# Programar desde cero

Fundamentos de programación con python y/o javascript



# Definiciones de andar por casa

- Datos unitarios o elementales: los más básicos de un lenguaje de programación
- Datos complejos: formados por composición de los elementales

# Tipos de datos en python

- Elementales
  - Números
    - int, float, complex
  - Cadenas
  - Booleanos

# Tipos de datos en python

- Complejos
  - Secuencias ordenadas
    - tupla (inmutable) → tuple
    - lista (mutable)  $\rightarrow$  list
  - Secuencias desordenadas
    - conjuntos → set → sólo elementos únicos (documentación)
    - diccionarios → dict → elementos de tipo clave: valor (documentación)

### **Tuplas**

- Secuencias ordenadas inmutables
  - Creación
    - (valor, ...)
    - tuple(iterable)
    - return 2 o más elementos separados por comas
  - Acceso
    - Igual que las cadenas [ini : fin+1 : salto ]

# **Tuplas**

- Secuencias ordenadas inmutables
  - Iteración
    - for valor in tupla:
  - Comprobar pertenencia
    - ini in tupla → bool

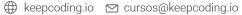
#### Listas

- Secuencias ordenadas Mutables
  - Creación
    - [valor...]
    - list(valor)
  - Acceso
    - lgual que las cadenas [ini : fin+1 : salto ]

### Listas

- Secuencias ordenadas mutables
  - Iteración
    - for valor in lista:
  - Comprobar pertenencia
    - ini in lista → bool





#### Listas

- Secuencias ordenadas mutables
  - Algunos métodos interesantes
    - lista.append(v) → añade valor al final de la lista
    - lista.clear() → vacía la lista
    - lista.extend(otra\_lista) → añade otra\_lista al final
    - lista.count(v)  $\rightarrow$  cuenta el número de veces que aparece v
    - lista.index(v) → posición de v en la lista (ini: 0) o Error
    - lista.insert(p, v)  $\rightarrow$  inserta v en posición p (ini: 0)
    - lista.pop() → saca último item de la lista
    - lista.pop(p)  $\rightarrow$  saca item en posicion p de la lista
    - lista.remove(v) → saca primer item con valor v de la lista

### **Diccionarios**

- Conjuntos mutables de pares clave: valor. Clave única
  - Creación
    - { k: v...}
    - dict(k=v...)
  - Acceso por clave
    - diccionario[k]
    - diccionario.get(k, default?)

### **Diccionarios**

- Conjuntos mutables de pares clave: valor. Clave unica
  - Iteración  $\bigcirc$ 
    - for clave in diccionario:
    - for valor in diccionario.values():
    - for clave, valor in diccionario.items()
  - Comprobar pertenencia
    - clave in diccionario → bool

### **Diccionarios**

- Conjuntos mutables de pares clave: valor. Clave unica
  - Algunos métodos interesantes
    - diccionario.pop(k, msg?) → elimina par con clave k
    - diccionario.clear() → vacía el diccionario

# **Conjuntos (sets)**

- Conjuntos mutables de valores únicos
  - Creación
    - {v...}
    - set(iterable)
  - Acceso
    - Al no estar ordenado no se puede acceder a un elemento sólo comprobar si existe

# **Conjuntos (sets)**

- Conjuntos mutables de valores únicos
  - Iteración
    - for item in conjunto:
  - Comprobar pertenencia
    - item in conjunto → bool

### **Conjuntos (sets)**

- Conjuntos mutables de valores únicos
  - Algunos métodos interesantes
    - conjunto.pop() → elimina item al azar
    - $\blacksquare$  conjunto.remove(v)  $\rightarrow$  elimina v del conjunto o Error
    - conjunto.discard(v) → elimina v del conjunto o nada
    - conjunto.clear() → vacía el conjunt
    - conjunto.add(v)  $\rightarrow$  añade v al conjunto (si no existe ya)
    - lacktriangleright conjunto.update(iterable) ightarrow añade los valores inexistentes de iterable al conjunto



Madrid | Barcelona | Bogotá

Datos de contacto:

cursos@keepcoding.io