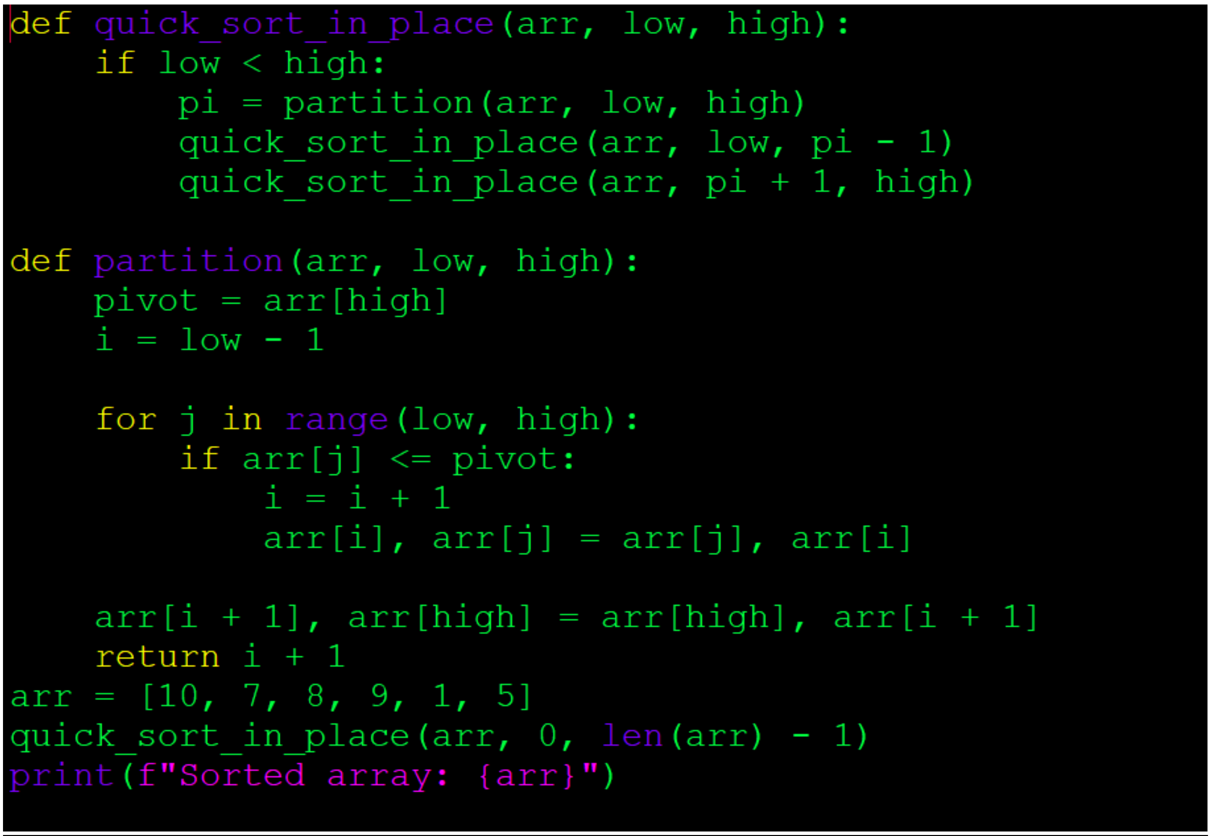
81. Quick sort

Code:



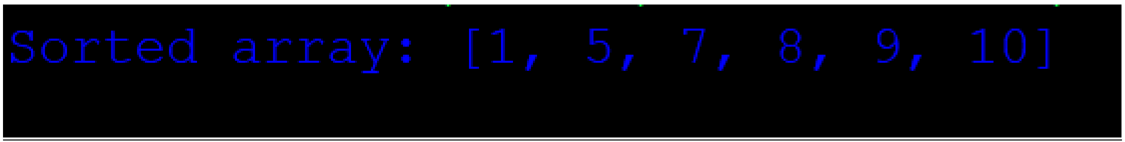
```
def quick_sort_in_place(arr, low, high):
    if low < high:
        pi = partition(arr, low, high)
        quick_sort_in_place(arr, low, pi - 1)
        quick_sort_in_place(arr, pi + 1, high)

def partition(arr, low, high):
    pivot = arr[high]
    i = low - 1

    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
arr = [10, 7, 8, 9, 1, 5]
quick_sort_in_place(arr, 0, len(arr) - 1)
print(f"Sorted array: {arr}")
```

Output:



```
Sorted array: [1, 5, 7, 8, 9, 10]
```

Time Complexity:

- T(n)= O(nlogn)