

Arrays

February 3, 2024

```
[3]: # # 1. Given an array, check if it contains any duplicates or not.
# arr = [1, 2, 4, 2, 5, 9]
# Output = True

def duplicates(arr):
    unique=set()
    for num in arr:
        if num in unique:
            return True
        unique.add(num)
    return False

arr = [1, 2, 4, 2, 5, 9]
v=duplicates(arr)
print(v)
```

True

```
[8]: # 2. Given an array and an integer k, rotate the array to the right by k steps.
# arr = [1, 2, 3, 4, 5, 6, 7] k = 3
# Output = [5, 6, 7, 1, 2, 3, 4]

def rotateright(arr,k):
    n=len(arr)

    if k==0 or n==1:
        return
    k %= n
    arr[:]= arr[-k:]+arr[:-k]

arr = [1, 2, 3, 4, 5, 6, 7]

rotateright(arr,2)
print(arr)
```

[6, 7, 1, 2, 3, 4, 5]

[9]: # 3. Reverse the given array in-place, means without using any extra data
↳ structure.

```
# arr = [2, 4, 5, 7, 9, 12]
# Output = [12, 9, 7, 5, 4, 2]
```

```
arr=[2,4,5,7,9,12]
arr=arr[::-1]
print(arr)
```

[12, 9, 7, 5, 4, 2]

[12]: # 4. Given an array of integers, find the maximum element in an array

```
# arr = [10, 5, 20, 8, 15]
# Output = 20
```

```
arr = [10, 5, 20, 8, 15]
a = max(arr)
print(a)
```

20

[14]: # 5. Given a sorted array, remove the duplicate element without using any extra
↳ data structure.

```
# arr = [1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5]
# Output = [1, 2, 3, 4, 5]
```

```
arr = [1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5]
arr=set(arr)
```

```
arr=list(arr)
print(arr)
```

[1, 2, 3, 4, 5]

[]: