

20 MARZO 2025

Lezione 5



Indice



01 Dizionari

02 Funzioni

03 Import

1. Dizionari

In italiano **dizionari**.

I dizionari contengono elementi organizzati in coppie **chiave:valore**.

Caratteristiche:

- **Ordinati**: a partire da Python 3.7, i dizionari sono ordinati, il che significa che mantengono l'ordine in cui gli elementi sono stati aggiunti.
- Nelle versioni precedenti (Python 3.6 e precedenti), i dizionari erano non ordinati.
- **Modificabili**: i dizionari sono modificabili, quindi puoi cambiare, aggiungere o rimuovere elementi dopo averli creati.
- **Duplicati non consentiti**: non è possibile avere due elementi con la stessa chiave in un dizionario, ma i valori possono essere duplicati.
- **Tipi di dato**: i valori possono essere di tutti i tipi di dato che abbiamo visto, possono anche contenere altri dizionari.

```
film = {  
    "titolo": "Bastardi senza gloria",  
    "regista": "Quentin Tarantino",  
    "anno": 2009,  
    "genere": ["Guerra", "Drammatico", "Azione"],  
}
```

Metodi

Accesso

```
# Riferendosi al nome della chiave  
film['titolo']
```

'Bastardi senza gloria'

```
# Utilizzando il metodo get()  
film.get('titolo')
```

NOTARE: Il metodo **get()** è importante e molto utile perché permette di specificare un valore di ritorno se la chiave specificata non esiste.

```
dictionary.get(keyname, value)
```

Aggiungere/modificare elementi

```
# Utilizzando una chiave  
macchina['targa'] = "FE587GB"
```

```
# Utilizzando il metodo update()  
macchina.update({"colore": "nero"})
```

In entrambi i casi, se la chiave in questione non è presente nel dizionario verrà aggiunto l'elemento, altrimenti verrà modificato

Metodi

Rimuovere elementi

Metodo	Descrizione
pop()	Rimuove l'elemento con la chiave passata come parametro, restituisce errore se non esiste la chiave
popitem()	Rimuove l'elemento inserito per ultimo
clear()	Rimuove tutti gli elementi

Miscellanea

Metodo	Descrizione
items()	Restituisce una lista contenente tutti gli elementi (chiave:valore) del dizionario
keys()	Restituisce una lista contenente tutte le chiavi del dizionario
values()	Restituisce una lista contenente tutti i valori all'interno del dizionario

Funzione len(): esiste sempre la funzione **len()**, che non è un metodo; permette di stampare il numero di elementi nel dizionario.

2. Funzioni

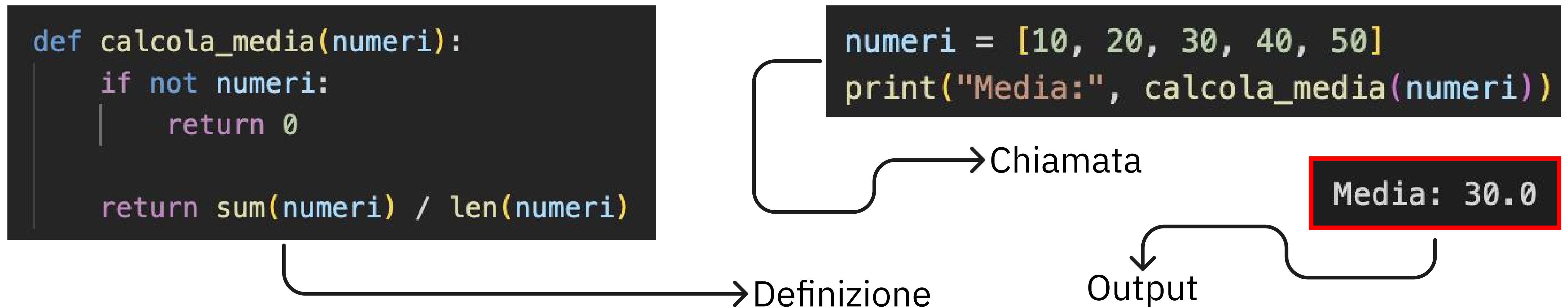
Quando si programma due cose sono fondamentali:

1. Non riscrivere il codice che hanno già scritto altri
2. Non duplicare il codice che hai già scritto tu

Le **funzioni** ci aiutano a rispettare questi due concetti.

Infatti permettono di definire blocchi di codice che viene eseguito solamente quando viene chiamato.

NOTARE: le **funzioni** possono avere o NON avere un valore di ritorno



3. Import

Per evitare di scrivere codice già implementato da altri ci viene in aiuto l'**import**.

Con **import** possiamo importare nel nostro foglio di lavoro / progetto dei **moduli** o anche solamente delle **funzioni** già create da altri programmatori.

Modulo	Descrizione
math	Funzioni matematiche avanzate, come radici e fattoriali.
random	Generazione di numeri casuali o scelta casuale da una lista.
datetime	Lavorare con date e orari.
time	Funzioni per gestire il tempo e creare pause.
os	Interagire con file, cartelle e sistema operativo.
sys	Informazioni su Python e il sistema.

```
def fattoriale(numero):  
    fattoriale = 1  
    if numero < 0:  
        return 0  
    elif numero == 0:  
        return 1  
    else:  
        for i in range(1, numero + 1):  
            fattoriale = fattoriale * i  
        return fattoriale  
fattoriale(5)
```

→ Invece di scrivere questo

120

```
from math import factorial  
factorial(5)
```

→ Si può importare direttamente la funzione