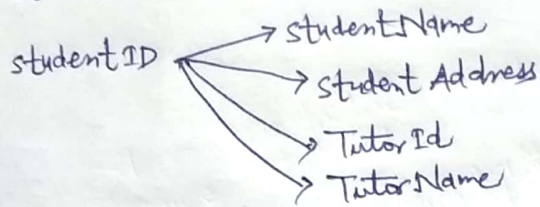


Ans 8-3

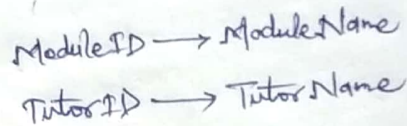
student

studentID	ModuleID	ModuleName	studentName	student Address	TutorID	TutorName

As studentID is Unique, So,



Similarly



functional dependencies

finding Candidate Key

closure of studentID, ModuleID = studentID, studentName, student Address, TutorID, TutorName, ModuleID, ModuleName

Hence, candidate Key = (studentID, ModuleID)

It contains partial dependency. Hence not in 2NF
(because studentID alone gives studentID, its name, address
TutorID, TutorName & ModuleID alone gives ModuleID &
ModuleName attributes uniquely)

In order to remove partial dependencies

student (studentID, studentName, studentAddress,

TutorID, TutorName)

Module (ModuleID, ModuleName)

So we have a new relation

Student-Module (studentID, ModuleID)

It contains transitive dependency (because $\text{studentID} \rightarrow \text{TutorID}$ & $\text{TutorID} \rightarrow \text{TutorName}$)

In order to remove these dependency, we have the following relation

student (studentID, studentName, studentAddress)

Tutor (TutorID, TutorName)

So, Final relation schemas in 3NF:

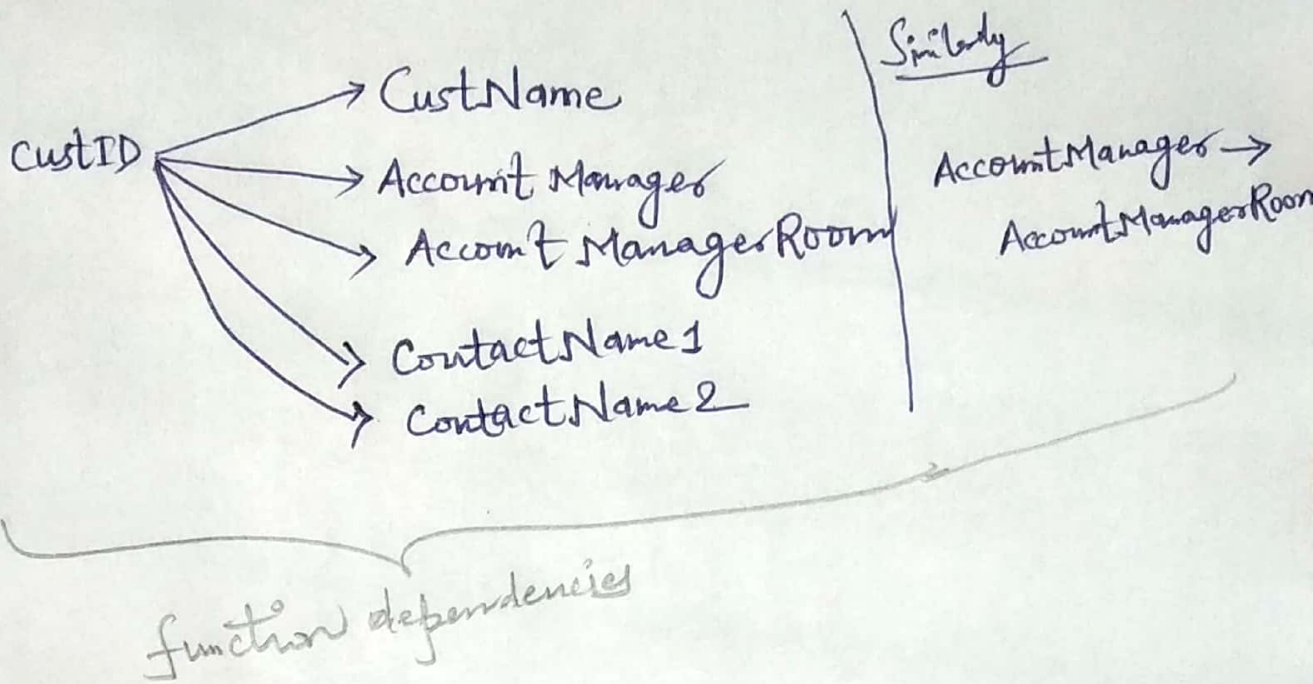
1. student (studentID, studentName, studentAddress)

2. student-Module (studentID, ModuleID)

3. Tutor (TutorID, TutorName)

4. Module (ModuleID, ModuleName)

Ans:-1 As CustID is Unique so,



As it contains dependency (transitive) because
 $\text{CustID} \rightarrow \text{Account Manager} \& \text{Account Manager} \rightarrow \text{Account Manager Room}$

So, the relation Schemas,

Customer (CustID, CustName, Account Manager)

cust_contact (CustID, contact)

Account (Account Manager, Account Manager Room)

Ans:-2 Entity : Patient, Hospital, Doctor, Record

Relationship : admitted in, has etc.

Attributes : P-id, P-add, P-Name

D-id, Salary etc.

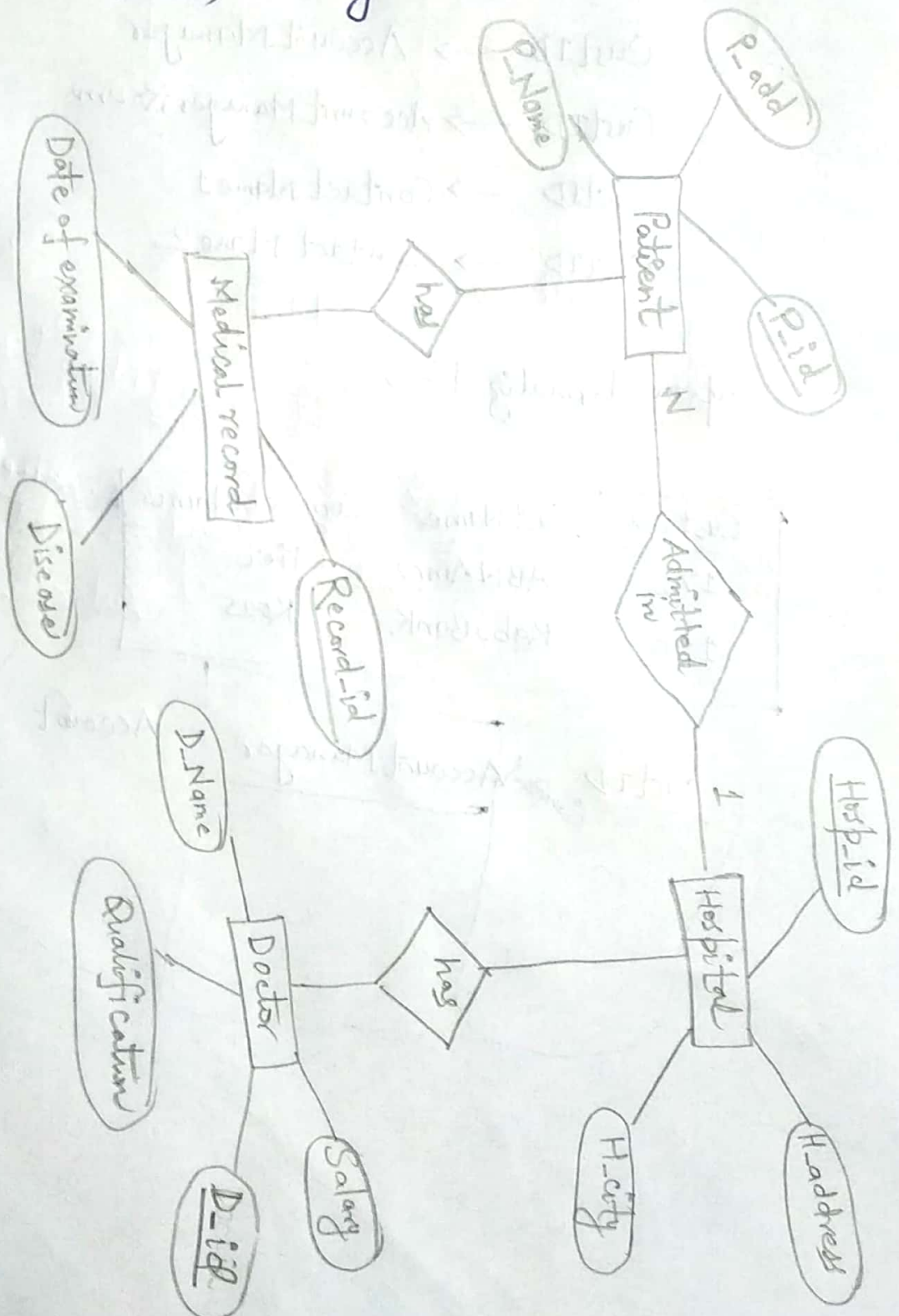


fig: ER diagram for hospital management system