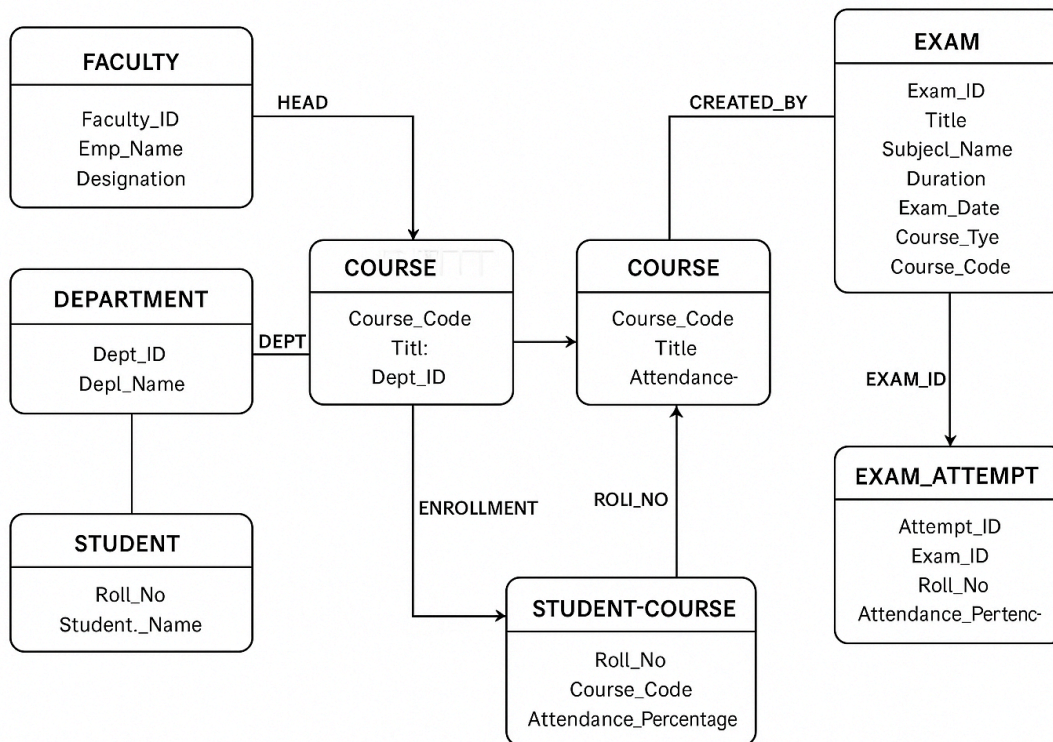


## **Problem 2: University Examination System**

**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.**



## 1. Faculty Table

```

CREATE TABLE Faculty (
  Faculty_ID INT PRIMARY KEY,
  Emp_Name VARCHAR(100),
  Designation VARCHAR(50)
);
  
```

## 2. Department Table

```

CREATE TABLE Department (
  Dept_ID INT PRIMARY KEY,
  Dept_Name VARCHAR(100),
  Head_Faculty_ID INT ,
  FOREIGN KEY (Head_Faculty_ID) REFERENCES Faculty(Faculty_ID) );
  
```

## 3. Course Table

```

CREATE TABLE Course (
  
```

```
Course_Code VARCHAR(10) PRIMARY KEY,  
Title VARCHAR(100),  
Dept_ID INT,  
Coordinator_ID INT,  
FOREIGN KEY (Dept_ID) REFERENCES Department(Dept_ID),  
FOREIGN KEY (Coordinator_ID) REFERENCES Faculty(Faculty_ID)  
);
```

#### 4. Student Table

```
CREATE TABLE Student (  
Roll_No INT PRIMARY KEY,  
Student_Name VARCHAR(100),  
Dept_ID INT,  
FOREIGN KEY (Dept_ID) REFERENCES Department(Dept_ID)  
);
```

#### 5. Student\_Course (Enrollment) Table

```
CREATE TABLE Student_Course (  
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 );
```

#### 6. Exam Table

```
CREATE TABLE Exam (  
Exam_ID INT PRIMARY KEY,  
Title VARCHAR(100),  
Subject_Name VARCHAR(100),  
Duration INT, -- in minutes  
Exam_Date DATE,
```

```
Exam_Type VARCHAR(20), -- 'internal' or 'external'  
Course_Code VARCHAR(10),  
Created_By INT,  
FOREIGN KEY (Course_Code) REFERENCES Course(Course_Code),  
FOREIGN KEY (Created_By) REFERENCES Faculty(Faculty_ID)  
);
```

## 7. Exam\_Attempt Table

```
CREATE TABLE Exam_Attempt (  
    Attempt_ID INT PRIMARY KEY,  
    Exam_ID INT,  
    Roll_No INT,  
    Attempt_Number INT,  
    Marks INT,  
    FOREIGN KEY (Exam_ID) REFERENCES Exam(Exam_ID),  
    FOREIGN KEY (Roll_No) REFERENCES Student(Roll_No)  
);
```

