

Assignment(Image processing).ipynb

```
#Implementation of Quicksort
def partition(arr, low, high):
    i = (low-1)
    pivot = arr[high]

    for j in range(low, high):
        if arr[j] <= pivot:
            i = i+1
            arr[i], arr[j] = arr[j], arr[i]

    arr[i+1], arr[high] = arr[high], arr[i+1]
    return (i+1)

def quickSort(arr, low, high):
    if len(arr) == 1:
        return arr
    if low < high:
        pi = partition(arr, low, high)
        quickSort(arr, low, pi-1)
        quickSort(arr, pi+1, high)

arr = [3,6,0,10,1,2,1]
n = len(arr)
quickSort(arr, 0, n-1)
print("Sorted array using Quicksort is:")
for i in range(n):
    print("%d" % arr[i]),
```

Sorted array using Quicksort is:
1
1
2

Assignment(Image processing).ipynb

```
#Generating 7 random number using random function and probability distribution
import random
for _ in range(7):
    print(random.gauss(0,1))
```

-1.3893538667068308
-0.9770895045587877
0.30888517842723534
-1.7420909678855905
0.18322373228470956
0.3841724742736049
-0.9181133442598315

[5] #convert 1D to 2D

```
import numpy as np

arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
arr_2d = np.reshape(arr, (2, 5))
# Modify the 2D numpy array (view object)
arr_2d[0][0] = 22
print('1D Numpy array:')
print(arr)
print('2D Numpy array:')
print(arr_2d)
```

1D Numpy array:
[22 1 2 3 4 5 6 7 8 9]
2D Numpy array:
[[22 1 2 3 4]
 [5 6 7 8 9]]