ASSIGNMENT 5

Write SQL query for these questions-

1) Retrieve the names and email addresses of all students

```
SELECT name,email from student
```

	name varchar(255)	email varchar(255)
1	John Doe	john.doe@example.com
2	Jane Smith	jane.smith@example.com
3	Robert Johnson	robert.j@example.com
4	Emily White	emily.white@example.com
5	Michael Lee	michael.lee@example.com
6	Sarah Brown	sarah.brown@example.com
7	David Clark	david.clark@example.com
8	Melissa Turner	melissa.turner@example.co

2) Find the courses that have more than three credits.

```
SELECT coursename from course where credits>3
```

Search results

	coursename varchar(255)
1	History
2	Chemistry
3	Physics
4	Biology

3) List the exams scheduled after November 15, 2023.

```
SELECT examdate from exam where examdate>"2023-11-15"
```

exam

Search results

	examid int
1	204
2	205

4) Get the faculty members who work in the "Mathematics" department.

```
SELECT name from faculty where department="Mathematics"
```

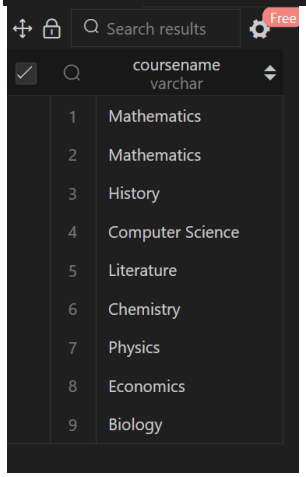
faculty

Search results

	name varchar(25!
1	Dr. Smith

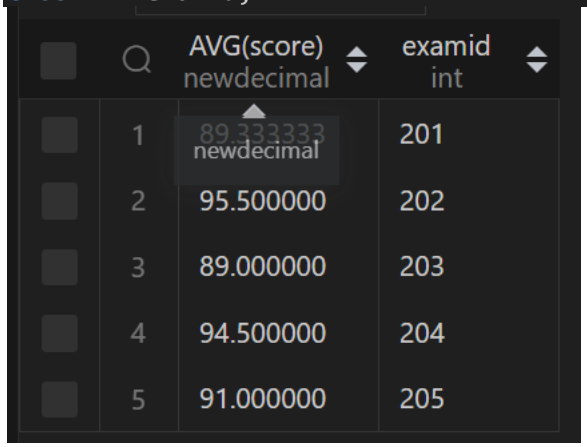
5) Retrieve the courses that each student is enrolled in.

```
SELECT coursename
FROM enrollment
JOIN course ON enrollment.course_id=course.course_id;
```



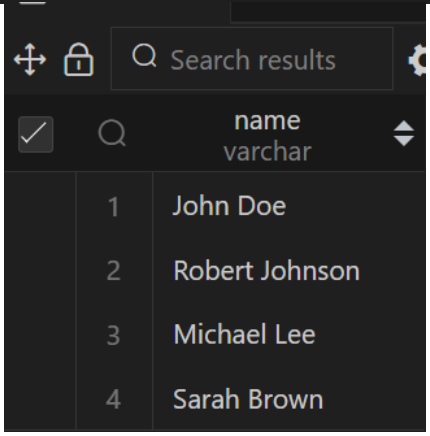
6) Find the average score for each exam.

```
SELECT AVG(score),examid
FROM examresult
GROUP BY examid;
```



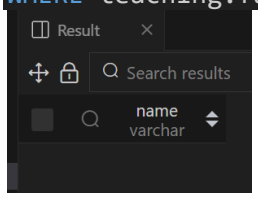
7) List the students who scored above 90 on any exam.

```
SELECT name
FROM student
JOIN examresult ON examresult.student_id=student.student_id where score>90;
```



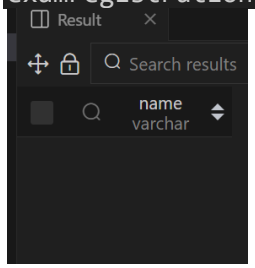
8) Retrieve the faculty members who teach multiple courses.

```
SELECT name FROM faculty WHERE (SELECT count(teaching.course_id) FROM teaching
WHERE teaching.facultyid = faculty.facultyid) > 1
```



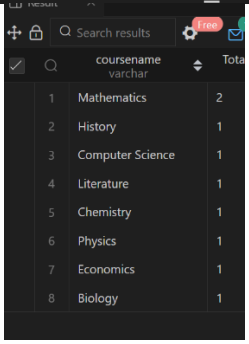
9) Find the students who have not registered for any exams.

```
SELECT name FROM student WHERE (SELECT count(examregistration.examid) FROM
examregistration WHERE examregistration.student_id =student.student_id) = 0
```



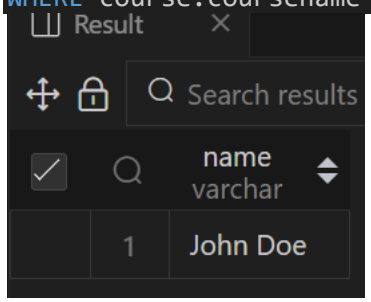
10) Retrieve the total number of enrollments for each course.

```
SELECT Course.coursename, COUNT(enrollment.course_id) AS TotalEnrollments
FROM course
LEFT JOIN enrollment ON course.course_id = enrollment.course_id GROUP BY
course.course_id;
```



11) Find the students who are enrolled in the "History" course.

```
SELECT student.name
FROM student
JOIN enrollment ON student.student_id = enrollment.student_id
JOIN course ON enrollment.course_id = course.course_id
WHERE course.coursename = 'History';
```



12) Retrieve the exams and their locations scheduled for November 2023.

```
SELECT examid, examdate, examlocation
FROM exam
WHERE YEAR(ExamDate) = 2023 AND MONTH(ExamDate) = 11;
```

exam

Search results

	examid int	examdate date	examlocation varchar(255)
1	201	2023-11-10	Exam Hall A
2	202	2023-11-12	Exam Hall B
3	203	2023-11-15	Exam Hall c
4	204	2023-11-18	Exam Hall D
5	205	2023-11-20	Exam Hall E

13) List the courses with the highest number of enrollments

```
SELECT c.coursename, MAX(e.Enrollments) AS MaxEnrollments
FROM course c
LEFT JOIN (
SELECT course_id, COUNT(student_id) AS Enrollments
FROM enrollment
GROUP BY course_id
) e ON c.course_id = e.course_id
GROUP BY c.coursename;
```

Result

Search results

	coursename varchar	MaxEnrollments bigint
1	Mathematics	2
2	History	1
3	Computer Science	1
4	Literature	1
5	Chemistry	1
6	Physics	1
7	Economics	1
8	Biology	1

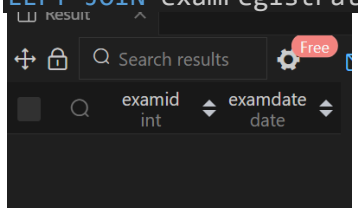
14) Find the average score for each student.

```
SELECT student.name, AVG(score) AS averagescore
FROM student
JOIN examresult ON student.student_id = examresult.student_id
GROUP BY student.student_id;
```

	name varchar	averagescore newdecimal
1	John Doe	92.500000
2	Jane Smith	88.000000
3	Robert Johnson	95.500000
4	Emily White	89.000000
5	Michael Lee	94.500000
6	Sarah Brown	91.000000
7	David Clark	87.500000

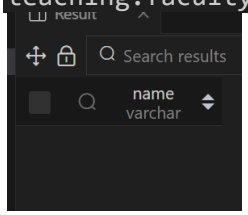
15) Retrieve the exams that have no registered students.

```
SELECT e.examid, e.examdate
FROM exam e
LEFT JOIN examregistration er ON e.examid = er.examid WHERE er.examid IS NULL;
```



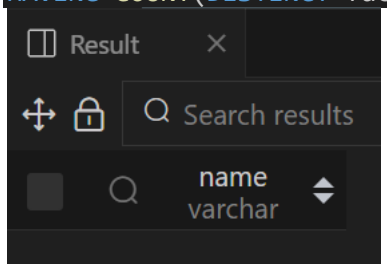
16) List the faculty members who have yet to teach any courses.

```
SELECT faculty.name
FROM faculty
LEFT JOIN teaching ON faculty.facultyid = teaching.facultyid WHERE
teaching.facultyid IS NULL;
```



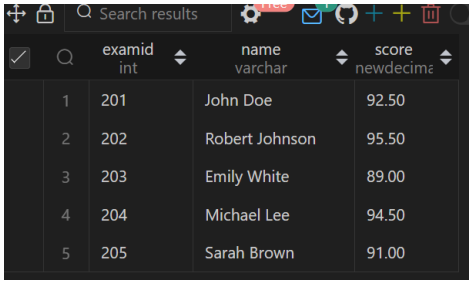
17) Find the students who have registered for exams in both "Mathematics" and "Computer Science" departments.

```
SELECT student.name
FROM student
JOIN enrollment ON student.student_id = enrollment.student_id JOIN course ON
enrollment.course_id = course.course_id
JOIN teaching ON course.course_id = teaching.course_id
JOIN faculty ON teaching.facultyid = faculty.facultyid
WHERE faculty.department IN ('Mathematics', 'Computer Science')
GROUP BY student.student_id
HAVING COUNT(DISTINCT faculty.department) = 2;
```



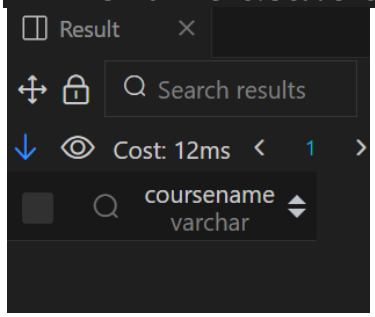
18) Retrieve the students who scored the highest in each exam.

```
SELECT examresult.examid, student.name, examresult.score
FROM examresult
JOIN student ON examresult.student_id = student.student_id
WHERE (examresult.examid, examresult.score) IN (
SELECT examid, MAX(score)
FROM examresult
GROUP BY examid);
```



19) Find the courses that no student has enrolled in.

```
SELECT coursename FROM course
LEFT JOIN enrollment ON course.course_id = enrollment.course_id
WHERE enrollment.student_id IS NULL;
```



20) Retrieve the faculty members who teach courses with an average enrollment count above 10.

```
SELECT faculty.name, AVG(enrollments) AS AverageEnrollments
FROM faculty
JOIN teaching ON faculty.facultyid = teaching.facultyid
JOIN (SELECT course.course_id, COUNT(enrollment.student_id) AS Enrollments
FROM course
LEFT JOIN enrollment ON course.course_id = enrollment.course_id
GROUP BY course.course_id) AS courseenrollments ON teaching.course_id =
courseenrollments.course_id
GROUP BY faculty.name
HAVING AVG(enrollments) > 10;
```