**Final Exam**

**Total Marks:100 Date: 14-11-2024**

| **This schema represents a basic Online Course Management System. Answer questions 1 - 5 based on this schema.**  CREATE TABLE Instructor (  InstructorID INT AUTO\_INCREMENT PRIMARY KEY,  Name VARCHAR(255) NOT NULL,  Email VARCHAR(255) NOT NULL UNIQUE,  Phone VARCHAR(15),  Department VARCHAR(50)  );  CREATE TABLE Course (  CourseID INT AUTO\_INCREMENT PRIMARY KEY,  Title VARCHAR(255) NOT NULL,  Credits INT NOT NULL,  InstructorID INT,  FOREIGN KEY (InstructorID) REFERENCES Instructor(InstructorID)  );  CREATE TABLE Enrollment (  EnrollmentID INT AUTO\_INCREMENT PRIMARY KEY,  StudentID INT,  CourseID INT,  EnrollmentDate DATE NOT NULL,  FOREIGN KEY (StudentID) REFERENCES Student(StudentID),  FOREIGN KEY (CourseID) REFERENCES Course(CourseID)  );  CREATE TABLE Student (  StudentID INT AUTO\_INCREMENT PRIMARY KEY,  Name VARCHAR(255) NOT NULL,  Email VARCHAR(255) NOT NULL UNIQUE,  Phone VARCHAR(15)  ); | | |
| --- | --- | --- |
|  | **QUESTIONS** | **Marks** |
| 1 | Draw an Entity-Relationship (ER) diagram to represent this Online Course Management System schema. | 10 |
| 2 | Write an SQL query to insert a new enrollment record for a student (e.g., StudentID 5) into the course with the highest credit hours.  INSERT INTO Enrollment (StudentID, CourseID, EnrollmentDate)  SELECT 5, CourseID, '2024-11-17'  FROM Course  ORDER BY Credits DESC  LIMIT 1; | 10 |
| 3 | Write an SQL UPDATE query to assign a new instructor to a course (e.g., CourseID 3) by updating the InstructorID.  UPDATE Course  SET InstructorID = 4  WHERE CourseID = 3; | 10 |
| 4 | Write an SQL query to find the names of instructors who teach the most credits (total).  SELECT Instructor.Name, SUM(Course.Credits) AS TotalCredits  FROM Instructor  JOIN Course ON Instructor.InstructorID = Course.InstructorID  GROUP BY Instructor.InstructorID, Instructor.Name  ORDER BY TotalCredits DESC  LIMIT 1; | 10 |
| 5 | Write an SQL query to list all students who are enrolled in more than two courses.  SELECT Student.Name, COUNT(Enrollment.CourseID) AS Cnt  FROM Student  JOIN Enrollment ON Student.StudentID = Enrollment.StudentID  GROUP BY Student.StudentID, Student.Name  HAVING COUNT(Enrollment.CourseID) > 2; | 10 |
| 6 | Design an ER diagram for a simple online retail system that includes entities such as Customers, Products, and Orders. Keep the diagram simple. | 10 |
| 7 | Explain the difference between GROUP BY and ORDER BY in SQL. Provide an example for each to illustrate.  GROUP BY is used to group rows based on one or more columns. It used aggregate functions.  ORDER BY is used for sorting query results in ascending or descending order. It organizes the data.  OrderDetails:   | orderId | productId | quantity | | --- | --- | --- | | 1 | 101 | 5 | | 2 | 102 | 10 | | 3 | 101 | 7 | | 4 | 103 | 3 | | 5 | 102 | 8 |   Group By (Sum of product Quantity):   | productId | TotalQuantity | | --- | --- | | 101 | 12 | | 102 | 18 | | 103 | 3 |   Order By Ascending for Quantity:   | orderID | productId | quantity | | --- | --- | --- | | 4 | 103 | 3 | | 1 | 101 | 5 | | 3 | 101 | 7 | | 5 | 102 | 8 | | 2 | 102 | 10 | | 10 |
| 8 | Given a table Instructor with a Salary column, write an SQL query to find the second-highest salary among instructors.  SELECT DISTINCT Salary  FROM Instructor  ORDER BY Salary DESC  LIMIT 1 OFFSET 1; | 10 |
| 9 | You have two tables, Instructor and Course. Use ON DELETE CASCADE on Course so that all courses are deleted when an instructor is removed.  CREATE TABLE Instructor (  InstructorID INT PRIMARY KEY,  Name VARCHAR(50)  );  CREATE TABLE Course (  CourseID INT PRIMARY KEY,  CourseName VARCHAR(50),  InstructorID INT,  FOREIGN KEY (InstructorID) REFERENCES Instructor(InstructorID) ON DELETE CASCADE  ); | 10 |
| 10 | Describe the most challenging topic you encountered in this course. Explain why it was challenging and how you overcame it.  Most challenging topic is drawing ER diagrams.  Because I face problems making correct relation.  I overcame drawing ER diagrams to study more and I saw the Conceptual session again. | 10 |