

1. Hola mundo en Python

Para eso usamos la función o comando print, para indicar a Python que debe imprimir una **cadena de caracteres(String)**, podemos usar `"""` (Comillas doble) o `' '` (Comillas simples), para representar una cadena.

```
# Imprimir hola mundo en python
print('Hola mundo :D')
```

2. Declaración de variables en Python

Variables

Una variable es un espacio de memoria (una caja) en el que podemos guardar y se recupera información (números, texto, etc). Las variables se nombran utilizando identificadores (nombre para las variables).

En Python se crea la variable asignándole un valor:

```
<identificador> = <valor>

# Ejemplo en consola python
>>> numero1 = 11
>>> numero2 = 7
>>> numero3 = numero1 + numero2
```

Reasignación de variables

Podemos en cualquier momento cambiar el valor de nuestra variable volviendo a asignar un valor al mismo identificador.

```
<identificador> = <nuevo_valor>
```

Reglas para identificadores en python

1. No pueden empezar con un número.
2. Deben estar en minúsculas.
3. Separamos las palabras usando guion bajo: `_`.

3. Poema en Python

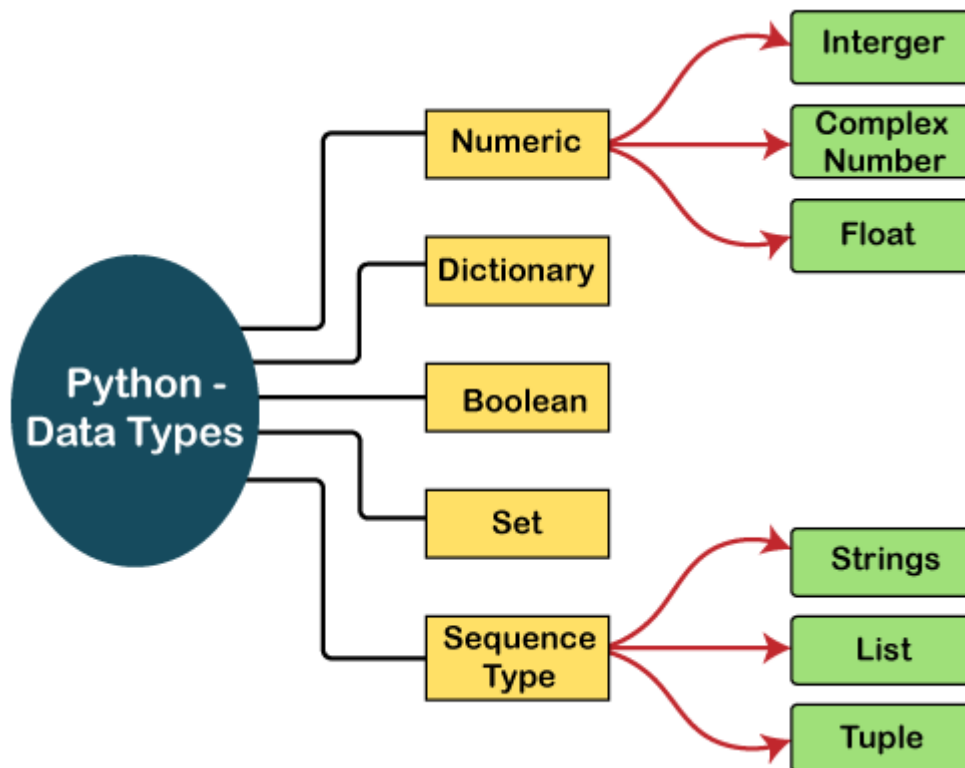
```
# Declaración de variables en python

frase1 = 'Tú dices que amas la lluvia,'
frase2 = 'sin embargo usas un paraguas'
frase3 = 'cuando llueve'


# Impresión de contenido de las variables
print(frase1, frase2, frase3)
```

4. Tipos de datos en Python


- Tipo texto (Cadena de caracteres): str
- Tipo numéricos: int, float, complex
- Secuenciales: list, tuple, range
- Mapeos: dict
- Conjuntos: set, frozenset
- Booleanos: bool
- Binary Types: bytes, bytearray, memoryview
- None Type: NoneType




5. Ejemplos de tipos de datos en python

```
  
# Tipos de variables en Python  
  
# int (entero)  
a = 28  
  
# float (decimales o reales)  
b = 1.5  
  
# str (string o cadena de texto)  
c = "Hello"  
c = 'Hello'  
  
# boolean (verdadero o falso)  
d = True  
  
# NoneType (Sin valor)  
e = None  
  
# str (5 y "5" no son lo mismo. La primera es un entero y la segunda una cadena de texto)  
f = "5"
```

6. Lectura de datos en python

```
  
# Lectura de datos en python  
nombre = input('Ingresa tu nombre: ')  
  
print(nombre)
```

7. Obtener el tipo de dato en una variable en Python

```
  
# Lectura de datos en python  
nombre = input('Ingresa tu nombre: ')  
  
print(nombre)  
print( type(nombre) )
```

8. Realiza un programa que sume dos números ingresados por el usuario.

```
# Sumar dos numeros en python con el operador aritmetico +

# Declaración de variables ( Entradas y Salida)

# Entradas
numero1 = None
numero2 = None

# Salida
resultado = None

# Lectura de datos
numero1 = float( input( 'Ingrese un número: ' ) )
numero2 = float( input( 'Ingrese otro número: ' ) )

# Procesamiento de datos
resultado = numero1 + numero2

# Impresión de resultados
print(resultado)
```

9. Operadores aritméticos en Python

- Suma: $1 + 2$
- Resta: $3 - 4$
- Multiplicación: $3 * 4$
- División (con decimales): $5 / 5$
- División (sin decimales): $21 // 5$
- Módulo: $25 \% 7$
- Potencia: $2 ** 2$

Alternativa para raíz cuadrada

```
import math

math.sqrt( 9 )
3.0

math.sqrt( 11.11 )
3.3331666624997918

math.sqrt( Decimal('6.25') )
2.5
```

Jerarquía de operadores aritméticos en python

1. Paréntesis ()
2. Exponentes o raíces
3. Multiplicaciones o divisiones
4. Sumas y restas
5. Asignacion =

10. Datos primitivos en python

Podemos encontrar cuatro tipos de datos que vienen definidos por defecto en Python, a estos tipos de datos los conocemos como primitivos. Tipos de datos primitivos en Python.

```
Integers: números Enteros
Floats: números de punto flotante (decimales)
Strings: cadena de caracteres (texto)
Boolean: booleanos (Verdadero o Falso)
```

Algunos operadores aritméticos pueden funcionar para operar con otros tipos de datos. Por ejemplo: podemos sumar strings, lo que concatena el texto o multiplicar un entero por un string, lo que repetirá el string las veces que indique el entero.

¿Cómo saber el tipo de dato que estoy usando?

Usando el comando type

```
## Resultado de consola
>>> x = 5
>>> type(x)
<class 'int'>
```

Ejecutar lo siguiente con consola

```
# Sumar dos textos(string)
>>> nombre1 = "Edgar Erik"
>>> nombre2 = "Andrés Urbano"
>>> nombre1 + nombre2
'Edgar ErikAndrés Urbano'

# Multiplicar un texto
>>> nombre1 * 4
'Edgar ErikEdgar ErikEdgar ErikEdgar Erik'

# Formateando texto
>>> nombre1 + ", " + nombre2
'Edgar Erik, Andrés Urbano'
```

Sume una variable que contenga un string y un numero y observe el resultado.

10 Realice un programa que pida al usuarios los datos A y B como reales, y realice:

- Suma
- Resta
- Multiplicación
- División (con decimales)
- División (sin decimales)
- Módulo
- Potencia

De estos.