Dr B R Ambedkar National Institute of Technology
Jalandhar-144011, Punjab, India.

July-Dec, 2021

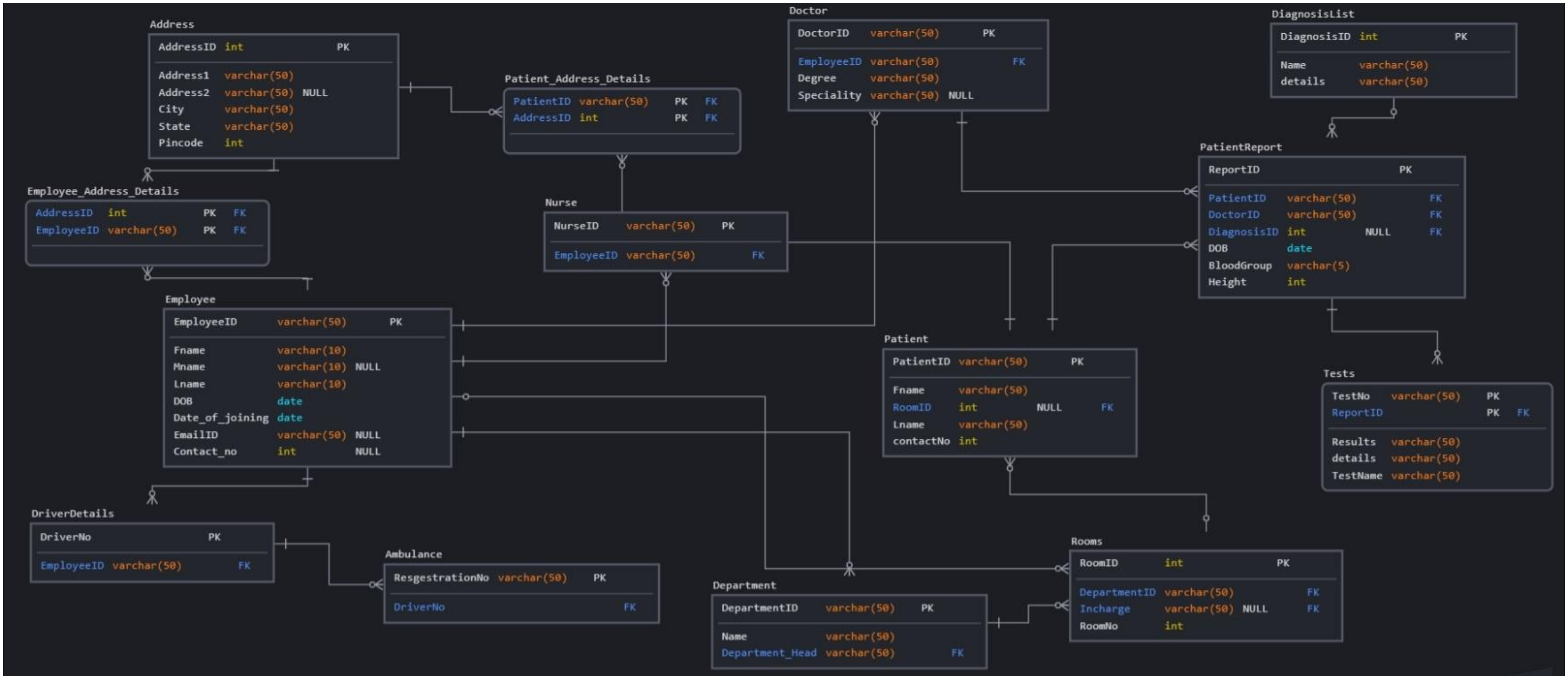
# Database Management Systems

## Assignment 1

(Hospital Management System)

Submitted to:
Dr. Rajneesh Rani

Submitted by:
Abhay (20103001)
Adarsh (20103003)
Akshay (20103010)
Amritdeep (20103013)



Address

| | | | | |
|---|---|---|---|---|
| AddressID | int | | PK | |
| Address1 | varchar(50) | | | |
| Address2 | varchar(50) | NULL | | |
| City | varchar(50) | | | |
| State | varchar(50) | | | |
| Pincode | int | | | |

Patient_Address_Details

| | | | | |
|---|---|---|---|---|
| PatientID | varchar(50) | | PK | FK |
| AddressID | int | | PK | FK |

Doctor

| | | | | |
|---|---|---|---|---|
| DoctorID | varchar(50) | | PK | |
| EmployeeID | varchar(50) | | | FK |
| Degree | varchar(50) | | | |
| Speciality | varchar(50) | NULL | | |

DiagnosisList

| | | | | |
|---|---|---|---|---|
| DiagnosisID | int | | PK | |
| Name | varchar(50) | | | |
| details | varchar(50) | | | |

PatientReport

| | | | | |
|---|---|---|---|---|
| ReportID | | | PK | |
| PatientID | varchar(50) | | | FK |
| DoctorID | varchar(50) | | | FK |
| DiagnosisID | int | NULL | | FK |
| DOB | date | | | |
| BloodGroup | varchar(5) | | | |
| Height | int | | | |

Employee_Address_Details

| | | | | |
|---|---|---|---|---|
| AddressID | int | | PK | FK |
| EmployeeID | varchar(50) | | PK | FK |

Nurse

| | | | | |
|---|---|---|---|---|
| NurseID | varchar(50) | | PK | |
| EmployeeID | varchar(50) | | | FK |

Employee

| | | | | |
|---|---|---|---|---|
| EmployeeID | varchar(50) | | PK | |
| Fname | varchar(10) | | | |
| Mname | varchar(10) | NULL | | |
| Lname | varchar(10) | | | |
| DOB | date | | | |
| Date_of_joining | date | | | |
| EmailID | varchar(50) | NULL | | |
| Contact_no | int | NULL | | |

Patient

| | | | | |
|---|---|---|---|---|
| PatientID | varchar(50) | | PK | |
| Fname | varchar(50) | | | |
| RoomID | int | NULL | | FK |
| Lname | varchar(50) | | | |
| contactNo | int | | | |

Tests

| | | | | |
|---|---|---|---|---|
| TestNo | varchar(50) | | PK | |
| ReportID | | | PK | FK |
| Results | varchar(50) | | | |
| details | varchar(50) | | | |
| TestName | varchar(50) | | | |

DriverDetails

| | | | | |
|---|---|---|---|---|
| DriverNo | | | PK | |
| EmployeeID | varchar(50) | | | FK |

Ambulance

| | | | | |
|---|---|---|---|---|
| ResgestrationNo | varchar(50) | | PK | |
| DriverNo | | | | FK |

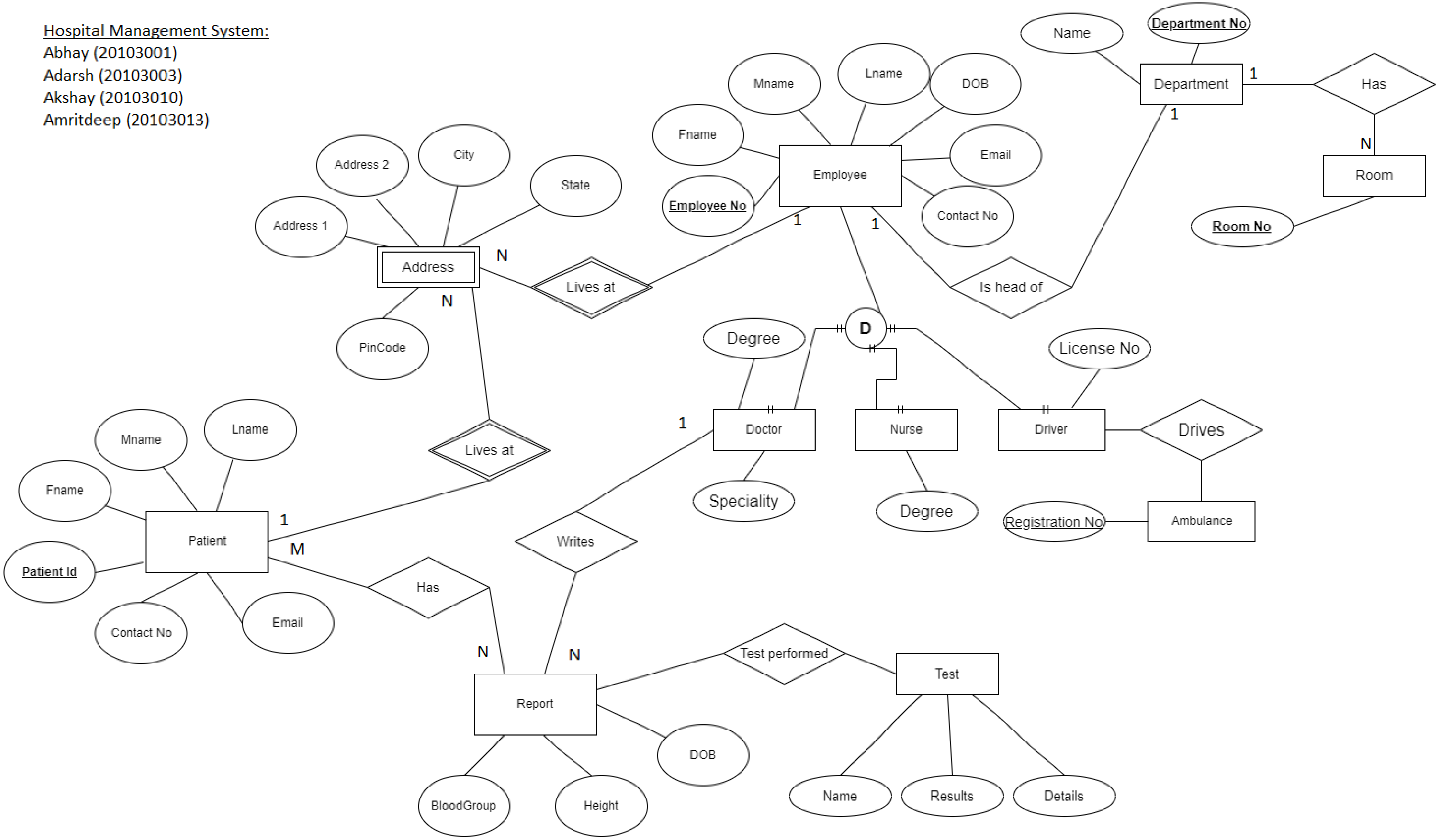
Department

| | | | | |
|---|---|---|---|---|
| DepartmentID | varchar(50) | | PK | |
| Name | varchar(50) | | | |
| Department_Head | varchar(50) | | | FK |

Rooms

| | | | | |
|---|---|---|---|---|
| RoomID | int | | PK | |
| DepartmentID | varchar(50) | | | FK |
| Incharge | varchar(50) | NULL | | FK |
| RoomNo | int | | | |

Hospital Management System:
Abhay (20103001)
Adarsh (20103003)
Akshay (20103010)
Amritdeep (20103013)



```
create database hospital; --creating our database called 'Hospital'
use hospital; --telling mysql to use our 'Hospital' database


CREATE TABLE Employee --creating table 'Employee'
(EmployeeID     varchar(50) NOT NULL ,
 Fname           varchar(10) NOT NULL ,
 Mname           varchar(10) NULL ,
 Lname           varchar(10) NULL ,
 DOB             date NOT NULL ,
 Date_of_joining date NOT NULL ,
 EmailID         varchar(50) NULL ,
 Contact_no      int NULL ,
 primary key (EmployeeID), -- primary key is Employee id
 check (EmployeeID like 'E%' or 'N%' or 'D%') -- this check is to
make sure that employee id starts from either E,N or D
);


create table Doctor(  --creating table 'Doctor'
DoctorID   varchar(50) NOT NULL ,
Degree     varchar(50) NOT NULL ,
Speciality varchar(50) NULL,
primary key(doctorID), -- primary key is doctor id
foreign key(doctorID) references Employee(EmployeeID), -- doctor id
referes employee id i.e. doctor id must be in employee id column
check (DoctorID like 'D%') -- this check is to make sure that people
with employee id starting from 'D' can be part of this table
 );


create table Nurse( --creating table 'Nurse'
NurseID   varchar(50) NOT NULL,
shift char NOT NULL,
primary key(NurseID), -- primary key is Nurse id
foreign key(NurseID) references Employee(EmployeeID), -- Nurse id
referes employee id i.e. Nurse id must be in employee id column
check (NurseID like 'N%'), -- this check is to make sure that people
with employee id starting from 'N' can be part of this table
```

```
check (shift in ('M','E','N')) -- this check is to make sure that a
nurse shift can only be 'M','E' or 'N' i.e. morning, evening or night
 );


create table Driver( -- creating table Driver
DriverID   varchar(50) NOT NULL ,
license varchar(10) NOT NULL,
shift char NOT NULL,
primary key (driverID), -- primary key is driver id
foreign key(DriverID) references Employee(EmployeeID), -- this check
is to make sure that people with employee id starting from 'N' can be
part of this table
check (shift in ('M','E','N')) -- this check is to make sure that a
nurse shift can only be 'M','E' or 'N' i.e. morning, evening or night
 );


 create table Ambulance( -- creating table Ambulance
 RegistrationNO varchar(10) NOT NULL,
 primary key(RegistrationNO), -- primary key is Registration no
 DriverID varchar(50) NOT NULL,
 foreign key(driverID) references Driver(DriverID) -- driver id must
be a part of driver table to be enter in this table
 );


 create table department( -- creating table department
 departmentID varchar(10) NOT NULL,
 primary key(departmentID), -- primary key is department id
 name varchar(10) NOT NULL,
 departmentHead varchar(50) NOT NULL,
 foreign key(departmentHead) references doctor(doctorID) --
department head can only be a doctor and from doctor table
 );


 create table room( -- create table room
 roomno int Not null,
```

```sql
 primary key(roomno), --primary key is roomno
 departmentID varchar(10) NOT NULL,
 foreign key(departmentID) references department(departmentID), --
department of room must be a part of department table
 roomIncharge varchar(50) NULL,
 foreign key (roomIncharge) references doctor(doctorID) -- room
incharge must be doctor from doctor table
 );


 create table patient( -- creating patient table
 patientID varchar(20) NOT NULL,
 primary key(patientID), -- primary key is patient id
 Fname           varchar(10) NOT NULL ,
 Mname           varchar(10) NULL ,
 Lname           varchar(10) NULL ,
 contactNo int NULL
 );


 create table ward( -- creating table ward
 roomNo int NOT NULL,
 primary key(roomno),-- primary key is roomno
 foreign key(roomNO) references room(roomNO), -- roomno must be part
of room table
 patientID varchar(20) NULL,
 foreign key (patientID)  references patient(patientID)-- patient
must be part of patient table
 );


 CREATE TABLE Address( -- creating table address this would contain
address of both employees and patient
 AddressID int NOT NULL ,
 Address1  varchar(50) NOT NULL ,
 Address2  varchar(50) NULL ,
 City      varchar(50) NOT NULL ,
 State     varchar(50) NOT NULL ,
 Pincode   int NOT NULL,
 primary key(addressID) -- primary key is address ID
 );


 create table Emp_Addr_Dtls( -- this table Emp_Addr_Dtls links table employees with their respective addresses
 addressID int NOT NULL,
 primary key(addressID), --primary key is address id
 EmployeeID varchar(50) NOT NULL,
 foreign key (addressID) references address(addressID),-- address must be a part of address table
 foreign key (employeeID) references employee(employeeID)-- employee must be a part of employee table
);


create table patient_Addr_Dtls(-- this table patient_Addr_Dtls links table patient with their respective addresses
 addressID int NOT NULL,
 primary key(addressID),--primary key is address id
 patientID varchar(50) NOT NULL,
 foreign key (addressID) references address(addressID),-- address must be a part of address table
 foreign key (patientID) references patient(patientID)-- patient must be part of patient table
);


CREATE TABLE DiagnosisList(-- this table Diagnosis list contains list of possible diagnosis and their details
 DiagnosisID int NOT NULL ,
 Name        varchar(50) NOT NULL ,
 details     varchar(50) NOT NULL ,
 primary key(diagnosisID));-- primary key is diagnosis ID


 create table patient_report(-- creating table patientID
 reportID int Not null,
 primary key(reportID),-- report ID is primary key
```

```sql
 patientID varchar(20) NOT NULL,
 foreign key(patientID) references patient(patientID), -- patient id entered here must be part of table patient
 diagnosisID int NULL,
 foreign key(diagnosisID) references diagnosisList(diagnosisID),--diagnosis must be from table diagnosisList
 DOB date NOT NULL,
 BloodGrp varchar(3) NOT NULL,
 check (bloodgrp in ('B+','A+','AB+','O+','B-','A-','AB-','O-')),--this check makes sure that correct blood group is entered
 height int Null
 );


 create table patient_test( --this table contains the name, their details and result of tests done on patient
 testno int NOT NULL,
 reportid int NOT NULL,
 primary key(reportid,testno),-- primary key is both report id and test no as one report can have multiple tests
 foreign key (reportid) references patient_report(reportid),-- report id must be inside report table
 testname varchar(20),
 results varchar(20),
 details varchar(20)
 );
```