**Hospital Billing System**

In partial submission of requirements in CSCC 20

Submitted by

DJ Renzo Emmanuel I. Bince

Eann Marten E. Lao

Submitted to:

Mr. Jessie Christopher Lagrosas

**Table of contents**

**Page No.**

Introduction 3

Requirements Analysis 3

Use-case Analysis 4

Class Diagram 7

Sequence Diagram 7

API 8

Appendices 12

**Introduction**

This paper serves as the documentation of the Hospital Billing System.

The program presented is a hospital billing system that utilizes java and MySQL as the program and the database respectively. The program is developed for the final requirements of the subjects CSCC20 and CC14.

The program is a simple billing system for a hospital with a few basic features such as adding a patient, generating a billing, and displaying it.

Requirements Analysis

Hospital Billing System

Functional Requirements:

* Allow user to add items to the billing (depending on which user)
* Allow different users (different hospital services)
* Allow different items/services to be added
* Show the user a current billing
* Allows user to output a receipt for payment (text file)

Non-Functional Requirements

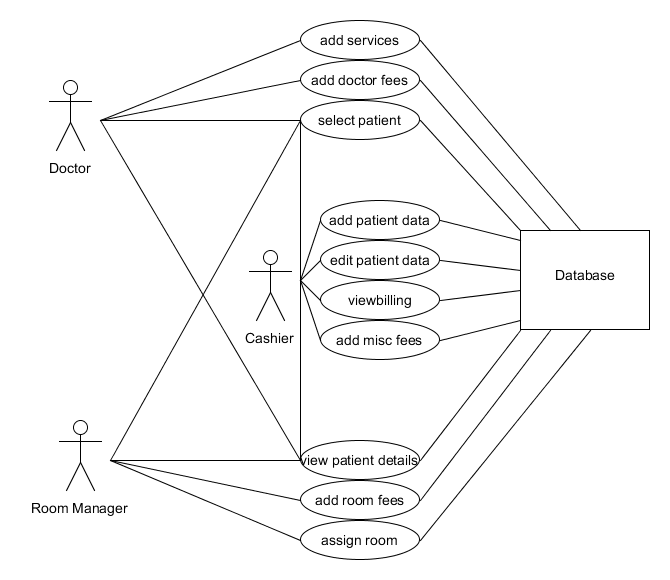
* Should allow storage of past billings
* Data should remain intact (data integrity)
* Program should accept keyboard input for data entry

Use-Case Analysis

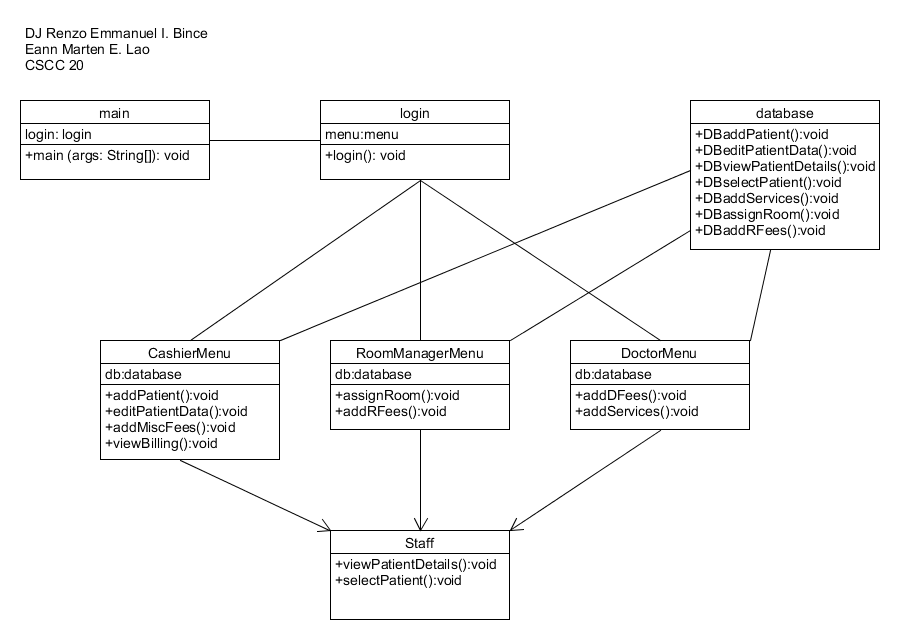
Hospital Billing System

Use Cases:

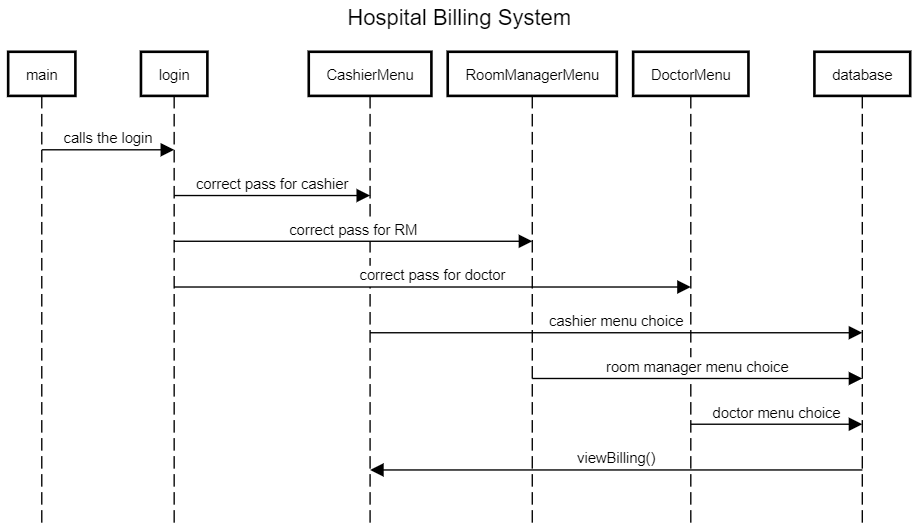
1. Add Patient
   1. Actor: Cashier
   2. Success Scenario: Patient’s details are added to the system
   3. Pre-Conditon: None
   4. Post-Condition: Patient data is added to the system
2. Edit Patient Data
   1. Actor: Cashier
   2. Success Scenario: Patient’s details are edited
   3. Pre-Condition: Patient must exist in the database
   4. Post-Condition: Patient Data is altered in the database
3. Add Misc Fees
   1. Actor: Cashier
   2. Success Scenario: Misc fees are reflected in the bill
   3. Pre-Condition: Patient data must exist
   4. Post-Condition: Misc fees are added to the bill in the database
4. View Billing
   1. Actor: Cashier
   2. Success Scenario: Billing is printed out
   3. Pre-Condition: Patient data must exist, must have at least one billing
   4. Post-Condition: None
5. Add Doctor Fees
   1. Actor: Doctor
   2. Success Scenario: Doctor fees are reflected in the billing
   3. Pre-Condition: Patient data must exist
   4. Post-Condition: Doctor fees are added in the billing database
6. Add Services
   1. Actor: Doctor
   2. Success Scenario: Services are reflected in the billing
   3. Pre-Condition: Patient details must exist, must
   4. Post-Condition: Services are added into billing in the database
7. Assign room
   1. Actor: Room Manager
   2. Success Scenario: Patient is assigned to a room
   3. Pre-Condition: Patient must be an in-patient
   4. Post-Condition: Patient and room ID is linked in database
8. Add Room Fees
   1. Actor: Room Manager
   2. Success Scenario: Room is reflected in billing
   3. Pre-Condition: Patient must be an in-patient,
   4. Post-Condition: Room billing is added to billing in the database
9. Select Patient
   1. Actor: Cashier, Doctor, or Room Manager
   2. Success Scenario: Selected a patient the system will interact with
   3. Pre-Condition: There has to be existing patients
   4. Post-Condition: Patient is selected
10. View Patient Details
    1. Actor: Cashier, Doctor, Room manager
    2. Success Scenario: Displays the patient data
    3. Pre-Condition: Patient data has to exist in the database
    4. Post-Condition: None

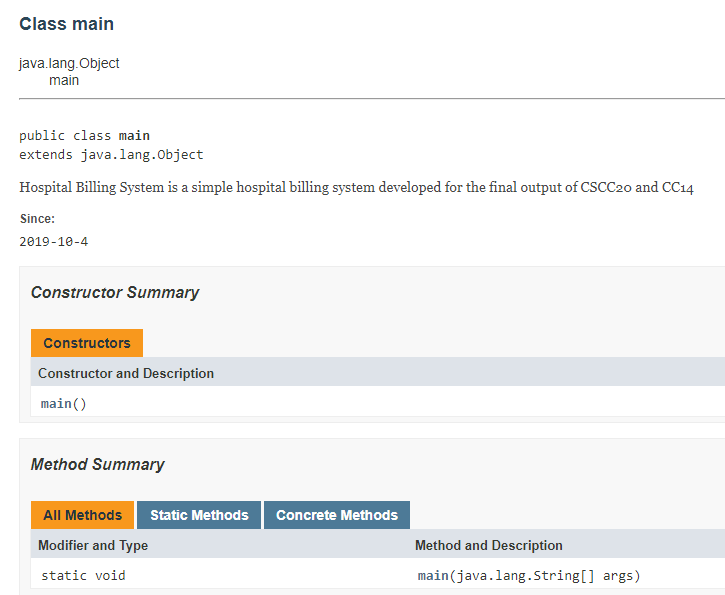


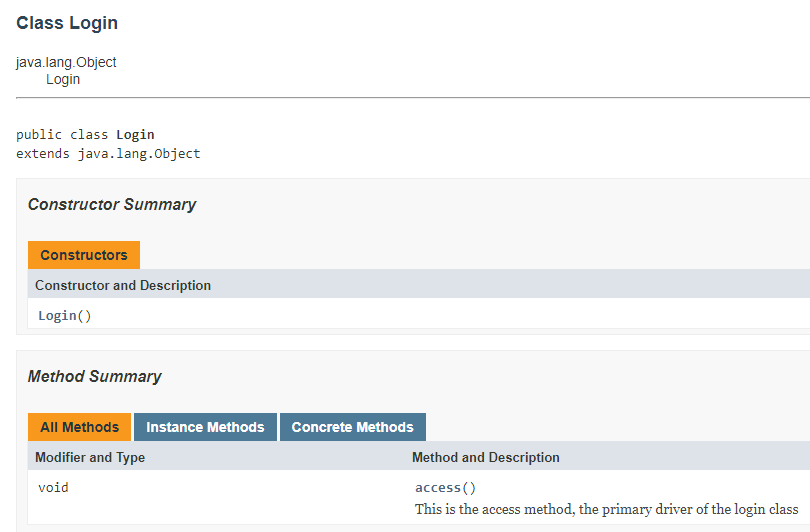
Class Diagram

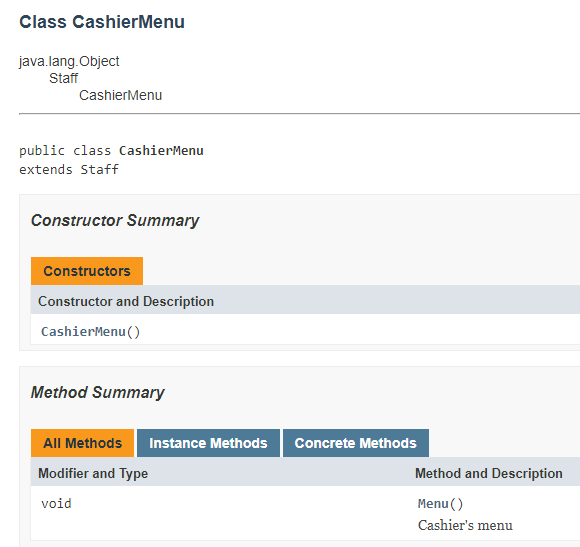


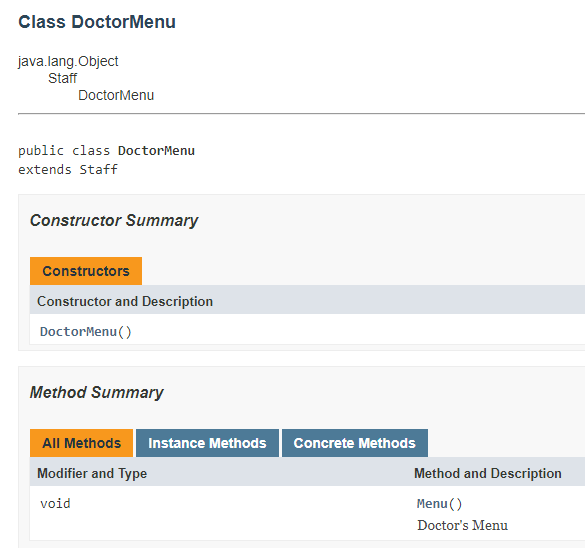
Sequence Diagram

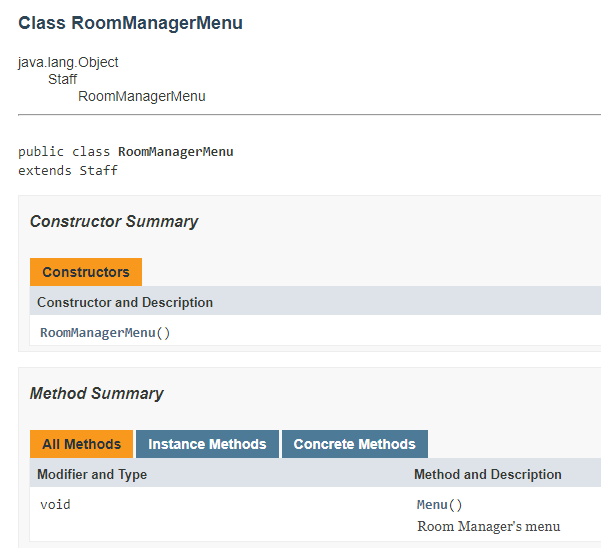


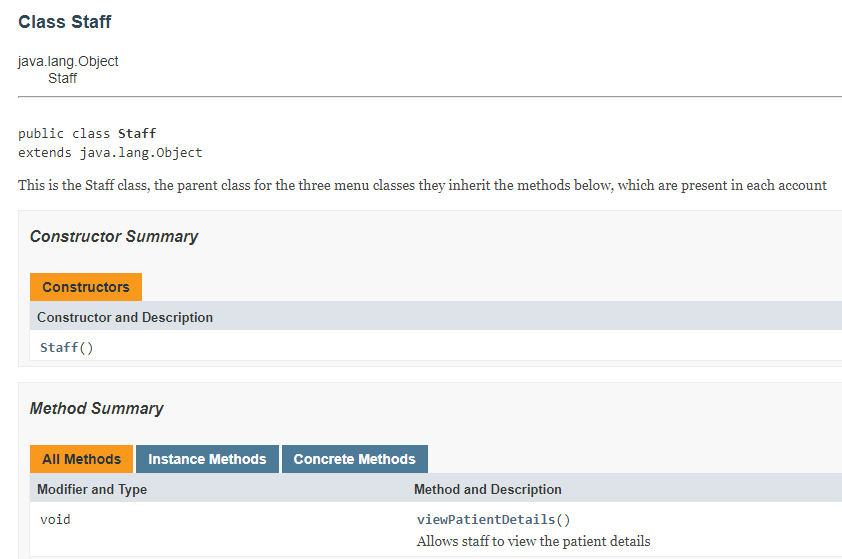
API

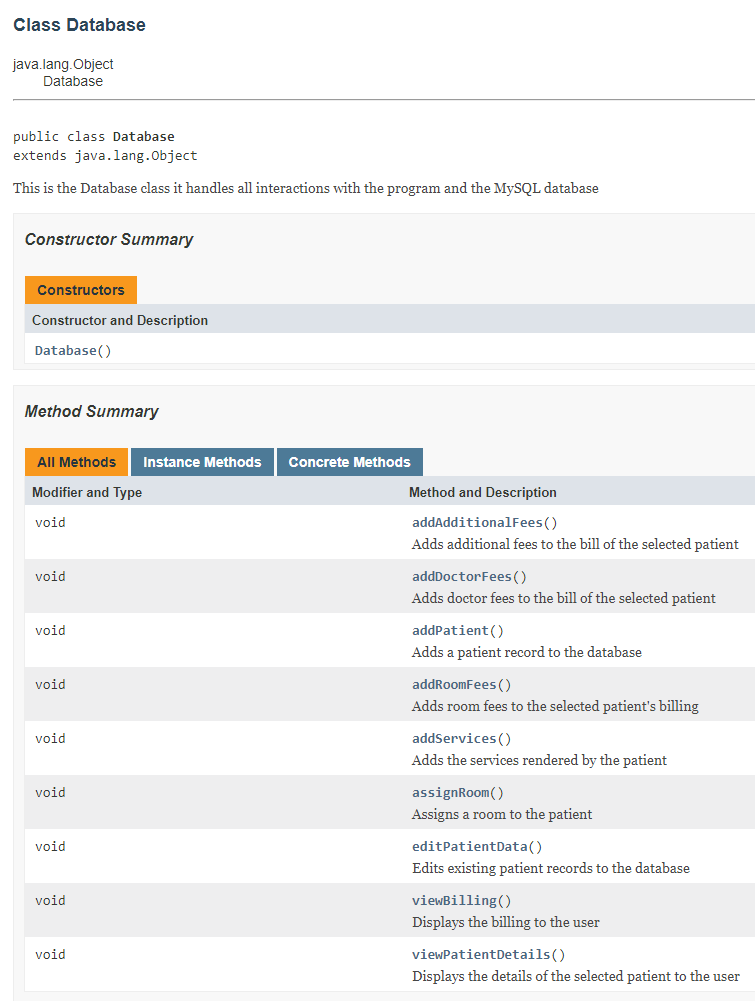




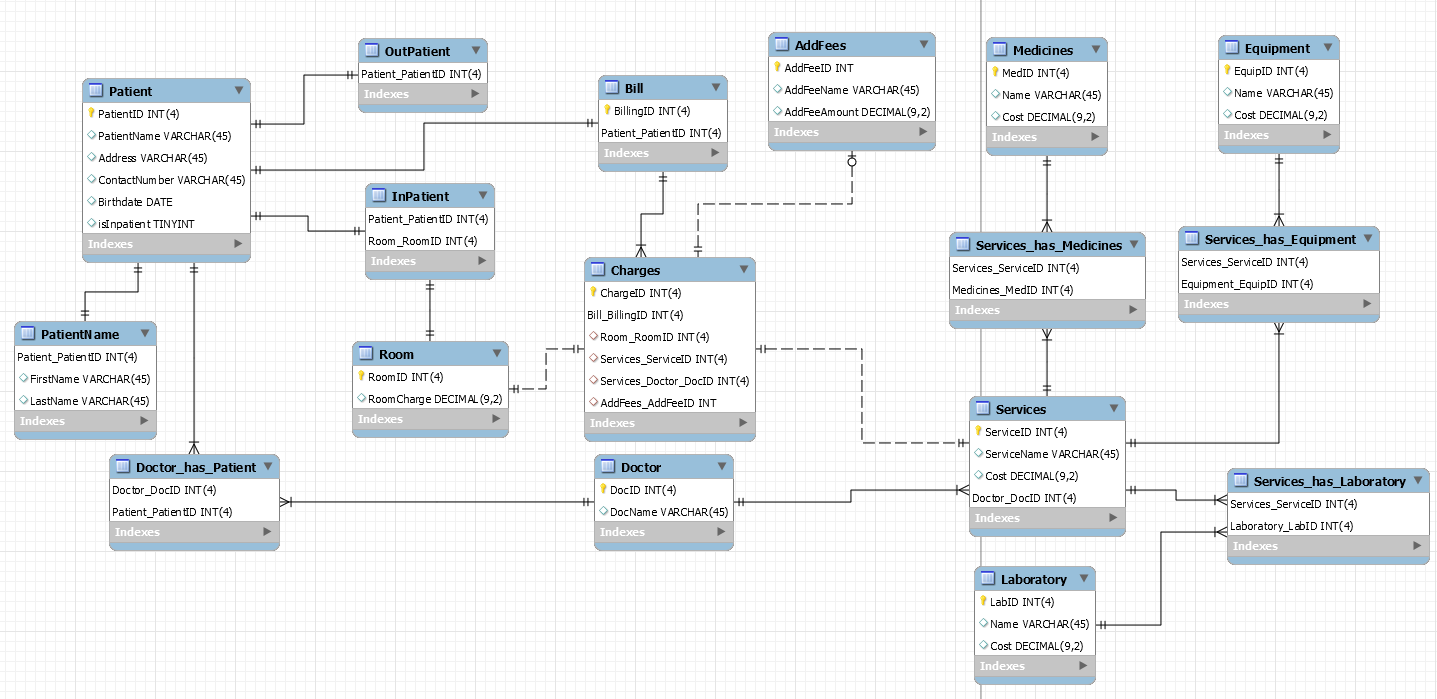








APPENDICES



-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

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

-- Schema mydb

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

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

-- Schema HospitalBillingSys

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

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

-- Schema HospitalBillingSys

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

CREATE SCHEMA IF NOT EXISTS `HospitalBillingSys` ;

USE `HospitalBillingSys` ;

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

-- Table `HospitalBillingSys`.`Patient`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Patient` (

`PatientID` INT(4) NOT NULL AUTO\_INCREMENT,

`PatientName` VARCHAR(45) NULL,

`Address` VARCHAR(45) NULL,

`ContactNumber` VARCHAR(45) NULL,

`Birthdate` DATE NULL,

`isInpatient` TINYINT NULL,

PRIMARY KEY (`PatientID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Bill`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Bill` (

`BillingID` INT(4) NOT NULL AUTO\_INCREMENT,

`Patient\_PatientID` INT(4) NOT NULL,

PRIMARY KEY (`BillingID`, `Patient\_PatientID`),

INDEX `fk\_Bill\_Patient1\_idx` (`Patient\_PatientID` ASC) VISIBLE,

CONSTRAINT `fk\_Bill\_Patient1`

FOREIGN KEY (`Patient\_PatientID`)

REFERENCES `HospitalBillingSys`.`Patient` (`PatientID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Room`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Room` (

`RoomID` INT(4) NOT NULL AUTO\_INCREMENT,

`RoomCharge` DECIMAL(9,2) NULL,

PRIMARY KEY (`RoomID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Doctor`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Doctor` (

`DocID` INT(4) NOT NULL AUTO\_INCREMENT,

`DocName` VARCHAR(45) NULL,

PRIMARY KEY (`DocID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Services`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Services` (

`ServiceID` INT(4) NOT NULL AUTO\_INCREMENT,

`ServiceName` VARCHAR(45) NULL,

`Cost` DECIMAL(9,2) NULL,

`Doctor\_DocID` INT(4) NOT NULL,

PRIMARY KEY (`ServiceID`, `Doctor\_DocID`),

INDEX `fk\_Services\_Doctor1\_idx` (`Doctor\_DocID` ASC) VISIBLE,

CONSTRAINT `fk\_Services\_Doctor1`

FOREIGN KEY (`Doctor\_DocID`)

REFERENCES `HospitalBillingSys`.`Doctor` (`DocID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`AddFees`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`AddFees` (

`AddFeeID` INT NOT NULL AUTO\_INCREMENT,

`AddFeeName` VARCHAR(45) NULL,

`AddFeeAmount` DECIMAL(9,2) NULL,

PRIMARY KEY (`AddFeeID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Charges`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Charges` (

`ChargeID` INT(4) NOT NULL AUTO\_INCREMENT,

`Bill\_BillingID` INT(4) NOT NULL,

`Room\_RoomID` INT(4) NULL DEFAULT NULL,

`Services\_ServiceID` INT(4) NULL DEFAULT NULL,

`Services\_Doctor\_DocID` INT(4) NULL DEFAULT NULL,

`AddFees\_AddFeeID` INT NULL DEFAULT NULL,

PRIMARY KEY (`ChargeID`, `Bill\_BillingID`),

INDEX `fk\_Charges\_Room1\_idx` (`Room\_RoomID` ASC) VISIBLE,

INDEX `fk\_Charges\_Services1\_idx` (`Services\_ServiceID` ASC, `Services\_Doctor\_DocID` ASC) VISIBLE,

INDEX `fk\_Charges\_AddFees1\_idx` (`AddFees\_AddFeeID` ASC) VISIBLE,

UNIQUE INDEX `Room\_RoomID\_UNIQUE` (`Room\_RoomID` ASC) VISIBLE,

UNIQUE INDEX `Services\_ServiceID\_UNIQUE` (`Services\_ServiceID` ASC) VISIBLE,

UNIQUE INDEX `Services\_Doctor\_DocID\_UNIQUE` (`Services\_Doctor\_DocID` ASC) VISIBLE,

UNIQUE INDEX `AddFees\_AddFeeID\_UNIQUE` (`AddFees\_AddFeeID` ASC) VISIBLE,

CONSTRAINT `fk\_Charges\_Bill`

FOREIGN KEY (`Bill\_BillingID`)

REFERENCES `HospitalBillingSys`.`Bill` (`BillingID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Charges\_Room1`

FOREIGN KEY (`Room\_RoomID`)

REFERENCES `HospitalBillingSys`.`Room` (`RoomID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Charges\_Services1`

FOREIGN KEY (`Services\_ServiceID` , `Services\_Doctor\_DocID`)

REFERENCES `HospitalBillingSys`.`Services` (`ServiceID` , `Doctor\_DocID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Charges\_AddFees1`

FOREIGN KEY (`AddFees\_AddFeeID`)

REFERENCES `HospitalBillingSys`.`AddFees` (`AddFeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Laboratory`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Laboratory` (

`LabID` INT(4) NOT NULL AUTO\_INCREMENT,

`Name` VARCHAR(45) NULL,

`Cost` DECIMAL(9,2) NULL,

PRIMARY KEY (`LabID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Medicines`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Medicines` (

`MedID` INT(4) NOT NULL AUTO\_INCREMENT,

`Name` VARCHAR(45) NULL,

`Cost` DECIMAL(9,2) NULL,

PRIMARY KEY (`MedID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Equipment`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Equipment` (

`EquipID` INT(4) NOT NULL AUTO\_INCREMENT,

`Name` VARCHAR(45) NULL,

`Cost` DECIMAL(9,2) NULL,

PRIMARY KEY (`EquipID`))

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`InPatient`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`InPatient` (

`Patient\_PatientID` INT(4) NOT NULL,

`Room\_RoomID` INT(4) NOT NULL,

PRIMARY KEY (`Patient\_PatientID`, `Room\_RoomID`),

INDEX `fk\_InPatient\_Room1\_idx` (`Room\_RoomID` ASC) VISIBLE,

CONSTRAINT `fk\_InPatient\_Patient1`

FOREIGN KEY (`Patient\_PatientID`)

REFERENCES `HospitalBillingSys`.`Patient` (`PatientID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_InPatient\_Room1`

FOREIGN KEY (`Room\_RoomID`)

REFERENCES `HospitalBillingSys`.`Room` (`RoomID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`OutPatient`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`OutPatient` (

`Patient\_PatientID` INT(4) NOT NULL,

PRIMARY KEY (`Patient\_PatientID`),

CONSTRAINT `fk\_OutPatient\_Patient1`

FOREIGN KEY (`Patient\_PatientID`)

REFERENCES `HospitalBillingSys`.`Patient` (`PatientID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Doctor\_has\_Patient`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Doctor\_has\_Patient` (

`Doctor\_DocID` INT(4) NOT NULL,

`Patient\_PatientID` INT(4) NOT NULL,

PRIMARY KEY (`Doctor\_DocID`, `Patient\_PatientID`),

INDEX `fk\_Doctor\_has\_Patient\_Patient1\_idx` (`Patient\_PatientID` ASC) VISIBLE,

INDEX `fk\_Doctor\_has\_Patient\_Doctor1\_idx` (`Doctor\_DocID` ASC) VISIBLE,

CONSTRAINT `fk\_Doctor\_has\_Patient\_Doctor1`

FOREIGN KEY (`Doctor\_DocID`)

REFERENCES `HospitalBillingSys`.`Doctor` (`DocID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Doctor\_has\_Patient\_Patient1`

FOREIGN KEY (`Patient\_PatientID`)

REFERENCES `HospitalBillingSys`.`Patient` (`PatientID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`PatientName`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`PatientName` (

`Patient\_PatientID` INT(4) NOT NULL,

`FirstName` VARCHAR(45) NULL,

`LastName` VARCHAR(45) NULL,

PRIMARY KEY (`Patient\_PatientID`),

CONSTRAINT `fk\_table1\_Patient1`

FOREIGN KEY (`Patient\_PatientID`)

REFERENCES `HospitalBillingSys`.`Patient` (`PatientID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Services\_has\_Medicines`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Services\_has\_Medicines` (

`Services\_ServiceID` INT(4) NOT NULL,

`Medicines\_MedID` INT(4) NOT NULL,

PRIMARY KEY (`Services\_ServiceID`, `Medicines\_MedID`),

INDEX `fk\_Services\_has\_Medicines\_Medicines1\_idx` (`Medicines\_MedID` ASC) VISIBLE,

INDEX `fk\_Services\_has\_Medicines\_Services1\_idx` (`Services\_ServiceID` ASC) VISIBLE,

CONSTRAINT `fk\_Services\_has\_Medicines\_Services1`

FOREIGN KEY (`Services\_ServiceID`)

REFERENCES `HospitalBillingSys`.`Services` (`ServiceID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Services\_has\_Medicines\_Medicines1`

FOREIGN KEY (`Medicines\_MedID`)

REFERENCES `HospitalBillingSys`.`Medicines` (`MedID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Services\_has\_Equipment`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Services\_has\_Equipment` (

`Services\_ServiceID` INT(4) NOT NULL,

`Equipment\_EquipID` INT(4) NOT NULL,

PRIMARY KEY (`Services\_ServiceID`, `Equipment\_EquipID`),

INDEX `fk\_Services\_has\_Equipment\_Equipment1\_idx` (`Equipment\_EquipID` ASC) VISIBLE,

INDEX `fk\_Services\_has\_Equipment\_Services1\_idx` (`Services\_ServiceID` ASC) VISIBLE,

CONSTRAINT `fk\_Services\_has\_Equipment\_Services1`

FOREIGN KEY (`Services\_ServiceID`)

REFERENCES `HospitalBillingSys`.`Services` (`ServiceID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Services\_has\_Equipment\_Equipment1`

FOREIGN KEY (`Equipment\_EquipID`)

REFERENCES `HospitalBillingSys`.`Equipment` (`EquipID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `HospitalBillingSys`.`Services\_has\_Laboratory`

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

CREATE TABLE IF NOT EXISTS `HospitalBillingSys`.`Services\_has\_Laboratory` (

`Services\_ServiceID` INT(4) NOT NULL,

`Laboratory\_LabID` INT(4) NOT NULL,

PRIMARY KEY (`Services\_ServiceID`, `Laboratory\_LabID`),

INDEX `fk\_Services\_has\_Laboratory\_Laboratory1\_idx` (`Laboratory\_LabID` ASC) VISIBLE,

INDEX `fk\_Services\_has\_Laboratory\_Services1\_idx` (`Services\_ServiceID` ASC) VISIBLE,

CONSTRAINT `fk\_Services\_has\_Laboratory\_Services1`

FOREIGN KEY (`Services\_ServiceID`)

REFERENCES `HospitalBillingSys`.`Services` (`ServiceID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Services\_has\_Laboratory\_Laboratory1`

FOREIGN KEY (`Laboratory\_LabID`)

REFERENCES `HospitalBillingSys`.`Laboratory` (`LabID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;