

Práctica 1: Variables, colecciones y estructuras.

Edwin Axel Vizuet Gil

El objetivo de la práctica es analizar el funcionamiento de las variables, colecciones (listas, tuplas, conjuntos, diccionarios) y las estructuras en Python.

Como primer ejercicio, se analizará el comportamiento de las variables, así como su declaración y operaciones:

- Las variables distinguen entre mayúsculas y minúsculas.
- No es necesario escribir el tipo de variable.

In [ ] : A = 1; a = 2

In [ ] : A + a

In [ ] : carros = 100  
potencia = 'turbo'  
cilindros = 24

En los siguientes ejemplos, con la función "type()" se puede determinar el tipo de variable asignada, debido a que Python las interpreta automáticamente.

In [ ] : type(carros)

In [ ] : type(potencia)

In [9]: type(cilindros)

Out[9]: int

In [11]: verdad = True

In [12]: print(verdad)

True

In [13]: print(not verdad)

False

In [17]: a = 1; b = 1

In [16]: a + b

Out[16]: 2

Colecciones

A continuación, se mostrará la declaración, uso y comportamiento de las listas, tuplas, conjuntos y diccionarios.

In [39]: lista = []  
lista = ['a',1,[1,2,3],1.5]

In [29]: print(lista)

['a', 1, [1, 2, 3], 1.5]

Los índices negativos sobre una lista recorren los últimos elementos de la misma.

In [35]: print(lista[-2])

[1, 2, 3]

In [40]: lista[1:3]

Out[40]: [1, [1, 2, 3]]

In [41]: a = ['gato','perro','perico','pinguino']

Se pueden usar estructuras iterativas para recorrer, manipular y mostrar los elementos de una lista.

In [42]: for i in a:  
print(i,len(i))

gato 4  
perro 5  
perico 6  
pinguino 8

In [43]: b = range(10)

Se puede usar el símbolo "?" después de una función para ver una descripción y los parámetros que requieren.

In [45]: range?

In [ ] : for i in range(len(b)):  
print(i,b[i])

In [48]: colores = ('rojo','azul','verde')  
print(colores)  
type(colores)

('rojo', 'azul', 'verde')

Out[48]: tuple

Se pueden crear tuplas dentro de una lista, incrementando el uso y comportamiento de las mismas.

In [1]: Lista=[('bus-stop','library'),('library','student-center')]

In [2]: print(Lista)

[('bus-stop', 'library'), ('library', 'student-center')]

Los conjuntos ordenan los elementos de menor a mayor, así mismo, eliminan los elementos repetidos.

In [6]: conjunto={}  
conjunto={1,2,3,4,5,1}  
print(conjunto)

{1, 2, 3, 4, 8, 9}

In [57]: subconjunto = {3,2,1}  
print(subconjunto)

{1, 2, 3}

Los diccionarios contienen un índice con su elemento en una sola posición.

In [7]: diccionario = {'a':1,  
'b':2}  
print(diccionario)

{'a': 1, 'b': 2}

Estructuras de control

Como último ejercicio, se mostrarán ejemplos de estructuras usadas en Python.

In [2]: print('Adivina un numero entre uno y diez')  
adivina = int(input())

Adivina un numero entre uno y diez  
3

In [3]: if adivina < 5:  
print('El numero es más alto')  
if adivina == 5:  
print('Adivinaste')  
else:  
print('No adivinaste')  
elif adivina >= 5:  
if adivina == 5:  
print('Adivinaste')  
else:  
print('El numero es menor')  
else:  
print('Adivinaste a la primera')

El numero es más alto  
No adivinaste

In [3]: edad = int(input("Escribe tu edad: "))

Escribe tu edad: 3

In [5]: if edad <= 16 or edad >= 65:  
print("Disfruta tu dia")  
else:  
print("Disfruta tu trabajo")

Disfruta tu dia

In [8]: parrot = 'El perico verde'  
letra = input('Ingrese una letra : ')  
if letra not in parrot:  
print("Esta letra no se encuentra en el enunciado")  
else:  
print("Si existe")

Ingrese una letra : verde  
Si existe

In [10]: numero = '9,223,372,036,854,775,807'  
numerossincomas = ''  
for char in numero:  
if char in '0123456789':  
numerossincomas = numerossincomas + char  
nuevonumero = int(numerossincomas)  
print('El numero es: {}'.format(nuevonumero))

El numero es: 9223372036854775807

In [18]: for estado in {'Colgo los tenis','Estiro la pata','Esta tiezo','Esta en el cielo de las aves'}:  
print('El perico '+estado)

El perico Esta tiezo  
El perico Estiro la pata  
El perico Colgo los tenis  
El perico Esta en el cielo de las aves

In [21]: for i in range(0,100,5):  
print("i es {}".format(i))

i es 0  
i es 5  
i es 10  
i es 15  
i es 20  
i es 25  
i es 30  
i es 35  
i es 40  
i es 45  
i es 50  
i es 55  
i es 60  
i es 65  
i es 70  
i es 75  
i es 80  
i es 85  
i es 90  
i es 95

In [29]: for i in range(1,11,1):  
for x in range(1,11,1):  
print(i,"x",x)

1 x 1  
1 x 2  
1 x 3  
1 x 4  
1 x 5  
1 x 6  
1 x 7  
1 x 8  
1 x 9  
1 x 10  
2 x 1  
2 x 2  
2 x 3  
2 x 4  
2 x 5  
2 x 6  
2 x 7  
2 x 8  
2 x 9  
2 x 10  
3 x 1  
3 x 2  
3 x 3  
3 x 4  
3 x 5  
3 x 6  
3 x 7  
3 x 8  
3 x 9  
3 x 10  
4 x 1  
4 x 2  
4 x 3  
4 x 4  
4 x 5  
4 x 6  
4 x 7  
4 x 8  
4 x 9  
4 x 10  
5 x 1  
5 x 2  
5 x 3  
5 x 4  
5 x 5  
5 x 6  
5 x 7  
5 x 8  
5 x 9  
5 x 10  
6 x 1  
6 x 2  
6 x 3  
6 x 4  
6 x 5  
6 x 6  
6 x 7  
6 x 8  
6 x 9  
6 x 10  
7 x 1  
7 x 2  
7 x 3  
7 x 4  
7 x 5  
7 x 6  
7 x 7  
7 x 8  
7 x 9  
7 x 10  
8 x 1  
8 x 2  
8 x 3  
8 x 4  
8 x 5  
8 x 6  
8 x 7  
8 x 8  
8 x 9  
8 x 10  
9 x 1  
9 x 2  
9 x 3  
9 x 4  
9 x 5  
9 x 6  
9 x 7  
9 x 8  
9 x 9  
9 x 10  
10 x 1  
10 x 2  
10 x 3  
10 x 4  
10 x 5  
10 x 6  
10 x 7  
10 x 8  
10 x 9  
10 x 10

In [5]: for i in range(1,21,1):  
if i % 3 == 0 and i % 5 == 0:  
print(i," es divisible entre 3 y 5")  
if i % 3 == 0:  
print(i," es divisible entre 3")  
if i % 5 == 0:  
print(i," es divisible entre 5")  
else:  
print(i, " no es divisible entre 3 o 5")

1 no es divisible entre 3 o 5  
2 no es divisible entre 3 o 5  
3 es divisible entre 3  
4 no es divisible entre 3 o 5  
5 es divisible entre 5  
6 es divisible entre 3  
7 no es divisible entre 3 o 5  
8 no es divisible entre 3 o 5  
9 es divisible entre 3  
10 es divisible entre 5  
11 no es divisible entre 3 o 5  
12 es divisible entre 3  
13 no es divisible entre 3 o 5  
14 no es divisible entre 3 o 5  
15 es divisible entre 3 y 5  
16 no es divisible entre 3 o 5  
17 no es divisible entre 3 o 5  
18 es divisible entre 3  
19 no es divisible entre 3 o 5  
20 es divisible entre 5

In [ ] :