

INSTITUTO TECNOLÓGICO SUPERIOR DE CHICONTEPEC

INGENIERÍA EN SISTEMAS COMPUTACIONALES



NOMBRE DE LA MATERIA:
MÉTODOS NUMÉRICOS.

NOMBRE DEL TEMA:
PYTHON.

NOMBRE DEL ALUMNO:
MIGUEL ANGEL MARTINEZ MARTINEZ.

NOMBRE DEL DOCENTE:
ING. EFRÉN FLORES CRUZ.

FECHA DE ENTREGA:
21/MARZO/2020.

Índice

Introducción.....	1
Información referente a Python	2
Elementos que conforma Python	3
Conclusión.....	8
Bibliografía.....	9

Introducción

En el presente trabajo tiene la función de mencionarte algunos componentes que componen a Python, además de darle los más importante que utiliza Python.

Python es un lenguaje de programación de propósito general muy poderoso y flexible, a la vez que también es muy sencillo el poder utilizar y fácil de aprender.

Información referente a Python

Python.

Es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

Python es un lenguaje de scripting de propósito y orientado a objetos, preparado para realizar cualquier tipo de programación, desde aplicaciones Windows a servidores de red o incluso páginas web. Es un lenguaje interpretado, lo que significa que no necesita compilarse el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad.

Python es la navaja suiza de los programadores. Se trata de un veterano lenguaje de programación presente en multitud de aplicaciones y sistemas operativos. Podemos encontrarlo corriendo en servidores, en aplicaciones iOS, Android, Linux, Windows o Mac. Esto es debido a que cuenta con una curva de aprendizaje moderada y, que su filosofía hace hincapié en ofrecer un sintaxis de código legible.

Es un lenguaje de programación versátil, multiplataforma y multiparadigma, que se destaca por su código legible y limpio. Uno de los motivos de su éxito es que cuenta con una licencia de código abierto que permite su utilización en cualquier escenario.

Elementos que conforma Python

Elementos.

Un programa en python es un archivo de texto (normalmente guardado con el juego de caracteres UTF-18) que contiene expresiones y sentencias del lenguaje python. Estas expresiones y sentencias consiguen combinando los elementos básicos del lenguaje.

- Palabras reservadas

Las palabras reservadas de Python son las formas básicas del lenguaje 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

- Literales

Los literales son los datos simples de Python en su forma más básica.

- números: valores lógicos, enteros, decimales y complejos, en notación decimal, octal o hexadecimal.
- cadenas de texto

- Operadores.

Los operadores son las características que definen operaciones matemáticas (lógicas y aritméticas). Son los siguientes:

+ - * ** / // % @
<< >> & | ^ ~
< > <= >= == !=

- Delimitadores.

Los delimitadores son los caracteres que permiten delimitar, separar o reparar expresiones. Son los siguientes:

' " # \
() [] { }
, : ; @ =
+= -= *= /= //= %=
&= |= ^= >>= <<= **=

- Identificadores.

Los identificadores son las palabras que utilizan para nombrar elementos creados por el usuario o otros usuarios. Estos elementos pueden ser variables u objetos que almacenan información, funciones que agrupan instrucciones, clases que combinan ambos, modelos que agrupan los elementos anteriores, etc.

Los identificadores están formados por letras (mayúsculas y minúsculas), números y caracteres guión bajo (_). Pueden ser caracteres Unicode, aunque normalmente se recomienda utilizar caracteres ASCII.

para evitar las complicaciones al uso de dichos países que otorgan
jugos de carácter diferente

El primer carácter del identificador debe ser una letra.

- Funciones.

Una función es un bloque de instrucciones agrupadas, que
permiten utilizar partes de un programa python incluye las siguientes
Para: predefinida (es decir, siempre estas funciones estan disponibles).

abs()	dict()	help()	min()
all()	dir()	hex()	next()
any()	divmod()	id()	object()
ascii()	enumerate()	input()	oct()
bin()	eval()	int()	open()
bool()	exec()	isinstance()	ord()
bytearray()	filter()	issubclass()	pow()
bytes()	float()	iter()	print()
callable()	format()	len()	property()
chr()	frozenset()	list()	range()
classmethod()	getattr()	locals()	repr()
compile()	globals()	map()	reversed()
complex()	hasattr()	max()	round()
delattr()	hash()	memoryview()	set()
setattr()	slice()	sorted()	staticmethod()
str()	sum()	super()	tuple()
type()	vars()	zip()	

- import -()

- Tipos de datos.

• Números enteros (int)

- Decimal: 24:60

- Binario: 0b010011, 0b1101

- Hexadecimal: 0x18, 0x3c14

- Octal: 030, 074

• Números de punto flotante (float)

• 3.141595

• 12.

• -45.3556

Los objetos tipo float corresponden al conjunto de los números reales.

• Números complejos (complex).

Los objetos de tipo complex corresponden al conjunto de los números complejos.

• $6.32 + 45j$

• $6.117j$

• $(2 + 0j)$

• $1j$

Siempre que el componente en los números sea distinto de 0, los objetos de tipo complex se expresarán como un par de números de tipo float separados por el operador de adición "+" en el que primer número correspondiente al componente de los números reales y el componente de los números imaginarios se identifica añadiéndole la letra "j" al final.

- Entrada / Salida en Python

• Entrada estandar

para poder informarnos al usuario, debe utilizar las funciones integradas en el intérprete del lenguaje, así como los argumentos de línea de comando.

Ejemplo:

La función `raw_input()` siempre devuelve un valor de cadena de caracteres.

La función `input()` siempre devuelve un valor numérico.

• Salida estandar

La forma general de mostrar información por pantalla es mediante una consola de comando, generalmente podemos mostrar texto y variables separados por comas pero esto se usa la sentencia `print`.

Conclusión

Cabe señalar que mediante la investigación me fui dando cuenta que Python es fácil de utilizar y de aprender a manejarlo, además que tiene grandes ventajas, tal es el caso de que no necesita compilarse como otros lenguajes de programación, ya que vi que es un lenguaje de programación libre de utilizar, donde el programador podrá hacer sus actividades.

La forma de trabajar con Python es de la misma manera que se utiliza con java, pero hay ciertas partes que cambian, una de ellas como ya mencionamos antes el aquel no necesita compilación del programa, también podemos trabajar con matemáticas, utilizando ciertos comandos que tendremos que ingresar.

Pero vi que también tiene la función del blog de notas, que una materia lo ocupamos cierto blog de notas, que es de la misma forma y estructura que tiene. Dándome cuenta que Python proviene también de Linux.

Esta investigación que hicimos fue el de tender una idea sobre como trabajar con Python, ya que instalamos el lenguaje de programación para la materia de Método Numéricos. Donde es una consola de trabajo.

Bibliografía

Elementos de un programa de Python. (s.f.). Recuperado el 20 de Marzo de 2020, de [www.mclibre.org: https://www.mclibre.org/consultar/python/lecciones/python-elementos.html](https://www.mclibre.org/consultar/python/lecciones/python-elementos.html)

Entrada/ Salida de Python. (s.f.). Recuperado el 21 de Marzo de 2020, de [entrenamiento-python-basico.readthedocs.io: https://entrenamiento-python-basico.readthedocs.io/es/latest/leccion7/entrada_salida.html](https://entrenamiento-python-basico.readthedocs.io/es/latest/leccion7/entrada_salida.html)

Tipos de datos basicos y operadores. (s.f.). Recuperado el 21 de Marzo de 2020, de [pythonista.io: https://pythonista.io/cursos/py101/tipos-de-datos-basicos-y-operadores](https://pythonista.io/cursos/py101/tipos-de-datos-basicos-y-operadores)