

## POR LUCHITO RÍOS

### TIPOS DE DATOS

int = enteros, ejemplo = 5

float = flotantes, ejemplo = 3.4

list = lista, ejemplo = [1,2,3,4]

complex = complejo, ejemplo = (2-4.5j)

str = string, ejemplo = "hola mundo"

booleanos = True False

### PRECEDENCIA:

#### Operaciones aritméticas

Operación	Operador	Aridad	Asociatividad	Prioridad
Exponenciación	**	Binaria	Por la derecha	1
Identidad	+	Unaria	----	2
Cambio de Signo	-	Unaria	----	2
Multiplicación	*	Binaria	Por la izquierda	3
División	/	Binaria	Por la izquierda	3
División entera	//	Binaria	Por la izquierda	3
Módulo (resto)	%	Binaria	Por la izquierda	3
Suma	+	Binaria	Por la izquierda	4
Resta	-	Binaria	Por la izquierda	4

#### General:

1. Primero paréntesis
2. Segundo operaciones % / \* + -
3. Tercero operadores de comparación <, >, >=, <=, !=
4. Cuarto not
5. Quinto and
6. Sexto or

### DIVISIÓN

#### División entera

5//2 = 2

#### División no entera:

5/2 = 2.5

5/2.0 = 2.5

### STRING

string = "hola"

### posiciones o índices

string[0] = string[-4] = "h"

string[1] = string[-3] = "o"

string[2] = string[-2] = "l"

string[3] = string[-1] = "a"

Los string son inmutables !

ejemplo: string[0] = "f" No se puede

### Métodos

**Concatenar:** se utiliza "+", siempre que sean dos variables de tipo string, ejemplo:

a = "hola"

b = " como estai"

a+b = "hola como estai"

**Repetir:** se multiplica por un número entero, ejemplo: "hola"\*3 = "holaholahola"

**len():** largo, ejemplo: len(string) = 4

**upper():** a mayúsculas, string.upper() = "HOLA"

**lower():** a minúsculas, string.lower() = "hola"

**strip()** = elimina espacios en blanco y saltos de línea iniciales y finales, ejemplo string = " hola mundo \n", string.strip() = "hola mundo"

### Métodos usados generalmente en sentencias IF retornar True o False

isalpha(): si es alfabético, string.isalpha() = True

islower(): si es minúscula, string.islower() = True

isupper(): si es mayúscula, string.isupper() = False

isdigit(): si es dígito, string.isdigit() = False

### LISTAS

lista = [1,2,3,4]

### posiciones o índices

lista [0] = lista [-4] = 1

lista [1] = lista [-3] = 2

lista [2] = lista [-2] = 3

lista [3] = lista [-1] = 4

Listas son mutables

### Métodos Listas

**Concatenar:** se utiliza "+", siempre que sean dos variables de tipo lista, ejemplo:

a = [1,2,3]

b = [4,5,6]

a+b = [1,2,3,4,5,6]

**Repetir:** se multiplica por un número entero, ejemplo: [1,2,3]\*3 = [1,2,3,1,2,3,1,2,3]

**len():** largo, ejemplo len(lista) = 4

**append():** agregar element al final, lista.append(5)

**pop(),** eliminar, sin parámetro elimina el último elemento, lista.pop()

con parámetro, elimina elemento posición dada.

lista.pop(2)

**index(),** primera posición (habiendo uno o más) elemento dado, lista.index(2)

**remove(),** elimina elemento, a diferencia del pop, elimina el elemento no por la posición. lista.remove(1)

**insert,** recibe dos parámetros, primero, posición a insertar, 2do elemento a insertar. lista.insert(2,9)

**sort():** Ordena una lista de menor a mayor, ejemplo: lista.sort()

**count():** Cuenta la cantidad de ocurrencias de un elemento en una lista, ejemplo: lista.count(1) → 1

## POR LUCHITO RÍOS

### De string a lista

split(): convierte string a lista, ejemplo:

```
a = "hola mundo"
```

```
lista = a.split()
```

```
lista = ["hola", "mundo"]
```

```
fecha = "DD/MM/AAAA"
```

Separa en el carácter

```
lista = fecha.split("/") = ["DD", "MM", "AAAA"]
```

```
string = "hola mundo"
```

```
lista = list(string)
```

```
['h', 'o', 'l', 'a', ' ', 'm', 'u', 'n', 'd', 'o']
```

### De lista a string

Join concatena los elementos de la lista con el parámetro indicado, los elementos de la lista solo deben ser string

```
Lista = ["hola", "como", "estas"]
```

```
String = " ".join(Lista)
```

```
"hola como estas"
```

```
String2 = "/".join(Lista)
```

```
"hola/como/estas"
```

```
String3 = str(Lista)
```

```
"["hola", "como", "estas"]"
```

## ENTRADAS TECLADO

```
entrada = input("ingrese algo: ")
```

convierte todo lo ingresado a string, para ingresar un string no es necesario hacerlo con comillas

## WHILE

### Listas

#### Recorrer por posición

```
i = 0
```

```
while i < len(lista):
```

```
    print (lista [i])
```

```
    i+=1
```

### String

#### Recorrer por posición

```
i = 0
```

```
while i < len(string):
```

```
    print (string[i])
```

```
    i+=1
```

### Se encuentra en

```
string = "hola"
```

```
if "a" in string:
```

```
    print ("La palabra se encuentra en el string")
```

```
lista = [1,2,3]
```

```
if 2 in lista:
```

```
    print ("El número se encuentra en la lista")
```

## FUNCIONES

```
# -*- coding: cp1252 -*-
#IMPORTACIÓN DE FUNCIONES
#DEFINICIÓN DE CONSTANTES
#DEFINICIÓN DE FUNCIONES
#BLOQUE PRINCIPAL
#ENTRADA DE DATOS
#PROCESAMIENTO
#SALIDA
```

Ejemplo:

```
# -*- coding: cp1252 -*-
#DEFINICIÓN DE CONSTANTES
#IMPORTACIÓN DE FUNCIONES
#DEFINICIÓN DE FUNCIONES
def Sumar(x,y):
    resultado = x +y
    return resultado
```

```
def multiplicar(x,y):
    resultado = x*y
    return resultado
```

```
#BLOQUE PRINCIPAL
#ENTRADA DE DATOS
A = int(input("Ingrese número: "))
B = int(input("Ingrese número: "))
C = input("Ingrese número: ")
#PROCESAMIENTO
Suma = Sumar(A,B)
Suma2 = Sumar(Suma,C)
Multiplicacion = Multiplicar(C,A)

#SALIDA
print ("El resultado de la suma entre
",A," y ",B," es ",Suma)
print ("El resultado de la suma entre
",Suma," y ",C," es ",Suma2)
print ("El resultado de la multiplicación
entre ",C," y ",A," es ",Multiplicacion)
```

**VARIABLES LOCALES:** Una variable local es aquella cuyo ámbito se restringe a la función que la ha declarado se dice entonces que la variable es local a esa función. Esto implica que esa variable sólo va a poder ser manipulada en dicha sección, y no se podrá hacer referencia fuera de dicha sección.

**VARIABLES GLOBALES:** Una variable global es aquella que se define fuera del cuerpo de cualquier función, El ámbito de una variable global son todas las funciones que componen el programa, cualquier función puede acceder a dichas variables para leer y escribir en ellas.

## MÓDULOS

### MATH

La función de seno y coseno se importan desde este módulo:

```
from math import sin,cos
```

USO: Variable = sin (90)

POR LUCHITO RÍOS

## **RANDOM**

```
from random import randint
```

USO: Randint toma 2 parámetros, si tengo 0 y 100 se generará un número aleatorio entre 0 y 100 (se incluye ambos)

```
valorEnteroRandom = randint(0,100)
```