

**University of Asia pacific**

***Department of Computer Science & Engineering***

Course code: cse-322.

Course title: Database Systems Lab

Sec: A (ii).

Project name: medical college database management system.

Date of Submission date: 18.09.2018

**Submitted by** **submitted to**

REg no. ; 16101031, 16101032, md. Nadeem ahmed

16101044. Assistant professor

University of Asia pacific

|  |  |
| --- | --- |
|  | **MEDICAL COLLEGE DATABASE**  **MANAGEMENT SYSTEM** |
|  |  |

**SUBMITTED BY:-**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Md. Aminul Islam**  **ID: 16101031** | **Zarif Khan mojlish**  **ID: 16101032** | **Md. Al-amin Khan Riad**  **ID: 16101044** |

Table of the content

Page

1. About the project …………………………………………………………4
2. E-r diagram ……………………….…………………………………………... 5
3. Schema diagram ………………………………….………………………... 6
4. Queries & screen shots …………………….………………………… 7
5. Appendix a ……………………………………………………………………. 19
6. Appendix B ………………………...…………………………………………. 21

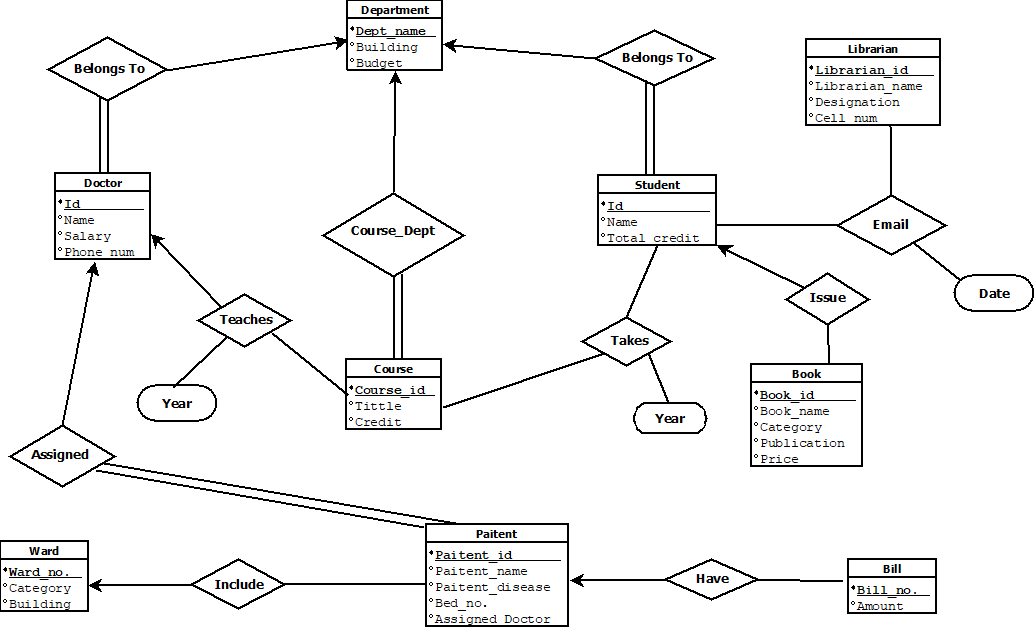
**About the Project:**

A Medical College started its journey in 1999-2000 and with 50 students but now there 200 students. Every year 50 students get allow for admission. Students have ID, name and the total credit they have completed. These students get admitted department-wise. Each department has name, building, budget. There are many courses under every department and courses have ID, title and credit. Students take courses where each course taught by a doctor. Every course is taken by many students. One student can take many courses and similarly, one doctor can teach many courses. All students and doctors must have a department and those departments that are included in this medical college department. Students and doctors also must have taken course year and taught course year respectively.

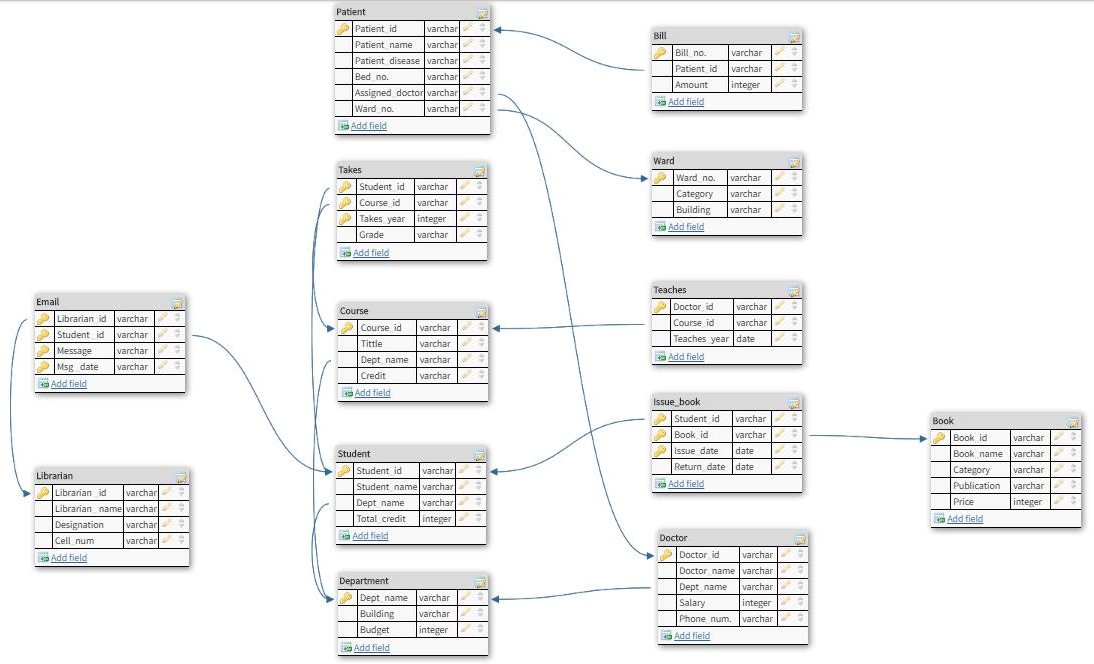
Our medical college is also a reputed hospital. This medical has some ward. Ward has ward no, category and building where its situated. Under these wards there are several beds and each bed for a single patient. Everyday there are many patients gets admitted and get proper treatment also. Every patient has ID, name, disease, bed no and assign doctor that must have every patient. Though every patient has one assign doctor but every single doctor has many patients. Each patient has some bills where bill no and amount are stored.

In this medical college, there is also a well-organized library. In this library there are many types of book where books have ID, name, category, publication, price. Students can issue book and every student can issue maximum two books at a time. They must return that within two weeks. A librarian can send message to a student for returning book or any other purpose. This messages must have date and a librarian has ID, name, designation and cell.

**E-R Diagram:**



**Schema Diagram:**

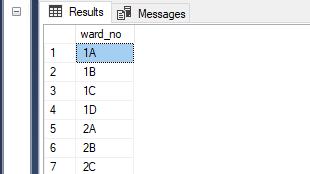


**Queries & Screen Shots:**

**Hospital Part**

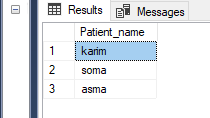
1. Find the ward no of all patient without duplicates.

select distinct ward\_no from Ward



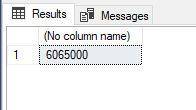
1. Find all patients in 1A no ward.

select Patient\_name from Patient where Ward\_no='1A';



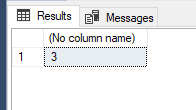
1. Sum of total bill of a patient.

select sum(Amount) from Bill;



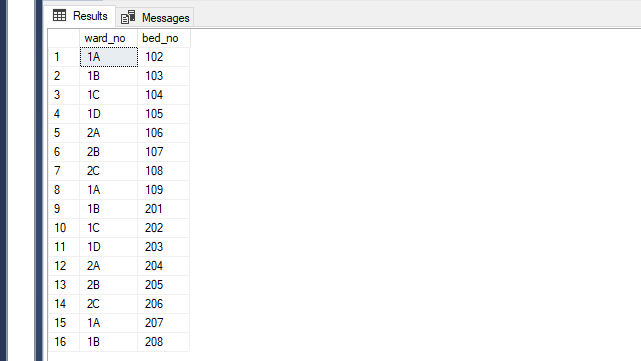
1. Find the total number of doctors whose assigned for patient in the “702”no ward.

select count(Assigned\_doctor) from Patient where Ward\_no='1A';



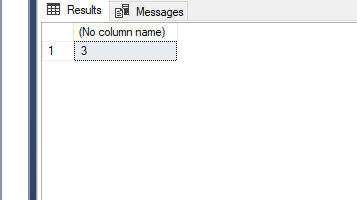
1. List all ward no along with the number of bed no in each ward.

select ward\_no,bed\_no from Patient;



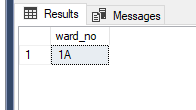
1. Find the total number of patient of “1A” no ward.

select count(distinct Patient\_id) from Patient where Ward\_no='1A';



1. Find all ward no of burn category.

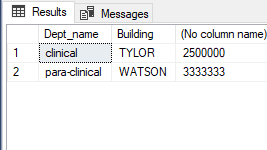
select all ward\_no from Ward where Category='burn unit';



**College Part**

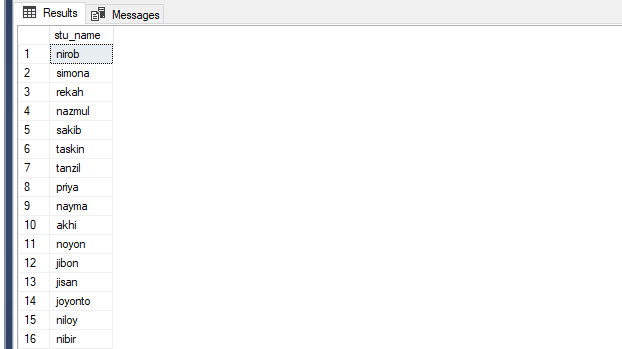
1. Find department name, building, budget/12 as monthly budget.

select Dept\_name,Building,Budget/12 from Department;



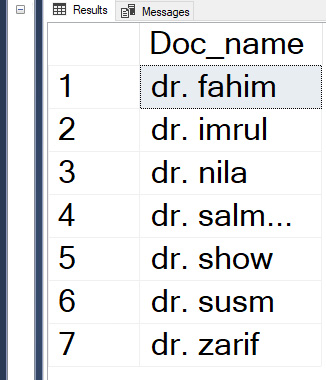
1. Find all students in clinical department with total credit>80

select stu\_name from student where Dept\_name ='clinical' and Total\_credit>40;



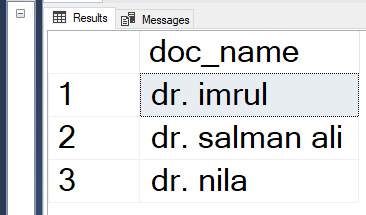
1. Find the names of all doctors who have a higher salary than some doctors in clinical department

Select distinct D.Doc\_name from Doctor as D,Doctor as S where D.Salary>S.Salary and S.Dept\_name = 'clinical';



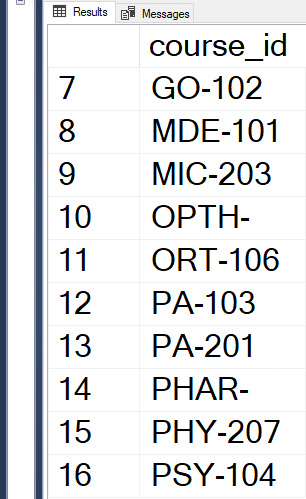
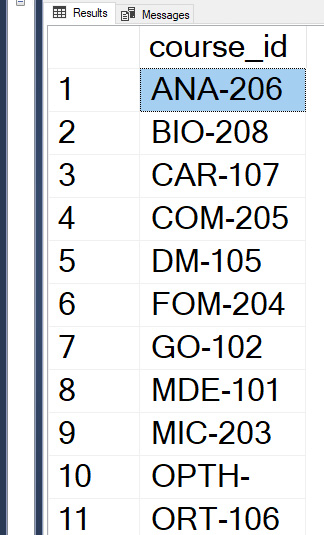
1. Find the names of all doctors with salary between 90000 and 100000.

select doc\_name from Doctor where Salary between 90000 and 100000;



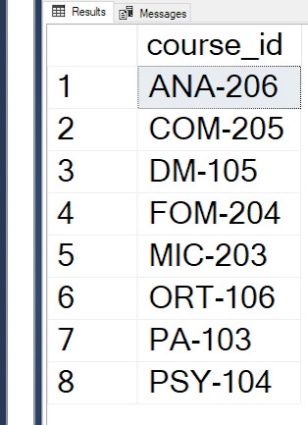
1. Find courses that teaches by doctors in 2011 or in 2012. (do this by union)

(select course\_id from Teaches where Teaches\_year = 2017) union (select course\_id from Teaches where Teaches\_year = 2018);



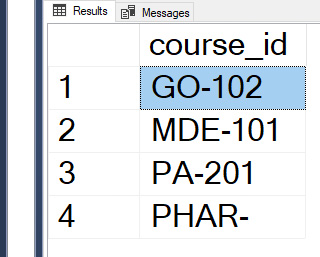
1. Find courses that teaches by doctors in 2009 and in 2010. (do this by intersect)

(select course\_id from Teaches where Teaches\_year = 2017) intersect (select course\_id from Teaches where Teaches\_year = 2018);



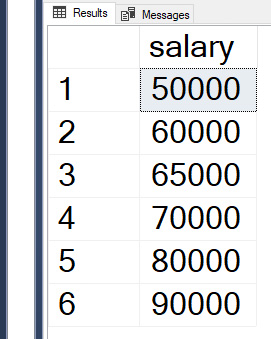
1. Find courses that takes by students in 2009 but not in 2018.(using except)

(select course\_id from Teaches where Teaches\_year = 2017) except (select course\_id from Teaches where Teaches\_year = 2018);



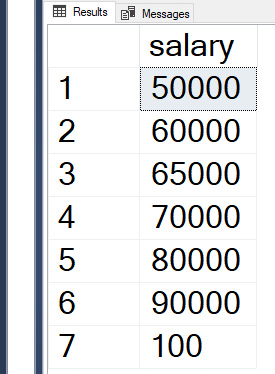
1. Find the salaries of all doctors that are less than the largest salary.

select distinct T.salary from Doctor as T, Doctor as S where T.Salary<S.Salary;



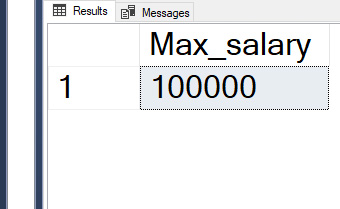
1. Find all the salaries of all doctors.

Select distinct salary from Doctor;



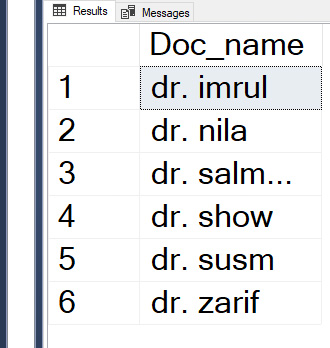
1. Find the maximum salary of doctors.

select max(salary)as Max\_salary from Doctor;



1. Find names of doctor with salary greater than that of some (at least one) doctor in the non-clinical department.

select distinct T.Doc\_name from Doctor as T, Doctor as S where T.Salary>S.Salary and S.Dept\_name = 'para-clinical';



**Library Part**

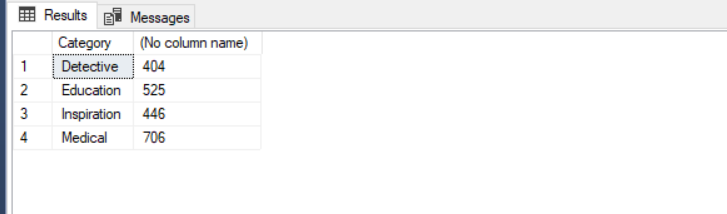
1. Names and average prices of all categories whose average price is greater than 360

select Category, avg (Price)

from Book

group by Category

having avg (Price) > 360;

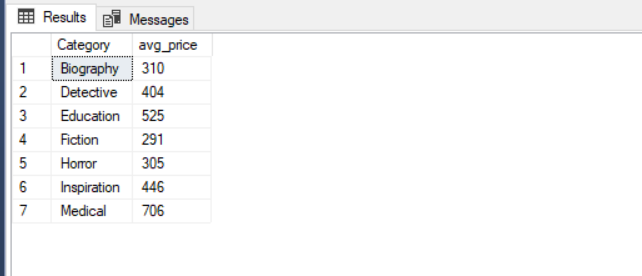


1. Average price of books in each category.

select Category, avg (Price) as avg\_price

from Book

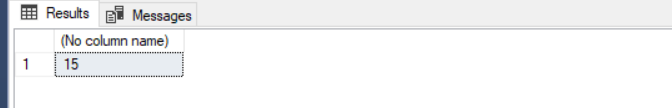
group by Category;



1. Total number of Issue books by students.

select count (\*)

from Issue\_book;

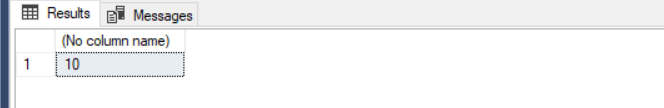


1. Total number of students who issued the books in the year of 2018.

select count(distinct Student\_id)

from Issue\_book

where (select Year(Issue\_date)) = '2018';

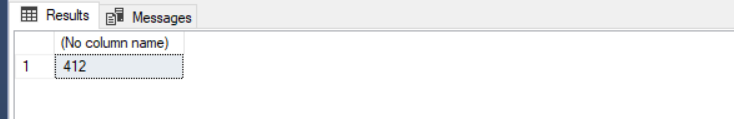


1. Average price of book in the DYD publication.

select avg (price)

from Book

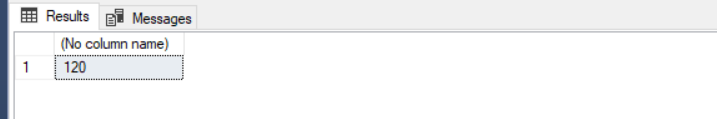
where Publication= 'DYD';



1. Minimum price of book.

select min (price)

from Book;

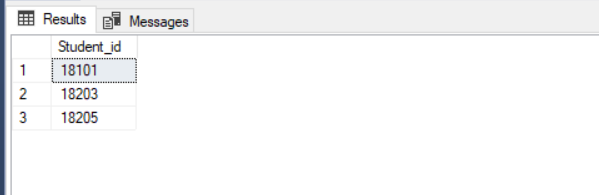


1. Students whose are not return their books.

select Student\_id\_

from Issue\_book\_

where Return\_date is null;



1. Maximum price of books in detective category.

select max (price)

from Book

where Category='detective';

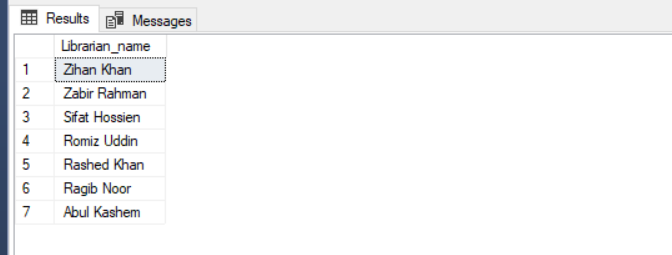


1. List in descending alphabetic order the names of all librarian.

select distinct Librarian\_name

from Librarian

order by Librarian\_name desc

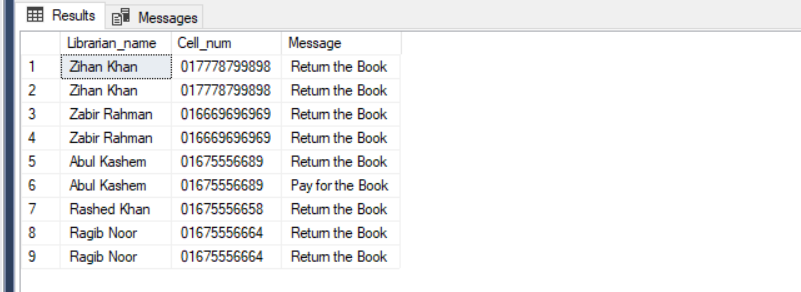


1. Names, contact number of all librarian who have sent some message and the message.

select Librarian\_name,Cell\_num,Message\_

from Librarian, Email

where Librarian.Librarian\_id=Email.Librarian\_id;



1. Names of all books where price is greater than the price of books in the fiction category.

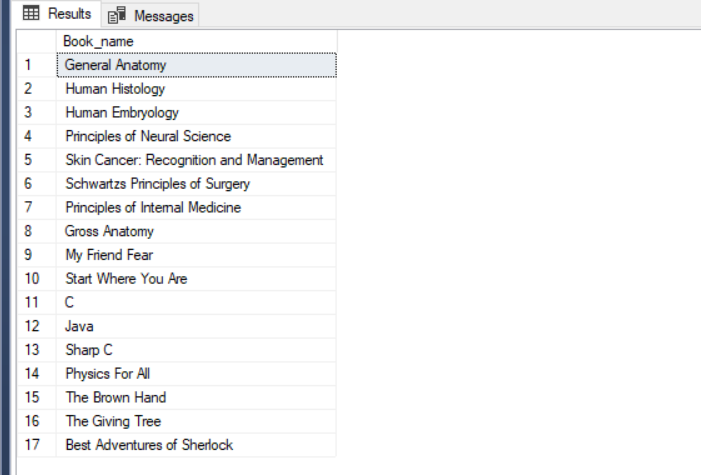
select Book\_name

from Book

where price > all (select Price

from Book

where Category = 'fiction');



1. All categories where the total price of books is greater than the average of the total price at all categories.

with category\_total (Category, value) as

(select Category, sum(Price)

from Book

group by Category),

category\_total\_avg(value) as

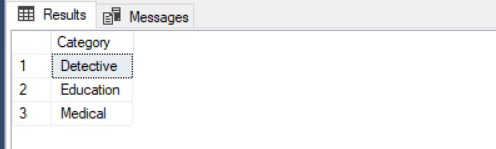
(select avg(value)

from category\_total)

select Category

from category\_total, category\_total\_avg

where category\_total.value > category\_total\_avg.value;



**Appendix A:**

create table Department(

Dept\_name varchar(60),

Building varchar(60),

Budget int,

constraint pk\_department primary key([Dept\_name])

);

create table Librarian(

Librarian\_id varchar(60),

Librarian\_name varchar(60),

Designation varchar(60),

Cell\_num varchar(60),

constraint pk\_librarian primary key([Librarian\_id])

);

create table Book(

Book\_id varchar(60),

Book\_name varchar(60),

Category varchar(60),

Publication varchar(60),

Price int,

constraint pk\_book primary key([Book\_id])

);

create table Ward(

Ward\_no varchar(60),

Category varchar(60),

Building varchar(60),

constraint pk\_ward primary key([Ward\_no])

);

create table Doctor(

Doctor\_id varchar(60),

Doc\_name varchar(60),

Dept\_name varchar(60),

Salary int,

Phone\_num varchar(60),

constraint pk\_doctor primary key([Doctor\_id]),

constraint fk\_doctor foreign key([Dept\_name]) references Department(Dept\_name)

);

create table Student(

Student\_id varchar(60),

Stu\_name varchar(60),

Dept\_name varchar(60),

Total\_credit float,

constraint pk\_student primary key([Student\_id]),

constraint fk\_student foreign key([Dept\_name]) references Department([Dept\_name])

);

create table Course(

Course\_id varchar(60),

Tittle varchar(60),

Dept\_name varchar(60),

Credit float,

constraint pk\_course primary key([Course\_id]),

constraint fk\_course foreign key([Dept\_name]) references Department([Dept\_name])

);

create table Takes(

Student\_id varchar(60),

Course\_id varchar(60),

Takes\_year varchar(60),

Grade varchar(60),

constraint pk\_takes primary key([Student\_id],[Course\_id],[Takes\_year]),

constraint fk\_takes1 foreign key([Student\_id]) references Student,

constraint fk\_takes2 foreign key([Course\_id]) references Course

);

create table Teaches(

Doctor\_id varchar(60),

Course\_id varchar(60),

Teaches\_year varchar(60),

constraint pk\_teaches primary key([Doctor\_id],[Course\_id],[Teaches\_year]),

constraint fk\_teaches1 foreign key([Doctor\_id]) references Doctor,

constraint fk\_teaches2 foreign key([Course\_id]) references Course

);

create table Issue\_book(

Student\_id varchar(60),

Book\_id varchar(60),

Issue\_date date,

Return\_date date,

constraint pk\_ibook primary key([Student\_id],[Book\_id],[Issue\_date]),

constraint fk\_ibook1 foreign key([Student\_id]) references Student,

constraint fk\_ibook2 foreign key([Book\_id]) references Book

);

create table Email(

Librarian\_id varchar(60),

Student\_id varchar(60),

Message varchar(600),

Msg\_date date,

constraint pk\_email primary key([Librarian\_id],[Student\_id],[Message],[Msg\_date]),

constraint fk\_email1 foreign key([Student\_id]) references Student,

constraint fk\_email2 foreign key([Librarian\_id]) references Librarian

);

create table Patient(

Patient\_id varchar(60),

Patient\_name varchar(60),

Patient\_disease varchar(60),

Bed\_no varchar(60),

Assigned\_doctor varchar(60),

Ward\_no varchar(60),

constraint pk\_patient primary key([Patient\_id]),

constraint fk\_patient foreign key([Ward\_no]) references Ward(Ward\_no),

constraint fk\_patient2 foreign key([Assigned\_doctor]) references Doctor(Doctor\_id)

);

create table Bill (

Bill\_no varchar(60),

Patient\_id varchar(60),

Amount int,

constraint pk\_bill primary key([Bill\_no]),

constraint fk\_bill foreign key([Patient\_id]) references Patient

);

**Appendix B:**

insert into Department values('clinical','TYLOR',30000000),

('para-clinical','WATSON',40000000);

insert into Librarian values('666','Romiz Uddin','Head','01848898999'),

('667','Sifat Hossien','Assistant Head','019965665667'),

('668','Zihan Khan','Assistant','017778799898'),

('669','Zabir Rahman','Assistant','016669696969'),

('670','Abul Kashem','Assistant','01675556689'),

('671','Rashed Khan','Assistant','01675556658'),

('672','Ragib Noor','Assistant','01675556664');

insert into Book values('760','General Anatomy','Medical','GLG',750),

('761','Physiology','Medical','CNC',450),

('762','Biochemistry','Medical','DLD',350),

('763','Human Histology','Medical','NLN',700),

('764','Human Embryology','Medical','KLL',900),

('765','Principles of Neural Science','Medical','LKL',800),

('766','Skin Cancer: Recognition and Management','Medical','NSN',600),

('767','Schwartzs Principles of Surgery','Medical','NPN',1000),

('768','Principles of Internal Medicine','Medical','SLS',950),

('769','Gross Anatomy','Medical','DYD',560),

('770','Blackberry Winter','Biography','GLG',200),

('771','Valentino','Biography','CNC',250),

('772','Swans','Biography','GLG',200),

('773','The Edge of Memory','Biography','DLD',400),

('774','Sea Prayer','Biography','NLN',500),

('775','War Like a Local','Fiction','GLG','200'),

('776','THE ROGER BROOK SERIES STARTER','Fiction','GLG',350),

('777','THE SWORD OF FATE','Fiction','DLD',300),

('778','UNCHARTED SEAS','Fiction','NLN',200),

('779','Holy Terror','Horror','KLL',350),

('780','Trauma','Horror','LKL',400),

('781','House of Bones','Horror','LKL',500),

('782','The Satanist','Horror','KLL',300),

('783','Made out of Stars','Inspiration','NSN',200),

('784','My Friend Fear','Inspiration','SLS',600),

('785','Start Where You Are','Inspiration','NPN',600),

('786','Coders','Education','NSN',200),

('787','Learn C','Education','CNC',200),

('788','C','Education','NPN',700),

('789','Java','Education','DLD',700),

('790','Sharp C','Education','NSN',800),

('791','Physics For All','Education','SLS',900),

('792','Vector','Education','DYD',500),

('793','Database','Education','GLG',200),

('794','Hound of the Vaskerbills','Detective','DYD',380),

('795','Adventure of the Empty House','Detective','DYD',250),

('796','Scandal in Bohemia','Detective','SLS',210),

('797','Chader Pahar','Fiction','DYD',520),

('798','The Secret Garden','Inspiration','NPN',330),

('799','Shikar','Horror','LKL',120),

('800','Life Without Limits','Inspiration','NSN',280),

('801','The Final Problem','Fiction','KLL',180),

('802','Satyanneshi Byomkesh','Detective','DYD',420),

('803','Hattyapuri','Detective','DYD',260),

('804','The Brown Hand','Detective','CNC',780),

('805','Sheyal Debota Rahassha','Detective','CNC',380),

('806','Gangtok e Gondogol','Detective','KLL',380),

('807','The Giving Tree','Inspiration','NLN',670),

('808','Best Adventures of Sherlock','Detective','DLD',580),

('809','Zombies','Horror','CNC',160);

insert into Ward values ('1A','burn unit','RH'),

('1B','coronary care unit','Packard'),

('1C','emergency unit','Painter'),

('1D','acute medical unit ','Packard'),

('2A','geriatric intensive- care unit','Painter'),

('2B','neonatal intensive care unit ','RH'),

('2C','pediatric intensive care unit ','Packard');

insert into Doctor values('031','dr. abdullah','clinical',50000,'01521207638'),

('032','dr. zarif','clinical',80000,'01944149959'),

('033','dr. imrul','clinical',90000,'01712369871'),

('034','dr. salman ali','clinical',100000,'01676600767'),

('035','dr. fahim','para-clinical',60000,'0176669990'),

('036','dr. showrav','para-clinical',70000,'01199788779'),

('037','dr. susmita','para-clinical',65000,'01981718789'),

('038','dr. nila','para-clinical',100000,'0174568928');

insert into Student values('18101','ahmed zamil','clinical',20.5),

('18102','zafar ali','clinical',20.5),

('18103','soma','clinical',24.0),

('18104','Hatem Tai','clinical',20.5),

('18105','Nyeem Azgar','clinical',21.5),

('18106','Shorif miya','clinical',22.5),

('18107','sophiya','clinical',23.5),

('18108','Hossain Sordar','clinical',20.5),

('18109','nirob','clinical',45.5),

('18110','simona','clinical',46.5),

('18111','rekah','clinical',48.5),

('18112','nazmul','clinical',49.0),

('18113','sakib','clinical',46.0),

('18114','taskin','clinical',45.5),

('18115','tanzil','clinical',46.5),

('18116','priya','clinical',46.5),

('18117','nayma','clinical',72.5),

('18118','akhi','clinical',74.0),

('18119','noyon','clinical',73.5),

('18120','jibon','clinical',70.5),

('18121','jisan','clinical',72.5),

('18122','joyonto','clinical',74.0),

('18123','niloy','clinical',72.5),

('18124','nibir','clinical',73.5),

('18201','antor','para-clinical',21.0),

('18202','labib','para-clinical',22.5),

('18203','hiron','para-clinical',21.5),

('18204','minhaz','para-clinical',24.5),

('18205','nakib','para-clinical',21.5),

('18206','sondip','para-clinical',21.5),

('18207','mow','para-clinical',21.5),

('18208','kona','para-clinical',21.5),

('18209','shakil','para-clinical',46.5),

('18210','shawon','para-clinical',46.5),

('18211','rony','para-clinical',46.5),

('18212','shojol','para-clinical',47.5),

('18213','rubel','para-clinical',48.5),

('18214','munna','para-clinical',49.5),

('18215','anik','para-clinical',49.5),

('18216','shojib','para-clinical',47.5),

('18217','hasif','para-clinical',71.5),

('18218','shorif','para-clinical',71.5),

('18219','momin','para-clinical',71.5),

('18220','arman','para-clinical',72.5),

('18221','hanif','para-clinical',73.5),

('18222','altaf','para-clinical',74.0),

('18223','bokul','para-clinical',74.5),

('18224','kotha','para-clinical',72.5);

insert into Course values('MDE-101','medicine','clinical',4.0),

('GO-102','gynea & obs','clinical',4.5),

('PA-103','paediatrics','clinical',4.0),

('PSY-104','psychiatry','clinical',4.0),

('DM-105','dermatology','clinical',4.5),

('ORT-106','orthopedics','clinical',4.5),

('CAR-107','cardiology','clinical',4.5),

('OPTH-108','ophthalmology','clinical',4.0),

('PA-201','pathology','para-clinical',4.5),

('PHAR-202','pharmacology','para-clinical',4.5),

('MIC-203','microbiology','para-clinical',4.5),

('FOM-204','forensic medicine','para-clinical',4.5),

('COM-205','community medicine','para-clinical',4.0),

('ANA-206','anatomy','para-clinical',4.5),

('PHY-207','physiology','para-clinical',4.5),

('BIO-208','biochemistry','para-clinical',4.5);

insert into Takes values('18101','MDE-101','2017','A'),

('18102','GO-102','2017','A-'),

('18103','MDE-101','2017','B'),

('18104','GO-102','2017','B+'),

('18107','MDE-101','2017','C+'),

('18108','GO-102','2017','D'),

('18109','PA-103','2017','C+'),

('18110','PSY-104','2017','C-'),

('18113','PA-103','2017','B-'),

('18114','PSY-104','2017','A'),

('18115','PA-103','2017','A+'),

('18116','PSY-104','2017','A+'),

('18119','DM-105','2017','A-'),

('18120','ORT-106','2017','A'),

('18121','DM-105','2017','A'),

('18122','ORT-106','2017','B+'),

('18103','PA-103','2018','B-'),

('18104','PSY-104','2018','B'),

('18105','PA-103','2018','B'),

('18106','PSY-104','2018','B-'),

('18109','DM-105','2018','A-'),

('18110','ORT-106','2018','D'),

('18111','DM-105','2018','F'),

('18112','ORT-106','2018','F'),

('18115','DM-105','2018','A+'),

('18116','ORT-106','2018','C'),

('18117','CAR-107','2018','C-'),

('18118','OPTH-108','2018','D'),

('18121','CAR-107','2018','A'),

('18122','OPTH-108','2018','B'),

('18123','CAR-107','2018','A-'),

('18124','OPTH-108','2018','B-'),

('18203','PA-201','2017','B'),

('18204','PHAR-202','2017','A+'),

('18205','PA-201','2017','B+'),

('18206','PHAR-202','2017','A-'),

('18209','MIC-203','2017','B-'),

('18210','FOM-204','2017','D'),

('18211','MIC-203','2017','C'),

('18212','FOM-204','2017','A'),

('18215','MIC-203','2017','C+'),

('18216','FOM-204','2017','C-'),

('18217','COM-205','2017','A-'),

('18218','ANA-206','2017','B'),

('18221','COM-205','2017','D'),

('18222','ANA-206','2017','C-'),

('18223','COM-205','2017','A'),

('18224','ANA-206','2017','B-'),

('18201','MIC-203','2018','D'),

('18202','FOM-204','2018','C+'),

('18203','MIC-203','2018','C-'),

('18204','FOM-204','2018','A+'),

('18207','MIC-203','2018','A'),

('18208','FOM-204','2018','C+'),

('18209','COM-205','2018','A-'),

('18210','ANA-206','2018','B-'),

('18213','COM-205','2018','C'),

('18214','ANA-206','2018','C-'),

('18215','COM-205','2018','A'),

('18216','ANA-206','2018','B'),

('18219','PHY-207','2018','B+'),

('18220','BIO-208','2018','B-'),

('18221','PHY-207','2018','D'),

('18222','BIO-208','2018','C+');

insert into Teaches values('031','MDE-101','2017'),

('032','GO-102','2017'),

('033','PA-103','2017'),

('034','PSY-104','2017'),

('033','DM-105','2017'),

('034','ORT-106','2017'),

('031','CAR-107','2018'),

('032','OPTH-108','2018'),

('033','PA-103','2018'),

('034','PSY-104','2018'),

('033','DM-105','2018'),

('034','ORT-106','2018'),

('035','PA-201','2017'),

('036','PHAR-202','2017'),

('037','MIC-203','2017'),

('038','FOM-204','2017'),

('037','COM-205','2017'),

('038','ANA-206','2017'),

('035','PHY-207','2018'),

('036','BIO-208','2018'),

('037','MIC-203','2018'),

('038','FOM-204','2018'),

('037','COM-205','2018'),

('038','ANA-206','2018');

insert into Issue\_book values('18107','760','2017-10-16','2017-10-26'),

('18109','760','2017-11-22','2017-12-05'),

('18107','783','2018-01-08','2018-01-26'),

('18106','795','2018-01-29','2018-02-05'),

('18203','763','2018-02-26',null),

('18108','768','2018-03-14','2018-03-22'),

('18115','765','2018-04-06','2018-05-02'),

('18212','801','2018-04-24','2018-05-16'),

('18104','770','2018-05-08','2018-05-28'),

('18101','786','2018-05-12','2018-05-21'),

('18106','772','2018-06-15','2018-06-28'),

('18121','764','2018-06-22','2018-07-10'),

('18108','798','2018-07-02','2018-07-22'),

('18205','806','2018-07-19',null),

('18101','769','2018-08-03',null);

insert into Email values('669','18107','Return the Book','2018-01-22'),

('668','18203','Return the Book','2018-03-10'),

('670','18203','Pay for the Book','2018-03-24'),

('671','18115','Return the Book','2018-04-20'),

('672','18212','Return the Book','2018-05-08'),

('670','18104','Return the Book','2018-05-22'),

('668','18121','Return the Book','2018-07-06'),

('672','18108','Return the Book','2018-07-16'),

('669','18205','Return the Book','2018-08-03');

insert into Patient values('1610','karim','pneumonia','102','031','1A'),

('1611','rahim','Blood Pressure','103','032','1B'),

('1612','jamil','skin','104','033','1C'),

('1613','sohel','fever','105','034','1D'),

('1614','mita','diabetes','106','035','2A'),

('1615','mira','overweight','107','036','2B'),

('1616','mona','stroke','108','037','2C'),

('1617','soma','influenza','109','038','1A'),

('1618','noman','foot & mouth','201','031','1B'),

('1619','silviya','fascioliasis','202','032','1C'),

('1620','abir','malaria','203','033','1D'),

('1621','akib','dengue','204','034','2A'),

('1622','asif','typhoid','205','035','2B'),

('1623','sorna','viral fevers','206','036','2C'),

('1624','asma','cholera','207','037','1A'),

('1625','payel','hepatitis B','208','038','1B');

insert into Bill values('16101031','1610',20000),

('16101032','1611',30000),

('16101033','1612',100000),

('16101034','1613',1200000),

('16101035','1614',200000),

('16101036','1615',2200000),

('16101037','1616',400000),

('16101038','1617',500000),

('16101039','1618',60000),

('16101040','1619',70000),

('16101041','1620',80000),

('16101042','1621',90000),

('16101043','1622',550000),

('16101044','1623',110000),

('16101045','1624',330000),

('16101046','1625',40000),

('16101047','1621',10000),

('16101048','1614',20000),

('16101049','1610',30000),

('16101050','1610',25000);