

CORREZIONE CODICE PYTHON S2L5

```
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.now()
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
```

Il codice crea un assistente virtuale che risponde a 3 domande :

- 1) qual è la data di oggi ?
- 2) che ore sono?
- 3) come ti chiami?

a queste tre domande il codice deve rispondere:

- 1) Oggi è (data di oggi)
- 2) l'ora attuale è (ora attuale)
- 3) mi chiamo Assistente Virtuale

Questo codice ha vari problemi che vanno da limitazioni per l'utente a errori veri e propri di sintassi.

ELENCO ERRORI

1)

Tra gli errori principali troviamo il fatto che il codice non ammette errori da parte dell' utente poichè usa solo ==.

Per renderlo più efficace si potrebbe modificare l'input appena viene ricevuto convertendo tutto in minuscolo e rimuovendo eventuali spazi iniziali e finali inseriti dall'utente.

ESEMPIO:

Codice originale RIGA 4- 5

```
comando_utente = input("cosa vuoi sapere?")
```

Soluzione

```
comando = input("Cosa vuoi sapere? ").strip().lower()
```

2)

Sempre a causa dell'uguaglianza == l'utente non può chiedere niente che non sia esattamente la frase accettata.

Si può migliorare questa cosa usando parole chiave all'interno della frase con (in).

Oppure dicendo all' utente esattamente cosa scrivere per avere risposta.

ESEMPIO:

Errore = Codice originale RIGA 12 if comando == "Qual è la data di oggi?":

Soluzione = if "data" in comando or "giorno" in comando:

3)

Errore = Il ciclo while da riga 3 a riga 9 viene definito a riga 11 quindi python non sa ancora cosa sia.

Soluzione = spostare la definizione def assistente_vocale_virtuale(comando): prima del ciclo while, dopo (import datetime)

4)

Errore = Alla riga 13 vi è scritto il metodo datetime.datetoday() ,Questo metodo non esiste.

Soluzione = Il metodo corretto è datetime.date.today()

5)

C'è una mancanza di presentazione iniziale dove il programma non spiega niente neanche cosa fa.

Soluzione: inserire frasi di presentazione e di cosa può fare l'assistente virtuale

```
print("Benvenuto\nPuoi chiedermi l'ora, la data o come mi chiamo.")
```

while True:

6)

Riga 3 = Assenza dei due punti : dopo il True.

7)

Riga 7 = indentazione del break scorretta.

Soluzione = mettere il break in linea con la riga 6 sotto il print, in questo modo si esce correttamente dal ciclo.

8)

Un miglioramento potrebbe essere di ricominciare il ciclo in caso di invio senza istruzioni.

Errore = se si preme invio senza aver scritto nulla il programma risponde non ho capito la tua domanda.

Soluzione = if not comando_utente:

```
    continue
```

9)

Errore = in caso di input errato il codice non suggerisce nessun tipo di aiuto all'utente.

Soluzione = si può scrivere un else con return : "non ho capito, se scrivi aiuto ti elencherò i comandi."

poi inserire input dove l'utente, se scrive aiuto il programma ripete cosa può fare.

CODICE CORRETTO

```
py esercizio S2L5.py ✘ ✘ |  
1 import datetime  
2  
3  
4 def assistente_virtuale(comando):  
5  
6  
7     if "data" in comando or "giorno" in comando:  
8         oggi = datetime.date.today()  
9         return f"La data di oggi è: {oggi.strftime('%d/%m/%Y')}"  
10  
11    elif "ora" in comando or "orario" in comando:  
12        ora_attuale = datetime.datetime.now().time()  
13        return f"L'ora attuale è: {ora_attuale.strftime('%H:%M')}"  
14  
15    elif "chi sei" in comando or "ti chiami" in comando or "nome" in comando:  
16        return "Mi chiamo Assistente Virtuali."  
17  
18    elif "aiuto" in comando or "help" in comando or "comandi" in comando:  
19        return ("Comandi disponibili:\n"  
20                 "- Chiedi l'ora (es. 'Che ora è?')\n"  
21                 "- Chiedi la data (es. 'Che giorno è oggi?')\n"  
22                 "- Chiedi il mio nome\n"  
23                 "- Scrivi 'esci' per chiudere.")  
24  
25    else:  
26        return "Non ho capito. Prova a scrivere 'aiuto' per vedere i comandi."  
27  
28  
29 while True:  
30  
31     comando_utente = input("\nCosa vuoi sapere? ").strip().lower()  
32  
33     if not comando_utente:  
34         continue  
35  
36     if comando_utente in ["esci", "stop", "basta", "exit"]:  
37         print("Arrivederci! Alla prossima.")  
38         break  
39  
40     risposta = assistente_virtuale(comando_utente)  
41  
42     print(f">> {risposta}")
```

