BENNETT UNIVERSITY

HOSPITAL DATABASE MANAGEMENT

TABLE OF CONTENTS

Problem Statement 1

2 Foreign Key Relationships

Hospital DB QuestionS 3

4 Additional QuestionS

PROBLEM STATEMENT

Abstract:

Structured Query Language (SQL) plays a critical role in managing hospital databases by providing an efficient and systematic way to store, retrieve, and manipulate healthcare-related data. In a hospital setting, maintaining patient records, doctor schedules, appointments, medical histories, and billing details is crucial for operational efficiency. SQL enables data integrity, security, and quick access to information through its powerful querying capabilities. It helps generate reports that aid in decision-making, such as identifying high-risk patients, tracking revenue, and managing resource allocation. By leveraging SQL, hospitals can ensure smooth workflow automation, minimize errors, and enhance patient care through real-time data management.

Objective:

You are tasked with designing and implementing a Hospital Database Management System using MySQL. The goal is to create a well-structured relational database that stores hospital-related data and allows efficient querying to generate reports and derive insights.

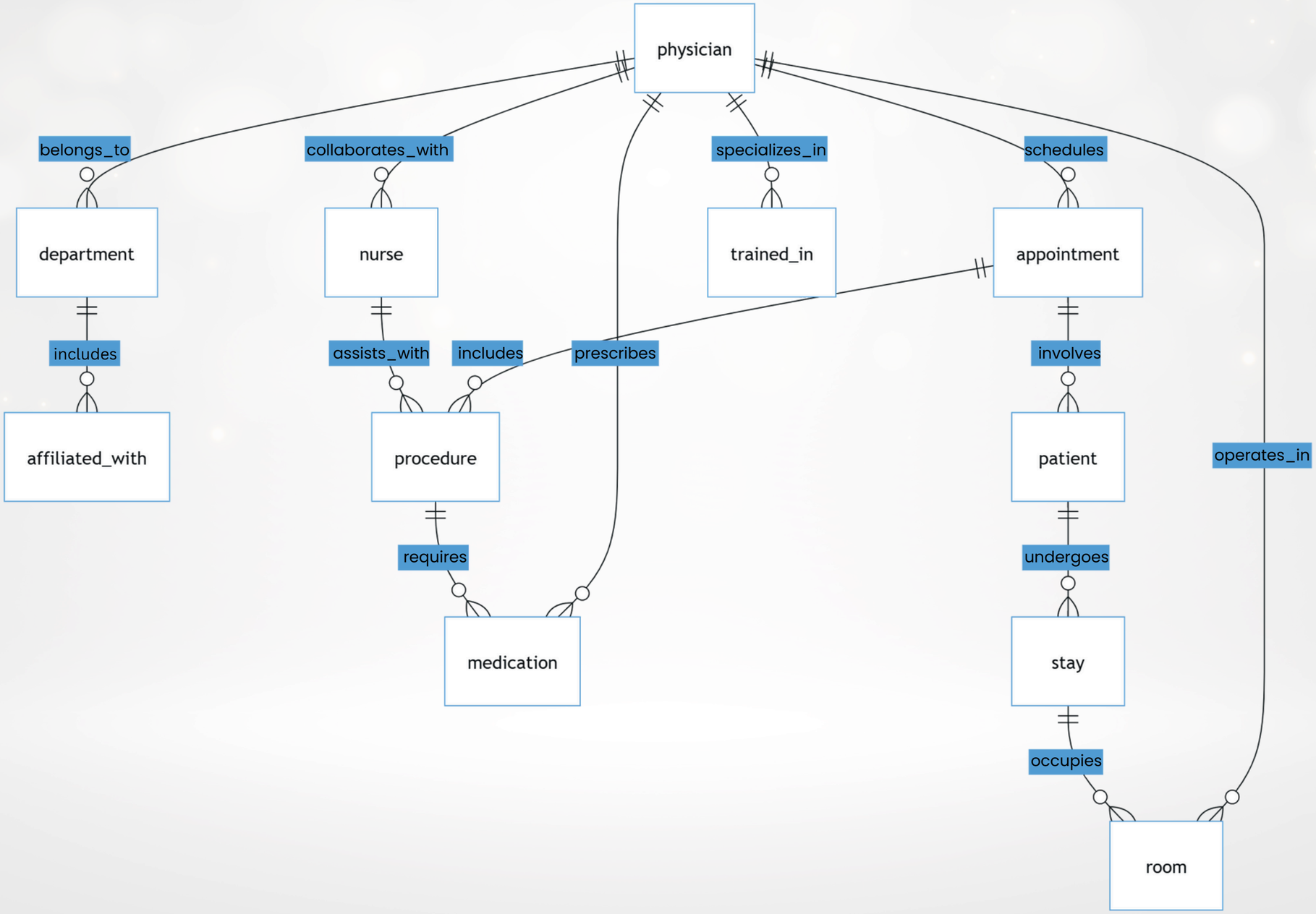
Foreign Key Relationships

- Overview of Referential Integrity Between Tables

Purpose of Foreign Keys

- Ensure Data Integrity: Foreign keys maintain consistency by linking related data across tables, preventing invalid references (e.g., a patient's pcp must reference an existing physician).
- Enforce Relationships: They define and enforce relationships between tables (e.g., one-to-many between physician and patient), ensuring accurate data connections.
- Support Data Consistency: Foreign keys prevent orphaned records (e.g., no stay record can exist for a non-existent patient).
- Facilitate Queries: They enable efficient joins to combine data from multiple tables for reporting and analysis.

Foreign Key Diagram (Visual Representation)



Foreign Keys Core Tables (Patient, Stay, Department, Appointment)

Table	Foreign Key	References Table	Primary Key
patient	pcp	physician	employeeid
department	head	physician	employeeid
stay	patient	patient	ssn
stay	room	room	roomnumber
appointment	patient	patient	ssn
appointment	physician	physician	employeeid
appointment	prepnurse	nurse	employeeid
appointment	appointment	appointment	appointmentid

Procedure-Related Tables (Undergoes, Trained_in, Affiliated_with, On_call)

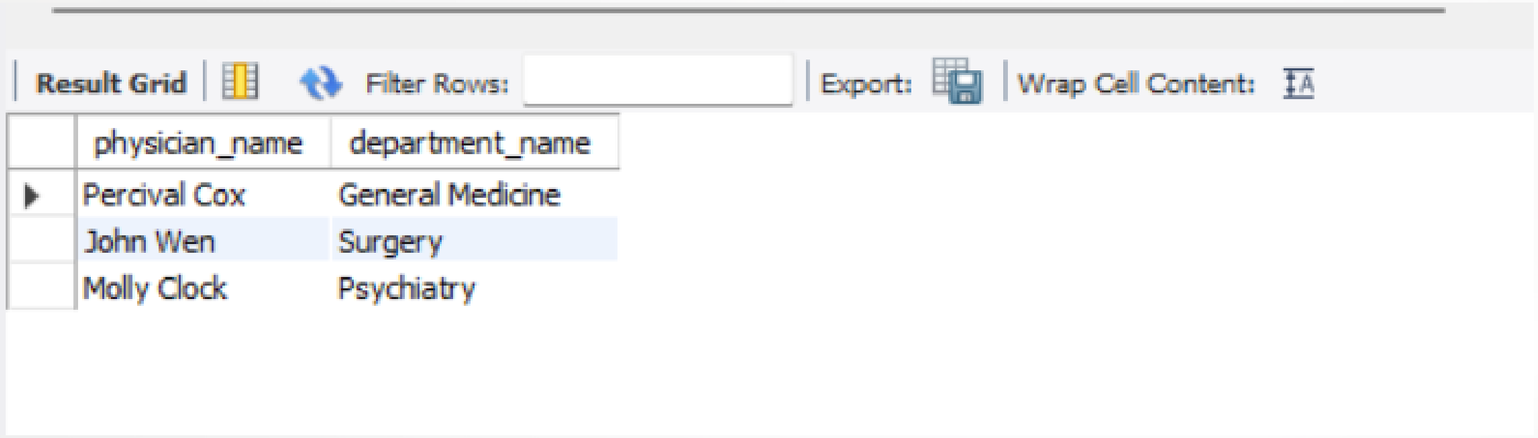
Table	Foreign Key	References Table	Primary Key
undergoes	patient	patient	ssn
undergoes	procedure	procedure	code
undergoes	stay	stay	stayid
undergoes	physician	physician	employeeid
undergoes	assistingnurse	nurse	employeeid
trained_in	physician	physician	employeeid
trained_in	treatment	procedure	code
affiliated_with	physician	physician	employeeid
affiliated_with	department	department	departmentid
on_call	nurse	nurse	employeeid

QUESTIONS

1. Write a SQL query to identify the physicians who are the department heads.

```
55      -- Question-1
56
57      SELECT p.name AS physician_name, d.name AS department_name
58      FROM physician p
59      JOIN department d ON p.employeeid = d.head;
60
```

OUTPUT



physician_name	department_name
Percival Cox	General Medicine
John Wen	Surgery
Molly Clock	Psychiatry

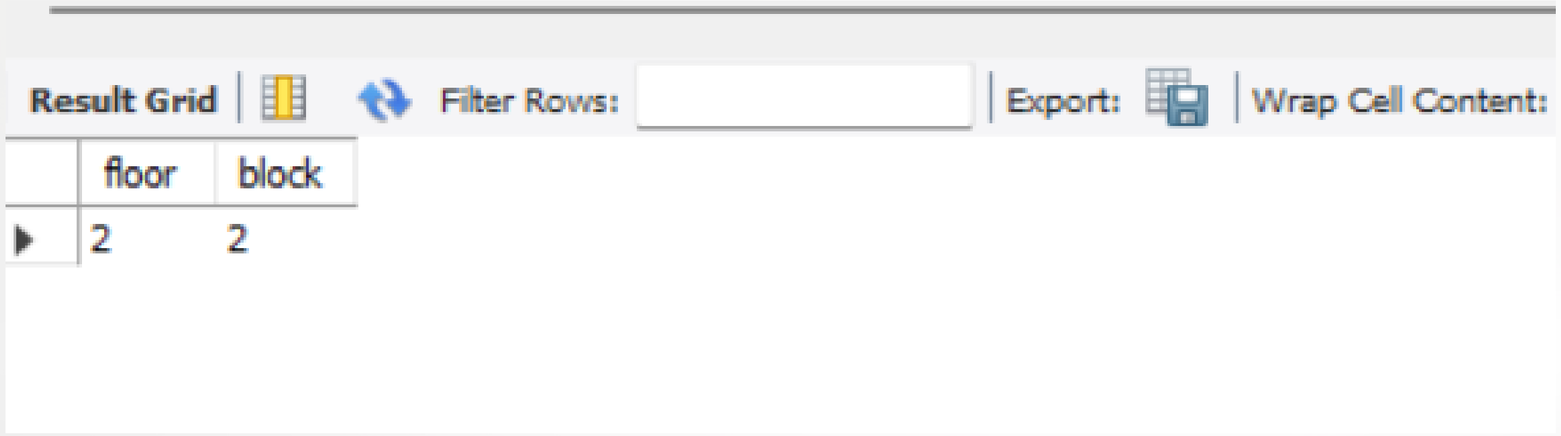
QUESTIONS

2. Write a SQL query to locate the floor and block where room number 212 is located.

```sql
-- Question-2

SELECT blockfloor AS floor, blockcode AS block
FROM room
WHERE roomnumber = 212;
```

OUTPUT



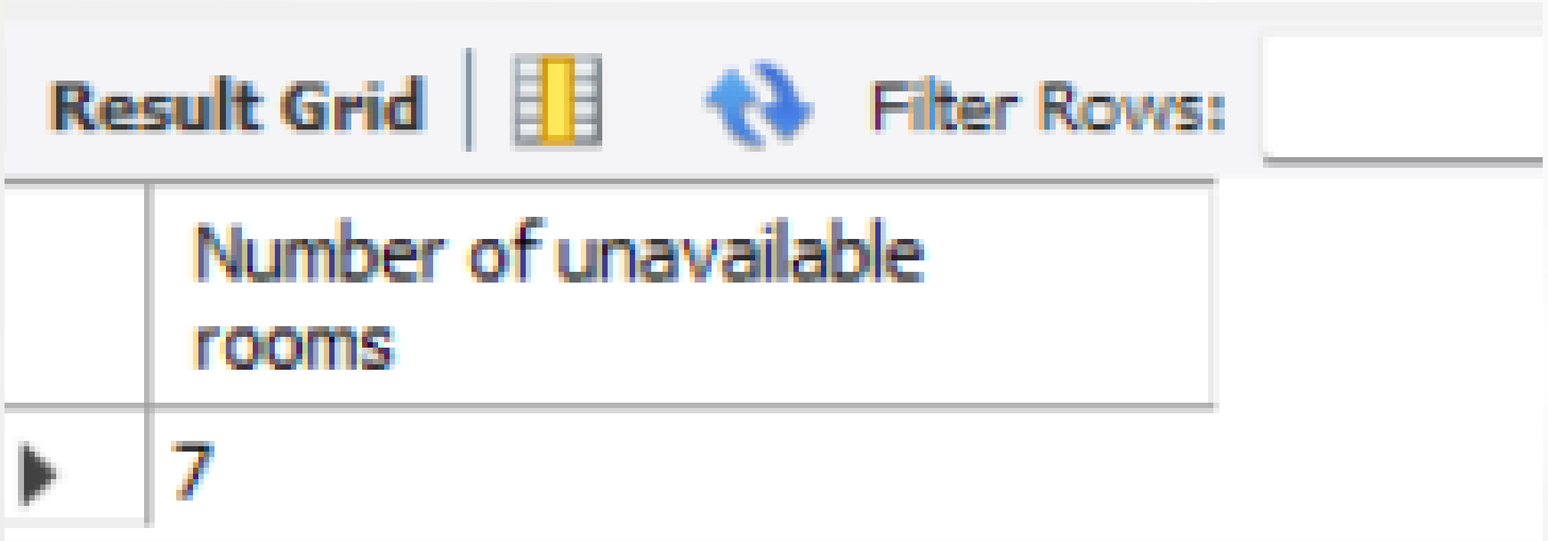
QUESTIONS

3. Write a SQL query to count the number of unavailable rooms. Return count as "Number of unavailable rooms".

```sql
-- Question-3

SELECT COUNT(*) AS "Number of unavailable rooms"
FROM room
WHERE unavailable = 't' or unavailable = 'TRUE';
```

OUTPUT



QUESTIONS

4.Write a SQL query to identify the physician and the department with which he or she is affiliated.

```sql
-- Question-4

SELECT p.name AS physician, d.name AS department
FROM physician p
JOIN affiliated_with a ON p.employeeid = a.physician
JOIN department d ON a.department = d.departmentid;
```

OUTPUT

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

physician	department
John Dorian	General Medicine
Eliot Red	General Medicine
Christopher Turk	General Medicine
Percival Cox	Christopher Turk
Bob Kelso	General Medicine
John Wen	General Medicine
Keith Dudemeister	General Medicine
Christopher Turk	Surgery
Todd Quinlan	Surgery
John Wen	Surgery
Molly Clock	Psychiatry

QUESTIONS

5.Write a SQL query to find those physicians who have received special training.

```sql
-- Question-5

SELECT DISTINCT p.employeeid, p.name
FROM physician p
JOIN trained_in t ON p.employeeid = t.physician;
```

OUTPUT

Result Grid | Filter Rows:

employeeid	name
3	Christopher Turk
7	John Wen
6	Todd Quinlan

QUESTIONS

6. Write a SQL query to identify the patients and the number of physicians with whom they have scheduled appointments.

```
90      -- Question-6
91
92  •   SELECT a.patient, COUNT(DISTINCT a.physician) AS physician_count
93      FROM appointment a
94      GROUP BY a.patient;
95
```

OUTPUT

Result Grid | Filter Rows:

patient	physician_count
100000001	2
100000002	1
100000003	1
100000004	2

QUESTIONS

7.Write a SQL query to count the number of unique patients who have been scheduled for examination room 'C'

```
96      -- Question-7
97
98      SELECT COUNT(DISTINCT patient) AS patient_count
99      FROM appointment
100     WHERE examinationroom = 'C';
101
```

OUTPUT

Result Grid | Filter Rows:

	patient_count
▶	3

QUESTIONS

8. Write a SQL query to count the number of available rooms for each floor in each block. Sort the result-set on floor ID, ID of the block.

```sql
-- Question-8

SELECT blockfloor, blockcode, COUNT(*) AS available_rooms
FROM room
WHERE unavailable = FALSE
GROUP BY blockfloor, blockcode
ORDER BY blockfloor, blockcode
LIMIT 0, 1000;
```

OUTPUT

Result Grid | Filter Rows:

blockfloor	blockcode	available_rooms
1	1	3
1	2	3
1	3	3
2	1	3
2	2	3
2	3	3
3	1	3
3	2	3
3	3	3
4	1	3
4	2	3
4	3	3

QUESTIONS

9. Create a view to display the name of the patients, their block, floor, and room number where they are admitted.

```
116     -- Question-9
117
118 •   CREATE VIEW patient_room_assignments AS
119     SELECT p.name AS patient, r.blockfloor, r.blockcode, r.roomnumber
120     FROM patient p
121     JOIN stay s ON p.ssn = s.patient
122     JOIN room r ON s.room = r.roomnumber;
123
124 •   SELECT * FROM patient_room_assignments;
```

OUTPUT

Result Grid | Filter Rows: | Export:

patient	blockfloor	blockcode	roomnumber
John Smith	1	2	111
Random J. Patient	1	3	123
Dennis Doe	1	2	112

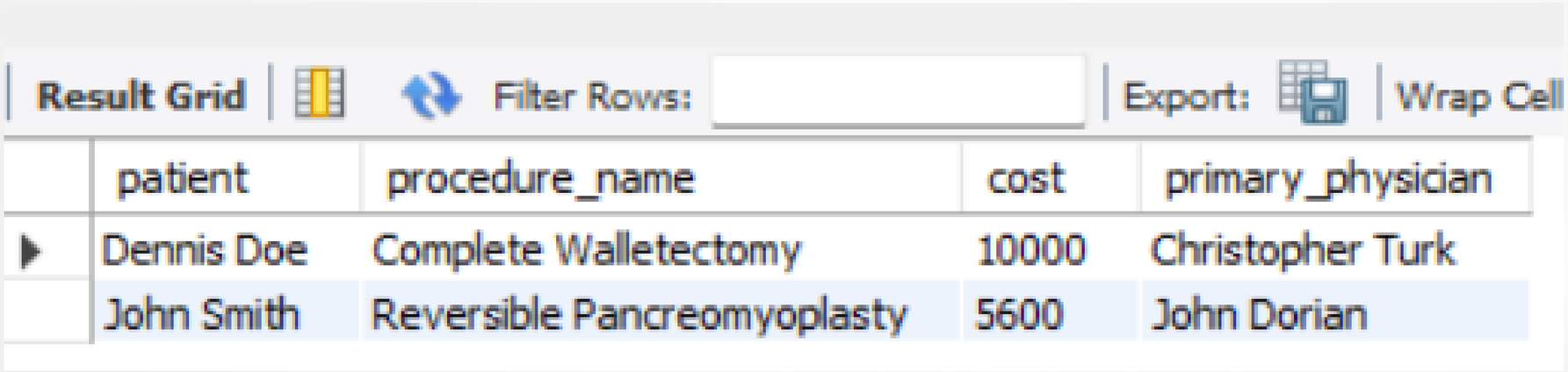
QUESTIONS

10.Write a SQL query to find those patients who have undergone a procedure costing more than $5,000, as well as the name of the physician who has provided primary care, should be identified.

```sql
-- Question-10

SELECT
    pa.name AS patient,
    pr.name AS procedure_name,
    pr.cost,
    ph.name AS primary_physician
FROM patient pa
JOIN undergoes u ON pa.ssn = u.patient
JOIN `procedure` pr ON u.procedure = pr.code
JOIN physician ph ON pa.pcp = ph.employeeid
WHERE pr.cost > 5000;
```

OUTPUT



patient	procedure_name	cost	primary_physician
Dennis Doe	Complete Walletectomy	10000	Christopher Turk
John Smith	Reversible Pancreomyoplasty	5600	John Dorian

QUESTIONS

11. Write a SQL query to identify those patients whose primary care is provided by a physician who is not the head of any department.

```sql
-- Question-11

SELECT pa.name AS patient, ph.name AS physician
FROM patient pa
JOIN physician ph ON pa.pcp = ph.employeeid
LEFT JOIN department d ON ph.employeeid = d.head
WHERE d.head IS NULL;
```

OUTPUT

Result Grid | Filter Rows:

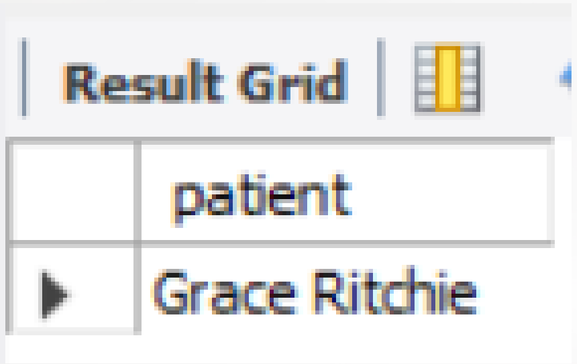
patient	physician
John Smith	John Dorian
Grace Ritchie	Eliot Red
Random J. Patient	Eliot Red
Dennis Doe	Christophe

QUESTIONS

12. Retrieve the names of patients who have been prescribed at least one medication by a physician from the Psychiatry department using a subquery.

```
151     -- Question-12
152
153     SELECT DISTINCT p.name AS patient
154     FROM patient p
155     JOIN prescribes pr ON p.ssn = pr.patient
156     JOIN physician ph ON pr.physician = ph.employeeid
157     JOIN affiliated_with a ON ph.employeeid = a.physician
158     JOIN department d ON a.department = d.departmentid
159     WHERE d.name = 'Psychiatry';
160
```

OUTPUT



QUESTIONS

13. Create a trigger that prevents inserting a new appointment if the physician does not have a primary affiliation with a department.

```
162    -- Question-13
163
164    DELIMITER //
165    CREATE TRIGGER validate_physician_affiliation
166    BEFORE INSERT ON appointment
167    FOR EACH ROW
168    BEGIN
169        IF NOT EXISTS (
170            SELECT 1 FROM affiliated_with
171            WHERE physician = NEW.physician AND primaryaffiliation = 't'
172        ) THEN
173            SIGNAL SQLSTATE '45000'
174            SET MESSAGE_TEXT = 'Physician must have primary department affiliation';
175        END IF;
176    END//
177    DELIMITER ;
178
179    describe appointment;
180    -- Try to check does delimiter work
181    INSERT INTO appointment (appointmentid, patient, prepnurse, physician, starttime, endtime, examinationroom)
182    VALUES (99999999, 100000001, 101, 3, '2023-01-01 10:00', '2023-01-01 11:00', 'A');
183
```

OUTPUT

Output

Action Output

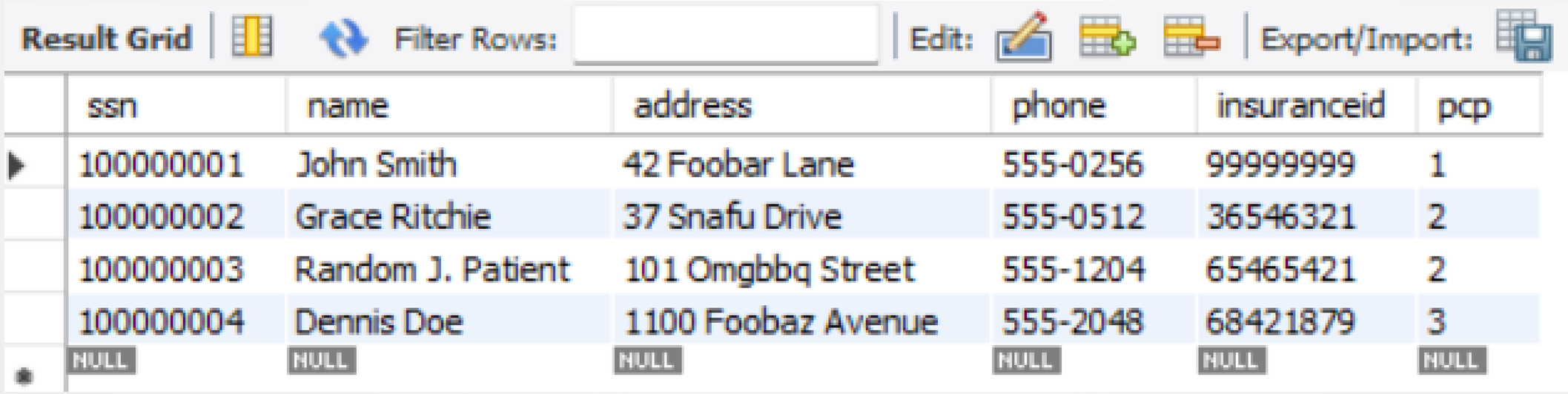
#	Time	Action	Message	Duration / Fetch
30	11:27:42	INSERT INTO appointment (appointmentid, patient, prepnurse, physician, startdatetime, enddatetime, examinat...	Error Code: 1054. Unknown column 'startdatetime' in 'field list'	0.031 sec
31	11:28:40	describe appointment	9 row(s) returned	0.016 sec / 0.000 sec
32	11:29:25	INSERT INTO appointment (appointmentid, patient, prepnurse, physician, starttime, endtime, examinationroom) ...	Error Code: 1644. Physician must have primary department affiliation	0.000 sec

QUESTIONS

14. Update the insurance ID of patients whose primary care physician (PCP) is 'John Dorian' to a new value '99999999'.

```
185     -- Question-14
186
187 •   UPDATE patient
188     SET insuranceid = '99999999'
189     WHERE pcp = (SELECT employeeid FROM physician WHERE name = 'John Dorian');
190
191 •   select * from patient;
```

OUTPUT



ssn	name	address	phone	insuranceid	pcp
100000001	John Smith	42 Foobar Lane	555-0256	99999999	1
100000002	Grace Ritchie	37 Snafu Drive	555-0512	36546321	2
100000003	Random J. Patient	101 Omgbbq Street	555-1204	65465421	2
100000004	Dennis Doe	1100 Foobaz Avenue	555-2048	68421879	3
NULL	NULL	NULL	NULL	NULL	NULL

QUESTIONS

15. Retrieve each physician's name along with the number of appointments they have and show the ranking of each physician based on the number of appointments in descending order.

```
193     -- Question-15
194
195     SELECT p.name AS physician,
196            COUNT(a.appointmentid) AS appointment_count,
197            RANK() OVER (ORDER BY COUNT(a.appointmentid) DESC)AS ranking
198     FROM physician p
199     LEFT JOIN appointment a ON p.employeeid = a.physician
200     GROUP BY p.name
201     ORDER BY appointment_count DESC;
```

OUTPUT

Result Grid | Filter Rows:

physician	appointment_count	ranking
Eliot Red	3	1
John Dorian	2	2
Percival Cox	2	2
Christopher Turk	1	4
Molly Clock	1	4
Bob Kelso	0	6
Todd Quinlan	0	6
John Wen	0	6
Keith Dudemeister	0	6

ADDITIONAL QUESTIONS

16.Which patients have never been prescribed any medication?

```
203     -- Question-16
204 •   SELECT examinationroom, COUNT(*) AS appointment_count
205     FROM appointment
206     GROUP BY examinationroom
207     ORDER BY appointment_count DESC;
208
```

OUTPUT

Result Grid | Filter Rows:

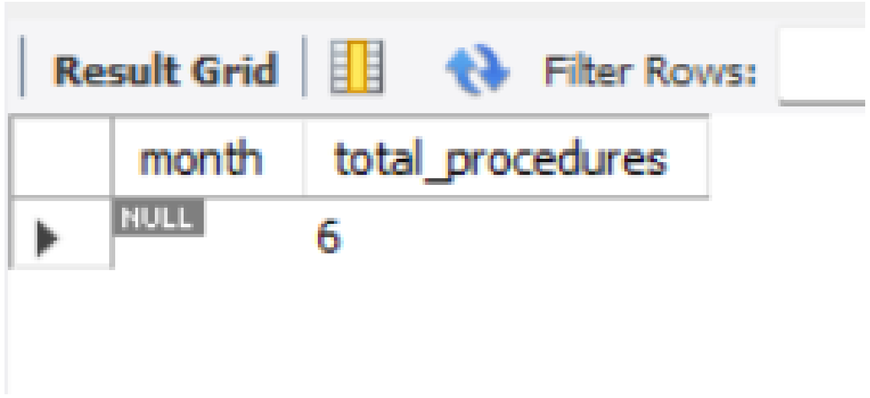
examinationroom	appointment_count
A	3
B	3
C	3

ADDITIONAL QUESTIONS

17. How many procedures were performed each month?

```
210     -- Question-17
211
212 •   SELECT
213         MONTH(date) AS month,
214         COUNT(*) AS total_procedures
215     FROM undergoes
216     GROUP BY MONTH(date)
217     ORDER BY month;
```

OUTPUT



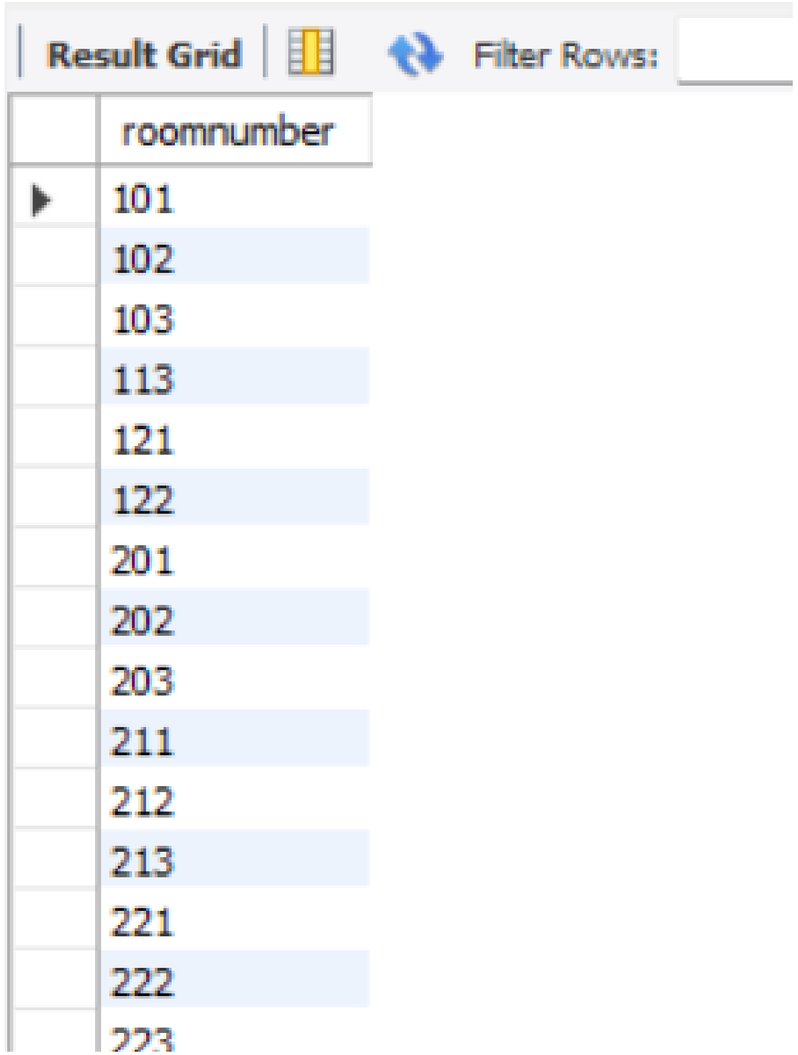
ADDITIONAL QUESTIONS

18. Which rooms have never been used for a patient stay?

CODE

```
219     -- Question-18
220
221 •   SELECT r.roomnumber
222     FROM room r
223     LEFT JOIN stay s ON r.roomnumber = s.room
224     WHERE s.room IS NULL;
```

OUTPUT



roomnumber
101
102
103
113
121
122
201
202
203
211
212
213
221
222
223

TEAM MEMBERS

Somya Aggarwal
E23BCAU0192
(Team Leader)

Rushay Gopani
E23CSEU1283

Navya Gupta
E23CSEU2083

Pranav Gupta
E23CSEU0301

Vishant Randhawa
E23CSEU1162

THANK YOU