

Hard

Table: Student

Column Name	Type
student_id	int
student_name	varchar

student_id is the primary key (column with unique values) for this table.
student_name is the name of the student.

Table: Exam

Column Name	Type
exam_id	int
student_id	int
score	int

(exam_id, student_id) is the primary key (combination of columns with unique values) for this table.
Each row of this table indicates that the student with student_id had a score points in the exam with id exam_id.

A **quiet student** is the one who took at least one exam and did not score the highest or the lowest score.

Write a solution to report the students (student_id, student_name) being quiet in all exams. Do not return the student who has never taken any exam.

Return the result table **ordered** by student_id.

The result format is in the following example.

Example 1:

Input:

Student table:

student_id	student_name
1	Daniel
2	Jade
3	Stella
4	Jonathan
5	Will

Exam table:

exam_id	student_id	score
10	1	70
10	2	80
10	3	90
20	1	80
30	1	70

30	3	80
30	4	90
40	1	60
40	2	70
40	4	80

Output:

student_id	student_name
2	Jade

Explanation:

For exam 1: Student 1 and 3 hold the lowest and high scores respectively.

For exam 2: Student 1 hold both highest and lowest score.

For exam 3 and 4: Student 1 and 4 hold the lowest and high scores respectively.

Student 2 and 5 have never got the highest or lowest in any of the exams.

Since student 5 is not taking any exam, he is excluded from the result.

So, we only return the information of Student 2.

Write your MySQL query statement below

```
-- WITH CTE AS(
--     SELECT exam_id, student_id, RANK() OVER (PARTITION BY exam_id ORDER BY score) AS R
--     FROM EXAM
-- ),
-- MAX_RANK AS (SELECT exam_id, MAX(R) AS R
--     FROM CTE
--     GROUP BY exam_id),
-- STU AS (SELECT DISTINCT CTE.student_id
--     FROM CTE
--     LEFT JOIN MAX_RANK
--     ON CTE.exam_id = MAX_RANK.exam_id AND CTE.R = MAX_RANK.R
--     WHERE MAX_RANK.exam_id IS NOT NULL OR CTE.R=1)

-- SELECT DISTINCT student_id, student_name
-- FROM EXAM e
-- JOIN Student s USING (student_id)
-- WHERE student_id NOT IN (SELECT *
--     FROM STU)
-- ORDER BY student_id
```

```
WITH CTE AS(
SELECT exam_id, e.student_id, student_name,
RANK() OVER(PARTITION BY exam_id ORDER BY score DESC) as max_score_rnk,
RANK() OVER(PARTITION BY exam_id ORDER BY score ASC) as min_score_rnk
FROM Exam e LEFT JOIN Student s ON e.student_id=s.student_id
)
```

```
SELECT DISTINCT student_id, student_name
FROM CTE
WHERE student_id NOT IN (SELECT student_id FROM CTE WHERE max_score_rnk=1 OR
min_score_rnk=1)
ORDER BY student_id
```