➔ In this problem I come up with 2 different the solutions this first one :

```
➔ select w.id from Weather w
➔ where w.temperature >
➔ ( select (t.temperature )
➔  from Weather t
➔ where t.recordDate = DATE_SUB(w.recordDate,interval 1 day)
➔ );
```

And that one I got up by drawing the 2 tables if I join them in self join and then I treat it like a nested loop logic so that lead me to think about making nested queries and theni will need exactly the yesterday day and to make that I search about a method which making that and I found very beautiful one it is DATE_SUB and the sub here cause the subtracting and it will give you the day that after you subtracting the current day with the interval you defined so I want yesterday so that what lead me to define just one interval

➔ And this solution get bad time

➔ This is the second solution :

here we will need to make a self join to make another copy to use it in the comparing
SELECT w1.id
FROM Weather w1
JOIN Weather w2
WHERE DATEDIFF(w1.recordDate,w2.recordDate)=1 AND w1.temperature > w2.temperature;
And that solution is getting better time complexity

➔ And here is very important information in self joining here if we use the on condition it will give us a wrong answer and that cause if I use ON w1.id = w2.id, it will put the same row info next to each other, and I won't be able to compare two different dates.