



MANUAL DE USUARIO

EQUIPO ALFA

Nombre: Luis Hervert
Damián Martínez 1504419
Nombre: Andre Sebastian
Gordillo Marquez

Github:
LHDM2000
Github:
androcop

ESPECIFICACIONES LEXICAS DEL LENGUAJE

```

Comentarios = {TraditionalComment} | {EndOfLineComment} |
{DocumentationComment}
TraditionalComment = "/*" [^*] ~"*/" | "/*" "*" + "/"
EndOfLineComment = "//" {ENTRADA}* {FinLinea}?
DocumentationComment = "/*" {CommentContent} "*" + "/"
CommentContent = ( [^*] | \*+ [^/*] ) *
If = "si"
Entonces = "entonces"
Sino = "sino"
Leer = "leer"
Escribir = "escribir"
Devolver = "devolver"
Constructor = "Constructor"
Destructor = "Destructor"
Principal = "Principal"
Instanciar = "instanciar"
Desde = "desde"
Clase = "clase" | "Clase"
Caracter_especial = "á"
NumeroDecimal = [1-9][0-9]* | 0
NumeroReal = [0-9]*[.][0-9]+[1-9]
Booleano = [Verdadero|Falso]
Cadena = ["\""]([a-zA-Z]*[0-9]*["\ "]*)*["\""]
Tabulacion = ["\ "]{4}|\t
Variables = [a-z]+{Reglas_variablesextra} | [a-z]
Propiedades = "Propiedades"|"propiedades"
Metodos = "Metodos"
Tipo_de_metodos =
"Publica"|"Publico"|"Privada"|"Privados"|"Protegidos"|"Protegidas"
Tipo_clase = "Impar"|"Par"
Tipo_entero = "entero"
Tipo_cadena = "cadena"
Tipo_real = "real"
Tipo_booleano = "booleano"
Tipo_nulo = "nulo"
Mientras = "mientras"
Incrementar = "incrementar"
Decrementar = "decrementar"
Hacer = "hacer"
Extiende = "extiende"
Eliminar = "Eliminar"
Incluir = "incluir"
Nuevo = "nuevo"
Ejecutar = "ejecutar"|"Ejecutar"
Verdadero = "verdadero"
Falso = "falso"

```

```

Parentesis1 = "("
Parentesis2 = ")"
Coma = ","
Reglas_variablesexta = [[a-z]|[0-9]|[A-Z]]*|[[a-z]|[0-9]|[A-Z]]* "_" [[a-
z]|[0-9]|[A-Z]]+
Excepciones_variables = [a-z]+|[[a-z]|[0-9]|[A-Z]]* "_" | [A-Z]+ {Variables}
Variables_clase = [A-Z]+{Reglas_variablesexta}
Excepciones_variables_clase = [a-z]+|[[a-z]|[0-9]|[A-Z]]* "_"
Suma = "+"
Resta = "-"
Multiplicacion = "*"
Division = "/"
Modulo = "%"
Exponente = "^"
Igual = "="
Incremento = "++"
Decremento = "--"
And = "AND"
Or = "OR"
MayorQue = ">"
MenorQue = "<"
IgualQue = "=="
DistintoQue = "!="
DosPuntos = ":"
Corchete1 = "["
Corchete2 = "]"
Punto = "."
CadenaEntero = "cadenaAEntero"
CadenaReal = "cadenaAReal"
CadenaBooleano = "cadenaABoleano"
Seno = "seno"
Coseno = "coseno"
Tangente = "tangente"
Logaritmo = "logaritmo"
Raiz = "raiz"
FinLinea = \r|\n|\r\n
ENTRADA = [^\r\n]
Ignorar = \n|\r\n|\r\n|\ "

```

ESPECIFICACIONES SINTACTICA DEL LENGUAJE

terminal CLASE, PROPIEDADES, METODOS;
 terminal IF, ENTONCES, SINO;
 terminal LEER, ESCRIBIR;
 terminal DEVOLVER, CONSTRUCTOR, DESTRUCTOR, PRINCIPAL, INSTANCIAR, DESDE, MIENTRAS;
 terminal INCREMENTO, DECREMENTO, HACER, EXTIENDE;
 terminal PARENTESIS1, PARENTESIS2, COMA;
 terminal IGUAL, AND, OR, DOS_PUNTOS;
 terminal PUNTO_Y_COMA;
 terminal VARIABLE, RAIZ;
 terminal MENOS, PUNTO;
 terminal VARIABLE_CLASE, ELIMINAR;
 terminal TIPO_DATO, OPERADORES;
 terminal SUMA, RESTA, MULTIPLICACION, DIVISION, EXPONENCIACION, MODULO;
 terminal MENOR_QUE, MAYOR_QUE, DISTINTO_QUE, IGUAL_QUE, INCREMENTAR, DECREMENTAR;
 terminal SENO, COSENO, TANGENTE, LOGARITMO, TIPO_DE_METODOS;
 terminal CADENA_A_BOOLEANO, CADENA_A_REAL, CADENA_A_ENTERO, CADENA;
 terminal SIGNOS, FUNCION_ESPECIAL, FUNCIONES_CLICLOS_ETC, COMENTARIO, SALTO_LINEA, EJECUTAR,
 NUMERO_DECIMAL;
 terminal VERDADERO, FALSO, NEW;
 non terminal S;
 non terminal CLASE_GENERAL, CLASE_PARTICULAR, VARIABLES_PROPIEDADES,
 VARIABLES_PROPIEDADES2;
 non terminal PROPIEDAD_CLASE, METODO_CLASE, METODOS_PARTICULAR;
 non terminal FINAL_DE_LINEA, SALTO, CODIGO_GENERAL, METODO_RETORNO, METODO_EJECUTAR;
 non terminal METODO_CODIGO, METODO_SINCODIGO, OPERACION, ASIGNACION,
 SIGNO_MATE, CONCATENACION, LOGICO, LEER_ESCRIBIR;
 non terminal DEVOLVER_METODO;
 non terminal CONDICION, CODIGO_CONDICION, SINO_GENERAL;
 non terminal CICLO_FOR, CICLO_DO_WHILE, DECLARAR_FOR;
 non terminal INSTANCIAR_CLASE, FUNCIONES, VARIABLES, VARIABLE_REPETIDAS;
 non terminal IGUAL_CLASE, METODOS_OBJETOS;
 start with S;
 S ::= CLASE_GENERAL;
 CLASE_GENERAL ::= CLASE_PARTICULAR CLASE_GENERAL
 | CLASE_PARTICULAR ;
 CLASE_PARTICULAR ::= CLASE VARIABLE_CLASE { :System.out.println("Encontre un de clase"); }
 | CLASE VARIABLE_CLASE PROPIEDAD_CLASE METODO_CLASE { :System.out.println("Encontre un
 de clase"); }
 | CLASE VARIABLE_CLASE PROPIEDAD_CLASE { :System.out.println("Encontre un de clase"); }
 | CLASE VARIABLE_CLASE METODO_CLASE { :System.out.println("Encontre un de clase"); }
 | CLASE TIPO_DATO { :System.out.println("Encontre un de clase"); }
 | CLASE TIPO_DATO PROPIEDAD_CLASE METODO_CLASE { :System.out.println("Encontre un de
 clase"); }
 | CLASE TIPO_DATO PROPIEDAD_CLASE { :System.out.println("Encontre un de clase"); }

```

|CLASE TIPO_DATO METODO_CLASE {:System.out.println("Encontre un de clase");;} ;
PROPIEDAD_CLASE ::= PROPIEDADES TIPO_DE_METODOS DOS_PUNTOS {:System.out.println("Encontre un
de clase");;}

|PROPIEDADES TIPO_DE_METODOS DOS_PUNTOS VARIABLES_PROPIEDADES
{:System.out.println("Encontre un de clase");;}
|PROPIEDADES TIPO_DE_METODOS DOS_PUNTOS VARIABLES_PROPIEDADES
PROPIEDAD_CLASE {:System.out.println("Encontre un de clase");;}
|PROPIEDADES TIPO_DE_METODOS DOS_PUNTOS PROPIEDAD_CLASE
{:System.out.println("Encontre un de clase");;}
VARIABLES_PROPIEDADES ::= TIPO_DATO VARIABLE_REPETIDAS {:System.out.println("Encontre una
variable");;}

|TIPO_DATO VARIABLE_REPETIDAS VARIABLES_PROPIEDADES {:System.out.println("Encontre
una variable");;}
VARIABLE_REPETIDAS ::= VARIABLE {:System.out.println("Encontre una variable");;}
|VARIABLE COMA VARIABLE_REPETIDAS {:System.out.println("Encontre una variable");;}
METODO_CLASE ::= METODOS TIPO_DE_METODOS DOS_PUNTOS METODOS_PARTICULAR
{:System.out.println("Encontre un de clase");;}
|METODOS TIPO_DE_METODOS DOS_PUNTOS METODOS_PARTICULAR METODO_CLASE
{:System.out.println("Encontre un de clase");;}
|METODOS TIPO_DE_METODOS DOS_PUNTOS {:System.out.println("Encontre un de clase");;}
METODOS_PARTICULAR ::= METODO_RETORNO {:System.out.println("Encontre Metodos");;}
|METODO_EJECUTAR {:System.out.println("Encontre Metodos");;}
|METODO_RETORNO METODOS_PARTICULAR {:System.out.println("Encontre Metodos");;}
|METODO_EJECUTAR METODOS_PARTICULAR {:System.out.println("Encontre Metodos");;}
METODO_RETORNO ::= TIPO_DATO VARIABLE PARENTESIS1 PARENTESIS2 DEVOLVER_METODO
{:System.out.println("Encontre variable");;}
|TIPO_DATO VARIABLE PARENTESIS1 VARIABLES_PROPIEDADES2 PARENTESIS2
DEVOLVER_METODO {:System.out.println("Encontre variable");;}
|TIPO_DATO PARENTESIS1 PARENTESIS2 {:System.out.println("Encontre parentesis");;}
|TIPO_DATO VARIABLE PARENTESIS1 PARENTESIS2 CODIGO_GENERAL DEVOLVER_METODO
{:System.out.println("Encontre variable");;}
|TIPO_DATO VARIABLE PARENTESIS1 VARIABLES_PROPIEDADES2 PARENTESIS2
CODIGO_GENERAL DEVOLVER_METODO {:System.out.println("Encontre variable");;}
|TIPO_DATO PARENTESIS1 PARENTESIS2 CODIGO_GENERAL {:System.out.println("Encontre
parentesis");;}
|TIPO_DATO PARENTESIS1 PARENTESIS2 CODIGO_GENERAL DEVOLVER_METODO
{:System.out.println("Encontre variable");;}
|TIPO_DATO PRINCIPAL PARENTESIS1 PARENTESIS2 CODIGO_GENERAL
{:System.out.println("Encontre variable");;} ;
DEVOLVER_METODO ::= DEVOLVER OPERACION {:System.out.println("Encontre metodo");;}
|DEVOLVER CONCATENACION {:System.out.println("Encontre metodo");;}
|DEVOLVER METODOS_OBJETOS {:System.out.println("Encontre metodo");;}
METODO_EJECUTAR ::= EJECUTAR VARIABLE PARENTESIS1 VARIABLES_PROPIEDADES2 PARENTESIS2
{:System.out.println("Encontre metodo");;}
|EJECUTAR VARIABLE PARENTESIS1 PARENTESIS2 {:System.out.println("Encontre metodo");;}
|EJECUTAR VARIABLE PARENTESIS1 VARIABLES_PROPIEDADES2 PARENTESIS2
CODIGO_GENERAL {:System.out.println("Encontre metodo");;}
|EJECUTAR VARIABLE PARENTESIS1 PARENTESIS2 CODIGO_GENERAL
{:System.out.println("Encontre metodo");;}

```

```

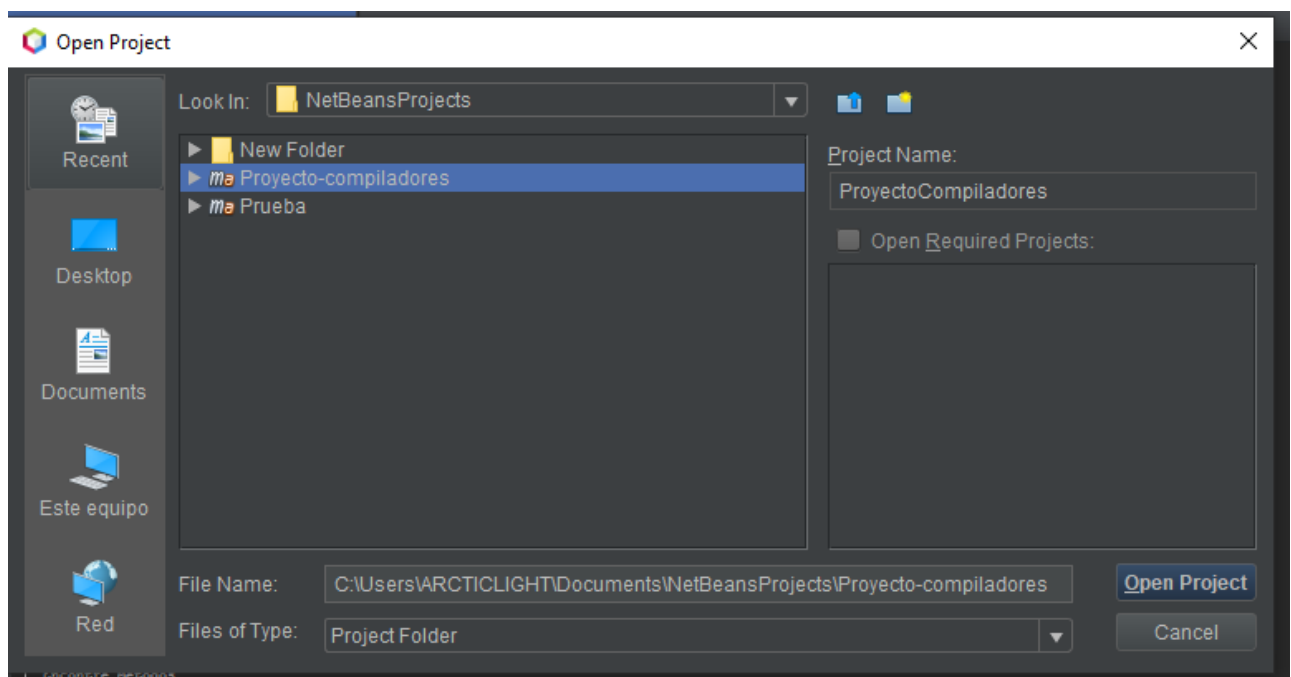
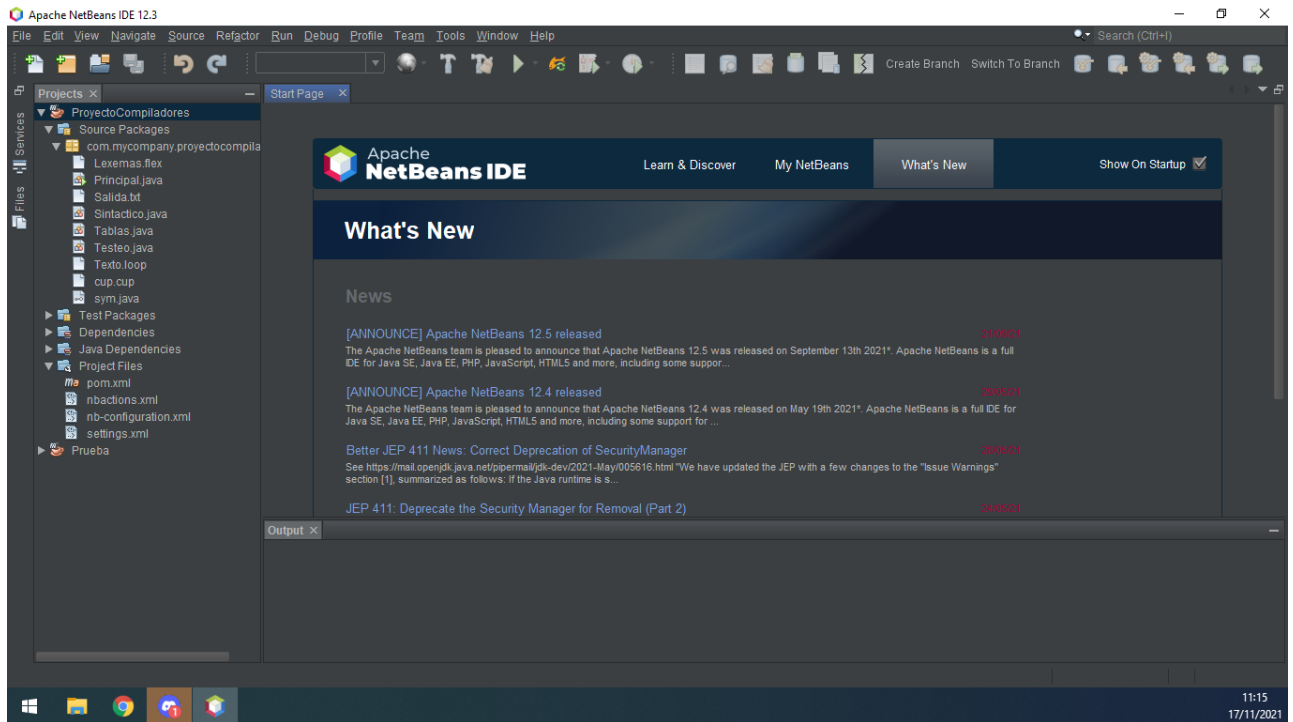
CODIGO_GENERAL ::=      ASIGNACION  {:System.out.println("Encontre Asignacion");;}
                        |LEER_ESCRIBIR  {:System.out.println("Encontre Asignacion");;}
                        |LEER_ESCRIBIR CODIGO_GENERAL {:System.out.println("Encontre Asignacion");;}
                        |ASIGNACION CODIGO_GENERAL {:System.out.println("Encontre Asignacion");;}
                        |CONDICION {:System.out.println("Encontre Sentencias");;}
                        |CICLO_FOR {:System.out.println("Encontre Sentencias");;}
                        |INSTANCIAR_CLASE {:System.out.println("Encontre Sentencias");;}
VARIABLES_PROPIEDADES2 ::= TIPO_DATO VARIABLE  {:System.out.println("Encontre Variable");;}
                        |TIPO_DATO VARIABLE COMA VARIABLES_PROPIEDADES2 {:System.out.println("Encontre
Sentencias");;}
                        |TIPO_DATO VARIABLE IGUAL OPERACION {:System.out.println("Encontre Sentencias");;}
ASIGNACION ::=          VARIABLE IGUAL OPERACION {:System.out.println("Encontre Operacion");;}
                        |VARIABLE IGUAL CONCATENACION {:System.out.println("Encontre Operacion");;}
                        |VARIABLE IGUAL LOGICO {:System.out.println("Encontre Operacion");;}
LEER_ESCRIBIR ::=      ESCRIBIR OPERACION {:System.out.println("Encontre Operacion");;}
                        |ESCRIBIR CONCATENACION {:System.out.println("Encontre Operacion");;}
                        |LEER VARIABLE {:System.out.println("Encontre Variable");;}
LOGICO ::=              VERDADERO
                        |FALSO ;
SIGNO_MATE ::=          SUMA
                        |RESTA
                        |MULTIPLICACION
                        |DIVISION
                        |MODULO      ;
CONCATENACION ::=      CADENA
                        |CADENA COMA CONCATENACION
                        |CADENA COMA OPERACION;
OPERACION ::=          VARIABLE
                        |NUMERO_DECIMAL
                        |VARIABLE SIGNO_MATE OPERACION
                        |NUMERO_DECIMAL SIGNO_MATE OPERACION;
CONDICION ::=          IF CODIGO_CONDICION ENTONCES CODIGO_GENERAL DEVOLVER_METODO SINO
                        |IF CODIGO_CONDICION ENTONCES CODIGO_GENERAL SINO
                        |IF CODIGO_CONDICION ENTONCES DEVOLVER_METODO SINO;
CODIGO_CONDICION ::=   OPERACION
                        |VARIABLE SIGNOS NUMERO_DECIMAL;
CICLO_FOR  ::=         DESDE VARIABLES_PROPIEDADES2 MIENTRAS CODIGO_CONDICION INCREMENTAR
NUMERO_DECIMAL HACER CODIGO_GENERAL
                        |HACER CODIGO_GENERAL MIENTRAS CODIGO_CONDICION CODIGO_GENERAL
                        |DESDE VARIABLES_PROPIEDADES2 MIENTRAS CODIGO_CONDICION INCREMENTAR
NUMERO_DECIMAL HACER
                        |HACER CODIGO_GENERAL MIENTRAS CODIGO_CONDICION;
INSTANCIAR_CLASE ::=   VARIABLE_CLASE VARIABLE IGUAL NEW VARIABLE_CLASE PARENTESIS1
PARENTESIS2 CODIGO_GENERAL
                        |VARIABLE_CLASE VARIABLE IGUAL NEW VARIABLE_CLASE PARENTESIS1 PARENTESIS2
                        |VARIABLE IGUAL NEW VARIABLE_CLASE PARENTESIS1 PARENTESIS2 CODIGO_GENERAL
                        |VARIABLE IGUAL NEW VARIABLE_CLASE PARENTESIS1 PARENTESIS2
                        |VARIABLE IGUAL NEW TIPO_DATO PARENTESIS1 PARENTESIS2 CODIGO_GENERAL
                        |VARIABLE IGUAL NEW TIPO_DATO PARENTESIS1 PARENTESIS2 ;

```

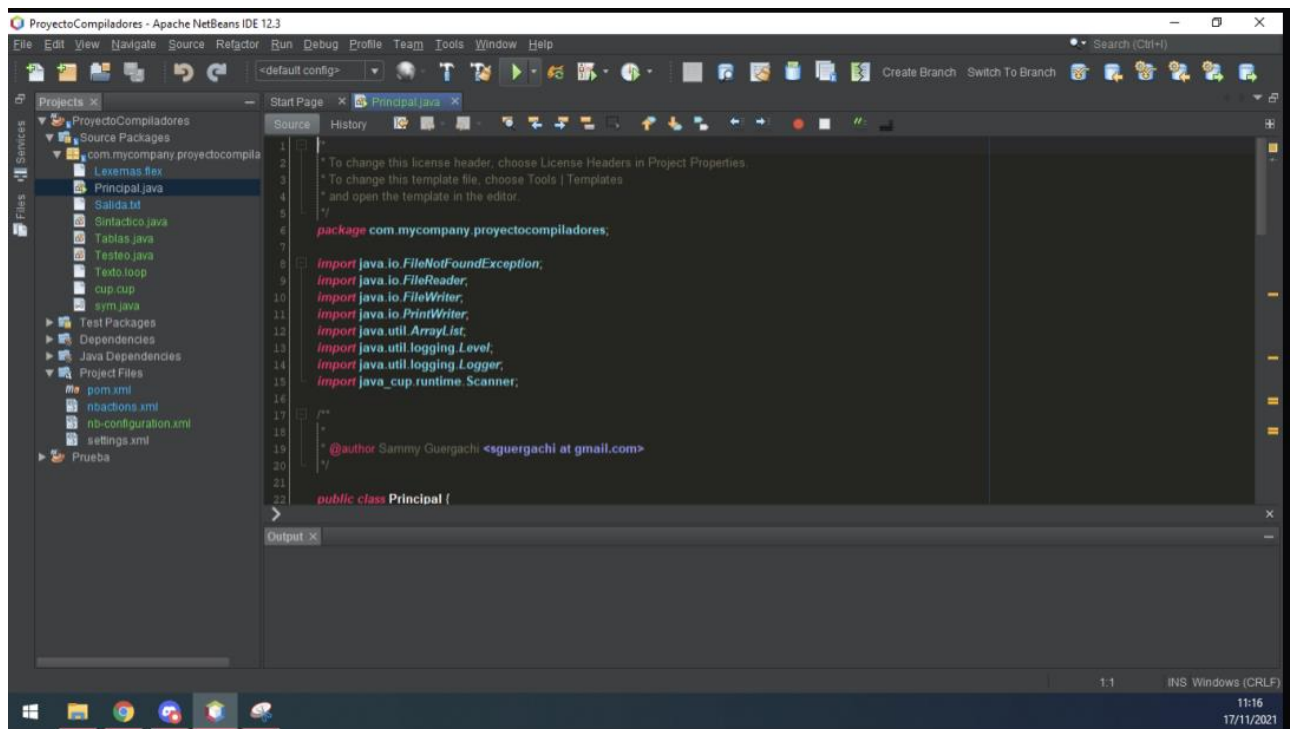
```
METODOS_OBJETOS ::= VARIABLE SIGNOS VARIABLE PARENTESIS1 VARIABLE_REPETIDAS  
PARENTESIS2  
|VARIABLE SIGNOS PARENTESIS1 PARENTESIS2;
```

MANUAL DE USUARIO

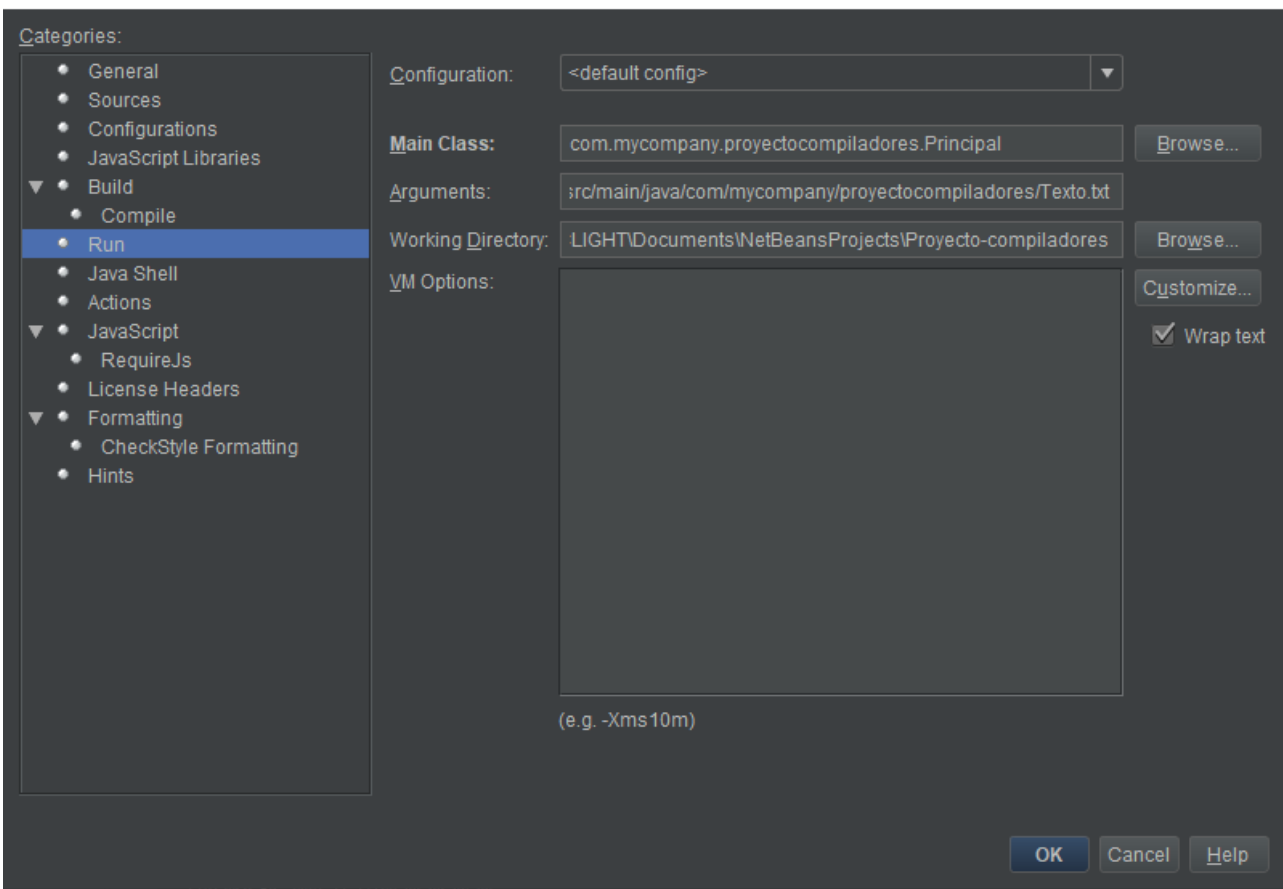
1. Abrimos NetBeans y nos ubicamos en la parte superior izquierda y daremos click en File y se nos desplegara una barra de tareas en la cual haremos click en OpenProject












2. A continuación abrimos el proyecto y procedemos a compilarlo en netbeans



Project Properties - ProyectoCompiladores



- Al ejecutar va a leer el archivo entrada.txt y se va a generar un archivo ".txt" en la carpeta del proyecto

	cup.cup	17/11/2021 10:31	Archivo CUP	11 KB
	Lexemas.flex	17/11/2021 06:58	Archivo FLEX	14 KB
	Principal.java	9/11/2021 10:29	Archivo de origen ...	3 KB
<input checked="" type="checkbox"/>	Salida.txt	17/11/2021 10:33	Documento de te...	1 KB
	Sintactico.java	9/11/2021 11:09	Archivo de origen ...	68 KB
	sym.java	9/11/2021 11:09	Archivo de origen ...	5 KB
	Tablas.java	9/11/2021 09:31	Archivo de origen ...	1 KB
	Testeo.java	9/11/2021 09:31	Archivo de origen ...	54 KB
	Testeo.java~	9/11/2021 09:30	Archivo JAVA~	54 KB
	Texto.loop	17/11/2021 10:50	Archivo LOOP	1 KB

- En este archivo se encontrará todos los tokens que se encontraran mostrándonos la palabra encontrada, el lexema con el que coincide y la línea y columna en la que se encuentra.

```

-----< com.mycompany:ProyectoCompiladores >-----
Building ProyectoCompiladores 1.0-SNAPSHOT
-----[ jar ]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ ProyectoCompiladores ---
Encontre un de clase
Encontre un de clase
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre Metodos
Encontre un de clase
Encontre un de clase
Encontre un de clase
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre metodo
Encontre variable
Encontre Metodos
Encontre Metodos
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre Metodos
Encontre un de clase
Encontre un de clase
Encontre un de clase
BUILD SUCCESS
Total time: 1.503 s
Finished at: 2021-11-17T22:33:53-06:00

```

- En caso que sea desde consola abrimos el CMD, una vez dentro usamos el comando `cd` para dirigirnos al archivo del proyecto y en esta dirigimos a la carpeta `target` en la cual encontraremos un archivo `jar`.

Nombre	Fecha de modificación	tipo	tamaño
archive-tmp	9/11/2021 10:29	Carpeta de archivos	
classes	17/11/2021 10:41	Carpeta de archivos	
generated-sources	9/11/2021 10:29	Carpeta de archivos	
maven-archiver	9/11/2021 10:29	Carpeta de archivos	
maven-status	9/11/2021 10:29	Carpeta de archivos	
test-classes	17/11/2021 10:41	Carpeta de archivos	
ProyectoCompiladores-1.0-SNAPSHOT-jar-with-dependencies.jar	9/11/2021 10:29	Executable Jar File	27 KB
ProyectoCompiladores-1.0-SNAPSHOT-jar-with-dependencies.jar	9/11/2021 10:29	Executable Jar File	49 KB

- Utilizaremos desde la consola CMD y nos situaremos en `target`, con el comando `java -jar ProyectoCompiladores-1.0-SNAPSHOT-jar-with-dependencies.jar` para próximo ejecutarlo

```
C:\Users\ARCTICLIGHT\Documents\NetBeansProjects\Proyecto-compiladores\target>java -jar ProyectoCompiladores-1.0-SNAPSHOT-jar-with-dependencies.jar
nov 17, 2021 11:37:10 PM com.mycompany.proyectocompiladores.Principal main
GRAVE: null
java.io.FileNotFoundException: src\main\java\com\mycompany\proyectocompiladores\Texto.loop (El sistema no puede encontrar la ruta especificada)
    at java.io.FileInputStream.open0(Native Method)
    at java.io.FileInputStream.open(Unknown Source)
    at java.io.FileInputStream.<init>(Unknown Source)
    at java.io.FileInputStream.<init>(Unknown Source)
    at java.io.FileReader.<init>(Unknown Source)
    at com.mycompany.proyectocompiladores.Principal.main(Principal.java:31)
C:\Users\ARCTICLIGHT\Documents\NetBeansProjects\Proyecto-compiladores\target>
```

- También podemos utilizar un archivo `.bat` en el cual implementaremos con un comando el cual es `java -jar target/ProyectoCompiladores-1.0-SNAPSHOT-jar-with-dependencies.jar`
 Pause
 El cual nos permitirá ejecutar el programa sin necesidad de tener netbeans o entrar por CMD

```
C:\Windows\system32\cmd.exe
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre Metodos
Encontre un de clase
Encontre un de clase
Encontre un de clase
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre metodo
Encontre variable
Encontre Metodos
Encontre Metodos
Encontre Variable
Encontre metodo
Encontre Sentencias
Encontre metodo
Encontre variable
Encontre Metodos
Encontre un de clase
Encontre un de clase
Encontre un de clase
C:\Users\ARCTICLIGHT\Documents\NetBeansProjects\Proyecto-compiladores>pause
Presione una tecla para continuar . . .
```