

# Report S2 L5

```
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     return risposta
23
```

## Errori di sintassi

### RIGA 3

- **Errore:** mancano : a fine frase
- **Versione corretta:** while true:

### RIGA 7

- **Errore:** break indentato male
- **Versione corretta:** indentarlo nell'IF

## RIGA 9

- **Errore:** `assistente_virtuale` non è stato definito
- **Versione corretta:** inserire il **while** in un **main()** e richiamarlo alla fine del codice oppure spostare il **while** dopo la funzione **assistente\_virtuale**

## RIGA 13

- **Errore:**  
`datetime.datetoday()`
- **Versione corretta:**
  - avendo importato la libreria con il comando **"import datetime"** il comando corretto è **datetime.datetime**, inoltre il comando `datetoday` non esiste.
  - la stringa corretta è **datetime.datetime.today**

## Errori logici

Sono presenti vari errori logici e di user experience che fanno comunque funzionare il programma ma rendono più probabile la presenza di errori

## RIGA 5

- **Problema:** `if comando_utente == "esci"`: Non viene indicato da nessuna parte il che per terminare il programma bisogna digitare **"esci"**
- **Soluzione:** Inserire un print tra **riga 4 e riga 5** che comunica all'utente il comando, ad esempio `print("Per terminare il programma scrivere 'esci'")`

**RIGA 12,15,18**

- **Problema:** essendo stringhe molto specifiche è facile che l'utente sbagli a digitare dimenticando una maiuscola o un segno di punteggiatura oppure scrivere parole in modo errato
- **Soluzione:**
  - **Possibilità 1.** aggiungere agli input `.lower` e `.strip` alla fine per formattare i caratteri in minuscolo e rimuovere la punteggiatura per uniformare i comando
  - **Possibilità 2.** creare una lista dove vengono fatte corrispondere le domande ad un numero per esempio
    - 1 - Qual è la data di oggi?
    - 2 - Che ore sono?
    - 3 - Come ti chiami?

in seguito rendere il comando **if comando == "1" :**  
**if comando == "2" :** **if comando == "3" :**

Una volta modificato il codice l'utente ha solo 3 scelte e 1 di 3 caratteri da digitare rendendo molto inferiore la possibilità di sbagliare