

Esercizio S2/L5

Traccia:

Per agire come un Hacker bisogna capire come pensare fuori dagli schemi.

L' esercizio di oggi ha lo scopo di allenare l'osservazione critica. Dato il codice si richiede allo studente di:

- Capire cosa fa il programma senza eseguirlo.
- Individuare nel 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.

COSA FA IL PROGRAMMA?

Il programma **simula** un Assistente virtuale scritto in **Python** che risponde solo a 3 domande:

- “Qual è la data di oggi?”
- “Che ore sono?”
- “Come ti chiami?”

Se vengono poste domande differenti da queste, l'Assistente risponderà “Non ho capito la domanda”.

Il ciclo **while True** permette all'utente di porre domande illimitate fino a quando non digita ‘esci’.

CASISTICHE NON STANDARD NON GESTITE

Il programma presenta alcune logiche che lo rendono **molto rigido**:

- richiede una **corrispondenza esatta** (punteggiatura inclusa). Ad esempio se l'utente scrive “che ore sono” (con la c minuscola e senza punto interrogativo), il programma risponde “Non ho capito la domanda”
- l'utente non viene informato sui comandi validi

ERRORI DI SINTASSI/LOGICI

Riga	Tipo di errore	Descrizione e soluzione
3	Sintassi	Alla fine dell'istruzione while True mancano i : Soluzione: while True:
7	Logico	Il break è indentato all'interno del while , quindi viene eseguito dopo il primo input, interrompendo subito il programma Soluzione: indentare il break all'interno dell if dato che è lui a gestire l'uscita
9 e 11	Logico	La funzione assistente_virtuale viene chiamata a riga 9, ma è definita a riga 11. Ciò significa che python non conosce la funzione quando viene chiamata. Soluzione: spostare la definizione della funzione a inizio file, sopra il ciclo while
13	Sintassi/Logico	Qui sono presenti 2 errori. La funzione datetoday() non esiste nel modulo datetime . Anche se datetoday() fosse esistito, la sintassi datetime.datetoday() è ugualmente sbagliata. Soluzione: la funzione corretta per ottenere la data di oggi è datetime.datetime.today() oppure cambiare l'import in from datetime import datetime per poter utilizzare datetime.today()

Altre migliorie che si possono applicare al codice sono:

- menu delle istruzioni stampato una sola volta all'inizio
- rendere l'input più flessibile con l'aggiunta di **.lower()** o **.upper()** e **.strip()**
- per gestire la rigidità di == negli **if**, si può usare l'operatore **in** (es. **if comando in 'Qual è la data di oggi?'**)