# Project Phase 3 - Team 51

## **1.** ER Diagram to Relational Model:-

a. **Mapping of Regular Entity Types**: For each strong entity type E in the ER schema, we created a relation R that includes all the simple attributes of E and the composite attributes were converted into simple component attributes like for example in doctor entity ,the composite attribute address was broken down into the simple component attributes like zipcode, state, country , street etc.

b. **Mapping of Weak Entity Types**:We did the same as point (a) for weak entities. Also we made sure to include the foreign key attributes of weak entities, that is the primary key attributes of the relations that correspond to the owner entity types. This took care of mapping the identifying relationship type of W. The primary key of a weak entity is the combination of the primary keys of the owners and the partial key of the weak entity.

c. **Mapping of Binary 1:1 Relationship Types:** We took the approach of foreign keys. For a relationship R and its participating entity types S and T, choose one of the relations—S, say—and include as a foreign key in S the primary key of T. For example, in Invoice we added the foreign key of Medical_Record as they have a 1:1 relationship.

d. **Mapping of Binary 1:N Relationship Types:** We did the same as in point(c). Just we added the foreign key of the relation whose entity type was on 1-side in the relation whose entity type was on the N-side. For example, in

"WORKS_IN" relationship we added primary key of Department as foreign key in Doctor's relation.

e. **Mapping of Binary M:N Relationship Types:** For each binary M:N relationship type R,we made a new relation S to represent R and included as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S. For example, we made a relation of the relationship "MONITORS" whose participants were Nurse and Patient. In the relation we included Patient_Id and Nurse_Id as foreign key and both together are primary key.

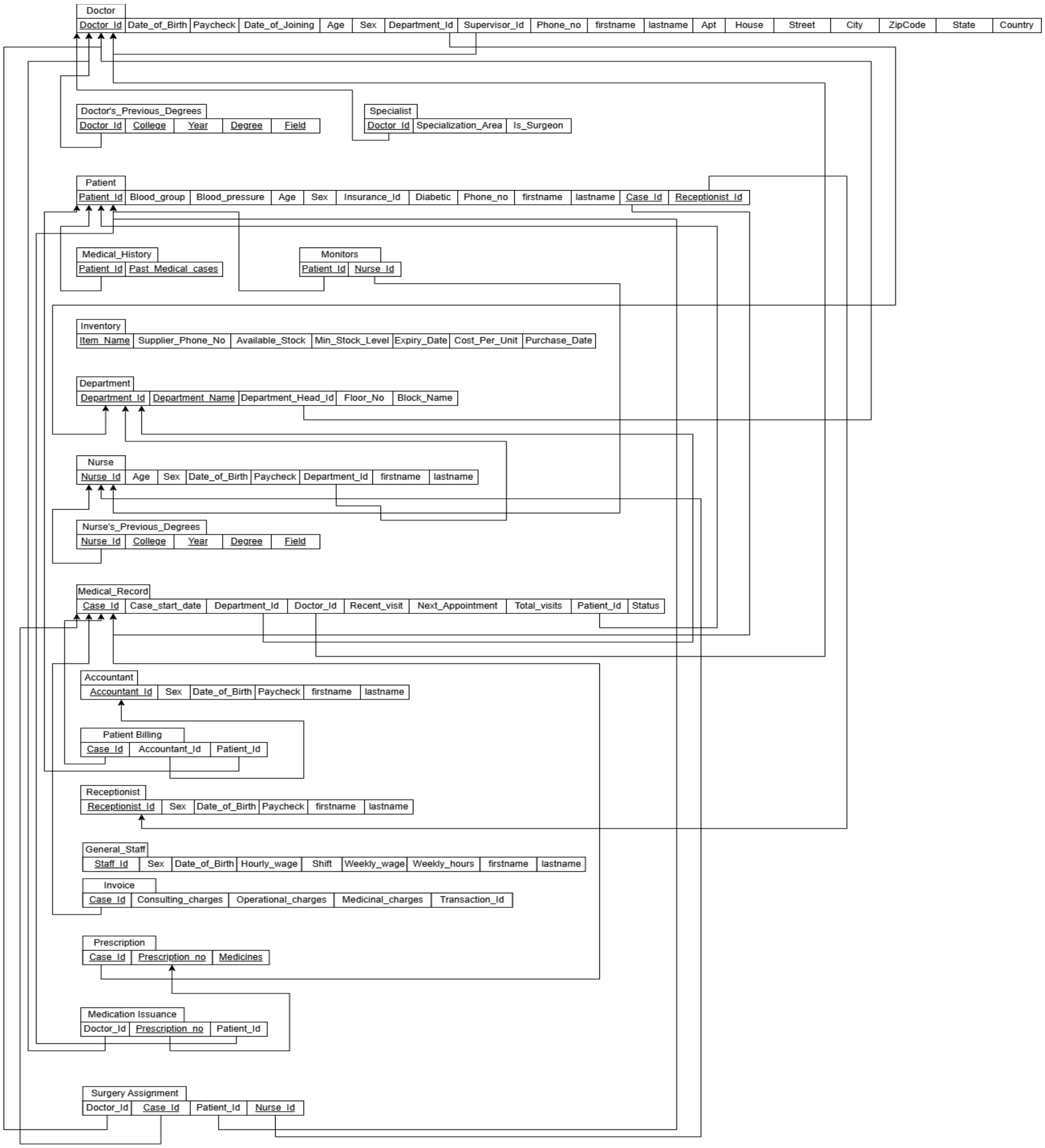
f. **Mapping of Multivalued Attributes:** For each multivalued attribute A, we made a new relation R. This relation R will include an attribute corresponding to A, plus the primary key attribute K—as a foreign key in R—of the relation that represents the entity type or relationship type that has A as a multivalued attribute. The primary key of R is the combination of A and K. For example, we had "Past_Medical_Cases" in Patient and "Previous_Degrees" in Nurse and Doctor.

g. **Mapping of N>2 Relationship Types:** For each n>2 relationship type R, we made a new relationship relation S to represent R. We made sure to include as foreign key attributes in S the primary keys of the relations that represent the participating entity types and we also included simple attributes of the n>2 relationship type as attributes of S. The primary key of S was chosen to be a combination of all the foreign keys that reference the relations representing the participating entity types whose cardinality constraints on any of the entity types E participating in R is not 1. For example, we had "Medical Issuance" and "Surgery

Assignment", which are n>2 degree relationships for which we made separate relation using the given steps.

h. **Mapping of Subclass:** For this we made a new relation R and made sure it includes the primary key of the main type . In our case the primary key of the subclass turned out to be the same as that of the main relation. We had a subclass "Specialist" for which we made a separate relation.

## The Relational Model Turned Out to be:



Doctor

| Doctor_Id | Date_of_Birth | Paycheck | Date_of_Joining | Age | Sex | Department_Id | Supervisor_Id | Phone_no | firstname | lastname | Apt | House | Street | City | ZipCode | State | Country |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Doctor's_Previous_Degrees

| Doctor_Id | College | Year | Degree | Field |
|---|---|---|---|---|

Specialist

| Doctor_Id | Specialization_Area | Is_Surgeon |
|---|---|---|

Patient

| Patient_Id | Blood_group | Blood_pressure | Age | Sex | Insurance_Id | Diabetic | Phone_no | firstname | lastname | Case_Id | Receptionist_Id |
|---|---|---|---|---|---|---|---|---|---|---|---|

Medical_History

| Patient_Id | Past_Medical_cases |
|---|---|

Monitors

| Patient_Id | Nurse_Id |
|---|---|

Inventory

| Item_Name | Supplier_Phone_No | Available_Stock | Min_Stock_Level | Expiry_Date | Cost_Per_Unit | Purchase_Date |
|---|---|---|---|---|---|---|

Department

| Department_Id | Department_Name | Department_Head_Id | Floor_No | Block_Name |
|---|---|---|---|---|

Nurse

| Nurse_Id | Age | Sex | Date_of_Birth | Paycheck | Department_Id | firstname | lastname |
|---|---|---|---|---|---|---|---|

Nurse's_Previous_Degrees

| Nurse_Id | College | Year | Degree | Field |
|---|---|---|---|---|

Medical_Record

| Case_Id | Case_start_date | Department_Id | Doctor_Id | Recent_visit | Next_Appointment | Total_visits | Patient_Id | Status |
|---|---|---|---|---|---|---|---|---|

Accountant

| Accountant_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname |
|---|---|---|---|---|---|

Patient Billing

| Case_Id | Accountant_Id | Patient_Id |
|---|---|---|

Receptionist

| Receptionist_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname |
|---|---|---|---|---|---|

General_Staff

| Staff_Id | Sex | Date_of_Birth | Hourly_wage | Shift | Weekly_wage | Weekly_hours | firstname | lastname |
|---|---|---|---|---|---|---|---|---|

Invoice

| Case_Id | Consulting_charges | Operational_charges | Medicinal_charges | Transaction_Id |
|---|---|---|---|---|

Prescription

| Case_Id | Prescription_no | Medicines |
|---|---|---|

Medication Issuance

| Doctor_Id | Prescription_no | Patient_Id |
|---|---|---|

Surgery Assignment

| Doctor_Id | Case_Id | Patient_Id | Nurse_Id |
|---|---|---|---|

## **2.** Converting Relational Model to 1NF:

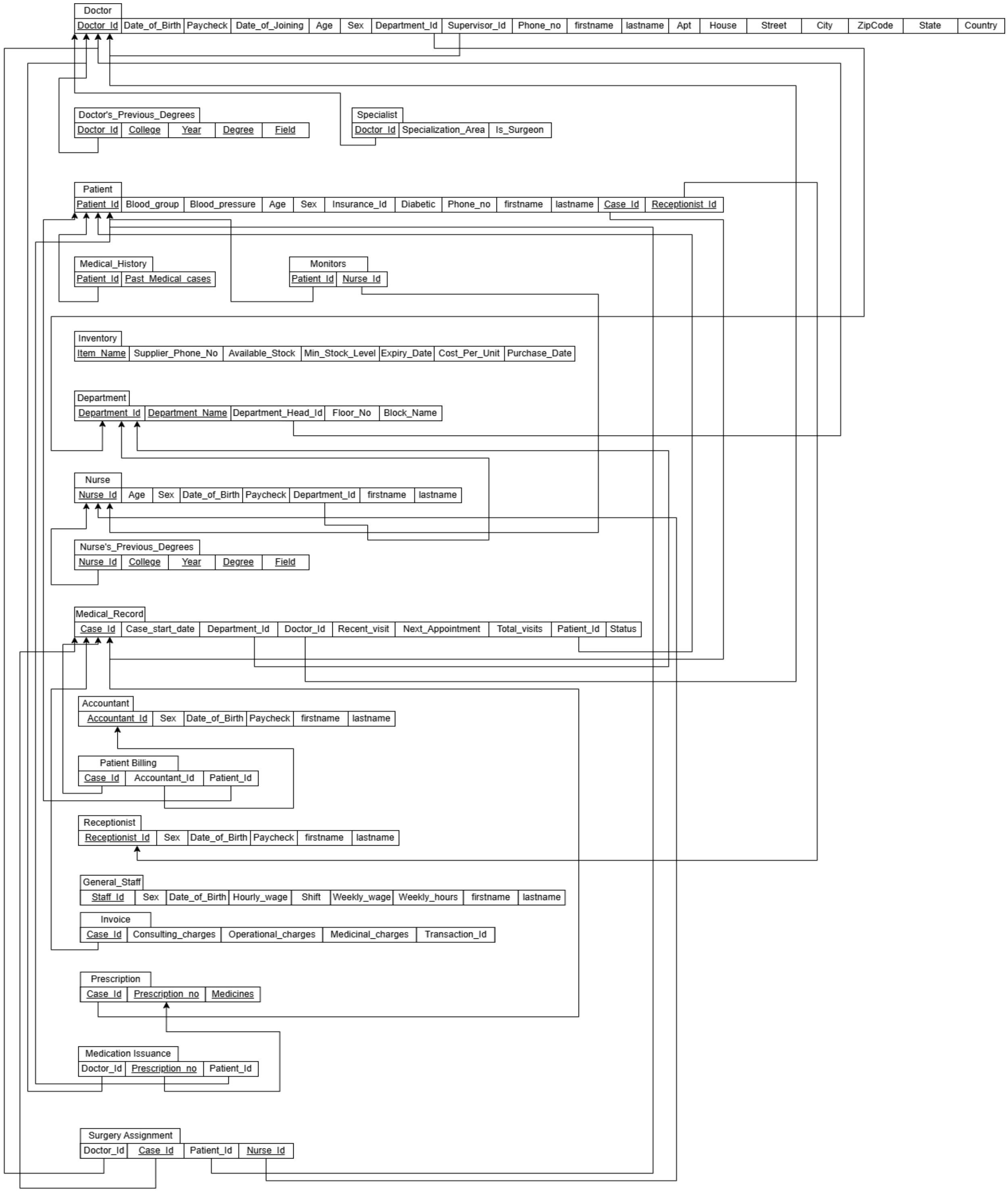
a. This form does not allow Multivalued attributes , Composite attributes and nested relations.

b. We already took care of Multivalued attributes by making a new table for them in order to reduce redundancy and we also converted composite attributes to simple component attributes. So both of these were taken care of in the relational model itself.

c. The relation generated in our relational model did not have any nested relations.

d. As a result the 1NF form turned out to be the same as the Relational Model.

## The 1NF form turned out to be:


Doctor
Doctor_Id | Date_of_Birth | Paycheck | Date_of_Joining | Age | Sex | Department_Id | Supervisor_Id | Phone_no | firstname | lastname | Apt | House | Street | City | ZipCode | State | Country
Doctor's_Previous_Degrees
Doctor_Id | College | Year | Degree | Field
Specialist
Doctor_Id | Specialization_Area | Is_Surgeon
Patient
Patient_Id | Blood_group | Blood_pressure | Age | Sex | Insurance_Id | Diabetic | Phone_no | firstname | lastname | Case_Id | Receptionist_Id
Medical_History
Patient_Id | Past_Medical_cases
Monitors
Patient_Id | Nurse_Id
Inventory
Item_Name | Supplier_Phone_No | Available_Stock | Min_Stock_Level | Expiry_Date | Cost_Per_Unit | Purchase_Date
Department
Department_Id | Department_Name | Department_Head_Id | Floor_No | Block_Name
Nurse
Nurse_Id | Age | Sex | Date_of_Birth | Paycheck | Department_Id | firstname | lastname
Nurse's_Previous_Degrees
Nurse_Id | College | Year | Degree | Field
Medical_Record
Case_Id | Case_start_date | Department_Id | Doctor_Id | Recent_visit | Next_Appointment | Total_visits | Patient_Id | Status
Accountant
Accountant_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname
Patient Billing
Case_Id | Accountant_Id | Patient_Id
Receptionist
Receptionist_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname
General_Staff
Staff_Id | Sex | Date_of_Birth | Hourly_wage | Shift | Weekly_wage | Weekly_hours | firstname | lastname
Invoice
Case_Id | Consulting_charges | Operational_charges | Medicinal_charges | Transaction_Id
Prescription
Case_Id | Prescription_no | Medicines
Medication Issuance
Doctor_Id | Prescription_no | Patient_Id
Surgery Assignment
Doctor_Id | Case_Id | Patient_Id | Nurse_Id

## **3.** Converting 1NF to 2NF:

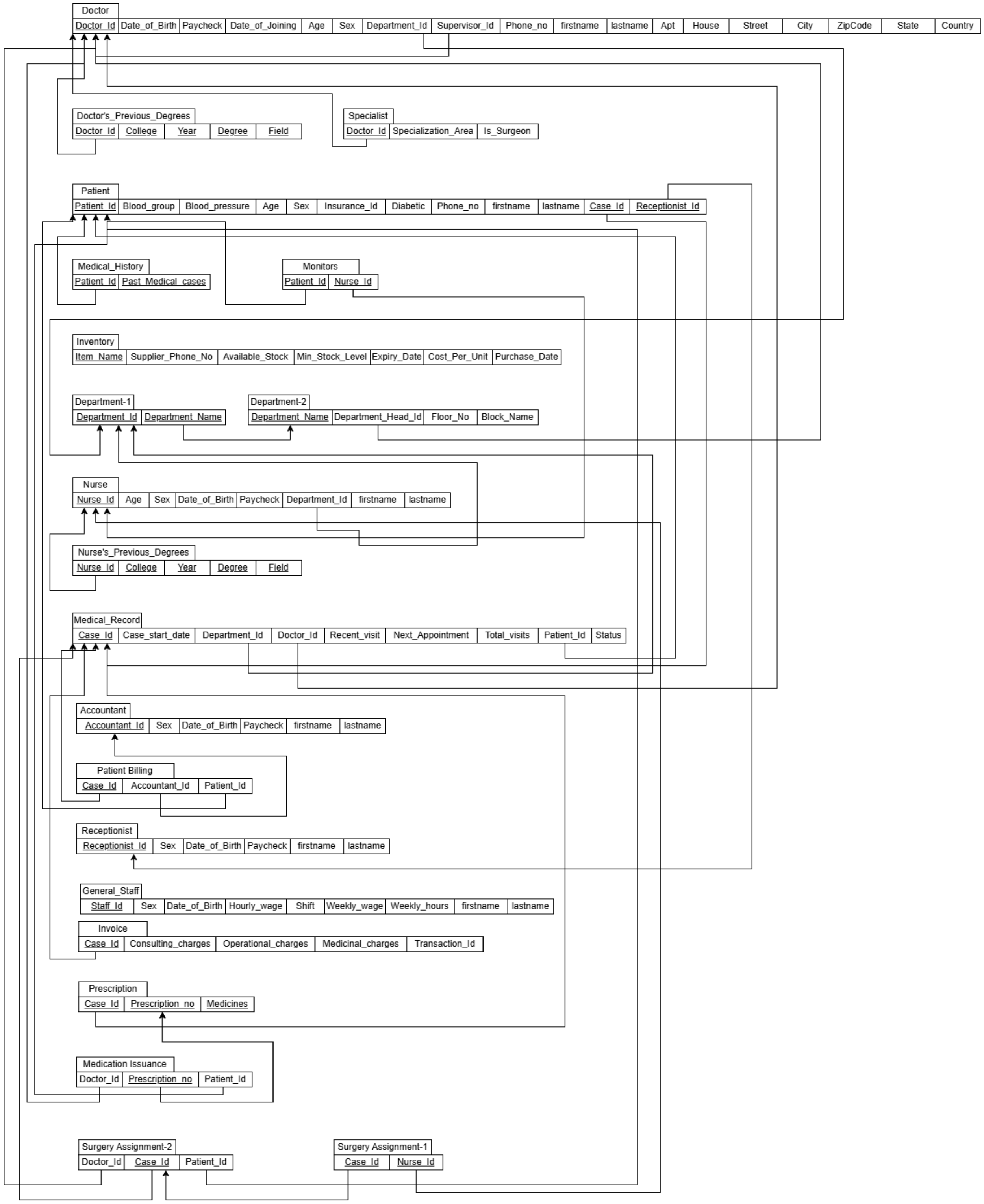
a. In this form we needed to ensure that in every relation R , every non prime attribute A in R was fully functionally dependent on the primary key K.

b. We checked each and every relation in the 1NF form , and checked if there was any partial dependency.

c. In “Surgery Assignment” relation: {Case_Id, Nurse_Id} -> Doctor_Id and {Case_Id, Nurse_Id} -> Patient_Id. But in this Doctor_Id and Patient_Id both depend on Case_Id. Hence, a partial dependency exists. So we made a new relation “Surgery Assignment-1” with attributes Case_Id and Nurse_Id and both are primary keys and removed Nurse_Id from the “Surgery Assignment” relation. We renamed “Surgery Assignment” to “Surgery Assignment-2”. Case_Id in “Surgery Assignment-1” is foreign key referencing Case_Id of “Surgery Assignment-2” and Nurse_Id in “Surgery Assignment-1” is referencing Nurse_Id of Nurse relation.

d. We did the same as above for the Department relation.

## The 2NF form turned out to be:


Doctor
Doctor_Id | Date_of_Birth | Paycheck | Date_of_Joining | Age | Sex | Department_Id | Supervisor_Id | Phone_no | firstname | lastname | Apt | House | Street | City | ZipCode | State | Country
Doctor's_Previous_Degrees
Doctor_Id | College | Year | Degree | Field
Specialist
Doctor_Id | Specialization_Area | Is_Surgeon
Patient
Patient_Id | Blood_group | Blood_pressure | Age | Sex | Insurance_Id | Diabetic | Phone_no | firstname | lastname | Case_Id | Receptionist_Id
Medical_History
Patient_Id | Past_Medical_cases
Monitors
Patient_Id | Nurse_Id
Inventory
Item_Name | Supplier_Phone_No | Available_Stock | Min_Stock_Level | Expiry_Date | Cost_Per_Unit | Purchase_Date
Department-1
Department_Id | Department_Name
Department-2
Department_Name | Department_Head_Id | Floor_No | Block_Name
Nurse
Nurse_Id | Age | Sex | Date_of_Birth | Paycheck | Department_Id | firstname | lastname
Nurse's_Previous_Degrees
Nurse_Id | College | Year | Degree | Field
Medical_Record
Case_Id | Case_start_date | Department_Id | Doctor_Id | Recent_visit | Next_Appointment | Total_visits | Patient_Id | Status
Accountant
Accountant_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname
Patient Billing
Case_Id | Accountant_Id | Patient_Id
Receptionist
Receptionist_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname
General_Staff
Staff_Id | Sex | Date_of_Birth | Hourly_wage | Shift | Weekly_wage | Weekly_hours | firstname | lastname
Invoice
Case_Id | Consulting_charges | Operational_charges | Medicinal_charges | Transaction_Id
Prescription
Case_Id | Prescription_no | Medicines
Medication Issuance
Doctor_Id | Prescription_no | Patient_Id
Surgery Assignment-2
Doctor_Id | Case_Id | Patient_Id
Surgery Assignment-1
Case_Id | Nurse_Id

## **4.** Converting 2NF to 3NF:

a. This form does not allow transitive dependency that is X->Y in R is transitive dependency, if there exists a set of attributes Z in R that is neither a candidate key nor a subset of any key in R and both X->Z and Z->Y holds.

b. We checked each and every relation in 2NF form and found out some transitive dependency in Doctor, Nurse, Medical_Record, General_Staff and Invoice.

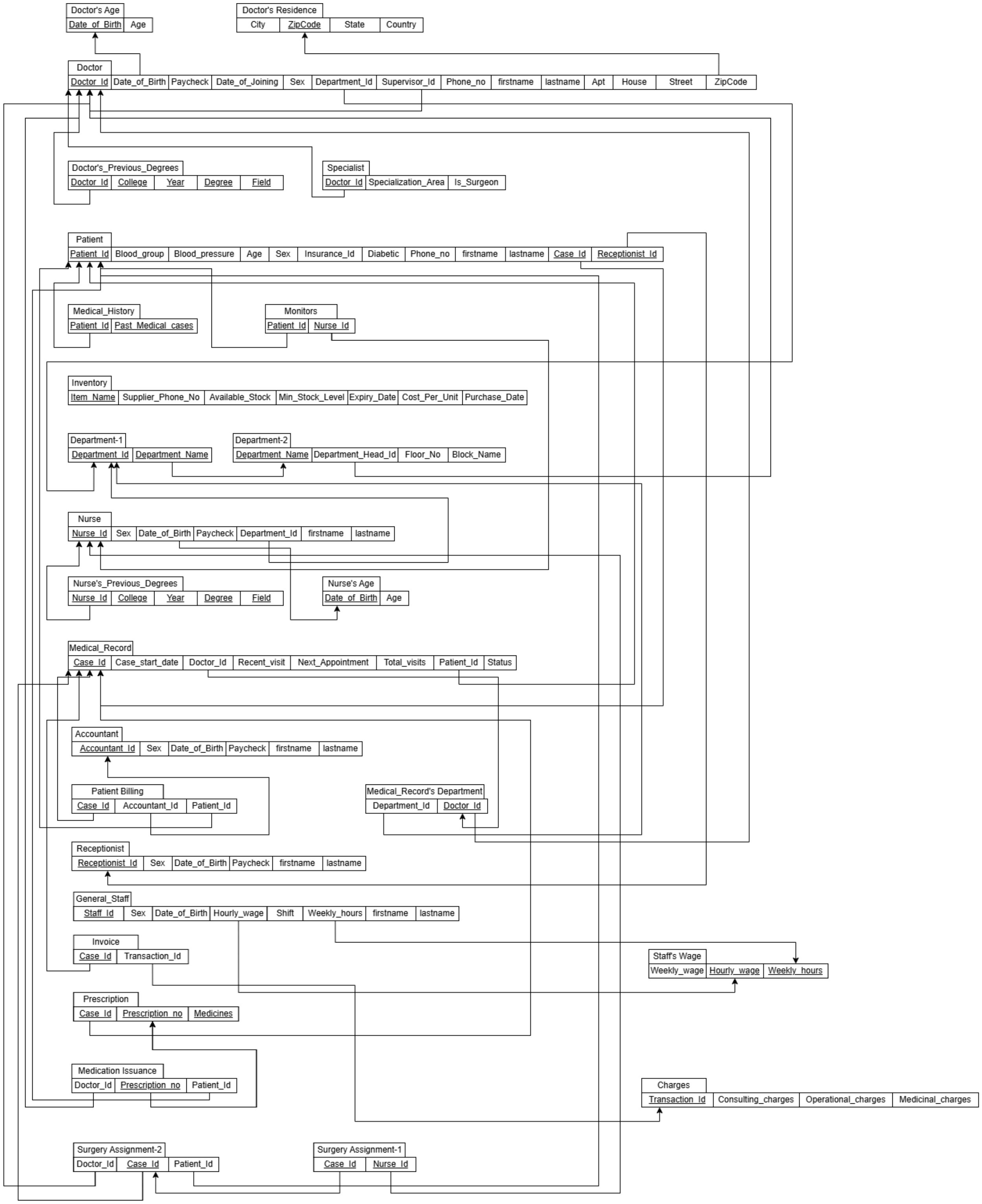
c. In relation Doctor we found that Doctor_Id -> Zip Code and Zip code -> City, State, Country and Doctor_Id -> City, State, Country. So we made a new relation in which we added City, ZipCode, Country and State where ZipCode is the primary key of it.

d. In relation Doctor we have Doctor_Id -> Date_of_Birth and Date_of_Birth -> Age and Doctor_Id -> Age. So there is transitive dependency. So we made another relation for this too. In relation Nurse too there was Date_of_Birth and Age, so we made another relation for it too.

e. In relation General_Staff we have Staff_Id -> {Hourly_wage, Weekly_hours} and {Hourly_wage, Weekly_hours} -> Weekly_wage and Staff_Id -> Weekly_wage. So for this transitive dependency we made a new relation with attributes Weekly_wage, Hourly_wage and Weekly_hours with Hourly_wage and Weekly_hours as primary keys.

f. We had transitive dependency in Invoice relation too which we solved by making a new relation similarly as above.

## The 3NF form turned out to be:



Doctor's Age
Date_of_Birth | Age

Doctor's Residence
City | ZipCode | State | Country

Doctor
Doctor_Id | Date_of_Birth | Paycheck | Date_of_Joining | Sex | Department_Id | Supervisor_Id | Phone_no | firstname | lastname | Apt | House | Street | ZipCode

Doctor's_Previous_Degrees
Doctor_Id | College | Year | Degree | Field

Specialist
Doctor_Id | Specialization_Area | Is_Surgeon

Patient
Patient_Id | Blood_group | Blood_pressure | Age | Sex | Insurance_Id | Diabetic | Phone_no | firstname | lastname | Case_Id | Receptionist_Id

Medical_History
Patient_Id | Past_Medical_cases

Monitors
Patient_Id | Nurse_Id

Inventory
Item_Name | Supplier_Phone_No | Available_Stock | Min_Stock_Level | Expiry_Date | Cost_Per_Unit | Purchase_Date

Department-1
Department_Id | Department_Name

Department-2
Department_Name | Department_Head_Id | Floor_No | Block_Name

Nurse
Nurse_Id | Sex | Date_of_Birth | Paycheck | Department_Id | firstname | lastname

Nurse's_Previous_Degrees
Nurse_Id | College | Year | Degree | Field

Nurse's Age
Date_of_Birth | Age

Medical_Record
Case_Id | Case_start_date | Doctor_Id | Recent_visit | Next_Appointment | Total_visits | Patient_Id | Status

Accountant
Accountant_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname

Patient Billing
Case_Id | Accountant_Id | Patient_Id

Medical_Record's Department
Department_Id | Doctor_Id

Receptionist
Receptionist_Id | Sex | Date_of_Birth | Paycheck | firstname | lastname

General_Staff
Staff_Id | Sex | Date_of_Birth | Hourly_wage | Shift | Weekly_hours | firstname | lastname

Invoice
Case_Id | Transaction_Id

Staff's Wage
Weekly_wage | Hourly_wage | Weekly_hours

Prescription
Case_Id | Prescription_no | Medicines

Medication Issuance
Doctor_Id | Prescription_no | Patient_Id

Charges
Transaction_Id | Consulting_charges | Operational_charges | Medicinal_charges

Surgery Assignment-2
Doctor_Id | Case_Id | Patient_Id

Surgery Assignment-1
Case_Id | Nurse_Id