



Business
School

Iniciación a Python. Colecciones Diccionarios

Marta Ramírez

Especialidad en Data science y Big data

- Definición.
- Métodos.
- Funciones.
- Estructura for para diccionarios.
- Ordenación de diccionarios.

I. Colecciones diccionarios

Colecciones

Diccionarios

Definición

Son colecciones que relacionan una clave y un valor.

Por ejemplo, veamos un diccionario de asignaturas y profesores:

```
Clase={"SGE":"Valentina","AD":"Marta","PDM":"Paloma"}
```

Para crear un diccionario vacío

```
Clase=dict{}
```

I. Colecciones diccionarios

Colecciones

Diccionarios

Definición

La diferencia principal entre los diccionarios y las listas es que a los valores almacenados en un diccionario se les accede no por su índice, porque de hecho no tienen orden, sino por su clave, utilizando de nuevo el operador [].

`Print(Clase["SGE"])` nos devuelve el valor que tiene esa clave.

I. Colecciones diccionarios

Colecciones

Diccionarios

Métodos

- #método que devuelve si una clave existe en un diccionario
clave in diccionario
 - # añadir elementos al diccionario
clase["clave nueva"] = "valor nuevo"
 - #borrar elementos
del diccionario ["clave"]
 - #borrar diccionario entero
del diccionario
- También podemos usar:
diccionario.clear()

I. Colecciones diccionarios

Colecciones

Diccionarios

For para diccionarios

```
for clave in mi_diccionario:
```

```
    print (clave, ": ", mi_diccionario[clave])
```

Usando el método `items()`, obtenemos una lista de tuplas (clave, valor), que podemos usar en el for:

```
for (clave, valor) in mi_diccionario.items():
```

```
    print (clave, ": ", valor)
```

Usando `enumerate` se puede obtener el índice de posición junto a su clave correspondiente:

```
for i, v in enumerate(diccionario):
```

```
    print (i, v)
```

I. Colecciones diccionarios

Colecciones

Diccionarios

Ordenación diccionarios

Por claves

Los diccionarios tienen un método de ordenación sobre claves `SORTED()`: Nos devuelve una lista de las claves ordenadas:

```
sorted(diccionario)
sorted (diccionario, reverse=True)
print (clave, ": ", mi_diccionario[clave])
```

Por sus valores

```
from operator import itemgetter
compra = {'leche':2.19, 'pan':3.09, 'cafe':4.59, 'huevos':1.79, 'quesos':4.99}
print(sorted(compra.items(), key=itemgetter(1)))
```

Ordena por el valor, ya que en `itemgetter(1)` miramos la posición 1 de las minilistas.

Conclusiones

Se han visto los conceptos más relevantes de las estructuras repetitivas en Python.

Para avanzar en la material **se deben realizar los casos prácticos**, tanto los que se han resuelto en clase como los que se os dejan pendientes para su realización.