



FACULTAD DE INGENIERIA MECANICA Y ELECTRICA



COMPILADOR



**INGENIERIA EN SISTEMAS COMPUTACIONALES
7MO SEMESTRE
SISTEMAS DE PROGRAMACION 1
PROF.ING.JOSE LUIS ZARAZUA GAYTAN
ALUMNAS:
NANCY VAZQUEZ VALLADARES
CELINA ELIZABETH BALLESTEROS OYUELA**

POLITICAS PYTHON

Tokens

- Se denominan *tokens* a los componentes léxicos del lenguaje de programación. Los tipos de *token* son: identificadores, palabras clave, operadores, delimitadores y literales. Para separar los *tokens* se utilizan espacios.

Identificadores

Un *identificador* es el nombre empleado para identificar una variable, una función, una clase, un módulo u otro objeto.

- Un identificador comienza con una letra (de la A la Z o de la a la z)
- **Python NO permite** signos de puntuación como @, \$ y %, excepto el guión bajo (_).
- Se recomienda nombrar las clases comenzando por una letra mayúscula y el resto de los identificadores por una letra minúscula.

```
# Identificadores validos:
radio
RaDio
contAlumnos
Cont_Alumnos
num1

# Identificadores NO validos:
2E2
_Miércoles
Cont-Alumnos
Conjunto Primero
print #palabras reservadas
1_cantidad
```

Recuerda que Python distingue las mayúsculas de las minúsculas, entonces por ejemplo **area**, **Area**, **AREA**, **aRea**, son 4 identificadores distintos. También comentar que, el último ejemplo, **aRea** no es recomendado, dado su poca legibilidad.

Palabras clave

- Python tiene 30 palabras reservadas que no podremos utilizar para ningún tipo de identificador:
- Las palabras reservadas de Python son las siguientes:

False	None	True	and	as	assert	async
await	break	class	continue	def	del	elif
else	except	finally	for	from	global	if
import	in	is	lambda	nonlocal	not	or
pass	raise	return	try	while	with	yield

Operadores

- Python utiliza caracteres no alfanuméricos y combinación de caracteres como operadores.
Se reconocen los siguientes operadores: + - * < <> >= > >= ==
- Operaciones que aparezcan más arriba se evalúan después que las que aparecen más abajo.
- Operaciones de la en la misma fila se evalúan de izquierda a derecha.^[1]

Delimitadores

- Python utiliza los siguientes símbolos y combinación de símbolos como delimitadores en expresiones, listas, diccionarios, etc: () [] { , : . ` = ; += -= *= /=
//= %= &= |= ^= >>= <<= **=
- Los caracteres \$ y ? y todos los caracteres de control nunca podrán ser parte de un programa Python, excepto en comentarios y *strings*.
- La utilización de @ también está limitado en algunas versiones.

Literales

- Un *literal* es un número o un *string* que aparece directamente en un programa. Los siguientes son literales en Python:

Definir dentro del código:

1. Empieza con comillas
2. Puede venir cualquier carácter especial
3. Salida con el mismo numero de comillas del inicio
4. El limite de comillas es de 3 para definir una constante string

Dentro de variables

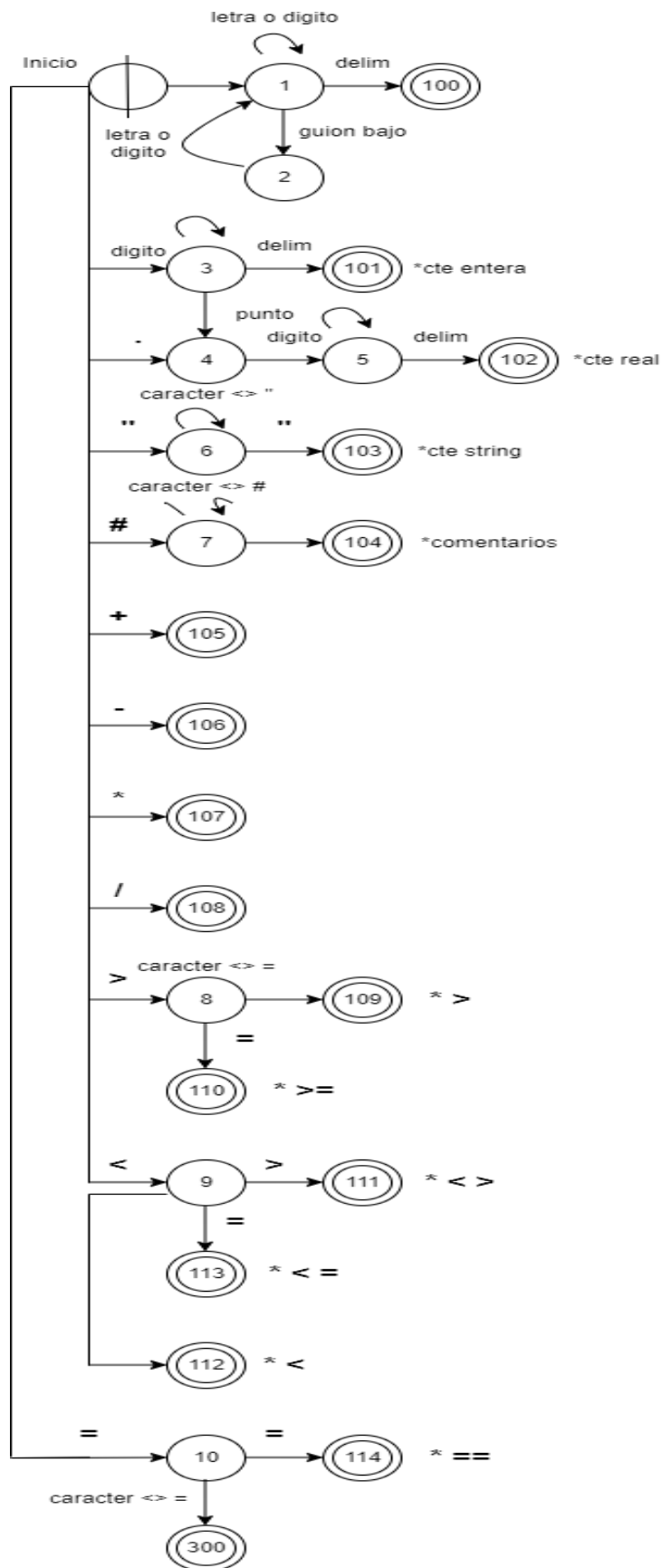
***No contiene reglas del tipo string**

```
42          # Entero
3.14        # Punto flotante
1.0j        # Imaginario
'hola'      # String
"mundo"     # Otro string
"""Buenas
noches"""   # String limitado con tres comillas
```

Constantes enteras y reales

1. Empieza con una constante entera
2. Puede seguir dígitos/ punto
3. No pueden ir mas de 2 puntos
4. **Después del punto debe ir por lo menos 1 dígito**
5. Puede seguir dígitos/ punto
6. Si comienza con dígito el siguiente carácter deberá llevar
7. No puede llevar mas de 1 punto
8. Después del punto el siguiente carácter deberá ser un dígito

Diagrama de estados



Matriz de estados

	0 Letra	1 Digito	2 Espacio	2 _	3 .	4 '	5 #
0	1	3	0	300	4	6	7
1	1	1	100	2	100	100	100
2	1	1	300	300	300	300	300
3	301	3	101	101	4	101	101
4	302	5	302	302	302	302	302
5	303	5	102	102	102	102	102
6	6	6	6	6	6	103	6
7	7	7	7	7	7	7	104
8	109	109	109	109	109	109	109
9	112	112	112	112	112	112	112
10	304	304	304	304	304	304	304

	6 +	7 -	8 *	9 /	10 >	11 <	12 =
0	105	106	107	108	8	9	10
1	100	100	100	100	100	100	100
2	300	300	300	300	300	300	300
3	101	101	101	101	101	101	101
4	302	302	302	302	302	302	302
5	102	102	102	102	102	102	102
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	109	109	109	109	109	109	110
9	112	112	112	112	111	112	113
10	304	304	304	304	304	304	114

Código

```
Public Class Proyecto
    Dim matriz(10, 13) As Integer
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        matriz(0, 0) = 1
        matriz(0, 1) = 3
        matriz(0, 2) = 0
        matriz(0, 3) = 300
        matriz(0, 4) = 4
        matriz(0, 5) = 6
        matriz(0, 6) = 7
        matriz(0, 7) = 105
        matriz(0, 8) = 106
        matriz(0, 9) = 107
        matriz(0, 10) = 108
        matriz(0, 11) = 8
        matriz(0, 12) = 9
        matriz(0, 13) = 10

        matriz(1, 0) = 1
        matriz(1, 1) = 1
        matriz(1, 2) = 100
        matriz(1, 3) = 2
        matriz(1, 4) = 100
        matriz(1, 5) = 100
        matriz(1, 6) = 100
        matriz(1, 7) = 100
        matriz(1, 8) = 100
        matriz(1, 9) = 100
        matriz(1, 10) = 100
        matriz(1, 11) = 100
        matriz(1, 12) = 100
        matriz(1, 13) = 100

        matriz(2, 0) = 1
        matriz(2, 1) = 1
        matriz(2, 2) = 300
        matriz(2, 3) = 300
        matriz(2, 4) = 300
        matriz(2, 5) = 300
        matriz(2, 6) = 300
        matriz(2, 7) = 300
        matriz(2, 8) = 300
        matriz(2, 9) = 300
        matriz(2, 10) = 300
        matriz(2, 11) = 300
        matriz(2, 12) = 300
        matriz(2, 13) = 300

        matriz(3, 0) = 301
        matriz(3, 1) = 3
        matriz(3, 2) = 101
        matriz(3, 3) = 101
        matriz(3, 4) = 4
        matriz(3, 5) = 101
        matriz(3, 6) = 101
        matriz(3, 7) = 101
        matriz(3, 8) = 101
        matriz(3, 9) = 101
        matriz(3, 10) = 101
        matriz(3, 11) = 101
    End Sub
End Class
```

```
matriz(3, 12) = 101
matriz(3, 13) = 101
```

```
matriz(4, 0) = 302
matriz(4, 1) = 5
matriz(4, 2) = 302
matriz(4, 3) = 302
matriz(4, 4) = 302
matriz(4, 5) = 302
matriz(4, 6) = 302
matriz(4, 7) = 302
matriz(4, 8) = 302
matriz(4, 9) = 302
matriz(4, 10) = 302
matriz(4, 11) = 302
matriz(4, 12) = 302
matriz(4, 13) = 302
```

```
matriz(5, 0) = 303
matriz(5, 1) = 5
matriz(5, 2) = 102
matriz(5, 3) = 102
matriz(5, 4) = 102
matriz(5, 5) = 102
matriz(5, 6) = 102
matriz(5, 7) = 102
matriz(5, 8) = 102
matriz(5, 9) = 102
matriz(5, 10) = 102
matriz(5, 11) = 102
matriz(5, 12) = 102
matriz(5, 13) = 102
```

```
matriz(6, 0) = 6
matriz(6, 1) = 6
matriz(6, 2) = 6
matriz(6, 3) = 6
matriz(6, 4) = 6
matriz(6, 5) = 103
matriz(6, 6) = 6
matriz(6, 7) = 6
matriz(6, 8) = 6
matriz(6, 9) = 6
matriz(6, 10) = 6
matriz(6, 11) = 6
matriz(6, 12) = 6
matriz(6, 13) = 6
```

```
matriz(7, 0) = 7
matriz(7, 1) = 7
matriz(7, 2) = 7
matriz(7, 3) = 7
matriz(7, 4) = 7
matriz(7, 5) = 7
matriz(7, 6) = 104
matriz(7, 7) = 7
matriz(7, 8) = 7
matriz(7, 9) = 7
matriz(7, 10) = 7
matriz(7, 11) = 7
matriz(7, 12) = 7
matriz(7, 13) = 7
```



```
matriz(8, 0) = 109
matriz(8, 1) = 109
matriz(8, 2) = 109
matriz(8, 3) = 109
matriz(8, 4) = 109
matriz(8, 5) = 109
matriz(8, 6) = 109
matriz(8, 7) = 109
matriz(8, 8) = 109
matriz(8, 9) = 109
matriz(8, 10) = 109
matriz(8, 11) = 109
matriz(8, 12) = 109
matriz(8, 13) = 110
```

```
matriz(9, 0) = 112
matriz(9, 1) = 112
matriz(9, 2) = 112
matriz(9, 3) = 112
matriz(9, 4) = 112
matriz(9, 5) = 112
matriz(9, 6) = 112
matriz(9, 7) = 112
matriz(9, 8) = 112
matriz(9, 9) = 112
matriz(9, 10) = 112
matriz(9, 11) = 111
matriz(9, 12) = 112
matriz(9, 13) = 113
```

```
matriz(10, 0) = 304
matriz(10, 1) = 304
matriz(10, 2) = 304
matriz(10, 3) = 304
matriz(10, 4) = 304
matriz(10, 5) = 304
matriz(10, 6) = 304
matriz(10, 7) = 304
matriz(10, 8) = 304
matriz(10, 9) = 304
matriz(10, 10) = 304
matriz(10, 11) = 304
matriz(10, 12) = 304
matriz(10, 13) = 114
```

End Sub

```
Private Sub Cargar_Click(sender As Object, e As EventArgs) Handles Cargar.Click
    OpenFileDialog1.ShowDialog()
    Dim archivo
    archivo = OpenFileDialog1.FileName
    Dim fileReader As System.IO.StreamReader
    fileReader = My.Computer.FileSystem.OpenTextFileReader(archivo)
    Dim stringReader As String
    While Not (fileReader.EndOfStream)
        stringReader = fileReader.ReadLine()
        ListBox1.Items.Add(stringReader)
    End While
End Sub
```

```

'*****
Dim renglon
Dim token, PalabraSalida
Dim Lineaz, direccion
'*****
Private Sub Compilar_Click(sender As Object, e As EventArgs) Handles Compilar.Click
    renglon = 0
    While (renglon < ListBox1.Items.Count) 'mientras que renglon =0 sea menor que la cantidad del
listbox1 entra
        ListBox1.SelectedIndex = renglon 'al indice va inicializar en el 0
        Lineaz = ListBox1.Text 'se le agregara a la linez
        BuscaToken()
        renglon = renglon + 1
    End While
End Sub

Private Sub BuscaId()
    Dim encontro = False
    Dim renglon2 = 0
    While ((encontro = False) And (renglon2 < ListBox4.Items.Count))
        If (UCase(token) = UCase(ListBox4.Text)) Then
            encontro = True
            direccion = renglon2
        End If
        renglon2 = renglon2 + 1
    End While
    If (encontro = False) Then
        ListBox4.Items.Add(token)
        direccion = ListBox4.Items.Count - 1
    End If
End Sub

Private Sub CntesEnteras()
    Dim encontro
    Dim renglon2
    encontro = False
    renglon2 = 0
    While ((encontro = False) And (renglon2 < ListBox6.Items.Count))
        ListBox6.SelectedIndex = renglon2
        If (UCase(token) = UCase(ListBox6.Text)) Then
            encontro = True
            direccion = renglon2
        End If
        renglon2 = renglon2 + 1
    End While
    If ((encontro = False)) Then
        ListBox6.Items.Add(token)
        direccion = ListBox6.Items.Count - 1
    End If
End Sub

Private Sub CntesReales()
    Dim encontro
    Dim renglon2
    encontro = False
    renglon2 = 0
    While ((encontro = False) And (renglon2 < ListBox7.Items.Count))
        ListBox7.SelectedIndex = renglon2
        If (UCase(token) = UCase(ListBox7.Text)) Then
            encontro = True
            direccion = renglon2
        End If
    End While
End Sub

```

```

        renglon2 = renglon2 + 1
    End While
    If ((encontro = False)) Then
        ListBox7.Items.Add(token)
        direccion = ListBox7.Items.Count - 1
    End If
End Sub
Private Sub Strings()
    Dim encontro
    Dim renglon2
    encontro = False
    renglon2 = 0
    While ((encontro = False) And (renglon2 < ListBox5.Items.Count))
        ListBox5.SelectedIndex = renglon2
        If (UCase(token) = UCase(ListBox5.Text)) Then
            encontro = True
            direccion = renglon2

            End If
            renglon2 = renglon2 + 1
        End While
        If ((encontro = False)) Then
            ListBox5.Items.Add(token)
            direccion = ListBox5.Items.Count - 1
        End If
    End Sub

'*****
Dim ColumnaN
Dim Posicion
Dim Caracter
'*****
Public Sub BuscaToken()
    Estado = 0
    token = ""
    Posicion = 1
    While (Posicion <= Len(Lineaz)) ' mientras que la pocicsion sea menor que la longitud de linea Z
        Caracter = Mid(Lineaz, Posicion, 1) 'extrae de una cadena el caracter indicado en la
posicion indicada
        ColumnaX()
        Estado = matriz(Estado, ColumnaN) 'caracter= cadena,inicio,#caracteres a sacar

        If (Estado >= 100) Then
            If (Len(token) > 0) Then
                ReconocimientoToken()
            ElseIf (Len(token) = 0) Then
                token = token + Caracter 'solo lee un caracter
                ReconocimientoToken()
            End If
            Estado = 0
            token = ""
        Else

            If (Estado <> 0) Then
                token = token + Caracter 'añadir a token el caracter (h)
            End If
        End If

        Posicion = Posicion + 1
    End While
    If (Len(token) > 0) Then 'el token hola es mayor si hola=4
        Caracter = " "
    End If

```

```

        ColumnaX()
        Estado = matriz(Estado, ColumnaN)
        ReconocimientoToken()
    End If
End Sub

```

```

Public Sub ColumnaX()
    If Character >= "A" And Character <= "Z" Or Character >= "a" And Character <= "z" Then
        ColumnaN = 0
    ElseIf (Character = " " Or Character = "") Then
        ColumnaN = 2
        'Constantes enteras
    ElseIf (Character >= "0" And Character <= "9") Then
        ColumnaN = 1
        'Delimitadores

    ElseIf (character = "_") Then
        ColumnaN = 3
    ElseIf (Character = ".") Then
        ColumnaN = 4
        'Strings
    ElseIf (Character = ChrW(34)) Then 'Identifica el caracter especial (Comillas) asckii
        ColumnaN = 5
    ElseIf (Character = "#") Then
        ColumnaN = 6
    ElseIf (Character = "+") Then
        ColumnaN = 7
    ElseIf (Character = "-") Then
        ColumnaN = 8
    ElseIf (Character = "*") Then
        ColumnaN = 9
    ElseIf (Character = "/" ) Then
        ColumnaN = 10
    ElseIf (Character = ">") Then
        ColumnaN = 11
    ElseIf (Character = "<") Then
        ColumnaN = 12
    ElseIf (Character = "=") Then
        ColumnaN = 13
    End If
End Sub

```

```

'*****
Dim Estado, Pr

```

```

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    ListBox1.Items.Clear()
    ListBox2.Items.Clear()
    ListBox4.Items.Clear()
    ListBox5.Items.Clear()
    ListBox6.Items.Clear()
    ListBox7.Items.Clear()
End Sub

```

```

'*****
Public Sub ReconocimientoToken()
    If Estado = 100 Then
        Pr = False
        BuscapalReservada()
    End If
End Sub

```

```

        If Pr Then
            PalabraSalida = token + " Pr " + Str(direccion)
            Posicion = Posicion - 1
            token = token + Caracter
        Else
            Buscaidentificadores()
            PalabraSalida = token + " Id " + Str(direccion)
            Posicion = Posicion - 1
        End If
    End If
    If Estado = 101 Then
        CntesEnteras()
        PalabraSalida = token + " Cnte. Entera" + Str(direccion)
        Posicion = Posicion - 1 'retrocede para volver a identificar el ultimo caracter
    ElseIf Estado = 102 Then
        CntesReales()
        PalabraSalida = token + " Cnte. Real" + Str(direccion)
        Posicion = Posicion - 1
    End If
    If Estado = 103 Then
        token = token + Caracter 'almacena las ultimas comillas
        Strings()
        PalabraSalida = token + " String" + Str(direccion)
    End If
    If Estado = 105 Then
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 106 Then
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 107 Then
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 108 Then
        PalabraSalida = token + " C.Esp"

        'operadores X
    ElseIf Estado = 109 Then
        PalabraSalida = token + " C.Esp"
        Posicion = Posicion - 1
    ElseIf Estado = 110 Then
        token = token + Caracter 'vacio = vacio+ ultimo caracter
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 111 Then
        token = token + Caracter
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 112 Then
        PalabraSalida = token + " C.Esp"
        Posicion = Posicion - 1
    ElseIf Estado = 113 Then
        token = token + Caracter
        PalabraSalida = token + " C.Esp"
    ElseIf Estado = 114 Then
        token = token + Caracter
        PalabraSalida = token + " C.Esp"
    End If

    ListBox2.Items.Add(PalabraSalida)
End Sub
Public Sub Buscaidentificadores()
    Dim encontro
    Dim renglon2
    encontro = False
    renglon2 = 0

```

```

While ((encontro = False) And (renglon2 < ListBox4.Items.Count))
    ListBox4.SelectedIndex = renglon2
    If (UCase(token) = UCase(ListBox4.Text)) Then
        encontro = True
        direccion = renglon2

    End If
    renglon2 = renglon2 + 1
End While
If ((encontro = False)) Then
    ListBox4.Items.Add(token)
    direccion = ListBox4.Items.Count - 1
End If
End Sub

Public Sub BuscapalReservada()
    Dim renglon2
    renglon2 = 0
    direccion = -1
    While ((Pr = False) And (renglon2 < ListBox3.Items.Count))
        ListBox3.SelectedIndex = renglon2
        If (UCase(token) = UCase(ListBox3.Text)) Then
            Pr = True
            direccion = renglon2 'guarda la direccion de renglon2
        End If
        renglon2 = renglon2 + 1
    End While
End Sub

End Class

```

ARCHIVO DE PRUEBA



```

pruebapython: Bloc de notas
Archivo Edición Formato Ver Ayuda
false ejemplo+

none a+b+c+true+
none a+b+c+true+

as

a<5+
a<5+
a<5.0+
a<5.0+
a<5.5+
a<5.5+


b+6+
b+6+
b+6.0+
b+6.0+
b+6.6+
b+6.6+

c+"Cte. String 1"+
c+"Cte. String 1"+
c+"Cte. String 2"+
c+"Cte. String 2"+
c+"Cte. String 3"+
c+"Cte. String 3"+
break

```

EJECUCION

ProyectoX

 **COMPILADOR**

(PYTHON)

```
a<5.0+
a<5.5+
a<5.5+




b+6+
b+6+
b+6.0+
b+6.0+
b+6.6+
b+6.6+

c+"Cte. String 1"+
c+"Cte. String 1"+
c+"Cte. String 2"+
c+"Cte. String 2"+
c+"Cte. String 3"+
c+"Cte. String 3"+
break
```

Palabras Reservadas

- False
- None
- True
- And
- As
- Await
- Break
- Class
- Continue
- Def
- Elif
- Else
- Except
- Finally
- For
- From
- Global
- If

Archivo de Entrada

Archivo de Salida

```
false Pr 0
ejemplo Id 0
+C.Esp
none Pr 1
a Id 1
+C.Esp
b Id 2
+C.Esp
c Id 3
+C.Esp
true Pr 2
+C.Esp
none Pr 1
a Id 1
+C.Esp
b Id 2
+C.Esp
c Id 3
```

Strings

```
"Cte. String 1"
"Cte. String 2"
"Cte. String 3"
```

Identificadores

```
ejemplo
a
b
c
```

Ctes. Enteras

```
5
6
```

Ctes. Reales

```
5.0
5.5
6.0
6.6
```