

# Esercizi (for)

## Esercizio 1

Utilizzando il for loop, stampare il seguente pattern. **CONSIGLIO:** utilizzare due for loop annidati.

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

```
In [7]: print("Pattern con numeri")

righe = 5

# start: 1
# stop: row+1 (range never include stop number in result)
# step: 1
# run loop 5 times
for i in range(1, righe + 1, 1):
    # Run inner loop i+1 times
    for j in range(1, i + 1):
        print(j, end=" ")
    # empty line after each row
    print("")
```

Pattern con numeri  
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5

## Esercizio 2

### Massimo e minimo

Scrivere un programma che trovi il valore massimo e minimo in una lista di numeri (senza usare max() e min()).

```
In [5]: numeri = [10.5, 25.2, 3.4, 99, 45, 67.1]
massimo = numeri[0]
minimo = numeri[0]

for numero in numeri:
    if numero > massimo:
        massimo = numero
    if numero < minimo:
        minimo = numero

print("Massimo:", massimo)
print("Minimo:", minimo)
```

Massimo: 99  
Minimo: 3.4

## Esercizio 3

### Conta numeri pari

Scrivi un programma che conti quanti numeri pari ci sono in una lista.

```
In [11]: numeri = [2, 7, 18, 33, 22, 42, 89]
conta_pari = 0

for numero in numeri:
    if numero % 2 == 0:
        conta_pari += 1

print("Numeri pari:", conta_pari)
```

Numeri pari: 4

## Esercizio 4

### Stampa quadrati

Usa un ciclo for per calcolare e stampare i quadrati dei numeri da 1 a 10.

```
In [13]: quadrati = []

for i in range(1, 11):
    quadrato = i ** 2
    quadrati.append(quadrato)

print("Quadrati:", quadrati)
```

Quadrati: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

## Esercizio 5

### Occorrenze

Scrivi un programma che conti quante volte un certo valore appare in una lista.

```
In [18]: lista = [3, 7, 3, 2, 3, 8, 3, 9]
elemento_da_contare = 3
conteggio = 0

for elemento in lista:
    if elemento == elemento_da_contare:
        conteggio += 1

print(f"Il numero {elemento_da_contare} appare {conteggio} volte nella lista.")
```

Il numero 3 appare 4 volte nella lista.

## Esercizio 6

### Successione di fibonacci

Chiedere all'utente di inserire un numero intero *n* e stampare i primi *n* numeri della successione di fibonacci.

```
In [16]: n = int(input("Inserire il numero di elementi della serie di fibonacci che si desidera"))
if n <= 0:
    print("Inserire un numero corretto")
elif n == 1:
    print([1])
else:
    l = [1, 1]
    for i in range(2, n):
        l.append(l[i-1] + l[i-2])
    print(l)
```

[1]

## Esercizio 7

### Calcolo del fattoriale

Chiedere all'utente di inserire un numero, calcolarne e stamparne il fattoriale.

```
In [25]: n = int(input("Inserire un numero intero non negativo"))
fattoriale = 1
if n < 0:
    print("Il fattoriale non esiste per numeri minori di 0")
elif n == 0:
    print("Il fattoriale di 0 è 1")
else:
    # run loop 5 times
    for i in range(1, n + 1):
        # multiply factorial by current number
        fattoriale = fattoriale * i
    print("Il fattoriale di", n, "è", fattoriale)
```

Il fattoriale di 5 è 120

## Esercizio 8

### Stampa numeri primi

Stampare tutti i numeri primi tra un range di numeri.

**Nota:**

- Un numero primo è un numero che non può essere ottenuto moltiplicando altri numeri interi.
- Un numero primo è un numero naturale maggiore di 1 che non è il prodotto di due numeri naturali minori.
- Un numero primo è un numero naturale maggiore di 1 divisibile solo per 1 o per sé stesso.

```
In [37]: start = 25
end = 50
print("Prime numbers between", start, "and", end, "are:")

for num in range(start, end + 1):
    # all prime numbers are greater than 1
    # if number is less than or equal to 1, it is not prime
    if num > 1:
        for i in range(2, num):
            # check for factors
            if (num % i) == 0:
                # not a prime number so break inner loop and
                # look for next number
                break
        else:
            print(num)
```

Prime numbers between 25 and 50 are:  
29  
31  
37  
41  
43  
47