

Introducción al lenguaje de programación Python

M.C.I. Victor Manuel Mora Romo



Modulo II

Variables y Operadores



GLOSARIO

Lenguaje informático: es un idioma artificial, utilizado por ordenadores, cuyo fin es transmitir información de algo a alguien.

Los lenguajes informáticos, pueden clasificarse en:

- a) lenguajes de programación (Python, PHP, Pearl, C, etc.);
- b) lenguajes de especificación (UML);
- c) lenguajes de consulta (SQL);
- d) lenguajes de marcas (HTML, XML);
- e) lenguajes de transformación (XSLT);
- f) protocolos de comunicaciones (HTTP, FTP).

Lenguaje de programación: es un lenguaje informático, diseñado para expresar órdenes e instrucciones precisas, que deben ser llevadas a cabo por una computadora. El mismo puede utilizarse para crear programas que controlen el comportamiento físico o lógico de un ordenador. Está compuesto por una serie de símbolos, reglas sintácticas y semánticas que definen la estructura del lenguaje.



GLOSARIO

Lenguajes de alto nivel: son aquellos cuya característica principal, consiste en una estructura sintáctica y semántica legible, acorde a las capacidades cognitivas humanas. A diferencia de los lenguajes de bajo nivel, son independientes de la arquitectura del hardware, motivo por el cual, asumen mayor portabilidad.

Lenguajes interpretados: a diferencia de los compilados, no requieren de un compilador para ser ejecutados sino de un intérprete. Un intérprete, actúa de manera casi idéntica a un compilador, con la salvedad de que ejecuta el programa directamente, sin necesidad de generar previamente un ejecutable. Ejemplo de lenguajes de programación interpretado son Python, PHP, Ruby, Lisp, entre otros.

Tipado dinámico: un lenguaje de tipado dinámico es aquel cuyas variables, no requieren ser definidas asignando su tipo de datos, sino que éste, se auto-asigna en tiempo de ejecución, según el valor declarado.

Multiplataforma: significa que puede ser interpretado en diversos Sistemas Operativos como GNU/Linux, Windows, Mac OS, Solaris, entre otros.



GLOSARIO

Multiparadigma: acepta diferentes paradigmas (técnicas) de programación, tales como la orientación a objetos, aspectos, la programación imperativa y funcional.

Código fuente: es un conjunto de instrucciones y órdenes lógicas, compuestos de algoritmos que se encuentran escritos en un determinado lenguaje de programación, las cuales deben ser interpretadas o compiladas, para permitir la ejecución del programa informático.



Elementos del Lenguaje

Como en la mayoría de los lenguajes de programación de alto nivel, Python se compone de una serie de elementos que alimentan su estructura. Entre ellos, podremos encontrar los siguientes:



I.- Comentarios

Un archivo, no solo puede contener código fuente. También puede incluir comentarios (notas que como programadores, indicamos en el código para poder comprenderlo mejor).

Los comentarios pueden ser de dos tipos: de una sola línea o multi-línea y se expresan de la siguiente manera:

Esto es un comentario de una sola línea

`mi_variable = 15`

"""Y este es un comentario

de varias líneas"""

`mi_variable = 15`

`mi_variable = 15 # Este comentario es de una línea también`



II.-Variables

Una variable es un espacio para almacenar datos modificables, en la memoria de un ordenador. En Python, una variable se define con la sintaxis:

nombre_de_la_variable = valor_de_la_variable

Cada variable, tiene un nombre y un valor, el cual define a la vez, el tipo de datos de la variable.

Existe un tipo de “variable”, denominada **constante**, la cual se utiliza para definir valores fijos, que no requieran ser modificados.



PEP 8: variables

Utilizar nombres descriptivos y en minúsculas. Para nombres compuestos, separar las palabras por guiones bajos. Antes y después del signo =, debe haber uno (y solo un) espacio en blanco

Correcto: `mi_variable = 12`

Incorrecto: `MiVariable = 12` | `mivariable = 12` | `mi_variable=12` |
`mi_variable = 12`

PEP 8: constantes

Utilizar nombres descriptivos y en mayúsculas separando palabras por guiones bajos.

Ejemplo: `MI_CONSTANTE = 12`

Para **imprimir un valor en pantalla**, en Python, se utiliza la palabra clave `print`:

```
mi_variable = 15  
print mi_variable
```

Lo anterior, imprimirá el valor de la variable `mi_variable` en pantalla.



III.- Operadores Aritméticos

Entre los operadores aritméticos que Python utiliza, podemos encontrar los siguientes:

Símbolo	Significado	Ejemplo	Resultado
+	Suma	<code>a = 10 + 5</code>	a es 15
-	Resta	<code>a = 12 - 7</code>	a es 5
-	Negación	<code>a = -5</code>	a es -5
*	Multiplicación	<code>a = 7 * 5</code>	a es 35
**	Exponente	<code>a = 2 ** 3</code>	a es 8
	raíz cuadrada	<code>a = 9**(0.5)</code>	a es 3
/	División	<code>a = 12.5 / 2</code>	a es 6.25
//	División entera	<code>a = 12.5 // 2</code>	a es 6.0
%	Módulo	<code>a = 27 % 4</code>	a es 3

PEP 8: operadores

Siempre colocar un espacio en blanco, antes y después de un operador



En los comentarios, pueden incluirse palabras que nos ayuden a identificar además, el subtipo de comentario:

TODO esto es algo por hacer
FIXME esto es algo que debe corregirse
XXX esto también, es algo que debe corregirse

PEP 8: comentarios

Comentarios en la misma línea del código deben separarse con dos espacios en blanco. Luego del símbolo # debe ir un solo espacio en blanco.

Correcto:

```
a = 15 # Edad de María
```

Incorrecto:

```
a = 15 # Edad de María
```





!!! Gracias !!!