Name:- Vaibhav kumar gupta

Date:- 10-02-2025

1. Relational Database Management System (RDBMS)

RDBMS is a type of database management system that stores data in a structured format using tables. It ensures data integrity, consistency, and supports relationships between tables.

2. Table and Database

- Database: A collection of organized data that can be easily accessed, managed, and updated.
- Table: A structured collection of related data in rows and columns within a database.

Commands:-

1 Creating a Database

CREATE DATABASE my_database;

Explanation:- This command creates a new database named my_database.

2. Creating a Table

```
id INT PRIMARY KEY,
name VARCHAR(100),
age INT,
department VARCHAR(50)
);
```

Explanation:- Defines an employees table with relevant columns.

3. Inserting Data

```
INSERT INTO employees (id, name, age, department)
```

```
VALUES (1, 'Alice', 30, 'HR');
```

Explanation:- Adds a record to the employees table.

4. Deleting Data

DELETE FROM employees WHERE id = 1;

Explanation:- Removes the record where id is 1.

5. Truncating a Table

TRUNCATE TABLE employees;

Explanation:- Clears all data from the employees table.

6. Renaming a Table

ALTER TABLE employees RENAME TO staff;

Explanation:- Changes table name from employees to staff.

7. Modifying a Column

ALTER TABLE employees RENAME COLUMN department TO dept;

Explanation:- Renames department column to dept.

8. Using Views

CREATE VIEW hr_employees AS

SELECT name, department FROM employees WHERE department = 'HR';

Explanation:- Creates a hr_employees view.

9. Ordering Data

SELECT * FROM employees ORDER BY age DESC;

Explanation:- Sorts results by age in descending order.

10. Filtering Data

SELECT * FROM employees WHERE age > 25;

Explanation:- Selects employees older than 25.

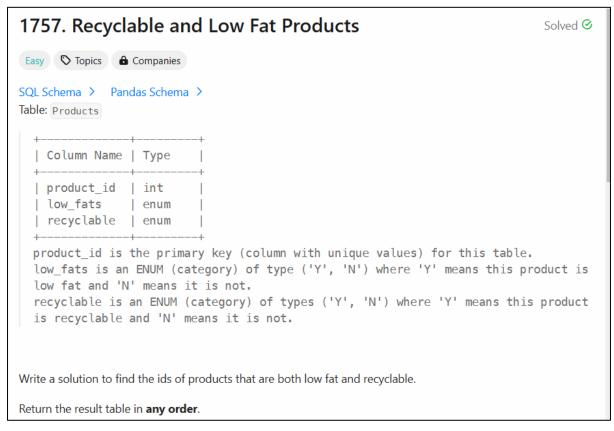
11. Deep vs Shallow Copy

• Shallow Copy:

CREATE TABLE employees_copy AS SELECT * FROM employees;

• Deep Copy:

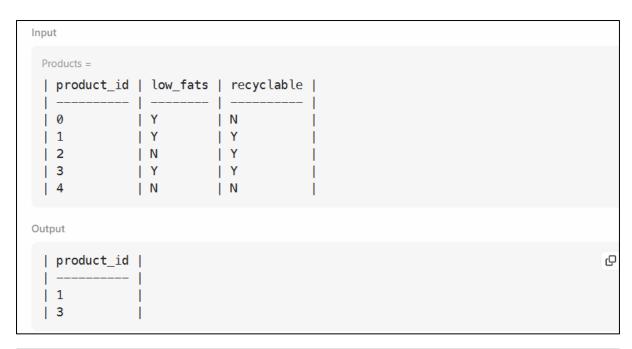
CREATE TABLE employees_copy like employees;

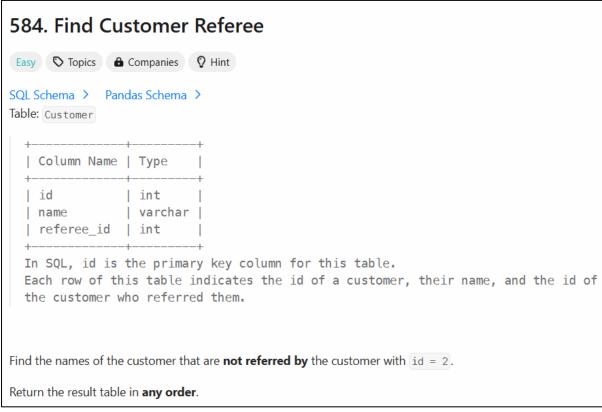


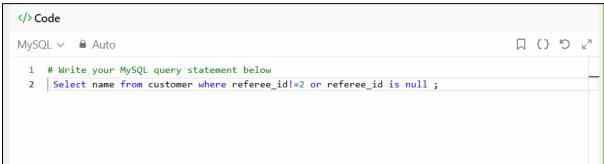
```
// Code

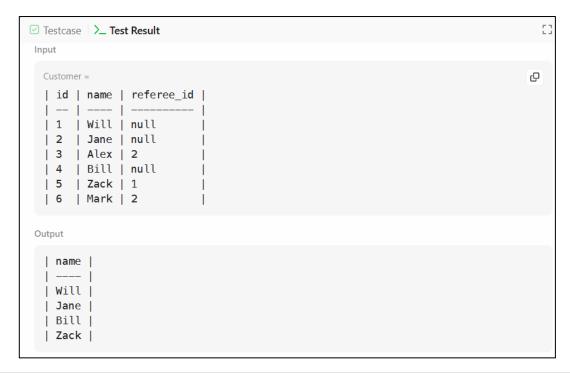
MySQL ∨ ♠ Auto

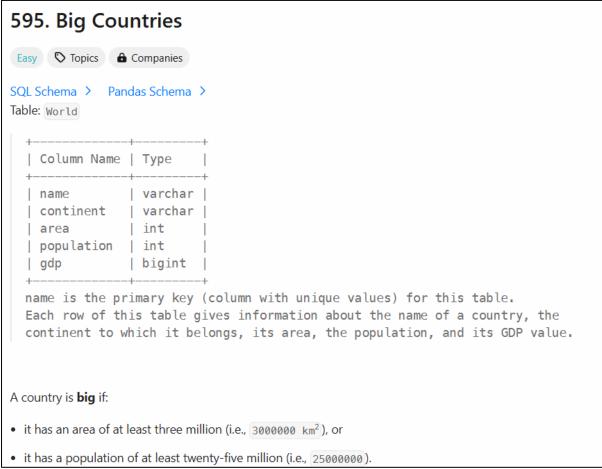
1 # Write your MySQL query statement below
2 select product_id from products where low_fats ='Y' and recyclable='Y';
3
```

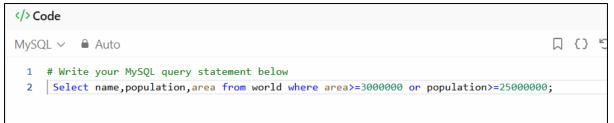












Input					
World =					
name	continent	area	population	gdp	
Algeria Andorra Angola	Europe Africa	652230 28748 2381741 468 1246700	25500100 2831741 37100000 78115 20609294	20343000000 12960000000 188681000000 3712000000 1009900000000	
Output	l nonulation	Larga	1		P
name Afghanistan Algeria	population 25500100 37100000	 652230 2381741			٦

1148. Article Views I



SQL Schema > Pandas Schema >

Table: Views

Column Name	Type
article_id	int
author_id	int
viewer_id	int
view_date	date

There is no primary key (column with unique values) for this table, the table may have duplicate rows.

Each row of this table indicates that some viewer viewed an article (written by some author) on some date.

Note that equal author_id and viewer_id indicate the same person.

Write a solution to find all the authors that viewed at least one of their own articles.

Return the result table sorted by id in ascending order.

The result format is in the following example.

```
</>Code

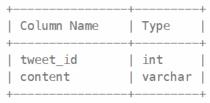
MySQL ∨ Auto

1 # Write your MySQL query statement below
2 Select distinct author_id as id
3 from views
4 where author_id=viewer_id
5 order by author_id;
```

Input			
Views =			
article_id	author_id	viewer_id	view_date
1	3	5	
1	3	6	2019-08-02
2	7	7	2019-08-01
2	7	6	2019-08-02
4	7	1	2019-07-22
3	4	4	2019-07-21
		₩ 1	View more
Output			
id 4 7			

SQL Schema > Pandas Schema >

Table: Tweets



tweet_id is the primary key (column with unique values) for this table. content consists of characters on an American Keyboard, and no other special characters.

This table contains all the tweets in a social media app.

Write a solution to find the IDs of the invalid tweets. The tweet is invalid if the number of characters used in the content of the tweet is **strictly greater** than 15.

Return the result table in any order.

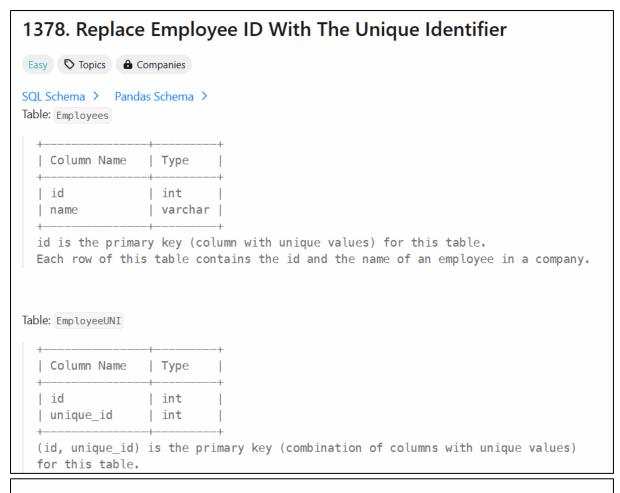
The result format is in the following example.

```
c/> Code

MySQL > Auto

1  # Write your MySQL query statement below
2  select tweet_id from tweets where length(content) >15;

Testcase > Test Result
Input
```



Write a solution to show the **unique ID** of each user, If a user does not have a unique ID replace just show null.

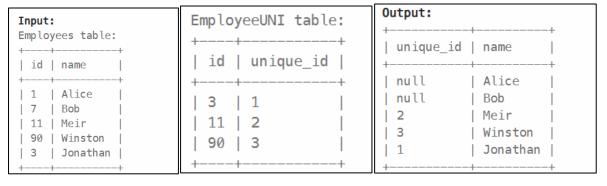
Return the result table in **any** order.

The result format is in the following example.

```
// Code

MySQL ✓ ♠ Auto

1  # Write your MySQL query statement below
2  | select unique_id , name from employees as ese
3  | left join
4  | EmployeeUni as euni
5  | on ese.id = euni.id;
```







SQL Schema > Pandas Schema >

Table: Cinema

Column Name	Type
id	int
movie	varchar
description	varchar
rating	float

id is the primary key (column with unique values) for this table. Each row contains information about the name of a movie, its genre, and its rating.

rating is a 2 decimal places float in the range [0, 10]

Write a solution to report the movies with an odd-numbered ID and a description that is not "boring".

Return the result table ordered by $\[\]$ rating $\[\]$ in descending order.

The result format is in the following example.

Example 1:

Input:

Cinema table:

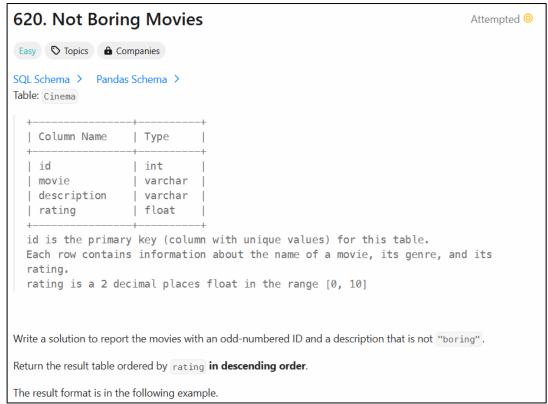
id	movie	description	 rating
1 2 3 4 5	War Science irish Ice song House card	great 3D fiction boring Fantacy Interesting	8.9 8.5 6.2 8.6

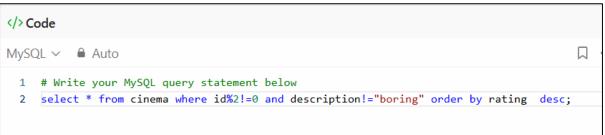
Output:

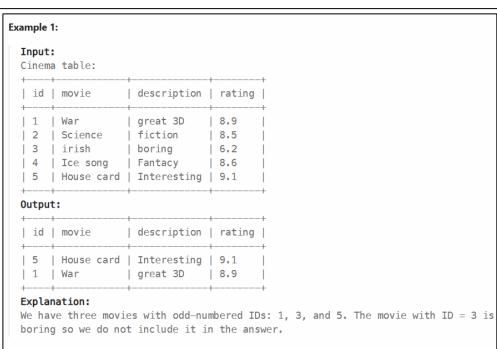
id movie	description	++ rating ++
5 House card 1 War	Interesting great 3D	9.1

Explanation:

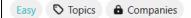
We have three movies with odd-numbered IDs: 1, 3, and 5. The movie with ID = 3 is boring so we do not include it in the answer.







1251. Average Selling Price



SQL Schema > Pandas Schema >

Table: Prices

Column Name	Type
product_id start_date end_date price	int

(product_id, start_date, end_date) is the primary key (combination of columns with unique values) for this table.

Each row of this table indicates the price of the product_id in the period from start_date to end_date.

For each product_id there will be no two overlapping periods. That means there will be no two intersecting periods for the same product_id.

Table: UnitsSold

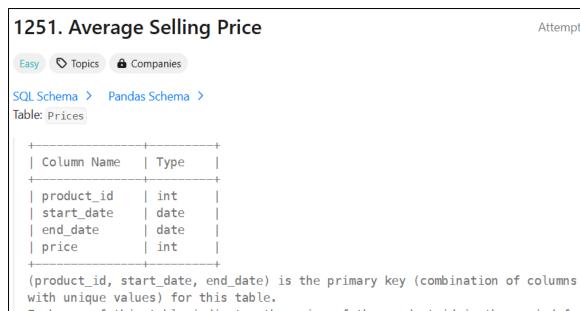
Column Name	++ Type +
product_id purchase_date units	int

This table may contain duplicate rows.

Each row of this table indicates the date, units, and product_id of each product sold.

Write a solution to find the average selling price for each product. average_price should be **rounded to 2 decimal places**. If a product does not have any sold units, its average selling price is assumed to be 0.

Return the result table in any order.



Each row of this table indicates the price of the product id in the period from start date to end date.

Attempted @

For each product id there will be no two overlapping periods. That means there will be no two intersecting periods for the same product_id.

