

ESERCIZIO W7D1 - Programmazione per Hacker (Python PT.2)

RICHIESTA

La parte obbligatoria dell'esercizio W7D1 prevedeva la scrittura di una funzione che restituisse, in output, una lista B di interi rappresentante la lunghezza delle parole contenute in una lista A precedentemente popolata.

La parte facoltativa dell'esercizio prevedeva invece la scrittura di una funzione che generasse una password. La password, a discrezione dell'utilizzatore del programma, poteva essere debole (8 caratteri con solo lettere e numeri) o forte (20 caratteri con lettere, numeri e simboli).

~~~

### SOLUZIONE

Per la risoluzione della prima parte dell'esercizio ho creato un programma basato su un apposito comando di Python, che consente proprio di calcolare di quanti caratteri è composto un elemento della lista.

Per la risoluzione della seconda parte, invece, ho utilizzato una serie di funzioni che consentono innanzitutto all'utente di scegliere la tipologia di password richiesta. Ho implementato poi le logiche richieste dall'esercizio in tema di caratteri consentiti, concludendo poi con la stampa a schermo della password generata.

~~~

ASPETTI TECNICI

Per la parte obbligatoria dell'esercizio, l'aspetto tecnico più interessante è l'utilizzo del comando **len**, che si occupa di calcolare di quanti caratteri è composto un elemento di una lista. Ho inserito questo comando all'interno di una funzione, richiamata poi dalla **main** del programma.

Per una maggiore leggibilità e modularità del codice, ho voluto poi creare anche una funzione di presentazione. In questo modo abbiamo **due funzioni separate** (presentazione e conteggio) richiamate, ognuna al momento giusto, dalla parte principale del programma.

Per quel che riguarda la parte facoltativa dell'esercizio, il primo aspetto tecnico da considerare è il richiamo ai metodi **random** e **string**. Random l'ho utilizzato per effettuare una selezione casuale fra i diversi elementi di una lista. String invece è un metodo che consente allo sviluppatore di utilizzare una serie di stringhe di caratteri già impostate. Un esempio è **string.ascii_letters**, che ho utilizzato per includere nella lista dei caratteri consentiti l'intero alfabeto inglese (26 caratteri) sia in maiuscolo che in minuscolo.

La struttura del programma è semplice, con **cinque diverse funzioni** (presentazione, sceltaUtente, passDebole, passForte e generazionePass) che vengono richiamate, ognuna al momento giusto, all'interno della **main** del programma.

Il metodo per la creazione di entrambe le password è uguale, con un ciclo **FOR** che itera la scelta di un carattere casuale fra quelli consentiti, prima per 8 volte e poi per 20 volte. La differenza più importante è che nella generazione della password forte sono inclusi **i caratteri simbolici**, esclusi invece nella generazione di password deboli.

LISTATO

Nelle successive immagini, ho incluso il listato e l'output di entrambi i programmi. Per ragioni di leggibilità, ho diviso il codice in varie immagini

```
W7D1Facoltativo.py W7D1.py
Users > pagizza > Code > W7D1.py > ...
1 #Questa è la lista delle parole su cui vogliamo effettuare il conteggio dei caratteri
2 lista_parole = ["Verde", "Giallo", "Rosso", "Azzurro"]
3
4 #Questa è una funzione che si occupa della presentazione del programma
5 def presentazione():
6     print ("*****")
7     print ("Questo programma calcola di quanti caratteri e' composta ogni parola di una lista")
8     print ("*****")
9
10 #Questa è la funzione che si occupa del conteggio dei caratteri
11 def conteggio (lista):
12     conteggioCaratteri = []
13     for parola in lista:
14         lunghezzaParola = len(parola)
15         conteggioCaratteri.append(lunghezzaParola)
16     return conteggioCaratteri
17
18 #Questa è la main del programma, con il richiamo alle altre funzioni
19 presentazione()
20 conteggioTotale = conteggio (lista_parole)
21 print ("La lista di parole e' composta da questi elementi:")
22 print (lista_parole)
23 print ("Il numero di caratteri delle parole nella nostra lista e' il seguente:")
24 print ([conteggioTotale])
```

```
Users > pagizza > Code > W7D1.py > ...
1 #Questa è la lista delle parole su cui vogliamo effettuare il conteggio dei caratteri
2 lista_parole = ["Verde", "Giallo", "Rosso", "Azzurro"]
3
4 #Questa è una funzione che si occupa della presentazione del programma
5 def presentazione():
6     print ("*****")
7     print ("Questo programma calcola di quanti caratteri e' composta ogni parola di una lista")
8     print ("*****")
9
10 #Questa è la funzione che si occupa del conteggio dei caratteri
11 def conteggio (lista):
12     conteggioCaratteri = []
13     for parola in lista:
14         lunghezzaParola = len(parola)
15         conteggioCaratteri.append(lunghezzaParola)
16     return conteggioCaratteri
17
18 #Questa è la main del programma, con il richiamo alle altre funzioni
PROBLEMI OUTPUT CONSOLE DI DEBUG TERMINALE PORTE
python3 -u "/Users/pagizza/Code/W7D1.py"
pagizza@MacBook-APPLE ~ % python3 -u "/Users/pagizza/Code/W7D1.py"
*****
Questo programma calcola di quanti caratteri e' composta ogni parola di una lista
*****
La lista di parole e' composta da questi elementi:
['Verde', 'Giallo', 'Rosso', 'Azzurro']
Il numero di caratteri delle parole nella nostra lista e' il seguente:
[5, 6, 5, 7]
pagizza@MacBook-APPLE ~ %
```

```
W7D1Facoltativo.py X
Users > pagizza > Code > W7D1Facoltativo.py > ...
1 #Importo i metodi random e string. Servono a scegliere un carattere casuale e utilizzare liste già definite
2 import random
3 import string
4
5 #Questa è la funzione relativa alla parte di presentazione del programma
6 def presentazione ():
7     print ("Epicode Password Generator")
8     print ("*****")
9     print ("L'Epicode Password Generator è un generatore di password")
10    print ("A seconda della tua scelta, può generare password deboli o password forti")
11    print ("*****")
12    print ("Una password debole è composta da 8 caratteri")
13    print ("Una password forte è composta da 20 caratteri")
14    print ("*****")
15
16 #Questa è la funzione che gestisce la scelta dell'utente e un'eventuale scelta non valida
17 def sceltaUtente ():
18     scelta=input ("Digita 1 sulla tastiera per generare una password debole o 2 per generare una password forte ")
19     validitaScelta = False
20     while not validitaScelta:
21         if scelta == "1":
22             validitaScelta = True
23             return scelta
24         elif scelta == "2":
25             validitaScelta = True
26             return scelta
27         else:
28             print ("La tua scelta non è valida")
29             scelta=input ("Digita 1 sulla tastiera per generare una password debole o 2 per generare una password forte ")
30
```

```
W7D1Facoltativo.py x
Users > pagizza > Code > W7D1Facoltativo.py > ...

31 #Questa è la funzione da richiamare per creare una password debole da 8 caratteri
32 def passDebole ():
33     stringaCaratteri = string.ascii_letters
34     stringaNumeri = string.digits
35     caratteriConsentiti = stringaCaratteri+stringaNumeri
36     lunghezzaPassword = 8
37     passwordDebole = ""
38     for i in range (lunghezzaPassword):
39         carattere = random.choice(caratteriConsentiti)
40         passwordDebole = passwordDebole + carattere
41     return passwordDebole
42
43 #Questa è la funzione da richiamare per creare una password forte da 20 caratteri, compresi simboli
44 def passForte ():
45     stringaCaratteri = string.ascii_letters
46     stringaNumeri = string.digits
47     stringaSimboli = string.punctuation
48     caratteriConsentiti = stringaCaratteri+stringaNumeri+stringaSimboli
49     lunghezzaPassword = 20
50     passwordForte = ""
51     for i in range (lunghezzaPassword):
52         carattere = random.choice(caratteriConsentiti)
53         passwordForte = passwordForte + carattere
54     return passwordForte
55
56 #Questa funzione richiama una o l'altra funzione di generazione, a seconda della scelta dell'utente
57 def generazionePass (digitazioneUtente):
58     if digitazioneUtente == "1":
59         password = passDebole()
60         return password
61     elif digitazioneUtente == "2":
62         password = passForte()
63         return password
64     else:
65         ""
66     return None
67
68 #Questa è la main del programma, con il richiamo delle varie funzioni e la stampa a video della password
69 presentazione()
70 digitazioneUtente = sceltaUtente()
71 passwordCompleta = generazionePass (digitazioneUtente)
72 print ("*****")
73 print ("Ecco la tua password: ", passwordCompleta)
```

```
PROBLEMI OUTPUT CONSOLE DI DEBUG TERMINALE PORTE
● pagizza@MacBook-APPLE ~ % python3 -u "/Users/pagizza/Code/W7D1Facoltativo.py"
Epicode Password Generator
*****
L'Epicode Password Generator è un generatore di password
A seconda della tua scelta, può generare password deboli o password forti
*****
Una password debole è composta da 8 caratteri
Una password forte è composta da 20 caratteri
*****
Digita 1 sulla tastiera per generare una password debole o 2 per generare una password forte 2
*****
Ecco la tua password: |g0,0h?5cjD8f/2fprrg
○ pagizza@MacBook-APPLE ~ %
```