

```
1 #!/usr/bin/env python
2 # coding: utf-8
3
4 # # Clase 2
5 # Variables y funciones
6
7 # Crear una variable
8
9 # TODO
10
11 # In[2]:
12
13
14 mi_primer_variable = "Hola Mundo"
15
16
17 # TODO
18
19 # In[4]:
20
21
22 Mi_primer_variable
23
24
25 # TODO
26
27 # In[5]:
28
29
30 for = "De nuevo hola"
31
32
33 # ### Números
34
35 # TODO
36
37 # In[7]:
38
39
```



Curso Básico de Python!

Arturo Téllez Cortés
Clase 2

29 de Abril del 2020

Contenido

- 1. Objetivo**
- 2. Creación de variables**
- 3. Clases de variables**
- 4. Funciones**
- 5. Ejercicio**
- 6. Dudas**

Objetivo

Objetivo de la clase

**Aprender a crear y
manipular variables
y funciones.**

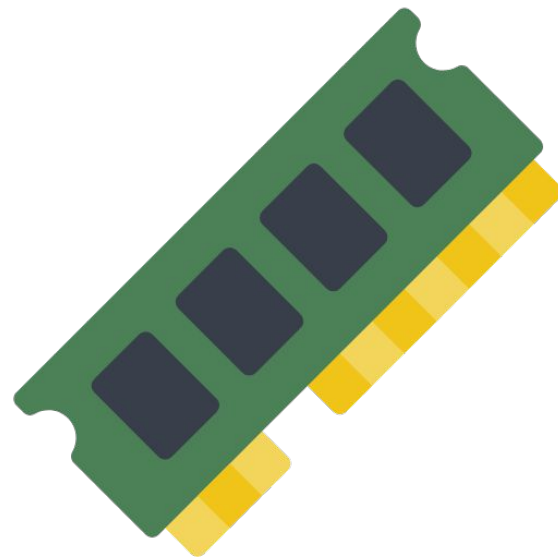
Creación de variables

¿Qué es una variable?

Es un sitio donde guardamos una determinada información, y se les clasificará dependiendo el tipo de información que almacenen.

Características de las variables:

- Case Sensitive
- No se pueden usar como nombres de variables palabras reservadas como: if, else, for
- No se pueden incluir en nombre de variables con tilde o con ñ
- No PUEDEN INICIAR CON un número



Clases de variables

Números

int

float

- Separador decimal es el punto

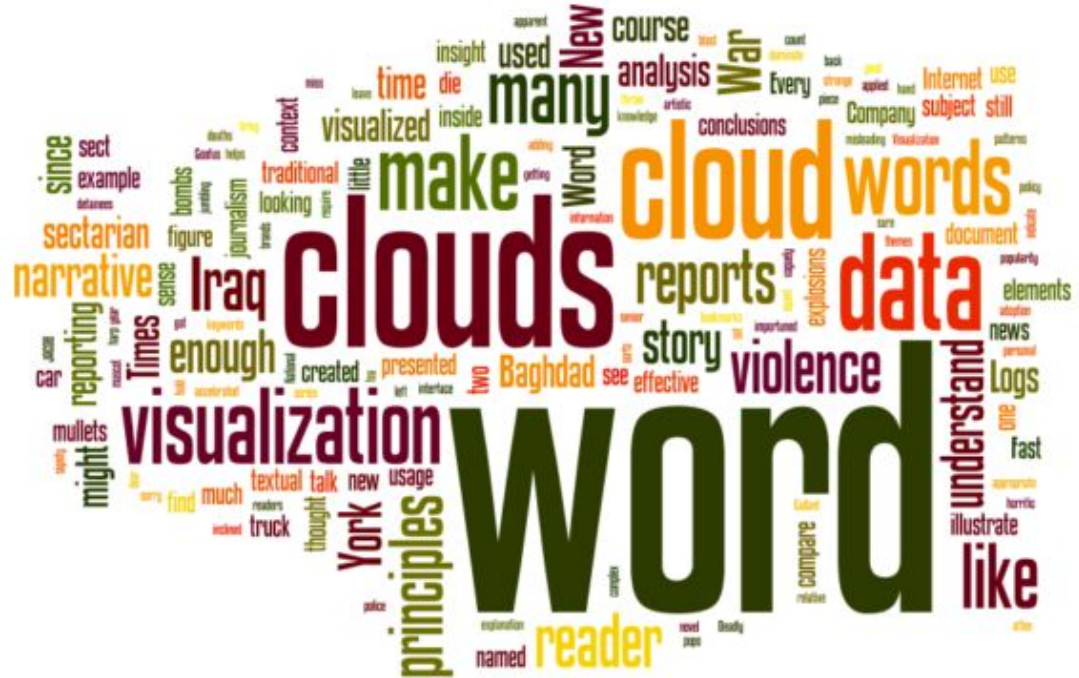


Manipular Números

- + suma
- - resta
- * multiplicación
- / división
- ** potencia
- // división entera
- % módulo
- pow
- round
- abs

Strings

- Delimitadas por comillas dobles o comillas simples.
- Para texto multilínea se usa ""



Manipular strings

- + concatenar texto
- * replicar texto
- [] acceder a un segmento del texto
Python empieza la selección de texto con 0
- len número de caracteres
- .find encontrar un texto
- .lower
- .upper
- .replace
- .split
- \ carácter de escape

Booleanos

- True
- False
- +, or o
- *, and y
- not negación

Oh!

- Existen valores que representan números complejos ($1 + 1j$)
- Puedo crear variables en una misma línea de código

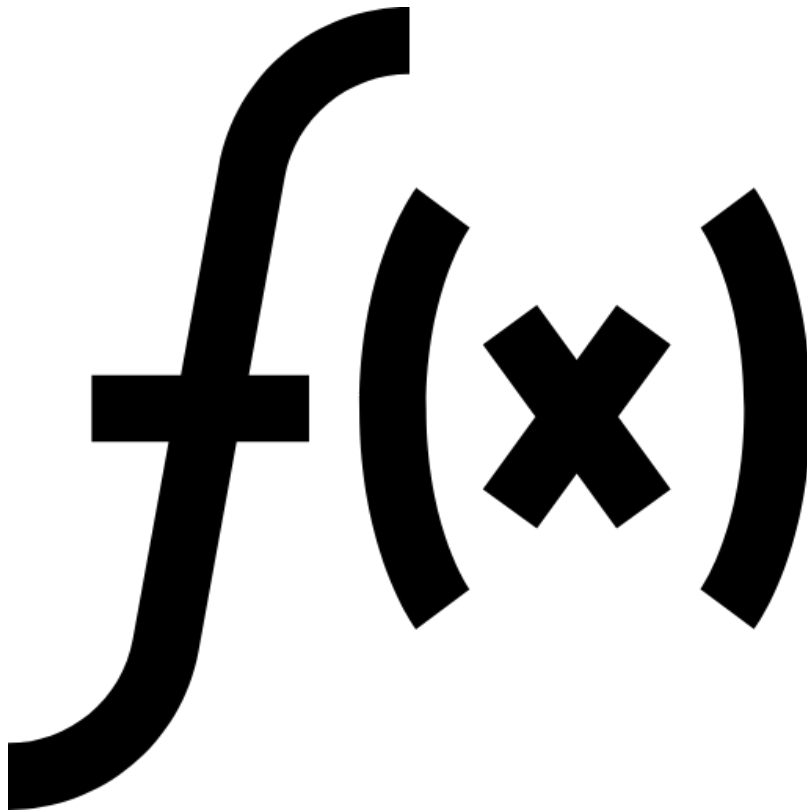
Funciones

Funciones

Se crean a partir de la palabra reservada **def**

```
def nombre_funcion(parametros):  
    código  
    return(salida)
```

Crear modular
Reutilizar código



Funciones

Argumentos

Posición

Nombre

Multi argumentos

Posición

Nombre

```
In [ ]: 1 def multiparametros_nombre(**nombres):  
        2     print(nombres)  
        3     multiparametros_nombre(animal = "perro", edad = 10, color = "12")
```

Funciones Lambda

Se crean a partir de la palabra reservada **lambda**

```
lambda x : x**2
```

