

## **Лабораторна робота №6**

**Тема: Мікрофреймворк Flask. Створення додатку “Список завдань”.**

**Мета роботи:** отримати навички створення Веб додатку.

Виконав: Томко Максим БСД-22

## Виконання

### Частина 1.

1. Вже маю встановлений пайтон останньої версії

```
[~ % python3 --version
Python 3.12.0
```

2. Створив папу lab6 для виконання проекту
3. Вже маю встановлений virtualenv

```
[~ % virtualenv --version
virtualenv 20.24.5 from /Lib
```

4. Створив віртуальне середовище

```
SystemExit: 2
user lab6 % virtualenv -p /usr/local/bin/python3 venv
created virtual environment CPython3.12.0.final.0-64 in 2320ms
creator CPython3Posix(dest=/Users/user/Desktop/uniukr/lab6/venv, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=bundle, via=copy, app_data_dir=/Users/user/Library/Application Support/virtual
added seed packages: pip==23.2.1
activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
user lab6 %
```

5. Активував віртуальне середовище

```
user lab6 % cd venv
user venv % cd bin
user bin % ls
activate          activate.nu      pip             pip3.12         python3.12
activate.csh      activate.ps1    pip-3.12        python          python3
activate.fish     activate_this.py pip3
user bin % source activate
(venv) user bin %
```

6. Встановив flask та базу даних

```
-----
blinker          1.7.0
click            8.1.7
Flask            3.0.0
Flask-SQLAlchemy 3.1.1
greenlet         3.0.1
itsdangerous     2.1.2
Jinja2           3.1.2
MarkupSafe       2.1.3
pip              23.2.1
SQLAlchemy       2.0.23
typing_extensions 4.8.0
Werkzeug         3.0.1
```

7. Створив файл app.py

```
(notice) To update, run: pip install --upgrade pip
(venv) user lab6 % touch app.py
(venv) user lab6 %
```

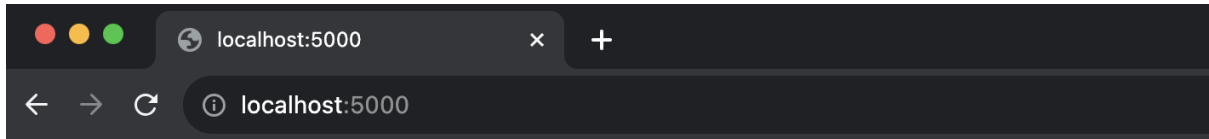
8. 9. 10. Написав шаблонний код

```
1  from flask import Flask
2
3  app = Flask(__name__)
4
5  @app.route('/')
6  def index():
7      return "Hello, world"
8
9  if __name__ == "__main__":
10     app.run(debug=True)
```

## 11. Запустив локальний сервер

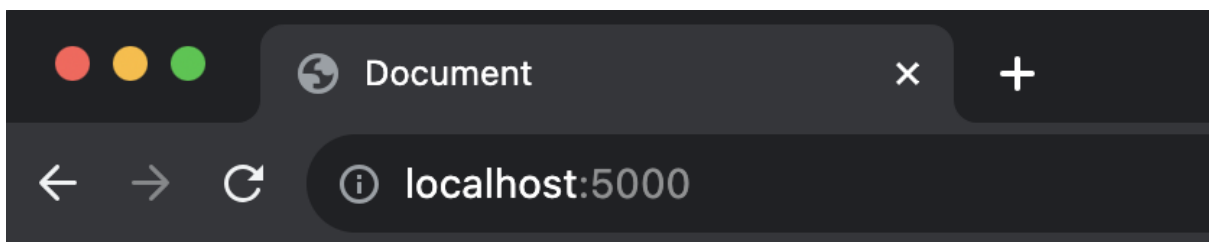
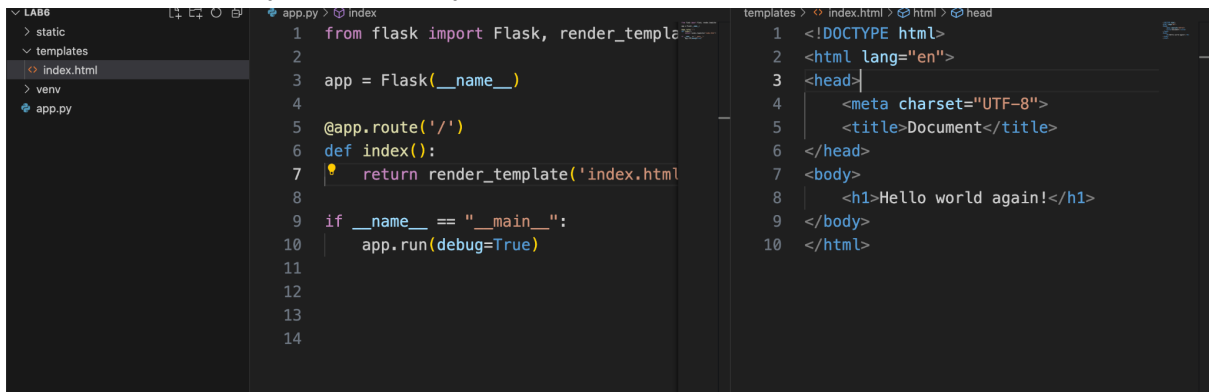
```
hello
(venv) user lab6 % python3 app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 717-068-910
127.0.0.1 - - [17/Nov/2023 16:48:01] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [17/Nov/2023 16:48:01] "GET /favicon.ico HTTP/1.1" 404 -
```

## 12. Перейшовши за адресою localhost:5000



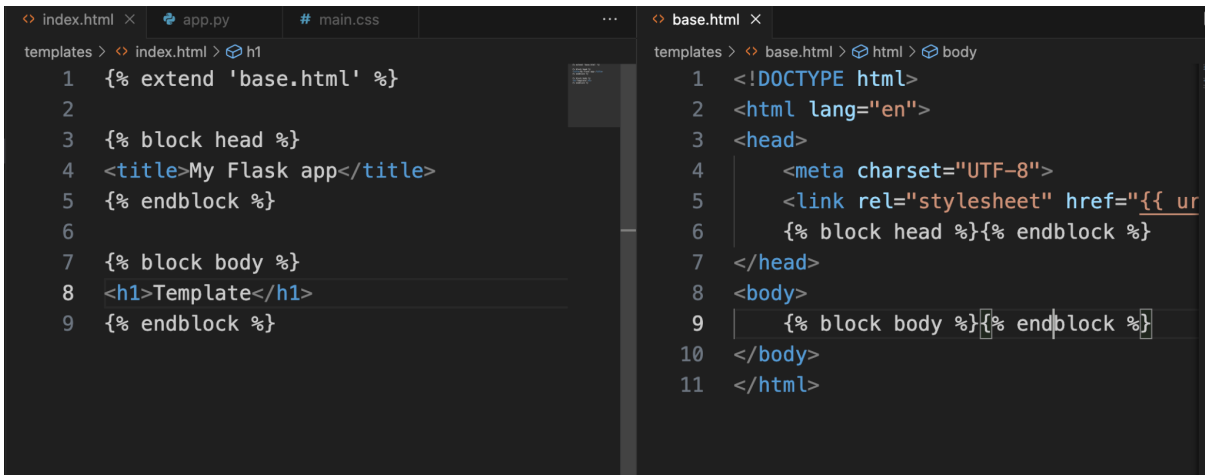
Hello, world

13.14. Створив дві папки static та templates, в останній створив файл index.html.  
Зробив деякі зміни у файлі app.py



# Hello world again!

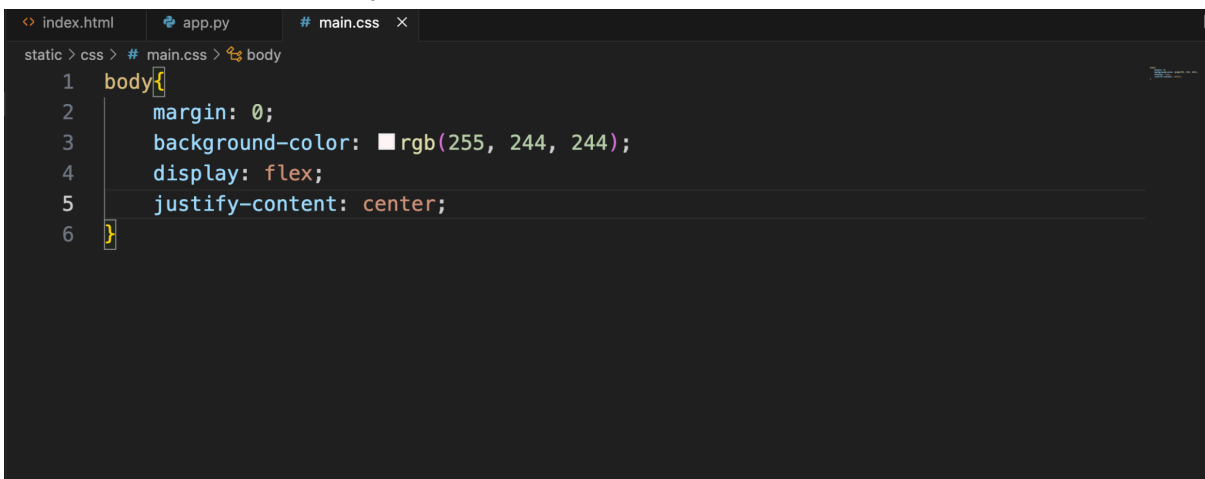
15. У папці templates , створив файл base.html , та вніс зміни у файл index.html



```
templates > index.html > h1
1 {% extend 'base.html' %}
2
3 {% block head %}
4 <title>My Flask app</title>
5 {% endblock %}
6
7 {% block body %}
8 <h1>Template</h1>
9 {% endblock %}

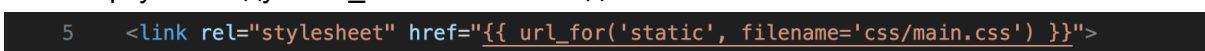
templates > base.html > html > body
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <link rel="stylesheet" href="{{ ur
6   {% block head %}{% endblock %}
7 </head>
8 <body>
9   {% block body %}{% endblock %}
10 </body>
11 </html>
```

16.Створив файл main.css у static/css



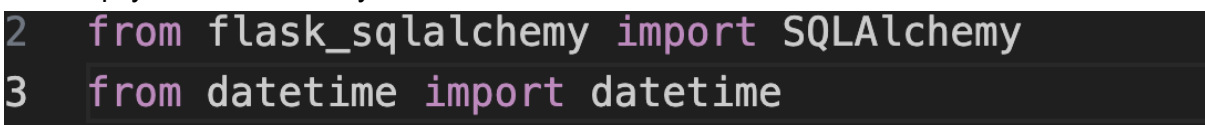
```
static > css > # main.css > body
1 body{
2   margin: 0;
3   background-color: #rgb(255, 244, 244);
4   display: flex;
5   justify-content: center;
6 }
```

17. Імпортував модуль url\_for та вніс зміни до base.html



```
5 <link rel="stylesheet" href="{{ url_for('static', filename='css/main.css') }}">
```

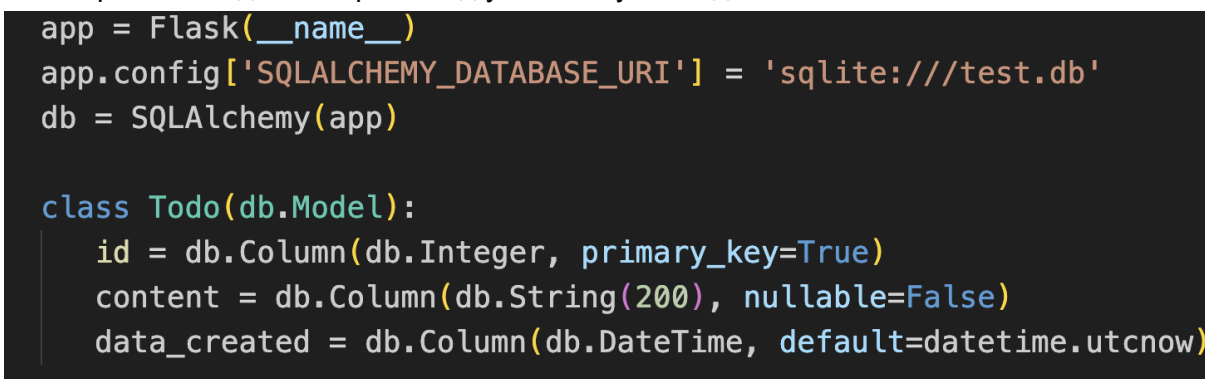
18.Імпортував SQLAlchemy та datetime



```
2 from flask_sqlalchemy import SQLAlchemy
3 from datetime import datetime
```

також та вказав шлях до бд

та створив клас для створення бд у готовому вигляді



```
app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///test.db'
db = SQLAlchemy(app)

class Todo(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    content = db.Column(db.String(200), nullable=False)
    data_created = db.Column(db.DateTime, default=datetime.utcnow)
```

```
index.html  base.html  app.py  # main.css
app.py > ...
1  from flask import Flask, render_template, url_for
2  from flask_sqlalchemy import SQLAlchemy
3  from datetime import datetime
4
5
6  app = Flask(__name__)
7  app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///test.db'
8  db = SQLAlchemy(app)
9
10 class Todo(db.Model):
11     id = db.Column(db.Integer, primary_key=True)
12     content = db.Column(db.String(200), nullable=False)
13     data_created = db.Column(db.DateTime, default=datetime.utcnow)
14     def __repr__(self) -> str:
15         return '<Task %r>' % self.id
16
17
18 @app.route('/')
19 def index():
20     return render_template('base.html')
21
22 if __name__ == "__main__":
23     app.run(debug=True)
24
25
```

19.Налаштував бд в інтерпретаторі

```
(venv) user lab6 % python3
Python 3.12.0 (v3.12.0:0fb18b02c8, Oct 2 2023, 09:45:56) [Clang 13.0.0 (clang-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> from app import db
db>>> db.create_all()
>>> exit()
(venv) user lab6 %
```

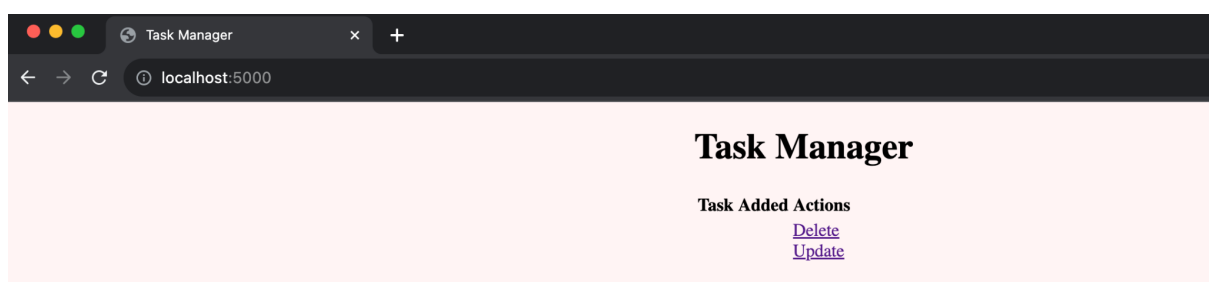
▼ instance

≡ test.db

20. Вніс зміни до файлу index.html

```
1  {% extends 'base.html' %}
2
3
4  {% block head %}
5  <title>Task Manager</title>
6  {% endblock %}
7
8
9  {% block body %}
10 <div class="content">
11   <h1>Task Manager</h1>
12   <table>
13     <tr>
14       <th>Task</th>
15       <th>Added</th>
16       <th>Actions</th>
17     </tr>
18     <tr>
19       <td></td>
20       <td></td>
21       <td>
22         <a href="">Delete</a>
23         <br>
24         <a href="">Update</a>
25       </td>
26     </tr>
27   </table>
28 </div>
29
30
31 {% endblock %}
```

запустивши проект та перейшовши по адресу localhost:5000, бачу, що все працює



21. У файлі app.py у декоратор вніс наступні зміни

```
@app.route('/', methods=['POST', 'GET'])
```

22. У файлі index.html додав тег form після таблиці

```
</table>
<form action="/" method="POST">
  <input type="text" name="content" id="content">
  <input type="submit" value="Add Task">
</form>
```

23. Імпортував request

```
from flask import Flask, render_template, url_for, request
```

та додав блок коду, який реагує на POST запити

```
if request.method == 'POST':
    return "Form submitted!"
else:
    return render_template('index.html')
```

24. Імпортував redirect

```
from flask import Flask, render_template, url_for, request, redirect
```

та додав логіку обробки в блок if

```
if request.method == 'POST':
    task_content = request.form['content']
    new_task = Todo(content=task_content)

    try:
        db.session.add(new_task)
        db.session.commit()
        return redirect('/')
    except:
        return 'There was an issue adding your task'

else:
    tasks = Todo.query.order_by(Todo.data_created).all()
    return render_template('index.html', tasks=tasks)
```

25.Вніс зміни до файлу index.html

```
{% for task in tasks %}
    <tr>
        <td>{{ task.content }}</td>
        <td>{{ task.data_created.date() }}</td>
        <td>
            <a href="">Delete</a>
            <br>
            <a href="">Update</a>
        </td>
    </tr>
{% endfor %}
```

26. Перевіряючи роботу додав кілька завдань

## Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити у спорт зал	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>

Add Task



## Частина 2.

### 27. Оновив файл main.css

```
body, html {
  margin: 0;
  font-family: sans-serif;
  background-color: #lightblue;
}

h1{
  text-align: center;
}

.content {
  margin: 0 auto;
  width: 400px;
}

table, td, th {
  border: 1px solid #aaa;
}

table {
  border-collapse: collapse;
  width: 100%;
}

th {
  height: 30px;
}

td {
  text-align: center;
  padding: 5px;
}

.form {
  margin-top: 20px;
}

#content {
  width: 70%;
}
```

## Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити у спорт зал	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
<input type="text"/>		<input type="button" value="Add Task"/>

28. Додав можливість видалення задач створивши новий роут

```
@app.route('/delete/<int:id>')
def delete(id):
    task_to_delete = Todo.query.get_or_404(id)

    try:
        db.session.delete(task_to_delete)
        db.session.commit()
        return redirect('/')
    except:
        return "Couldn't delete that task."
```

29. Додав посилання на id елемента, який я хочу видалити

```
<a href="/delete/{{ task.id }}">Delete</a>
```

30. Видалення не працює, знайшов та виправив помилку на 42 рядку comit -> commit

```
@app.route('/delete/<int:id>')
def delete(id):
    task_to_delete = Todo.query.get_or_404(id)

    try:
        db.session.delete(task_to_delete)
        db.session.commit()
        return redirect('/')
    except:
        return "Couldn't delete that task."
```

видалення працює

## Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити у спорт зал	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
<input type="text"/>	<input type="button" value="Add Task"/>	

## Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
<input type="text"/>	<input type="button" value="Add Task"/>	

31. Створив новий роут update, для зміни задач

```
@app.route('/update/<int:id>', methods=['GET', 'POST'])
def update(id):
    task = Todo.query.get_or_404(id)

    if request.method == 'POST':
        task.content = request.form['content']

        try:
            db.session.commit()
            return redirect('/')
        except:
            return "Can't update task!"
    else:
        return render_template('update.html', task=task)
```

та вніс зміни у файл index.html , додавши посилання

```
<a href="/update/{{ task.id }}">Update</a>
```

32. Створив новий файл update.html

```
{% extends 'base.html' %}

{% block head %}
<title>Task Manager</title>
{% endblock %}

{% block body %}
<div class="content">
    <h1>Task Manager | Update task</h1>

    <form action="/update/{{ task.id }}" method="POST">
        <input type="text" name="content" id="content" value="{{ task.content }}">
        <input type="submit" value="Update">
    </form>
</div>

{% endblock %}
```

# Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити за колою в магазин	2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>
<input type="text"/>		<input type="button" value="Add Task"/>

# Task Manager | Update task

# Task Manager

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити за пепси з манго в магазин	2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>
<input type="text"/>		<input type="button" value="Add Task"/>

### Частина 3. Bootstrap

Модифікував файл base.html , підключивши bootstrap

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
<link rel="stylesheet" href="{{ url_for('static', filename='css/main.css') }}">
{% block head %}{% endblock %}
</head>
<body>
    {% block body %}{% endblock %}
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js"></script>
</body>
```

Модифікував форму

```
<form action="/" method="POST" class="mb-4">
    <input type="text" name="content" id="content" class="form-control" required>
    <button type="submit" class="btn btn-primary">Add Task</button>
</form>
```

щоб не було такого

	Май 13	10	<a href="#">Update</a>
		2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>
		2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>
		2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>

Add Task

Task	Added	Actions
Захистити Лабораторні роботи	2023-11-17	<a href="#">Delete</a> <a href="#">Update</a>
Сходити на ...	2023-11-18	<a href="#">Delete</a> <a href="#">Update</a>

Add Task

також додав нових кнопок

```
<td>{{ task.content }}</td>
<td>{{ task.data_created.date() }}</td>
<td>
    <a href="/delete/{{ task.id }}" class="btn btn-danger">Delete</a>
    <a href="/update/{{ task.id }}" class="btn btn-warning">Update</a>
</td>
</tr>
</table>
```

# Task Manager

Task	Added	Actions
Сходити за пепси з манго в магазин	2023- 11-18	<div>Delete</div> <div>Update</div>

Add Task

та інші косметичні зміни

# Task Manager

Task	Added	Actions
Сходити за пепси з манго в магазин	2023- 11-18	<div>Delete</div> <div>Update</div>

Add Task