**Bài báo cáo**

**Xây dựng ứng dụng quản lý công việc**

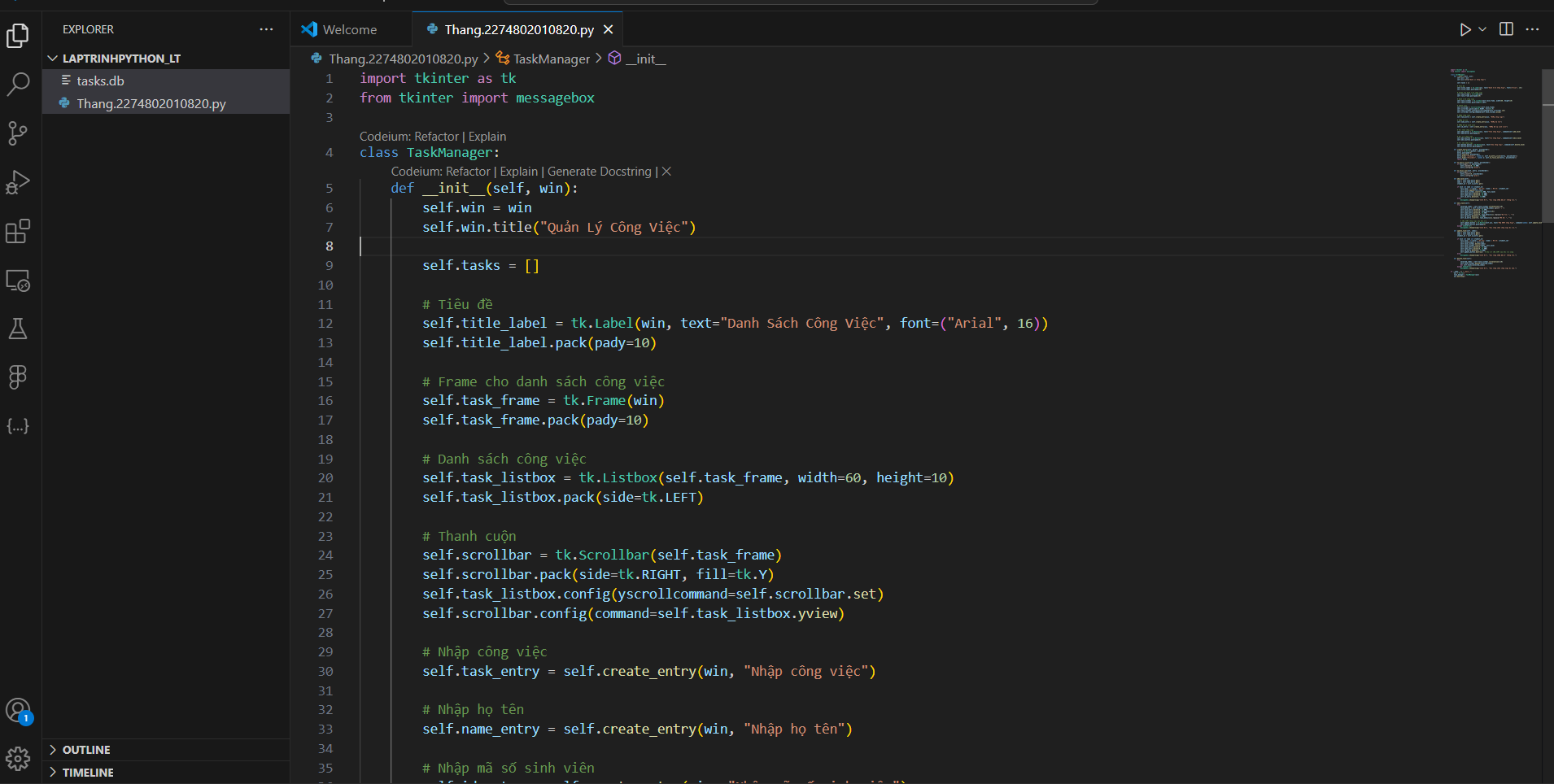
**Mssv: 2274802010820**

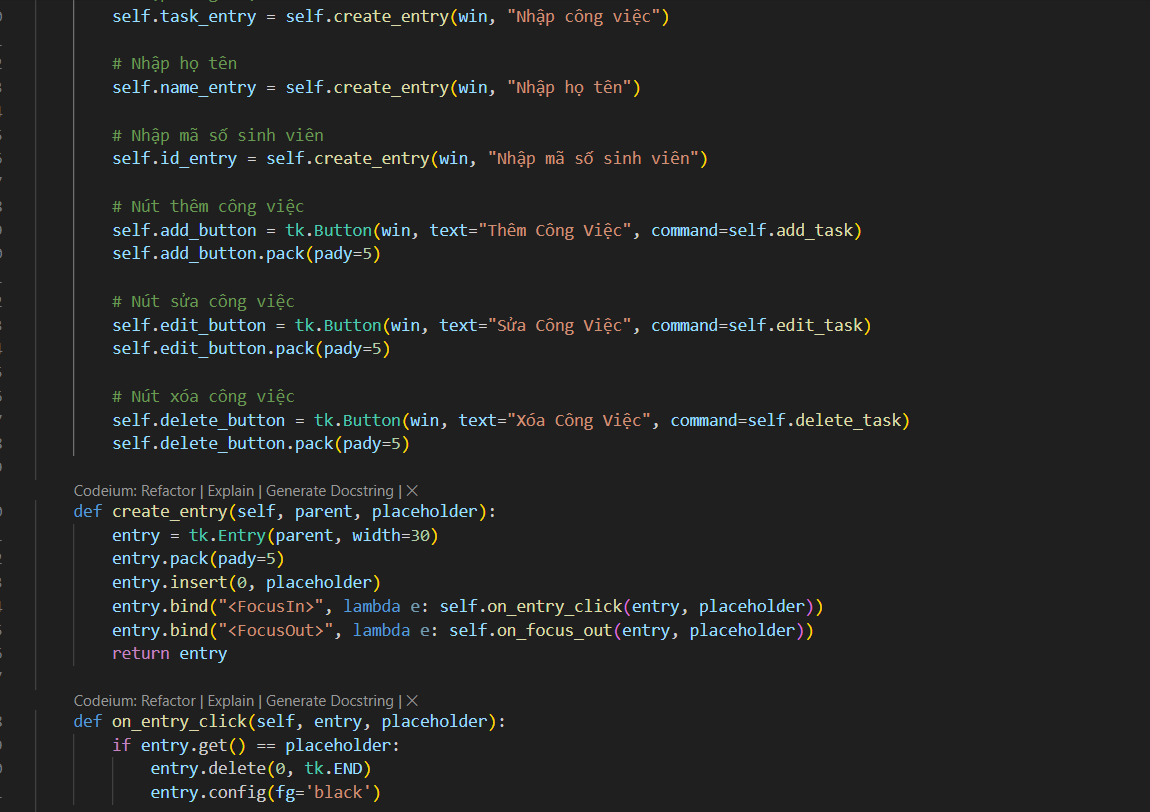
**Họ tên: Dương Thắng**

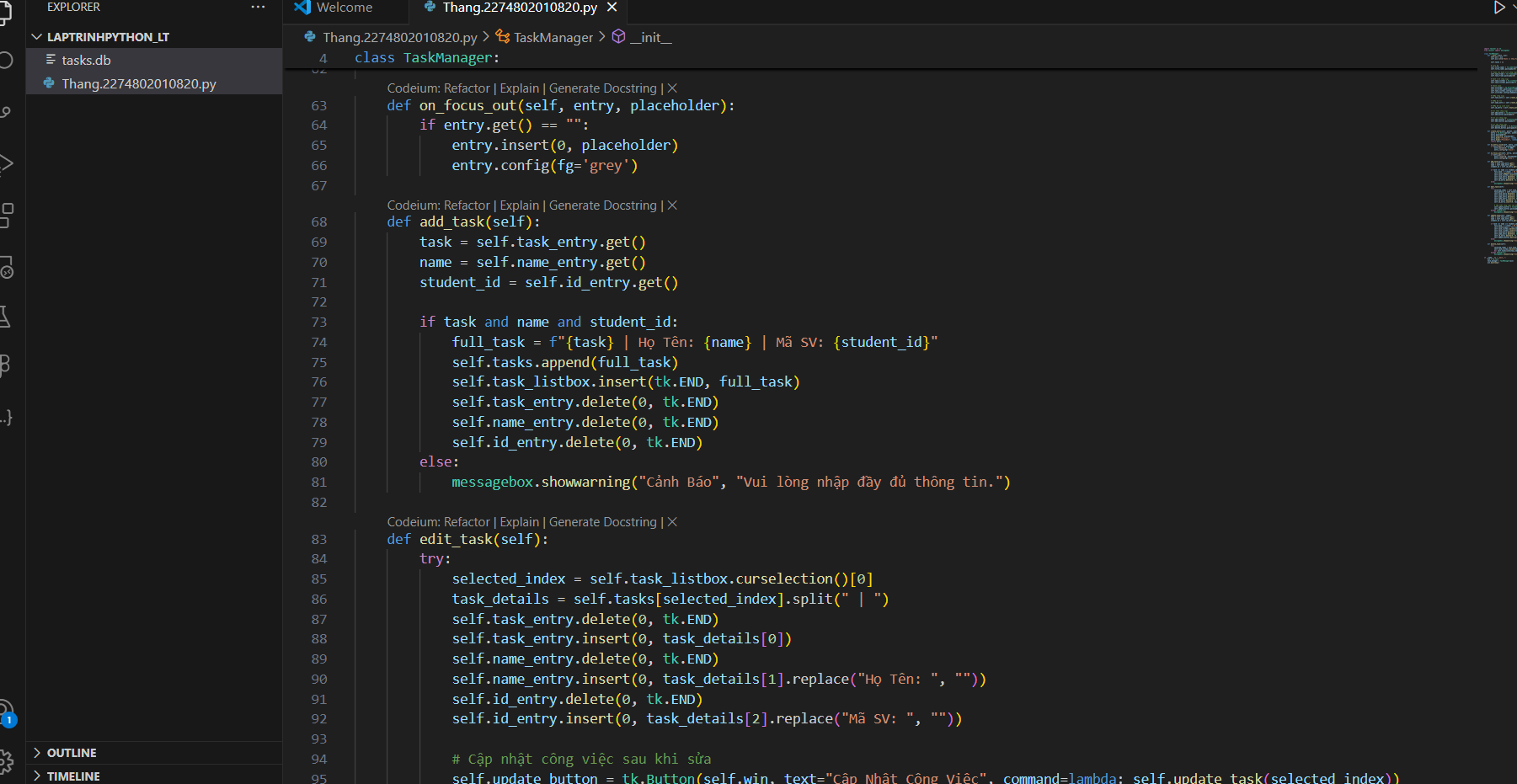
**Nội dung:**

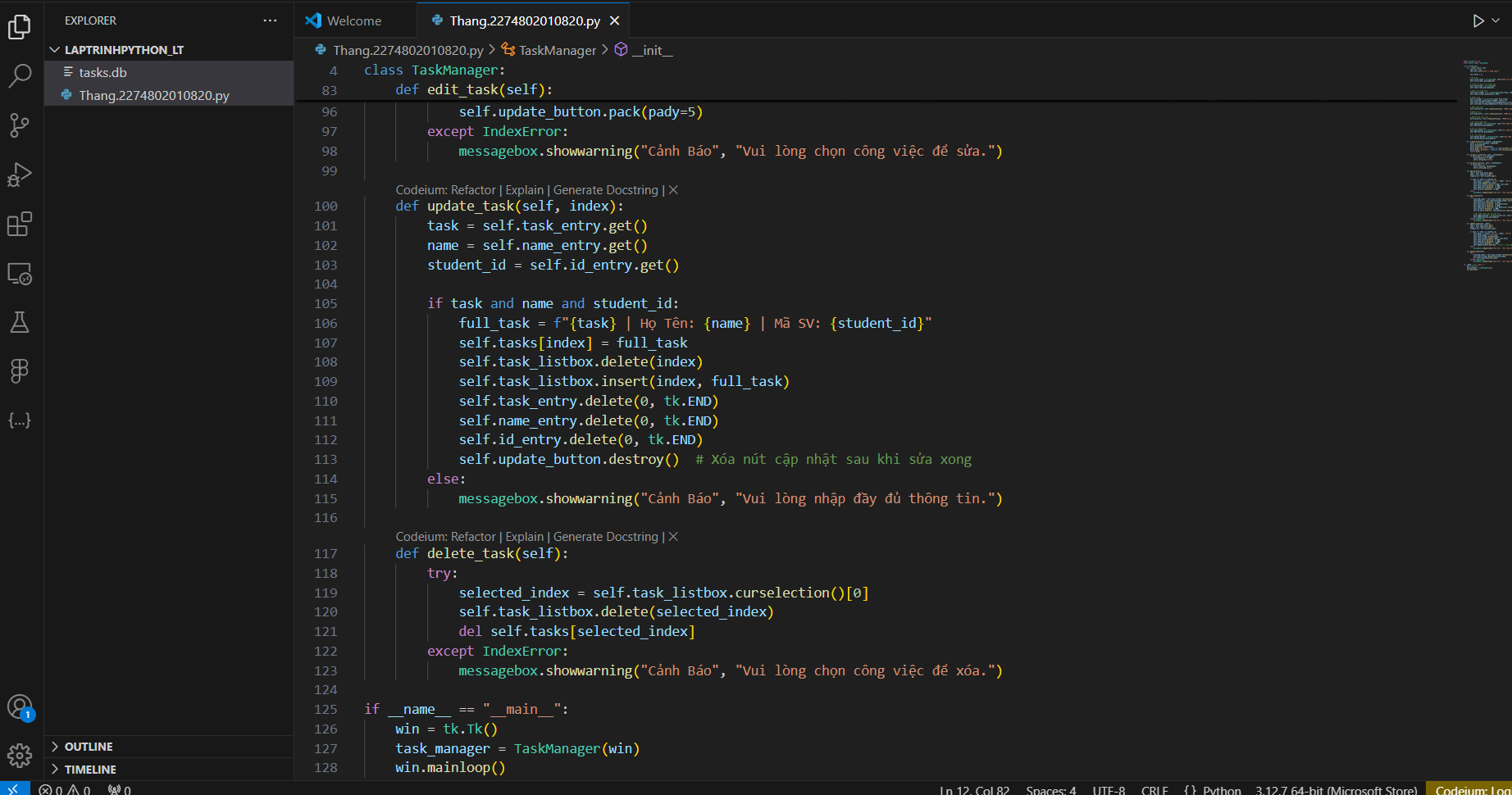
**Giao Diện**

1. **Tiêu Đề**:
   * Ở phần đầu cửa sổ, có một tiêu đề lớn "Danh Sách Công Việc", giúp người dùng dễ dàng nhận biết mục đích của ứng dụng.
2. **Danh Sách Công Việc**:
   * Một khung danh sách (Listbox) hiển thị tất cả các công việc đã được thêm vào. Người dùng có thể cuộn lên/xuống để xem tất cả các công việc.
3. **Thanh Cuộn**:
   * Bên phải danh sách công việc có một thanh cuộn giúp người dùng di chuyển qua lại trong danh sách nếu có nhiều công việc.
4. **Ô Nhập**:
   * Có ba ô nhập:
     + **Nhập công việc**: Để nhập tên công việc.
     + **Nhập họ tên**: Để nhập tên của người thực hiện công việc.
     + **Nhập mã số sinh viên**: Để nhập mã số sinh viên liên quan đến công việc.
5. **Nút Chức Năng**:
   * **Thêm Công Việc**: Nút này cho phép người dùng thêm một công việc mới vào danh sách.
   * **Sửa Công Việc**: Nút này cho phép người dùng sửa thông tin của công việc đã chọn.
   * **Xóa Công Việc**: Nút này cho phép người dùng xóa công việc đã chọn khỏi danh sách.





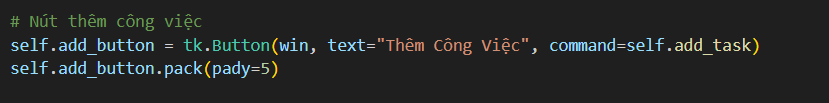




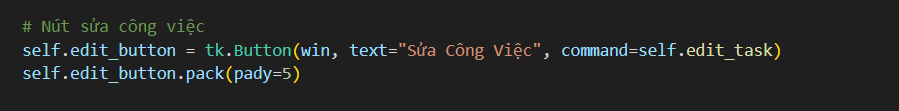
**Chức Năng**

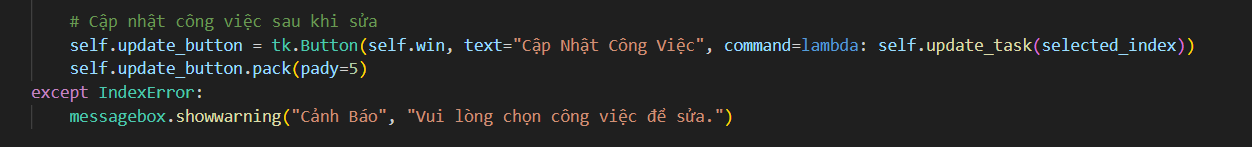
1. **Thêm Công Việc**:
   * Khi người dùng nhập thông tin vào các ô nhập và nhấn nút "Thêm Công Việc", thông tin sẽ được xác nhận (phải đầy đủ) và thêm vào danh sách công việc. Một thông báo sẽ xuất hiện nếu thông tin không đầy đủ.



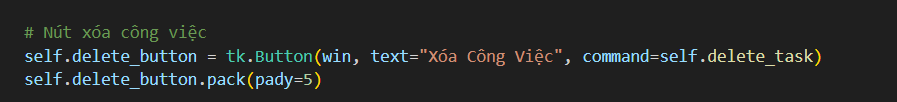


1. **Sửa Công Việc**:
   * Người dùng có thể chọn một công việc trong danh sách và nhấn nút "Sửa Công Việc". Thông tin của công việc sẽ được tải vào các ô nhập, cho phép người dùng chỉnh sửa. Sau khi nhập thông tin mới, người dùng nhấn nút "Cập Nhật Công Việc" để lưu thay đổi.

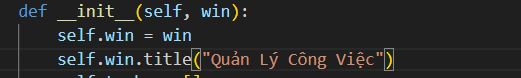




1. **Xóa Công Việc**:
   * Khi người dùng chọn một công việc trong danh sách và nhấn nút "Xóa Công Việc", công việc đó sẽ bị xóa khỏi danh sách. Một thông báo sẽ xuất hiện nếu không có công việc nào được chọn.



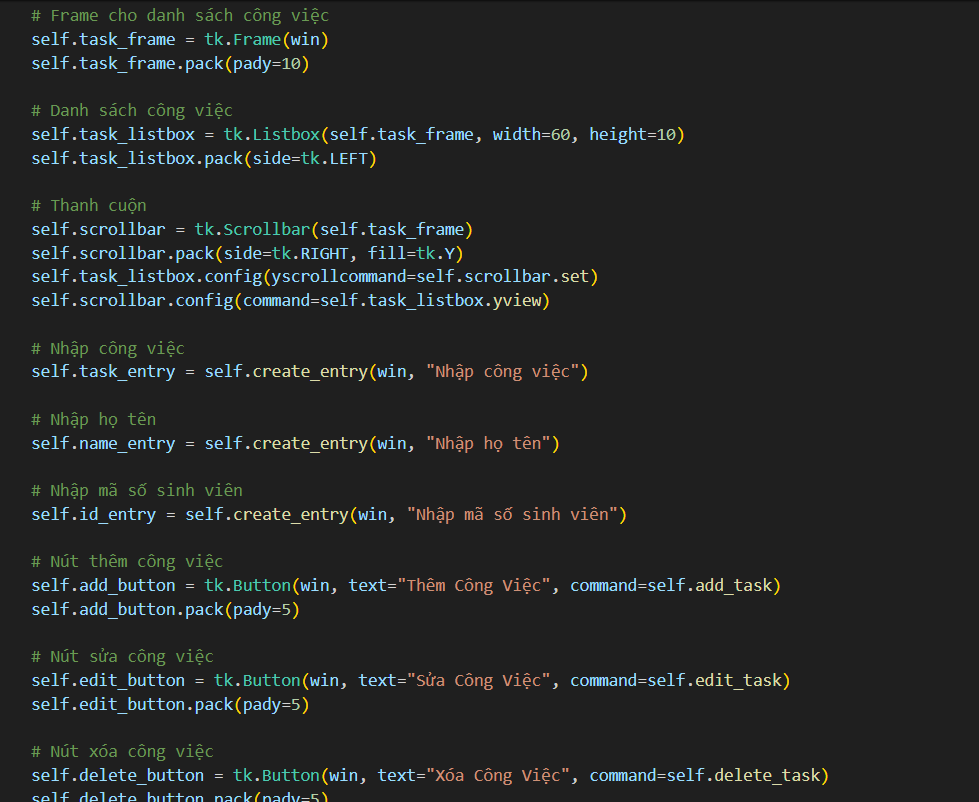
**Các hàm khởi tạo trong mã chương trình và các đoạn mã chi tiết**



**init\_\_**: Đây là phương thức khởi tạo, được gọi khi một đối tượng của lớp TaskManager được tạo.

**self.win**: Biến này lưu trữ tham chiếu đến cửa sổ chính của ứng dụng (đối tượng Tk).

**self.win.title("Quản Lý Công Việc")**: Đặt tiêu đề cho cửa sổ là "Quản Lý Công Việc".



**tk.Frame**: Tạo một frame (khung) để chứa danh sách công việc và thanh cuộn.

**pack(pady=10)**: Đặt frame vào cửa sổ với khoảng cách dọc 10 pixel.

**tk.Listbox**: Tạo một danh sách có thể chọn để hiển thị các công việc.

**width=60** và **height=10**: Đặt chiều rộng và chiều cao cho danh sách.

**pack(side=tk.LEFT)**: Đặt danh sách bên trái trong frame.

**tk.Scrollbar**: Tạo thanh cuộn để cuộn qua danh sách.

**pack(side=tk.RIGHT, fill=tk.Y)**: Đặt thanh cuộn bên phải và cho phép nó mở rộng theo chiều dọc.

**self.task\_listbox.config(yscrollcommand=self.scrollbar.set)**: Kết nối thanh cuộn với danh sách để khi cuộn, danh sách cũng di chuyển.

**self.scrollbar.config(command=self.task\_listbox.yview)**: Kết nối thanh cuộn với chức năng cuộn của danh sách.

**elf.create\_entry**: Gọi phương thức create\_entry để tạo các ô nhập cho việc nhập công việc, họ tên và mã số sinh viên.

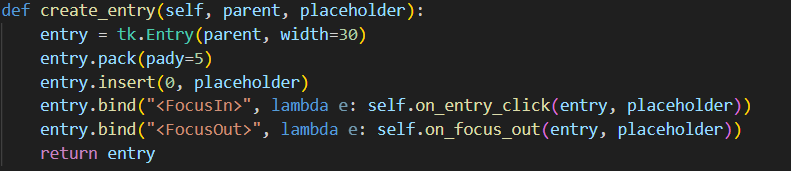
Mỗi ô nhập có một placeholder để hướng dẫn người dùng về thông tin cần nhập.

**tk.Button**: Tạo các nút để thực hiện các chức năng như thêm, sửa và xóa công việc.

**command=self.add\_task**: Liên kết nút với các phương thức xử lý tương ứng (add\_task, edit\_task, delete\_task).

**pack(pady=5)**: Đặt các nút vào cửa sổ với khoảng cách dọc 5 pixel.

* **Phương thức create\_entry**



parent: Cửa sổ hoặc frame chứa ô nhập.

placeholder: Văn bản hiển thị trong ô nhập khi chưa có dữ liệu.

**Tạo ô nhập**: tk.Entry để tạo một ô nhập với chiều rộng 30.

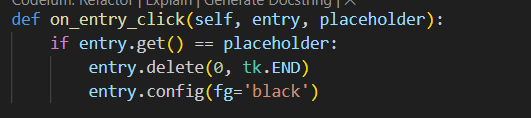
**Đặt vào cửa sổ**: pack(pady=5) để căn chỉnh ô nhập với khoảng cách dọc.

**Thêm placeholder**: insert(0, placeholder) để hiển thị văn bản hướng dẫn.

Khi nhấp vào ô nhập (<FocusIn>), gọi on\_entry\_click để xóa placeholder.

Khi rời khỏi ô nhập (<FocusOut>), gọi on\_focus\_out để kiểm tra và có thể khôi phục placeholder nếu ô nhập rỗng.

* **Phương thức on\_entry\_click**



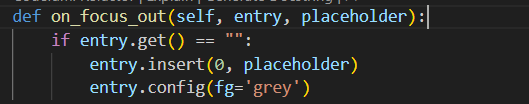
Xử lý sự kiện khi người dùng nhấp vào ô nhập.

entry: Ô nhập đang được nhấp vào.

placeholder: Văn bản hướng dẫn cho ô nhập.

Xóa nội dung (placeholder) và thay đổi màu chữ thành đen (fg='black'), cho phép người dùng nhập văn bản mới.

* **Phương thức on\_focus\_out**



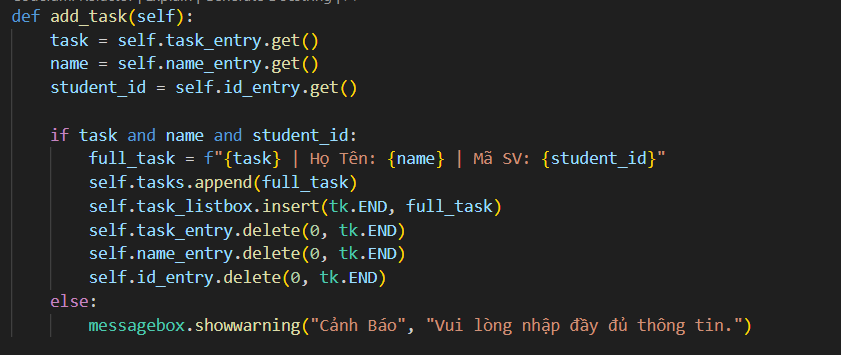
Xử lý sự kiện khi người dùng rời khỏi ô nhập.

entry: Ô nhập mà người dùng đã rời khỏi.

placeholder: Văn bản hướng dẫn cho ô nhập.

Khôi phục placeholder vào ô và thay đổi màu chữ thành xám (fg='grey'), để người dùng biết rằng ô này cần thông tin.

* **Phương thức add\_task**



task: Lấy giá trị từ ô nhập công việc (task\_entry).

name: Lấy giá trị từ ô nhập họ tên (name\_entry).

student\_id: Lấy giá trị từ ô nhập mã số sinh viên (id\_entry).

Câu lệnh if kiểm tra xem tất cả các trường đều có dữ liệu hay không (không rỗng).

Nếu tất cả đều có dữ liệu, chương trình sẽ tiếp tục thêm công việc.

Tạo một chuỗi full\_task kết hợp thông tin từ ba trường nhập vào thành một định dạng rõ ràng:

Ví dụ: "Công việc | Họ Tên: Nguyễn Văn A | Mã SV: 123456".

Thêm full\_task vào danh sách self.tasks, nơi lưu trữ tất cả các công việc.

Cập nhật danh sách hiển thị (task\_listbox) bằng cách thêm full\_task vào cuối danh sách (tk.END).

Xóa nội dung của các ô nhập sau khi thêm công việc thành công, để người dùng có thể nhập công việc mới mà không cần xóa thủ công.

Nếu một hoặc nhiều trường không có dữ liệu, hiển thị một hộp thoại cảnh báo cho người dùng, yêu cầu họ nhập đầy đủ thông tin.

* **Phương thức edit task**



Sử dụng self.task\_listbox.curselection() để lấy chỉ số của công việc đang được chọn trong danh sách.

Selected\_index sẽ chứa chỉ số đầu tiên (vì người dùng chỉ có thể chọn một công việc trong danh sách).

Lấy thông tin công việc tương ứng từ danh sách self.tasks và tách nó thành các phần riêng biệt bằng cách sử dụng phương thức split(" | ").

Task\_details sẽ chứa một danh sách với các phần: tên công việc, họ tên, và mã số sinh viên.

Xóa nội dung hiện tại của ô nhập công việc và chèn tên công việc đã được chọn.(insert)

Tương tự như trên, xóa nội dung của ô nhập họ tên và mã số sinh viên, sau đó chèn thông tin tương ứng đã tách được từ task\_details.

Sử dụng replace để loại bỏ các phần không cần thiết ("Họ Tên: " và "Mã SV: ") trước khi chèn vào ô.

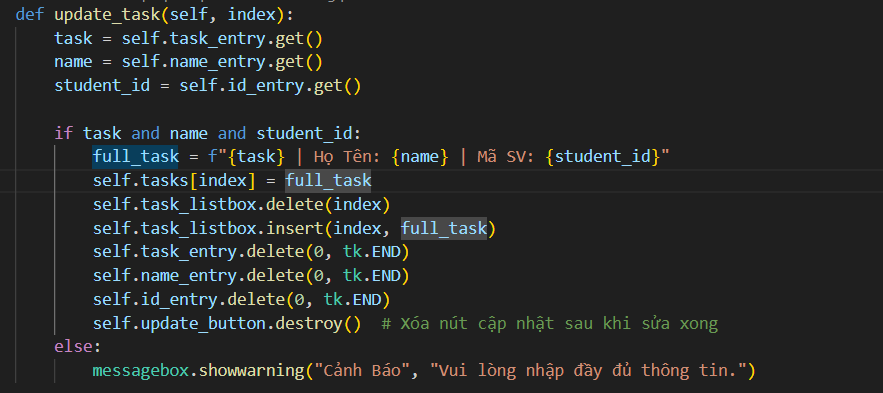
Tạo một nút mới có nhãn "Cập Nhật Công Việc" để người dùng có thể lưu các thay đổi.

Nút này được liên kết với phương thức update\_task, sẽ được gọi khi người dùng nhấn nút. Chỉ số của công việc được chọn (selected\_index) được truyền vào để xác định công việc nào cần được cập nhật.

Nếu không có công việc nào được chọn (người dùng chưa chọn gì trong danh sách), sẽ phát sinh lỗi IndexError.

Trong trường hợp này, một hộp thoại cảnh báo sẽ hiển thị thông báo yêu cầu người dùng chọn công việc trước khi sửa.

* **Phương thức update task**



Lấy giá trị từ các ô nhập:

* Task: Tên công việc.
* Name: Họ tên.
* Student\_id: Mã số sinh viên.

Câu lệnh if kiểm tra xem tất cả các trường có dữ liệu hay không (không rỗng).

Nếu tất cả đều có dữ liệu, chương trình tiếp tục cập nhật công việc

Xóa công việc cũ trong danh sách hiển thị (task\_listbox) tại vị trí index.

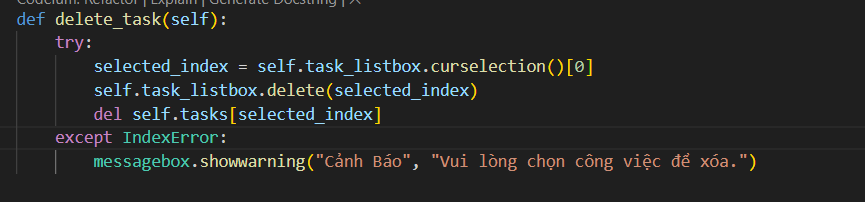
Chèn full\_task mới vào danh sách ở cùng vị trí.

Xóa nội dung của các ô nhập sau khi cập nhật thành công, để người dùng có thể nhập dữ liệu mới mà không cần xóa thủ công.

Hủy bỏ nút "Cập Nhật Công Việc" sau khi hoàn thành việc cập nhật

Nếu một hoặc nhiều trường không có dữ liệu, hiển thị một hộp thoại cảnh báo cho người dùng, yêu cầu họ nhập đầy đủ thông tin.

* Phương thức delete task



Sử dụng self.task\_listbox.curselection() để lấy chỉ số của công việc đang được chọn trong danh sách.

selected\_index sẽ chứa chỉ số đầu tiên (vì người dùng chỉ có thể chọn một công việc trong danh sách).

Gọi phương thức delete(selected\_index) trên task\_listbox để xóa công việc đã chọn khỏi danh sách hiển thị.

Sử dụng del để xóa công việc tương ứng trong danh sách self.tasks tại vị trí selected\_index.

Nếu không có công việc nào được chọn (người dùng chưa chọn gì trong danh sách), sẽ phát sinh lỗi IndexError.

Trong trường hợp này, hiển thị một hộp thoại cảnh báo với thông báo yêu cầu người dùng chọn công việc trước khi thực hiện xóa.

<https://github.com/Thang204/Python_Nang_Cao>