TRƯỜNG ĐẠI HỌC AN GIANG

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**

# Chuyên đề Python (COS525)

XÂY DỰNG ỨNG DỤNG QUẢN LÝ HỌC SINH VỚI PYTHON, TKINTER VÀ MYSQL

**Giảng viên hướng dẫn**: **Sinh viên thực hiện**:

Ths. Nguyễn Ngọc Minh Võ Trình Thanh Trúc – DTH235800

Trần Thị Bích Tuyền – DTH235810

Lớp DH24TH3\_Nhóm3\_TổTH1

## An Giang, 11/2025

MỤC LỤC

[PHẦN MỞ ĐẦU 3](#_Toc214221576)

[1. Đặt vấn đề 3](#_Toc214221577)

[2. Tổng quan và cơ sở lý thuyết 3](#_Toc214221578)

[PHẦN NỘI DUNG 5](#_Toc214221579)

[3. Kết quả đạt được và Demo Code 5](#_Toc214221580)

[A. THIẾT KẾ GIAO DIỆN QUẢN LÝ NHÂN VIÊN 6](#_Toc214221581)

[B. CÁC BƯỚC XÂY DỰNG ỨNG DỤNG 8](#_Toc214221582)

[1. Cài đặt thư viện cần thiết: 8](#_Toc214221583)

[2. Thiết kế cơ sở dữ liệu SQL Server: 9](#_Toc214221584)

[3. Xây dựng giao diện người dùng (GUI) bằng Tkinter:\ 9](#_Toc214221585)

[4. Viết mã Python kết nối cơ sở dữ liệu 10](#_Toc214221586)

[5. Xây dựng giao diện Tkinter 11](#_Toc214221587)

[C. HƯỚNG DẪN CÀI ĐẶT & CODE ỨNG DỤNG 12](#_Toc214221588)

[1. Cài đặt môi trường 13](#_Toc214221589)

[2. Chuẩn bị CSDL SQL Server 13](#_Toc214221590)

[3. Viết code ứng dụng với Tkinter + SQL Server 16](#_Toc214221591)

[*Form “Đăng Nhập”:* 16](#_Toc214221592)

[*Form “Quản lý Học sinh”* 18](#_Toc214221593)

[*Form “Quản lý điểm”* 27](#_Toc214221594)

[PHẦN CUỐI 37](#_Toc214221595)

[Danh mục tài liệu tham 37](#_Toc214221596)

PHẦN MỞ ĐẦU

1. Đặt vấn đề

- Tính cần thiết của đề tài:

Hiện nay, việc quản lý học sinh tại nhiều trường học vẫn còn thực hiện thủ công (ghi chép giấy , sử dụng điểm danh bằng miệng) hoặc sử dụng các công cụ đơn giản như Excel. Cách làm này dễ dẫn đến sai sót, mất dữ liệu và khó khăn trong việc tổng hợp, tốn nhân sự trong thống kê thông tin học sinh. Các đề tài trước đây chủ yếu tập trung vào việc quản lý điểm hoặc danh sách học sinh, nhưng chưa tích hợp đầy đủ các chức năng quản lý lớp học, thông tin cá nhân và kết quả học tập trong cùng một hệ thống.

Vì vậy, đề tài này là cần thiết nhằm khắc phục những hạn chế đó. Đề tài mang ý nghĩa lý luận và thực tiễn trong việc áp dụng ngôn ngữ lập trình Python, thư viện Tkinter cũng như MySQL vào xây dựng hệ thống quản lý dữ liệu, giúp người học rèn luyện kỹ năng lập trình, xử lý dữ liệu và thiết kế giao diện ứng dụng thực tế. Đồng thời, đề tài góp phần hỗ trợ nhà trường trong công tác quản lý, lưu trữ và tra cứu thông tin học sinh một cách nhanh chóng, chính xác và hiệu quả.

Ngoài ra, trong bối cảnh chuyển đổi số trong giáo dục, việc áp dụng công nghệ thông tin vào quản lý học sinh là xu hướng tất yếu. Python là ngôn ngữ lập trình mạnh mẽ, phổ biến và dễ học, rất phù hợp cho sinh viên trong việc phát triển các ứng dụng quản lý có quy mô nhỏ và vừa. Đề tài này không chỉ có ý nghĩa thực tiễn trong quản lý dữ liệu học sinh mà còn giúp người thực hiện rèn luyện kỹ năng lập trình, thiết kế cơ sở dữ liệu và tư duy hệ thống, phục vụ cho học tập và nghiên cứu sau này.

- Mục tiêu nghiên cứu của đề tài:

Xây dựng một chương trình quản lý học sinh có khả năng:

+ Thêm, sửa, xóa thông tin học sinh.

+ Quản lý lớp học, điểm số và kết quả học tập của từng học sinh.

+ Thống kê số lượng học sinh, kết quả học tập và xếp loại học lực.

+ Giao diện thân thiện, dễ sử dụng, có thể áp dụng trong thực tế tại các trường học nhỏ và trung tâm giáo dục.

Tạo ra một phần mềm có tính ứng dụng cao, dễ triển khai, giúp tiết kiệm thời gian và nâng cao hiệu quả trong công tác quản lý học sinh.

Bên cạnh các chức năng cơ bản, chương trình hướng đến việc sử dụng MySQL để lưu trữ dữ liệu học sinh, đảm bảo tính toàn vẹn và bảo mật thông tin.

2. Tổng quan và cơ sở lý thuyết

- Đặt vấn đề

Trong công tác quản lý giáo dục và tối ưu nguồn nhân lực như hiện nay, việc lưu trữ, cập nhật và tìm kiếm thông tin học sinh là một khâu quan trọng. Tuy nhiên, hầu hết các trường học hiện nay vẫn đang sử dụng hình thức quản lý thủ công bằng sổ sách hoặc file Excel, gây khó khăn khi cần tra cứu nhanh hoặc thống kê dữ liệu. Do đó, việc xây dựng một ứng dụng quản lý học sinh bằng Python giúp tự động hóa quy trình quản lý, giảm thiểu sai sót và nâng cao hiệu quả công việc là rất cần thiết.

- Lịch sử giải quyết vấn đề:

Đã có một số đề tài trước đây xây dựng chương trình quản lý học sinh sử dụng các ngôn ngữ như C, Java hoặc PHP. Tuy nhiên, các đề tài này chủ yếu tập trung vào chức năng cơ bản, giao diện còn hạn chế hoặc chưa tận dụng được sự đơn giản và mạnh mẽ của Python. Đề tài này được thực hiện với mục tiêu sử dụng Python và thư viện Tkinter để xây dựng một ứng dụng có giao diện trực quan, thân thiện với người dùng và có khả năng mở rộng trong tương lai.

- Phạm vi của đề tài:

Đề tài tập trung xây dựng phần mềm quản lý thông tin học sinh, bao gồm các chức năng: thêm, sửa, xóa, tìm kiếm, hiển thị danh sách và quản lý điểm học tập.

Phạm vi không bao gồm quản lý giáo viên, môn học hoặc điểm rèn luyện. Ứng dụng phù hợp với các trường tiểu học, THCS hoặc trung tâm giáo dục quy mô nhỏ.

- Phương pháp nghiên cứu/ hướng giải quyết vấn đề:

Đề tài được thực hiện dựa trên phương pháp nghiên cứu ứng dụng, bao gồm:

+ Phân tích yêu cầu hệ thống và thiết kế cơ sở dữ liệu.

+ Sử dụng ngôn ngữ lập trình Python để xây dựng ứng dụng.

+ Áp dụng thư viện Tkinter để tạo giao diện đồ họa (GUI).

+ Dùng My SQL để lưu trữ và truy xuất dữ liệu học sinh.

+ Kiểm thử chương trình nhằm đảm bảo tính ổn định và khả năng hoạt động thực tế.

- Cơ sở lý thuyết:

+ Ngôn ngữ Python: Là ngôn ngữ lập trình hướng đối tượng, dễ đọc, dễ hiểu và có thư viện hỗ trợ mạnh mẽ cho phát triển phần mềm.

+ Thư viện Tkinter: Là thư viện mặc định của Python dùng để xây dựng giao diện đồ họa, giúp người dùng thao tác trực quan với các chức năng như nút bấm, hộp nhập liệu, bảng dữ liệu...

+ Cơ sở dữ liệu MySQL: Là hệ quản trị cơ sở dữ liệu nhẹ, phù hợp cho ứng dụng có quy mô nhỏ, dễ tích hợp vào chương trình Python.

PHẦN NỘI DUNG

3. Kết quả đạt được và Demo Code

Đề tài “Xây dựng ứng dụng Quản lý Học Sinh” được thực hiện bằng ngôn ngữ Python, sử dụng thư viện Tkinter để thiết kế giao diện người dùng. Ứng dụng gồm các Form cơ bản sau:

- Form “Đăng nhập”:

+ Cho phép người dùng nhập tên đăng nhập, mật khẩu và phân quyền.

+ Ứng dụng kiểm tra thông tin đăng nhập trong cơ sở dữ liệu.

+ Nếu hợp lệ, người dùng sẽ được chuyển đến giao diện tương ứng với quyền hạn (Admin hoặc Giáo viên).

- Form “Quản lý học sinh”: dành cho người dùng có quyền Admin, với các chức năng:

+ Thêm học sinh mới vào hệ thống.

+ Sửa thông tin học sinh đã có.

+ Xóa học sinh khỏi danh sách

+ Cập nhật số lượng học sinh theo lớp hoặc học sinh có trong danh sách.

- Form “Quản lý điểm”: dành cho Giáo viên, hỗ trợ các chức năng:

+ Nhập và chỉnh sửa điểm học kỳ I, học kỳ II của từng môn học.

+ Tính điểm trung bình hai học kỳ.

+ Xếp loại học lực cho từng học sinh.

+ Cập nhật số lượng học sinh theo lớp hoặc học sinh có trong danh sách.

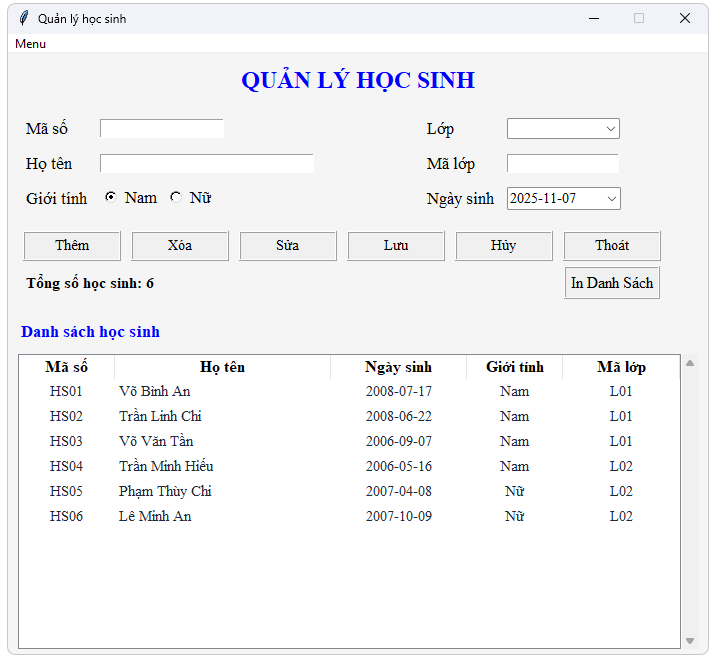
Vận dụng kỹ thuật lập trình ngôn ngữ Python để giải quyết các vấn đề trong thực tế, ứng với đề tài mà Sinh viên đã lựa chọn.

1. THIẾT KẾ GIAO DIỆN QUẢN LÝ NHÂN VIÊN

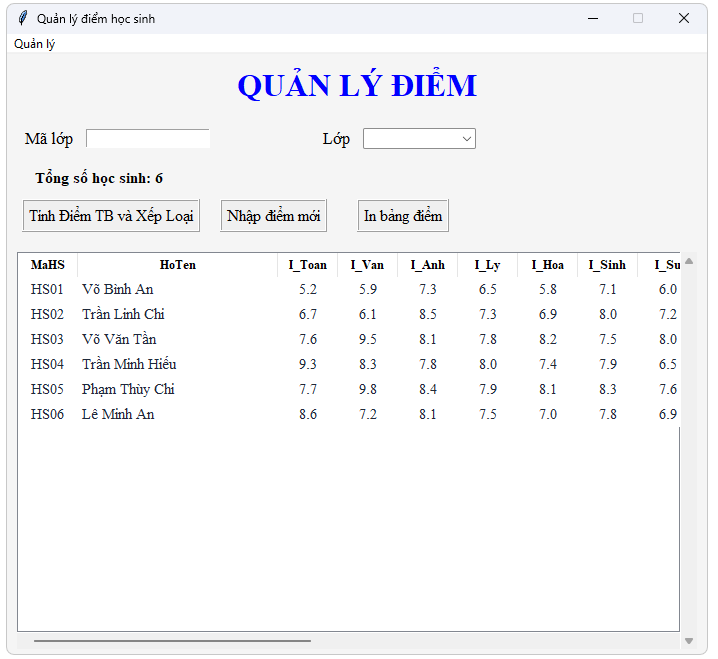
Form “Đăng nhập”:



Form “Quản lý học sinh”



Form “Quản lý điểm”



1. CÁC BƯỚC XÂY DỰNG ỨNG DỤNG

Để xây dựng ứng dụng quản lý học sinh bằng Python, Tkinter và SQL Server, trước tiên cần cài đặt các thư viện cần thiết, bao gồm Tkinter để tạo giao diện người dùng và pyodbc để kết nối với cơ sở dữ liệu SQL Server. Tiếp theo, thiết kế cơ sở dữ liệu SQL Server nhằm lưu trữ các thông tin về học sinh như mã học sinh, họ tên, ngày sinh, giới tính, lớp, và điểm số. Sau đó, sử dụng Tkinter để xây dựng giao diện trực quan, giúp người dùng dễ dàng thao tác với dữ liệu. Cuối cùng, kết hợp Python với thư viện pyodbc để thực hiện các chức năng thêm, xóa, sửa và hiển thị danh sách học sinh, giúp ứng dụng hoạt động hiệu quả và hỗ trợ việc quản lý thông tin học sinh một cách khoa học, thuận tiện và chính xác.

1. Cài đặt thư viện cần thiết:

Cài đặt các thư viện cần thiết:

- Python: Đảm bảo bạn đã cài đặt Python trên máy tính của mình.

- SQL Server: Cài đặt máy chủ SQLServer và tạo một cơ sở dữ liệu cho ứng dụng của bạn.

- pyodbc: Cài đặt thư viện này để kết nối Python với cơ sở dữ liệu SQLServer (pip install pyodbc).

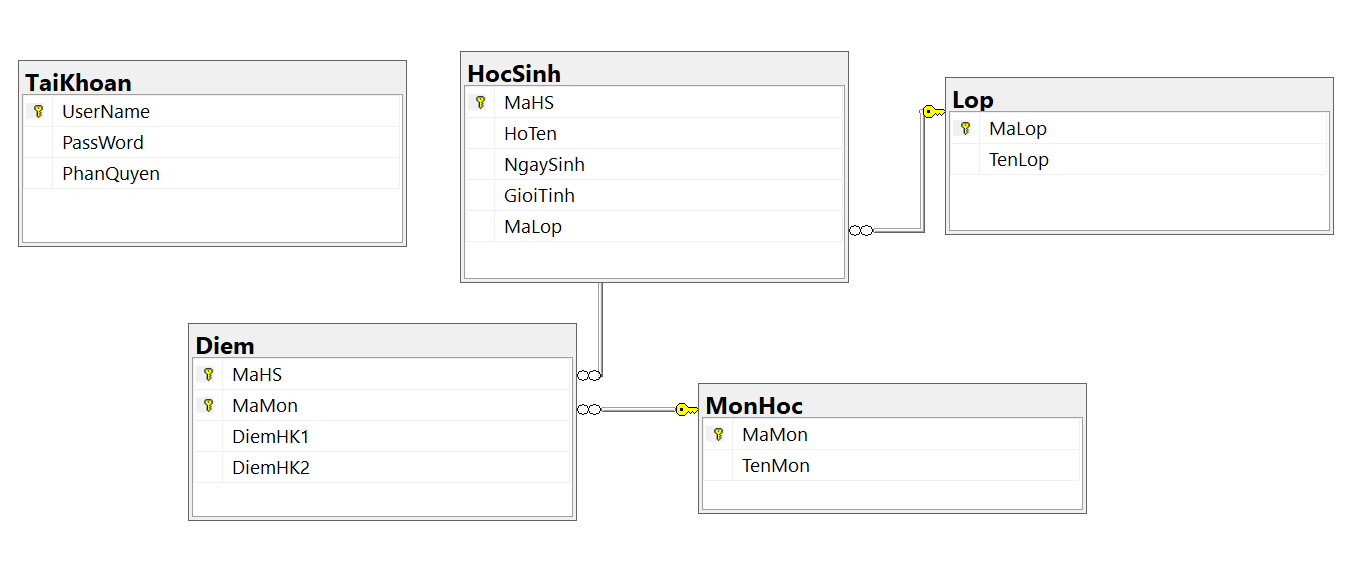
- tkcalendar: Cài đặt thư viện này để cho phép chọn ngày dễ dàng trong giao diện Tkinter (pip install tkcalendar).

- subprocess: Import thư viện này để chạy các file Python khác từ bên trong chương trình Python đang chạy.

- sys: Import thư viện này để cho phép làm việc với trình thông dịch Python.

- filedialog: Import thư viện này để lưu file mà không cần tự gõ đường dẫn.

2. Thiết kế cơ sở dữ liệu SQL Server:



3. Xây dựng giao diện người dùng (GUI) bằng Tkinter:\

- Sử dụng thư viện Tkinter để tạo các cửa sổ, trường nhập liệu, nút bấm và bảng hiển thị thông tin Học sinh.

- Thiết kế giao diện cho phép người dùng thực hiện các chức năng sau:

+ Đăng nhập với tên và mật khẩu được cấp, cùng phân quyền của người dùng.

+ Thêm thông tin Học sinh mới.

+ Lưu thông tin Học sinh.

+ Sửa thông tin Học sinh.

+ Hủy thông tin Học sinh.

+ Xóa thông tin Học sinh.

+ Tính điểm trung bình và xếp loại Học sinh theo lớp.

+ Sửa điểm cho các Học sinh đã có điểm hoặc nhập thêm điểm khi có Học sinh chưa có dữ liệu điểm số.

+ In danh sách thông tin Học sinh theo lớp.

+ In danh sách điểm Học sinh theo lớp.

4. Viết mã Python kết nối cơ sở dữ liệu

- Sử dụng thư viện pyodbc để kết nối với cơ sở dữ liệu SQL Server

- Form “Đăng nhập” viết các hàm:

+ Đăng nhập:

Lấy dữ liệu từ các trường nhập liệu trên giao diện như username, password và phân quyền để so sánh với cơ sở dữ liệu từ đó đưa người dùng đến đúng các form có thể truy cập và sử dụng.

Trong hàm đăng nhập nếu phân quyền là “Admin” thì trỏ đến hàm mở form “Quản lý học sinh”. Nếu phân quyền là “Giáo viên” thì trỏ đến hàm mở form “Quản lý điểm.

- Form “Quản lý học sinh” viết các hàm:

+ Hàm xử lý

* Tải và hiển thị: tải và hiển thị dữ liệu Học sinh lên giao diện bảng.
* Chọn lớp: khi chọn lớp tự hiển thị mã lớp và hiển thị danh sách Học sinh theo lớp.
* Xóa các dữ liệu trên các trường nhập liệu trên giao diện sau các thao tác

+ Hàm cho các nút

* Menu: chứa nút đăng xuất khỏi form.
* Thêm Học sinh: lấy dữ liệu từ các trường nhập liệu trên giao diện và chèn vào bảng hocsinh nếu chưa có mã học sinh đó.
* Xóa thông tin Học sinh: xóa nhân viên khỏi cơ sở dữ liệu dựa trên maso.
* Sửa thông tin Học sinh: sửa đổi thông tin Học sinh trên danh sách học sinh.
* Lưu thông tin sửa Học sinh: sửa đổi thông tin Học sinh trong cơ sở dữ liệu dựa trên maso.
* Hủy thông tin Học sinh: hủy các thông tin khi nhấn nút sửa và chưa lưu.
* Thoát: thoát khỏi form “Quản lý học sinh”.
* In danh sách học sinh: xuất dữ liệu danh sách Học sinh dưới dạng file excel dựa vào lớp.

- Form “Quản lý điểm” viết các hàm:

+ Hàm xử lý

* Menu: chứa nút đăng xuất khỏi form.
* Thêm cột: thêm cột tính điểm trung bình và xếp loại vào cơ sở dữ liệu
* Chọn lớp: khi chọn lớp tự hiển thị mã lớp và hiển thị danh sách Học sinh theo lớp.
* Tải và hiển thị: tải và hiển thị dữ liệu Học sinh lên giao diện bảng.
* Mở cửa sổ sửa điểm: mở 1 Form nhập lại điểm với các học sinh vừa được thêm vào ở Form “Quản lý Học sinh” hoặc các Học sinh cần sửa và lưu điểm dựa vào maso và hoten.

+ Hàm cho các nút

* Tính điểm và xếp loại: dựa vào các trường như điểm toán, văn, anh có trong cơ sở dữ liệu và các dữ liệu mới được nhập để tính điểm và xếp loại Học sinh theo lớp.
* Sửa điểm: được sử dụng đối với các Học sinh vừa mới được thêm ở Form “ Quản lý Học sinh” hoặc các Học sinh cần sửa điểm lại dựa vào hàm mở cửa sổ sửa điểm.
* In danh sách học sinh: xuất dữ liệu danh sách Học sinh dưới dạng file excel dựa vào lớp.

5. Xây dựng giao diện Tkinter

- Liên kết các nút bấm trên giao diện Tkinter với các hàm Python tương ứng để xử lý dữ liệu và tương tác với cơ sở dữ liệu.

**Các Chức Năng**

***Form “Đăng Nhập”:***

o Đăng nhập: Khi người dùng nhập đúng tên đăng nhập, mật khẩu, phân quyền giống như các trường “username”, “password”, “phân quyền” trên cơ sở dữ liệu và nhấn nút “Đăng nhập”, mã Python sẽ mở đúng Form theo “phân quyền” người dùng có thể mở trên giao diện Tkinter.

***Form “ Quản lý Học sinh”:***

o Đăng xuất: Khi người dùng nhấn vào nút “Quản lý ” trên menu sẽ hiện ra nút “Đăng xuất”. Nếu người nhấn nút “Đăng xuất”, mã Python sẽ tiếng hành mở Form “Đẳng nhập” và đóng Form “Quản lý Học sinh” lại.

o Chọn danh sách theo lớp: Khi người dùng chọn lớp, danh sách Học sinh sẽ được sắp xếp theo lớp trên giao diện Tkinter.

o Thêm Học sinh: Khi người dùng nhập mã số, họ tên, mã lớp (hoặc chọn tên lớp), giới tính, ngày sinh và nhân nút “Thêm”, mã Python sẽ lấy các giá trị này và thêm 1 bảng ghi mới vào bảng hocsinh trong SQL Server.

o Xem Học sinh: Khi người dùng mở cửa sổ ứng dụng, mã Python sẽ truy vấn cơ sở dữ liệu, lấy tất thông tin Học sinh hiển thị chúng dưới dạng một danh sách trên giao diện Tkinter.

o Xóa Học sinh: Khi người dùng chọn 1 Học sinh trên danh sách giao diện Tkinter và nhấn nút “Xóa”, Học sinh đó sẽ bị xóa trong cơ sở dữ liệu dựa vào mã học sinh, bị xóa trong danh sách trên giao diện Tkinter.

o Sửa Học sinh: Khi người dùng chọn 1 Học sinh trên danh sách giao diện Tkinter và nhấn nút “Sửa”. Các trường dữ liệu sẽ hiện lên thông tin để người dùng sửa.

o Lưu Học sinh: Khi người dùng “Sửa” xong thông tin Học sinh và nhấn nút “Lưu” dữ liệu mới của học sinh sẽ được lưu lại trên danh sách của giao diện Tkinter và trong cơ sở dữ liệu.

o Hủy thao tác: Khi người dùng nhấn nút “Hủy” thao tác “Sửa” vừa được thực hiện sẽ bị hủy.

o Thoát: Nhấn nút “Thoát” để thoát khỏi Form “Quản lý Học sinh”.

o In danh sách: Khi người dùng muốn in danh sách thông tin học sinh theo lớp. Người dùng cần chọn lớp cần in và nhấn vào nút “In danh sách” để xuất file dữ liệu dưới định dạng file excel.

**Form “Quản lý điểm”:**

o Thêm 2 cột “Tính điểm trung bình” và “Xếp loại”: Khi người dùng từ đăng nhập vào Form “Quản lý điểm”, mã Python tiến hành kiểm tra lớp đó đã được thêm 2 cột “Tính điểm trung bình” và “Xếp loại” trên cơ sở dữ liệu hay chưa. Nếu chưa mã Python sẽ tiếng hành thêm vào 2 cột. Nếu 2 cột đã tồn tại sẽ in ra màn hình 2 cột đã tồn tại.

o Đăng xuất: Khi người dùng nhấn vào nút “Quản lý ” trên menu sẽ hiện ra nút “Đăng xuất”. Nếu người nhấn nút “Đăng xuất”, mã Python sẽ tiếng hành mở Form “Đẳng nhập” và đóng Form “Quản lý điểm” lại.

o Chọn danh sách theo lớp: Khi người dùng chọn lớp, danh sách Học sinh sẽ được sắp xếp theo lớp trên giao diện Tkinter.

o Tính điểm trung bình và xếp loại: Khi người dùng nhấn chọn “Lớp” cần tính và nhấn nút “Tính điểm TB và xếp loại”, mã Python tiến hành dựa vào dữ liệu điểm có trong cơ sở dữ liệu để tính cho cả lớp đó và cập nhật lên cơ sở dữ liệu cũng như lên danh sác Học sinh trên giao diện Tkinter.

o Nhập điểm mới: Khi người dùng muốn nhập điểm cho Học sinh chưa có cơ sở dữ liệu về điểm hoặc muốn sửa điểm cho Học sinh. Người dùng cần chọn Học sinh cần thực hiện nhập điểm và nháy nút “Nhập điểm mới”, mã Python sẽ hiện ra cửa sổ cho phép người dùng nhập và lưu điểm mới cho Học sinh.

o In danh sách: Khi người dùng muốn in danh sách điểm học sinh theo lớp. Người dùng cần chọn lớp cần in và nhấn vào nút “In danh sách” để xuất file dữ liệu dưới định dạng file excel.

o Xem điểm Học sinh: Khi người dùng mở cửa sổ ứng dụng, mã Python sẽ truy vấn cơ sở dữ liệu, lấy tất thông tin điểm Học sinh hiển thị chúng dưới dạng một danh sách trên giao diện Tkinter.

1. HƯỚNG DẪN CÀI ĐẶT & CODE ỨNG DỤNG

Ứng dụng sẽ có:

+ Giao diện nhập thông tin Học sinh (Mã số, Họ tên, Lớp, Mã lớp, Giới tính, Ngày sinh).

+ Giao diện tính điểm cho học sinh (Tính điểm trung bình 2 học kỳ, xếp loại).

+ Giao diện nhập thông tin để đăng nhập (Tên đăng nhập, mật khẩu, và phân quyền).

+ Bảng danh sách Học sinh.

+ Chức năng CRUD (Thêm, Sửa, Xóa, Lưu).

+ Bảng danh sách điểm Học sinh.

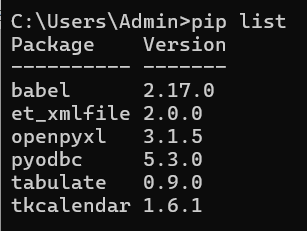
+ Chức năng Nhập điểm mới, tính điểm trung bình và xếp loại.

+ Dữ liệu lưu trực tiếp vào SQL Server Database (không mất khi tắt ứng dụng).

+ Giao diện Tkinter thiết kế chuẩn như hình minh họa ở trên

1. Cài đặt môi trường

Trước tiên cài đặt thư viện cần thiết:

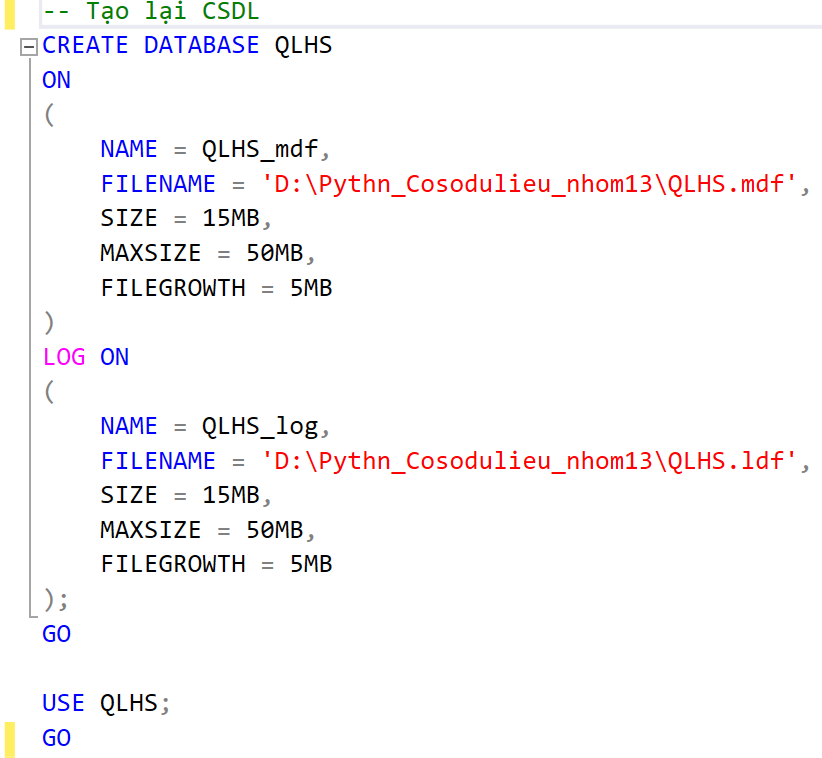


Giải thích:

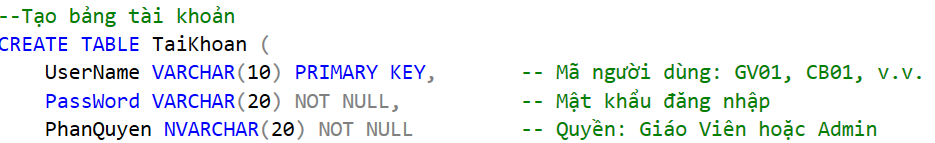
* **Babel:** định dạng ngày tháng, số và tiền tệ phù hợp với ngôn ngữ hoặc khu vực của người dùng.
* **et xmlfile:** là thư viện phụ trợ được sử dụng bởi **openpyxl**, có nhiệm vụ ghi dữ liệu XML một cách hiệu quả, đặc biệt hữu ích khi xử lý các tệp Excel định dạng .xlsx.
* **openpyxl** là thư viện cho phép đọc, ghi và chỉnh sửa các tệp Excel (.xlsx) bằng Python, thường được dùng để xuất danh sách, bảng điểm trong ứng dụng.
* **pyodbc:** là thư viện giúp kết nối Python với các hệ quản trị cơ sở dữ liệu SQL Server phục vụ cho việc truy xuất và xử lý dữ liệu.
* **tabulate:** hiển thị dữ liệu dạng bảng trong terminal, giúp in ra bảng kết quả rõ ràng, dễ đọc và có thể xuất theo nhiều kiểu định dạng khác nhau.
* **Tkcalendar:** là tiện ích mở rộng cho Tkinter, cung cấp các widget như Calendar và DateEntry để chọn ngày tháng.

2. Chuẩn bị CSDL SQL Server

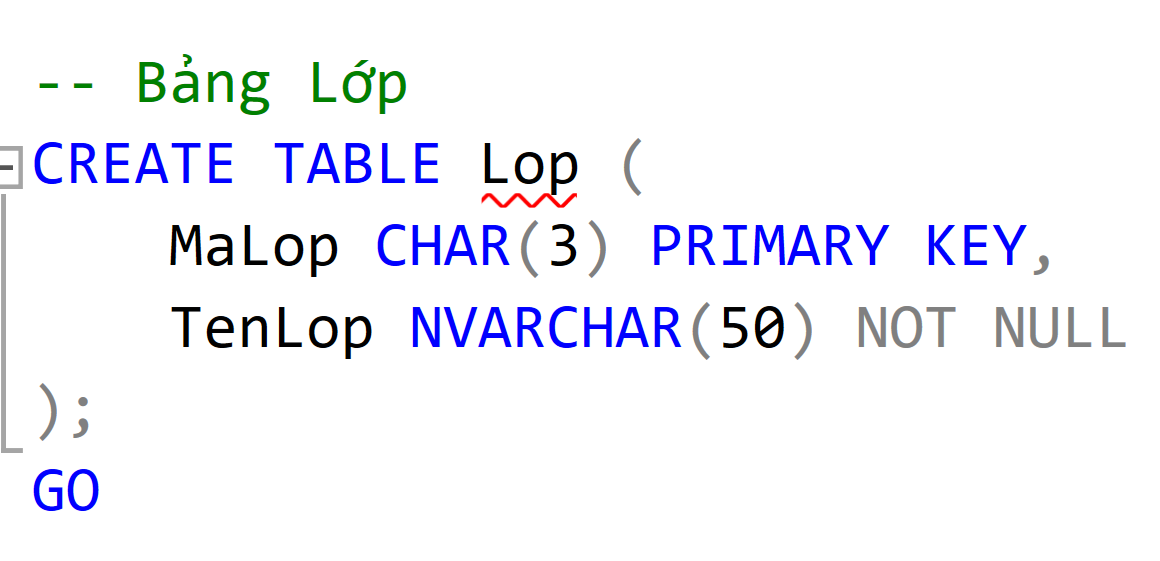
Mở SQL Server rồi tạo database và bảng:



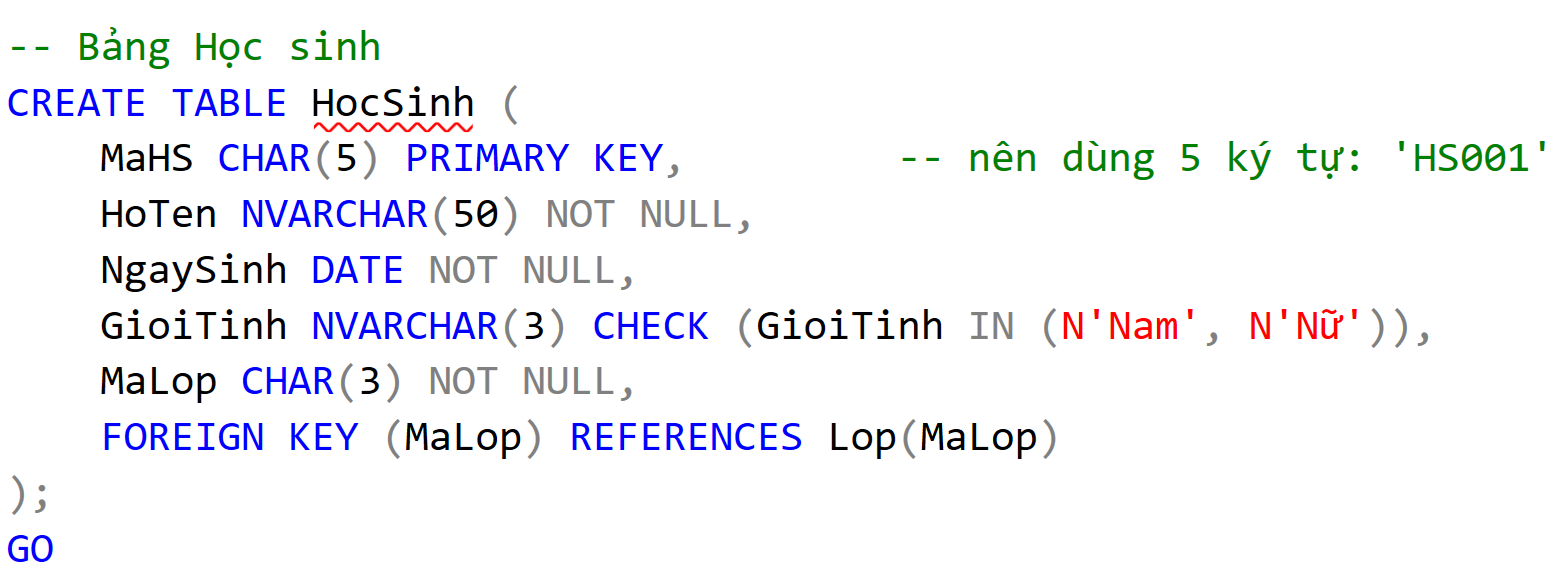
**Bảng TaiKhoan**



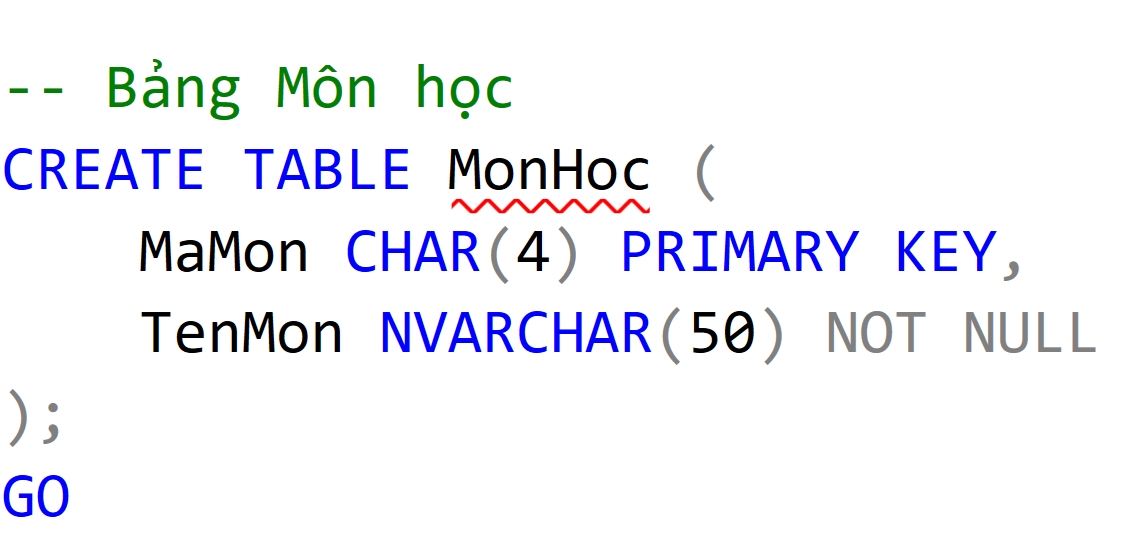
**Bảng Lớp:**



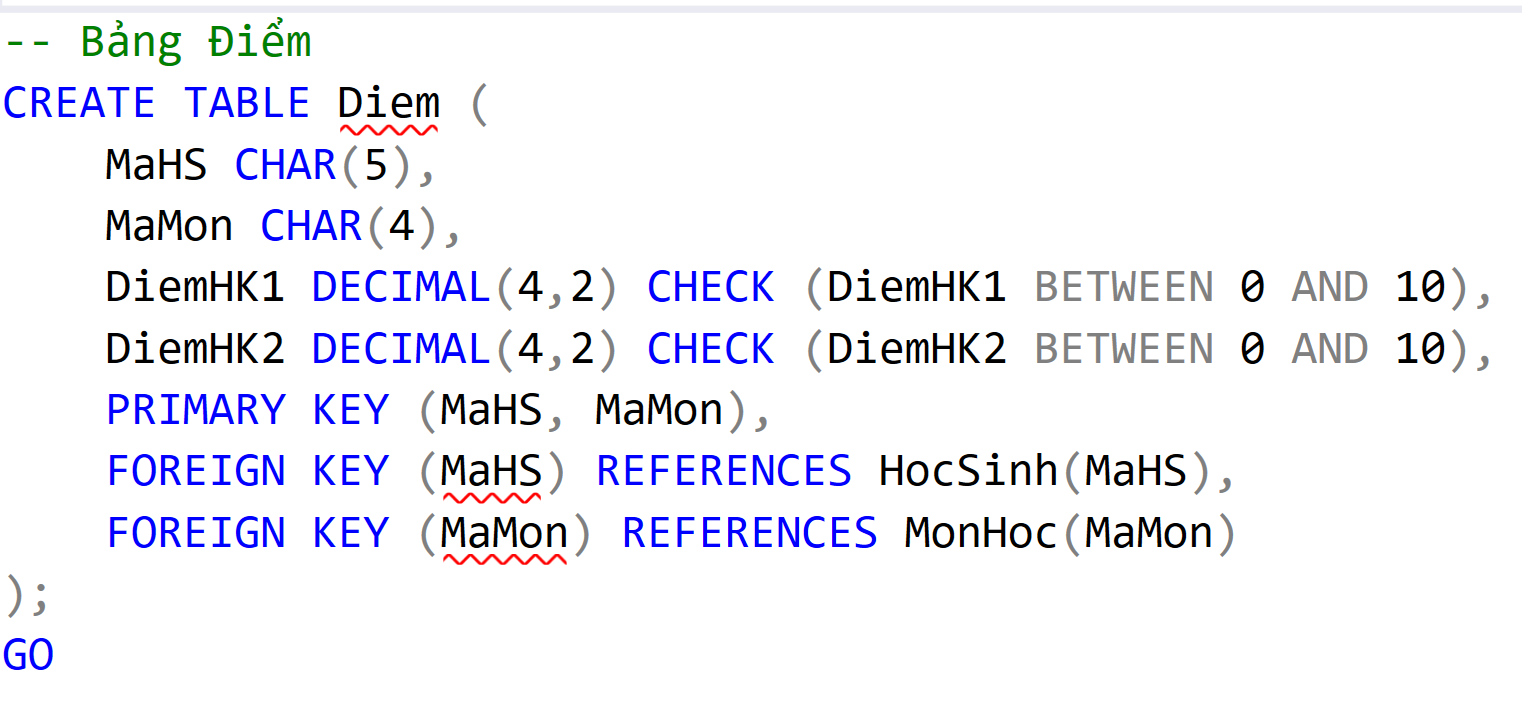
**Bảng Học Sinh:**



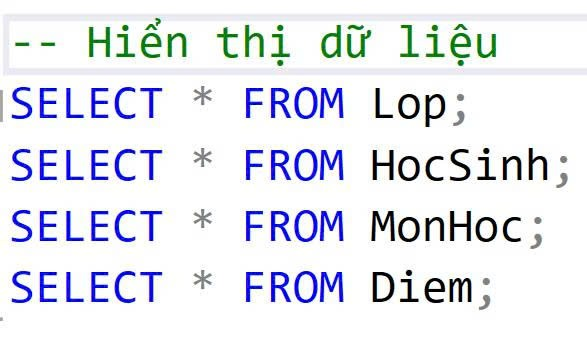
**Bảng Môn Học:**



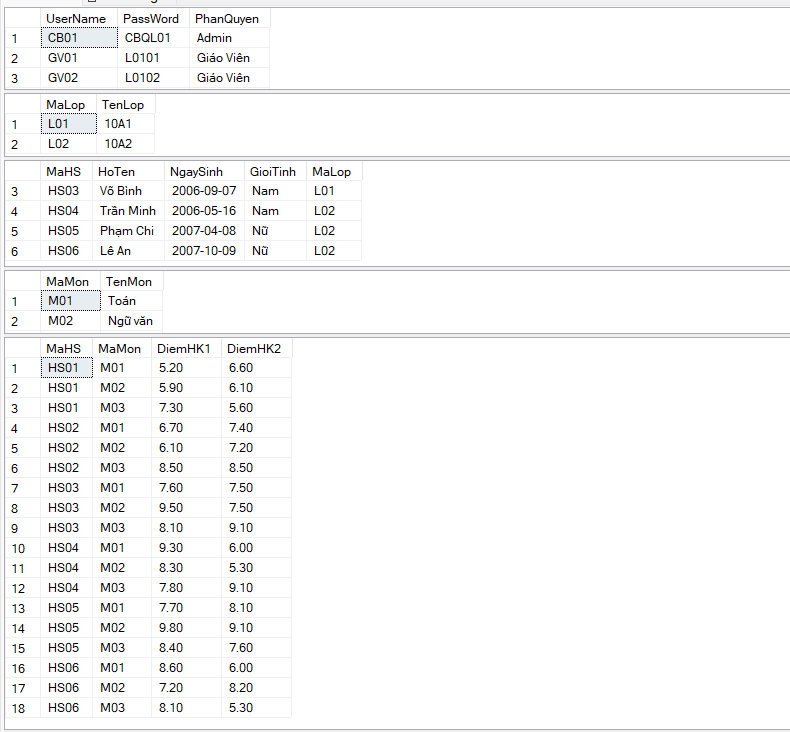
**Bảng Điểm:**



**Kiểm tra dữ liêu:**

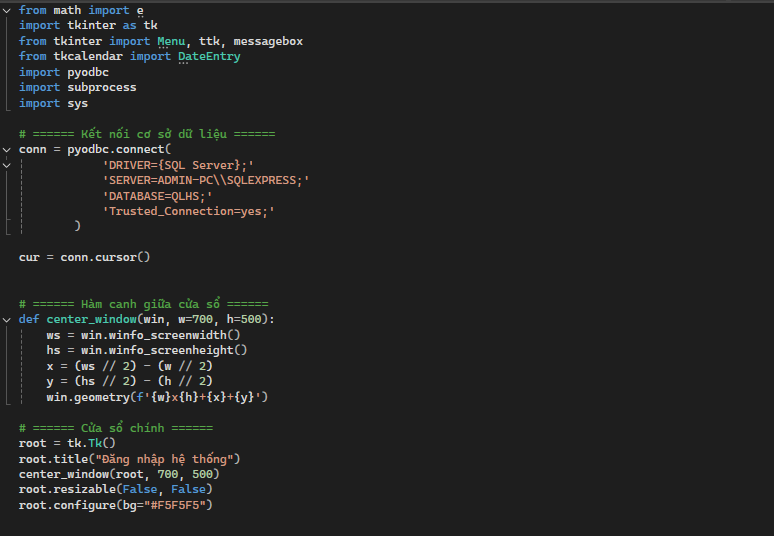


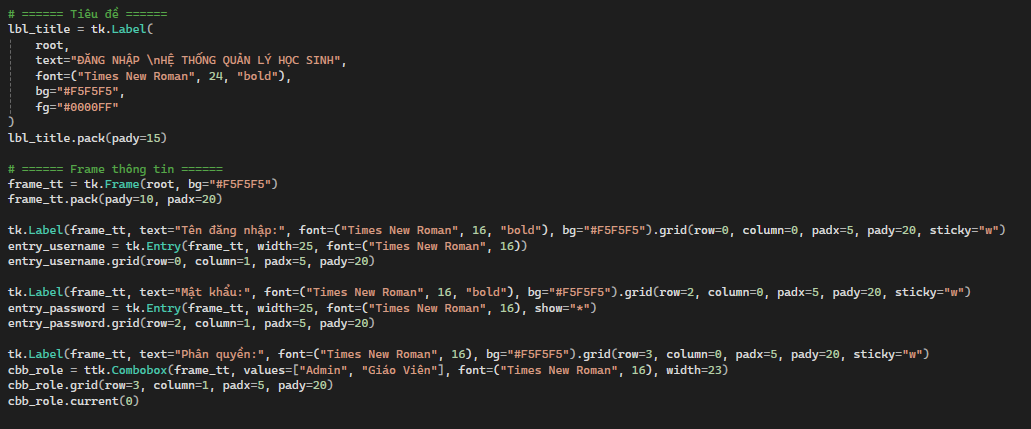
**Dữ liêu sau khi import:**

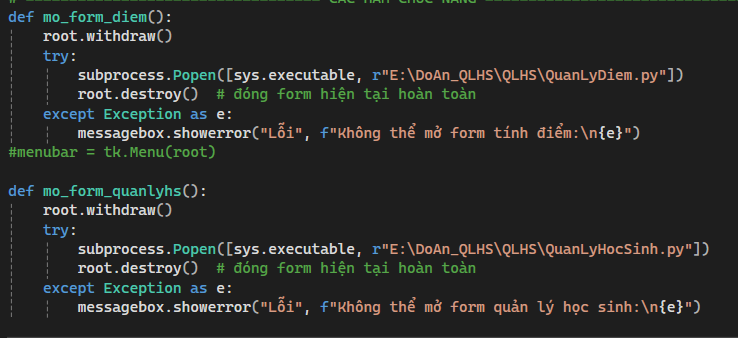


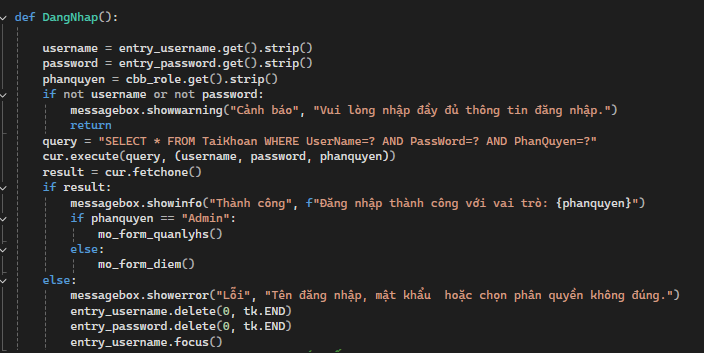
3. Viết code ứng dụng với Tkinter + SQL Server

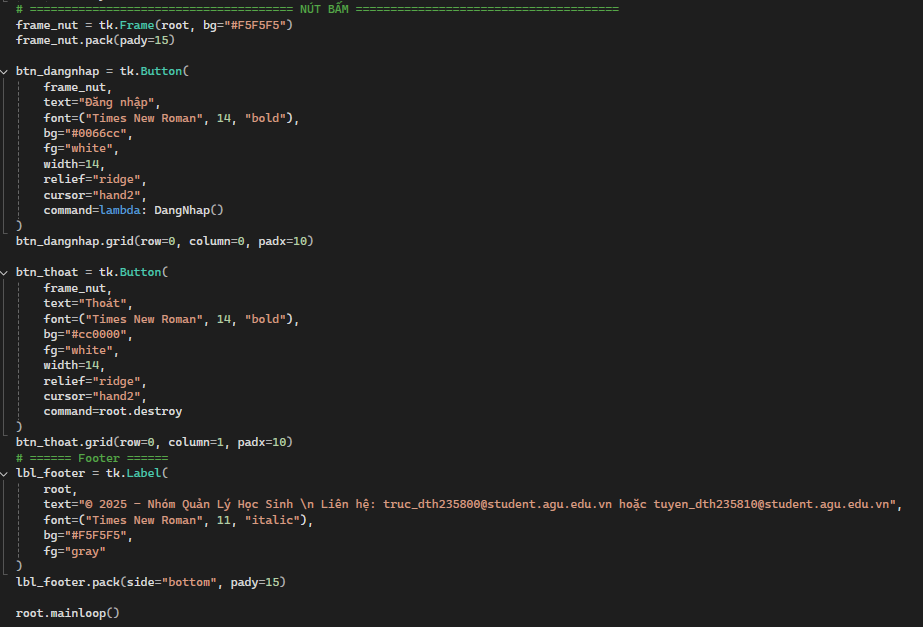
*Form “Đăng Nhập”:*



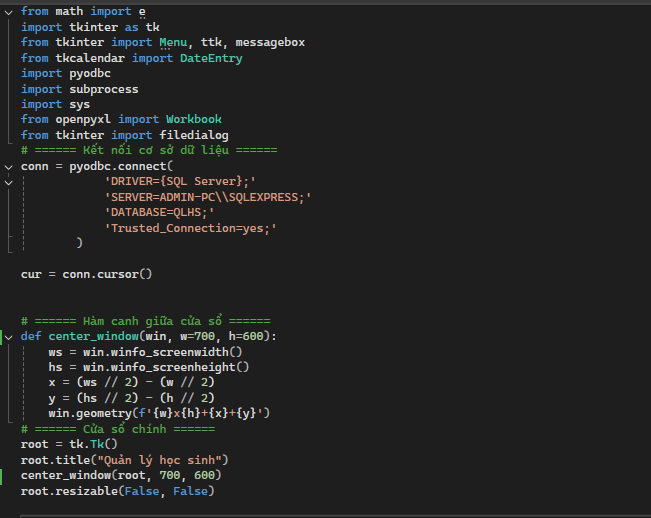


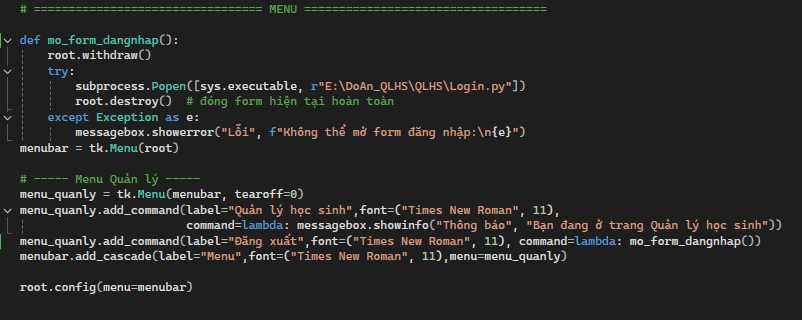


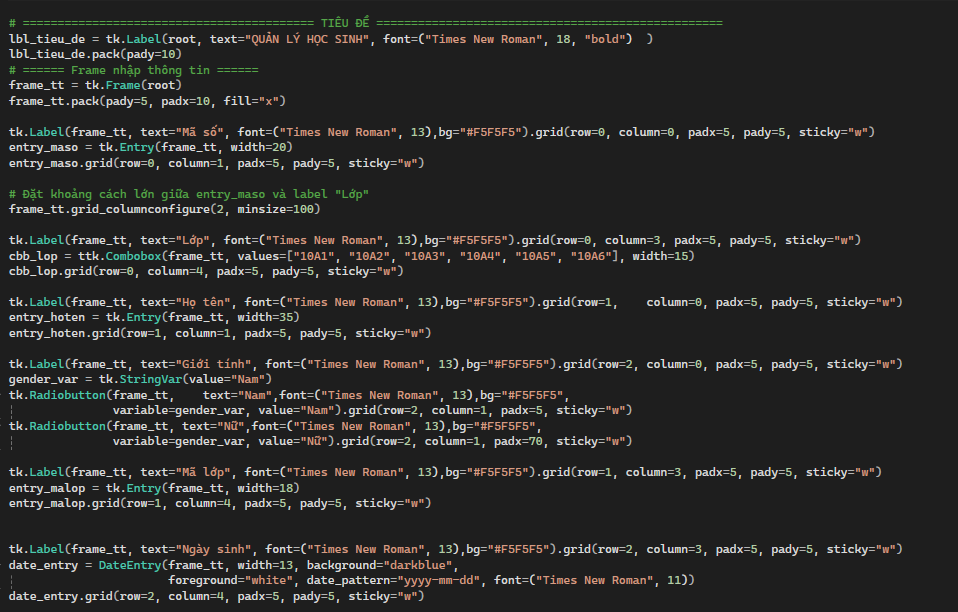


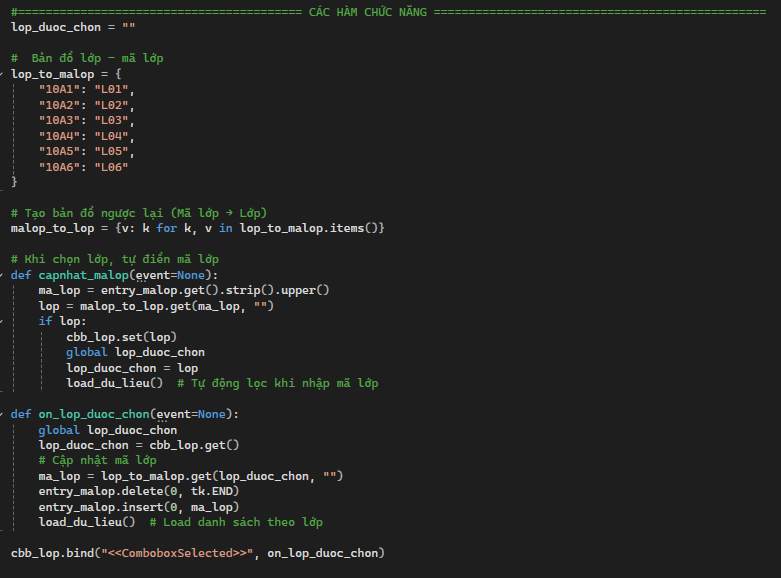


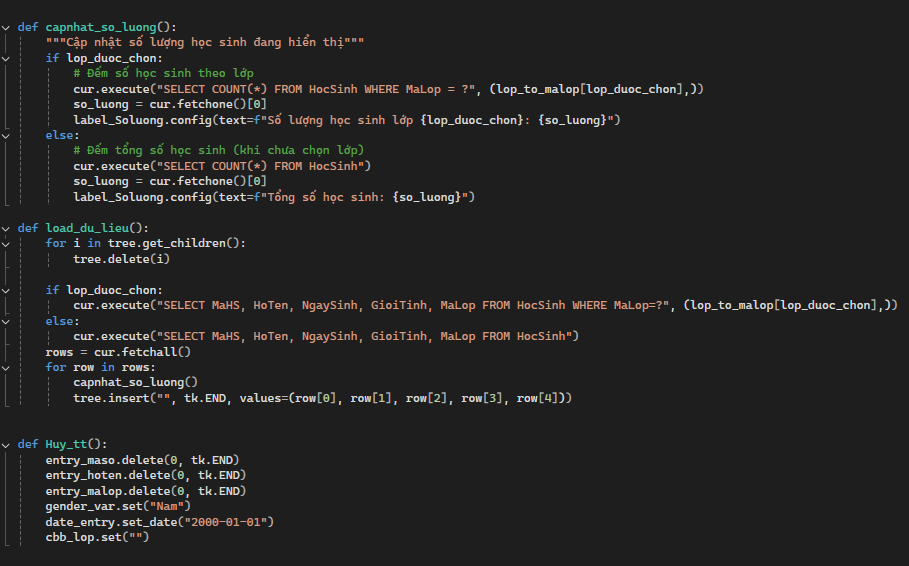
*Form “Quản lý Học sinh”*



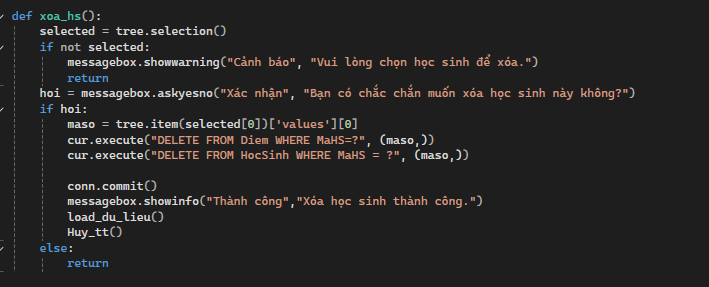




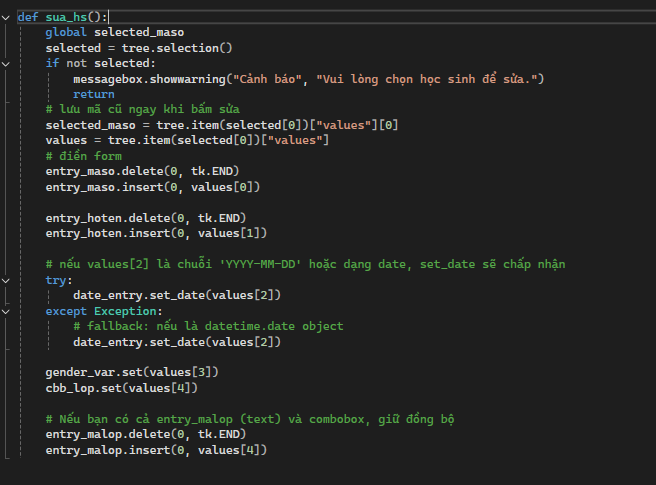


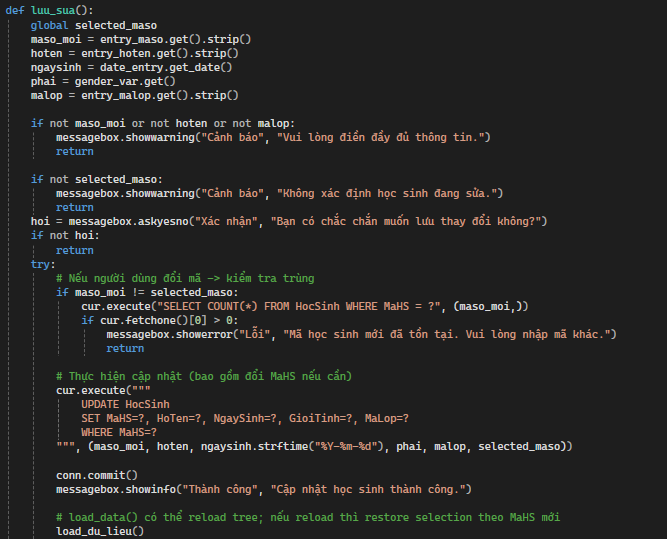






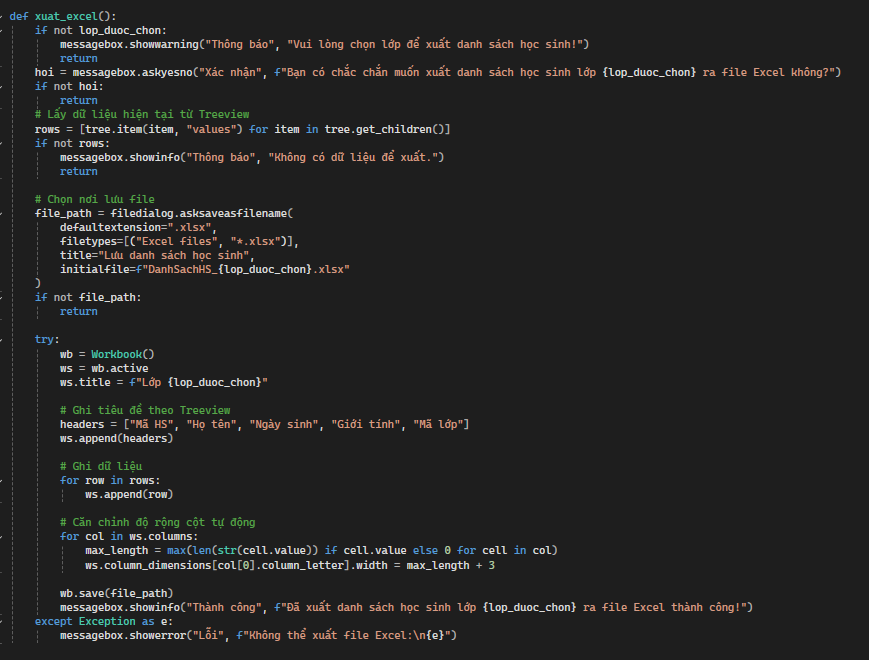


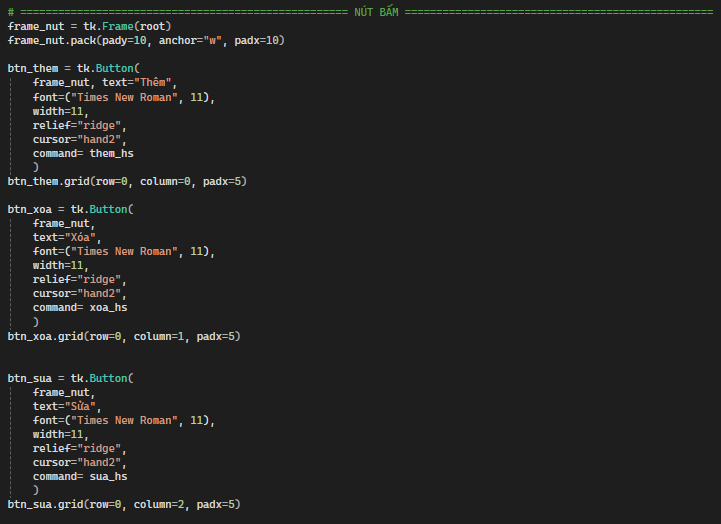


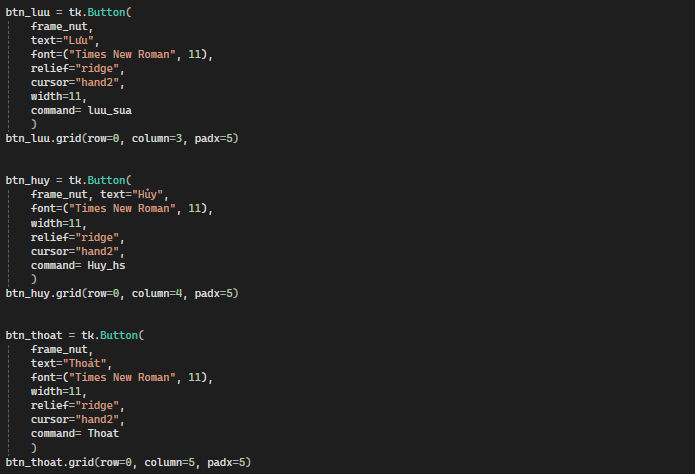


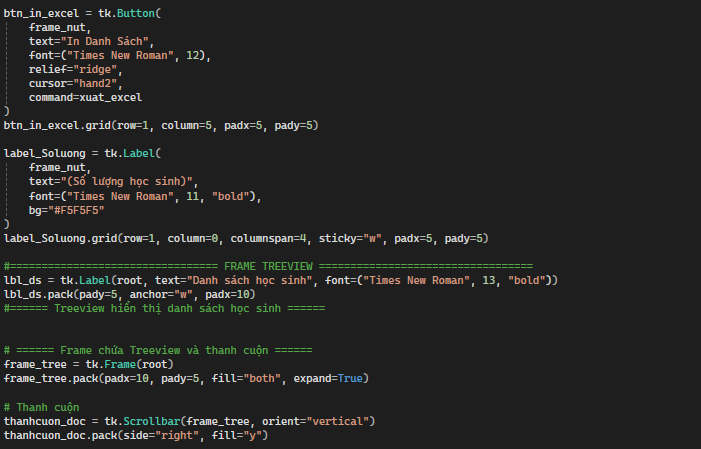


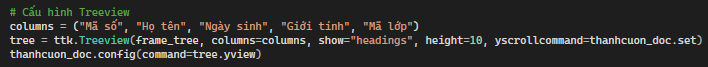


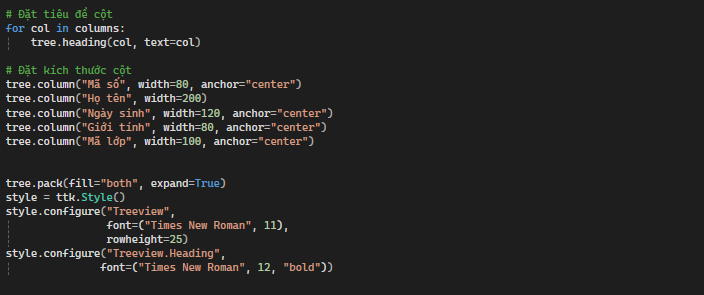


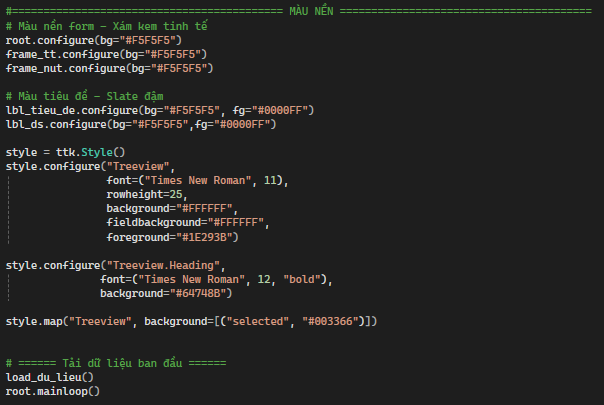




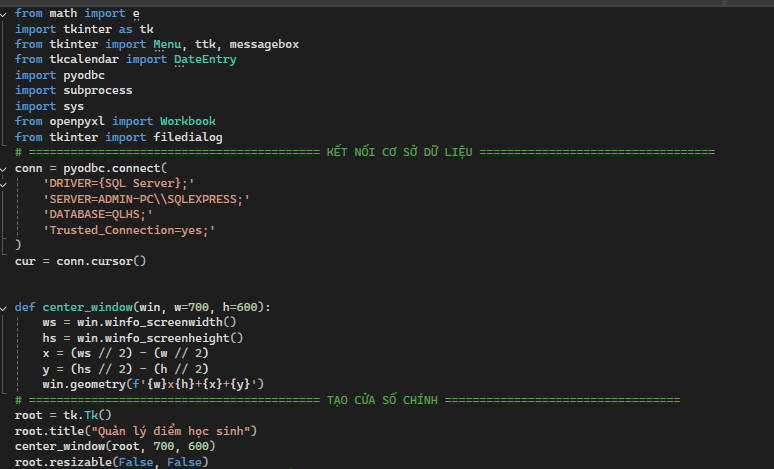


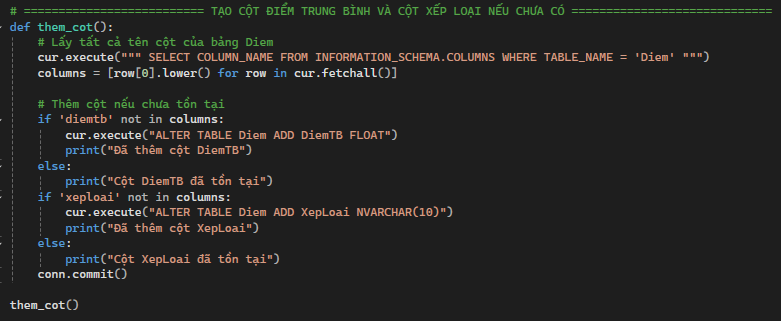


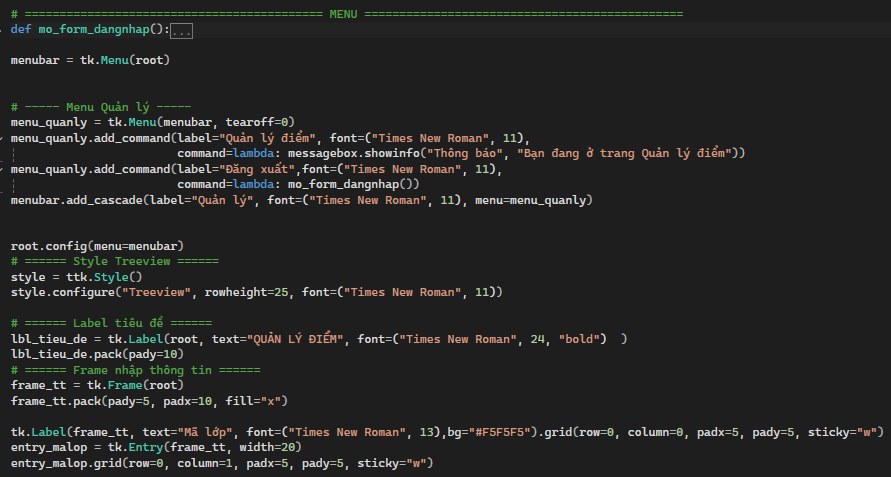


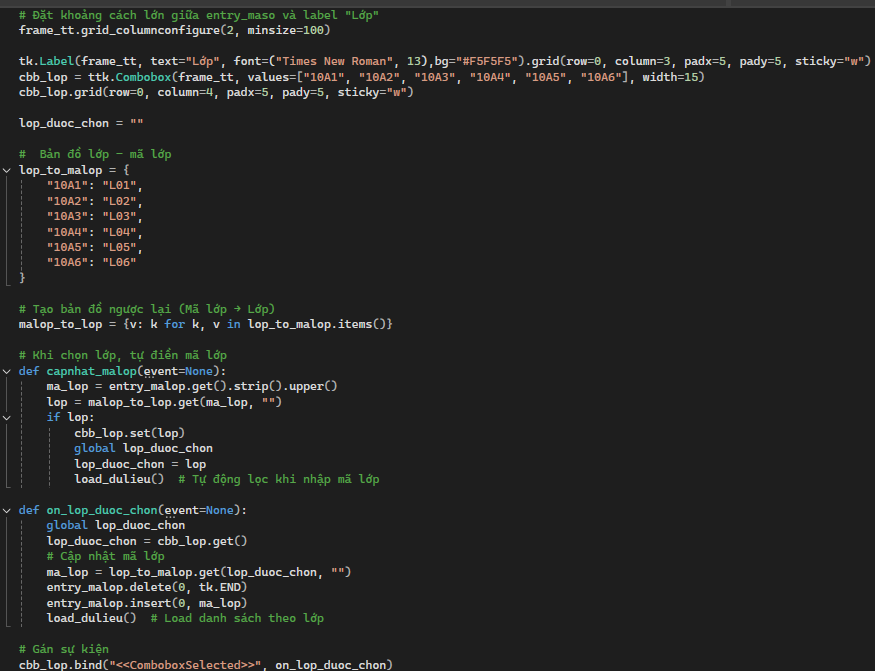


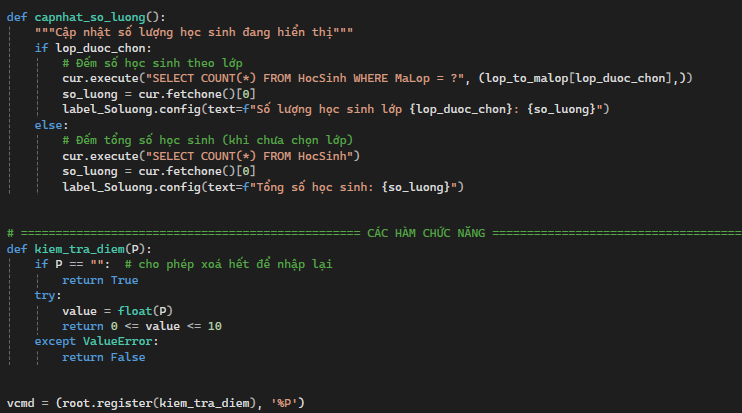
*Form “Quản lý điểm”*



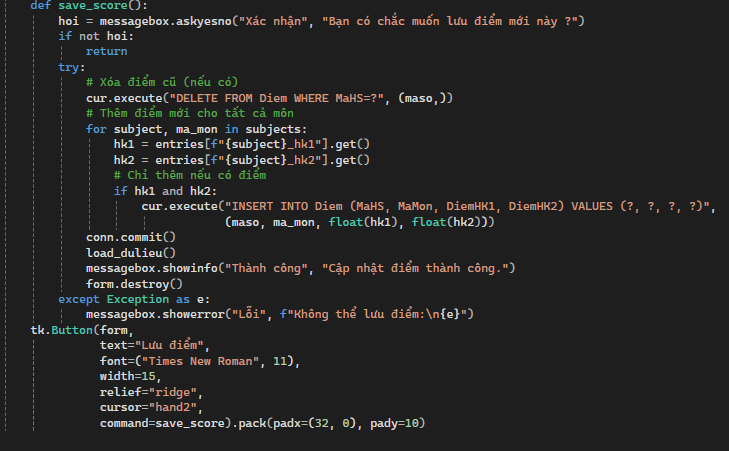


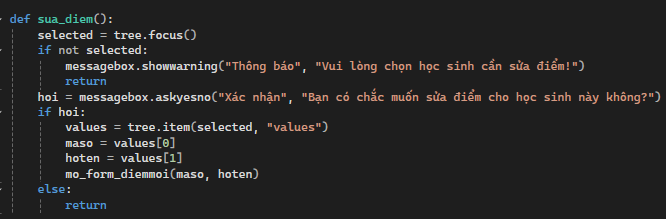


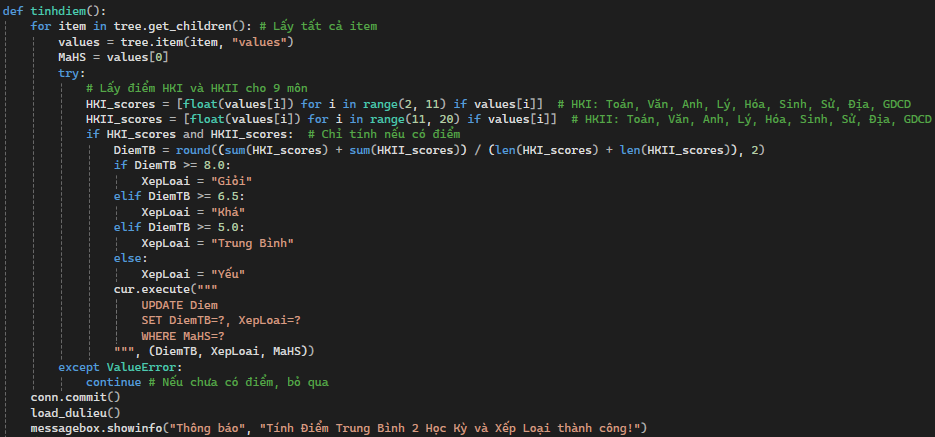


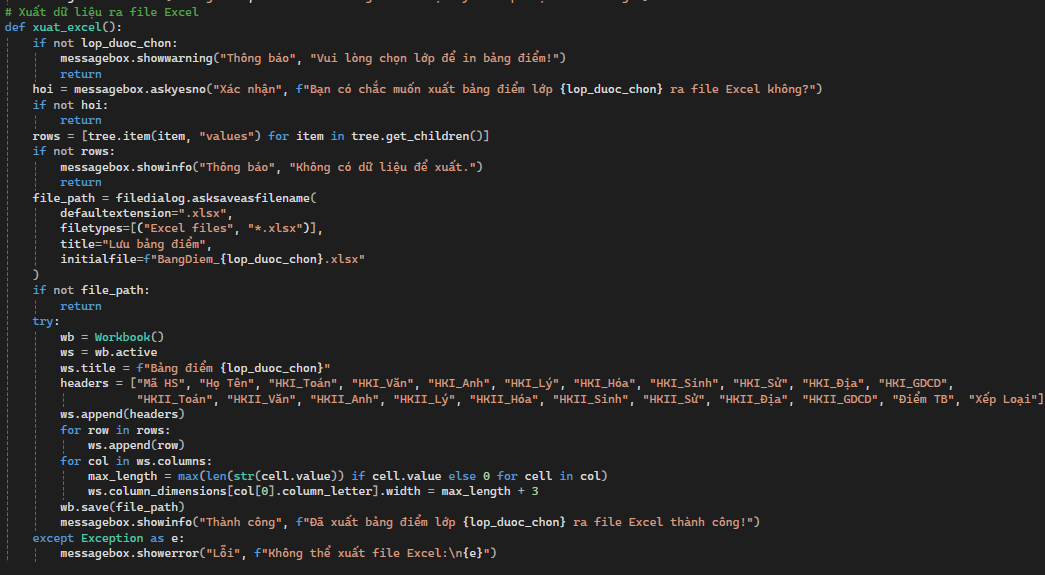


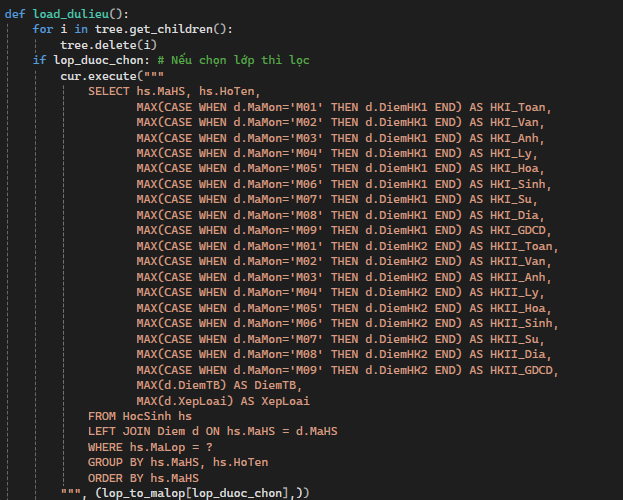


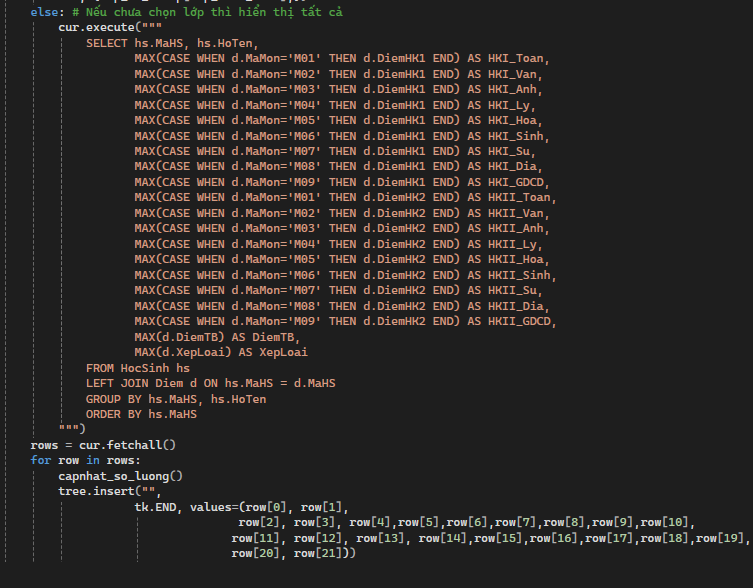


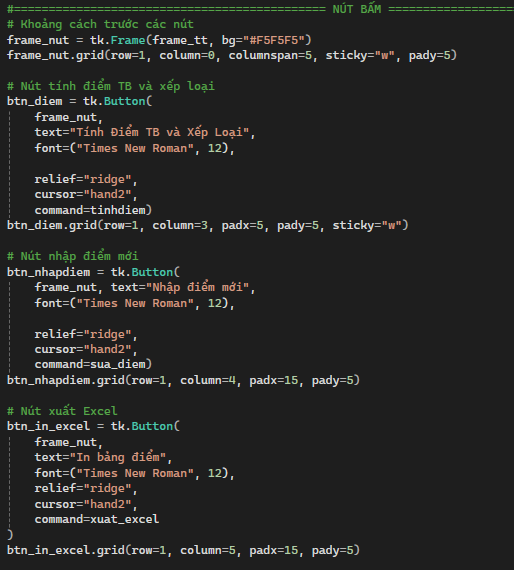




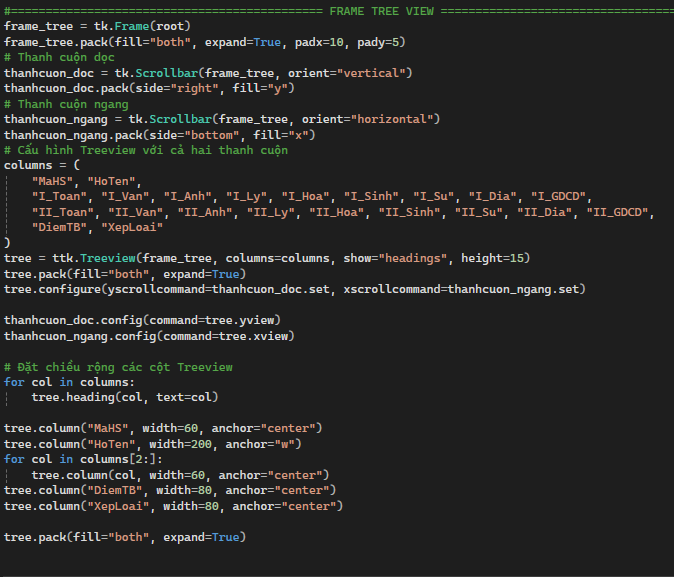


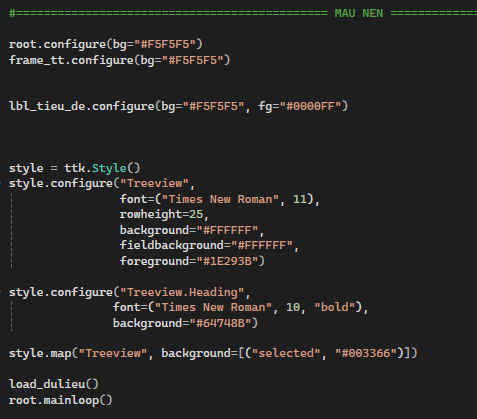












PHẦN CUỐI

Danh mục tài liệu tham

- Đồ án tham khảo của thầy Ths.Nguyễn Ngọc Minh.

- Công cụ hỗ trợ sửa lỗi phát sinh Chat GPT, Grok.