ABV- Indian Institute of Information Technology,Gwalior

SUBMITTED BY:

GROUP 18

2019IMT-010 Akhil Badoni
2019IMT-023 Astha Goyal
2019IMT-031 Deependra Yadav
2019IMT-073 Piyush Garg
2019IMT-074 Piyush Rajput
2019IMT-105 Sushant Basak

SUBMITTED TO:-

Dr. Debanjan Sadhya

REAL WORLD PROBLEM STATEMENT

A hospital database has been presented in this project, we only placed those values which are very much needed for hospitals in order to function properly. We created this project so that hospitals can be set in more numbers in less amount of time and function properly. This database can also be used in medical camps,military purpose or mini hospitals that are set for temporary basis.

Entity Sets:

1. Doctor
2. Patient
3. Nurse
4. Department
5. Board of members

Relationship Sets

1. Treats: One to many from doctor to patient entity set. (Ex. A heart patient consults the cardiologist of the hospital.)

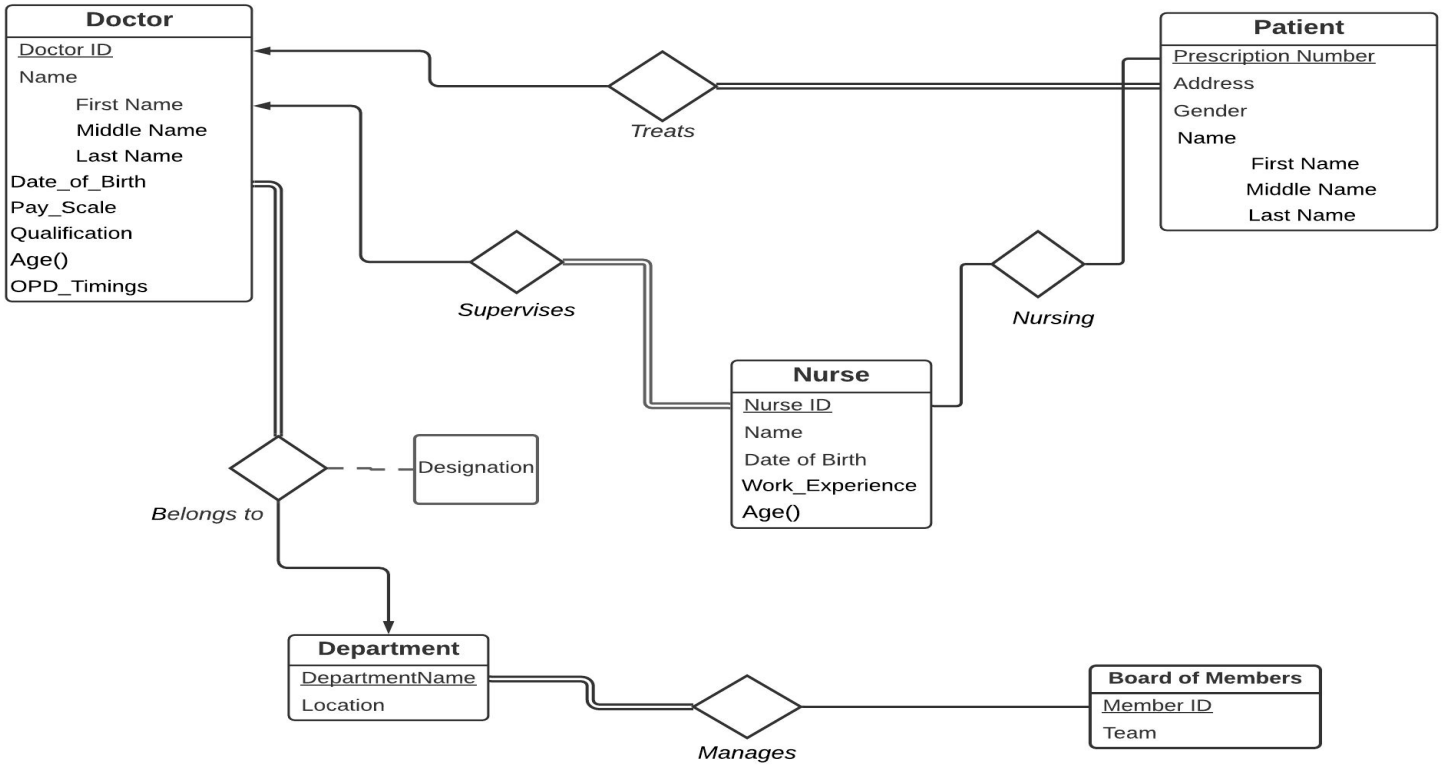
2. Nursing: Many to many from nurse to patient entity set. (Ex. More than one nurse monitor progress report of more than one patient.)

3. Supervises: One to many from doctor to nurse entity set. (Ex. More than one nurse may work under the supervision of only one doctor.)

4. Belongs_to: Many to one from doctor to department entity set. (Ex. Multiple cardiologists belong to the cardiology department of the hospital.)

5. Manages: Many to many from department to board of members entity set. (Ex. The board of members oversees the managerial affairs of all departments of the hospital.)

ER Diagram


Doctor
Doctor ID
Name
First Name
Middle Name
Last Name
Date_of_Birth
Pay_Scale
Qualification
Age()
OPD_Timings
Patient
Prescription Number
Address
Gender
Name
First Name
Middle Name
Last Name
Treats
Supervises
Nursing
Nurse
Nurse ID
Name
Date of Birth
Work_Experience
Age()
Designation
Belongs to
Department
DepartmentName
Location
Manages
Board of Members
Member ID
Team

Initial Schema:-

Entity Sets

DOCTOR(Doctor ID, First Name, Middle Name, Last Name,OPD Timings, Pay_scale, Date of Birth, Qualification)

PATIENT(Prescription Number, Address, First Name, Middle Name, Last Name, Gender)

NURSE(Nurse ID, Name, Work experience, Date of Birth)

DEPARTMENT(Department Name, Location)

BOARD OF MEMBERS(Member ID, Team)

Relationship Sets:-

TREATS(Prescription Number, Doctor ID)

NURSING(Nurse ID, Prescription Number)

SUPERVISES(Nurse ID, Doctor ID)

BELONGS TO(Doctor ID, Department Name,Designation)

MANAGES(Department Name, Member ID)

Schema of **Treats**, **Supervises** and **Belongs to** have :-

- Many to one relationships
- Total participation on many sides

Hence, these schemas can be reduced with the help of entity sets of many sides.

1st Normal Form :- (Atomic Attributes)

Schema:-

DOCTOR (Doctor ID, First Name, Middle Name, Last Name,OPD Timings, Pay_scale, Date of Birth, Qualification, Department, Designation)

PATIENT (Prescription Number, Doctor ID, Address, First Name, Middle Name, Last Name, Gender)

NURSE (Nurse ID, Doctor ID, Name, Work experience, Date of Birth)

DEPARTMENT(Department Name, Location)

BOARD OF MEMBERS (Member ID, Team)

NURSING (Nurse ID, Prescription Number)

MANAGES (Department Name, Member ID)

Functional Dependencies:-

DOCTOR : F={ Doctor ID→R, (OPD Timings,Qualification,Designation)→Pay_Scale}

PATIENT : F={Prescription Number→R}

NURSING : F={(Nurse ID, Prescription Number)→R}

NURSE: F={Nurse ID→R}

DEPARTMENT :F={Department Name→R}

BELONGS TO : F={Doctor ID→R}

BOARD OF MEMBERS :F={Member ID→R}

MANAGES :F={(Department Name, Member Id)→R}

2ND NORMAL FORM :-

As there are no partial dependencies present so there will not be any changes in the schemas. So, Current Schema are already in 2NF.

3rd Normal Form:-

Notice the functional dependency **(OPDTimings, Designation, Qualification) → Pay_scale** is a transitive functional dependency so it should have to be removed so we will decompose Doctor schema into Doctor_Details and Doctor_pay.(**OPD Timings, Designation,Qualification)** will become foregin key in Doctor_details and Doctor_pay will contain **(OPDTimings, Designation, Qualification)** as primary key .

Schema:-

***Doctor_Details** (Doctor ID, First Name, Middle Name, Last Name,OPD Timings, Date of Birth, Qualification, Department, Designation)

***Doctor_Pay** (OPD Timings, Designation,Qualification,Pay_scale)

PATIENT (Prescription Number, Doctor ID, Address, First Name, Middle Name, Last Name, Gender)

NURSE (Nurse ID, Doctor ID, Name, Work experience, Date of Birth)

DEPARTMENT(Department Name, Location)

BOARD OF MEMBERS (Member ID, Team)

NURSING (Nurse ID, Prescription Number)

MANAGES (Department Name, Member ID)

Functional Dependencies

***DOCTOR_DETAILS**: F={Doctor ID→R}

***DOCTOR_PAY**: F={OPD Timings, Designation,Qualification→R}

PATIENT: F={Prescription Number→R}

NURSE: F={Nurse ID→R}

DEPARTMENT: F={Department Name→R}

BOARD OF MEMBERS: F={Member ID→R}

MANAGES: F={(Department Name, Member Id)→R

NURSING: F={(Nurse ID, Prescription Number)→R}

BCNF(Boyce -Codd Normal Form) :-

A relation schema R is in BCNF with respect to a set F of functional dependencies if for all functional dependencies in F+ of the form **α→ β** where **α ⊆** R and **β ⊆** R, at least one of the following holds:

- **α → β** is trivial (i.e., **β ⊆ α**)
- **α** is a superkey for R

Since previous schemas follows all the condition of BCNF so the it is also in BCNF.

SQL Tables

department_name	location
Ayurveda	1
Cardiology	4
Dental	6
Eye	2
Homeopathy	3
Surgery	5
NULL	NULL

department

doctor_pay

opd_timings	qualification	designation	pay_scale
00:00-8:00	MBBS	Junior	140000
10:00-15:00	MD	Visiting	75000
10:00-16:00	MBBS	Junior	80000
10:00-16:00	MD	Junior	100000
12:00-16:00	MD	Senior	150000
12:00-18:00	MBBS	Junior	100000
15:00-20:00	MD	HOD	195000
20:00-00:00	MBBS	Junior	115000
9:00-12:00	DNB	HOD	200000
9:00-13:00	MD	Visiting	50000
9:00-14:00	MD	HOD	180000
NULL	NULL	NULL	NULL

doctor_details

doctor_id	first_name	middle_name	last_name	opd_timings	dob	qualification	designation	department_name
1	Leonard	NULL	Hofstader	15:00-20:00	1992-05-15	MD	HOD	Cardiology
2	Howard	Joel	Wolowitz	20:00-00:00	1990-08-06	MBBS	Junior	Ayurveda
3	Rajesh	Kumar	Dogra	9:00-14:00	1966-02-10	DNB	HOD	Ayurveda
4	kakashi	NULL	Hatake	12:00-16:00	1979-01-01	MD	Senior	Eye
5	Ross	NULL	Geller	9:00-12:00	1975-09-11	DNB	HOD	Surgery
6	Jamie	NULL	Lannister	9:00-12:00	1977-05-10	DNB	HOD	Dental
7	John	NULL	Snow	00:00-8:00	1985-05-10	MBBS	Junior	Ayurveda
8	Rachel	Karen	Green	9:00-14:00	1975-01-11	DNB	HOD	Homeopathy
9	Mukesh	NULL	Tripathi	15:00-20:00	1968-03-03	MD	HOD	Eye
10	Oona	NULL	Chaplin	10:00-16:00	1977-10-05	MD	Junior	Surgery
11	Light	NULL	Yagami	9:00-13:00	1980-10-05	MD	Visiting	Dental
12	Sheldon	Lee	Cooper	10:00-15:00	1977-05-18	MD	Visiting	Homeopathy
13	Lalit	Kumar	Khatri	00:00-8:00	1985-04-08	MBBS	Junior	Eye
14	Charlie	NULL	Sheen	12:00-16:00	1985-10-10	MD	Senior	Cardiology
15	Aman	Singh	Sodhi	12:00-18:00	1975-09-11	MBBS	Junior	Homeopathy
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

prescription_number	doctor_id	address	first_name	middle_name	last_name	gender
101	1	2792 Conference Center Way	Wayne	A	Judge	M
102	1	4936 Broadway Avenue	Anna	R	Langston	F
103	2	2823 Burwell Heights Road	Carl		Klein	M
104	3	4936 Broadway Avenue	Ricardo	C	Sutton	F
105	7	4822 Meadow View Drive	Gayatri	Yadunandan	Chaudhry	F
106	3	4538 Romrog Way	Anand	Anil	Warrior	F
107	2	3028 Pine Garden Lane	Devendra	NULL	Badal	M
108	9	3112 Black Oak Hollow Road	Wafiq	NULL	Kaur	F
109	2	1804 Woodside Circle	David	NULL	Chowdhury	M
110	6	378 Heliport Loop	Anand	Lal	Meda	F
NULL	NULL	NULL	NULL	NULL	NULL	NULL

patient

nurse

nurse_id	doctor_id	name	work_experience	dob
51	9	Shweta	6	1982-12-12
52	4	Singh	5	1995-06-07
53	8	Sally	4	1991-12-12
54	5	Tridevi	3	1998-03-07
55	5	Prerna	3	1992-08-15
56	7	Sunidhi	2	1990-02-12
57	8	Shivani	8	1996-09-03
58	3	Anuradha	5	1985-09-26
59	6	Rebecca	6	1993-08-15
60	5	Hermoine	5	1986-06-23
61	8	Akansha	2	1996-10-12
62	5	Rose	1	2000-02-12
63	1	Sharma	6	1994-05-28
64	2	Kumari	2	1998-11-13
65	4	Priya	5	1997-01-26
NULL	NULL	NULL	NULL	NULL

nursing

nurse_id	prescription_number
51	101
54	102
52	103
53	104
55	105
60	106
56	108
58	109
59	110
NULL	NULL

board_of_members

member_id	team
501	Hiring
502	Sanitation
503	ICU
504	Waste Management
505	Administration
506	Hiring
507	laboratory
NULL	NULL

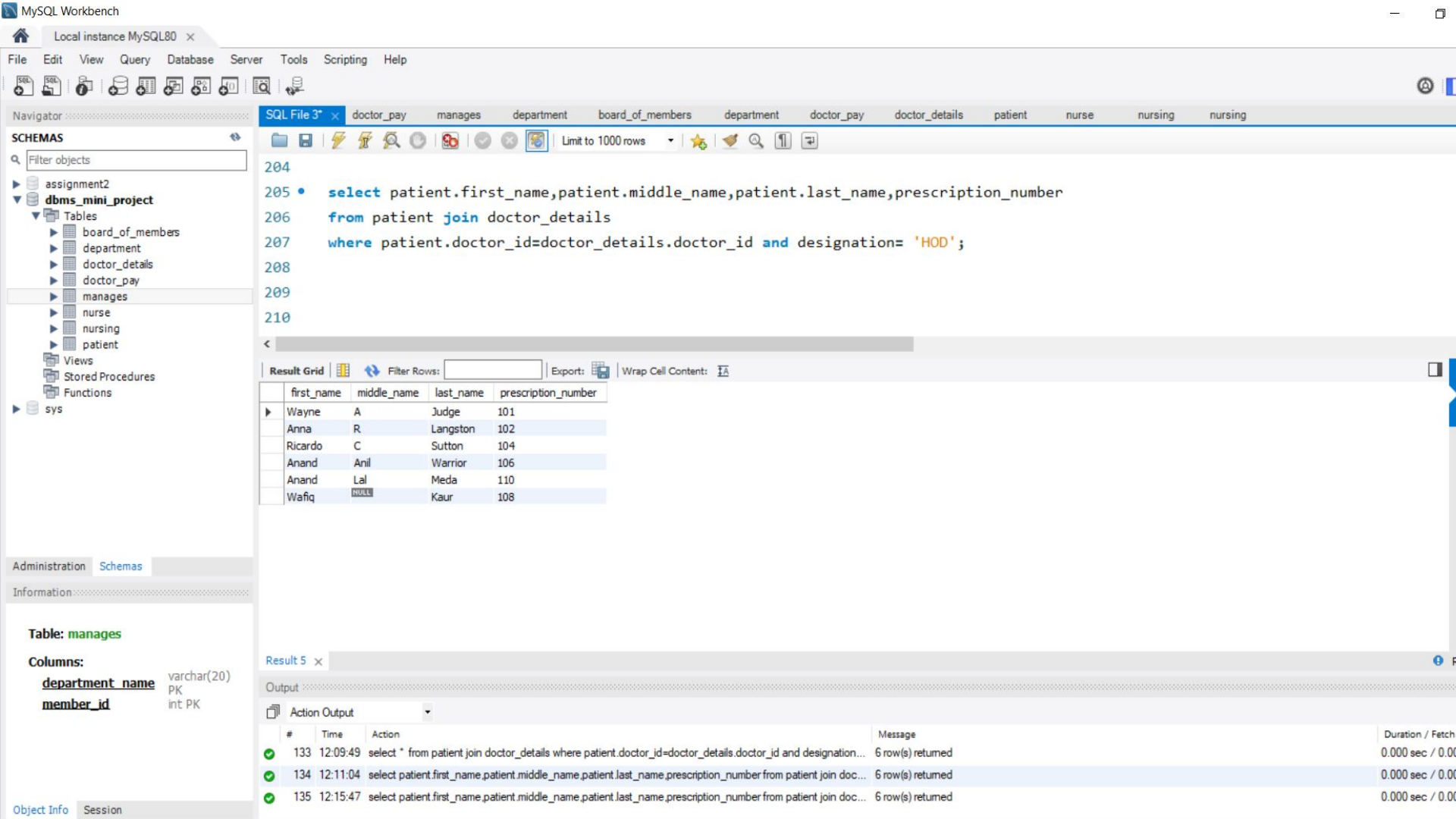
manages

department_name	member_id
Ayurveda	501
Dental	506
Cardiology	502
Eye	503
Homeopathy	504
Surgery	507
Cardiology	505
Homeopathy	501
Dental	503
NULL	NULL

SQL Queries

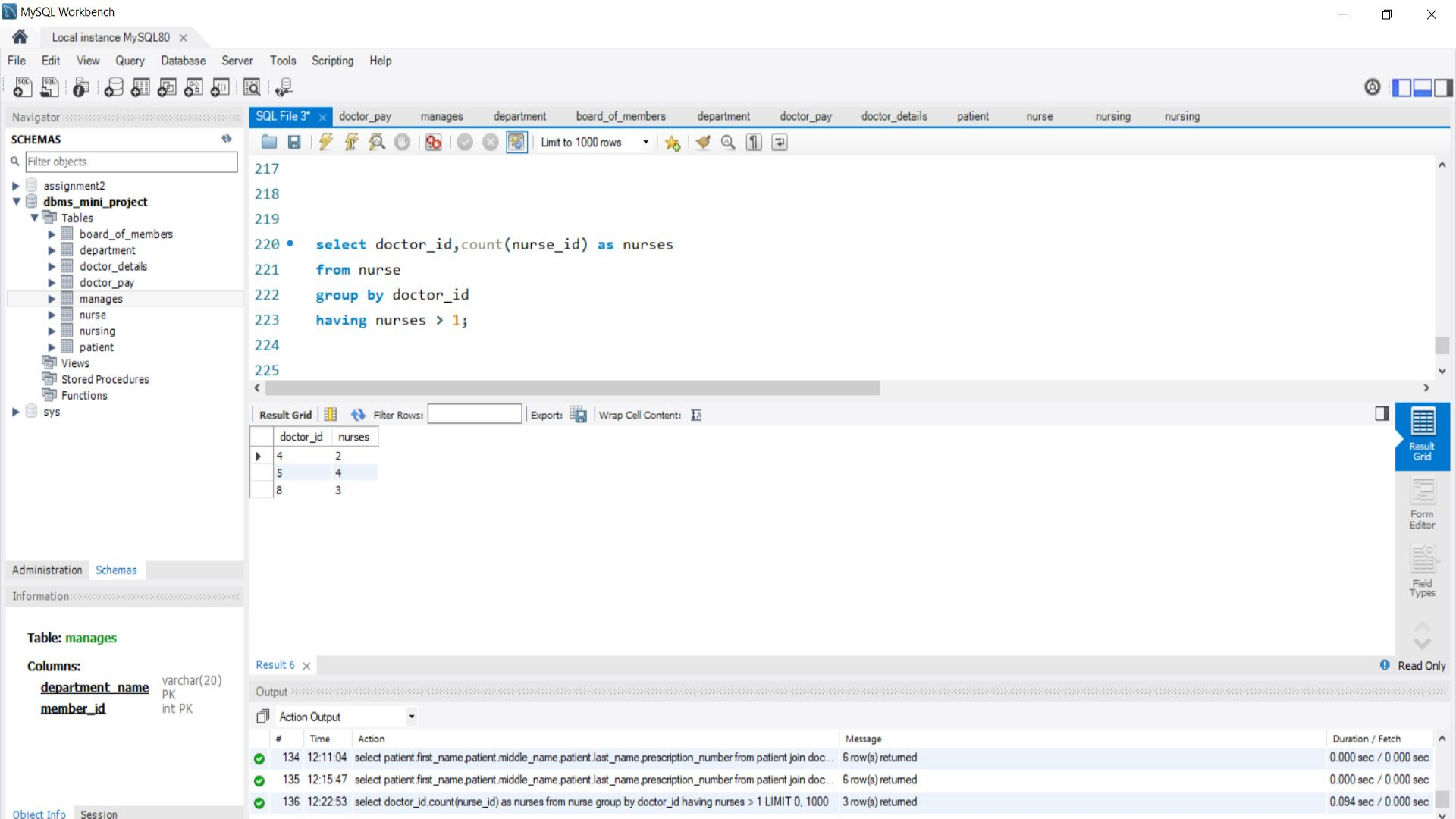
Query 1 - Patients who are treated by HOD

```
select patient.first_name,patient.middle_name,patient.last_name,prescription_number
from patient join doctor_details
where patient.doctor_id=doctor_details.doctor_id and designation= 'HOD';
```


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
assignment2
dbms_mini_project
Tables
board_of_members
department
doctor_details
doctor_pay
manages
nurse
nursing
patient
Views
Stored Procedures
Functions
sys
Administration Schemas
Information
Table: manages
Columns:
department_name varchar(20) PK
member_id int PK
Object Info Session
SQL File 3* doctor_pay manages department board_of_members department doctor_pay doctor_details patient nurse nursing nursing
Limit to 1000 rows
204
205 • select patient.first_name,patient.middle_name,patient.last_name,prescription_number
206 from patient join doctor_details
207 where patient.doctor_id=doctor_details.doctor_id and designation= 'HOD';
208
209
210
Result Grid Filter Rows: Export: Wrap Cell Content:
first_name middle_name last_name prescription_number
Wayne A Judge 101
Anna R Langston 102
Ricardo C Sutton 104
Anand Anil Warrior 106
Anand Lal Meda 110
Wafiq NULL Kaur 108
Result 5
Output
Action Output
Time Action Message Duration / Fetch
133 12:09:49 select * from patient join doctor_details where patient.doctor_id=doctor_details.doctor_id and designation... 6 row(s) returned 0.000 sec / 0.00
134 12:11:04 select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc... 6 row(s) returned 0.000 sec / 0.00
135 12:15:47 select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc... 6 row(s) returned 0.000 sec / 0.00

Query 2- Doctors who supervises more than one nurses

```
select doctor_id,count(nurse_id) as nurses
from nurse
group by doctor_id
having nurses > 1
```


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
assignment2
dbms_mini_project
Tables
board_of_members
department
doctor_details
doctor_pay
manages
nurse
nursing
patient
Views
Stored Procedures
Functions
sys
Administration Schemas
Information
Table: manages
Columns:
department_name varchar(20) PK
member_id int PK
Object Info Session
SQL File 3* doctor_pay manages department board_of_members department doctor_pay doctor_details patient nurse nursing nursing
Limit to 1000 rows
217
218
219
220 • select doctor_id,count(nurse_id) as nurses
221 from nurse
222 group by doctor_id
223 having nurses > 1;
224
225
Result Grid Filter Rows: Export: Wrap Cell Content:
doctor_id nurses
4 2
5 4
8 3
Result Grid
Form Editor
Field Types
Result 6
Read Only
Output
Action Output
Time Action Message Duration / Fetch
134 12:11:04 select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc... 6 row(s) returned 0.000 sec / 0.000 sec
135 12:15:47 select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc... 6 row(s) returned 0.000 sec / 0.000 sec
136 12:22:53 select doctor_id,count(nurse_id) as nurses from nurse group by doctor_id having nurses > 1 LIMIT 0, 1000 3 row(s) returned 0.094 sec / 0.000 sec

Query 3- Patients who are treated by a Cardiologist having a designation HOD.

```
Select patient.first_name,patient.middle_name,patient.last_name,prescription_number
from patient join doctor_details
where patient.doctor_id=doctor_details.doctor_id and department_name='Cardiology'
and designation='HOD';
```

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- assignment2
- **dbms_mini_project**
  - Tables
    - board_of_members
    - department
    - doctor_details
    - doctor_pay
    - manages
    - nurse
    - nursing
    - patient
  - Views
  - Stored Procedures
  - Functions
- sys

Administration Schemas

Information

Table: manages

Columns:

department_name	varchar(20) PK
member_id	int PK

Object Info Session

SQL File 3* | doctor_pay | manages | department | board_of_members | department | doctor_pay | patient | nurse | nursing | nursing | doctor_details

Limit to 1000 rows

```
227
228
229
230 •  select patient.first_name,patient.middle_name,patient.last_name,prescription_number
231    from patient join doctor_details
232    where patient.doctor_id=doctor_details.doctor_id and department_name='Cardiology' and designation='HOD';
233
234
235
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

first_name	middle_name	last_name	prescription_number
Wayne	A	Judge	101
Anna	R	Langston	102

Result Grid | Form Editor | Field Types

Result 8

Read Only

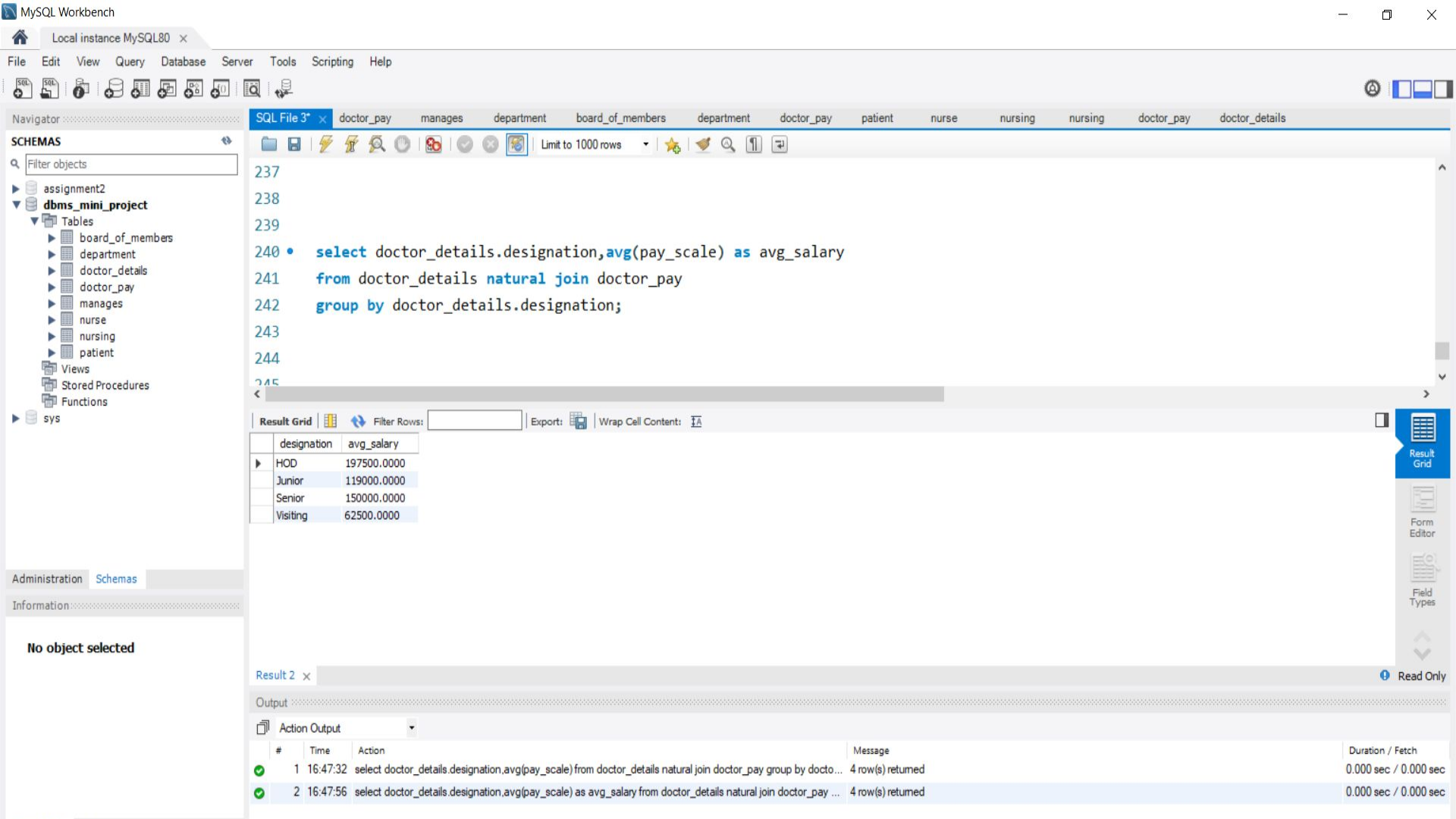
Output

Action Output

#	Time	Action	Message	Duration / Fetch
137	12:29:54	SELECT * FROM dbms_mini_project.doctor_details LIMIT 0, 1000	15 row(s) returned	0.000 sec / 0.000 sec
138	12:30:39	select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc...	0 row(s) returned	0.063 sec / 0.000 sec
139	12:33:01	select patient.first_name,patient.middle_name,patient.last_name,prescription_number from patient join doc...	2 row(s) returned	0.000 sec / 0.000 sec

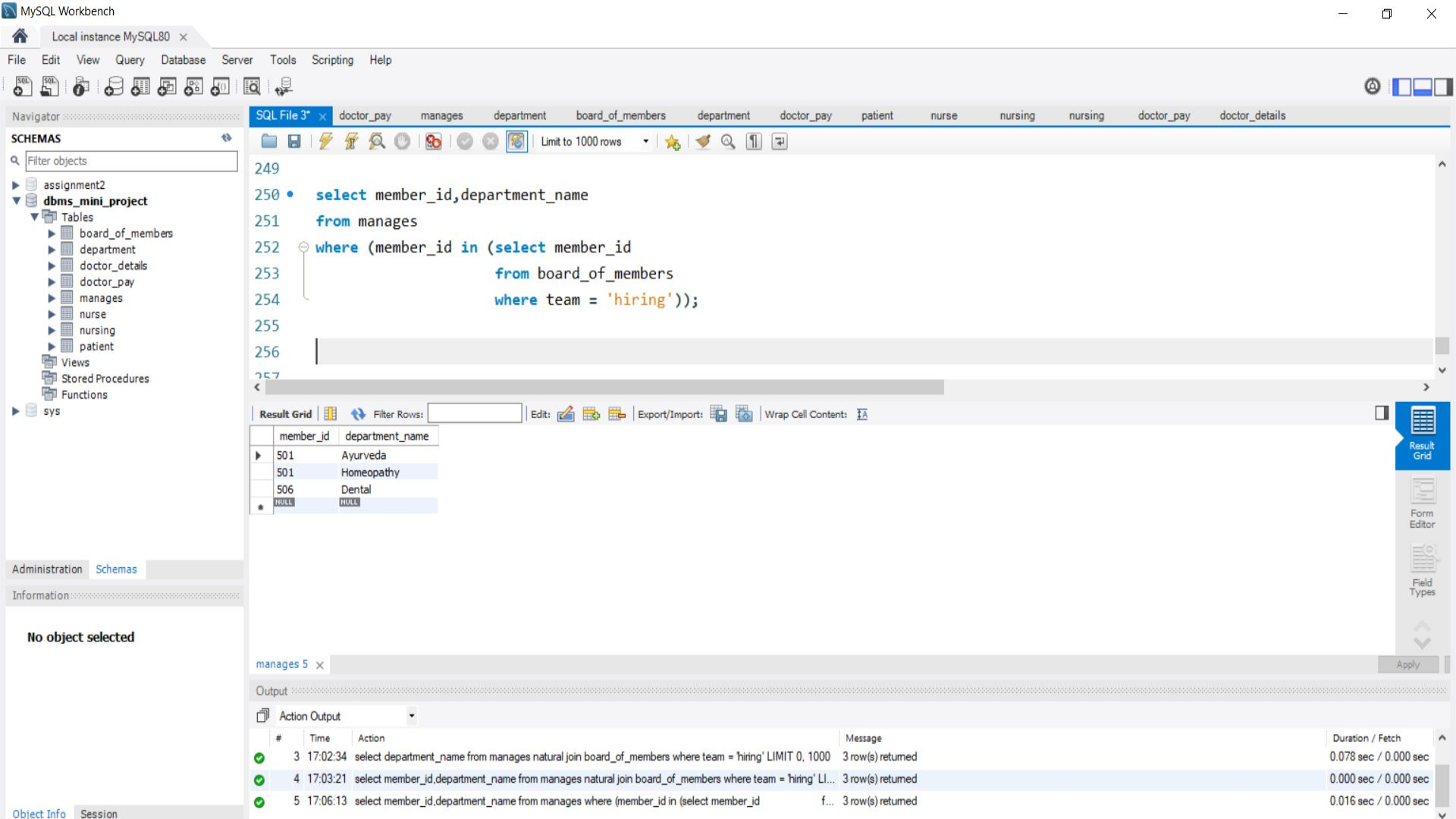
Query 4- Find the average salary of each designations of doctors..

```
select doctor_details.designation,avg(pay_scale) as avg_salary
from doctor_details natural join doctor_pay
group by doctor_details.designation;
```


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
assignment2
dbms_mini_project
Tables
board_of_members
department
doctor_details
doctor_pay
manages
nurse
nursing
patient
Views
Stored Procedures
Functions
sys
Administration Schemas
Information
No object selected
SQL File 3* doctor_pay manages department board_of_members department doctor_pay patient nurse nursing nursing doctor_pay doctor_details
Limit to 1000 rows
237
238
239
240 • select doctor_details.designation,avg(pay_scale) as avg_salary
241 from doctor_details natural join doctor_pay
242 group by doctor_details.designation;
243
244
Result Grid Filter Rows: Export: Wrap Cell Content:
designation avg_salary
HOD 197500.0000
Junior 119000.0000
Senior 150000.0000
Visiting 62500.0000
Result Grid
Form Editor
Field Types
Result 2
Read Only
Output
Action Output
Time Action Message Duration / Fetch
1 16:47:32 select doctor_details.designation,avg(pay_scale) from doctor_details natural join doctor_pay group by docto... 4 row(s) returned 0.000 sec / 0.000 sec
2 16:47:56 select doctor_details.designation,avg(pay_scale) as avg_salary from doctor_details natural join doctor_pay ... 4 row(s) returned 0.000 sec / 0.000 sec

Query 5- Find the department name of the members of the hiring team.

```
select member_id,department_name
from manages
where (member_id in (select member_id
                     from board_of_members
                                  where team = 'hiring'));
```


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
assignment2
dbms_mini_project
Tables
board_of_members
department
doctor_details
doctor_pay
manages
nurse
nursing
patient
Views
Stored Procedures
Functions
sys
Administration Schemas
Information
No object selected
Object Info Session
SQL File 3* doctor_pay manages department board_of_members department doctor_pay patient nurse nursing nursing doctor_pay doctor_details
Limit to 1000 rows
249
250 • select member_id,department_name
251 from manages
252 where (member_id in (select member_id
253 from board_of_members
254 where team = 'hiring'));
255
256
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
member_id department_name
501 Ayurveda
501 Homeopathy
506 Dental
NULL NULL
Result Grid
Form Editor
Field Types
manages 5
Apply
Output
Action Output
Time Action Message Duration / Fetch
3 17:02:34 select department_name from manages natural join board_of_members where team = 'hiring' LIMIT 0, 1000 3 row(s) returned 0.078 sec / 0.000 sec
4 17:03:21 select member_id,department_name from manages natural join board_of_members where team = 'hiring' LI... 3 row(s) returned 0.000 sec / 0.000 sec
5 17:06:13 select member_id,department_name from manages where (member_id in (select member_id f... 3 row(s) returned 0.016 sec / 0.000 sec

Relax Queries :-

Query 1- Nurse having work experience>5

$$\pi\ \text{nurse\_id}\ (\ \sigma\ \text{work\_experience} > 5\ (\ \text{Nurse}\ )\ )$$

https://dbis-uibk.github.io/relax/calc/local/uibk/local/0

60%

Calculator Language

Select DB (Hospital_Management_Sy...

Department
- department_name string
- location number

Nursing
- nurse_id number
- prescription_number number

Nurse
- nurse_id number
- doctor_id number
- name string
- work_experience number
- dob string

Manages
- department_name string
- member_id number

Doctor_Pay
- opd_timings string
- qualification string
- designation string
- pay_scale number

doctor_details
- doctor_id number
- first_name string
- middle_name string
- last_name string
- opd_timings string
- dob string
- qualification string
- designation string
- department_name string

Patient
- prescription_number number
- doctor_id number
- address string
- first_name string
- middle_name string
- last_name string
- gender string

Relational Algebra | SQL | Group Editor

π σ ρ ← → τ γ ∧ ∨ ¬ = ≠ ≥ ≤ ∩ ∪ ÷ - × ⋈ ⋉ ⋊ ⟕ ⟖ ⟗ ▷ = – {}

```
1 π nurse_id ( σ work_experience > 5 ( Nurse ) )
```

execute query

download history

π nurse_id (σ work_experience > 5 (Nurse))

Nurse.nurse_id
57
59
63
51

Query2 - first_name and doctor_id of doctor who are HOD

π doctor_id , first_name (σ designation='HOD' (doctor_details))

department_name string
location number

Nursing
nurse_id number
prescription_number number

Nurse
nurse_id number
doctor_id number
name string
work_experience number
dob string

Manages
department_name string
member_id number

Doctor_Pay
opd_timings string
qualification string
designation string
pay_scale number

doctor_details
doctor_id number
first_name string
middle_name string
last_name string
opd_timings string
dob string
qualification string
designation string
department_name string

Patient
prescription_number number
doctor_id number
address string
first_name string
middle_name string
last_name string
gender string

```
1 π doctor_id , first_name (σ designation='HOD' (doctor_details))
```

execute query

download history

π doctor_id, first_name (σ designation = 'HOD' (doctor_details))

doctor_details.doctor_id	doctor_details.first_name
3	'Rajesh'
5	'Ross'
6	'Jamie'
8	'Rachel'
9	'Mukesh'
1	'Leonard'

Query 3 - Find all the Male Patients who are treated by doctor having doctor id 2.

$$\sigma_{\text{gender='M'}}(\text{Patient}) \cap \sigma_{\text{doctor\_id} = 2}(\text{Patient})$$

https://dbis-uibk.github.io/relax/calc/local/uibk/local/0

60%

Calculator Language

Department
department_name string
location number

Nursing
nurse_id number
prescription_number number

Nurse
nurse_id number
doctor_id number
name string
work_experience number
dob string

Manages
department_name string
member_id number

Doctor_Pay
opd_timings string
qualification string
designation string
pay_scale number

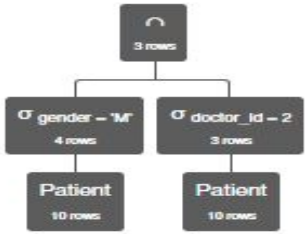
doctor_details
doctor_id number
first_name string
middle_name string
last_name string
opd_timings string
dob string
qualification string
designation string
department_name string

Patient
prescription_number number
doctor_id number
address string
first_name string
middle_name string
last_name string
gender string

```
1  σ gender='M' (Patient) ∩ σ doctor_id = 2 (Patient)
```

execute query

download history



$$\sigma_{\text{gender = 'M'}}(\text{Patient}) \cap \sigma_{\text{doctor\_id = 2}}(\text{Patient})$$

_number	Patient.doctor_id	Patient.address	Patient.first_name	Patient.middle_name	Patient.last_name	Patient.gender
	2	'2823 Burwell Heights Road'	'Carl'	'C'	'Klein'	'M'
	2	'3028 Pine Garden Lane'	'Devendra'	'NULL'	'Badal'	'M'
	2	'1804 Woodside Circle'	'David'	'Lal'	'Chowdhury'	'M'

Query 4 - Find all the doctors along with details whose qualification is MBBS.

σ qualification = 'MBBS' (doctor_details))

https://dbis-uibk.github.io/relax/calc/local/uibk/local/0

60%

Calculator Language

Department
- department_name string
- location number

Nursing
- nurse_id number
- prescription_number number

Nurse
- nurse_id number
- doctor_id number
- name string
- work_experience number
- dob string

Manages
- department_name string
- member_id number

Doctor_Pay
- opd_timings string
- qualification string
- designation string
- pay_scale number

doctor_details
- doctor_id number
- first_name string
- middle_name string
- last_name string
- opd_timings string
- dob string
- qualification string
- designation string
- department_name string

Patient
- prescription_number number
- doctor_id number
- address string
- first_name string
- middle_name string
- last_name string
- gender string

```
1 σ qualification = 'MBBS' (doctor_details)
```

execute query

download history

$\sigma_{\text{qualification = 'MBBS'}}(\text{doctor\_details})$

doctor_details.doctor_id	doctor_details.first_name	doctor_details.middle_name	doctor_details.last_name	doctor_details.c
2	'Howard'	'Joel'	'Wolowitz'	'20:00-0(
7	'John'	'NULL'	'Snow'	'00:00-8
13	'Lalit'	'Kumar'	'Khatri'	'00:00-8
15	'Aman'	'Singh'	'Sodhi'	'12:00-1{

‹ 1 ›

Query 5-Doctor names and doctor_id whose salary is more than 150000

π doctor_id , first_name (σ pay_scale > 150000 (doctor_details ⋈ Doctor_Pay))

https://dbis-uibk.github.io/relax/calc/local/uibk/local/0

50%

RelaX

Calculator | Language | Feedback | Help | Imprint

Select DB (Hospital_Management_System)

Department
- department_name string
- location number

Nursing
- nurse_id number
- prescription_number number

Nurse
- nurse_id number
- doctor_id number
- name string
- work_experience number
- dob string

Manages
- department_name string
- member_id number

Doctor_Pay
- opd_timings string
- qualification string
- designation string
- pay_scale number

doctor_details
- doctor_id number
- first_name string
- middle_name string
- last_name string
- opd_timings string
- dob string
- qualification string
- designation string
- department_name string

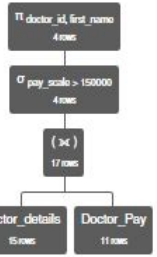
Patient
- prescription_number number
- doctor_id number
- address string
- first_name string
- middle_name string
- last_name string
- gender string

Relational Algebra | SQL | Group Editor

```
1 π doctor_id , first_name (σ pay_scale > 150000 (doctor_details ⋉ Doctor_Pay))
```

execute query

download | history



$$\pi_{\text{doctor\_id, first\_name}}(\sigma_{\text{pay\_scale} > 150000}(\text{doctor\_details} \ltimes \text{Doctor\_Pay}))$$

doctor_details.doctor_id	doctor_details.first_name
5	'Ross'
6	'Jamie'
9	'Mukesh'
1	'Leonard'

‹ 1 ›

Links :-

Relax Code:-

ps://docs.google.com/document/d/1FcCZW9KEEpgT-lh3dRmiqUEUadoU8Xm5ub2uWoYSCBE/edit?usp=sharinghtt

SQL Code:-

https://docs.google.com/document/d/1bjmQekRX19InDF4yX8dOX4_3I8Vafwgm6BIRFRpxAIs/edit?usp=sharing

Er Diagram:-

https://lucid.app/lucidchart/invitations/accept/2ecf241b-477f-49b6-a9e4-1bbddb9b66f2