

Progetto

Individuare gli errori all'interno di un codice.

Progetto S2/L5

```
1 import datetime
2
3 while True:
4     comando_utente = input("Cosa vuoi sapere? ")
5     if comando_utente == "esce!":
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
23 return risposta
```

L'immagine rappresenta il codice di un programma scritto in Python. Il programma dovrebbe permettere all'utente di interagire con un'assistente virtuale che in questo caso specifico dovrebbe calcolare la data e l'ora al momento della richiesta. Nella riga 1 vediamo infatti che è stata importata la libreria Python 'Datetime' tramite comando '**import datetime**'. Il modulo 'Datetime' serve per lavorare con date e orari, fornisce classi e funzioni per ottenere la data e l'ora correnti, creare date e orari personalizzati, formattare date e orari in stringhe leggibili.

Casistiche non standard.

Possiamo notare che all'interno di questo codice sorgente ci sono delle casistiche non standard che il programma non gestisce, ad esempio la funzione a partire dalla riga 11 dovrebbe consentire al programma di calcolare soltanto la data e l'ora come da richiesta utente ma troviamo anche richieste che non fanno riferimenti a data o ora e quindi il programma non le gestisce e va in errore.

Errori di sintassi/logici

- Riga 3 dopo il 'True' manca il carattere ':'

- Riga 5 mancano le parentesi () attorno ad ‘esci’
- Riga 7 errore di indentazione, il break è allineato con l’if ma dovrebbe stare all’interno, di conseguenza anche l’else successivo presenta un errore di indentazione
- riga 9 assenza di “ “ all’interno delle parentesi
- riga 12 mancano le parentesi
- riga 14 e 17 bisognava aggiungere il comando **print()** che ci permette di stampare il risultato che vorremmo ottenere.

Riporto di seguito come avrei scritto io il codice per renderlo più comprensibile ed efficace:

```
import datetime

def mostra_data():
    oggi = datetime.datetime.today()
    print('La data di oggi è:', oggi.strftime("%d/%m/%Y"))

def mostra_ora():
    adesso = datetime.datetime.now()
    print ("L'ora attuale è:", adesso.strftime("%H:%M:%S"))

def mostra_data_ora():
    adesso = datetime.datetime.now()
    print("Data e ora attuali:", adesso.strftime("%d/%m/%Y %H:%M:%S"))

print("\nScegli cosa vuoi visualizzare:")
print("1. Data")
print("2. Ora")
print("3. Data e ora")
print("0. Esci")

scelta = input("Inserisci la tua scelta (0-3): ")

if scelta == "1":
    mostra_data()
elif scelta == "2":
    mostra_ora()
elif scelta == "3":
```

```
    mostra_data_ora()  
elif scelta == "0":  
    print("Uscita dal programma. Arrivederci!")
```

Dopo aver importato la libreria ‘datetime’ ho scritto tre funzioni e assegnato ad ogni funzione una singola responsabilità. In questo modo il codice appare già più leggibile e nel caso riutilizzabile. Dopodichè tramite il comando print() ho fatto sì che si presentasse il menù con le opzioni che si possono calcolare e successivamente con i costrutti if -elife-else eseguiamo il codice in base al valore di una delle condizioni scelte affinché siano vere.