

RDBMS concepts Morning Assignment

Question:1

Functional dependencies are:

CustID->CustName,AccountManager,AccountManagerRoom,Conatct1,Conact2

AccountManager->AccountManagerRoom

CustID-> Conatct1,Conact2

In 1NF we should not have repeating groups. A customer has multiple contacts (contact1 and contact 2). Here we can create a new entity **Contact_id** to store all names of contacts.

We have 2 tables now.

Table 1: **Customers** – CustID , CustName, AccountManager, AccountManagerRoom, Contact_id

Table 2: **Contacts** – Conatct_id, ContactName

In 2NF there are no partial dependency. Now the key is (CustID). But

CustID->Contact_ID is also valid. Contact_id is a primary key in Table-Contacts. So here partial dependency exists. So we need to split table and a new table **Cust_Contacts** is created.

So now the table are :

Customers: CustID , CustName, AccountManager, AccountManagerRoom,

Cust_Contacts: CustID, Contact_id

Contacts – Conatct_id, ContactName

In 3NF we need to eliminate transitive dependencies. AccountManagerRoom is trasitively dependent on AccountManager. So we need to split and create a new entity AccountManager_id. So the tables become

Customers: CustID , CustName, AccountManager_id

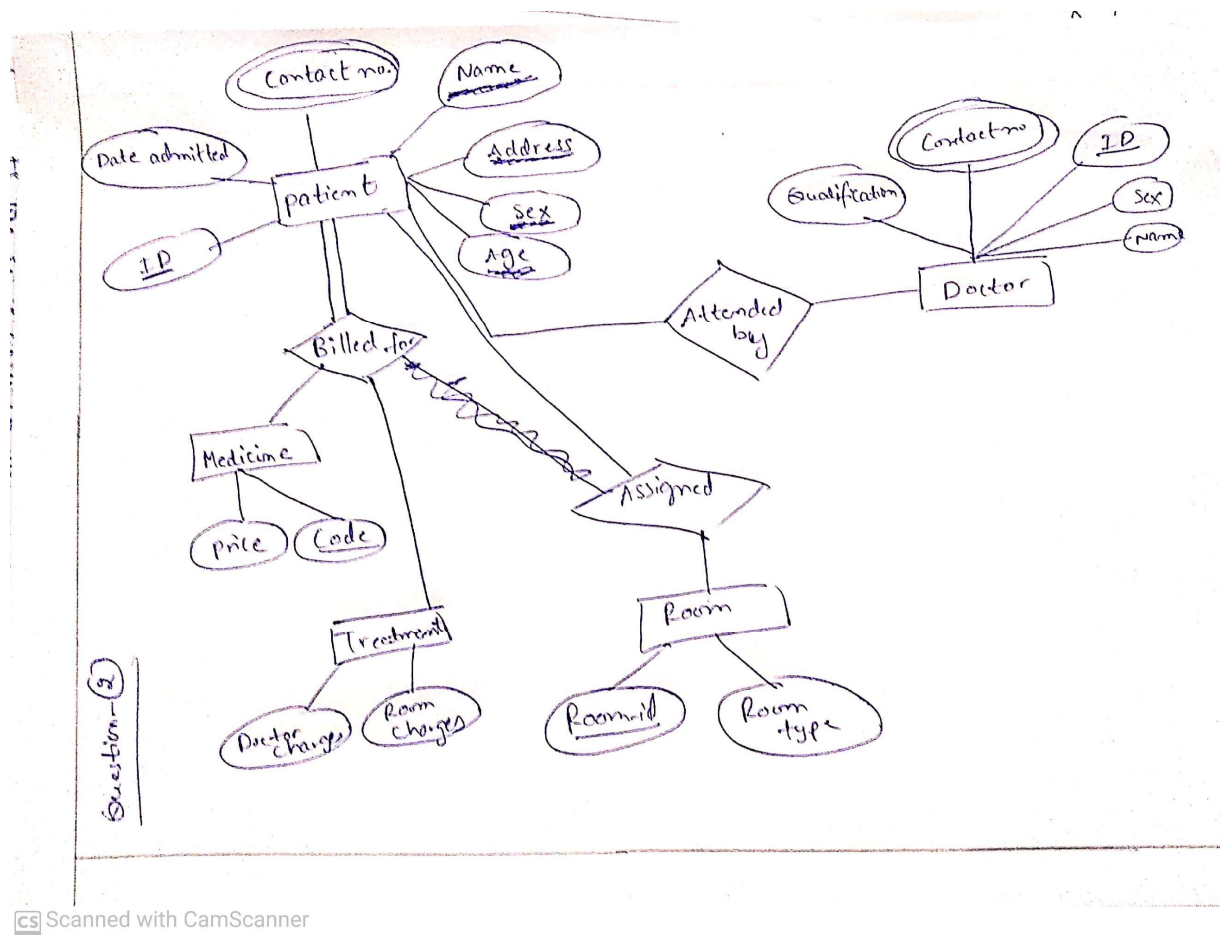
AccountManger: AccountManager_id, AccountManager,
AccountManagerRoom

Cust_Contacts: CustID, Contact_id

Contacts – Conatct_id, ContactName

It is in BCNF .

Question :2



Question 3:

Each student has unique id and tutor

StudentID → StudentName, StudentAddress, TutorId, TutorName

$\text{ModuleID} \rightarrow \text{ModuleName}$

$\text{TutorID} \rightarrow \text{TutorName}$

The key is (StudentID, ModuleID)

It is in 1NF.

But StudentID also identifies StudentName, StudentAddress, TutorID, TutorName uniquely. And also ModuleID identifies ModuleID, ModuleName.

Two partial dependencies exists. So we need to split student table into 3 tables as follows:

Student (StudentID, StudentName, StudentAddress, TutorID, TutorName)

Module (ModuleID, ModuleName)

Stu_Module (StudentID, ModuleID)

Now it is in 2NF.

$\text{TutorID} \rightarrow \text{TutorName}$ is transitive. ($\text{StudentID} \rightarrow \text{TutorID}$ and $\text{TutorID} \rightarrow \text{TutorName}$)

We need to split to be in 3NF.

Final tables are

Stu (StudentID, StudentName, StudentAddress)

Tutor (TutorID, TutorName)

Module (ModuleID, ModuleName)

Stu_Module (StudentID, ModuleID)