

Problem 1988: Find Cutoff Score for Each School

Problem Information

Difficulty: Medium

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

Table:

Schools

+-----+-----+ | Column Name | Type | +-----+-----+ | school_id | int | | capacity | int |
+-----+-----+ school_id is the column with unique values for this table. This table
contains information about the capacity of some schools. The capacity is the maximum
number of students the school can accept.

Table:

Exam

+-----+-----+ | Column Name | Type | +-----+-----+ | score | int | | student_count
| int | +-----+-----+ score is the column with unique values for this table. Each row in
this table indicates that there are student_count students that got at least score points in the
exam. The data in this table will be logically correct, meaning a row recording a higher score
will have the same or smaller student_count compared to a row recording a lower score. More
formally, for every two rows i and j in the table, if score

i

> score

j

then student_count

i

<= student_count

j

.

Every year, each school announces a

minimum score requirement

that a student needs to apply to it. The school chooses the minimum score requirement based on the exam results of all the students:

They want to ensure that even if

every

student meeting the requirement applies, the school can accept everyone.

They also want to

maximize

the possible number of students that can apply.

They

must

use a score that is in the

Exam

table.

Write a solution to report the

minimum score requirement

for each school. If there are multiple score values satisfying the above conditions, choose the

smallest

one. If the input data is not enough to determine the score, report

-1

Return the result table in

any order

The result format is in the following example.

Example 1:

Input:

Schools table: +-----+-----+ | school_id | capacity | +-----+-----+ | 11 | 151 || 5 |
48 || 9 | 9 || 10 | 99 | +-----+-----+ Exam table: +-----+-----+ | score |
student_count | +-----+-----+ | 975 | 10 || 966 | 60 || 844 | 76 || 749 | 76 || 744 | 100 |
+-----+-----+

Output:

+-----+-----+ | school_id | score | +-----+-----+ | 5 | 975 || 9 | -1 || 10 | 749 || 11 |
744 | +-----+-----+

Explanation:

- School 5: The school's capacity is 48. Choosing 975 as the min score requirement, the school will get at most 10 applications, which is within capacity. - School 10: The school's capacity is 99. Choosing 844 or 749 as the min score requirement, the school will get at most 76 applications, which is within capacity. We choose the smallest of them, which is 749. - School 11: The school's capacity is 151. Choosing 744 as the min score requirement, the school will get at most 100 applications, which is within capacity. - School 9: The data given is not enough to determine the min score requirement. Choosing 975 as the min score, the school may get 10 requests while its capacity is 9. We do not have information about higher scores, hence we report -1.

Code Snippets

MySQL:

```
# Write your MySQL query statement below
```

MS SQL Server:

```
/* Write your T-SQL query statement below */
```

PostgreSQL:

```
-- Write your PostgreSQL query statement below
```

Oracle:

```
/* Write your PL/SQL query statement below */
```

Pandas:

```
import pandas as pd

def find_cutoff_score(schools: pd.DataFrame, exam: pd.DataFrame) ->
    pd.DataFrame:
```

Solutions

MySQL Solution:

```
# Write your MySQL query statement below
```

MS SQL Server Solution:

```
/* Write your T-SQL query statement below */
```

PostgreSQL Solution:

```
-- Write your PostgreSQL query statement below
```

Oracle Solution:

```
/* Write your PL/SQL query statement below */
```

Pandas Solution:

```
import pandas as pd

def find_cutoff_score(schools: pd.DataFrame, exam: pd.DataFrame) ->
    pd.DataFrame:
```