

Introduzione a Python

Cos'è Python?

Python è un linguaggio di programmazione creato da Guido van Rossum nel 1991. È tipo il coltellino svizzero della programmazione: fa tutto, lo capiscono tutti, e non ti fa venire il mal di testa.

Perché è così popolare? Perché:

- **Sembra quasi inglese:** la sintassi è talmente chiara che sembra di leggere delle istruzioni, non del codice alieno
- **Fa TUTTO:** web, intelligenza artificiale, analisi dati, automazione, giochi... praticamente tutto tranne il caffè (ma ci stanno lavorando)
- **Community enorme:** qualsiasi errore tu faccia, qualcuno l'ha già fatto prima di te e ha scritto la soluzione su Stack Overflow 🤖

Ricordi Flowgorithm? Ecco, Python è il **livello successivo**. La logica è identica (input → elaborazione → output), cambia solo il modo di scrivere. Se hai capito i flowchart, qui ti sentirai a casa!

!!! tip "Editor interattivo 🎮"

In queste pagine troverai degli ****editor Python interattivi****: puoi modi

Il tuo primo programma

Tradizionalmente, il primo programma che si scrive in qualsiasi linguaggio è il mitico "Hello World". In Flowgorithm serviva un blocco Output... in Python basta UNA riga:

```
print("Hello World!")
```

Boom. Fatto. Prova a cambiare il messaggio e clicca **Run!**

La funzione `print()`

`print()` è la funzione che **mostra roba sullo schermo**. È l'equivalente del blocco Output di Flowgorithm, ma molto più flessibile.

```
print("Ciao!")  
print("Mi chiamo Python")  
print("2 + 3 =", 2 + 3)
```

Puoi stampare:

- **Testo** (tra virgolette `"..."` o apici `'...'`)
- **Numeri** (senza virgolette, altrimenti Python li tratta come testo!)
- **Più cose insieme**, separate da virgole — Python mette automaticamente uno spazio tra una e l'altra, gentile no?

La funzione `input()`

`input()` **chiede un dato all'utente**. È il blocco Input di Flowgorithm, ma con il messaggio di richiesta incluso direttamente. Niente più doppio blocco Output+Input, qui fai tutto in una riga sola!

```
nome = input("Come ti chiami? ")  
print("Ciao", nome, "!")
```

!!! warning "ATTENZIONE — Trappola per principianti! ⚠"

```
`input()` restituisce **SEMPRE una stringa** (testo), anche se l'utente
```

Commenti

I commenti servono a **spiegare il codice** a chi lo legge. E quel "chi lo legge" spesso sei tu tra due settimane, che hai già dimenticato tutto! Python li ignora completamente, sono solo per gli umani.

```
# Questo è un commento: Python lo ignora
print("Questa riga viene eseguita")  # Commento a fine riga

# I commenti aiutano a organizzare il codice:
# ===== INPUT =====
nome = input("Il tuo nome: ")

# ===== OUTPUT =====
print("Benvenuto,", nome)
```

Regola d'oro: commenta il **perché**, non il **cosa**. Scrivere `# incrementa x` sopra `x +=`
1 è inutile (grazie, lo vedo!). Scrivere `# aggiungo il bonus del weekend` è utile!

Struttura di un programma Python

Un programma Python si legge **dall'alto verso il basso**, riga per riga. Niente blocchi "Start" e "End" come in Flowgorithm: inizi a scrivere e quando il file finisce, il programma finisce. Semplice!

La struttura tipica segue sempre il solito schema che ormai dovresti conoscere a memoria:

```
# 1. INPUT - Raccolta dati
# 2. ELABORAZIONE - Calcoli e operazioni
# 3. OUTPUT - Mostra i risultati
```

Esempio completo (riconosci la struttura? Sì, è uguale a Flowgorithm!):

```
# Programma che calcola l'area di un rettangolo

# INPUT
base = float(input("Inserisci la base: "))
altezza = float(input("Inserisci l'altezza: "))

# ELABORAZIONE
area = base * altezza

# OUTPUT
print("L'area del rettangolo è:", area)
```

Visto? **Identico** a Flowgorithm, solo che invece di trascinare blocchetti, scrivi righe di codice. Stessa logica, diverso strumento!

Come eseguire Python sul tuo computer

L'editor interattivo di queste pagine è comodissimo per provare, ma prima o poi vorrai eseguire Python sul tuo PC "per davvero". Ecco come fare, senza impazzire:

Installazione

1. Scarica Python da python.org
2. Durante l'installazione, **spunta "Add Python to PATH"** — se non lo fai, preparati a soffrire
3. Clicca Installa e aspetta

Editor consigliati

- **IDLE** — viene installato insieme a Python, è basic ma funziona. Perfetto per iniziare
- **Visual Studio Code** — gratuito, professionale, con mille estensioni. Il preferito di chi fa sul serio
- **PyCharm Community** — potentissimo, ma un po' pesante. Per quando diventi un pro

Esecuzione

1. Crea un file con estensione `.py` (esempio: `programma.py`)

2. Scrivi il codice
3. Apri il terminale e digita: `python programma.py`
4. Guarda la magia accadere ✨