**Analizador Léxico:**

**analisisLexico(String texto):**

Este método es el que se encarga del análisis léxico el cual genera los tokens que se utilizan para el análisis sintáctico más adelante, se puede seguir el funcionamiento del método por los comentarios en este.

XD

**clasificadorCaracter(char carac):**

Este método se encarga de regresar un entero según el carácter que le llegue, con este se consigue clasificar los chars y luego crear los reportes.

XD

**reporteCaracter(char Carac):**

Método que se encarga de crear el reporte por cada carácter, utiliza un switch para clasificar según lo que llegue, este método depende del método anterior.

XD

**Analizador Sintáctico:**

Expresión para Declaración

Declaración |ID ASIGNACION EXPRESION

|ID OPERADOR/ ASIGNACION EXPRESION

EXPRESION -> |Constante(Operador Y Constante)\*;

| ID

|Otros(ID|Constantes|Otros)\*

Expresión para Métodos:

Métodos |PALABRA\_CLAVE ID OTROS ((ID|Constantes)(Otros)?)\* OTROS OTROS

Expresión para invocación de método:

Métodos |ID OTROS ((ID|Constantes)(Otros)?)\* OTROS

Expresión para Condicionales

if expresión [condicional expresión]\* :

Condicional -> | if expresión [condicional expresión]\*

EXPRESION -> | boolean

| ID

| (Entero|Double|ID) Comparación (Entero|Double|ID);

Expresión para For

For x in range:

For -> | For Expresión

Expresion - > | ID in (ID|Range ((ID|Constante)(Otros)?)\*)OTROS

Expresión para While

While -> | While Expresión

Expresión -> | (ID|Constante) Condicional (ID|Constante)

Tabla de expresiones:

la tabla de simbolos va mas o menos de la siguiente manera

# simbolo # tipo # valor # linea # columna

x entero 5 10 12

ejemplo funcion -- 1 1