

Introduzione a Python

Basi di informatica, gestione dati e statistica (BIGDS)

Python

- Python è uno dei *più diffusi* linguaggi di programmazione e ha una *curva di apprendimento* pressoché piatta. Enorme disponibilità di *librerie*: NumPy, Pandas, scikit-learn...
- Funzionale e orientato agli oggetti
- Consolidato nel tempo
- Semplice e versatile

Download

<https://www.python.org/downloads/>

Dipendenza dal SO

IDE – Integrated Development Environment

Windows (multiplatforma)

- **pycharm:** <https://www.jetbrains.com/pycharm/>
- **wing:** <http://wingware.com/downloads>
- **spyder:** <https://www.spyder-ide.org/>

web-based tools

- **codeboard:** codeboard.io
- **replit:** <https://replit.com/>
- **colab:** <https://colab.research.google.com/>

Setup e primo programma

- Preferenze del IDE
- Creazione di un nuovo progetto

Primo programma: `print("Hello world")`

dove `print` rappresenta una **funzione** predefinita (*built-in*)

- Ogni istruzione termina con un *a capo*
- Importanza della *indentazione*

I commenti

```
# Il mio primo programma  
print("Hello world")
```

Variabili

Contenitori che posso *accogliere* al loro interno dei dati (*tipo e nome*)

Esempio:

```
print("Matteo insegna Informatica")  
print("Federica insegna Matematica")
```

con le **variabili** e operatore di **assegnamento**:

```
char_nome = "Matteo"  
char_materia = "Informatica"  
print(char_nome + " insegna " + char_materia)  
char_nome = "Federica"  
char_materia = "Matematica"  
print(char_nome + " insegna " + char_materia)
```


Differenti tipo di dato

La definizione degli *oggetti* (variabili) ha come conseguenza l'assegnazione di uno specifico spazio in *memoria centrale*

Stringa "Matteo insegna Informatica"

Numeri

anni = 49

Boolean True o False

Il **tipo** definisce l'insieme di operazioni che possono essere svolte su quella specifica variabile

Le stringhe

```
frase = "Matteo insegna Informatica"  
print(frase + " e Statistica")
```

Le stringhe e funzioni

```
frase = "Matteo insegna Informatica"  
print(frase.lower())
```

```
frase = "Matteo insegna Informatica"  
print(frase.upper())
```

```
frase = "Matteo insegna Informatica"  
print(frase.replace("Matteo", "Luca"))
```

Le stringhe e funzioni

```
frase = "Matteo insegna Informatica"  
print(len(frase))
```

```
frase = "Matteo insegna Informatica"  
print(frase[0])
```

Python usa lo 0 come indice del primo elemento di una stringa

Numeri

```
print(1)
print(1.234)
print(-1)
print(-1 + 1)
print(4 / 2)
print(4 % 3)
```

```
num = 3
print(num)
```

Numeri e stringhe

```
frase = "Matteo insegna Informatica"  
anni = 49  
print(frase + " e ha " + str(anni) + " anni")
```

Numeri e funzioni

```
num = -3  
print(abs(num))
```

```
num = 3  
print(pow(num, 2))
```

```
num = 3.3  
print(round(num))
```

```
from math import *  
num = 3.3  
print(sqrt(num))
```

Input da utente

```
input()
```

```
nome = input("Inserisci il tuo nome: ")  
print("Ciao " + nome + "!!")
```

La variabile `nome` conterrà la **stringa** inserita dall'utente

Calcolatrice

```
num_1 = input("Inserisci un numero: ")
num_2 = input("Inserisci un altro numero: ")
risultato = int(num_1) + int(num_2)
print(risultato)
```

...ma attenzione!

```
num_1 = input("Inserisci un numero: ")
num_2 = input("Inserisci un altro numero: ")
risultato = float(num_1) + float(num_2)
print("Risultato = ", risultato)
```

Istruzioni condizionali

Istruzione if

```
if condizione:  
    istruzione/i
```

```
i = 1  
if i < 10:  
    print(str(i) + " e' minore di 10")
```

- La valutazione di un'espressione restituisce `True` o `False` (1 o 0)

Istruzioni condizionali

Ramo else

```
if condizione:  
    istruzione/i  
else:  
    istruzione/i
```

```
i = 100  
if i < 10:  
    print(str(i) + " e' minore di 10")  
else:  
    print(str(i) + " e' maggiore di 10")
```

Istruzioni condizionali

if annidati

```
i = -1
if i < 10:
    if i < 0:
        print(str(i) + " e' negativo")
    else:
        print(str(i) + " e' minore di 10")
else:
    print(str(i) + " e' maggiore di 10")
```

Istruzioni condizionali

if annidati

```
i = 10
if i < 10:
    if i < 0:
        print(str(i) + " e' negativo")
    else:
        print(str(i) + " e' minore di 10")
else:
    if i == 10:
        print(str(i) + " e' uguale a 10")
    else:
        print(str(i) + " e' maggiore di 10")
```

Istruzioni condizionali

Esercizio

Scrivere un programma che permetta di valutare se un numero intero (letto da tastiera) è pari o dispari

```
print("Inserisci un numero: ")
num = int(input())

if num % 2 == 0:
    print("Il numero inserito e' pari")
else:
    print("Il numero inserito e' dispari")
```

Istruzioni condizionali

elif

```
if condizione:  
    istruzione/i  
elif condizione:  
    istruzione/i  
else:  
    istruzione/i
```


Esercizio

Creare un programma che presi in ingresso il peso (in Kg) e l'altezza (in m) di una persona, calcoli il BMI.

$$BMI = peso / (altezza^2)$$

Il programma dovrà successivamente assegnare una delle seguenti tre classi:

- sottopeso se $BMI \leq 20$
- normopeso se $20 < BMI \leq 30$
- sovrappeso se $BMI > 30$

```
peso = float(input("Inserisci il peso in Kg: "))
altezza = float(input("Inserisci l'altezza in m: "))

bmi = peso / altezza**2

print("Il bmi vale: {:.2f}".format(bmi))

if bmi <= 20:
    print("Sei sottopeso")
elif 20 < bmi <= 30:
    print("Sei normopeso")
else:
    print("Sei sovrappeso")
```

Calcolatrice v.2

```
num_1 = float(input("Inserisci un numero: "))
op = input("Inserisci l'operazione: ")
num_2 = float(input("Inserisci un altro numero: "))

if op == "+":
    print("Risultato = ", num_1 + num_2)
elif op == "-":
    print("Risultato = ", num_1 - num_2)
elif op == "*":
    print("Risultato = ", num_1 * num_2)
elif op == "/":
    print("Risultato = ", num_1 / num_2)
else:
    print("Operazione non valida!")
```

Istruzioni iterative

Ripetere un'operazione un certo numero di volte

```
print(1)  
print(2)  
print(3)
```

```
for i in (1, 2, 3):  
    print(i)
```

```
for valore in sequenza:  
    istruzione/i
```

- ad ogni iterazione `i` assume i valori indicati nelle `()` ed viene eseguita l'istruzione che segue `i` `:`
- l'operatore `in` assume `True` o `False`

Istruzioni iterative

```
somma = 0
for i in (0, 1, 2):
    somma = somma + i
print(somma)
```

```
somma = 0
for i in range(3):
    somma = somma + i
print(somma)
```

Istruzioni iterative

```
somma = 0
for i in range(3):
    n = int(input("Inserisci un numero: "))
    somma = somma + n
print("La somma vale: " + str(somma))
```

Istruzioni iterative

```
while espressione:  
    istruzione/i
```

```
somma = 0  
i=1  
numero = 1  
  
while numero != 0 and i <= 10:  
    numero = input("Ins. valore: ")  
    numero = int(numero)  
    somma = somma + numero  
    i = i + 1  
print("Somma = ", somma)
```

Istruzioni iterative: media

```
somma = 0
i = 0
numero = 1

while numero != 0:
    numero = input("Ins. valore: ")
    numero = int(numero)
    somma = somma + numero
    i = i + 1
print("Somma = ", somma)
print("i = ", i)
media = somma / (i - 1)
print("Media = ", media)
```


Esercizio

Scrivere un programma che calcoli la media di N numeri. N è definito dall'utente.

```
print("Quanti numeri vuoi inserire: ")
n = int(input())
print("Inserisci i " + str(n) + " numeri: ")
somma = 0
for i in range(n):
    numero = int(input("Inserisci un numero: "))
    somma = somma + numero
media = somma / n
print("La media vale: " + str(media))
```

Domande?

www.menti.com