

# Report

**Oggetto: Valutazione e correzioni sul codice “Assistente digitale”**

## 1.Introduzione

Il presente report si focalizza sull'analisi e la valutazione del codice Python sottostante che implementa un rudimentale Assistente Virtuale. L'obiettivo primario è quello di esaminare la struttura del codice, le funzionalità e l'identificazione degli eventuali errori, le limitazioni logiche e le aree di potenziale miglioramento implementabili nel codice fornito.

In questa fase iniziale, l'analisi si concentrerà sullo stato attuale del codice, evidenziando i suoi aspetti funzionali e, gli eventuali errori, le limitazioni logiche presenti in esso. Successivamente, si potranno considerare le possibili strategie di correzione e miglioramento per superare le criticità riscontrate durante l'analisi.

In seguito, verrà mostrato il codice così come è stato fornito per iniziare a dare un'analisi approfondita, il codice viene visualizzato tramite l'interprete di Pycharm.

```
import datetime
def assistente_virtuale(comando): 1 usage
    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))
```

## 2. Descrizione e Funzionalità del codice

Il codice così come è stato fornito implementa un semplice assistente virtuale con il quale si può interagire in maniera testuale tramite la riga di comando. La sua funzionalità principale consiste nel rispondere a tre domande specifiche relative alla data corrente, all'ora attuale e al proprio nome (quello dell'assistente).

Secondo il codice, le funzionalità del codice sono:

- **Richiesta della data corrente:** Quando l'utente digita la frase "Qual è la data di oggi?", il programma utilizza la libreria *datetime* di Python per recuperare la data odierna dal sistema operativo, la quale viene formattata utilizzando "+oggi.strftime("%d,%m,%")" per dare la formattazione della data in giorno/mese/anno.
- **Richiesta dell'ora Attuale:** Se viene digitato dall'utente precisamente "Che ore sono?", il programma acquisirà l'ora corrente dal sistema utilizzando sempre la stessa libreria citata sopra, venendo formattata in ore:minuti e stampata a schermo per l'utente.
- **Richiesta del Nome dell'Assistente:** In caso l'utente chieda "Come ti chiami?", l'assistente risponderà con una stringa predefinita "Mi chiamo Assistente Virtuale".
- **Gestione comandi non riconosciuti:** Per qualsiasi input da parte dell'utente che non siano le domande predefinite il programma risponderà con "Non ho capito la tua domanda." Ripetendo il ciclo.
- **Funzionalità di uscita:** L'utente può terminare l'interazione con il programma digitando la parola "esci" (indipendentemente dalla capitalizzazione, grazie alla conversione in minuscolo dell'input). In questo caso, il programma visualizza il messaggio "Arrivederci" e si chiude.

L'interazione con l'utente avviene in un ciclo continuo grazie al ciclo *while*: il programma attende l'input dell'utente, elabora il comando (se riconosciuto) e fornisce una risposta, per poi attendere un nuovo input, fino a quando l'utente non desidera uscire.

## 3. Analisi degli Errori e Limitazioni

Andando a guardare il codice così come è stato presentato nello screenshot è possibile trovare vari errori sia logici che strutturali affinché il programma funzioni correttamente:

- **Errori Sintattici:** Inizialmente, il codice presenta un codice di battitura nel metodo per ottenere la data corrente(*datetime.datetime.today()* anziché *datetime.date.today()*) e la mancanza dei due punti ( : ) alla fine dell'istruzione *while True*. Questi errori sintattici una volta corretti permetteranno la corretta esecuzione del programma.
- **Limitazione nella Corrispondenza dei Comandi:** L'assistente virtuale si basa su una corrispondenza esatta delle stringhe dei comandi. Variazioni linguistiche, sinonimi o ordini di parole diversi per esprimere la stessa richiesta non vengono contemplati e quindi non verranno riconosciuti dal programma. Ad esempio, "Che

data è oggi?" o "Dimmi l'ora" porterebbero alla risposta "Non ho capito la tua domanda".

- **Mancanza di Guida o Indicazione dei Comandi Disponibili:** L'utente che interagisce con l'assistente virtuale non ha modo di sapere quali sono i comandi validi che il sistema è in grado di comprendere. Non viene fornita alcuna istruzione, menù o suggerimento che fornisce quali comandi possono essere utilizzati. Ciò costringe l'utente ad indovinare o a fare tentativi, portando inevitabilmente alla risposta "Non ho capito la tua domanda".
- **Assenza di suggerimenti:** Il codice non implementa alcuna forma di elaborazione del linguaggio naturale, neanche a livello basilare (come la ricerca di parole chiave). Ciò limita significativamente la capacità di comprendere l'intento dell'utente al di là delle frasi predefinite.
- **Interazione Limitata:** L'assistente virtuale è limitato a rispondere a 3 domande specifiche e non mantiene alcuno stato o contesto conversazionale. Non è in grado di gestire domande più complesse, correlate o di fornire informazioni più complesse ed elaborate.
- **Limitazione dell'uso di `.lower()`:** L'input forzato della domanda così come è stata scritta nel codice limita l'utente a come deve scrivere, stando attento alla capitalizzazione della frase.

## 4. Conclusione

Il codice analizzato implementa una forma molto basilare di un assistente virtuale, capace di rispondere a un numero estremamente limitato di comandi predefiniti. Sebbene funzionale (una volta corretti gli errori) con questi comandi basilari, presenta significative limitazioni in termini di robustezza, usabilità e capacità di comprensione del linguaggio naturale. La dipendenza da una corrispondenza esatta delle stringhe rende l'interazione rigida e poco intuitiva. L'assenza di meccanismi di gestione degli errori e la natura statica dell'interazione ne limitano ulteriormente l'applicabilità in contesti più ampi.

## 5. Proposte di Miglioramento

Per superare le limitazioni identificate, si potrebbero considerare delle strategie di miglioramento, verrà proposto un codice con un esempio di migliorazioni utilizzando sempre il ciclo while, ma implementando anche un'interfaccia di menù.

- **Introduzione di una struttura a menù:** un menù eliminerebbe gli errori di digitazione e renderebbe l'interfaccia più chiara per l'utente.

```
import datetime

def menu(): 1 usage
    print("\nCosa vuoi sapere?")
    print("1 - Data di oggi")
    print("2 - Ora Attuale")
    print("3 - Nome dell'Assistente")
    print("4 - Esci")

def data(): 1 usage
    oggi=datetime.date.today()
    return f"La data di oggi è {oggi.strftime('%d/%m/%Y')}"

def ora(): 1 usage
    ora = datetime.datetime.now().time()
    return f"L'ora attuale è {ora.strftime('%H:%M')}"

def nome(): 1 usage
    return "Mi chiamo Assistente Virtuale"
```

linguistiche.

- Questo report fornisce una valutazione completa del codice fornito, evidenziando sia i suoi aspetti funzionali di base che le sue significative limitazioni e potenziali aree di miglioramento. Può servire come base per ulteriori discussioni e per l'implementazione di modifiche volte a rendere l'assistente virtuale più efficace e usabile.

- **Gestione degli errori:** Implementare blocchi try-except per gestire potenziali eccezioni e rendere il programma più robusto.
- **Espansione delle funzionalità:** Aggiungere la capacità di rispondere a un numero maggiore di domande e di eseguire azioni più complesse.
- **Miglioramento della gestione dell'input:** Considerare approcci più sofisticati per la normalizzazione dell'input utente, tenendo conto potenzialmente della capitalizzazione o di altre variazioni