



5-9-2021

GRAMATICA DE LOS LENGUAJES

FIUSAC Copy Analyzer – Proyecto 1

Eduardo René Agustin Mendoza 201801627
ORGANIZACIÓN DE LENGUAJES Y COMPILADORES 1 - N

GRAMATICA DEL LENGUAJE FCA

Terminales = { Compare, DefinirGlobales, GenerarReporteEstadistico, string, double, GraficaBarras, Titulo, Ejex, Valores, TituloX, TituloY, GraficaPie, GraficaLineas, Archivo, ;, :, ,, {, }, (,), [,], =, IDENTIFICADOR, E_CADENA, DECI }

Donde:

ENTERO: [0-9]⁺

LETRA: [a-zA-Z]

IDENTIFICADOR, es una expresión regular la cual es: {LETRA}{LETRA}|"_{ENTERO})*

DECI: [0-9]⁺+".[0-9]⁺?

E_CADENA = "[^"]^{*}"

BPRUTA = "[^"]^{*}"

No terminales = {ini, contenidoGeneral, graphbarra, graphpie, graphline, titulos, tituloX, tituloY, archivo, comentario, graficas, cuerpo, contenidoX, contenidoB, ejesX, valoresGeneral, comparar, ejepie, contenidoPie, valoresPie, contenidoP}

No terminal inicial: ini

Producciones:

Ini ::= comentario GenerarReporteEstadistico { comparar contenidoGeneral }
 | GenerarReporteEstadistico { comentario comparar contenidoGeneral }
 | GenerarReporteEstadistico { comparar contenidoGeneral comentario }
 | GenerarReporteEstadistico { comparar contenidoGeneral } comentario

ContenidoGeneral ::= DefinirGlobales { cuerpo }
 | DefinirGlobales { comentario cuerpo }
 | DefinirGlobales { cuerpo comentario }
 | DefinirGlobales { cuerpo } graficas

Cuerpo ::= cuerpo string IDENTIFICADOR = E_CADENA ;
 | cuerpo double IDENTIFICADOR = DECI ;
 | string IDENTIFICADOR = E_CADENA ;
 | double IDENTIFICADOR = DECI ;

Graficas ::= Graficas comparar GraficaBarras { graphbarra }
 | Graficas comparar GraficaPie { graphpie }
 | Graficas comparar GraficaLineas { graphline }
 | Graficas GraficaBarras { graphbarra }
 | Graficas GraficaPie { graphpie }
 | Graficas GraficaLineas { graphline }
 | GraficaBarras { graphbarra }

```
|GraficaPie{ graphpie }  
|GraficaLineas{ graphline }  
|comparar GraficaBarras{ graphbarra }  
|comparar GraficaPie{ graphpie }  
|comparar GraficaLineas{ graphline }
```

```
Graphbarra::=graphbarra titulos  
|graphbarra ejesx  
|graphbarra valoresgeneral  
|graphbarra titulox  
|graphbarra tituloy  
|titulos  
|ejesx  
|valoresgeneral  
|titulox  
|tituloy
```

```
graphpie::=graphpie titulos  
|graphpie ejepie  
|graphpie valorespie  
|titulos  
|ejepie  
|valorespie
```

```
graphline::=graphline titulos  
|graphline Archivo  
|titulos  
|Archivo
```

```
titulos::= Titulo : IDENTIFICADOR ;  
|Titulo : E_CADENA ;
```

```
ejesx::= Ejex: [ contenidox ]
```

```
ejepie::= Ejex: [ contenidopie ]
```

```
contenidox::=  
contenidox E_CADENA  
|contenidox IDENTIFICADOR  
|contenidox ,  
|E_CADENA
```

|IDENTIFICADOR

contenidopie::=contenidopie E_CADENA
|contenidopie IDENTIFICADOR
|contenidopie ,
|E_CADENA
|IDENTIFICADOR

valoresgeneral::= Valores : [contenidob] ;

valorespie::= Valores : [contenidopie] ;

contenidob::=contenidob DECI
|contenidob IDENTIFICADOR
|contenidob ,
|contenidob E_CADENA
|DECI
|IDENTIFICADOR
|E_CADENA

contenidop::=contenidop DECI
|contenidop IDENTIFICADOR
|contenidop ,
|contenidop E_CADENA
|DECI
|IDENTIFICADOR
|E_CADENA

titulox::= TituloX: IDENTIFICADOR ;
| TituloX: E_CADENA ;

tituloy::= TituloY: IDENTIFICADOR ;
| TituloY : E_CADENA ;

Archivo::= Archivo : IDENTIFICADOR ;
| Archivo : E_CADENA ;

compara::= Compare (E_CADENA ,E_CADENA) ;

GRAMATICA DEL LENGUAJE JAVASCRIPT

Terminales = { class , var , let , const , if , else , for , while , do , switch , case , console.log , requiere , break , ; , : , . , , , (,) , { , } , default , + , - , * , ** , / , > , < , % , && , || , = , == , != , >= , <= , ! , IDENTIFICADOR , BPE_CADENA , DECI , false , true , ++ , -- , LINECOMMENT , MULTILINECOMMENT }

Donde=

LINECOMMENT = (//.*\n)|(//.*\r\n)|(//.*\r)

MULTILINECOMMENT = "<!"!""*([^\>]|[\^!]">"!""[^\>])*""!""!>"

ENTERO=[0-9]+

DECI=[0-9]+("." [0-9])?

LETRA = [a-zA-Z]

IDENTIFICADOR = {LETRA}{LETRA}|_"|{ENTERO})*

E_CADENA = \"[^\"]*

No Terminal = { ini , procesos , parametros , variable , impresión , opmate , concatenar , condición , forvariable , condicionfor , foraumento , casos , signos , comentario , siguiente }

No terminal Inicial = ini

Producciones:

ini::=ini comentario class IDENTIFICADOR () { procesos }

|ini class IDENTIFICADOR { comentario procesos }

|ini class IDENTIFICADOR { procesos comentario }

|ini class IDENTIFICADOR { procesos }

|ini IDENTIFICADOR () { procesos }

|ini IDENTIFICADOR (parametros) { procesos }

|ini var IDENTIFICADOR = variable

|ini let IDENTIFICADOR = variable

|ini const IDENTIFICADOR = variable

|ini var IDENTIFICADOR

|ini let IDENTIFICADOR

|ini const IDENTIFICADOR

|ini var IDENTIFICADOR

|ini let IDENTIFICADOR

|ini const IDENTIFICADOR

|class IDENTIFICADOR () { procesos }

|IDENTIFICADOR () { procesos }

|IDENTIFICADOR () ;

|IDENTIFICADOR (parametros) ;

|IDENTIFICADOR ()

|IDENTIFICADOR (parametros)

|IDENTIFICADOR (parametros) { procesos }

|var IDENTIFICADOR = variable

|let IDENTIFICADOR = variable
|const IDENTIFICADOR = variable
|var IDENTIFICADOR
|let IDENTIFICADOR
|const IDENTIFICADOR
|var IDENTIFICADOR ;
|let IDENTIFICADOR ;
|const IDENTIFICADOR ;

comentario::=comentario LINECOMMENT
|comentario MULTILINECOMMENT
| LINECOMMENT
|MULTILINECOMMENT

parametros::=IDENTIFICADOR
|parametros , IDENTIFICADOR
|parametros signos DECI
|parametros signos (variable)

variable::=E_CADENA ;
|E_CADEN
|DECI ;
|DECI
|DECI signos DECI ;
|DECI signos DECI
|true ;
|true
|false ;
|false
|(opmate) opmate ;
|(opmate) opmate
|(opmate) - DECI ;
|(opmate) - DECI
|(opmate) + DECI ;
|(opmate) + DECI
|(opmate) * DECI ;
|(opmate) * DECI
|(opmate) / DECI ;
|(opmate) / DECI
|(opmate) ** DECI ;
|(opmate) ** DECI
|(opmate) % DECI ;
|(opmate) % DECI

|(opmate) - opmate ;
 |(opmate) - opmate
 |(opmate) + opmate ;
 |(opmate) + opmate
 |(opmate) * opmate ;
 |(opmate) * opmate
 |(opmate) / opmate ;
 |(opmate) / opmate
 |(opmate) ** opmate ;
 |(opmate) ** opmate
 |(opmate) % opmate ;
 |(opmate) % opmate
 |(opmate) ;
 |(opmate)
 |requiere (E_CADENA) ;
 |requiere (E_CADENA)
 |signos DECI signos IDENTIFICADOR signos DECI
 |signos DECI signos IDENTIFICADOR signos DECI ;
 |signos DECI signos IDENTIFICADOR signos DECI signos variable
 |DECI signos IDENTIFICADOR signos DECI
 |DECI signos IDENTIFICADOR signos DECI ;
 |DECI signos IDENTIFICADOR signos DECI signos variable
 |signos DECI signos IDENTIFICADOR signos IDENTIFICADOR
 |signos DECI signos IDENTIFICADOR signos IDENTIFICADOR ;
 |signos DECI signos IDENTIFICADOR signos IDENTIFICADOR signos variable
 |DECI signos IDENTIFICADOR signos IDENTIFICADOR
 |DECI signos IDENTIFICADOR signos IDENTIFICADOR ;
 |DECI signos IDENTIFICADOR signos IDENTIFICADOR signos variable
 |signos IDENTIFICADOR signos DECI signos DECI
 |signos IDENTIFICADOR signos DECI signos DECI ;
 |signos IDENTIFICADOR signos DECI signos DECI signos variable
 |IDENTIFICADOR signos DECI signos DECI
 |IDENTIFICADOR signos DECI signos DECI ;
 |IDENTIFICADOR signos DECI signos DECI signos variable
 |signos IDENTIFICADOR signos IDENTIFICADOR signos DECI
 |signos IDENTIFICADOR signos IDENTIFICADOR signos DECI ;
 |signos IDENTIFICADOR signos IDENTIFICADOR signos DECI signos variable
 |IDENTIFICADOR signos IDENTIFICADOR signos DECI
 |IDENTIFICADOR signos IDENTIFICADOR signos DECI ;
 |IDENTIFICADOR signos IDENTIFICADOR signos DECI signos variable
 |signos IDENTIFICADOR signos DECI signos IDENTIFICADOR
 |signos IDENTIFICADOR signos DECI signos IDENTIFICADOR ;
 |signos IDENTIFICADOR signos DECI signos IDENTIFICADOR signos variable

|IDENTIFICADOR signos DECI signos IDENTIFICADOR
 |IDENTIFICADOR signos DECI signos IDENTIFICADOR ;
 |IDENTIFICADOR signos DECI signos IDENTIFICADOR signos variable
 |signos DECI signos DECI signos IDENTIFICADOR
 |signos DECI signos DECI signos IDENTIFICADOR ;
 |signos DECI signos DECI signos IDENTIFICADOR signos variable
 |DECI signos DECI signos IDENTIFICADOR
 |DECI signos DECI signos IDENTIFICADOR ;
 |DECI signos DECI signos IDENTIFICADOR signos variable
 |signos IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR
 |signos IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR ;
 |signos IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR signos variable
 |IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR
 |IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR ;
 |IDENTIFICADOR signos IDENTIFICADOR signos IDENTIFICADOR signos variable

signos::=+

|-
 |*
 |/
 |**
 |%

procesos::=procesos IDENTIFICADOR () { procesos }
 |procesos IDENTIFICADOR (parametros) { procesos }
 |procesos var IDENTIFICADOR = variable
 |procesos let IDENTIFICADOR = variable
 |procesos const IDENTIFICADOR = variable
 |procesos var IDENTIFICADOR
 |procesos let IDENTIFICADOR
 |procesos const IDENTIFICADOR
 |procesos var IDENTIFICADOR ;
 |procesos let IDENTIFICADOR ;
 |procesos const IDENTIFICADOR ;
 |procesos if (condicion) { procesos } siguiente
 |procesos if (condicion) { procesos }
 |procesos for (forvariable condicionfor foraumento) { procesos }
 |procesos while (condicion) { procesos }
 |procesos do { procesos } while (condicion)
 |procesos do { procesos } while (condicion) ;
 |procesos switch (IDENTIFICADOR) { casos }
 |procesos console.log (impresion) ;
 |procesos console.log (impresion)


```

|procesos IDENTIFICADOR = IDENTIFICADOR
|procesos IDENTIFICADOR = concatenar
|procesos IDENTIFICADOR . IDENTIFICADOR ( parametros )
|procesos IDENTIFICADOR . IDENTIFICADOR ( )
|procesos IDENTIFICADOR . IDENTIFICADOR ( parametros ) ;
|procesos IDENTIFICADOR . IDENTIFICADOR ( ) ;
|procesos IDENTIFICADOR ( parametros ) ;
|procesos IDENTIFICADOR ( parametros )
|IDENTIFICADOR ( ) { procesos }
|IDENTIFICADOR ( parametros ) { procesos }
|var IDENTIFICADOR = variable
|let IDENTIFICADOR = variable
|const IDENTIFICADOR = variable
|var IDENTIFICADOR
|let IDENTIFICADOR
|const IDENTIFICADOR
|var IDENTIFICADOR ;
|let IDENTIFICADOR ;
|const IDENTIFICADOR ;
| if ( condicion ) { procesos } siguiente
| if ( condicion ) { procesos }
|for ( forvariable condicionfor foraumento ) { procesos }
|while ( condicion ) { procesos }
|do { procesos } while ( condicion )
|do { procesos } while ( condicion ) ;
|switch ( IDENTIFICADOR ) { casos }
|console.log ( impresion ) ;
|console.log ( impresion )
|IDENTIFICADOR = IDENTIFICADOR
|IDENTIFICADOR = concatenar
|IDENTIFICADOR . IDENTIFICADOR ( parametros )
|IDENTIFICADOR . IDENTIFICADOR ( )
|IDENTIFICADOR . IDENTIFICADOR ( parametros ) ;
|IDENTIFICADOR . IDENTIFICADOR ( ) ;
|IDENTIFICADOR ( parametros ) ;
|IDENTIFICADOR ( parametros )

```

```

siguiente::=siguiente else { procesos }
|siguiente else if ( condicion ) { procesos }
|else { procesos }
|else if ( condicion ) { procesos }

```

```

concatenar::=concatenar + IDENTIFICADOR

```

```
|concatenar + DECI
|concatenar + E_CADENA
|concatenar
|IDENTIFICADOR + DECI
|IDENTIFICADOR + E_CADENA
|IDENTIFICADOR
|DECI + IDENTIFICADOR
|E_CADENA + IDENTIFICADOR + IDENTIFICADOR
|IDENTIFICADOR + IDENTIFICADOR
```

```
forvariable::=var IDENTIFICADOR = variable
|let IDENTIFICADOR = variable
|IDENTIFICADOR = IDENTIFICADOR ;
|IDENTIFICADOR opmate ;
|IDENTIFICADOR ;
|IDENTIFICADOR = variable
```

```
condicionfor::=IDENTIFICADOR > IDENTIFICADOR ;
|IDENTIFICADOR < IDENTIFICADOR ;
|IDENTIFICADOR >= IDENTIFICADOR ;
|IDENTIFICADOR <= IDENTIFICADOR ;
|IDENTIFICADOR > DECI ;
|IDENTIFICADOR < DECI ;
|IDENTIFICADOR >= DECI ;
|IDENTIFICADOR <= DECI ;
|IDENTIFICADOR > opmate ;
|IDENTIFICADOR < opmate ;
|IDENTIFICADOR >= opmate ;
|IDENTIFICADOR <= opmate ;
```

```
foraumentto::=IDENTIFICADOR ++
|IDENTIFICADOR --
|IDENTIFICADOR + = DECI
```

```
condicion::=condicion E_CADENA == E_CADENA
|condicion DECI == DECI
|condicion DECI > DECI
|condicion DECI < DECI
|condicion DECI >= DECI
|condicion DECI <= DECI
|condicion opmate == DECI
|condicion opmate > DECI
|condicion opmate < DECI
```

|condicion opmate >= DECI
|condicion opmate <= DECI
|condicion opmate == IDENTIFICADOR
|condicion opmate > IDENTIFICADOR
|condicion opmate < IDENTIFICADOR
|condicion opmate >= IDENTIFICADOR
|condicion opmate <= IDENTIFICADOR
|condicion IDENTIFICADOR == DECI
|condicion IDENTIFICADOR > DECI signos DECI
|condicion IDENTIFICADOR > IDENTIFICADOR signos DECI
|condicion IDENTIFICADOR > DECI signos IDENTIFICADOR
|condicion IDENTIFICADOR > (opmate)
|condicion IDENTIFICADOR == true
|condicion IDENTIFICADOR == false
|condicion IDENTIFICADOR == E_CADENA
|condicion IDENTIFICADOR > DECI
|condicion IDENTIFICADOR < DECI
|condicion IDENTIFICADOR < DECI signos DECI
|condicion IDENTIFICADOR < IDENTIFICADOR signos DECI
|condicion IDENTIFICADOR < DECI signos IDENTIFICADOR
|condicion IDENTIFICADOR < (opmate)
|condicion IDENTIFICADOR >= DECI
|condicion IDENTIFICADOR >= DECI signos DECI
|condicion IDENTIFICADOR >= IDENTIFICADOR signos DECI
|condicion IDENTIFICADOR >= DECI signos IDENTIFICADOR
|condicion IDENTIFICADOR >= (opmate)
|condicion IDENTIFICADOR <= DECI
|condicion IDENTIFICADOR <= DECI signos DECI
|condicion IDENTIFICADOR <= IDENTIFICADOR signos DECI
|condicion IDENTIFICADOR <= DECI signos IDENTIFICADOR
|condicion IDENTIFICADOR <= (opmate)
|condicion IDENTIFICADOR > IDENTIFICADOR
|condicion IDENTIFICADOR < IDENTIFICADOR
|condicion IDENTIFICADOR >= IDENTIFICADOR
|condicion IDENTIFICADOR <= IDENTIFICADOR
|condicion IDENTIFICADOR == IDENTIFICADOR
|condicion DECI ! DECI
|condicion E_CADENA ! E_CADENA
|condicion IDENTIFICADOR ! DECI
|condicion IDENTIFICADOR ! true
|condicion IDENTIFICADOR ! false
|condicion IDENTIFICADOR ! E_CADENA
|condicion IDENTIFICADOR ! IDENTIFICADOR

|condicion != IDENTIFICADOR
|condicion (condicion)
|condicion IDENTIFICADOR
|condicion ||
|condicion &&
|E_CADENA == E_CADENA
|DECI == DECI
|DECI > DECI
|DECI < DECI
|DECI >= DECI
|DECI <= DECI
|opmate == DECI
|opmate > DECI
|opmate < DECI
|opmate >= DECI
|opmate <= DECI
|opmate ! DECI
|opmate == IDENTIFICADOR
|opmate > IDENTIFICADOR
|opmate < IDENTIFICADOR
|opmate >= IDENTIFICADOR
|opmate <= IDENTIFICADOR
|opmate ! IDENTIFICADOR
|IDENTIFICADOR
|IDENTIFICADOR == DECI
|IDENTIFICADOR == DECI signos DECI
|IDENTIFICADOR == IDENTIFICADOR signos DECI
|IDENTIFICADOR == DECI signos IDENTIFICADOR
|IDENTIFICADOR == (opmate)
|IDENTIFICADOR == true
|IDENTIFICADOR == false
|IDENTIFICADOR == E_CADENA
|IDENTIFICADOR > DECI
|IDENTIFICADOR > DECI signos DECI
|IDENTIFICADOR > IDENTIFICADOR signos DECI
|IDENTIFICADOR > DECI signos IDENTIFICADOR
|IDENTIFICADOR > (opmate)
|IDENTIFICADOR < DECI
|IDENTIFICADOR < DECI signos DECI
|IDENTIFICADOR < IDENTIFICADOR signos DECI
|IDENTIFICADOR < DECI signos IDENTIFICADOR
|IDENTIFICADOR < (opmate)
|IDENTIFICADOR >= DECI

```

|IDENTIFICADOR >= DECI signos DECI
|IDENTIFICADOR >= IDENTIFICADOR signos DECI
|IDENTIFICADOR >= DECI signos IDENTIFICADOR
|IDENTIFICADOR >= ( opmate )
|IDENTIFICADOR <= DECI
|IDENTIFICADOR <= DECI signos DECI
|IDENTIFICADOR <= IDENTIFICADOR signos DECI
|IDENTIFICADOR <= DECI signos IDENTIFICADOR
|IDENTIFICADOR <= ( opmate )
|IDENTIFICADOR > IDENTIFICADOR
|IDENTIFICADOR < IDENTIFICADOR
|IDENTIFICADOR >= IDENTIFICADOR
|IDENTIFICADOR <= IDENTIFICADOR
|IDENTIFICADOR == IDENTIFICADOR
|DECI ! DECI
|E_CADENA ! E_CADENA
|IDENTIFICADOR ! DECI
|IDENTIFICADOR ! true
|IDENTIFICADOR ! false
|IDENTIFICADOR ! E_CADENA
|IDENTIFICADOR ! IDENTIFICADOR
|!= IDENTIFICADOR
|( condicion )

```

```

casos::=casos case DECI : procesos break ;
|casos default : procesos break ;
|casos default : procesos
|casos case E_CADENA : procesos break ;
|case DECI : procesos break ;
|default : procesos break ;
|default : procesos
|case E_CADENA : procesos break ;

```

```

impresion::=impresion + DECI
|impresion + E_CADENA
|impresion + != IDENTIFICADOR
|impresion + IDENTIFICADOR
|impresion + ( opmate )
|E_CADENA
|IDENTIFICADOR
|DECI
|- DECI
|- ( opmate )

```

| - (DECI)
| - (IDENTIFICADOR)
| (opmate)
| IDENTIFICADOR ++
| IDENTIFICADOR --
| != IDENTIFICADOR

opmate::=opmate (opmate)
| opmate DECI ** DECI
| opmate DECI / DECI
| opmate DECI % DECI
| opmate DECI * DECI
| opmate DECI + DECI
| opmate DECI - DECI
| opmate DECI ** IDENTIFICADOR
| opmate DECI / IDENTIFICADOR
| opmate DECI % IDENTIFICADOR
| opmate DECI * IDENTIFICADOR
| opmate DECI + IDENTIFICADOR
| opmate DECI - IDENTIFICADOR
| (opmate)
| DECI ** DECI
| DECI / DECI
| DECI % DECI
| DECI * DECI
| DECI + DECI
| DECI - DECI
| DECI ** IDENTIFICADOR
| DECI / IDENTIFICADOR
| DECI % IDENTIFICADOR
| DECI * IDENTIFICADOR
| DECI + IDENTIFICADOR
| DECI – IDENTIFICADOR