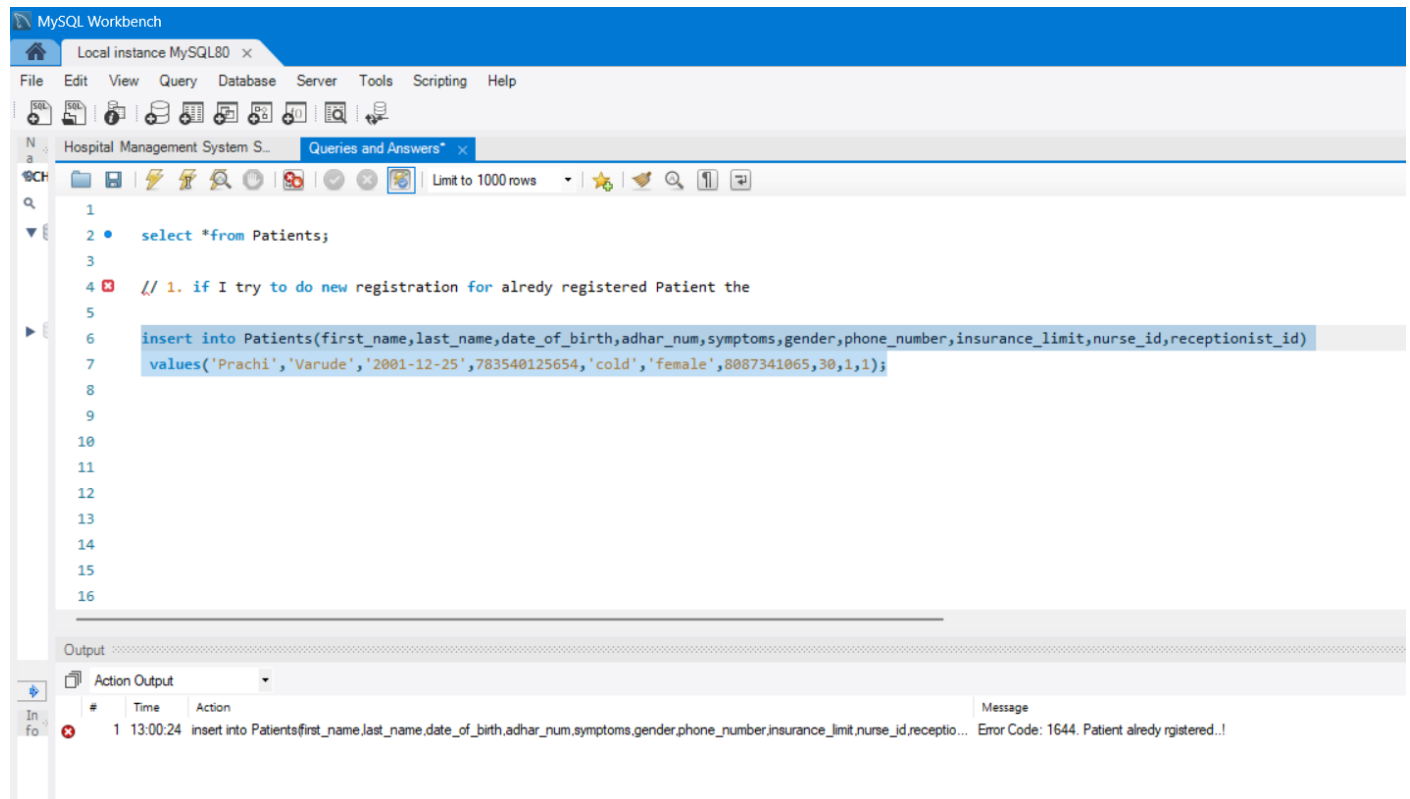


### //// Questions and Answers of queries

select \*from Patients;

// 1. if I try to do new registration for alredy registered Patient the

```
insert into Patients(first_name,last_name,date_of_birth,adhar_num,symptoms,gender,phone_number,insurance_limit)
values('Prachi','Varude','2001-12-25',783540125654,'cold','female',8087341065,30);
```



## // 2. Adding data for roles

insert into Roles(role\_name) values ('Doctor');

insert into Roles(role\_name) values ('Nurse');

insert into Roles(role\_name) values ('Receptionist');

select \*from Roles;

The screenshot shows the MySQL Workbench interface. The main editor window contains the following SQL queries:

```
3
4 // 1. if I try to do new registration for already registered Patient the
5
6 insert into Patients(first_name,last_name,date_of_birth,adhar_num,symptoms,gender,phone_number,insurance_limit) values('Prachi','Varude','2001-12-25',783540125654,'col
7
8 // 2. Adding data for roles
9
10 insert into Roles(role_name) values ('Doctor');
11 insert into Roles(role_name) values ('Nurse');
12 insert into Roles(role_name) values ('Receptionist');
13
```

The Results window shows the output of the queries. The first query returns 3 rows of data for the Roles table:

role_id	role_name
1	Doctor
2	Nurse
3	Receptionist

The Output window shows the execution details of the queries:

#	Time	Action	Message	Duration
1	19:09:19	select *from Roles LIMIT 0, 1000	3 row(s) returned	0.016 s

### // 3. Adding diagnosis data for patient

```
insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-01-10','Regular checkup','painkiller',5,5);
```

```
insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-02-20','Regular checkup','dolo',6,6);
```

```
select *from Diagnosis;
```

The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The main window displays a SQL script with the following queries:

```
13
14 • select *from Roles;
15
16 // 3. Adding diagnosis data for patient
17
18 insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-01-10','Regular checkup','painkiller',5,5);
19 • insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-02-20','Regular checkup','dolo',6,6);
20
21
22 • select *from Diagnosis;
```

Below the script, the 'Result Grid' shows the results of the queries. The first query returned 2 rows. The second query inserted 2 rows into the 'Diagnosis' table. The third query returned 2 rows.

diagnosis_id	patient_id	appointment_id	diagnosis_date	diagnosis_details	prescription
1	5	5	2024-01-10	Regular checkup	painkiller
2	6	6	2024-02-20	Regular checkup	dolo

The 'Output' pane at the bottom shows the execution log with the following messages:

#	Time	Action	Message
70	19:46:46	select *from Appointments LIMIT 0, 1000	2 row(s) returned
71	19:47:10	insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-01-10','Regular checku...	1 row(s) affected
72	19:47:10	insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2024-02-20','Regular checku...	1 row(s) affected
73	19:47:28	insert into Bill(bill_amount,bill_date,patient_id) values(500,'2024-01-10',5)	1 row(s) affected
74	19:47:28	insert into Bill(bill_amount,bill_date,patient_id) values(500,'2024-02-20',5)	1 row(s) affected
75	19:47:28	insert into Bill(bill_amount,bill_date,patient_id) values(100,'2024-01-10',6)	1 row(s) affected
76	19:47:28	insert into Bill(bill_amount,bill_date,patient_id) values(100,'2024-02-20',6)	1 row(s) affected

#### // 4.Fetching details of particular patient

```
select *from Patients
where insurance_limit>30;
```

The screenshot displays the MySQL Workbench interface. The query editor at the top contains the following SQL query:

```
// 4.Fetching details of particular patient
select *from Patients
where insurance_limit>30;
```

Below the query editor, the 'Result Grid' shows the output of the query. It contains two rows of patient data:

patient_id	first_name	last_name	date_of_birth	adhar_num	symptoms	gender	phone_number	insurance_limit	total_bill	receptionist_id	nurse_id
5	Ramesh	Janawade	2002-07-01	583547762003	cough	male	7276774873	40	0.00	1	1
7	Kiran	Sabale	1981-09-05	102030405060	headache	female	9730083826	60	0.00	2	1

At the bottom of the interface, the 'Output' tab is active, showing the execution log:

#	Time	Action	Message
1	19:51:12	select *from Patients where insurance_limit>30 LIMIT 0, 1000	2 row(s) returned

## // 5. Generating bill for Patient using stored procedure

call Generate\_bill(5);

select total\_bill from Patients

where patient\_id=5;

The screenshot shows the MySQL Workbench interface. The main window displays a SQL query in the 'Queries and Answers' tab. The query is as follows:

```
28
29 // 5. Generating bill for Patient
30
31 call Generate_bill(5);
32
33 select total_bill from Patients
34 where patient_id=5;
35
36
37
```

Below the query editor, the 'Result Grid' is visible, showing the results of the query:

total_bill
1000.00

At the bottom of the interface, the 'Output' tab is active, showing a log of actions performed:

#	Time	Action
✓ 1	19:51:12	select *from Patients where insurance_limit>30 LIMIT 0, 1000
✓ 2	19:52:45	select *from Bill LIMIT 0, 1000
✓ 3	19:54:10	call Generate_bill(5)
✓ 4	19:55:36	select total_bill from Patients where patient_id=5 LIMIT 0, 1000

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Hospital Management System S... Queries and Answers\* x

Limit to 1000 rows

```
28
29 // 5. Generating bill for Patient
30
31 select *from Bill;
32
33
34
35
36
37
--
```

Result Grid

bill_id	patient_id	bill_amount	bill_date
2	5	500.00	2024-01-10
3	5	500.00	2024-02-20
4	6	100.00	2024-01-10
5	6	100.00	2024-02-20
NULL	NULL	NULL	NULL

Diagnosis4 Bill 6 x

Output

Action Output

#	Time	Action
1	19:51:12	select *from Patients where insurance_limit>30 LIMIT 0, 1000
2	19:52:45	select *from Bill LIMIT 0, 1000

## // 6. Fetching details of the Patients which are treated by Nurses havin name 'Susmita'

select \*from Patients

join Nurses

on Patients.nurse\_id=Nurses.nurse\_id and nurse\_name='Susmita';

The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for file operations, query execution, and database management. The main window displays a query in the 'Queries and Answers' tab:

```
// 6. Fetching details of the Patients which are treated by Nurses havin name 'Susmita'  
  
select *from Patients  
join Nurses  
on Patients.nurse_id=Nurses.nurse_id and nurse_name='Susmita';
```

Below the query editor, the 'Result Grid' shows the results of the query. The grid has 16 columns: patient\_id, first\_name, last\_name, date\_of\_birth, adhar\_num, symptoms, gender, phone\_number, insurance\_limit, total\_bill, receptionist\_id, nurse\_id, nurse\_id, nurse\_name, nurse\_address, and phone\_num. The results are as follows:

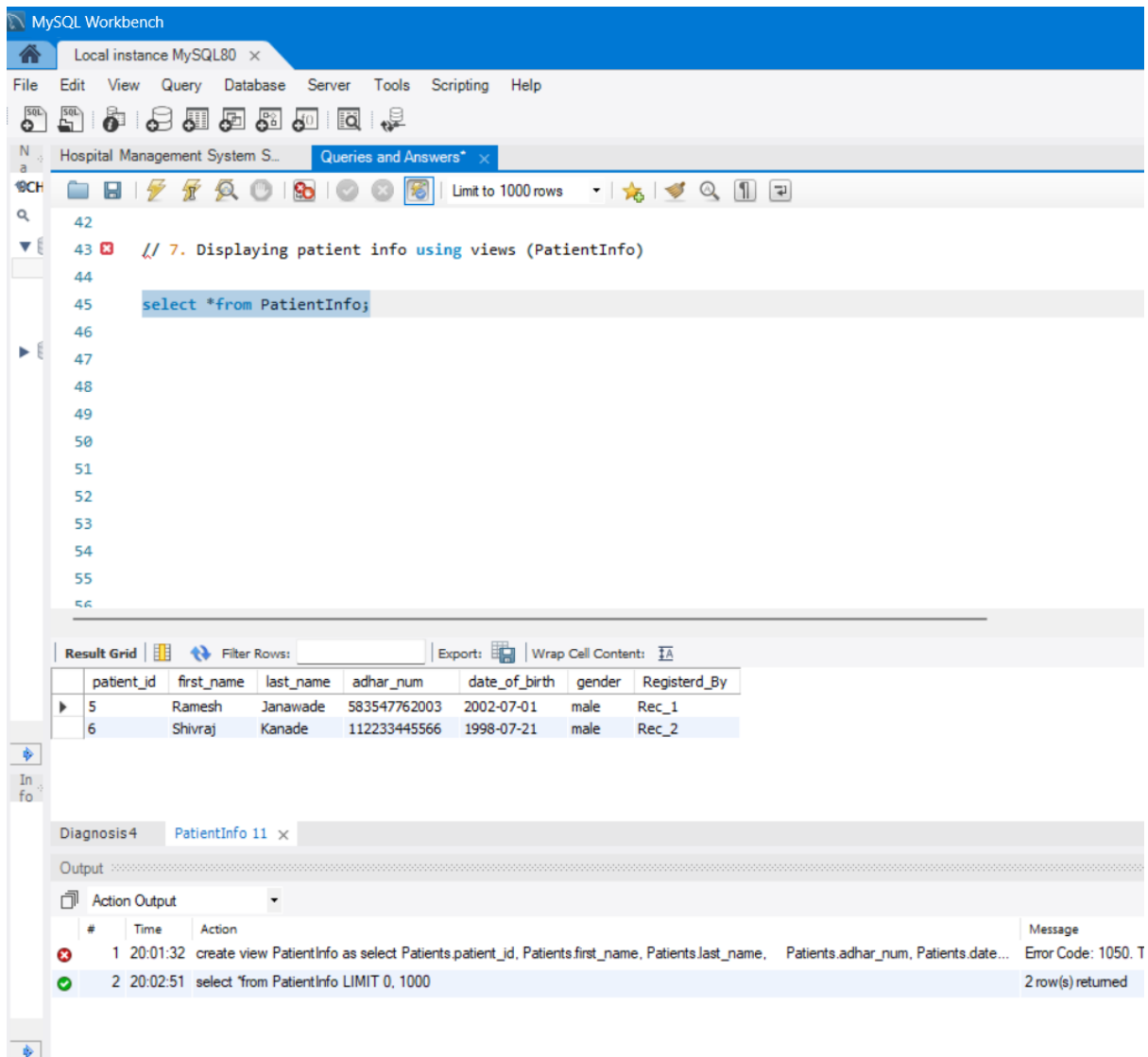
patient_id	first_name	last_name	date_of_birth	adhar_num	symptoms	gender	phone_number	insurance_limit	total_bill	receptionist_id	nurse_id	nurse_id	nurse_name	nurse_address	phone_num
5	Ramesh	Janawade	2002-07-01	583547762003	cough	male	7276774873	40	1000.00	1	1	1	Susmita	Gadhinglaj	1203005640
7	Kiran	Sabale	1981-09-05	102030405060	headache	female	9730083826	60	0.00	2	1	1	Susmita	Gadhinglaj	1203005640

At the bottom, the 'Output' tab shows the execution log. It lists 7 actions with their timestamps and messages:

#	Time	Action	Message
1	19:51:12	select *from Patients where insurance_limit>30 LIMIT 0, 1000	2 row(s) returned
2	19:52:45	select *from Bill LIMIT 0, 1000	4 row(s) returned
3	19:54:10	call Generate_bill(5)	1 row(s) affected
4	19:55:36	select total_bill from Patients where patient_id=5 LIMIT 0, 1000	1 row(s) returned
5	19:58:13	select *from Patients join Nurses on Patients.nurse_id=Nurses.nurse_id LIMIT 0, 1000	3 row(s) returned
6	19:58:41	select *from Patients join Nurses on Patients.nurse_id=Nurses.nurse_id LIMIT 0, 1000	3 row(s) returned
7	19:59:22	select *from Patients join Nurses on Patients.nurse_id=Nurses.nurse_id and nurse_name='Susmita' LIMIT 0, 1000	2 row(s) returned

## // 7. Displaying patient info using views (PatientInfo)

select \*from PatientInfo;



The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The main window displays a query in the 'Queries and Answers' tab. The query is:

```
// 7. Displaying patient info using views (PatientInfo)

select *from PatientInfo;
```

Below the query editor, the 'Result Grid' shows two rows of data:

patient_id	first_name	last_name	adhar_num	date_of_birth	gender	Registered_By
5	Ramesh	Janawade	583547762003	2002-07-01	male	Rec_1
6	Shivraj	Kanade	112233445566	1998-07-21	male	Rec_2

At the bottom, the 'Output' tab shows the execution log:

#	Time	Action	Message
1	20:01:32	create view PatientInfo as select Patients.patient_id, Patients.first_name, Patients.last_name, Patients.adhar_num, Patients.date...	Error Code: 1050. T
2	20:02:51	select *from PatientInfo LIMIT 0, 1000	2 row(s) returned



## // 8. Index of Patients and Bill table

show index from Patients;

The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The main window displays a query in the 'Queries and Answers' tab:

```
46
47 // 8. Index of Patients and Bill table
48
49 show index from Patients;
50
51
```

Below the query editor, the 'Result Grid' shows the output of the query. The table has 15 columns: Table, Non\_unique, Key\_name, Seq\_in\_index, Column\_name, Collation, Cardinality, Sub\_part, Packed, Null, Index\_type, Comment, Index\_comment, Visible, and Expression. The results are as follows:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
patients	0	PRIMARY	1	patient_id	A	0				BTREE			YES	
patients	0	adhar_num	1	adhar_num	A	0			YES	BTREE			YES	
patients	1	nurse_id	1	nurse_id	A	0				BTREE			YES	
patients	1	receptionist_id	1	receptionist_id	A	0				BTREE			YES	

At the bottom, the 'Output' tab shows the execution log. The first action failed with an error: 'Error Code: 1050. Table 'PatientInfo' already exists'. The second and third actions succeeded, returning 2 and 4 rows respectively.

#	Time	Action	Message
1	20:01:32	create view PatientInfo as select Patients.patient_id, Patients.first_name, Patients.last_name, Patients.adhar_num, Patients.date...	Error Code: 1050. Table 'PatientInfo' already exists
2	20:02:51	select *from PatientInfo LIMIT 0, 1000	2 row(s) returned
3	20:04:41	show index from Patients	4 row(s) returned

// 9. checking whether insurance is expired while inserting into Diagnosis table

// if I try to add new patient data as

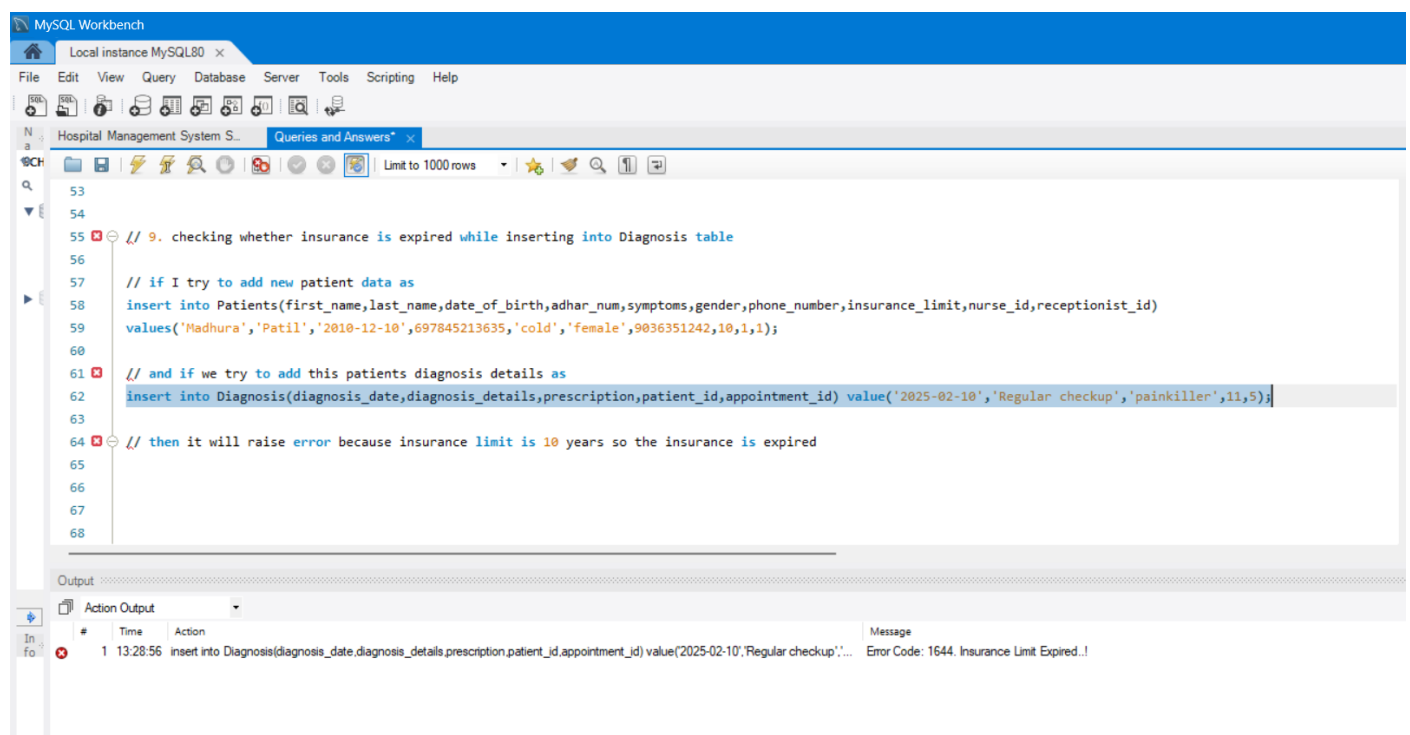
insert into

```
Patients(first_name,last_name,date_of_birth,adhar_num,symptoms,gender,phone_number,insurance_limit,nurse_id,receptionist_id) values('Madhura','Patil','2010-12-10','697845213635','cold','female','9036351242',10,1,1);
```

// and if we try to add this patients diagnosis details as

```
insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2025-02-10','Regular checkup','painkiller',9,5);
```

// then it will raise error because insurance limit is 10 years so the insurance is expired



The screenshot shows the MySQL Workbench interface. The main editor displays a SQL script with the following content:

```
53
54
55 // 9. checking whether insurance is expired while inserting into Diagnosis table
56
57 // if I try to add new patient data as
58 insert into Patients(first_name,last_name,date_of_birth,adhar_num,symptoms,gender,phone_number,insurance_limit,nurse_id,receptionist_id)
59 values('Madhura','Patil','2010-12-10','697845213635','cold','female','9036351242',10,1,1);
60
61 // and if we try to add this patients diagnosis details as
62 insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2025-02-10','Regular checkup','painkiller',11,5);
63
64 // then it will raise error because insurance limit is 10 years so the insurance is expired
65
66
67
68
```

The 'Output' panel at the bottom shows the 'Action Output' tab with the following message:

#	Time	Action	Message
1	13:28:56	insert into Diagnosis(diagnosis_date,diagnosis_details,prescription,patient_id,appointment_id) value('2025-02-10','Regular checkup',...	Error Code: 1644. Insurance Limit Expired..!