

## ESERCIZIO S2-L5/VENERDI'

### TRACCIA:

***Dato il codice si richiede allo studente di:***

- 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.***

Per eseguire questo esercizio, iniziamo analizzando il codice per capire prima di tutto qual è la funzione che svolge il programma in python:

```
import datetime

def assistente_virtuale(comando):

    if comando == "Qual è la data di oggi?":

        oggi = datetime.datetime.today()

        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")

    elif comando == "Che ore sono?":

        ora_attuale = datetime.datetime.now().time()

        risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")

    elif comando == "Come ti chiami?":

        risposta = "Mi chiamo Assistente Virtuale"

    else:

        risposta = "Non ho capito la tua domanda."

    return risposta

while True

    comando_utente = input("Cosa vuoi sapere? ")

    if comando_utente.lower() == "esci":

        print("Arrivederci!")

        break

    else:

        print(assistente_virtuale(comando_utente))
```

Leggendo il codice, possiamo constatare che ci fornisce un programma che simula un assistente virtuale che risponde ad alcuni specifici comandi dell'utente, come la data corrente, l'ora attuale, e il nome dell'assistente.

Per prima cosa andiamo a vedere cosa fa nel dettaglio il programma:

Il programma chiede all'utente un input (tramite `input("Cosa vuoi sapere? ")`).

Se l'utente scrive "esci", il programma risponde con "Arrivederci!" e termina.

Se l'utente dà uno dei seguenti comandi:

- "Qual è la data di oggi?": il programma risponde con la data corrente.
- "Che ore sono?": il programma risponde con l'ora corrente.
- "Come ti chiami?": il programma risponde con il nome "Assistente Virtuale".

Se il comando non corrisponde a nessuna delle casistiche previste, il programma risponde con "Non ho capito la tua domanda."

Prima di andare ad analizzare eventuali errori logici o di sintassi, l'esercizio ci chiede di Individuare nel codice sorgente le casistiche non standard che il programma non gestisce (esempio, comportamenti potenziali che non sono stati contemplati).

Se l'utente scrive un comando che non è "Qual è la data di oggi?", "Che ore sono?", o "Come ti chiami?", il programma risponde con "Non ho capito la tua domanda."

Sarebbe utile gestire anche comandi con spazi in più o in minuscolo/maiuscolo.

In questi casi, potrebbe essere utile normalizzare l'input dell'utente (ad esempio, usando **`comando_utente.lower()` o `.strip()` per rimuovere gli spazi superflui**).

Per migliorare il programma, inoltre, sarebbe utile aggiungere una gestione degli errori, nel caso in cui l'utente inserisca comandi che

non siano del tipo che ci aspettiamo (ad esempio, "Che ora è?" anziché "Che ore sono?").

```
def main():
    comandi_possibili = [
        "Qual è la data di oggi?",
        "Che ore sono?",
        "Come ti chiami?"
    ]

    while True:
        comando_utente = input("Cosa vuoi sapere? ").strip().lower()

        if comando_utente == "esci":
            print("Arrivederci!")
            break
        elif comando_utente not in [comando.lower() for comando in comandi_possibili]:
            print("Non ho capito la tua domanda. Ecco alcuni comandi che posso capire:")
            print("- Qual è la data di oggi?")
            print("- Che ore sono?")
            print("- Come ti chiami?")
        else:
            print(assistente_virtuale(comando_utente))
```

Passiamo agli errori di **sintassi e/o logici**.

Il primo errore di sintassi che notiamo è la funzione `datetime.datetoday()`.

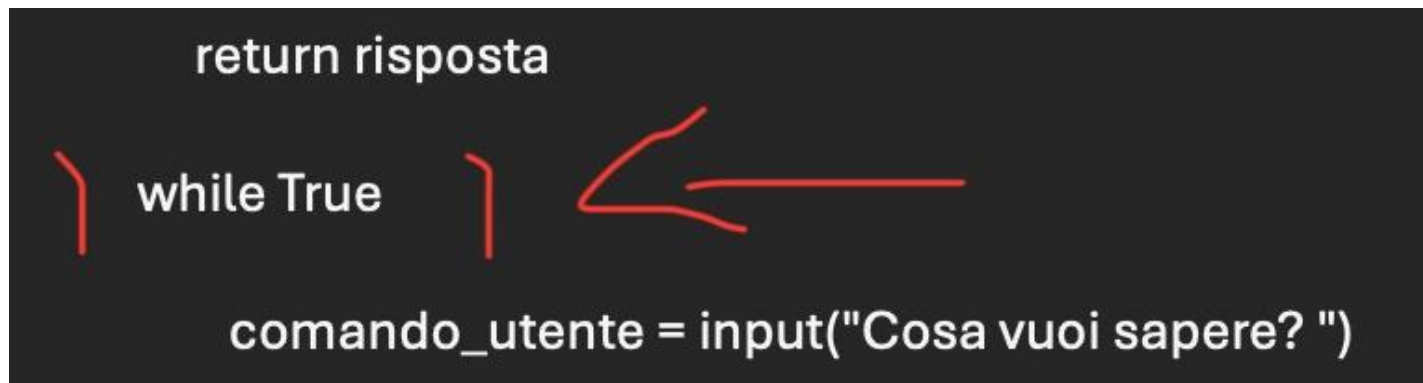
```
if comando == "Qual è la data di oggi?":
```

```
| oggi = datetime.datetoday() |
```

```
risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
```

Il metodo corretto per ottenere la data corrente è `datetime.date.today()`.

Quindi, per correggere questo errore di sintassi, dobbiamo sostituire **`datetime.datetime.today()`** con **`datetime.date.today()`**.  
Un altro errore visibile è presente nel ciclo while, dopo **`while true`** infatti, mancano i **due punti “:”** alla fine; quindi, il codice non è sintatticamente corretto.



```
return risposta  
while True  
comando_utente = input("Cosa vuoi sapere? ")
```

The image shows a snippet of Python code on a dark background. The code is: `return risposta`, `while True`, and `comando_utente = input("Cosa vuoi sapere? ")`. Red annotations highlight syntax errors: a red bracket on the left of `while True` and a red arrow pointing to the missing colon at the end of the `while True` line.

In questo caso è sufficiente aggiungere i due punti per correggere l'errore di sintassi.

Sembrano esserci inoltre, alcuni problemi di indentazione.

- La struttura del ciclo while non è indentata in modo corretto, in particolare la parte che riguarda l'input dell'utente e il controllo del comando "esci".

- L'indentazione delle linee all'interno del ciclo while e della funzione `assistente_virtuale` non è corretta. Le linee dopo `if` o `else` devono essere correttamente indentate.

Per correggere, è sufficiente allineare correttamente l'indentazione all'interno delle funzioni e dei blocchi condizionali.

Nelle prossime immagini possiamo vedere in **rosso** le correzioni applicate al programma.

```

import datetime

def assistente_virtuale(comando):
    if comando == "Qual è la data di oggi?":
        oggi = datetime.date.today() # Correzione: datetime.date.today()
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")

    while True: # Correzione: aggiunto il ":" alla fine di "while True"
        comando_utente = input("Cosa vuoi sapere? ").strip().lower() # Correzione: normalizzato
        l'input
        if comando_utente == "esci":
            print("Arrivederci!")
            break
        else:
            print(assistente_virtuale(comando_utente))

```

## RIEPILOGO DELLE CORREZIONI:

**Corretta chiamata a `datetime.date.today()`.**

**Corretta indentazione** del codice

**Aggiunto `strip()`** per rimuovere spazi superflui **e `lower()`** per rendere il confronto insensibile alla maiuscolizzazione.

**Sintassi corretta per `while True:`**, con l'aggiunta dei due punti alla fine della dichiarazione.

## QUALI MODIFICHE SONO STATE EFFETTUATE PER MIGLIORARE IL PROGRAMMA?

È stato aggiunto un confronto semplice, se l'input dell'utente non corrisponde esattamente a uno dei comandi predefiniti, il programma restituisce un messaggio di errore con la lista dei comandi possibili.

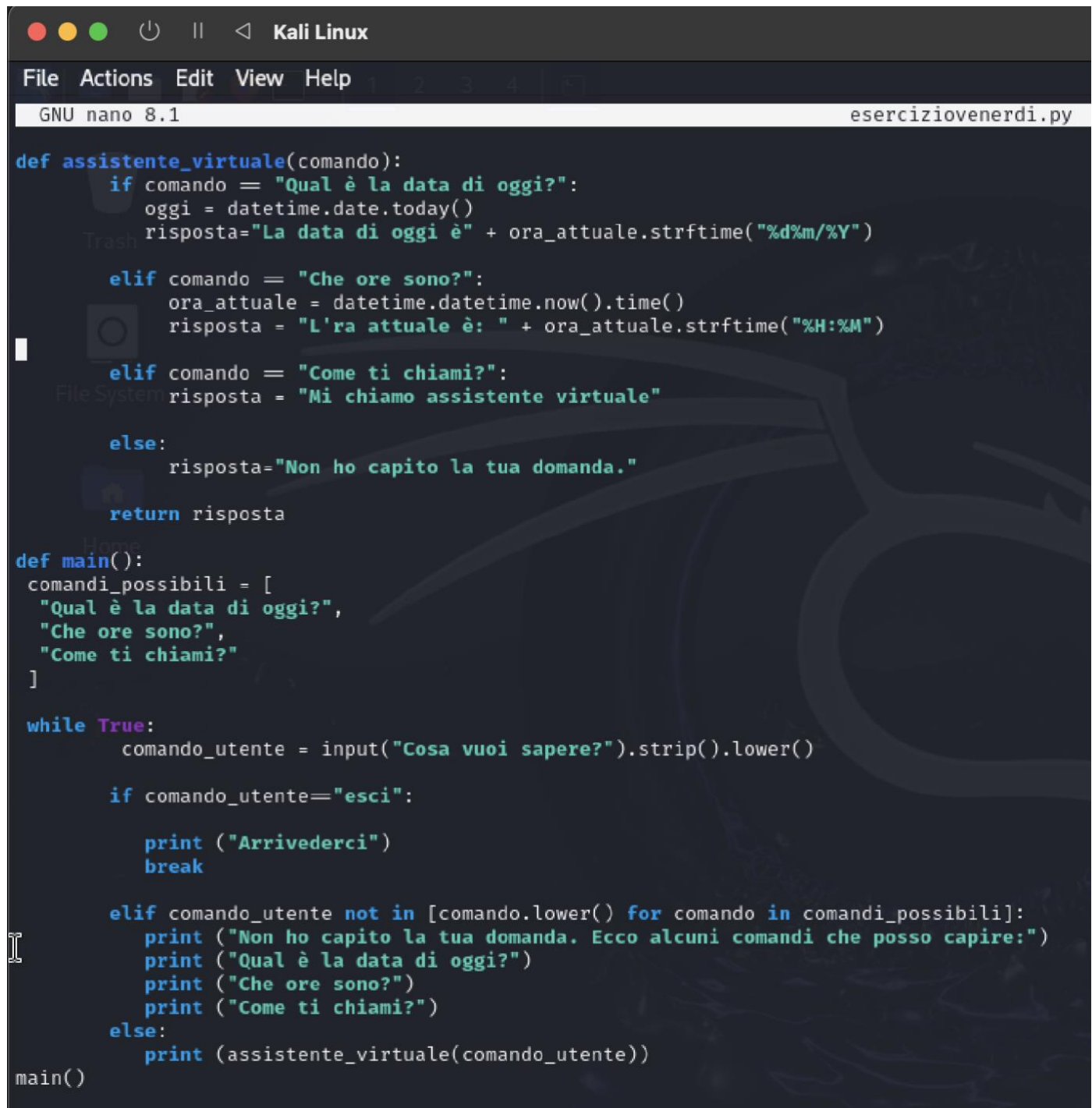
**Messaggio generico:**

Quando il comando non è riconosciuto, il programma fornisce una lista di comandi che l'utente può usare.

**Normalizzazione dell'input:**

L'input viene sempre convertito in minuscolo con **.lower()** e rimosso eventuali spazi extra con **.strip()**, per evitare problemi legati a maiuscole/minuscole o spazi aggiuntivi.

Nell'immagine seguente, il codice completo dopo le modifiche:



```
def assistente_virtuale(comando):
    if comando == "Qual è la data di oggi?":
        oggi = datetime.date.today()
        risposta="La data di oggi è" + ora_attuale.strftime("%d%m/%Y")

    elif comando == "Che ore sono?":
        ora_attuale = datetime.datetime.now().time()
        risposta = "L'ra attuale è: " + ora_attuale.strftime("%H:%M")

    elif comando == "Come ti chiami?":
        risposta = "Mi chiamo assistente virtuale"

    else:
        risposta="Non ho capito la tua domanda."

    return risposta

def main():
    comandi_possibili = [
        "Qual è la data di oggi?",
        "Che ore sono?",
        "Come ti chiami?"
    ]

    while True:
        comando_utente = input("Cosa vuoi sapere?").strip().lower()

        if comando_utente=="esci":

            print ("Arrivederci")
            break

        elif comando_utente not in [comando.lower() for comando in comandi_possibili]:
            print ("Non ho capito la tua domanda. Ecco alcuni comandi che posso capire:")
            print ("Qual è la data di oggi?")
            print ("Che ore sono?")
            print ("Come ti chiami?")
        else:
            print (assistente_virtuale(comando_utente))

main()
```