



Python in pratica III

Pascal Brunot

24/01/2026

FabLab Bergamo

Regole del corso



ORARI

15,30 – 16,30
pausa 15 minuti
16.45 – 18.00

Le domande sono molto gradite, a qualunque momento. Alzate la mano se non vi noto.

Usate il PC e fare gli esercizi/quizz man mano, la pratica è fondamentale per imparare.

Piano Lezione 3

- ❖ Review lezione 2
 - ❖ Liste, dizionari, for
 - ❖ Esercizio calcolo medie
- ❖ Librerie Python: Pillow, Pandas, Matplotlib
 - ❖ Manipolare immagini
 - ❖ Lavorare su tabelle di dati con Panas
 - ❖ Grafici con Matplotlib

Usare librerie Python

Trovare la libreria giusta

Chiedere all'AI

«proponimi una libreria Python semplice per fare <....> con un esempio di installazione e uso per farmi capire le funzioni chiave

Alcune sono molto molto diffuse e li vedremo nel corso : pandas, numpy, PIL,....

1

Installare

`!pip install <nome_libreria>`

`!pip install pandas`

PIP è un gestore di pacchetti che scarica la libreria su Internet e la installa per Python.

2

Importare nel programma

```
import pandas  
import numpy as np  
  
# Import parziali  
from PIL import Image  
import matplotlib.pyplot as plt
```

3

Liste

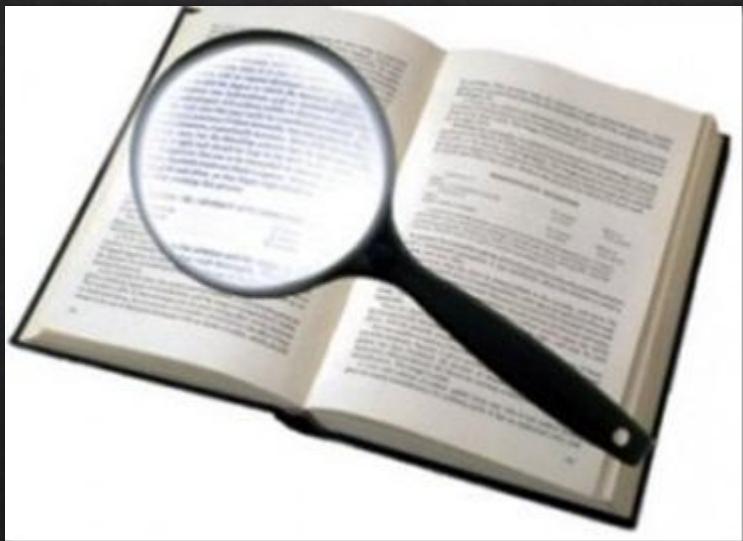
- ❖ Parentesi quadre
- ❖ Accesso agli elementi con indice intero
l[1] , l[-1]

```
# Lo zaino del videogiocatore
zaino = ["spada", "pozione", "mappa",
"torcia"]
print(zaino[0]) # spada - il primo oggetto!
```



Dizionari

- ❖ Parentesi graffe { } (ALTGR SHIFT [])
- ❖ Accesso agli elementi con la loro **chiave**



```
# Rubrica  
contatti = {  
    "pizzeria": "035-123456",  
    "mamma": "333-1234567",  
    "meccanico": "349-7654321"  
}  
  
# Estraiamo il contatto di emergenza  
print(contatti["pizzeria"]) # Emergenza pizza!
```

A large yellow arrow points from the word "chiave" in the list of bullet points to the colon character ":" in the code example, highlighting the correspondence between the concept and its implementation.

Dizionari

- ❖ Un dizionario è un insieme di **valori** indicizzati da **chiavi** univoche

```
dizionario = {  
    chiave1: valore1,  
    chiave2: valore2,  
    ...  
}
```

- ❖ Non posso avere chiavi duplicate nel dizionario.

- ❖ Cancellare un'entrata del dizionario
`del(contatti["pizzeria"])`

- ❖ Aggiungere un elemento
`contatti["papa"] = "3311234564"`

- ❖ Enumerare le chiavi del dizionario:
`for nome in contatti:
 print(f"{nome} = {contatti[nome]}")`

- ❖ Enumerare entrate e valori
`for nome, telefono in contatti.items():
 print(f"{nome} : {telefono}")`



Voti scolastici

- ❖ Creare una variabile voti che contenga elenchi di voti, organizzati per nome della materia.
- ❖ Tipi utilizzabili ?
- ❖ Elenco di voti :
 - Matematica 8**
 - Italiano 10**
 - Matematica 8.5**
 - Geografia 7**
 - Matematica 9**
 - Italiano 8**
 - Italiano 8.5**
- ❖ Come posso organizzare questi dati in una variabile python ?

L'esercizio

- ❖ Calcolare la media per materia_

Matematica 8,5

Italiano 8,83

Geografia 7

- ❖ Calcolare la media complessiva dello studente con due decimali

8,43

Definizione della media aritmetica

$$\text{Media} = \frac{\text{somma dei voti}}{\text{numero di voti}}$$

Non ci sono coefficienti per materia.

Esercizio 2

- ❖ Aggiungere una materia Latino con i voti
8,9,10
- ❖ Ricalcolare la media complessiva.

Manipolare immagine

- ❖ Scaricare un'immagine dal web con Request
- ❖ Ridimensionare
- ❖ Ruotare
- ❖ Convertire in livelli di grigio
- ❖ Libreria PILLOW

Manipolare dati

- ❖ Caricare un file di dati in COLAB
- ❖ Libreria **PANDAS**
- ❖ Dati tabellari («dataframes»)

Titolo 1	Titolo 2		
df.loc[0] → Valore 1	df["Titolo 1"][0]	Valore 4	df["Titolo 2"][0]
df.loc[1] → Valore 2	df["Titolo 1"][1]	Valore 5	df["Titolo 2"][1]
df.loc[2] → Valore 3	df["Titolo 1"][2]	Valore 6	df["Titolo 2"][2]

Colonna 1
df["Titolo 1"]

Colonna 2
df["Titolo 2"]

Dataframe
df

```
graph LR; A["df.loc[0] → Valore 1"] --- B["df['Titolo 1'][0]"]; A --- C["Valore 4"]; A --- D["df['Titolo 2'][0]"]; B --- C; B --- D; E["df.loc[1] → Valore 2"] --- F["df['Titolo 1'][1]"]; E --- G["Valore 5"]; E --- H["df['Titolo 2'][1]"]; F --- G; F --- H; I["df.loc[2] → Valore 3"] --- J["df['Titolo 1'][2]"]; I --- K["Valore 6"]; I --- L["df['Titolo 2'][2]"]; J --- K; J --- L; M["Colonna 1  
df['Titolo 1']"]; N["Colonna 2  
df['Titolo 2']"]; O["Dataframe  
df"]
```

Pandas – caricare dati

- ❖ File voti_scolastici.csv

https://www.fablabbergamo.it/wp-content/uploads/2026/01/voti_scolastici.csv

1	Nome	Cognome	Classe	Materia	Voto	Data
2	Laura	Bianchi	3A	Geografia	5	25/09/2025
3	Laura	Bianchi	3A	Geografia	7	11/10/2025
4	Laura	Bianchi	3A	Geografia	8	28/10/2025
5	Laura	Bianchi	3A	Geografia	5	28/11/2025
6	Laura	Bianchi	3A	Inglese	6	05/09/2025
7	Laura	Bianchi	3A	Inglese	6	15/09/2025
8	Laura	Bianchi	3A	Inglese	6	01/11/2025
9	Laura	Bianchi	3A	Inglese	5	14/11/2025
10	Laura	Bianchi	3A	Italiano	6	27/09/2025
11	Laura	Bianchi	3A	Italiano	5	28/09/2025
12	Laura	Bianchi	3A	Italiano	4	01/10/2025

```
import pandas as pd
```

```
# Leggi il file CSV
```

```
df_voti = pd.read_csv('voti_scolastici.csv')
```

*Come posso avere la colonna di tutti i cognomi ?
Come posso avere la colonna con tutti i voti ?*



Pandas

- ❖ Esplorare i dati caricati (head())
- ❖ Tabella pivot (Excel) – medie per materia raggruppate per studente

```
[24] 0s
▶ # Prime righe (come stampare le prime chiavi di un dizionario)
print("■ PRIME 5 RIGHE DEL REGISTRO")
print("*60)
print(df_voti.head())
...
■ PRIME 5 RIGHE DEL REGISTRO
=====
    Nome Cognome Classe Materia Voto      Data
0  Laura  Bianchi    3A Geografia   5 2025-09-25
1  Laura  Bianchi    3A Geografia   7 2025-10-11
2  Laura  Bianchi    3A Geografia   8 2025-10-28
3  Laura  Bianchi    3A Geografia   5 2025-11-28
4  Laura  Bianchi    3A Inglese     6 2025-09-05
```

Media di Voto	Etichette di colonna	Inglese	Italiano	Matematica	Scienze Storia
Etichette di riga	Geografia				
■Bianchi	Laura	6.25	5.75	6.20	6.00 6.67 6.20
■Bruno	Alessandro	6.25	6.00	5.25	7.40 5.67 6.20
■Colombo	Luca	8.25	6.33	6.50	7.25 6.33 6.75
■Conti	Davide	8.25	7.25	8.00	6.33 7.60 8.75
■Costa	Andrea	7.33	8.75	7.75	8.00 7.80 8.60
■De Luca	Elena	4.50	6.67	5.20	6.50 7.00 6.67
■Ferrari	Sofia	7.67	7.33	8.00	8.00 7.20 5.40
■Gallo	Francesca	7.00	6.33	9.20	8.20 7.60 8.25
■Greco	Giulia	8.50	9.33	7.33	7.20 6.33 8.00
■Marino	Matteo	6.33	7.67	8.75	7.50 8.67 6.25
■Neri	Anna	7.25	5.33	6.00	5.60 6.20 6.00
■Ricci	Chiara	6.40	6.40	4.00	7.00 6.00 6.33
■Rossi	Marco	7.50	5.40	6.50	6.60 6.00 6.00
■Russo	Roberto	7.00	5.00	6.00	6.80 5.50 5.60
■Verdi	Giovanni	8.00	7.25	6.80	7.75 8.00 6.67

- ❖ Qual'è lo studente con la media più alta?

Definire le liste con proprietà (list comprehensions)

- ❖ Voglio tutte le frutta che iniziano con a

```
frutti1 = []
for frutto in frutta:
    if frutto[0] == 'a':
        frutti1.append(frutto)
```

```
[frutto for frutto in frutti if frutto[0] == 'a']
```

- ❖ Voglio la tabellina del 3

```
[3*n for n in range(1,11)]
```

Sintassi

```
[<elemento> for <elemento> in  
<sorgente> if <condizione>]
```

Interi divisibili da 7 fra 0 e 99 :

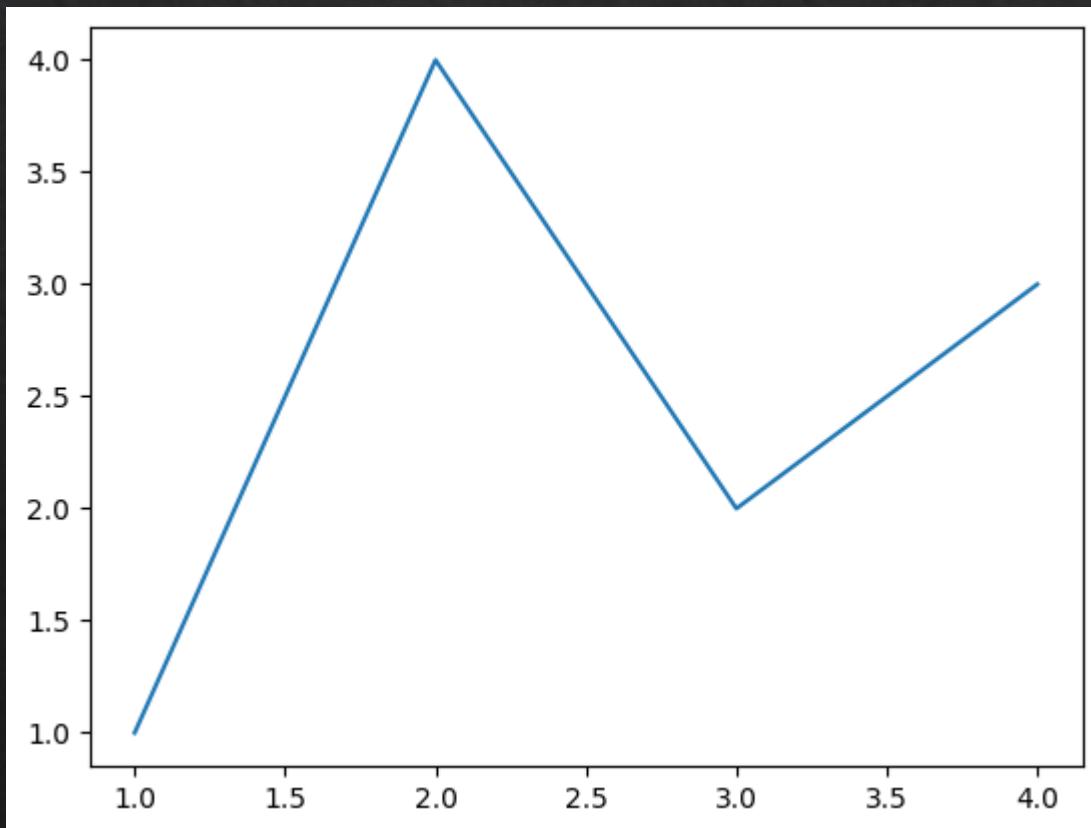
```
[i for i in range(0,100) if i % 7 ==  
0]
```

Matplotlib

- ❖ **Libreria Python** per la creazione di grafici e visualizzazioni
 - ❖ **Basata su NumPy** per l'elaborazione dati numerici
 - ❖ **MATLAB-inspired** - sintassi facile e intuitiva
 - ❖ **Versatile** - dal prototipo rapido alle pubblicazioni professionali
- ❖ **Installazione:**
`!pip install matplotlib`
 - ❖ **Nel vostro script:**
`import matplotlib.pyplot as plt`

Il primo grafico

- ❖ Eseguire



- ❖ **Codice base:**

```
x = [1, 2, 3, 4]  
y = [1, 4, 2, 3]  
plt.plot(x, y)  
plt.show()
```

- ❖ **Risultato:** un grafico a linee con i vostri dati

Personalizzazione grafico

❖ Personalizzazione Essenziale

`plt.title()` - Titolo del grafico

`plt.xlabel()` e `plt.ylabel()` - Etichette assi

`plt.legend()` - Legenda delle serie

`plt.grid()` - Griglia di background

