

RequisitiLezione

March 20, 2019

1 LEZIONE 1 - Cosa è necessario conoscere

1. La variabile e come le si assegna un valore (tramite l'istruzione di assegnazione) `variabile1 = 100` `variabile2 = 10.2` `variabile3 = 'Hello world!'` `variabile4 = True`
2. L'operatore `/` di divisione (tra due valori numerici) `divisione = variabile1 / variabile2`
3. L'operatore `//` di divisione intera (tra due valori numerici) `divisione = variabile1 // variabile2`
4. L'operatore `%` di resto della divisione intera (tra due valori numerici) `resto_divisione_intera = variabile1 % variabile2`
5. L'operatore `+` di somma (tra due valori numerici) `somma = variabile1 + variabile2`
6. L'operatore `*` di prodotto (tra due valori numerici) `prodotto = variabile1 * variabile2`
7. L'operatore `**` di elevamento a potenza (tra due valori numerici) `quadrato = variabile1**2` `radice_quadrata = variabile1**0.5`
8. L'operatore `-` di negazione aritmetica `negazione_aritmetica = -variabile1`
9. Gli operatori di confronto `==`, `!=`, `>` e `<` (usati in espressioni booleana, che restituiscono i valori `True` e `False`) `variabile1 == variabile2` `variabile1 != variabile2` `variabile2 < variabile1` `variabile2 > variabile1`
10. Gli operatori logici `and` e `or` (usati in espressioni booleana, che restituiscono i valori `True` e `False`) `variabile1 == variabile2 and variabile1 < variabile3` `variabile1 == variabile2 or variabile1 < variabile3`
11. La stringa di caratteri e come si assegna una stringa a una variabile `stringa = 'Hello world!'`
12. La funzione predefinita `len()` per determinare la dimensione di una stringa `dimensione_stringa = len(stringa)`
13. L'indicizzazione positiva dei caratteri in una stringa: 0 è l'indice del primo carattere, 1 quello del secondo, etc., `dimensione_stringa-1` è l'indice dell'ultimo carattere)
14. L'indicizzazione negativa dei caratteri in una stringa: -1 è l'indice dell'ultimo carattere, -2 quello del penultimo, etc., `-dimensione_stringa` è l'indice del primo carattere

15. La sintassi di accesso al singolo carattere di una stringa tramite l'indice della sua posizione
`primo_carattere = stringa[0]` `primo_carattere = stringa[-dimensione_stringa]`
16. La sintassi di *slicing* per accedere a una sottostringa
`sottostringa_da_terzo_a_ottavo_carattere = stringa[2:8]`
17. L'operatore `in` per verificare la presenza di un carattere all'interno di una stringa (usato in un'espressione booleana, che restituisce i valori `True` o `False`) `'w' in stringa`
18. Cos'è una lista e come si assegna una lista a una variabile `lista = [False, 100, 345.789, 'Hello world!']`
19. La funzione predefinita `len()` per determinare la dimensione di una lista
`dimensione_lista = len(lista)`
20. L'indicizzazione positiva degli elementi di una lista: 0 è l'indice del primo elemento, 1 quello del secondo, etc., `dimensione_lista-1` è l'indice dell'ultimo elemento)
21. L'indicizzazione negativa degli elementi di una lista: -1 è l'indice dell'ultimo elemento, -2 quello del penultimo, etc., `-dimensione_lista` è l'indice del primo elemento
22. La sintassi di accesso al singolo elemento di una lista tramite l'indice della sua posizione
`terzo_elemento = lista[2]` `terzo_elemento = lista[-dimensione_lista+2]`
23. La sintassi di aggiornamento del singolo elemento di una lista tramite l'indice della sua posizione
`lista[2] = 100`
24. La sintassi di *slicing* per accedere a una sottolista
`sottolista_da_secondo_a_terzo_elemento = lista[1:3]`
25. La sintassi di aggiornamento di una sottolista tramite *slicing*
`lista[1:3] = [10,10,10,10]`
26. L'operatore `in` per verificare la presenza di una chiave all'interno di un dizionario (usato in un'espressione booleana, che restituisce `True` e `False`) `456 in lista`
27. Il dizionario e come si assegna un dizionario a una variabile
`dizionario = {'mele' : 4, 'pere' : 6, 'fragole' : 15}`
28. La funzione predefinita `len()` per determinare il numero di coppie chiave-valore di un dizionario
`dimensione_dizionario = len(dizionario)`
29. La sintassi di accesso al singolo valore tramite la sua chiave
`numero_mele = dizionario['mele']`
30. La sintassi di aggiornamento del singolo valore tramite la sua chiave
`dizionario['mele'] = 6`
31. L'operatore `in` per verificare la presenza di una chiave all'interno di un dizionario (usato in un'espressione booleana, che restituisce `True` e `False`) `'ciliegie' in dizionario`
32. Il costrutto di selezione `if`
`if condizione_da_verificare_vera: faccio_qualcosa`
33. Il costrutto di selezione alternativa `if-else`
`if condizione_da_verificare_vera: faccio_qualcosa else: faccio_altro`

34. I costrutti di selezione in serie if-elif-else
- ```
if condizione1_da_verificare_vera:
 eseguo_task1
elif condizione2_da_verificare_vera:
 eseguo_task2
elif condizione3_da_verificare_vera:
 eseguo_task3
...
elif condizioneN_da_verificare_vera:
 eseguo_taskN
else:
 eseguo_task_finale
```
35. I costrutti di selezione nidificati
36. La sintassi attraversamento dei caratteri di una stringa o degli elementi di una lista tramite for
- ```
for carattere in stringa:
    faccio_qualcosa
```

```
for elemento in lista:
    faccio_qualcosa
```