

Manual tecnico

Compiladores 1

MINIMAL REACT CMS

proyecto 1

Organización del Proyecto

Tienes dos principales áreas de análisis:

1. **Análisis sHTTP** → Procesa las solicitudes sHTTP verificando su estructura de las distintas peticiones.
2. **Análisis sCL** → Procesa la solicitud o instrucción que se quiere realizar en el servidor.
3. **Análisis toml** → Procesa la estructura de la página.
4. **Análisis minimal react** → Procesa el lenguaje para poder determinar si está correcto, y luego transpilarlo.

Tecnologías usadas:

- Antlr4
- jflex
- java Cup parse
- lenguaje java
- WebSockets

Estructura de Clases y Relaciones

1. Análisis sHTTP

- Usa la clase sHTTP AnalizadorLexico_sHTTP → se encarga de analizar los distintos tokens posibles.
- Usa la clase sCL AnalizadorSintactico_sHTTP → se encarga de analizar la sintaxis de la petición.

2. Análisis sCL

- Usa la clase AnalizadorLexico_sCL → se encarga de analizar los distintos tokens posibles.
- Usa la clase AnalizadorLexico_sCL → se encarga de analizar la sintaxis de la instrucción.

3. Análisis toml

- Usa una clase toml → para interpretar el archivo toml.

4. Análisis minimal react

- Usa clase TypeScriptLexer→ para reconocer los distintos tokens.
- Una clase TypeScriptParse→ para analizar la sintaxis del lenguaje.
- Una clase Listeners→ para ir recogiendo las instrucciones valida e invalidas.

Explicación del Flujo de programa

1. Llega la petición al servidor.
2. Se envía el empaquetado a la clase flujo.
3. Se descompone el empaquetado en tres partes, el body, el encabezado http, y la instrucción scl.
4. Se envían a la clase compilador, y se envían a sus respectivos, compiladores.
5. Se analiza la semántica de la petición y la semántica de la instrucción.
6. Luego se analiza el body si se requiere.
7. Si existen errores, se mandará una lista de errores, de lo contrario el análisis es exitoso.
8. Se envía el response en un json, y se envía al editor.

Tokens

Tokens o lexers sHTTP:

```
package Archivos_sHTTP;

import java_cup.runtime.Symbol;

%%

%{
    private String errorToken = "";
    private int errorLinea = 0;
    private int errorColumna = 0;
}%

%init{
    yyline=0;
    yycolumn=0;
%init}

%cup
%class analizadorLexico_sHTTP
%public
%line
%char
%column
%full
%debug

espacionBlanco = [ \t\f\r]+
POST="POST"
GET="GET"
SITIO="SITIO"
SITIOS="SITIOS"
PAGINA="PAGINA"
PATCH="PATCH"
PAGINAS="PAGINAS"
PAGINASITIOS="PAGINASITIOS"
%%

<YYINITIAL> {espacionBlanco} { }

<YYINITIAL> {POST} {
    return new java_cup.runtime.Symbol(sym.POST, yyline, yycolumn,
yytext().trim());
}
```

```

<YYINITIAL> {GET} {
    return new java_cup.runtime.Symbol(sym.GET, yyline, yycolumn,
yytext().trim());
}

<YYINITIAL> {PATCH} {
    return new java_cup.runtime.Symbol(sym.PATCH, yyline, yycolumn,
yytext().trim());
}

<YYINITIAL> {SITIO} {
    return new java_cup.runtime.Symbol(sym.SITIO, yyline, yycolumn,
yytext().trim());
}

<YYINITIAL> {SITIOS} {
    return new java_cup.runtime.Symbol(sym.SITIOS, yyline, yycolumn,
yytext().trim());
}

<YYINITIAL> {PAGINA} {
    return new java_cup.runtime.Symbol(sym.PAGINA, yyline, yycolumn,
yytext().trim());
}
<YYINITIAL> {PAGINAS} {
    return new java_cup.runtime.Symbol(sym.PAGINAS, yyline, yycolumn,
yytext().trim());
}
<YYINITIAL> {PAGINASITIOS} {
    return new java_cup.runtime.Symbol(sym.PAGINASITIOS, yyline, yycolumn,
yytext().trim());
}

. {
    errorToken = yytext();
    errorLinea = yyline ;
    errorColumna = yycolumn+1;
    if(yyline==0){
        errorLinea=yyline+1;
    }
    System.out.println("Error léxico detectado: '" + errorToken + "' en
línea " + errorLinea + ", columna " + errorColumna);
    throw new RuntimeException("Error léxico: '" + errorToken + "' en línea
" + errorLinea + ", columna " + errorColumna);
}

```

Tokens o lexers sCL:

```
package Archivos_sCL;
```

```
import java_cup.runtime.Symbol;
```

```
parser code {:
```

```
    public int accion=0;
    public int objetivo=0;
    public boolean instruccionValida = false;
    public String mensaje = "";
    public String orde = "";
    public String instrccion = "";
```

```
};
```

```
action code {:
```

```
};
```

```
terminal String CREAR, OBTENER, SITIO,AGREGAR,
PAGINA,PAGINAS,PAGINASITIOS,SITIOS, PARAMETRO, COMA,DELETE, MODIFICAR ,
TRANPILAR ;
```

```
nonterminal INICIO;
nonterminal INSTRUCCION, INSTRUCCIONES;
nonterminal PARAMETROS;
start with INICIO;
```

```
INICIO::=INSTRUCCIONES { :RESULT = instrccion; :};
```

```
INSTRUCCIONES ::= INSTRUCCIONES INSTRUCCION
                | INSTRUCCION
                | error
                { :
                  mensaje="  ERROR DE SINTAXIS";
                  instrccion = "Error de sintaxis: instrucción inválida";
                  instruccionValida = false;
                  RESULT = null;
                :};
```

```
INSTRUCCION ::= CREAR:c SITIO:s PARAMETRO:pa
{ :
```

```

    orde = c + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=1;
    objetivo=2;
    mensaje=" COMPILACION EXITOSA: crear sitio";
:}
| CREAR:c PAGINA:p PARAMETRO:pa
{:
    orde = c + p + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=1;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: crear pagina";
:}
;

```

```

PARAMETROS::=PARAMETRO COMA PARAMETRO
            |PARAMETRO;

```

```

INSTRUCCION ::= AGREGAR:a PAGINA:s PARAMETROS:pa
{:
    orde = a + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=2;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: AGREGAR PAGINA";
:}
;

```

```

INSTRUCCION ::= MODIFICAR:a PAGINA:s PARAMETROS:pa
{:
    orde = a + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=3;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: MODIFICAR PAGINA";
:}

```

;

INSTRUCCION ::= TRANPILAR:a PAGINA:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=6;

objetivo=1;

mensaje=" COMPILACION EXITOSA: TRAMPILAR PAGINA";

:}

;

INSTRUCCION ::= DELETE:a PAGINA:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=4;

objetivo=1;

mensaje=" COMPILACION EXITOSA: DELETE PAGINA";

:}

|DELETE:a SITIO:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=4;

objetivo=2;

mensaje=" COMPILACION EXITOSA: DELETE PAGINA";

:}

;

INSTRUCCION ::= OBTENER:o SITIO:s PARAMETRO:pa

{:

orde= o+s+pa+"\n";

instruccionValida = true;

instrccion = orde;

accion=5;

objetivo=2;

mensaje=" COMPILACION EXITOSA: OBTENER SITIO";


```

:}
|
OBTENER:o PAGINA:p PARAMETROS:pa
{:
    orde= o+p+pa+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINA";
:}
|
OBTENER:o PAGINASITIOS:p PARAMETRO:pa
{:
    orde= o+p+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=5;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINAS";
:}
|
OBTENER:o SITIOS:s
{:
    orde= o+s+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=3;
    mensaje=" COMPILACION EXITOSA: OBTENER SITIOS";
:}
|
OBTENER:o PAGINAS:p
{:
    orde= o+p+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=4;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINAS";
:}
;

```

Tokens o lexers MinimalReact:

LEXERTipeScript////////////////////////////////////

```
MAS : '+';
MENOS : '-';
MUL : '*';
DIV : '/';
POT : '^';
CONST: 'const';
REGRESO: 'return';
COMA: ',';
PAR1: '(';
PAR2: ')';
ASIGNACIO: '=';
ASIGNACIO_PAGINA: '=>';
CORCH1: '{';
CORCH2: '}';
MAYOR_QUE: '>';
MENOR_QUE: '<';
DIFERENTE: '!=';
FINI_NSTRUCCION: ';';
DECIMAL: [0-9]+'.'[0-9]+;
ENTERO: [0-9]+;
DOS_PUNTOS: ':';
CADENA: '"'.*?'"';
CARACTER : '\\'. '\\';
IDENTIFICADOR: [a-zA-Z]+[\\_a-zA-Z0-9]*;
WS: [ \\t\\r\\n]+ -> skip ;

//##### palabras reservadas
IF: 'if';
ELSE: 'else';
VAR: 'var';
CONSOLE: 'console';
LOG: 'log';
FUNCTION: 'function';

//tipos
NUMBER: 'number';
```

```
STRING:'string';  
CHAR:'char';  
BOOLEAN:'boolean';  
VOID:'void';
```

```
//booleanos  
TRUE:'true';  
FALSE:'false';
```

```
////////////////////////////////////
```

```
//Lexer HTML //////////////////////////////////////
```

```
MAIN:'main';  
HEADERS: 'h'[1-6];  
PARRAFO:'p';  
BOTON:'button';  
ONCLICK:'onClick';  
INPUT:'input';
```

```
////////////////////////////////////
```

Gramáticas

Gramatica sHTTP

```
package Archivos_sHTTP;
import java_cup.runtime.Symbol;
parser code {:

    public int metodo=0;
    public int objetivo=0;
    public boolean instruccionValida = false;
    public String mensaje = "";
    public String orde = "";
    public String instrccion = "";
:}

terminal String POST, SITIO, SITIOS, PAGINA,PAGINAS,PAGINASITIOS , GET,
PATCH;
nonterminal INICIO;
nonterminal INSTRUCCION;

start with INICIO;

INICIO ::= INSTRUCCION
    {:
        RESULT = instrccion;
    :}
|error
    {:
        mensaje=" ERROR DE SINTAXIS";
        instrccion = "Error de sintaxis: instrucción inválida";
        instruccionValida = false;
        RESULT = null;
    :};

INSTRUCCION ::= POST:p SITIO:s
    {:
        orde = p + s + "\n";
        instruccionValida = true;
        metodo=1;
        objetivo=2;
        instrccion = orde;
        mensaje="COMPILACION EXITOSA: POST SITIO";
```

```

:}
| POST:p PAGINA:pa
{:
    orde = p + pa + "\n";
    instruccionValida = true;
    metodo=1;
    objetivo=1;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: POST PAGINA";
:}
;

```

INSTRUCCION ::= GET:p SITIO:s

```

{:
    orde = p + s + "\n";
    instruccionValida = true;
    metodo=2;
    objetivo=2;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: GET SITIO";
:}
| GET:p PAGINA:pa
{:
    orde = p + pa + "\n";
    instruccionValida = true;
    metodo=2;
    objetivo=1;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: GET PAGINA";
:}
;

```

INSTRUCCION ::= PATCH:p SITIO:s

```

{:
    orde = p + s + "\n";
    instruccionValida = true;
    metodo=3;
    objetivo=2;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: PATCH SITIO";
:}
| PATCH:p PAGINA:pa
{:

```

```

    orde = p + pa + "\n";
    instruccionValida = true;
    metodo=3;
    objetivo=1;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: PATCH PAGINA";
:}
;

```

INSTRUCCION ::= GET:p SITIOS:s

```

{:
    orde = p + s + "\n";
    instruccionValida = true;
    metodo=2;
    objetivo=3;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: GET SITIOS";
:}
|
GET:g PAGINAS:p
{:
    orde = g + p + "\n";
    instruccionValida = true;
    metodo=2;
    objetivo=4;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: GET PAGINAS";
:}
|
GET:g PAGINASITIOS:p
{:
    orde = g + p + "\n";
    instruccionValida = true;
    metodo=2;
    objetivo=5;
    instrccion = orde;
    mensaje=" COMPILACION EXITOSA: GET PAGINASITIO";
:}
;

```

Gramática sCL

```
package Archivos_sCL;
```

```
import java_cup.runtime.Symbol;
```

```
parser code {:
```

```
    public int accion=0;
    public int objetivo=0;
    public boolean instruccionValida = false;
    public String mensaje = "";
    public String orde = "";
    public String instrccion = "";
```

```
};
```

```
action code {:
```

```
};
```

```
terminal String CREAR, OBTENER, SITIO,AGREGAR,
PAGINA,PAGINAS,PAGINASITIOS,SITIOS, PARAMETRO, COMA,DELETE, MODIFICAR ,
TRANPILAR ;
```

```
nonterminal INICIO;
nonterminal INSTRUCCION, INSTRUCCIONES;
nonterminal PARAMETROS;
start with INICIO;
```

```
INICIO::=INSTRUCCIONES { :RESULT = instrccion;:};
```

```
INSTRUCCIONES ::= INSTRUCCIONES INSTRUCCION
```

```
    | INSTRUCCION
```

```
    | error
```

```
    {:
```

```
        mensaje="    ERROR DE SINTAXIS";
```

```
        instrccion = "Error de sintaxis: instrucción inválida";
```

```
        instruccionValida = false;
```

```
        RESULT = null;
```

```
    :};
```

```
INSTRUCCION ::= CREAR:c SITIO:s PARAMETRO:pa
```

```
    {:
```

```

    orde = c + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=1;
    objetivo=2;
    mensaje=" COMPILACION EXITOSA: crear sitio";
:}
| CREAR:c PAGINA:p PARAMETRO:pa
{:
    orde = c + p + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=1;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: crear pagina";
:}
;

```

```

PARAMETROS::=PARAMETRO COMA PARAMETRO
          |PARAMETRO;

```

```

INSTRUCCION ::= AGREGAR:a PAGINA:s PARAMETROS:pa
{:
    orde = a + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=2;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: AGREGAR PAGINA";
:}
;

```

```

INSTRUCCION ::= MODIFICAR:a PAGINA:s PARAMETROS:pa
{:
    orde = a + s + pa + "\n";
    instruccionValida = true;
    instrccion = orde;
    accion=3;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: MODIFICAR PAGINA";
:}

```


;

INSTRUCCION ::= TRANPILAR:a PAGINA:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=6;

objetivo=1;

mensaje=" COMPILACION EXITOSA: TRAMPILAR PAGINA";

:}

;

INSTRUCCION ::= DELETE:a PAGINA:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=4;

objetivo=1;

mensaje=" COMPILACION EXITOSA: DELETE PAGINA";

:}

|DELETE:a SITIO:s PARAMETROS:pa

{:

orde = a + s + pa + "\n";

instruccionValida = true;

instrccion = orde;

accion=4;

objetivo=2;

mensaje=" COMPILACION EXITOSA: DELETE PAGINA";

:}

;

INSTRUCCION ::= OBTENER:o SITIO:s PARAMETRO:pa

{:

orde= o+s+pa+"\n";

instruccionValida = true;

instrccion = orde;

accion=5;

objetivo=2;

mensaje=" COMPILACION EXITOSA: OBTENER SITIO";

```

:}
|
OBTENER:o PAGINA:p PARAMETROS:pa
{:
    orde= o+p+pa+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=1;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINA";
:}
|
OBTENER:o PAGINASITIOS:p PARAMETRO:pa
{:
    orde= o+p+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=5;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINAS";
:}
|
OBTENER:o SITIOS:s
{:
    orde= o+s+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=3;
    mensaje=" COMPILACION EXITOSA: OBTENER SITIOS";
:}
|
OBTENER:o PAGINAS:p
{:
    orde= o+p+"\n";
    instruccionValida = true;
    instrccion = orde;
    accion=5;
    objetivo=4;
    mensaje=" COMPILACION EXITOSA: OBTENER PAGINAS";
:}
;

```

Gramática TypeScript

```
//parserTipeScript
inicio: declaracion_pagina;

declaracion_pagina:CONST IDENTIFICADOR ASIGNACIO PAR1 PAR2 ASIGNACIO_PAGINA CORCH1
bloque_declaracion_pagina? regreso CORCH2| ;

bloque_declaracion_pagina: instrucciones_tipe_scrip bloque_declaracion_pagina_prima
                           | declaracion_funcion bloque_declaracion_pagina_prima;

bloque_declaracion_pagina_prima: instrucciones_tipe_scrip
bloque_declaracion_pagina_prima
                           | declaracion_funcion bloque_declaracion_pagina_prima ;

instrucciones_tipe_scrip returns[Resultado value]:
    instruccion_tipo_script
    {
        if($instruccion_tipo_script.value==null)
        {
            $value = new Resultado();
            $value.setBandera0(true);
        }else{
            if($instruccion_tipo_script.value.isBandera0() )
            {
                $value = new Resultado();
                $value.setBandera0(true);
            }
        }
    } (instrucciones_tipe_scrip_prima
    {
        if($instruccion_tipo_script.value==null)
        {
            $value = new Resultado();
            $value.setBandera0(true);
        }else{

if($instruccion_tipo_script.value.isBandera0() )
        {
            $value = new Resultado();
            $value.setBandera0(true);
        }
    }
    )
;

instrucciones_tipe_scrip_prima returns[Resultado value]:
```

```

instruccion_tipo_script
{
    if($instruccion_tipo_script.value==null)
    {
        $value = new Resultado();
        $value.setBandera0(true);
    }else{
        if($instruccion_tipo_script.value.isBandera0() )
        {
            $value = new Resultado();
            $value.setBandera0(true);
        }
    }
} (intrucciones_tipe_scrip_prima
{
    if($instruccion_tipo_script.value==null)
    {
        $value = new Resultado();
        $value.setBandera0(true);
    }else{

if($instruccion_tipo_script.value.isBandera0() )
    {
        $value = new Resultado();
        $value.setBandera0(true);
    }
    }
}

)

| { $value = new Resultado();}
;

```

instruccion_tipo_script returns[Resultado value] locals[boolean error=false, String instruccion, boolean funcion=false, boolean if=false] :

```

creacionVariable
{
    if($creacionVariable.value==null){

        $value = new Resultado("ss");
        $value.setBandera0(true);

        errorif=true; errorEnFuncion=true;
    }else{

        if($creacionVariable.value.isBandera0() )

```

```

        {
            $value = new Resultado("ss");
$value.setBandera0(true);    errorEnFuncion=true;    errorif=true;

        }else{ $error=true;
$funcion=declaracionFuncionEnCurso; $if=declaracioniFEnCurso;}
        }
        an($screacionVariable.ctx,5);
    }

    |asignacionVariable
    {
        if($asignacionVariable.value==null){

            $value = new Resultado("ss");
$value.setBandera0(true);    errorif=true; errorEnFuncion=true;
        }
        else{
            if($asignacionVariable.value.isBandera0() )
            {
                $value = new Resultado("ss");
$value.setBandera0(true);    errorif=true; errorEnFuncion=true;

            }else{ $error=true;
$funcion=declaracionFuncionEnCurso; $if=declaracioniFEnCurso;}
            }
            an($asignacionVariable.ctx,6);
        }
        |impresion_consola
        {
            if($impresion_consola.value==null){

                $value = new Resultado("ss");
$value.setBandera0(true);errorif=true;    errorEnFuncion=true;
            }else{
                if($impresion_consola.value.isBandera1() ||
$impresion_consola.value.isBandera2() )
                {
                    $value = new Resultado("ss");
$value.setBandera0(true);    errorif=true; errorEnFuncion=true;

                }else{ $error=true;
$funcion=declaracionFuncionEnCurso; $if=declaracioniFEnCurso;}
                }
                an($impresion_consola.ctx,7);
            }
        |condicional {

```

```

        $error=false;
        if($condicional.value==null){

                $value = new Resultado("ss");
$value.setBandera0(true);  errorif=true; errorEnFuncion=true;

        }else{   System.out.println("3");

                if(errorif )
                {
                        $value = new Resultado("ss");
$value.setBandera0(true); errorif=true;errorEnFuncion=true;
                }else{
                        $error=true;
$funcion=declaracionFuncionEnCurso; $if=declaracioniFEnCurso;
                }
        }
    }

    |llamada_funcion

    {

        if($llamada_funcion.value==null){

                $value = new Resultado("ss");
$value.setBandera0(true); errorEnFuncion=true;

        }else{   if($llamada_funcion.value.isBandera0() )
                {
                        $value = new Resultado("ss");
$value.setBandera0(true);  errorEnFuncion=true;

                }else{ $error=true;
$funcion=declaracionFuncionEnCurso; $if=declaracioniFEnCurso;
                }
        }
    }
;

```

```

regreso_funcion  locals[boolean error1=false, boolean error2=false]: REGRESO
tipoAsignacionVariable?  FINI_NSTRUCCION

```

```

{
    Object valor=null;
    if(!returnFuncion)

```

```

{
    if (tablaSimbolosLocales.get ($REGRESO.text) != null) {
        if ($tipoAsignacionVariable.value != null) {

if ($tipoAsignacionVariable.value.value != null)

{
            if (!
tablaSimbolosLocales.get ($REGRESO.text).getClass().getSimpleName().equals ($tipoAsig
nacionVariable.value.value.getClass().getSimpleName()) ) {

valor=$tipoAsignacionVariable.value.value.getClass().getSimpleName()+" no
compatible con
"+tablaSimbolosLocales.get ($REGRESO.text).getClass().getSimpleName();

errorEnFuncion=true;  String mensaje =  $REGRESO.text + " "  +
$tipoAsignacionVariable.value.value + " " + $FINI_NSTRUCCION.text;

} else {

valor="Guardando en tabla: " + $REGRESO.text + " -> " +
$tipoAsignacionVariable.value.value;

tablaSimbolosLocales.put ($REGRESO.text, $tipoAsignacionVariable.value.value);
String

returnnn=$REGRESO.text+" "+$tipoAsignacionVariable.text+" ";

instrucciones.add(returnnn); tipoInstruccion.add(0); returnFuncion=true;

}

}

} else {

if ($tipoAsignacionVariable.ctx != null) {
    errorEnFuncion=true;

$error2=true;

} else {

String

returnnn=$REGRESO.text+" "; instrucciones.add(returnnn); tipoInstruccion.add(0);
returnFuncion=true;

}

}

} else {

errorEnFuncion=true;  $error1=true;

}

}

;

```

```

declaracion_funcion locals[boolean error=false]:
{
    tablaSimbolosLocales=new HashMap<String,Object>();
    instruccionesFuncion = new ArrayList<>();
    tipoInstruccionFuncion= new ArrayList<>();
    errorEnFuncion=false; returnFuncion=false;
    declaracionFuncionEnCurso=true;

}
f=FUNCTION id=IDENTIFICADOR p1=PAR1 parametros_funcion

p2=PAR2 dp=DOS_PUNTOS tipo{

    Object valor=null;

    switch ($tipo.value.toString()) {
        case "String" -> { valor="a"; }
        case "Character" -> {valor='a'; }
        case "Boolean" -> {valor=false; }
        case "Double" -> {valor=0.0; }
        case "void" -> { valor=null; }
    }

    tablaSimbolosLocales.put("return", valor );

} CORCH1{

    String encabezadoFucion=$f.text+" "+$id.text+" ("+$parametros_funcion.text+") : "+$tipo.text+" {";
    instrucciones.add(encabezadoFucion);
tipoInstruccion.add(8);

    } bloque_instrucciones CORCH2{

        if(errorEnFuncion){
            $error=true;
        }else{

            Object tipoRetorno=null;

            if(tablaSimbolosLocales.get("return")!=null){

                tipoRetorno =

                tablaSimbolosLocales.get("return");

            }

            List<Token> tokens =

            ((CommonTokenStream)_input)

            .getTokens($bloque_

            instrucciones.start.getTokenIndex(),

```



```

$bloque_instrucciones.stop.getTokenIndex());

Funcion miFuncion =new
Funcion($id.text,$parametros_funcion.parametros,tablaSimbolosLocales,
$bloque_instrucciones.ctx,tipoRetorno,tokens);

miFuncion.setInstrucciones(instruccionesFuncion);

miFuncion.setTipoInstruccion(tipoInstruccionFuncion);

tablaFunciones.put($id.text,miFuncion );
    }
    errorEnFuncion=false;
    tablaSimbolosLocales=null;
    instruccionesFuncion = null;
    tipoInstruccionFuncion= null;
    declaracionFuncionEnCurso=false;
    instrucciones.add("}");
tipoInstruccion.add(0);
    }

;

parametros_funcion returns[Map<String, String> parametros]:
    id=IDENTIFICADOR DOS_PUNTOS t=tipo
    {
        $parametros = new HashMap<String, String>();
        Object valor=null;

        switch ($t.value.toString()) {
            case "String" -> {valor="a"; }
            case "Character" -> { valor='a';}
            case "Boolean" -> {valor=false; }
            case "Double" -> {valor=0.0; }
        }

        $parametros.put($id.getText(), $t.value.toString());
        tablaSimbolosLocales.put($id.getText(), valor);
    }
( COMA id2=IDENTIFICADOR DOS_PUNTOS t2=tipo
    {
        if($parametros!=null){
            switch ($t.value.toString()) {
                case "String" -> { valor="a"; }
                case "Character" -> { valor='a'; }
            }
        }
    }

```

```

        case "Boolean" -> { valor=false;}
        case "Double" -> { valor=0.0; }
    }

    $parametros.put($id2.getText(), $t2.value.toString());
    tablaSimbolosLocales.put($id2.getText(), valor);
}

}

)*

;

bloque_instrucciones returns []
: instr=instrucciones_tipe_scrip  intr2=bloque_instrucciones_prima
|instr2=regreso_funcion bloque_instrucciones_prima
;

bloque_instrucciones_prima : instrucciones_tipe_scrip  bloque_instrucciones_prima |
regreso_funcion  bloque_instrucciones_prima| ;

llamada_funcion returns[Resultado value, boolean exito=true, Object retorno]
locals[boolean error1=false,boolean error4=false,boolean error2=false,boolean
error3=false , int parametros=0, String tiposParamalos="", String
tiposParama="",boolean error=false]:
    IDENTIFICADOR PAR1 llamada_parametros_funcion?  PAR2 FINI_NSTRUCCION

    {

        Map <String,Object> tablaEnTurno=null;
        if(tablaSimbolosLocales!=null){tablaEnTurno=tablaSimbolosLocales;}else
{tablaEnTurno=tablaSimbolos;}

        if(tablaFunciones.containsKey($IDENTIFICADOR.text)){

            Funcion funcion=(Funcion)tablaFunciones.get($IDENTIFICADOR.text);

            $retorno=funcion.getTipoRetorno();
            if(funcion.getParametros()!=null){
                $parametros=funcion.getParametros().size();
                $tiposParama=funcion.getParametros().values().toString();
            }

            if(_localctx.llamada_parametros_funcion!=null){

if(funcion.getParametros().size()==$llamada_parametros_funcion.parametros.size()){

```

```

        List<String> listaKeysParametros = new
ArrayList<>(funcion.getParametros().keySet());
        List<String> listaValoresParametros = new
ArrayList<>(funcion.getParametros().values());
        List<Object> listaValoresParametrosFuncion =
$llamada_parametros_funcion.parametros;

        for (int i = 0; i < listaValoresParametros.size();
i++) {

if(listaValoresParametros.get(i).equals(listaValoresParametrosFuncion.get(i).getCla
ss().getSimpleName())){

if(tablaEnTurno.containsKey($IDENTIFICADOR.text)){

funcion.getVariables().put(listaKeysParametros.get(i), listaValoresParametrosFuncion
.get(i));

}
}
else{

$value = new Resultado(null, "");
$value.setBandera0(true);
$tiposParamalos+=
listaValoresParametrosFuncion.get(i).toString();
$error4=true;
$exito=false;

}

}
$value = new Resultado(null, "ss");
}
else{
if(funcion.getParametros().size()==0){
$error2=true; $exito=false;
}
else{
$error3=true; $exito=false;

}

}
}
else{
if(funcion.getParametros()!=null){

if(funcion.getParametros().size()!=0){
$error3=true; $exito=false;

}

}

}
}
}
else{

```

```

        $error1=true; $exit0=false;
    }
}
;

llamada_parametros_funcion returns[ArrayList<Object> parametros]:
{
    $parametros = new ArrayList<>();
}
tv1=tipoAsignacionVariable
{
    if($tv1.value!=null){
        if($tv1.value.value!=null){
            $parametros.add($tv1.value.value);
        }
    }
}
( COMA    tv2=tipoAsignacionVariable
{
    if($tv2.value!=null){
        if($tv2.value.value!=null){
            $parametros.add($tv2.value.value);
        }
    }
}
)*
;

//
*****
*****
*****

condicional returns[Resultado value] locals[boolean error1=false, boolean
error2=false, boolean error3=false; , String Instvalido="", String elseIf=""]
:{returnif=false; $error1=false; $error2=false; errorif=false;
declaracioniFEnCurso=true; $value = new Resultado( ""); } IF PAR1
t2=tipoAsignacionVariable
{
    if($t2.value!=null){
        if($t2.value.value!=null)
        {
            if (!($t2.value.value instanceof Boolean)) {
                $error3=true; $value.setBandera0(true);
$Instvalido=$t2.text;
            }
        }
    }
}

```

```

        }else{
            $error2=true; $value.setBandera0(true); $Instvalido=$t2.text;
        }
    }
    PAR2 CORCH1 {String encabezado=$IF.text+"("+ $t2.text+"){";
instrucciones.add(encabezado); tipoInstruccion.add(9);}
    bloque_instrucciones_condicional? CORCH2{instrucciones.add(")");
tipoInstruccion.add(0);} (ELSE (IF PAR1 t=tipoAsignacionVariable {
        if($t.value!=null){
            if($t.value.value!=null)
                {
                    if (!( $t.value.value instanceof Boolean)) {
                        $error3=true; $value.setBandera0(true);
$Instvalido=$t.text;
                    }}
                }else{
                    $error2=true;
                    $value.setBandera0(true);
                    $Instvalido=$t.text;
                }
            } PAR2 { $selfIf=$IF.text+"("+ $t.text+"){"; }? CORCH1 {String
encabezadoElse=$ELSE.text+" "+$selfIf+"{"; instrucciones.add(encabezadoElse);
tipoInstruccion.add(10);}
        {
        }
        bloque_instrucciones_condicional? CORCH2 {instrucciones.add(")");
tipoInstruccion.add(0);} )?
        { $error1=errorif;

            declaracioniFEnCurso=false;
            if($value==null)
            { Object valor=null ; $value = new Resultado(valor, "");
            if(errorif){ $value.setBandera0(true);}
            }
        }
    }
;

bloque_instrucciones_condicional returns[Resultado value]
    : instr=instrucciones_tipe_scrip intr2=bloque_instrucciones_condicional_prima

    |instr2=regreso_if bloque_instrucciones_condicional_prima
;

bloque_instrucciones_condicional_prima : instrucciones_tipe_scrip
bloque_instrucciones_condicional_prima | regreso_if
bloque_instrucciones_condicional_prima| ;

```

```

regreso_if locals[boolean error1=false, boolean error2=false,boolean
error3=false]: REGRESO tipoAsignacionVariable? FINI_NSTRUCCION

{
    Object valor=null;
    if(tablaSimbolosLocales!=null)
    {
        if(!returnif)
        {
            if(tablaSimbolosLocales.get($REGRESO.text)!=null){
                if($tipoAsignacionVariable.value!=null){

if($tipoAsignacionVariable.value.value!=null)

{
                                if (!
tablaSimbolosLocales.get($REGRESO.text).getClass().getSimpleName().equals($tipoAsig
nacionVariable.value.value.getClass().getSimpleName())){

valor=$tipoAsignacionVariable.value.value.getClass().getSimpleName()+" no
compatible con
"+tablaSimbolosLocales.get($REGRESO.text).getClass().getSimpleName();
                                errorif=true;
                                String mensaje =
$REGRESO.text + " " + $tipoAsignacionVariable.value.value + " " +
$FINI_NSTRUCCION.text;
                                }else{

valor="Guardando en tabla: " + $REGRESO.text + " -> " +
$tipoAsignacionVariable.value.value;

tablaSimbolosLocales.put($REGRESO.text, $tipoAsignacionVariable.value.value);
                                String
returnn=$REGRESO.text+" "+$tipoAsignacionVariable.text+" ";

instrucciones.add(returnn); tipoInstruccion.add(0); returnif=true;
                                }
                                }
                                }
                                }else{
                                if ($tipoAsignacionVariable.ctx != null) {
                                    errorif=true;$error2=true;
                                }else{
                                    returnif=true; String
returnn=$REGRESO.text+""; instrucciones.add(returnn); tipoInstruccion.add(0);
                                }
                                }
}

```

```

        } else{
            errorif=true; $error1=true;
        }
    }else{
        $error3=true;errorif=true;
    }
}

;
//
*****
*****
*****

```

```

impresion_consola returns[Resultado value]:
    CONSOLE '.' LOG PAR1 t1=tipo_impresion t2=tipos_impresion PAR2 FINI_NSTRUCCION
{
    Object valor = $t1.text+" "+$t2.value ;
    $value = new Resultado(valor, "ss");
    $value.setBandera2(true);
}
|CONSOLE '.' LOG PAR1 tipo_impresion PAR2 FINI_NSTRUCCION
{
    if($tipo_impresion.value != null && $tipo_impresion.value.value != null){
        Object valor = null;
        $value = new Resultado(valor, "ss");
        if($tipo_impresion.value.isBandera1()){
            $value.setBandera1(true);
        }
    } else{
        Object valor = null;
        $value = new Resultado(valor, "ss");
        $value.setBandera1(true);
    }
}
;

```

```

tipos_impresion returns[String value]: tipo_impresion tipos_impresiones_prima
{ $value= $tipo_impresion.text+" "+$tipo_impresiones_prima.text;      } ;

```

```

tipo_impresiones_prima returns[String value]: tipo_impresion
tipo_impresiones_prima { $value= String.valueOf($tipo_impresion.text);};

```

```

tipo_impresion returns[Resultado value]:

```

```

CADENA
{
    Object valor = null;
    valor=$CADENA.text ;
    $value = new Resultado(valor, "");
}

| IDENTIFICADOR
{   Map <String,Object> tablaEnTurno=null;

    if(tablaSimbolosLocales!=null)
    {
        tablaEnTurno=tablaSimbolosLocales;
        if (!
tablaEnTurno.containsKey($IDENTIFICADOR.text)) {
            tablaEnTurno=tablaSimbolos;
        }

    }else
    {

        tablaEnTurno=tablaSimbolos;
    }

    boolean ban1=false;
    Object valor = null;
    if (!$tablaEnTurno.containsKey($IDENTIFICADOR.text)) {

        ban1=true;
    } else {
        valor = tablaEnTurno.get($IDENTIFICADOR.text);
    }
    $value = new Resultado(valor, "");
    $value.setBandera1(ban1);

}

|expresion
{

    Object valor = null;

```



```

        if($expresion.value!=null){

            valor=$expresion.value.value ;
        }
        $value = new Resultado(valor, "");

    }

|CHAR
    {

        Object valor = null;
        valor=(String)$CHAR.text ;
        $value = new Resultado(valor, "");

    }

|tipoBoleano
    {

        Object valor = null;
        valor= Boolean.parseBoolean($tipoBoleano.text);
        $value = new Resultado(valor, "");

    }

;

creacionVariable returns[Resultado value]:

    VAR IDENTIFICADOR DOS_PUNTOS tipo ASIGNACIO    tipoAsignacionVariable
FINI_NSTRUCCION

    {

        Map <String,Object> tablaEnTurno=null;

        if(tablaSimbolosLocales!=null)
        {
            tablaEnTurno=tablaSimbolosLocales;

        }else
        {

            tablaEnTurno=tablaSimbolos;

        }

        String valor = null;

        if($tipoAsignacionVariable.value!=null){

```

```

        if($tipoAsignacionVariable.value.value!=null)
        {
            if (!
$tipo.value.equals($tipoAsignacionVariable.value.value.getClass().getSimpleName()))
{

valor=$tipoAsignacionVariable.value.value.getClass().getSimpleName()+" no
compatible con "+$tipo.value;

String mensaje = $VAR.text + " " +
$IDENTIFICADOR.text + " " +
+
$DOS_PUNTOS.text + " " + $tipo.value + " "
$ASIGNACIO.text + " " +
$tipoAsignacionVariable.value.value + " " +
$FINI_NSTRUCCION.text;

$value = new Resultado(mensaje);
$value.añadirMensaje(valor);
$value.setBandera1(true);
$value.setBandera0(true);
}else{

if (tablaEnTurno.containsKey($IDENTIFICADOR.text)) {

$value = new
Resultado(valor, $IDENTIFICADOR.text);
$value.setBandera5(true);
$value.setBandera0(true);
}else{

valor="Guardando en tabla:
" + $IDENTIFICADOR.text + " -> " + $tipoAsignacionVariable.value.value;

tablaEnTurno.put($IDENTIFICADOR.text, $tipoAsignacionVariable.value.value);
$value = new
Resultado(valor, $IDENTIFICADOR.text);
}

}

}
}

```

```

        }
        | VAR IDENTIFICADOR DOS_PUNTOS ASIGNACIO tipoAsignacionVariable
FINI_NSTRUCCION
        {
            String topoAsig="----";
            if($tipoAsignacionVariable.value!=null){

topoAsig=$tipoAsignacionVariable.value.value.toString();
            }
            String mensaje = $VAR.text + " " + $IDENTIFICADOR.text + "
" +

            $DOS_PUNTOS.text + " " +
            $ASIGNACIO.text + " " + topoAsig + " " +
            $FINI_NSTRUCCION.text;
            Object valor="tipo no reconocido";
            $value = new Resultado(valor, mensaje);
            $value.setBandera2(true);
            $value.setBandera0(true);
        }
        |VAR IDENTIFICADOR DOS_PUNTOS tipo ASIGNACIO FINI_NSTRUCCION
        {
            String mensaje = $VAR.text + " " + $IDENTIFICADOR.text + "
" +

            $DOS_PUNTOS.text + " " + $tipo.value + " "+
            $ASIGNACIO.text + " " + $FINI_NSTRUCCION.text;

            Object valor="nada asignado no reconocido";
            $value = new Resultado(valor, mensaje);
            $value.setBandera3(true); $value.setBandera0(true);
        }
        | VAR DOS_PUNTOS tipo ASIGNACIO tipoAsignacionVariable FINI_NSTRUCCION
        {

            String mensaje = $VAR.text + " " + $DOS_PUNTOS.text + " " +
            $tipo.value + " " + $ASIGNACIO.text + " " + $tipoAsignacionVariable.value.value +
            " " +$FINI_NSTRUCCION.text;
            Object valor="ID no reconocido"; $value = new
Resultado(valor, mensaje); $value.setBandera4(true); $value.setBandera0(true);
        }

;

asignacionVariable returns[Resultado value]:

        IDENTIFICADOR ASIGNACIO tipoAsignacionVariable FINI_NSTRUCCION

        {

```

```

        boolean funcion=false;
        Map <String,Object> tablaEnTurno=null;

        if(tablaSimbolosLocales!=null)
        {
            tablaEnTurno=tablaSimbolosLocales;
            if (!
tablaEnTurno.containsKey($IDENTIFICADOR.text)) {
                String mensaje= $IDENTIFICADOR.text; $value
= new Resultado(mensaje);    $value.setBandera0(true); $value.setBandera2(true);
tablaEnTurno=tablaSimbolos;
            }
        }else
        {
            tablaEnTurno=tablaSimbolos;
        }

        if (!tablaEnTurno.containsKey($IDENTIFICADOR.text)) {
            String mensaje= $IDENTIFICADOR.text;    $value = new
Resultado(mensaje);    $value.setBandera0(true);$value.setBandera2(true);
        } else {
            String valor = null;
            if($tipoAsignacionVariable.value!=null){

                if($tipoAsignacionVariable.value.value!
=null)

                {
                    if (!
tablaEnTurno.get($IDENTIFICADOR.text).getClass().getSimpleName().equals($tipoAsigna
cionVariable.value.value.getClass().getSimpleName())){

valor=$tipoAsignacionVariable.value.value.getClass().getSimpleName()+" no
compatible con "+tablaEnTurno.get($IDENTIFICADOR.text).getClass().getSimpleName();

                    String mensaje =
$IDENTIFICADOR.text + " " +
$ASIGNACIO.text + " " +
$tipoAsignacionVariable.value.value + " " +
$FINI_NSTRUCCION.text;

                    $value = new Resultado(mensaje);
$value.añadirMensaje(valor);    $value.setBandera1(true); $value.setBandera0(true);
                }else{
                    valor="Guardando en tabla:
" + $IDENTIFICADOR.text + " -> " + $tipoAsignacionVariable.value.value;

```

```

tablaEnTurno.put ($IDENTIFICADOR.text, $tipoAsignacionVariable.value.value);
                                                                    $value = new
Resultado(valor, "");
                                                                    }
                                                                    }

                                                                    }else{
                                                                    $value = new Resultado();
                                                                    $value.setBandera3(true);
                                                                    $value.setBandera0(true);
                                                                    }
                                                                    }
                                                                    };

tipoAsignacionVariable returns[Resultado value] locals[boolean errorId=false]:

    CADENA
    {
        Object valor = null;
        valor=(String)$CADENA.text ;
        $value = new Resultado(valor, "");
    }

    | CARACTER
    {
        Object valor = null;
        valor=$CARACTER.text.charAt(0);
        $value = new Resultado(valor, "");
    }

    | tipoBoleano
    {
        Object valor = null;
        valor= Boolean.parseBoolean($tipoBoleano.text);
        $value = new Resultado(valor, "");
    }

    | IDENTIFICADOR
    {

System.out.println("ingrese se algo"+$IDENTIFICADOR.text);
        Map <String,Object> tablaEnTurno=null;

        if(tablaSimbolosLocales!=null)
        {
            tablaEnTurno=tablaSimbolosLocales;
            if (!tablaEnTurno.containsKey($IDENTIFICADOR.text))
{

```


expresion returns [Resultado value]:

```
t1=terminoN2
{
    Object valort1 = null;
    if($t1.value.value!=null){
        valort1 = $t1.value.value;
        $value = new Resultado(valort1, "");
    }
    $value = new Resultado(valort1, "");
}
(MAS t2=terminoN2
{ Object valor = null;
    if($t2.value.value!=null && $value.value!=null){
        valor = (Double)$value.value+ (Double)
$t2.value.value; $value = new Resultado(valor, "");
    }
    $value = new Resultado(valor, "");
}
)*
|t1=terminoN2
{
    Object valort1 = null;
    if($t1.value.value!=null){
        valort1 = $t1.value.value;    $value = new
Resultado(valort1, "");
    }
    $value = new Resultado(valort1, "");
}
(MENOS t2=terminoN2
{ Object valor = null;
    if($t2.value.value!=null && $value.value!=null){
        valor = (Double)$value.value- (Double)
$t2.value.value;    $value = new Resultado(valor, "");
    }
    $value = new Resultado(valor, "");
}
)*
;
```

terminoN2 returns [Resultado value]:

```
t1=terminoN3
{
    Object valort1 = null;
    if($t1.value.value!=null ){
```

```

                                valort1 = $t1.value.value; $value = new
Resultado(valort1, "");
                                }
                                $value = new Resultado(valort1, "");
                                }
(MUL t2=terminoN3
    { Object valor = null;
      if($t2.value.value!=null && $value.value!=null){
          valor = (Double)$value.value* (Double)
$t2.value.value;$value = new Resultado(valor, "");
      }
      $value = new Resultado(valor, "");
    }
)*
| t1=terminoN3
    {
        Object valort1 = null;
        if($t1.value.value!=null ){
            valort1 = $t1.value.value;
        }
        $value = new Resultado(valort1, "");
    }
(DIV t2=terminoN3
    { Object valor = null;
      if($t2.value.value!=null && $value.value!=null){
          valor = (Double)$value.value / (Double)
$t2.value.value; $value = new Resultado(valor, "");
      }
      $value = new Resultado(valor, "");
    }
)*
;

```

terminoN3 returns [Resultado value]:

```

n1=terminoN4
{
    Object valorn1 = null;
    if($n1.value.value!=null){
        valorn1 = $n1.value.value; $value = new
Resultado(valorn1, "");
    }
    $value = new Resultado(valorn1, "");
}
(POT n2=terminoN4
    { Object valor = null;

```



```

        if($n2.value.value!=null && $value.value!=null ){
            valor= Math.pow((Double)$value.value, (Double)
$n2.value.value);    $value = new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");
    }
    )*
;

```

terminoN4 returns [Resultado value]:

```

    n1=numero
    {
        Object valorn1 = null;
        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value;    $value = new
Resultado(valorn1, "");
        }
        $value = new Resultado(valorn1, "");

    }
    (MAYOR_QUE n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = (Double)$value.value > (Double)
$n2.value.value; $value = new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");
    }
    )*
    | n1=numero
    {
        Object valorn1 = null;
        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value;    $value = new
Resultado(valorn1, "");
        }
        $value = new Resultado(valorn1, "");
    }
    (MENOR_QUE n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = (Double)$value.value < (Double)
$n2.value.value;

            $value = new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");
    }

```

```

    }
)*
| n1=numero
    {
        Object valorn1 = null;
        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value;
            $value = new Resultado(valorn1, "");
        }
        $value = new Resultado(valorn1, "");

    }
(MAYOR_QUE ASIGNACION n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = (Double)$value.value >= (Double)
$n2.value.value;

            $value = new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");

    }
)*
| n1=numero
    {
        Object valorn1 = null;
        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value;
            $value = new Resultado(valorn1, "");
        }
        $value = new Resultado(valorn1, "");

    }
(MENOR_QUE n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = (Double)$value.value <= (Double)
$n2.value.value;

            $value = new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");

    }
)*
| n1=numero
    {
        Object valorn1 = null;

```

```

        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value; $value = new
Resultado(valorn1, "");
        }

        $value = new Resultado(valorn1, "");
    }
(DIFERENTE n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = $value.value != $n2.value.value; $value =
new Resultado(valor, "");
        }
        $value = new Resultado(valor, "");
    }
)*
|n1=numero
    {
        Object valorn1 = null;
        if($n1.value.value!=null){
            valorn1 = (Double)$n1.value.value; $value = new
Resultado(valorn1, "");
        }
        $value = new Resultado(valorn1, "");
    }
(ASIGNACIO ASIGNACIO n2=numero
    { Object valor = null;
        if($n2.value.value!=null && $value.value!=null ){
            valor = $value.value == $n2.value.value;
            $value = new Resultado(valor, "");
        }

        $value = new Resultado(valor, "");
    }
)*
;

numero returns [Resultado value]
: ENTERO
    {
        Object valor = Double.parseDouble($ENTERO.text);
        $value = new Resultado(valor, "");
    }
| DECIMAL
    {
        Object valor = Double.parseDouble($DECIMAL.text);
        $value = new Resultado(valor, "");
    }

```

```

    }
| PAR1 expresion PAR2
    {
        Object valor = $expresion.value;
        $value = new Resultado(valor, "");
    }
| IDENTIFICADOR
    {
        Map <String,Object> tablaEnTurno=null;

        if(tablaSimbolosLocales!=null)
        {
            tablaEnTurno=tablaSimbolosLocales;
            if (!tablaEnTurno.containsKey($IDENTIFICADOR.text))
{
                tablaEnTurno=tablaSimbolos;
            }
        }else
        {
            tablaEnTurno=tablaSimbolos;
        }
        boolean ban1 = false;boolean ban2 = false;    Object valor = null;

        if (!tablaEnTurno.containsKey($IDENTIFICADOR.text)) {
            valor = null;    ban1=true;
        } else {

            if(tablaEnTurno.get($IDENTIFICADOR.text) instanceof Double)
{
                valor = tablaEnTurno.get($IDENTIFICADOR.text);
            } else{
                ban2=true; valor= null;
            }
        }

        $value = new Resultado(valor, "ss");
        $value.setBandera1(ban1);
        $value.setBandera2(ban2);

    }
| MENOS numero
    {
        Object valor = -((Number) $numero.value.value).doubleValue();
        $value = new Resultado(valor, "");
    }
| (CADENA|CARACTER|BOOLEAN)
    {

```

```

        Object valor= null;
        $value = new Resultado(valor, "");
    }
;

regreso returns[boolean exito=false] locals[boolean errorInterno=false, boolean
errorBloque=false]:

    REGRESO PAR1 bloque_main
    {
        if(!$bloque_main.exito){
            $errorInterno=true;
            $errorBloque=true;
        }

    }PAR2 FINI_NSTRUCCION

    {
        if(!$errorInterno){
            $exito=true;
        }
    }
;

bloque_main returns[boolean exito=false]:

    '<' MAIN '>' bloque_instrucciones_html '</' MAIN '>'
    {
        $exito=true;
    }
;

bloque_instrucciones_html:

    instruccion_html bloque_instrucciones_html_prima
;

bloque_instrucciones_html_prima:

    instruccion_html bloque_instrucciones_html_prima
    |
;
instruccion_html:

    etiquetaEncabezado {an($etiquetaEncabezado.ctx,1);}
    | etiquetaParrafo {an($etiquetaParrafo.ctx,2);}
    | etiquetaBoton {an($etiquetaBoton.ctx,3);}

```

```

| etiquetaInput {an($etiquetaInput.ctx,4);}
;

etiquetaEncabezado returns[boolean exito=false] locals[ boolean
errorId=false,boolean errorTipo=false,boolean errorInterno=false, String ids=""]:

    '<' h1=HEADERS '>' (CADENA|variableTipeScript
        {
            if(!$variableTipeScript.exito){
                $errorId=true; $errorInterno=true;
$ids+=$variableTipeScript.text+" ";
            }
        }
    )* '</' h2=HEADERS '>'
        {
            if(!$h1.text.equals($h2.text))
            {
                $errorTipo=true;
$errorInterno=true;
            }
            if(!$errorInterno){
                $exito=true;
            }
        }
;

variableTipeScript returns[boolean exito=false] locals[ boolean
errorId=false,boolean errorInterno=false]:

    CORCH1 IDENTIFICADOR
    {
        if (!tablaSimbolos.containsKey($IDENTIFICADOR.text))
        {
            $errorId=true;$errorInterno=true;
        }
    } CORCH2
    {
        if(!$errorInterno){
            $exito=true;
        }
    }
;

etiquetaParrafo returns[boolean exito=false] locals[ boolean errorId=false,boolean
errorTipo=true,boolean errorInterno=false, String ids=""]:

    '<p1=PARRAFO>' (CADENA|variableTipeScript

```

```

        {
            if(!$variableTipeScript.exito){
                $errorId=true; $errorInterno=true;
                $sids+=$variableTipeScript.text+" ";
            }
        }
    ) * '</'p2=PARRAFO'>'
        {
            if(!$errorInterno){
                $errorTipo=false;
                $exito=true;
            }
        }
    }

```

```
;
```

etiquetaBoton returns[boolean exito=false] locals[boolean errorId=false,boolean errorTipo=true,boolean errorOnClock=true,boolean errorInterno=false, String ids=""]:

```

    '<'BOTON  onclick?
        {
            if(_localctx.onclick!=null){
                if(!$onclick.exito){
                    $errorOnClock=true;
                }
            }
        }

    '>' (CADENA|variableTipeScript
        {
            if(!$variableTipeScript.exito){
                $errorId=true; $errorInterno=true;
                $sids+=$variableTipeScript.text+" ";
            }
        }
    ) * '</'BOTON'>'
        {
            if(!$errorInterno){
                $errorTipo=false;
                $exito=true;
            }
        }
    }

```

```
;
```

```
onclick returns[boolean exito=false] locals[boolean errorInterno=false,boolean  
errorLlamadaFuncion=false]:
```

```
    ONCLICK ASIGNACIO CORCH1 llamada_funcion  
    {  
        if(!$llamada_funcion.exito){ $errorInterno=true; $errorLlamadaFuncion=  
true;}  
    } CORCH2  
    {  
        if(!$errorInterno){  
            $exito=true;  
        }  
    }
```

```
;
```

```
etiquetaInput returns[boolean exito=false] locals[ boolean errorId=false,boolean  
errorTipo=true,boolean errorInterno=false, String ids=""]:
```

```
    '<'INPUT ('value' ASIGNACIO CORCH1 variableTipeScript CORCH2  
        {  
            if(!$variableTipeScript.exito){  
                $errorId=true; $errorInterno=true;  
$sids+=$variableTipeScript.text+" ";  
            }  
        }  
    )? '>'  
    {  
        if(!$errorInterno){  
            $errorTipo=false;  
$exito=true;  
        }  
    }
```

```
;
```