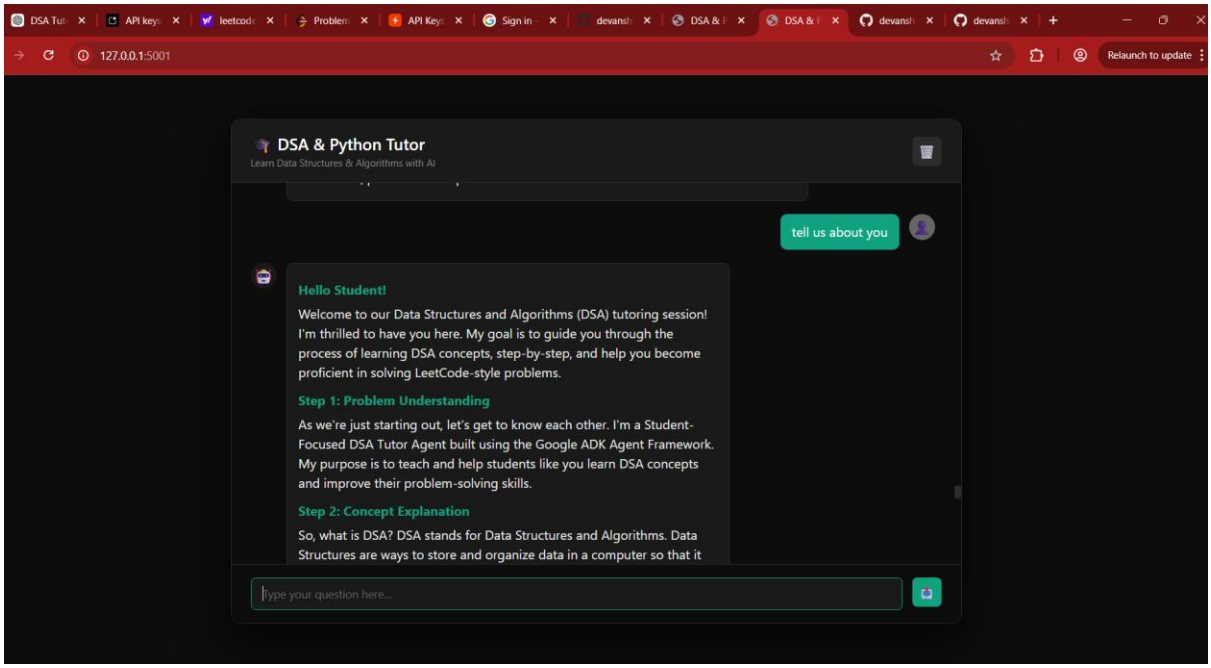
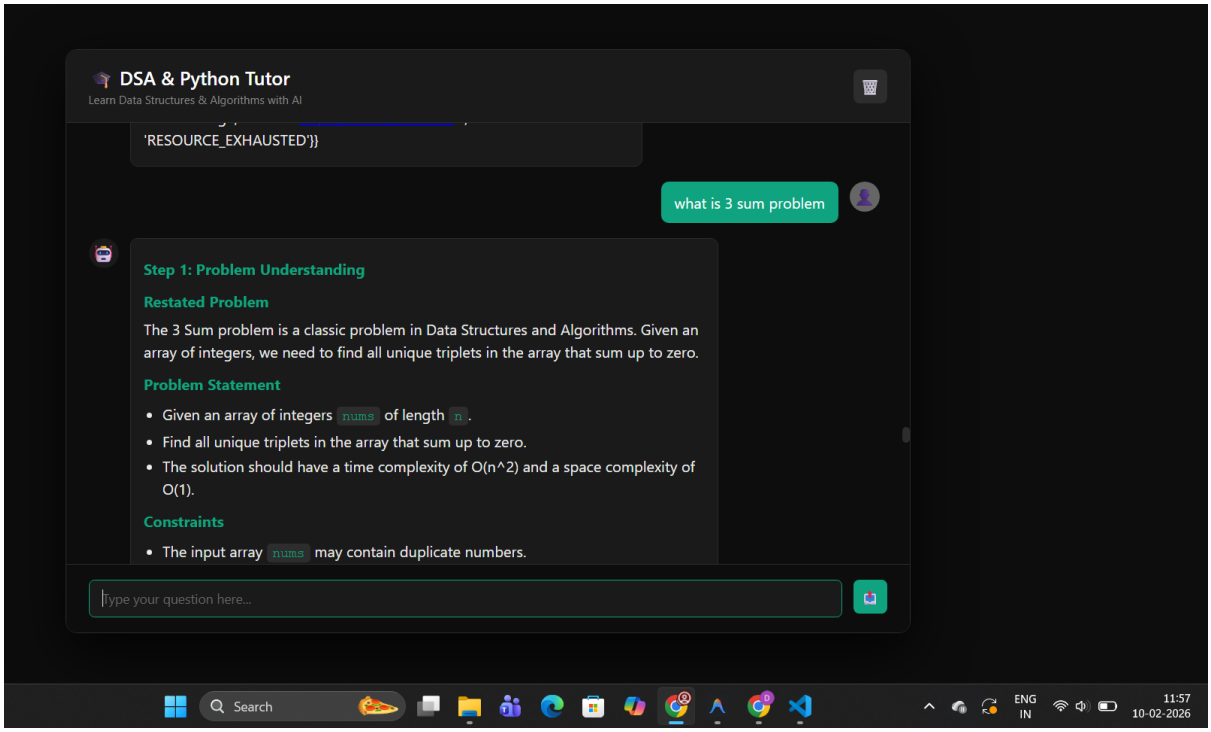
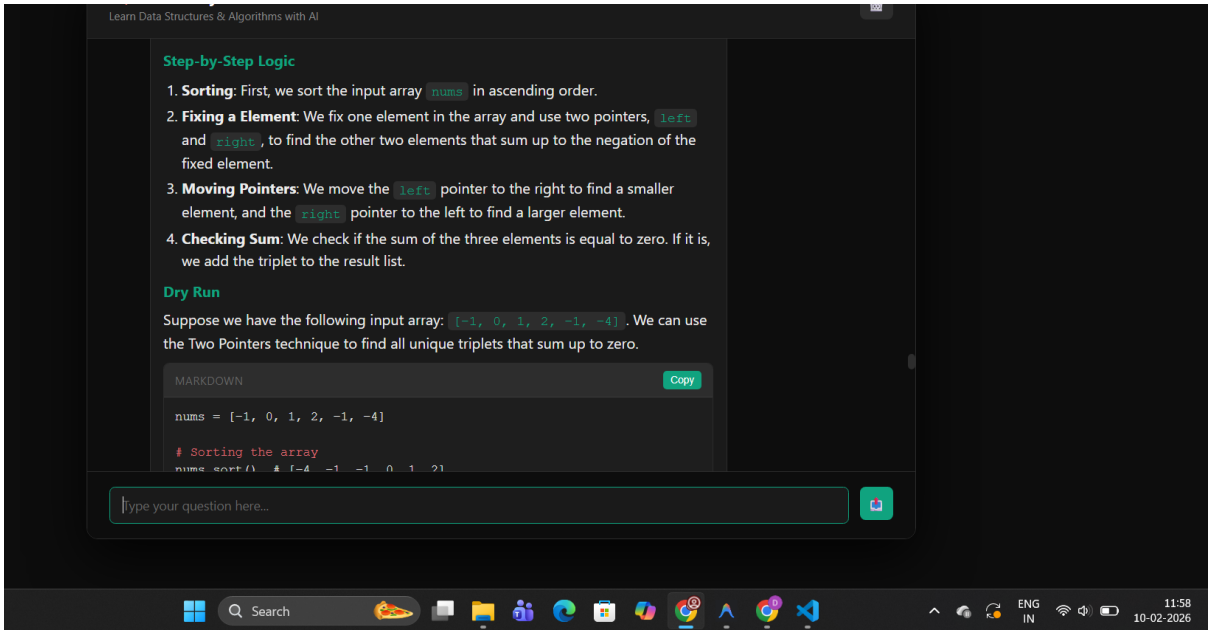
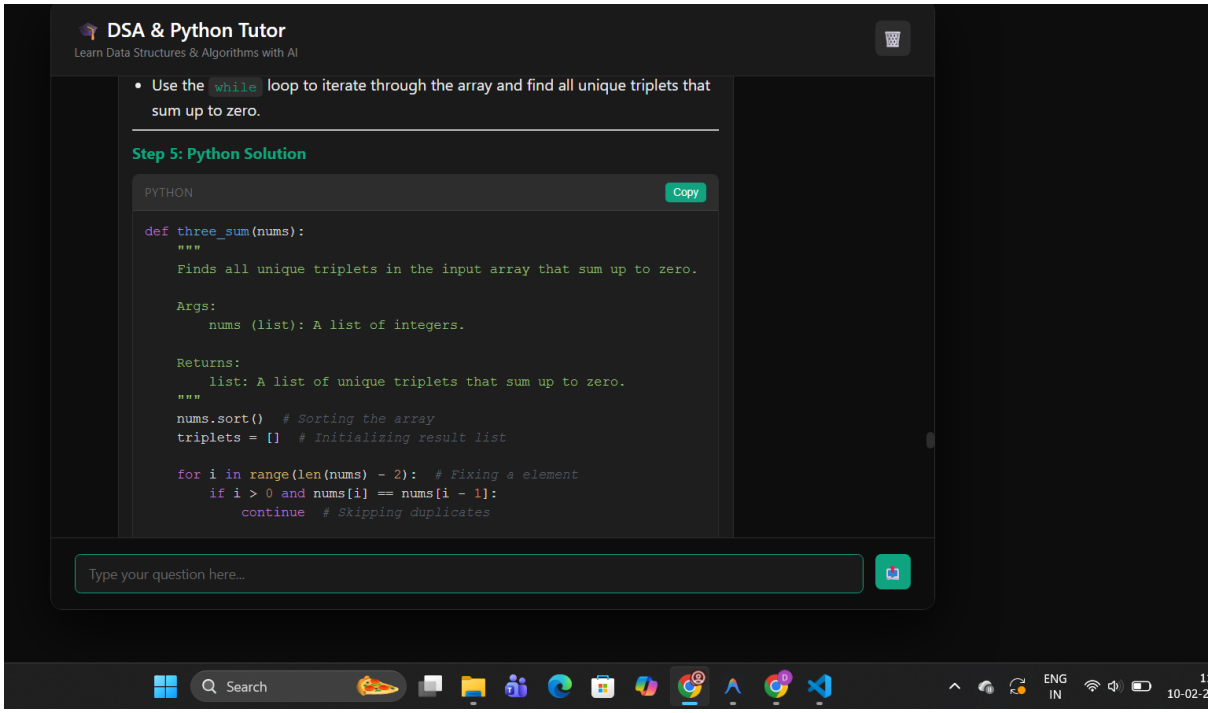
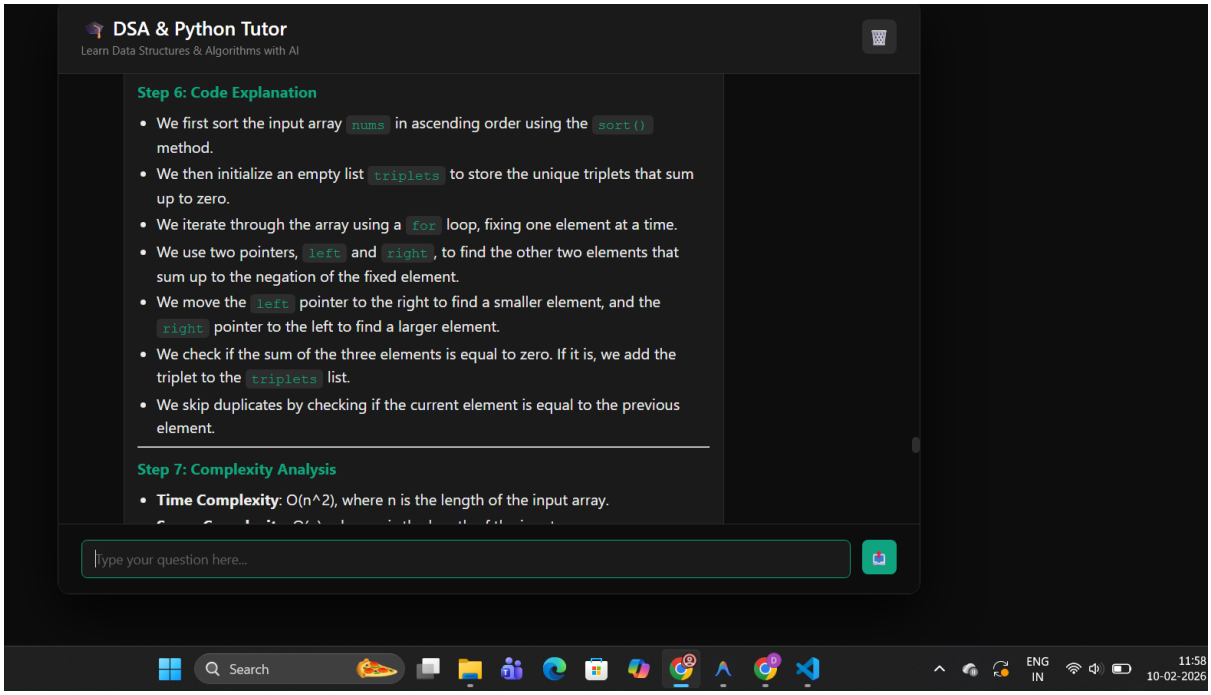

127.0.0.1:5001
Relaunch to update
DSA & Python Tutor
Learn Data Structures & Algorithms with AI
tell us about you
Hello Student!
Welcome to our Data Structures and Algorithms (DSA) tutoring session! I'm thrilled to have you here. My goal is to guide you through the process of learning DSA concepts, step-by-step, and help you become proficient in solving LeetCode-style problems.
Step 1: Problem Understanding
As we're just starting out, let's get to know each other. I'm a Student-Focused DSA Tutor Agent built using the Google ADK Agent Framework. My purpose is to teach and help students like you learn DSA concepts and improve their problem-solving skills.
Step 2: Concept Explanation
So, what is DSA? DSA stands for Data Structures and Algorithms. Data Structures are ways to store and organize data in a computer so that it
Type your question here...


DSA & Python Tutor
Learn Data Structures & Algorithms with AI
'RESOURCE_EXHAUSTED'}}
what is 3 sum problem
Step 1: Problem Understanding
Restated Problem
The 3 Sum problem is a classic problem in Data Structures and Algorithms. Given an array of integers, we need to find all unique triplets in the array that sum up to zero.
Problem Statement
Given an array of integers nums of length n .
Find all unique triplets in the array that sum up to zero.
The solution should have a time complexity of O(n^2) and a space complexity of O(1).
Constraints
The input array nums may contain duplicate numbers.
Type your question here...
Search
ENG
IN
11:57
10-02-2026


Learn Data Structures & Algorithms with AI
Step-by-Step Logic
1. Sorting: First, we sort the input array nums in ascending order.
2. Fixing a Element: We fix one element in the array and use two pointers, left and right, to find the other two elements that sum up to the negation of the fixed element.
3. Moving Pointers: We move the left pointer to the right to find a smaller element, and the right pointer to the left to find a larger element.
4. Checking Sum: We check if the sum of the three elements is equal to zero. If it is, we add the triplet to the result list.
Dry Run
Suppose we have the following input array: [-1, 0, 1, 2, -1, -4]. We can use the Two Pointers technique to find all unique triplets that sum up to zero.
MARKDOWN
Copy
nums = [-1, 0, 1, 2, -1, -4]
# Sorting the array
Type your question here...
Search
ENG
IN
11:58
10-02-2026


DSA & Python Tutor
Learn Data Structures & Algorithms with AI
• Use the while loop to iterate through the array and find all unique triplets that sum up to zero.
Step 5: Python Solution
PYTHON
Copy
def three_sum(nums):
"""
Finds all unique triplets in the input array that sum up to zero.
Args:
nums (list): A list of integers.
Returns:
list: A list of unique triplets that sum up to zero.
"""
nums.sort() # Sorting the array
triplets = [] # Initializing result list
for i in range(len(nums) - 2): # Fixing a element
if i > 0 and nums[i] == nums[i - 1]:
continue # Skipping duplicates
Type your question here...
Search
ENG
IN


DSA & Python Tutor
Learn Data Structures & Algorithms with AI
Step 6: Code Explanation
We first sort the input array nums in ascending order using the sort() method.
We then initialize an empty list triplets to store the unique triplets that sum up to zero.
We iterate through the array using a for loop, fixing one element at a time.
We use two pointers, left and right, to find the other two elements that sum up to the negation of the fixed element.
We move the left pointer to the right to find a smaller element, and the right pointer to the left to find a larger element.
We check if the sum of the three elements is equal to zero. If it is, we add the triplet to the triplets list.
We skip duplicates by checking if the current element is equal to the previous element.
Step 7: Complexity Analysis
Time Complexity: O(n^2), where n is the length of the input array.
Type your question here...
Search
ENG
IN
11:58
10-02-2026