American International University-Bangladesh

Project Title: Doctor Appointment Management System

Course Name: Introduction to Database

Section: F

Course Teacher: Juena Ahmed Noshin

Group Member Info:

Name	ID	Contribution
Mst. Nadiya Noor	22-46454-1	Scenario, Normalization, Table creation, Subquery, user s creation, Data insertion, Schema Diagram
Most. Sayma Khatun	22-47035-1	ER Diagram, Normalization, Joining, Data insertion, Schema Diagram
Md. Nafiun Ahmed Ovi	22-46453-1	Group function, Normalization, View.
Effat Ara	22-46090-1	Single row function, Short description.
Ahsanul Haque Joar	22-47183-1	Conclusion, Normalization, Relational algebra.

Contents Page

Introduction 3

Scenario Description 3

ER Diagram 4

Normalization 5-12

Final Table 12

Schema Diagram 13

User Creation 14

Table Creation 15-25

Data Insertion 26-35

Single Row Function 36

Group Function 37

Subquery 38-39

Joining 40-41

View 42-43

Relational Algebra 44

Conclusion 45

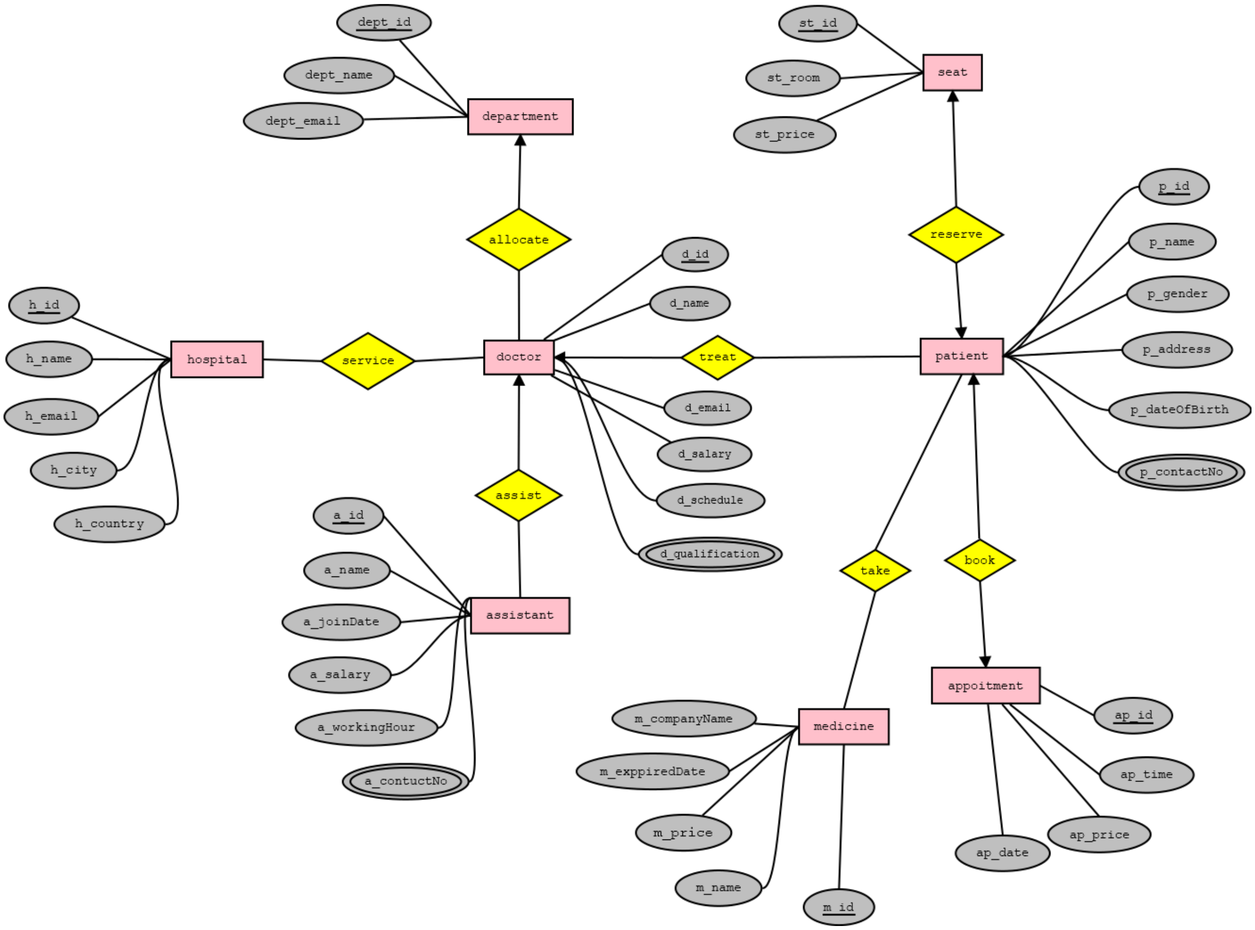
Introduction

In a ‘Doctor Appointment Management System’ generally the patient is appointed and treated by a doctor. In this system, patient information is recorded. For the appointment, a hospital also includes this system. Doctors are connected with specific hospitals and manage their appointment. There remains appointment information of a patient. Such as appointment time, date, price etc. Doctors are also from various departments. Doctors' information is also recorded here. Such as doctor name, email, schedule, qualification etc. There are also some assistants to help doctors. The assistant generally takes the appointment from the patient. For the emergency case, patients may reserve seats for resting and treatment. There are also various kinds of medicine. From where patients may take medicine.

Scenario Description

In a Doctor Appointment management system, a doctor may treat many patients. A patient may be treated by only one doctor. A doctor is identified by a doctor ID. The system also stores doctor name, email, salary, schedule for visiting patients and qualification. There may be multiple qualifications. A patient is identified by a patient ID. The system also stores patient name, date of birth, gender, contact number, address. There may be multiple contact numbers of patients. In this system, a doctor may give his/her service in at least one hospital. A hospital may be serviced by many doctors. A hospital is identified by hospital id. The system also stores hospital name, email, city, country. A doctor may be allocated by exactly one department. A department may have many doctors. A department is identified by department ID. The system also stores the department name and email. A doctor may have at least one assistant. An assistant can assist at most one doctor. An assistant is identified by an assistant ID. The system also stores assistant name, join date, salary, working hour and contact number. There may be multiple contact numbers of assistants. A patient may reserve at most one seat. A seat can be reserved by exactly one patient. A seat is identified by seat id. The system also stores room no of seat and seat price. A patient may book at most one appointment. An appointment is booked by only one patient. An appointment is identified by appointment id. The system also stores appointment price, date and time. A patient may take many medicines. Medicine can be taken by many patients. A medicine is identified by medicine id. Medicine name, company name, price, expire date of a medicine are also stored.

ER Diagram


dept_id
dept_name
dept_email
department
allocate
st_id
st_room
st_price
seat
reserve
p_id
p_name
p_gender
p_address
p_dateOfBirth
p_contactNo
patient
h_id
h_name
h_email
h_city
h_country
hospital
service
doctor
d_id
d_name
treat
d_email
d_salary
d_schedule
d_qualification
assist
a_id
a_name
a_joinDate
a_salary
a_workingHour
a_contuctNo
assistant
take
book
medicine
m_companyName
m_exppiredDate
m_price
m_name
m_id
appoitment
ap_id
ap_time
ap_price
ap_date

Normalization

book

UNF

book (ap_id, ap_price, ap_date, ap_time, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address)

1NF

a_contactNo is multivalued attribute

1. ap_id, ap_price, ap_date, ap_time, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

2NF

1. ap_id, ap_price, ap_date, ap_time

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

3NF

There is no transitive attribute. Relation is already in 3NF

1. ap_id, ap_price, ap_date, ap_time

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

Table Creation

1. ap_id, ap_price, ap_date, ap_time, **p_id**

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

take

UNF

take (m_id, m_name, m_companyName, m_price, m_expiredDate, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address)

1NF

p_contactNo is multivalued attribute

1. m_id, m_name, m_companyName, m_price, m_expiredDate, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

2NF

1. m_id, m_name, m_companyName, m_price, m_expiredDate

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

3NF

There is no transitive attribute. Relation is already in 3NF

1. m_id, m_name, m_companyName, m_price, m_expiredDate

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

Table Creation

1. m_id, m_name, m_companyName, m_price, m_expiredDate

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

4. **m_id, p_id**

assist

UNF

assist (d_id, d_name, d_email, d_salary, d_schedule, d_qualification, a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo)

1NF

a_contactNo, d_qualification is multivalued attribute

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo

2NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification,

2. a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo

3NF

There is no transitive attribute. Relation is already in 3NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification,

2. a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo

Table Creation

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification,

2. a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo, **d_id**

allocate

UNF

allocate (d_id, d_name, d_email, d_salary, d_schedule, d_qualification, dept_id, dept_name, dept_email)

1NF

d_qualification is multivalued attribute

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, dept_id, dept_name, dept_email

2NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. dept_id, dept_name, dept_email

3NF

There is no transitive attribute. Relation is already in 3NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. dept_id, dept_name, dept_email

Table Creation

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, **dept_id**

2. dept_id, dept_name, dept_email

treat

UNF

treat(d_id, d_name, d_email, d_salary, d_schedule, d_qualification, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address)

1NF

p_contactNo, d_qualification is multivalued attribute

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address)

2NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

3NF

There is no transitive attribute. Relation is already in 3NF

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

Table Creation

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address**, d_id**

service

UNF

service (d_id, d_name, d_email, d_salary, d_schedule, d_qualification, h_id, h_name, h_email, h_city, h_country)

1NF

d_qualification is multivalued attribute

1. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, h_id, h_name, h_email, h_city, h_country

2NF

1.d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. h_id, h_name, h_email, h_city, h_country

3NF

1.d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. h_id, h_name, h_email

3. h_city, h_country

Table Creation

1.d_id, d_name, d_email, d_salary, d_schedule, d_qualification

2. h_id, h_name, h_email, **address_id**

3. address_id, h_city, h_country

4. **d_id**, **h_id**

reserve

UNF

reserve (st_id, st_room, st_price, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address)

1NF

p_contactNo is multivalued attribute

1. st_id, st_room, st_price, p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

2NF

1. st_id, st_room, st_price

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

3NF

1. st_id, st_room, st_price

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

Table Creation

1. st_id, st_room, st_price, **p_id**

2. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

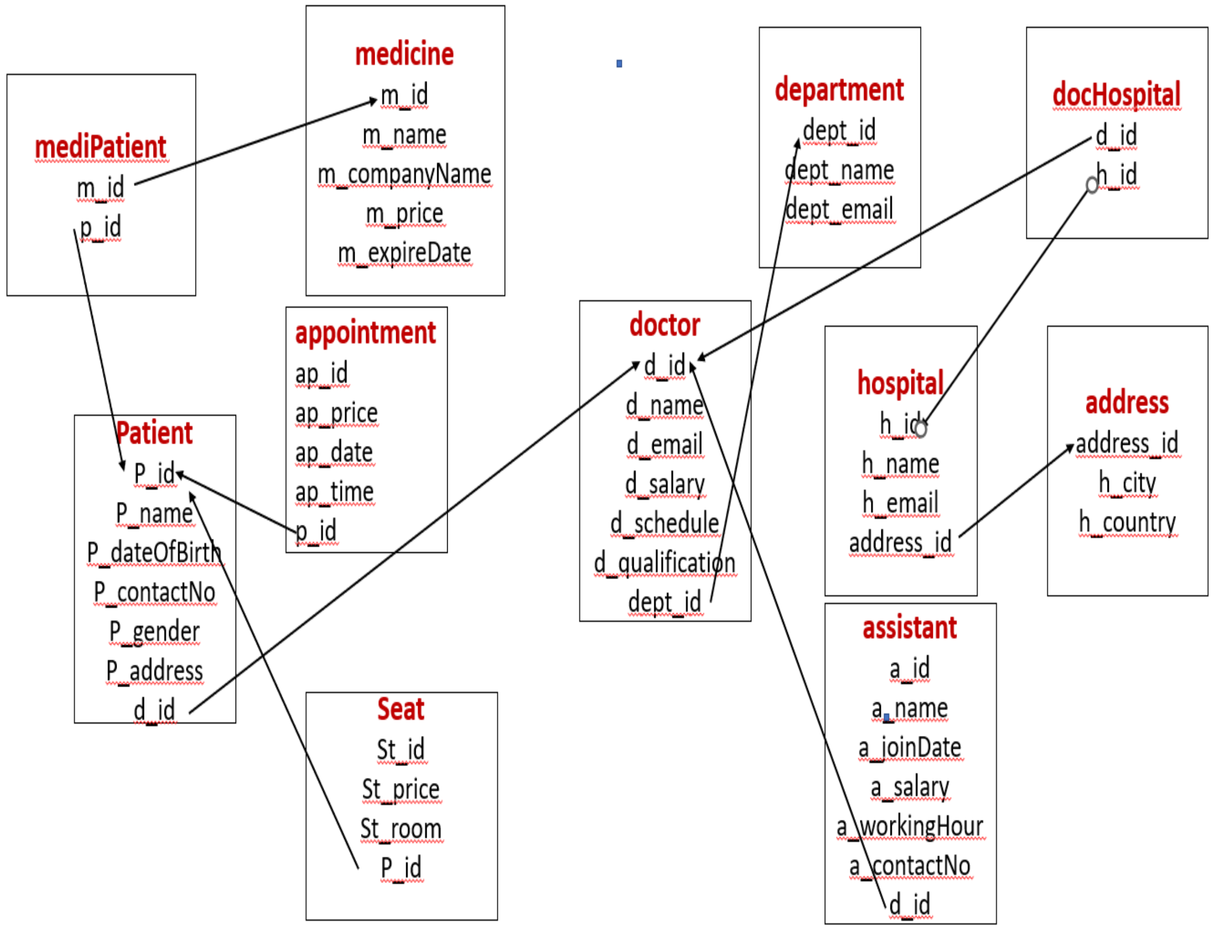
Temporary Table

1. ap_id, ap_price, ap_date, ap_time, **p_id**
2. a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo
3. *m_id, m_name, m_companyName, m_price, m_expiredDate*
4. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address
5. ***m_id, p_id***
6. d_id, d_name, d_email, d_salary, d_schedule, d_qualification
7. *a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo,* ***d_id***
8. *d_id, d_name, d_email, d_salary, d_schedule, d_qualification,* ***dept_id***
9. *dept_id, dept_name, dept_email*
10. d_id, d_name, d_email, d_salary, d_schedule, d_qualification
11. *p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address***, d_id**
12. d_id, d_name, d_email, d_salary, d_schedule, d_qualification
13. *h_id, h_name, h_email,* ***address_id***
14. *address_id, h_city, h_country*
15. ***d_id**, **h_id***
16. *st_id, st_room, st_price,* ***p_id***
17. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address

Final Table

1. ap_id, ap_price, ap_date, ap_time**, p_id**
2. m_id, m_name, m_companyName, m_price, m_expiredDate
3. **m_id, p_id**
4. a_id, a_name, a_joinDate, a_salary, a_workingHour, a_contactNo, **d_id**
5. d_id, d_name, d_email, d_salary, d_schedule, d_qualification, **dept_id**
6. dept_id, dept_name, dept_email
7. p_id, p_name, p_dateOfBirth, p_contactNo, p_gender, p_address**, d_id**
8. h_id, h_name, h_email, **address_id**
9. address_id, h_city, h_country
10. **d_id**, **h_id**
11. st_id, st_room, st_price, **p_id**

Schema Diagram



User Creation

User: admin

create user admin identified by admin;

grant dba to admin;

grant create table to admin;

grant create session to admin;

grant create sequence to admin;

grant create view to admin;

```
grant create procedure to admin;
```

Table Creation

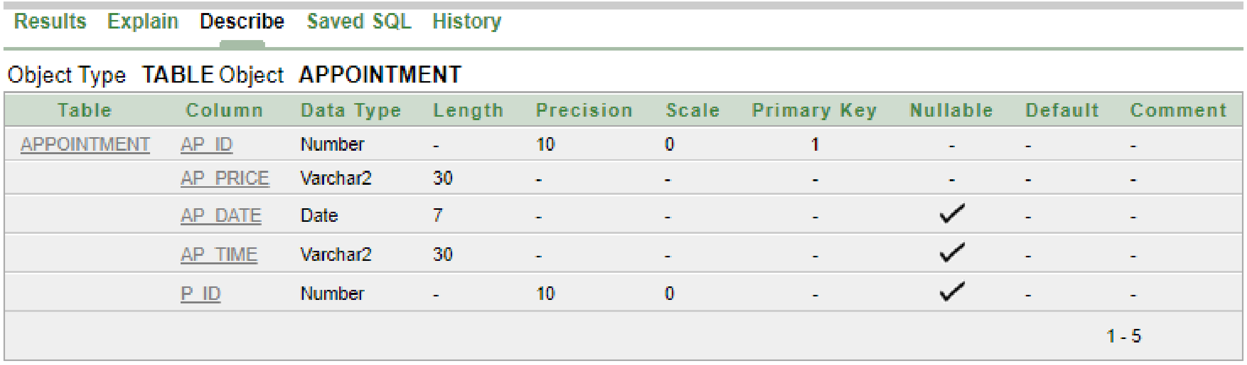
1. appointment

Table Creation: create table appointment

```
(ap_id number(10)primary key,
ap_price varchar2(30)not null,
ap_date date,
ap_time varchar2(30)unique,
p_id number(10),
FOREIGN KEY (p_id) REFERENCES patient(p_id));
```

Sequence Creation:

```
create sequence ap_id
increment by 1
start with 01
maxvalue 1000;
desc appointment;
```



Object Type TABLE Object APPOINTMENT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
APPOINTMENT	AP_ID	Number	-	10	0	1	-	-	-
	AP_PRICE	Varchar2	30	-	-	-	-	-	-
	AP_DATE	Date	7	-	-	-	✓	-	-
	AP_TIME	Varchar2	30	-	-	-	✓	-	-
	P_ID	Number	-	10	0	-	✓	-	-

1 - 5

2. medicine

Table Creation: create table medicine

(m_id number(10)primary key,

m_name varchar2(50)not null,

m_companyName varchar2(50), m_price varchar2(20),

m_expireDate date);

Sequence Creation:

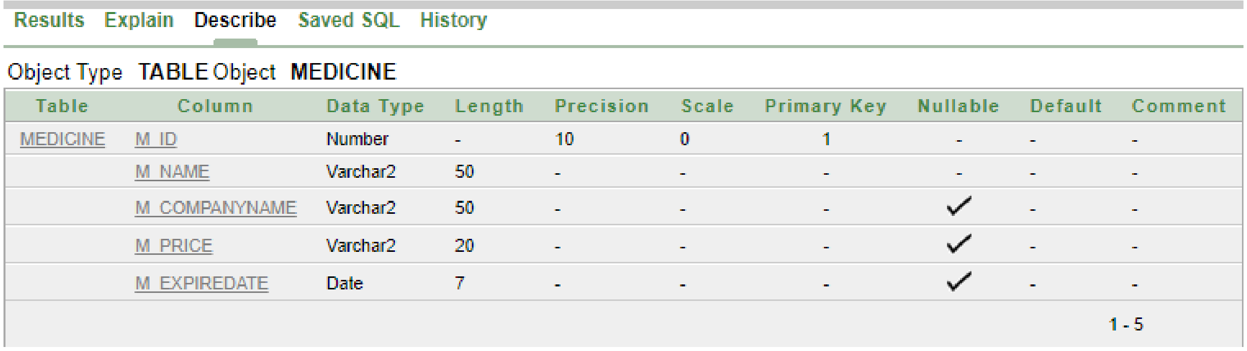
create sequence m_id

increment by 5

start with 10

maxvalue 1000;

desc medicine;



Object Type TABLE Object MEDICINE

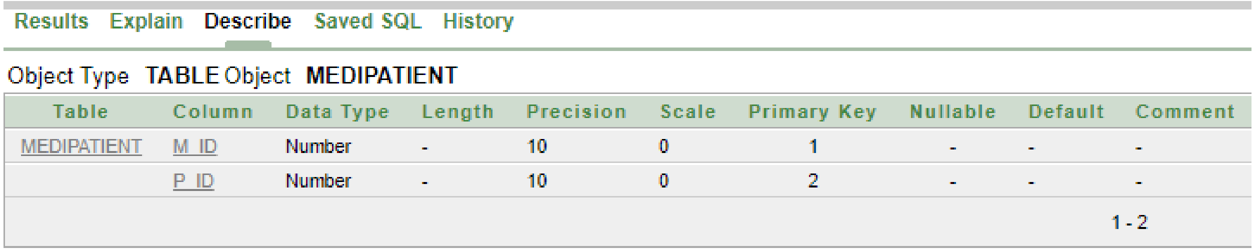
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
MEDICINE	M_ID	Number	-	10	0	1	-	-	-
	M_NAME	Varchar2	50	-	-	-	-	-	-
	M_COMPANYNAME	Varchar2	50	-	-	-	✓	-	-
	M_PRICE	Varchar2	20	-	-	-	✓	-	-
	M_EXPIREDATE	Date	7	-	-	-	✓	-	-

1 - 5

3. mediPatient

Table Creation:

```
create table mediPatient

(m_id number(10), p_id number(10),

primary key(m_id,p_id),FOREIGN KEY (m_id)

REFERENCES medicine(m_id), FOREIGN KEY

 (p_id) REFERENCES patient(p_id));


desc mediPatient;
```



Object Type TABLE Object MEDIPATIENT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
MEDIPATIENT	M_ID	Number	-	10	0	1	-	-	-
	P_ID	Number	-	10	0	2	-	-	-

1 - 2

4. assistant

Table Creation: create table assistant

```
( a_id number(10)primary key,

 a_name varchar2(50)not null,

 a_contactNo1 number(20)unique,  a_contactNo2 number(20)unique,
```

a_salary varchar2(20), a_workingHour varchar2(30),

a_joinDate date, d_id number(10), FOREIGN KEY (d_id) REFERENCES doctor(d_id));

Sequence Creation:

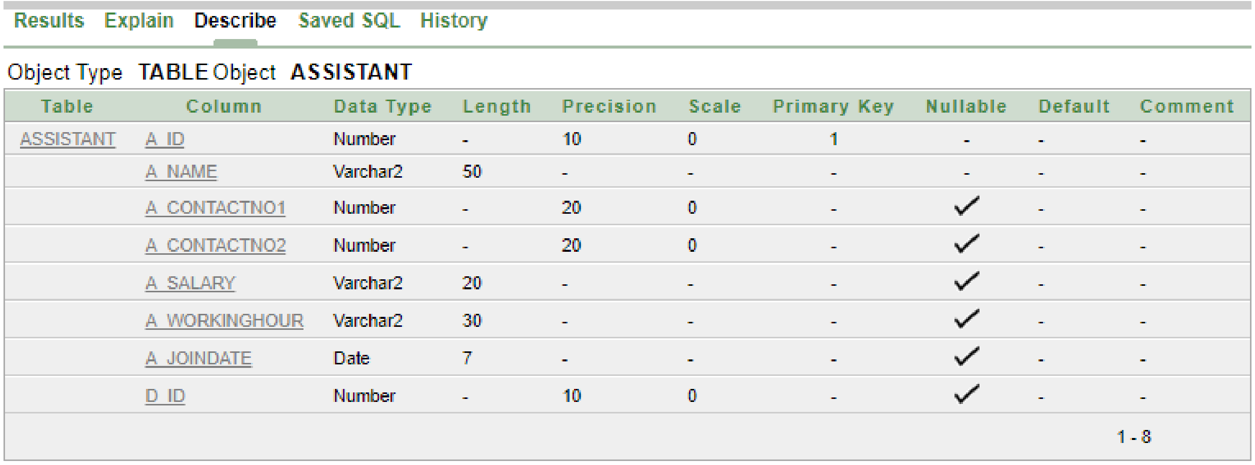
create sequence a_id

increment by 2

start with 300

maxvalue 1000;

desc assistant;



Object Type TABLE Object ASSISTANT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ASSISTANT	A_ID	Number	-	10	0	1	-	-	-
	A_NAME	Varchar2	50	-	-	-	-	-	-
	A_CONTACTNO1	Number	-	20	0	-	✓	-	-
	A_CONTACTNO2	Number	-	20	0	-	✓	-	-
	A_SALARY	Varchar2	20	-	-	-	✓	-	-
	A_WORKINGHOUR	Varchar2	30	-	-	-	✓	-	-
	A_JOINDATE	Date	7	-	-	-	✓	-	-
	D_ID	Number	-	10	0	-	✓	-	-

1 - 8

5. doctor

Table Creation: create table doctor

(d_id number(10)primary key,

d_name varchar2(30)not null,

d_salary varchar2(30), d_qualification1 varchar2(30),

d_qualification2 varchar2(30),dept_id number(10),

FOREIGN KEY (dept_id) REFERENCES department(dept_id));

Alter Table:

alter table doctor add(d_email varchar2(30), d_schedule varchar2(30));

Sequence Creation:

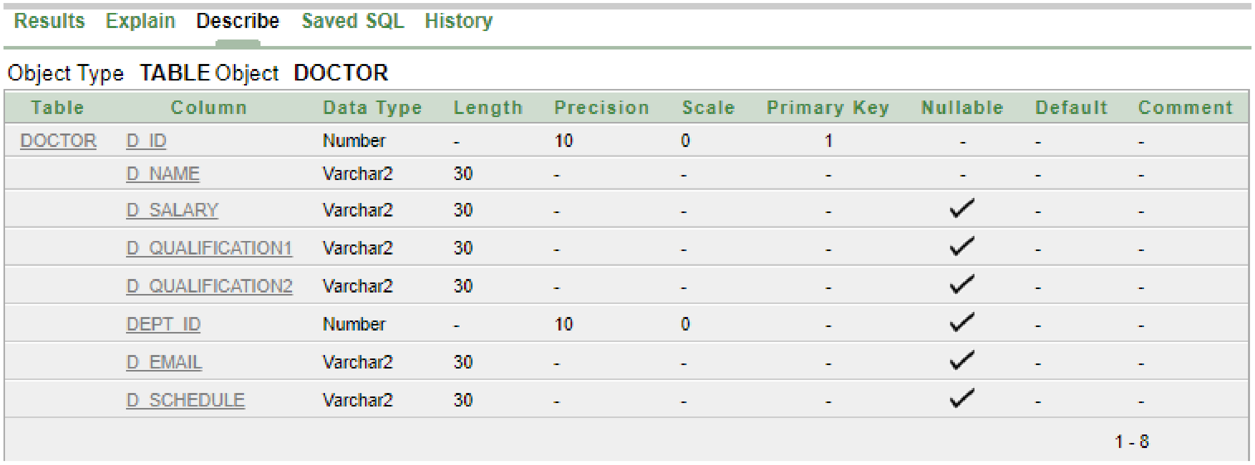
create sequence d_id

increment by 2

start with 140

maxvalue 1000;

desc doctor;



Object Type TABLE Object DOCTOR

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DOCTOR	D_ID	Number	-	10	0	1	-	-	-
	D_NAME	Varchar2	30	-	-	-	-	-	-
	D_SALARY	Varchar2	30	-	-	-	✓	-	-
	D_QUALIFICATION1	Varchar2	30	-	-	-	✓	-	-
	D_QUALIFICATION2	Varchar2	30	-	-	-	✓	-	-
	DEPT_ID	Number	-	10	0	-	✓	-	-
	D_EMAIL	Varchar2	30	-	-	-	✓	-	-
	D_SCHEDULE	Varchar2	30	-	-	-	✓	-	-

1 - 8

6. department

Table Creation:

create table department(dept_id number(10)primary key,

dept_name varchar2(50)not null,

dept_email varchar2(50));

Sequence Creation:

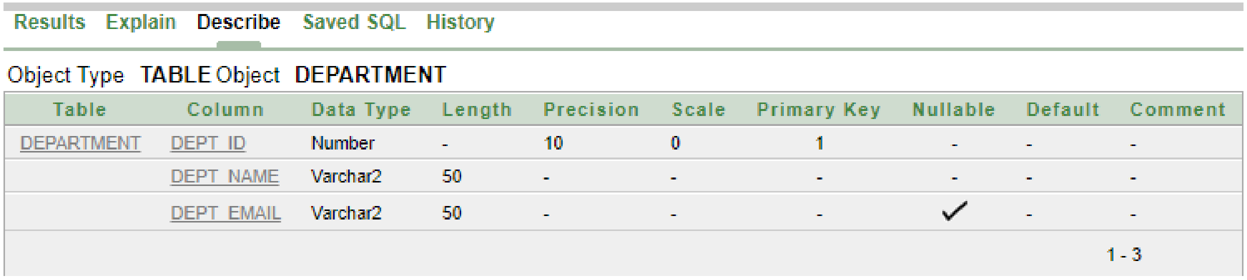
create sequence dept_id

increment by 10

start with 100

maxvalue 1000;

desc department;



Object Type TABLE Object DEPARTMENT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPARTMENT	DEPT_ID	Number	-	10	0	1	-	-	-
	DEPT_NAME	Varchar2	50	-	-	-	-	-	-
	DEPT_EMAIL	Varchar2	50	-	-	-	✓	-	-

1 - 3

7. patient

Table Creation: create table patient

(p_id number(10)primary key,

p_name varchar2(50)not null,

p_contactNo1 number(20)unique, p_contactNo2 number(20)unique,

p_gender varchar2(20), p_address varchar2(30),

m_expireDate date, d_id number(10), FOREIGN KEY (d_id) REFERENCES doctor(d_id));

Alter Table:

alter table patient drop column m_expireDate;

alter table patient add(p_dateOfBirth date);

Sequence Creation:

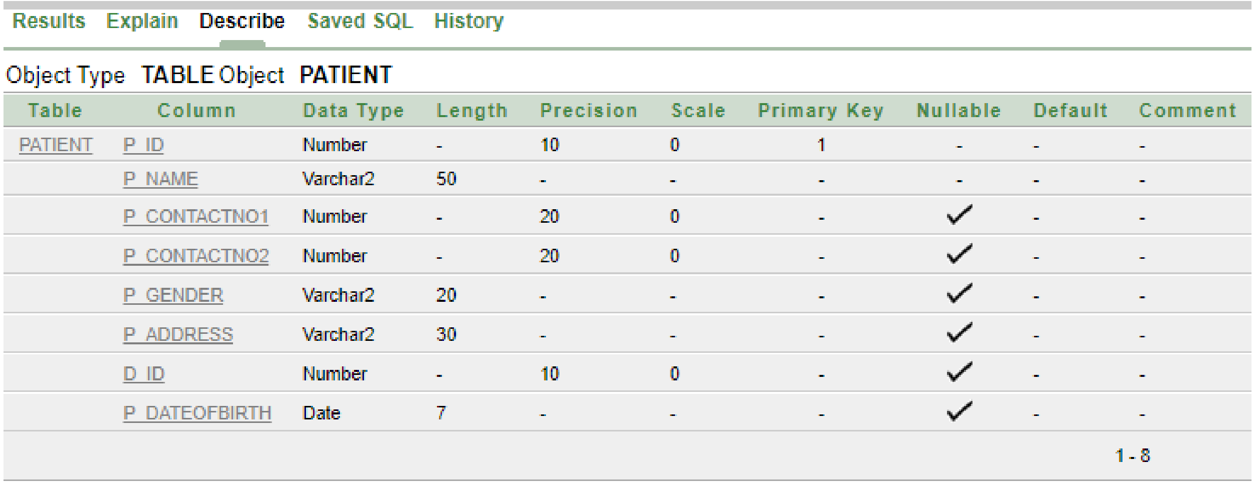
create sequence p_id

increment by 2

start with 150

maxvalue 1000;

desc patient;



Object Type TABLE Object PATIENT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PATIENT	P_ID	Number	-	10	0	1	-	-	-
	P_NAME	Varchar2	50	-	-	-	-	-	-
	P_CONTACTNO1	Number	-	20	0	-	✓	-	-
	P_CONTACTNO2	Number	-	20	0	-	✓	-	-
	P_GENDER	Varchar2	20	-	-	-	✓	-	-
	P_ADDRESS	Varchar2	30	-	-	-	✓	-	-
	D_ID	Number	-	10	0	-	✓	-	-
	P_DATEOFBIRTH	Date	7	-	-	-	✓	-	-

1 - 8

8. hospital

Table Creation: create table hospital

(h_id number(10)primary key,

h_name varchar2(30)not null,

h_email varchar2(30)unique,

address_id number(10),

FOREIGN KEY (address_id) REFERENCES address(address_id));

Sequence Creation:

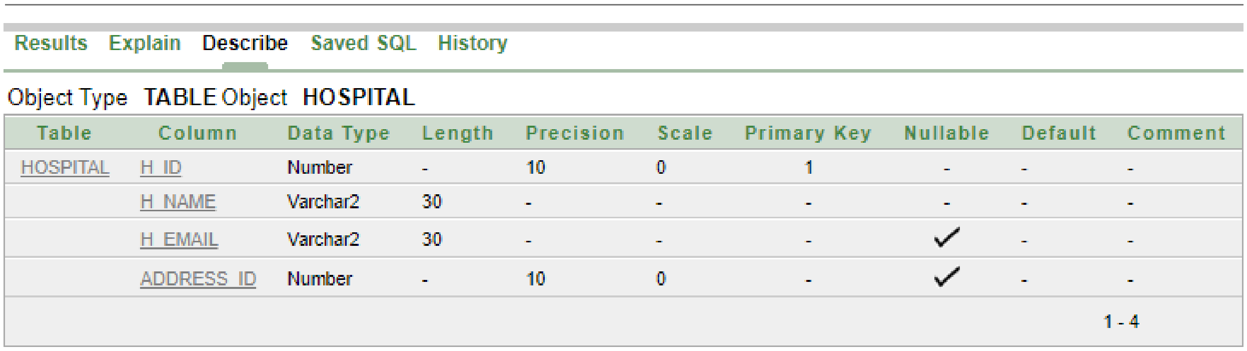
create sequence h_id

increment by 2

start with 20

maxvalue 1000;

desc hospital;



Object Type TABLE Object HOSPITAL

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
HOSPITAL	H_ID	Number	-	10	0	1	-	-	-
	H_NAME	Varchar2	30	-	-	-	-	-	-
	H_EMAIL	Varchar2	30	-	-	-	✓	-	-
	ADDRESS_ID	Number	-	10	0	-	✓	-	-

1 - 4

9. address

Table Creation: create table address

(address_id number(10)primary key, h_city varchar2(50),

h_country varchar2(50));

Sequence Creation:

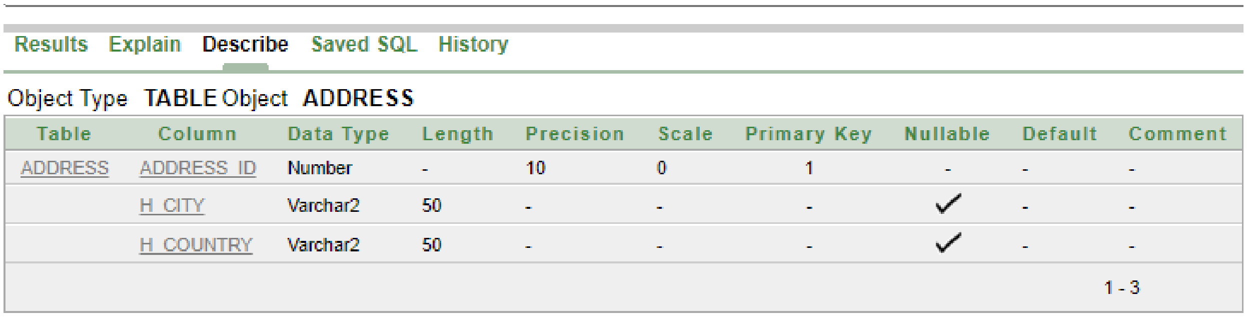
create sequence address_id

increment by 2

start with 200

maxvalue 1000;

desc hospital;



Object Type TABLE Object ADDRESS

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADDRESS	ADDRESS_ID	Number	-	10	0	1	-	-	-
	H_CITY	Varchar2	50	-	-	-	✓	-	-
	H_COUNTRY	Varchar2	50	-	-	-	✓	-	-

1 - 3

10. docHospital

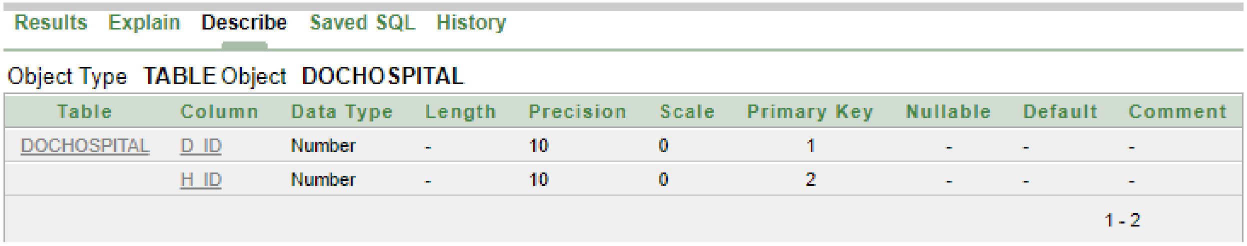
Table Creation:

create table docHospital

(d_id number(10), h_id number(10), primary key(d_id,h_id), FOREIGN KEY (d_id)

REFERENCES doctor(d_id), FOREIGN KEY (h_id) REFERENCES hospital(h_id));

desc docHospital;



Object Type TABLE Object DOCHOSPITAL

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DOCHOSPITAL	D_ID	Number	-	10	0	1	-	-	-
	H_ID	Number	-	10	0	2	-	-	-

1 - 2

11. seatBooking

Table Creation: create table seatBooking

(st_id number(10)primary key,

st_roomNo varchar2(30)default 'Not yet booking',

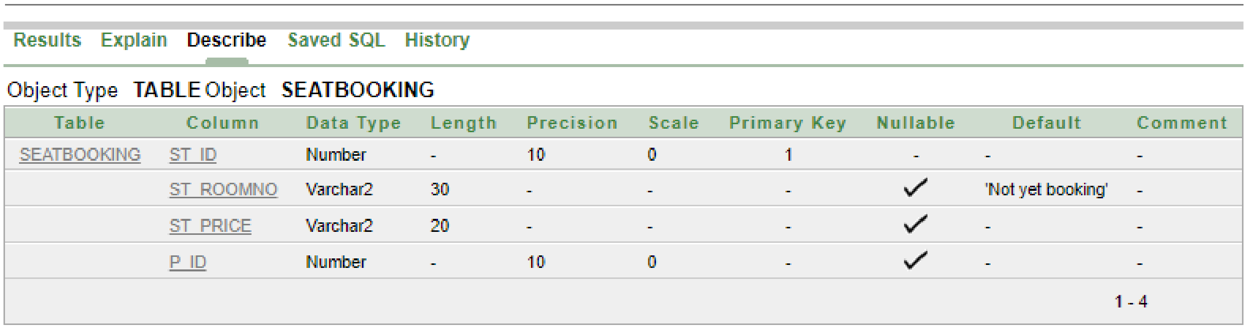
st_price varchar2(20),

p_id number(10),

FOREIGN KEY (p_id) REFERENCES patient(p_id));

Sequence Creation:

```
create sequence st_id

increment by 2

start with 20

maxvalue 1000;

desc seatBooking;
```



Object Type TABLE Object SEATBOOKING

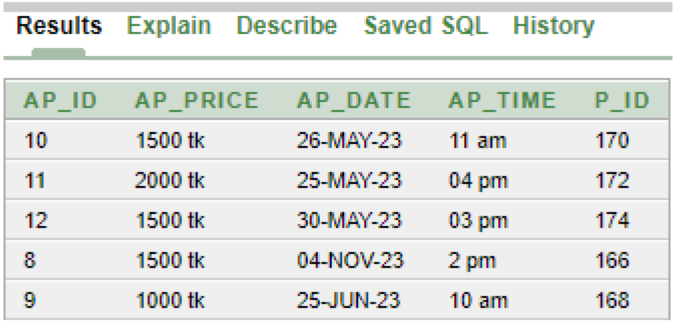
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SEATBOOKING	ST_ID	Number	-	10	0	1	-	-	-
	ST_ROOMNO	Varchar2	30	-	-	-	✓	'Not yet booking'	-
	ST_PRICE	Varchar2	20	-	-	-	✓	-	-
	P_ID	Number	-	10	0	-	✓	-	-

1 - 4

Data Insertion

1. appointment:

```
insert into appointment values(ap_id.nextval, '1500 tk','04 Nov 2023','2 pm','166');

insert into appointment values(ap_id.nextval, '1000 tk','25 June 2023','10 am','168');

insert into appointment values(ap_id.nextval, '1500 tk','26 May 2023','11 am','170');

insert into appointment values(ap_id.nextval, '2000 tk','25 May 2023',' 04 pm','172');

insert into appointment values(ap_id.nextval, '1500 tk','30 May 2023','03 pm','174');
```



AP_ID	AP_PRICE	AP_DATE	AP_TIME	P_ID
10	1500 tk	26-MAY-23	11 am	170
11	2000 tk	25-MAY-23	04 pm	172
12	1500 tk	30-MAY-23	03 pm	174
8	1500 tk	04-NOV-23	2 pm	166
9	1000 tk	25-JUN-23	10 am	168

2. medicine:

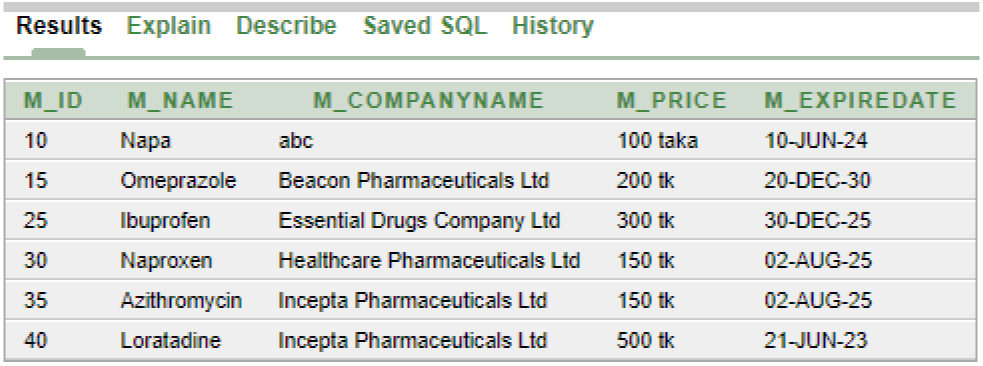
insert into medicine values(m_id.nextval, 'Omeprazole', 'Beacon Pharmaceuticals Ltd','200 tk',' 20 Dec 2030');

insert into medicine values(m_id.nextval, 'Ibuprofen', 'Essential Drugs Company Ltd','300 tk',' 30 Dec 2025');

insert into medicine values(m_id.nextval, 'Naproxen', 'Healthcare Pharmaceuticals Ltd','150 tk',' 02 Aug 2025');

insert into medicine values(m_id.nextval, 'Azithromycin', 'Incepta Pharmaceuticals Ltd','150 tk',' 02 Aug 2025');

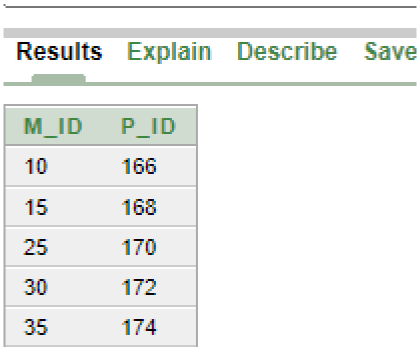
insert into medicine values(m_id.nextval, 'Loratadine', 'Incepta Pharmaceuticals Ltd','500 tk',' 21 June 2023');



M_ID	M_NAME	M_COMPANYNAME	M_PRICE	M_EXPIREDATE
10	Napa	abc	100 taka	10-JUN-24
15	Omeprazole	Beacon Pharmaceuticals Ltd	200 tk	20-DEC-30
25	Ibuprofen	Essential Drugs Company Ltd	300 tk	30-DEC-25
30	Naproxen	Healthcare Pharmaceuticals Ltd	150 tk	02-AUG-25
35	Azithromycin	Incepta Pharmaceuticals Ltd	150 tk	02-AUG-25
40	Loratadine	Incepta Pharmaceuticals Ltd	500 tk	21-JUN-23

3. mediPatient:

```
insert into mediPatient values('10','166');

insert into mediPatient values('15','168');

insert into mediPatient values('25','170');

insert into mediPatient values('30','172');

insert into mediPatient values('35','174');
```



M_ID	P_ID
10	166
15	168
25	170
30	172
35	174

4. assistant:

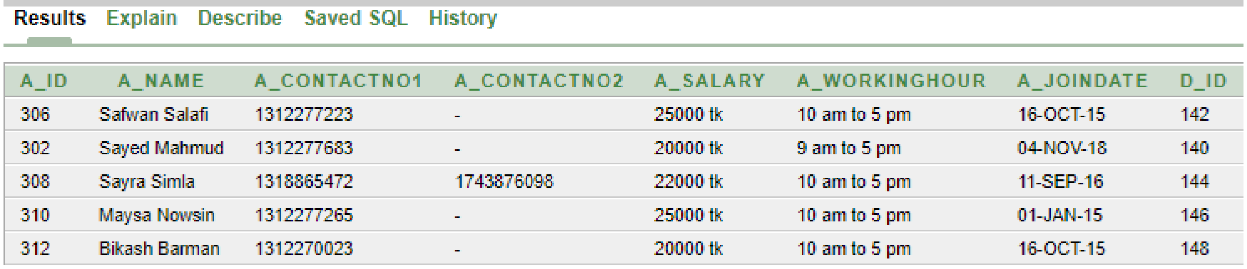
```
insert into assistant values(a_id.nextval, 'Sayed Mahmud','01312277683',null,'20000 tk','9 am to 5 pm','04 Nov 2018','140');
```

insert into assistant values(a_id.nextval, 'Safwan Salafi','01312277223',null,'25000 tk','10 am to 5 pm','16 Oct 2015','142');

insert into assistant values(a_id.nextval, 'Sayra Simla','01318865472',01743876098,'22000 tk','10 am to 5 pm','11 Sep 2016','144');

insert into assistant values(a_id.nextval, 'Maysa Nowsin','01312277265',null,'25000 tk','10 am to 5 pm','01 Jan 2015','146');

insert into assistant values(a_id.nextval, 'Bikash Barman','01312270023',null,'20000 tk','10 am to 5 pm','16 Oct 2015','148');



A_ID	A_NAME	A_CONTACTNO1	A_CONTACTNO2	A_SALARY	A_WORKINGHOUR	A_JOINDATE	D_ID
306	Safwan Salafi	1312277223	-	25000 tk	10 am to 5 pm	16-OCT-15	142
302	Sayed Mahmud	1312277683	-	20000 tk	9 am to 5 pm	04-NOV-18	140
308	Sayra Simla	1318865472	1743876098	22000 tk	10 am to 5 pm	11-SEP-16	144
310	Maysa Nowsin	1312277265	-	25000 tk	10 am to 5 pm	01-JAN-15	146
312	Bikash Barman	1312270023	-	20000 tk	10 am to 5 pm	16-OCT-15	148

5. doctor:

insert into doctor values(d_id.nextval, 'Md.Asrafil Haque','90000 tk','MD','MBBS','110');

insert into doctor values(d_id.nextval, 'Samima Farhat','95000 tk','D.N.B','MBBS','120');

insert into doctor values(d_id.nextval, 'Hasan Tariq','80000 tk','FCPS','MD','130');

insert into doctor values(d_id.nextval, 'Tahmid Mahdi','100000 tk','FCPS','MBBS','140');

insert into doctor values(d_id.nextval, 'Mahdiya Rahman','80000 tk','MS','DNB','150');

Data Insertion after alter table:

update doctor set d_email='asraf567@gmail.com'where d_id=140;

update doctor set d_email='samiafarhat99@gmail.com'where d_id=142;

update doctor set d_email='hasantariq33@gmail.com'where d_id=144;

update doctor set d_email='tahmidmahdi01@gmail.com'where d_id=146;

update doctor set d_email='rmahdiya543@gmail.com'where d_id=148;

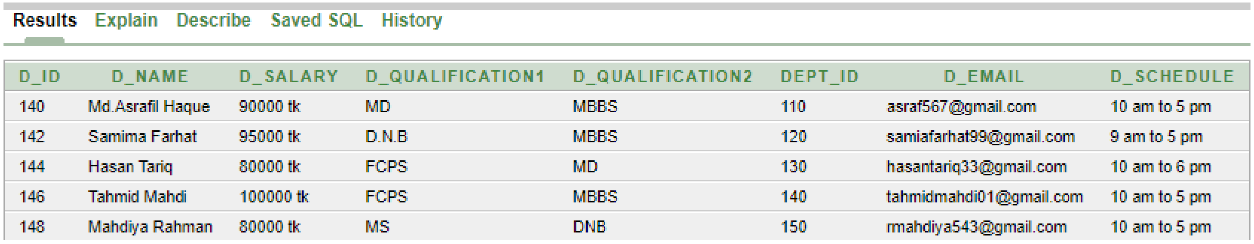
update doctor set d_schedule='10 am to 5 pm'where d_id=140;

update doctor set d_schedule='9 am to 5 pm'where d_id=142;

update doctor set d_schedule='10 am to 6 pm'where d_id=144;

update doctor set d_schedule='10 am to 5 pm'where d_id=146;

update doctor set d_schedule='10 am to 5 pm'where d_id=148;



D_ID	D_NAME	D_SALARY	D_QUALIFICATION1	D_QUALIFICATION2	DEPT_ID	D_EMAIL	D_SCHEDULE
140	Md.Asrafil Haque	90000 tk	MD	MBBS	110	asraf567@gmail.com	10 am to 5 pm
142	Samima Farhat	95000 tk	D.N.B	MBBS	120	samiafarhat99@gmail.com	9 am to 5 pm
144	Hasan Tariq	80000 tk	FCPS	MD	130	hasantariq33@gmail.com	10 am to 6 pm
146	Tahmid Mahdi	100000 tk	FCPS	MBBS	140	tahmidmahdi01@gmail.com	10 am to 5 pm
148	Mahdiya Rahman	80000 tk	MS	DNB	150	rmahdiya543@gmail.com	10 am to 5 pm

6. department:

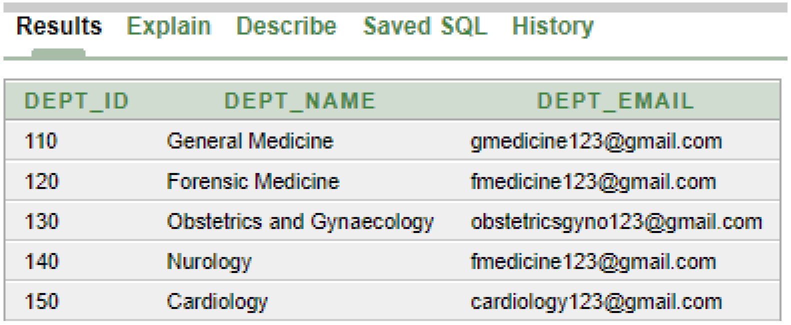
insert into department values(dept_id.nextval, 'Cardiology','cardiology123@gmail.com');

insert into department values(dept_id.nextval, 'General Medicine','gmedicine123@gmail.com');

insert into department values(dept_id.nextval, 'Forensic Medicine','fmedicine123@gmail.com');

insert into department values(dept_id.nextval, 'Obstetrics and Gynaecology','obstetricsgyno123@gmail.com');

insert into department values(dept_id.nextval, 'Nurology','fmedicine123@gmail.com');



DEPT_ID	DEPT_NAME	DEPT_EMAIL
110	General Medicine	gmedicine123@gmail.com
120	Forensic Medicine	fmedicine123@gmail.com
130	Obstetrics and Gynaecology	obstetricsgyno123@gmail.com
140	Nurology	fmedicine123@gmail.com
150	Cardiology	cardiology123@gmail.com

7. patient:

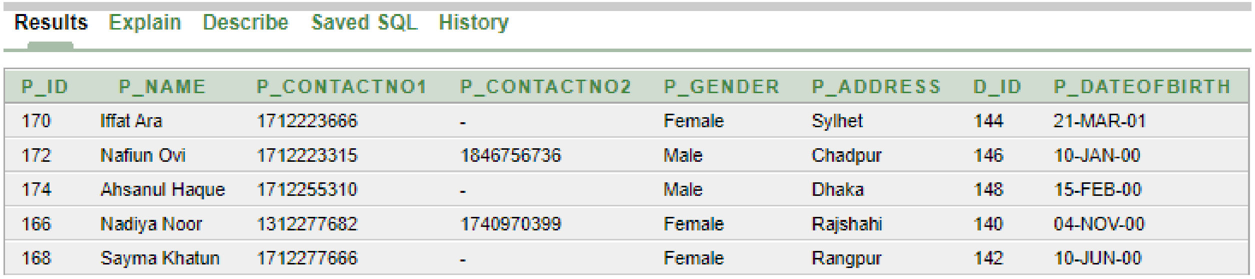
insert into patient values(p_id.nextval, 'Nadiya Noor','01312277682','01740970399','Female','Rajshahi','140','04 Nov 2000');

insert into patient values(p_id.nextval, 'Sayma Khatun','01712277666',null,'Female','Rangpur','142','10 June 2000');

insert into patient values(p_id.nextval, 'Iffat Ara','01712223666',null,'Female','Sylhet','144','21 March 2001');

insert into patient values(p_id.nextval, 'Nafiun Ovi','01712223315','01846756736','Male','Chadpur','146','10 Jan 2000');

insert into patient values(p_id.nextval, 'Ahsanul Haque','01712255310',null,'Male','Dhaka','148','15 Feb 2000');

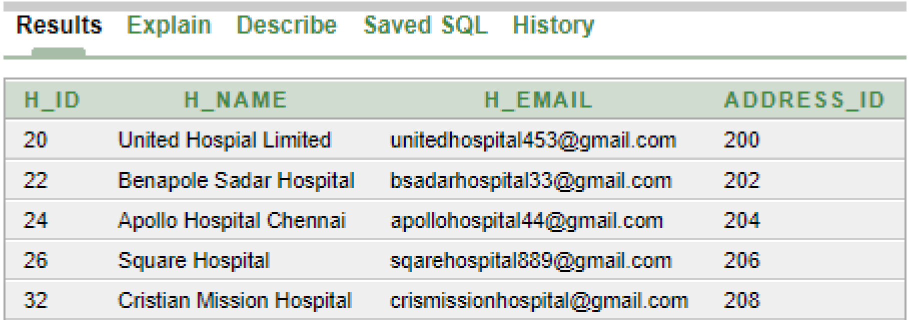


P_ID	P_NAME	P_CONTACTNO1	P_CONTACTNO2	P_GENDER	P_ADDRESS	D_ID	P_DATEOFBIRTH
170	Iffat Ara	1712223666	-	Female	Sylhet	144	21-MAR-01
172	Nafiun Ovi	1712223315	1846756736	Male	Chadpur	146	10-JAN-00
174	Ahsanul Haque	1712255310	-	Male	Dhaka	148	15-FEB-00
166	Nadiya Noor	1312277682	1740970399	Female	Rajshahi	140	04-NOV-00
168	Sayma Khatun	1712277666	-	Female	Rangpur	142	10-JUN-00

8. hospital:

insert into hospital values(h_id.nextval, 'United Hospial Limited','unitedhospital453@gmail.com', '200');

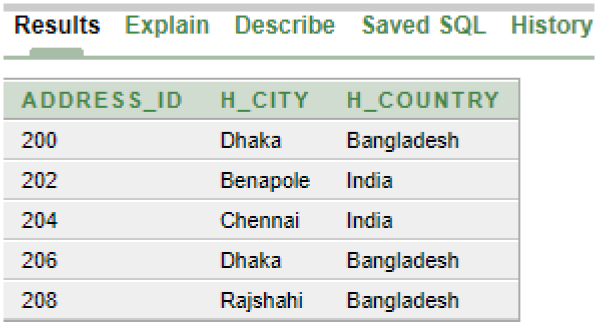
```
insert into hospital values(h_id.nextval, 'Benapole Sadar Hospital','bsadarhospital33@gmail.com', '202');

insert into hospital values(h_id.nextval, 'Apollo Hospital Chennai','apollohospital44@gmail.com', '204');

insert into hospital values(h_id.nextval, 'Square Hospital','sqarehospital889@gmail.com', '206');

insert into hospital values(h_id.nextval, 'Cristian Mission Hospital','crismissionhospital@gmail.com', '208');
```



H_ID	H_NAME	H_EMAIL	ADDRESS_ID
20	United Hospial Limited	unitedhospital453@gmail.com	200
22	Benapole Sadar Hospital	bsadarhospital33@gmail.com	202
24	Apollo Hospital Chennai	apollohospital44@gmail.com	204
26	Square Hospital	sqarehospital889@gmail.com	206
32	Cristian Mission Hospital	crismissionhospital@gmail.com	208

9. address:

```
insert into address values(address_id.nextval, 'Dhaka','Bangladesh');
insert into address values(address_id.nextval, 'Benapole','India');
insert into address values(address_id.nextval, 'Chennai','India');
insert into address values(address_id.nextval, 'Dhaka','Bangladesh');
insert into address values(address_id.nextval, 'Rajshahi','Bangladesh');
```



ADDRESS_ID	H_CITY	H_COUNTRY
200	Dhaka	Bangladesh
202	Benapole	India
204	Chennai	India
206	Dhaka	Bangladesh
208	Rajshahi	Bangladesh

10. docHospital:

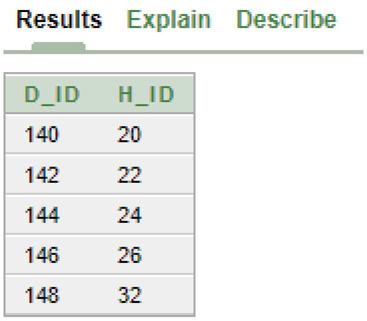
insert into docHospital values('140','20');

insert into docHospital values('142','22');

insert into docHospital values('144','24');

insert into docHospital values('146','26');

insert into docHospital values('148','32');



D_ID	H_ID
140	20
142	22
144	24
146	26
148	32

11. seatBooking:

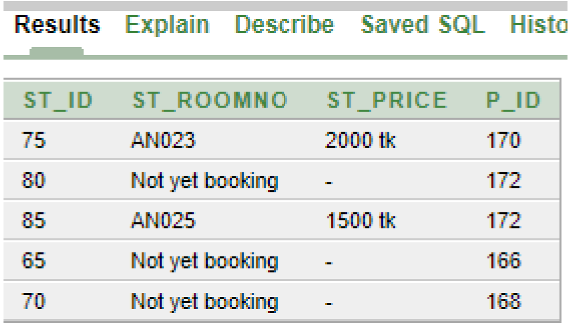
insert into seatBooking values(st_id.nextval, default,null,'166');

insert into seatBooking values(st_id.nextval,default,null,'168');

insert into seatBooking values(st_id.nextval,'AN023','2000 tk','170');

insert into seatBooking values(st_id.nextval,default,null,'172');

insert into seatBooking values(st_id.nextval,'AN025','1500 tk','172');

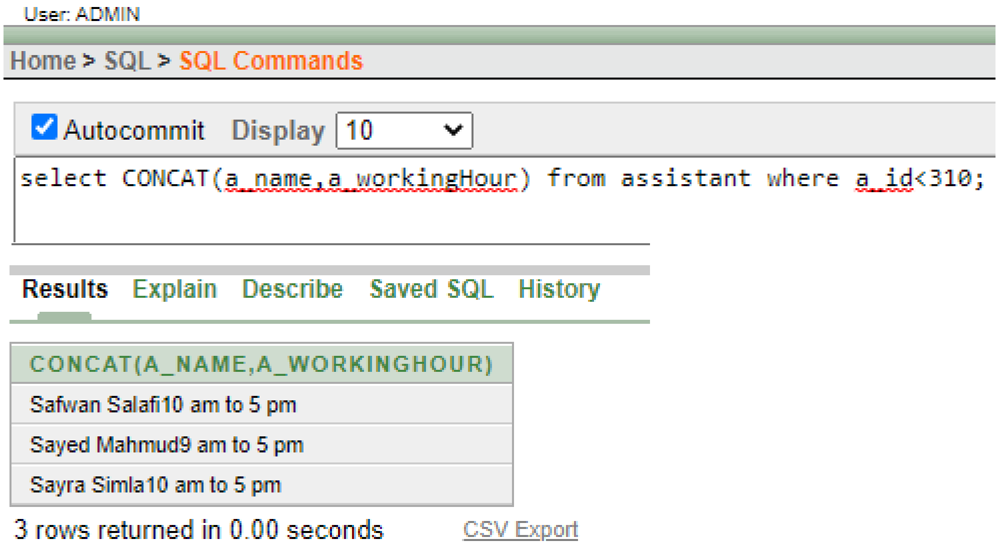


ST_ID	ST_ROOMNO	ST_PRICE	P_ID
75	AN023	2000 tk	170
80	Not yet booking	-	172
85	AN025	1500 tk	172
65	Not yet booking	-	166
70	Not yet booking	-	168

Single Row Function

No1: Write a query to concat name and working hour of assistant who have a_id less than 310.

Ans: select CONCAT(a_name,a_workingHour) from assistant where a_id<310;


User: ADMIN
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History
CSV Export

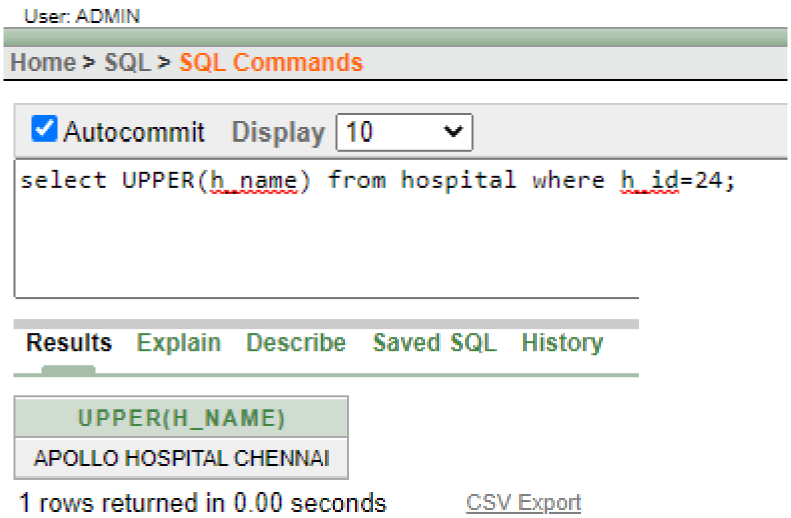
```
select CONCAT(a_name,a_workingHour) from assistant where a_id<310;
```

CONCAT(A_NAME,A_WORKINGHOUR)
Safwan Salafi10 am to 5 pm
Sayed Mahmud9 am to 5 pm
Sayra Simla10 am to 5 pm

3 rows returned in 0.00 seconds

No2: Write a query to upper case the hospital name having h_id=24.

Ans: select UPPER(h_name) from hospital where h_id=24;



☑ Autocommit Display 10

```
select UPPER(h_name) from hospital where h_id=24;
```

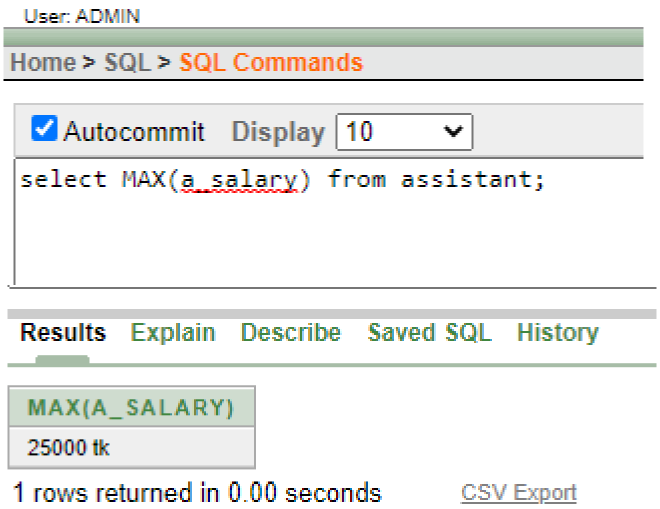
UPPER(H_NAME)
APOLLO HOSPITAL CHENNAI

1 rows returned in 0.00 seconds

Group Function

No1: Write a query to show the maximum salary of assistant.

Ans: select MAX(a_salary) from assistant;



☑ Autocommit Display 10

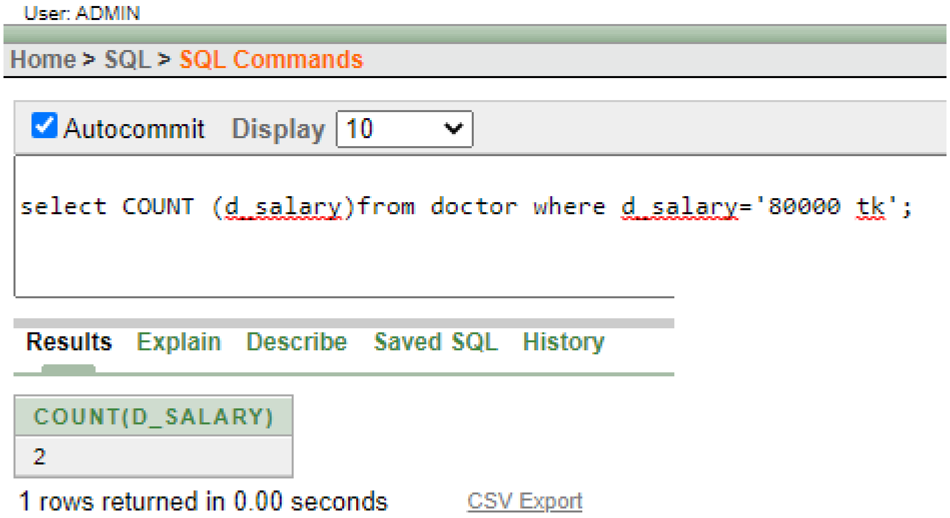
```
select MAX(a_salary) from assistant;
```

MAX(A_SALARY)
25000 tk

1 rows returned in 0.00 seconds

No2: Write a query to count the total row of doctor table having d_salary=’80000 tk’.

Ans: select COUNT (d_salary)from doctor where d_salary='80000 tk';



```
select COUNT (d_salary)from doctor where d_salary='80000 tk';
```

COUNT(D_SALARY)
2

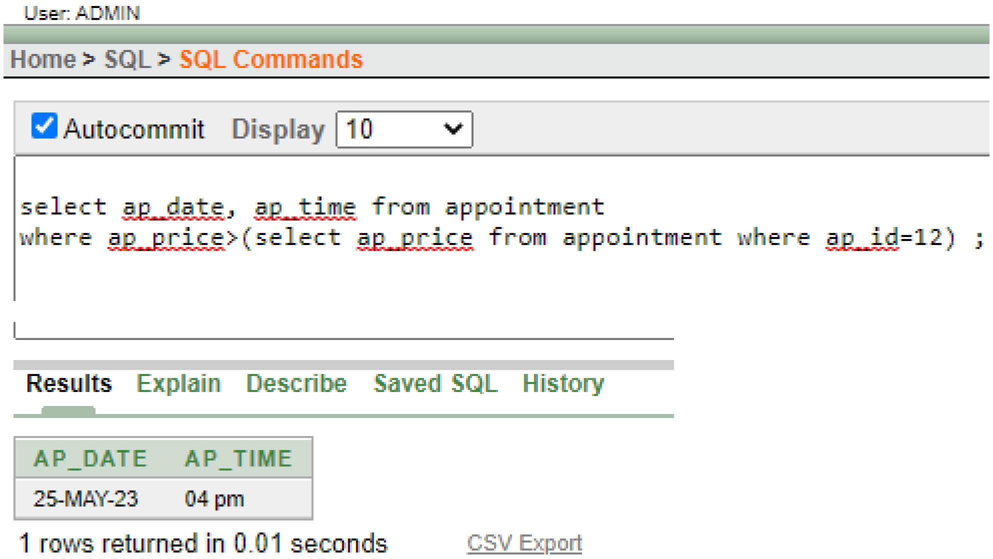
1 rows returned in 0.00 seconds

SubQuery

Subquery No1:

Display the title and release date of music which release date is greater than id number 12?

Ans: select ap_date, ap_time from appointment where ap_price>(select ap_price from appointment where ap_id=12) ;



```
select ap_date, ap_time from appointment
where ap_price>(select ap_price from appointment where ap_id=12) ;
```

AP_DATE	AP_TIME
25-MAY-23	04 pm

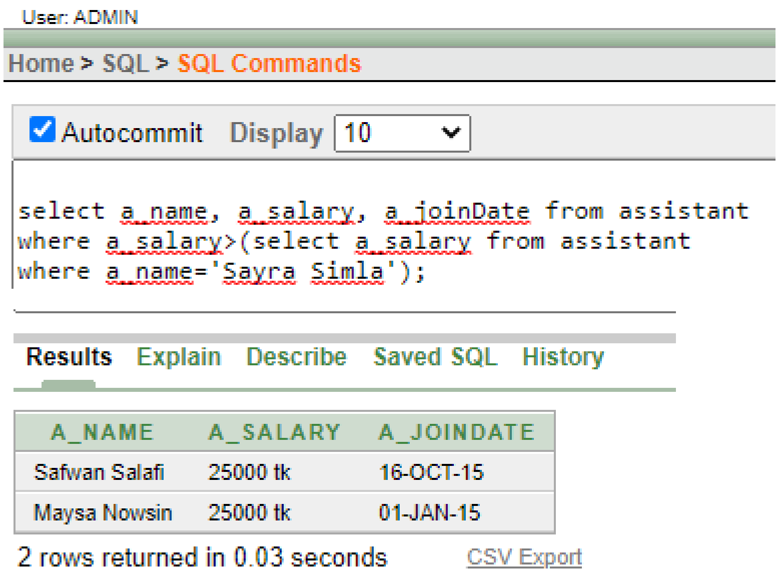
1 rows returned in 0.01 seconds

Subquery No2:

Display the name, salary and join date for the assistant who earn more than “Sayra Simla”?

Ans: select a_name, a_salary, a_joinDate from assistant where

a_salary>(select a_salary from assistant where a_name='Sayra Simla');



☑ Autocommit Display 10

```
select a_name, a_salary, a_joinDate from assistant
where a_salary>(select a_salary from assistant
where a_name='Sayra Simla');
```

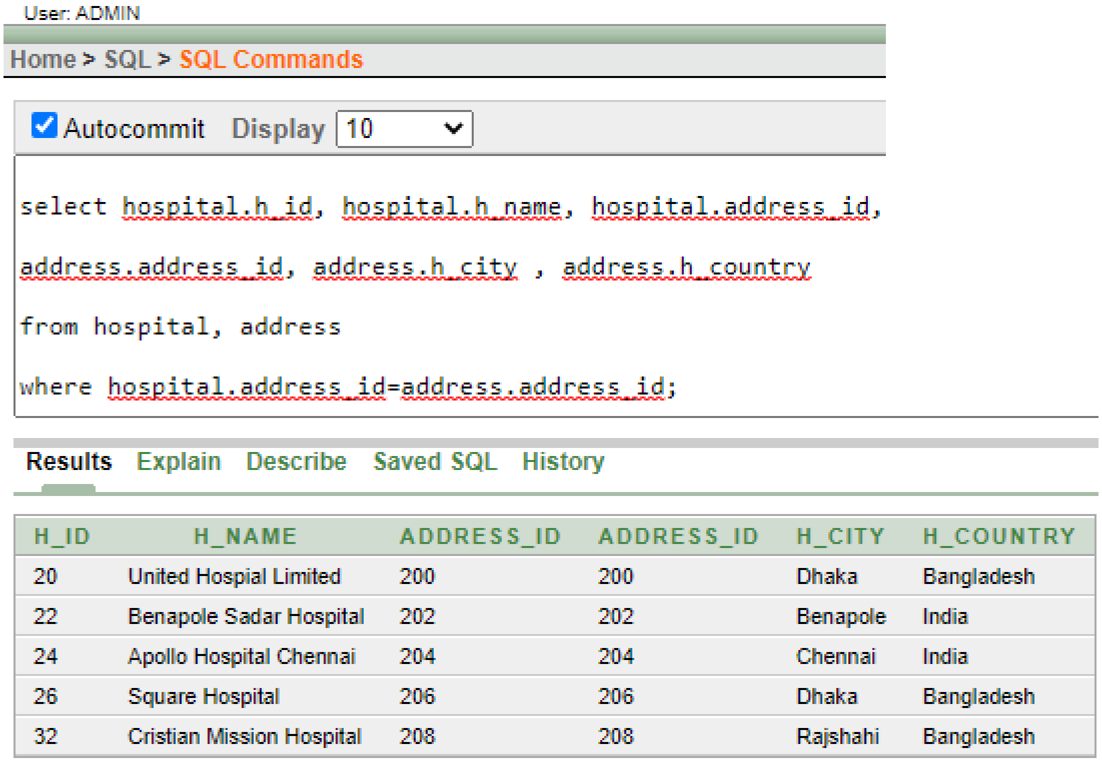
A_NAME	A_SALARY	A_JOINDATE
Safwan Salafi	25000 tk	16-OCT-15
Maysa Nowsin	25000 tk	01-JAN-15

2 rows returned in 0.03 seconds

Joining

Joining No1:

```
select hospital.h_id, hospital.h_name, hospital.address_id,
address.address_id, address.h_city , address.h_country
from hospital, address
where hospital.address_id=address.address_id;
```



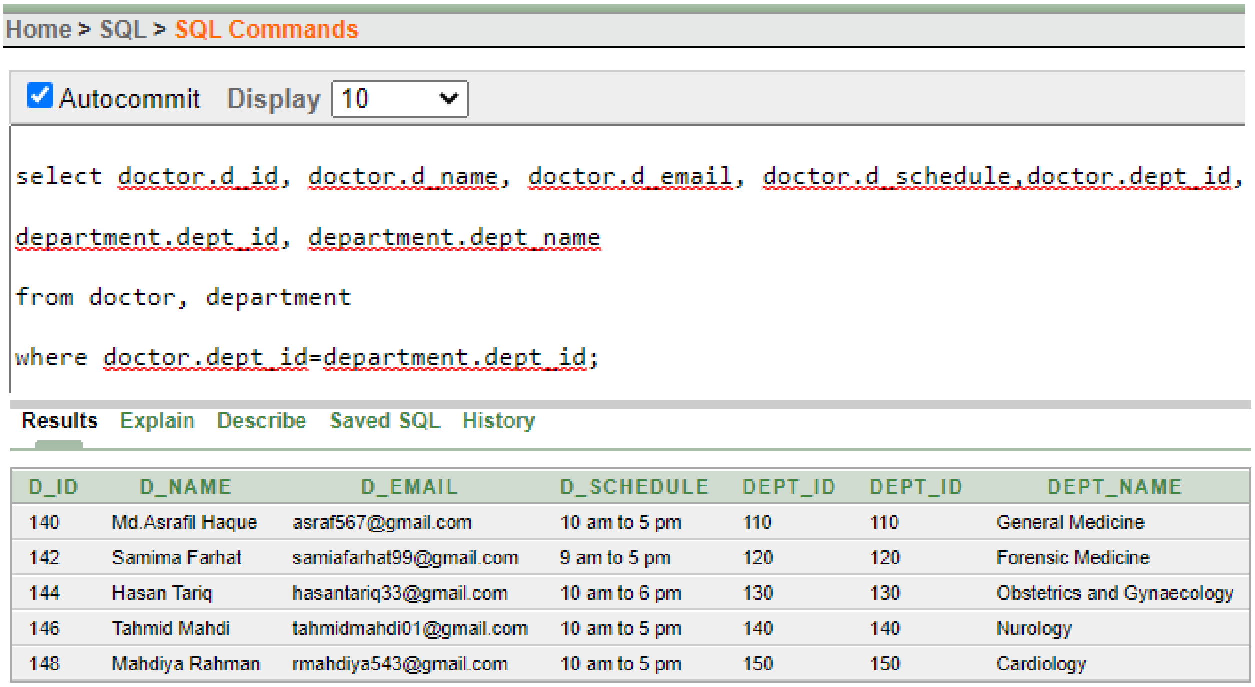
☑ Autocommit Display 10

```
select hospital.h_id, hospital.h_name, hospital.address_id,
address.address_id, address.h_city , address.h_country
from hospital, address
where hospital.address_id=address.address_id;
```

H_ID	H_NAME	ADDRESS_ID	ADDRESS_ID	H_CITY	H_COUNTRY
20	United Hospial Limited	200	200	Dhaka	Bangladesh
22	Benapole Sadar Hospital	202	202	Benapole	India
24	Apollo Hospital Chennai	204	204	Chennai	India
26	Square Hospital	206	206	Dhaka	Bangladesh
32	Cristian Mission Hospital	208	208	Rajshahi	Bangladesh

Joining No2:

```
select doctor.d_id, doctor.d_name, doctor.d_email, doctor.d_schedule,doctor.dept_id,
department.dept_id, department.dept_name
from doctor, department
where doctor.dept_id=department.dept_id;
```



Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

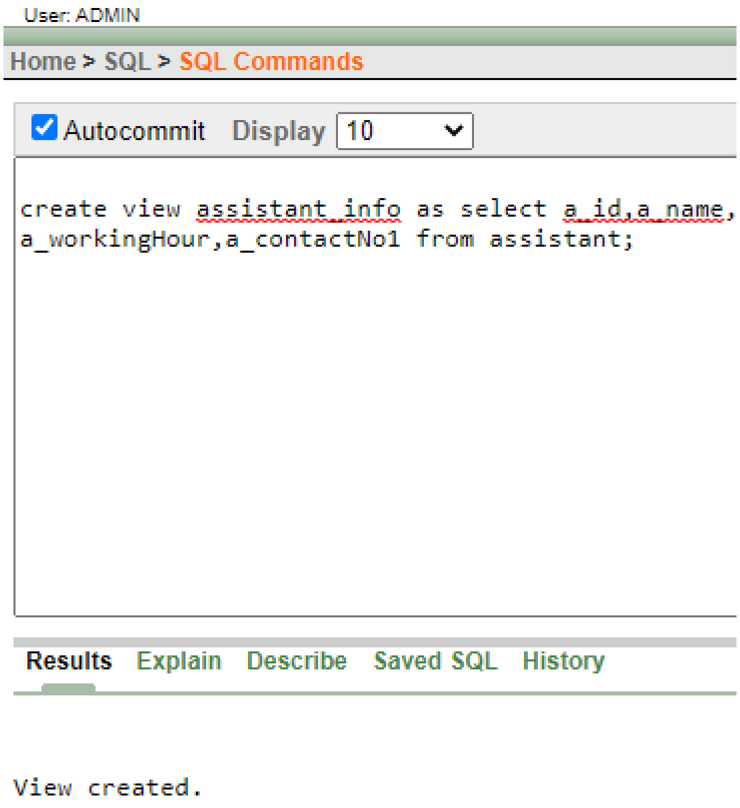
```
select doctor.d_id, doctor.d_name, doctor.d_email, doctor.d_schedule,doctor.dept_id,

department.dept_id, department.dept_name

from doctor, department

where doctor.dept_id=department.dept_id;
```

D_ID	D_NAME	D_EMAIL	D_SCHEDULE	DEPT_ID	DEPT_ID	DEPT_NAME
140	Md.Asrafil Haque	asraf567@gmail.com	10 am to 5 pm	110	110	General Medicine
142	Samima Farhat	samiafarhat99@gmail.com	9 am to 5 pm	120	120	Forensic Medicine
144	Hasan Tariq	hasantariq33@gmail.com	10 am to 6 pm	130	130	Obstetrics and Gynaecology
146	Tahmid Mahdi	tahmidmahdi01@gmail.com	10 am to 5 pm	140	140	Nurology
148	Mahdiya Rahman	rmahdiya543@gmail.com	10 am to 5 pm	150	150	Cardiology

View

View No1:

Create a view assistant_info as select a_id, a_name, a_workingHour, a_contactNo1 from assistant.

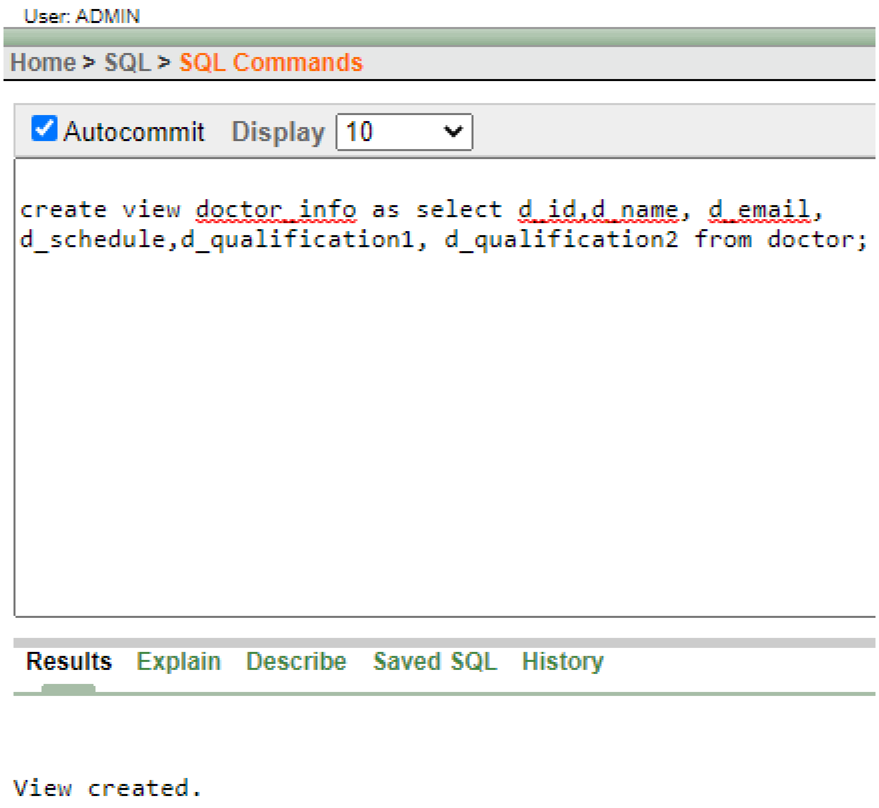


```
create view assistant_info as select a_id,a_name,
a_workingHour,a_contactNo1 from assistant;
```

```
View created.
```

View No2:

Create a view doctor_info as select d_id, d_name, d_email, d_schedule, d_qualification1, d_qualification2 from doctor.



```
create view doctor_info as select d_id,d_name, d_email,
d_schedule,d_qualification1, d_qualification2 from doctor;
```

```
View created.
```

Relational Algebra

No1: Find out m price information from medicine table where m_price>150.

Ans: $\sigma_{m\_price>150}(medicine);$

No2: Project the dept_name, dept_id from the department table.

Ans: $\pi_{dept\_name,dept\_id}(department);$

No3: Fetch p gender from patient table for those patient where p gender=Female.

Ans: $\pi_{p\_gender}(\sigma_{p\_gender=\text{"female"}}(patient));$

No4: Find out a id and a salary from assistant table where a salary=2500tk.

Ans: $\pi_{d\_id,d\_salary}(\sigma_{a\_salary=2500}(assistant));$

No5: Find out st id, st price and st roomno from seat table where st roomno=Not yet booking.

Ans: $\pi_{st\_id,st\_price,st\_roomno}(\sigma_{st\_roomno=\text{"not yet booking"}}(seat));$

Conclusion

In conclusion, the "Doctor Appointment System" database project utilizing Oracle SQL has been successfully implemented. The project aimed to streamline the process of scheduling and managing doctor appointments, and it incorporated several tables such as "Doctor", "Patient", "Doctor's Assistant", "Appointment" , "Medicine", "Hospital", "Doctors Department " and "Seat".

Throughout the project, various key findings were discovered. The system effectively stores and manages information about doctors, patients, appointments, medicines, hospitals, doctor departments, and seating arrangements. It provides the necessary functionality to schedule and track appointments, assign doctors to specific departments, and manage medication records.

To further improve the existing project, we should consider the following future considerations:

1. Enhance User Interface: We should improve the user interface to be more intuitive, user-friendly, and visually appealing. Incorporate modern design principles and usability testing for a seamless experience.

2. Advanced Scheduling Features: We need to implement features like automated appointment reminders, waitlist management, and real-time doctor availability to enhance efficiency and improve patient satisfaction.

3. Integration with Electronic Health Records (EHR): We should explore integrating the appointment system with electronic health records, enabling doctors to access patient medical history and relevant information for informed decisions and better care.

4. Analytics and Reporting: We need to incorporate analytics and reporting capabilities to generate insights about appointment trends, patient demographics, and resource utilization for improved resource allocation.

5. Mobile Application: In future, we try to develop a mobile app for convenient access to the appointment system on smartphones and tablets, enabling patients to schedule, reschedule, or cancel appointments on the go.

6. Security and Privacy Enhancements: We also try to strengthen security measures, including secure authentication, data encryption, and regular audits, to protect patient data and ensure compliance with privacy regulations.

By implementing these future improvements, the Doctor Appointment System can become even more efficient, user-friendly, and feature-rich, ultimately enhancing the overall healthcare experience for both patients and medical professionals.