

# Problem 1459: Rectangles Area

## Problem Information

**Difficulty:** Medium

**Acceptance Rate:** 0.00%

**Paid Only:** No

## Problem Description

Table:

Points

Column Name	Type
id	int
x_value	int
y_value	int

id is the column with unique values for this table. Each point is represented as a 2D coordinate (x\_value, y\_value).

Write a solution to report all possible

axis-aligned

rectangles with a

non-zero area

that can be formed by any two points from the

Points

table.

Each row in the result should contain three columns

(p1, p2, area)

where:

p1

and

p2

are the

id

's of the two points that determine the opposite corners of a rectangle.

area

is the area of the rectangle and must be

non-zero

Return the result table

ordered

by

area

in descending order

. If there is a tie, order them by

p1

in ascending order

. If there is still a tie, order them by

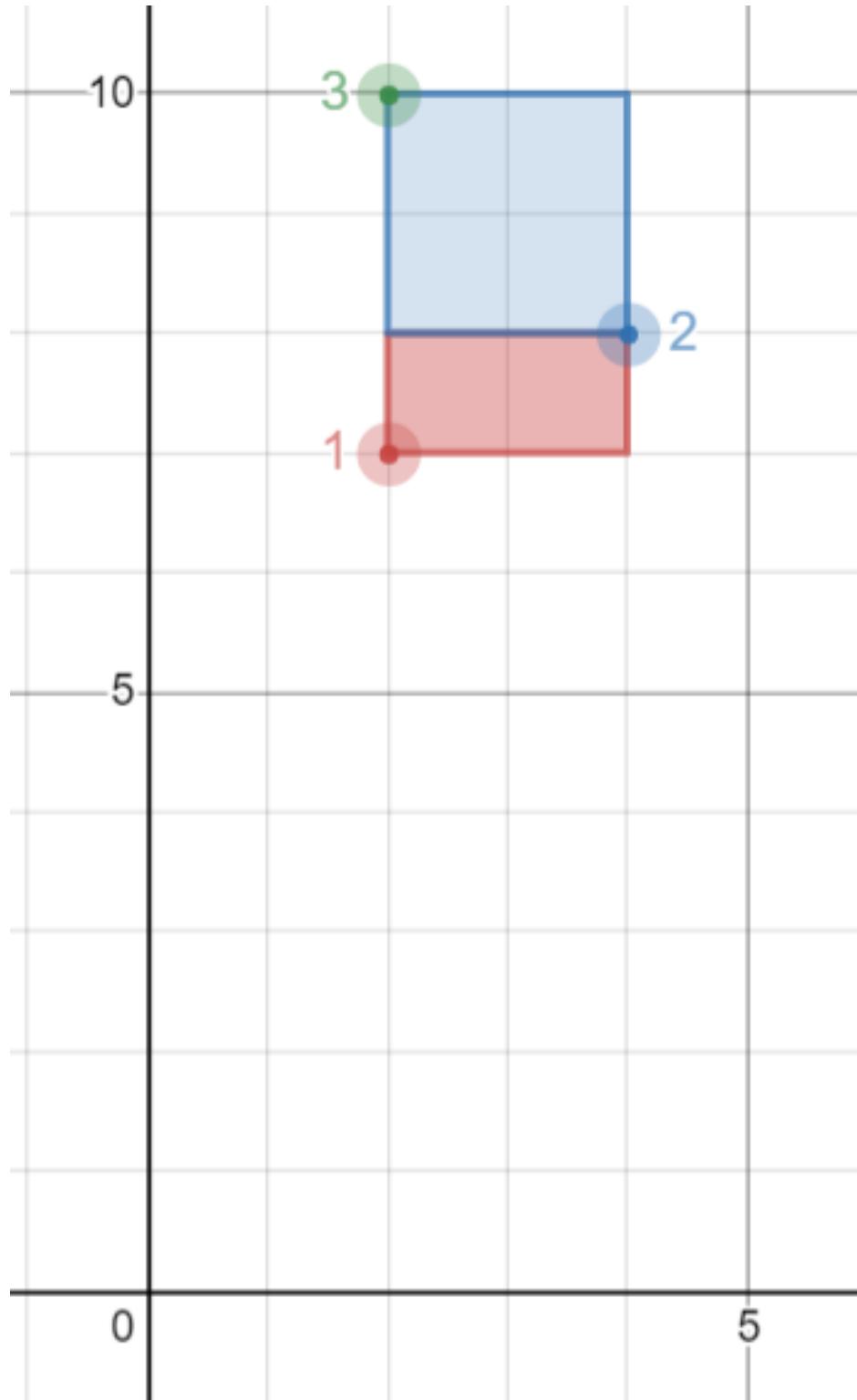
p2

in ascending order

.

The result format is in the following table.

Example 1:



Input:

Points table: +-----+-----+-----+ | id | x\_value | y\_value |  
+-----+-----+-----+ | 1 | 2 | 7 | | 2 | 4 | 8 | | 3 | 2 | 10 |  
+-----+-----+-----+

Output:

		p1	p2	area				2	3	4	
1	2	2						2	3	4	

Explanation:

The rectangle formed by  $p1 = 2$  and  $p2 = 3$  has an area equal to  $|4-2| * |8-10| = 4$ . The rectangle formed by  $p1 = 1$  and  $p2 = 2$  has an area equal to  $|2-4| * |7-8| = 2$ . Note that the rectangle formed by  $p1 = 1$  and  $p2 = 3$  is invalid because the area is 0.

## 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 rectangles_area(points: 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 rectangles_area(points: pd.DataFrame) -> pd.DataFrame:
```