Two Sum — hash map (two pass)

```
class Solution:
    def two_sum(self, nums: List[int], target: int) -> List[int]:
        indices = {}

        for i, num in enumerate(nums):
            indices[num] = i

        for i, num in enumerate(nums):
            diff = target - num
            if diff in indices and indices[diff] != i:
                return [i, indices[diff]]
print(Solution().two_sum(nums = [3,4,5,6], target = 7)
```

Step 1:

self -> solution instance

nums -> [3,4,5,6]

target -> 7

indices → {} (empty)

Step 2: (for i, num in enumerate(nums):)

i → 0

num → 3

Step 3: indices[num] = i

indices[3] = 0 => indices = {3:0}

Step 4: (for i, num in enumerate(nums))

i → 1

num → 4

Step 5: (indices[num] = i)

indices[4] = 1 => indices → {3:0; 4:1}

Step 6: we iterate another 2 times

i → 2, and then 3

num → 5 and then 6

indices → {3:0; 4:1; 5:2; 6:3}

second iteration

Step 7: (for i, num in enumerate(nums))

i -> 0

num -> 3

Step 8: (diff = target - num)

diff = 7 - 3 => 4

Step 9: (if diff in indices and indices[diff] != i)

4 is in indices

indices[4] = 1 => indices[4] != 0

Step 10: return [i, indices[diff]]

return [0, 1] ✓