**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**

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

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

|  |
| --- |
| **select student.s\_id, student.s\_name, AVG(score.s\_score)**  **from student**  **JOIN score on student.s\_id = score.s\_id**  **group by student.s\_id, student.s\_name**  **having AVG(score.s\_score) >= 60;** |

A screenshot of a computer

Description automatically generated

**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  student.s\_id, student.s\_name, AVG(score.s\_score)  FROM  student  JOIN  score ON student.s\_id = score.s\_id  GROUP BY student.s\_id , student.s\_name  HAVING AVG(score.s\_score) < 60; |

A screenshot of a computer

Description automatically generated

**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,  COUNT(course.c\_id),  SUM(score.s\_score)  FROM  student  JOIN  score ON score.s\_id = student.s\_id  JOIN  course ON course.c\_id = score.c\_id  GROUP BY student.s\_id , student.s\_name; |

A screenshot of a computer

Description automatically generated

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

|  |
| --- |
| SELECT  COUNT(t\_id)  FROM  teacher  WHERE  t\_name LIKE '%li%'; |

A screenshot of a computer

Description automatically generated

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

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

A screenshot of a computer

Description automatically generated

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

|  |
| --- |
| select student.s\_id, student.s\_name, student.s\_brith, student.s\_sex  from student  INNER JOIN score on score.s\_id = student.s\_id  INNER JOIN course on course.c\_id = score.c\_id  JOIN teacher on course.t\_id = teacher.t\_id  where teacher.t\_id NOT IN (select teacher.t\_id  from student  INNER JOIN score on score.s\_id = student.s\_id  INNER JOIN course on course.c\_id = score.c\_id  JOIN teacher on course.t\_id = teacher.t\_id  where teacher.t\_id = '01') |

A screenshot of a computer

Description automatically generated

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

|  |
| --- |
| select \* from student  INNER JOIN score on score.s\_id = student.s\_id  INNER JOIN course on course.c\_id = score.c\_id  where course.c\_id = '01' OR course.c\_id = '02'; |

A screenshot of a computer

Description automatically generated

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

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

|  |
| --- |
| select student.s\_id, student.s\_name, count(course.c\_id)  from student  JOIN score on score.s\_id = student.s\_id  JOIN course on course.c\_id = score.c\_id  group by student.s\_id, student.s\_name  having count(course.c\_id) < 3 |

A screenshot of a computer

Description automatically generated

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

|  |
| --- |
| select student.s\_id, student.s\_name, course.c\_id  from student  JOIN score on score.s\_id = student.s\_id  JOIN course on course.c\_id = score.c\_id  where student.s\_id != '01' AND course.c\_id IN (  select course.c\_id  from student  JOIN score on score.s\_id = student.s\_id  JOIN course on course.c\_id = score.c\_id  where student.s\_id = '01') |

A screenshot of a computer

Description automatically generated

**Q13: Query the information of other students who are studying the same course as the student numbered "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 \***  **from student**  **JOIN score on score.s\_id = student.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** |

A screenshot of a computer

Description automatically generated

**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)**

**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**

|  |
| --- |
| **select student.s\_id, student.s\_name, student.s\_brith, student.s\_sex, course.c\_id, score.s\_score**  **from student**  **JOIN score on score.s\_id = student.s\_id**  **JOIN course on course.c\_id = score.c\_id**  **order by course.c\_id asc, score.s\_score desc** |

A screenshot of a computer

Description automatically generated

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