Gramática analizador léxico Servidor

Gramática: import

G = (N, T, P, S)

N = {i, m, p, o, r, t}

T = {s0}

S = {s0}

P =

s0 -> import

Gramática: class

G = (N, T, P, S)

N = {c, l, a, s}

T = {s0}

S = {s0}

P =

s0 -> class

Gramática: int

G = (N, T, P, S)

N = {int}

T = {s0}

S = {s0}

P =

s0 -> int

Gramática: boolean

G = (N, T, P, S)

N = {b, o, l, e, a, n}

T = {s0}

S = {s0}

P =

s0 -> boolean

Gramática: String

G = (N, T, P, S)

N = {S, t, r, i, n, g}

T = {s0}

S = {s0}

P =

s0 -> String

Gramática: char

G = (N, T, P, S)

N = {c, h, a, r}

T = {s0}

S = {s0}

P =

s0 -> char

Gramática: double

G = (N, T, P, S)

N = {d, o, u, b, l, e}

T = {s0}

S = {s0}

P =

s0 -> double

Gramática: double

G = (N, T, P, S)

N = {d, o, u, b, l, e}

T = {s0}

S = {s0}

P =

s0 -> double

Gramática: Object

G = (N, T, P, S)

N = {O, b, j, e, c, t}

T = {s0}

S = {s0}

P =

s0 -> Object

Gramática: if

G = (N, T, P, S)

N = {i, f}

T = {s0}

S = {s0}

P =

s0 -> if

Gramática: else

G = (N, T, P, S)

N = {e, l, s}

T = {s0}

S = {s0}

P =

s0 -> else

Gramática: for

G = (N, T, P, S)

N = {f, o, r}

T = {s0}

S = {s0}

P =

s0 -> for

Gramática: while

G = (N, T, P, S)

N = {w, h, i, l, e}

T = {s0}

S = {s0}

P =

s0 -> while

Gramática: do

G = (N, T, P, S)

N = {d, o}

T = {s0}

S = {s0}

P =

s0 -> do

Gramática: switch

G = (N, T, P, S)

N = {s, w, i, t, c}

T = {s0}

S = {s0}

P =

s0 -> switch

Gramática: case

G = (N, T, P, S)

N = {c, a, s, e}

T = {s0}

S = {s0}

P =

s0 -> case

Gramática: public

G = (N, T, P, S)

N = {p, u, b, l, i, c}

T = {s0}

S = {s0}

P =

s0 -> public

Gramática: protected

G = (N, T, P, S)

N = {p, r, o, t, e, c, d}

T = {s0}

S = {s0}

P =

s0 -> protected

Gramática: private

G = (N, T, P, S)

N = {p, r, i, v, a, t, e}

T = {s0}

S = {s0}

P =

s0 -> private

Gramática: final

G = (N, T, P, S)

N = {f, i, n, a, l}

T = {s0}

S = {s0}

P =

s0 -> final

Gramática: break

G = (N, T, P, S)

N = {b, r, e, a, k}

T = {s0}

S = {s0}

P =

s0 -> break

Gramática: return

G = (N, T, P, S)

N = {r, e, t, n}

T = {s0}

S = {s0}

P =

s0 -> return

Gramática: void

G = (N, T, P, S)

N = {v, o, i, d}

T = {s0}

S = {s0}

P =

s0 -> void

Gramática: this

G = (N, T, P, S)

N = {t, h, i, s}

T = {s0}

S = {s0}

P =

s0 -> this

Gramática: true

G = (N, T, P, S)

N = {t, r, u, e}

T = {s0}

S = {s0}

P =

s0 -> true

Gramática: false

G = (N, T, P, S)

N = {f, a, l, s, e}

T = {s0}

S = {s0}

P =

s0 -> false

Gramática: new

G = (N, T, P, S)

N = {n, e, w}

T = {s0}

S = {s0}

P =

s0 -> new

Gramática: default

G = (N, T, P, S)

N = {d, e, f, a, u, l, t}

T = {s0}

S = {s0}

P =

s0 -> default

Gramatica: entero

d = [0 - 9]

G = (N, T, P, S)

N = {d}

T = {s0, s1}

S = {s0}

P =

s0 -> d s1

s0 -> d

s1 -> d s0

Gramatica: decimal

d = [0 - 9]

G = (N, T, P, S)

N = {d}

T = {s0, s1, s2, s3, s4}

S = {s0}

P =

s0 -> d s1

s0 -> d s2

s1 -> d s0

s2 -> . s3

s3 -> d s4

s3 -> d

s4 -> d s3

Gramatida: id

id = [a-zA-Z_]

G = (N, T, P, S)

N = {id}

T = {s0, s1, s2, s3, s4}

S = {s0}

P =

s0 -> id s0

s0 -> id

Gramatica: Cadena

cadena = [a - z, A - Z]

G = (N, T, P, S)

N = {'', cadena, entero, decimal}

T = {s0, s1, s2}

S = {s0}

P =

s0 -> " s1

s1 -> cadena s2

s1 -> entero s2

s1 -> decimal s2

s2 -> s1

s2 -> "

Gramatica: ==

G = (N, T, P, S)

N = {=}

T = {s0}

S = {s0}

P =

s0 -> ==

Gramatica: !=

G = (N, T, P, S)

N = {!, =}

T = {s0}

S = {s0}

P =

s0 -> !=

Gramatica: <=

G = (N, T, P, S)

N = {<, =}

T = {s0}

S = {s0}

P =

s0 -> <=

Gramatica: >=

G = (N, T, P, S)

N = {>, =}

T = {s0}

S = {s0}

P =

s0 -> >=

Gramatica: &&

G = (N, T, P, S)

N = {&}

T = {s0}

S = {s0}

P =

s0 -> &

Gramatica: ||

G = (N, T, P, S)

N = {|}

T = {s0}

S = {s0}

P =

s0 -> ||

Gramatica: ++

G = (N, T, P, S)

N = {+}

T = {s0}

S = {s0}

P =

s0 -> ++

Gramatica: --

G = (N, T, P, S)

N = {-}

T = {s0}

S = {s0}

P =

s0 -> --

Gramatica: +

G = (N, T, P, S)

N = {+}

T = {s0}

S = {s0}

P =

s0 -> +

Gramatica: -

G = (N, T, P, S)

N = {-}

T = {s0}

S = {s0}

P =

s0 -> -

Gramatica: *

G = (N, T, P, S)

N = {*}

T = {s0}

S = {s0}

P =

s0 -> *

Gramatica: /

G = (N, T, P, S)

N = {/}

T = {s0}

S = {s0}

P =

s0 -> /

Gramatica: (

G = (N, T, P, S)

N = {(}

T = {s0}

S = {s0}

P =

s0 -> (

Gramatica:)

G = (N, T, P, S)

N = {)}

T = {s0}

S = {s0}

P =

s0 ->)

Gramatica: :

G = (N, T, P, S)

N = {:}

T = {s0}

S = {s0}

P =

s0 -> :

Gramatica: ;

G = (N, T, P, S)

N = {;}

T = {s0}

S = {s0}

P =

s0 -> ;

Gramatica: {

G = (N, T, P, S)

N = {"{"}

T = {s0}

S = {s0}

P =

s0 -> {

Gramatica: }

G = (N, T, P, S)

N = {"}"}

T = {s0}

S = {s0}

P =

s0 -> }

Gramatica: [

G = (N, T, P, S)

N = {"["}

T = {s0}

S = {s0}

P =

s0 -> [

Gramatica:]

G = (N, T, P, S)

N = {"]"}

T = {s0}

S = {s0}

P =

s0 ->]

Gramatica: =

G = (N, T, P, S)

N = {"="}

T = {s0}

S = {s0}

P =

s0 -> =

Gramatica: ,

G = (N, T, P, S)

N = {","}

T = {s0}

S = {s0}

P =

s0 -> ,