729. My Calendar I

Implement a MyCalendar class to store your events. A new event can be added if adding the event will not cause a double booking.

Your class will have the method, book(int start, int end). Formally, this represents a booking on the half open interval [start, end), the range of real numbers x such that start <= x < end.

A double booking happens when two events have some non-empty intersection (ie., there is some time that is common to both events.)

For each call to the method MyCalendar.book, return true if the event can be added to the calendar successfully without causing a double booking. Otherwise, return false and do not add the event to the calendar.

Your class will be called like this: MyCalendar cal = new MyCalendar(); MyCalendar.book(start, end)

Example 1:

```
MyCalendar();
MyCalendar.book(10, 20); // returns true
MyCalendar.book(15, 25); // returns false
MyCalendar.book(20, 30); // returns true
```

```
class MyCalendar {

    TreeMap<Integer, Integer> calendar;

    public MyCalendar() {
        calendar = new TreeMap<>();
    }

    public boolean book(int start, int end) {
        Integer floorKey = calendar.floorKey(start);
        if (floorKey != null &&
           calendar.get(floorKey) > start)
            return false;

        Integer ceilingKey = calendar.ceilingKey(start);
        if (ceilingKey != null &&
           ceilingKey < end)
            return false;

        calendar.put(start, end);
        return true;
    }
}
```