

Hospital Management System

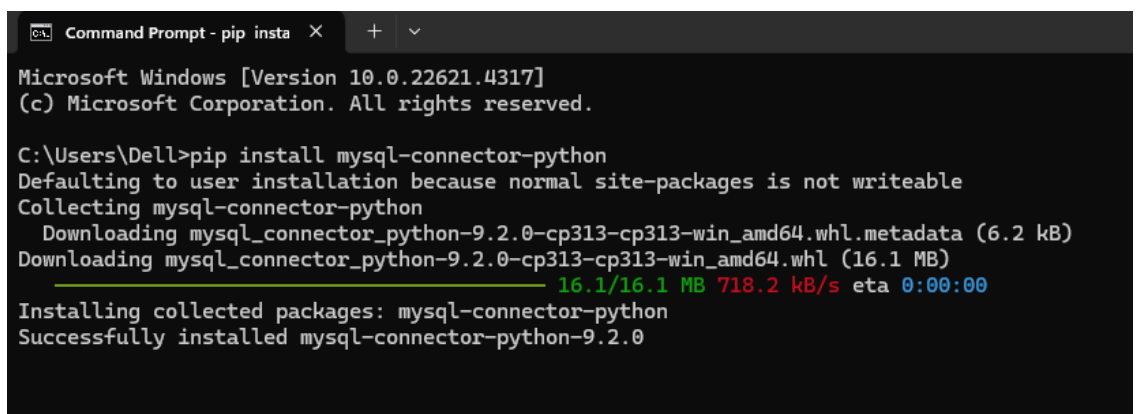
Project Description:

Developed a Hospital Management System using Python (Tkinter) for the frontend and MySQL for the backend. The system allows hospitals to manage patient records, including adding, updating, deleting, and viewing patient details. It ensures efficient record-keeping and retrieval of patient information.

Technologies Used: Python (Tkinter), MySQL, SQL, Database Management

Steps Include:

- Open the Python IDE and type the code for the hospital management system.
- Before this ,for connecting the python and MySQL we need to install the library **mysql.connector**
- Open the command prompt,type the command
pip install mysql-connector-python
- After the successful installation , it shows



```
Command Prompt - pip insta X + v
Microsoft Windows [Version 10.0.22621.4317]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell>pip install mysql-connector-python
Defaulting to user installation because normal site-packages is not writeable
Collecting mysql-connector-python
  Downloading mysql_connector_python-9.2.0-cp313-cp313-win_amd64.whl.metadata (6.2 kB)
  Downloading mysql_connector_python-9.2.0-cp313-cp313-win_amd64.whl (16.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 16.1/16.1 MB 718.2 kB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.2.0
```

- Then import the mysql.connector library in the code.
- And then import **tkinter** library for Graphical User Interface(GUI).

- After importing these library ,type the code the

```

projectHMS.py - C:\Users\Deell\projectHMS.py (3.11.4)
File Edit Format Run Options Window Help

import mysql.connector
from tkinter import *
from tkinter import messagebox, ttk

# Database Connection
conn = mysql.connector.connect(
    host='localhost',
    user='root',
    password='Mukund200@',
    database='hospital_management'
)
cursor = conn.cursor()

# Create Table if not exists
cursor.execute('''CREATE TABLE IF NOT EXISTS patients (
    id INT AUTO INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    age INT,
    gender VARCHAR(10),
    contact VARCHAR(20),
    disease VARCHAR(100)
)''')
conn.commit()

# GUI Setup
root = Tk()
root.title("Hospital Management System")
root.geometry("700x500")
root.configure(bg="#f0f0f0")

frame = Frame(root, padx=20, pady=20, bg="#ffffff")
frame.pack(pady=20)

Label(frame, text="Patient Name", font=("Arial", 12)).grid(row=0, column=0, pady=5, sticky=W)
name_entry = Entry(frame, width=30)
name_entry.grid(row=0, column=1, pady=5)

Label(frame, text="Age", font=("Arial", 12)).grid(row=1, column=0, pady=5, sticky=W)
age_entry = Entry(frame, width=30)
age_entry.grid(row=1, column=1, pady=5)

Label(frame, text="Gender", font=("Arial", 12)).grid(row=2, column=0, pady=5, sticky=W)
gender_var = StringVar()
gender_var.set("Male")
radio_male = Radiobutton(frame, text="Male", variable=gender_var)
radio_female = Radiobutton(frame, text="Female", variable=gender_var)
radio_male.grid(row=2, column=1, pady=5)
radio_female.grid(row=2, column=2, pady=5)

Label(frame, text="Contact", font=("Arial", 12)).grid(row=3, column=0, pady=5, sticky=W)
contact_entry = Entry(frame, width=30)
contact_entry.grid(row=3, column=1, pady=5)

Label(frame, text="Disease", font=("Arial", 12)).grid(row=4, column=0, pady=5, sticky=W)
disease_entry = Entry(frame, width=30)
disease_entry.grid(row=4, column=1, pady=5)

Add Patient View Patients

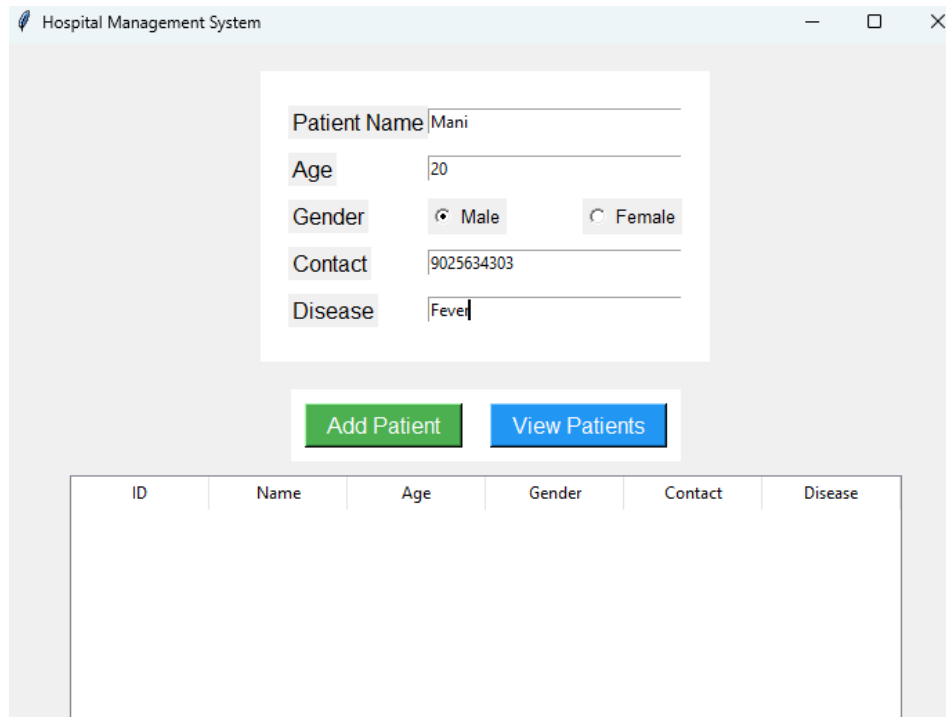
ID Name Age Gender Contact Disease

```

- After typing the code go to menu bar Click on **Run** option to run the code.
- After the successful compilation , it opens the GUI page

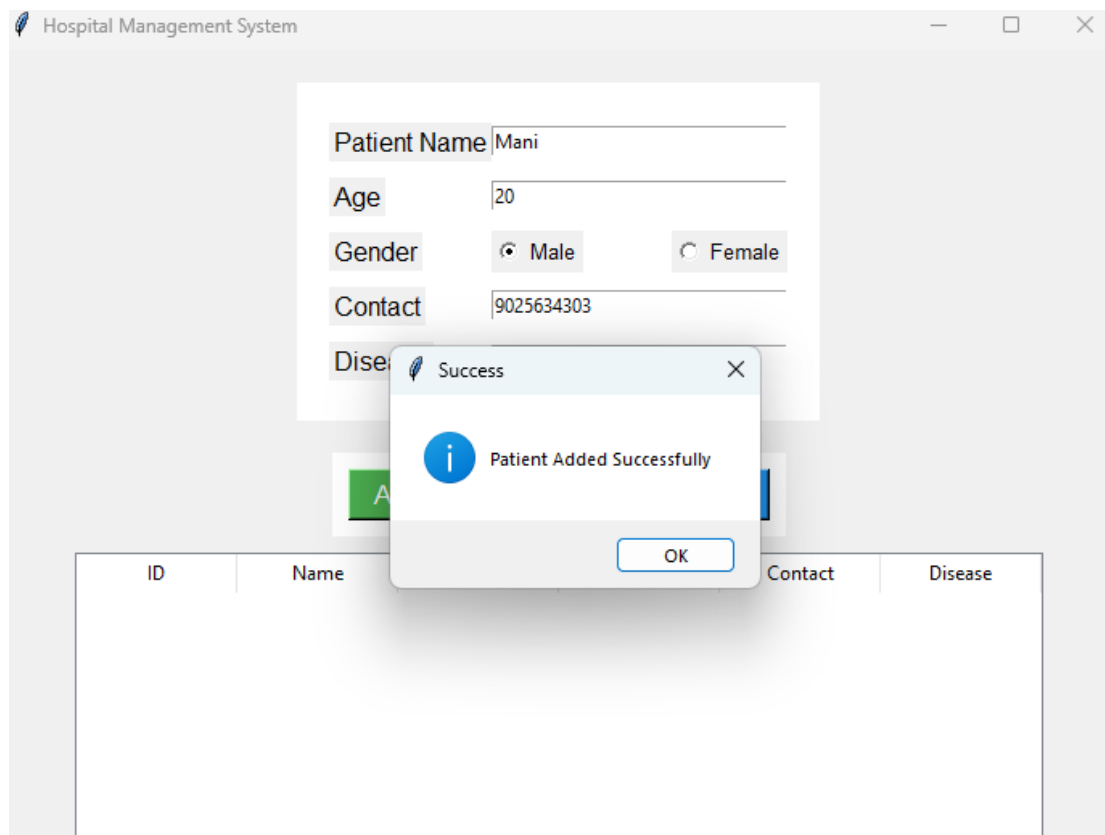
| ID | Name | Age | Gender | Contact | Disease |
|----|------|-----|--------|---------|---------|
|----|------|-----|--------|---------|---------|

- Fill the details and Click on **Add Patient** button



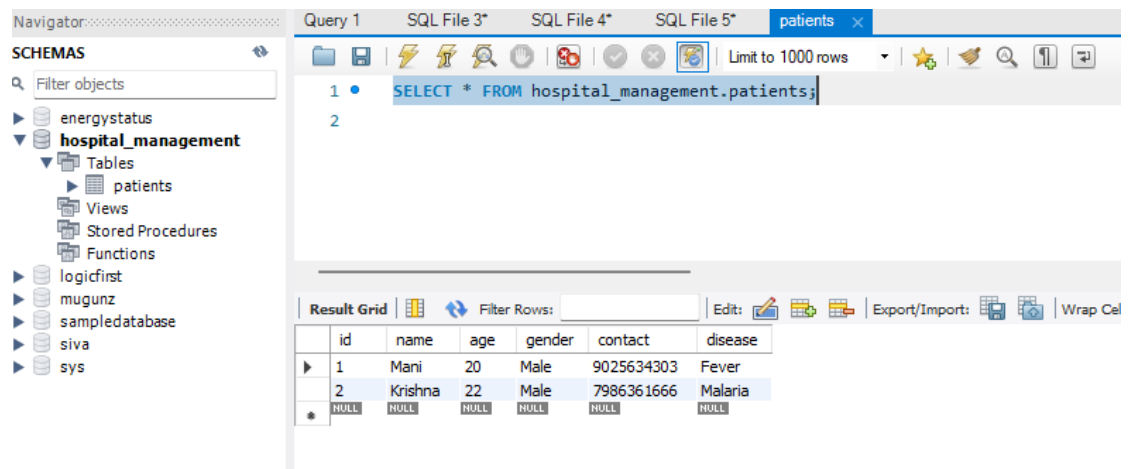
The screenshot shows a web application window titled "Hospital Management System". Inside, there is a form with the following fields: "Patient Name" (Mani), "Age" (20), "Gender" (radio buttons for Male and Female, with Male selected), "Contact" (9025634303), and "Disease" (Feve). Below the form are two buttons: "Add Patient" (green) and "View Patients" (blue). At the bottom of the window is a table with columns: ID, Name, Age, Gender, Contact, and Disease. The table is currently empty.

- Then it shows **Patient Added successfully**



This screenshot shows the same "Hospital Management System" window as before, but with a modal dialog box in the foreground. The dialog box is titled "Success" and contains an information icon (i) and the text "Patient Added Successfully". There is an "OK" button at the bottom right of the dialog. The background form and table are still visible but slightly dimmed.

- We can add many details like this
- These datas are to be stored in a **MySQL** database.



- Using MySQL database we can manage patient records, including adding, updating, deleting, and viewing patient details.