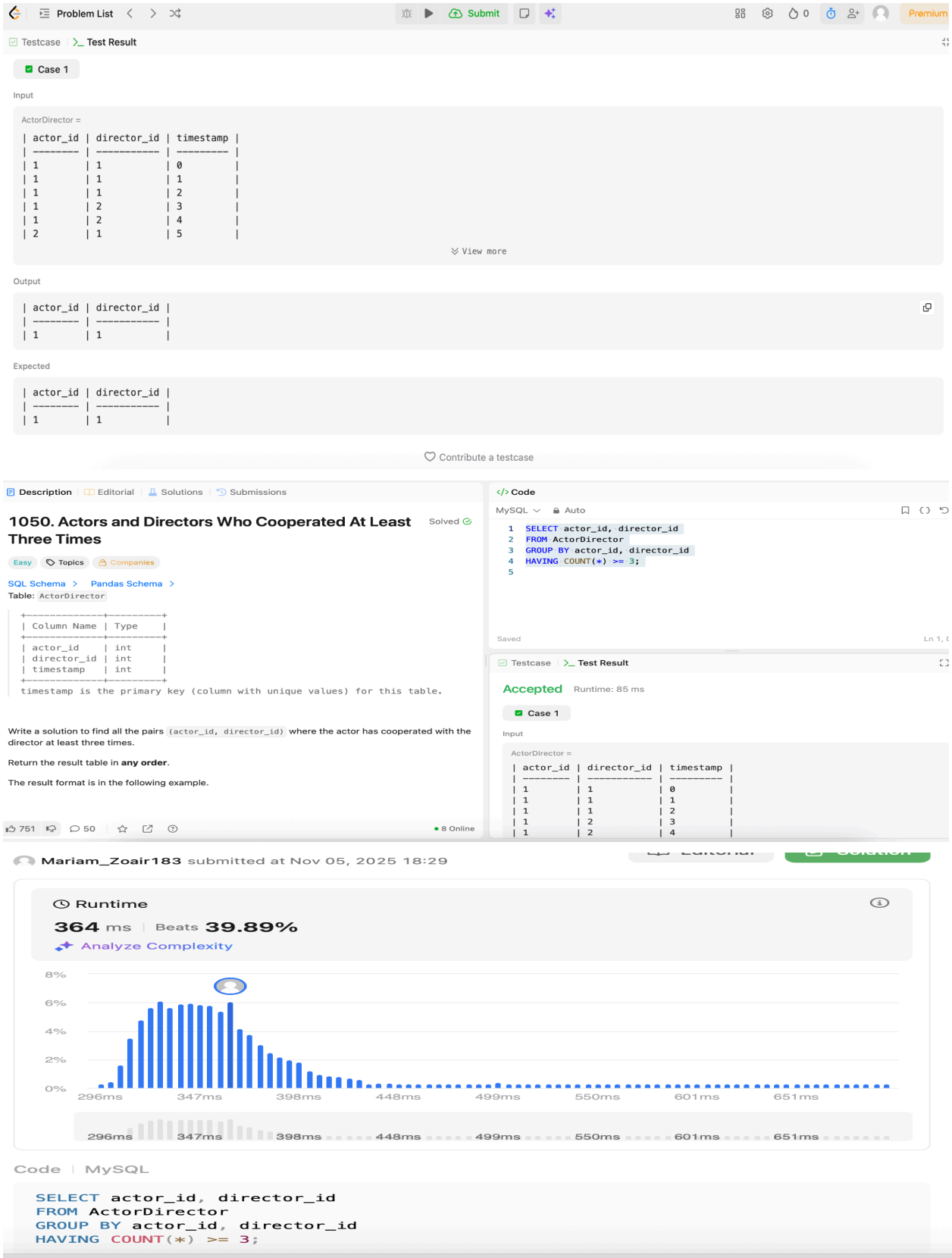
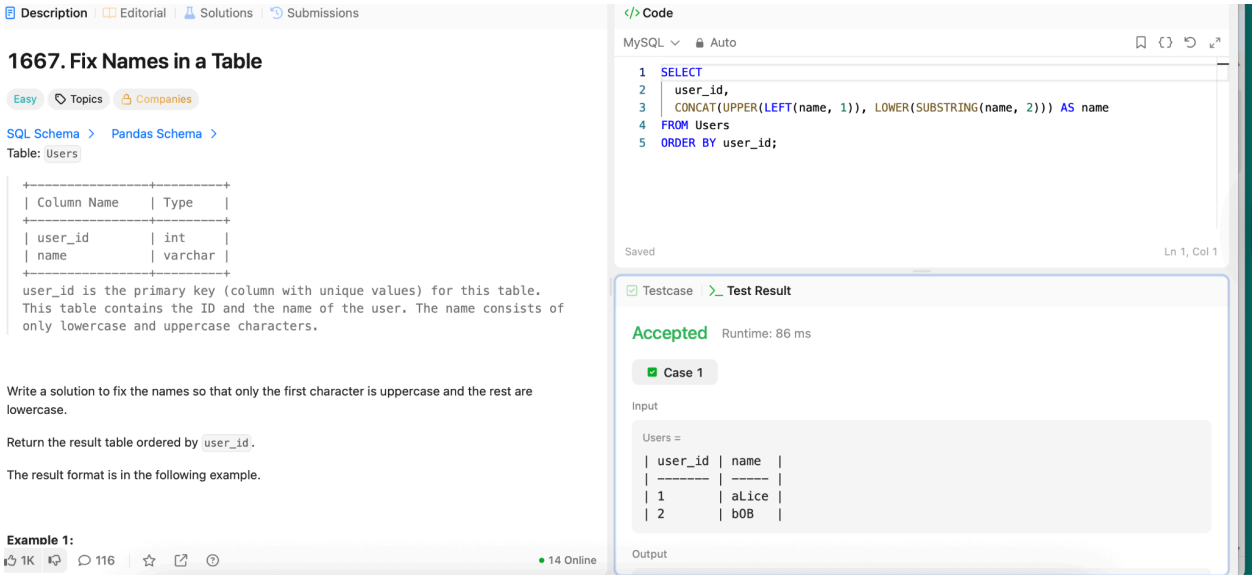
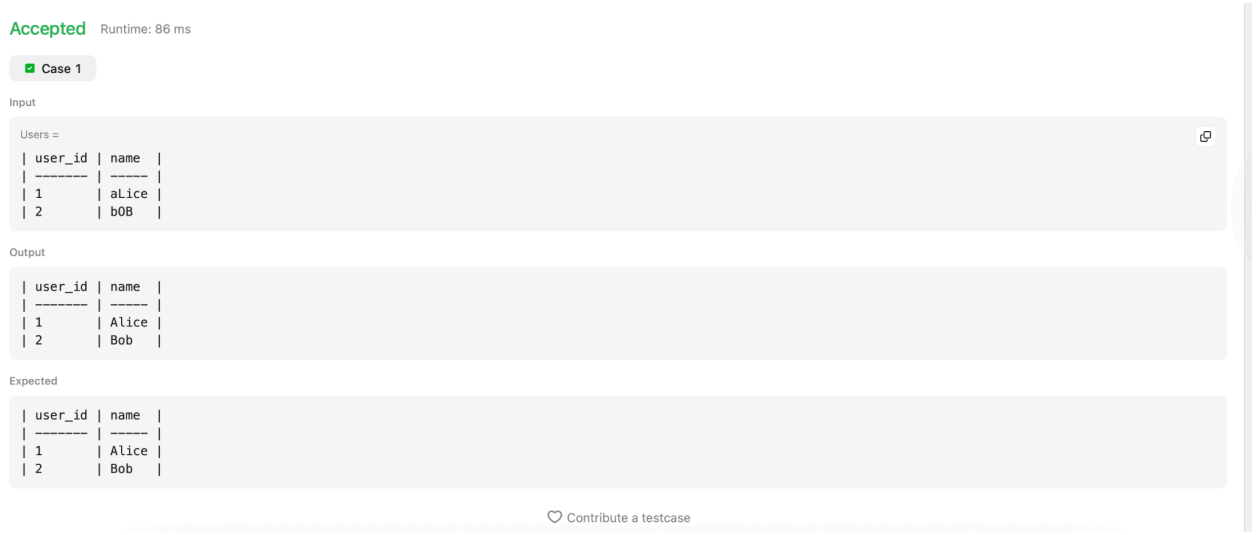
Question #1

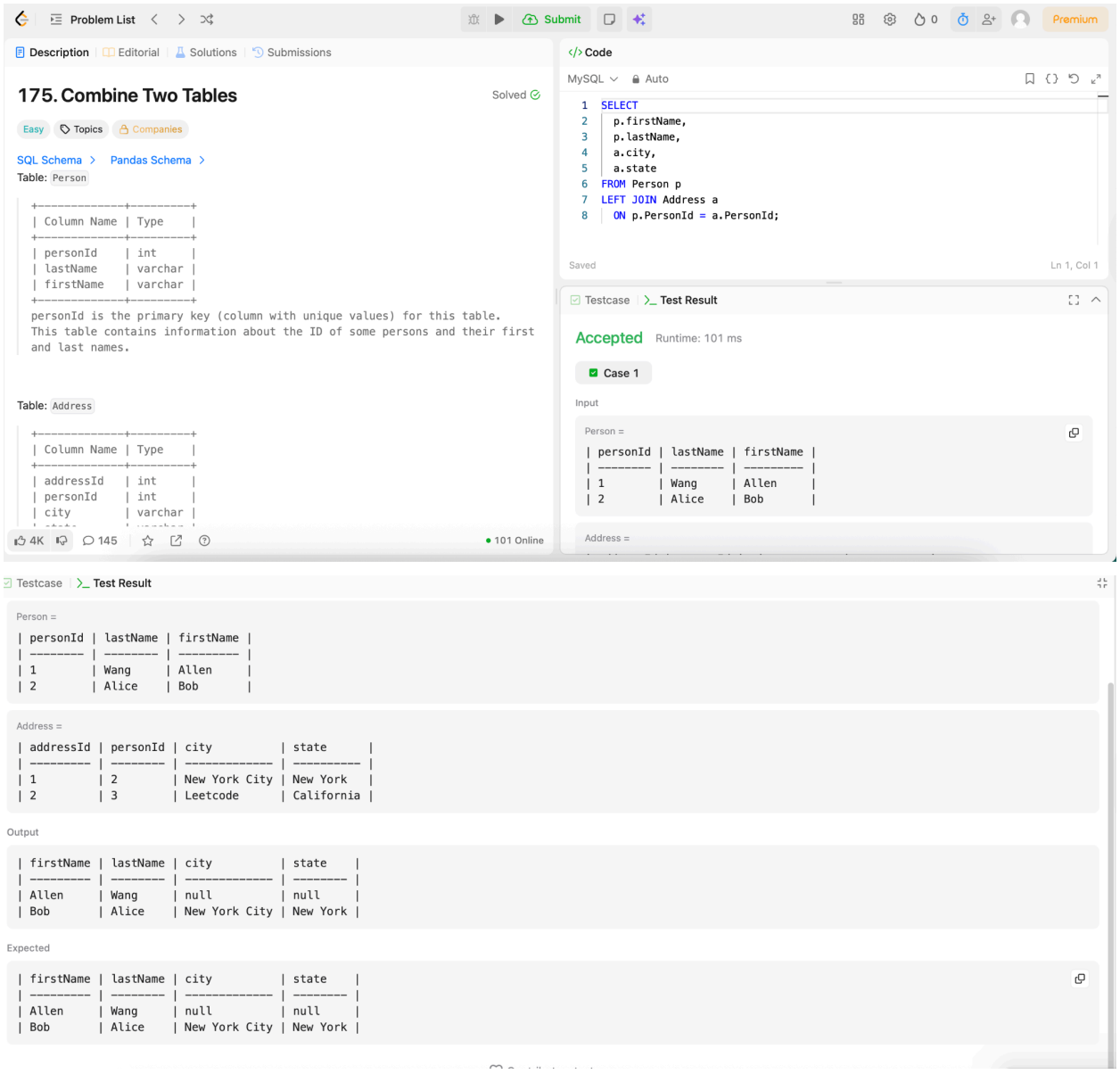

Problem List
Submit
Premium
Testcase
Test Result
Case 1
Input
ActorDirector =
actor_id	director_id	timestamp
1	1	0
1	1	1
1	1	2
1	2	3
1	2	4
2	1	5
View more
Output
actor_id	director_id
1	1
Expected
actor_id	director_id
1	1
Contribute a testcase
Description
Editorial
Solutions
Submissions
1050. Actors and Directors Who Cooperated At Least Three Times
Solved
Easy
Topics
Companies
SQL Schema
Pandas Schema
Table: ActorDirector
Column Name	Type
actor_id	int
director_id	int
timestamp	int
timestamp is the primary key (column with unique values) for this table.
Write a solution to find all the pairs (actor_id, director_id) where the actor has cooperated with the director at least three times.
Return the result table in any order.
The result format is in the following example.
751
50
8 Online
Code
MySQL
Auto
SELECT actor_id, director_id
FROM ActorDirector
GROUP BY actor_id, director_id
HAVING COUNT(*) >= 3;
Saved
Testcase
Test Result
Accepted
Runtime: 85 ms
Case 1
Input
ActorDirector =
actor_id	director_id	timestamp
1	1	0
1	1	1
1	1	2
1	2	3
1	2	4
Mariam_Zoair183 submitted at Nov 05, 2025 18:29
Runtime
364 ms | Beats 39.89%
Analyze Complexity
8%
6%
4%
2%
0%
296ms
347ms
398ms
448ms
499ms
550ms
601ms
651ms
Code | MySQL
SELECT actor_id, director_id
FROM ActorDirector
GROUP BY actor_id, director_id
HAVING COUNT(*) >= 3;

Question #2



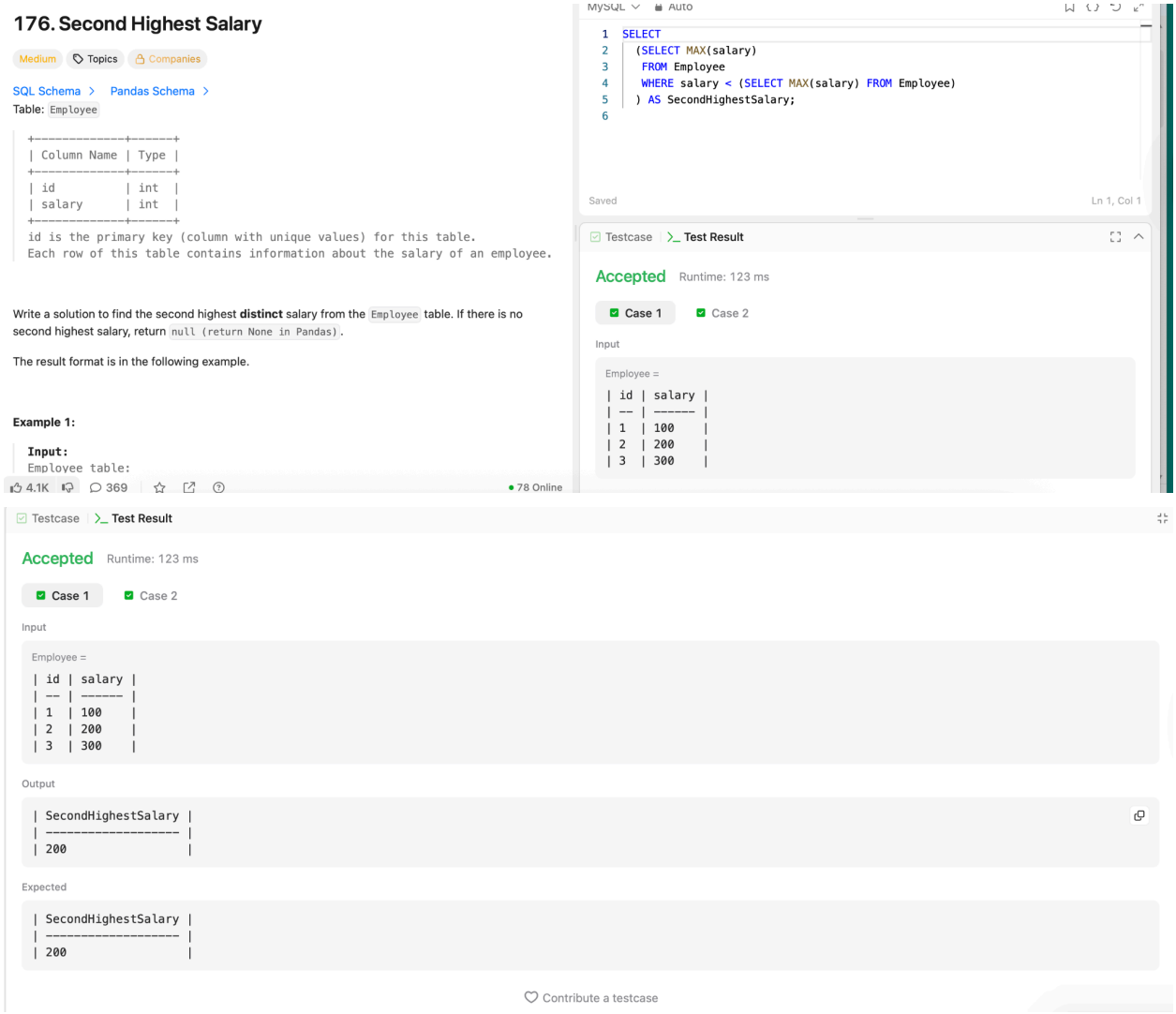


Question #3


Problem List
Submit
Premium
Description
Editorial
Solutions
Submissions
175. Combine Two Tables
Solved
Easy
Topics
Companies
SQL Schema
Pandas Schema
Table: Person
+-------------+---------+
| Column Name | Type |
+-------------+---------+
personId	int
lastName	varchar
firstName	varchar
+-------------+---------+
personId is the primary key (column with unique values) for this table.
This table contains information about the ID of some persons and their first and last names.
Table: Address
+-------------+---------+
| Column Name | Type |
+-------------+---------+
addressId	int
personId	int
city	varchar
4K
145
101 Online
Code
MySQL
Auto
1 SELECT
2 p.firstName,
3 p.lastName,
4 a.city,
5 a.state
6 FROM Person p
7 LEFT JOIN Address a
8 ON p.PersonId = a.PersonId;
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 101 ms
Case 1
Input
Person =
personId	lastName	firstName
1	Wang	Allen
2	Alice	Bob
Address =

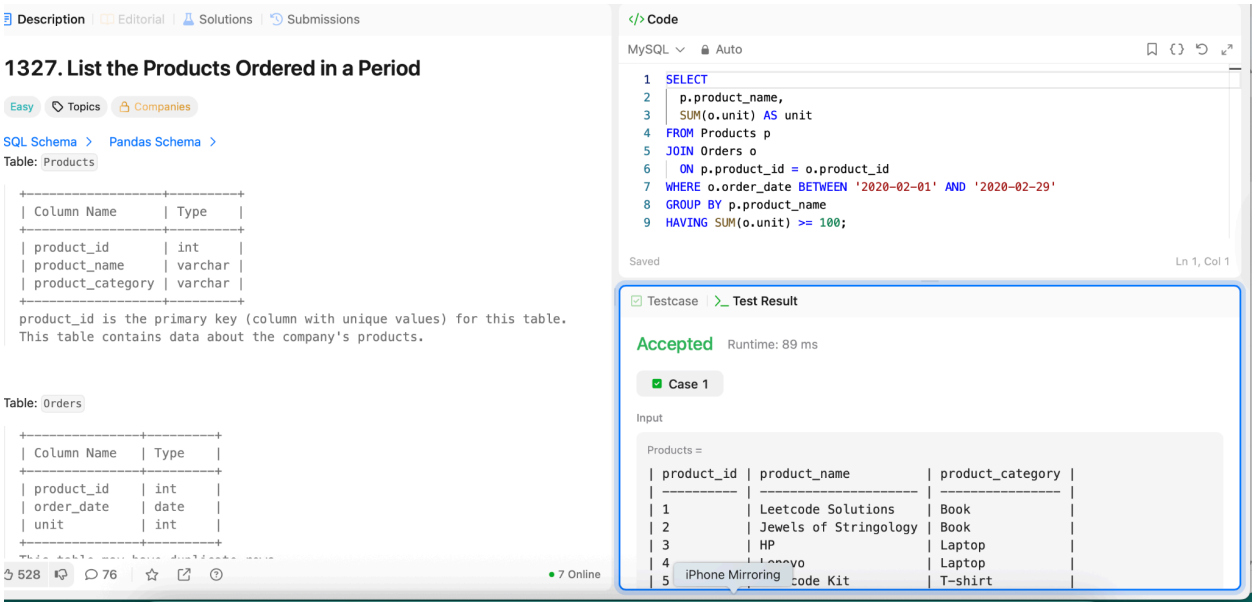
Testcase
Test Result
Person =
personId	lastName	firstName
1	Wang	Allen
2	Alice	Bob
Address =
addressId	personId	city	state
1	2	New York City	New York
2	3	Leetcode	California
Output
firstName	lastName	city	state
Allen	Wang	null	null
Bob	Alice	New York City	New York
Expected
firstName	lastName	city	state
Allen	Wang	null	null
Bob	Alice	New York City	New York

Question #4

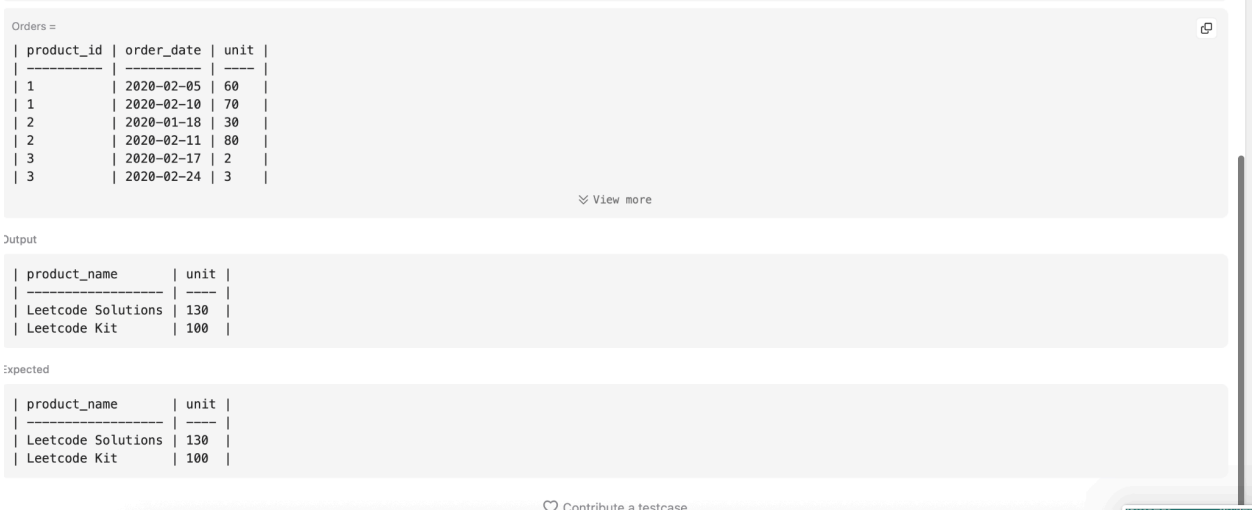

176. Second Highest Salary
Medium
Topics
Companies
SQL Schema
Pandas Schema
Table: Employee
+-------------+------+
| Column Name | Type |
+-------------+------+
| id | int |
| salary | int |
+-------------+------+
id is the primary key (column with unique values) for this table.
Each row of this table contains information about the salary of an employee.
Write a solution to find the second highest distinct salary from the Employee table. If there is no second highest salary, return null (return None in Pandas).
The result format is in the following example.
Example 1:
Input:
Employee table:
4.1K
369
78 Online
MySQL
Auto
1 SELECT
2 (SELECT MAX(salary)
3 FROM Employee
4 WHERE salary < (SELECT MAX(salary) FROM Employee)
5) AS SecondHighestSalary;
6
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 123 ms
Case 1
Case 2
Input
Employee =
id	salary
1	100
2	200
3	300

Testcase
Test Result
Accepted
Runtime: 123 ms
Case 1
Case 2
Input
Employee =
id	salary
1	100
2	200
3	300
Output
SecondHighestSalary
200
Expected
SecondHighestSalary
200
Contribute a testcase

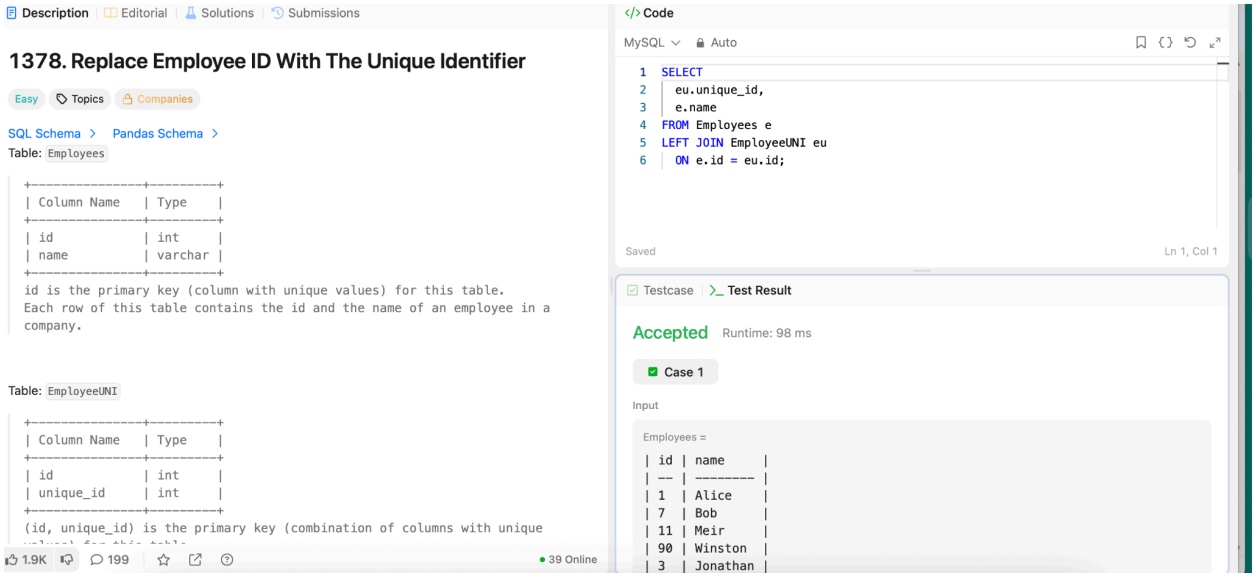
Question #5


Description
Editorial
Solutions
Submissions
1327. List the Products Ordered in a Period
Easy
Topics
Companies
SQL Schema
Pandas Schema
Table: Products
+------------------+---------+
| Column Name | Type |
+------------------+---------+
product_id	int
product_name	varchar
product_category	varchar
+------------------+---------+
product_id is the primary key (column with unique values) for this table.
This table contains data about the company's products.
Table: Orders
+---------------+---------+
| Column Name | Type |
+---------------+---------+
product_id	int
order_date	date
unit	int
+---------------+---------+
528
76
7 Online
Code
MySQL
Auto
SELECT
p.product_name,
SUM(o.unit) AS unit
FROM Products p
JOIN Orders o
ON p.product_id = o.product_id
WHERE o.order_date BETWEEN '2020-02-01' AND '2020-02-29'
GROUP BY p.product_name
HAVING SUM(o.unit) >= 100;
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 89 ms
Case 1
Input
Products =
product_id	product_name	product_category
1	Leetcode Solutions	Book
2	Jewels of Stringology	Book
3	HP	Laptop
4	Lenovo	Laptop
5	Leetcode Kit	T-shirt
iPhone Mirroring

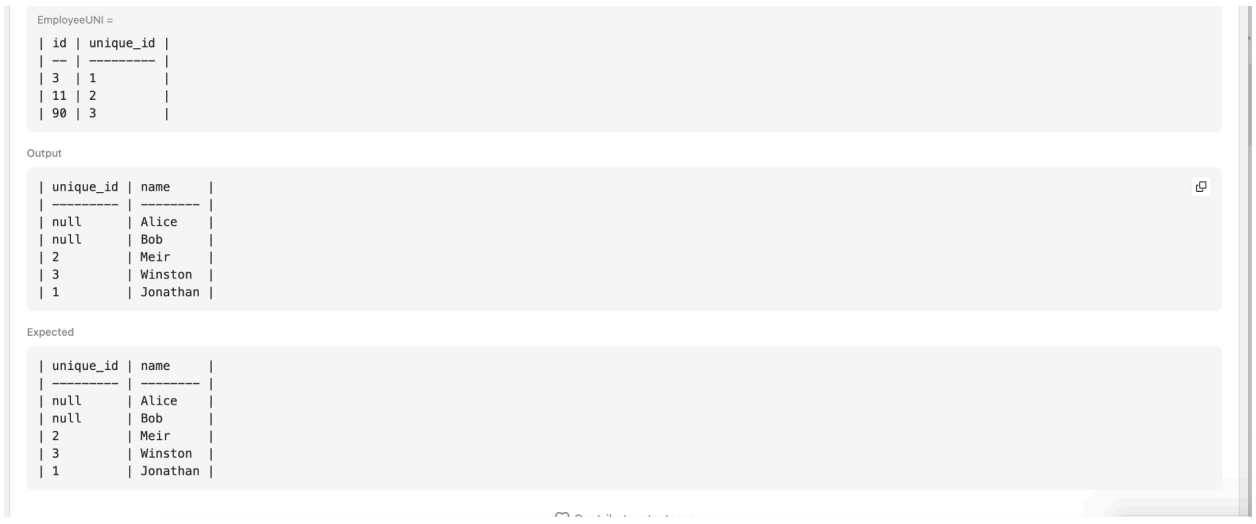

Accepted
Runtime: 89 ms
Case 1
Input
Products =
product_id	product_name	product_category
1	Leetcode Solutions	Book
2	Jewels of Stringology	Book
3	HP	Laptop
4	Lenovo	Laptop
5	Leetcode Kit	T-shirt
Orders =
product_id	order_date	unit
1	2020-02-05	60
1	2020-02-10	70
2	2020-01-18	30
2	2020-02-11	80
3	2020-02-17	2
3	2020-02-24	3
View more
Output


Orders =
product_id	order_date	unit
1	2020-02-05	60
1	2020-02-10	70
2	2020-01-18	30
2	2020-02-11	80
3	2020-02-17	2
3	2020-02-24	3
View more
Output
product_name	unit
Leetcode Solutions	130
Leetcode Kit	100
Expected
product_name	unit
Leetcode Solutions	130
Leetcode Kit	100
Contribute a testcase

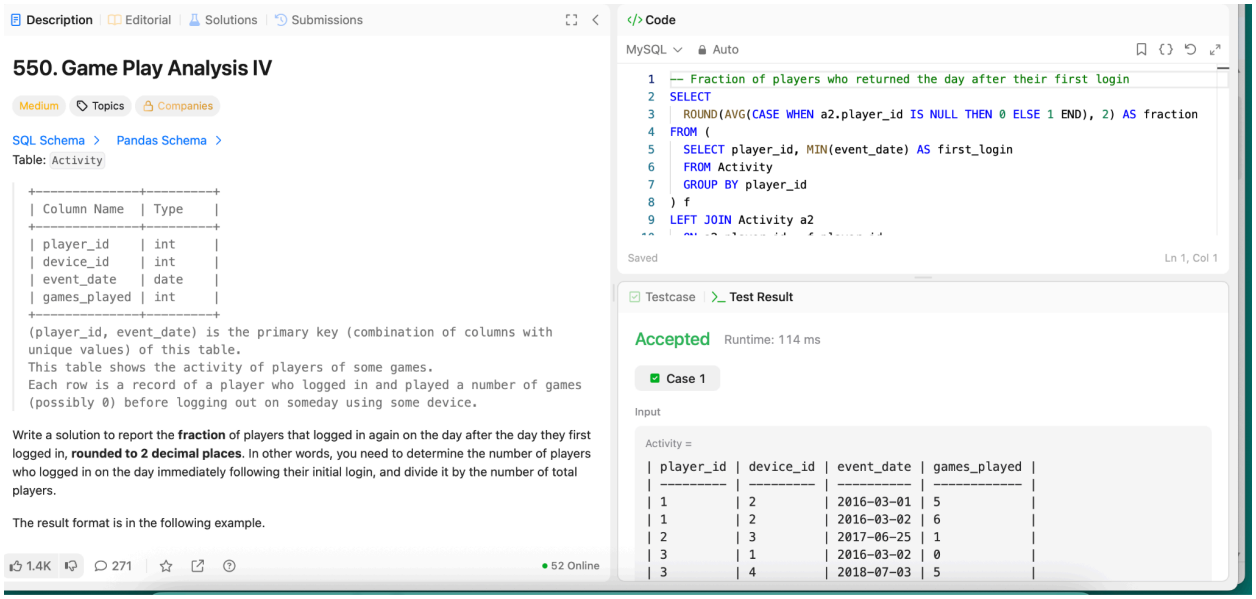
Question #6

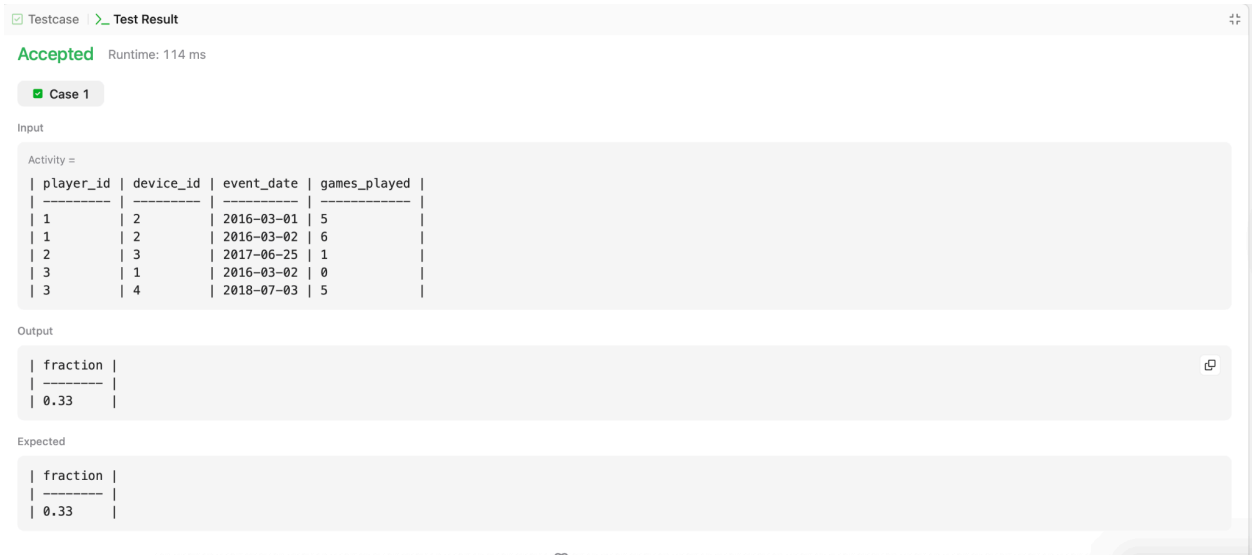

Description
Editorial
Solutions
Submissions
1378. Replace Employee ID With The Unique Identifier
Easy
Topics
Companies
SQL Schema
Pandas Schema
Table: Employees
+---------------+---------+
| Column Name | Type |
+---------------+---------+
| id | int |
| name | varchar |
+---------------+---------+
id is the primary key (column with unique values) for this table.
Each row of this table contains the id and the name of an employee in a company.
Table: EmployeeUNI
+---------------+---------+
| Column Name | Type |
+---------------+---------+
| id | int |
| unique_id | int |
+---------------+---------+
(id, unique_id) is the primary key (combination of columns with unique
1.9K
199
39 Online
Code
MySQL
Auto
SELECT
eu.unique_id,
e.name
FROM Employees e
LEFT JOIN EmployeeUNI eu
ON e.id = eu.id;
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 98 ms
Case 1
Input
Employees =
id	name
1	Alice
7	Bob
11	Meir
90	Winston
3	Jonathan


Testcase
Test Result
Input
Employees =
id	name
1	Alice
7	Bob
11	Meir
90	Winston
3	Jonathan
EmployeeUNI =
id	unique_id
3	1
11	2
90	3
Output
unique_id	name
null	Alice
null	Bob
2	Meir
3	Winston
1	Jonathan

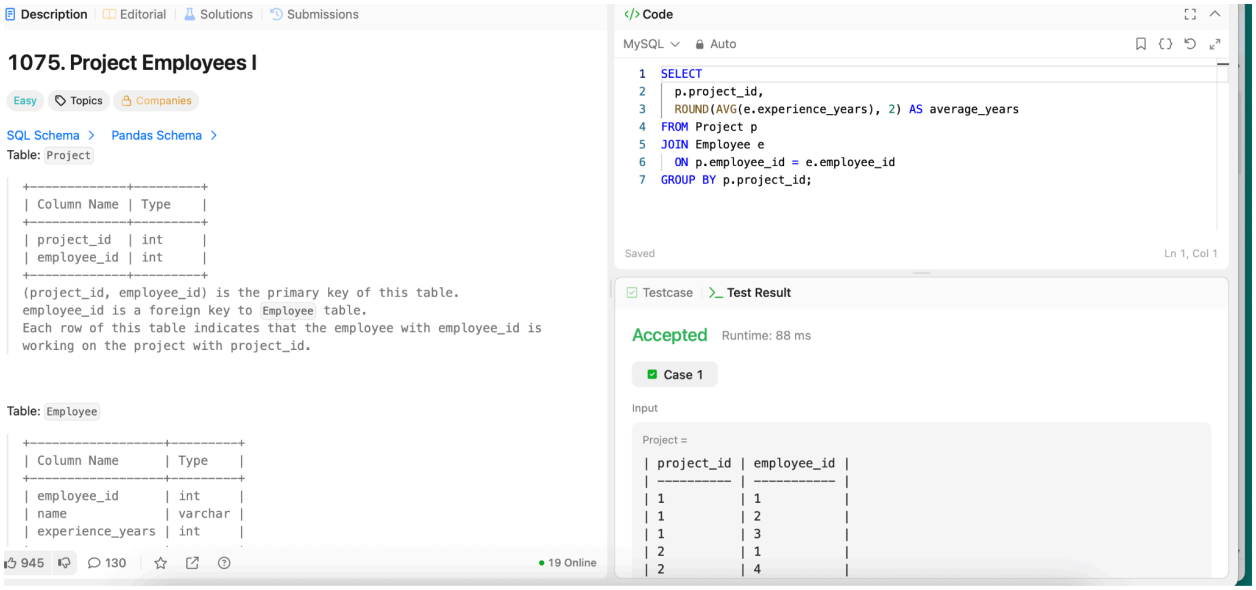

EmployeeUNI =
id	unique_id
3	1
11	2
90	3
Output
unique_id	name
null	Alice
null	Bob
2	Meir
3	Winston
1	Jonathan
Expected
unique_id	name
null	Alice
null	Bob
2	Meir
3	Winston
1	Jonathan

Question #7

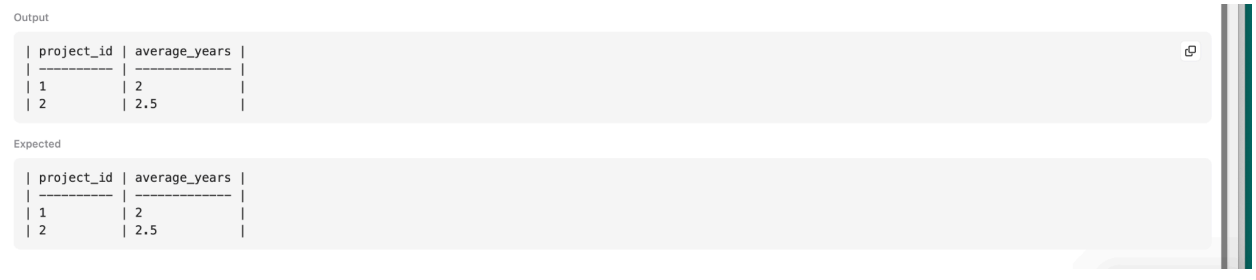




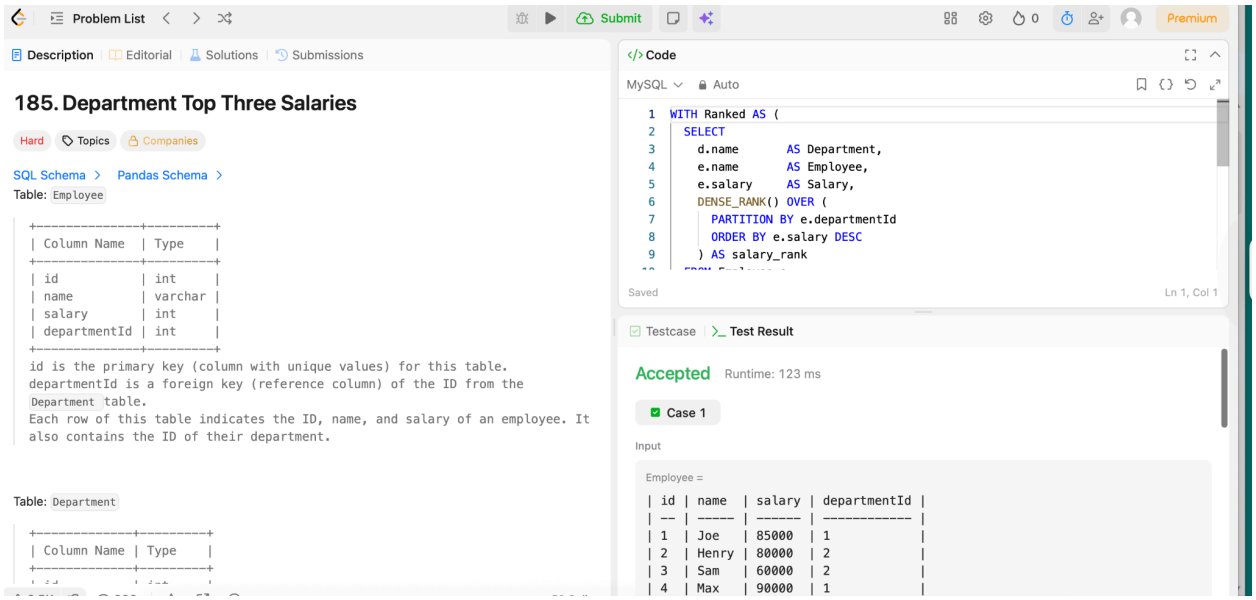
Question # 8

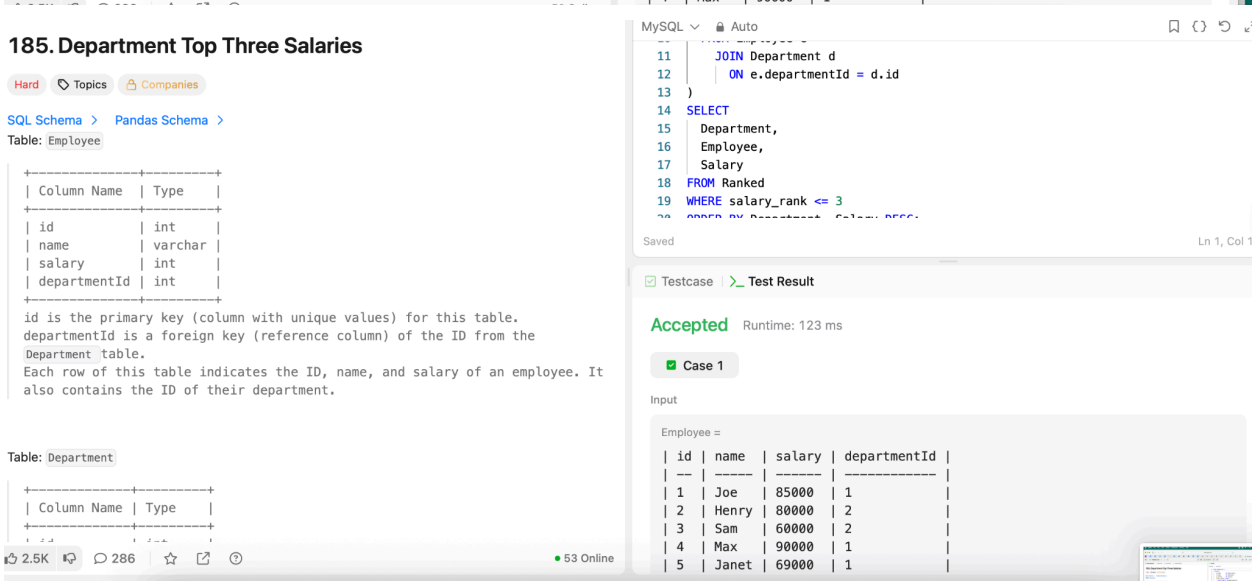

Description
Editorial
Solutions
Submissions
1075. Project Employees I
Easy
Topics
Companies
SQL Schema
Pandas Schema
Table: Project
+-------------+---------+
| Column Name | Type |
+-------------+---------+
| project_id | int |
| employee_id | int |
+-------------+---------+
(project_id, employee_id) is the primary key of this table.
employee_id is a foreign key to Employee table.
Each row of this table indicates that the employee with employee_id is working on the project with project_id.
Table: Employee
+------------------+---------+
| Column Name | Type |
+------------------+---------+
employee_id	int
name	varchar
experience_years	int
945
130
19 Online
Code
MySQL
Auto
1 SELECT
2 p.project_id,
3 ROUND(AVG(e.experience_years), 2) AS average_years
4 FROM Project p
5 JOIN Employee e
6 ON p.employee_id = e.employee_id
7 GROUP BY p.project_id;
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 88 ms
Case 1
Input
Project =
project_id	employee_id
1	1
1	2
1	3
2	1
2	4

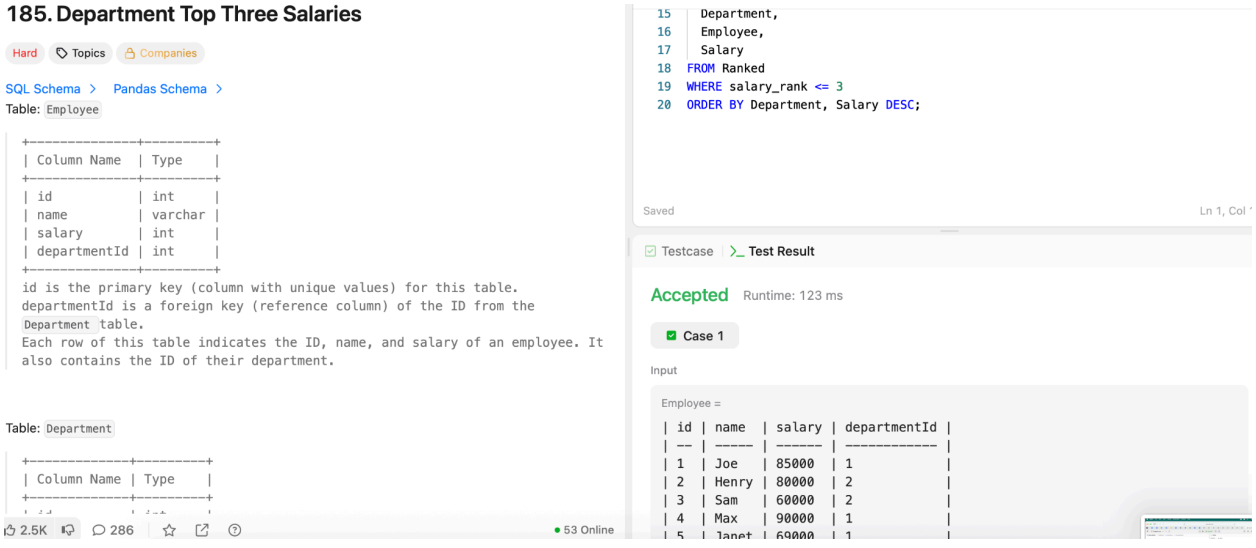

Testcase
Test Result
Project =
project_id	employee_id
1	1
1	2
1	3
2	1
2	4
Employee =
employee_id	name	experience_years
1	Khaled	3
2	Ali	2
3	John	1
4	Doe	2
Output
project_id	average_years
1	2
2	2.5


Output
project_id	average_years
1	2
2	2.5
Expected
project_id	average_years
1	2
2	2.5

Question #9


Problem List
Submit
Premium
Description
Editorial
Solutions
Submissions
Code
MySQL
Auto
185. Department Top Three Salaries
Hard
Topics
Companies
SQL Schema
Pandas Schema
Table: Employee
Column Name	Type
id	int
name	varchar
salary	int
departmentId	int
id is the primary key (column with unique values) for this table.
departmentId is a foreign key (reference column) of the ID from the Department table.
Each row of this table indicates the ID, name, and salary of an employee. It also contains the ID of their department.
Table: Department
Column Name	Type
WITH Ranked AS (
SELECT
d.name AS Department,
e.name AS Employee,
e.salary AS Salary,
DENSE_RANK() OVER (
PARTITION BY e.departmentId
ORDER BY e.salary DESC
) AS salary_rank
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 123 ms
Case 1
Input
Employee =
id	name	salary	departmentId
1	Joe	85000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1


185. Department Top Three Salaries
Hard
Topics
Companies
SQL Schema
Pandas Schema
Table: Employee
Column Name	Type
id	int
name	varchar
salary	int
departmentId	int
id is the primary key (column with unique values) for this table.
departmentId is a foreign key (reference column) of the ID from the Department table.
Each row of this table indicates the ID, name, and salary of an employee. It also contains the ID of their department.
Table: Department
Column Name	Type
2.5K
286
53 Online
MySQL
Auto
JOIN Department d
ON e.departmentId = d.id
)
SELECT
Department,
Employee,
Salary
FROM Ranked
WHERE salary_rank <= 3
Saved
Ln 1, Col 1
Testcase
Test Result
Accepted
Runtime: 123 ms
Case 1
Input
Employee =
id	name	salary	departmentId
1	Joe	85000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1
5	Janet	69000	1


185. Department Top Three Salaries
Hard
Topics
Companies
SQL Schema
Pandas Schema
Table: Employee
Column Name	Type
id	int
name	varchar
salary	int
departmentId	int
id is the primary key (column with unique values) for this table.
departmentId is a foreign key (reference column) of the ID from the Department table.
Each row of this table indicates the ID, name, and salary of an employee. It also contains the ID of their department.
Table: Department
Column Name	Type
2.5K
286
53 Online
Department,
Employee,
Salary
FROM Ranked
WHERE salary_rank <= 3
ORDER BY Department, Salary DESC;
Saved
Testcase
Test Result
Accepted
Runtime: 123 ms
Case 1
Input
Employee =
id	name	salary	departmentId
1	Joe	85000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1
5	Janet	69000	1

Input

Employee =

id	name	salary	departmentId
1	Joe	85000	1
2	Henry	80000	2
3	Sam	60000	2
4	Max	90000	1
5	Janet	69000	1
6	Randy	85000	1

View more

Department =

id	name
1	IT
2	Sales

Output

Department	Employee	Salary
IT	Max	90000
IT	Joe	85000
IT	Randy	85000
IT	Will	70000
Sales	Henry	80000
Sales	Sam	60000

Expected

Department	Employee	Salary
IT	Joe	85000
Sales	Henry	80000
Sales	Sam	60000
IT	Max	90000
IT	Randy	85000
IT	Will	70000