**eg:Query the information of students.**

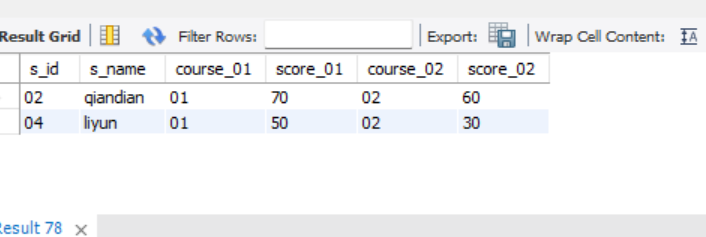
select \* from student;

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

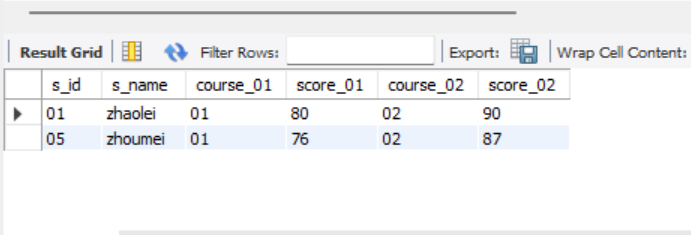
**Q1: Query the information and course scores of students whose grades in the "01" course are higher than those in the "02" course**

|  |
| --- |
| SELECT student.s\_id, student.s\_name, score.c\_id AS course\_01, score.s\_score AS score\_01, s2.c\_id AS course\_02, s2.s\_score AS score\_02  FROM student  JOIN score ON student.s\_id = score.s\_id AND score.c\_id = '01'  JOIN score s2 ON student.s\_id = s2.s\_id AND s2.c\_id = '02'  WHERE score.s\_score > s2.s\_score; |



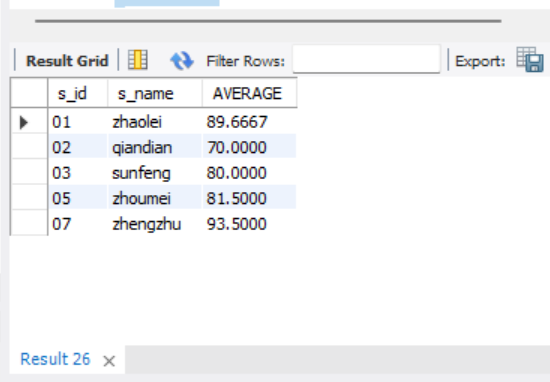
**Q2:** **Query the information and course scores of students whose grades in the "01" course are lower than those in the "02" course**

|  |
| --- |
| SELECT student.s\_id, student.s\_name, score.c\_id AS course\_01, score.s\_score AS score\_01, s2.c\_id AS course\_02, s2.s\_score AS score\_02  FROM student  JOIN score ON student.s\_id = score.s\_id AND score.c\_id = '01'  JOIN score s2 ON student.s\_id = s2.s\_id AND s2.c\_id = '02'  WHERE score.s\_score < s2.s\_score; |



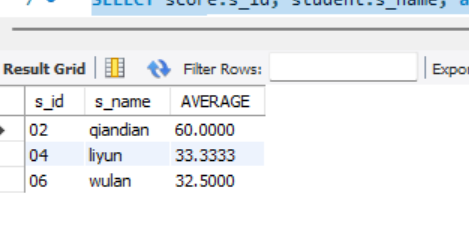
**Q3:** **Query the student number, student name and average score of students whose average score is greater than or equal to 60 points**

|  |
| --- |
| SELECT score.s\_id, student.s\_name, avg(score.s\_score) AS AVERAGE  from score  join student on student.s\_id = score.s\_id  WHERE score.s\_score >= 60  GROUP BY s\_id |



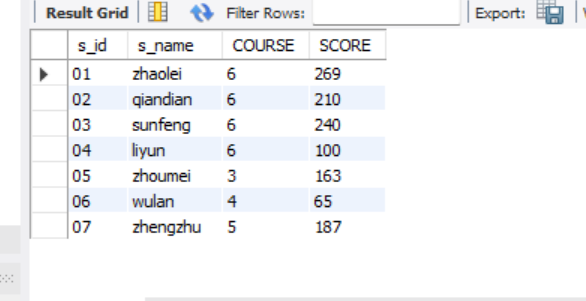
**Q4: Query the student number, name and average score of students whose average score is less than 60 points (including those with and without scores)**

|  |
| --- |
| SELECT score.s\_id, student.s\_name, avg(score.s\_score) AS AVERAGE  from score  join student on student.s\_id = score.s\_id  WHERE score.s\_score <= 60  GROUP BY s\_id |



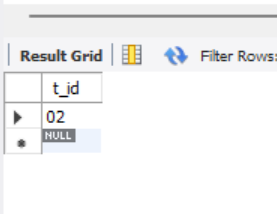
**Q5: Query the student number, student name, total number of courses selected, and total grades of all courses (including those without selected courses) of all students**

|  |
| --- |
| SELECT student.s\_id, student.s\_name, sum(score.c\_id) AS COURSE, sum(score.s\_score) AS SCORE  from score  JOIN student ON student.s\_id = score.s\_id  JOIN course ON course.c\_id = score.c\_id  GROUP BY s\_id |



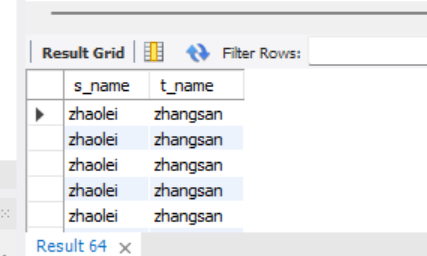
**Q6: Query the number of teachers with the surname "Li"**

|  |
| --- |
| SELECT t\_id from teacher  where t\_name like 'li%' |



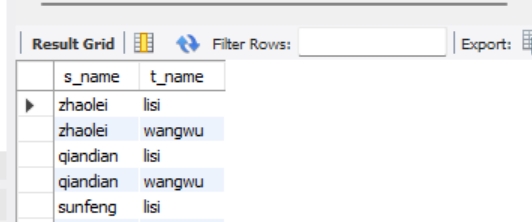
**Q7: Query the information of students who have studied with teacher "zhangsan"**

|  |
| --- |
| **SELECT student.s\_name, teacher.t\_name**  **FROM student**  **JOIN score ON score.s\_id = score.s\_id**  **JOIN course ON score.c\_id = course.c\_id**  **JOIN teacher ON course.t\_id = teacher.t\_id**  **WHERE teacher.t\_name = 'zhangsan';** |



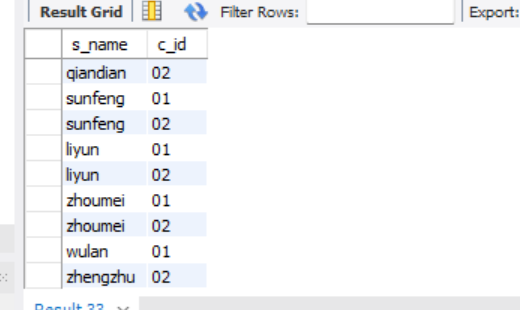
**Q8: Query the information of students who have not studied under teacher "zhangsan"**

|  |
| --- |
| SELECT student.s\_name, teacher.t\_name  FROM student  JOIN score ON student.s\_id = score.s\_id  JOIN course ON score.c\_id = course.c\_id  JOIN teacher ON course.t\_id = teacher.t\_id  WHERE teacher.t\_name <> 'zhangsan'; |



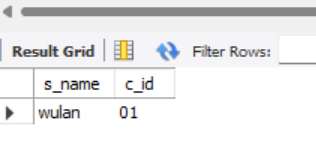
**Q9: Query the information of students who have studied the course numbered "01" and also studied the course numbered "02"**

|  |
| --- |
| SELECT student.s\_name, course.c\_id  from score  JOIN student on student.s\_id = score.s\_id  JOIN course on course.c\_id = score.c\_id  where course.c\_id IN (01, 02) |



**Q10: Query the information of students who have studied the course numbered "01" but not the course numbered "02"**

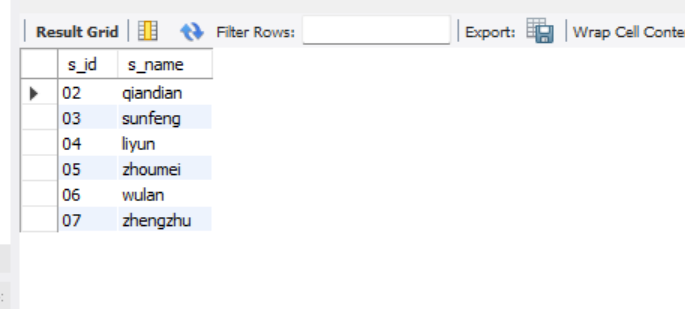
|  |
| --- |
| SELECT student.s\_name, course.c\_id  FROM student  JOIN score ON student.s\_id = score.s\_id  JOIN course ON course.c\_id = score.c\_id  LEFT JOIN score AS score\_2 ON student.s\_id = score\_2.s\_id AND score\_2.c\_id = '02'  WHERE course.c\_id = '01' AND score\_2.s\_id IS NULL; |



**Q11: Query information about students who have not completed all courses**

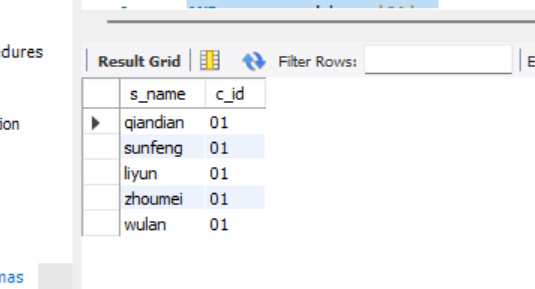
**Q12: Query the information of students who have at least one course that is the same as the student with student number "01"**

|  |
| --- |
| SELECT DISTINCT student.s\_id, student.s\_name  FROM student  JOIN score ON score.s\_id = student.s\_id  WHERE score.c\_id IN (SELECT s2.c\_id  FROM score s2  WHERE s2.s\_id = '01')  AND student.s\_id <> '01'; |



**Q13: Query the information of other students who are studying the same course as the student numbered "01"**

|  |
| --- |
| SELECT student.s\_name, course.c\_id  FROM score  JOIN student ON student.s\_id = score.s\_id  JOIN course ON course.c\_id = score.c\_id  WHERE score.c\_id = '01'  AND score.s\_id <> '01'; |

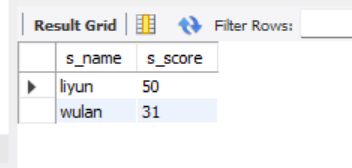


**Q14: Query the names of students who have not studied any course taught by "zhangsan"**

**Q15: Query the student ID number, name and average grade of students who failed two or more courses (Even if you did not take the exam, your score will be recorded as zero, so it should be considered a failure.)**

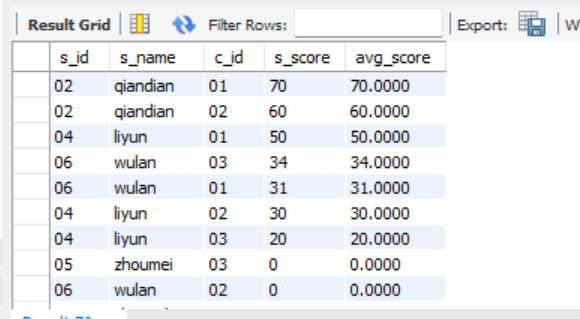
**Q16: Query the student information of "01" course whose score is less than 60 and sorted by score in descending order**

|  |
| --- |
| SELECT student.s\_name, score.s\_score  FROM score  JOIN student ON student.s\_id = score.s\_id  JOIN course ON course.c\_id = score.c\_id  WHERE course.c\_id = '01' AND score.s\_score < 60  ORDER BY score.s\_score DESC; |



**Q17: Displays the grades for all students in all courses and the average grade by average grade from highest to lowest (Students who did not take the exam will have a score of 0)**

|  |
| --- |
| **SELECT student.s\_id, student.s\_name, course.c\_id, IFNULL(score.s\_score, 0) AS s\_score,**  **IFNULL(AVG(score.s\_score), 0) AS avg\_score**  **FROM student**  **CROSS JOIN course**  **LEFT JOIN score ON student.s\_id = score.s\_id AND course.c\_id = score.c\_id**  **GROUP BY student.s\_id, student.s\_name, course.c\_id**  **ORDER BY avg\_score DESC;** |

****

**Q18: Query the highest score, lowest score and average score of each subject: displayed in the following form: course ID, course name, highest score, lowest score, average score, passing rate, medium rate, good rate, excellent rate – passing is >=60 , medium is: 70-80, good is: 80-90, excellent is: >=90**

**Q19: Query the student information, Sort by the scores of each subject and display the ranking**

**Q20: Query students’ total scores and rank them**