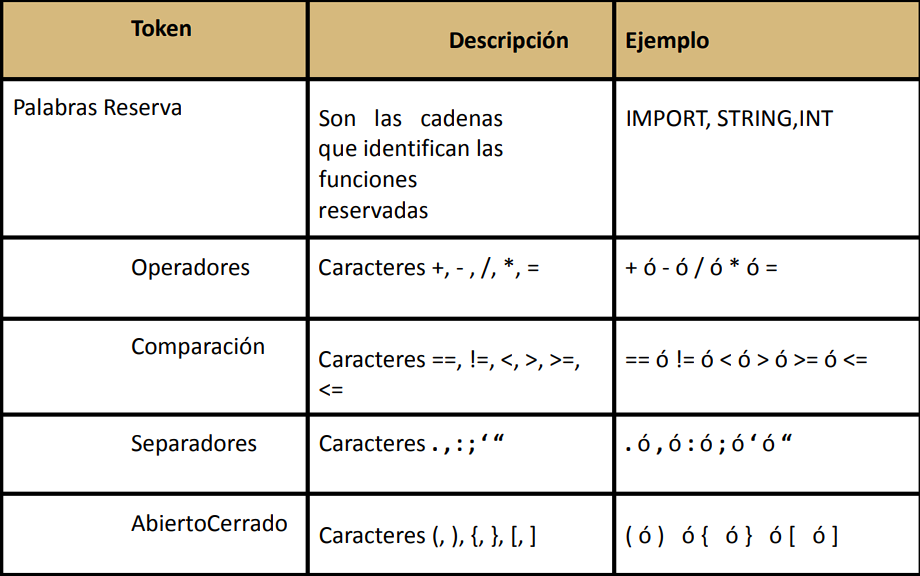
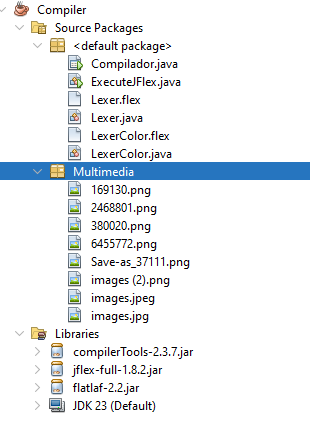
**Palabras reservadas y detalles específicos:** El programa cuenta con 11 tipos de tokens y 25 palabras reservadas. Tokens



**Sobre el compilador:** Un compilador es un programa que traduce un programa escrito en lenguaje fuente y produce otro equivalente escrito en un lenguaje destino. Lenguaje de alto nivel. Por ejemplo: C, Pascal, C++. Vista general: Estructura del proyecto El compilador se compone de 4 carpetas Source Packages, Test Packages, Libraries y Test Librares. En la carpeta Source Packages, se cuenta con dos paquetes java: y Multimedia. En el segundo, son guardadas todas las imágenes que se emplean en el compilador para ser más visualmente atractivo al usuario.



**¿Qué es el análisis semántico?** La fase de análisis semántico revisa el programa fuente para tratar de encontrar errores semánticos y reúne la información sobre los tipos para la fase posterior de generación de código. En ella se utiliza la estructura jerárquica determinada por la clase de análisis sintáctico para identificar los operadores y operandos de expresiones y proposiciones. Clase del analizador semantico.java Después de haber guardado anteriormente las producciones generadas en el análisis sintáctico, estas se tratan de diferente manera dependiendo de las instrucciones que guardan. Análisis de Identificadores: Lo primero que hace esta parte es generar un HashMap que contiene los tipos de dato que manejamos en el lenguaje, además de los valores que aceptan estos tipos de datos; esto para comparar las asignaciones más adelante. Después, se va iterando el arreglo de producciones identProd, que es el que contiene las producciones que se generan al declarar una variable. Los valores que están en la posición 1 son los ID (nombres de las variables), y los que están en la posición 0 son los tipos de datos. Estos se van agregando al HashMap de identificadores, excepto en los casos donde ya haya una variable con el mismo nombre en este HashMap, lo que genera un error. Cuando este proceso termina, se itera el arreglo asigProd, que almacena las líneas de código donde hay una asignación de valores a una variable. Esta vez, los ID están en la posición 0, ya que estas instrucciones no especifican el tipo de dato. Entonces, se revisa en el HashMap de identificadores mencionado anteriormente si esta variable está declarada, además de que si el tipo de dato asignado coincide con el que debe almacenar la variable, usando el HashMap tiposDatos del principio. El arreglo asigProdConID tiene la misma función, con la excepción de que el valor asignado es también una variable, por lo que se debe revisar el tipo de dato de ambas variables. Cabe destacar que las producciones en identProd están en la forma: TIPODATO ID [ = VALOR] ; //(valor opcional)