

6 - Diccionarios



Horas de Libre Configuración
Curso 2023 - 2024

¿Qué son?



- ▶ Es un tipo de datos secuencia y mutable.
- ▶ Básicamente es una tabla hash.
- ▶ Mantiene un conjunto de pares clave, valor.
 - ▶ *Clave* debe ser de tipo inmutable (número, string o tupla).
 - ▶ *Valor* puede ser de cualquier tipo

Creación de diccionarios

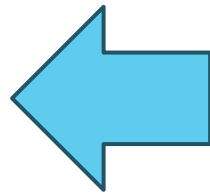


```
vacio = {}  
precios = {  
    "Pera":1.60,  
    "Ciruela": 1.24,  
    "Manzana": 1.75  
}
```

Creación de diccionarios

- También se puede usar la función `dict()`.

```
vacio = dict()  
precios = dict(  
    Pera=1.60,  
    Ciruela=1.24,  
    Manzana=1.75)
```



¡OJO!
Las claves deben
ser identificadores
válidos

Operaciones



- ▶ Añadir o modificar un elemento:

```
precios["Sandía"] = 0.99
```

- ▶ Comprobar si una clave existe en el diccionario:

```
if "Pera" in precios:
```

Operaciones



► Consultar un valor:

```
pSandia = precios["Sandía"]
```

Se produce un error si la clave no se encuentra en el diccionario.

Operaciones



► Consultar un valor sin error:

```
if "Sandía" in precios:
```

```
    pSandia = precios["Sandía"]
```

```
pSandia = precios.get("Sandía")
```

```
pMelon = precios.get("Melón", -1)
```

Operaciones



► **Borrar un elemento:**
`del precios["Sandía"]`

Se produce un error si la clave no se encuentra en el diccionario.

Operaciones



- Obtener un elemento y eliminarlo:

```
pSandia = precios.pop("Sandía")
```

Se produce un error si la clave no se encuentra en el diccionario. Para evitar el error:

```
pMelon = precios.pop("Melón", -1)
```

Operaciones



- ▶ Vaciar el diccionario:

```
precios.clear()
```

- ▶ Longitud del diccionario:

```
len(precios)
```

Elementos del diccionario

- ▶ `keys()` devuelve la lista de claves.
- ▶ `values()` devuelve la lista de valores.
- ▶ `items()` devuelve la lista de elementos.

```
for c in precios.keys():  
    print(c)
```