AI ASSISTED CODING

**LAB-5.1**

NAME:A.DINESH

ENROLL.NO:2403A52090 BATCH-04

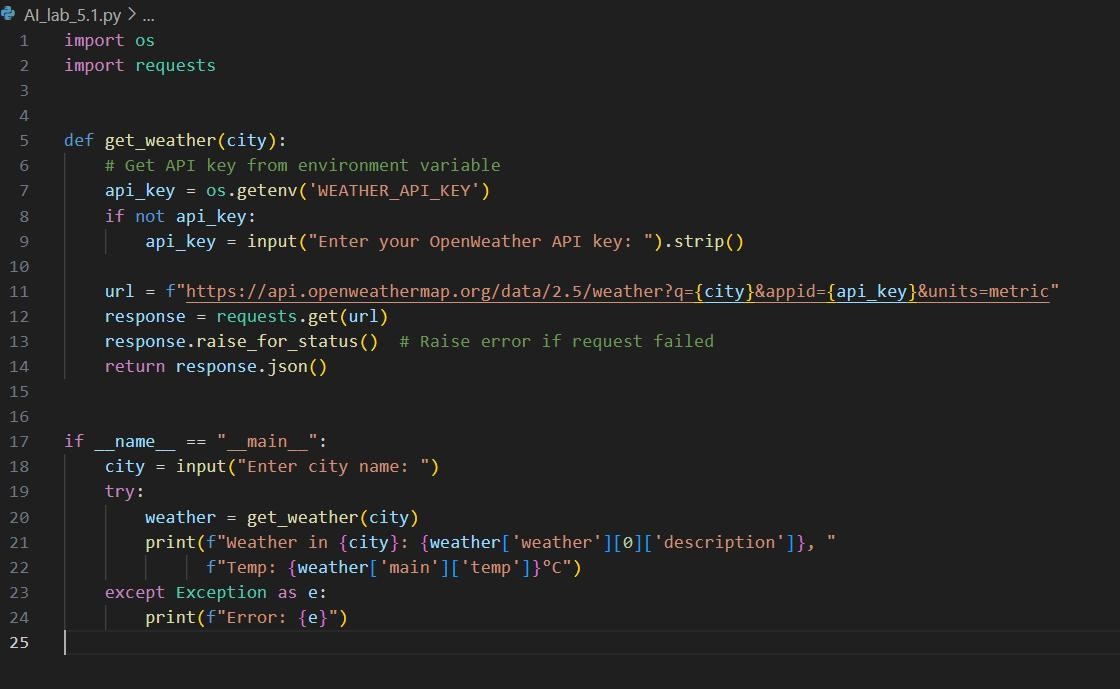
TASK-01:

Use an AI tool to generate a Python program that connects to a weather API.

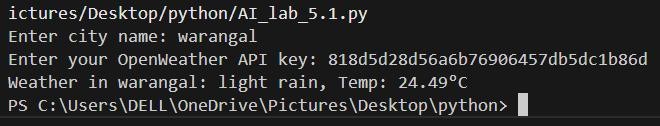
PROMPT:

Generate a python program that connects to a weather API and displays the climate of the particular city.

Code:



OUTPUT:



Observation:

The code generated by the github copilot is to check the weather of a particular city or village by entering its name and it asks for the API key which is generated in Open WeatherApp.

TASK-02:

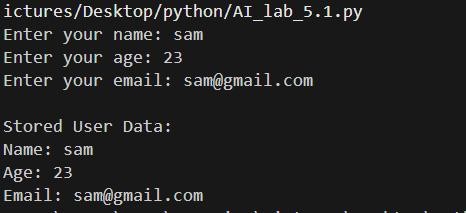
Use an AI tool to generate a Python script that stores user data (name, email, password) in a file.

PROMPT:

Write a python program to store the users data in a file. CODE:



OUTPUT:



OBSERVATION:

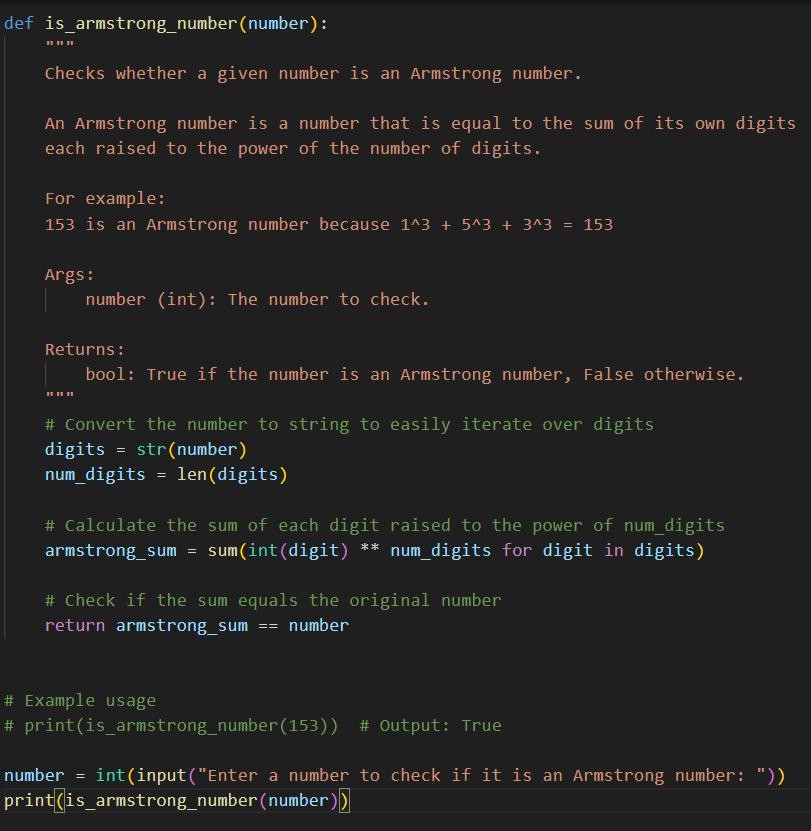
The code takes the input as name , age, email and stores them in a file which is created by it by asking several permissions.

TASK-03: Use AI to generate an Armstrong number checking function with comments and explanations

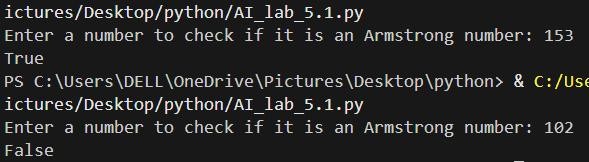
PROMPT:

Write a python function to check whether the given number is Armstrong or not also provide clear explanation using comments.

CODE:



OUTPUT:



OBSERVATIONS:

The code is about the python function which accepts a number as an input and checks whether the given number is Armstrong or not. The copilot gave a clear explanation using comments.

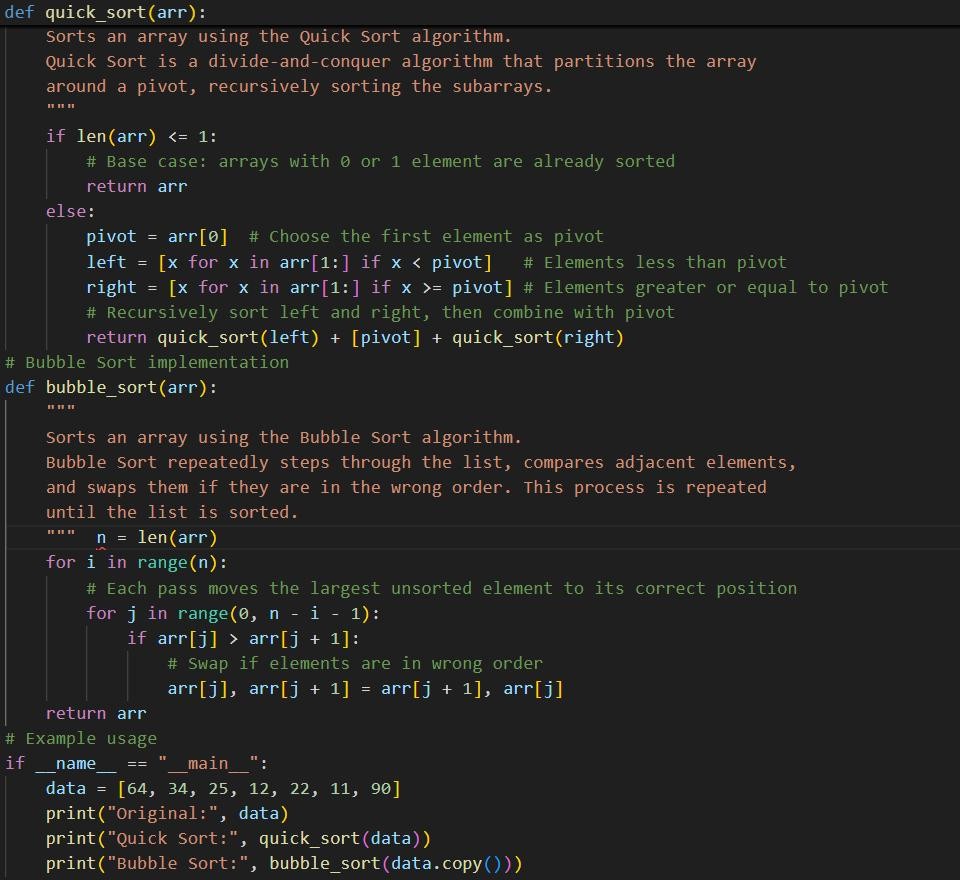
TASK-04:

Use AI to implement two sorting algorithms (e.g., QuickSort and BubbleSort).

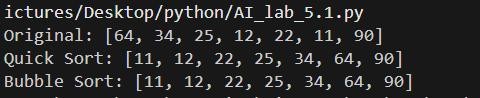
PROMPT:

Generate Python code for QuickSort and BubbleSort, and include comments explaining step-by-step how each works and where they differ.

CODE:



OUTPUT:



OBSERVATION:

The code generated by the copilot is the technique quick sort and bubble sort of the array and the code is explained step-by-step using comments

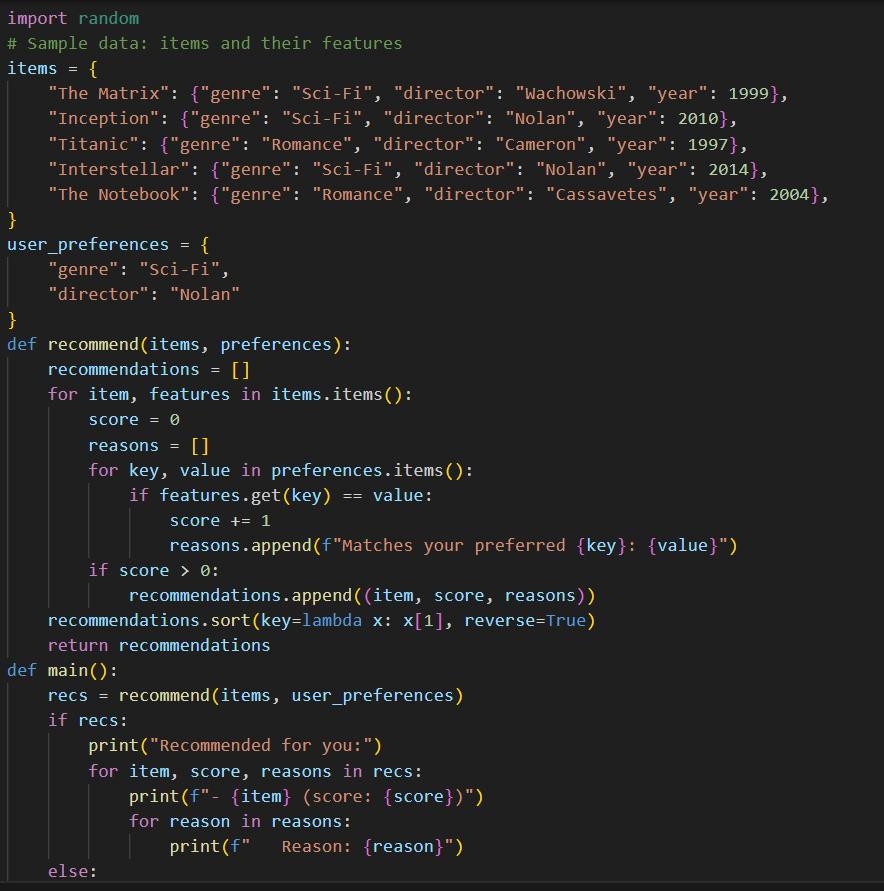
.TASK-05:

Use AI to create a product recommendation system

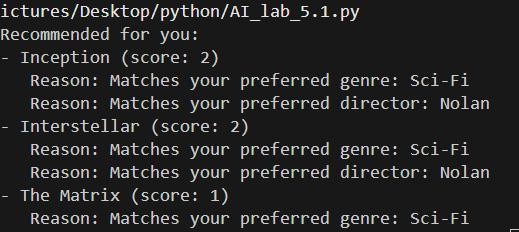
PROMPT:

Generate a recommendation system that also provides reasons for each suggestion.

CODE:



OUTPUT:



OBSERVATION:

The code is about the product recommendation system and it also explains the reason why the product is recommended for you.

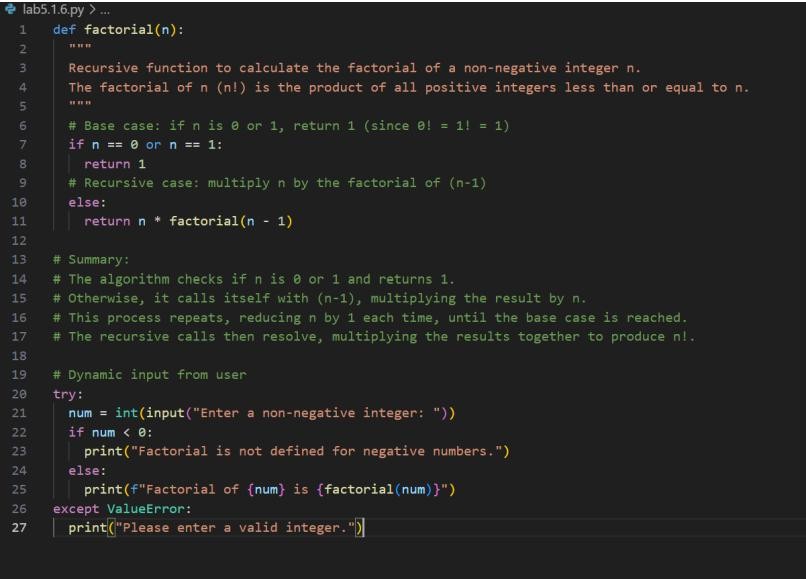
TASK-06:

Ask AI to generate a Python function for calculating factorial using recursion

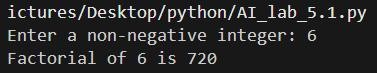
PROMPT:

Generate a recursive factorial function with comments that explain each line and a final summary of the algorithm’s flow.

CODE:



OUTPUT:



OBSERVATION:

The code is to find the factorial of the given number by using recursive function. It also explains how the algorithm works in comments.

TASK-07:

def support\_reply(name,gender): if gender.lower() == “male”:

prefix = “Mr.”

else:

prefix = “Mrs.”

return f”Dear{prefix} {name},we have resolved the problem.”

Regenerate the code so that support messages use neutral language (e.g., “Dear {name}”) and optionally accept preferred titles

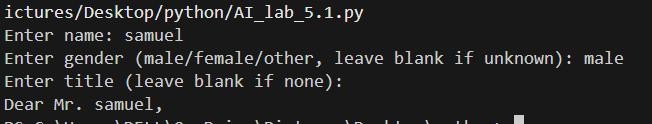
PROMPT:

Regenerate the code to accept the neutral comments and give the appropriate output.

CODE:



OUTPUT:



OBSERVATION:

The code is about generating message based on their gender and greet the person in an appropriate way. The copilot generated very well without creating any nuisense