



Curso de Python

Tipos de Datos Primitivos Simples

- Números (numbers)
 - Ejemplo: 0, -1, 3.1416
 - Se pueden usar operadores aritméticos y lógicos
- Cadenas
 - Ejemplo: 'Hola', "Python"
 - Accesos, subcadenas, formato y comparación
- Booleanos
 - True y False
 - Operaciones y tablas de verdad

Tipos de Datos Primitivos Compuestos (Contenedores)

- Lista (list)
 - Ejemplo: [1, "dos", [3,4]]
- Tupla (tuple)
 - Ejemplo: (1, 'dos', 3)
- Diccionario ({})
 - Ejemplo: {'hola':'hello', 'adiós':'goodbye'}

Diferencia entre Lista, Tupla y Diccionario

Listas	Tuplas	Diccionarios
Tienen orden	Tienen orden	No tienen orden
Pueden contener elementos de distintos tipos	Pueden contener elementos de distintos tipos	Pueden contener elementos de distintos tipos
Son mutables, es decir, pueden alterarse durante la ejecución del programa	Son inmutables, es decir, no pueden alterarse durante la ejecución de un programa	Son mutables, es decir, pueden alterarse durante la ejecución de un programa.
		Las claves son únicas, es decir, no pueden repetirse en un mismo diccionario, y pueden ser de cualquier tipo de datos inmutables
<code>list()</code>	<code>tuple()</code>	<code>{ }</code>

Estructuras de Control (1/3)

- Condicionales (**if**)

```
age = 20
```

```
if edad <= 18:
```

```
    print ( 'Menor' )
```

```
elif edad >65:
```

```
    print ( 'Jubilado' )
```

```
else:
```

```
    print ( 'Activo' )
```

Estructuras de Control (2/3)

- Bucles condicionales (**while**)

```
while True:
```

```
    num = int(input('Introduce un  
    número: '))
```

```
    if num == 0:
```

```
        break
```

Estructuras de Control (3/3)

- Bucles iterativos (**for**)

```
for i in range(1,10,2):  
    print(i, end=", ")
```

Archivos y Funciones

- Archivos y operaciones de abrir, modificar, leer y cerrar
- Funciones
 - Uso del keyword **def**
 - **Manejo de parámetros y retorno**
 - Ambito de los parámetros y variables de una función
 - Funciones recursivas

Programación Funcional

- Funciones anónimas **lambda**
- En colecciones iterables:
- Aplicar una función a una colección iterable (**map**)
- Filtrar los elementos (**filter**)
- Combinar los elementos (**zip**)
- Operar todos los elementos (**reduce**)

Librerías relevantes

- **Datetime** (tipos de datos **date**, **time** y **datetime**)
- **Numpy** (clase de objetos **array**)
- **Pandas**
 - **Series** (Estructura de una dimensión)
 - **Dataframe** (Estructura de dos dimensiones (tabla))
 - **Cubo** (Estructur de tres dimensiones)
- **Matplotlib** (creación de gráficos)