**TRƯỜNG ĐẠI HỌC AN GIANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO ĐỒ ÁN**

***Chuyên đề Python(COS525)***

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ CỬA HÀNG TIVI VỚI PYTHON ,TKINTER VÀ MYSQL**

Người hướng dẫn :

Ths. Nguyễn Ngọc Minh

Sinh Viên Thực Hiện:

DTH235771.Nguyễn Ngọc Thảo .DH24TH3.Nhóm3\_Tổ1

DTH235797.Nguyễn Văn Trọng.DH24TH3.Nhóm 3\_Tổ 1

**An Giang,11-2025**

1. **Đặt vấn đề**

***Tính cần thiết của đề tài*** :

**Hạn chế của các đề tài đã có:**

Các mô hình quản lý bán hàng trước đây chủ yếu phục vụ bán hàng tổng quát, chưa phù hợp với đặc thù của mặt hàng Tivi có nhiều hãng, model và thông số kỹ thuật phức tạp. Phần lớn chưa tích hợp quản lý bảo hành, chưa theo dõi được thông tin kỹ thuật chi tiết và không hỗ trợ so sánh model. Một số đề tài còn hạn chế về giao diện, khó sử dụng và chưa cập nhật theo công nghệ mới, nên chưa đáp ứng tốt nhu cầu thực tế của cửa hàng Tivi.

### ****Vấn đề các đề tài trước chưa đề cập:****

- Tự động hóa quá trình cập nhật tồn kho theo nhập – xuất.

- Quản lý chi tiết từng Tivi theo **mã serial**, tránh thất thoát.

- Theo dõi doanh thu từng dòng sản phẩm, từng hãng.

- Gợi ý nhập hàng theo số liệu bán ra.

- Tích hợp quản lý khách hàng – bảo hành – đổi trả.

### ****Sự cần thiết của đề tài:****

Việc quản lý thủ công dễ dẫn đến sai lệch dữ liệu và khó kiểm soát hàng hóa, đặc biệt với mặt hàng Tivi có giá trị cao. Một hệ thống quản lý chuyên biệt giúp cửa hàng giảm thất thoát, nâng cao hiệu quả xử lý công việc và phù hợp xu hướng số hóa trong kinh doanh bán lẻ hiện nay.

### ****Ý nghĩa lý luận và thực tiễn:****

**Ý nghĩa lý luận**

Đề tài góp phần vận dụng các kiến thức về mô hình hóa dữ liệu, thiết kế hệ thống và lập trình để xây dựng một mô hình quản lý phù hợp với thực tế.

**Ý nghĩa thực tiễn:**

- Giúp cửa hàng theo dõi số lượng Tivi, loại hàng, hãng sản xuất, giá cả, doanh thu chính xác.

- Hỗ trợ ra quyết định: nhập hàng, thanh lý, chính sách giá.

- Tăng hiệu quả quản lý thông qua giao diện trực quan và chức năng tra cứu nhanh chóng.

### ****Cơ sở để cụ thể hóa mục tiêu và định hướng nội dung chính:****

- Dựa trên nhu cầu thực tế của cửa hàng Tivi.

- Dựa trên hạn chế của các đề tài tương tự.

- Áp dụng các kiến thức môn học về cơ sở dữ liệu, thiết kế hệ thống, lập trình ứng dụng.

**Nội dung chính của đề tài sẽ tập trung vào:**

- Phân tích yêu cầu hệ thống quản lý cửa hàng Tivi.

- Thiết kế cơ sở dữ liệu (sản phẩm, hãng sản xuất, hóa đơn, khách hàng…).

- Thiết kế các chức năng chính: nhập hàng, xuất hàng, quản lý tồn kho, thống kê doanh thu.

- Xây dựng giao diện phần mềm vả kiểm thử và đánh giá hệ thống

### Mục Tiêu Nghiên Cứu của Đề Tài

Mục tiêu của đề tài là xây dựng một hệ thống phần mềm quản lý cửa hàng Tivi sử dụng Python và MySQL, đạt được các kết quả cụ thể sau:

**-Thiết kế CSDL:** Thiết kế và triển khai một Cơ sở dữ liệu quan hệ (gồm các bảng loai, sanpham, nhanvien, khachhang, hoadon, cthd) để lưu trữ dữ liệu một cách nhất quán và hiệu quả. (Định lượng: 6 bảng, Quan hệ 1-nhiều rõ ràng).

**- Xây dựng Giao diện:** Phát triển giao diện người dùng (Tkinter) thân thiện, trực quan, cho phép người dùng thực hiện đầy đủ các chức năng **Thêm, Sửa, Xóa (CRUD)** đối với các danh mục chính (Sản phẩm, Khách hàng, Loại). (Định tính: Thân thiện, đầy đủ CRUD).

**- Xử lý Nghiệp vụ Bán hàng:** Xây dựng module tạo hóa đơn bán hàng, bao gồm khả năng **tự động tính toán tổng tiền**, **cập nhật số lượng tồn kho** sau mỗi giao dịch, và hiển thị chi tiết hóa đơn. (Khả thi: Tự động hóa).

**- Phân quyền Cơ bản:** Triển khai chức năng **Đăng nhập** và **Phân quyền** cơ bản (Admin có quyền quản lý Nhân viên, Nhân viên thường chỉ có quyền bán hàng/quản lý danh mục). (Khả thi).

.**2. Tổng quan và cơ sở lý thuyết:**

### 1. Đặt Vấn Đề

Bán lẻ Tivi là một ngành hàng có giá trị cao, mẫu mã đa dạng và cạnh tranh lớn. Việc quản lý thông tin sản phẩm (mã, giá, tồn kho) và các giao dịch bán hàng (hóa đơn, chi tiết hóa đơn) đòi hỏi sự chính xác tuyệt đối. Hệ thống quản lý được đề xuất nhằm cung cấp công cụ tự động hóa, giảm thiểu sai sót, và nâng cao hiệu suất làm việc của nhân viên và khả năng ra quyết định của quản lý.

### 2. Lịch sử Giải quyết Vấn Đề

**- Giải pháp truyền thống:** Nhiều cửa hàng nhỏ dùng Excel/sổ sách. Hạn chế: Dễ mất dữ liệu, khó kiểm soát đồng thời, không thể tạo báo cáo phức tạp.

**- Giải pháp phần mềm:** Các phần mềm POS và quản lý kho tổng quát (ví dụ: Sapo, KiotViet) đã giải quyết vấn đề chung, nhưng chi phí cao và thường đi kèm với các tính năng không cần thiết, phức tạp hóa việc sử dụng.

**- Tính mới của đề tài:** Đề tài này giải quyết khoảng trống giữa giải pháp thủ công và phần mềm chuyên nghiệp đắt tiền, bằng cách xây dựng một hệ thống tùy chỉnh, tập trung vào nghiệp vụ cốt lõi của cửa hàng Tivi, sử dụng **công nghệ mã nguồn mở (Python/MySQL)** để đảm bảo tính linh hoạt và chi phí thấp.

### 3. Phạm Vi của Đề Tài

Đề tài tập trung giải quyết các vấn đề sau:

**-Phạm vi nghiệp vụ:** Quản lý danh mục (loai, sanpham, khachhang, nhanvien) và Quản lý giao dịch bán hàng (hoadon, cthd).

**- Phạm vi công nghệ:** Sử dụng **Python 3** cho logic lập trình và giao diện, **Tkinter** cho giao diện đồ họa, và **MySQL Server** cho hệ quản trị cơ sở dữ liệu.

**- Mức độ:** Chỉ quản lý tồn kho theo số lượng tổng (không đi sâu vào quản lý kho nhập/xuất chi tiết hoặc bảo hành theo Serial Number phức tạp).

### 4. Phương Pháp Nghiên Cứu và Hướng Giải Quyết

**Phương pháp nghiên cứu**

**- Nghiên cứu lý thuyết:** Tổng hợp kiến thức về thiết kế CSDL quan hệ (Mô hình thực thể - ERD), nguyên lý lập trình hướng đối tượng (OOP) và kiến trúc phân lớp (3-layer Architecture).

**- Nghiên cứu thực nghiệm:** Thực hiện xây dựng phần mềm theo mô hình **Phát triển Tăng trưởng (Incremental Model)**, tức là xây dựng từng module nhỏ (ví dụ: Quản lý Loại -> Quản lý Sản phẩm -> Quản lý Khách hàng -> Quản lý Hóa đơn) và kiểm thử liên tục.

**Hướng giải quyết:** Xây dựng phần mềm theo kiến trúc phân lớp:

**- Tầng Dữ liệu (database.py):** Chịu trách nhiệm kết nối, truy vấn (CRUD) với MySQL.

**- Tầng Logic/Mô hình (models.py):** Chứa các hàm nghiệp vụ, gọi các hàm từ tầng dữ liệu.

**- Tầng Giao diện (hoadon.py, sanpham.py, v.v.):** Chứa code Tkinter, hiển thị dữ liệu và nhận lệnh từ người dùng.

***Tóm tắt:***

|  |  |
| --- | --- |
| **Cơ sở Lí thuyết** | **Ứng dụng trong Đề tài** |
| **Lập trình Python** | Ngôn ngữ chính để phát triển ứng dụng, tận dụng tính dễ đọc và thư viện phong phú. |
| **Tkinter** | Thư viện giao diện chuẩn, được sử dụng để xây dựng giao diện người dùng đơn giản, nhanh chóng và hoạt động độc lập (standalone). |
| **MySQL** | Hệ quản trị CSDL quan hệ (RDBMS) mã nguồn mở, được sử dụng để lưu trữ tập trung và quản lý các quan hệ dữ liệu. |
| **Nguyên tắc OOP** | Áp dụng trong thiết kế các lớp (ví dụ: các hàm trong models.py hoặc các đối tượng giao diện Tkinter) để tăng khả năng tái sử dụng và dễ bảo trì. |
| **SQL**(Structured Query Language) | Ngôn ngữ được sử dụng để giao tiếp với MySQL (lệnh SELECT, INSERT, UPDATE, DELETE) trong file database.py. |

## 3. Kết Quả Đạt Được và Demo Code

Đề tài đã hoàn thành việc xây dựng ứng dụng quản lý cửa hàng Tivi, vận dụng thành công ngôn ngữ lập trình Python, thư viện giao diện Tkinter và hệ quản trị CSDL MySQL để giải quyết các vấn đề nghiệp vụ thực tế.

### 1. Phân Tích và Xây Dựng Ứng Dụng

Ứng dụng được xây dựng theo kiến trúc phân lớp cơ bản (Data Layer, Logic Layer, Presentation Layer) để đảm bảo tính dễ bảo trì và mở rộng.

#### Vận dụng Kỹ thuật Lập trình Python

**- Lập trình Hướng đối tượng (OOP):** Các file module được thiết kế để tách biệt chức năng (ví dụ: sanpham.py quản lý giao diện sản phẩm, models.py chứa các hàm nghiệp vụ, database.py xử lý kết nối CSDL).

**- Xử lý Ngoại lệ (Error Handling):** Sử dụng khối try...except (đặc biệt trong database.py) để xử lý các lỗi kết nối CSDL (mysql.connector.Error) và các lỗi nhập liệu, giúp chương trình không bị dừng đột ngột.

**- Quản lý CSDL:** Sử dụng thư viện **mysql-connector-python** để tạo các hàm truy vấn và thực thi dữ liệu (fetch\_all, execute\_query) một cách bảo mật thông qua tham số hóa truy vấn (sử dụng %s).

#### Chức năng Cơ bản và Công cụ Tkinter

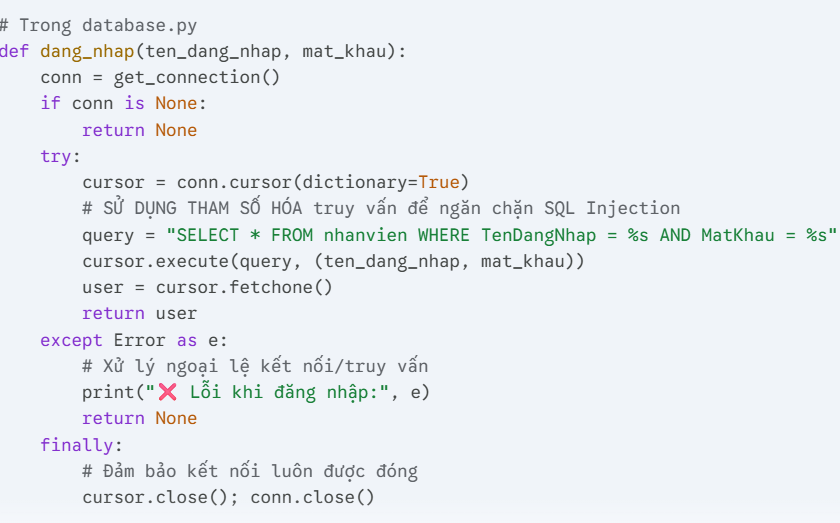
Ứng dụng sử dụng Tkinter để xây dựng giao diện người dùng Desktop (GUI), với các widget cơ bản và nâng cao:

| **Chức năng Cơ bản** | **File Module Chính** | **Công cụ Tkinter Ứng dụng** |
| --- | --- | --- |
| **Đăng nhập/Phân quyền** | login\_view.py | Tk(), Entry (có show='\*' cho mật khẩu), messagebox. |
| **Quản lý Danh mục (CRUD)** | sanpham.py, loai.py, khachhang.py | Toplevel(), LabelFrame, Entry, Button, **ttk.Treeview** (hiển thị dữ liệu dạng bảng), **ttk.Combobox**. |
| **Quản lý Giao dịch (Hóa đơn)** | hoadon.py | DateEntry (từ tkcalendar cho ngày), ttk.Treeview (hiển thị chi tiết HĐ). |
| **Menu Chính** | main.py | Frame, Button (gọi các module con). |

### *Demo Code (Minh họa Kỹ thuật)*

Phần demo tập trung vào các đoạn mã minh họa việc vận dụng Python để giải quyết các vấn đề thực tế:

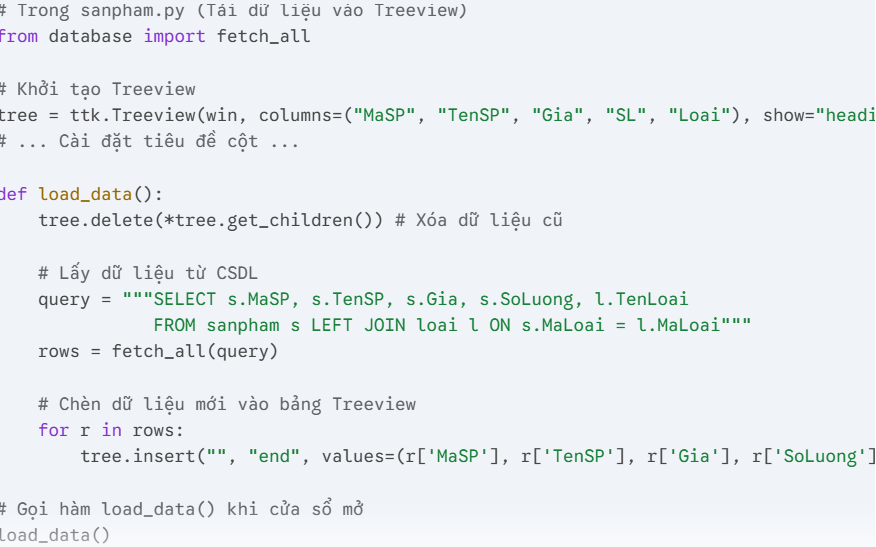
#### 1. Kết nối và Đăng nhập An toàn (database.py & login\_view.py)



#### Vận hành CRUD và Tkinter (loai.py hoặc sanpham.py)



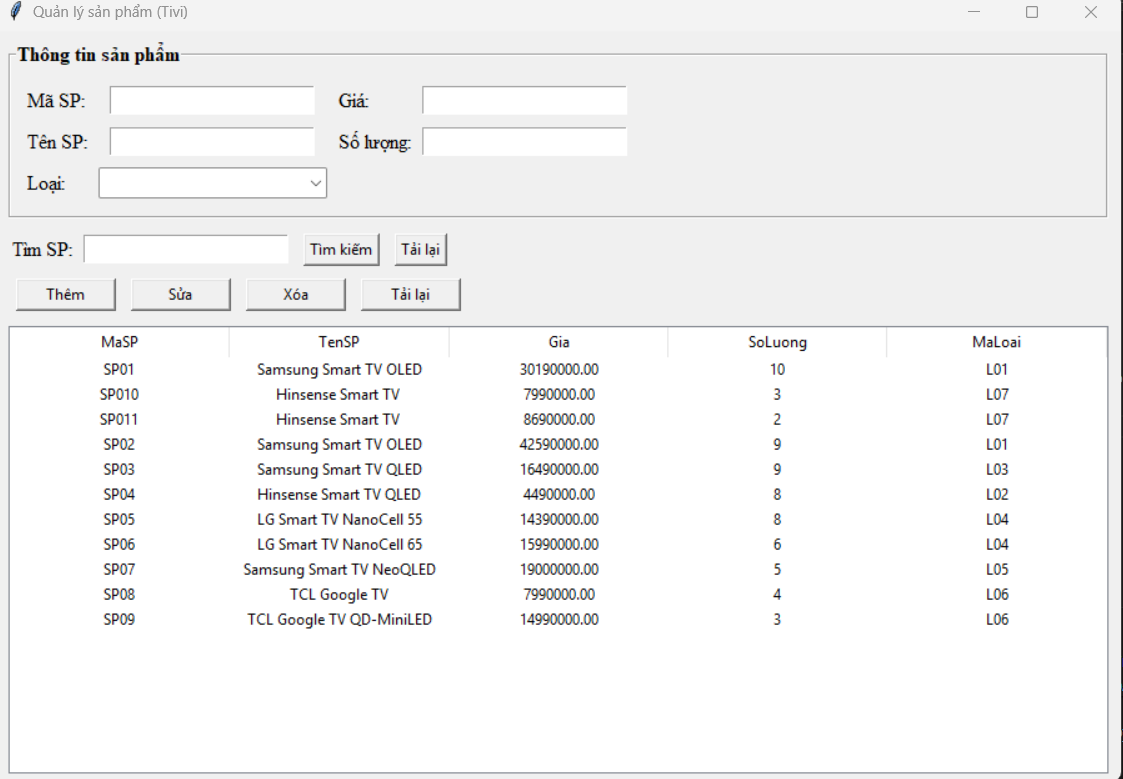
#### 3. Hiển thị Dữ liệu Dạng Bảng (ttk.Treeview)

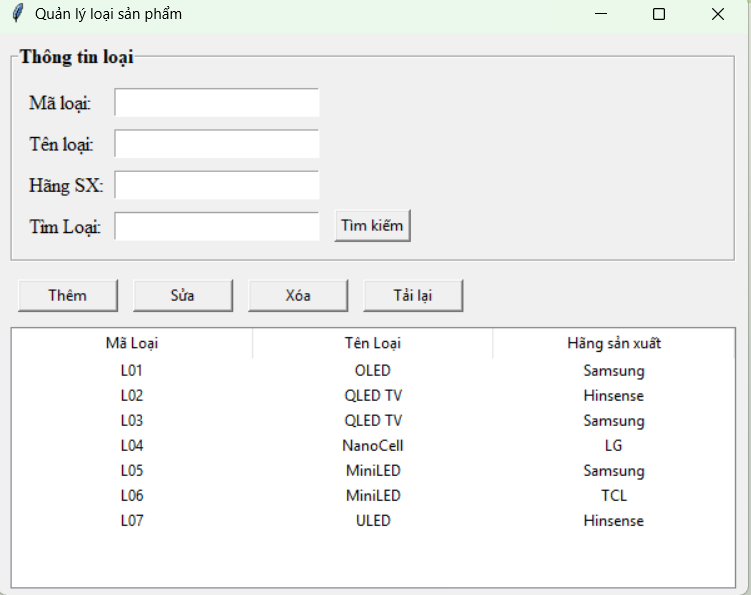


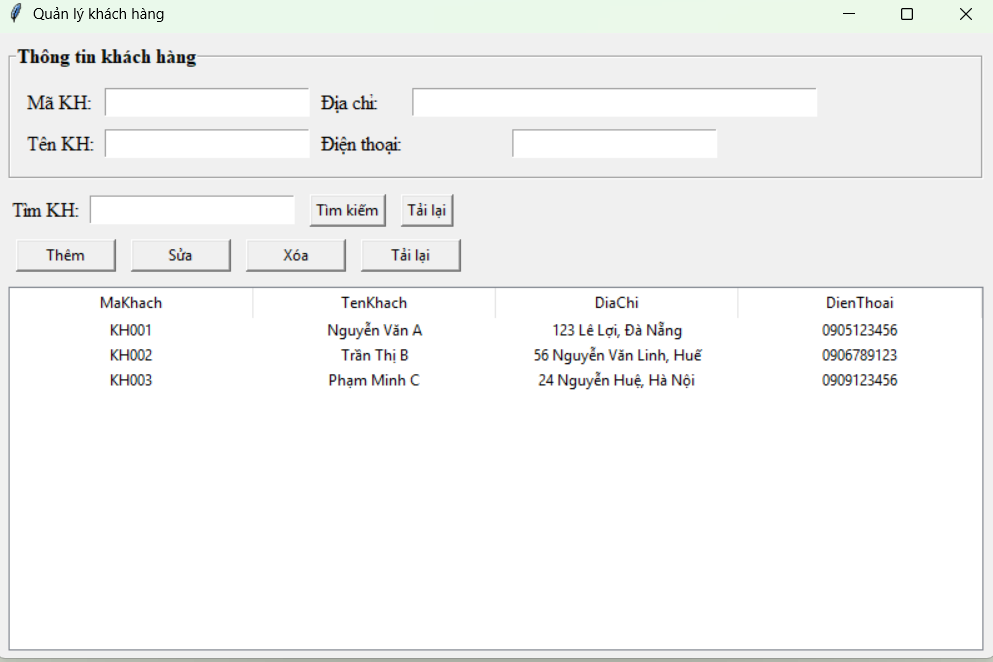
1. **THIẾT KÉ GIAO DIỆN QUẢN LÝ CỬA HÀNG TIVI**

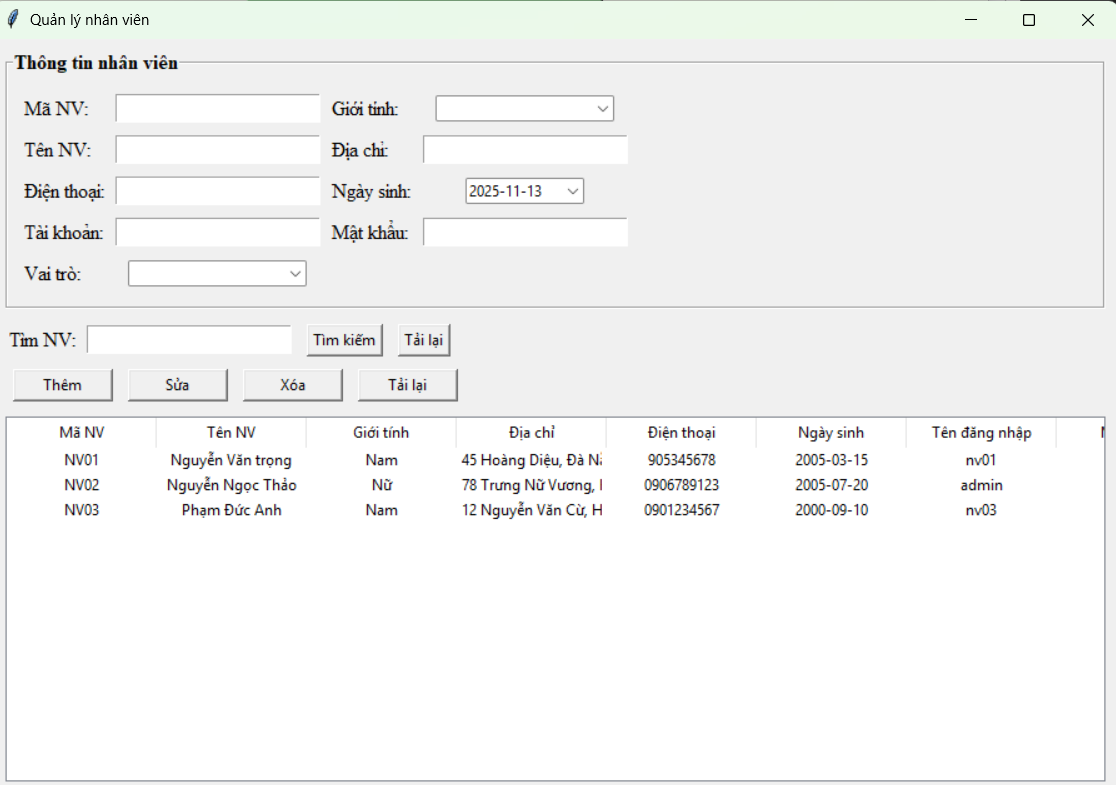
****

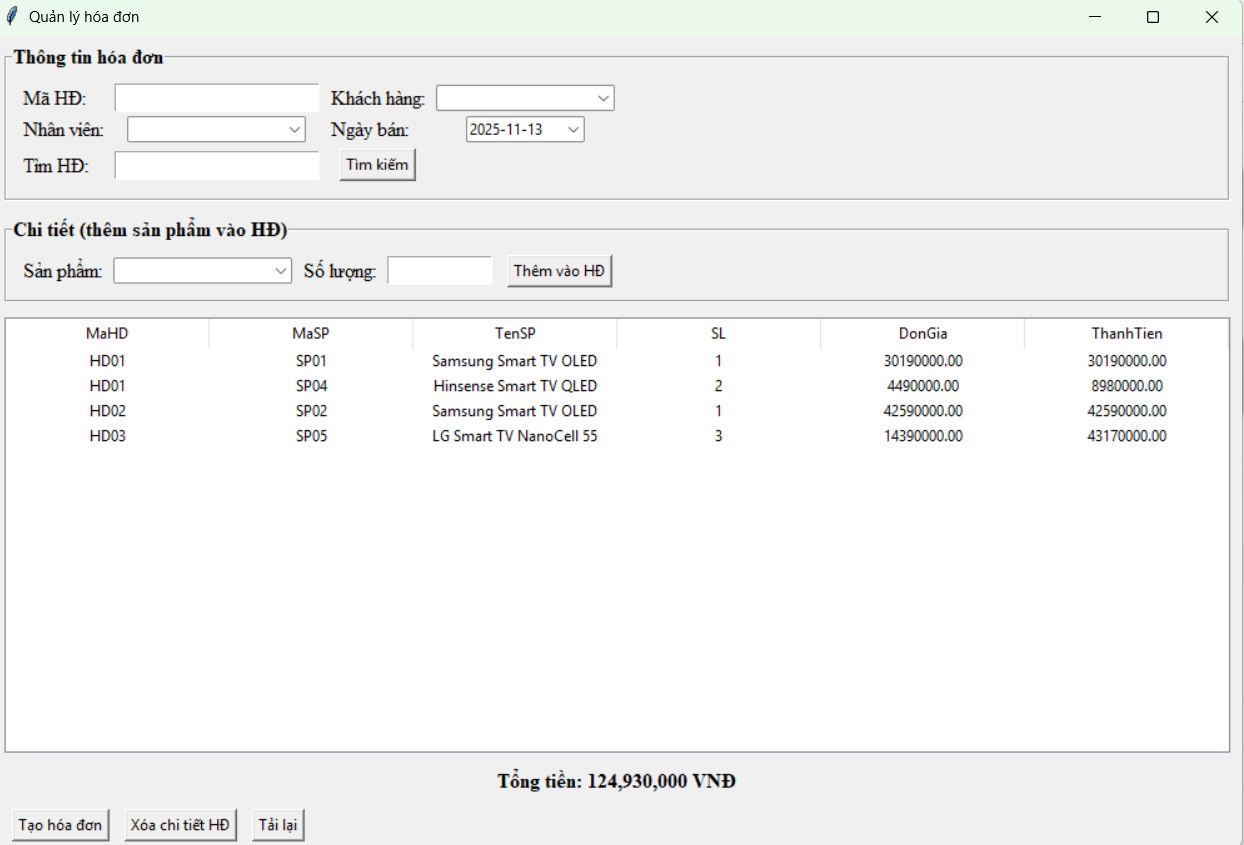
****

****

****

****

****

****

**.B.CÁC BƯỚC XÂY DỰNG ỨNG DỤNG**

Để xây dựng **Ứng dụng Quản lý Cửa hàng Tivi** hoàn chỉnh bằng Python, Tkinter và MySQL, cần tuân thủ một quy trình phát triển có cấu trúc, bao gồm việc thiết lập môi trường, thiết kế CSDL nền tảng và xây dựng giao diện người dùng theo kiến trúc phân lớp.

### 1. Cài đặt các Thư viện Cần thiết

Bước đầu tiên là thiết lập môi trường Python để có thể giao tiếp với MySQL và xây dựng giao diện đồ họa.

* **Python:** Đảm bảo đã cài đặt Python 3.x.
* **MySQL Server:** Cài đặt và khởi động dịch vụ máy chủ MySQL.
* **Thư viện Python (Sử dụng Terminal):**

**mysql-connector-python:** Thư viện tiêu chuẩn để kết nối code Python với MySQL Server.

**Bash**

**pip install mysql-connector-python**

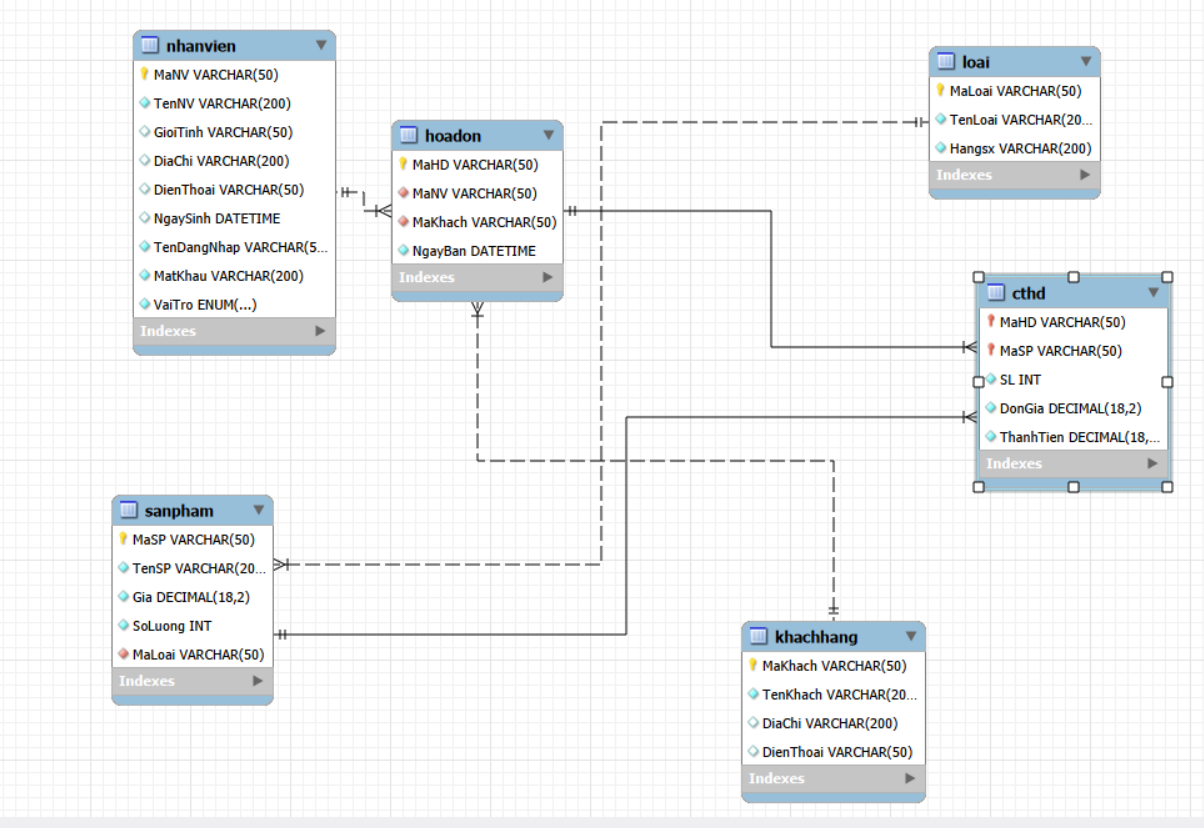
* **tkcalendar:** Thư viện Tkinter mở rộng, cần thiết cho việc chọn ngày

(ví dụ: ngày lập hóa đơn, ngày sinh nhân viên) một cách trực quan.

**Bash**

**pip install tkcalendar**

**2.Thiết kế cơ sở dữ liệu MySQL**

****

1. ***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 của các bảng quản lý của cửa hàng TiVi

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

- **Quản lý Sản phẩm:** Thêm, Sửa, Xóa, Tìm kiếm thông tin chi tiết của Tivi (Mã, Tên, Giá, Số lượng tồn, Liên kết với Loại sản phẩm).

**- Quản lý Loại sản phẩm:** Quản lý các danh mục phân loại (ví dụ: Smart TV, Full HD, UHD 4K, Tivi hãng Samsung/Sony/LG).

- **Quản lý Khách hàng:** Quản lý thông tin Khách hàng để phục vụ việc lập Hóa đơn và chăm sóc sau bán hàng.

- **Quản lý Nhân viên:** Quản lý thông tin cá nhân và tài khoản (Tên đăng nhập, Mật khẩu, Phân quyền) để đảm bảo tính bảo mật và phân công trách nhiệm.

- **Quản lý Hóa đơn:** Lập hóa đơn, quản lý chi tiết hóa đơn (liên kết Sản phẩm, Khách hàng, Nhân viên) và cập nhật tồn kho theo thời gian thực.

- **Thoát** : Nhấn xác nhận để thoát khỏi giao diện

**4.Viết mã Python kết nối cơ sở dữ liệu:**

Sử dụng thư viện **mysql.connector** để thiết lập kết nối và vận dụng các hàm xử lý CSDL đã định nghĩa trong file **database.py** để thao tác với tất cả các bảng

**- Thêm** Sản phẩm, Loại, KH,NV: INSERT INTO [tên bảng] VALUES(...)

- **Sửa** Sản phẩm, Loại, KH, NV : UPDATE [tên bảng] SET [trường]=%s, ... WHERE [Mã khóa chính]=%s

**- Xóa** Sản phẩm, Loại, KH, NV : DELETE FROM [tên bảng] WHERE [Mã khóa chính]=%s

**Tìm kiếm** Nhập từ khóa vào ô tìm kiếm, nhấn nút **"Tìm"**.

SELECT \* FROM [tên bảng] WHERE [Trường] LIKE %s

**Xem/Đọc** Tất cả 5 module : SELECT \* FROM [tên bảng]

1. **Xây dựng giao diện Tkinter**

**- Thêm:**Khi người dùng nhập đầy đủ thông tin vào các trường nhập liệu (**Entry** và **Combobox**) và nhấn nút **"Thêm"**.Nếu thành công, hiển thị hộp thoại **messagebox.showinfo**; sau đó, hàm **load\_data()** được gọi để **cập nhật lại bảng Treeview** (làm mới danh sách hiển thị).

**- Xem**:Khi người dùng mở cửa sổ module ứng dụng (ví dụ: sanpham.py, hoadon.py).mã Python sẽ truy vấn cơ sở dữ liệu, lấy tất thông tin muc đã mở hiển thị chúng dưới dạng một danh sách hoặc bảng rên giao diện Tkinter.

**- Tìm kiếm:**Khi người dùng nhập từ khóa (Mã hoặc Tên) vào ô tìm kiếm và nhấn nút **"Tìm"**.Mã Python sẽ lấy từ khóa, thực hiện truy vấn **SELECT \* FROM [tên bảng] WHERE [Trường] LIKE %s** khỏi CSDL. Kết quả trả về sẽ được dùng để **lọc** và hiển thị lại trên bảng **ttk.Treeview**

**- Xóa** :Khi người dùng chọn một dòng trên bảng danh sách (**Treeview**) và nhấn nút **"Xóa"**Hiển thị hộp thoại **messagebox.askyesno** để xác nhận. Nếu người dùng chọn **"Yes"**, lệnh DELETE được thực thi, và sau đó **load\_data()** được gọi để xóa bản ghi đó khỏi bảng **Treeview** trên giao diện.

**C.HƯỚNG DẪN CÀI ĐẶT & CODE ỨNG DỤNG**

Ứng dụng sẽ có:

- Giao diện đăng nhập cho nhân viên và quản lý(quản lý mới có thể chỉnh sửa thông tin).

- Giao diện lựa chọn mục cần xem - Nhập thông tin sản phẩm ,loại ,khách hàng ,nhân viên

- Bảng danh sách sản phẩm ,loại,khách hàng ,nhân viên ,hóa đơn

- Chức năng CRUD (Thêm, Sửa, Xóa, Lưu,Tìm kiếm).

- Dữ liệu lưu trực tiếp vào **MySQL 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:

Screenshot 2025-11-15 134114

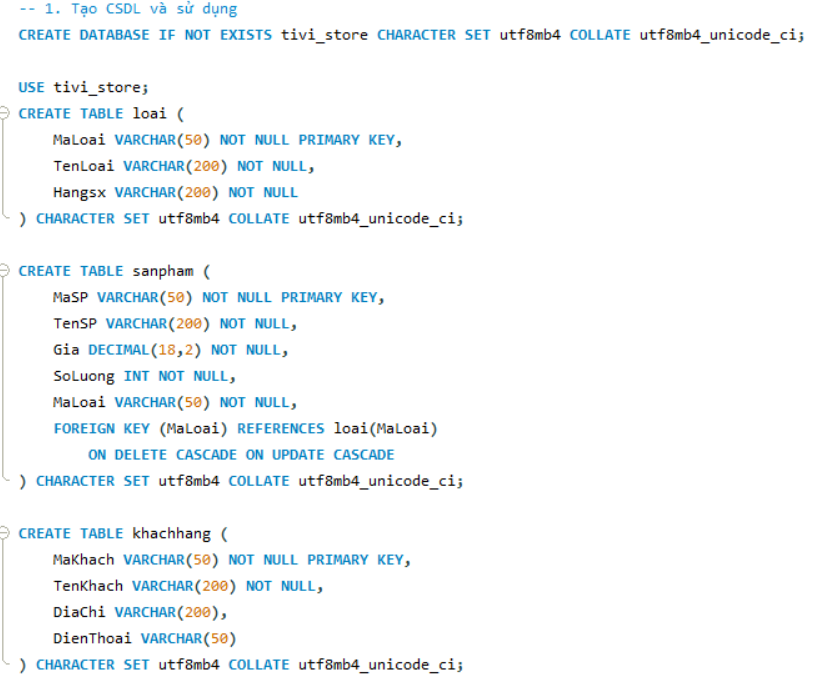
Giải thích:

mysql-connector-python: dùng để kết nối Python với MySQL.

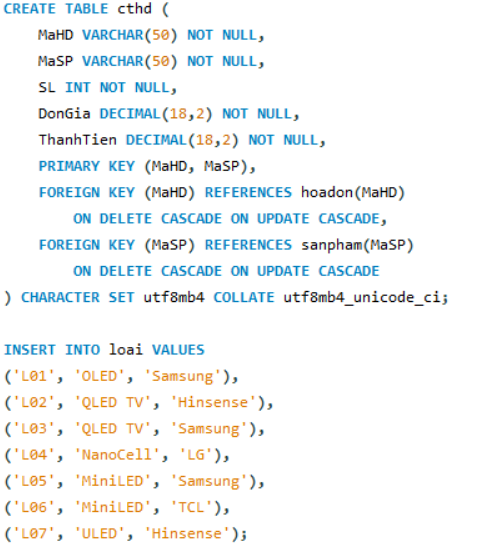
tkcalendar: để chọn ngày sinh bằng DateEntry.

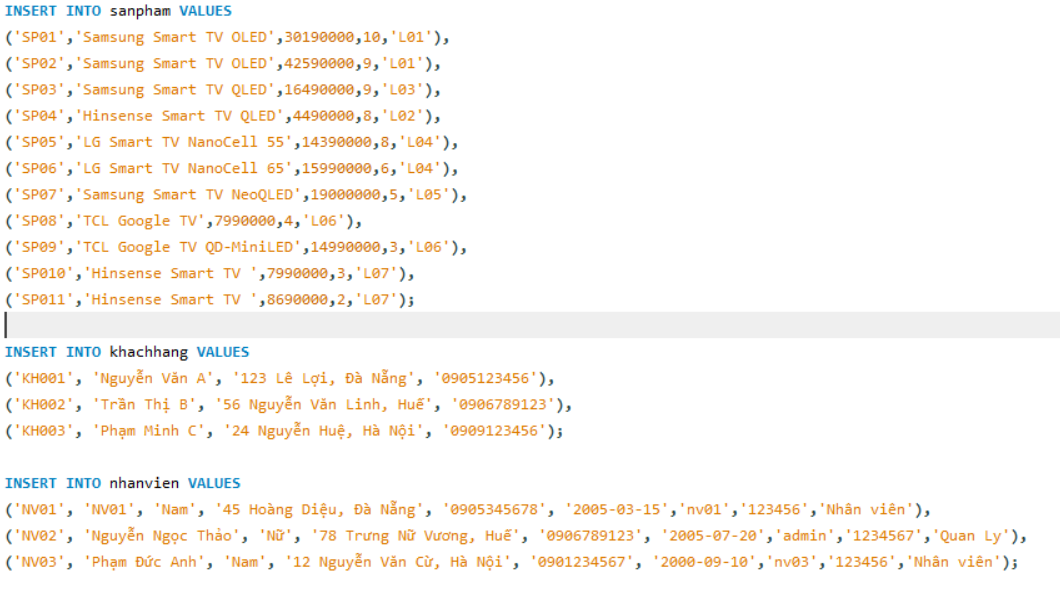
1. **Chuẩn bị CSDLMySQL**

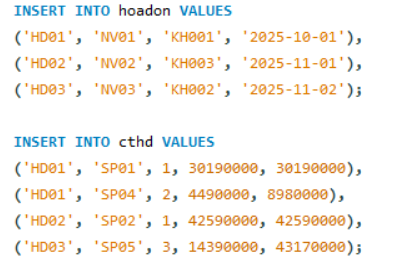
Mở MySQL (WorkBench, phpMyAdmin hoặc terminal) rồi tạo database và bảng











**3. Viết code ứng dụng với Tkinter + MySQL**

