2025/11/30 17:24 1/4 Introducción a python

# Introducción a python

#### **Variables**

Python no es un lenguaje fuertemente tipado, por lo que muchas veces para obtener una variable simplemente se pone el nombre seguido de un igual y el valor:

```
variable = "hola"
numero = 1

#Para saber el tipo de dato de una variable usamos type:
print(type(variable), type(numero))
```

### **Arrays**

En python se puede cargar casi cualquier valor a un array independientemente del tipo, por lo que hay que tener cuidado.

```
Array = [1,2,3,4] #Array standard
ArrayMultiTipo = [1, 2, "patata", "pato", 5] #Array de elementos de varios
tipos

DelUnoAl100 = [x for x in range (1000)] #Array compuesto por los números del
1 al 1000
De2en2 = [x fo x in range(0,1000,2)] #Array compuesto por los números pares
hasta el 1000.
#Range(NumInicio, NumFin, Salto)
```

Para el manejo de los array tenemos los siguientes ejemplos:

```
array = [x for x in range (0,100)]#array con valores del 0 al 100
size = len(array) #0btenemos el tamaño del array

print(array[:10])#Mostramos los 10 primeros elementos del array
print(array[-10:0])#Mostramos los 10 últimos elementos del array
print(array[5:20])#Mostramos los elementos de la posición 5 a la 20
print(array[85:len(array)-5])#Mostramos los elementos de la posición 85 a la
última posición -5 elementos

for value in array[::2]:#Recorremos las posiciones del array de 2 en 2
    print(value)
```

#### **Comentarios**

Para hacer comentarios utilizamos # para una sola línea y 3 comillas (""") para un comentario multilínea:

```
#Comentario de una línea
"""
Comentario
multi
línea
"""
```

## Imprimir en pantalla

Para mostrar una variable en pantalla con el print escribimos f delante de las comillas y dentro ponemos entre llaves la variable a imprimir:

```
a = "1"
print (f"hola {a}")
```

## **Operaciones**

Para hacer el módulo utilizamos doble barram por ejemplo 6 en módulo 7:

```
a = 6//7
```

## **Condicionales**

En python tenemos el condicional if que funciona de forma similar a otros lenguajes de programación. OJO: Aquí no hay llaves se hace todo con tabulaciones.

```
if (variable == true):
    print("Hola")
else:
    print("Adios")
```

### **Bucle For**

```
#ejemplo de uso del bucle for para recorrer un array:
Array = [1,2,3,4]
```

http://knoppia.net/ Printed on 2025/11/30 17:24

2025/11/30 17:24 3/4 Introducción a python

```
for x in Array:
  print(x)
```

#### **Bucle While**

```
y = true
while(y):
    print(1)
    y=False
```

#### **Funciones**

```
def funcion(parametroA, parametroB):
    a = "salida"
    print("Contenido")
    return a
```

Si quisiéramos devolver múltiples valores los separaríamos con comillas:

```
def funcion():
    return 1,2
```

# **Diccionarios en Python**

Se utilizan para almacenar datos.

```
dicc = dict{
    "clave_1": "valor_1"
    "clave_2": "valor_2"
}

print(dicc)#Mostramos contenido del diccionario

#Para recorrer las claves del diccionario:
for key, value in dicc.items():
    print(f"clave:{key}, valor: {value}")
```

```
From:
```

http://knoppia.net/ - Knoppia

Permanent link:

http://knoppia.net/doku.php?id=modelado:introduccion\_python&rev=1706776652

Last update: 2024/02/01 08:37



modelado:introduccion\_python http://knoppia.net/doku.php?id=modelado:introduccion\_python&rev=1706776652

http://knoppia.net/ Printed on 2025/11/30 17:24