

# Introduzione a python

November 9, 2023

```
[1]: #Addizioni
numero1= int(input("inserisci il primo numero:"))
numero2= int(input("inserisci il secondo numero:"))
Addizione = numero1 + numero2
print("L'Addizione è:", Addizione)
```

```
inserisci il primo numero:5
inserisci il secondo numero:7
L'Addizione è: 12
```

```
[2]: #Sottrazioni
numero1= int(input("inserisci il primo numero:"))
numero2= int(input("inserisci il secondo numero:"))
Sottrazione = numero1 - numero2
print("La Sottrazione è:", Sottrazione)
```

```
inserisci il primo numero:80
inserisci il secondo numero:5
La Sottrazione è: 75
```

```
[3]: #Divisoni
numero1= int(input("inserisci il primo numero:"))
numero2= int(input("inserisci il secondo numero:"))
Divisione = numero1 / numero2
print("La Divisione è:", Divisione)
```

```
inserisci il primo numero:8
inserisci il secondo numero:2
La Divisione è: 4.0
```

```
[4]: #Moltiplicazioni
numero1= int(input("inserisci il primo numero:"))
numero2= int(input("inserisci il secondo numero:"))
Moltiplicazione = numero1 * numero2
print("La Moltiplicazione è:", Moltiplicazione)
```

```
inserisci il primo numero:7
inserisci il secondo numero:8
La Moltiplicazione è: 56
```

```
[5]: #Fattoriale
n = int(input("inserisci un numero intero positivo: "))
Fattoriale = 1

for numero in range(1, n + 1):

    Fattoriale *= numero
print("il Fattoriale di", n, "è:", Fattoriale)
```

```
inserisci un numero intero positivo: 5
il Fattoriale di 5 è: 120
```

```
[6]: #Media dei numeri
numeri = []

n = int(input("Quanti numeri vuoi inserire?"))

for i in range (n):
    numero = float(input("inserisci un numero:"))
    numeri.append(numero)

media = sum(numeri) / len(numeri)

print("La media dei numeri inseriti è:", media, "la lista completa è:", numeri)
```

```
Quanti numeri vuoi inserire?18
inserisci un numero:3
inserisci un numero:4
inserisci un numero:5
inserisci un numero:6
inserisci un numero:7
inserisci un numero:8
inserisci un numero:9
inserisci un numero:0
inserisci un numero:1
inserisci un numero:2
inserisci un numero:3
inserisci un numero:4
inserisci un numero:5
inserisci un numero:6
inserisci un numero:7
inserisci un numero:8
inserisci un numero:9
inserisci un numero:0
La media dei numeri inseriti è: 4.833333333333333 la lista completa è: [3.0,
4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 0.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0,
0.0]
```

```
[7]: #Gioco dell'indovinello
import random

numero_da_indovinare = random.randint(1, 100)
tentativi = 0

while True:
    tentativo = int(input("Indovina il numero (1-100): "))
    tentativi += 1

    if tentativo == numero_da_indovinare:
        print("Bravo! Hai indovinato il numero", numero_da_indovinare, "in",
→tentativi, "tentativi.")
        break
    elif tentativo < numero_da_indovinare:
        print ("Il numero da indovinare è più grande")
    else:
        print("il numero è più piccolo.")
```

```
Indovina il numero (1-100): 70
Il numero da indovinare è più grande
Indovina il numero (1-100): 90
Il numero da indovinare è più grande
Indovina il numero (1-100): 95
Il numero da indovinare è più grande
Indovina il numero (1-100): 99
Bravo! Hai indovinato il numero 99 in 4 tentativi.
```

```
[8]: #Gioco della morra cinese
import random

mosse = ["carta", "forbici", "sasso"]

computer_mossa = random.choice(mosse)

print("Benvenuti al Gioco della morra cinese!")
scelta_giocatore = input("Segli la tua mossa (carta, forbici, sasso):")

if scelta_giocatore not in mosse:
    print("mossa non permessa")
else:
    print("Il computer ha scelto:", computer_mossa)
    if scelta_giocatore == computer_mossa:
        print("Pareggio!")
    elif (scelta_giocatore == "carta" and computer_mossa == "sasso") or\
        (scelta_giocatore == "forbici" and computer_mossa == "carta") or\
        (scelta_giocatore == "sasso" and computer_mossa == "forbici"):
```

```

        print("Hai vinto!")
    else:
        print("Hai perso!")

```

Benvenuti al Gioco della morra cinese!  
 Segli la tua mossa (carta, forbici, sasso):carta  
 Il computer ha scelto: forbici  
 Hai perso!

```

[9]: #Fattoriale.2
numero=int(input("Inserisci un numero intero:"))

fattoriale=1

if fattoriale<0:
    print("numero negativo")
elif fattoriale==0:
    print("il fattoriale di zero è 1")
else:
    for numero in range(1, numero+1):
        fattoriale*=numero
    print(f"il fattoriale di {numero} è {fattoriale}")

```

Inserisci un numero intero:6  
 il fattoriale di 6 è 720

```

[10]: #Media dei numeri.2
numeri= []
n= int(input("quanti numeri vuoi inserire?"))
for i in range (n):
    numero = float(input("inserisci un numero:"))
    numeri.append(numero)
media = sum(numeri)/len(numeri)
print("La media dei numeri inseriti è", media, "La lista completa dei numeri è",
      numeri)

```

quanti numeri vuoi inserire?5  
 inserisci un numero:1  
 inserisci un numero:2  
 inserisci un numero:3  
 inserisci un numero:4  
 inserisci un numero:5  
 La media dei numeri inseriti è 3.0 La lista completa dei numeri è [1.0, 2.0, 3.0, 4.0, 5.0]

```

[11]: #Somma di tutte le cifre di un numero
N = int(input("inserisci un numero intero positivo di N:"))

somma = 0

```

```

for numero in range (2, 2 * N + 1, 2):
    somma += numero
print (f"la somma dei primi {N} numeri è pari è {somma}")

```

inserisci un numero intero positivo di N:4  
la somma dei primi 4 numeri è pari è 20

```

[12]: #Comando append
N = int(input("inserisci un numero intero positivo N:"))
lista = []

for numero in range(2,2 * N + 1,2):
    lista.append(numero)

print (lista)

```

inserisci un numero intero positivo N:7  
[2, 4, 6, 8, 10, 12, 14]

```

[13]: #Quadrato di un numero
n = int(input("inserisci un numero intero positivo:"))
print("Quadrati dei primi",n,"numeri")
for numero in range (1,n+1):
    quadrato = numero **2
print("il quadrato di", n, "è", quadrato)

```

inserisci un numero intero positivo:8  
Quadrati dei primi 8 numeri  
il quadrato di 8 è 64

```

[14]: #Conteggio vocali
frase = input("inserisci una frase o una parola:").lower()#per ridurre tutte le
↳ lettere minuscole

conteggio_vocali = 0

vocali = "aeiou"

for carattere in frase:
    if carattere in vocali:
        conteggio_vocali += 1

print(f"nella frase inserita ci sono {conteggio_vocali} vocali.")
#per ogni vocale somma uno al conteggio

```

inserisci una frase o una parola:buongiorno  
nella frase inserita ci sono 5 vocali.

```
[15]: #Comando for in range
for numero in range (3):
    print(numero)
```

```
0
1
2
```

```
[16]: #Tutti i segni d'operazione in un codice
operazione= input("inserisci l'operazione(+,-,*,/):")
numero1=float(input("inserisci il primo numero"))
numero2=float(input("inserisci il secondo numero"))
if operazione== "+":
    risultato=numero1+numero2
elif operazione== "-":
    risultato=numero1-numero2
elif operazione== "*":
    risultato=numero1*numero2
elif operazione== "/":
    risultato = numero1/numero2
else:
    risultato = "operazione non valida"
print("il risultato è,", risultato)
```

```
inserisci l'operazione(+,-,*,/):+
inserisci il primo numero90
inserisci il secondo numero7
il risultato è, 97.0
```

```
[17]: #Gioco del dado
import random

numero_dado = random.randint(1,6)

indovina = int(input("indovina il numero del dado (da 1 a 6:)"))

if indovina <1 or indovina >6:
    print("numero non ammesso")
elif indovina == numero_dado:
    print(f"complimento il numero del dado era {numero_dado}. hai indovinato!")
else:
    print(f"mi dispiace, il numero del dado era {numero_dado}. meglio fortuna,
    →alla prossima!")
```

```
indovina il numero del dado (da 1 a 6:)5
mi dispiace, il numero del dado era 3. meglio fortuna alla prossima!
```

```
[18]: #Conteggio della popolazione
popolazione = int(input("inserire popolazione iniziale:"))
anni = int(input("inserire numero di anni da simulare:"))

tasso_natalità = float(input("inserire tasso di natalità:"))
tasso_mortalità = float(input("inserire tasso di mortalità:"))

for anno in range(anni):
    nascite = (popolazione * tasso_natalità) / 100
    morti = (popolazione * tasso_mortalità) / 100
    popolazione += (nascite - morti)

    print(f"anno {anno+1}: popolazione = {int(popolazione)}")
print("simulazione completata")
```

```
inserire popolazione iniziale:300
inserire numero di anni da simulare:10
inserire tasso di natalità:3
inserire tasso di mortalità:2
anno 1: popolazione = 303
anno 2: popolazione = 306
anno 3: popolazione = 309
anno 4: popolazione = 312
anno 5: popolazione = 315
anno 6: popolazione = 318
anno 7: popolazione = 321
anno 8: popolazione = 324
anno 9: popolazione = 328
anno 10: popolazione = 331
simulazione completata
```

```
[19]: #Comando sulla data d'oggi
import datetime

today= datetime.datetime.today()
print(f"oggi è il giorno: {today:%d %m %Y} ore: {today:%H %M %S}")
```

```
oggi è il giorno: 09 11 2023 ore: 19 33 42
```

```
[20]: #Convertitore di unità di misure
print("benvenuto nel convertitore di unità di misura!")
scelta = input("cosa desideri convertire?(metri/piedi/chilogrammi/libbre):").
    ↪lower()

if scelta == "metri":
    valore = float(input("inserisci il valore in metri:"))
    risultato = valore * 3.28084
    print(f"{valore} metri corrispondono a {risultato} piedi")
```

```

if scelta == "piedi":
    valore = float(input("inserisci il valore in piedi:"))
    risultato = valore * 3.28084
    print(f"{valore} piedi corrispondono a {risultato} chilogrammi")
if scelta == "chilogrammi":
    valore = float(input("inserisci il valore in chilogrammi:"))
    risultato = valore * 2.20462
    print(f"{valore}chilogrammi corrispondono a {risultato} libbre")
if scelta == "libbre":
    valore = float(input("inserisci il valore in libbre:"))
    risultato = valore * 3.28084
    print(f"{valore} libbre corrispondono a {risultato} chilogrammi")
else:
    print("scelta non valida sceltri tra 'metri', 'piedi', 'chilogrammi', o
↪'libbre'")

```

benvenuto nel convertitore di unità di misura!  
cosa desideri convertire?(metri/piedi/chilogrammi/libbre):metri  
inserisci il valore in metri:30  
30.0 metri corrispondono a 98.4252 piedi  
scelta non valida sceltri tra 'metri', 'piedi', 'chilogrammi', o 'libbre'

```

[21]: #Calcolo numero di fibonacci
n = int(input("inserisci un numero n per calcolare l'n'esimo numero di
↪fibonacci"))

a=0
b=1
c=1

if n <= 0:
    print("il numero deve essere maggiore di zero.")
elif n == 1:
    risultato = a
else:
    for iterazione in range(n-3):
        a = b
        b = c
        c = a + b
    risultato = c
print(f"l'n-esimo numero di fibonacci è:", risultato)

```

inserisci un numero n per calcolare l'n'esimo numero di fibonacci10  
l'n-esimo numero di fibonacci è: 34

```

[22]: #Calcolo numero di fibonacci.2
def fibonacci(n):
    fib_series = [0, 1]

```



```

while len(fib_series) < n:
    fib_series.append(fib_series[-1]+ fib_series[-2])

return fib_series

```

[23]: fibonacci(5)

[23]: [0, 1, 1, 2, 3]

```

[24]: #Calcolo numero di fibonacci.3
n = int(input("inserisci il numero di termini della serie di fibonacci"))

if n <= 0:
    print("inserisci un numero positivo")
else:
    result = fibonacci(n)
    print(result)

```

inserisci il numero di termini della serie di fibonacci30

[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811, 514229]

```

[25]: #Calcolo d'aree
import math
def calcola_area_cerchio(raggio):
    return math.pi * (raggio** 2)

def calcola_area Rettangolo(base,altezza):
    return(base * altezza)

def calcola_area triangolo(base,altezza):
    return(base * altezza) / 2

```

```

[26]: print("benvenuti nella calcolatrice delle aree")

scelta = input("vuoi calcolare l'area di un cerchio(c), rettangolo(r) o
↳triangolo(t)").lower()

if scelta == 'c':
    raggio = float(input("inserisci il raggio del cerchio:"))
    area = calcola_area_cerchio(raggio)
    print(f"l'area del cerchio è {area:.2f}")
if scelta == 'r':
    base = float(input("inserisci la base del rettangolo:"))
    area = calcola_area Rettangolo(base,altezza)
    print(f"l'area del rettangolo è {area:.2f}")
if scelta == 't':

```

```

base = float(input("inserisci la base del triangolo:"))
area = calcola_area_triangolo(base,altezza)
print(f"l'area del triangolo è{area:.2f}")
else:
    print("scelta non valida. si prega di inserire 'c', 'r' o 't'.")

```

benvenuti nella calcolatrice delle aree  
 vuoi calcolare l'area di un cerchio(c), rettangolo(r) o triangolo(t)c  
 inserisci il raggio del cerchio:30  
 l'area del cerchio è 2827.43  
 scelta non valida. si prega di inserire 'c', 'r' o 't'.

```

[27]: #Liste
lista=[15,66,78]
x=int(input("inserisci un numero:"))
if x in lista:
    print("numero è presente nella lista")
else:
    print("il numero non è presente nella lista")

```

inserisci un numero:19  
 il numero non è presente nella lista

```

[28]: #Liste.2
prodotti={}
prodotti["pan bauletto"]=2
prodotti["coca cola"]=3

```

```

[29]: #Comando datetime
from datetime import datetime
import pytz

print("Benvenuto nell'orologio mondiale")

citta_fusi_orari = {
    "New york": "America/New_York",
    "Londra": "Europe/London",
    "Tokyo": "Asia/Tokyo",
    "Sydney": "Australia/Sydney",
    "Rio de Janeiro": "America/Sao_Paulo"
}

while True:
    print("\nCittà disponibili:")
    for citta in citta_fusi_orari.keys():
        print(citta)

```

```

    scelta_citta = input("Inserisci il nome della citta per visualizzare l'ora_
↳(o 'esci' per uscire):").strip()

    if scelta_citta.lower() == 'esci':
        break

    if scelta_citta in citta_fusi_orari.keys():
        fuso_orario = pytz.timezone(citta_fusi_orari[scelta_citta])
        ora_corrente = datetime.now(fuso_orario)
        print(f"L'ora corrente a {scelta_citta} è: {ora_corrente.strftime('%H:%M:
↳%S')}")
    else:
        print("Città non valida.")

```

Benvenuto nell'orologio mondiale

Città disponibili:

New york

Londra

Tokyo

Sydney

Rio de Janeiro

Inserisci il nome della citta per visualizzare l'ora (o 'esci' per uscire):Londra

L'ora corrente a Londra è: 18:34:54

Città disponibili:

New york

Londra

Tokyo

Sydney

Rio de Janeiro

Inserisci il nome della citta per visualizzare l'ora (o 'esci' per uscire):esci

```

[33]: def paolo():
        print("mi chiamo Paolo")

```

```

[34]: if __name__=="__main__":
        paolo()

```

mi chiamo Paolo

```

[39]: #funzione calcolo bmi
def calcola_bmi(peso,altezza):
    return peso/(altezza**2)
def valuta_bmi(bmi):
    if bmi<=18.5:
        return"sottopeso"

```

```

elif 19<=bmi <24:
    return"normopeso"
elif 25<= bmi <29.9:
    return"sovrappeso"
else:
    return"obeso"

```

```

[ ]: #calcolatrice bmi
def main():
    n=int(input("inserisci un numero di persone da valutare:"))
    for persone in range(n):
        print("Benvenuto nella calcolatrice bmi")
        peso = float(input("inserisci il tuo peso in chilo grammi:"))
        altezza = float(input("inserisci la tua altezza in metri:"))

        bmi = calcola_bmi(peso,altezza)
        valutazione = valuta_bmi(bmi)

        print(f"il tuo bmi è {bmi:.2f},sei calssificato come '{valutazione}'")

if __name__ == "__main__":
    main()

```

```

inserisci un numero di persone da valutare:6
Benvenuto nella calcolatrice bmi
inserisci il tuo peso in chilo grammi:67
inserisci la tua altezza in metri:1.70
il tuo bmi è 23.18,sei calssificato come 'normopeso'
Benvenuto nella calcolatrice bmi

```

```

[ ]: #convertitore unità di misura
def metri_a_piedi(metri):
    return metri * 3.28084
def piedi_a_metri(piedi):
    return piedi / 3.28084
def chilogrammi_a_libbre(chilogrammi):
    return chilogrammi * 2.20462
def libbre_a_chilogrammi(libbre):
    return libbre / 2.20462

def selezione(scelta):
    if scelta == "metri":
        valore = float(input("inserisci il valore in metri:"))
        risultato = metri_a_piedi(valore)
        print(f"{valore: .3f}metri corrispondono a {risultato: .3f}piedi.")
    if scelta == "piedi":
        valore = float(input("inserisci il valore in piedi:"))
        risultato = piedi_a_metri(valore)

```

```

        print(f"{valore: .3f}piedi corrispondono a {risultato: .3f}metri.")
    if scelta == "chilogrammi":
        valore = float(input("inserisci il valore in chilogrammi:"))
        risultato = chilogrammi_a_libbre(valore)
        print(f"{valore: .3f}chilogrammi corrispondono a {risultato: .3f}libbre.
↪")
    if scelta == "libbre":
        valore = float(input("inserisci il valore in libbre:"))
        risultato = libbre_a_chilogrammi(valore)
        print(f"{valore: .3f}libbre corrispondono a {risultato: .3f}chilogrammi")
    else:
        print("sveglia scelta non valida")

def main():
    print("Ciao")
    scelta = ("scegli").lower()
if __name__=="__main__":
    main()

```

```

[ ]: #Calcolo delle calorie
cibo_calorie = {
    "Pizza": 285,
    "hamburger": 250,
    "insalata": 100,
    "pollo arrosto": 335,
    "yogurt": 150
}

def calorie_consumate(cibo, quantita):
    if cibo not in cibo_calorie.keys():
        print("cibo non presente")
    elif cibo in cibo_calorie.keys():
        calorie_per_100g = cibo_calorie[cibo]
        calorie_totali = (calorie_per_100g / 100) * quantita
        return calorie_totali

def main():
    cibo_consumato = []

    while True:
        print("Menu:")
        print("\n 1. aggiungi cibo consumato")
        print("\n 2. calcolacalorie totali")
        print("\n 3. esci")

        scelta = input("scegli un'opzione")

```

```

    if scelta == "1":
        print("\n")

        for key, value in cibo_calorie.items():
            print(key, value)

        cibo = input("inserisci il cibo consumato:").lower()
        quantita = float(input("inserisci la quantità (in grammi):"))
        cibo_consumato.append((cibo, quantita))

    elif scelta == "2":
        calorie_totali = sum(calorie_consumate(c, q) for c, q in cibo_consumato)
        print(f"\nCalorie totali consumate: {calorie_totali} calorie")
    elif scelta == "3":
        break
    else:
        print("\nScelta non valida. riprova")
if __name__=="__main__":
    main()

```

```

[ ]: acquistidue={
    "pan bauletto":10,
    "nutella":10,
}

```

```

[ ]: acquisti={}
acquisti["pan bauletto"]=10
acquisti["nutella"]=10

```

```

[ ]: #generatore di personaggi fantasy
import random

speci = ["elfo", "umano", "nano", "orco", "gnomo"]
classi = ["guerriero", "mago", "ranger", "ladro", "chierico"]
armi = ["spada", "arco", "bachetta magica", "ascia", "daga"]
abilita = ["furtivita", "magia dell'acqua", "camuffamento", "estrazione",
    "mineraria", "incantesimi di guarigione"]

specie = random.choice(speci)
classe = random.choice(classi)
arma = random.choice(armi)
abilita_scelta = random.sample(abilita, random.randint(1, 3))

print(f"personaggio fantasy generato:")
print(f"specie:{specie}")
print(f"classe:{classe}")

```

```
print(f"arma:{arma}")
print(f"abilità: {' '.join(abilita_scelta)}")
```

```
[ ]: #generatore di personaggi fantasy 2
import random
speci=["elfo","nano","orco","gnomo"]
classi=["guerriero","mago","ranger","ladro","chierico"]
armi=["spada","arco","Bacchetta magica","ascia","daga"]
abilità=["furtività","magia dell'acqua","camuffamento","estrazione_
↳mineraria","guarigione"]
def crea_personaggio():
    return{
        "specie":random.choice(speci),
        "classe":random.hoice(classi),
        "arma":random.choice(armi),
        "abilità":random.sample(abilità,random.randint(1,3))
    }
    return personaggio
def main():
    personaggio_generato=crea_personaggio()
    print("personaggio generato")
    for chiave,valore in personaggio_generato.items():
if __name__=="__main__":
    main()
```

```
[ ]: #generatore di anagrammi
from itertools import permutations
k=0

def trova_anagrammi(parola):
    anagrammi=[" ".join(p)for p in permutations(parola)]
    return anagrammi

print("benvenuti nel risolutore di anagrammi")
parola_input=input("inserisci una parola").strip().lower()

if len(parola_input)<2:
    print("inserisci una parola con almeno due caratteri")
else:
    anagrammi=trova_anagrammi8parola_input)

    for elemento in anagrammi:
        if elemento != parola_input:
            k += 1
        print(elemento)
print(f"gli anagrammi della parola{parola_input} sono {k}")
```

```
[ ]: #calcolatore interessi
def calcola_interessi(importo_iniziale,tasso_interesse,periodi_investimento):
    importo_finale=importo_iniziale*(1+tasso_interesse/100)**periodi_investimento
    return valore_finale

print("benvenuto nel calcolatore di interessi")
importo=float(input("inserisci valore iniziale"))
tasso=float(input("inserisci interesse"))
periodo=int(input("inserisci il periodo di investimento"))
importo_finale=calcola_interessi(importo,tasso,periodo)
print(f"l'importo finale dopo{periodo} anni è di {importo_finale:.2f}euro")
```

```
[ ]: #calcolatore tassi di cambio
tassi_di_cambio={
    "dollari":1.0
    "euro".0.85
    "yen":110.41
}

importo=float(input("inserisci l'importo da convertire"))
valuta_di_partenza=input("inserisci la valuta di partenza:").lower
valuta_destinazione=input("inserisci la valuta di destinazione").lower

if valuta_di_partenza in tassi_di_cambio and valuta_destinazione in tassi_di_cambio:
    tasso_di_cambio=tassi_di_cambio[valuta_destinazione]/tassi_di_cambio[valuta_di_partenza]
    importo_convertito=importo*tasso_di_cambio

    print(f"{importo}{valuta_di_partenza}sono equivalenti a {importo_convertito:.2f}{valuta_destinazione}")
else:
    print("valute non supportate")
```

```
[ ]: #conteggio lettere
frase=input("inserisci una frase")
frase=frase.lower()
alfabeto='abcdefghijklmnopqrstuvwxyz'
conteggio_lettere={}
for lettera in alfabeto:
    conteggio=frase.count(lettera)
    if conteggio>0:
        conteggio_lettere[lettera]=conteggio
for lettera, conteggio in conteggio_lettere.items():
    print(f"{lettera}:{conteggio}")
```



```
[ ]: #calcolo forza gravitazionale
def forza_gravitazionale(m1,m2,r):
    #costante gravitazionale
    G=6.67430_11
    #calcolo della forza gravitazionale
    F=(G*m1*m2)/(r**2)
    return f
#esempio di utilizzo
massa_terra=5.97224
massa_luna=7.3422
distanza_terra_luna=384000000

forza=forza_gravitazionale(massa_terra,massa_luna,distanza_terra_luna)
print(f"forza gravitazionale tra la terra e la luna{forza}Newton")
```