class Solution:

    def twoSum(self, nums: List[int], target: int) -> List[int]:

        # Create a dictionary to store numbers and their indices

        num\_map = {}

        # Iterate through the list with both index and value

        for i, num in enumerate(nums):

            # Calculate the complement needed to reach the target

            complement = target - num

            # Check if the complement already exists in the dictionary

            if complement in num\_map:

                # If found, return the index of the complement and the current index

                return [num\_map[complement], i]

            # If not found, add the current number and its index to the dictionary

            num\_map[num] = i

        # If no such pair is found (though the problem usually guarantees one)

        return []