



Todo list app using Flask | Python

Last Updated : 15 May, 2020

There are many frameworks that allow building your webpage using Python, like Django, flask, etc. Flask is a web application framework written in Python. Flask is based on WSGI(Web Server Gateway Interface) toolkit and Jinja2 template engine. Its modules and libraries that help the developer to write applications without writing the low-level codes such as protocols, thread management, etc.

In this article, we will learn how to make a todo list app using the Flask framework. In this app, you can add your todo items and mark them as complete or incomplete.

Installation:

```
pip install Flask
```

Basic setup :

Step 1: First make basic folders

```
mkdir app && cd app && mkdir static && mkdir templates
```

Step 2: Make some basic python files to write the code and name it as you desire.

Step 3: Run the below command to start the server

```
touch run.py the app
```

Step 4: Change directory to *app*–

```
cd app
```

Step 5: create models.py for database, routes.py for urls/views and __init__ file to package our app

```
touch models.py routes.py __init__.py
```

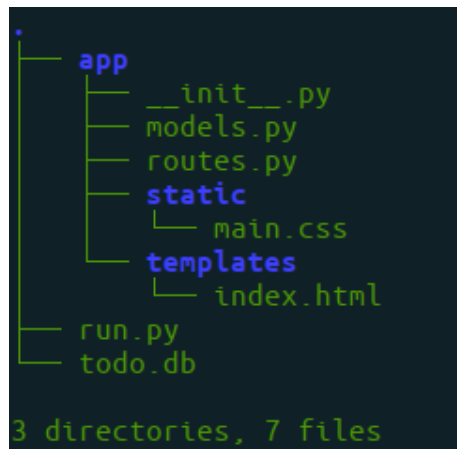
Step 6: Goto *templates/* directory and create index.html file

```
cd templates && touch index.html
```

Step 7: Goto *static/* directory and create main.css

```
cd static && touch main.css
```

Now, open the project folder using a text editor. The directory structure should look like this :



run.py file

```
from app import app

if __name__ == '__main__':
    app.run(debug=True)
```

app/__init__.py file

```
from flask import Flask
from flask_sqlalchemy import SQLAlchemy
import os

file_path = os.path.abspath(os.getcwd())+"/todo.db"

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

```
from app import routes
```

app/routes.py file

```
from flask import render_template, request, redirect, url_for
from app import app
from app.models import Todo
from app import db

@app.route('/')
def index():
    incomplete = Todo.query.filter_by(complete=False).all()
    complete = Todo.query.filter_by(complete=True).all()

    return render_template('index.html', incomplete=incomplete, complete=complete)

@app.route('/add', methods=['POST'])
def add():
    todo = Todo(text=request.form['todoitem'], complete=False)
    db.session.add(todo)
    db.session.commit()

    return redirect(url_for('index'))

@app.route('/complete/<id>')
def complete(id):
    todo = Todo.query.filter_by(id=int(id)).first()
    todo.complete = True
    db.session.commit()

    return redirect(url_for('index'))
```

app/models.py file

```
from app import db

class Todo(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    text = db.Column(db.String(200))
    complete = db.Column(db.Boolean)
```

```
def __repr__(self):
    return self.text
```

app/main.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Todo App</title>
    <link rel="stylesheet" type="text/css" href="{{url_for('static', filename='ma

</head>
<body>
    <h1>Todo List</h1>
    <div>Add a new todo item:
        <form action="{{ url_for('add') }}" method="POST">
            <input type="text" name="todoitem">
            <input type="submit" value="Add Item" class="button">
        </form>
    </div>
    <div>
        <h2>Incomplete Items</h2>
        <ul>
            {% for todo in incomplete %}
            <li style="font-size: 30pt" class='mark'>{{ todo.text }} <a href="{{
            {% endfor %}
        </ul>
        <h2>Completed Items</h2>
        <ul>
            {% for todo in complete %}
            <li style="font-size: 30pt">{{ todo.text }}</li>
            {% endfor %}
        </ul>
    </div>
</body>
</html>
```

app/main.css

```
body{
    background:black;
    color:red;
    margin-top: 5px;
}
.button{
```

```
color:green;
}
.mark{font-size: 10px;}
```

Run the todo app using the below command

```
python run.py
```



s Somil...



5

Previous Article

Create Contact Us using WTForms in Flask

Next Article

Build a Flask application to validate CAPTCHA

Similar Reads

Documenting Flask Endpoint using Flask-Autodoc

Documentation of endpoints is an essential task in web development and being able to apply it in different frameworks is always a utility. This article discusses...

4 min read

Minify HTML in Flask using Flask-Minify

Flask offers HTML rendering as output, it is usually desired that the output HTML should be concise and it serves the purpose as well. In this article, we would...

12 min read

How to use Flask-Session in Python Flask ?

Flask Session - Flask-Session is an extension for Flask that supports Server-side Session to your application. The Session is the time between the client logs in to...

4 min read

How to make a Todo List CLI application using Python ?

In this article, we see how to make a Command Line application todo list in python. Todo list is a basic application in which users can add items. It's a list of...

7 min read

How to Integrate Flask-Admin and Flask-Login

In order to merge the admin and login pages, we can utilize a short form or any other login method that only requires the username and password. This is know...

8 min read

Flask URL Helper Function - Flask url_for()

In this article, we are going to learn about the flask url_for() function of the flask URL helper in Python. Flask is a straightforward, speedy, scalable library, used f...

11 min read

Python | ToDo GUI Application using Tkinter

Prerequisites : Introduction to tkinter Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, Tkinter is...

5 min read

ToDo webapp using Django

Django is a high-level Python Web framework-based web framework that allows rapid development and clean, pragmatic design. today we will create a todo app...

3 min read

Create a Weather app using Flask | Python

Prerequisite : Flask installation Flask is a lightweight framework written in Python. It is lightweight because it does not require particular tools or libraries...

2 min read

Create a Real Time Currency Converter app using Flask | Python

Prerequisite: installation of Python What is Flask in Python ? Flask is a popular and lightweight Python web framework, meaning it is a third-party Python libra...

8 min read

Article Tags : [Python](#)

Practice Tags : [python](#)



Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305



Company

[About Us](#)
[Legal](#)
[In Media](#)
[Contact Us](#)
[Advertise with us](#)
[GFG Corporate Solution](#)
[Placement Training Program](#)
[GeeksforGeeks Community](#)

Languages

[Python](#)
[Java](#)
[C++](#)
[PHP](#)
[GoLang](#)
[SQL](#)
[R Language](#)
[Android Tutorial](#)
[Tutorials Archive](#)

DSA

- Data Structures
- Algorithms
- DSA for Beginners
- Basic DSA Problems
- DSA Roadmap
- Top 100 DSA Interview Problems
- DSA Roadmap by Sandeep Jain
- All Cheat Sheets

Web Technologies

- HTML
- CSS
- JavaScript
- TypeScript
- ReactJS
- NextJS
- Bootstrap
- Web Design

Computer Science

- Operating Systems
- Computer Network
- Database Management System
- Software Engineering
- Digital Logic Design
- Engineering Maths
- Software Development
- Software Testing

System Design

- High Level Design
- Low Level Design
- UML Diagrams
- Interview Guide
- Design Patterns
- OOAD
- System Design Bootcamp
- Interview Questions

School Subjects

- Mathematics
- Physics
- Chemistry
- Biology
- Social Science
- English Grammar

Data Science & ML

- Data Science With Python
- Data Science For Beginner
- Machine Learning
- ML Maths
- Data Visualisation
- Pandas
- NumPy
- NLP
- Deep Learning

Python Tutorial

- Python Programming Examples
- Python Projects
- Python Tkinter
- Web Scraping
- OpenCV Tutorial
- Python Interview Question
- Django

DevOps

- Git
- Linux
- AWS
- Docker
- Kubernetes
- Azure
- GCP
- DevOps Roadmap

Interview Preparation

- Competitive Programming
- Top DS or Algo for CP
- Company-Wise Recruitment Process
- Company-Wise Preparation
- Aptitude Preparation
- Puzzles

GeeksforGeeks Videos

- DSA
- Python
- Java
- C++
- Web Development
- Data Science

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved