

Lección 03: Léxico

Índice

Anterior

Un programa en Python

Un programa en Python es un archivo de texto, por defecto en codificación UTF-8. El programa se divide en líneas lógicas que son leídas por un *parser* (analizador sintáctico) e interpretadas. Los programas en python tienen extensión “.py”.

En Python, las líneas de código NO terminan con ;

Comentarios

Los comentarios en Python pueden ser: - De una línea: Comienzan por # - Múltiples líneas: Comienzan por """ y acaban con """

```
# Esto es un comentario de una línea
```

```
""" Esto también es
un comentario, pero
de múltiples líneas
"""
```

Unión de líneas

Varias líneas físicas pueden unirse en una única línea lógica. Esto se utiliza para escribir código más legible.

- Pueden unirse explícitamente utilizando la barra invertida
- Pueden unirse implícitamente las expresiones entre paréntesis, corchetes o llaves.

```
# Línea lógica en varias líneas físicas (unión explícita usando \)
if 1900 < year < 2100 and 1 <= month <= 12 \
    and 1 <= day <= 31 and 0 <= hour < 24 \
    and 0 <= minute < 60 and 0 <= second < 60:    # Looks like a valid date
    return 1
```

```
# Línea lógica en varias líneas físicas (unión implícita)
month_names = ['Januari', 'Februari', 'Maart',      # These are the
               'April',  'Mei',      'Juni',        # Dutch names
               'Juli',   'Augustus', 'September',   # for the months
               'Oktober', 'November', 'December']   # of the year
```

Tabuladores

Los tabuladores al principio de línea en Python son muy importantes porque se utilizan para determinar la agrupación de las declaraciones. A diferencia de otros lenguajes, la agrupación se hace con tabuladores, no con llaves, paréntesis y otros caracteres.

Nota: Los errores de tabulación hacen que el código no pueda ser interpretado por el parser y serán detectados como errores de indentación.

Keywords

Palabras reservadas del lenguaje que no pueden utilizarse como identificadores de variable, nombre de función, ni para otra cosa. Son estas:

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

Funciones Built-in

El intérprete de Python tiene una serie de funciones y tipos incluidos en él que están siempre disponibles. Son estas:

abs()	delattr()	hash()	memoryview()	set()	all()
min()	setattr()	any()	dir()	hex()	next()
divmod()	id()	object()	sorted()	bin()	enumerate()
staticmethod()	bool()	eval()	int()	open()	str()
isinstance()	ord()	sum()	bytearray()	filter()	issubclass()
bytes()	float()	iter()	print()	tuple()	callable()
property()	type()	chr()	frozenset()	list()	range()
getattr()	locals()	repr()	zip()	compile()	globals()
__import__()	complex()	hasattr()	max()	round()	

Siguiente