

INSTITUTO TECNOLÓGICO SUPERIOR DE CHICONTEPÉC

Materia:

Métodos Numéricos

Nombre del trabajo:

Investigación Ubuntu

Alumno:

Leopoldo Bautista Ramírez.

Docente:

Ing. Efrén Flores Cruz

N. de control:

1817V0016

29 de Marzo del 2020

Números y operaciones aritméticas

FOLIO

Tipos de datos enteros, decimales y números complejos.

* Enteros y decimales

Python distingue entre números decimales y complejos.

Al ~~dejar~~ escribir un número decimal, el separador entre la parte entera y decimal es un punto.

```
>>> 3
```

```
3
```

```
>>> 4.5
```

```
4.5
```

* Números complejos

Python puede hacer cálculos con números complejos.

La parte imaginaria se acompaña de la letra "j".

```
>>> 1 + 1j + 2j + 3j
```

```
3 + 4j
```

```
>>> (1 + 1j) * 1j
```

```
(-1 + 1j)
```

* Las 4 operaciones básicas

Las 4 operaciones básicas son suma, resta, multiplicación y división.

Al hacer operaciones en las que se involucran números enteros y decimales, el resultado es siempre decimal. En el caso de que el resultado no tenga decimal, Python escribe 0 como parte decimal para indicar que el resultado es un número decimal.

>>> 4.5 * 3

13.5

>>> 4.5 * 2

9.0

Al sumar, restar o multiplicar números enteros el resultado es entero

>>> 1 + 2

3

>>> 3 - 4

-1

>>> 5 * 6

30

Al dividir números enteros, el resultado es siempre decimal aunque sea un número entero. Cuando Python escribe un número decimal, lo escribe siempre con parte decimal, aunque sea nula

>>> 9 / 2

4.5

>>> 9 / 3

3.0

Tipos de Booleanos true y false

FOLIO

Una variable booleana es una variable que solo puede tomar dos posibles valores: True (Verdadero) o False (Falso)

En Python cualquier variable (en general, cualquier objeto) puede considerarse como una variable booleana. En general los elementos nulos y vacíos se consideran False y el resto se consideran True

Operadores lógicos

- Los operadores lógicos son unas operaciones que trabajan con valores booleanos

*and: y lógico. Este operador da como resultado true si y solo si sus dos operadores es true:

>> True and True

True

>> True and False

False

*> False and False

False

>> False and True

False

Cadenas de texto.

DIA

MES

AÑO

FOLIO

Manipular cadenas de caracteres en Python si has estado expuesto antes a otro lenguaje de programación sabrás que necesitas declarar o escribir variables antes de que puedas almacenar algo en ella, esto no es necesario cuando trabajas con cadenas de caracteres en Python. Podemos crear una cadena de caracteres simplemente encerrando contenido entre comillas después de un signo de igual.

```
= mensaje = "Hola mundo"
```

Concatenar: En termino significa juntar cadenas de caracteres. El proceso de concatenación se realiza mediante el operador de suma (+). Ten en cuenta que deberás marcar explícitamente donde quieres los espacios en blanco y colocarlos entre comillas.

```
Mensaje = "Hola " + "Mundo"
```

```
print(Mensaje)
```

```
-> Hola Mundo
```

Multiplicar

si quieres varias copias de una cadena de caracteres podemos utilizar el operador de multiplicación (*).

```
Mensaje2a = "Hola" * 3
```

```
mensaje2b = "Mundo"
```

```
print(Mensaje2a + mensaje2b)
```

```
-> Hola Hola Hola Mundo
```


Tipos de variables en Python

Una variable es un sitio donde guardamos una determinada información.

En Python hay que tener en cuenta que si escribimos en mayúsculas o minúsculas la variable no significará lo mismo y tendremos dos distintos.

Números:

En los números tenemos dos tipos, los enteros llamados `int` y los reales llamados `float`. El separador decimal que tenemos que usar será el punto.