

# Contains duplicates

part 3 using hash set

```
class Solution:
```

```
def contains_duplicates(self, nums: List[int]) -> bool:
```

```
    seen = set()
```

```
    for num in nums:
```

```
        if num in seen:
```

```
            return True
```

```
        seen.add(num)
```

```
    return False
```

```
print(Solution().contains_duplicates(nums = [1, 2, 3, 4, 1]))
```

- Step 1 :

self	→	solution	instance
nums	→	0 1 2 3 4	index
		1 2 3 4 1	

- Step 2:

seen  $\rightarrow$  empty set ( )

!info: set items are unchangeable, and duplicates are not allowed.

- Step 3 (for num in nums:)

num  $\rightarrow$  1

- Step 4 (if num in seen:)  
False

- Step 5 (seen.add(num))

seen  $\rightarrow$  (1)

- Step 6 (for num in nums:)

num  $\rightarrow$  2

- Step 7 (if num in seen:)  
False

- Step 8 (seen.add(num))

seen  $\rightarrow$  (1, 2)

- Step 9 (for num in nums:)  
num  $\rightarrow$  3

- Step 10 (if num in seen:)  
False

- Step 11 (seen.add(num))  
seen  $\rightarrow$  (1, 2, 3)

- Step 12 (for num in nums:)  
num  $\rightarrow$  4

- Step 13 (if num in seen:)  
False

- Step 14 (seen.add(num))  
seen  $\rightarrow$  (1, 2, 3, 4)

- Step 15 (for num in nums:)  
num  $\rightarrow$  1

- Step 16 ( if num in seen)  
return True. ✓