



PYTHON CHEATSHEET

OPERADORES

- aritméticos: `+`, `-`, `*`, `/`, `**`
- lógicos: `True`, `False`, `<`, `>`, `<=`, `>=`, `==`, `!=`, `and`, `or`, `&`, `|`, `not`

INSERCIÓN DINÁMICA

- `print('Soy {} y tengo {} años'.format(nombre, edad))`
- `print(f'Soy {nombre} y tengo {edad} años')`
- `print('Tengo %d años y %d hijos' %(41, 2))` donde `%d` para enteros, `%f` para reales, `%s` para texto

TIPOS DE DATOS

- individuales: *int*, *float*, *str*, *None*
- conversión: *int()*, *float()*, *str()*
- Secuencias: *range*, *list*, *tuple*, *dict*, *set*
- Listas:
 - crear lista: `[]` o `list()`
 - añadir elemento-s: `.append()` para uno, o `.extend()` para varios
 - eliminar por índice: `del lista[2]`
 - eliminar por nombre: `lista.remove('nombre')`
 - ¿elemento en lista?: `elemento in lista`
 - número de elementos: `len(lista)`
 - ordenar lista: `lista.sort(key = str.lower)`
- Tuplas:
 - crear tupla: `()` o `tuple(lista)`
- Diccionarios:
 - crear diccionario: `{}` o `dict(lista de tuplas)`
 - modificar elemento: `diccionario['clave'] = valor`
 - eliminar elemento: `del diccionario['clave']`
- Conjuntos:
 - crear conjunto: `set(lista)`
 - añadir elemento-s: `.add()` para uno, o `.update()` para varios
 - eliminar elemento por valor: `.discard('elemento')`
 - unión conjuntos: `conj1.unión(conj2)`
 - intersección conjuntos: `conj1.intersection(conj2)`
 - diferencia conjuntos: `conj1.difference(conj2)`



PYTHON CHEATSHEET

INDEXACIÓN

- cadenas: *cadena[0], cadena[0:3], cadena[-1]*
- listas: *lista[0], lista[0:3], lista[-1]*
- tuplas: *tupla [0], tupla[0:3], tupla[-1]*
- diccionarios: *dicc['clave']*
- conjuntos: *no se pueden indexar*

CONTROL DE FLUJO

- if $x > y$:
 acción
- elif $y > x$:
 acción
- else:
 acción
- for cada in range(5):
 acción
- Iterar diccionarios:
 - .items(): *devuelve clave y valor*
 - .keys(): *devuelve clave*
 - .values(): *devuelve valor*
 - enumerate(diccionario): *devuelve posición y clave*
- List comprehension: *[haz esto para cada in iterable if condición]*

FUNCIONES

- definir función: *def nombre(par1,par2):*
 cuerpo función
 return(resultado)
- funciones Lambda: *lambda x: x * 2*
- map: *aplica una función a un iterable: map(función,iterable)*
- input: *captura datos del usuario: input('Cómo te llamas')*
- zip: *une dos secuencias elemento a elemento: zip(lista1,lista2)*

IMPORTACIÓN DE MÓDULOS

- import modulo as alias
- from modulo import *
- from modulo import funcion