

Richiesta

```
1 import datetime
2
3 while True:
4     comando_utente = input("cosa vuoi sapere? ")
5     if comando_utente == "esci":
6         print("Arrivederci!")
7         break
8     else:
9         print(assistente_virtuale(comando_utente))
10
11 def assistente_virtuale(comando):
12     if comando == "Qual è la data di oggi?":
13         oggi = datetime.datetime.today()
14         risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
15     elif comando == "Che ore sono?":
16         ora_attuale = datetime.datetime.now().time()
17         risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")
18     elif comando == "Come ti chiami?":
19         risposta = "Mi chiamo Assistente Virtuale"
20     else:
21         risposta = "Non ho capito la tua domanda."
22
23     return risposta
```

- 1) Capire cosa fa il programma senza eseguirlo.
- 2) Individuare nel codice sorgente le casistiche non standard che il programma non gestisce (esempio, comportamenti potenziali che non sono stati contemplati).
- 3) Individuare eventuali errori di sintassi / logici.
- 4) Proporre una soluzione per ognuno di essi.

Relazione

- 1) Il programma funziona come un semplice assistente testuale: continua a chiedere all'utente cosa vuole sapere e, in base alla risposta, fornisce alcune informazioni già previste, come la data di oggi, l'ora attuale o il proprio nome. Se la domanda non rientra tra quelle riconosciute, comunica di non aver capito. L'esecuzione prosegue finché l'utente non scrive "esci", momento in cui il programma saluta ed è terminato.
- 2) Il programma tiene conto solo di poche domande molto precise, quindi tutto ciò che esce da quelle non viene riconosciuto. Non comprende nessuna variazione di scrittura (maiuscole, minuscole ecc.). Non prevede nemmeno casi come input vuoti, si limita a tre funzioni molto precise.
- 3) Errore di sintassi alla riga 2: il ciclo while True è scritto senza i due punti finali.
Soluzione --> while True:

Errore di posizione alla riga 7: l'else: break non è incluso nell'if e quindi chiuderebbe sempre il programma a prescindere dall'input.

Soluzione: allineare il break al print

```
if comando_utente == "esci":
    print("Arrivederci!")
    break
```

Errore di sintassi a riga 13: la funzione datetime.datetoday è scritta nel modo sbagliato
Soluzione --> datetime.date.today

Errore di posizione della funzione: il def assistente_virtuale (comando) definisce una funzione pertanto bisognerebbe posizionarlo prima di utilizzare la funzione in questione.

Soluzione --> spostare il blocco def sopra il blocco while true