

Problem -2: Given an array of intervals, merge all the overlapping intervals and return an array of non-overlapping intervals.

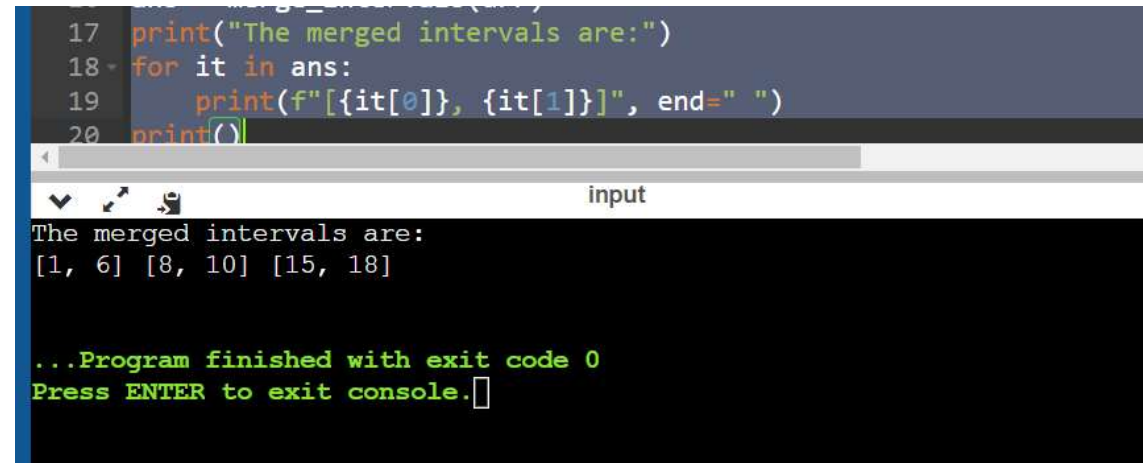
```
def merge_intervals(intervals):  
    if not intervals:  
        return []  
  
    intervals.sort(key=lambda x: x[0])  
    merged = [intervals[0]]  
  
    for interval in intervals[1:]:  
        if interval[0] <= merged[-1][1]:  
            merged[-1][1] = max(merged[-1][1], interval[1])  
        else:  
            merged.append(interval)  
  
    return merged  
  
arr = [[1, 3], [8, 10], [2, 6], [15, 18]]  
ans = merge_intervals(arr)
```

```
print("The merged intervals are:")
```

```
for it in ans:
```

```
    print(f"[{it[0]}, {it[1]]", end=" ")
```

```
print()
```



The screenshot shows a code editor with the following Python code:

```
17 print("The merged intervals are:")
18 for it in ans:
19     print(f"[{it[0]}, {it[1]]", end=" ")
20 print()
```

Below the code editor is a terminal window titled "input". The terminal output is:

```
The merged intervals are:
[1, 6] [8, 10] [15, 18]

...Program finished with exit code 0
Press ENTER to exit console.
```