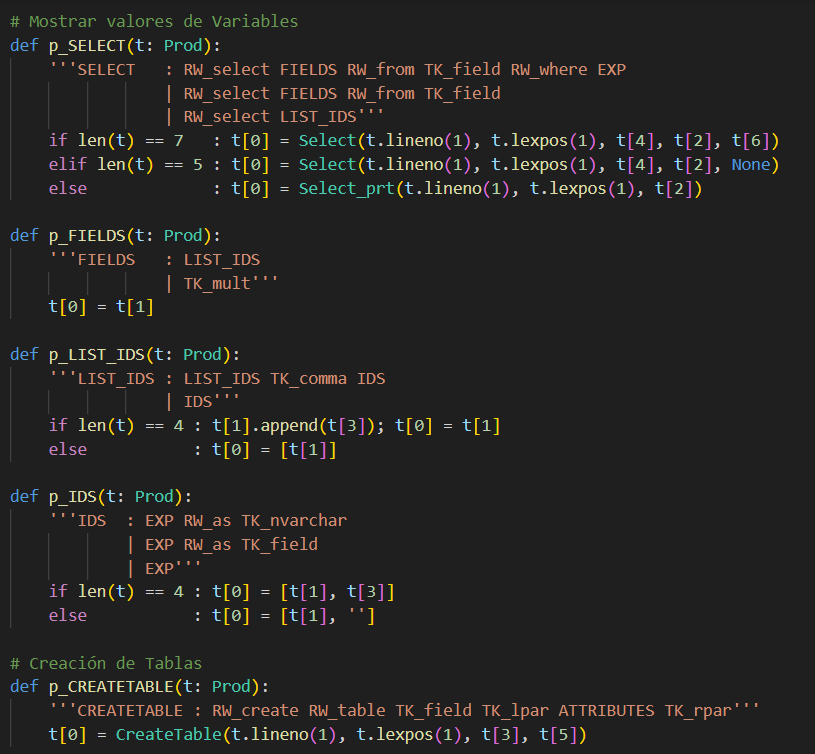
**ANALIZADOR SINTACTICO(PARSER)**

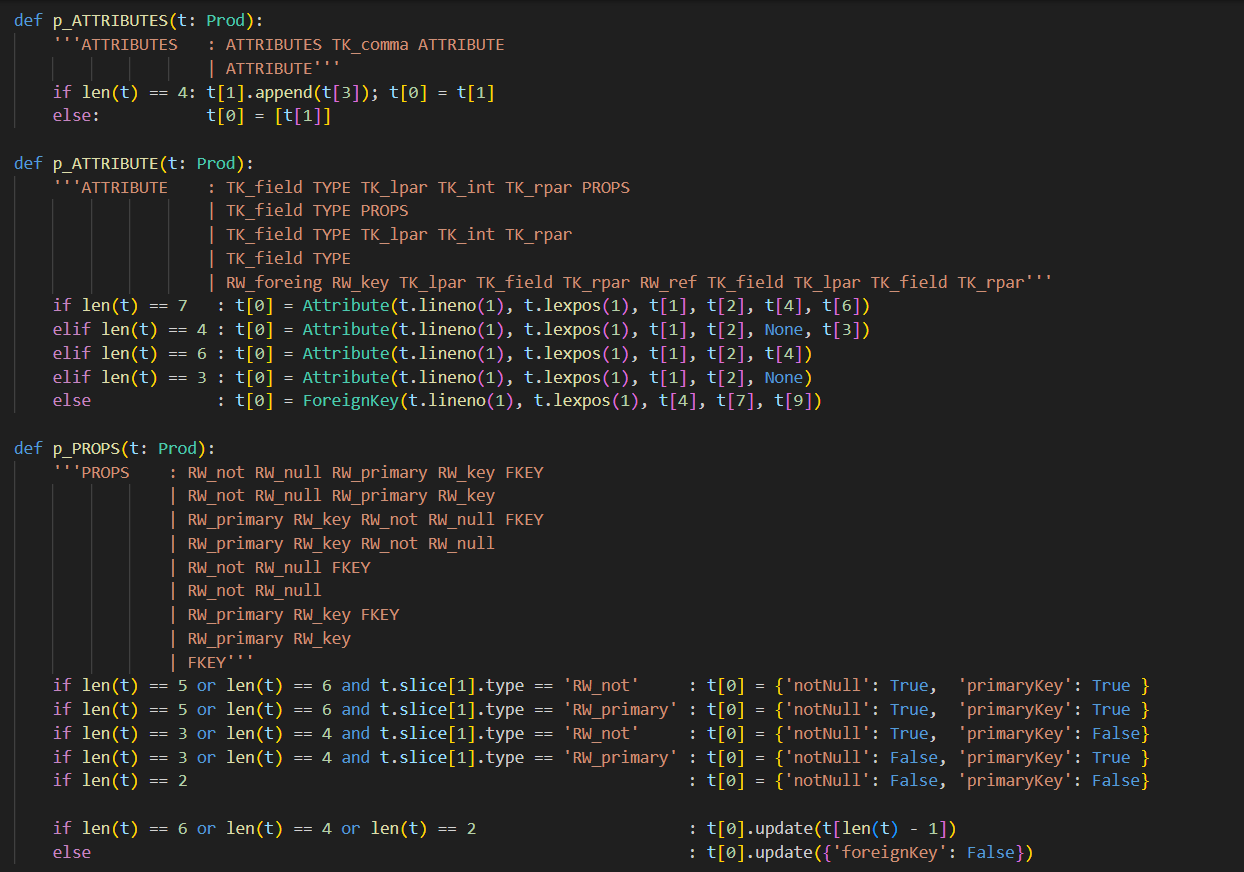
**Gramática:** Creación de las reglas sintácticas que el programa aceptará y leerá durante la ejecución de este mismo.

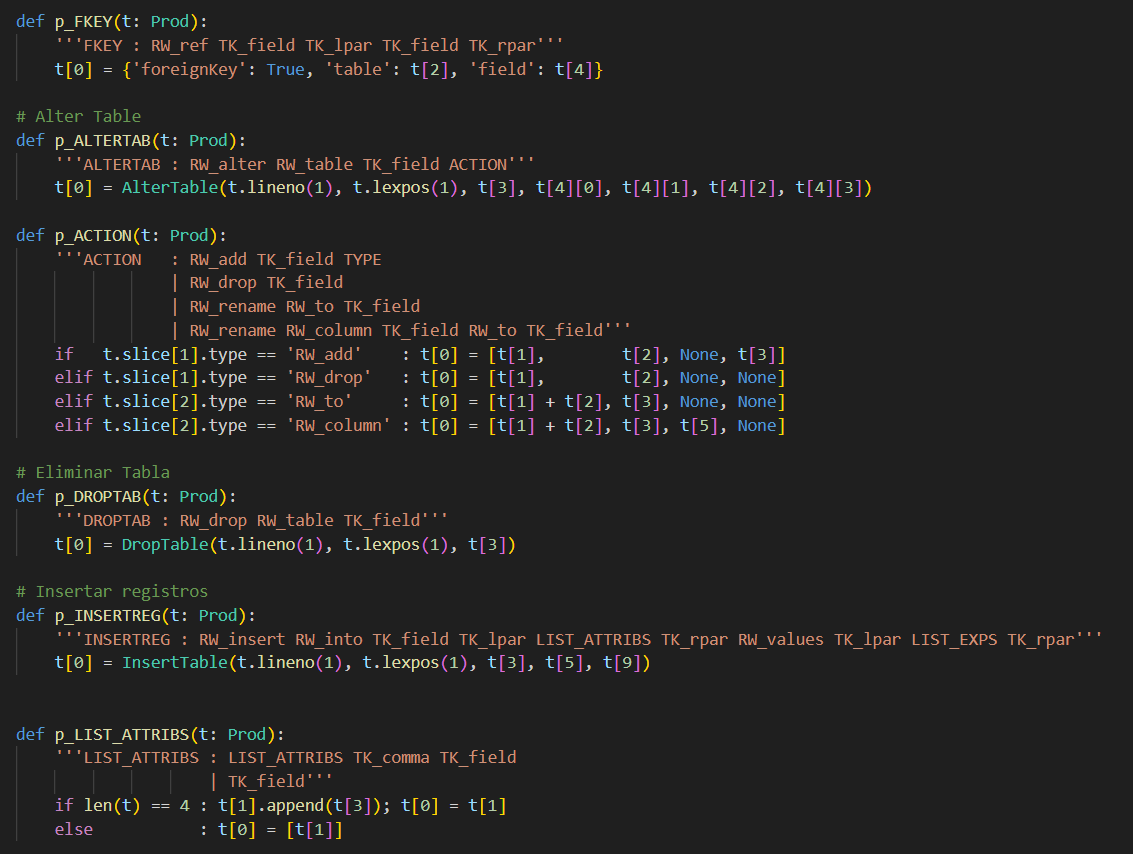


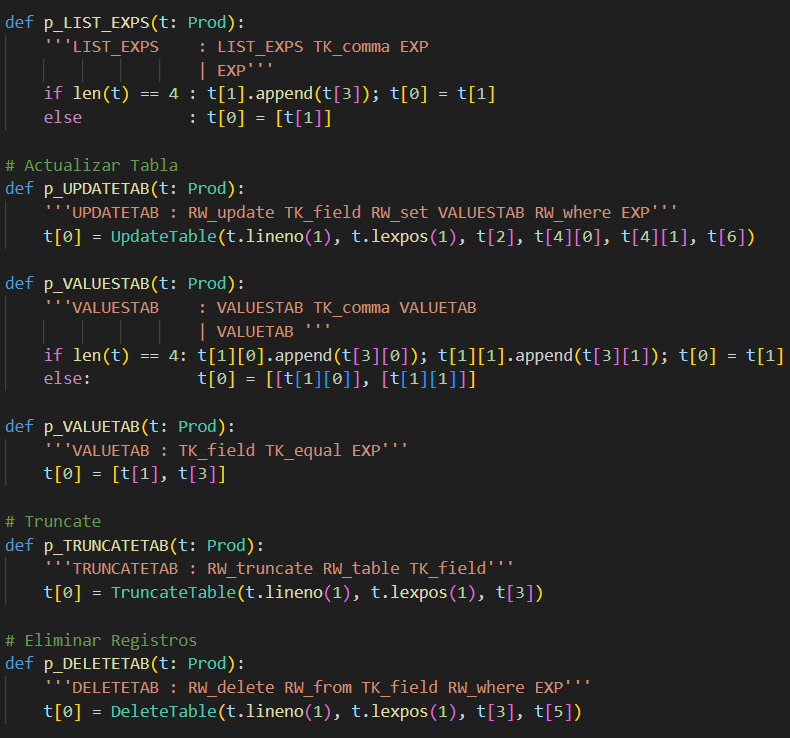


****

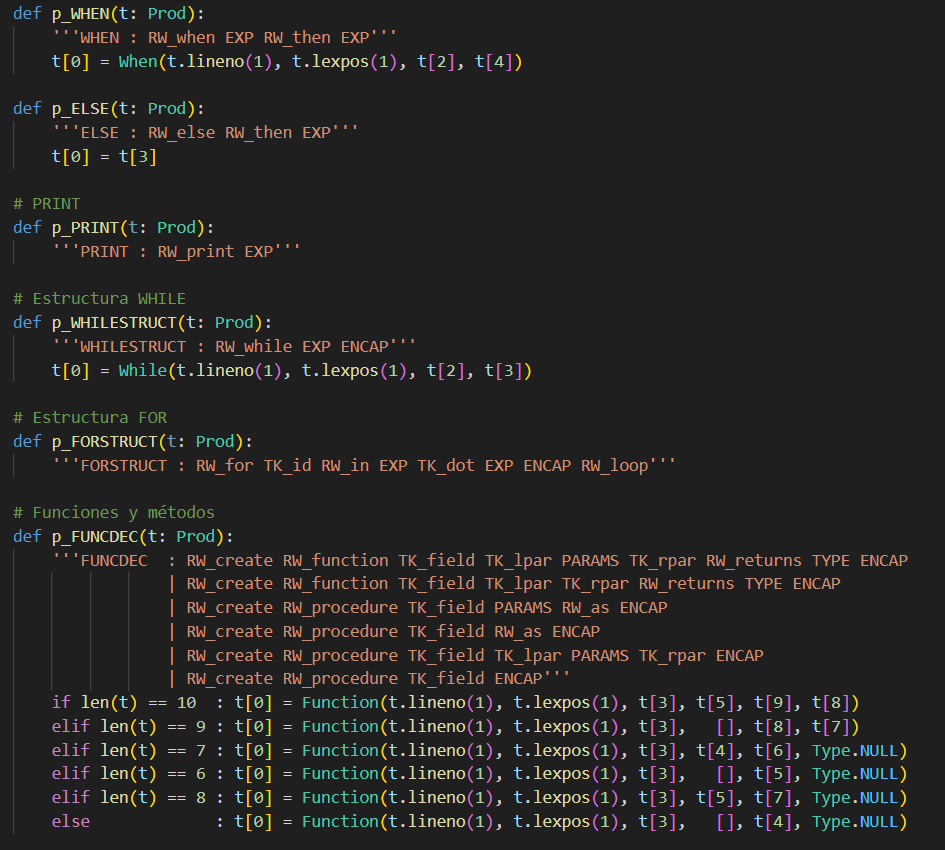
****

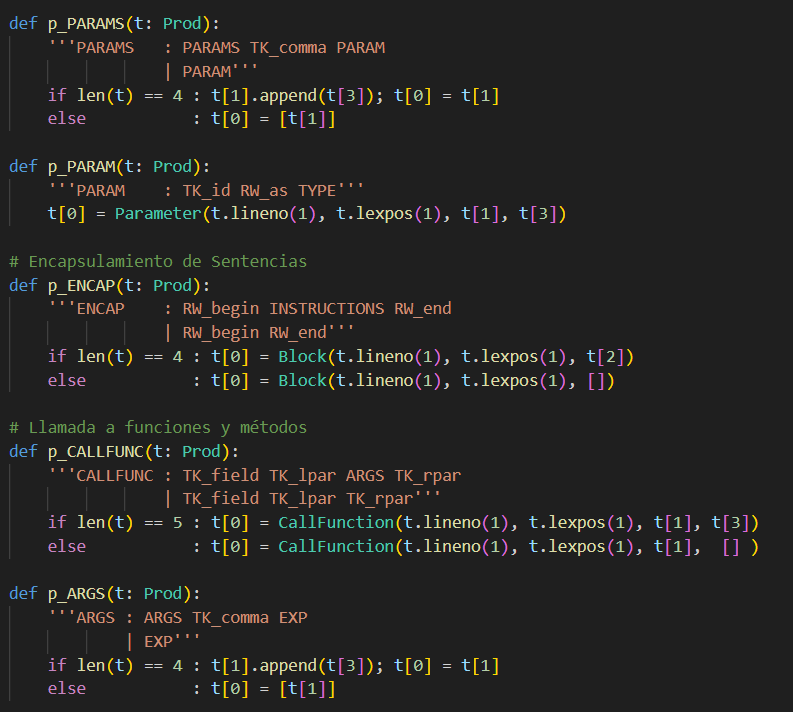
****

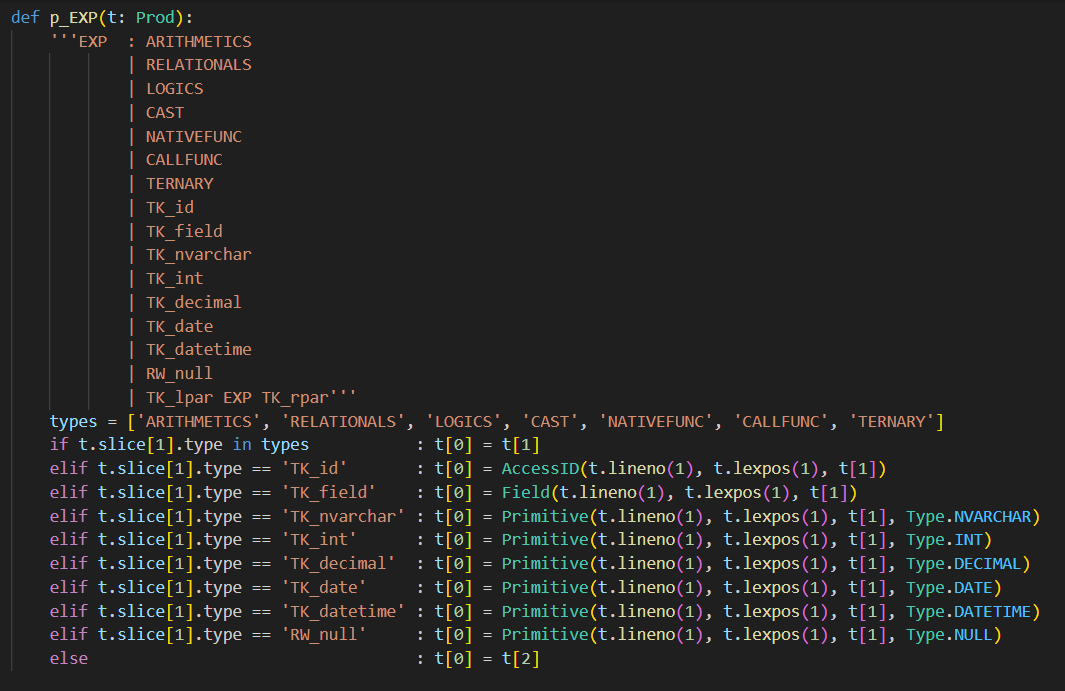
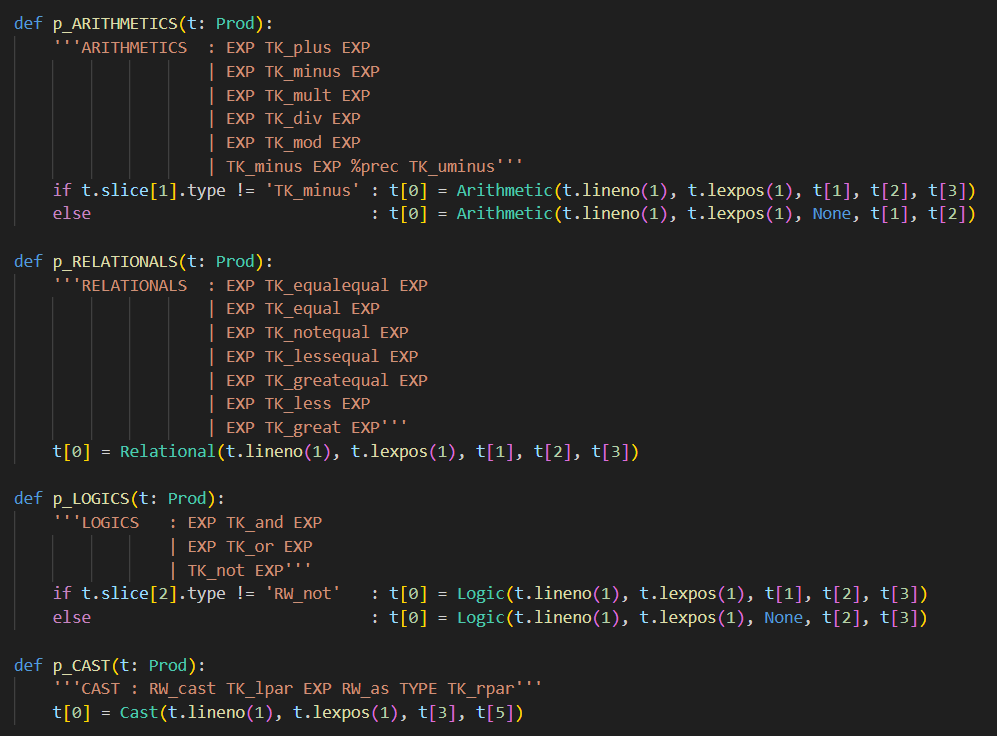
****

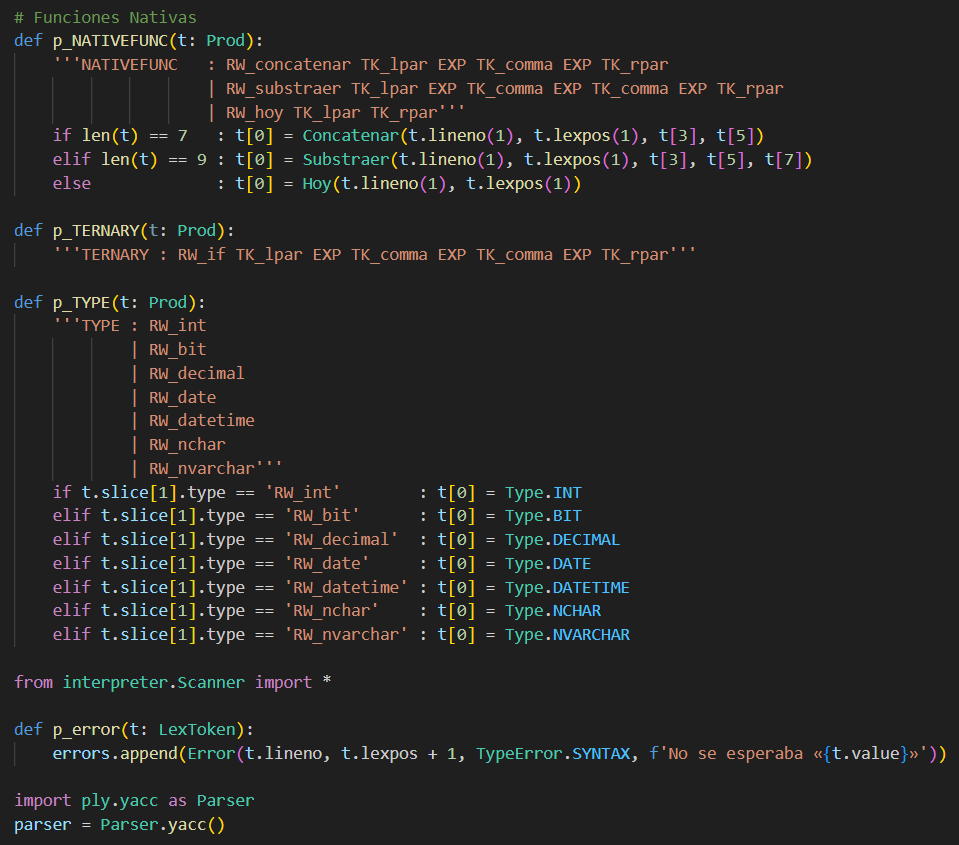
****

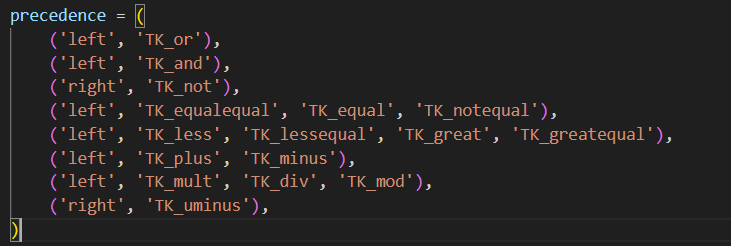




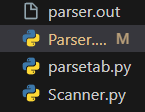






**Precedencias**: Durante la ejecución el programa tomara algunas reglas léxicas (Token’s) como prioridad.

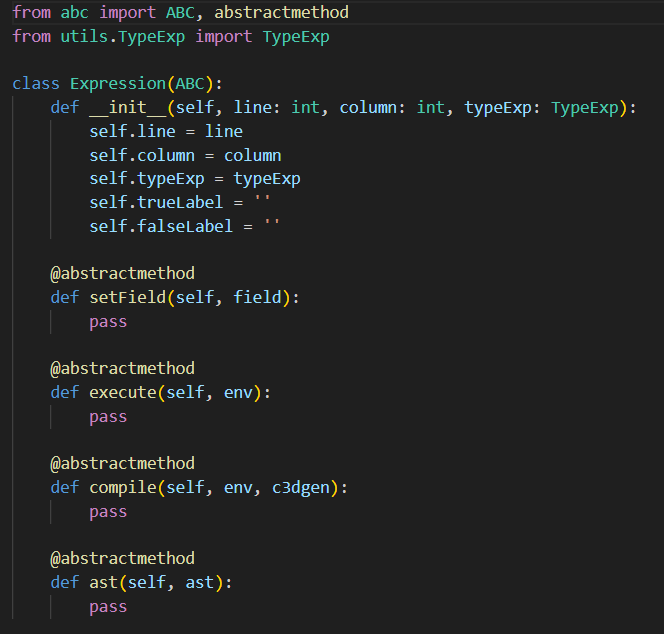
Archivos generados por la herramienta PLY:



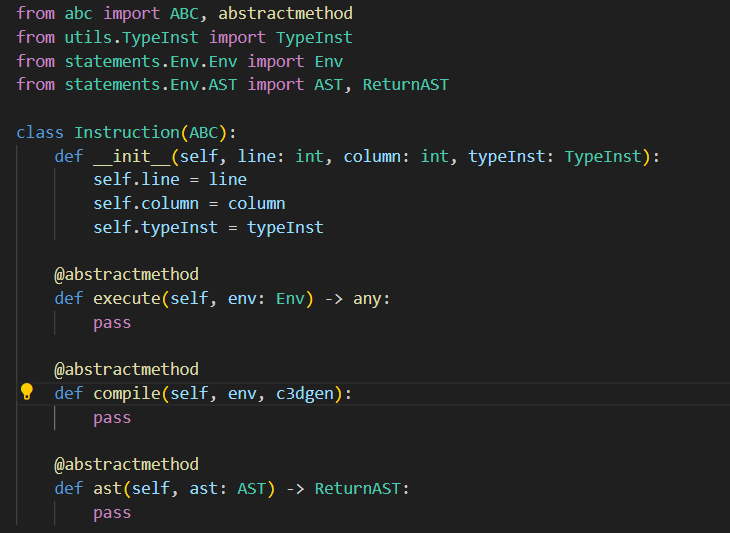
**PATRON INTERPRETE**

Para la creación de este proyecto se utilizo el patrón interprete como modelo a seguir para la implementación de las deferentes clases donde se trabajaron las funcionalidades de este mismo.

**Clase Expresión:** Se construyo una clase padre llamada expresión que se implementará en las expresiones que el programa podrá ejecutar durante la interacción con este mismo.

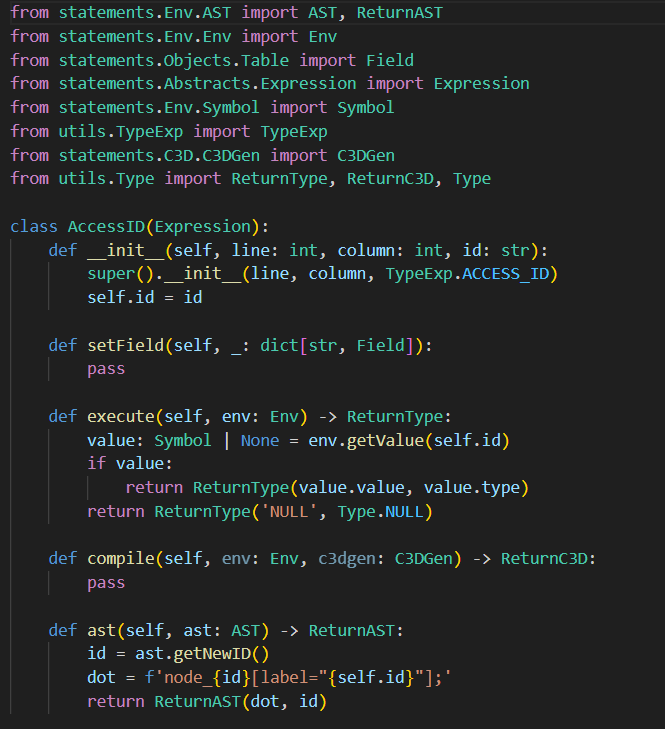


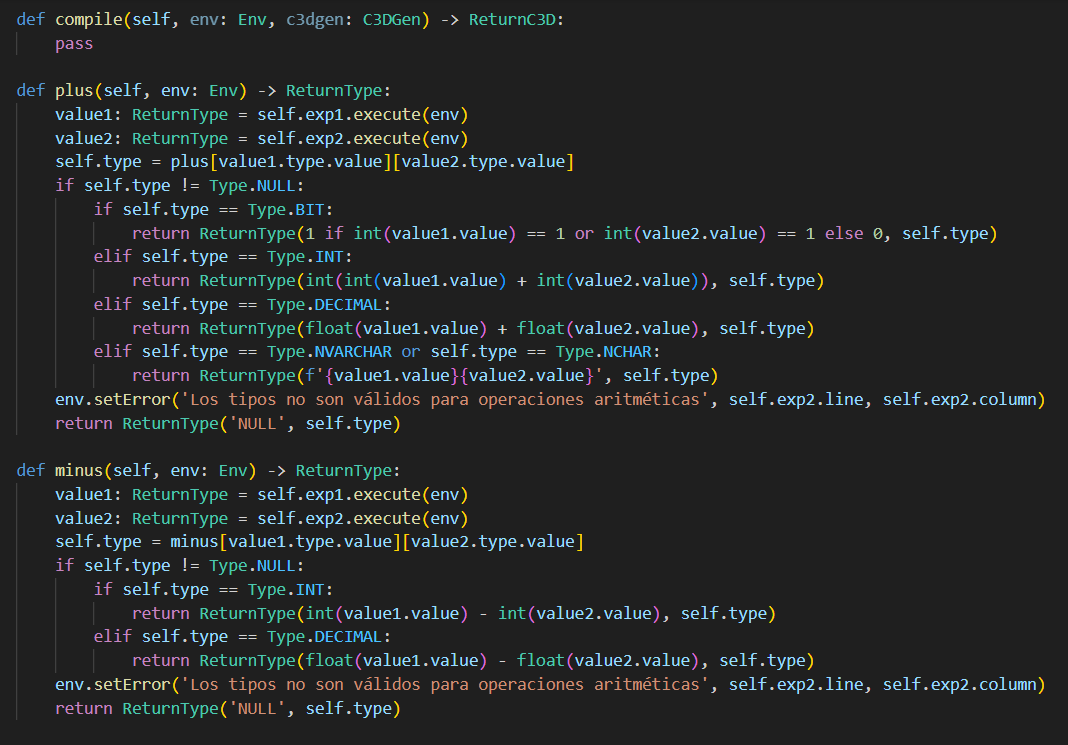
**Clase Instrucción:** Se construyo una clase padre llamada Instrucción que se implementará en las Instrucciones que el programa podrá ejecutar durante la interacción con este mismo.

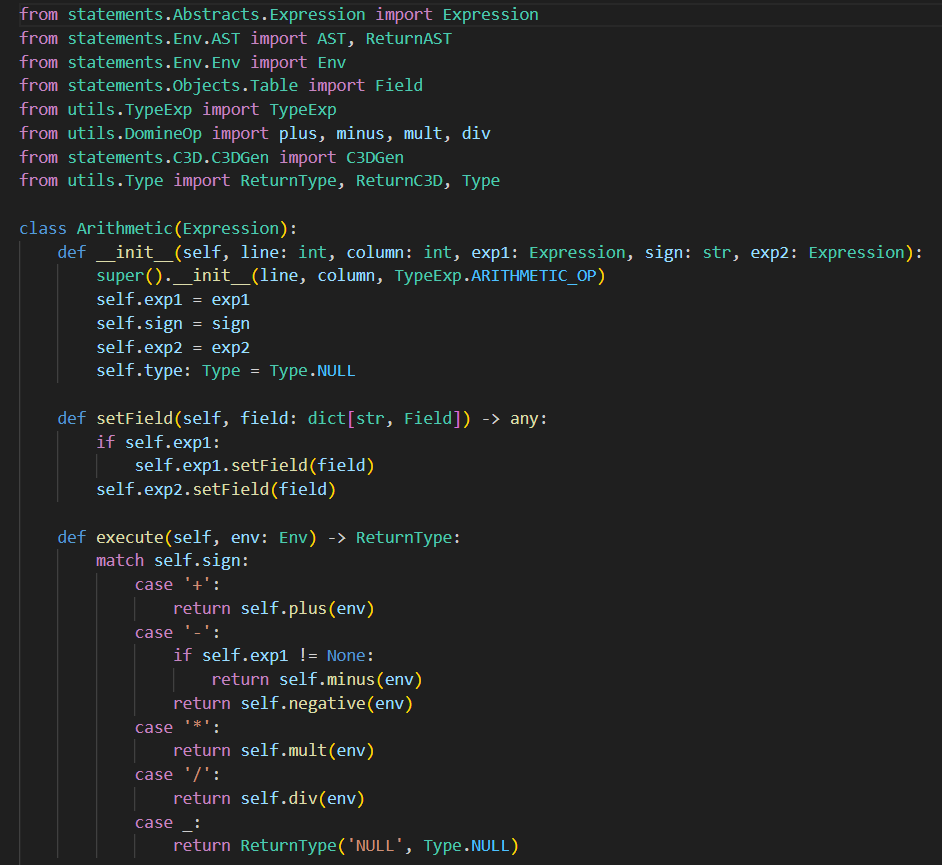
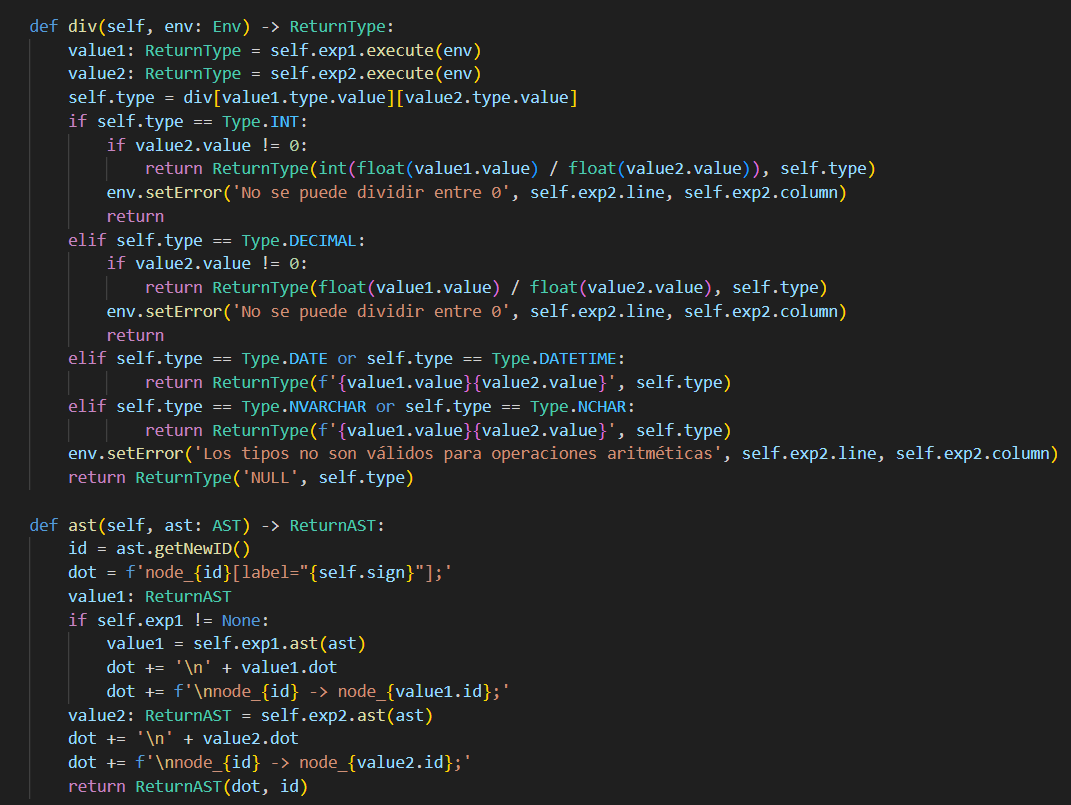


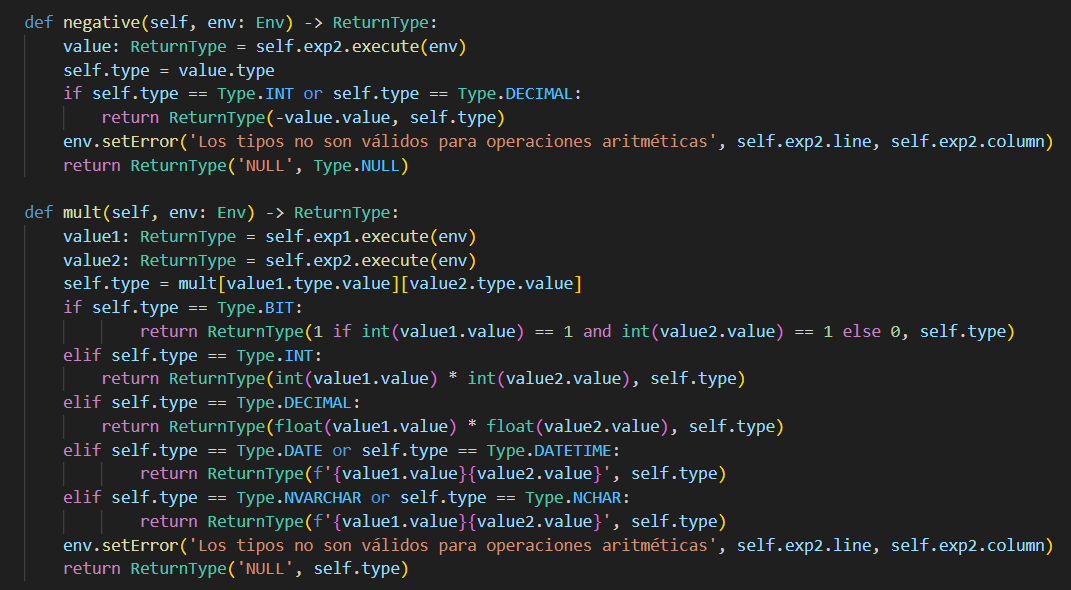
Las expresiones que el programa utilizara son las siguientes:

AccessID

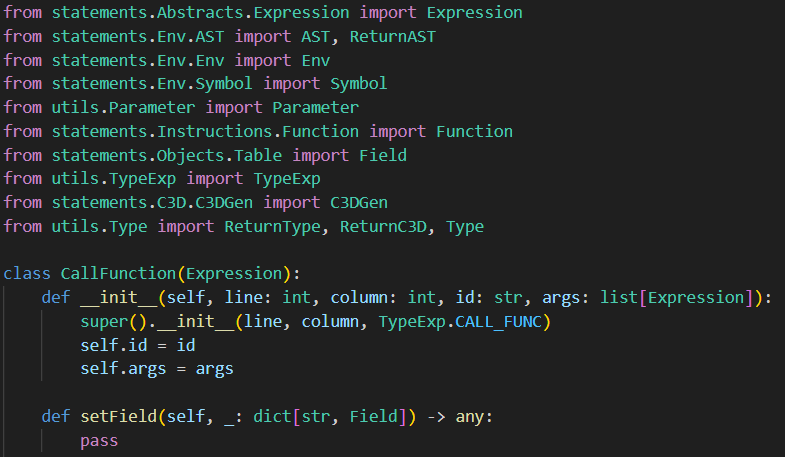


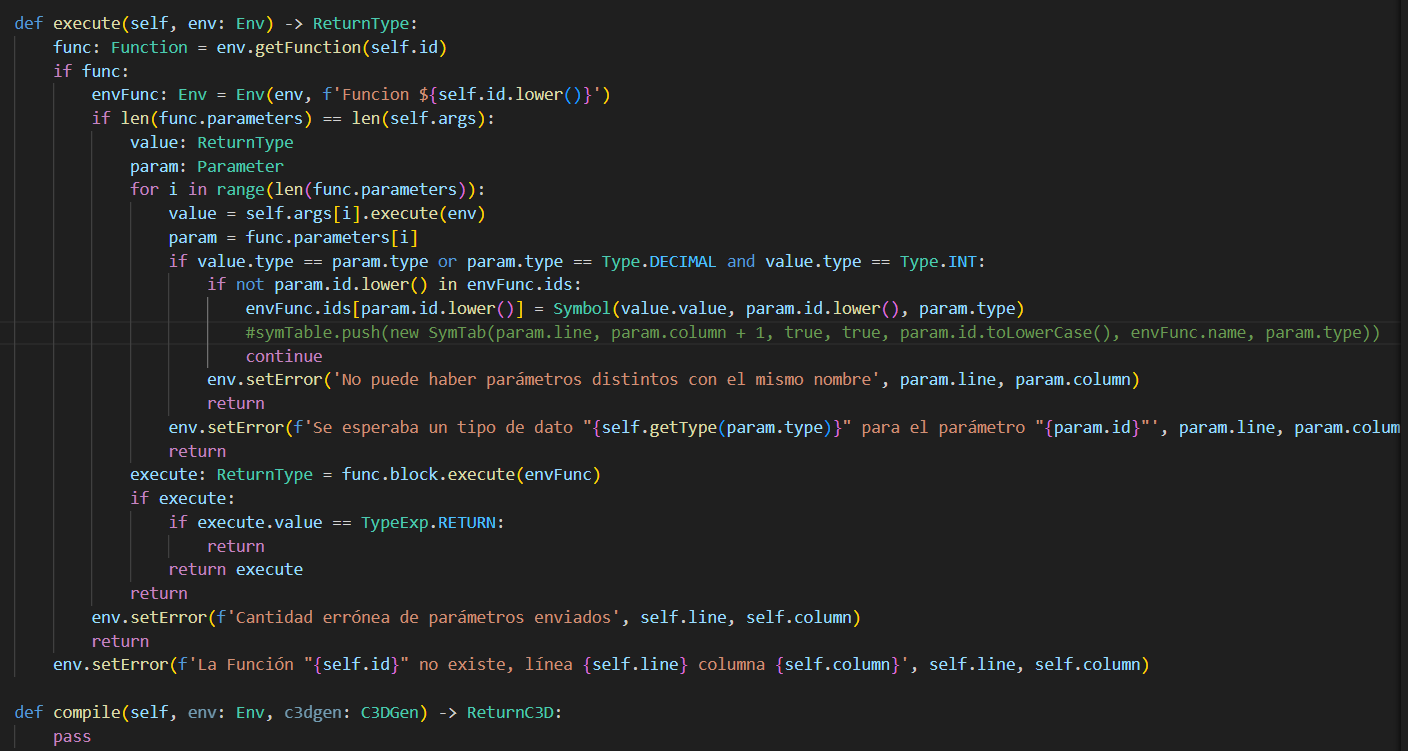
Arithmetic

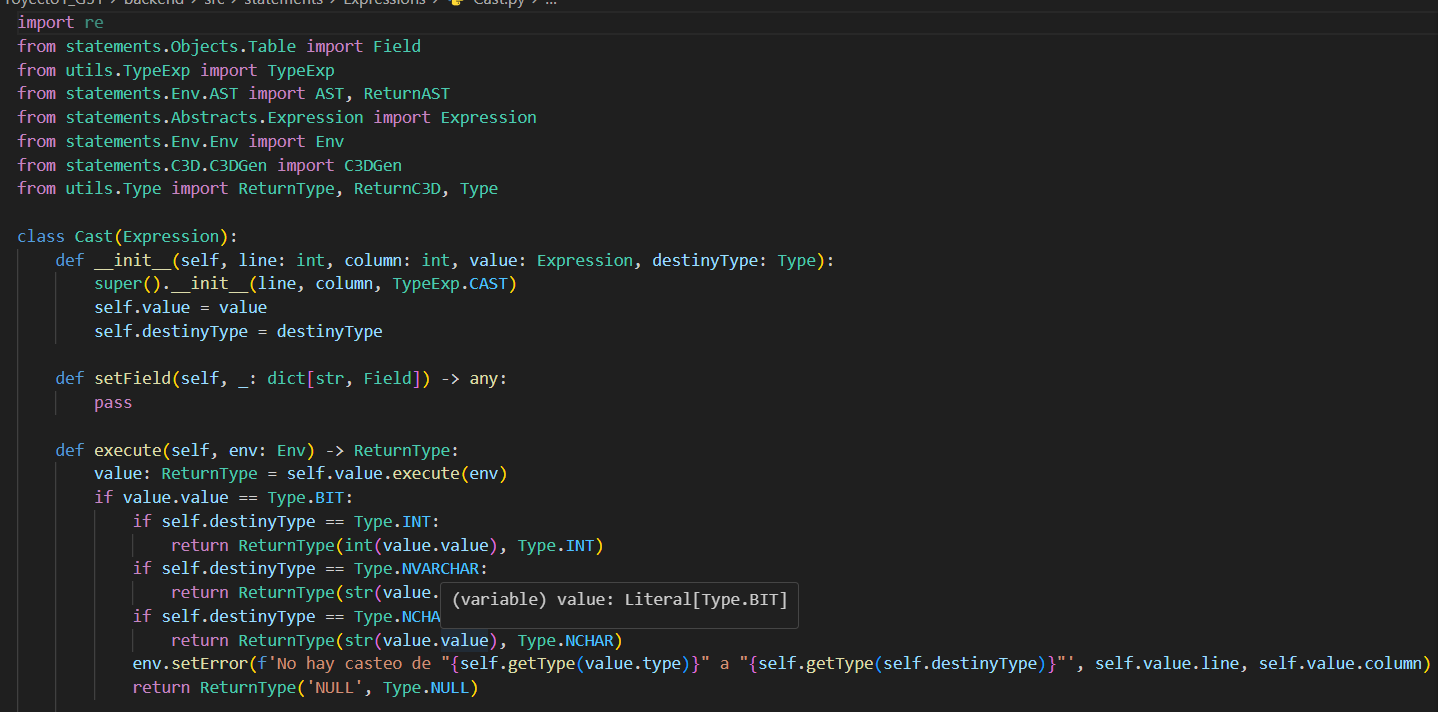


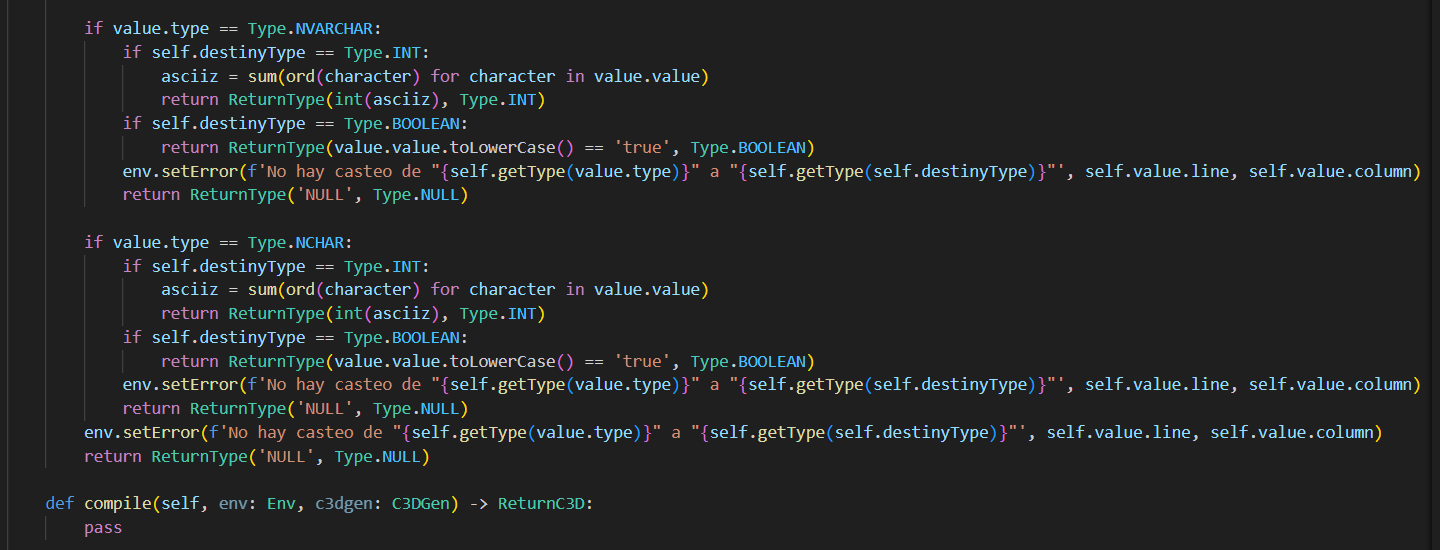


CallFunction



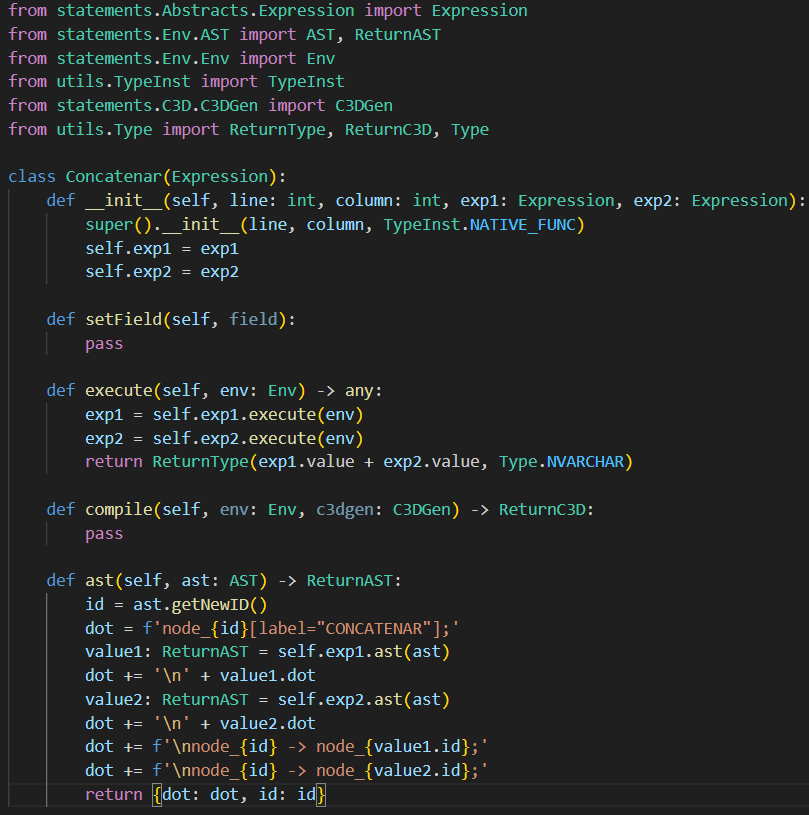


Cast

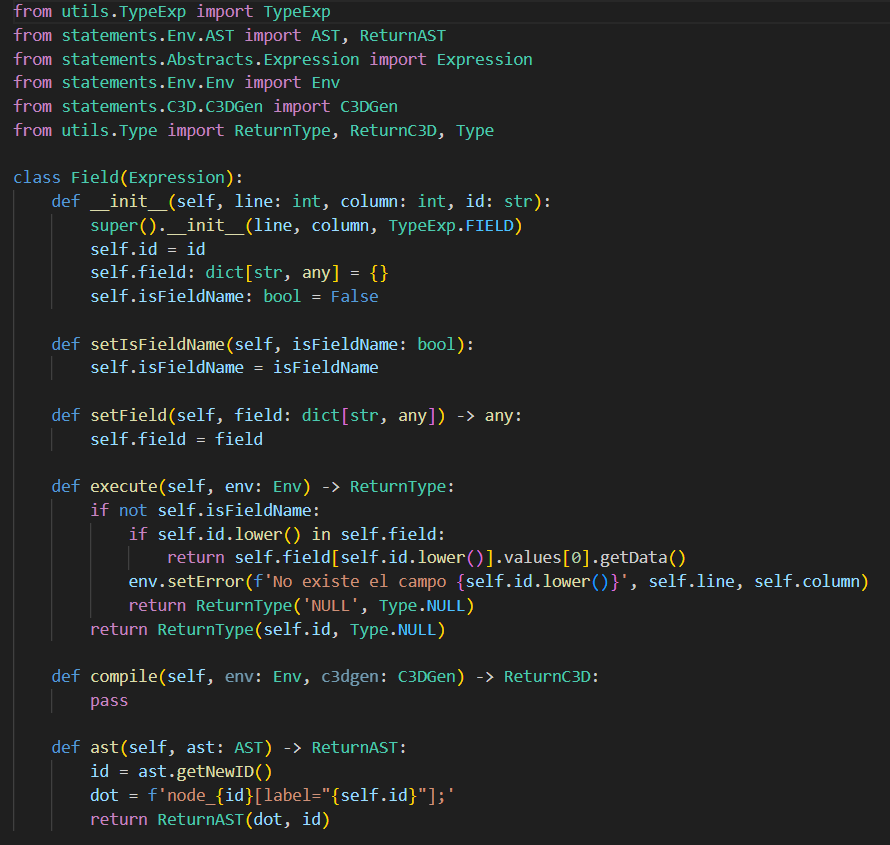




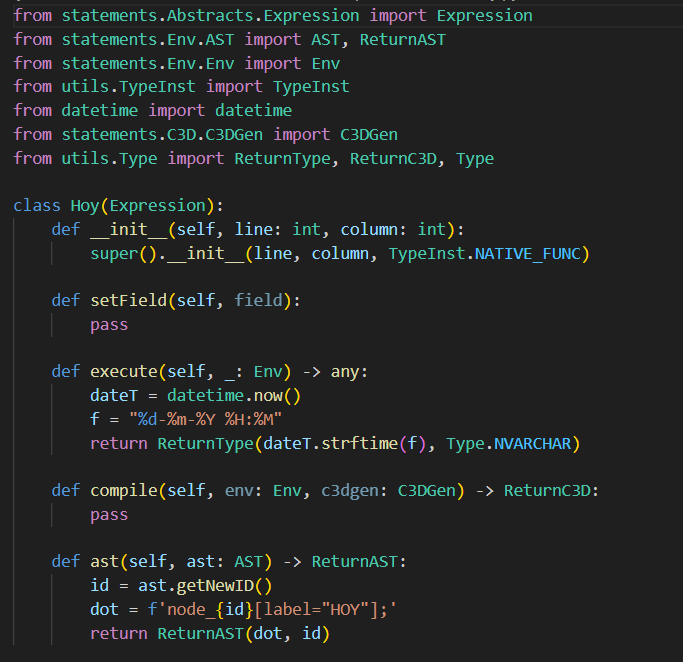
Concatenar



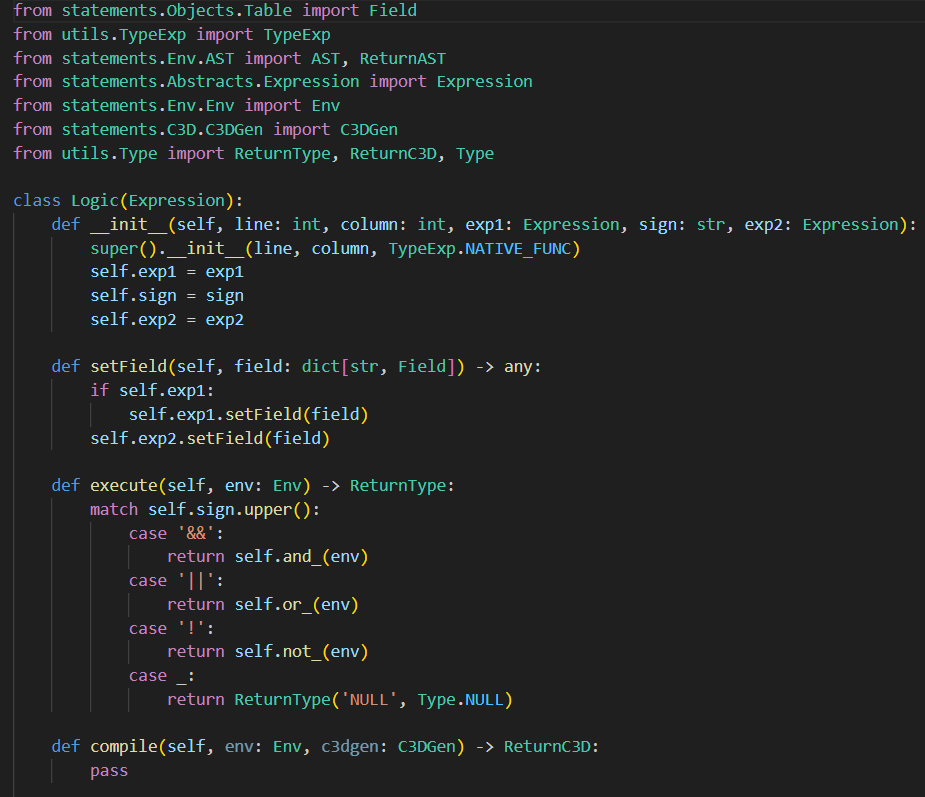
Field

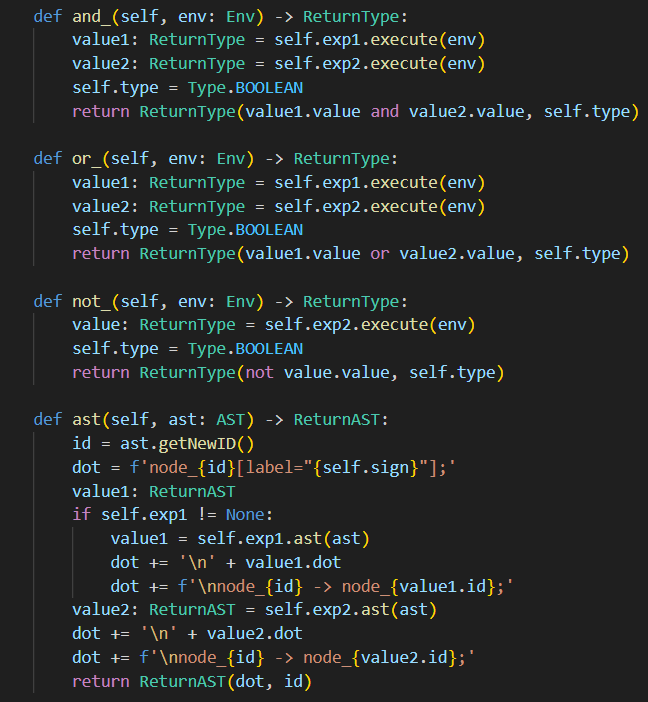


Hoy

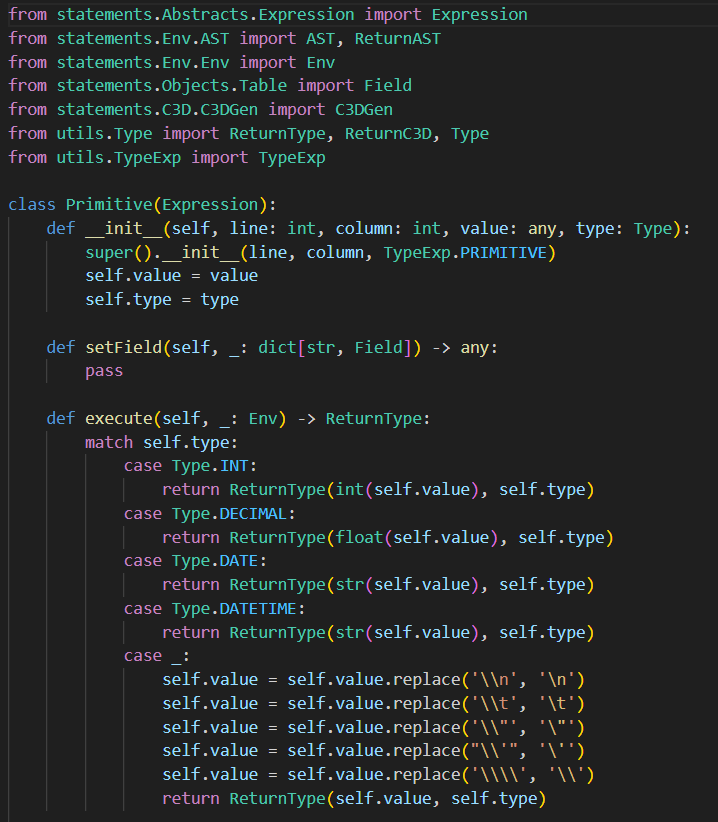


Logic

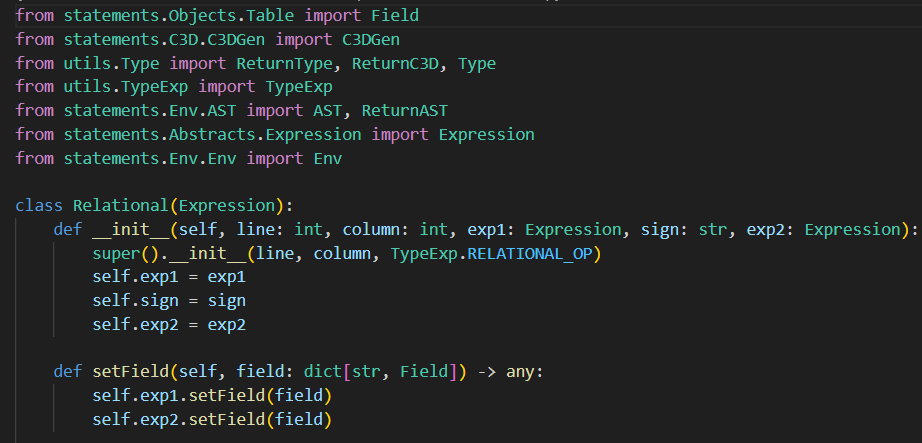


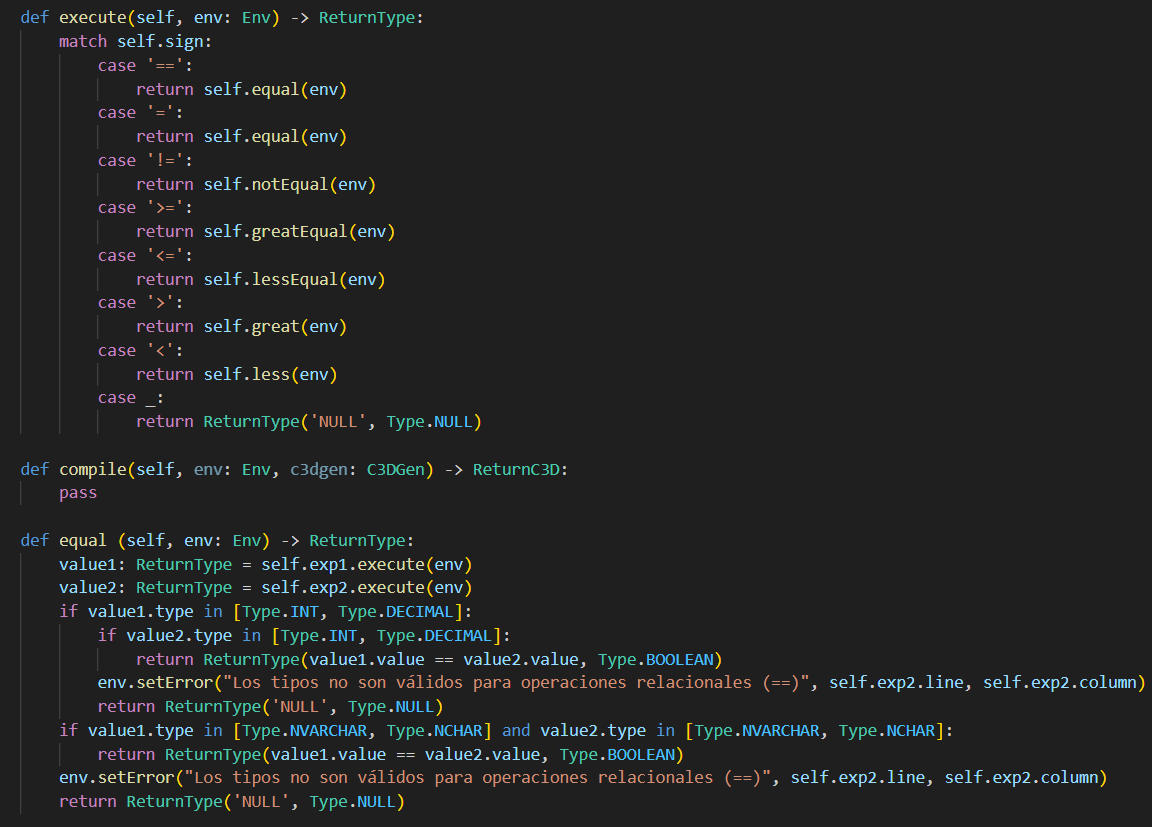


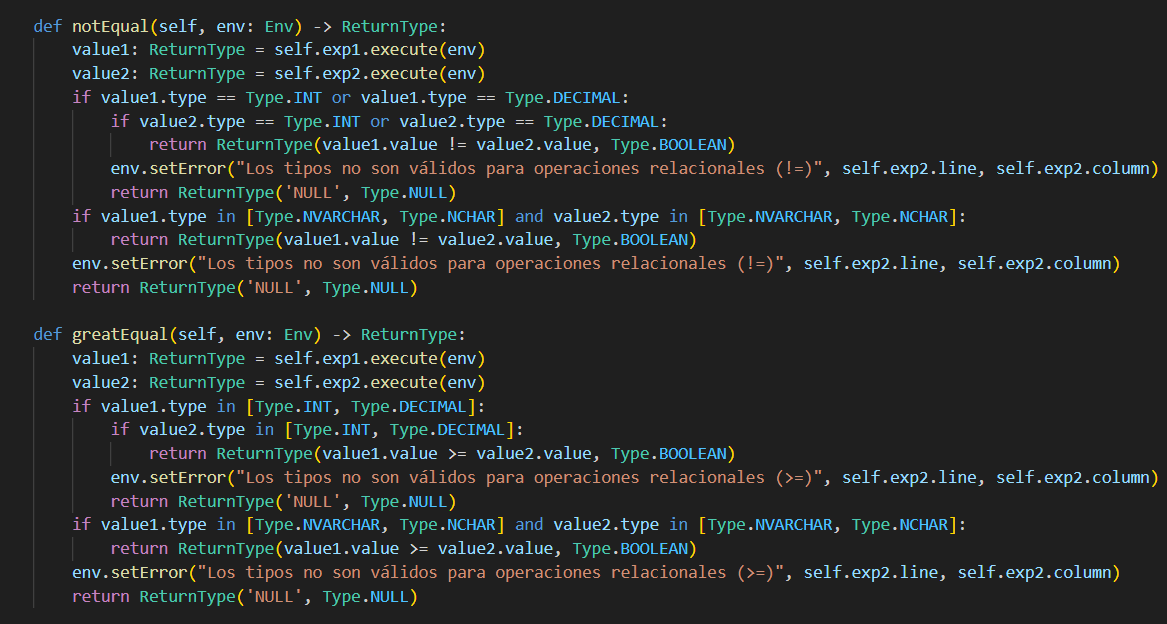
Primitive

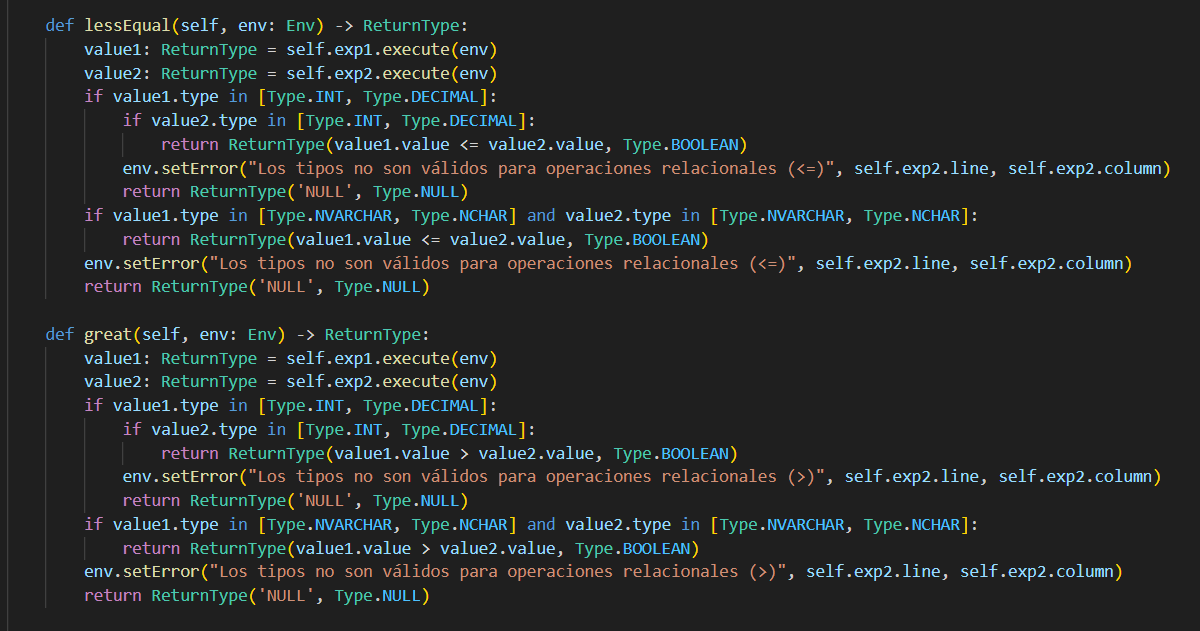


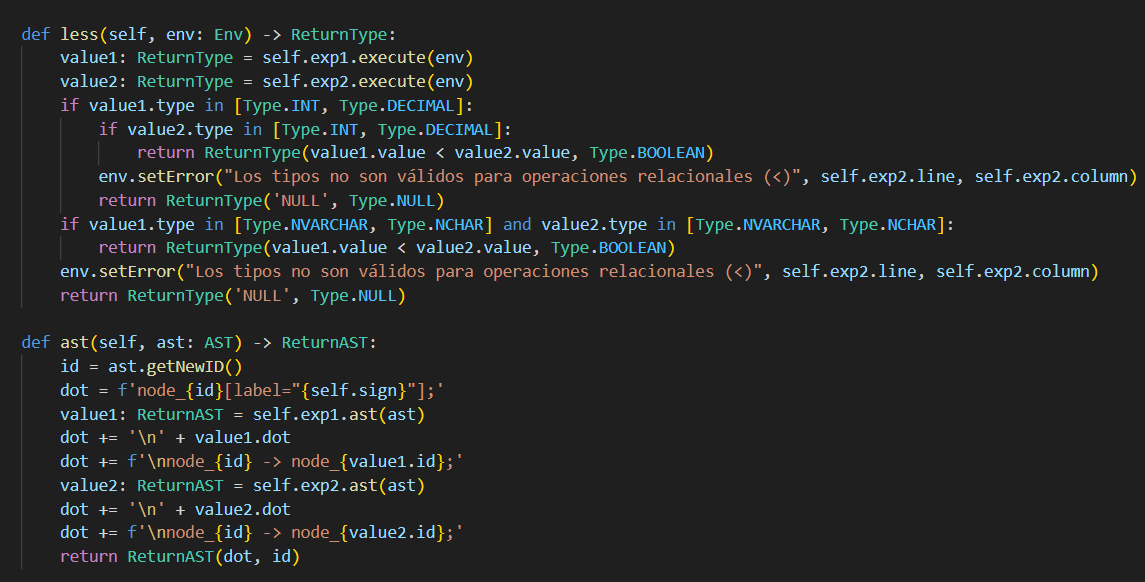


Relacional

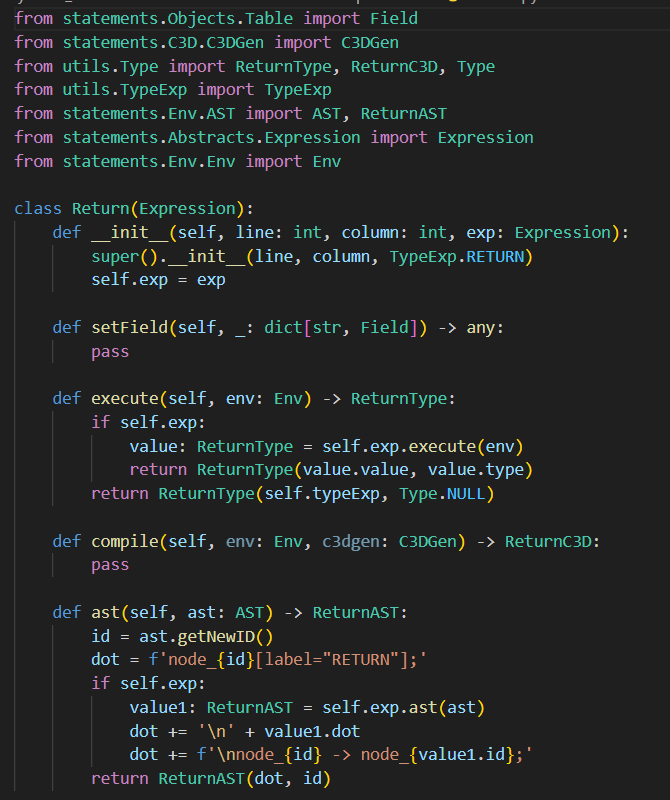


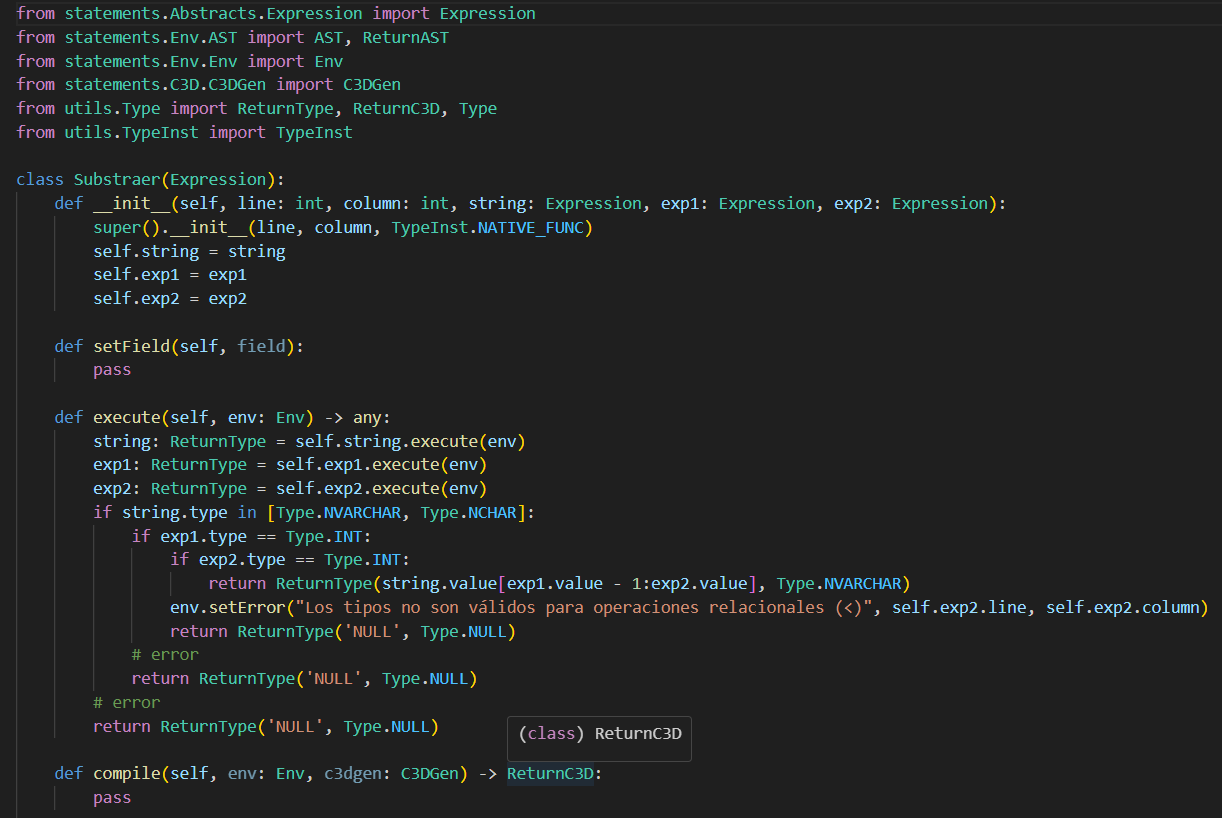


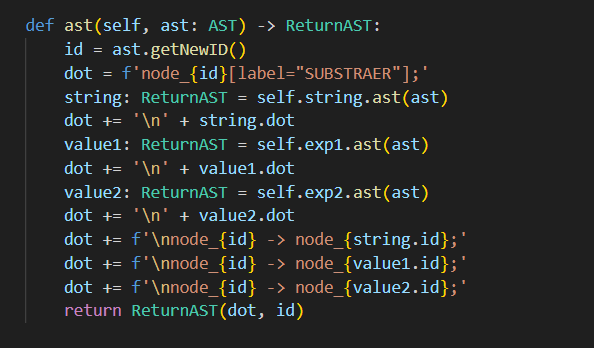




Return



Subtraer

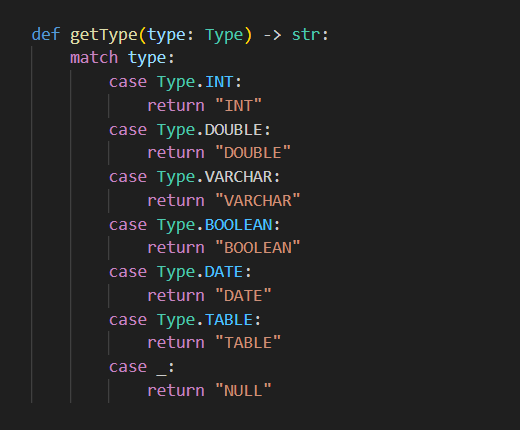


Las Instrucciones que el programa utilizara son las siguientes:

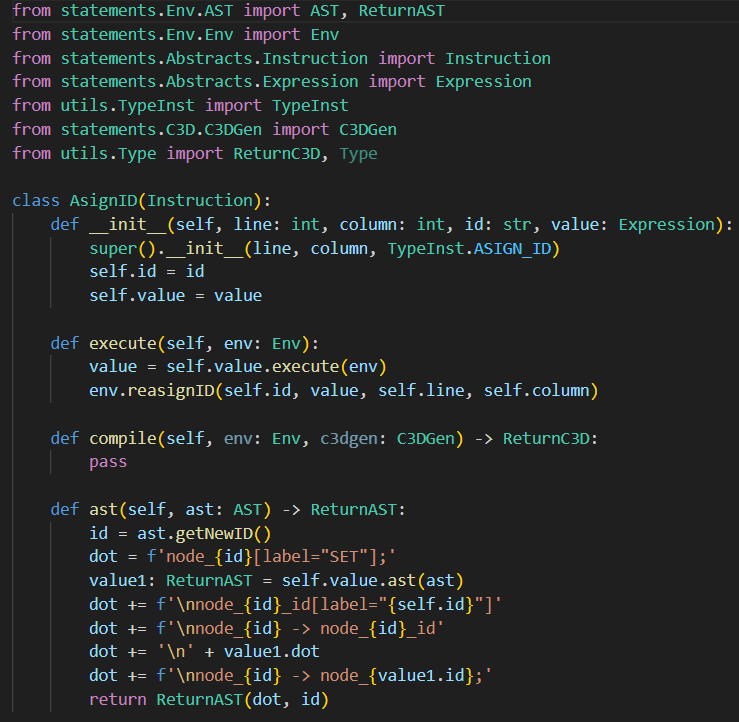
AlterTable







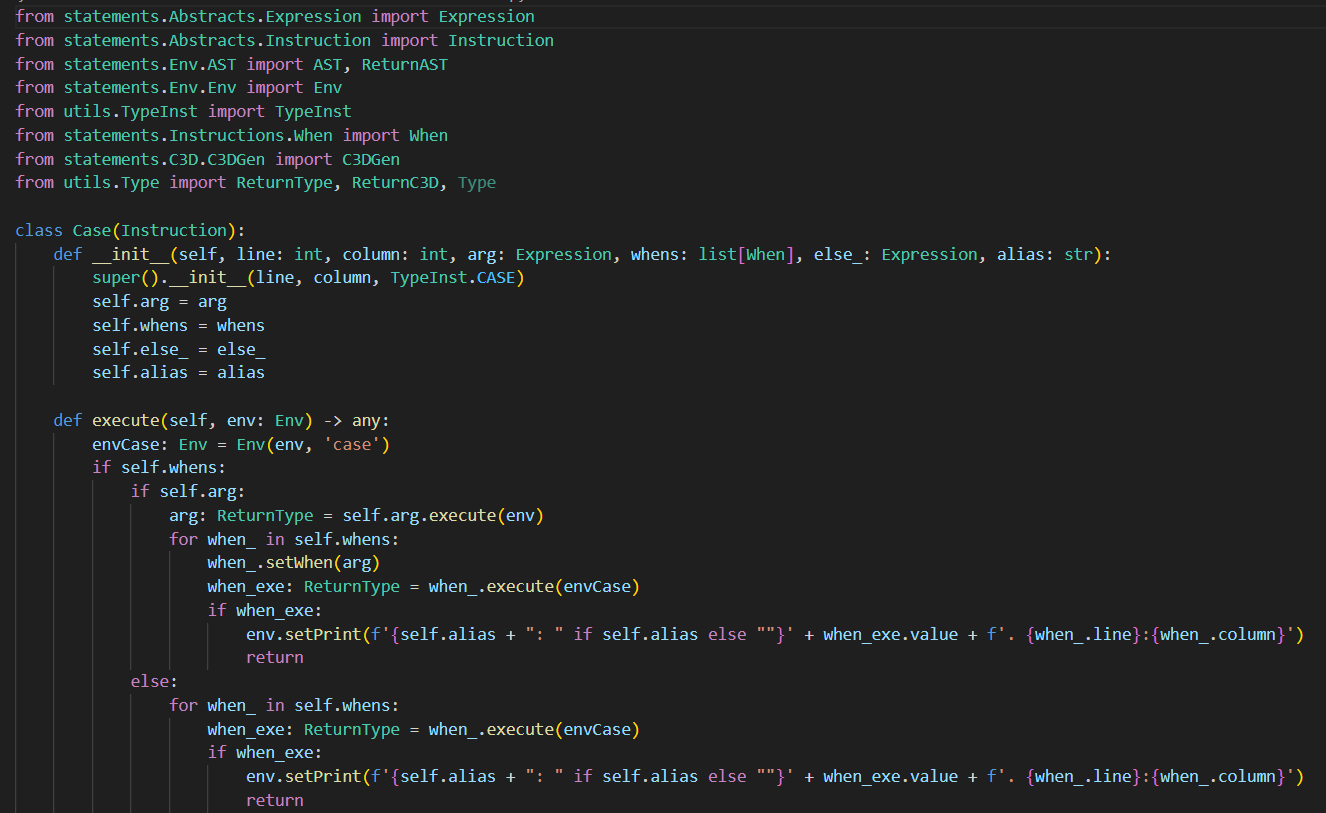
AsignID

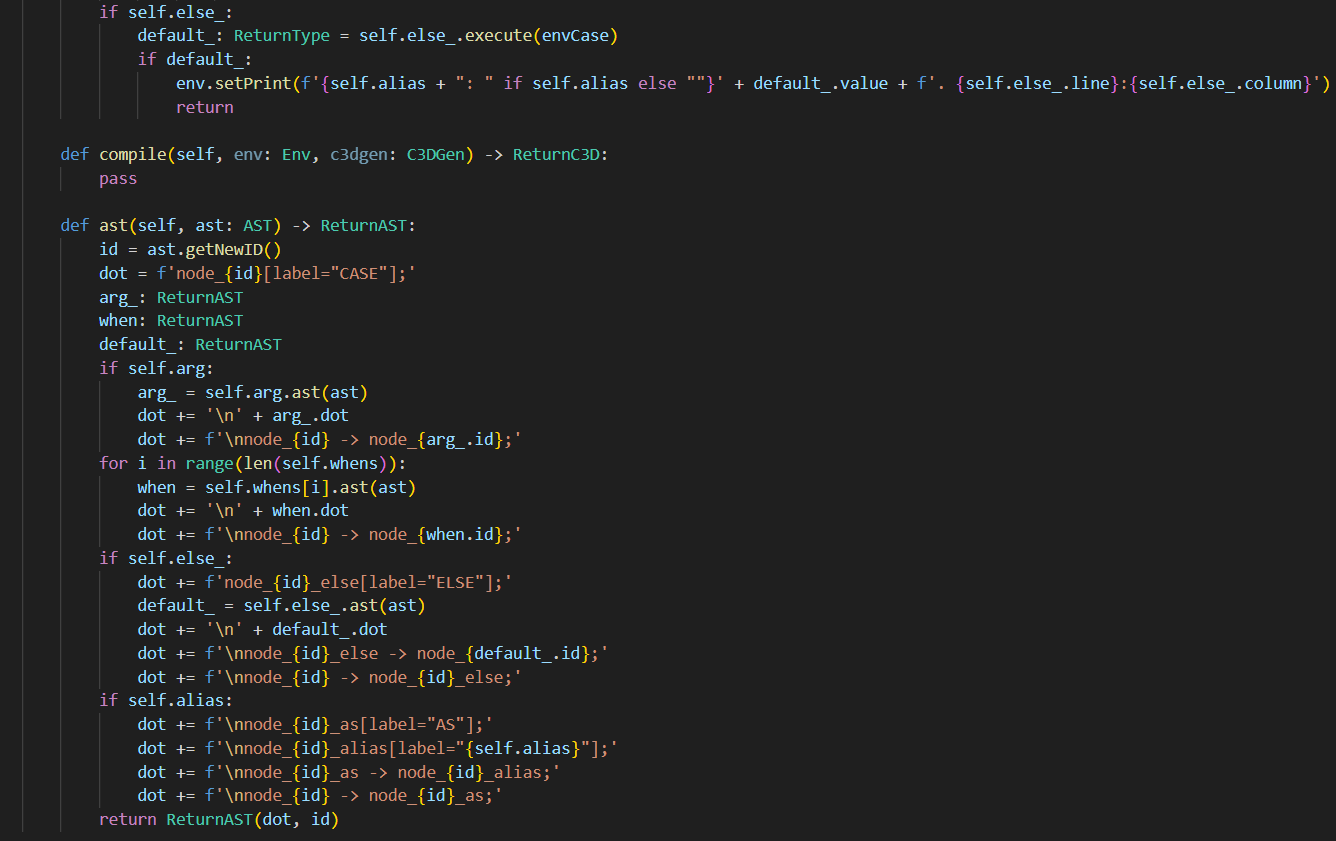


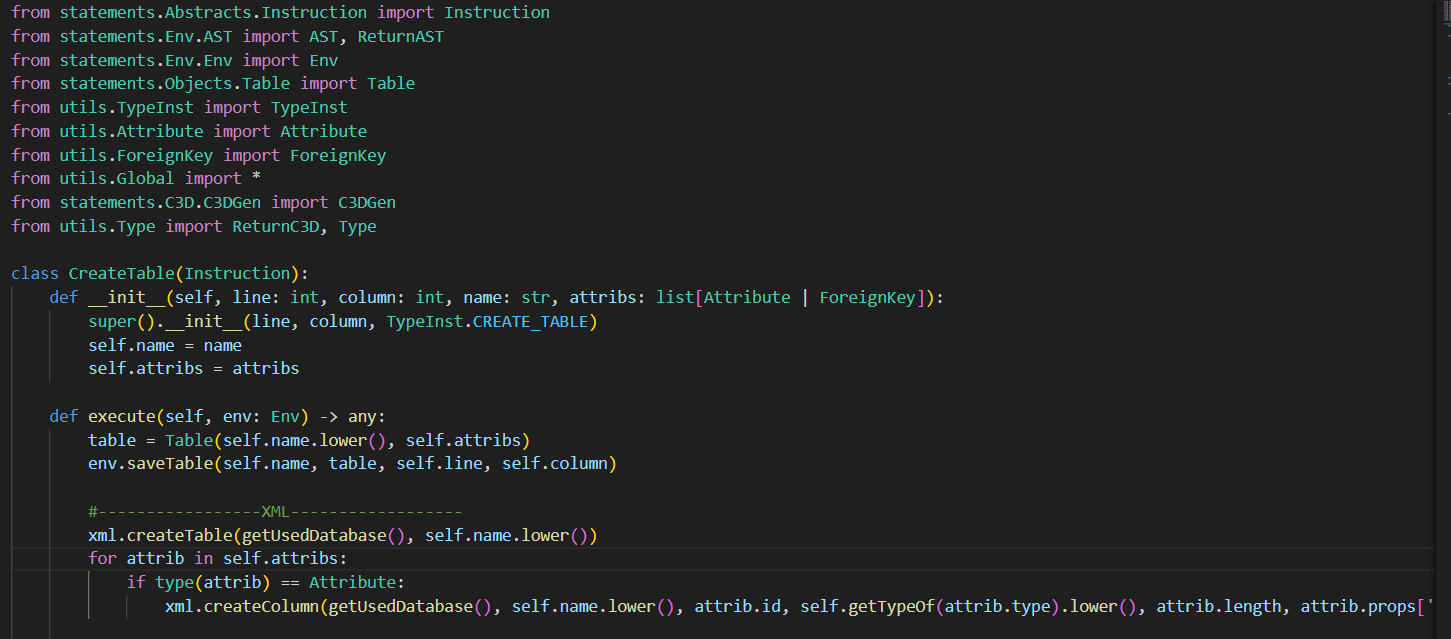
Block



Case

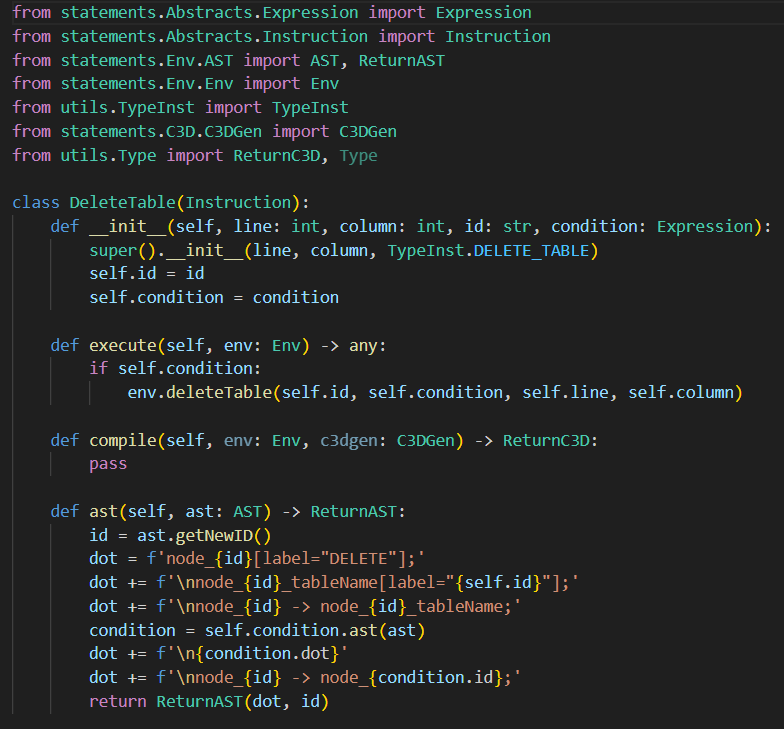




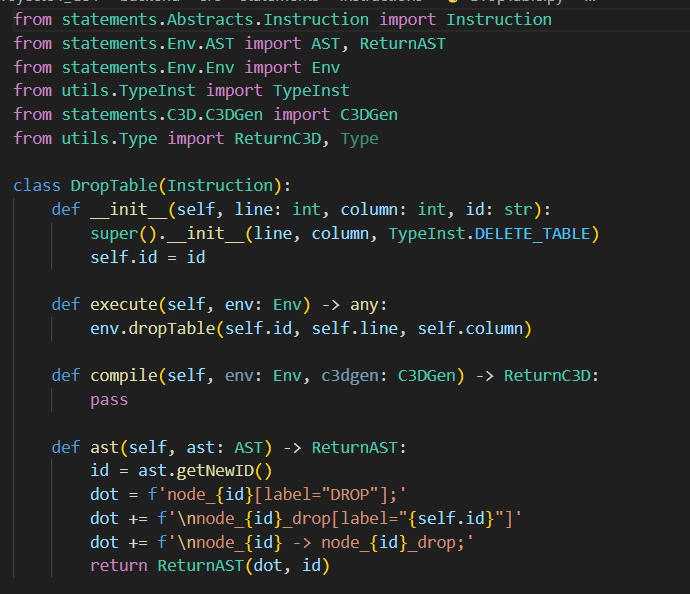
CreateTable



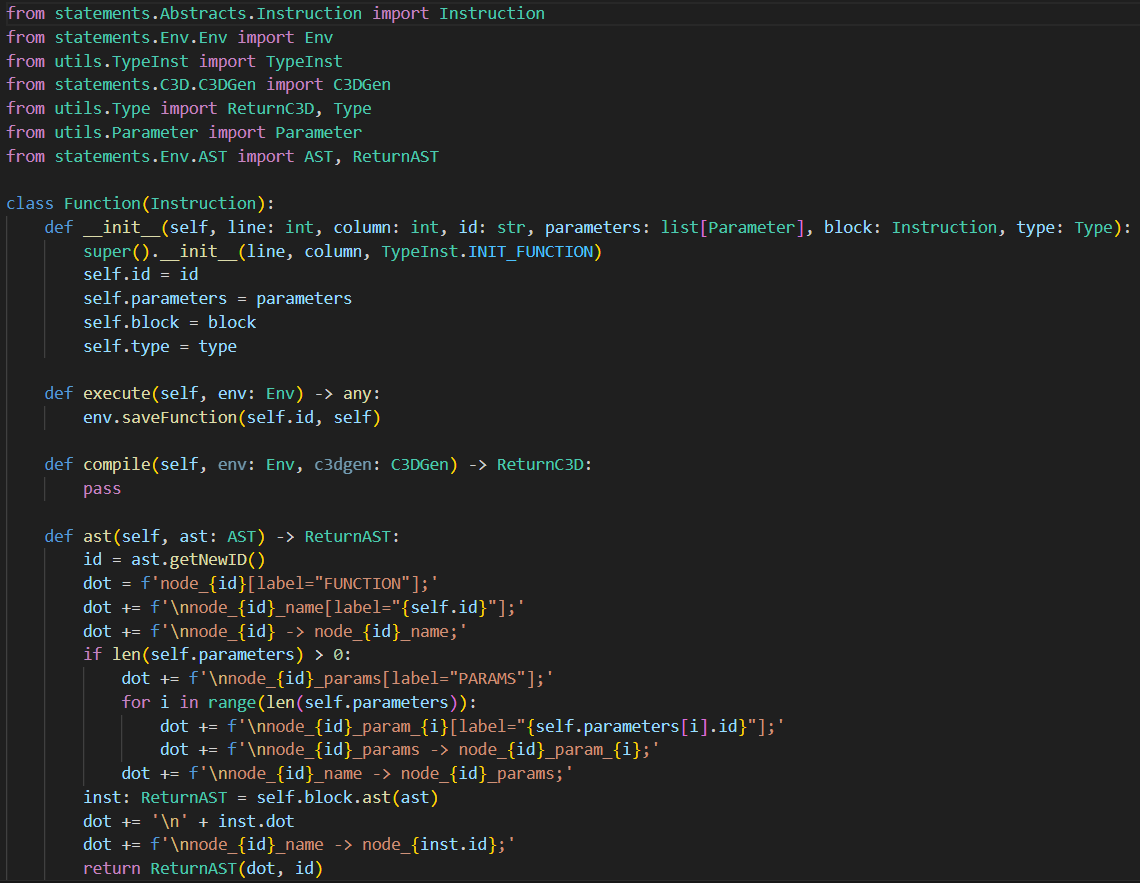
DeleteTable

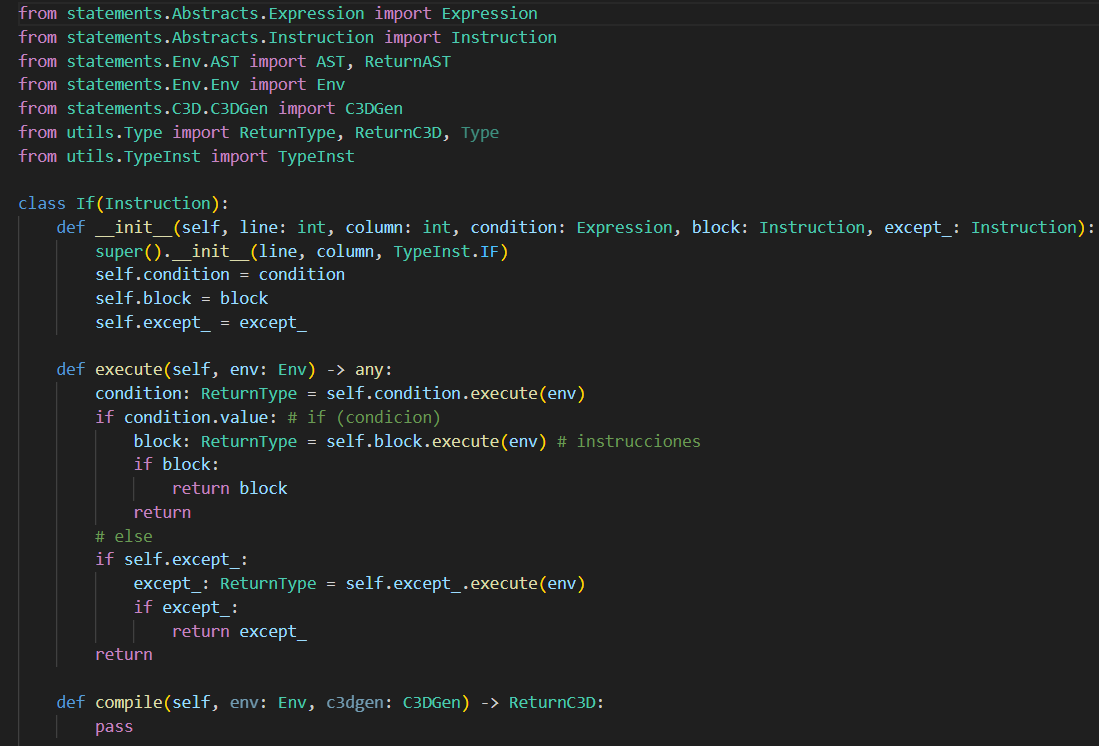


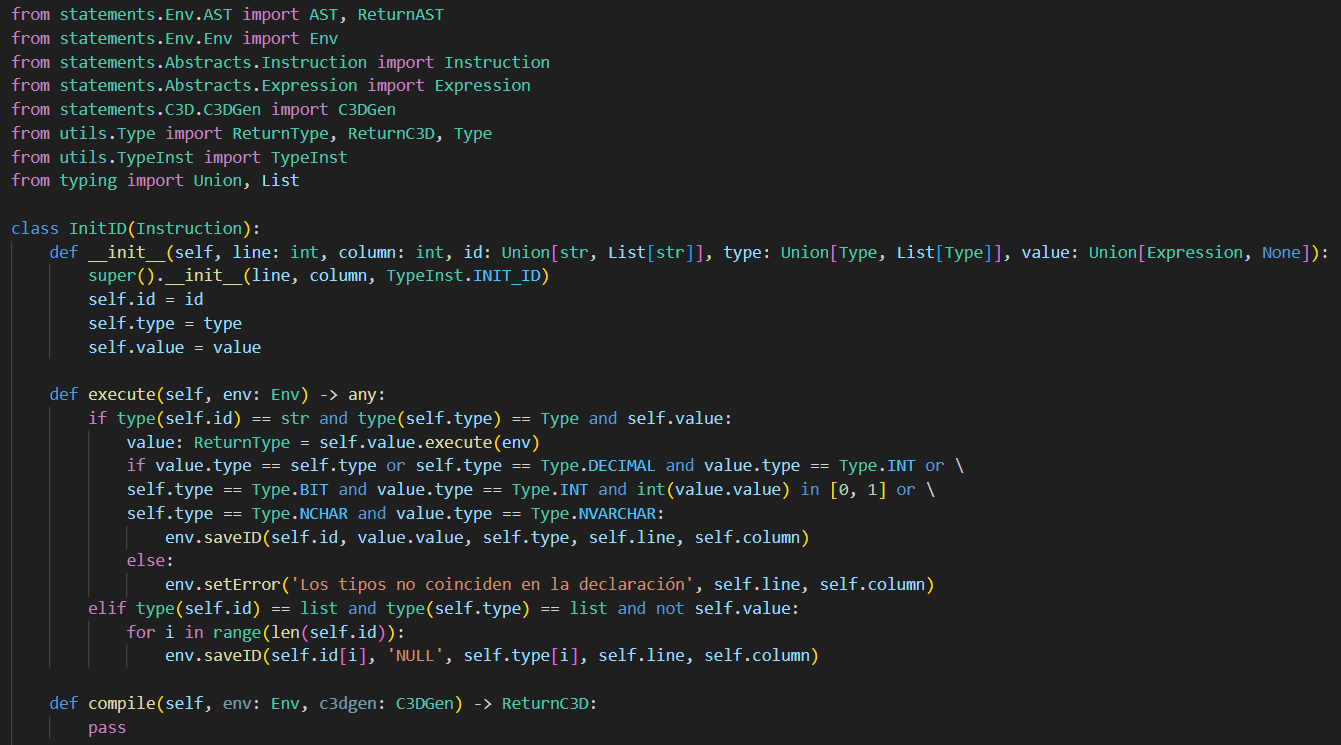
DropTable



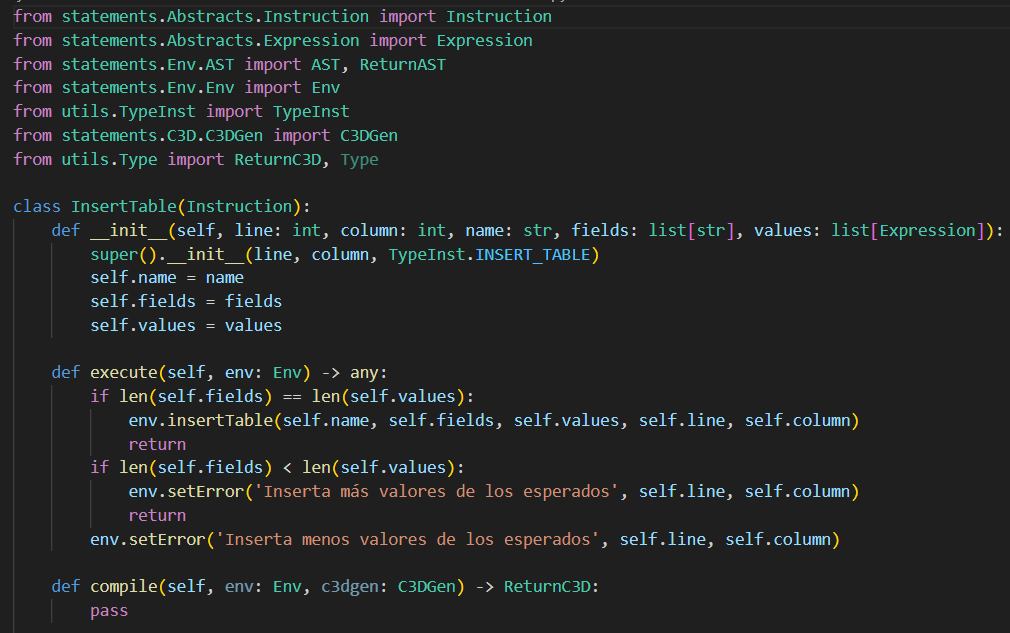
Function



If

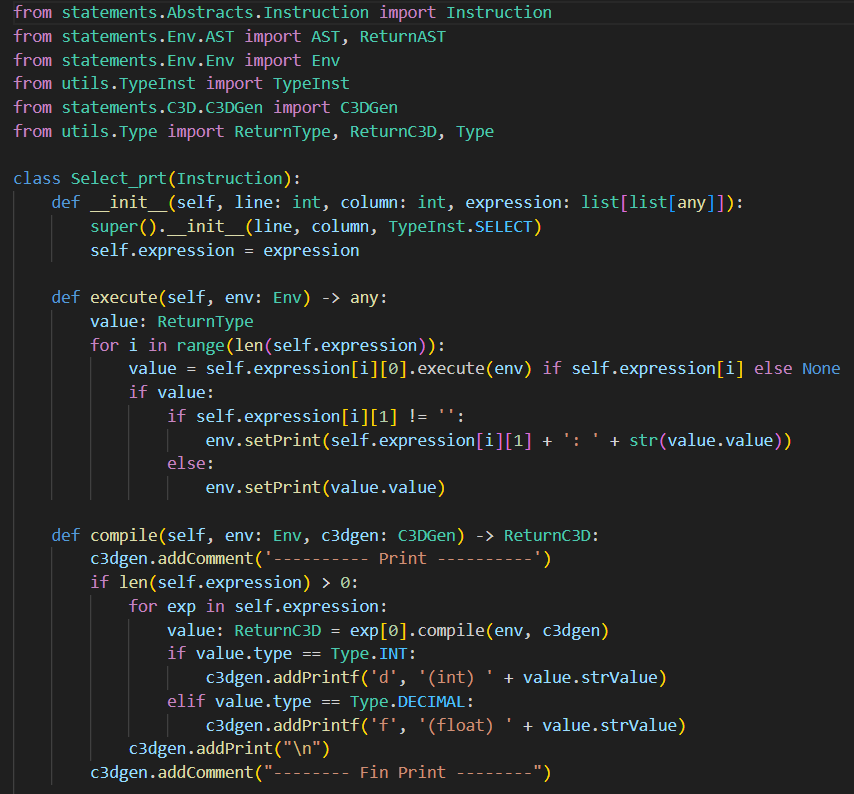
InitID

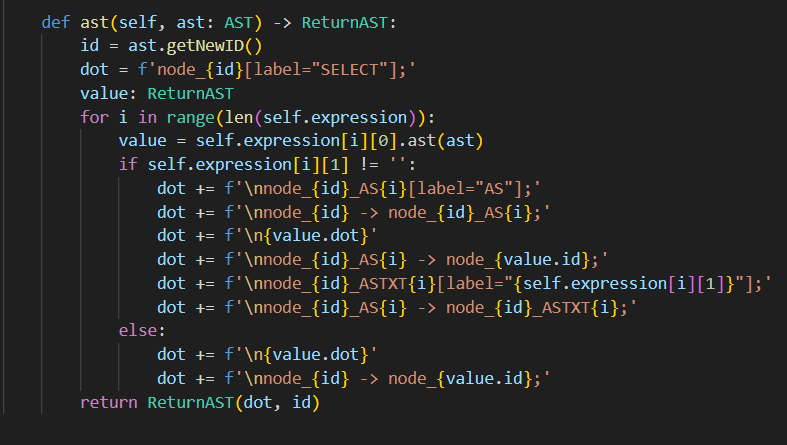


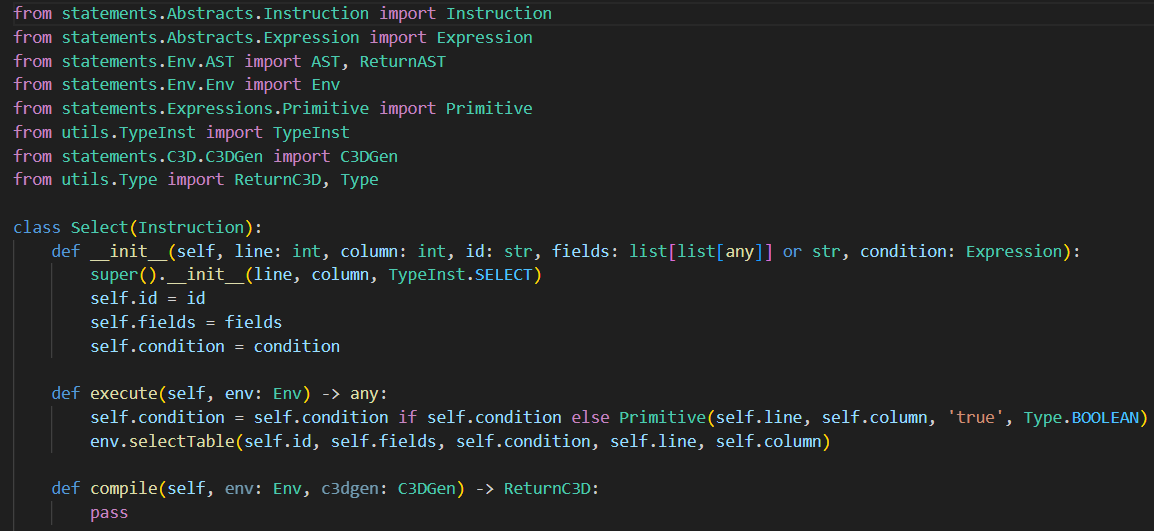
InsertTable



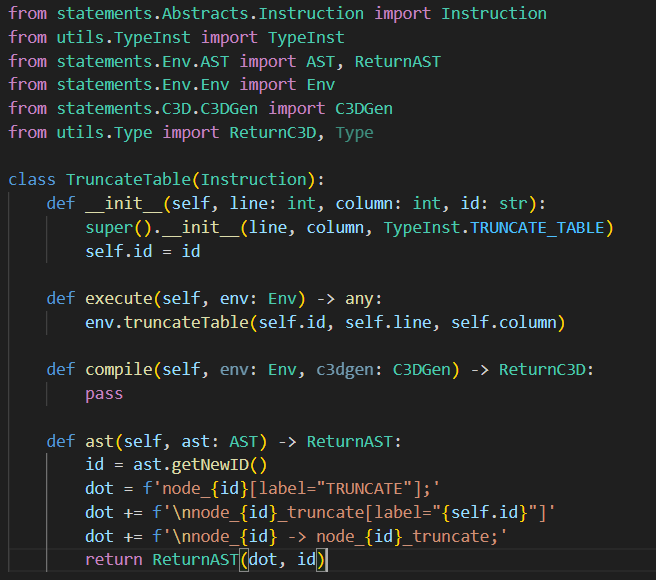
Select\_prt



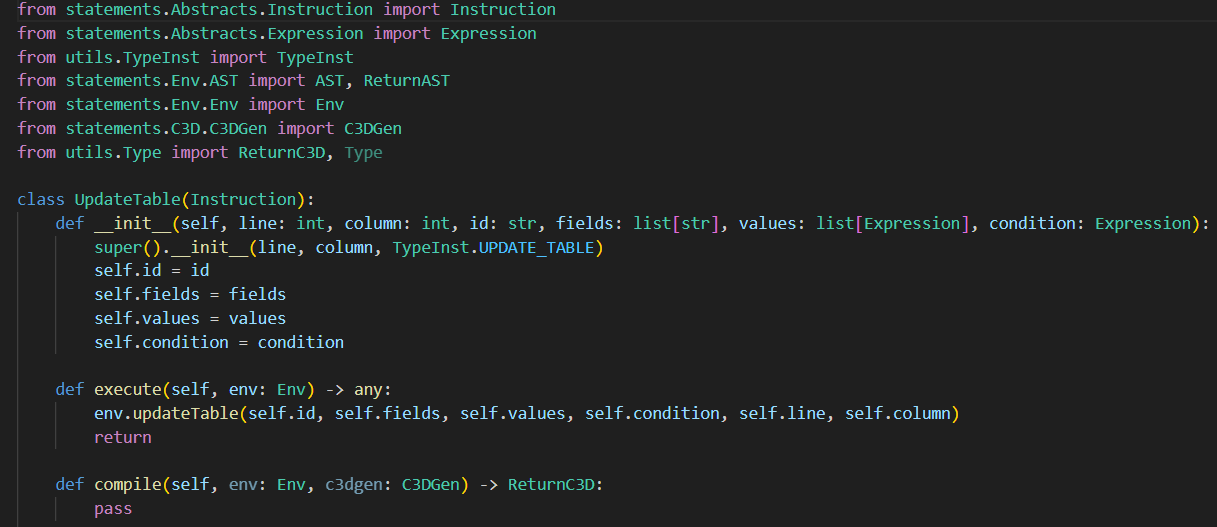


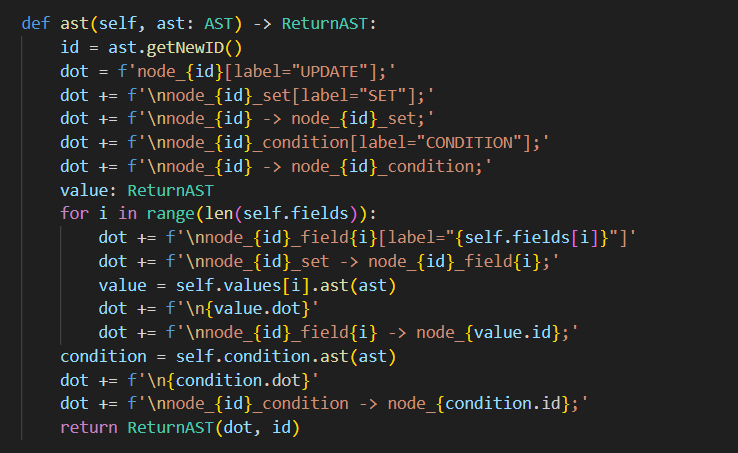
Select



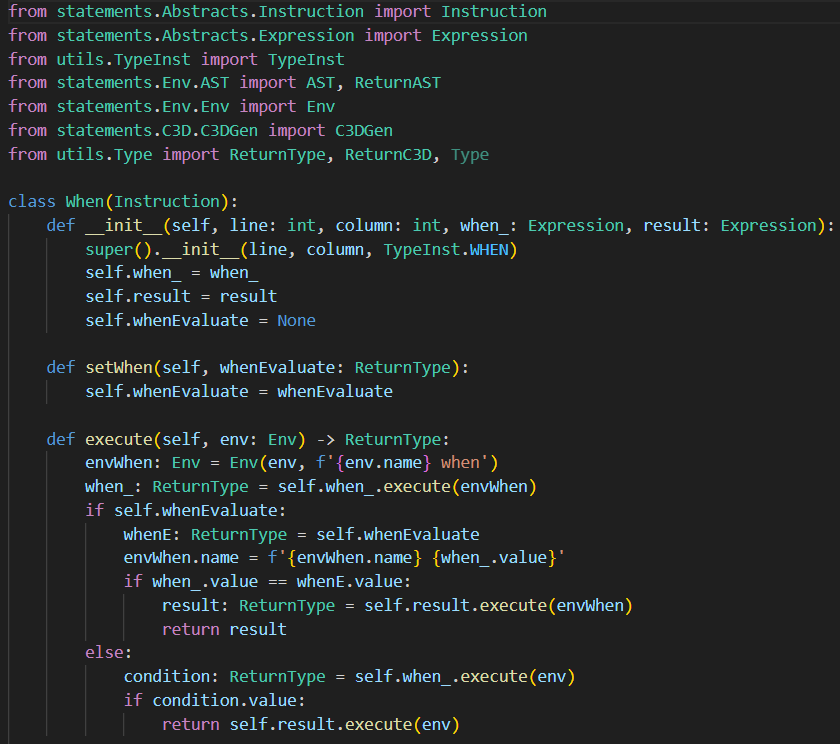
TruncateTable

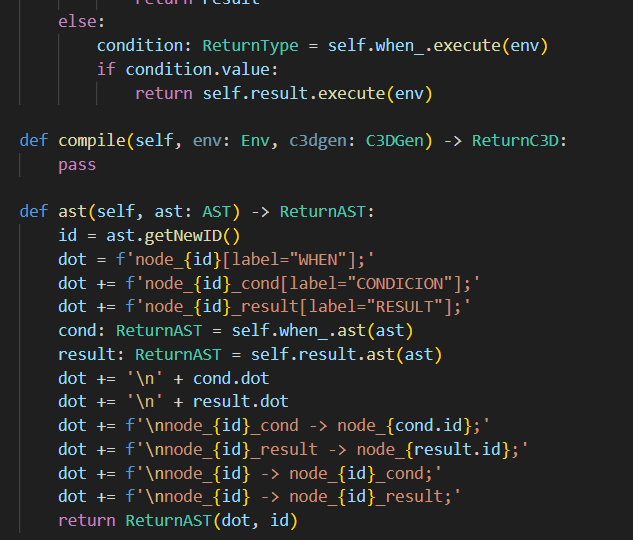
UpdateTable



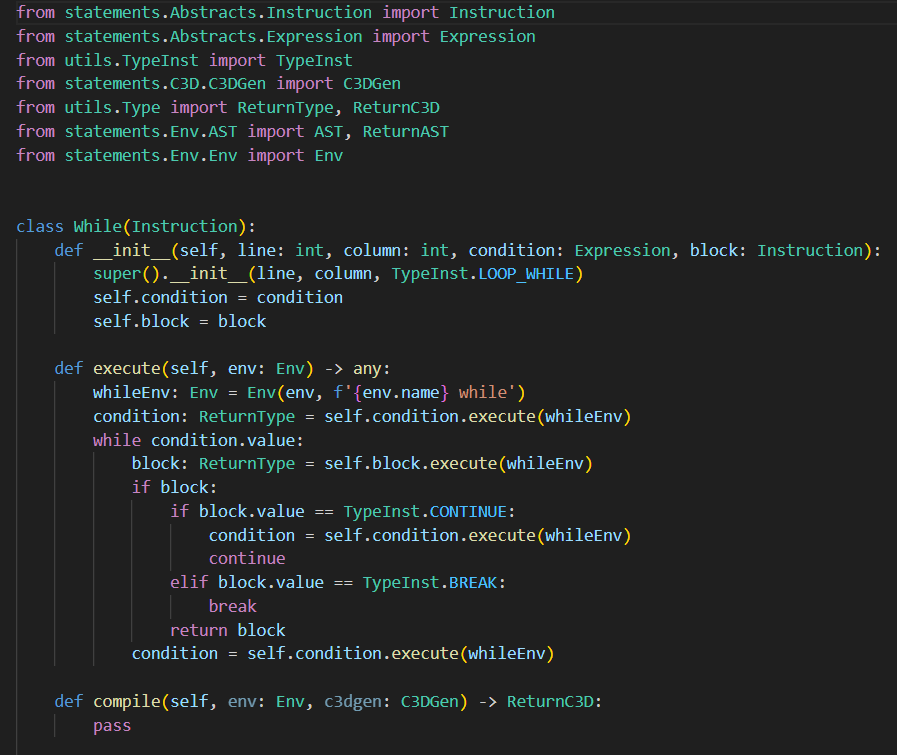


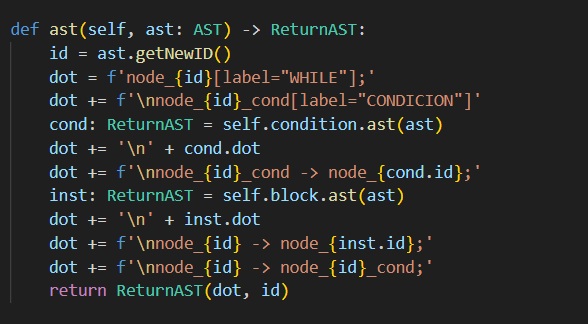
Truncate



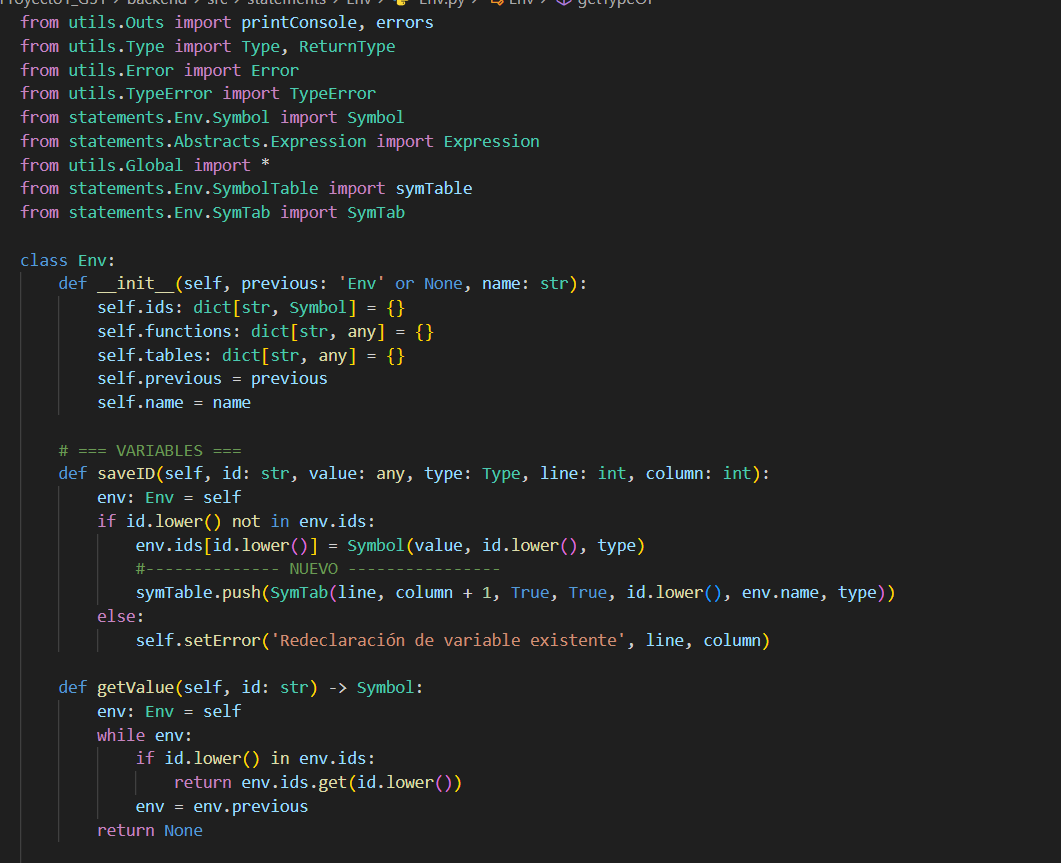


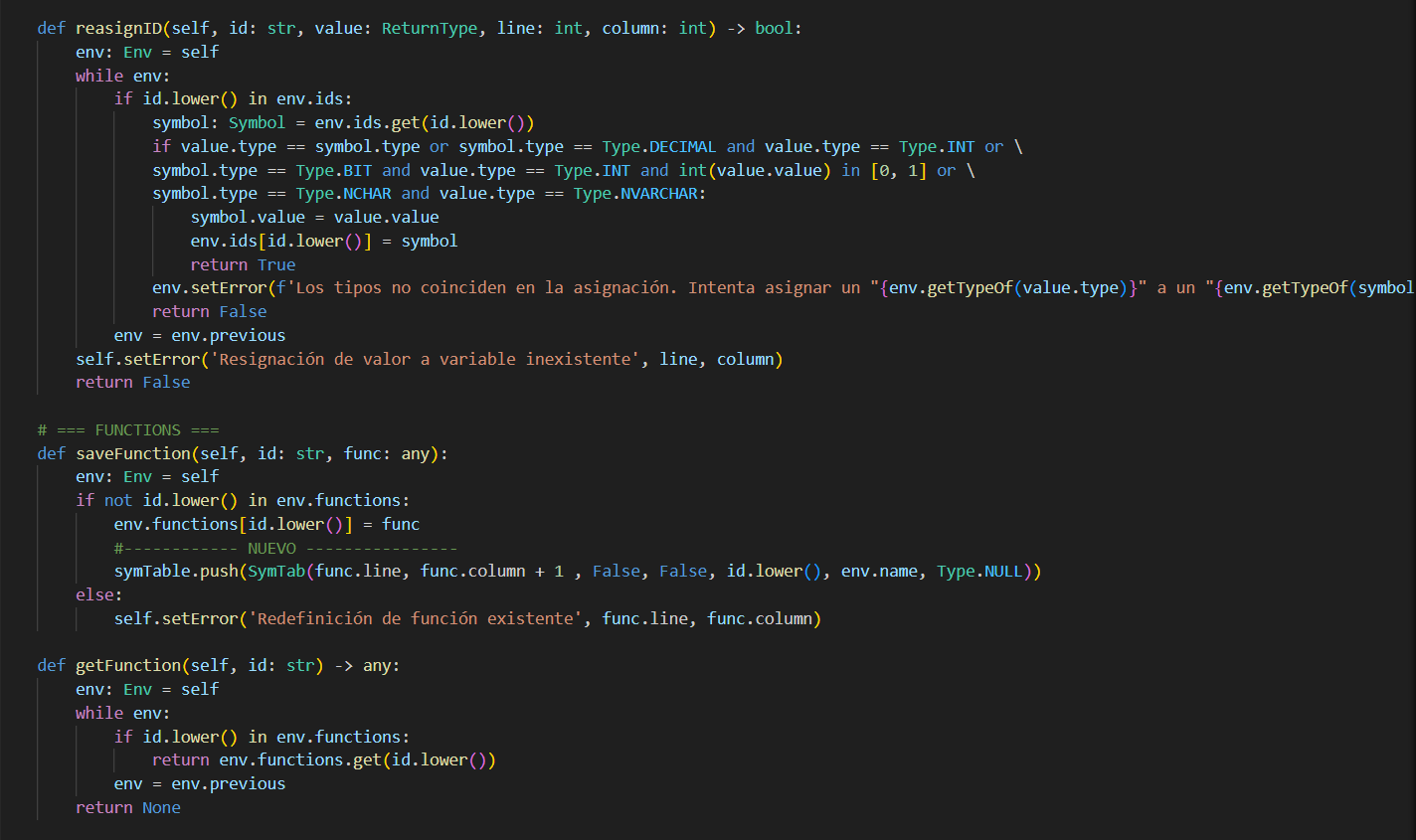
While

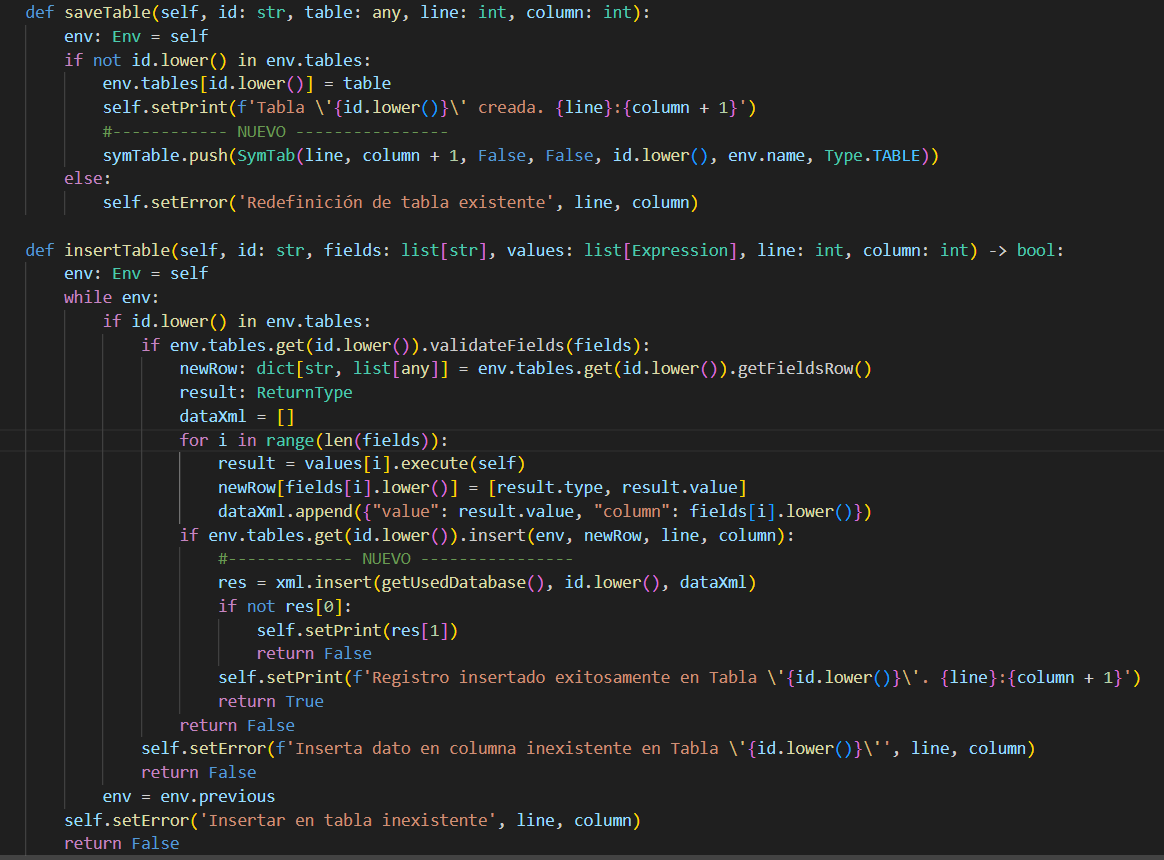


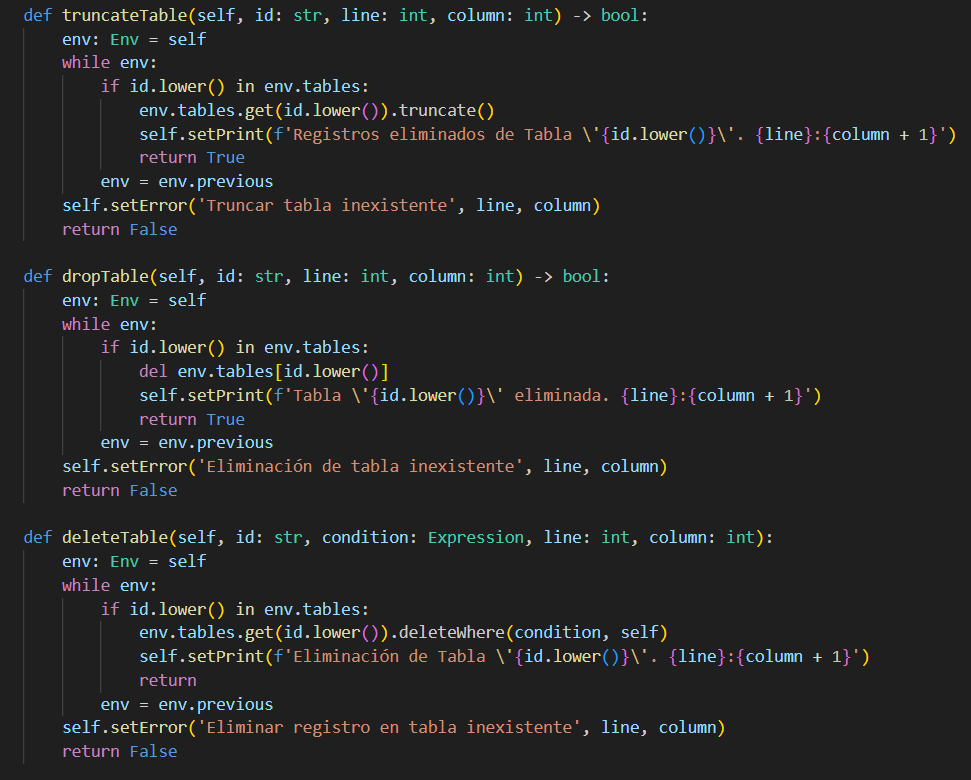


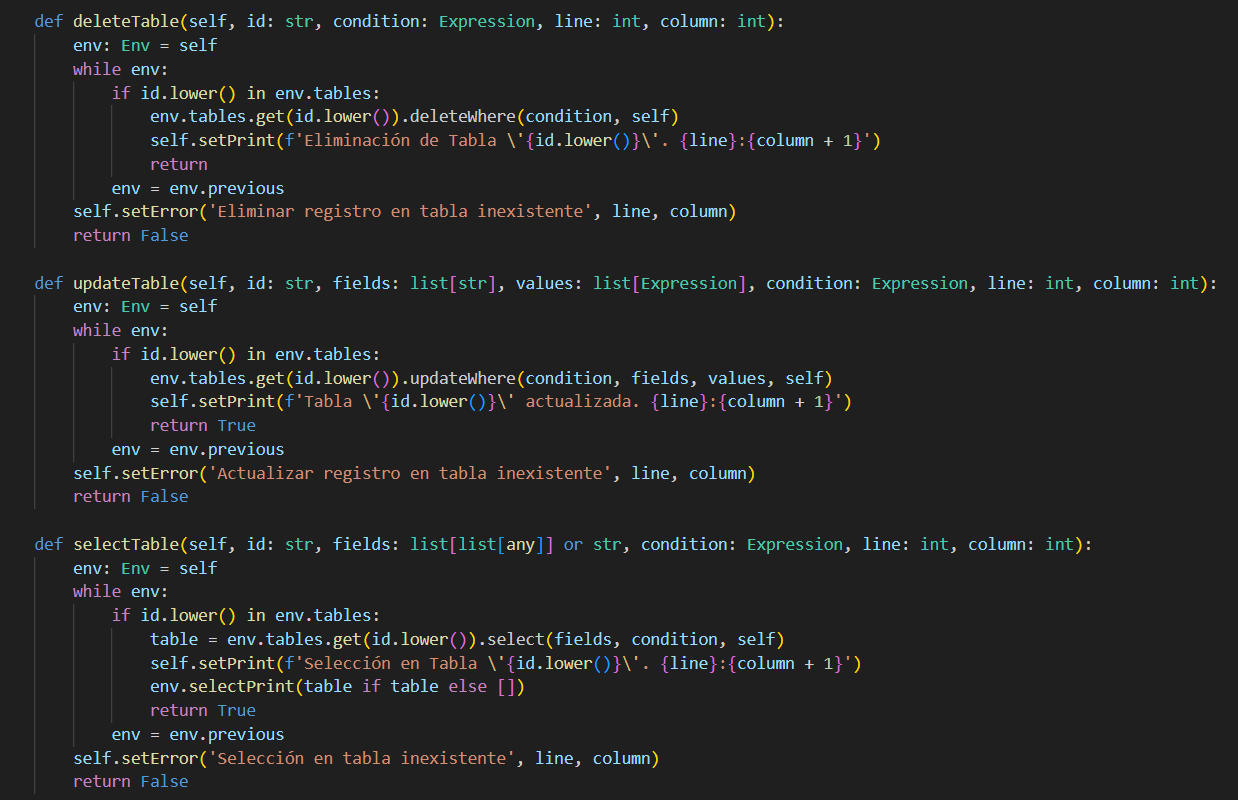
Los entornos que se utilizaron para la creación del programa son los siguientes:

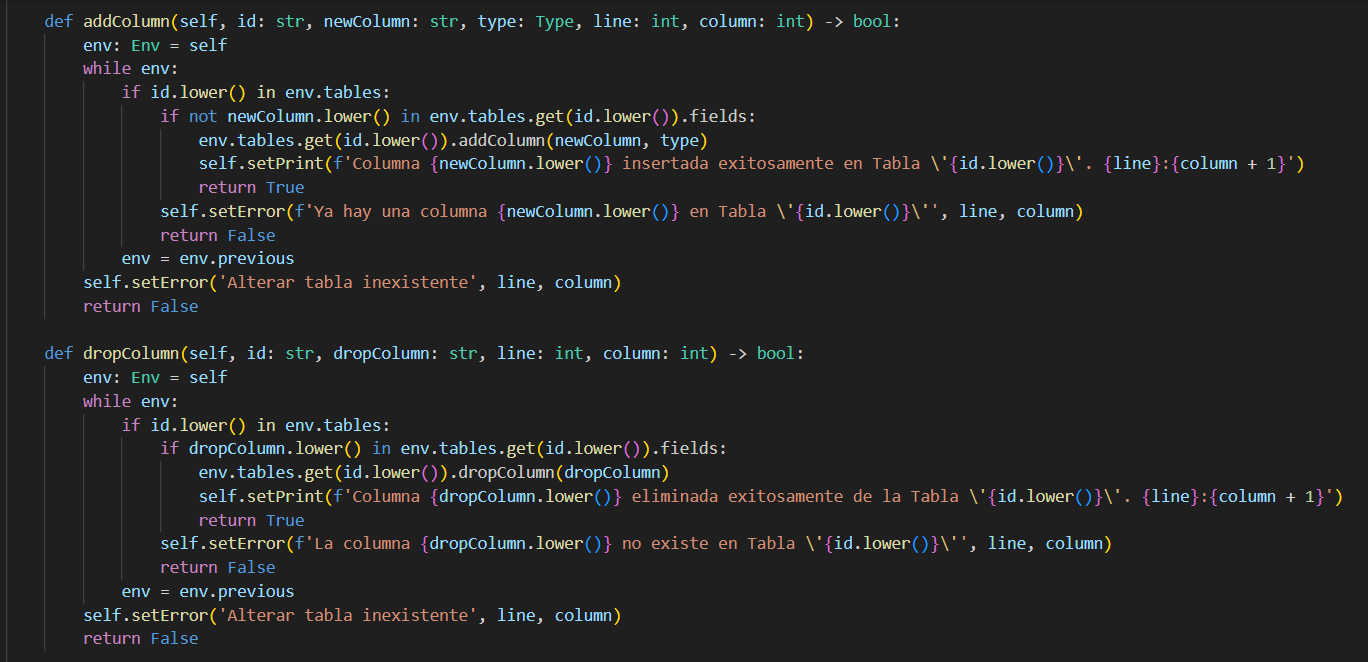




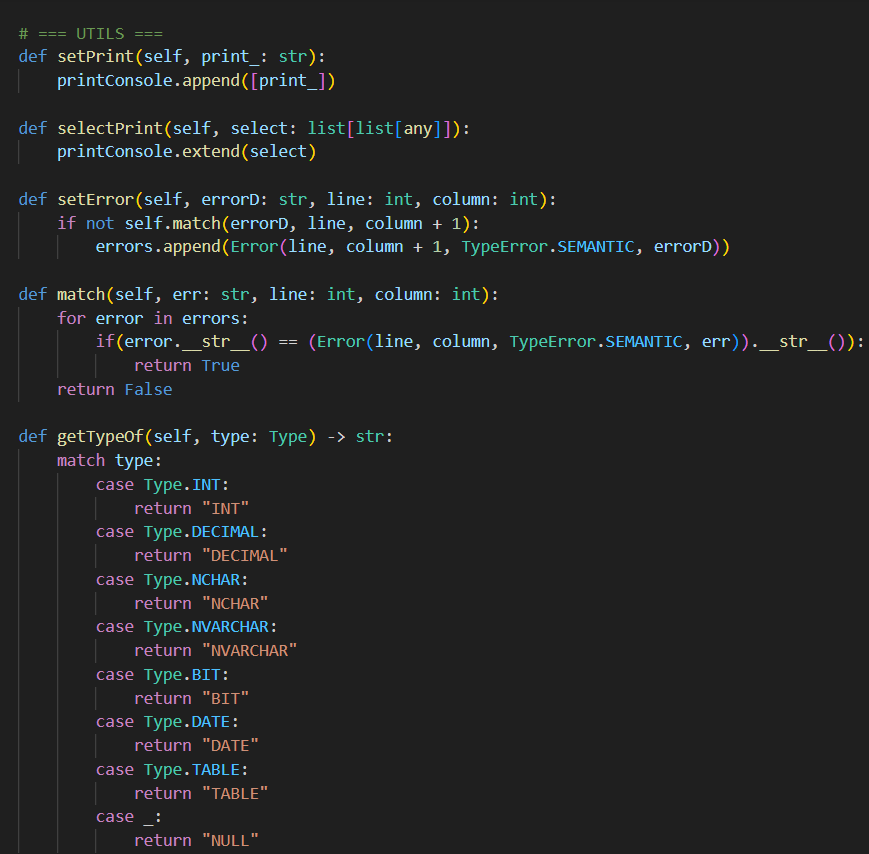












Abstract Sintact Tree (AST)

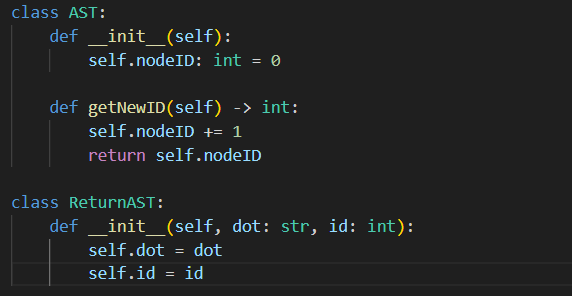
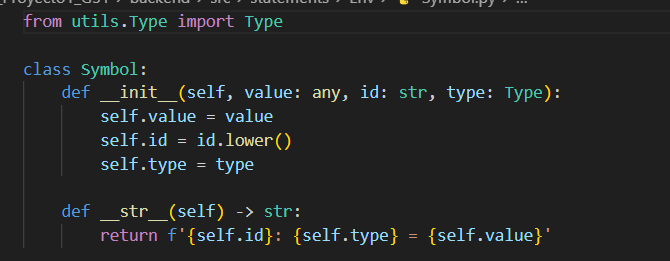


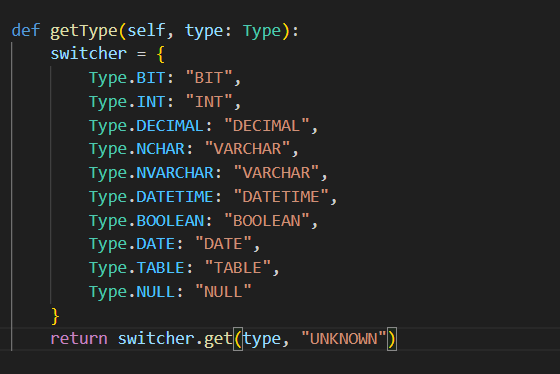
Tabla de símbolos



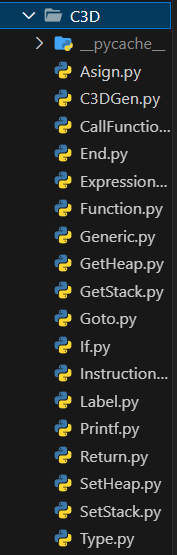




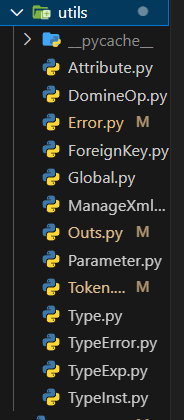




Codigo De Tres Direcciones(C3D) Clases



Clases de apoyo



Frontend

