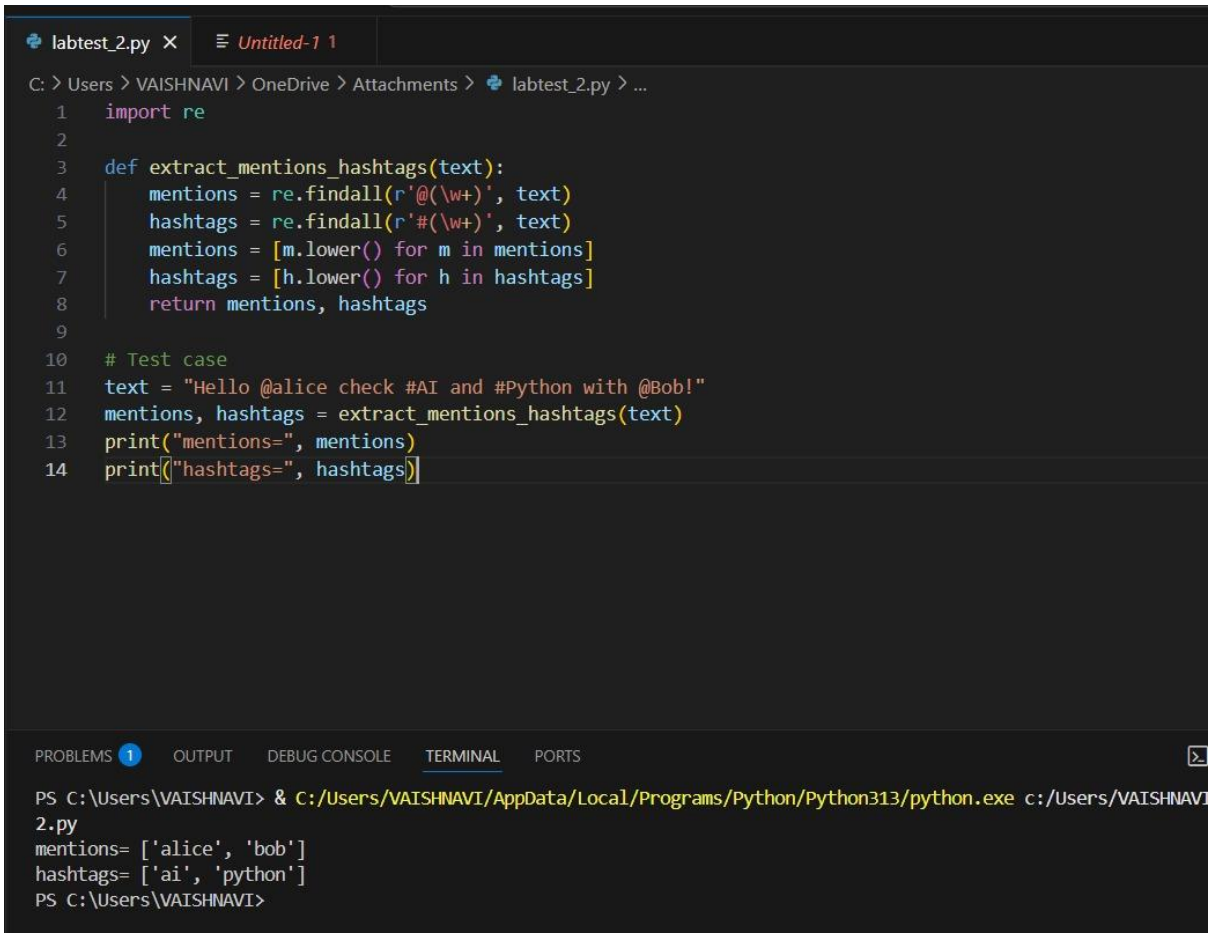
AI ASSITED CODING LAB TEST – 2

(SET - H)

1.TASK : Regex extract mentions/hashtags, lowercase lists.

PROMPT : Write a Python function to extract all @mentions and #hashtags from a string, ignoring punctuation, and return two lowercase lists

CODE:



OBSERVATION :

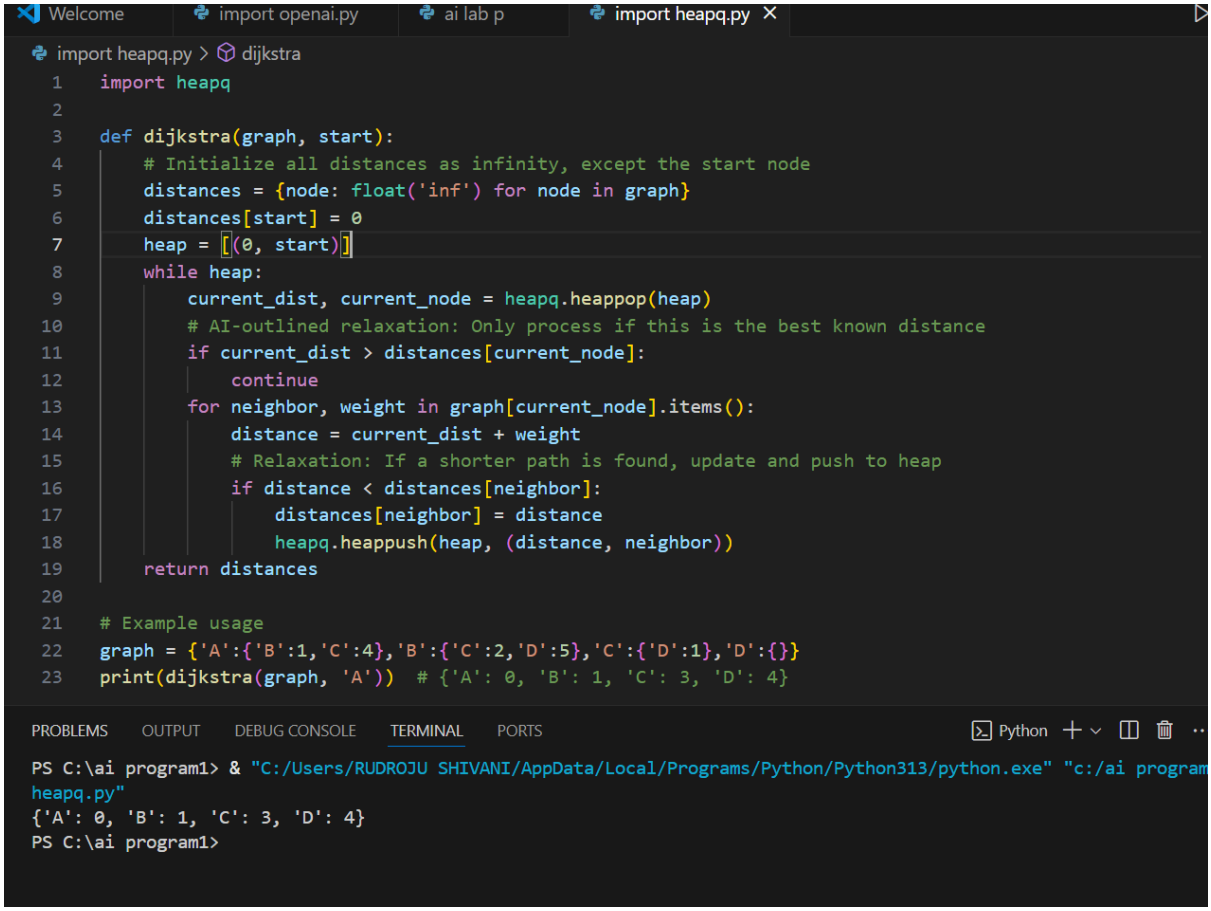
- The function extract_mentions_hashtags(text) uses regular expressions to find all words that follow @ (mentions) and # (hashtags) in the input string.
- re.findall(r'@(\w+)', text) finds all mentions, and re.findall(r'#(\w+)', text) finds all hashtags.
- Both lists are converted to lowercase for normalization.
- The function returns two lists: one for mentions and one for hashtags.
- The test cases check that the function works with typical input, punctuation, and edge cases (like no tags).

2 .TASK : Dijkstra from 'A' using heapq.

PROMPT :

Given a graph as a positive-weight adjacency dictionary, implement Dijkstra’s algorithm using heapq to find the shortest distances from node 'A'. Return a dictionary of shortest distances. AI should outline the relaxation step.

CODE :



OBSERVATION :

All nodes start with infinite distance except 'A' (0).

The heap always pops the node with the smallest known distance.

For each neighbor, the algorithm checks if the path through the current node is shorter (relaxation).

If so, it updates the distance and adds the neighbor to the heap.

Continues until all nodes are processed.

The output matches the sample.