

TRƯỜNG ĐẠI HỌC VĂN LANG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CUỐI KỲ
MÔN HỌC LẬP TRÌNH PYTHON NÂNG CAO

Bài tập 03:

Website Quản lý sinh viên

SVTH: Trần Ngọc Phước – 2274802010699

LỚP: 241_71ITSE31003_02

GVHD: Huỳnh Thái Học

TP. Hồ Chí Minh – 11/2024

MỤC LỤC

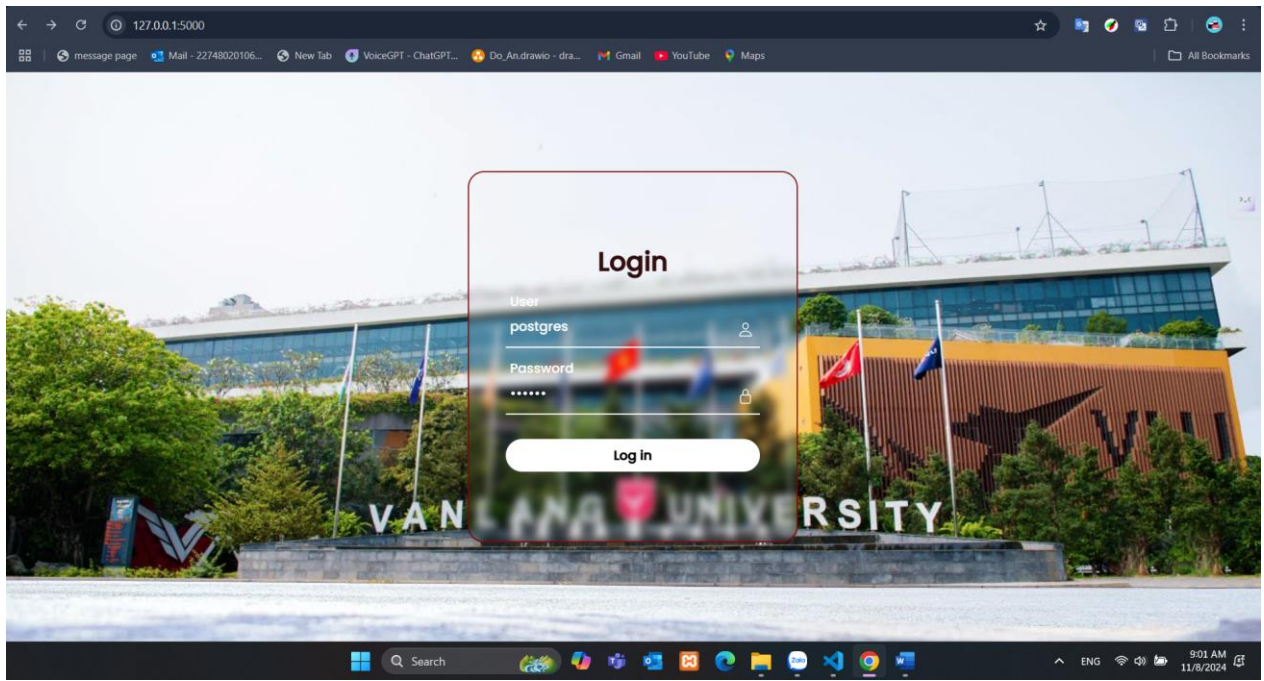
1. Giới thiệu về flask.....	3
2. Giao diện Website Quản lý sinh viên.	4
3. Chức năng.....	6
3.1 Login vào cơ sở dữ liệu:	6
3.2 Xóa dữ liệu sinh viên:	6
3.3 Thêm sinh viên mới vào cơ sở dữ liệu:	6
3.4 Tìm kiếm sinh viên dựa vào mssv:.....	6
4. Mã chương trình.	7
5. Đường link github.	12
--- HẾT---	12

1. Giới thiệu về flask.

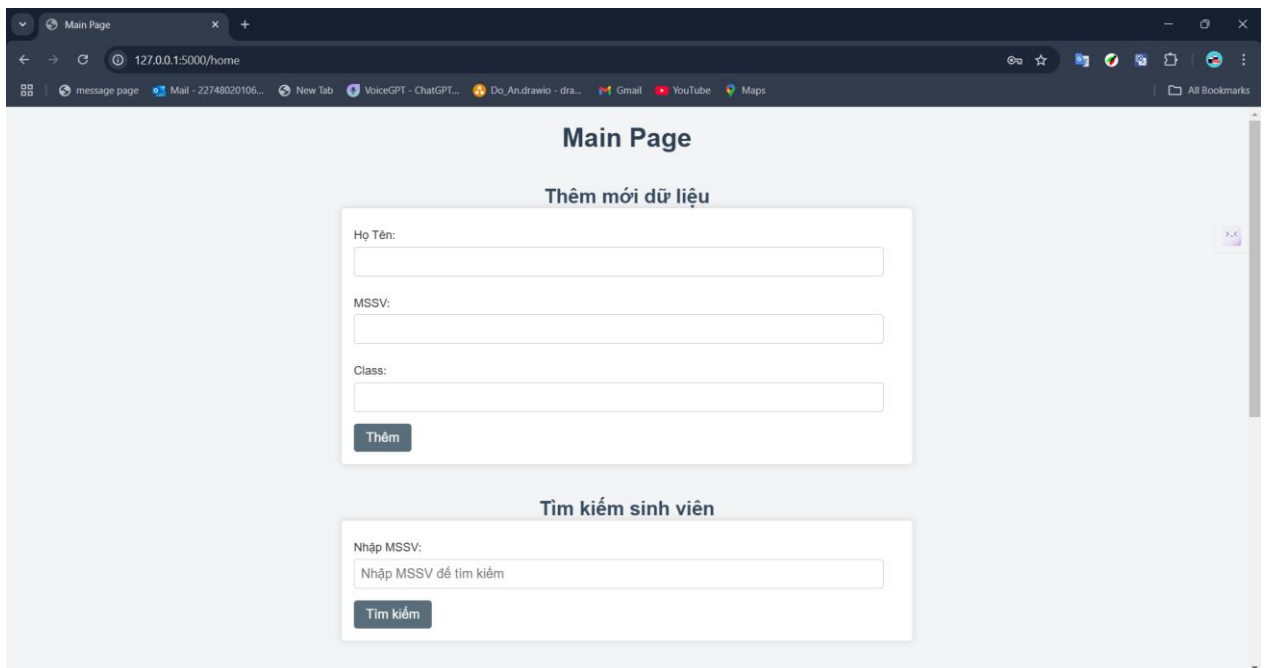
Flask là một microframework phổ biến trong Python, được thiết kế để phát triển các ứng dụng web nhanh chóng và dễ dàng. Ra mắt lần đầu vào năm 2010 bởi Armin Ronacher, Flask nổi bật với triết lý “đơn giản nhưng mở rộng”, mang lại cho các nhà phát triển sự linh hoạt trong việc lựa chọn các công cụ và thư viện tùy ý mà không bị ràng buộc vào cấu trúc cứng nhắc. Với thiết kế nhẹ và tối giản, Flask không đi kèm với các tính năng như ORM hay xác thực, giúp người dùng dễ dàng kiểm soát và mở rộng theo nhu cầu dự án. Flask cũng hỗ trợ hệ thống định tuyến, hệ thống mẫu Jinja2, và khả năng tích hợp dễ dàng với các tiện ích mở rộng, làm cho nó trở thành lựa chọn lý tưởng cho cả các dự án nhỏ và các ứng dụng web phức tạp.

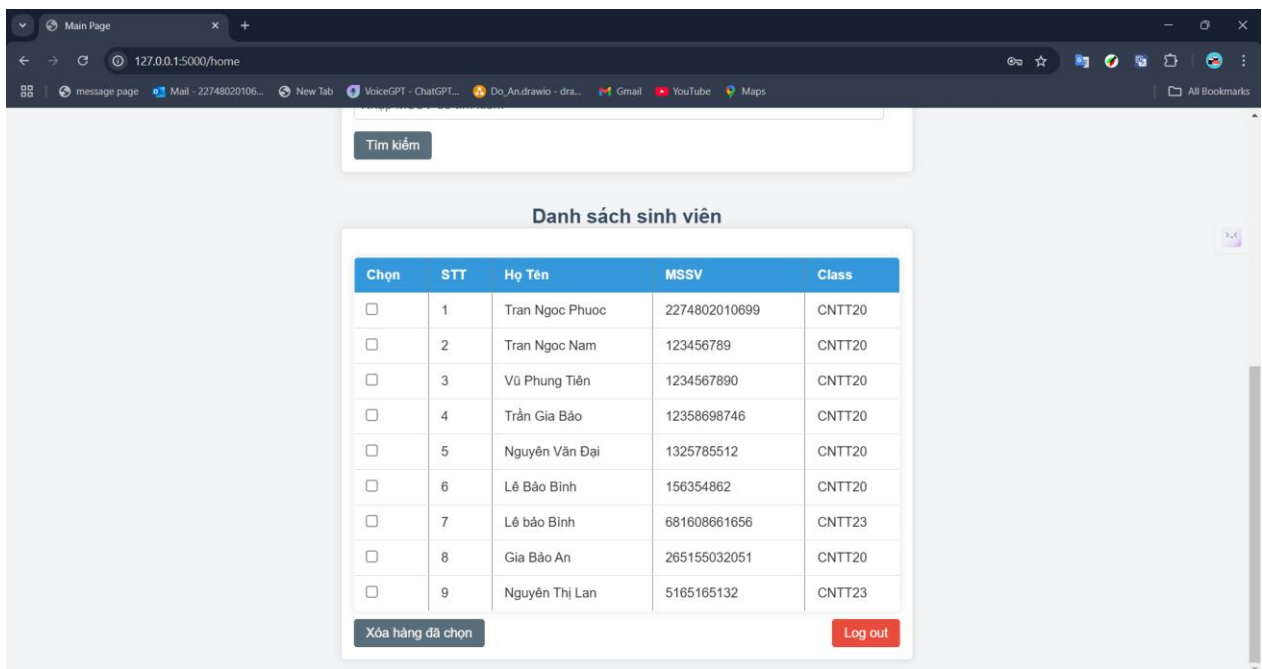
2. Giao diện Website Quản lý sinh viên.

Giao diện để login vào Website

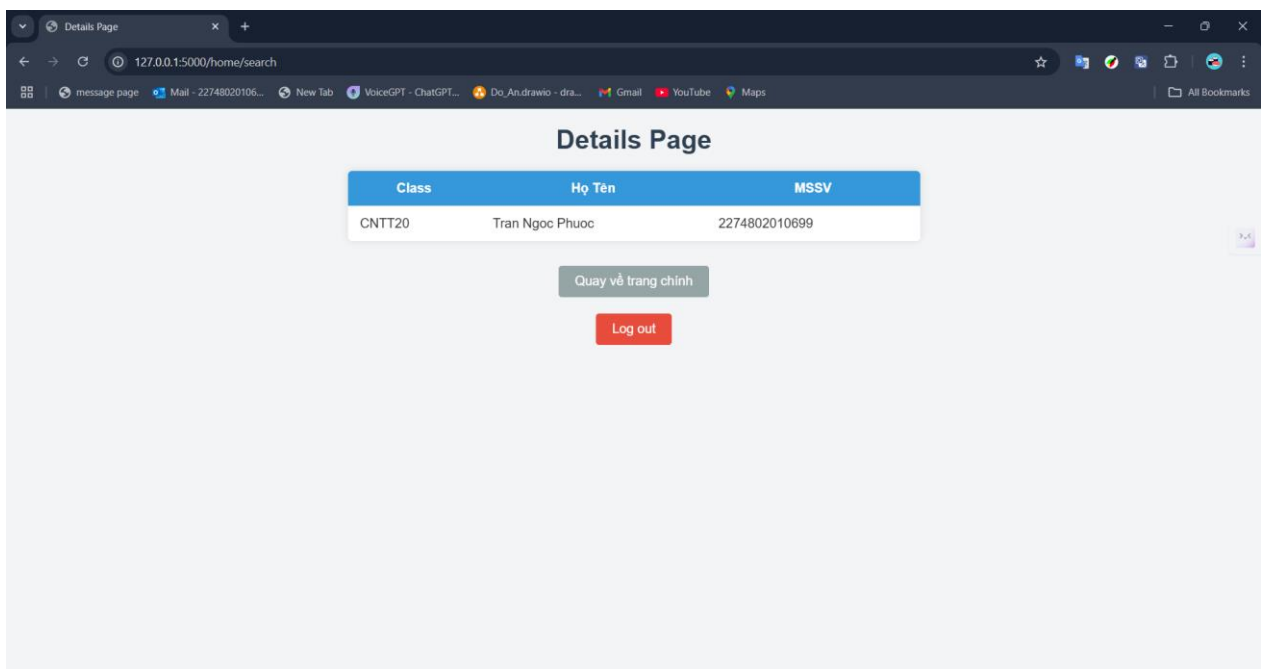


Giao diện khi login thành công vào website





Giao diện khi tìm kiếm sinh viên dựa vào mssv



Và khi người dùng nhấn **“Log out”** thì sẽ quay lại trang login

3.Chức năng.

Website Quản lý sinh viên được xây dựng bằng Python và thư viện flask có các chức năng chính sau:

3.1 Login vào cơ sở dữ liệu:

- Người dùng nhập user: “postgres”
- Người dùng nhập Password: “123456”
- Phần này người dùng cung cấp dữ liệu đầu vào là user và Pass và nhấn nút “Log in”, website sẽ tải toàn bộ danh sách sinh viên có trong cơ sở dữ liệu và hiển thị vào trang “Main Page”

3.2 Xóa dữ liệu sinh viên:

- Người dùng chọn vào hàng dữ liệu cần xóa “Chọn”
- Phần này người dùng đã chọn “vào hàng dữ liệu cần xóa” và nhấn nút “Xóa hàng đã chọn”, Website sẽ xóa dữ liệu sinh viên đó.

3.3 Thêm sinh viên mới vào cơ sở dữ liệu:

- Người dùng nhập tên, mssv và lớp vào ba ô nhập liệu tương ứng.
- Phần này người dùng cung cấp dữ liệu đầu vào là “tên sinh viên mới”, “MSSV” và “Mã lớp học” khi nhấn nút “Thêm”, ứng dụng sẽ thêm sinh viên mới đó vào mã lớp học tương ứng.

3.4 Tìm kiếm sinh viên dựa vào mssv:

- Người dùng nhập mssv vào ô nhập liệu tương ứng để tìm kiếm.
- Phần này người dùng cung cấp dữ liệu đầu vào là “MSSV” khi nhấn nút “Tìm kiếm”, ứng dụng sẽ hiển thị sinh viên có mssv tương ứng

4. Mã chương trình.

Về giao diện:

- Trang Index.html để phục vụ cho việc Login

```
app.py 4. U  index.html U  home.html U  details.html U  database.py U  index.css U
Do_An_Cuoi_Ky > project_01 > test_03 > templates > index.html > html > body > section > div.form-box > div.form-value > form > p.messages
2 <html lang="en">
9 <body>
10 {% block content %}
11 <section>
12
13     <div class="form-box">
14         <div class="form-value">
15             <form action="#" method="post">
16                 <h2>Login</h2>
17                 {% with messages = get_flashed_messages() %}
18                     {% if messages %}
19                         {% for m in messages %}
20                             <p class="messages">{{ m|safe }}</p>
21                         {% endfor %}
22                     {% endif %}
23                 {% endwith %}
24                 <div class="inputbox">
25                     <ion-icon name="person-outline"></ion-icon>
26                     <input type="text" name="user" required> </input>
27                     <label for="">User</label>
28                 </div>
29                 <div class="inputbox">
30                     <ion-icon name="lock-closed-outline"></ion-icon>
31                     <input type="password" name="password" required> </input>
32                     <label for="">Password</label>
33                 </div>
34                 <button>Log in</button>
35             </form>
36         </div>
37     </div>
38 </section>
39 {% endblock %}
```

- Trang home.html để phục vụ cho việc hiển thị cơ sở dữ liệu và chức năng thêm sinh viên mới

```
<body>
<h1>Main Page</h1>
<!-- Hiển thị thông báo -->
{% with messages = get_flashed_messages(with_categories=true) %}
{% if messages %}
{% for category, m in messages %}
<p class="{{ category }}">{{ m }}</p>
{% endfor %}
{% endif %}
{% endwith %}

<!-- Form thêm mới dữ liệu -->
<h2>Thêm mới dữ liệu</h2>
<form action="{{ url_for('add_data') }}" method="post">
    <label for="name">Họ Tên:</label>
    <input type="text" name="name" required>
    <label for="mssv">MSSV:</label>
    <input type="text" name="mssv" required>
    <label for="class">Class:</label>
    <input type="text" name="class" required>
    <button type="submit">Thêm</button>
</form>

<!-- Form tìm kiếm sinh viên -->
<h2>Tìm kiếm sinh viên</h2>
<form action="{{ url_for('search_data') }}" method="post">
    <label for="mssv">Nhập MSSV:</label>
    <input type="text" name="mssv" placeholder="Nhập MSSV để tìm kiếm" required>
    <button type="submit">Tìm kiếm</button>
</form>
```

```

<h2>Danh sách sinh viên</h2>
<form action="{{ url_for('delete_data') }}" method="post">
    <table border="1" cellpadding="5" cellspacing="0">
        <thead>
            <tr>
                <th>Chọn</th>
                <th>STT</th>
                <th>Họ Tên</th>
                <th>MSSV</th>
                <th>Class</th>
            </tr>
        </thead>
        <tbody>
            {% for row in data %}
            <tr>
                <td><input type="checkbox" name="selected" value="{{ row[0] }}"></td>
                <td>{{ loop.index }}</td>
                <td>{{ row[1] }}</td>
                <td>{{ row[2] }}</td>
                <td>{{ row[3] }}</td>
            </tr>
            {% endfor %}
        </tbody>
    </table>
    <div class="form-buttons">
        <button type="submit" class="delete">Xóa hàng đã chọn</button>
        <a href="{{ url_for('login') }}">Log out</a>
    </div>

```

- Trang details.html để phục vụ cho việc tìm kiếm sinh viên theo mssv

```

100 <body>
101     {% with messages = get_flashed_messages() %}
102     {% if messages %}
103     {% for m in messages %}
104     <p class="message">{{ m }}</p>
105     {% endfor %}
106     {% endif %}
107     {% endwith %}
108
109     <h1>Details Page</h1>
110
111     <table>
112     <tr>
113         <th>Class</th>
114         <th>Họ Tên</th>
115         <th>MSSV</th>
116     </tr>
117     {% for row in data %}
118     <tr>
119         <td>{{ row[3] }}</td> <!-- Class (Cột thứ 4 trong bảng khoacntt) -->
120         <td>{{ row[1] }}</td> <!-- Name (Cột thứ 2 trong bảng khoacntt) -->
121         <td>{{ row[2] }}</td> <!-- MSSV (Cột thứ 3 trong bảng khoacntt) -->
122     </tr>
123     {% endfor %}
124     </table>
125
126     <a href="{{ url_for('home') }}" class="back-link">Quay về trang chính</a>
127     <a href="{{ url_for('login') }}">Log out</a>
128 </body>

```


Về các chức năng chính:

- Kết nối vào database

```
app.py 4, U | index.html U | home.html U | details.html U | database.py 1, U X | index.css U
Do_An_Cuoi_Ky > project_01 > test_03 > database.py > ...
1 # db_connection.py
2 import psycopg2
3
4 class Database:
5     def __init__(self, db_name, user, password, host, port):
6         self.db_name = db_name
7         self.user = user
8         self.password = password
9         self.host = host
10        self.port = port
11
12    def connect(self):
13        try:
14            self.conn = psycopg2.connect(
15                dbname=self.db_name,
16                user=self.user,
17                password=self.password,
18                host=self.host,
19                port=self.port
20            )
21            self.cur = self.conn.cursor()
22            return True
23        except Exception as e:
24            print(f"Error connecting to the database: {e}")
25            return False
26
27
28
```

```
app.py 4, M X | home.html M | details.html M
Do_An_Cuoi_Ky_Ly_Thuyet > project_01 > BaiTap03 > app.py > home
1 from flask import Flask, render_template, request, redirect, url_for, flash, session, make_response
2 import psycopg2
3 from psycopg2 import sql
4 from database import Database
5
6 app = Flask(__name__)
7 app.secret_key = 'your_secret_key' # Bảo mật session
8
9
10 # Trang đăng nhập
11 @app.route('/', methods=['GET', 'POST'])
12 def login():
13     if request.method == 'POST':
14         user = request.form['user']
15         password = request.form['password']
16         db = Database(db_name='quanlysinhvien', user=user, password=password, host='localhost', port=5432)
17         if db.connect():
18             session['user'] = user
19             session['password'] = password
20             db.cur.close()
21             db.conn.close()
22             return redirect(url_for('home'))
23         else:
24             flash("Kết nối không thành công. Vui lòng kiểm tra thông tin đăng nhập.", 'error')
25         return render_template('index.html')
26
```

• Load Data

```
28 # Trang chính
29 @app.route('/home', methods=['GET', 'POST'])
30 def home():
31     if 'user' in session and 'password' in session:
32         db = Database(db_name='quanlysinhvien', user=session['user'], password=session['password'], host='localhost', port=5432)
33         if db.connect():
34             db.cur.execute('SELECT * FROM khoacontt ORDER BY id')
35             data = db.cur.fetchall()
36             db.cur.close()
37             db.conn.close()
38             return render_template('home.html', data=data)
39     flash("Bạn cần đăng nhập để truy cập trang này.", 'error')
40     return redirect(url_for('login'))
```

• Insert data

```
app.py 4, M x home.html M details.html M
Do_An_Cuoi_Ky_Ly_Thuyet > project_01 > BaiTap03 > app.py > home
43 # them dữ liệu mới
44 @app.route('/home/add', methods=['POST'])
45 def add_data():
46     if 'user' in session and 'password' in session:
47         new_name = request.form.get('name', '').strip()
48         new_mssv = request.form.get('mssv', '').strip()
49         new_class = request.form.get('class', '').strip()
50
51         if not new_name or not new_mssv or not new_class:
52             flash("Tất cả các trường (name, mssv, class) đều bắt buộc.", 'error')
53             return redirect(url_for('home'))
54
55         db = Database(db_name='quanlysinhvien', user=session['user'], password=session['password'], host='localhost', port=5432)
56         if db.connect():
57             try:
58                 # Kiểm tra mssv trùng lặp
59                 db.cur.execute("SELECT COUNT(*) FROM khoacontt WHERE mssv = %s", (new_mssv,))
60                 if db.cur.fetchone()[0] > 0:
61                     flash("MSSV đã tồn tại, vui lòng nhập MSSV khác.", 'error')
62                 else:
63                     # Thêm dữ liệu mới
64                     db.cur.execute("INSERT INTO khoacontt (name, mssv, class) VALUES (%s, %s, %s)",
65                                   (new_name, new_mssv, new_class))
66                     db.conn.commit()
67                     flash("Thêm dữ liệu thành công.", 'success')
68             except Exception as e:
69                 flash(f"Lỗi khi thêm dữ liệu: {e}", 'error')
70             finally:
71                 db.cur.close()
72                 db.conn.close()
73     return redirect(url_for('home'))
```

- Search data

```
app.py 4, M x home.html M details.html M
Do_An_Cuoi_Ky_Ly_Thuyet > project_01 > Baitap03 > app.py > home
99
100 # Tìm kiếm sinh viên theo MSSV
101 @app.route('/home/search', methods=['POST'])
102 def search_data():
103     if 'user' in session and 'password' in session:
104         mssv = request.form['mssv'] # Lấy MSSV từ form tìm kiếm (parameter) host: str
105         db = Database(db_name='quanlysinhvien', user=session['user'], password=session['password'], host='localhost', port=5432)
106         if db.connect():
107             try:
108                 # Tìm kiếm sinh viên theo MSSV
109                 query = "SELECT * FROM khoacontt WHERE mssv = %s"
110                 db.cur.execute(query, (mssv,))
111                 data = db.cur.fetchall()
112
113                 db.cur.close()
114                 db.conn.close()
115
116                 if data:
117                     # Truyền kết quả tìm kiếm vào trang details.html
118                     return render_template('details.html', data=data)
119                 else:
120                     flash("Không tìm thấy sinh viên với MSSV đã nhập.", 'error')
121             except Exception as e:
122                 flash(f"Lỗi khi tìm kiếm: {e}", 'error')
123             finally:
124                 db.cur.close()
125                 db.conn.close()
126
127         return redirect(url_for('home'))
```

- delete data

```
app.py 4, M x home.html M details.html M
Do_An_Cuoi_Ky_Ly_Thuyet > project_01 > Baitap03 > app.py > home
75
76 # Xóa dữ liệu đã chọn
77 @app.route('/home/delete', methods=['POST'])
78 def delete_data():
79     if 'user' in session and 'password' in session:
80         selected_ids = request.form.getlist('selected')
81         if not selected_ids:
82             flash("Vui lòng chọn hàng cần xóa.", 'error')
83             return redirect(url_for('home'))
84
85         db = Database(db_name='quanlysinhvien', user=session['user'], password=session['password'], host='localhost', port=5432)
86         if db.connect():
87             try:
88                 db.cur.execute(
89                     "DELETE FROM khoacontt WHERE id IN (%s)" % ','.join(map(str, selected_ids))
90                 )
91                 db.conn.commit()
92                 flash("Xóa dữ liệu thành công.", 'success')
93             except Exception as e:
94                 flash(f"Lỗi khi xóa dữ liệu: {e}", 'error')
95             finally:
96                 db.cur.close()
97                 db.conn.close()
98         return redirect(url_for('home'))
99
```

5. Đường link github.

<https://github.com/phuocsasc/PythonNC/tree/master/Baitap01>

--- HẾT ---