page2image568

SQL Training

Lesson-End Project Solution



**Patient Diagnosis Report**

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

**SQL code:**

CREATE TABLE lep\_6.patients (

date DATE NOT NULL,

pid varchar(45) NOT NULL,

p\_name varchar(45) NOT NULL,

age INT NOT NULL,

weight INT NOT NULL,

gender varchar(45) NOT NULL,

location varchar(45) NOT NULL,

phone\_no INT NOT NULL,

disease varchar(45) NOT NULL,

doctor\_name varchar(45) NOT NULL,

doctor\_id INT NOT NULL,

PRIMARY KEY(pid));

1. Write a query to insert values into the **patients** table

**SQL code:**

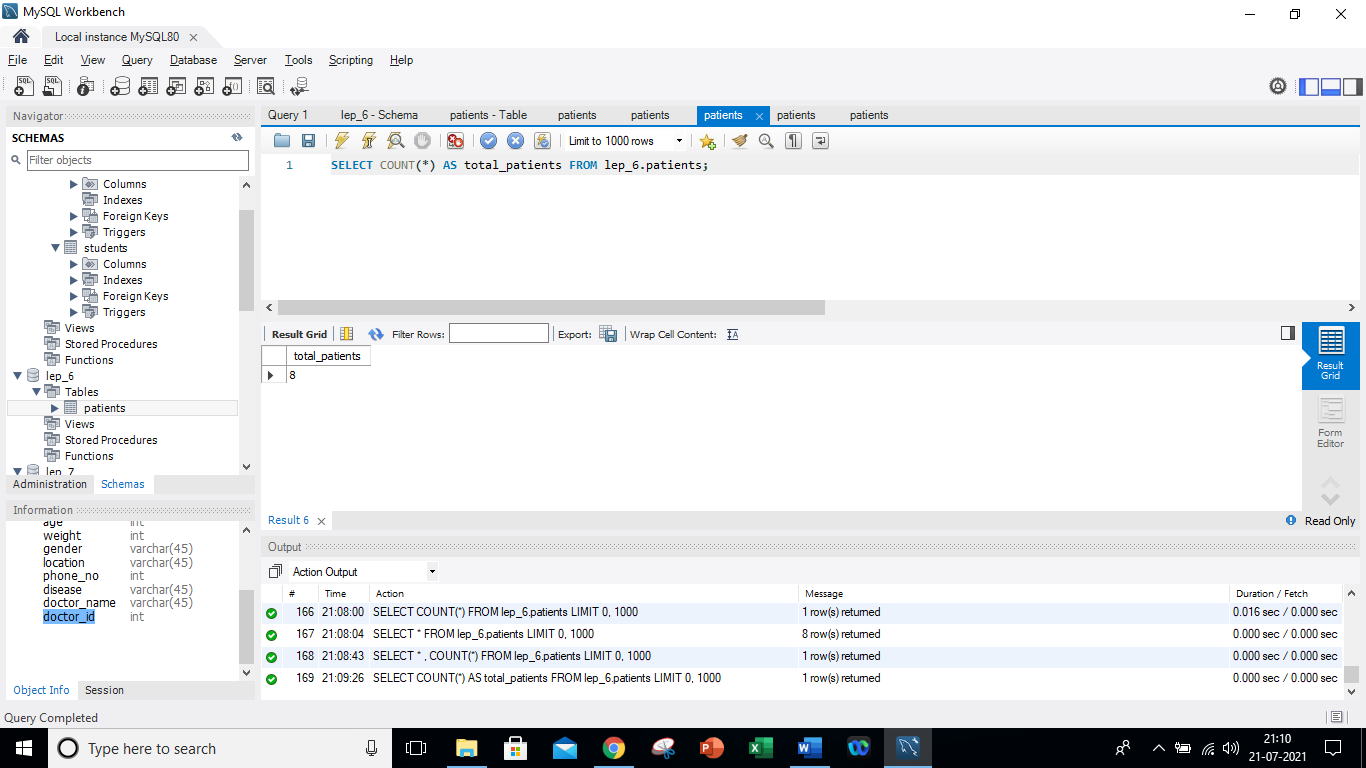
INSERT INTO lep\_6. patients (date,pid,p\_name,age,weight,gender,location,phone\_no,disease,doctor\_name,doctor\_id) VALUES ('2019-06-15','AP2021','Sarath','67','76','Male','chennai','5462829','Cardiac','Mohan','21');

1. Write a query to display the total number of patients in the table

**SQL code:**

SELECT COUNT(\*) AS total\_patients FROM lep\_6.patients;

**Output:**



1. Write a query to display the patient ID, patient name, gender, and disease of the oldest (age) patient

**SQL code:**

SELECT pid,p\_name,gender,disease, MAX(AGE) AS MAX\_AGE FROM lep\_6.patients;

**Output:**



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

**SQL code:**

SELECT pid,p\_name ,NOW() as CurrentDate FROM lep\_6.patients;

**Output:**

Graphical user interface, text, application

Description automatically generated

1. Write a query to display the old patient name and new patient name in uppercase

**SQL code:**

SELECT doctor\_name,UCASE(doctor\_name) AS UpperCase\_D\_name FROM lep\_6.patients;

**Output:**

Table

Description automatically generated

1. Write a query to display the patients' names along with the total number of characters in their name

**SQL code:**

SELECT p\_name,length(p\_name) AS lengthofp\_name FROM lep\_6.patients;

**Output:**

Graphical user interface, table

Description automatically generated

1. Write a query to display the gender of the patient as M or F along with the patient's name

**SQL code:**

SELECT p\_name,MID(gender,1,1) AS GENDER FROM lep\_6.patients;

**Output:**

Table

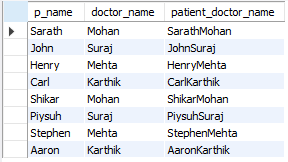
Description automatically generated

1. Write a query to combine the patient's name and doctor's name in a new column

**SQL code:**

SELECT p\_name,doctor\_name,CONCAT(p\_name,doctor\_name) AS patient\_doctor\_name FROM lep\_6.patients;

**Output:**



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

**SQL code:**

SELECT age,LOG10(age) as LOG\_AGE FROM lep\_6.patients;

**Output:**

Graphical user interface, text, application

Description automatically generated

1. Write a query to extract the year for a given date and place it in a separate column

**SQL code:**

SELECT \*,YEAR(date) AS Year FROM lep\_6.patients;

**Output:**

Table

Description automatically generated

1. Write a query to check the patient’s name and doctor’s name are similar and display **NULL**, else return the patient’s name

**SQL code:**

SELECT NULLIF(p\_name,doctor\_name) FROM lep\_6.patients;

**Output:**

Table

Description automatically generated

1. Write a query to check if a patient’s age is greater than 40 and display **Yes** if it is and **No** if it isn't

**SQL code:**

SELECT age,IF(age>40,'Yes','No') AS Agegreater40 FROM lep\_6.patients;

**Output:**

Table

Description automatically generated

1. Write a query to display duplicate entries in the doctor name column

**SQL code:**

SELECT doctor\_name,COUNT(\*) occurences FROM lep\_6.patients GROUP BY doctor\_name HAVING COUNT(\*)>1;

**Output:**

Table

Description automatically generated