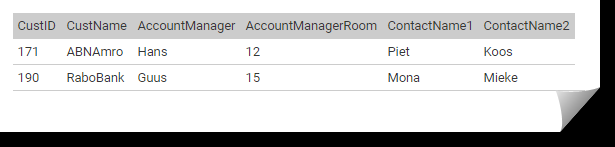
**RDBMS Assignment**

**Ans1:**

****

As we can see that CustId can be used to determine the various attributes in the table.

But we can see that a Transitive Dependency in the table. CustId uniquely identify Account Manager , and Account Manager can also be used to identify Account Manager Room

So one can see A->B and B->C , so we can see that A->C (Transitive Dependency)

So we can remove this by creating new table

We can create two tables from this

1.

|  |  |  |  |
| --- | --- | --- | --- |
| CustId | CustName | ContactName1 | ContactName2 |

2.

|  |  |
| --- | --- |
| AccountManager | AccountManagerRoom |

**Ans2:**

My ER Diagram consist of various entities, which are as follows:

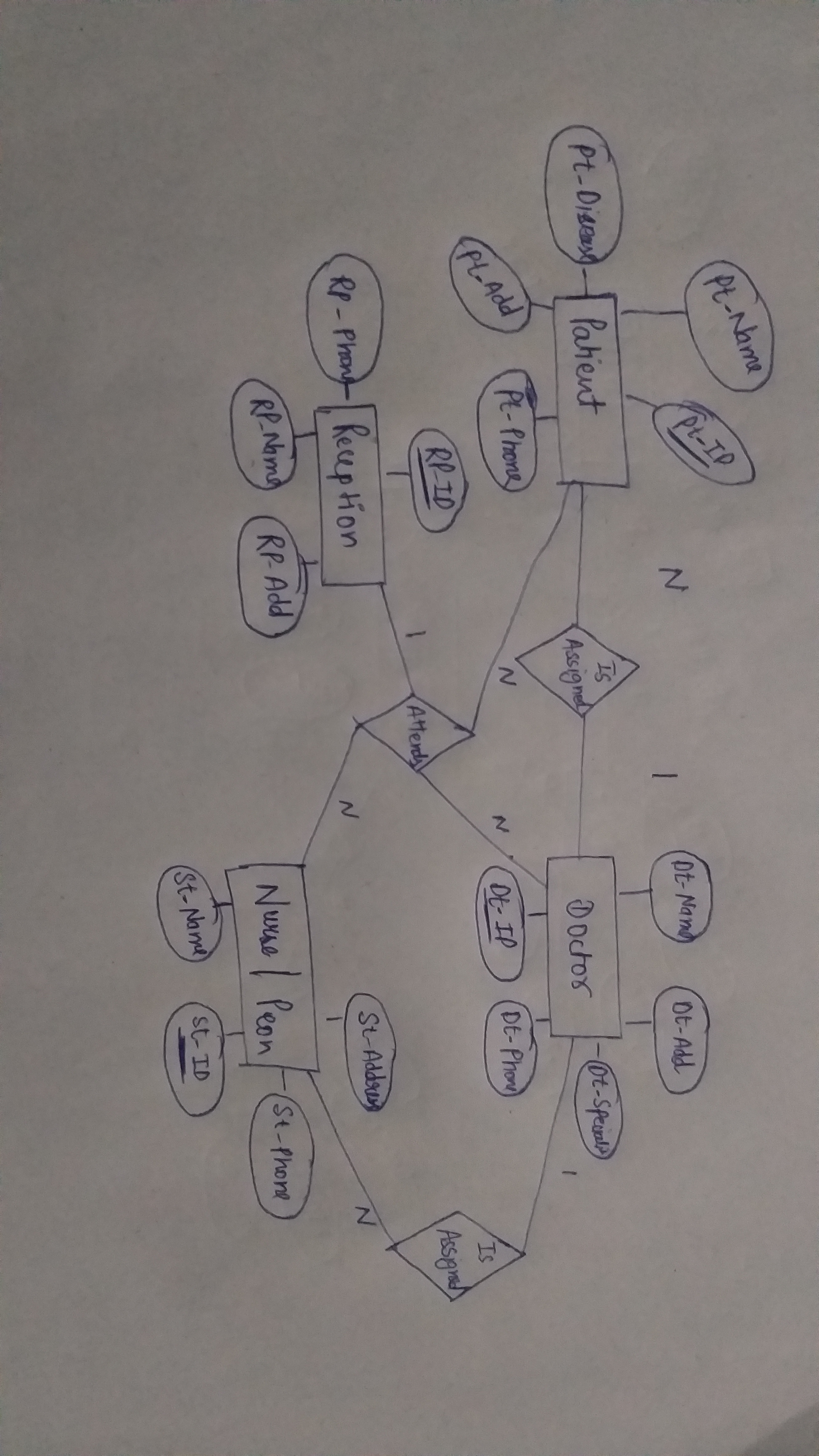
1. Patient

2. Doctor

3. Reception

4. Nurse/peon

and the diagram looks as follows:

****

**Ans3:**

As we can see there are many dependecies in the relation described .

StudentId->StudentName

StudentId->StudentAddress

ModuleId->ModuleName

TutorId->TutorName

StudentId->ModuleId

ModuleId->StudentId

So to normalize this into 3NF we need to distribute the reation into 5 relations.

StudetnId and ModuleId will be the primary keys.

R1: StudentId,ModuleId.

R2: StudentId,StudentName,StudentAddress.

R3: StudentId,TutorId.

R4: TutorId,TutorName.

R5: ModuleId,ModuleName.

**Breaking the table into 5 relations will convert the relation into 3NF.**