TASK -12 : MINI PROJECT

E-learning course management system

- **Identify the Entities : instructors ,students ,course and assessment**

1. Instructor

- Instructor_ID (Primary Key)
- Name
- Email
- Department
- Phone Number
- Qualification

2. Student

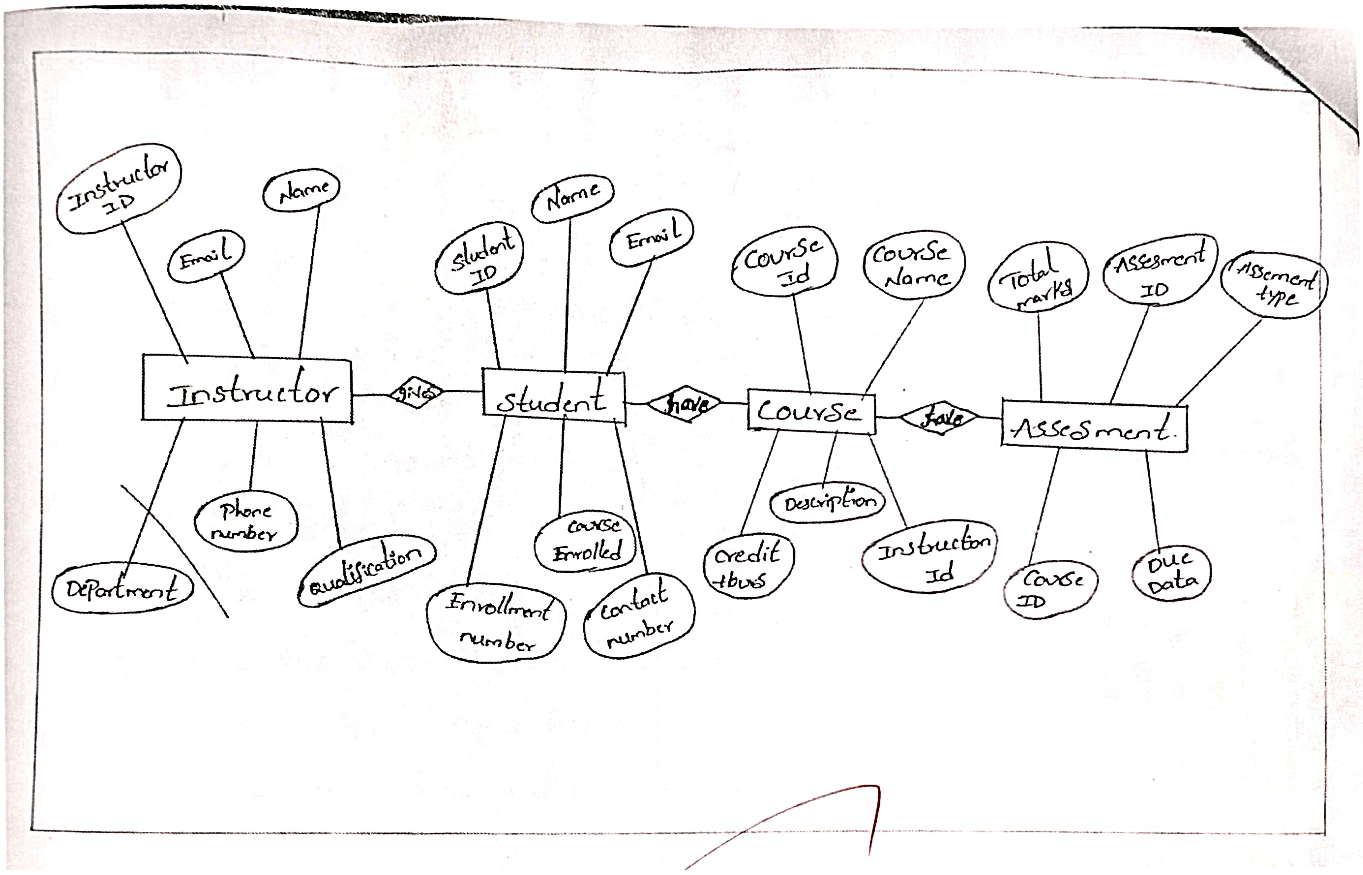
- Student_ID (Primary Key)
- Name
- Email
- Enrollment Number
- Course Enrolled
- Contact Number

. Course

- Course_ID (Primary Key)
- Course Name
- Description
- Credit Hours
- Instructor_ID

4. Assessment

- Assessment_ID (Primary Key)
- Assessment Type
- Total Marks
- Course_ID
- Due Date



❖ Apply normalization to the relational schema to remove redundancy for instruction, students ,course and assessment

Step- 1 : identify the unnormalized relation (UNF)

Student ID	Student name	Course ID	Course name	Instructor ID	Instructor name	Assessment type	Assessment ID	Marks
S01	Ananya	C101	DBMS	101	Dr . Mehta	Quiz	A1	18
S01	Ananya	C101	DBMS	101	Dr . Mehta	Exam	A2	85
S02	Priya	C102	DBMS	101	Dr . Mehta	Quiz	A1	15
S02	Priya	C102	JAVA	102	Prof. Sharma	Project	A3	90

Step 2: First Normal Form (1NF)

- Each cell contains atomic values.
- No repeating groups.

Step 3: Second Normal Form (2NF)

- Remove partial dependencies.
- Student Name depends only on Student ID.
- Course Name and Instructor ID depend only on Course ID.
- Instructor Name depends only on Instructor ID.

a) Student

Student _ID	Student Name
S01	Ananya
S02	Priya

b) Course

Course ID	Course Name	Instructor ID
C101	DBMS	101
C102	Java	102

c) Instructor

Instructor ID	Instructor Name
101	Dr . Mehta
102	Prof .shrama

d) Assesment

Assesment ID	Course ID	Assesment type
A1	C101	Quiz
A2	C101	Exam
A1	C101	Quiz
A3	C102	Project

Step 4: Third Normal Form (3NF)

- No transitive dependencies.

Final Schema:

- STUDENT(Student_ID, Student_Name)
- INSTRUCTOR(Instructor_ID, Instructor_Name)
- COURSE(Course_ID, Course_Name, Instructor_ID)
- ASSESSMENT(Assessment_ID, Course_ID, Assessment_Type, Marks, Student_ID)

❖ Wri

calculate grades

Assign Instructor to Course:

```
UPDATE COURSE SET Instructor_ID = '102' WHERE Course_ID = 'C101'; UPDATE
COURSE SET Instructor_ID = '102' WHERE Course_ID = 'C102';
```

Enroll Students in Courses:

```
CREATE TABLE ENROLLMENT (

Enrollment_ID INT PRIMARY KEY AUTO_INCREMENT,

Student_ID VARCHAR(10),

Course_ID VARCHAR(10),

FOREIGN KEY (Student_ID) REFERENCES STUDENT(Student_ID),

FOREIGN KEY (Course_ID) REFERENCES COURSE(Course_ID)

);
```

Now insert enrollment data :

```
INSERT INTO ENROLLMENT (Student_ID, Course_ID) VALUES ('S01', 'C101');

INSERT INTO ENROLLMENT (Student_ID, Course_ID) VALUES ('S01', 'C102');

INSERT INTO ENROLLMENT (Student_ID, Course_ID) VALUES ('S02', 'C101');

INSERT INTO ENROLLMENT (Student_ID, Course_ID) VALUES ('S03', 'C102';
```

a)Total Marks per Student per Course

```
SELECT

 S.Student_ID,

 S.Student_Name,

 C.Course_Name,

 SUM(A.Marks) AS Total_Marks

FROM ASSESSMENT A

JOIN STUDENT S ON A.Student_ID = S.Student_ID

JOIN COURSE C ON A.Course_ID = C.Course_ID

GROUP BY S.Student_ID, C.Course_Name;
```

b) Average Marks and Grade Assignment

```
SELECT
S.Student_Name,
C.Course_Name,
AVG(A.Marks) AS Average_Marks,
CASE
 WHEN AVG(A.Marks) >= 85 THEN 'A'
 WHEN AVG(A.Marks) >= 80 THEN 'B'
 WHEN AVG(A.Marks) >= 50 THEN 'C'
 ELSE 'F'
END AS Grade
FROM ASSESSMENT A
JOIN STUDENT S ON A.Student_ID = S.Student_ID
JOIN COURSE C ON A.Course_ID = C.Course_ID
GROUP BY S.Student_Name, C.Course_Name;
```

✓

∷ Use mango DB to manage course (RUD operations)

Database & collection setup

Open the mango DB :

```
use learningDB;

db.createCollection("courses");
```

Create a collection :

```
db.courses.insertOne({

 Course_ID: "C102",

 CourseName: "Advanced Java",

 Modules: [

  { Module_ID: "1", Title: "Java Basics", Content: "Intro to Java..." },

  { Module_ID: "2", Title: "OOP Concepts", Content: "Classes and objects..." }

 ],

 Assessments: [

  { Assessment_ID: "A5", Type: "Project", Max_Marks: 100 }

 ]

});
```

Read :

```
db.courses.find(); // All courses

db.courses.findOne({ Course_ID: "C102" }); // Specific course
```

Update :

```
db.courses.updateOne(

 { Course_ID: "C102" },

 { $set: { CourseName: "Advanced Database Systems" } }

);

db.courses.updateOne(

 { Course_ID: "C102" },
```