

## Generatore password

```
1 import string
2 import random
3
4 # Definizione della funzione che genera la password
5 def genera_password(complexita):
```

### 1. Argomento complessita:

- La funzione prende un parametro **complexita**, che rappresenta il livello di complessità desiderato per la password.

```
7     # Impostazione della lunghezza della password a 8 se semplice, o a 20 se complicata
8     lunghezza = 8 if complexita == 'semplice' else 20
9
```

### 2. Determinazione della Lunghezza:

- La variabile **lunghezza** viene impostata a 8 se **complexita** è 'semplice', altrimenti a 20, attraverso la l'operazione ternario.

```
10    # Caratteri consentiti in base al livello di complessità
11    caratteri = string.ascii_letters + string.digits
12
```

### 3. Caratteri Consentiti:

- La variabile **caratteri** contiene inizialmente lettere maiuscole e minuscole (**string.ascii\_letters**) e cifre (**string.digits**).
- Se **complexita** è 'complicata', vengono aggiunti anche caratteri di punteggiatura (**string.punctuation**).

```
15
16    # Generare la password selezionando casualmente i caratteri dalla stringa di caratteri consentiti
17    password = ''.join(random.choice(caratteri) for _ in range(lunghezza))
```

### 4. Generazione della Password:

- La password viene generata selezionando casualmente i caratteri dalla stringa di **caratteri** consentiti.
- Utilizza la funzione **random.choice** in un ciclo per selezionare casualmente i caratteri.

```
18    return password
```

### 5. Restituzione della Password:

- La password generata viene restituita dalla funzione.

```
20 # Chiedere all'utente il livello di complessità desiderato
21 livello_complessita = input("Vuoi una password semplice o complicata? ").lower()
22
```

Input dell'Utente:

Chiede all'utente di specificare se desidera una password semplice o complicata.

```
23 # Verifica che l'utente abbia inserito una risposta valida
24 while livello_complessita not in ['semplice', 'complicata']:
25     print("Per favore, rispondi con 'semplice' o 'complicata'.")
26     livello_complessita = input("Vuoi una password semplice o complicata? ").lower()
27
28
```

Verifica di Validità della Risposta:

Verifica che la risposta dell'utente sia valida ('semplice' o 'complicata').

Se la risposta non è valida, richiede all'utente di inserirla nuovamente.

```
# Generare e stampare la password
password_generata = genera_password(livello_complessita)
```

Chiamata della Funzione:

Chiama la funzione `genera_password` con il livello di complessità specificato dall'utente.

```
print("La tua password generata è:", password_generata)
```

Stampa della password:

Stampa la password generata a schermo.

```
Vuoi una password semplice o complicata? semplice
La tua password generata è: dSFNcH8W
```

```
Vuoi una password semplice o complicata? complicata
La tua password generata è: Q.FAS=m'=)#K{d-wcHU+
```

Risultato delle password, sia in semplice e complicato.

## Conclusione:

Il programma chiede all'utente il livello di complessità desiderato per la password, verifica che la risposta sia valida e genera quindi una password in base a tale risposta. Infine, stampa la password generata a schermo. Il codice sembra funzionare correttamente.