Database (MySQL)

Create a database named **EDGE Training Management System (etms)** with the following tables and **insert data** in each table.

courses(id, title, course_code, course_level, credits, *instructor_id*)

instructors(id, name, email, designation, phone, salary)

learners(id, name , roll, email, date_of_birth, city)

enrollments(id, *students_id, course_id,* enroll_date, status)

assessments(id, *course_id,* assessment_title, total_marks)

results(id, *student_id, assessment_id*, obtained_marks)

Instructions:

You need to submit all SQL code for the following questions. So, save all your code in a SQL file.

Q1: Create a Database named **etms** and corresponding tables using SQL Commands. [5]

Q2: Insert data into each table using SQL Commands. [5]

Q3: Write SQL commands for the following questions: [10]

a) Find the list of students who enrolled in Database course.

```
SELECT students.name
FROM students
JOIN enrollments ON students.id = enrollments.student_id
JOIN courses ON enrollments.course_id = courses.id
WHERE courses.course_title = 'Database';
```

b) How many students enrolled in all courses?

```
SELECT COUNT(*) FROM enrollments;
```

c) How many students enrolled for course having id 2?

```
SELECT COUNT(*) FROM enrollments where course_id=2;
```

d) Find the list of senior trainers who get salary more than 50,000.

```
SELECT COUNT(*) FROM instructors WHERE
designation='Senior Trainer' AND salary>50000;
```

e) Retrieve the list of learners and show them sorted alphabetically by their city.

```
SELECT name, city FROM learners ORDER BY city;
```

f) Increase the salary by 5% for those instructors who earn less than 50,000.

```
UPDATE instructors
SET salary=salary*1.05
WHERE salary<50000;
```

g) What is the total marks for the students with id 3?

```
SELECT SUM(obtained_marks) FROM results WHERE
student_id=3;
```

h) What are the mark assessment criteria?

```
SELECT COUNT(DISTINCT assessment_title) FROM assessments;
```

i) Show the list of all basic courses that are taken by instructor with id 1.

```
SELECT title, course_code, level FROM courses WHERE
instructor_id=1;
```

j) Find the list of employees who get more than average salary.

```
SELECT name, salary
FROM teachers
WHERE salary > (SELECT AVG(salary) FROM teachers);
```

Table 1: courses

Structure →

Column	Type	Description
id	INT	Primary Key, Auto Increment
title	VARCHAR(255)	The name of the course which is unique
course_code	VARCHAR(50)	Unique course code
level	ENUM	Course level (e.g., Basic, Intermediate, Advanced)
credits	INT	Number of credits (1-5)
instructor_id	INT	Foreign key to instructors table (id)

Data →

id	title	course_code	level	credits	instructor_id
1	Python Programming	P101	Basic	3	1
2	Database	DB202	Intermediate	4	2
3	Web Development	WD201	Intermediate	4	3
4	Web Design	WD102	Basic	3	1

Table 2: instructors

Structure →

Column	Type	Description
id	INT	Primary Key, Auto Increment
name	VARCHAR(255)	The name of the instructor
email	VARCHAR(100)	Unique email
designation	ENUM	Designation of Instrcutors (e.g., Junior Trainer, Trainer, Senior Trainer, Mentor, Industry Expert)
phone	VARCHAR(20)	Unique phone number
salary	DECIMAL(10,2)	Salary for instructor

Data →

id	name	email	designation	phone	salary
1	Alice B	ab@edu.com	Trainer	9876	60,000
2	Bob Dan	bd@edu.com	Senior Trainer	456789	80,000
3	Charlie Choe	cc@edu.com	Junior Trainer	98765	40000

Table 3: learners

Structure →

Column	Type	Description
id	INT	Primary Key, Auto Increment
name	VARCHAR(255)	The name of the students
roll	VARCHAR(100)	Unique roll
email	VARCHAR(100)	Unique email
date_of_birth	DATE	Date of birth of learner
city	VARCHAR(100)	City of learners

Data →

id	name	roll	email	date_of_birth	city
1	Alice Johnson	123	alice.j@students.com	10/10/99	Dhaka
2	Bob Williams	223	bob.w@students.com	22/6/99	Barishal
3	Carol Stevens	124	carol.s@students.com	12/11/01	Dhaka

Table 4: enrollments

Column	Type	Description
id	INT	Primary Key, Auto Increment
student_id	INT	Foreign key to students table (id)
course_id	INT	Foreign key to courses table (id)
enrollment_date	DATE	When a learner is enrolled in a course.
status	ENUM('Active', 'Completed', 'Dropped')	Current enrollment status

id	student_id	course_id	enrollment_date	status
1	1	1	1/9/23	1
2	2	2	15/8/23	2
3	3	3	1/9/23	1

Table 5: assessments

Column	Type	Description
id	INT	Primary Key, Auto Increment
course_id	INT	Foreign key to courses table (id)
assessment_title	VARCHAR(200)	Assessment title
total_marks	INT	Allocated marks

id	course_id	assessment_title	total_marks
1	1	Python Midterm	20
2	2	Quiz 1	10
3	3	Project	20

Table 6: results

Column	Type	Description
id	INT	Primary Key, Auto Increment
student_id	INT	Foreign key to students table (id)
assessment_id	INT	Foreign key to assessments table (id)
Marks_obtained	DECIMAL(5,2)	Total marks achieved by a student

id	student_id	assessment_id	marks_obtained
1	1	1	85
2	2	2	45
3	3	3	80