1. Relational schemas

   Patient(<u>patientID</u>: string, firstName: string, lastName: string, emergencyContanctFirstName: string, emergencyContactLastName: string, insuranceCompany: string, insurancePolicyNum: string)

   Admission(<u>admitNum</u>: string, <u>patientID</u>: string, admitDate: date, dischargeDate: date)

   Employee(<u>employeeID</u>: string, firstName: string, lastName: string, priviledgeType: string, employeeType: string)

   Service(<u>serviceNum</u>: string, patientID: string, employeeID: string, serviceType: string, admitNum: string)

   Treatment(<u>treatmentID</u>: string, treatmentName: string, <u>treatmentDate</u>: timestamp, <u>serviceNum</u>: string, technicianID: string, nurseID: string, docID: string)

   Primary keys have been underlined. If multiple attributes make up the primary key, multiple attributes were underlined.

2. Description of how the ER model was translated into schemas

   The ER model defined all of the data elements and relationships within the hospital system. Patients are provided services and are admitted. Admissions result in services, but services can occur without admittance. Treatments are prescribed or administered as a result of a service. Services and treatments have associated employees, and the employeeID associated with the services of an admitted patient has admitting privileges and was responsible for said patient's admittance. The employeeID(s) associated with an instance of treatment are the ones responsible for prescribing or administering that treatment described.

   In simpler terms, patients are admitted. Patients are provided services. Services often result in treatment. Employees oversee treatments. Admissions result in services. Doctors with admitting privileges admit patients.

   The employeeID associated with the admitting doctor was made to be an attribute of services instead of admissions, but all admissions result in services. This decision was made to reduce complexity of the overall system, and because all services are associated with an employee of some sort. The functionality is the same had employeeID been made an attribute of admissions instead.

   All of the primary keys are noted in the ER model and are underlined in the relational schemas. In addition, all attributes listed in the ER model are listed in the relational schemas.

3. Text description of normalization to BCNF

The goal of achieving BCNF is to reduce the risk of anomaly as a result of the underlying functional dependencies. BCNF defines a schema R such that *every non-trivial X -> Y which holds on R, X is a superkey of R* (defined in our Module 6 notes slides)*.

For patient:

patientID -> firstName; patientID -> lastName; patientID -> emergencyContanctFirstName; patientID -> emergencyContactLastName; patientID -> insuranceCompany; patientID -> insurancePolicyNum

Key: patientID
Highest NF: BCNF

For admissions:

admitNum -> patientID, admitNum -> admitDate, admitNum -> dischargeDate

Key: admitNum
Highest NF: BCNF

For employees:

employeeID -> firstName; employeeID -> lastName; employeeID -> privilegeType; employeeID -> employeeType

Key: employeeID
Highest NF: BCNF

For services:

serviceNum -> patientID; serviceNum -> employeeID; serviceNum -> serviceType; serviceNum -> admitNum;

Key: serviceNum

Highhest NF: BCNF

For treatments:

treatmentID, serviceNum, treatmentDate -> treatmentName, treatmentID, serviceNum, treatmentDate -> technicianID; treatmentID, serviceNum, treatmentDate -> nurseID; treatmentID, serviceNum, treatmentDate -> docID

Key: treatmentID, serviceNum, treatmentDate

Highest NF: BCNF

In the examples above, anomalies will likely arise if any attributes are removed from the keys. Decomposition would not be necessary here.

4. Text annotations of assumptions, constraints, decisions of 3NF

Each table has a single unique attribute except treatments. Since a treatment can be administered an infinite number of times, and because multiple treatments can be administered per service, a combination of treatmentID, serviceNum, and treatmentDate (a timestamp, which is more specific than a generic date) is necessary to properly identify a treatment instance.

PatientID cannot be used to identify anything outside of the patient table because patients can have multiple admissions, services, and treatments.

Decisions on 3NF/BCNF were made by evaluating prime vs. non-prime attributes. The following attributes are all considered to be prime: patientID, admitNum, employeeID, and serviceNum. Those values must all be unique.