Day-4

Python DSA

https://github.com/im-amit-kumar/Python-DSA/tree/main/Day-4

1. Leetcode 1. Two Sum

https://leetcode.com/problems/two-sum/description/

Optimal Solution

```python
class Solution:
    def twoSum(self, nums: List[int], target: int) -> List[int]:
        n = len(nums)
        hash_map = dict()
        for i in range(n):
            remaining = target - nums[i]
            if remaining in hash_map:
                return [hash_map[remaining],i]
            hash_map[nums[i]]=i
```

Time Complexity – O(N)

Space Complexity – O(N)

2. Leetcode 75 Sort Colors

https://leetcode.com/problems/sort-colors/description/

Optimal Solution

```python
class Solution:
    def sortColors(self, nums: List[int]) -> None:
        """
        Do not return anything, modify nums in-place instead.
        """
        low = 0
        mid = 0
        high = len(nums) - 1

        while mid <= high:
            if nums[mid] == 0:
                nums[low], nums[mid] = nums[mid], nums[low]
                low += 1
                mid += 1
            elif nums[mid] == 1:
                mid += 1
            else:
                nums[mid], nums[high] = nums[high], nums[mid]
                high -= 1
```

Time Complexity – O(N)

Space Complexity – O(1)

3. Leetcode 169 Majority Element

https://leetcode.com/problems/majority-element/

Optimal Solution

```
class Solution:
    def majorityElement(self, nums: List[int]) -> int:
        candidate= nums[0]
        count=0
        for i in range(0,len(nums)):
            if count ==0:
                candidate = nums[i]
                count=1
```

TC – O(N)

SC- O(1)

4. Leetcode 53 Maximum Subarray

https://leetcode.com/problems/maximum-subarray/

Optimal Solution

```
class Solution:
    def maxSubArray(self, nums: List[int]) -> int:
        maxi= float("-inf")
        Sum=0
        n= len(nums)
        for i in range(n):
            Sum+=nums[i]
            if Sum>maxi:
                maxi=Sum
            if Sum<0:
                Sum=0
        return maxi
```

Time Complexity - O(N)

Space Complexity - O(1)