

MANUAL TECNICO

Entornos que el que se desarrolló TypeWise

Windows

- Procesador: Intel Core i7-10510U CPU @ 1.80GHz 3.10GHz
- Memoria: 16GB RAM.
- Disco memoria: 500GB SSD
- java versión "1.8.0_341"

Para realizar los análisis léxico y sintáctico:

- "jison": "0.4.18"

Framerworks y librerías frontend y backend:

- "cors": "^2.8.5"
- "react": "^18.2.0"
- "bootstrap": "^5.2.3",

BACKEND

CLASES:

- **InterpreteController:** clase implementada para el manejo de peticiones en el lado cliente
- **Environment:** Clase desarrollada para el manejo de ámbitos en la ejecución
- **Expression:** Clase que maneja las expresiones ingresadas
- **Instruction:** Clase que maneja las funciones ingresadas
- **Return:** Clase que hace le manejo de retorno de tipo de datos
- **Simbolo:** Clase que realiza el manejo de símbolos, variables, etc.
- **Length:** Clase que realiza el retorno de longitud de alguna expresión
- **Round:** Clase que realiza el retorno el valor redondeado de alguna expresión
- **ToLower:** Clase que realiza el retorno el valor en minusculas de alguna expresión
- **ToString:** Clase que realiza el retorno el valor en string(cadena) de alguna expresión
- **ToUpper:** Clase que realiza el retorno el valor en mayusculas de alguna expresión
- **Truncate:** Clase que realiza el retorno el valor sin decimales de alguna expresión
- **Typeoef:** Clase que realiza el retorno el tipo de dato de alguna expresión
- **Main:** Clase que hace la ejecucion del código ingresado
- **Acceso:** Clase que accede a una variable ya declarada
- **Aritmetica:** Clase que realia las operaciones aritméticas
- **LlamadaFuncion:** Hace el llamado de funciones ya guardas en el entorno
- **Logica** Clase que realia las operaciones Logicas
- **OperacionesUnarios:** Clase que realia las operaciones Incremento y decremento como expresión
- **Parametros:** Para el manejo de parámetros
- **Primitivo:** Retorna el valor
- **Relacional:** Clase que realia las operaciones relacionales
- **Arreglo:** para el guardado de arreglos
- **DoWhile:** Clase para el manejo de instrucción DoWhile
- **For** Clase para el manejo de instrucción For
- **While** Clase para el manejo de instrucción While
- **If:** Clase para el manejo de instrucción IF
- **Declara:** Clase para el manejo de instrucción declarar

- **Decre** : Clase para el manejo de instrucción Decrementar
- **Funcion**: Clase para el manejo de instrucción Funcion
- **Incre** Clase para el manejo de instrucción Incrementar
- **Print** Clase para el manejo de instrucción Print de expresiones
- **Statement** Clase para el manejo de instrucción statement bloque de código entre llave
- **Error** Clase para el manejo de instrucción errores dentro del código
- **Printlist**: Clase para el manejo de una lista de prints y mostrarlos en consola
- **TablaSimbolos**: Clase para el manejo de símbolos y ámbitos
- **TablaDominante**: Clase para corbar operación aritmética
- **TipoAritmetica**: Manejo de operaciones tipo
- **TipoLogica**: Manejo de operaciones lógicas y tipo
- **TipoRelacionales**: Manejo de operaciones relacionales y tipo

METODOS EMPLEADOS

- **Guardar**: Guarda la variables
- **getVar**: Obtiene la variable
- **guardarFuncion**: Guarda la función
- **getFuncion**: Retorna la función
- **getGlobal**: Retorna el entorno global
- **modificar**: Actualiza variables
- **guardar_arreglo**: Guarda la arreglos
- **get_array**: Retorna el arreglo
- **update_array**: Actualiza el valor del arreglo

Existen 2 metodos abstractos para cada clase

- **Execute**: Ejecuta el código
- **drawAst**: Retorna el ast como grafo

FRONTEND

CLASES:

- **Button**: clase implementada para el manejo de botones
- **Console**: Clase desarrollada para el manejo de la consola
- **Editor**: Clase que manejo del editor
- **Modal**: Clase que manejo de modals de reportes
- **Home**: Manejo de la pagina principal

METODOS EMPLEADOS

- **Interpretar**: Implementación para la llamada del interprete
- **uploadDocument**: para cargar archivos