

**ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC AN GIANG**



ĐỒ ÁN LẬP TRÌNH PYTHON CĂN BẢN

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ BÁN SÁCH VỚI
PYTHON, TKINTER VÀ SQL SERVER**

AN GIANG, 11-2025

Chuyên Đề: Python (COS525)

Giảng viên hướng dẫn: Ths. Nguyễn Ngọc Minh

Sinh viên thực hiện:

- DTH235.Lê Đoàn Nguyên Anh. DH24TH1_NhómTH1_Tô TH2

An Giang, 11-2025

GIỚI THIỆU CHUNG

Đồ án "Xây dựng ứng dụng quản lý bài hát với Python, Tkinter và SQL Server" được thực hiện nhằm mục tiêu xây dựng một hệ thống thông tin hoàn chỉnh quản lý bài hát, đáp ứng các yêu cầu thực tế của ứng dụng phát nhạc hiện đại.

Mục tiêu chính của đồ án là:

- Thiết kế giao diện người dùng thân thiện
- Kết nối và quản lý cơ sở dữ liệu SQL Server
- Triển khai các chức năng CRUD (Create, Read, Update, Delete)
- Xây dựng hệ thống tìm kiếm nâng cao

CHƯƠNG 1: GIỚI THIỆU

1.1 Đặt Vấn Đề

Trong các hiệu sách và cửa hàng bán sách hiện đại, việc quản lý thông tin sách là một trong những nhiệm vụ quan trọng và phức tạp. Cần phải có một hệ thống thông tin quản lý bán sách hiệu quả, cho phép:

- Thêm thông tin bài hát (mã bài hát, tên bài hát, ca sĩ, giới tính, ngày phát hành, thể loại)
- Sửa đổi thông tin bài hát
- Xóa bài hát khỏi danh sách
- Xem danh sách tất cả bài hát
- Tìm kiếm thông tin bài hát nhanh chóng (theo mã bài hát, tên bài hát, ca sĩ, giới tính, ngày, thể loại)
- Lưu Bài Hát

Đồ án này xây dựng một ứng dụng quản lý bán sách sử dụng Python, Tkinter và SQL Server nhằm giải quyết các yêu cầu trên.

1.2 Mục Tiêu Nghiên Cứu

Mục Tiêu Chung

Xây dựng một ứng dụng quản lý bán sách hoàn chỉnh, có giao diện thân thiện và các chức năng CRUD cơ bản, kết nối với SQL Server.

Mục Tiêu Cụ Thể

- Thiết kế giao diện người dùng bằng thư viện Tkinter
- Kết nối và thao tác với cơ sở dữ liệu SQL Server sử dụng pyodbc
- Triển khai các chức năng thêm, sửa, xóa, và xem danh sách sách
- Xây dựng chức năng tìm kiếm nâng cao
- Áp dụng kỹ thuật lập trình Python để xử lý dữ liệu an toàn

1.3 Nội Dung Đồ Án

Đồ án này sẽ bao gồm:

1. Thiết kế giao diện quản lý bài hát
2. Chuẩn bị cơ sở dữ liệu SQL Server (QLBS)
3. Viết code ứng dụng với Tkinter + SQL Server
4. Triển khai các chức năng CRUD
5. Hướng dẫn cài đặt và sử dụng

CHƯƠNG 2: TỔNG QUAN VỀ NGHIÊN CỨU

2.1 Giới Thiệu Về Đề

Hệ thống quản lý bài hát là một phần không thể thiếu trong hoạt động kinh doanh của các ứng dụng nghe nhạc. Các ứng dụng quản lý bài hát cần phải cung cấp:

- Giao diện dễ sử dụng và thân thiện
- Khả năng lưu trữ dữ liệu sách an toàn
- Tốc độ truy xuất dữ liệu nhanh
- Các chức năng cơ bản nhưng đầy đủ (thêm, sửa, xóa, tìm kiếm)

2.2 Cơ Sở Lý Thuyết

Python

Python là ngôn ngữ lập trình bậc cao, dễ học và mạnh mẽ. Nó được sử dụng rộng rãi trong phát triển ứng dụng web, phân tích dữ liệu, và ứng dụng desktop. Python cung cấp các thư viện phong phú và cộng đồng phát triển lớn.

Tkinter

Tkinter là thư viện giao diện người dùng (GUI) chuẩn của Python. Nó cho phép tạo các ứng dụng desktop với:

- Cửa sổ (Window)
- Trường nhập liệu (Entry)
- Nút bấm (Button)
- Bảng hiển thị (Treeview) cho danh sách dữ liệu
- Các widget khác

SQL Server

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ mạnh mẽ. Nó cung cấp:

- Lưu trữ dữ liệu an toàn và hiệu quả
- Truy xuất dữ liệu nhanh
- Hỗ trợ nhiều bảng và quan hệ (thể loại, tác giả, nhà xuất bản, sách)
- Tính toán vẹn dữ liệu thông qua khóa ngoại

PyODBC

PyODBC là thư viện Python cho phép kết nối đến SQL Server thông qua ODBC Driver. Nó cung cấp:

- Kết nối an toàn với SQL Server
- Thực thi các lệnh SQL
- Xử lý lỗi kết nối

CHƯƠNG 3: PHƯƠNG PHÁP NGHIÊN CỨU VÀ CÁC BƯỚC XÂY DỰNG

3.1 Cài Đặt Môi Trường

Cài Đặt Python

Đảm bảo Python đã được cài đặt trên máy tính. Kiểm tra bằng lệnh:

```
python --version
```

Cài Đặt SQL Server

Cài đặt SQL Server và công cụ quản trị như SQL Server Management Studio (SSMS) hoặc SQL Server Data Tools.

Cài Đặt Thư Viện Cần Thiết

Sử dụng pip để cài đặt các thư viện:

```
pip install pyodbc
```

3.2 Thiết Kế Cơ Sở Dữ Liệu SQL Server

Tạo Database

Mở MySQL workbench 8.0 CE rồi chạy lệnh:

```
CREATE DATABASE IF NOT EXISTS `qlbaihat`
```

```
USE `qlbaihat`
```

Cấu Trúc Bảng Chính (baihat)

Trường	Kiểu Dữ Liệu	Mô Tả
‘ma_bh’	varchar(20)	Mã bài hát (khóa chính)
‘tenbaihat’	VARCHAR(50)	Tên bài hát
‘casi’	varchar(50)	Ca Sĩ
‘phai’	varchar(10)	Phái
‘ngay_phathanh’	date	Ngày Phát Hành Default NULL
‘loai’	Varchar(10)	Thể loại nhạc

3.3 Thiết Kế Giao Diện

Giao diện ứng dụng bao gồm:

- **Tiêu đề:** Quản Lý Bài Hát- Dữ liệu MySQL workbench 8.0 CE

- **Bảng danh sách:** Hiển thị toàn bộ sách (mabh, tenbaihat, casi, phai, ngayphathanh,loai)
 - **Ô tìm kiếm:** Tìm sách theo mã bài hát, tên bài hát, ca sĩ, phái , ngày phát hành , loài bài hát
 - **Nút chức năng:** Thêm bài hát, Sửa bài hát, Xóa bài hát, Tìm kiếm, xuất file exel , Lưu bài hát , Thoát
-

CHƯƠNG 4: KẾT QUẢ ĐẠT ĐƯỢC VÀ DEMO CODE

4.1 Các Chức Năng Chính

Chức Năng CRUD

Ứng dụng triển khai đầy đủ các chức năng:

- **Create (C):** Thêm bài hát mới
- **Finding(F):**Tìm Kiếm Bài hát
- **Update (U):** Sửa thông tin Bài Hát
- **Delete (D):** Xóa Bài Hát
- **Save (S) :** Lưu Bài Hát

Các Nút Chức Năng

1. **Thêm Bài Hát:** Thêm bài hát mới vào cơ sở dữ liệu (mã bài hát, tên bài hát, ca sĩ, giới tính, ngày phát hành, thể loại)
2. **Sửa Bài Hát:** Sửa thông tin bài hát được chọn
3. **Xóa Bài Hát:** Xóa một hoặc nhiều bài hát khỏi cơ sở dữ liệu
4. **Tìm:** Tìm kiếm sách theo mã bài hát, tên bài hát, ca sĩ, giới tính, ngày phát hành,
5. **Thoát:** Đóng ứng dụng

4.2 Hướng Dẫn Cài Đặt & Chạy Code

Bước 1: Chuẩn Bị Cơ Sở Dữ Liệu

Chạy script SQL để tạo các bảng và insert dữ liệu mẫu.

Bước 2: Kết Nối SQL Server

Chỉnh sửa thông tin kết nối trong hàm ket_noi_db():

```
def connect_db():
    return mysql.connector.connect(
        host="127.0.0.1",
        user="root",
        password="12345",
        database="qlbaihat"
    )
```

Lưu ý: Thay đổi giá trị SERVER, UID, PWD phù hợp với môi trường của bạn.

Bước 3: Chạy Ứng Dụng

[python app.py](#)

4.3 Đoạn Code Chính

Kết Nối SQL Server

```
import mysql.connector
def connect_db():
    return mysql.connector.connect(
        host="127.0.0.1",
        user="root",
        password="12345",
        database="qlbaihat"
    )
```

Hàm Tải Dữ Liệu

```
def load_data():
    for i in tree.get_children():
        tree.delete(i)
```

```

conn = connect_db()
try:
    cur = conn.cursor()
    cur.execute("SELECT * FROM baihat")
    rows = cur.fetchall()
    for row in rows:
        tree.insert("", tk.END, values=row)
except Exception as e:
    messagebox.showerror("Lỗi kết nối", str(e))
finally:
    conn.close()

```

Hàm Thêm Bài Hát

```

def clear_input():
    entry_mabh.delete(0, tk.END)
    entry_tenbh.delete(0, tk.END)
    entry_casi.delete(0, tk.END)
    gender_var.set("Nam")
    date_entry.set_date("2000-01-01")
    cbb_loai.set("") entry_mabh.config(state='normal')

def ThemBaiHat(): mabh = entry_mabh.get()
    tenbh = entry_tenbh.get()
    gioitinh = gender_var.get()
    casi = entry_casi.get()
    ngayph = date_entry.get_date()
    loaibh = cbb_loai.get()

    if mabh == "" or tenbh == "" or casi == "":
        messagebox.showwarning("Thiếu dữ liệu", "Vui lòng nhập Mã, Họ và tên Ca Sĩ")
        return
    conn = connect_db()
    try:
        cur = conn.cursor()
        cur.execute("INSERT INTO baihat VALUES (%s, %s, %s, %s, %s, %s)",


```

```

        (mabh, tenbh, casi,gioitinh,ngayph, loaibh))
    conn.commit()
    messagebox.showinfo("Thành công", "Thêm Bài Hát thành công")
    load_data()
    clear_input()
except mysql.connector.IntegrityError:
    messagebox.showerror("Lỗi", f"Mã Bài Hát '{mabh}' đã tồn tại!")
except Exception as e:
    messagebox.showerror("Lỗi", str(e))
finally:
    conn.close()

```

Hàm Xóa Bài Hát

```

def XoaBaiHat():
    selected = tree.selection()
    if not selected:
        messagebox.showwarning("Chưa chọn", "Hãy chọn Bài Hát trên bảng để xóa")
        return
    mabh = tree.item(selected)["values"][0]

    confirm = messagebox.askyesno("Xác nhận", f'Bạn có chắc muốn xóa giáo viên có mã {mabh}?')
    if confirm:
        conn = connect_db()
        cur = conn.cursor()
        cur.execute("DELETE FROM BaiHat WHERE ma_bh = %s", (mabh,))
        conn.commit()
        conn.close()
        load_data()
        clear_input()
        messagebox.showinfo("Thành công", "Đã xóa bài hát")

```

Hàm Tìm Kiếm Bài Hát

```

def TimKiem(): search_win = tk.Toplevel(root) search_win.title("Tìm kiếm Bài Hát")
search_win.geometry("300x120")

tk.Label(search_win, text="Nhập mã hoặc tên Bài Hát:").pack(pady=10)
entry_search = tk.Entry(search_win, width=30)
entry_search.pack(pady=5)
def ThucHienTim():

```

```

keyword = entry_search.get()
if keyword == "":
    messagebox.showwarning("Thông báo", "Vui lòng nhập từ khóa!")
    return
for i in tree.get_children():
    tree.delete(i)
conn = connect_db()
try:
    cur = conn.cursor()
    sql = "SELECT * FROM baihat WHERE ma_bh LIKE %s OR tenbaihat LIKE %s OR casi LIKE %s"
    val = (f"%{keyword}%", f"%{keyword}%", f"%{keyword}%")
    cur.execute(sql, val)
    rows = cur.fetchall()
    if len(rows) == 0:
        messagebox.showinfo("Kết quả", "Không tìm thấy bài hát nào.")
        load_data()
    else:
        for row in rows:
            tree.insert("", tk.END, values=row)
search_win.destroy()
except Exception as e:
    messagebox.showerror("Lỗi", str(e))
finally:
    conn.close()

tk.Button(search_win, text="Tìm kiếm", command=ThucHienTim).pack(pady=10)

```

CHƯƠNG 5: KẾT LUẬN VÀ KHUYẾN NGHỊ

5.1 Kết Luận

Đồ án đã thành công xây dựng một ứng dụng quản lý bài hát hoàn chỉnh với các tính năng:

- Giao diện người dùng thân thiện được xây dựng bằng Tkinter
- Kết nối và quản lý dữ liệu trong SQL Server

- Các chức năng CRUD đầy đủ
- Xử lý dữ liệu an toàn và hiệu quả

Ứng dụng có thể được sử dụng để quản lý thông tin bài hát của các ứng dụng phát nhạc nhỏ và vừa.

5.2 Hạn Chế

- Ứng dụng hiện tại chỉ hỗ trợ quản lý cơ bản
- Chưa có chức năng báo cáo thống kê chi tiết
- Chưa có hệ thống quyền truy cập (authentication)
- Chưa tối ưu hóa hiệu suất cho dữ liệu lớn

5.3 Khuyến Nghị

Để cải thiện ứng dụng trong tương lai, có thể:

- Thêm chức năng tìm kiếm nâng cao
- Triển khai hệ thống quyền truy cập (login/logout)
- Xây dựng các báo cáo thống kê
- Nâng cấp giao diện với các công cụ hiện đại hơn
- Tối ưu hóa cơ sở dữ liệu cho hiệu suất cao hơn

TÀI LIỆU THAM KHẢO

1 Nguyễn Ngọc Minh. "Qui định hình thức báo cáo đồ án môn Python." Trường Đại học An Giang, 2025.

2 Official Python Documentation. "tkinter --- Python interface to Tcl/Tk."
<https://docs.python.org/3/library/tkinter.html>

3 Microsoft SQL Server Documentation. "SQL Server Documentation." Microsoft, 2024. <https://learn.microsoft.com/en-us/sql/>

4 PyODBC Contributors. "pyodbc." PyPI Package Repository.
<https://pypi.org/project/pyodbc/>

5 Van Rossum, G. "Python Enhancement Proposals." [Python.org](https://www.python.org/).
<https://www.python.org/>