

Universidad de San Carlos de Guatemala
Primer semestre
Facultad de ingeniería
Escuela de Ciencias y Sistemas
Organización de Lenguajes y Compiladores 1

MANUAL TECNICO

Roberto Carlos Gómez Donis

202000544

03/05/2023

TypeWise

Introducción

Un lenguaje de programación interpretado diseñado para ayudar a los estudiantes de Introducción a la Programación y Computación 1 a familiarizarse con los conceptos básicos de la programación. TypeWise ha sido desarrollado utilizando las tecnologías de Node.js, React y TypeScript, y su gramática ha sido definida utilizando la herramienta Jison.

Este manual técnico está destinado a proporcionar información detallada sobre cómo utilizar TypeWise, tanto desde la perspectiva del usuario como desde la del desarrollador. En él, encontrarás una descripción completa de la sintaxis del lenguaje, incluyendo sus palabras clave, operadores y estructuras de control. También se proporcionará información sobre cómo utilizar el intérprete de TypeWise para ejecutar programas escritos en este lenguaje.

Para los desarrolladores interesados en contribuir al proyecto, este manual técnico también incluirá información sobre cómo trabajar con el código fuente de TypeWise utilizando herramientas como Node.js, React y TypeScript, así como cómo personalizar la gramática del lenguaje utilizando Jison.

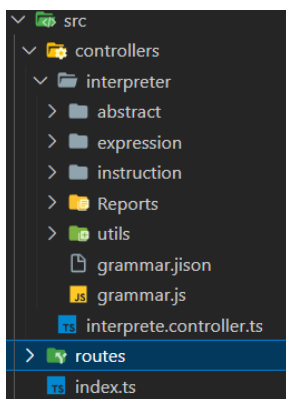
Tecnologías utilizadas

Node.js, React, TypeScript, Jison

La aplicación web utiliza las siguientes tecnologías:

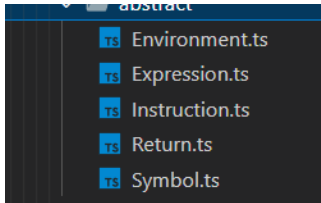
- React: Para el frontend.
- Node.js: Backend
- TypeScript: Backend.
- Jison: La producción de la gramática para el analizador.

Clases y carpetas utilizadas:



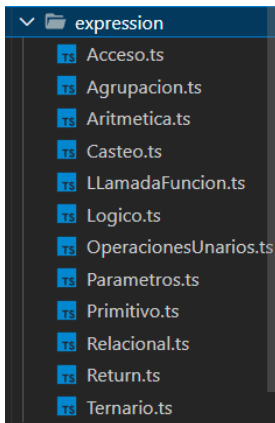
En esta imagen se logran ver las carpetas y la clase index la cual es la conexión del front con el back, también se logra apreciar el archivo json el cual nos permitió realizar el análisis.

Empecemos por la Carpeta abstract:



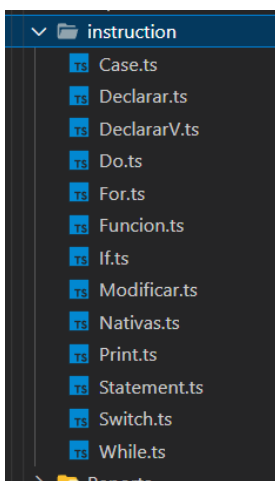
En esta carpeta se encuentran clases abstractas que simplemente heredan a otras clases los atributos que estas contienen

Luego sigue la carpeta "Expression"



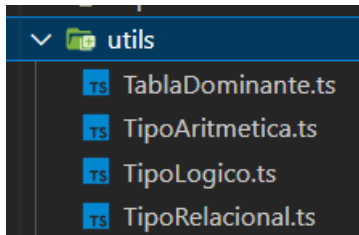
Estas clases nos permiten que la expresión tome cualquiera de estos tipos de datos cuando lo requiera

Siguiente le sigue la carpeta instruction:



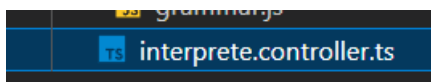
Esta carpeta y sus clases nos permiten que el interprete pueda realizar las acciones necesarias para ejecutar las sentencias, como se logra apreciar acá se encuentra todo lo necesario para que el interprete pueda realizar las acciones ya que las expresiones lo único que hace es que el dato tome diferentes tipos.

Por ultimo le sigue la carpeta Utils:



Esta carpeta solo es una auxiliar para poder hacer algunos tipos de datos para la expresión

Por ultimo queda la clase



La cual es muy importante, ya que gracias a esta clase hace que el interprete funcione porque esta clase nos proporciona un entorno en el cual trabajar, así a su vez puede hacer que tenga sub entornos.