BestHealthCare Management System

Lakkappa, Ravichandra

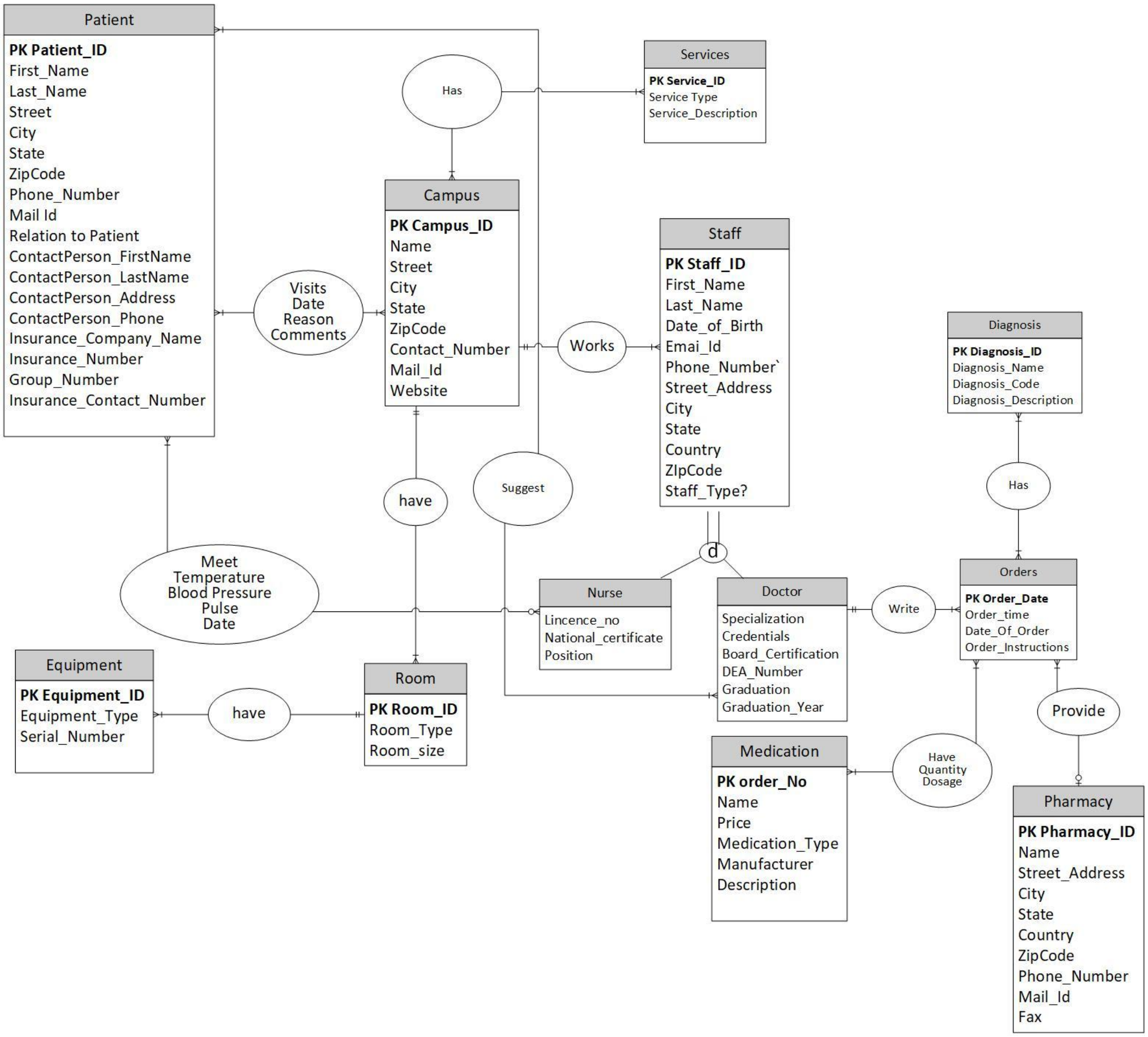
01/26/2025

Professor Earl Charles

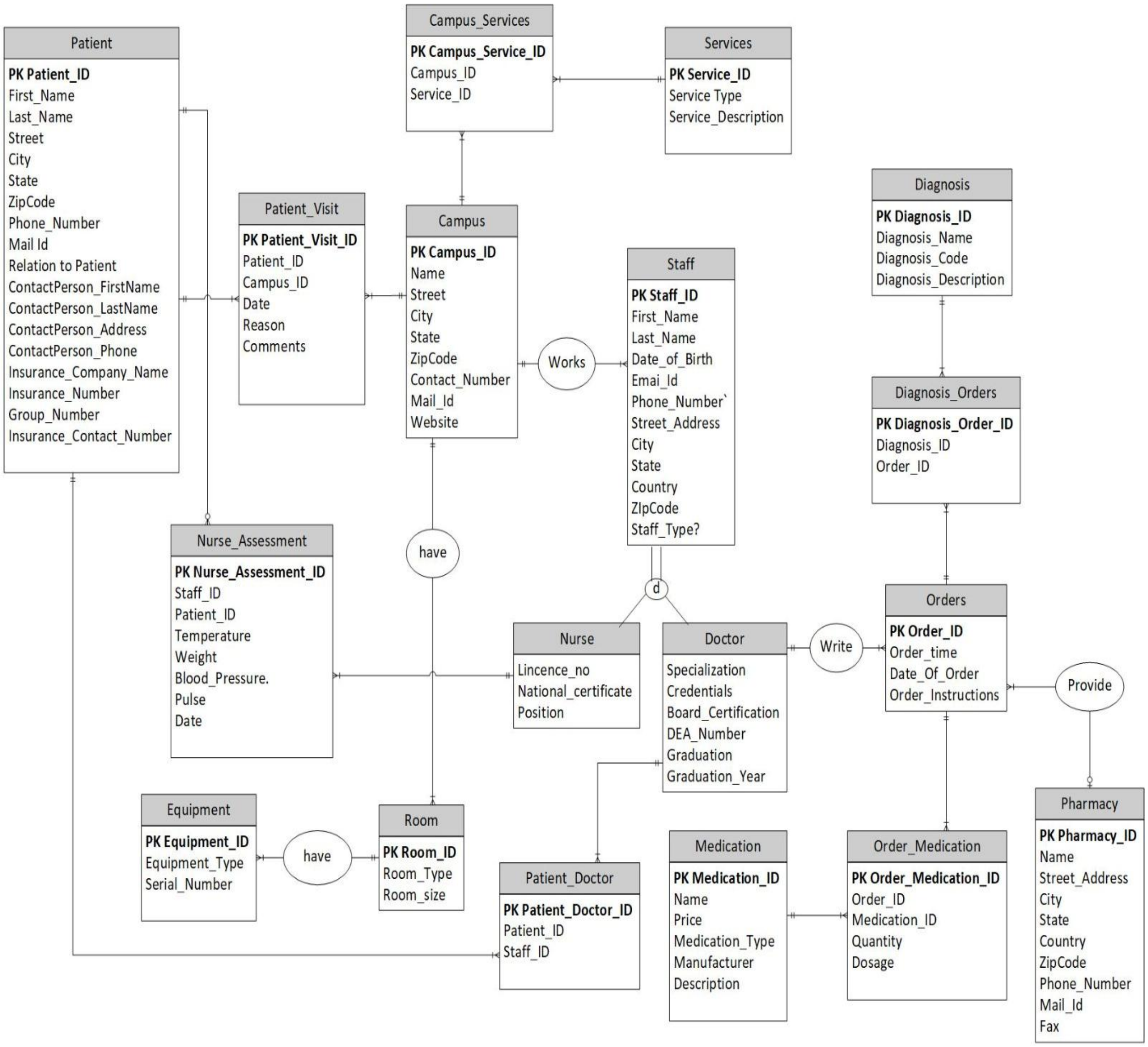
INDEX

Table of Contents

ERD #1


Patient
PK Patient_ID
First_Name
Last_Name
Street
City
State
ZipCode
Phone_Number
Mail Id
Relation to Patient
ContactPerson_FirstName
ContactPerson_LastName
ContactPerson_Address
ContactPerson_Phone
Insurance_Company_Name
Insurance_Number
Group_Number
Insurance_Contact_Number
Has
Services
PK Service_ID
Service Type
Service_Description
Campus
PK Campus_ID
Name
Street
City
State
ZipCode
Contact_Number
Mail_Id
Website
Visits
Date
Reason
Comments
Works
Staff
PK Staff_ID
First_Name
Last_Name
Date_of_Birth
Emai_Id
Phone_Number`
Street_Address
City
State
Country
ZIpCode
Staff_Type?
Diagnosis
PK Diagnosis_ID
Diagnosis_Name
Diagnosis_Code
Diagnosis_Description
Has
have
Suggest
d
Meet
Temperature
Blood Pressure
Pulse
Date
Nurse
Lincence_no
National_certificate
Position
Doctor
Specialization
Credentials
Board_Certification
DEA_Number
Graduation
Graduation_Year
Write
Orders
PK Order_Date
Order_time
Date_Of_Order
Order_Instructions
Equipment
PK Equipment_ID
Equipment_Type
Serial_Number
have
Room
PK Room_ID
Room_Type
Room_size
Medication
PK order_No
Name
Price
Medication_Type
Manufacturer
Description
Have
Quantity
Dosage
Provide
Pharmacy
PK Pharmacy_ID
Name
Street_Address
City
State
Country
ZipCode
Phone_Number
Mail_Id
Fax

ERD #2

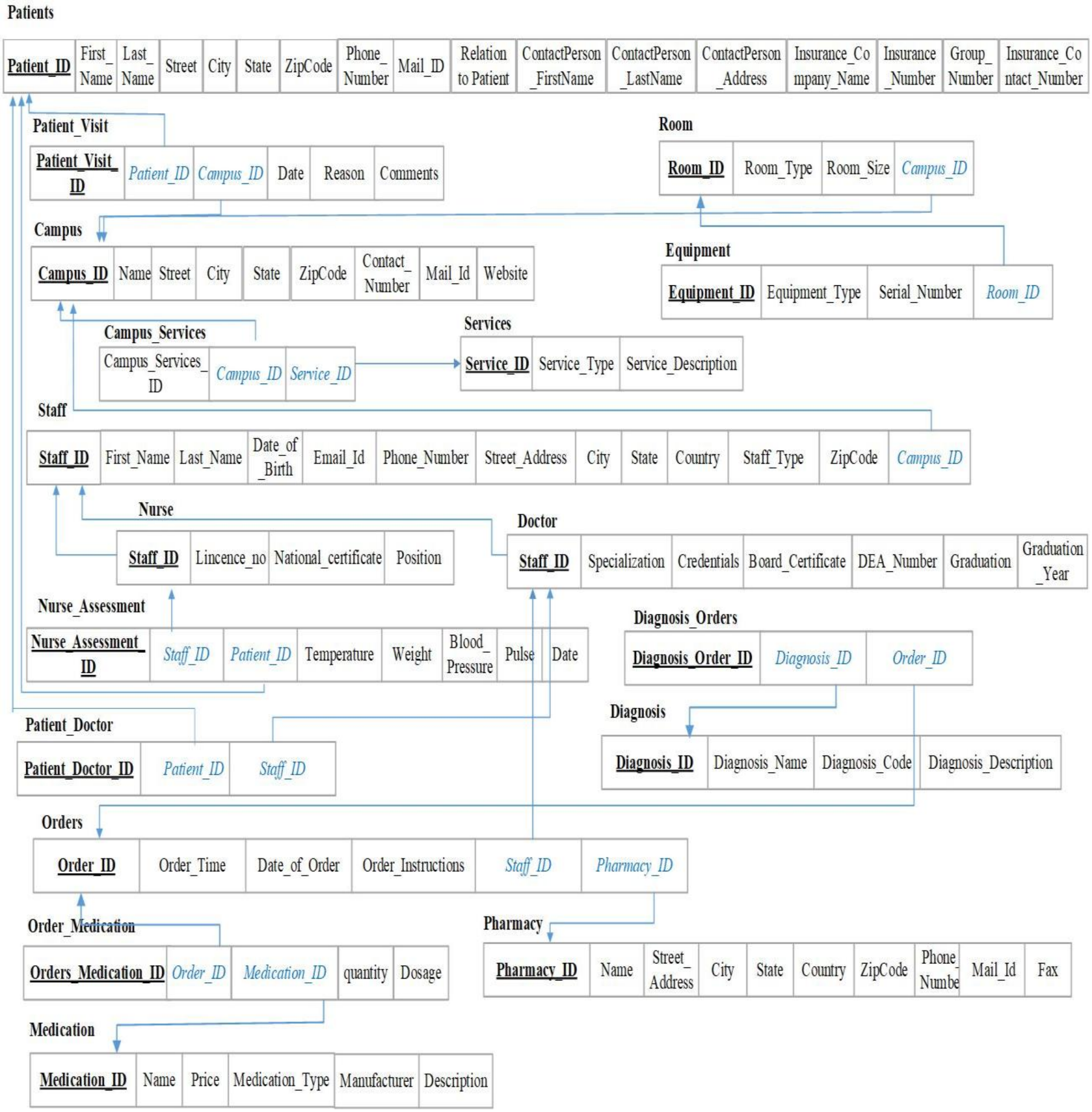

Patient
PK Patient_ID
First_Name
Last_Name
Street
City
State
ZipCode
Phone_Number
Mail Id
Relation to Patient
ContactPerson_FirstName
ContactPerson_LastName
ContactPerson_Address
ContactPerson_Phone
Insurance_Company_Name
Insurance_Number
Group_Number
Insurance_Contact_Number
Campus_Services
PK Campus_Service_ID
Campus_ID
Service_ID
Services
PK Service_ID
Service Type
Service_Description
Patient_Visit
PK Patient_Visit_ID
Patient_ID
Campus_ID
Date
Reason
Comments
Campus
PK Campus_ID
Name
Street
City
State
ZipCode
Contact_Number
Mail_Id
Website
Works
Staff
PK Staff_ID
First_Name
Last_Name
Date_of_Birth
Emai_Id
Phone_Number`
Street_Address
City
State
Country
ZIpCode
Staff_Type?
Diagnosis
PK Diagnosis_ID
Diagnosis_Name
Diagnosis_Code
Diagnosis_Description
Diagnosis_Orders
PK Diagnosis_Order_ID
Diagnosis_ID
Order_ID
have
d
Nurse_Assessment
PK Nurse_Assessment_ID
Staff_ID
Patient_ID
Temperature
Weight
Blood_Pressure.
Pulse
Date
Nurse
Lincence_no
National_certificate
Position
Doctor
Specialization
Credentials
Board_Certification
DEA_Number
Graduation
Graduation_Year
Write
Orders
PK Order_ID
Order_time
Date_Of_Order
Order_Instructions
Provide
Equipment
PK Equipment_ID
Equipment_Type
Serial_Number
have
Room
PK Room_ID
Room_Type
Room_size
Patient_Doctor
PK Patient_Doctor_ID
Patient_ID
Staff_ID
Medication
PK Medication_ID
Name
Price
Medication_Type
Manufacturer
Description
Order_Medication
PK Order_Medication_ID
Order_ID
Medication_ID
Quantity
Dosage
Pharmacy
PK Pharmacy_ID
Name
Street_Address
City
State
Country
ZipCode
Phone_Number
Mail_Id
Fax

Assumptions

- Patients can visit multiple Campuses for treatment and each campus can treat multiple patients.
- Campus can provide multiple services to patients and same service can be provided by multiple campuses.
- In a campus there will be various staff, but one staff person can work only in one campus.
- Staff has two sub types which are Nurse and Doctor.
- Nurse can meet any number of patients; each patient may or may not meet multiple nurses for assessment.
- Patient can visit multiple doctors for treatment and date, reason, and comments might vary based on each visit.
- Doctor can write many orders for patient, but each order should be written by a particular doctor.
- Each order has multiple medications, and particular medications can present in various orders. For every order quantity and dosage will vary based on the medication history.
- Orders can be taken by particular pharmacy based on the patient's choice and pharmacy can take multiple orders.
- Each order can have multiple Diagnosis for a patient and same Diagnosis can be done for many orders based on patient’s condition.
- Rooms can be present in many numbers of campuses and a particular room will be present in a campus.

- Room will have multiple equipment, but same equipment can’t present in same room multiple times.

Logical Schema



Patients

Patient_ID | First_Name | Last_Name | Street | City | State | ZipCode | Phone_Number | Mail_ID | Relation to Patient | ContactPerson_FirstName | ContactPerson_LastName | ContactPerson_Address | Insurance_Company_Name | Insurance_Number | Group_Number | Insurance_Contact_Number

Patient_Visit

Patient_Visit_ID | Patient_ID | Campus_ID | Date | Reason | Comments

Room

Room_ID | Room_Type | Room_Size | Campus_ID

Campus

Campus_ID | Name | Street | City | State | ZipCode | Contact_Number | Mail_Id | Website

Equipment

Equipment_ID | Equipment_Type | Serial_Number | Room_ID

Campus_Services

Campus_Services_ID | Campus_ID | Service_ID

Services

Service_ID | Service_Type | Service_Description

Staff

Staff_ID | First_Name | Last_Name | Date_of_Birth | Email_Id | Phone_Number | Street_Address | City | State | Country | Staff_Type | ZipCode | Campus_ID

Nurse

Staff_ID | Lincence_no | National_certificate | Position

Doctor

Staff_ID | Specialization | Credentials | Board_Certificate | DEA_Number | Graduation | Graduation_Year

Nurse_Assessment

Nurse_Assessment_ID | Staff_ID | Patient_ID | Temperature | Weight | Blood_Pressure | Pulse | Date

Diagnosis_Orders

Diagnosis_Order_ID | Diagnosis_ID | Order_ID

Diagnosis

Diagnosis_ID | Diagnosis_Name | Diagnosis_Code | Diagnosis_Description

Patient_Doctor

Patient_Doctor_ID | Patient_ID | Staff_ID

Orders

Order_ID | Order_Time | Date_of_Order | Order_Instructions | Staff_ID | Pharmacy_ID

Order_Medication

Orders_Medication_ID | Order_ID | Medication_ID | quantity | Dosage

Pharmacy

Pharmacy_ID | Name | Street_Address | City | State | Country | ZipCode | Phone_Numbe | Mail_Id | Fax

Medication

Medication_ID | Name | Price | Medication_Type | Manufacturer | Description

Research on SQL Server datatypes

DATA TYPE	DESCRIPTION	WHEN TO USE
Numeric		
bigint	When integer values might exceed the range that is supported by the int data type this must be used	For Integer Values *Range*: -2^63 to 2^63-1 *Storage*: 8 Bytes
bit	An integer data type that can hold a value from 1 to 64.	For values 1, 0, or null *Storage*: 2 Bytes
decimal	Numeric data types that have fixed precision and scale.	For Numeric data *Range*: - 10^38 +1 through 10^38 -1. (Maximum precision) *Storage:* 5-17 Bytes
int	Integer data type	For Integer Values *Range*: -2^31 to 2^31-1 *Storage*: 4 Bytes
money	Data Type for Monetary Units.	Currency /Monetary Data

		Range: -922,337,203,685,477.5808 to 922,337,203,685,477.5807 *Storage*: 8 Bytes
numeric	Measurable data type.	For Numeric data *Range*: - 10^38 +1 through 10^38 - 1. (Maximum precision) *Storage*: 5-17 Bytes
smallint	Data type that stores small whole numbers.	For Integer Values *Range*: -2^15 to 2^15-1 *Storage*: 2 Bytes
smallmoney	Data Type for Monetary Units. Accurate to a ten-thousandth of the monetary units that they represent.	Currency /Monetary Data *Range*: - 214,748.3648 to 214,748.3647 *Storage*: 4 Bytes
tinyint	The smallest data type for integer values	For Integer Values *Range*: 0 to 255 *Storage*: 1 Byte

float	Approximate-number data types for use with floating point numeric data.	For Approximate Numbers *Range*: '1.79E+308 to -2.23E-308, 0 and 2.23E-308 to 1.79E+308 *Storage*: 4-8 Bytes
real	Approximate-number data types for use with floating point numeric data.	For Approximate Numbers *Range*: 3.40E + 38 to -1.18E - 38, 0 and 1.18E - 38 to 3.40E + 38 *Storage*: 4 Bytes
Character Strings		
char	Character data types that are of fixed size.	For uniform character entries *Range*: 1 through 8,000 *Storage*: Table specific
varchar	Character data types that are of variable size.	For non-uniform character entries *Range*: 1 through 8,000 *Storage*: 2 Bytes + Data size
text	It is used for Variable-length non-Unicode data.	For binary string

		Range: Maximum string length of 2^31-1 *Storage*: Data Size
Unicode Characters Strings		
nchar	Character data types that are of fixed size	For fixed-size strings *Range*: 1 through 4,000 *Storage*: 2 * n bytes
nvarchar	Character data types that are of variable-size	For variable-size strings *Range*: 1 through 4,000 *Storage*: 2 bytes + 2 * n bytes
ntext	Variable-length data types for storing large Unicode data.	For string data *Range*: Maximum string length of 2^30 - 1 *Storage*: 2 * Stringlength
Binary Strings		
binary	Binary data type for fixed length Binary data.	For Binary values *Range*: 1 through 8,000 *Storage*: Table Specific

image	The variable-length data type for storing large binary data.	for Image files *Range*: 0 through 2^31-1 *Storage*: 2 bytes + data size
varbinary	Binary data types for Variable-length Binary data.	for images, pdf or word *Range*: 1 through 8,000 *Storage*: 2 bytes + data size
Date & Time		
date	For Date Description, Format = YYYY-MM-DD	for Date *Range*: 0001-01-01 through 9999-12-31 *Storage*: 3 bytes
datetime	Defines a date combined with a time of day with fractional seconds based on a 24-hour clock.	for Date & Time *Range*: January 1, 1753, through December 31, 9999 & 00:00:00 through 23:59:59.997 *Storage*: 8 bytes
datetime2	An extension of the existing datetime type	For Date & Time 24 hr

		Range: 0001-01-01 through 9999-12-31 & 00:00:00 through 23:59:59.9999999 *Storage*: 6-8 bytes
datetimeoffset	Defines a date that is combined with a time of a day that has time zone awareness and is based on a 24-hour clock.	Date, time 24 hrs & time zone *Range*: 0001-01-01 through 9999-12-31 & 00:00:00 through 23:59:59.9999999 & -14:00 through +14:00 *Storage*: 10 Bytes
smalldatetime	Date combined with a time of day based on a 24-hour day, with seconds always zero& without fractional seconds	Date time without fraction seconds *Range*: 1900-01-01 through 2079-06-06 & 00:00:00 through 23:59:59 *Storage*: 4 bytes
time	Defines a time of the day. The time is without time zone awareness and is based on a 24-hour clock.	Time of the day *Range*: 00:00:00.0000000 through 23:59:59.9999999 *Storage*: 5 bytes

Other Data Types		
cursor	A data type for variables or stored procedure OUTPUT parameters containing a cursor reference.	for Cursor reference
hierarchyrid	Variable length, a system data type that represents a position in a tree hierarchy.	For the value of Position in the Hierarchy *Storage*: 5 bytes
sql_variant	A data type that stores values of various SQL Server-supported data types.	For the Value of Objects of other data types *Storage*: 8016 bytes
uniqueidentifier	It is a globally unique identifier.	For a column or local variable *Storage*: 16 Bytes
xml	This Data type stores XML data.	For well-formed XML fragments. *Storage*: 2 GB Max
Spatial Geometry types	This data type represents data in a Euclidean (flat) coordinate system.	For Geometry methods
table	Special data type used to store a result set for processing later	For Functions and variables

Spatial Geography types	Data type represents data in a round-earth coordinate system.	For Co-ordinates
rowversion	This data type exposes automatically generated, unique binary numbers within a database	For the value of Incrementing number *Storage*: 8 Bytes

Data Dictionary

Column Name	**Data Type (Size)**	**Constraints**
Patients		
Patient ID	Int	Primary key for PATIENT table
First_Name	Varchar (50)	
Last_Name	Varchar (50)	
Date_of_Birth	Date	
Street	Varchar (50)	
City	Varchar (50)	
State	Varchar (50)	
ZipCode	Int	
Phone_Number	Int	
Mail_ID	Varchar (50)	
Relation_to_Patient	Varchar (50)	
ContactPerson_FirstName	Varchar (50)	
ContactPerson_LastName	Varchar (50)	
ContactPerson_Address	Varchar (100)	
Insurance_Company_Name	Varchar (50)	
Insurance_Number	Int	
Group_Number	Int	

Insurance_Contact_Number	Int	
Patient_Visit		
Patient_Visit_ID	Int	Primary key for PATIENT _VISIT table.
Patient_ID	Int	Foreign key for Patients table.
Campus_ID	Int	Foreign key for Campuses table.
Date	Date	
Reason	Varchar (50)	
Comments	Varchar (50)	
Campuses		
Campus_ID	Int	Primary key for CAMPUSES table.
Name	Varchar (50)	
Street	Varchar (50)	
City	Varchar (50)	
State	Varchar (50)	
ZipCode	Int	
Contact_Number	Int	
Mail_ID	Varchar (50)	
Website	Varchar (50)	
Campus_Services		

Campus_Services_ID	Int	Primary key for CAMPUS_SERVICES table.
Campus_ID	Int	Foreign key for CAMPUS table.
Service_ID	Int	Foreign key for SERVICES table.
Services		
Service_ID	Int	Primary for key SERVICE table.
Service_Type	Varchar (50)	
Service_Description	Varchar (250)	
Staff		
Staff_ID	Int	Primary for key STAFF table.
First_Name	Varchar (50)	
Last_Name	Varchar (50)	
Date_of_Birth	Date	
Email_Id	Varchar (50)	
Phone_Number	Int	
Street	Varchar (50)	
City	Varchar (50)	
State	Varchar (50)	

Country	Varchar (50)	
Staff_Type	Varchar (15)	
ZipCode	Int	
Campus_ID	Int	Foreign key to Campuses table.
Nurses		
Staff_ID	Int	Primary key for NURSE table.
Lincence_no	Varchar (25)	
National_Certificate	Varchar (50)	
Position	Varchar (50)	
Doctors		
Staff_ID	Int	Primary key for STAFF table.
Specialization	Varchar (50)	
Credentials	Varchar (50)	
Board_Certificate	Varchar (50)	
DEA_Number	Int	
Graduation	Varchar (50)	
Graduation_Year	Int	
Nurse_Assessment		

Nurse_Assessment_ID	Int	Primary key for NURSE_ASSESSMENT table.
Staff_ID	Int	Foreign key to Nurses table.
Patient_ID	Int	Foreign key to Patients table.
Temperature	Varchar (50)	
Weight	Varchar (50)	
Blood_Pressure	Varchar (50)	
Pulse	Varchar (50)	
Date	Date	
Room		
Room_ID	Int	Primary key for ROOM table.
Room_Type	Varchar (50)	
Room_Size	nVarchar(50)	
Campus_ID	Int	Foreign key to Campuses table.
Equipment		
Equipment_ID	Int	Primary key for EQUIPMENT table.
Equipment_Type	Varchar (50)	
Serial_Number	Varchar (50)	

Room_ID	Int	Foreign key to Room table.
Patient_Doctor		
Patient_Doctor_ID	Int	Primary key for PATIENT_DOCTOR table.
Patient_ID	Int	Foreign key to PATIENT table.
Staff_ID	Int	Foreign key to Doctors table.
Orders		
Order_ID	Int	Primary key for ORDERS table.
Order_Time	Time	
Date_of_Order	Date	
Order_Instructions	Varchar (250)	
Staff_ID	Int	Foreign key to Staff table.
Pharmacy_ID	Int	Foreign key to Pharmacy table.
Order_Medication		
Orders_Medication_ID	Int	Primary key for ORDER_MEDICATION table.
Order_ID	Int	Foreign key to ORDER table.

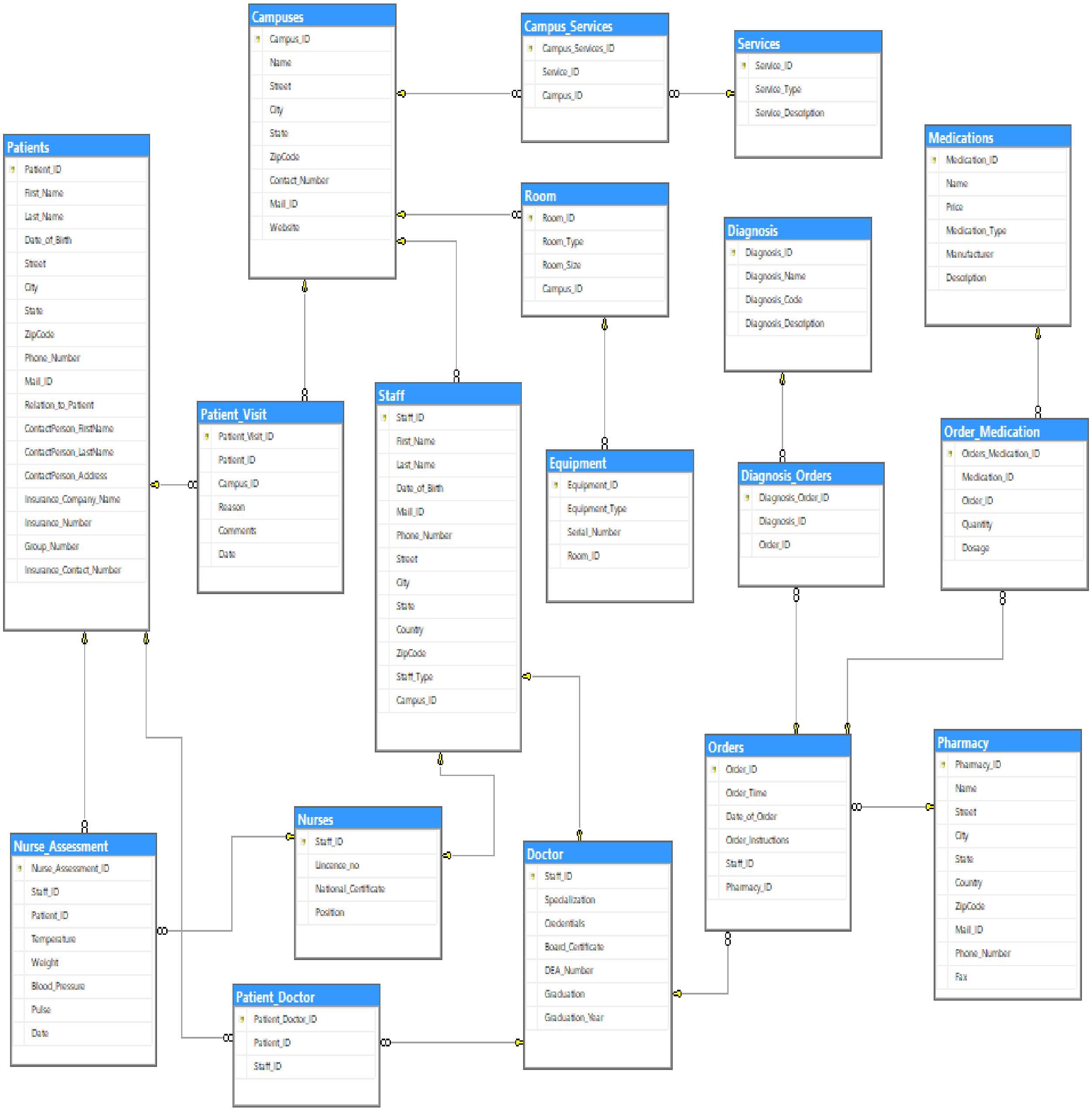
Medication_ID	Int	Foreign key to Medication table.
Quantity	Int	
Dosage	Varchar (50)	
Medications		
Medication_ID	Int	Primary key for MEDICATION table.
Name	Varchar (50)	
Price	Decimal (10,2)	
Medication_Type	Varchar (50)	
Manufacturer	Varchar (50)	
Description	Varchar (250)	
Diagnosis_Orders		
Diagnosis_Order_ID	Int	Primary key for DIAGNOSIS _ORDERS table.
Diagnosis_ID	Int	Foreign key to DIADNOSIS table.
Order_ID	Int	Foreign key to ORDERS table.
Diagnosis		
Diagnosis_ID	Int	Primary key for DIAGNOSIS table.

Diagnosis_Name	Varchar (50)	
Diagnosis_Code	Int	
Diagnosis_Description	Varchar (250)	
Pharmacy		
Pharmacy_ID	Int	Primary key for PHARMACY table.
Name	Varchar (50)	
Street	Varchar (50)	
City	Varchar (50)	
State	Varchar (50)	
Country	Varchar (50)	
ZipCode	Int	
Phone_Number	Int	
Mail_Id	Varchar (50)	
Fax	Int	

Data Normalization

Numerous tables in the BestCare electronic medical record system are used to hold various sorts of data, including information on the campus, services provided, treatment rooms, equipment, staff, patients, physician-patient interactions, nurse assessments, and medication orders. In our ERD's Entities do not have any Multivalued valued attributes in them so it meets the 1st Normalization Form. For minimizing data anomalies and ensuring data integrity, the tables are created based on their logical relationships. Each table has a unique set of columns for storing pertinent data. To guarantee that the data is in at least a third normal form, assure data consistency, and minimize data redundancy, the information is saved in different tables.

Database Schema



```
-- 1 QUERY

SELECT DISTINCT Patient_ID,First_Name,Street,City,[State],Phone_Number
FROM Patients
ORDER BY [State];

---------------------------------------------------------
-- 2 QUERY

SELECT Orders.Order_ID,Medications.Medication_ID,SUM(Order_Medication.Quantity) AS [Total_Quantity]
FROM Orders,Order_Medication,Medications
WHERE Orders.Order_ID = Order_Medication.Order_ID
AND Order_Medication.Medication_ID = Medications.Medication_ID
GROUP BY Medications.Medication_ID,Order_Medication.Quantity,Orders.Order_ID
ORDER BY Order_Medication.Quantity;
```

```
------------------------------------------------------------------------------------------------

--3 QUERY


SELECT Order_ID,Order_Time,Pharmacy.Pharmacy_ID,Pharmacy.[Name]

FROM Orders,Pharmacy

WHERE Orders.Pharmacy_ID= Pharmacy.Pharmacy_ID

ORDER BY Order_Time;


-------------------------------------------------------------------------------------------------

-- 4 QUERY


SELECT Patients.Patient_ID,Patients.First_Name, Patient_Visit.Reason,Campuses.Campus_ID

FROM Patients,Patient_Visit,Campuses

WHERE Patients.Patient_ID=Patient_Visit.Patient_ID

AND Patient_Visit.Campus_ID= Campuses.Campus_ID;


--------------------------------------------------------------------------------------------
```

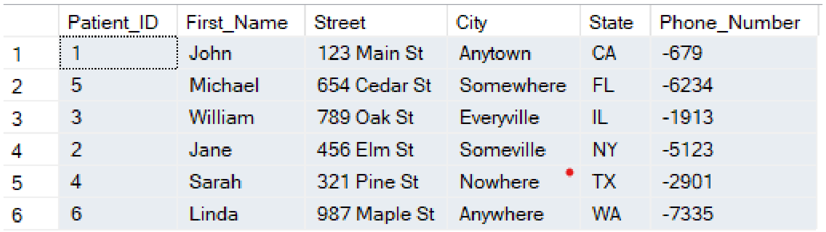
```
--5 QUERY


SELECT
Doctor.Staff_ID,Doctor.Specialization,Orders.Order_ID,Orders.Date_of_Order,Medications.[Name],Order_Medication.Quantity

FROM Doctor,Orders,Order_Medication,Medications

WHERE Doctor.Staff_ID= Orders.Staff_ID

AND Orders.Order_ID= Order_Medication.Order_ID

AND Order_Medication.Medication_ID=Medications.Medication_ID

ORDER BY Orders.Order_ID;


------------------------------------------------------------------------------------------

--6 QUERY


SELECT Patients.First_Name AS [PATIENT NAME], Patient_Visit.Reason AS [Reason Of Visit],Campuses.Campus_ID,Services.Service_Type

FROM Patients,Patient_Visit,Campuses,Campus_Services,Services

WHERE Patients.Patient_ID= Patient_Visit.Patient_ID

AND Patient_Visit.Campus_ID= Campuses.Campus_ID
```

AND Campuses.Campus_ID=Campus_Services.Campus_ID

AND Campus_Services.Service_ID=Services.Service_ID;

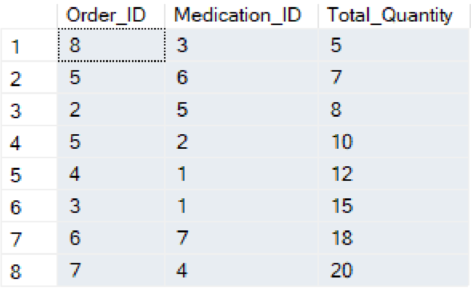
Results

Query 1



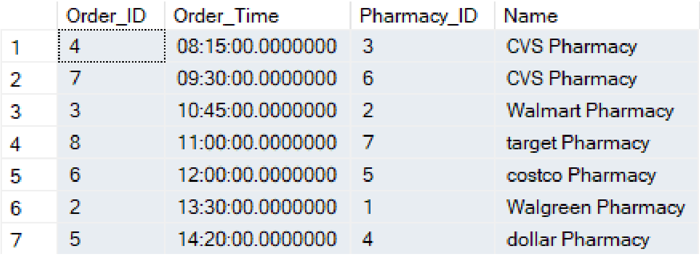
	Patient_ID	First_Name	Street	City	State	Phone_Number
1	1	John	123 Main St	Anytown	CA	-679
2	5	Michael	654 Cedar St	Somewhere	FL	-6234
3	3	William	789 Oak St	Everyville	IL	-1913
4	2	Jane	456 Elm St	Someville	NY	-5123
5	4	Sarah	321 Pine St	Nowhere	TX	-2901
6	6	Linda	987 Maple St	Anywhere	WA	-7335

Query 2



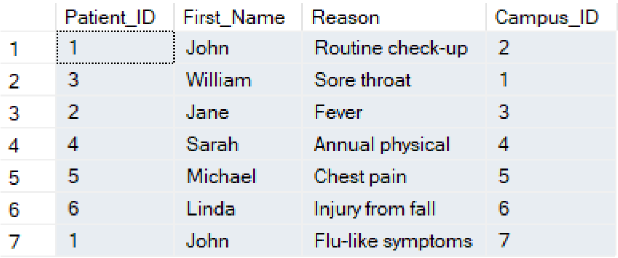
	Order_ID	Medication_ID	Total_Quantity
1	8	3	5
2	5	6	7
3	2	5	8
4	5	2	10
5	4	1	12
6	3	1	15
7	6	7	18
8	7	4	20

Query 3



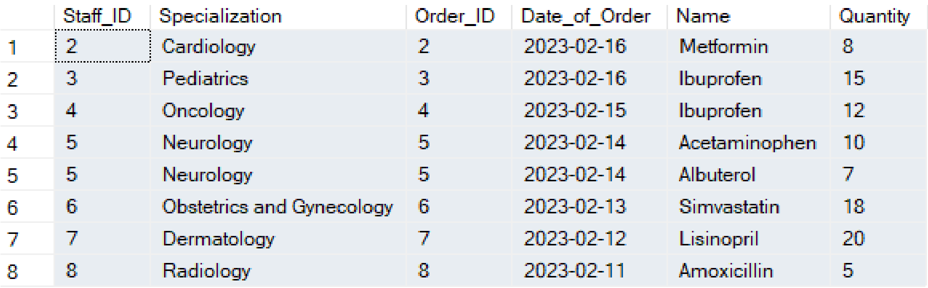
	Order_ID	Order_Time	Pharmacy_ID	Name
1	4	08:15:00.0000000	3	CVS Pharmacy
2	7	09:30:00.0000000	6	CVS Pharmacy
3	3	10:45:00.0000000	2	Walmart Pharmacy
4	8	11:00:00.0000000	7	target Pharmacy
5	6	12:00:00.0000000	5	costco Pharmacy
6	2	13:30:00.0000000	1	Walgreen Pharmacy
7	5	14:20:00.0000000	4	dollar Pharmacy

Query 4



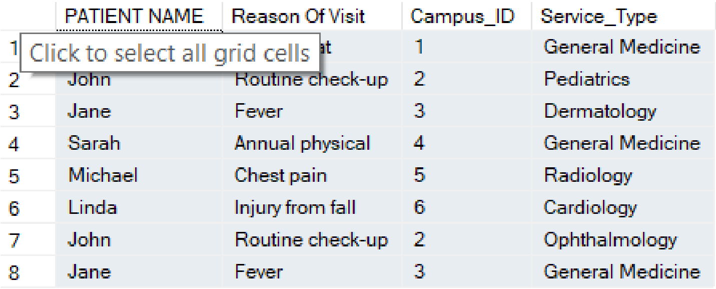
	Patient_ID	First_Name	Reason	Campus_ID
1	1	John	Routine check-up	2
2	3	William	Sore throat	1
3	2	Jane	Fever	3
4	4	Sarah	Annual physical	4
5	5	Michael	Chest pain	5
6	6	Linda	Injury from fall	6
7	1	John	Flu-like symptoms	7

Query 5



	Staff_ID	Specialization	Order_ID	Date_of_Order	Name	Quantity
1	2	Cardiology	2	2023-02-16	Metformin	8
2	3	Pediatrics	3	2023-02-16	Ibuprofen	15
3	4	Oncology	4	2023-02-15	Ibuprofen	12
4	5	Neurology	5	2023-02-14	Acetaminophen	10
5	5	Neurology	5	2023-02-14	Albuterol	7
6	6	Obstetrics and Gynecology	6	2023-02-13	Simvastatin	18
7	7	Dermatology	7	2023-02-12	Lisinopril	20
8	8	Radiology	8	2023-02-11	Amoxicillin	5

Query 6



	PATIENT NAME	Reason Of Visit	Campus_ID	Service_Type
1		t	1	General Medicine
2	John	Routine check-up	2	Pediatrics
3	Jane	Fever	3	Dermatology
4	Sarah	Annual physical	4	General Medicine
5	Michael	Chest pain	5	Radiology
6	Linda	Injury from fall	6	Cardiology
7	John	Routine check-up	2	Ophthalmology
8	Jane	Fever	3	General Medicine

Query Requirements

Query Number	Report Requirements	SQL Code	Recordset

Query1	Select Patient details from patients and order by State	SELECT DISTINCT Patient_ID,First_Name,Street,City,[State],Phone_Number
		FROM Patients
		ORDER BY [State];
			6

Query2	Select orders and medications from order, medication, order_medication
	Where orderID
	Group by and order by for quantity.
		SELECT Orders.Order_ID,Medications.Medication_ID,
		SUM(Order_Medication.Quantity)
		AS [Total_Quantity]
		FROM Orders,Order_Medication,Medications
		WHERE Orders.Order_ID = Order_Medication.Order_ID
		AND Order_Medication.Medication_ID = Medications.Medication_ID
		GROUP BY Medications.Medication_ID, Order_Medication.Quantity,Orders.Order_ID
		ORDER BY Order_Medication.Quantity;
			8

Query3	Select [details of pharmacy order]
	From Pharmacy and order table
	Where [ID]
	Order by time
		SELECT Order_ID,Order_Time,Pharmacy.Pharmacy_ID,Pharmacy.[Name]
		FROM Orders,Pharmacy
		WHERE Orders.Pharmacy_ID= Pharmacy.Pharmacy_ID
		ORDER BY Order_Time;
			7

Query4	Select[patient campus details]
	From patient, visit campus
	Where [ID's]
	Joining 3 Tables
		SELECT Patients.Patient_ID,Patients.First_Name,
		Patient_Visit.Reason,Campuses.Campus_ID
		FROM Patients,Patient_Visit,Campuses
		WHERE Patients.Patient_ID=Patient_Visit.Patient_ID
		AND Patient_Visit.Campus_ID= Campuses.Campus_ID;
			7

Query5	Select [Doctor medication details]
	From Doctors orders, order_medication
	Where[ID]
	Order by order ID
	Joining 4 tables
		SELECT Doctor.Staff_ID,Doctor.Specialization,Orders.Order_ID,
		Orders.Date_of_Order,Medications.[Name],
		Order_Medication.Quantity
		FROM Doctor,Orders,Order_Medication,Medications
		WHERE Doctor.Staff_ID= Orders.Staff_ID
		AND Orders.Order_ID= Order_Medication.Order_ID
		AND Order_Medication.Medication_ID=Medications.Medication_ID
		ORDER BY Orders.Order_ID;
			8

Query6	Joining 5 Tables Selec[Detils] From [Tables]	SELECT Patients.First_Name AS [PATIENT NAME], Patient_Visit.Reason AS [Reason Of Visit], Campuses.Campus_ID,Services.Service_Type	8

	Where [ID of 5 tables]	FROM
		Patients,Patient_Visit,Campuses,Campus_Services,Services
		WHERE Patients.Patient_ID= Patient_Visit.Patient_ID
		AND Patient_Visit.Campus_ID= Campuses.Campus_ID
		AND Campuses.Campus_ID=Campus_Services.Campus_ID
		AND Campus_Services.Service_ID=Services.Service_ID;	

First, connect to the database management system and select the appropriate database to work. Use the CREATE TABLE command to define the structure of the table. This includes specifying the name of the table, the names and data types of each column, and any constraints or indices that should be applied to the table.

```
CREATE TABLE Patient (

PatientID INT PRIMARY KEY,

Name VARCHAR(50) NOT NULL,

Address VARCHAR(255) NOT NULL,

City VARCHAR(50),

State VARCHAR(50),

ZipCode VARCHAR(10),

Email VARCHAR(255) NOT NULL UNIQUE);
```

This command creates a table named "Patients" with columns for PatientsID, name, address, city, state, zip code, and email. The PRIMARY KEY constraint is added to the PatientsID column to ensure it is unique and serves as the primary key for the table. The NOT NULL constraint is added to the Name, Address, and Email columns to ensure they cannot contain null values, and the UNIQUE constraint is added to the Email column to ensure that no two customers can have the same email address.

Once the table is created, you can use the INSERT INTO command to add new rows to the table.

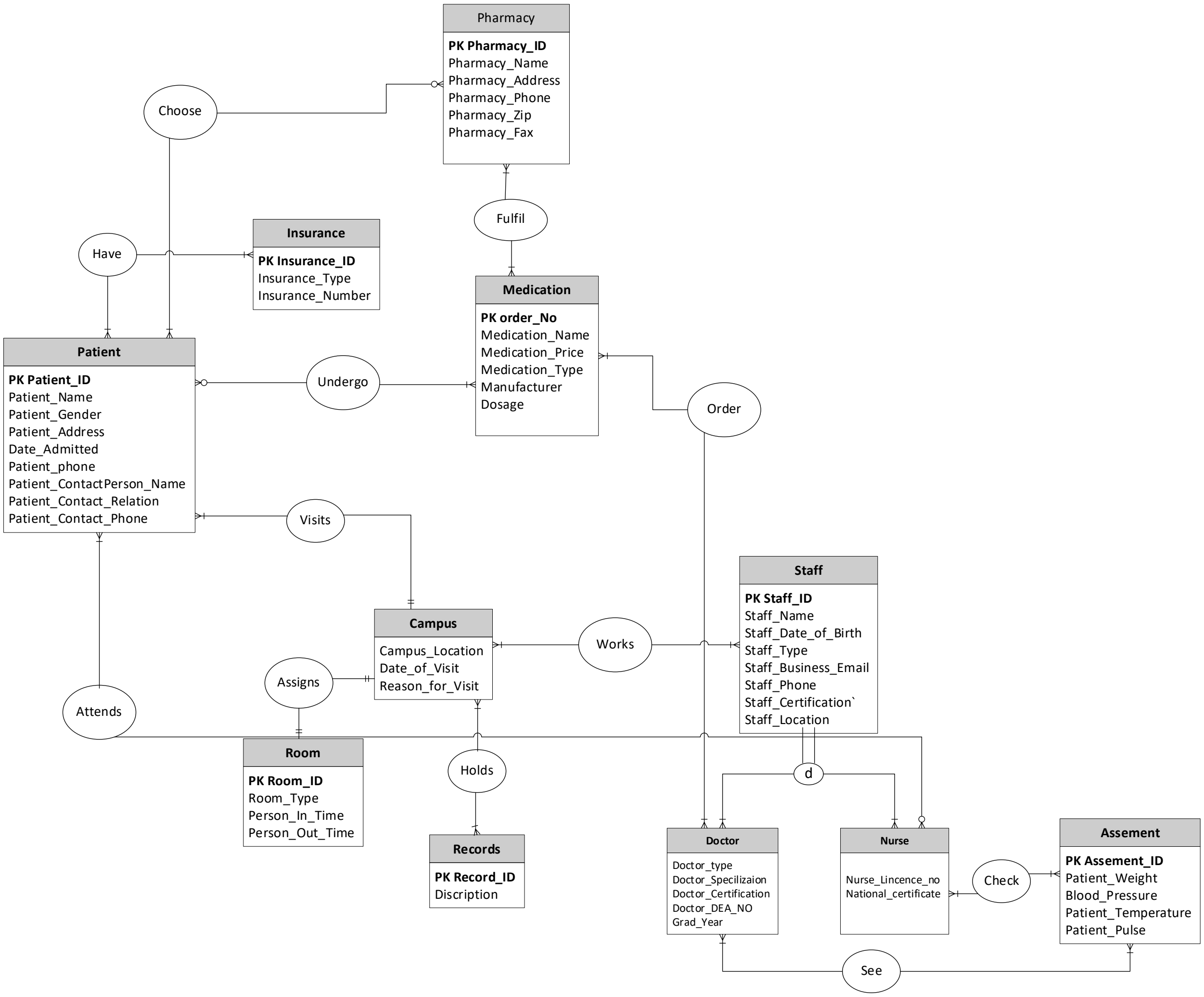
```
INSERT INTO Patients (ID, FirstName, LastName, Age, Email)

VALUES (1, 'John', 'Doe', 30, 'john.doe@example.com'),

    (2, 'Jane', 'Smith', 25, 'jane.smith@example.com'),

    (3, 'Bob', 'Johnson', 40, 'bob.johnson@example.com');
```

Retrieve data from the table, you can use the SELECT command.

```
SELECT ID, FirstName, LastName

FROM Patient

WHERE Age > 30;
```

Appendix

Ravichandra Lakkappa ERD


Pharmacy
PK Pharmacy_ID
Pharmacy_Name
Pharmacy_Address
Pharmacy_Phone
Pharmacy_Zip
Pharmacy_Fax
Choose
Fulfil
Have
Insurance
PK Insurance_ID
Insurance_Type
Insurance_Number
Medication
PK order_No
Medication_Name
Medication_Price
Medication_Type
Manufacturer
Dosage
Patient
PK Patient_ID
Patient_Name
Patient_Gender
Patient_Address
Date_Admitted
Patient_phone
Patient_ContactPerson_Name
Patient_Contact_Relation
Patient_Contact_Phone
Undergo
Order
Visits
Staff
PK Staff_ID
Staff_Name
Staff_Date_of_Birth
Staff_Type
Staff_Business_Email
Staff_Phone
Staff_Certification`
Staff_Location
Campus
Campus_Location
Date_of_Visit
Reason_for_Visit
Works
Assigns
Attends
Room
PK Room_ID
Room_Type
Person_In_Time
Person_Out_Time
Holds
Records
PK Record_ID
Discription
d
Doctor
Doctor_type
Doctor_Specilizaion
Doctor_Certification
Doctor_DEA_NO
Grad_Year
Nurse
Nurse_Lincence_no
National_certificate
Check
Assement
PK Assement_ID
Patient_Weight
Blood_Pressure
Patient_Temperature
Patient_Pulse
See