Sample sql project with dataset link and question and answers with respect to education platform ? It should have 20 questions(10 easy , 5 medium and 5 hard) .

Sample sql project with dataset link and question and answers with respect to education platform ? It should have 20 questions(10 easy , 5 medium and 5 hard) .

Here’s a sample SQL project idea tailored for an education platform, complete with a dataset link and 20 questions (10 easy, 5 medium, and 5 hard):

**Project Overview**

**Title:** Student Performance Analysis **Objective:** Analyze student performance data to derive insights about academic trends, strengths, and areas for improvement. **Dataset:** You can use the Student Performance Dataset from Kaggle. It contains information about students' demographics, academic performance, and other related factors.

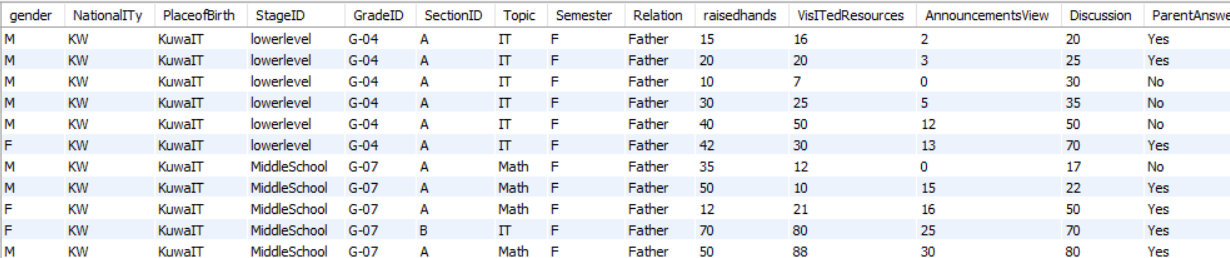
Q1 : Retrieve all student records from the dataset.

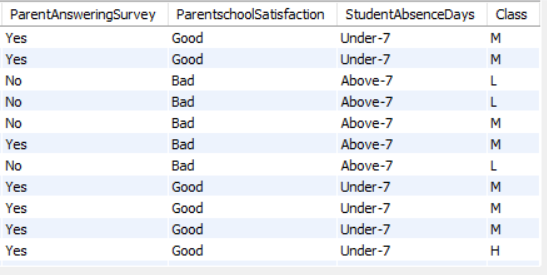
A1 :

Code :

SELECT \* FROM Testfile;

Output :





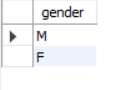
Q2 : List all distinct values in the gender column.

A2 :

Code :

SELECT DISTINCT gender FROM Testfile;

Output :



Q3 :

Count the total number of rows in the table.

A3 :

Code :

SELECT COUNT(\*) AS total\_rows FROM Testfile;

Output :



Q4 :

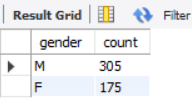
Count the number of students for each gender.

A4 :

Code :

SELECT gender, COUNT(\*) AS count FROM Testfile GROUP BY gender;

Output:



Q5 :

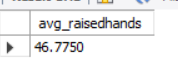
Calculate the average number of raisedhands.

A5 :

Code :

SELECT AVG(raisedhands) AS avg\_raisedhands FROM Testfile;

Output :



Q6 :

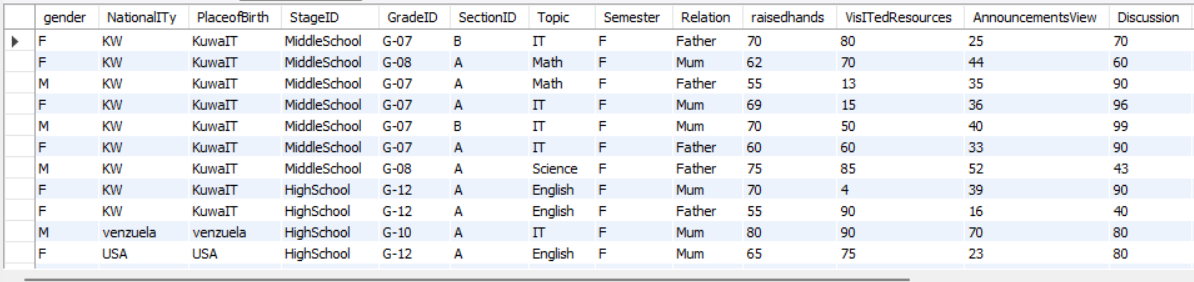
Retrieve all rows where raisedhands is greater than 50.

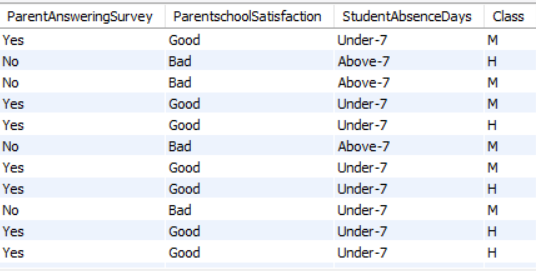
A6 :

Code :

SELECT \* FROM Testfile WHERE raisedhands > 50;

Output :





Q7 :

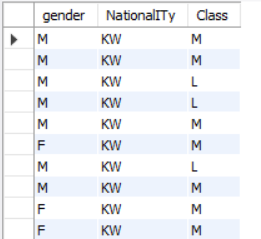
Show only gender, NationalITy, and Class.

A7 :

Code :

SELECT gender, NationalITy, Class FROM Testfile;

Output :



Q8 :

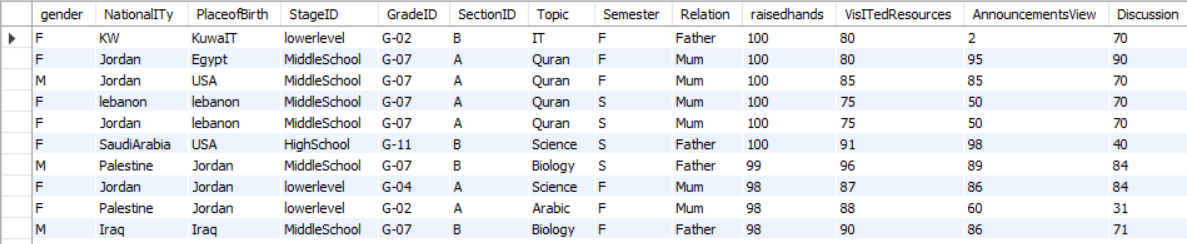
List all rows ordered by raisedhands in descending order.

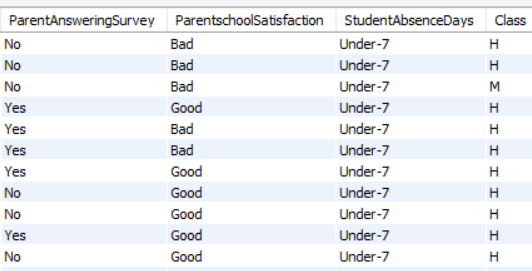
A8 :

Code :

SELECT \* FROM Testfile ORDER BY raisedhands DESC;

Output :





Q9 :

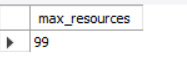
Find the maximum value in the VisITedResources column.

A9 :

Code :

SELECT MAX(VisITedResources) AS max\_resources FROM Testfile;

Output :



Q10 :

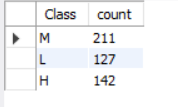
Count how many students are in each Class.

A10 :

Code :

SELECT Class, COUNT(\*) AS count FROM Testfile GROUP BY Class;

Output :



Q11 :

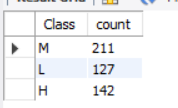
Find classes that have more than 20 students.

A11 :

Code :

SELECT Class, COUNT(\*) AS count FROM Testfile GROUP BY Class HAVING count > 20;

Output :



Q12 :

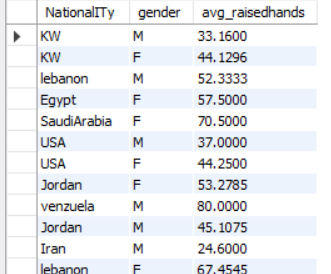
Compute the average raisedhands for each combination of NationalITy and gender.

A12 :

Code :

SELECT NationalITy, gender, AVG(raisedhands) AS avg\_raisedhands FROM Testfile GROUP BY NationalITy, gender;

Output :



Q13 :

Retrieve rows where AnnouncementsView is greater than 5 and Discussion is less than 10.

A13 :

Code :

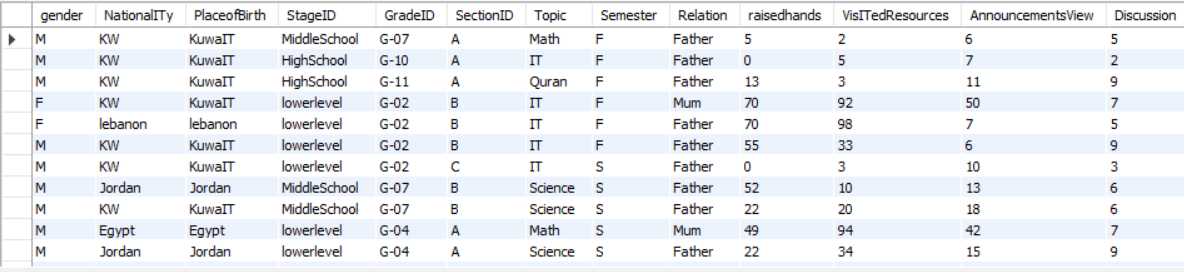
SELECT \*

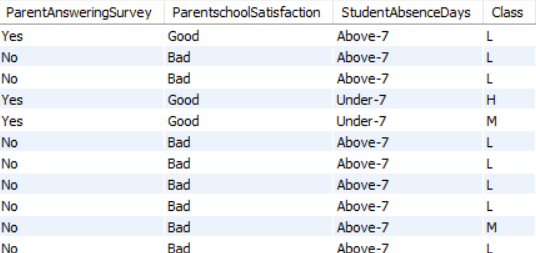
FROM Testfile

WHERE AnnouncementsView > 5

AND Discussion < 10;

Output :

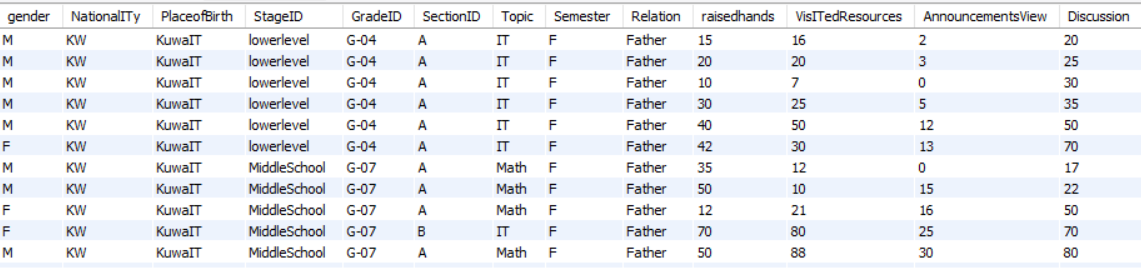


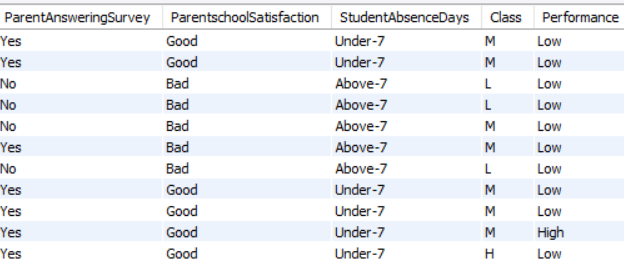


Q14 :

Add a column called Performance that shows 'High' if raisedhands is above 50 and 'Low' otherwise.

Code :





Q15 :

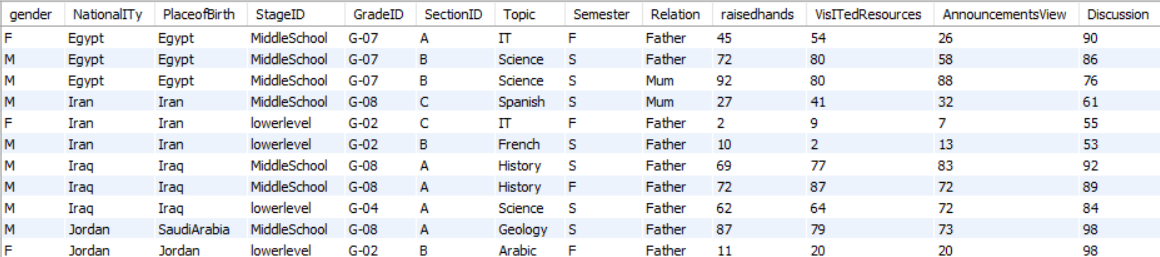
For each NationalITy, list the top 3 rows with the highest Discussion scores (requires MySQL 8.0+ for window functions).

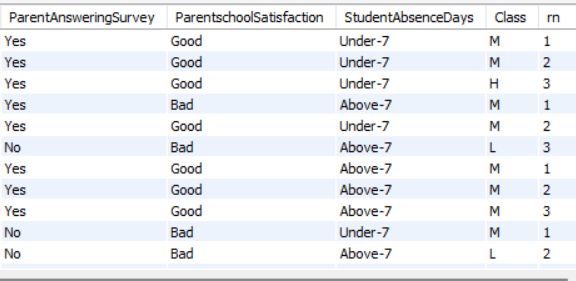
A15 :

Code :

SELECT \* FROM ( SELECT \*, ROW\_NUMBER() OVER (PARTITION BY NationalITy ORDER BY Discussion DESC) AS rn FROM Testfile ) AS sub WHERE rn <= 3;

Output :





Q16 :

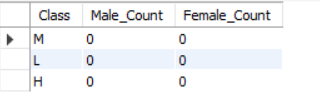
Create a pivot-like table that shows the number of students by gender for each Class.

A16 :

Code :

SELECT Class, SUM(CASE WHEN gender = 'Male' THEN 1 ELSE 0 END) AS Male\_Count, SUM(CASE WHEN gender = 'Female' THEN 1 ELSE 0 END) AS Female\_Count FROM Testfile GROUP BY Class;

Output :



Q17 :

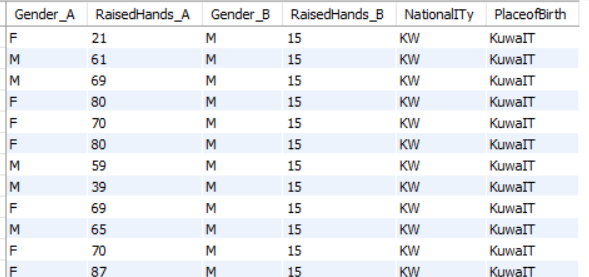
Compare students within the same NationalITy and PlaceofBirth by joining the table to itself; list pairs where one student has a higher raisedhands value than the other.

A :

Code :

SELECT a.gender AS Gender\_A, a.raisedhands AS RaisedHands\_A, b.gender AS Gender\_B, b.raisedhands AS RaisedHands\_B, a.NationalITy, a.PlaceofBirth FROM Testfile a JOIN Testfile b ON a.NationalITy = b.NationalITy AND a.PlaceofBirth = b.PlaceofBirth WHERE a.raisedhands > b.raisedhands;

Output :



Q18 :

For each row, calculate the ratio of raisedhands to VisITedResources, while avoiding division by zero.

A :

Code :

SELECT \*,

CASE

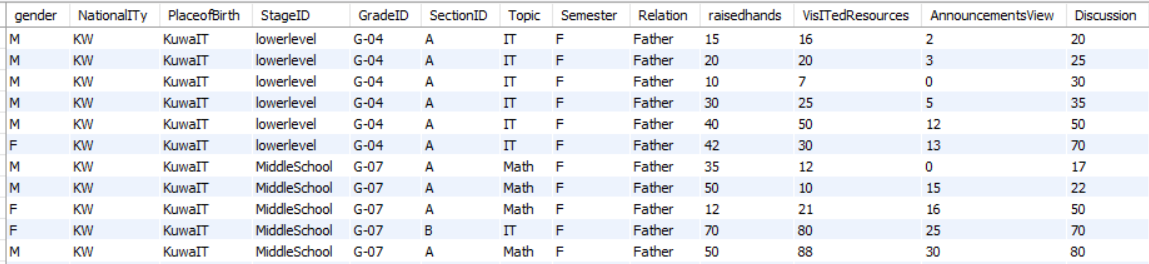
WHEN VisITedResources = 0 THEN NULL

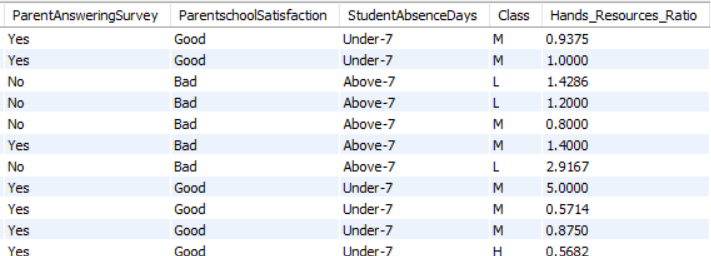
ELSE raisedhands / VisITedResources

END AS Hands\_Resources\_Ratio

FROM Testfile;

Output :





Q19 :

Rank students by Discussion scores within each NationalITy using window functions (MySQL 8.0+ required).

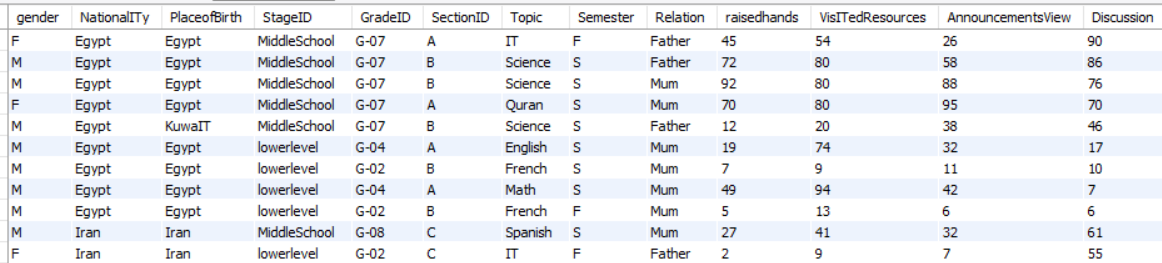
A19 :

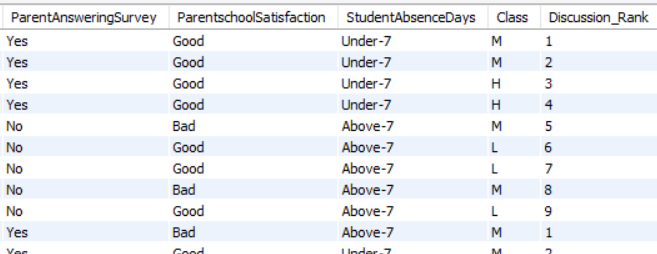
SELECT \*,

RANK() OVER (PARTITION BY NationalITy ORDER BY Discussion DESC) AS Discussion\_Rank

FROM Testfile;

Output :





Q20 :

Calculate a weighted score for each student using the formula  
**(raisedhands \* 0.4 + AnnouncementsView \* 0.3 + Discussion \* 0.3)**  
and rank students by this score.

A20 :

Code :

SELECT \*, (raisedhands \* 0.4 + AnnouncementsView \* 0.3 + Discussion \* 0.3) AS WeightedScore, RANK() OVER (ORDER BY (raisedhands \* 0.4 + AnnouncementsView \* 0.3 + Discussion \* 0.3) DESC) AS ScoreRank FROM Testfile;

Output :

