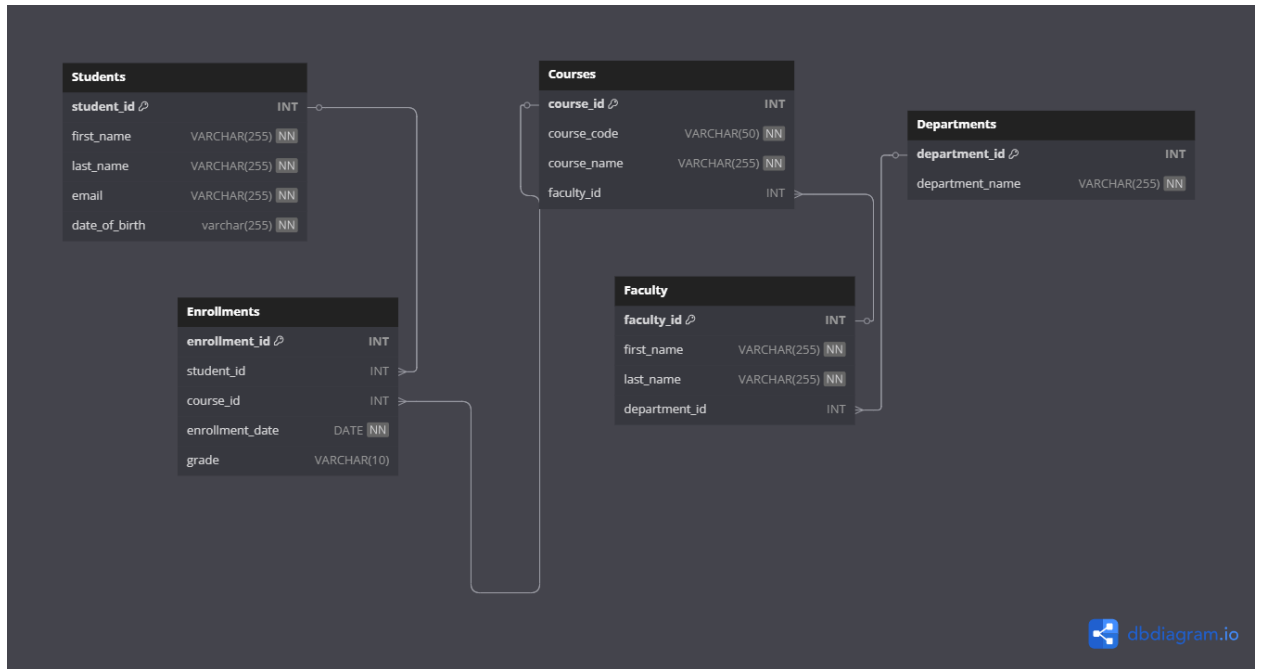


Name: Bunroeun sereysarakroetth

Class:ITE G9A3

- Er diagram (image or PDF)



- SQL file create table and insert statement file



insert and add table.sql

- sql queries with explanation

1. Retrieve all students who enrolled in a specific course.

```
-- 1. Retrieve all students who enrolled in a specific course.
-- To retrieve all students enrolled in a specific course (e.g., course with code "CS101"):
SELECT
    s.first_name AS student_first_name,s.last_name AS student_last_name,
    c.course_code,c.course_name,e.enrollment_date,e.grade
FROM Enrollments e
JOIN Students s ON e.student_id = s.student_id
JOIN Courses c ON e.course_id = c.course_id
WHERE c.course_code = 'CS101';
```

2. Find all faculty members in a particular department.

```
-- 2. Find all faculty members in a particular department.
-- To find all faculty members in a specific department (e.g., "Computer Science"):
SELECT
    f.first_name AS faculty_first_name,
    f.last_name AS faculty_last_name,
    d.department_name
FROM Faculty f
JOIN Departments d ON f.department_id = d.department_id
WHERE d.department_name = 'Computer Science';
```

3. List all courses a particular student is enrolled in.

```
-- 3. List all courses a particular student is enrolled in.
-- To list all courses a particular student is enrolled in (e.g., student with student_id 1):

SELECT
    c.course_code,
    c.course_name,
    e.enrollment_date,
    e.grade
FROM Enrollments e
JOIN Courses c ON e.course_id = c.course_id
WHERE e.student_id = 1;
```

4. Retrieve students who have not enrolled in any course.

```
-- 4. Retrieve students who have not enrolled in any course.  
-- To retrieve all students who have not enrolled in any course:
```

```
SELECT  
    s.first_name AS student_first_name,  
    s.last_name AS student_last_name  
FROM Students s  
LEFT JOIN Enrollments e ON s.student_id = e.student_id  
WHERE e.student_id IS NULL;
```

5. Find the average grade of students in a specific course

```
-- 5. Find the average grade of students in a specific course.  
-- To find the average grade of students in a specific course (e.g., course with course_code 'CS101'):
```

```
SELECT  
    c.course_code,  
    c.course_name,  
    AVG(CASE  
        WHEN e.grade = 'A' THEN 4.0  
        WHEN e.grade = 'B' THEN 3.0  
        WHEN e.grade = 'C' THEN 2.0  
        WHEN e.grade = 'D' THEN 1.0  
        WHEN e.grade = 'F' THEN 0.0  
    )
```

- screenshots of query results

1. Retrieve all students who enrolled in a specific course.

insert and add table SQL File 1\* x

Limit to 1000 rows

```

4 • SELECT
5     s.first_name AS student_first_name,
6     s.last_name AS student_last_name,
7     c.course_code,
8     c.course_name,
9     e.enrollment_date,
10    e.grade
11 FROM Enrollments e
12 JOIN Students s ON e.student_id = s.student_id
13 JOIN Courses c ON e.course_id = c.course_id
14 WHERE c.course_code = 'CS101';

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	student_first_name	student_last_name	course_code	course_name	enrollment_date	grade
▶	Alice	Johnson	CS101	Introduction to Programming	2024-03-01	A

2. Find all faculty members in a particular department.

insert and add table SQL File 1\* x

Limit to 1000 rows

```

16 -- 2. Find all faculty members in a particular department
17 -- Execute the selected portion of the script or everything, if there is no selection
18 • SELECT
19     f.first_name AS faculty_first_name,
20     f.last_name AS faculty_last_name,
21     d.department_name
22 FROM Faculty f
23 JOIN Departments d ON f.department_id = d.department_id
24 WHERE d.department_name = 'Computer Science';
25
26

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	faculty_first_name	faculty_last_name	department_name
▶	John	Doe	Computer Science

3. List all courses a particular student is enrolled in.

Insert and add table SQL File 1 x

Limit to 1000 rows

```
27 -- 3. List all courses a particular student is enrolled in.
28 -- To list all courses a particular student is enrolled in (e.g., student with student_id 1):
29
30 • SELECT
31     c.course_code,
32     c.course_name,
33     e.enrollment_date,
34     e.grade
35 FROM Enrollments e
36 JOIN Courses c ON e.course_id = c.course_id
37 WHERE e.student_id = 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	course_code	course_name	enrollment_date	grade
▶	CS101	Introduction to Programming	2024-03-01	A

4. Retrieve students who have not enrolled in any course.

Limit to 1000 rows

```
39
40
41 -- 4. Retrieve students who have not enrolled in any course.
42 -- To retrieve all students who have not enrolled in any course:
43
44 • SELECT
45     s.first_name AS student_first_name,
46     s.last_name AS student_last_name
47 FROM Students s
48 LEFT JOIN Enrollments e ON s.student_id = e.student_id
49 WHERE e.student_id IS NULL;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |





	student_first_name	student_last_name
--	--------------------	-------------------

5. Find the average grade of students in a specific course

```

50
51 -- 5. Find the average grade of students in a specific course.
52 -- To find the average grade of students in a specific course (e.g., course with course_code 'CS101'):
53 • SELECT
54     c.course_code,
55     c.course_name,
56     AVG(CASE
57         WHEN e.grade = 'A' THEN 4.0
58         WHEN e.grade = 'B' THEN 3.0
59         WHEN e.grade = 'C' THEN 2.0
60         WHEN e.grade = 'D' THEN 1.0
61         WHEN e.grade = 'F' THEN 0.0
62     END) AS average_grade
54 FROM course c
55 JOIN enrollment e ON c.course_code = e.course_code
56 WHERE c.course_code = 'CS101'
57 GROUP BY c.course_code, c.course_name
58 ORDER BY c.course_code

```

Result Grid   Filter Rows:  | Export:  | Wrap Cell Content: 

	course_code	course_name	average_grade
▶	CS101	Introduction to Programming	4.00000