



# **Hospital Management System**

BUAN 6320

## Contents

Overview .....	2
Objective .....	2
Relational Data Model .....	3
Assumptions/Notes About Data Entities and Relationships .....	3
Entity-Relationship Diagram .....	3
Physical MySQL Database .....	5
Assumptions/Notes About Data Set .....	5
Screen shot of Physical Database objects .....	5
Data in the Database .....	6

## Overview

### **Key stakeholders/beneficiaries - Who is involved in the system?**

- Decision makers/higher management
- Doctors
- Patients
- Administration
- Support staff

### **Data – What kind of data is generated in the system?**

- Employees' data
- Patients' data
- Lab Tests records
- Inventory – Existing stock, upcoming orders, etc.
- Business numbers - billing, revenues, etc.

## Objective

The purpose of the project entitled as “Hospital Management System” is to computerize the Front Office Management of Hospital to develop software, which is user friendly simple, fast, and cost – effective. The primary aim of the project is to develop a database for a multi-specialty hospital to maintain events and records of employees, Doctors, Patients, and day to day management operations. It further includes a subset of services under each segment such as patients' history, registration, Doctor schedules, appointments, and billing. The goal of the project is to showcase how data can be managed systematically with the help of the database. Database Management System in place would lead to-

- Design and maintain a database of the patient as well as employee details of the hospital
- It also includes the patients' appointment, billing information, lab tests, disease history
- A feedback table is maintained in the database to store the feedbacks of the patients
- Reports are generated to visualize the data in a better manner
- Organized structure – all data in one place – improved accessibility.
- Better utilization of resources by reducing redundancy.
- Helps aid business decision making, to improve outcomes.

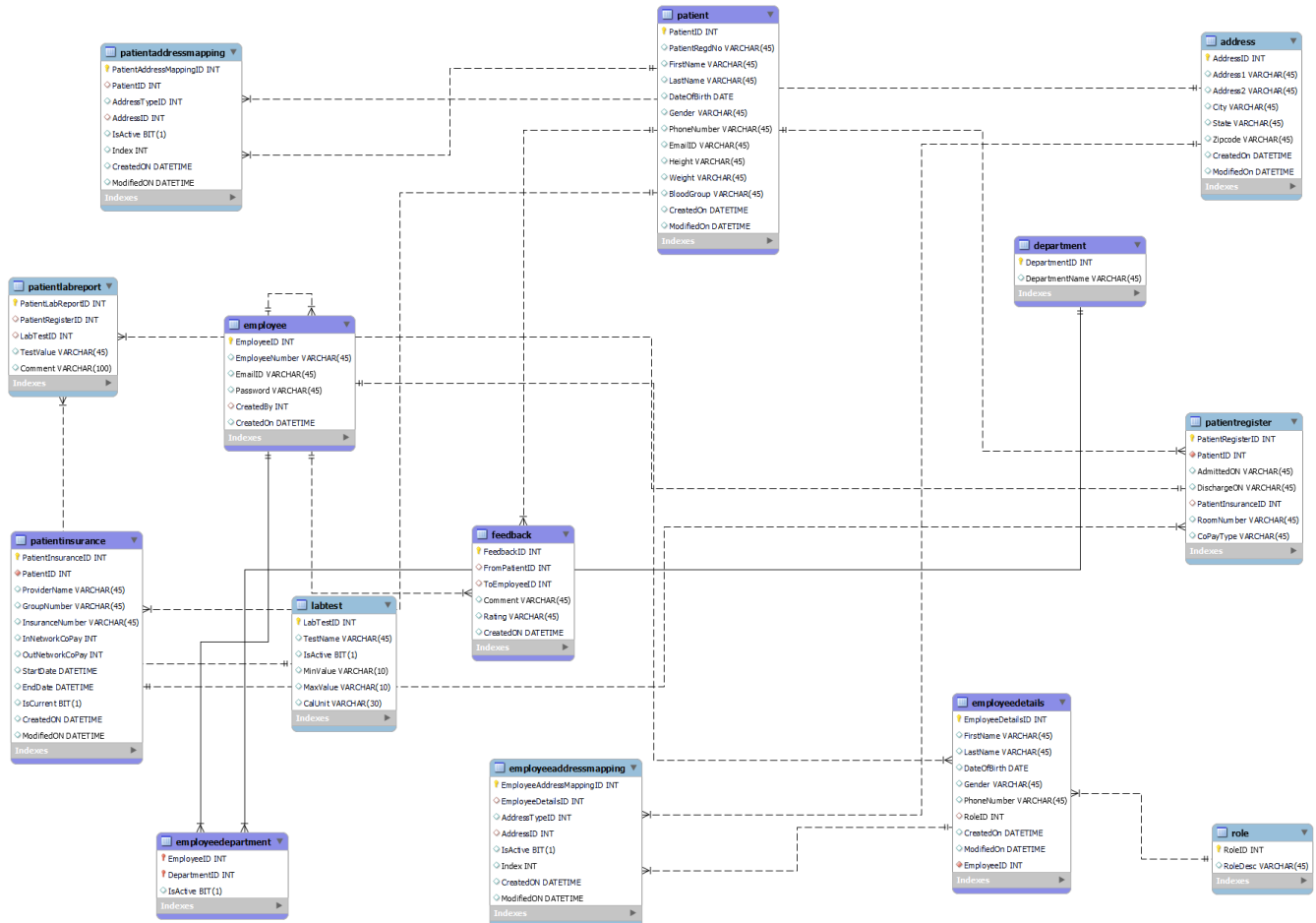
## Relational Data Model

## Assumptions/Notes About Data Entities and Relationships

Include assumptions about data entities and their relationships with each other.

Include reasons why the data model is in 3NF.

## Entity-Relationship Diagram



## Entity-Relationship Across Data

Single Employee - many Departments( One to many reslationship defined)

Patient details - one to many parts, each part to be clubbed to multiple data, Patientinsurance, Billing, patientaddress.

Employee table is connected to different tables with foreign key as Employee ID.

Patient: One to many

Employee: One to many

Department: Many to one



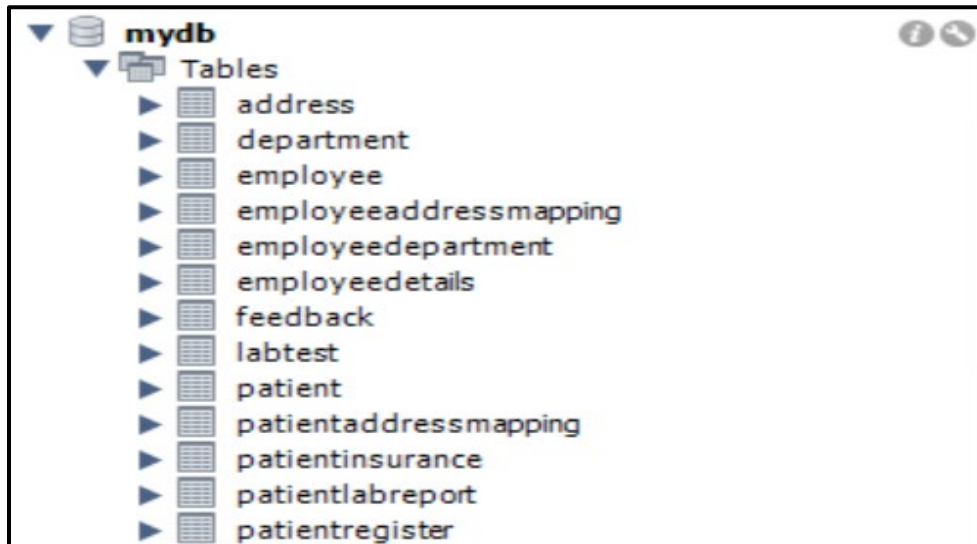
## Physical MySQL Database

### Assumptions/Notes About Data Set

Include any assumptions made about data such as empty fields, sparse data, bad data, etc.

### Screen shot of Physical Database objects

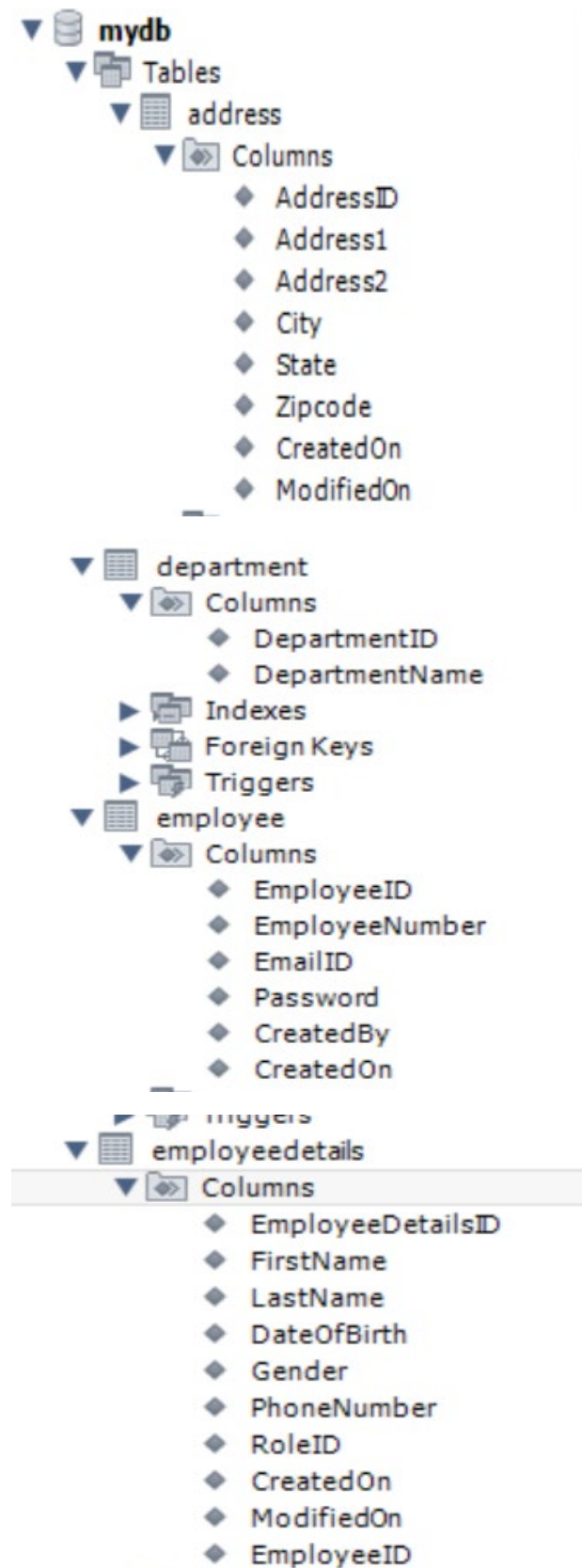
#### Tables



#### Views and Stored Procedures



## Data in the Database



- ▼ feedback
  - ▼ Columns
    - ◆ FeedbackID
    - ◆ FromPatientID
    - ◆ ToEmployeeID
    - ◆ Comment
    - ◆ Rating
    - ◆ CreatedON
  - ▶ Indexes
  - ▶ Foreign Keys
  - ▶ Triggers

- ▼ labtest
  - ▼ Columns
    - ◆ LabTestID
    - ◆ TestName
    - ◆ IsActive
    - ◆ MinValue
    - ◆ MaxValue
    - ◆ CalUnit
  - ▶ Indexes
  - ▶ Foreign Keys
  - ▶ Triggers

- ▼ patient
  - ▼ Columns
    - ◆ PatientID
    - ◆ PatientRegdNo
    - ◆ FirstName
    - ◆ LastName
    - ◆ DateOfBirth
    - ◆ Gender
    - ◆ PhoneNumber
    - ◆ EmailID
    - ◆ Height
    - ◆ Weight
    - ◆ BloodGroup
    - ◆ CreatedOn
    - ◆ ModifiedOn

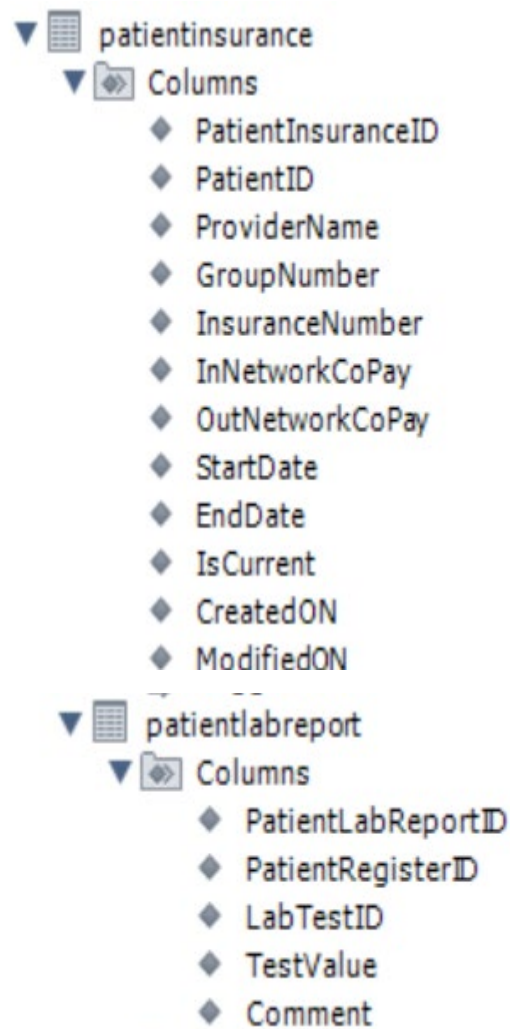
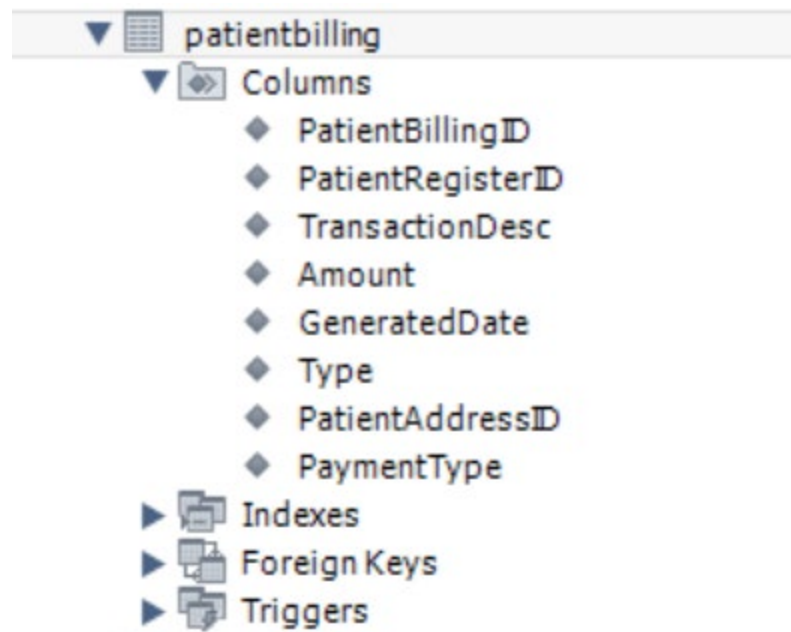


- ▼ patientaddressmapping
  - ▼ Columns
    - ◆ PatientAddressMappingID
    - ◆ PatientID
    - ◆ AddressTypeID
    - ◆ AddressID
    - ◆ IsActive
    - ◆ Index
    - ◆ CreatedON
    - ◆ ModifiedON
  - ▶ Indexes
  - ▶ Foreign Keys
  - ▶ Triggers

- ▼ patientappointment
  - ▼ Columns
    - ◆ PatientID
    - ◆ EmployeeID
    - ◆ AppoitmentDate
    - ◆ IsComplete
    - ◆ IsCancelled
    - ◆ IsNoShow
    - ◆ CreatedBy
    - ◆ CreatedOn
  - ▶ Indexes
  - ▶ Foreign Keys
  - ▶ Triggers

- ▼ patientattendant
  - ▼ Columns
    - ◆ PatientRegisterID
    - ◆ EmployeeID
  - ▶ Indexes
  - ▶ Foreign Keys
  - ▶ Triggers

- ▼ patientbilling





## Data in the Database:

Table Name	Primary Key	Foreign Key
Employee	Employee ID	CreatedBy
Department	Department ID	-
EmployeeDepartment	Employee ID	Employee ID, Department ID
Employeeedetails	EmployeeedetailsID	RoleID, Employee ID
Address	AddressID	-
Patient	PatientID	-
Patientregister	PatientRegisterID	PatientID, PatientInsuranceID
Role	RoleID	-
PatientInsurance	PatientInsuranceID	PatientID
Labtest	LabtestID	-
Feedback	FeedbackID	FromPatientID, ToEmployeeID
Employeeaddressmapping	EmployeeaddressmappingID	EmployeeDetailsID
Patientaddressmapping	PatientaddressmappingID	PatientID, AddressID
patientlabreport	patientlabreportID	PatientRegisterID, LabtestID