Vidyavardhini’s College of Engineering & Technology

Department of Computer Engineering

Exploring Files and directories: Python program to append data to existing file and then display the entire file

Date of Performance:

Date of Submission:

Experiment No. 5

Title: Exploring Files and directories: Python program to append data to existing file and then display the entire file

Aim: To Exploring Files and directories: Python program to append data to existing file and then display the entire file

Objective: To Exploring Files and directories

Theory:

Directory also sometimes known as a folder are unit organizational structure in computer's file system for storing and locating files or more folders. Python now supports a number of APIs to list the directory contents. For instance, we can use the Path.iterdir, os.scandir, os.walk, Path.rglob, or os.listdir functions.

Python too supports file handling and allows users to handle files i.e., to read and write files, along with many other file handling options, to operate on files. The concept of file handling has stretched over various other languages, but the implementation is either complicated or lengthy, but alike other concepts of Python, this concept here is also easy and short. Python treats file differently as text or binary and this is important. Each line of code includes a sequence of characters and they form text file. Each line of a file is terminated with a special character, called the EOL or End of Line characters like comma {,} or newline character. It ends the current line and tells the interpreter a new one has begun. Let's start with Reading and Writing files.

Working of open() function

We use open () function in Python to open a file in read or write mode. As explained above, open ( ) will return a file object. To return a file object we use open() function along with two arguments, that accepts file name and the mode, whether to read or write. So, the syntax being: open(filename, mode). There are three kinds of mode, that Python provides and how files can be opened:

“ r “, for reading.

“ w “, for writing.

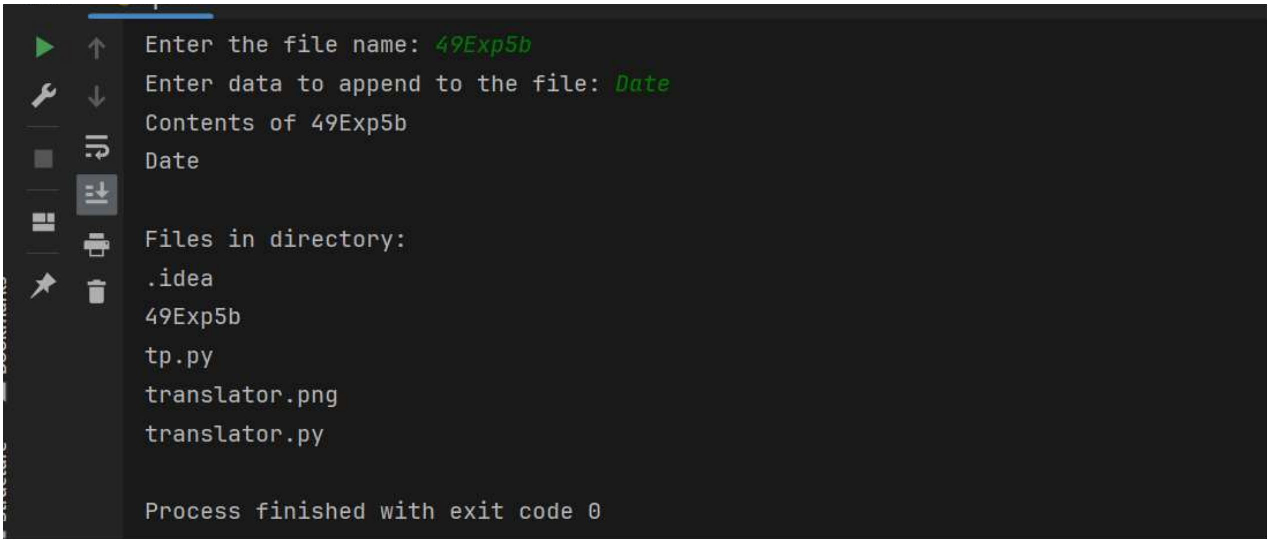
“ a “, for appending.

“ r+

Code:

```
import os
def append_to_file(file_name, data):
   try:
       with open(file_name, 'a') as file:
           file.write(data + '\n')
   except FileNotFoundError:
       print("File not found.")
def display_file(file_name):
   try:
       with open(file_name, 'r') as file:
           print("Contents of", file_name)
           print(file.read())
   except FileNotFoundError:
       print("File not found.")
def list_directory():
   directory = os.getcwd()
   print("Files in directory:")
   for file_name in os.listdir(directory):
       print(file_name)
def main():
   file_name = input("Enter the file name: ")
   data = input("Enter data to append to the file: ")
   append_to_file(file_name, data)
   display_file(file_name)
   list_directory()
if __name__ == "__main__":
   main()
```

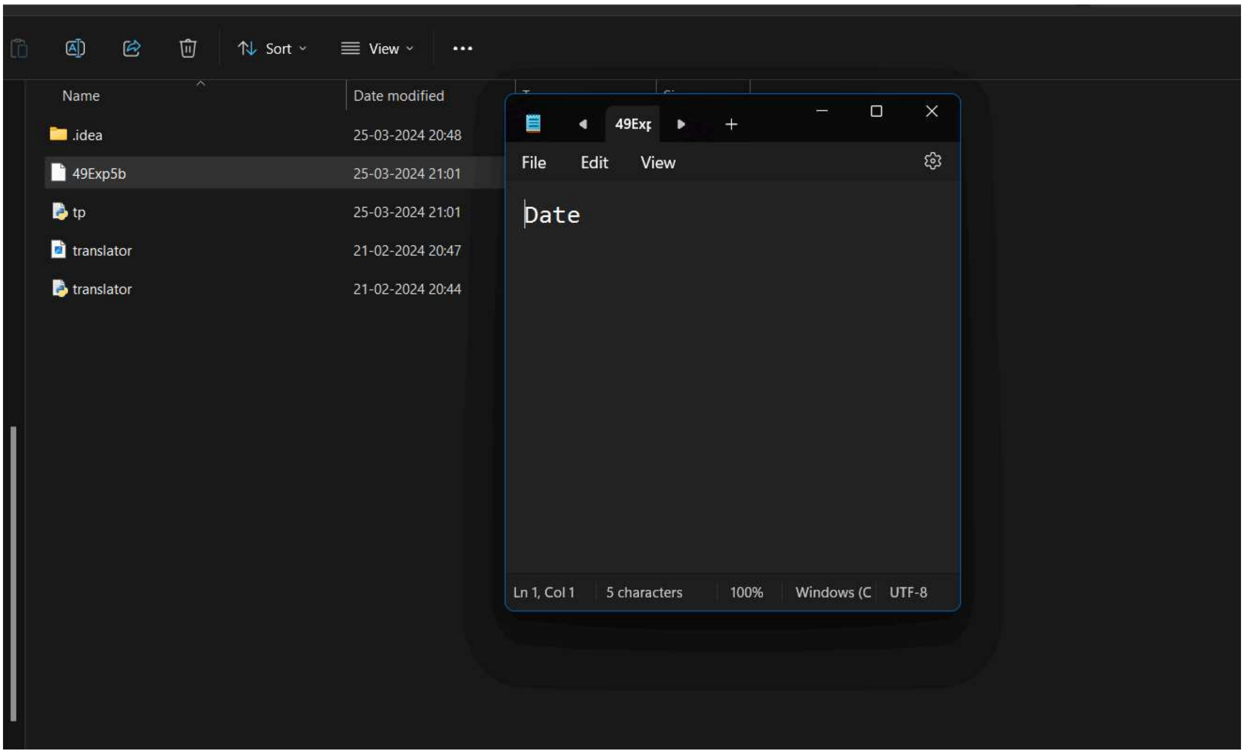
Output:



```
Enter the file name: 49Exp5b
Enter data to append to the file: Date
Contents of 49Exp5b
Date

Files in directory:
.idea
49Exp5b
tp.py
translator.png
translator.py

Process finished with exit code 0
```



Name	Date modified
.idea	25-03-2024 20:48
49Exp5b	25-03-2024 21:01
tp	25-03-2024 21:01
translator	21-02-2024 20:47
translator	21-02-2024 20:44

```
Date
```

Conclusion: Directories and files have been explored.