```sql
-- Patient Diagnosis Report Project - Sonal Ranpise
-- DESCRIPTION:
-- The data analyst of a hospital wants to store the patient diagnosis reports with the details of the doctors and the patients for good medical practice and continuity of care.
-- Objective:
-- The database design helps to retrieve, update, and modify the patient’s details to keep track of the patient's health care routine.
-- Task to be performed:

-- Task 01 :
-- 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.
CREATE DATABASE Patient_Diagnosis_Report;
USE Patient_Diagnosis_Report;
CREATE TABLE Patient_Diagnosis_Report.patients_table
(
date DATE NOT NULL,
pid VARCHAR (100) PRIMARY KEY NOT NULL,
p_name VARCHAR (100) NOT NULL,
age INT NOT NULL,
weight INT NOT NULL,
gender VARCHAR (100) NOT NULL,
location VARCHAR (100) NOT NULL,
phone_no INT NOT NULL,
disease VARCHAR (100) NOT NULL,
doctor_name VARCHAR (100) NOT NULL,
doctor_id INT NOT NULL
);
DESCRIBE Patient_Diagnosis_Report.patients_table;

-- Task 02 :
-- Write a query to insert values into the patients table.
INSERT INTO  Patient_Diagnosis_Report.patients_table (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"),
("2019-02-13", "AP2022", "John", "62", "80",  "Male",  "banglore",  "1234731",  "Cancer",  "Suraj",  "22"),
("2018-08-01", "AP2023", "Henry", "43", "65",  "Male",  "Kerala",  "9028320",  "Liver",  "Mehta",  "23"),
("2020-04-02", "AP2024", "Carl", "56", "72",  "Female",  "Mumbai",  "9293829",  "Asthma",  "Karthik",  "24"),
("2017-09-15", "AP2025", "Shikar", "55", "71",  "Male",  "Delhi",  "7821281",  "Cardiac",  "Mohan",  "21"),
("2018-07-22", "AP2026", "Piysuh", "47", "59",  "Male",  "Haryana",  "8912819",  "Cancer",  "Suraj",  "22"),
("2017-03-25", "AP2027", "Stephen", "69", "55",  "Male",  "Gujarat",  "8888211",  "Liver",  "Mehta",  "23"),
("2019-04-22", "AP2028", "Aaron", "75", "53",  "Male",  "Banglore",  "9012192",  "Asthma",  "Karthik",  "24");
SELECT*FROM Patient_Diagnosis_Report.patients_table;

-- Task 03 :
-- Write a query to display the total number of patients in the table.
SELECT COUNT(PID) AS total_number_of_patients FROM Patient_Diagnosis_Report.patients_table;
-- OR
SELECT COUNT(*)  AS total_number_of_patients FROM Patient_Diagnosis_Report.patients_table;

-- Task 04 :
-- Write a query to display the patient id, patient name, gender, and disease of the patient whose age is maximum.
SELECT pid, p_name, gender, disease, MAX(AGE) AS maximum_age FROM Patient_Diagnosis_Report.patients_table;

-- Task 05 :
```

```sql
-- Write a query to display patient id and patient name with the current date.
SELECT pid, p_name, NOW() AS currentdate FROM Patient_Diagnosis_Report.patients_table;

-- Task 06
-- Write a query to display the old patient’s name and new patient's name in uppercase.
SELECT p_name, UCASE(p_name) AS p_name_uppercase  FROM Patient_Diagnosis_Report.
patients_table;

-- Task 07
-- Write a query to display the patient’s name along with the length of their name.
SELECT p_name, lENGTH(p_name) AS p_name_length  FROM Patient_Diagnosis_Report.
patients_table;

-- Task 08
-- Write a query to display the patient’s name, and the gender of the patient must be
mentioned as M or F.
SELECT p_name, MID(gender,1,1) AS gender FROM Patient_Diagnosis_Report.patients_table;

-- Task 09
-- Write a query to combine the names of the patient and the doctor in a new column.
SELECT concat(p_name," ", doctor_name) AS patient_doctor_combine_name FROM
Patient_Diagnosis_Report.patients_table;

-- Task 10
-- Write a query to display the patients’ age along with the logarithmic value (base 10)
of their age.
SELECT age, log10(age) AS logarithmic_value_age FROM Patient_Diagnosis_Report.
patients_table;

-- Task 11
-- Write a query to extract the year from the given date in a separate column.
SELECT*,  YEAR(DATE) AS year FROM Patient_Diagnosis_Report.patients_table;
-- OR
SELECT YEAR(DATE) AS year FROM Patient_Diagnosis_Report.patients_table;

-- TASK 12
-- Write a query to return NULL if the patient’s name and doctor’s name are similar else
return the patient’s name.
SELECT IFNULL(p_name,doctor_name) FROM Patient_Diagnosis_Report.patients_table;

-- TASK 13
-- Write a query to return Yes if the patient’s age is greater than 40 else return No.
SELECT AGE, IF(AGE > 40 ,"YES", "NO") AS age_greater_than_40 FROM
Patient_Diagnosis_Report.patients_table;

-- TASK 14
-- Write a query to display the doctor’s duplicate name from the table.
SELECT doctor_name, COUNT(*) AS doctor_name_duplicate_name FROM Patient_Diagnosis_Report.
patients_table GROUP BY doctor_name  HAVING COUNT(*) > 1;

```