Task 1.1: Superkey and Candidate Key Analysis

Relation A: Employee

```
Employee(EmpID, SSN, Email, Phone, Name, Department, Salary)
```

Sample Data:

EmpID	SSN	Email	Phone	Name	Department	Salary
101	123-45-6789	john@company.com	555-0101	John	IT	75000
102	987-65-4321	mary@company.com	555-0102	Mary	HR	68000
103	456-78-9123	bob@company.com	555-0103	Bob	IT	72000

Your Tasks: 1. List at least 6 different superkeys 2. Identify all candidate keys 3. Which candidate key would you choose as primary key and why? 4. Can two employees have the same phone number? Justify your answer based on the data shown.

1. Super keys: [EmpID], [SSN], [Email], [Phone], [EmpID, SSN], [Email, phone]
2. Candidate keys: EmpID, SSN, Email, Phone
3. the best primary key is empID, since it is short, automatically created, unique.
4. based on the data shown, the phone can not repeat since it is created from empId which is unique

Relation B: Course Registration

```
Registration(StudentID, CourseCode, Section, Semester, Year, Grade, Credits)
```

Task 1.2: Foreign Key Design

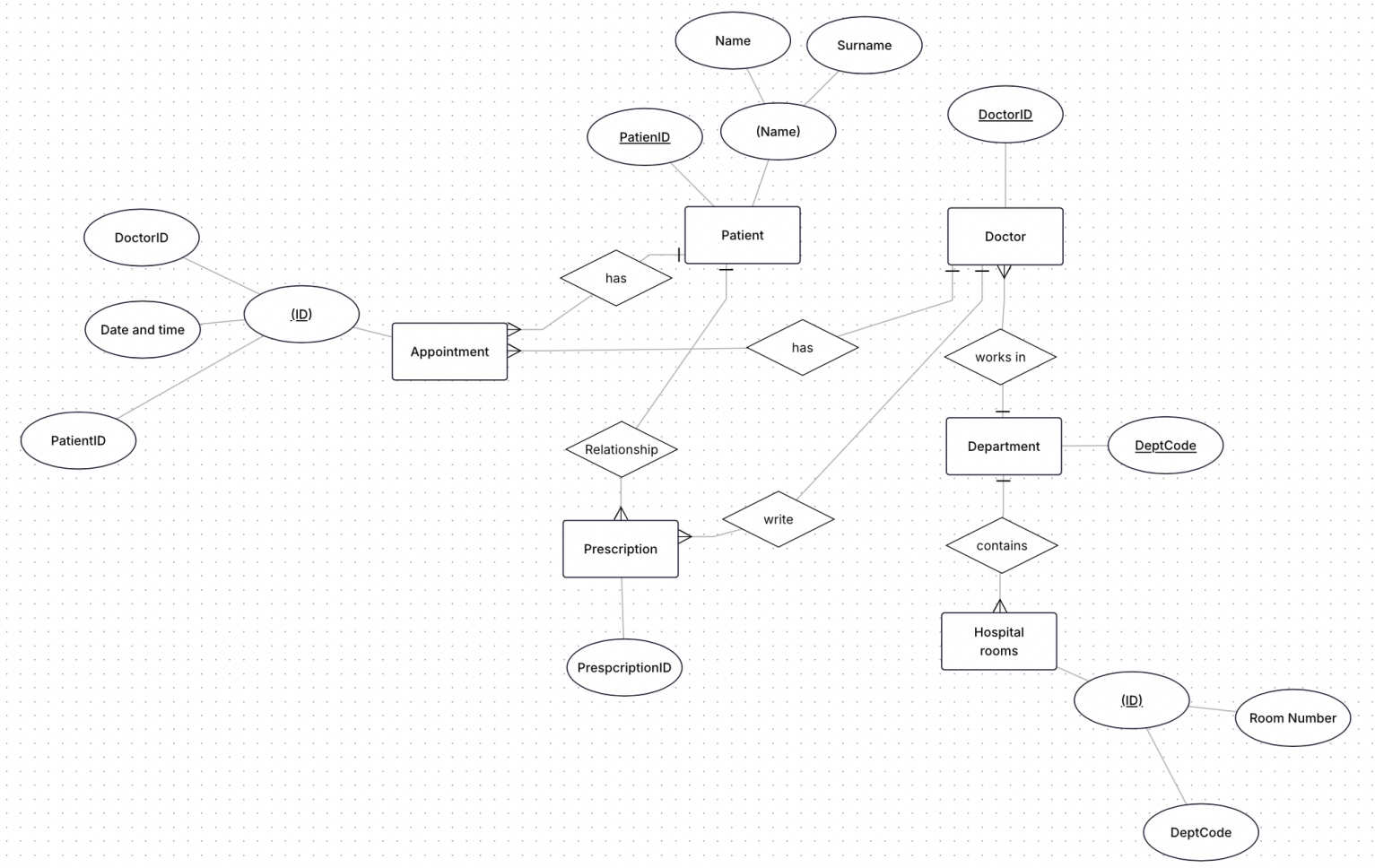
Design the foreign key relationships for this university system:

Table	Foreign key	Reference	Cardinality
Student	AdvisorID	Professor(profID)	Many to 1
Professor	Department	Department(deptcode)	Many to 1
Course	DepartmentCode	Department(deptcode)	Many to 1
Department	ChairID	Professor(profID)	1 to 1
Enrollment	StudentID	Student(studentID)	Many to 1
Enrollment	CourseID	Course(CourseID)	Many to 1

Your Tasks: 1. Identify all foreign key relationships

Part 2: ER Diagram Construction

Task 2.1: Hospital Management System



Attributes:
simple: id's
composite: addresses(zip, street, city), fullname(name, surname)
multi-valed: phone
derived: age(from birthdate)

Task 2.2: E-commerce Platform

Scenario: Design a simplified e-commerce database.

Requirements:

- **Customers** place **Orders** for **Products**
- **Products** belong to **Categories** and are supplied by **Vendors**
- **Orders** contain multiple **Order Items** (quantity and price at time of order)
- **Products** have reviews and ratings from customers
- Track **Inventory** levels for each product
- **Shipping addresses** can be different from customer billing addresses

Your Tasks: 1. Create a complete ER diagram 2. Identify at least one weak entity and justify why it's weak 3. Identify at least one many-to-many relationship that needs attributes

Part 4: Normalization Workshop

Task 4.1: Denormalized Table Analysis

Given Table:

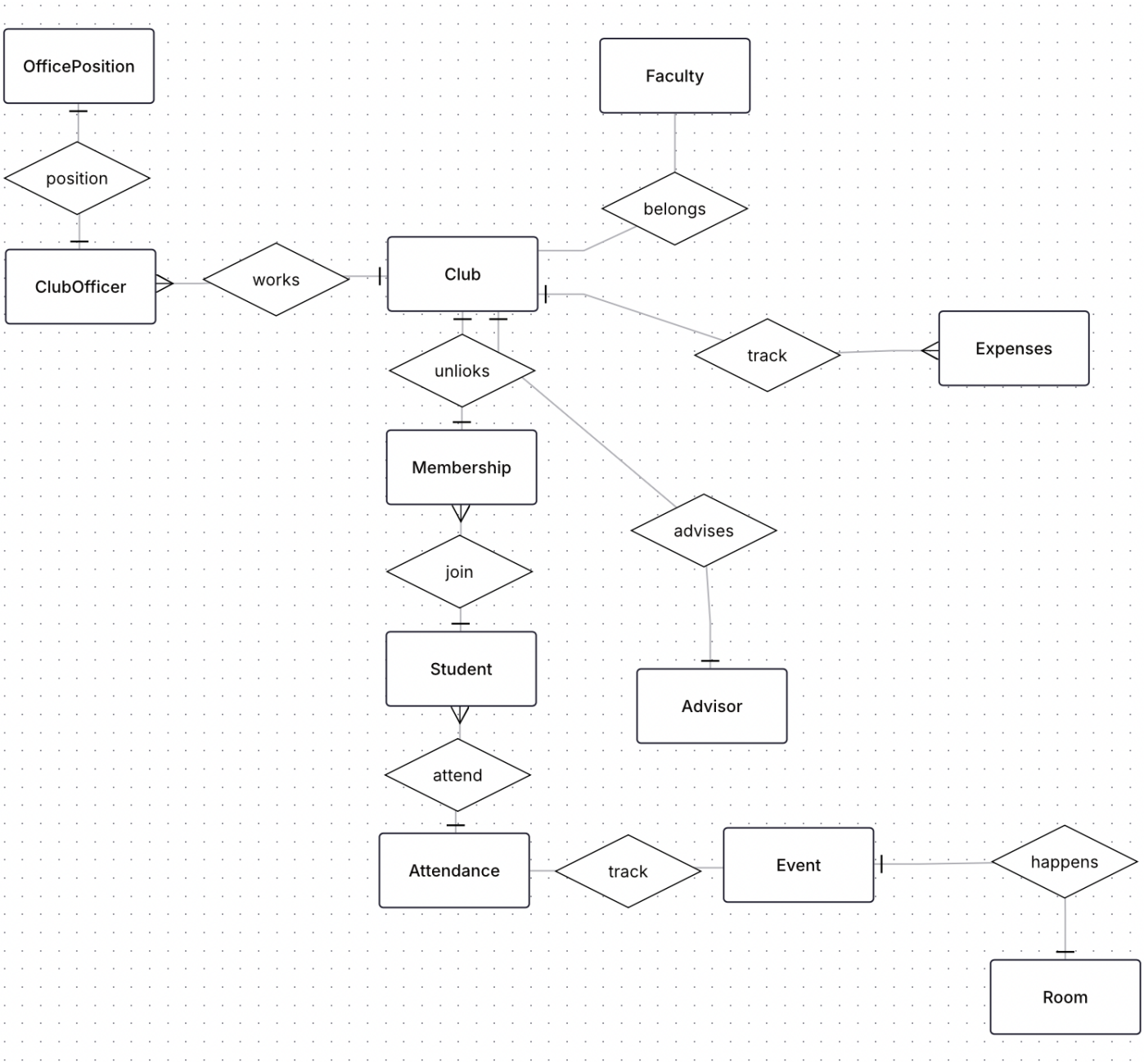
```
StudentProject(StudentID, StudentName, StudentMajor, ProjectID, ProjectTitle,
               ProjectType, SupervisorID, SupervisorName, SupervisorDept,
               Role, HoursWorked, StartDate, EndDate)
FD's:
StudentID -> StudentName, StudentMajor
ProjectID -> ProjectTitle, ProjectType, startDate, endDate
SupervisorID -> SupervisorName, SupervisorDept
[StudentID, ProjectID] -> Role. hoursWorked
Problems with table: repeating info names, when we update we would have to
               update all tables instead of one cell.
1nf -> atomic data
2nf -> remove partial dependencies
```

Student(StudentID PK, StudentName, StudentMajor)
Project(ProjectID PK, ProjectTitle, ProjectType, StartDate, EndDate, SupervisorID FK)
Supervisor(SupervisorID PK, SupervisorName, SupervisorDept)
StudentProject(StudentID FK, ProjectID FK, Role, HoursWorked)
3nf -> already no transitive dependencies

Part 5: Design Challenge

Task 5.1: Real-World Application

Scenario: Your university wants to track student clubs and organizations with the following requirements:


OfficePosition
position
ClubOfficer
works
Club
Faculty
belongs
track
Expenses
unlioks
Membership
join
Student
attend
Attendance
advises
Advisor
track
Event
happens
Room