Name: Md Mainul Hossain Fahim Chowdhury

Id: 0112230085

Section:A

1.. List All Students and Their Departments o Query the student table to list all students with their corresponding department names

Ans:  select ID,name,dept_name

FROM student

| | | | | ID | name | dept_name |
|---|---|---|---|---|---|---|
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 00128 | Zhang | Comp. Sci. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 12345 | Shankar | Comp. Sci. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 19991 | Brandt | History |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 23121 | Chavez | Finance |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 44553 | Peltier | Physics |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 45678 | Levy | Physics |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 54321 | Williams | Comp. Sci. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 55739 | Sanchez | Music |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 70557 | Snow | Physics |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 76543 | Brown | Comp. Sci. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 76653 | Aoi | Elec. Eng. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 98765 | Bourikas | Elec. Eng. |
| ☐ | 🖊 Edit | ⬚ Copy | ⊖ Delete | 98988 | Tanaka | Biology |

2. Retrieve all courses and the names of the instructors teaching them.

Ans: select c.course_id,i.name

from course AS c

inner join teaches AS t

ON c.course_id=t.course_id

INNER JOIN instructor AS i

ON  t.id=i.id

| course_id | name |
|-----------|------|
| CS-101 | Srinivasan |
| CS-315 | Srinivasan |
| CS-347 | Srinivasan |
| FIN-201 | Wu |
| MU-199 | Mozart |
| PHY-101 | Einstein |
| HIS-351 | El Said |
| CS-101 | Katz |
| CS-319 | Katz |
| BIO-101 | Crick |
| BIO-301 | Crick |
| CS-190 | Brandt |
| CS-190 | Brandt |
| CS-319 | Brandt |
| EE-181 | Kim |

3. Find all sections for the course with course code 'CS-101' and show the term, year, building, and room number.

Ans: select *

from section

where course_id='CS-101'

| | | | course_id | sec_id | semester | year | building | room_number | time_slot_id |
|---|---|---|-----------|--------|----------|------|----------|-------------|--------------|
| ☐ | Edit | Copy | Delete | CS-101 | 1 | Fall | 2017 | Packard | 101 | H |
| ☐ | Edit | Copy | Delete | CS-101 | 1 | Spring | 2018 | Packard | 101 | F |

4. Display each instructor's ID, name, and the course section they are teaching.

Ans: SELECT i.ID,i.name,s.sec_id

from instructor AS i

INNER JOIN teaches AS t

ON i.ID=t.ID

inner join section AS s

ON t.course_id=s.course_id

| ID | name | sec_id |
|---|---|---|
| 10101 | Srinivasan | 1 |
| 10101 | Srinivasan | 1 |
| 10101 | Srinivasan | 1 |
| 10101 | Srinivasan | 1 |
| 12121 | Wu | 1 |
| 15151 | Mozart | 1 |
| 22222 | Einstein | 1 |
| 32343 | El Said | 1 |
| 45565 | Katz | 1 |
| 45565 | Katz | 1 |
| 45565 | Katz | 1 |
| 45565 | Katz | 2 |
| 76766 | Crick | 1 |
| 76766 | Crick | 1 |
| 83821 | Brandt | 1 |
| 83821 | Brandt | 2 |
| 83821 | Brandt | 1 |
| 83821 | Brandt | 2 |
| 83821 | Brandt | 1 |
| 83821 | Brandt | 2 |
| 98345 | Kim | 1 |

5. List all students enrolled in the course 'BIO-101', including their IDs and names.

Ans: SELECT c.course_id,s.ID,s.name

from course AS c

Inner JOIN takes AS t

ON t.course_id=c.course_id

INNER JOIN student AS s

ON s.ID=t.ID

WHERE c.course_id='BIO-101'

| course_id | ID | name |
|-----------|-------|--------|
| BIO-101 | 98988 | Tanaka |

6. Retrieve all students with their advisor's ID and name

Ans : SELECT

   s.ID AS student_id,

   s.name AS student_name,

   a.i_ID AS advisor_id,

   i.name AS advisor_name

FROM

   student AS s

JOIN

   advisor AS a ON s.ID = a.s_ID

JOIN

   instructor AS i ON a.i_ID = i.ID;

| student_id | student_name | advisor_id | advisor_name |
|---|---|---|---|
| 12345 | Shankar | 10101 | Srinivasan |
| 44553 | Peltier | 22222 | Einstein |
| 45678 | Levy | 22222 | Einstein |
| 00128 | Zhang | 45565 | Katz |
| 76543 | Brown | 45565 | Katz |
| 23121 | Chavez | 76543 | Singh |
| 98988 | Tanaka | 76766 | Crick |
| 76653 | Aoi | 98345 | Kim |
| 98765 | Bourikas | 98345 | Kim |

7. List all courses that have prerequisites, including the course code, course name, and the prerequisite course code.

Ans: SELECT

   c.course_id AS course_code,

   c.title AS course_name,

   p.prereq_id AS prerequisite_course_code

FROM

   course AS c

JOIN

   prereq AS p ON c.course_id = p.course_id;

| course_code | course_name | prerequisite_course_code |
|---|---|---|
| BIO-301 | Genetics | BIO-101 |
| BIO-399 | Computational Biology | BIO-101 |
| CS-190 | Game Design | CS-101 |
| CS-315 | Robotics | CS-101 |
| CS-319 | Image Processing | CS-101 |
| CS-347 | Database System Concepts | CS-101 |
| EE-181 | Intro. to Digital Systems | PHY-101 |

8. Show each department with the courses it offers, including department name and course title.

Ans: SELECT

   d.dept_name AS department_name,

   c.title AS course_title

FROM

   department AS d

JOIN

   course AS c ON d.dept_name = c.dept_name

ORDER BY

   d.dept_name, c.title;

| department_name | course_title |
| --- | --- |
| Biology | Computational Biology |
| Biology | Genetics |
| Biology | Intro. to Biology |
| Comp. Sci. | Database System Concepts |
| Comp. Sci. | Game Design |
| Comp. Sci. | Image Processing |
| Comp. Sci. | Intro. to Computer Science |
| Comp. Sci. | Robotics |
| Elec. Eng. | Intro. to Digital Systems |
| Finance | Investment Banking |
| History | World History |
| Music | Music Video Production |
| Physics | Physical Principles |

9. Find all courses taught by the instructor with ID '83821', including course codes and titles.

Ans:  SELECT c.course_id,i.ID

from course AS c

INNER JOIN department AS d

ON d.dept_name=c.dept_name

INNER JOIN instructor AS i

ON i.dept_name=d.dept_name

WHERE i.ID='83821'

| course_id | ID |
|-----------|-------|
| CS-101 | 83821 |
| CS-190 | 83821 |
| CS-315 | 83821 |
| CS-319 | 83821 |
| CS-347 | 83821 |

10. List students and their grades for the course 'CS-319'.

Ans: SELECT  s.ID,t.grade,c.course_id

from student AS s

INNER join takes as t

on t.ID=s.ID

INNER JOIN course as c

on c.course_id=t.course_id

where c.course_id='CS-319'

| ID | grade | course_id |
|-------|-------|-----------|
| 45678 | B | CS-319 |
| 76543 | A | CS-319 |

11. Find students who have taken all courses offered by the Comp. Sci. Department.

Ans : SELECT  d.dept_name,c.course_id,s.name

from student as s

INNER join department as d

on d.dept_name=s.dept_name

INNER join course as c

on c.dept_name=d.dept_name

WHERE d.dept_name="Comp. Sci."

| dept_name | course_id | name |
|---|---|---|
| Comp. Sci. | CS-101 | Zhang |
| Comp. Sci. | CS-190 | Zhang |
| Comp. Sci. | CS-315 | Zhang |
| Comp. Sci. | CS-319 | Zhang |
| Comp. Sci. | CS-347 | Zhang |
| Comp. Sci. | CS-101 | Shankar |
| Comp. Sci. | CS-190 | Shankar |
| Comp. Sci. | CS-315 | Shankar |
| Comp. Sci. | CS-319 | Shankar |
| Comp. Sci. | CS-347 | Shankar |
| Comp. Sci. | CS-101 | Williams |
| Comp. Sci. | CS-190 | Williams |
| Comp. Sci. | CS-315 | Williams |
| Comp. Sci. | CS-319 | Williams |
| Comp. Sci. | CS-347 | Williams |
| Comp. Sci. | CS-101 | Brown |
| Comp. Sci. | CS-190 | Brown |
| Comp. Sci. | CS-315 | Brown |
| Comp. Sci. | CS-319 | Brown |
| Comp. Sci. | CS-347 | Brown |

12. List instructors who teach more than one course, including their IDs and the count of courses they teach.

Ans: SELECT i.ID, i.name, COUNT(t.course_id) AS course_count

FROM instructor AS i

JOIN teaches AS t ON i.ID = t.ID

GROUP BY i.ID, i.name

HAVING COUNT(t.course_id) > 1;

| ID | name | course_count |
|---|---|---|
| 10101 | Srinivasan | 3 |
| 45565 | Katz | 2 |
| 76766 | Crick | 2 |
| 83821 | Brandt | 3 |

13. List students whose advisors are from the Comp. Sci. department, including student IDs and advisor names

Ans  SELECT s.ID AS student_id, s.name AS student_name, i.name AS advisor_name

FROM student AS s

JOIN advisor AS a ON s.ID = a.s_ID

JOIN instructor AS i ON a.i_ID = i.ID

WHERE i.dept_name = 'Comp. Sci.';

| student_id | student_name | advisor_name |
|---|---|---|
| 12345 | Shankar | Srinivasan |
| 00128 | Zhang | Katz |
| 76543 | Brown | Katz |

14. Find departments that do not offer any courses, including department names.

Ans: SELECT d.dept_name

FROM department AS d

LEFT JOIN course AS c ON d.dept_name = c.dept_name

WHERE c.course_id IS NULL;

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0005 seconds.)

SELECT d.dept_name FROM department AS d LEFT JOIN course AS c ON d.dept_name = c.dept_name WHERE c.course_id IS NULL;

☐ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

**dept_name**

Query results operations

Create view

15. Calculate the average grade for each course, including course codes, titles, and average grades

Ans: SELECT

    c.course_id,

    c.title,

    AVG(

        CASE

            WHEN t.grade = 'A' THEN 4.0

            WHEN t.grade = 'B' THEN 3.0

            WHEN t.grade = 'C' THEN 2.0

            WHEN t.grade = 'D' THEN 1.0

            WHEN t.grade = 'F' THEN 0.0

            ELSE NULL

        END

    ) AS average_grade

FROM

    course AS c

JOIN

    takes AS t ON c.course_id = t.course_id

GROUP BY

  c.course_id, c.title;

| course_id | title | average_grade |
|---|---|---|
| BIO-101 | Intro. to Biology | 4.00000 |
| BIO-301 | Genetics | NULL |
| CS-101 | Intro. to Computer Science | 2.50000 |
| CS-190 | Game Design | 4.00000 |
| CS-315 | Robotics | 3.50000 |
| CS-319 | Image Processing | 3.50000 |
| CS-347 | Database System Concepts | 4.00000 |
| EE-181 | Intro. to Digital Systems | 2.00000 |
| FIN-201 | Investment Banking | NULL |
| HIS-351 | World History | 3.00000 |
| MU-199 | Music Video Production | NULL |
| PHY-101 | Physical Principles | NULL |

16. Determine which time slot (e.g., 'A', 'B', 'C', etc.) is most frequently assigned to sections of each course.

Ans: WITH TimeSlotFrequency AS (

  SELECT

    s.course_id,

    s.time_slot_id,

    COUNT(*) AS frequency

  FROM

    section AS s

  GROUP BY

    s.course_id, s.time_slot_id

),

MaxFrequency AS (

```sql
    SELECT
        course_id,
        MAX(frequency) AS max_frequency
    FROM
        TimeSlotFrequency
    GROUP BY
        course_id
)
SELECT
    tf.course_id,
    tf.time_slot_id,
    tf.frequency
FROM
    TimeSlotFrequency AS tf
JOIN
    MaxFrequency AS mf
    ON tf.course_id = mf.course_id AND tf.frequency = mf.max_frequency;
```

| course_id | time_slot_id | frequency |
|-----------|--------------|-----------|
| BIO-101 | B | 1 |
| BIO-301 | A | 1 |
| CS-101 | F | 1 |
| CS-101 | H | 1 |
| CS-190 | A | 1 |
| CS-190 | E | 1 |
| CS-315 | D | 1 |
| CS-319 | B | 1 |
| CS-319 | C | 1 |
| CS-347 | A | 1 |
| EE-181 | C | 1 |
| FIN-201 | B | 1 |
| HIS-351 | C | 1 |
| MU-199 | D | 1 |
| PHY-101 | A | 1 |

17. Find the student who has taken the highest number of credits in each department.

Ans: WITH StudentCredits AS (

  SELECT

    s.ID,

    s.name,

    s.dept_name,

    SUM(c.credits) AS total_credits

  FROM

    student AS s

  JOIN

    takes AS t ON s.ID = t.ID

  JOIN

    course AS c ON t.course_id = c.course_id

```
    GROUP BY

        s.ID, s.name, s.dept_name

),

MaxCredits AS (

    SELECT

        dept_name,

        MAX(total_credits) AS max_credits

    FROM

        StudentCredits

    GROUP BY

        dept_name

)

SELECT

    sc.ID,

    sc.name,

    sc.dept_name,

    sc.total_credits

FROM

    StudentCredits AS sc

JOIN

    MaxCredits AS mc ON sc.dept_name = mc.dept_name AND sc.total_credits =
mc.max_credits;
```

| ID | name | dept_name | total_credits |
|---|---|---|---|
| 98988 | Tanaka | Biology | 8 |
| 12345 | Shankar | Comp. Sci. | 14 |
| 98765 | Bourikas | Elec. Eng. | 7 |
| 23121 | Chavez | Finance | 3 |
| 19991 | Brandt | History | 3 |
| 55739 | Sanchez | Music | 3 |
| 45678 | Levy | Physics | 11 |

18. List courses that were not taught in the Spring 2018 semester.

Ans: SELECT c.course_id, c.title

FROM course AS c

WHERE NOT EXISTS (

  SELECT 1

  FROM teaches AS t

  WHERE t.course_id = c.course_id

   AND t.semester = 'Spring'

   AND t.year = 2018

);

| ←T→ | | | | course_id | title |
|------|------|------|------|-----------|-------|
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | BIO-101 | Intro. to Biology |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | BIO-301 | Genetics |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | BIO-399 | Computational Biology |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | CS-190 | Game Design |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | CS-347 | Database System Concepts |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | EE-181 | Intro. to Digital Systems |
| ☐ | 🖊 Edit | ⧉ Copy | ⊖ Delete | PHY-101 | Physical Principles |

19. Count the number of courses taught by each department and list them in descending order.

Ans: SELECT

   d.dept_name,

   COUNT(DISTINCT c.course_id) AS course_count

FROM

   department AS d

JOIN

   course AS c ON d.dept_name = c.dept_name

JOIN

   teaches AS t ON c.course_id = t.course_id

GROUP BY

   d.dept_name

ORDER BY

   course_count DESC;

| dept_name | course_count ▾ 1 |
|-----------|------------------|
| Comp. Sci. | 5 |
| Biology | 2 |
| Elec. Eng. | 1 |
| Music | 1 |
| Finance | 1 |
| Physics | 1 |
| History | 1 |

20. Display the relationship between instructors, students, and the courses they are enrolled in, including the course code, student ID, student name, instructor ID, and instructor name

Ans:  SELECT

   c.course_id,

   s.ID AS student_id,

   s.name AS student_name,

   i.ID AS instructor_id,

   i.name AS instructor_name

FROM

   takes AS t

JOIN

   course AS c ON t.course_id = c.course_id

JOIN

   student AS s ON t.ID = s.ID

JOIN

   section AS sec ON t.course_id = sec.course_id AND t.sec_id = sec.sec_id AND t.semester = sec.semester AND t.year = sec.year

JOIN

teaches AS te ON sec.course_id = te.course_id AND sec.sec_id = te.sec_id AND sec.semester = te.semester AND sec.year = te.year

JOIN

   instructor AS i ON te.ID = i.ID

| course_id | student_id | student_name | instructor_id | instructor_name |
|-----------|-----------|--------------|---------------|-----------------|
| CS-101 | 00128 | Zhang | 10101 | Srinivasan |
| CS-101 | 12345 | Shankar | 10101 | Srinivasan |
| CS-101 | 45678 | Levy | 10101 | Srinivasan |
| CS-101 | 54321 | Williams | 10101 | Srinivasan |
| CS-101 | 76543 | Brown | 10101 | Srinivasan |
| CS-101 | 98765 | Bourikas | 10101 | Srinivasan |
| CS-315 | 12345 | Shankar | 10101 | Srinivasan |
| CS-315 | 98765 | Bourikas | 10101 | Srinivasan |
| CS-347 | 00128 | Zhang | 10101 | Srinivasan |
| CS-347 | 12345 | Shankar | 10101 | Srinivasan |
| FIN-201 | 23121 | Chavez | 12121 | Wu |
| MU-199 | 55739 | Sanchez | 15151 | Mozart |
| PHY-101 | 44553 | Peltier | 22222 | Einstein |
| HIS-351 | 19991 | Brandt | 32343 | El Said |
| CS-101 | 45678 | Levy | 45565 | Katz |
| CS-319 | 45678 | Levy | 45565 | Katz |
| BIO-101 | 98988 | Tanaka | 76766 | Crick |
| BIO-301 | 98988 | Tanaka | 76766 | Crick |
| CS-190 | 12345 | Shankar | 83821 | Brandt |
| CS-190 | 54321 | Williams | 83821 | Brandt |
| CS-319 | 76543 | Brown | 83821 | Brandt |
| EE-181 | 76653 | Aoi | 98345 | Kim |