DAY 42

INTERVIEW BIT PROBLEMS :

1. Merge Intervals

Given a set of non-overlapping intervals, insert a new interval into the intervals (**merge if necessary**).
You may assume that the intervals were initially sorted according to their start times.
Example 1:
Given intervals [1,3],[6,9] insert and merge [2,5] would result in [1,5],[6,9].
Example 2:
Given [1,2],[3,5],[6,7],[8,10],[12,16], insert and merge [4,9] would result
in [1,2],[3,10],[12,16].
This is because the new interval [4,9] overlaps with [3,5],[6,7],[8,10].
Make sure the **returned intervals are also sorted**.

CODE :

PYTHON

```
# Definition for an interval.
# class Interval:
#     def __init__(self, s=0, e=0):
#         self.start = s
#         self.end = e

class Solution:
    def insert(self, intervals, newInterval):
        start = newInterval.start
        end = newInterval.end
        right = 0
        left=0
        while right < len(intervals):
            if start <= intervals[right].end:
                if end < intervals[right].start:
                    break
                start = min(start, intervals[right].start)
                end = max(end, intervals[right].end)
            else:
                left += 1
            right += 1
        return intervals[:left] + [Interval(start, end)] + intervals[right:]
```