5.2

normalized relational schemaStudent(StudentID PK, Name, Email, Phone, Major)

* Club(ClubID PK, ClubName, Description, Budget)
* Membership(StudentID FK, ClubID FK, JoinDate, PRIMARY KEY(StudentID, ClubID))
* OfficerPosition(PositionID PK, PositionName)
* ClubOfficer(ClubID FK, PositionID FK, StudentID FK, TermStart DATE, TermEnd DATE, PRIMARY KEY(ClubID, PositionID, TermStart))
* FacultyAdvisor(AdvisorID PK, Name, Dept)
* ClubAdvisor(ClubID FK, AdvisorID FK, StartDate, EndDate, PRIMARY KEY(ClubID, StartDate))
* Event(EventID PK, ClubID FK, Title, Description, EventDate, StartTime, EndTime)
* Room(RoomID PK, Building, RoomNumber)
* RoomReservation(ReservationID PK, EventID FK, RoomID FK, ReservationStart, ReservationEnd)
* Attendance(EventID FK, StudentID FK, Attended BOOLEAN, PRIMARY KEY(EventID, StudentID))
* Expense(ExpenseID PK, ClubID FK, Amount DECIMAL, Date, Description)

5.3

I have allocated ClubOfficer with a separate table with PositionID and TermStart/TermEnd period.

• Positions have the nature of roles and terms — you need to keep a history (who was president in the past).

• We need a guarantee that there will be no more than one president in one club during the same period → the simplest way to express it is through unique restrictions in ClubOfficer.

• If you put the Role in Membership, it is difficult to keep a history of officer shifts and impose restrictions on the uniqueness of the position.

5.4

1.Find all students who are officers in the Computer Science Club.

2.List all events scheduled for next week with their room reservations.

3.Show current club budget and total expenses for each club for the current semester.