

Cosa fa il programma?

A prima vista, il codice presenta un'unica funzione che gestisce una sorta di assistente virtuale, al quale chiedere alcune domande e lui restituirà una stringa con una risposta.

Ci sono un paio di casistiche fatte con degli "if, elif, else" che analizzano e gestiscono le varie domande che possono essergli fatte.

È presente un While loop, quindi, è intuibile che il codice venga eseguito all'infinito fino a quando l'utente non desidera di terminare con la sua esecuzione.

Gli input dell'utente vengono inseriti all'interno della variabile "comando_utente" e successivamente analizzati.

Dentro la funzione "assistente_vocale" viene preso l'input dell'utente e viene sottoposto a tre condizioni: "Qual è la data di oggi?"; "Che ore sono?"; "Come ti chiami?" e nel caso sia una di queste tre, verrà data informazione, altrimenti dirà che non ha capito la domanda

Quali sono le caratteristiche "non standard"?

Ci sono alcune caratteristiche "non standard" che il programma non esegue come la corretta gestione degli input utente che potrebbero avere diverse combinazioni di lettere maiuscole o minuscole e la mancata comunicazione di quali input è possibile inserire, come ad esempio le possibili domande da poter fare all'assistente o anche la possibilità di interrompere l'esecuzione del programma.

Sono presenti degli errori?

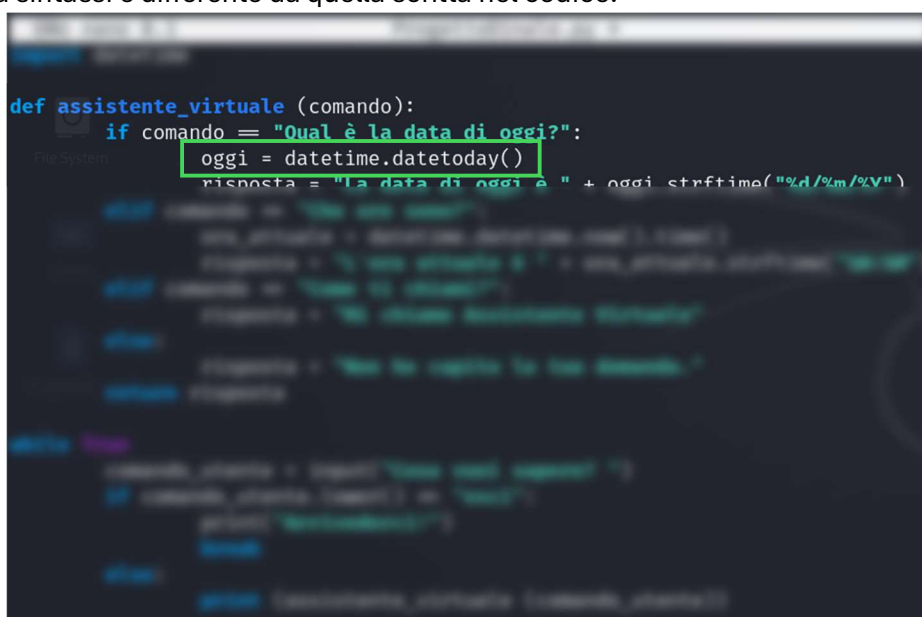
Errori sintattici?

Per trovarli sono andato ad analizzare in maniera più approfondita il codice per vedere se fossero presenti degli errori di tipo sintattico o logico.

Come prima cosa, mi sono documentato sulla documentazione ufficiale nel sito "docs.python.org", e ho notato che nella sezione che parla del modulo "datetime" (<https://docs.python.org/3/library/datetime>) c'è una parte che cita proprio il metodo che restituisce la data locale corrente:

```
classmethod date.today()
    Return the current local date.
```

e indica che la sintassi è differente da quella scritta nel codice:



```
def assistente_virtuale (comando):
    if comando == "Qual è la data di oggi?":
        oggi = datetime.datetoday()
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
        print(risposta)
    elif comando == "Che ore sono?":
        ora = datetime.datetime.now().strftime("%H:%M")
        risposta = "Le ore sono " + ora
        print(risposta)
    elif comando == "Come ti chiami?":
        nome = input("Inserisci il tuo nome: ")
        risposta = "Mi ha risposto il tuo nome."
        print(risposta)
    else:
        print("Non ho capito la tua domanda.")
    print()
    comando_utente = input("Inserisci la tua domanda: ")
    if comando_utente == "":
        print("Fine")
    else:
        assistente_virtuale(comando_utente)
```

Comportando una segnalazione di errore nel caso in cui si dovesse entrare in questa condizione.

Il secondo errore sintattico che ho riscontrato è stata una mancanza dei ":" nel ciclo while che gestisce il loop del codice fino a quando non viene chiesto dall'utente:

```
def assistente_virtuale (comando):  
    if comando == "Qual è la data di oggi?":  
        oggi = datetime.datetime.now().date()  
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")  
    elif comando == "Che ora è?":  
        ora_attuale = datetime.datetime.now().time()  
        risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")  
    elif comando == "Salve tu, assistente":  
        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))
```

Errori logici?

Nella sezione del ciclo while che gestisce l'uscita e successiva interruzione del programma viene convertito il testo in testo minuscolo, questa caratteristica non è presente nel caso in cui si inserisca una stringa diversa da "esci" e la condizione vada nell'else.

Questo potrebbe portare ad una errata esecuzione dato che, se l'utente dovesse inserire una stringa non esattamente scritta come quelle indicate nei possibili casi della funzione, come ad esempio la lettera iniziale maiuscola o la "e" non accentata, verrà indicata la mancata comprensione da parte dell'assistente virtuale.

```
def assistente_virtuale (comando):  
    if comando == "Qual è la data di oggi?":  
        oggi = datetime.datetime.now().date()  
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")  
    elif comando == "Che ora è?":  
        ora_attuale = datetime.datetime.now().time()  
        risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")  
    elif comando == "Salve tu, assistente":  
        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))
```

Soluzioni:

dopo aver corretto gli errori sintattici nel modo descritto precedentemente, per migliorare il codice, lo renderei “a prova di utente” dato che potrebbero essere inseriti un grande numero di possibili input errati.

Inizierei col comunicare all’utente le possibili operazioni che può eseguire il nostro assistente virtuale, aggiungendo la frase:

<<Non ho capito la tua domanda, prova a chiedermi ‘che ore sono?’, ‘qual è la data di oggi?’ o ‘come ti chiami?’>>

così da portare l’utente a inserire un corretto input richiesto dalla funzione.

```
def assistente_virtuale (comando):  
    if comando == "qual è la data di oggi?" or comando == "qual e la data di oggi?":  
        oggi = datetime.date.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. \nProva a chiedermi: \n'qual e' la data di oggi?' 'che ore sono?' 'come ti chiami?'"  
  
    return risposta
```

Per risolvere la mancanza di comunicazione all’utente della possibilità di interrompere l’esecuzione del programma tramite il comando “esci”, basta comunicarlo ad inizio dell’esecuzione:

```
while True:  
    comando_utente = input("Cosa vuoi sapere? \nscrivi 'esci' per uscire dal programma\n").lower()  
    if comando_utente == "esci":  
        print("Arrivederci!")  
        break  
    else:  
        print (assistente_virtuale (comando_utente))
```

Per poter gestire il problema del possibile inserimento di stringhe diverse da quelle dentro la funzione, potremmo innanzitutto trasformare in minuscolo tutto la testo immediatamente dopo l’inserimento dall’utente inserendo “.lower”:

```
while True:  
    comando_utente = input("Cosa vuoi sapere? \nscrivi 'esci' per uscire dal programma\n").lower()  
    if comando_utente == "esci":  
        print("Arrivederci!")  
        break  
    else:  
        print (assistente_virtuale (comando_utente))
```

E successivamente modificare le stringhe che vengono accettate dai casi della funzione dagli “if, elif” in stringhe tutte in minuscolo (così come verranno convertite dopo l’inserimento dell’utente):

```
def assistente_virtuale (comando):
    if comando == "qual è la data di oggi?" or comando == "qual e la data di oggi?":
        oggi = datetime.date.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. \nProva a chiedermi: \n'qual e' la data di oggi?' 'che ore sono?' 'come ti chiami?'"

    return risposta
```

Per risolvere il problema della possibile “e” non accentata, aggiungerei un operatore logico “OR” per poter inserire delle varianti alla frase “Qual è la data di oggi?”, come: “qual e la data di oggi?”

```
def assistente_virtuale (comando):
    if comando == "qual è la data di oggi?" or comando == "qual e la data di oggi?":
        oggi = datetime.date.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. \nProva a chiedermi: \n'qual e' la data di oggi?' 'che ore sono?' 'come ti chiami?'"

    return risposta
```