May 14, 2020

Bradley De Ath

DEA18001355

Database system AND PRESENTATION

SCDT44 – Assignment 2

Table of Contents

[The Database Requirements 1](#_Toc40264387)

[Relational Database Justification 1](#_Toc40264388)

[Redundant Data 1](#_Toc40264389)

[Future Preparation 2](#_Toc40264390)

[Hierarchical Databases 2](#_Toc40264391)

[Dataset Normalisation 2](#_Toc40264392)

[Database Design 5](#_Toc40264393)

[Entity Relationship Diagram 5](#_Toc40264394)

[Data Flow Diagram 5](#_Toc40264395)

[Query Design and Testing 6](#_Toc40264396)

[Bibliography 7](#_Toc40264397)

[Database SQL 8](#_Toc40264398)

# The Database Requirements

It is important to meet the requirements laid out by the client when developing a database. There are several features and details that need to be accessible and useable on the system for it to meet them. This section of the documentation will go over the requirements and what each entails.

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| View all patients and their Insurance Number | Users are able to view and sort all patient names along with their Insurance Number |
| Filter by Insurance Number | Users can filter patient information by their Insurance Number’s content |
| Patient’s Staff | Along side the patient information, the staff that are assigned to each patient can be viewed |
| Medication Assignment | Users can view where medication has been prescribed, with an appointment either with or without a staff member |
| Staff Training | Users can view what specialisation staff have been trained it; results can be filtered by training type, such as all those in Popology |
| Edit Staff Details | Queries can be used to edit the details of staff members, such as names. |
| Patient Appointment History | Users can see which patients have had appointments with different staff types, such as doctors |
| Procedure costs | Users can have different procedure types, such as Lineology’s, costs displayed and totalled |
| Surgery Costs | The cost of different surgery types (Minor or Major) can be totalled |
| Procedure cost Averages | The average cost for different procedure types can be calculated |

# Relational Database Justification

While there are several different database types that can be used for the development of the hospital system, I believe that the use of a Relational model is best for efficiency and the features offered.

## Redundant Data

With the large amount of information stored on the hospital database, it can become easy for data to be repeated unnecessarily. With repeated data, comes the chance of inconsistencies with the relevance of the data, with only some instances of the repeated information being up-to-date, leaving some to hold inaccurate information on details such as contact numbers and schedules.

Redundant data in a database can lead to the wrong information being used that, especially in the context of a hospital system, can make the process of completing daily tasks more tedious than necessary. It is important for hospitals and medical centres to have the correct contact information for their patients, as without easy means to contact their patients, medical staff cannot provide any necessary support or information as soon as it is needed **(Does your GP have your correct contact information?, 2015)**.

Using relational databases can help to reduce/remove any redundant data present **(Kroenke, 2005)**. This is because of how these databases handle the use of the same information in more than one location. Through the use of links between tables (done through foreign and primary keys), information on one table can be passed to another for use in queries, removing the need for the same information to be recorded twice across the two tables. For example, as can be seen in the normalised data in the section below, through the use of matching AppointmentIDs, information from separate tables can be grouped together into a single result with their corresponding ID.

## Future Preparation

Another benefit to the use of relational databases is the adaptability for future entries into the system or more properties/columns being added to a table. Provided that unique identifiers (such as IDs) remain the same, more columns and details can be added to existing values with minimal risk of breaking processes or calculations that are already in place. With the unique identifiers being referenced in other tables, any new information added instantly becomes available for being passed and used elsewhere in the system. This remains true for any new entries that are added to a table. With the ‘Select \*’ query that returns all information present in the table, any new values are automatically included, ensuring that the displayed content always contains all information available **(SQL SELECT Statement, n.d.)**.

## Hierarchical Databases

Alternative to relational databases, the hierarchical model has its own features and techniques that can be useful for creating efficient systems. However, I believe that this model is not as effective for use in the hospital database, due to its handling of data transfer. As the name suggests, the sorting and connections between data is managed based on a hierarchy, with data building off those previously established. With the need for data to build off pre-existing content, the database becomes less adaptable to change. With it being harder to insert new/modify existing values and columns due to data in the system that is built off and requiring its current state. This issue is avoided in relational Databases with the use of primary/foreign keys, and so is a better fit to help keep the hospital running on the system even when changes need to be made.

# Dataset Normalisation

*The Doctors Table The Nurses Table*

*The Physicians Table The Medication Table*

*The Patient Table*



*The Appointment Table*

*The Appointment-Procedure Joiner Table*

*The Appointment-Staff Joiner Table*

|  |  |  |  |
| --- | --- | --- | --- |
| **PatientID** | **NurseID** | **PhysicianID** | **DoctorID** |
| 1 | 1 | 1 | *Null* |
| 2 | *Null* | 2 | 4 |
| 3 | 2 | *Null* | 1 |
| 4 | 3 | *Null* | 2 |
| 5 | 4 | *Null* | 3 |
| 6 | *Null* | *Null* | *Null* |
| 7 | *Null* | *Null* | *Null* |
| 8 | *Null* | *Null* | *Null* |
| 9 | *Null* | *Null* | *Null* |
| 10 | *Null* | *Null* | *Null* |
| 9 | *Null* | *Null* | *Null* |
| 10 | *Null* | *Null* | *Null* |
| 11 | *Null* | 3 | *Null* |
| 12 | *Null* | 4 | *Null* |
| 13 | *Null* | 5 | *Null* |
| 14 | *Null* | 6 | *Null* |
| 15 | *Null* | 7 | *Null* |

*The Staff-Patient Joiner Table*

# Database Design

## Entity Relationship Diagram

*The Entity Relationship Diagram for the Hospital Database*

## Data Flow Diagram

*The Data Flow Diagram for the Hospital Database*

# Query Design and Testing

This section of the report displays the database’s implemented queries in both the design and the execution stages, as well as if it meets the requirements laid out by the client.

|  |  |  |
| --- | --- | --- |
| **Query** | **Design** | **Testing** |
| Names of all patients and corresponding insurance number in alphabetical order by Name. |  |  |
| Names of all patients whose insurance ID ends in 42 into ascending order. |  |  |
| Full Name, address and name of staff of all patients. |  |  |
| All the staff details who have prescribed medication without an appointment by a Staff Member |  |  |
| All staff names who have been trained in Popology. |  |  |
| Change the name of physician Allan Fitzgerald to Allan Cross |  |  |
| Names of patients that have or have had appointments with a doctor. |  |  |
| The count and total cost of all Lineology procedures in descending order. |  |  |
| The sum cost of all minor surgery procedures for all patients. |  |  |
| The average cost of a Boneology Procedure in ascending order. |  |  |

# Bibliography

Gloucestershireccg.nhs.uk. 2015. *Does Your GP Have Your Correct Contact Information?*. [online] Available at: <https://www.gloucestershireccg.nhs.uk/does-your-gp-have-your-correct-contact-information/> [Accessed 28 April 2020].

Kroenke, D.M., 2005. Beyond the relational database model. *Computer*, *38*(5), pp.89-90.

W3schools.com. n.d. *SQL SELECT Statement*. [online] Available at: <https://www.w3schools.com/sql/sql\_select.asp> [Accessed 28 April 2020].

# Database SQL

*-- phpMyAdmin SQL Dump*

*-- version 4.9.1*

*-- https://www.phpmyadmin.net/*

*--*

*-- Host: 127.0.0.1*

*-- Generation Time: May 08, 2020 at 02:21 PM*

*-- Server version: 10.4.8-MariaDB*

*-- PHP Version: 7.3.11*

**SET** SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

**SET** AUTOCOMMIT = 0;

**START** TRANSACTION;

**SET** time\_zone = "+00:00";

*/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/*;

*/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/*;

*/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/*;

*/\*!40101 SET NAMES utf8mb4 \*/*;

*--*

*-- Database: `scdt44 cw2`*

*--*

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `appointment`*

*--*

**CREATE** **TABLE** `appointment` (

`AppointmentID` int(11) **NOT** **NULL**,

`PatientID` int(11) **NOT** **NULL**,

`Date\_of\_Appointment` date **NOT** **NULL**,

`Start\_Time` time **NOT** **NULL**,

`End\_Time` time **NOT** **NULL**,

`Duration\_(Hours)` double **NOT** **NULL**,

`Room\_Number` int(11) **NOT** **NULL**,

`Availability` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `appointment`*

*--*

**INSERT** **INTO** `appointment` (`AppointmentID`, `PatientID`, `Date\_of\_Appointment`, `Start\_Time`, `End\_Time`, `Duration\_(Hours)`, `Room\_Number`, `Availability`) **VALUES**

(1, 1, '2020-12-14', '10:00:00', '12:00:00', 2, 1, 'Unavailable'),

(2, 2, '2020-04-20', '11:00:00', '13:00:00', 2, 2, 'Unavaliable\r\n'),

(3, 3, '2020-04-20', '11:00:00', '13:00:00', 2, 3, 'Unavaliable\r\n'),

(4, 4, '2020-12-14', '15:00:00', '16:00:00', 1, 4, 'Unavaliable\r\n'),

(5, 5, '2020-03-15', '09:00:00', '11:00:00', 2, 5, 'Unavaliable\r\n'),

(6, 6, '2020-12-14', '17:00:00', '18:00:00', 1, 6, 'Unavaliable\r\n'),

(7, 7, '2020-03-15', '18:00:00', '20:00:00', 2, 7, 'Unavaliable'),

(8, 8, '2020-12-14', '15:00:00', '15:30:00', 0.5, 8, 'Unavaliable'),

(9, 9, '2020-12-14', '18:00:00', '20:00:00', 2, 9, 'Unavaliable'),

(10, 15, '2020-03-15', '10:00:00', '16:00:00', 6, 10, 'Unavaliable'),

(11, 8, '2020-12-16', '15:00:00', '15:30:00', 0.5, 8, 'Unavaliable'),

(12, 9, '2020-12-16', '18:00:00', '20:00:00', 2, 9, 'Unavaliable');

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `appointment-procedures`*

*--*

**CREATE** **TABLE** `appointment-procedures` (

`AppointmentID` int(11) **NOT** **NULL**,

`Procedures` text **NOT** **NULL**,

`Cost` double **NOT** **NULL**,

`MedicationID` int(11) **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `appointment-procedures`*

*--*

**INSERT** **INTO** `appointment-procedures` (`AppointmentID`, `Procedures`, `Cost`, `MedicationID`) **VALUES**

(1, 'Minor Surgery\r\n', 600, 1),

(2, 'Minor Surgery\r\n', 600, 2),

(3, 'Minor Surgery\r\n', 800, 3),

(4, 'Minor Surgery\r\n', 900, 1),

(5, 'Minor Surgery\r\n', 600, 2),

(6, 'Minor Surgery\r\n', 600, 3),

(7, 'Major Surgery\r\n', 1000, 1),

(8, 'Minor Surgery\r\n', 250, 2),

(9, 'Major Surgery\r\n', 1000, 3),

(10, 'Major Surgery\r\n', 2000, 1),

(11, 'Minor Surgery\r\n', 750, 2),

(12, 'Major Surgery\r\n', 1250, 3);

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `appointment-staff`*

*--*

**CREATE** **TABLE** `appointment-staff` (

`AppointmentID` int(11) **DEFAULT** **NULL**,

`NurseID` int(11) **DEFAULT** **NULL**,

`PhysicianID` int(11) **DEFAULT** **NULL**,

`DoctorID` int(11) **DEFAULT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `appointment-staff`*

*--*

**INSERT** **INTO** `appointment-staff` (`AppointmentID`, `NurseID`, `PhysicianID`, `DoctorID`) **VALUES**

(1, 1, 1, **NULL**),

(2, **NULL**, 2, **NULL**),

(3, 2, **NULL**, 1),

(4, 3, **NULL**, 2),

(5, 4, **NULL**, 3),

(6, **NULL**, **NULL**, **NULL**),

(7, **NULL**, **NULL**, **NULL**),

(8, **NULL**, **NULL**, **NULL**),

(9, **NULL**, **NULL**, **NULL**),

(10, **NULL**, 7, **NULL**),

(11, **NULL**, **NULL**, **NULL**),

(12, **NULL**, **NULL**, **NULL**);

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `doctor`*

*--*

**CREATE** **TABLE** `doctor` (

`DoctorID` int(11) **NOT** **NULL**,

`Doctor\_Name` text **NOT** **NULL**,

`Trained\_In` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `doctor`*

*--*

**INSERT** **INTO** `doctor` (`DoctorID`, `Doctor\_Name`, `Trained\_In`) **VALUES**

(1, 'Rachel Mathis\r\n', 'Lineology'),

(2, 'Chester Bains\r\n', 'Jamology'),

(3, 'Aditi Wainwright\r\n', 'Boneology'),

(4, 'Mia-Rose Espinosa\r\n', 'Popology');

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `medication`*

*--*

**CREATE** **TABLE** `medication` (

`MedicationID` int(11) **NOT** **NULL**,

`Medication` text **NOT** **NULL**,

`Description` text **NOT** **NULL**,

`Brand` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `medication`*

*--*

**INSERT** **INTO** `medication` (`MedicationID`, `Medication`, `Description`, `Brand`) **VALUES**

(1, 'Paracetamol', 'Pain Killer\r\n', 'PainResolve\r\n'),

(2, 'Ibruprofen\r\n', 'Heart Burn\r\n', 'HeartRelief\r\n'),

(3, 'Zapain\r\n', 'Pain Killer\r\n', 'Zap!\r\n');

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `nurse`*

*--*

**CREATE** **TABLE** `nurse` (

`NurseID` int(11) **NOT** **NULL**,

`Nurse\_Name` text **NOT** **NULL**,

`Trained\_In` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `nurse`*

*--*

**INSERT** **INTO** `nurse` (`NurseID`, `Nurse\_Name`, `Trained\_In`) **VALUES**

(1, 'Edwin Huff\r\n', 'Boneology'),

(2, 'Zayyan Saunders\r\n', 'Lineology'),

(3, 'Isobelle Hinton\r\n', 'Boneology'),

(4, 'Bobby Phelps\r\n', 'Jamology');

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `patient`*

*--*

**CREATE** **TABLE** `patient` (

`PatientID` int(11) **NOT** **NULL**,

`Name` text **NOT** **NULL**,

`Address` text **NOT** **NULL**,

`Phone` text **NOT** **NULL**,

`Insurance\_Number` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `patient`*

*--*

**INSERT** **INTO** `patient` (`PatientID`, `Name`, `Address`, `Phone`, `Insurance\_Number`) **VALUES**

(1, 'Barry Gates\r\n', '66 Rockingham Street, Barnsley, S71 1JR\r\n', '01743 887228\r\n', 'BAZ12342\r\n'),

(2, 'Stanislaw Bassett\r\n', '53 Queen Street, Rushden, NN10 0AY\r\n', '01743 887223\r\n', 'STA12545\r\n'),

(3, 'Kalvin Dickens\r\n', '1 Goudhurst Close, Maidstone, ME16 8QT\r\n', '01743 833228\r\n', 'KAL12342\r\n'),

(4, 'Kaydee Stuart\r\n', '9 Siskin Close, Horsham, RH12 5YR\r\n', '01743 882328\r\n', 'KAY13345\r\n'),

(5, 'Hazel Mcneil\r\n', '21 Doric Road, Liverpool, L13 3DS\r\n', '01743 187222\r\n', 'HAZ14355\r\n'),

(6, 'Sanaya Mendoza\r\n', 'Flat 10, Pine Lodge, Leigh Corner, Cobham, KT11 2RB\r\n', '01743 837428\r\n', 'SAN12645\r\n'),

(7, 'Yehuda Allan\r\n', '18 Waterside Lodges, Elland Road, Elland, HX5 9AQ\r\n', '01743 227226\r\n', 'YEH12345\r\n'),

(8, 'Patricia Mccarthy\r\n', '5 Crabtree Close, Danesmoor, S45 9BH\r\n', '01743 887255\r\n', 'PAT12345\r\n'),

(9, 'Kaan Landry\r\n', '2 Larchcroft, Chatham, ME5 0NL\r\n', '01743 667228\r\n', 'KAA12342\r\n'),

(10, 'River Gale\r\n', '7 Prospect Place, South Brent, TQ10 9AH\r\n', '01743 882311\r\n', 'GAL12345\r\n'),

(11, 'Sky Rooney\r\n', '24 Merrimans Hill Road, Worcester, WR3 8LW\r\n', '01743 127222\r\n', 'ROO12345\r\n'),

(12, 'Hallam Shields\r\n', '2 - 3 Sunny Bank, Cragg Vale, HX7 5SL\r\n', '01743 835528\r\n', 'SHI15545\r\n'),

(13, 'Chantal Page\r\n', 'Pytte Cottage, Clyst St George, EX3 0NU\r\n', '01743 226626\r\n', 'PAG12645\r\n'),

(14, 'Kurtis Rivera\r\n', 'Park House, Church Street, Talgarth, LD3 0DW\r\n', '01743 887233\r\n', 'RIV17345\r\n'),

(15, 'Dru Battle\r\n', 'Flat 6, Aperture House, 6 Honduras Street, London, EC1Y 0TH\r\n', '01743 667265\r\n', 'BAT12345\r\n');

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `patient-staff`*

*--*

**CREATE** **TABLE** `patient-staff` (

`PatientID` int(11) **NOT** **NULL**,

`DoctorID` int(11) **DEFAULT** **NULL**,

`NurseID` int(11) **DEFAULT** **NULL**,

`PhysicianID` int(11) **DEFAULT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `patient-staff`*

*--*

**INSERT** **INTO** `patient-staff` (`PatientID`, `DoctorID`, `NurseID`, `PhysicianID`) **VALUES**

(1, **NULL**, 1, 1),

(2, 4, **NULL**, 2),

(3, 1, 2, **NULL**),

(4, 2, 3, **NULL**),

(5, 3, 4, **NULL**),

(6, **NULL**, **NULL**, **NULL**),

(7, **NULL**, **NULL**, **NULL**),

(8, **NULL**, **NULL**, **NULL**),

(9, **NULL**, **NULL**, **NULL**),

(10, **NULL**, **NULL**, **NULL**),

(11, **NULL**, **NULL**, 3),

(12, **NULL**, **NULL**, 4),

(13, **NULL**, **NULL**, 5),

(14, **NULL**, **NULL**, 6),

(15, **NULL**, **NULL**, 7);

*-- --------------------------------------------------------*

*--*

*-- Table structure for table `physician`*

*--*

**CREATE** **TABLE** `physician` (

`PhysicianID` int(11) **NOT** **NULL**,

`Physician\_Name` text **NOT** **NULL**,

`Trained\_In` text **NOT** **NULL**

) ENGINE=InnoDB **DEFAULT** CHARSET=utf8mb4;

*--*

*-- Dumping data for table `physician`*

*--*

**INSERT** **INTO** `physician` (`PhysicianID`, `Physician\_Name`, `Trained\_In`) **VALUES**

(1, 'Bob Harris\r\n', 'Kingstology'),

(2, 'Allan Cross', 'Lineology'),

(3, 'Jeremy Shipton\r\n', 'Kingstology'),

(4, 'Charles Ulter\r\n', 'Jamology'),

(5, 'Christina Bishop', 'Popology'),

(6, 'Sarah Smith\r\n', 'Popology'),

(7, 'William. B Hampton \r\n', 'Kingstology');

*--*

*-- Indexes for dumped tables*

*--*

*--*

*-- Indexes for table `appointment`*

*--*

**ALTER** **TABLE** `appointment`

**ADD** **UNIQUE** **KEY** `AppointmentID` (`AppointmentID`),

**ADD** **KEY** `patient-link` (`PatientID`);

*--*

*-- Indexes for table `appointment-procedures`*

*--*

**ALTER** **TABLE** `appointment-procedures`

**ADD** **UNIQUE** **KEY** `AppointmentID` (`AppointmentID`),

**ADD** **KEY** `medication-link` (`MedicationID`);

*--*

*-- Indexes for table `appointment-staff`*

*--*

**ALTER** **TABLE** `appointment-staff`

**ADD** **UNIQUE** **KEY** `AppointmentID` (`AppointmentID`),

**ADD** **KEY** `doctor-**connect**` (`DoctorID`),

**ADD** **KEY** `nurse-**connect**` (`NurseID`),

**ADD** **KEY** `physician-**connect**` (`PhysicianID`);

*--*

*-- Indexes for table `doctor`*

*--*

**ALTER** **TABLE** `doctor`

**ADD** **UNIQUE** **KEY** `DoctorID` (`DoctorID`);

*--*

*-- Indexes for table `medication`*

*--*

**ALTER** **TABLE** `medication`

**ADD** **UNIQUE** **KEY** `MedicationID` (`MedicationID`);

*--*

*-- Indexes for table `nurse`*

*--*

**ALTER** **TABLE** `nurse`

**ADD** **UNIQUE** **KEY** `NurseID` (`NurseID`);

*--*

*-- Indexes for table `patient`*

*--*

**ALTER** **TABLE** `patient`

**ADD** **UNIQUE** **KEY** `PatientID` (`PatientID`);

*--*

*-- Indexes for table `patient-staff`*

*--*

**ALTER** **TABLE** `patient-staff`

**ADD** **PRIMARY** **KEY** (`PatientID`),

**ADD** **KEY** `doctor-**check**` (`DoctorID`),

**ADD** **KEY** `nurse-**check**` (`NurseID`),

**ADD** **KEY** `physician-**check**` (`PhysicianID`);

*--*

*-- Indexes for table `physician`*

*--*

**ALTER** **TABLE** `physician`

**ADD** **UNIQUE** **KEY** `PhysicianID` (`PhysicianID`);

*--*

*-- Constraints for dumped tables*

*--*

*--*

*-- Constraints for table `appointment`*

*--*

**ALTER** **TABLE** `appointment`

**ADD** **CONSTRAINT** `patient-link` **FOREIGN** **KEY** (`PatientID`) **REFERENCES** `patient` (`PatientID`);

*--*

*-- Constraints for table `appointment-procedures`*

*--*

**ALTER** **TABLE** `appointment-procedures`

**ADD** **CONSTRAINT** `appointment-link` **FOREIGN** **KEY** (`AppointmentID`) **REFERENCES** `appointment` (`AppointmentID`),

**ADD** **CONSTRAINT** `medication-link` **FOREIGN** **KEY** (`MedicationID`) **REFERENCES** `medication` (`MedicationID`);

*--*

*-- Constraints for table `appointment-staff`*

*--*

**ALTER** **TABLE** `appointment-staff`

**ADD** **CONSTRAINT** `appointment-**connect**` **FOREIGN** **KEY** (`AppointmentID`) **REFERENCES** `appointment` (`AppointmentID`) **ON** **DELETE** **NO** ACTION **ON** **UPDATE** **NO** ACTION,

**ADD** **CONSTRAINT** `doctor-**connect**` **FOREIGN** **KEY** (`DoctorID`) **REFERENCES** `doctor` (`DoctorID`) **ON** **DELETE** **NO** ACTION **ON** **UPDATE** **NO** ACTION,

**ADD** **CONSTRAINT** `nurse-**connect**` **FOREIGN** **KEY** (`NurseID`) **REFERENCES** `nurse` (`NurseID`),

**ADD** **CONSTRAINT** `physician-**connect**` **FOREIGN** **KEY** (`PhysicianID`) **REFERENCES** `physician` (`PhysicianID`);

*--*

*-- Constraints for table `patient-staff`*

*--*

**ALTER** **TABLE** `patient-staff`

**ADD** **CONSTRAINT** `doctor-**check**` **FOREIGN** **KEY** (`DoctorID`) **REFERENCES** `doctor` (`DoctorID`) **ON** **DELETE** **NO** ACTION **ON** **UPDATE** **NO** ACTION,

**ADD** **CONSTRAINT** `nurse-**check**` **FOREIGN** **KEY** (`NurseID`) **REFERENCES** `nurse` (`NurseID`) **ON** **DELETE** **NO** ACTION **ON** **UPDATE** **NO** ACTION,

**ADD** **CONSTRAINT** `patient-verify` **FOREIGN** **KEY** (`PatientID`) **REFERENCES** `patient` (`PatientID`),

**ADD** **CONSTRAINT** `physician-**check**` **FOREIGN** **KEY** (`PhysicianID`) **REFERENCES** `physician` (`PhysicianID`) **ON** **DELETE** **NO** ACTION **ON** **UPDATE** **NO** ACTION;

**COMMIT**;

*/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/*;

*/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/*;

*/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/*;