

DICCIONARIOS

Estructura de datos que almacena elementos en pares clave : valor (key : value).

Características

- La **clave** es **única e inmutable**: booleano, strings int, float,tuplas
- El **valor** puede ser de **cualquier tipo**: números, cadenas, listas, tuplas y otros diccionarios
- Son **mutables**, se puede agregar, modificar y eliminar elementos

Se pueden **crear**:

- Usando **{}** y clave : valor separados por dos puntos

```
frutas = {"manzana": 0.99, "plátano": 0.75, "naranja": 0.60}
```



```
diccionario4 = { 'lunes': [1, "peras"], 'martes':[2, "manzanas"], 'miércoles':[3, "brocoli"] }
```
- **dict()**
 - diccionario1 = dict(jueves = "thursday", viernes = Friday")
 - diccionario6 = dict([('sábado','saturday'), ('domingo','sunday')]) →
{'sábado': 'saturday', 'domingo': 'sunday'}

Se puede crear un diccionario de una **tupla o lista de tuplas** dict(**tuple**)

- **.fromkeys()** para crear un diccionario con claves y valores **predeterminados**
 - claves = ['jueves', 'viernes', 'sabado'] valor_predeterminado = []
diccionario7 = dict.fromkeys(claves, valor_predeterminado) → {'jueves': [], 'viernes': [], 'sabado': []}

Propiedades:

- **len()**: **cantidad** de elementos del diccionario, **clave:valor** es un elemento.
Se puede conocer **solo las keys o values**: len(diccionario.keys())
- **.keys()**: devuelve una **lista con las claves del diccionario**
- **.values()**: devuelve una **lista con los valores del diccionario**
- **.items()**: devuelve una **lista de tuplas clave-valor**

- **in, not in:** puede ser in diccionario y lo **busca** sobre las **claves** → ‘clave’ in diccionario. Para buscar en los **valores** in diccionario.values()

Métodos:

- **.copy(): copia** el diccionario en **uno nuevo**
- **.clear(): elimina** todos los elementos
- **.update(): actualiza** el diccionario **con nuevas claves-valor, con otro diccionario o una clave o valor.** Sobreescribe.
diccionario1.update(diccnuevo) O diccionario1.update(‘clave’ : ‘valor’) O diccionario.update({‘clavepredeterminada’ : ‘nuevovalor’})
- **.get(): devuelve el valor de una clave** → diccionario.get(clave) es igual que diccionario[clave]. Si **no existe** devuelve **None** y se puede añadir después del valor una , ‘una frase’ → diccionario.get(clave, ‘una frase’). **No sobreescribe.**
- **.setdefault():** igual que get, pero **si no existe sobreescribe** el diccionario insertando una con **valor None** o uno **‘predeterminado’**.
diccionario.get(clave, ‘una frase’)
- **sorted(): ordena alfabéticamente** los elementos por las **claves**: sorted(diccionario1) → la claves de diccionario1 ordenadas alfabéticamente. También se pueden **ordenar los valores** solo si son del **mismo type**: sorted(diccionario1.values()) o sorted(diccionario1.values(), reverse=True)
- **.pop(): elimina y devuelve el valor de una clave especificada:** diccionario.pop(‘clave’)→devuelve el valor. **Sobreescribe** el diccionario, si la clave no existe genera un error.
- **.popitem(): elimina y devuelve el valor de la última clave añadida.** diccionario.popitem(‘clave’)→devuelve el valor. **Sobreescribe** el diccionario.

Notas

1. No se puede acceder por índice. Para **acceder a clave** → diccionario[‘clave’]

2. Para **acceder al valor** o valores de una clave: diccionario['clave'][posicióndelvalor/es] → valor/es
3. Cuando es un **diccionario con diccionarios** dentro y queremos **acceder a un valor** concreto. → Diccionario1[clave1][indicedentroclave1]

Para **saber si hay una clave** en el diccionario: diccionario['clave']

Para **modificar el valor** de una clave, o cambiar de string a lista: diccionario['clave'] = 'nuevovalor' , o []

Para **añadir valores** a una clave que es una lista podemos usar el método .append() de las listas: diccionario[clave].append('nuevovalor').

Cuando hay una lista se puede usar las mismas funciones que en listas.

Ej1: Diccionario[valorqueeslista].append(9) añade el valor 9 dentro de la key lista Ana

Ej: Diccionario[valorqueeslista][nºindice] accede al dato que queremos dentro de la lista que son valores.

Input:

Variable = input('qué mascota quieres eliminar') ENTER, EN LA BARRA ESCRIBIR mascota2

Diccionario_animales[opcion] DEVUELVE el valor de mascota2 que es perro

SETS

elementos únicos y mutables y vienen definidos por `{}'