User Profile:

Create a table for UserProfile. Create column headers for UserID, FirstName, LastName, Username, Password, Email, Address, City, State, ZipCode, Phone, DOB, SSN, Role, and Department. Set the Primary Key as UserID. All fields are required for all users, however Department is not applicable to Patient role. UserID must be a valid ID. Password should meet specific requirements and should be protected. Email should be in correct email format. Address, City, State, and ZipCode should all be valid entries. SSN should also be protected.

Authentication & Authorization:

Create a table for Authentication&Authorization. Column headers include Username, Password, Role, UserID. Username is Primary Key and UserID is Foreign Key which references UserID from UserProfile Table. Role will determine what aspects of the database the user is authorized to view. Username and Password must be correct to be granted access to database (authentication). All fields are required. UserID must be valid ID and Role must be a valid Role.

Appointments:

Create table for Appointments. Create column headers for AppointmentID, Appointment Date, AppointmentTime, PatientID, PhysicianID, Appointment Reason. Primary Key should be set to AppointmentID. PatientID is a Foreign Key that references PatientID in Patient Table. PhysicianID is also a Foreign Key that references PhysicianID in Physician Table. AppointmentDate should be a valid date and AppointmentTime should be a valid time. All fields are required. PatientID and PhysicianID should be valid IDs.

Patient:

Create table for Patient. Column headers should include PatientID, InPatient, BedID, BalanceDue, InsuranceID. PatiendID should be the Primary Key. BedID is a Foreign Key that references BedID in Bed Table and InsuranceID is Foreign Key that references InsuranceID in Insurance Table. InPatient indicates whether patient is inpatient or outpatient. All fields are required. IDs must all be valid.

Physician:

Create Table for Physician. Column headers include PhysicianID, PhysicianType, PhysicianRank, DepartmentID. PhysicianID is the Primary Key and DepartmentID is a Foreign Key that references Department Table. All fields are required. PhysicianID and DepartmentID must be valid IDs. Physician Type and Physician Rank must be valid entries.

Department:

Create Table for Department. Create column headers labeled DepartmentID, Name, and Location. DepartmentID is the Primary Key. All fields are required. DepartmentID must be a valid ID. Name must be a valid department and Location must be a valid location.

Orders:

Create Table named Orders. Create column headers labeled OrderNumber, PatientID, PatientLastName, PatientFirstName, DateRequested, PhysicianID, Location, Status, SetName, and Comments. OrderNumber is the Primary Key. PatientID and PhysicianID are Foreign Keys referencing Patient table and Physician table. Status should be one of a few options (Pending, Processing, Complete, Incomplete, etc.). PatientID and PhysicianID must be valid IDs. Comments are an optional field. Comments can include any additional information needed about the order. All other fields are required. SetName is the name of item being ordered. DateRequested should be in Date format and should be a valid date.

Prescriptions:

Create Table named Prescriptions. Column headers include PrescriptionID, PatientID, PrescriptionName, PhysicianID, PharmacyID, and DatePrescribed. PrescriptionID is the Primary Key. PatientID, PhysicianID, and PharmacyID are Foreign Keys referencing their respective tables. All fields are required. DatePrescribed should be in Date format and should be a valid date. All IDs should be valid.

Billing:

Create a new Table named Billing. Create Column headers labeled BillNumber, PatientID, PatientFirstName, PatientLastName, BillAmount, InsuranceCoverage, AmountOwed, BillDetails, Status, BillDue. Primary Key is BillNumber. PatientID is Foreign Key referencing

Patient Table. BillDetails is an optional field to enter additional information about the bill. InsuranceCoverage is the amount Patient's insurance covers and AmountOwed is how much the patient owes after insurance. Status can be one of a few options (unpaid, processing, paid, denied, etc.). BillDue is the date the bill is due, so should be in Date format and should be valid date.

Insurance:

Create a new table named Insurance. Column headers include InsuranceID, ProviderName, CoverageDesc, Deductible, Copay, Coinsurance, OutOfPocketMax. InsuranceID is the Primary Key. All fields are required. Some insurance may not have Deductible, Copay, Coinsurance, or OutOfPocketMax which in that case those fields will be set to 0.

Pharmacy:

Create a new table named Pharmacy. Column headers include PharmacyID, PharmacyName, Location, InventoryID. PharmacyID is Primary Key. All fields are required. PharmacyID must be valid as well as Location. InventoryID is Foreign Key which references Inventory Table.

Inventory:

Create new table named Inventory. Create column headers labeled MedicationID, MedicationName, Strength, DoseageForm, Quantity, ExpiryDate, Cost, and Vendor. MedicationID is Primary Key. All fields are required. ExpiryDate must be in Date format and must be a valid date. MedicationID must be a valid ID.

Lab Test:

Create new table named LabTest. Create column headers labeled LabTestID, TestName, TestType, TestResult, PhysicianID. LabTestID is the Primary Key and PhysicianID is a Foreign Key that references Physician table. All fields are required. PhysicianID is the ID of the Physician submitting Lab Test Request. Test Names and Types must be valid. IDs must also be valid.

Notification:

Create table named Notification. Create column headers labeled NotificationID, PatientID, RecipientID. NotificationID is the Primary Key and PatientID is a Foreign Key referencing the Patient Table. RecipientID is the ID of the person the message is being sent to. This must be a valid ID and can be a Nurse, Physician, or Staff. PatientID must be valid ID.