

## Practice Project 7.12: Patient Diagnosis Report

### Course 4: SQL Training

#### Task:

1. Write a query to create a patients table with the fields such as date, patient id, patient name, age, weight, gender, location, phone number, disease, doctor name, and doctor id.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' selected. The main editor window shows the SQL File 5\* with the following SQL code:

```
1 • Create Table Patients_table(  
2     DATE date,  
3     P_id int,  
4     P_name varchar(40),  
5     P_age int,  
6     P_weight int,  
7     Gender char(1),  
8     Location varchar(30),  
9     Ph_no int,  
10    Disease varchar(30),  
11    Doc_name varchar(30),  
12    Doc_id int)  
13
```

The bottom pane shows the 'Output' tab with the following log:

#	Time	Action	
2	13:05:12	Use Patient_diagnosis_report	0
3	13:05:26	Use Patient_diagnosis_report	0
4	13:07:17	Create Table Patients_table( DATE date, P_id int,...	0

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' selected, and 'patients\_table' is highlighted under the 'Tables' folder. The main editor window shows the SQL File 5\* with the following SQL code:

```
1 • SELECT * FROM patient_diagnosis_report.patients_table;
```

The bottom pane shows the 'Result Grid' tab with the following columns:

DATE	P_id	P_name	P_age	P_weight	Gender	Location	Ph_no	Disease	Doc_name	Doc_id
------	------	--------	-------	----------	--------	----------	-------	---------	----------	--------

## 2. Write a query to **insert** values into the **Patients table**.

The screenshot shows the SQL Studio interface. The left pane displays the 'patient\_diagnosis\_report' schema with a table named 'patients\_table'. The main editor shows the query: `SELECT * FROM patient_diagnosis_report.patients_table;`. The 'Result Grid' at the bottom displays the following data:

	date	pid	p_name	age	weight	gender	location	phone_no	disease	doctor_name	doctor_id
▶	15-06-2019	AP2021	Sarath	67	76	Male	chennai	5462829	Cardiac	Mohan	21
	13-02-2019	AP2022	John	62	80	Male	banglore	1234731	Cancer	Suraj	22
	08-01-2018	AP2023	Henry	43	65	Male	Kerala	9028320	Liver	Mehta	23
	04-02-2020	AP2024	Carl	56	72	Female	Mumbai	9293829	Asthma	Karthik	24
	15-09-2017	AP2025	Shikar	55	71	Male	Delhi	7821281	Cardiac	Mohan	21
	22-07-2018	AP2026	Piyush	47	59	Male	Haryana	8912819	Cancer	Suraj	22
	25-03-2017	AP2027	Stephen	69	55	Male	Gujarat	8888211	Liver	Mehta	23
	22-04-2019	AP2028	Aaron	75	53	Male	Banglore	9012192	Asthma	Karthik	24

## 3. Write a query to display the **total number of patients** in the table.

The screenshot shows the SQL Studio interface. The left pane displays the 'patient\_diagnosis\_report' schema with a table named 'patients\_table'. The main editor shows the query: `Select count(pid) as Total_no_of_patients from Patients_table`. The 'Result Grid' at the bottom displays the following data:

	Total_no_of_patients
▶	8

4. Write a query to display the patient id, patient name, gender, and disease of the patient whose **age is maximum**.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded, showing 'patients\_table'. The main editor window contains the following SQL query:

```
1 • Select Pid, P_name, Gender, Disease,  
2     MAX(age) as max_age from Patients_table  
3  
4
```

The 'Result Grid' at the bottom shows the following data:

Pid	P_name	Gender	Disease	max_age
AP2021	Sarath	Male	Cardiac	75

5. Write a query to display patient id and patient name with the **current date**.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded, showing 'patients\_table'. The main editor window contains the following SQL query:

```
1 • Select Pid, P_name, now() as current_date_time  
2     from Patients_table;  
3
```

The 'Result Grid' at the bottom shows the following data:

Pid	P_name	current_date_time
AP2021	Sarath	2022-03-07 15:20:33
AP2022	John	2022-03-07 15:20:33
AP2023	Henry	2022-03-07 15:20:33
AP2024	Carl	2022-03-07 15:20:33
AP2025	Shikar	2022-03-07 15:20:33
AP2026	Piyush	2022-03-07 15:20:33
AP2027	Stephen	2022-03-07 15:20:33
AP2028	Aaron	2022-03-07 15:20:33

6. Write a query to display the **old patient's name** and **new patient's name in uppercase**.

The screenshot shows the SQL Enterprise Manager interface. The left pane displays the 'SCHEMAS' tree with 'patients\_datasets' selected. The right pane shows a query window with the following SQL query:

```
1 • SELECT date, ucase(p_name) from patients_datasets where date= "25-03-2017" or date= "04-02-2020";
```

Below the query window, the 'Result Grid' displays the results of the query:

date	ucase(p_name)
04-02-2020	CARL
25-03-2017	STEPHEN

7. Write a query to display the patient's name along with the **length of their name**.

The screenshot shows the SQL Enterprise Manager interface. The left pane displays the 'SCHEMAS' tree with 'patients\_table' selected. The right pane shows a query window with the following SQL query:

```
1 • SELECT P_name "Patient's Name",  
2     LENGTH(P_name) "Length"  
3     from Patients_table
```

Below the query window, the 'Result Grid' displays the results of the query:

Patient's Name	Length
SARATH	6
JOHN	4
HENRY	5
CARL	4
SHIKAR	6
PIYSUH	6
STEPHEN	7
AARON	5

8. Write a query to display the patient's name, and the **gender** of the patient must be mentioned as **M or F**.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded. The main pane shows a query in 'SQL File 5':

```
1 • Select P_name,  
2 SUBSTRING(Gender,1,1)  
3 from Patients_table
```

The 'Result Grid' at the bottom displays the following data:

P_name	SUBSTRING(Gender,1,1)
SARATH	M
JOHN	M
HENRY	M
CARL	F
SHIKAR	M
PIYSUH	M
STEPHEN	M
AARON	M

9. Write a query to **combine the names of the patient** and the doctor in a new column.

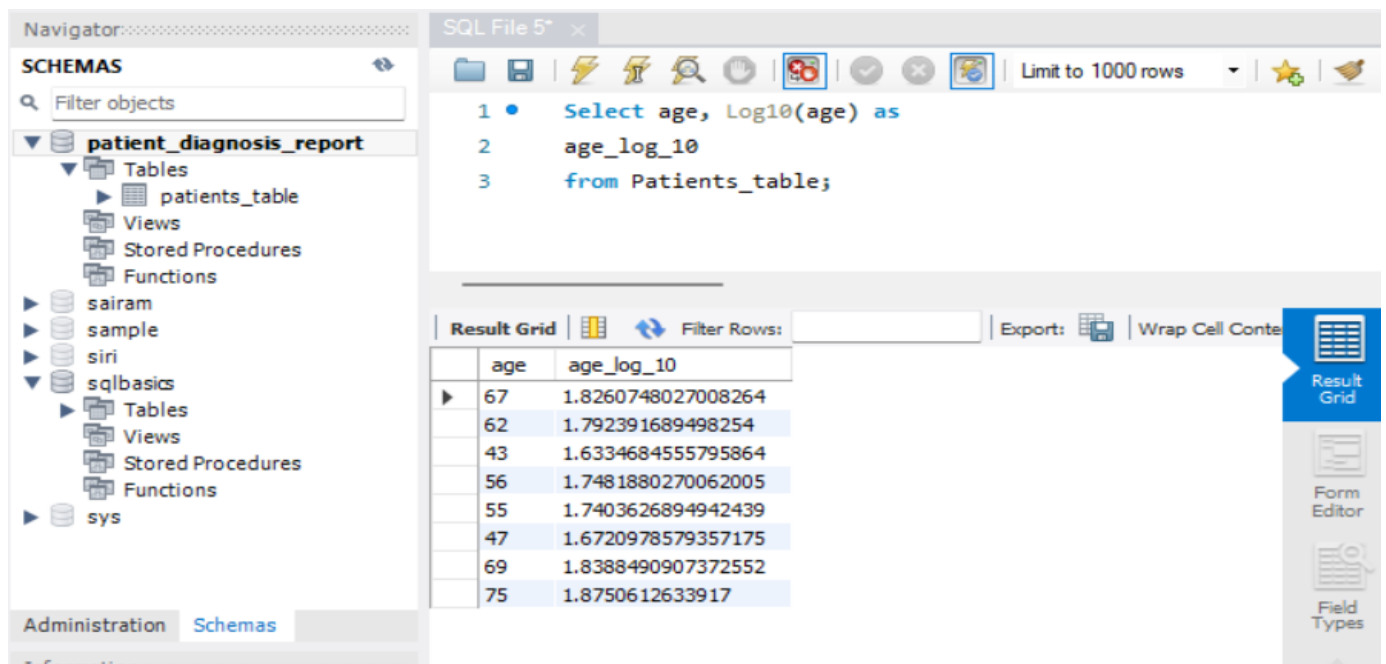
The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded. The main pane shows a query in 'SQL File 5':

```
1 • Select concat(P_name, "      ", doctor_name)  
2 as Patient_ref_Doctor  
3 from Patients_table
```

The 'Result Grid' at the bottom displays the following data:

Patient_ref_Doctor
SARATH Mohan
JOHN Suraj
HENRY Mehta
CARL Karthik
SHIKAR Mohan
PIYSUH Suraj
STEPHEN Mehta
AARON Karthik

10. Write a query to display the patients' age along with the **logarithmic value** (base 10) of **their age**.



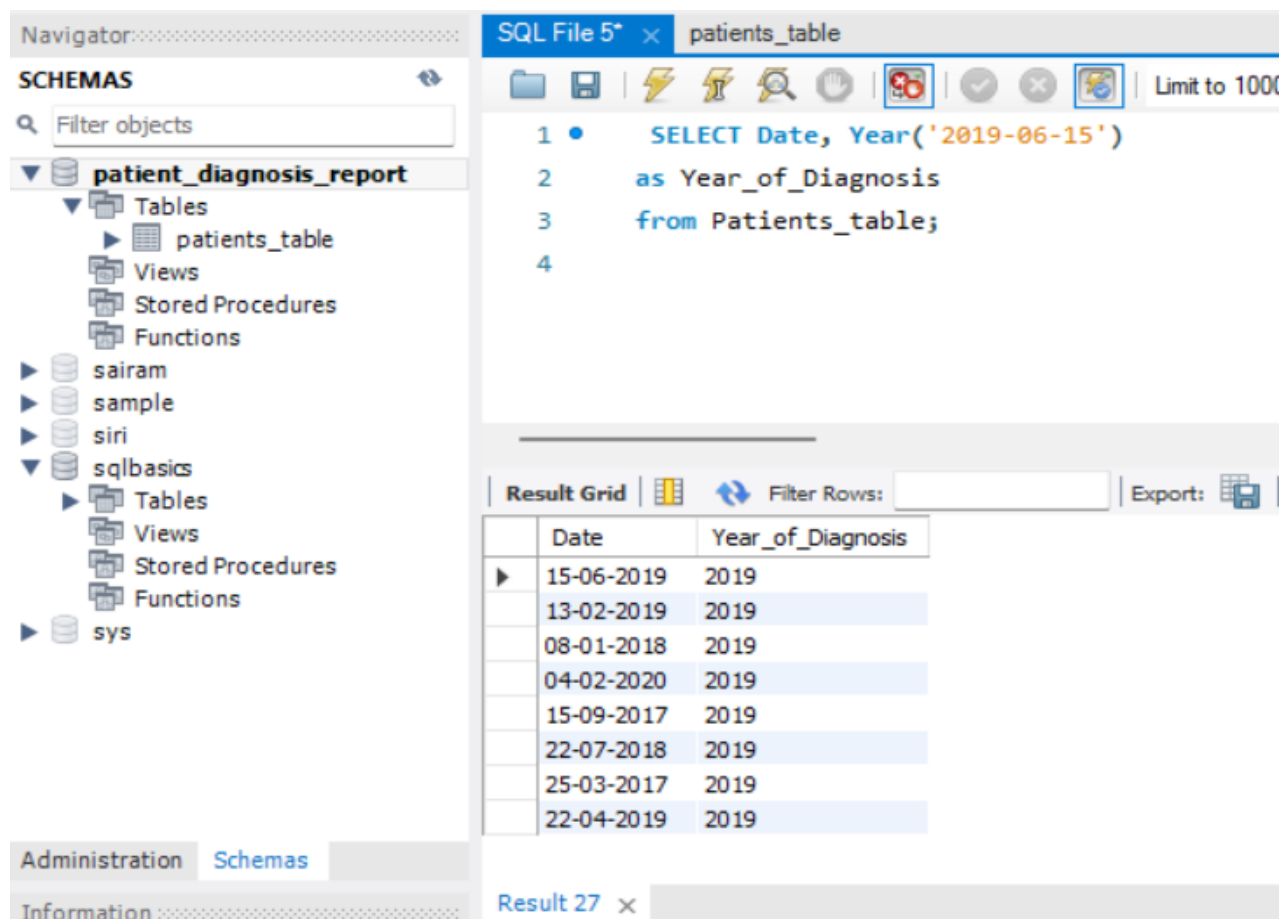
The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane is expanded to show the 'patient\_diagnosis\_report' database, with the 'patients\_table' selected. The main pane displays the following SQL query:

```
1 • Select age, Log10(age) as
2   age_log_10
3   from Patients_table;
```

Below the query, the 'Result Grid' shows the output of the query. The grid has two columns: 'age' and 'age\_log\_10'. The results are as follows:

age	age_log_10
67	1.8260748027008264
62	1.792391689498254
43	1.6334684555795864
56	1.7481880270062005
55	1.7403626894942439
47	1.6720978579357175
69	1.8388490907372552
75	1.8750612633917

11. Write a query to **extract the year** from the given date in a separate column.



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane is expanded to show the 'patient\_diagnosis\_report' database, with the 'patients\_table' selected. The main pane displays the following SQL query:

```
1 • SELECT Date, Year('2019-06-15')
2   as Year_of_Diagnosis
3   from Patients_table;
4
```

Below the query, the 'Result Grid' shows the output of the query. The grid has two columns: 'Date' and 'Year\_of\_Diagnosis'. The results are as follows:

Date	Year_of_Diagnosis
15-06-2019	2019
13-02-2019	2019
08-01-2018	2019
04-02-2020	2019
15-09-2017	2019
22-07-2018	2019
25-03-2017	2019
22-04-2019	2019



12. Write a query to return **NULL** if the **patient's name and doctor's name are similar** else return the **patient's name**.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded, showing 'patients\_datasets' and 'patients\_table'. The main editor shows a query: `select p_name from patients_table where p_name!=doctor_name`. The 'Result Grid' at the bottom shows the following data:

p_name
SARATH
JOHN
HENRY
CARL
SHIKAR
PIYSUH
STEPHEN
AARON

13. Write a query to return **Yes** if the **patient's age is greater than 40** else return **No**.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'patient\_diagnosis\_report' expanded, showing 'patients\_table'. The main editor shows a query: `SELECT age, case when age >40 then 'YES' else 'No' End from patients_table;`. The 'Result Grid' at the bottom shows the following data:

age	case when age >40 then 'YES' else 'No' End
67	YES
62	YES
43	YES
56	YES
55	YES
47	YES
69	YES
75	YES

14. Write a query to display the **doctor's duplicate name** from the table.

The screenshot shows a SQL IDE interface. On the left, the 'SCHEMAS' pane displays a tree view with 'patient\_diagnosis\_report' expanded, showing 'patients\_table' under 'Tables'. The main editor window, titled 'SQL File 5\* x patients\_table', contains the following SQL query:

```
1 • SELECT doctor_name, count(doctor_name)
2   from patients_table
3  Group by doctor_name
4  Having count(doctor_name )>1
```

Below the query editor, the 'Result Grid' tab is active, displaying the query results in a table:

doctor_name	count(doctor_name)
Mohan	2
Suraj	2
Mehta	2
Karthik	2