University Examination System

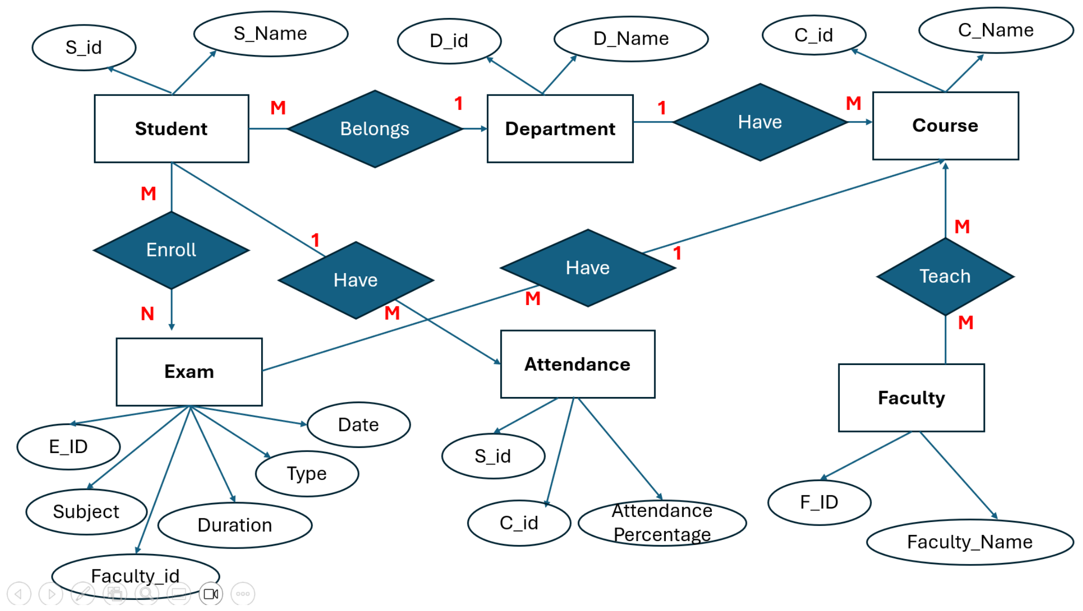
Design an Entity-Relationship schema for a university examination system that manages data about **exams**, **students**, **faculty members**, **courses**, and **departments**.

Each **department** has a unique name and is headed by a **faculty member**. A department can offer multiple **courses**, and each course has a unique course code, title, and is coordinated by a faculty member.**Faculty members** have an employee ID, name, and designation. They can teach multiple courses, coordinate specific courses, and also serve as heads of departments. A faculty member may handle multiple roles at once.

Students have a roll number and name, and each student belongs to one department. A student can enroll in multiple courses offered by that department. For each enrolled course, a student has an **attendance percentage** recorded.

Exams are created by faculty members .Each exam has a title, subject name (which is assumed to be the same as the course name), duration, date, type (internal or external), and is always linked to a specific course. Students may appear in multiple exams related to their courses, and for each exam, a student may have multiple attempts, with marks and attempt dates recorded for each.

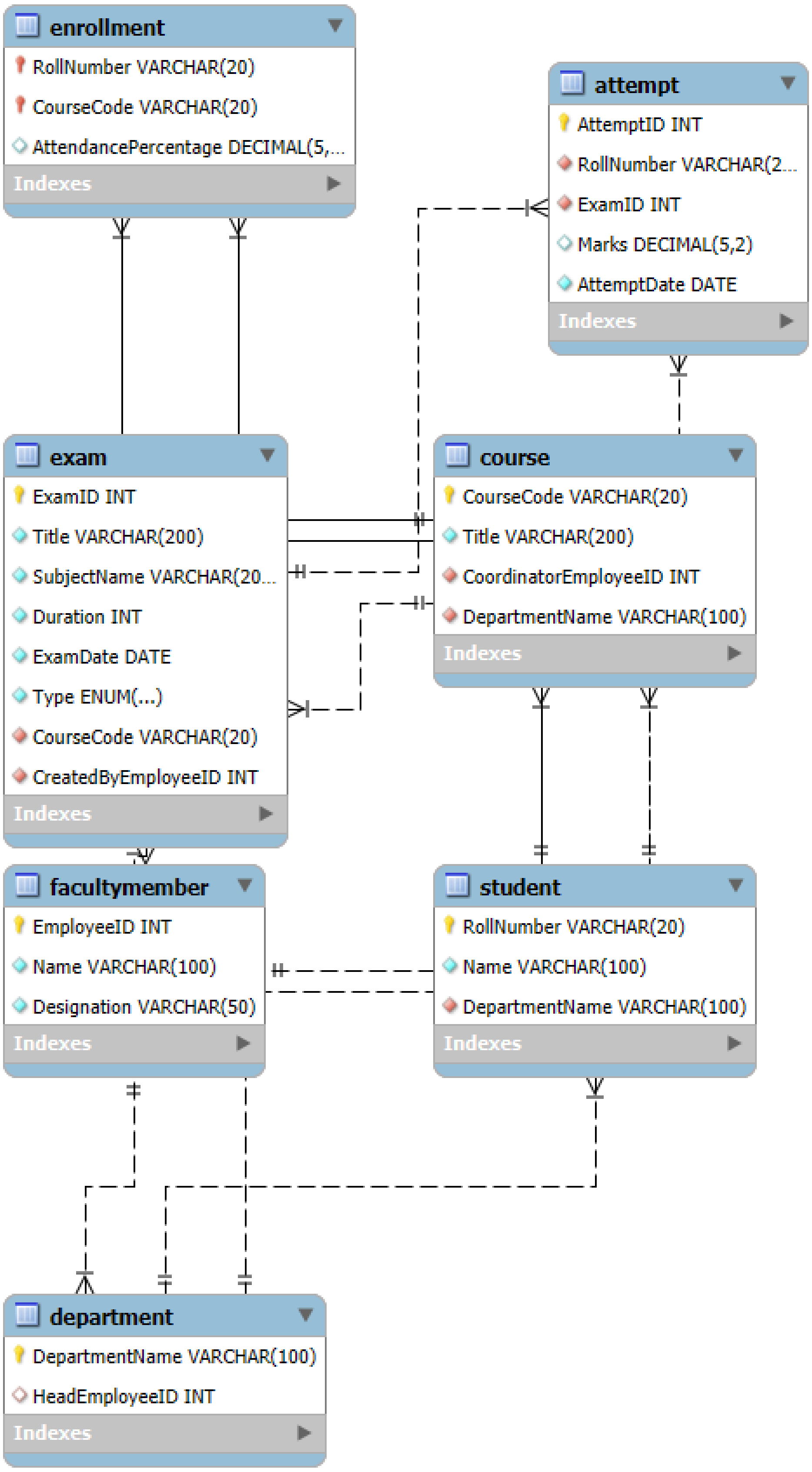
All relationships between students, courses, faculty, and exams must reflect these associations clearly — such as student-course enrollment, faculty-course teaching, course-department mapping, and exam-course ownership.



🧾 SQL Table Creation Statements:

```
-- 1. Department Table
CREATE TABLE Department (
    Dept_ID INT PRIMARY KEY AUTO_INCREMENT,
    Dept_Name VARCHAR(100) UNIQUE NOT NULL,
    Head_ID INT,  -- FK to Faculty
    FOREIGN KEY (Head_ID) REFERENCES
Faculty(Faculty_ID)
);

-- 2. Faculty Table
CREATE TABLE Faculty (
    Faculty_ID INT PRIMARY KEY,
    Faculty_Name VARCHAR(100) NOT NULL,
    Designation VARCHAR(50)
```

```
);

-- 3. Course Table
CREATE TABLE Course (
   Course_Code VARCHAR(10) PRIMARY KEY,
   Title VARCHAR(100) NOT NULL,
   Dept_ID INT,
   Coordinator_ID INT,
   FOREIGN KEY (Dept_ID) REFERENCES
Department(Dept_ID),
   FOREIGN KEY (Coordinator_ID) REFERENCES
Faculty(Faculty_ID)
);

-- 4. Faculty_Course_Teaching Table (Many-to-Many:
Faculty teaches multiple courses)
CREATE TABLE Faculty_Course_Teaching (
   Faculty_ID INT,
   Course_Code VARCHAR(10),
   PRIMARY KEY (Faculty_ID, Course_Code),
   FOREIGN KEY (Faculty_ID) REFERENCES
Faculty(Faculty_ID),
   FOREIGN KEY (Course_Code) REFERENCES
Course(Course_Code)
);

-- 5. Student Table
CREATE TABLE Student (
   Roll_No INT PRIMARY KEY,
    Student_Name VARCHAR(100) NOT NULL,
    Dept_ID INT,
    FOREIGN KEY (Dept_ID) REFERENCES
Department(Dept_ID)
);

-- 6. Student_Course_Enrollment Table (Many-to-Many:
Students enroll in multiple courses)
CREATE TABLE Student_Course_Enrollment (
    Roll_No INT,
    Course_Code VARCHAR(10),
    Attendance_Percentage DECIMAL(5,2),
    PRIMARY KEY (Roll_No, Course_Code),
    FOREIGN KEY (Roll_No) REFERENCES
Student(Roll_No),
    FOREIGN KEY (Course_Code) REFERENCES
Course(Course_Code)
);

-- 7. Exam Table
CREATE TABLE Exam (
    Exam_ID INT PRIMARY KEY AUTO_INCREMENT,
    Title VARCHAR(100) NOT NULL,
    Subject_Name VARCHAR(100) NOT NULL,  --
Usually same as course title
    Duration INT,  -- in minutes
    Exam_Date DATE,
    Exam_Type ENUM('Internal', 'External'),
    Course_Code VARCHAR(10),
    Creator_ID INT,
    FOREIGN KEY (Course_Code) REFERENCES
Course(Course_Code),
    FOREIGN KEY (Creator_ID) REFERENCES
Faculty(Faculty_ID)
);

-- 8. Student_Exam_Attempt Table
CREATE TABLE Student_Exam_Attempt (
    Attempt_ID INT PRIMARY KEY AUTO_INCREMENT,
    Roll_No INT,
    Exam_ID INT,
    Attempt_Date DATE,
    Marks DECIMAL(5,2),
    FOREIGN KEY (Roll_No) REFERENCES
Student(Roll_No),
    FOREIGN KEY (Exam_ID) REFERENCES
Exam(Exam_ID)
);
```


enrollment
RollNumber VARCHAR(20)
CourseCode VARCHAR(20)
AttendancePercentage DECIMAL(5,...
Indexes
attempt
AttemptID INT
RollNumber VARCHAR(2...
ExamID INT
Marks DECIMAL(5,2)
AttemptDate DATE
Indexes
exam
ExamID INT
Title VARCHAR(200)
SubjectName VARCHAR(20...
Duration INT
ExamDate DATE
Type ENUM(...)
CourseCode VARCHAR(20)
CreatedByEmployeeID INT
Indexes
course
CourseCode VARCHAR(20)
Title VARCHAR(200)
CoordinatorEmployeeID INT
DepartmentName VARCHAR(100)
Indexes
facultymember
EmployeeID INT
Name VARCHAR(100)
Designation VARCHAR(50)
Indexes
student
RollNumber VARCHAR(20)
Name VARCHAR(100)
DepartmentName VARCHAR(100)
Indexes
department
DepartmentName VARCHAR(100)
HeadEmployeeID INT
Indexes