

TRƯỜNG ĐẠI HỌC VĂN LANG  
KHOA CÔNG NGHỆ THÔNG TIN



**BÁO CÁO PROJECT 2**  
**LẬP TRÌNH PYTHON NÂNG CAO**

**Chủ đề:**

**GIAO DIỆN GUI CƠ BẢN**

**SVTH: Trần Văn Hiếu – 2274802010262**

**LỚP: 241\_71ITSE31003\_0202**

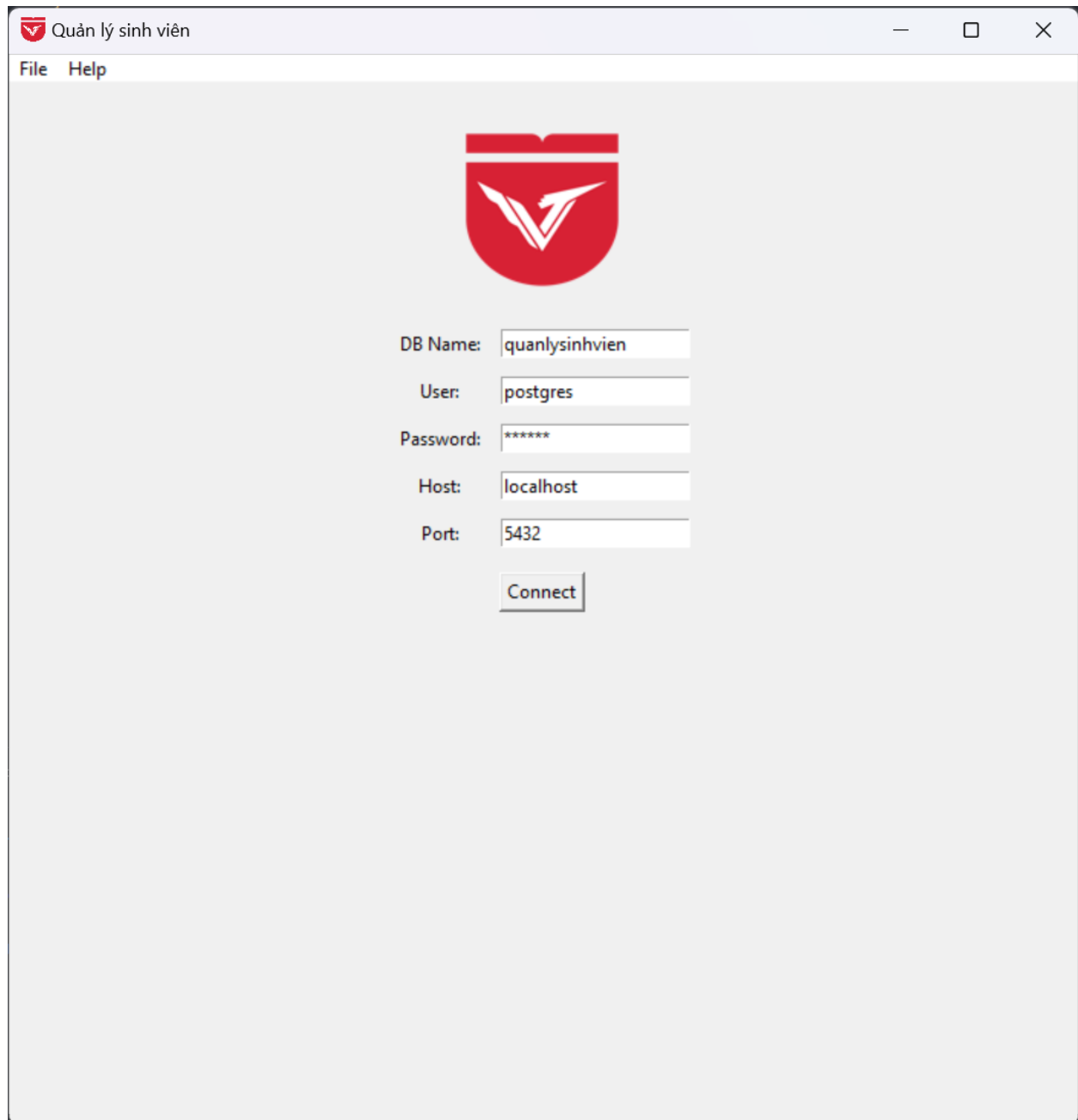
**GVHD: Huỳnh Thái Học**

**TP. Hồ Chí Minh – 10/2024**

# Mục Lục


1	Giao diện .....	3
2	Chức năng .....	4
2.1	Thêm hàng dữ liệu mới .....	4
2.2	Load dữ liệu.....	6
2.3	Chọn 1 hay nhiều hàng để xóa .....	7
2.4	Xóa dữ liệu được nhập trong mssv .....	7
2.5	Xóa nội dung trong ô nhập.....	9
2.6	Cập nhật dữ liệu .....	10
2.7	Xuất file excel .....	11
2.8	Thoát GUI .....	14
2.9	Hiển thị bản quyền .....	14
3	Mã chương trình .....	17
4	Github .....	25

# 1 Giao diện



Quản lý sinh viên

File Help



DB Name:

User:

Password:

Host:

Port:

Quản lý sinh viên

File Help

Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
2274802010971	Bùi Kế Tôn	Tường

Entry Frame

MSSV :

Họ :

Tên :

Button Frame

Thêm hàng mới

Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

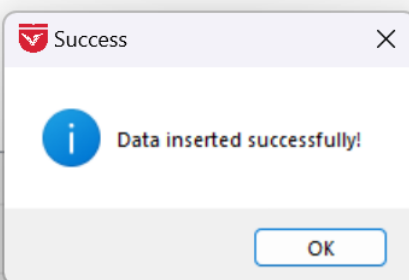
Cập nhật nội dung trong ô

## 2 Chức năng

### 2.1 Thêm hàng dữ liệu mới

## Quản lý sinh viên VLU

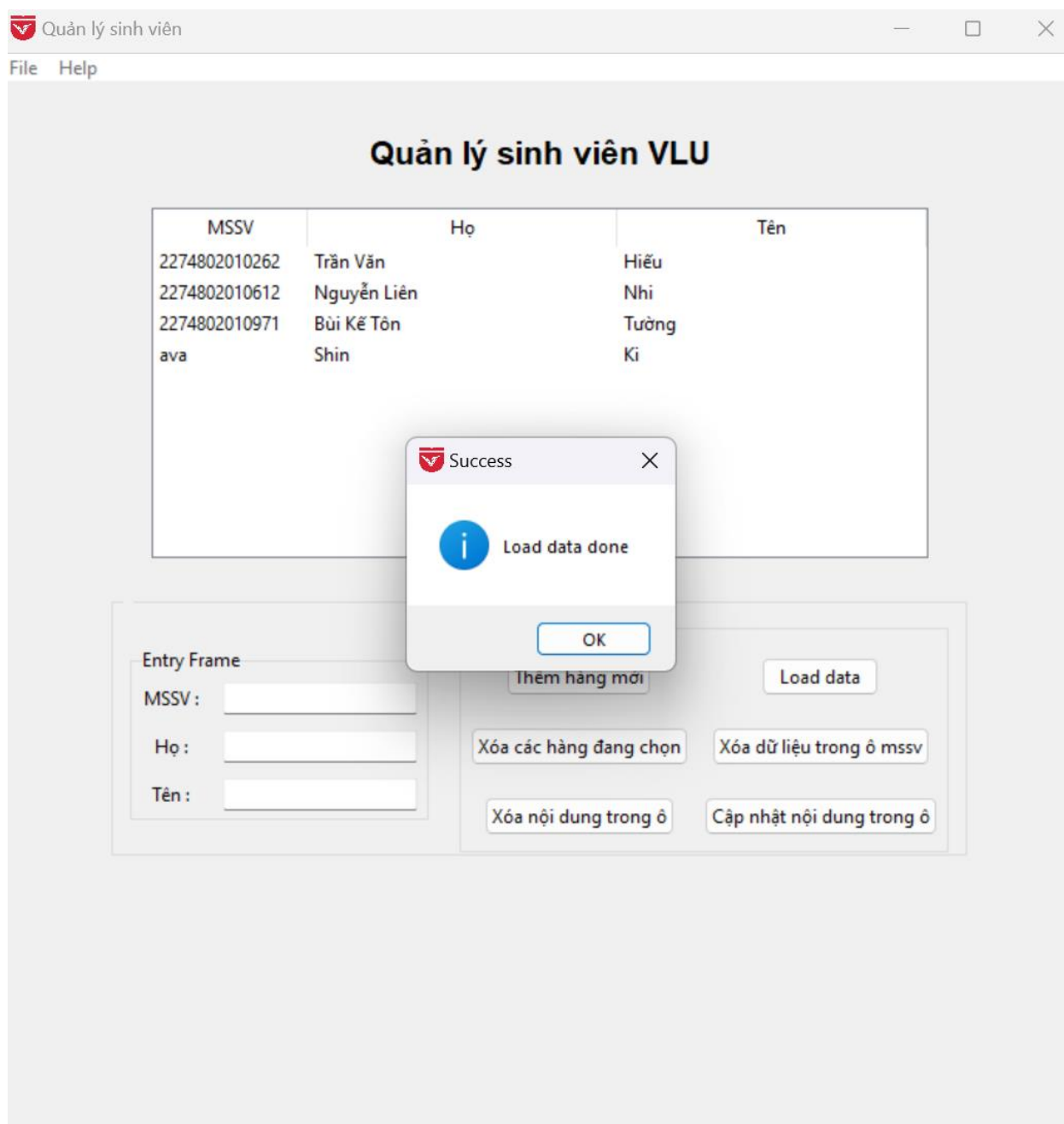
MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
2274802010971	Bùi Kế Tôn	Tường



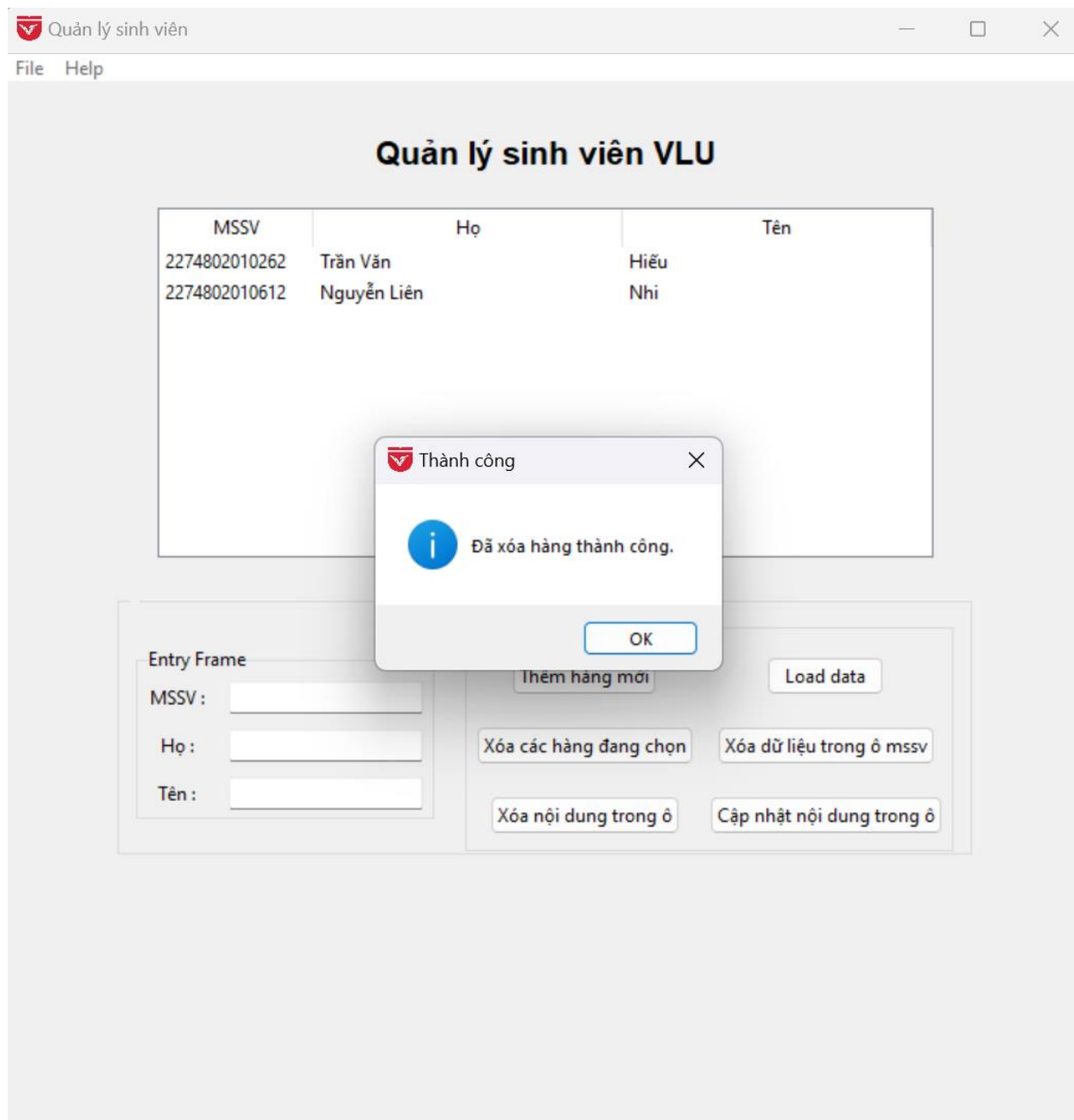
## Entry Frame

MSSV : Họ : Tên :

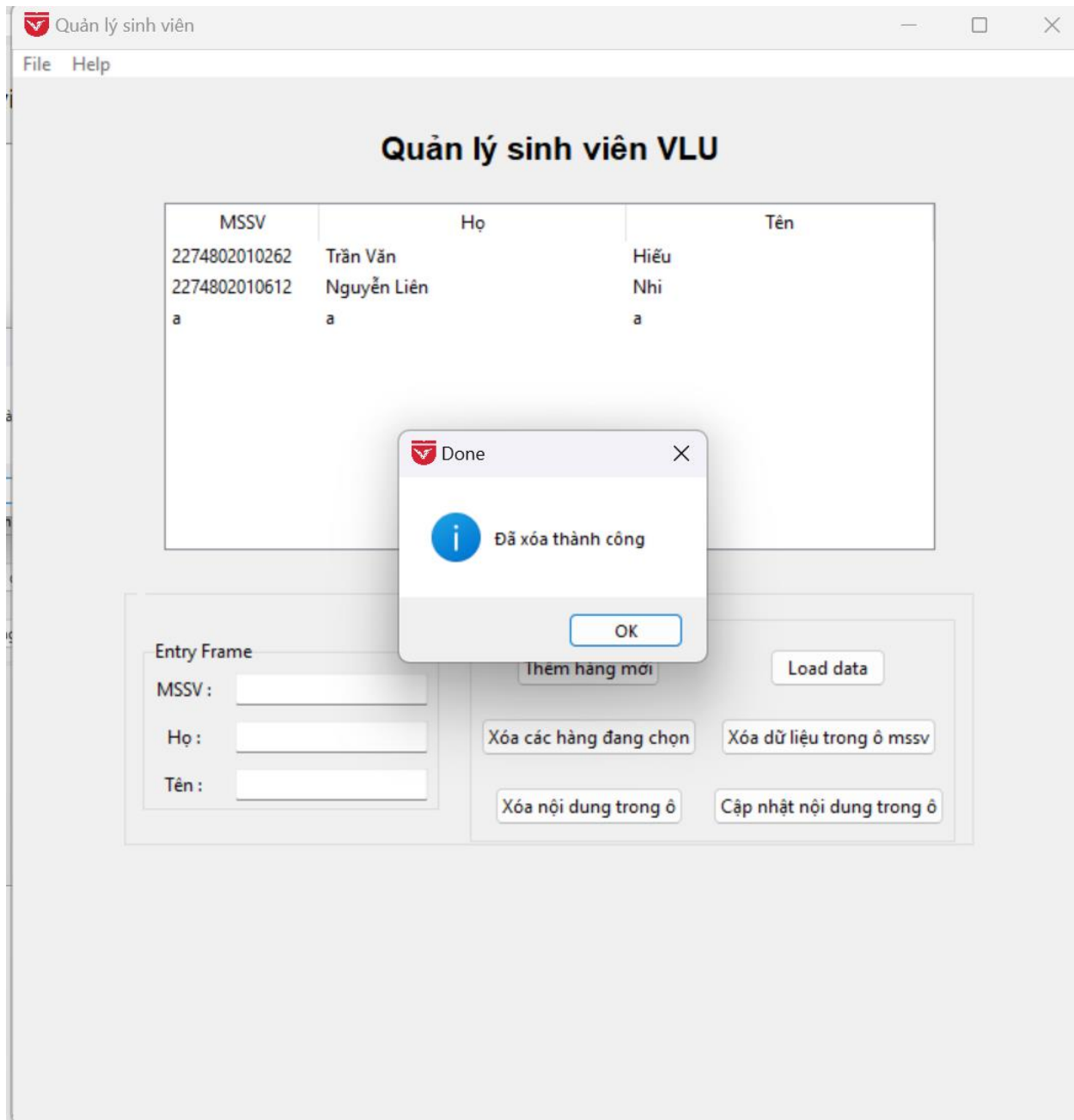
## 2.2 Load dữ liệu



## 2.3 Chọn 1 hay nhiều hàng để xóa



## 2.4 Xóa dữ liệu được nhập trong mssv





## 2.5 Xóa nội dung trong ô nhập

Quản lý sinh viên

File Help

### Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi

Entry Frame

MSSV :

Họ :

Tên :

Button Frame

Thêm hàng mới

Load data

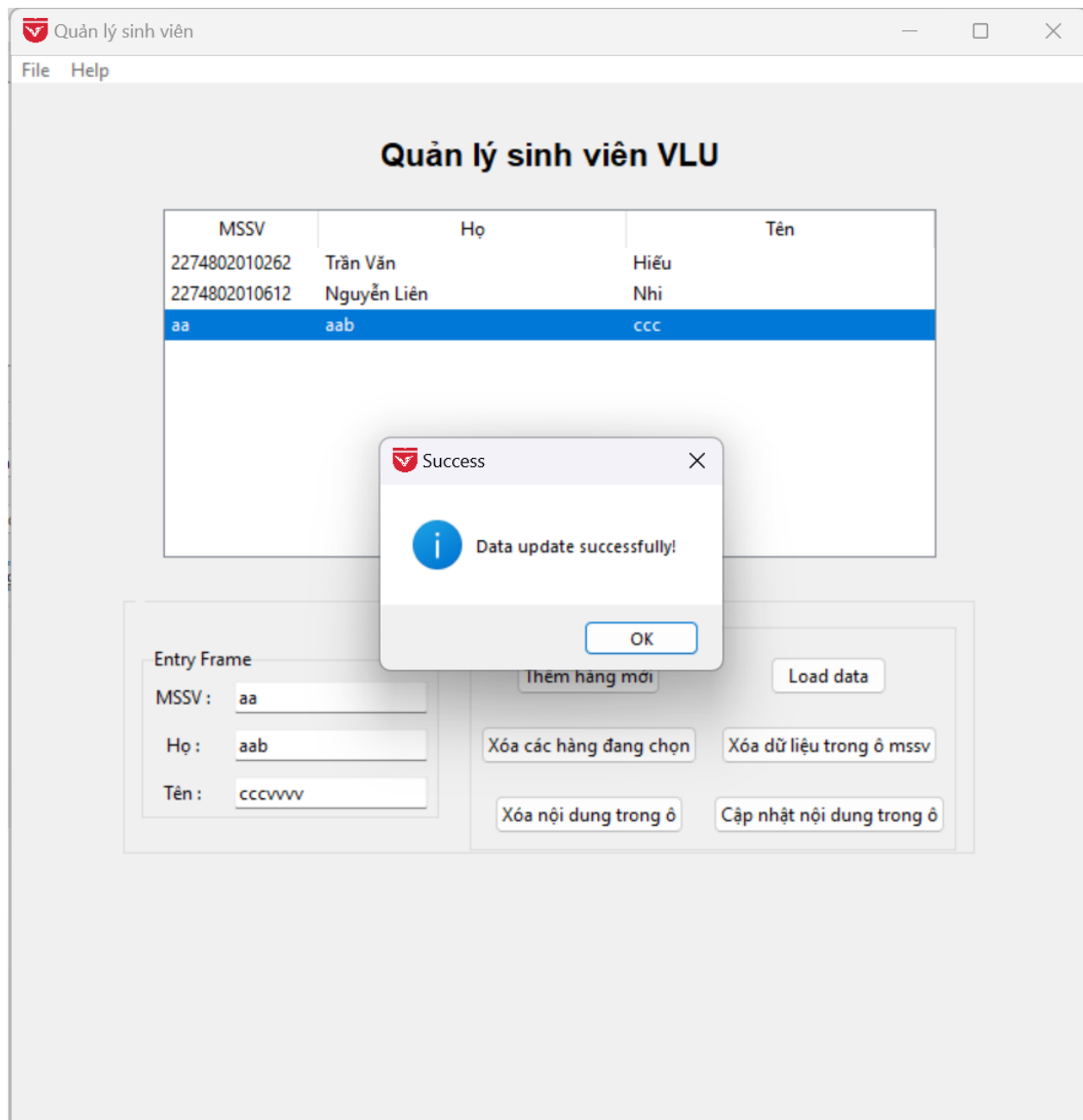
Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô

## 2.6 Cập nhật dữ liệu



## 2.7 Xuất file excel

Quản lý sinh viên

File Help

Xuất  
Exit

### Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
aa	aab	cccvvvv

Entry Frame

MSSV :

Họ :

Tên :

Button Frame

Thêm hàng mới

Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô

## Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
aa	aab	ccccvvv

Xuất file



Dữ liệu đã được xuất ra file Excel thành công!

OK

Entry Frame

MSSV :

Họ :

Tên :

Thêm hàng mới

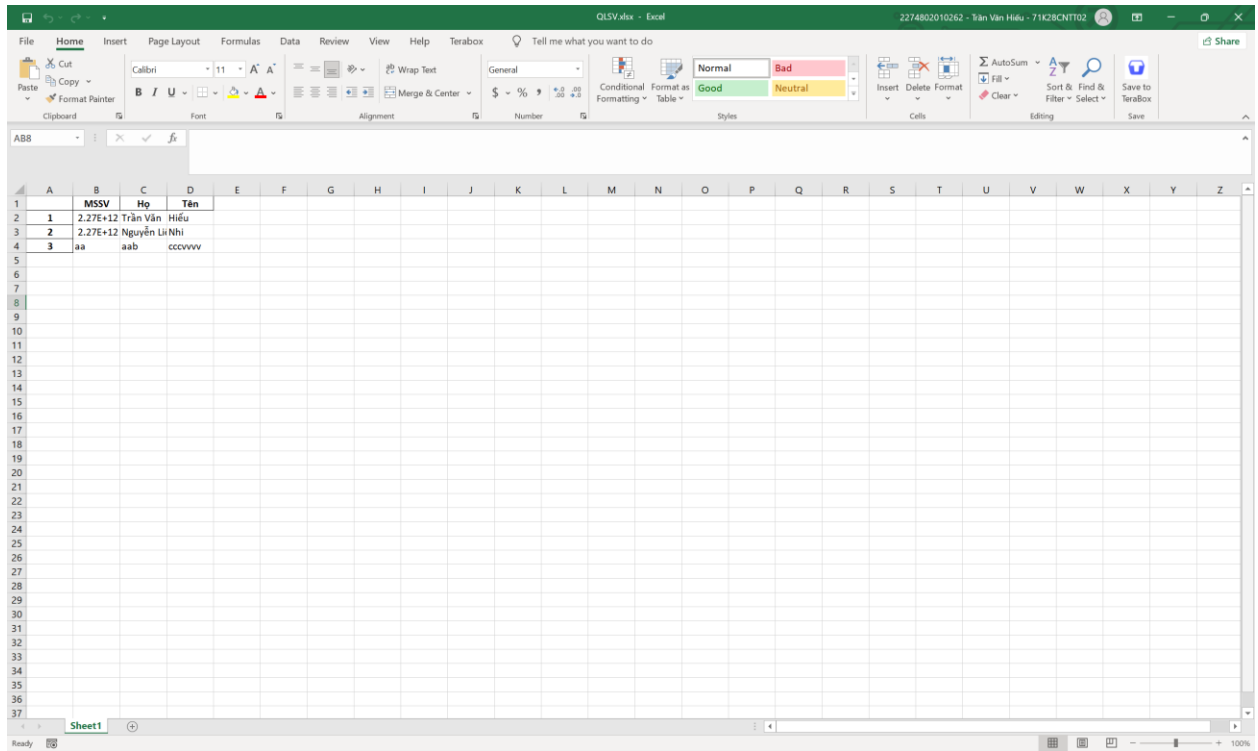
Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô



## 2.8 Thoát GUI

Quản lý sinh viên

File Help

Xuất  
Exit

### Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
aa	aab	ccccvvv

Entry Frame

MSSV :

Họ :

Tên :

Button Frame

Thêm hàng mới

Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô

## 2.9 Hiển thị bản quyền



File Help

About

## Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
aa	aab	cccvvvv

Entry Frame

MSSV :

Họ :

Tên :

Button Frame

Thêm hàng mới

Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô

## Quản lý sinh viên VLU

MSSV	Họ	Tên
2274802010262	Trần Văn	Hiếu
2274802010612	Nguyễn Liên	Nhi
aa	aab	cccwww



About



This is a GUI made by Hieu Tran version 1.0  
Release in 10/13/2024

OK

Entry Frame

MSSV :

Họ :

Tên :

Thêm hàng mới

Load data

Xóa các hàng đang chọn

Xóa dữ liệu trong ô mssv

Xóa nội dung trong ô

Cập nhật nội dung trong ô



### **3 Mã chương trình**

```

1
2 import tkinter as tk
3 from tkinter import Menu, ttk, messagebox, filedialog
4 import psycopg2
5 from psycopg2 import sql
6 import pandas as pd
7 from PIL import Image, ImageTk
8
9
10 class TableApp():
11     def __init__(self, root):
12         self.root = root
13         self.root.iconbitmap('logo.ico')
14         self.root.title("Kết nối database")
15         # File menu
16         self.menu_bar = Menu(self.root)
17         file_menu = Menu(self.menu_bar, tearoff=0)
18         self.menu_bar.add_cascade(label="File", menu=file_menu)
19         file_menu.add_command(label="Xuất", command=self.export_to_excel)
20         file_menu.add_command(label="Exit", command=self.quit_app)
21         # Help menu
22         help_menu = Menu(self.menu_bar, tearoff=0)
23         self.menu_bar.add_cascade(label="Help", menu=help_menu)
24         help_menu.add_command(label="About", command=self.msg_box_info)
25         self.root.config(menu=self.menu_bar)
26         self.root.title("Quản lý sinh viên")
27         self.db_name = tk.StringVar(value='quanlysinhvien')
28         self.user = tk.StringVar(value='postgres')
29         self.password = tk.StringVar(value='123456')
30         self.host = tk.StringVar(value='localhost')
31         self.port = tk.StringVar(value='5432')
32         self.table_name = tk.StringVar(value='sinhvien')
33         original_image = Image.open("logovlu.png")
34         resized_image = original_image.resize((100, 100))
35         self.image = ImageTk.PhotoImage(resized_image)
36
37         width = 700
38         height = 700
39         self.root.geometry(f"{height}x{width}")
40         screen_width = self.root.winfo_screenwidth()
41         screen_height = self.root.winfo_screenheight()
42         x = (screen_width - width) // 2
43         y = (screen_height - height) // 2
44         self.root.geometry(f'{width}x{height}+{x}+{y}')
45
46         self.widgets_connect()
47

```



```
1
2 def widgets_connect(self):
3
4     self.connection_frame = tk.Frame(self.root)
5     self.connection_frame.pack(pady=10)
6
7     image_label = tk.Label(self.connection_frame, image=self.image)
8     image_label.grid(row=0, column=0, columnspan=3, pady=20)
9
10    tk.Label(self.connection_frame, text="DB Name:").grid(
11        row=1, column=0, padx=5, pady=5)
12    tk.Entry(self.connection_frame, textvariable=self.db_name).grid(
13        row=1, column=1, padx=5, pady=5)
14
15    tk.Label(self.connection_frame, text="User:").grid(
16        row=2, column=0, padx=5, pady=5)
17    tk.Entry(self.connection_frame, textvariable=self.user).grid(
18        row=2, column=1, padx=5, pady=5)
19
20    tk.Label(self.connection_frame, text="Password:").grid(
21        row=3, column=0, padx=5, pady=5)
22    tk.Entry(self.connection_frame, textvariable=self.password,
23        show="*").grid(row=3, column=1, padx=5, pady=5)
24
25    tk.Label(self.connection_frame, text="Host:").grid(
26        row=4, column=0, padx=5, pady=5)
27    tk.Entry(self.connection_frame, textvariable=self.host).grid(
28        row=4, column=1, padx=5, pady=5)
29
30    tk.Label(self.connection_frame, text="Port:").grid(
31        row=5, column=0, padx=5, pady=5)
32    tk.Entry(self.connection_frame, textvariable=self.port).grid(
33        row=5, column=1, padx=5, pady=5)
34
35    tk.Button(self.connection_frame, text="Connect",
36        command=self.connect_to_manage).grid(row=6, columnspan=2, pady=10)
37
```

```

1
2 def widgets_manage(self):
3
4     # Khung chứa bảng
5     self.table_frame = tk.Frame(self.root)
6     self.table_frame.pack(pady=10)
7     tk.Label(self.table_frame, text="Quản lý sinh viên VLU", border=2, font=(
8         "Helvetica", 16, "bold")).grid(column=0, row=0, columnspan=3, pady=20)
9     # Tạo bảng Treeview
10    self.data_table = ttk.Treeview(self.table_frame, columns=(
11        "MSSV", "Họ", "Tên"), show="headings", height=10)
12    self.data_table.heading("MSSV", text="MSSV")
13    self.data_table.heading("Họ", text="Họ")
14    self.data_table.heading("Tên", text="Tên")
15
16    self.data_table.column("MSSV", width=100)
17    self.data_table.column("Họ", width=200)
18    self.data_table.column("Tên", width=200)
19    self.data_table.grid(column=0, row=2)
20    self.data_table.bind("<ButtonRelease-1>", self.on_tree_select)
21    self.load_data()
22
23    # Khung chứa các ô nhập liệu
24    form_frame = ttk.LabelFrame(self.root, text="")
25    form_frame.pack(pady=10)
26
27    # Nhân và ô nhập MSSV
28    input_frame = ttk.LabelFrame(form_frame, text="Entry Frame")
29    input_frame.grid(padx=10, row=0, column=0)
30    ttk.Label(input_frame, text="MSSV : ").grid(
31        row=0, column=0, padx=5, pady=5)
32    self.mssv = tk.StringVar()
33    ttk.Entry(input_frame, textvariable=self.mssv).grid(
34        row=0, column=1, padx=5, pady=5)
35
36    # Nhân và ô nhập Họ Tên
37    ttk.Label(input_frame, text="Họ : ").grid(
38        row=1, column=0, padx=5, pady=5)
39    self.ho = tk.StringVar()
40    ttk.Entry(input_frame, textvariable=self.ho).grid(
41        row=1, column=1, padx=5, pady=5)
42
43    ttk.Label(input_frame, text="Tên : ").grid(
44        row=2, column=0, padx=5, pady=5)
45    self.ten = tk.StringVar()
46    ttk.Entry(input_frame, textvariable=self.ten).grid(
47        row=2, column=1, padx=5, pady=5)
48
49    # Frame button
50    button_frame = ttk.LabelFrame(form_frame, text="Button Frame")
51    button_frame.grid(padx=10, row=0, column=1)
52    # Nút để thêm hàng mới
53    add_button = ttk.Button(
54        button_frame, text="Thêm hàng mới", command=self.add_data_button)
55    add_button.grid(column=0, row=0, padx=5, pady=10)
56
57    # Nút để Load data
58    load_button = ttk.Button(
59        button_frame, text="Load data", command=self.load_data_button)
60    load_button.grid(column=1, row=0, padx=5, pady=10)
61    # Nút để xóa data đang chọn
62    delete_button = ttk.Button(
63        button_frame, text="Xóa các hàng đang chọn", command=self.delete_selected_row)
64    delete_button.grid(column=0, row=1, padx=5, pady=10)
65
66    # Nút để xóa data trong ô
67    delete_input_button = ttk.Button(
68        button_frame, text="Xóa dữ liệu trong ô mssv", command=self.delete_data_button)
69    delete_input_button.grid(column=1, row=1, padx=5, pady=10)
70
71    # Nút clear input
72    clear_button = ttk.Button(
73        button_frame, text="Xóa nội dung trong ô", command=self.clear_inputs)
74    clear_button.grid(column=0, row=2, padx=5, pady=10)
75    # Nút update data
76    update_button = ttk.Button(
77        button_frame, text="Cập nhật nội dung trong ô", command=self.update_data_button)
78    update_button.grid(column=1, row=2, padx=5, pady=10)
79

```

```

1
2 def on_tree_select(self, event):
3     """Xử lý khi người dùng chọn hàng trong Treeview"""
4     selected_items = self.data_table.selection()
5     if selected_items:
6         item = selected_items[0]
7         values = self.data_table.item(item, 'values')
8         mssv = values[0]
9         ho = values[1]
10        ten = values[2]
11        self.mssv.set(mssv)
12        self.ho.set(ho)
13        self.ten.set(ten)
14
15 def insert_data_button(self):
16     mssv = self.mssv.get().strip()
17     ho = self.ho.get().strip()
18     ten = self.ten.get().strip()
19
20     if self.check_exsit(self.mssv.get()):
21         self.insert_data()
22         self.add_table(mssv, ho, ten)
23         self.clear_inputs()
24         self.load_data()
25     else:
26         messagebox.showerror("Error", "Mã sinh viên không tồn tại")
27
28 def add_data_button(self):
29     """Thêm sinh viên mới vào cơ sở dữ liệu và bảng Treeview"""
30     mssv = self.mssv.get().strip()
31     ho = self.ho.get().strip()
32     ten = self.ten.get().strip()
33
34     if self.validate_input(mssv, ho, ten):
35         if self.check_exist(mssv):
36             messagebox.showerror("Error", "Mã sinh viên đã tồn tại")
37         else:
38             self.insert_data()
39             self.add_table(mssv, ho, ten)
40             self.clear_inputs()
41             self.load_data()
42
43 def update_data_button(self):
44     mssv = self.mssv.get().strip()
45     if self.check_exist(mssv):
46         self.update_data()
47         self.clear_inputs()
48         self.load_data()
49     else:
50         messagebox.showerror("Error", "Mã sinh viên không tồn tại")
51
52 def delete_data_button(self):
53     mssv = self.mssv.get().strip()
54     if self.check_exist(mssv):
55         self.delete_data(mssv)
56         self.clear_inputs()
57         messagebox.showinfo("Done", "Đã xóa thành công ")
58         self.load_data()
59
60     else:
61         messagebox.showerror("Error", "Mã sinh viên không tồn tại")
62

```



```

1
2 def load_data_button(self):
3     if self.load_data():
4         messagebox.showinfo("Success", "Load data done")
5
6 def validate_input(self, mssv, ho, ten):
7     """Kiểm tra đầu vào không được để trống."""
8     if not mssv or not ho or not ten:
9         messagebox.showwarning(
10             "Lỗi nhập liệu", "MSSV và Họ Tên không được để trống.")
11     return False
12     return True
13
14 def add_table(self, mssv, ho, ten):
15     """Thêm hàng mới vào bảng Treeview."""
16     new_row = (mssv, ho, ten)
17     self.data_table.insert("", tk.END, values=new_row)
18
19 def clear_inputs(self):
20     """Xóa các ô nhập liệu sau khi thêm dữ liệu thành công."""
21     self.mssv.set("")
22     self.ho.set("")
23     self.ten.set("")
24
25 # ! Còn Return True False
26 def connect_db(self):
27     """Kết nối tới cơ sở dữ liệu PostgreSQL"""
28     try:
29         self.conn = psycopg2.connect(
30             dbname=self.db_name.get(),
31             user=self.user.get(),
32             password=self.password.get(),
33             host=self.host.get(),
34             port=self.port.get()
35         )
36         self.cur = self.conn.cursor()
37         messagebox.showinfo(
38             "Success", "Connected to the database successfully!")
39         return True
40     except Exception as e:
41         messagebox.showerror(
42             "Error", f"Error connecting to the database: {e}")
43         return False
44
45 def insert_data_tabletree(self, rows):
46     # Dấu * là giải nén tuple thành các phần tử riêng lẻ
47     # Xóa toàn bộ dữ liệu hiện tại
48     self.data_table.delete(*self.data_table.get_children())
49     for row in rows:
50         # các hàng được chèn không có quan hệ cha-con
51         self.data_table.insert("", tk.END, values=row)
52
53 def load_data(self):
54     """Tải dữ liệu từ cơ sở dữ liệu và hiển thị trong bảng Treeview"""
55     try:
56         query = sql.SQL(
57             "SELECT * FROM {}".format(sql.Identifier(self.table_name.get())))
58         self.cur.execute(query)
59         rows = self.cur.fetchall()
60         self.insert_data_tabletree(rows)
61         return True
62     except Exception as e:
63         self.conn.rollback()
64         messagebox.showerror("Error", f"Error loading data: {e}")
65         return False
66

```

```

1
2 def insert_data(self):
3     """Chèn dữ liệu sinh viên vào cơ sở dữ liệu"""
4     try:
5         insert_query = sql.SQL("INSERT INTO {} (mssv, ho,ten) VALUES (%s, %s,%s)").format(
6             sql.Identifier(self.table_name.get()))
7         data_to_insert = (self.mssv.get(), self.ho.get(), self.ten.get())
8         self.cur.execute(insert_query, data_to_insert)
9         self.conn.commit()
10        messagebox.showinfo("Success", "Data inserted successfully!")
11    except Exception as e:
12        self.conn.rollback()
13        messagebox.showerror("Error", f"Error inserting data: {e}")
14
15 def update_data(self):
16     try:
17         update_query = sql.SQL("UPDATE {} SET ho = %s, ten = %s WHERE mssv = %s").format(
18             sql.Identifier(self.table_name.get()))
19         data_to_update = (self.ho.get(), self.ten.get(), self.mssv.get())
20         self.cur.execute(update_query, data_to_update)
21         self.conn.commit()
22         messagebox.showinfo("Success", "Data update successfully!")
23    except Exception as e:
24        self.conn.rollback()
25        messagebox.showerror("Error", f"Error inserting data: {e}")
26
27 def delete_selected_row(self):
28     """Xóa hàng đang được chọn trong Treeview"""
29     try:
30         # Lấy tất cả các hàng đang được chọn
31         selected_items = self.data_table.selection()
32         if not selected_items:
33             messagebox.showwarning("Lỗi", "Vui lòng chọn hàng để xóa.")
34             return
35
36         for item in selected_items:
37             values = self.data_table.item(item, 'values')
38             mssv = values[0]
39             self.delete_data(mssv)
40             if item in self.data_table.get_children():
41                 self.data_table.delete(item)
42             else:
43                 messagebox.showwarning(
44                     "Lỗi", f"Item {item} không tồn tại trong bảng.")
45             self.clear_inputs()
46             messagebox.showinfo("Thành công", "Đã xóa hàng thành công.")
47    except Exception as e:
48        messagebox.showerror("Error", f"Error deleting selected row: {e}")
49
50 def delete_data_input(self):
51     try:
52         self.delete_data(self.mssv.get())
53         self.load_data()
54         self.clear_inputs()
55         messagebox.showinfo("Success", "Đã xóa thành công")
56    except Exception as e:
57        messagebox.showerror("Error", f"{e}")
58
59 def delete_data(self, mssv):
60     """Xóa dữ liệu sinh viên khỏi cơ sở dữ liệu theo MSSV"""
61     try:
62         delete_query = sql.SQL("DELETE FROM {} WHERE mssv = %s").format(
63             sql.Identifier(self.table_name.get()))
64         self.cur.execute(delete_query, (mssv,))
65         self.conn.commit()
66    except Exception as e:
67        self.conn.rollback()
68        messagebox.showerror(
69            "Error", f"Error deleting data from database: {e}")
70

```

```

1      def connect_to_manage(self):
2          check = self.connect_db()
3          if check:
4              self.connection_frame.pack_forget()
5              self.widgets_manage()
6
7      def msg_box_info(self):
8          messagebox.showinfo(
9              "About", "This is a GUI made by Hieu Tran version 1.0\n Release in 10/13/2024\n")
10
11     def quit_app(self):
12         self.root.destroy()
13
14     def check_exist(self, mssv):
15         children = self.data_table.get_children()
16         for child in children:
17             values = self.data_table.item(child, 'values')
18             if values[0] == mssv:
19                 return True
20         return False
21
22     def export_to_excel(self):
23         try:
24             rows = []
25             for item in self.data_table.get_children():
26                 row_data = self.data_table.item(item)['values']
27                 rows.append(row_data)
28
29             # Tạo DataFrame
30             df = pd.DataFrame(rows, columns=["MSSV", "Họ", "Tên"])
31             df.index = df.index + 1
32
33             # Mở hộp thoại Lưu tệp để chọn đường dẫn Lưu
34             file_path = filedialog.asksaveasfilename(defaultextension=".xlsx",
35                                                         filetypes=[
36                                                             ("Excel files", "*.xlsx"), ("All files", "*.*")],
37                                                         title="Chọn nơi lưu file")
38
39             if file_path: # Nếu người dùng không hủy
40                 # Xuất ra file Excel
41                 df.to_excel(file_path, index=True)
42                 messagebox.showinfo(
43                     "Xuất file", "Dữ liệu đã được xuất ra file Excel thành công!")
44             else:
45                 messagebox.showwarning("Hủy", "Bạn đã hủy lưu file.")
46
47         except Exception as e:
48             messagebox.showwarning("Warning", f"Có lỗi xảy ra: {str(e)}")
49
50 if __name__ == "__main__":
51     root = tk.Tk()
52     TableApp(root)
53     root.mainloop()
54

```



## **4 Github**

[https://github.com/Nonope2/AdvancePython\\_Project.git](https://github.com/Nonope2/AdvancePython_Project.git)