TRƯỜNG ĐẠI HỌC ĐẠI NAM

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**HỌC PHẦN: LẬP TRÌNH PYTHON**

**ĐỀ TÀI: QUẢN LÝ LỊCH KHÁM BỆNH**

**Giảng viên: ThS. Lê Tuấn Anh**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Họ tên** | **Mã SV** |
| **1** | **Đặng Thanh Bình** | **1871020078** |
| **2** | **Đào Tiến Dũng** | **1871020174** |

**Hà Nội, năm 2025**

TRƯỜNG ĐẠI HỌC ĐẠI NAM

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**HỌC PHẦN: LẬP TRÌNH PYTHON**

**ĐỀ TÀI: QUẢN LÝ LỊCH KHÁM BỆNH**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **Mã SV** | **Họ và Tên** | **Ngày Sinh** | **Điểm** | |
| **Bằng Số** | **Bằng Chữ** |
| **1** | **1871020078** | **Đặng Thanh Bình** | **18/08/2003** |  |  |
| **2** | **1871020174** | **Đào Tiến Dũng** | **22/03/2006** |  |  |

|  |  |
| --- | --- |
| **CÁN BỘ CHẤM THI 1**  **ThS. Lê Tuấn Anh** | **CÁN BỘ CHẤM THI 2** |

### 

**Hà Nội, năm 2025**

LỜI NÓI ĐẦU

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng lập trình vào quản lý dữ liệu đã trở thành xu hướng tất yếu trong nhiều lĩnh vực, đặc biệt là trong ngành y tế. Việc xây dựng các phần mềm hỗ trợ quản lý lịch khám bệnh không chỉ giúp tối ưu hóa công việc của nhân viên y tế mà còn nâng cao chất lượng phục vụ bệnh nhân.

Xuất phát từ nhu cầu thực tiễn đó, em đã lựa chọn đề tài **“Quản lý lịch khám bệnh”** để thực hiện bài tập lớn môn Lập trình Python. Mục tiêu của đề tài là xây dựng một chương trình ứng dụng bằng Python có khả năng quản lý thông tin bệnh nhân, bác sĩ, lịch khám và đơn thuốc một cách khoa học, hiệu quả và dễ sử dụng. Dự án được phát triển trên nền tảng dòng lệnh với cấu trúc dữ liệu linh hoạt, sử dụng các kỹ thuật thao tác tệp JSON và danh sách – từ điển, đồng thời áp dụng các khái niệm lập trình hướng thủ tục đã học.

Trong quá trình thực hiện đề tài, em đã có cơ hội củng cố và nâng cao kiến thức về lập trình Python, rèn luyện kỹ năng phân tích, thiết kế chương trình, cũng như tư duy tổ chức dữ liệu và xử lý thông tin. Mặc dù còn nhiều điểm hạn chế và đơn giản so với các hệ thống thực tế, nhưng đề tài là bước khởi đầu quan trọng để phát triển các ứng dụng quản lý chuyên nghiệp hơn trong tương lai.

Em xin chân thành cảm ơn thầy Lê Tuấn Anh và thầy Đào Duy Khương đã hướng dẫn, hỗ trợ em trong suốt quá trình học tập và thực hiện đề tài này.

MỤC LỤC

[LỜI NÓI ĐẦU 3](#_Toc201851364)

[MỤC LỤC 4](#_Toc201851365)

[CHƯƠNG 1. GIỚI THIỆU VỀ MỘT SỐ KHÁI NIỆM TRONG PYTHON 6](#_Toc201851366)

[1.1. Định nghĩa của Python 6](#_Toc201851367)

[1.1.1. Các ứng dụng thực tế của Python 6](#_Toc201851368)

[1.1.2. Ưu điểm và khuyết điểm của Python 6](#_Toc201851369)

[1.2. Những cú pháp Python cơ bản 8](#_Toc201851370)

[a) Biến và kiểu dữ liệu cơ bản 8](#_Toc201851371)

[b) Câu lệnh điều kiện 8](#_Toc201851372)

[c) Vòng lặp 8](#_Toc201851373)

[d) Hàm và cách sử dụng 9](#_Toc201851374)

[CHƯƠNG 2. GIỚI THIỆU, MÔ TẢ ĐỀ TÀI 10](#_Toc201851375)

[2.1. Giới thiệu đề tài 10](#_Toc201851376)

[2.1.1. Mục tiêu 10](#_Toc201851377)

[2.1.2. Lý do chọn đề tài 10](#_Toc201851378)

[2.2. Công cụ và phương pháp thực hiện 11](#_Toc201851379)

[2.2.1. Ngôn ngữ lập trình 11](#_Toc201851380)

[2.2.2. Phương pháp lưu trữ và xử lý dữ liệu 11](#_Toc201851381)

[CHƯƠNG 3. PHÁT TRIỂN VÀ TRIỂN KHAI ỨNG DỤNG 12](#_Toc201851382)

[3.1. Phát triển chức năng chính 12](#_Toc201851383)

[3.1.1. Quản lý bệnh nhân và bác sĩ 12](#_Toc201851384)

[3.1.2. Quản lý lịch khám 12](#_Toc201851385)

[3.2. Quản lý đơn thuốc 12](#_Toc201851386)

[3.2.1. Tạo và sửa đơn thuốc 12](#_Toc201851387)

[3.2.2. Thống kê và lưu dữ liệu 13](#_Toc201851388)

[3.3. Một số đoạn code trong menu. 13](#_Toc201851389)

[3.4. Đánh giá và hướng phát triển 16](#_Toc201851390)

[KẾT LUẬN 18](#_Toc201851391)

[DANH MỤC TÀI LIỆU THAM KHẢO 19](#_Toc201851392)

DANH MỤC HÌNH ẢNH

[Hình 3.1. 14](#_Toc201851416)

[Hình 3.2. 15](#_Toc201851417)

[Hình 3.3. 16](#_Toc201851418)

[Hình 3.4. 17](#_Toc201851419)

DANH MỤC BẢNG

[Bảng 1.1. So sánh giữa các ngôn ngữ 9](#_Toc201851451)

BẢNG CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **Ý NGHĨA** |
| 1 | AI | Artificial Intelligence |
| 2 | API | Application Programming Interface |
| 3 | CLI | Command Line Interface |
| 4 | IDE | Integrated Development Environment |
| 5 | GUI | Graphical User Interface |
| 6 | JSON | JavaScript Object Notation |

CHƯƠNG 1. GIỚI THIỆU VỀ MỘT SỐ KHÁI NIỆM TRONG PYTHON

# 1.1. Định nghĩa của Python

Python, như bạn thấy, là một ngôn ngữ sử dụng trình thông dịch, điều này mang lại cho nó sự linh hoạt và dễ sử dụng – đặc biệt trong quá trình phát triển và thử nghiệm. Giờ thì chúng ta đã hiểu được cách Python hoạt động dưới “nắp ca-pô,” hãy cùng tiếp tục khám phá sâu hơn về những gì Python có thể làm được trong phần tiếp theo.

Python là một ngôn ngữ lập trình cấp cao, được Guido van Rossum tạo ra vào cuối những năm 1980. Được đặt tên theo nhóm hài kịch Monty Python, ngôn ngữ này không chỉ dễ tiếp cận mà còn hài hước và thú vị như cái tên của nó. Python nổi bật với cú pháp đơn giản, dễ đọc, và một cộng đồng lập trình viên hỗ trợ mạnh mẽ.

Chính những yếu tố này đã giúp Python trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới.

## 1.1.1. Các ứng dụng thực tế của Python

Lập trình web với Django và Flask: Hãy tưởng tượng bạn muốn tạo một trang web cá nhân hay một ứng dụng web nào đó. Python đã có sẵn những “vũ khí bí mật” là Django và Flask, hai framework mạnh mẽ giúp bạn xây dựng các ứng dụng web từ nhỏ đến lớn, từ đơn giản đến phức tạp.

Phân tích dữ liệu và khoa học dữ liệu: Dữ liệu là “vàng” trong thế kỷ 21, và Python chính là “khoan vàng” giúp bạn tiếp cận nguồn tài nguyên vô tận này. Với các thư viện như Pandas, NumPy, bạn có thể phân tích, trực quan hóa và xử lý dữ liệu một cách dễ dàng.

Machine Learning và AI với Python: Nếu bạn mơ ước trở thành một nhà khoa học dữ liệu hoặc chuyên gia trí tuệ nhân tạo, thì Python chính là tấm vé vàng của bạn. Các thư viện như Scikit-learn và TensorFlow sẽ biến bạn từ một “tay mơ” thành “cao thủ” AI trong thời gian ngắn.

Automation và scripting với Python: Bạn có mệt mỏi với những công việc lặp đi lặp lại? Hãy để Python làm thay bạn! Với vài dòng code, Python có thể giúp bạn tự động hóa mọi thứ, từ quản lý file, gửi email đến kiểm tra hệ thống.

## 1.1.2. Ưu điểm và khuyết điểm của Python

**Ưu điểm:**

Dễ học và đọc hiểu: Cú pháp gần gũi với ngôn ngữ tự nhiên, giúp việc học lập trình dễ dàng hơn cho cả người mới bắt đầu.

Tính linh hoạt cao: Phù hợp với nhiều loại dự án, từ phát triển web đến khoa học dữ liệu và AI, nhờ vào hệ sinh thái thư viện phong phú.

Cộng đồng lớn mạnh: Một cộng đồng đông đảo và năng động, hỗ trợ tạo ra nhiều thư viện và công cụ giúp giảm thiểu thời gian phát triển.

**Khuyết điểm:**

Tốc độ thực thi chậm: Là ngôn ngữ thông dịch, Python chạy chậm hơn so với các ngôn ngữ biên dịch như C/C++.

Hạn chế trong lập trình di động: Python chưa phổ biến trong phát triển ứng dụng di động so với các ngôn ngữ như Java hay Swift.

Không tối ưu cho hệ thống đòi hỏi hiệu năng cao: Do tiêu thụ nhiều bộ nhớ và có thể không phù hợp cho các ứng dụng yêu cầu tối ưu hóa bộ nhớ và tốc độ cao.

Để có cái nhìn khách quan ta nên đặt Python cạnh những ngôn ngữ tương tự khác để so sánh như:

Bảng 1.1. So sánh giữa các ngôn ngữ

|  | **GO** | **Java** | **Python** | **C++** |
| --- | --- | --- | --- | --- |
| **Kiểu dữ liệu** | Tĩnh | Tĩnh | Động | Tĩnh |
| **Biên dịch** | Được biên dịch | Được biên dịch | Không cần biên dịch | Được biên dịch |
| **Đồng thời** | Hỗ trợ  goroutines | Hỗ trợ thông qua frameworks | Thông qua mô-đun của bên thứ 3 | Thủ công với threading |
| **Quản lý bộ nhớ** | Tự động | Tự động | Tự động | Thủ công |
| **Cú pháp** | Đơn giản | Phức tạp | Đơn giản | Phức tạp |
| **Ứng dụng** | Lập trình hệ thống máy chủ, Web | Ứng dụng doanh nghiệp, ứng dụng android | Phát triển web, scripting | Lập trình hệ thống, phát triển game |

# 1.2. Những cú pháp Python cơ bản

#### a) Biến và kiểu dữ liệu cơ bản

Python hỗ trợ nhiều kiểu dữ liệu như số nguyên (int), số thực (float), chuỗi (str), và boolean (bool). Biến trong Python không cần khai báo kiểu dữ liệu trước, giá trị của biến sẽ xác định kiểu dữ liệu.

Cú pháp:

x = 5  # Số nguyên

y = 3.14  # Số thực

z = "Hello"  # Chuỗi ký tự

#### b) Câu lệnh điều kiện

Python sử dụng các câu lệnh if, elif, và else để thực hiện các hành động dựa trên điều kiện. Đây là cách cơ bản để điều hướng luồng chương trình dựa trên các tình huống khác nhau.

Cú pháp:

if condition:

    # Khối mã thực thi nếu điều kiện đúng

elif another\_condition:

    # Khối mã thực thi nếu điều kiện khác đúng

else:

    # Khối mã thực thi nếu tất cả điều kiện trên đều sai

#### c) Vòng lặp

Python cung cấp hai loại vòng lặp chính: for và while. Vòng lặp for được dùng để lặp qua một chuỗi các phần tử, còn while tiếp tục lặp cho đến khi một điều kiện nhất định bị sai.

Cú pháp:

# For loop

for i in range(5):

    # Khối mã lặp lại 5 lần

# While loop

count = 0

while count < 5:

    # Khối mã lặp lại khi điều kiện đúng

    count += 1

#### d) Hàm và cách sử dụng

Hàm trong Python giúp đóng gói mã thành các khối có thể tái sử dụng, giúp mã gọn gàng và dễ bảo trì. Hàm có thể nhận tham số và trả về giá trị.

Cú pháp:

def function\_name(parameters):

    # Khối mã thực thi

    return value

# Gọi hàm

result = function\_name(arguments)

CHƯƠNG 2. GIỚI THIỆU, MÔ TẢ ĐỀ TÀI

# **2.1. Giới thiệu đề tài**

Hệ thống “Quản lý lịch khám bệnh” được xây dựng nhằm hỗ trợ các cơ sở y tế trong việc quản lý thông tin bệnh nhân, bác sĩ, lịch khám và đơn thuốc một cách có hệ thống và hiệu quả. Dự án được lập trình bằng ngôn ngữ Python, với mục tiêu cung cấp một công cụ đơn giản nhưng đầy đủ chức năng, giúp tối ưu quy trình làm việc tại các phòng khám vừa và nhỏ. Ứng dụng được triển khai trên môi trường dòng lệnh, sử dụng các cấu trúc dữ liệu cơ bản của Python để xử lý và lưu trữ thông tin.

## **2.1.1. Mục tiêu**

Cho phép tạo mới, chỉnh sửa, xóa, tìm kiếm và lọc thông tin bệnh nhân và bác sĩ một cách linh hoạt.

Quản lý lịch khám bệnh, bao gồm:

+ Tạo lịch khám mới theo mã bệnh nhân và mã bác sĩ.

+ Chỉnh sửa hoặc hủy lịch khi cần thiết.

+ Tra cứu lịch khám theo ngày, theo bệnh nhân hoặc theo bác sĩ.

Quản lý đơn thuốc:

+ Tạo đơn thuốc, thêm/xóa thuốc trong đơn.

+ Tính tổng chi phí của đơn thuốc.

+ Thống kê thuốc được kê nhiều nhất, bệnh nhân sử dụng nhiều thuốc nhất.

## 2.1.2. Lý do chọn đề tài

Gần gũi với thực tế trong lĩnh vực y tế, có thể phát triển thêm trong tương lai.

Giúp người học rèn luyện kỹ năng:

+ Quản lý dữ liệu bằng cấu trúc danh sách và từ điển trong Python.

+ Xử lý file JSON để lưu trữ và truy xuất thông tin hiệu quả.

+ Thiết kế menu điều hướng đa chức năng trong chương trình dòng lệnh.

+ Là nền tảng thực hành tốt cho những dự án phần mềm thực tế quy mô nhỏ.

# 2.2. Công cụ và phương pháp thực hiện

## 2.2.1. Ngôn ngữ lập trình

Dự án sử dụng ngôn ngữ Python 3.x vì:

+ Cú pháp đơn giản, dễ học, phù hợp với sinh viên.

+ Hỗ trợ tốt việc thao tác với dữ liệu dạng từ điển, danh sách và file JSON.

+ Có sẵn nhiều thư viện hỗ trợ như datetime, json, os...

2.2.2. Phương pháp lưu trữ và xử lý dữ liệu

Dữ liệu được lưu trữ dưới định dạng JSON với các tệp:

+ benhnhan.json: thông tin bệnh nhân.

+ bacsi.json: thông tin bác sĩ.

+ lichkham.json: danh sách lịch khám.

+ donthuoc.json: danh sách đơn thuốc.

Mỗi module có chức năng riêng:

+ Đọc file → nạp vào danh sách trong bộ nhớ.

+ Chỉnh sửa → cập nhật danh sách → ghi lại vào file JSON.

Cách này giúp dữ liệu được bảo toàn giữa các lần sử dụng, không bị mất khi thoát chương trình.

CHƯƠNG 3. PHÁT TRIỂN VÀ TRIỂN KHAI ỨNG DỤNG

# 3.1. Phát triển chức năng chính

## 3.1.1. Quản lý bệnh nhân và bác sĩ

Hệ thống cho phép:

+ Thêm mới, chỉnh sửa, xóa bệnh nhân/bác sĩ bằng cách nhập mã số định danh.

+ Hiển thị danh sách đầy đủ kèm thông tin như: họ tên, ngày sinh, giới tính, chuyên khoa (với bác sĩ).

Tính năng nâng cao:

+ Lọc danh sách theo giới tính, độ tuổi, hoặc chuyên khoa.

+ Tìm kiếm theo từ khóa tên hoặc mã để thao tác nhanh hơn.

+ Dữ liệu hiển thị dạng bảng từ điển hoặc bảng in dòng lệnh.

## 3.1.2. Quản lý lịch khám

Người dùng có thể:

+ Tạo lịch khám mới với thông tin: mã bác sĩ, mã bệnh nhân, ngày, giờ.

+ Chỉnh sửa hoặc xóa lịch đã tạo.

Tra cứu lịch khám:

+ Theo ngày cụ thể.

+ Theo bệnh nhân.

+ Theo bác sĩ → giúp tránh việc trùng lịch trong cùng khung giờ.

+ Dữ liệu lịch khám được tự động kiểm tra trùng lặp trước khi lưu.

# 3.2. Quản lý đơn thuốc

## 3.2.1. Tạo và sửa đơn thuốc

Đơn thuốc gắn với mã bệnh nhân và ngày kê.

Người dùng có thể:

+ Thêm các loại thuốc (tên, liều lượng, giá) vào đơn.

+ Xóa thuốc trong đơn nếu nhập sai.

+ Kiểm tra và tránh trùng tên thuốc khi kê.

+ Cấu trúc đơn thuốc: mỗi đơn là một từ điển gồm thông tin bệnh nhân, ngày kê và danh sách thuốc (dưới dạng list).

## 3.2.2. Thống kê và lưu dữ liệu

Các thống kê cơ bản:

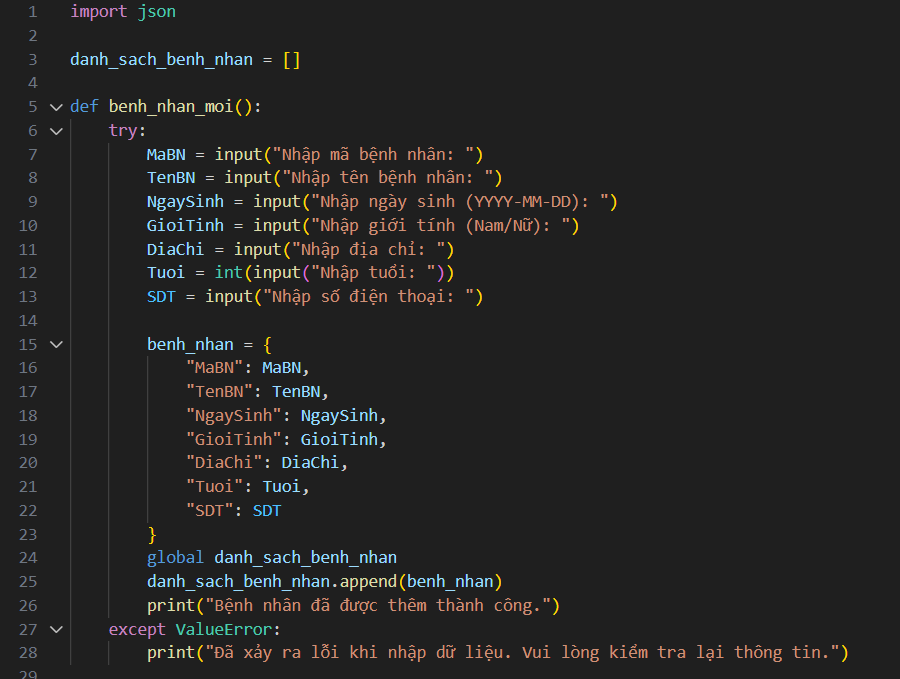
+ Tổng tiền từng đơn thuốc.

+ Thuốc được kê nhiều nhất.

+ Bệnh nhân có số lượng thuốc kê nhiều nhất.

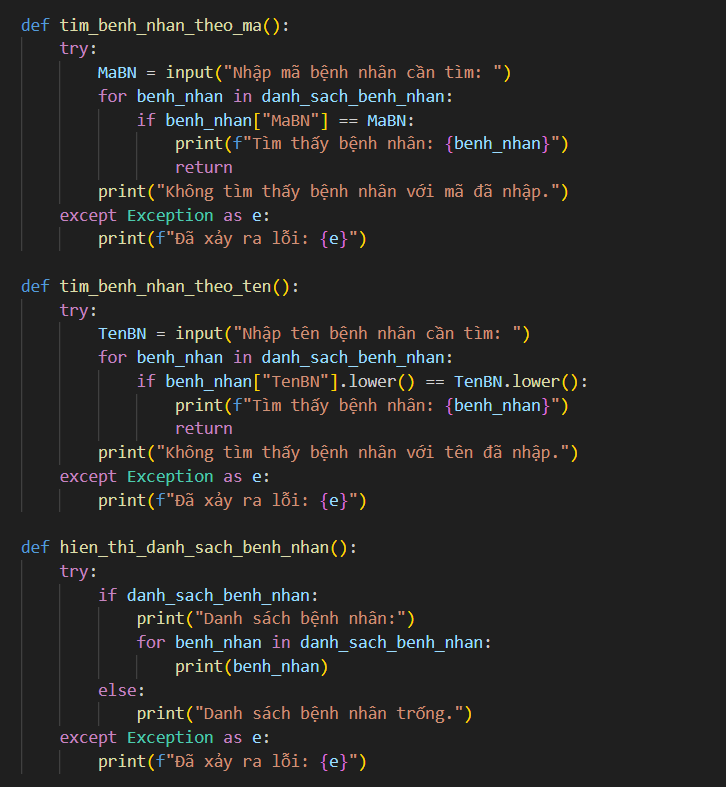
+ Dữ liệu đơn thuốc được lưu vào donthuoc.json, dễ dàng kiểm tra, thống kê và mở rộng.

# 3.3. Một số đoạn code trong menu.



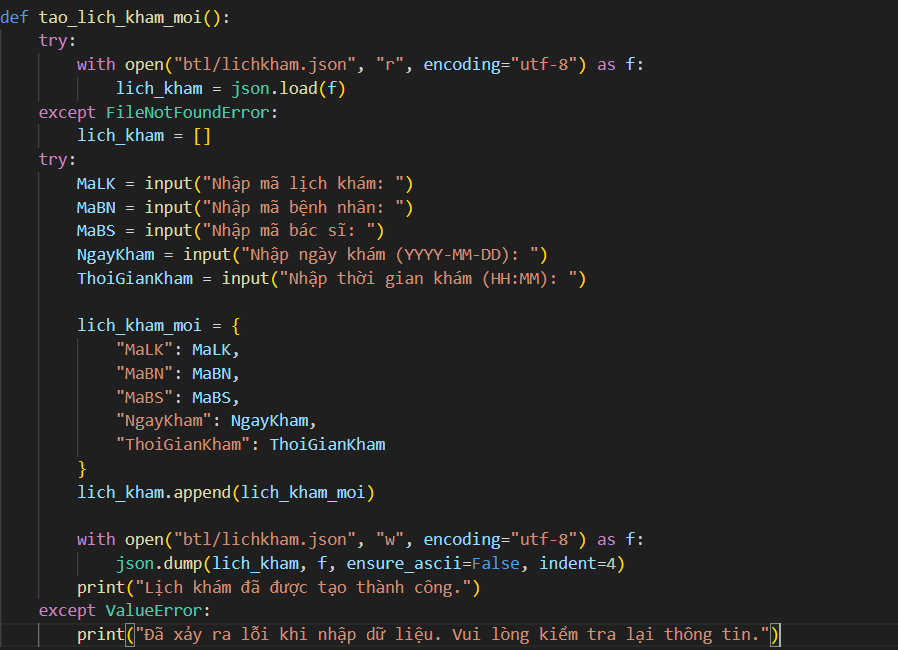
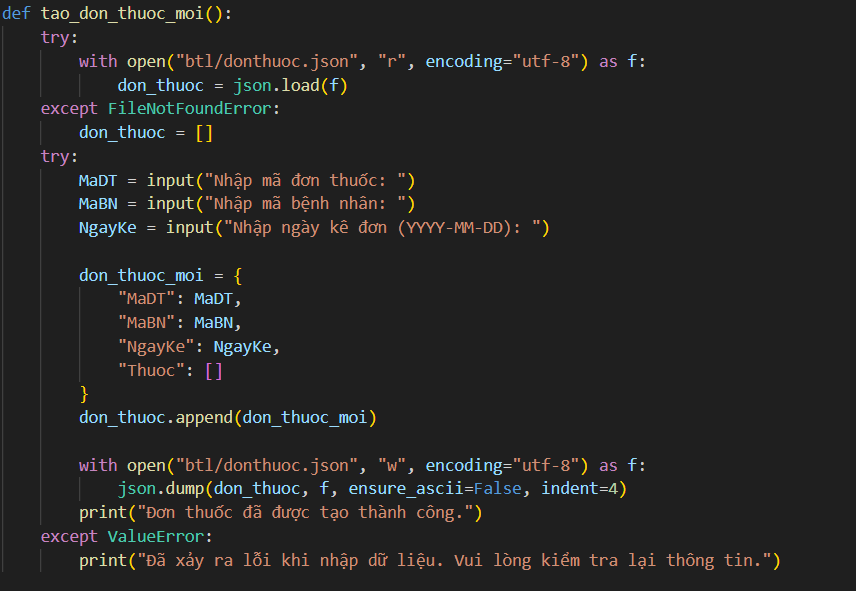
Hình 3.1.

Đoạn code được sử dụng để nhập thông tin bao gồm: “Mã bệnh nhân, tên, ngày sinh, giới tính, địa chỉ, tuổi và số điện thoại”.



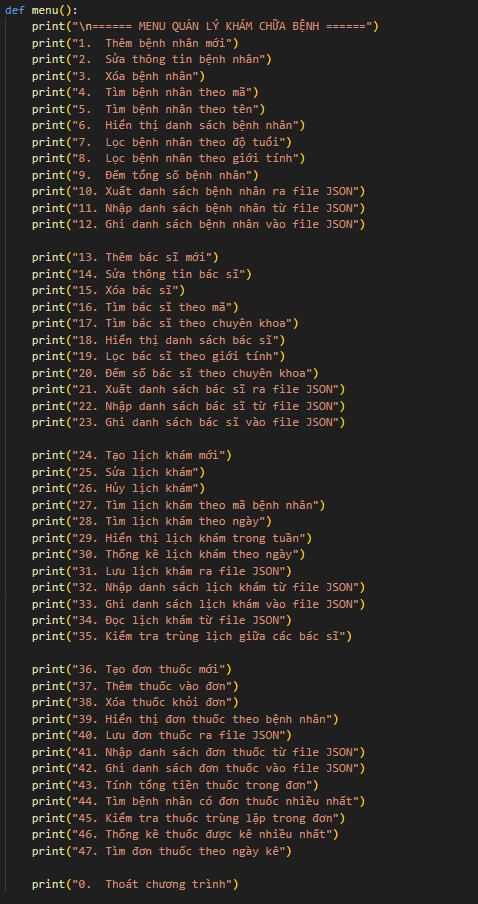
Hình 3.2.

Các chức năng trong menu bao gồm: Tìm bệnh nhân theo mã bệnh nhân, Tìm bệnh nhân theo tên và Hiện thị danh sách tất cả bệnh nhân đã nhập.

Hình 3.3.

Hai đoạn code trên được sử dụng để “Tạo lịch khám mới” và “Tạo đơn thuốc mới”.



Hình 3.4.

Đây là danh sách menu hiện thị đầy đủ bao gồm 47 chức năng chính và thoát chương trình.

# 3.4. Đánh giá và hướng phát triển

+ Chương trình hoạt động ổn định, thực hiện được các chức năng đặt ra.

+ Giao diện dòng lệnh thân thiện, dễ thao tác, phù hợp với sinh viên.

+ Quản lý dữ liệu đơn giản mà hiệu quả, nhờ sử dụng JSON.

**Hạn chế**

+ Chưa có phân quyền người dùng (bác sĩ / quản trị).

+ Giao diện còn đơn giản, chưa trực quan như phần mềm thực tế.

+ Chưa có xác thực đầu vào (kiểm tra định dạng số điện thoại, ngày giờ…).

**Hướng phát triển**

+ Chuyển sang giao diện đồ họa sử dụng tkinter hoặc xây dựng web app bằng Flask/Django.

+ Sử dụng cơ sở dữ liệu như SQLite hoặc MySQL để quản lý dữ liệu lớn và truy vấn hiệu quả hơn.

+ Thêm chức năng **đăng nhập**, **quản lý người dùng**, phân quyền và xuất báo cáo.

KẾT LUẬN

Sau khi hoàn thành dự án "Quản lý lịch khám bệnh", nhóm đã vận dụng thành thạo các kiến thức đã học trong môn Lập trình Python. Chương trình được xây dựng theo hướng modular hóa với nhiều chức năng tương tác thông qua menu.

Dự án không chỉ giúp nhóm hiểu rõ hơn về cách quản lý dữ liệu với danh sách và từ điển trong Python mà còn cung cấp trải nghiệm thực tế về thao tác với file JSON, xử lý lỗi và tổ chức cấu trúc chương trình.

Thông qua việc phát triển dự án, nhóm rút ra được tầm quan trọng của việc lên kế hoạch, chia nhỏ bài toán và kiểm thử thường xuyên để xây dựng một ứng dụng hoàn chỉnh.

Trong tương lai, nhóm có thể mở rộng thêm giao diện đồ họa (GUI) hoặc tích hợp với cơ sở dữ liệu SQL để nâng cao tính thực tiễn của hệ thống.

DANH MỤC TÀI LIỆU THAM KHẢO

1. Kuhlman, D. (2013). *A Python Book: Beginning Python, Advanced.*