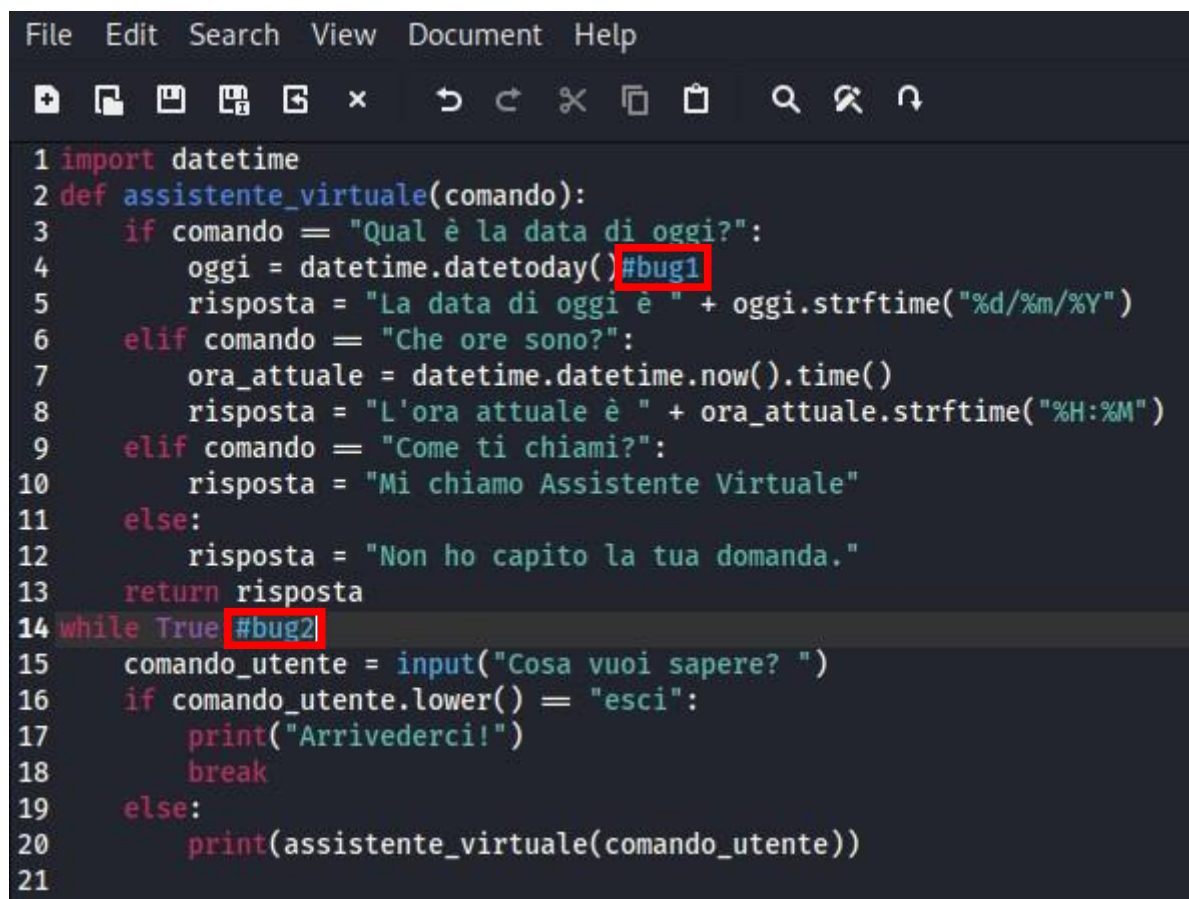


Dato il codice si richiede allo studente di:

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

Il programma simula un'assistente virtuale a cui chiedere determinate informazioni. Nella figura sottostante sono riportati i bug di sintassi riscontrati:



```
File Edit Search View Document Help
[Icons]
1 import datetime
2 def assistente_virtuale(comando):
3     if comando == "Qual è la data di oggi?":
4         oggi = datetime.datetime.today() #bug1
5         risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
6     elif comando == "Che ore sono?":
7         ora_attuale = datetime.datetime.now().time()
8         risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")
9     elif comando == "Come ti chiami?":
10        risposta = "Mi chiamo Assistente Virtuale"
11    else:
12        risposta = "Non ho capito la tua domanda."
13    return risposta
14 while True #bug2
15     comando_utente = input("Cosa vuoi sapere? ")
16     if comando_utente.lower() == "esci":
17         print("Arrivederci!")
18         break
19     else:
20         print(assistente_virtuale(comando_utente))
21
```

*bug1* – Nella riga 4, è assente il "." tra le parole *date* e *today*.

*bug2* – Nella riga 14, sono assenti i ":" che in un ciclo devono essere sempre presenti.

Altra considerazione da fare, è quella sulle istruzioni per l'utente; esse, infatti, sono mancanti. Invero, se si avvia il programma, l'assistente chiede subito "*cosa vuoi sapere?*". In questo caso si può presupporre che si possa chiedere qualsiasi cosa all' assistente virtuale. Ma dando una prima occhiata al codice, ci si accorge subito che le informazioni date in risposta dall' assistente virtuale sono limitate alla data, all' orario e a fornire il suo nome.

Per tanto, per evitare di avere come risposta automatica impostata "*non ho capito la tua domanda*", ma bensì una delle risposte positive impostate dal codice; si dovranno inserire delle istruzioni per l'utente con delle domande ben precise da porre attraverso l'uso del comando *print*. Nell' immagine sottostante, è possibile notare dove sono state inserite le istruzioni:

```
1 print("Ciao! Sono il tuo assistente virtuale, puoi farmi le seguenti domande:")
2 print("Qual è la data di oggi?")
3 print("Che ore sono?")
4 print("Come ti chiami?\n")
5
6
7 import datetime
8 def assistente_virtuale(comando):
9     if comando == "Qual è la data di oggi?":
10         oggi = datetime.date.today()#bug1
11         risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
12     elif comando == "Che ore sono?":
13         ora_attuale = datetime.datetime.now().time()
14         risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")
15     elif comando == "Come ti chiami?":
16         risposta = "Mi chiamo Assistente Virtuale"
17     else:
18         risposta = "Non ho capito la tua domanda."
19     return risposta
20
21 while True:#bug2
22     comando_utente = input("Cosa vuoi sapere?\n")
23     if comando_utente.lower() == "esci":
24         print("Arrivederci!")
25         break
26     else:
27         print(assistente_virtuale(comando_utente))
28
```