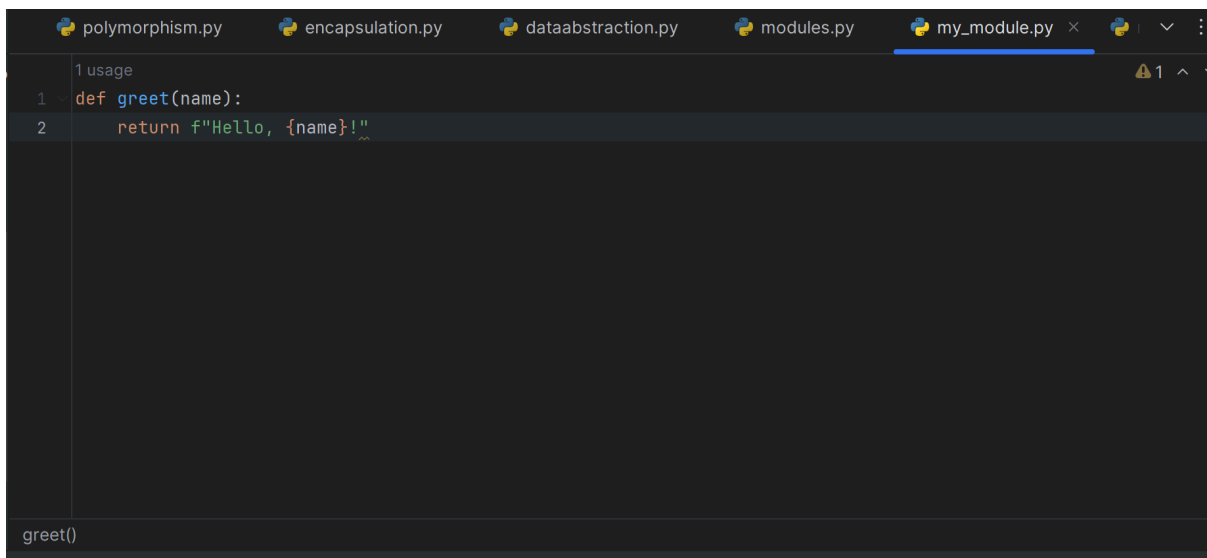


NAME: AKULA SHARATH CHANDRA
BATCH: DATA ENGINEERING
TOPICS:Python Modules,File handling

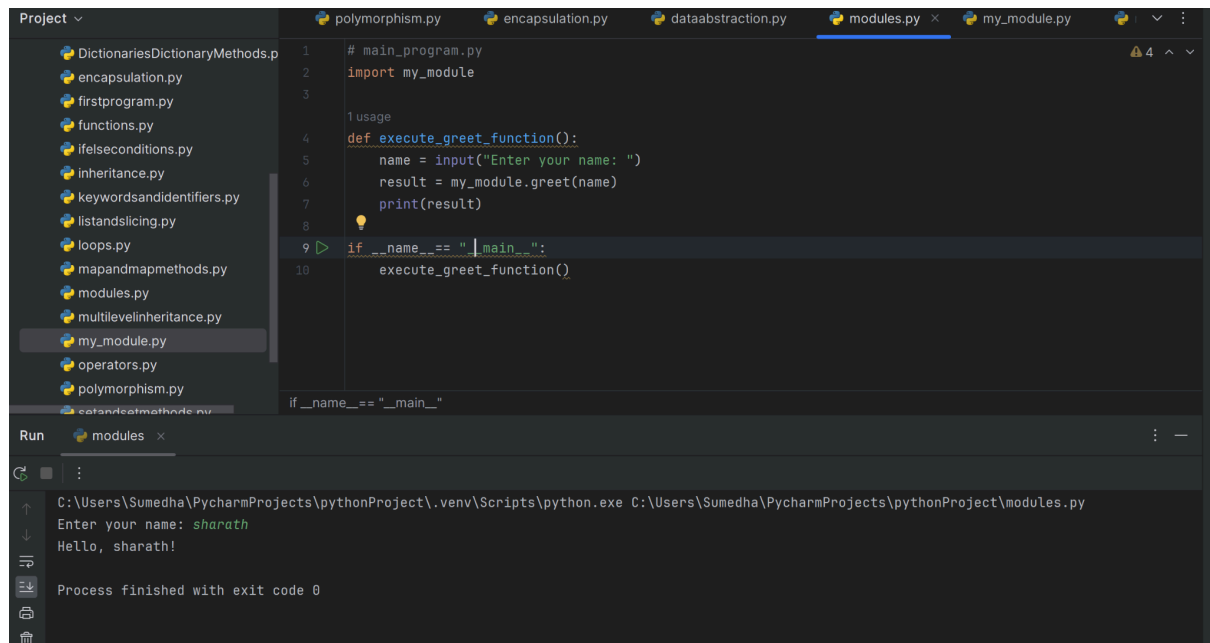
i)Python Modules:a module is a file containing Python definitions and statements. It serves as a way to organise and structure Python code by grouping related functionalities together.



The image shows a screenshot of a Python IDE with several tabs open: polymorphism.py, encapsulation.py, dataabstraction.py, modules.py, and my_module.py. The my_module.py tab is active and displays the following code:

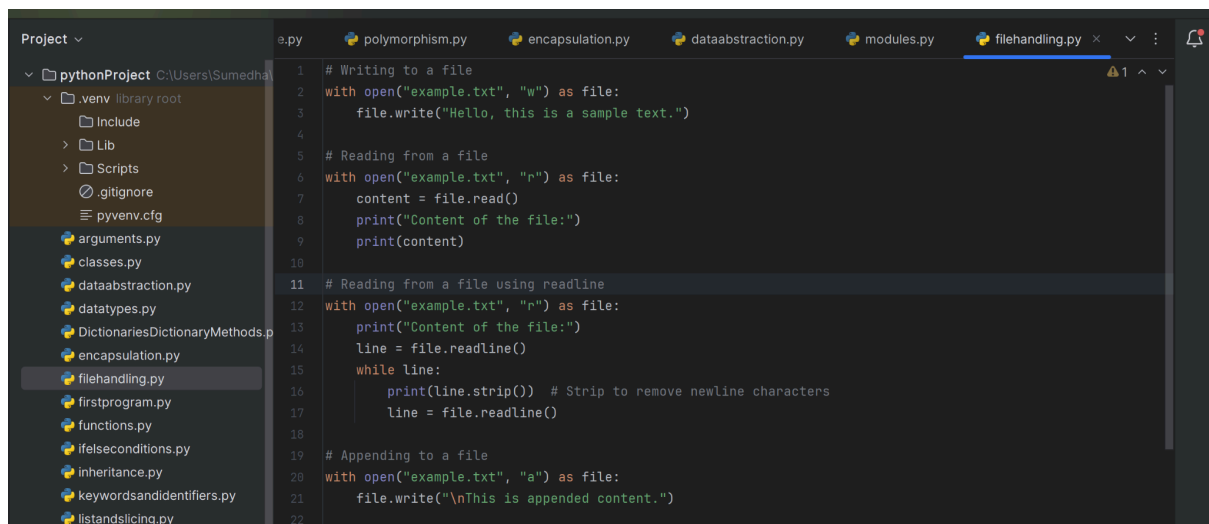
```
1 usage
2 def greet(name):
3     return f"Hello, {name}!"
```

The variable explorer on the left shows a variable named greet().

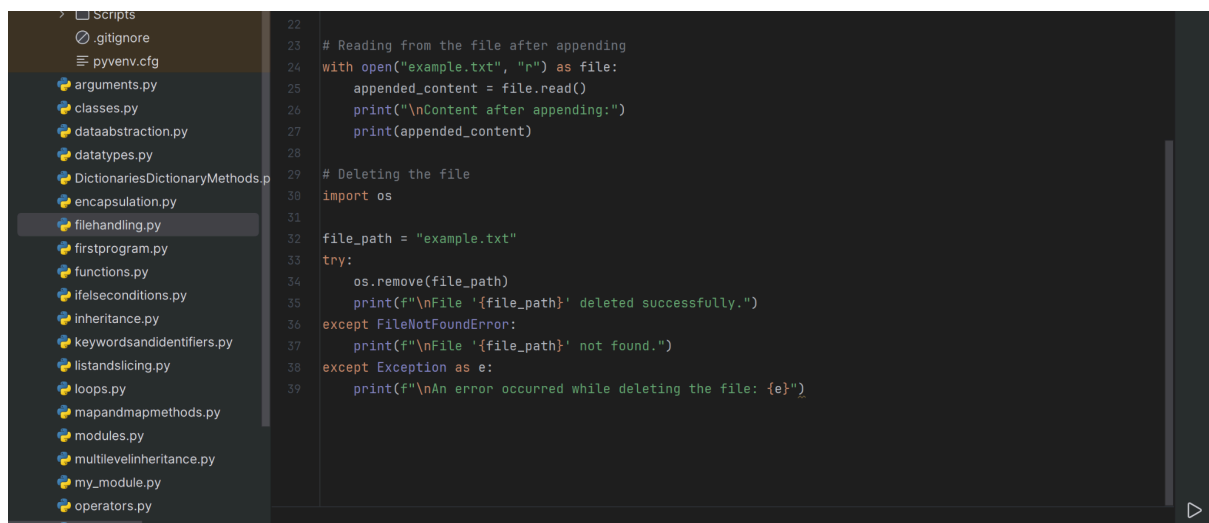


ii) File handling: File handling in Python allows you to read from and write to files. Python provides built-in functions and methods to perform various operations on files, such as opening, reading, writing, and closing.

—> i have done how to write into a file, how to read from a file , reading from a file using a read line, appending to a file, reading from the file after appending ,deleting the file .



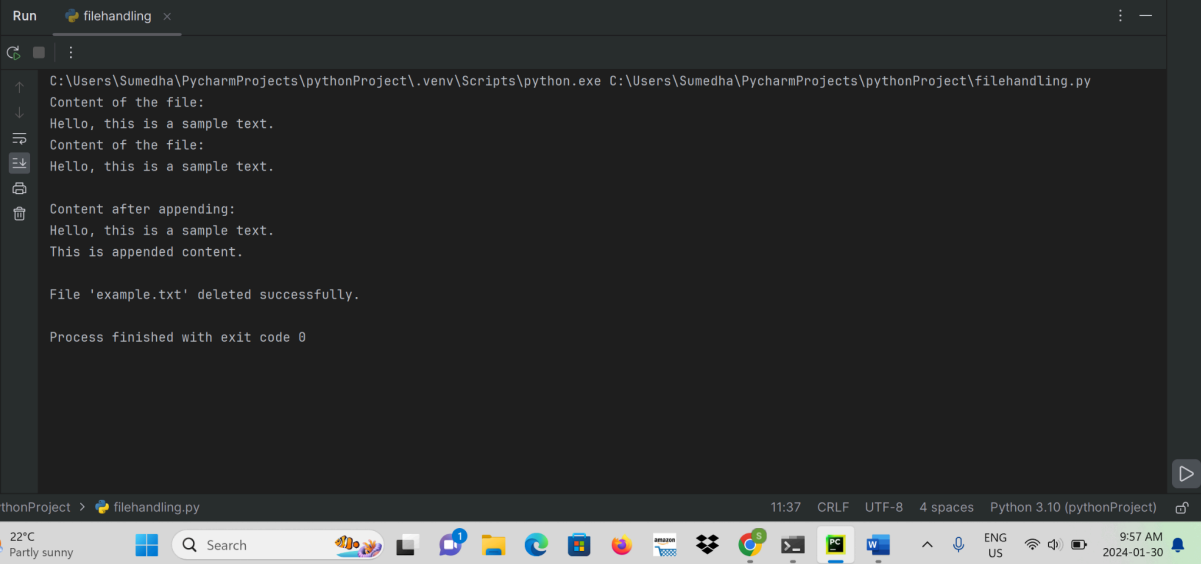
```
1 # Writing to a file
2 with open("example.txt", "w") as file:
3     file.write("Hello, this is a sample text.")
4
5 # Reading from a file
6 with open("example.txt", "r") as file:
7     content = file.read()
8     print("Content of the file:")
9     print(content)
10
11 # Reading from a file using readline
12 with open("example.txt", "r") as file:
13     print("Content of the file:")
14     line = file.readline()
15     while line:
16         print(line.strip()) # Strip to remove newline characters
17         line = file.readline()
18
19 # Appending to a file
20 with open("example.txt", "a") as file:
21     file.write("\nThis is appended content.")
22
```



```
22
23 # Reading from the file after appending
24 with open("example.txt", "r") as file:
25     appended_content = file.read()
26     print("\nContent after appending:")
27     print(appended_content)
28
29 # Deleting the file
30 import os
31
32 file_path = "example.txt"
33 try:
34     os.remove(file_path)
35     print(f"\nFile '{file_path}' deleted successfully.")
36 except FileNotFoundError:
37     print(f"\nFile '{file_path}' not found.")
38 except Exception as e:
39     print(f"\nAn error occurred while deleting the file: {e}")

```

Output:



```
Run filehandling x
C:\Users\Sumedha\PycharmProjects\pythonProject\.venv\Scripts\python.exe C:\Users\Sumedha\PycharmProjects\pythonProject\filehandling.py
Content of the file:
Hello, this is a sample text.
Content of the file:
Hello, this is a sample text.

Content after appending:
Hello, this is a sample text.
This is appended content.

File 'example.txt' deleted successfully.

Process finished with exit code 0
```

pythonProject > filehandling.py 11:37 CRLF UTF-8 4 spaces Python 3.10 (pythonProject)

22°C Partly sunny 9:57 AM 2024-01-30