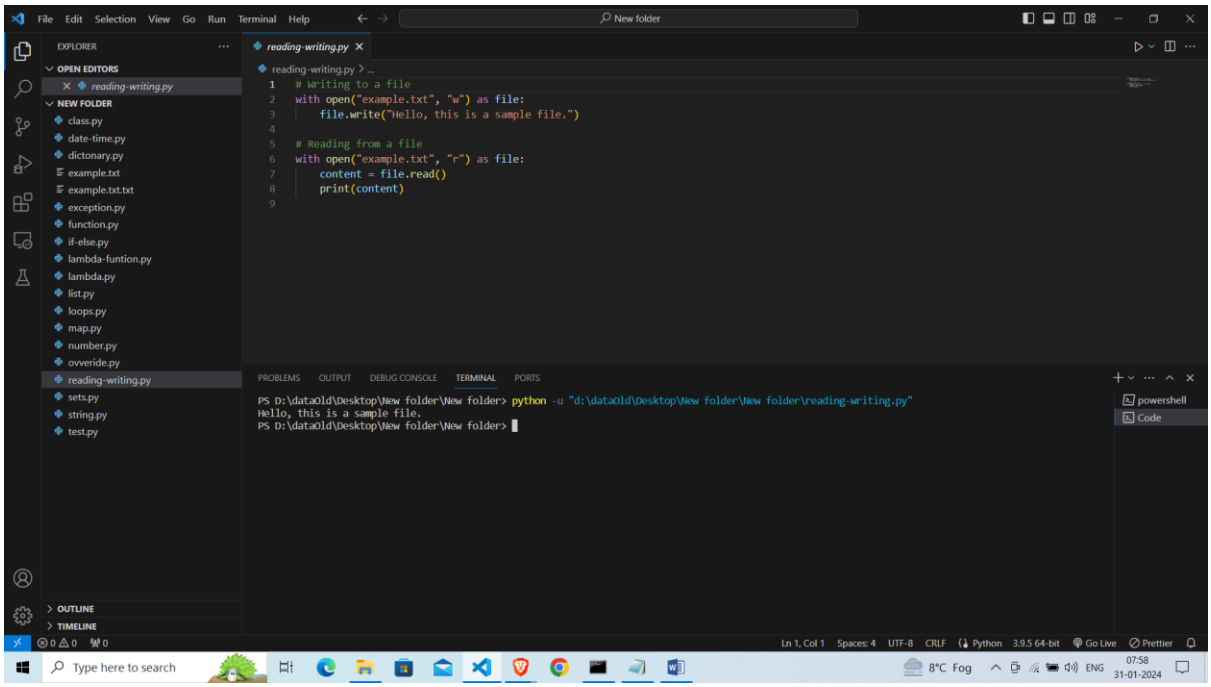
Assignment - 8

File I/O using Python:

File Input/Output (I/O) in Python allows you to interact with external files. The open() function is typically used to open a file. It takes two arguments - the name of the file and the mode in which the file is opened ('r' for reading, 'w' for writing, 'a' for appending, etc.). Once you're done with the file, it's good practice to close it using the close() method.

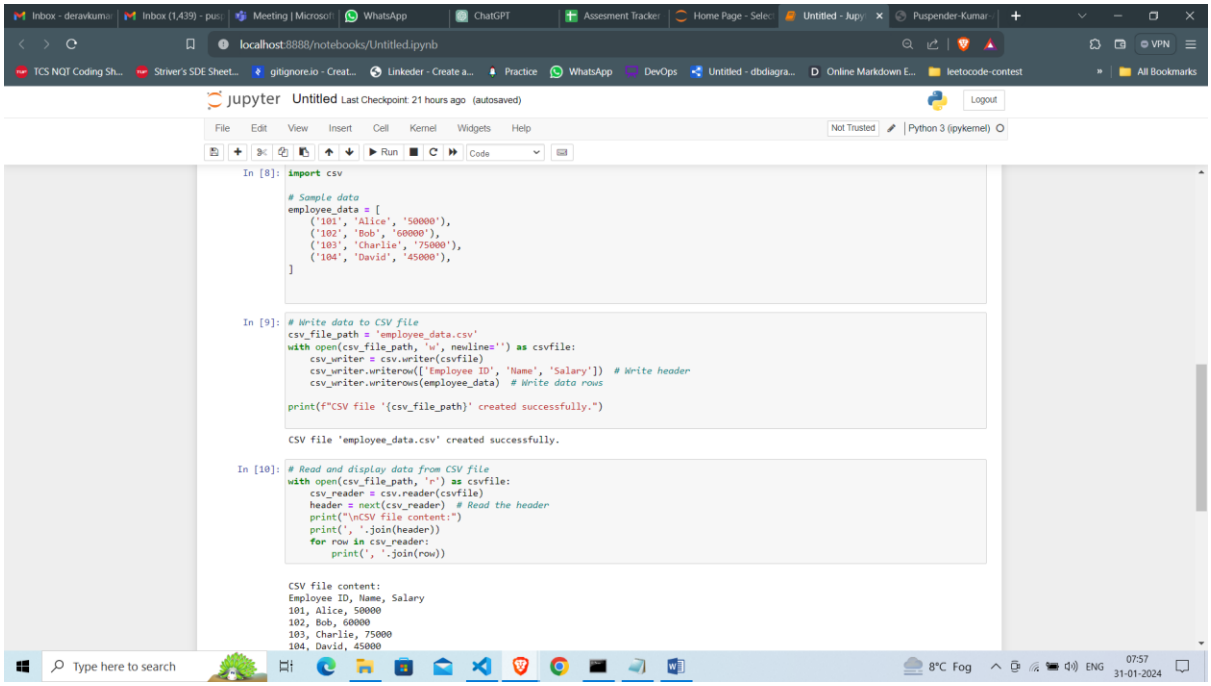
Example of reading and writing a file:



Read Data from CSV File into Python List:

CSV (Comma-Separated Values) files are a common format for storing tabular data. Python provides the csv module to handle CSV files.

Example of reading data from a CSV file into a list:



Processing Python Lists:

In Python, a list is a versatile and commonly used data structure that allows you to store and manipulate a collection of items. Lists in Python are ordered and mutable, meaning you can change the contents of a list after it is created. You can perform various operations on lists, such as iterating through them, adding or removing elements, and applying functions to each element.

Lambda Functions in Python:

A lambda function in Python is a small, anonymous function defined using the lambda keyword. Lambda functions are useful for creating short, throwaway functions without the need to formally define them using the def keyword. Lambda functions can take any number of arguments, but they can only have one expression. The syntax for a lambda function is: lambda arguments: expression.

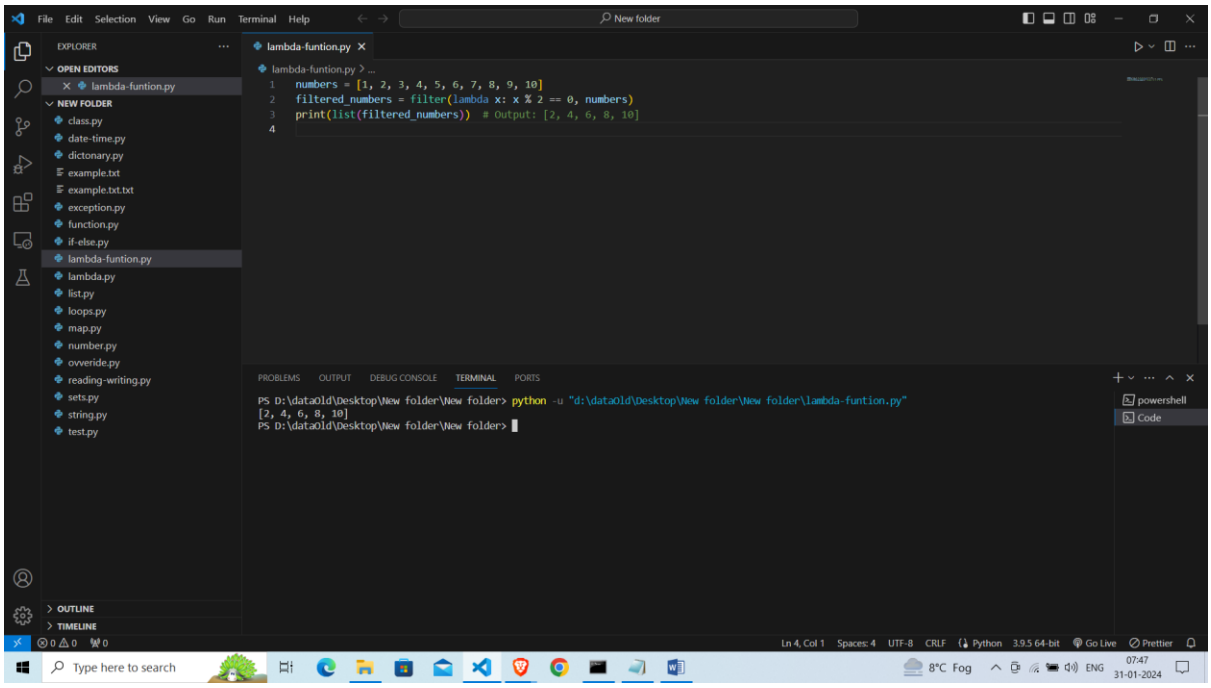
Usage of Lambda Functions:

Lambda functions are often used in situations where a small function is required for a short period and defining a full function using def would be unnecessary. They are commonly used in functional programming constructs like map(), filter(), and reduce().

Filter Data in Python Lists using filter and lambda:

The filter() function in Python is used to filter elements of an iterable based on a specified function. When combined with a lambda function, it becomes a concise way to filter elements from a list. The filter() function takes a function and an iterable, applying the function to each element in the iterable and returning only those for which the function evaluates to True.

Example:

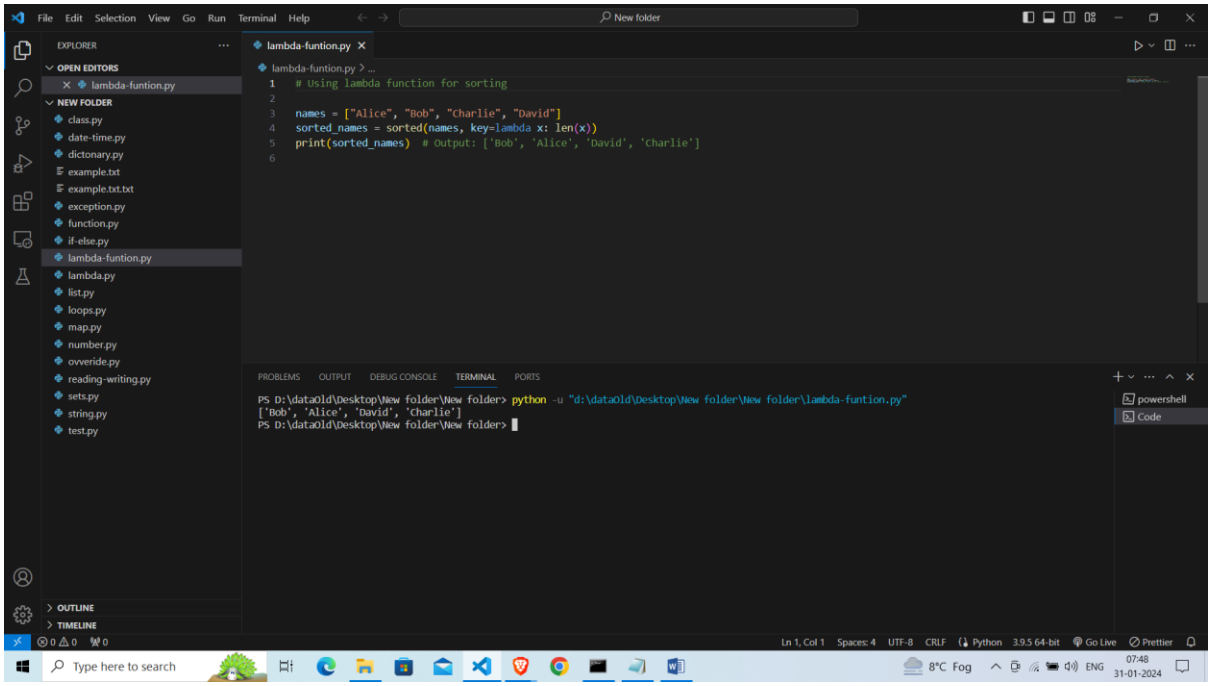


Use of Lambda Function in Python:

Lambda functions find application in situations where a small, anonymous function is needed for a short duration, such as in functional programming, sorting, filtering, and mapping.

Practical Uses of Python lambda function:

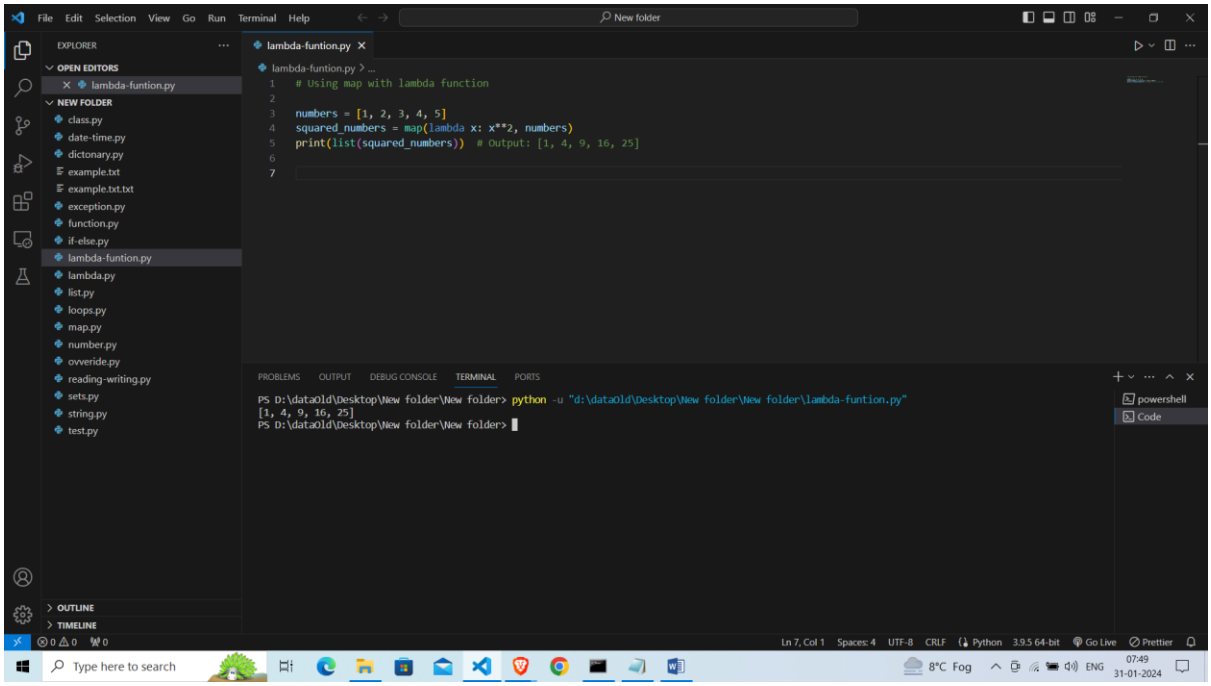
- Sorting: Lambda functions are commonly used as the key function in sorting operations.



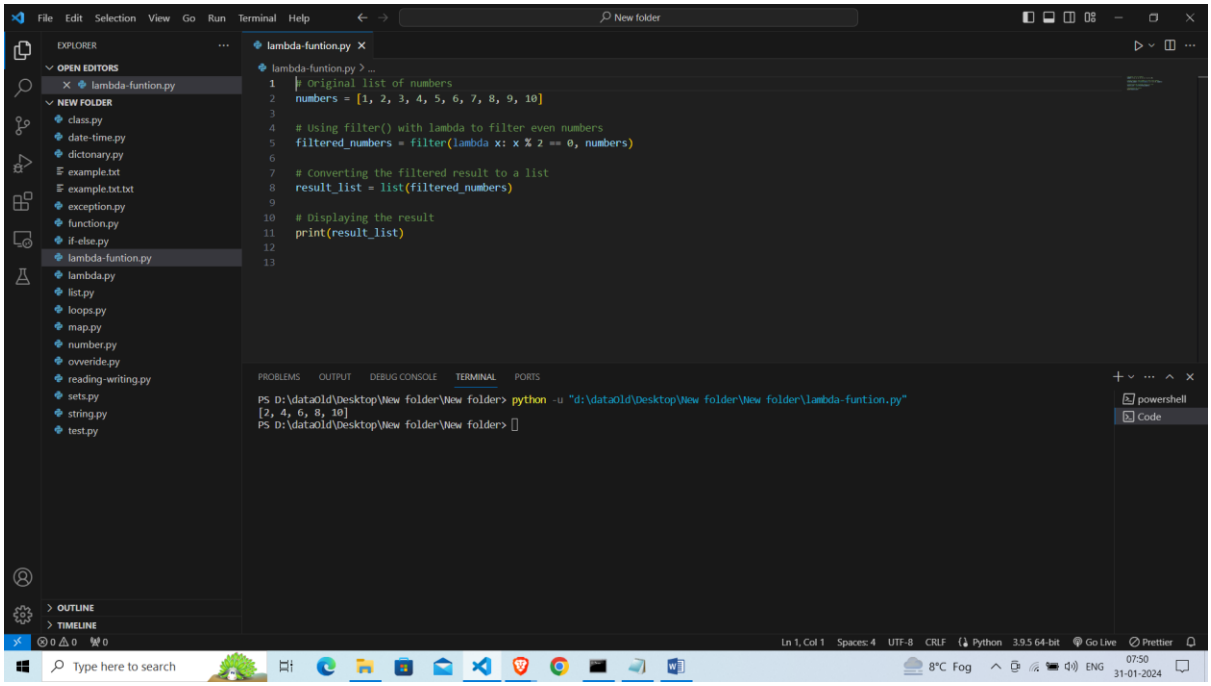
- GUI Applications: Lambda functions are sometimes used in GUI applications as quick event handlers.

Using lambda() Function with map(), filter(), reduce():

- **map():** The map() function applies a given function to all the items in an iterable and returns an iterator that produces the results.



- **filter():** As discussed earlier, filter() can be used in conjunction with lambda functions to filter elements based on a condition.



- **reduce():** The reduce() function is part of the functools module and is used to successively apply a binary function to the items of an iterable, reducing them to a single accumulated result.