

Workshop: Using Flask & HTML

@ UNC Charlotte
October 19th, 2023

{clt/hack}

By Ethan Nguyen & Simone Tezzo

Agenda

- Setting up the Development Environment (10 min)
- Introduction to Flask (10 min)
- Basic Flask Application (15 min)
- Incorporating HTML into Flask (20 min)
- Projects
- Conclusion and Next Steps (5 min)

Resources



<https://github.com/ThatE10/charlotte-hack/tree/main>

Installation

Python
PyCharm
Flask

Go to <https://www.python.org/downloads>

Go to <https://www.jetbrains.com/pycharm/download/?section=windows>

Run `$ pip install flask`

OR Mac/Linux

`$ pip3 install flask`

If you have any questions on please Google it or Ask ChatGPT first

Intro to Flask

What is Flask?

- Flask is a web framework for Python.
- Lightweight, simple, and easy to use.

Key Features of Flask

- Minimalistic design.
- Flexible and extensible.
- Ideal for small to medium-sized web applications.

Common Uses of Flask

Building web applications and websites.

Creating RESTful APIs.

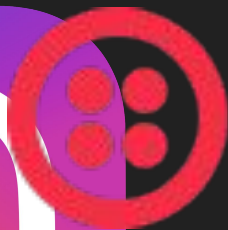
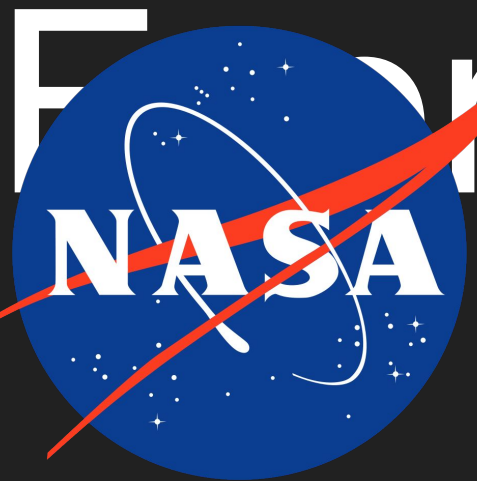
Prototyping and quick development.

Integrating with other Python libraries.

Microservices architecture.

Flas

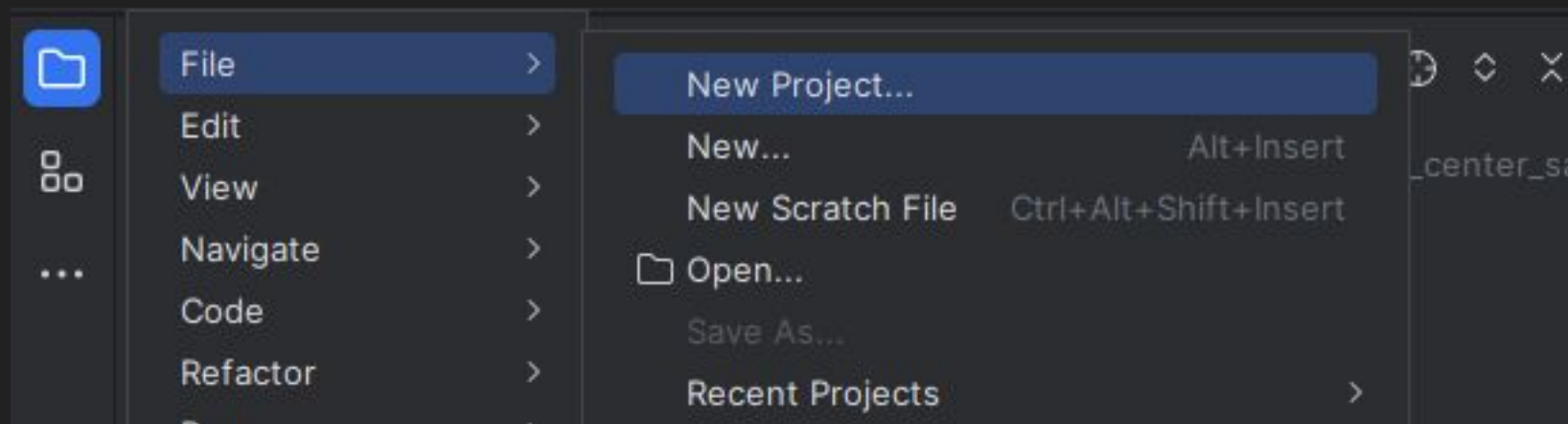
NETFLIX



twilio



Creating a Basic Project



```
11 # Function to convert text to
```



Create Project



Location: C:\Users\Ethan Nguyen\PycharmProjects\flask_app



Python Interpreter: New Virtualenv environment

☒ New environment using Virtualenv

Location: C:\Users\Ethan Nguyen\PycharmProjects\flask_app\venv



Base interpreter: Python 3.11 C:\Program Files\Python311\python.exe



☐ Inherit global site-packages

☐ Make available to all projects

☐ Previously configured interpreter


Interpreter: Python 3.11 C:\Program Files\Python311\python.exe



Create


Create the following folder structure

```
flask_app/  
├── main.py  
├── template/  
│   └── index.html  
└── static/  
    └── styles.css
```

 main.py ×

```
1 import flask
```

```
2 |
```

 main.py x

1 `import flask`

2

1 usage

3

`def print_`

4

`# Use a breakpoint in the code line below to debug your script.`



`print(f'Hi, {name}')` # Press Ctrl+F8 to toggle the breakpoint.

No module named 'flask'



[Install package flask](#)

Alt+Shift+Enter

[More actions...](#)

Alt+Enter



```
from flask import Flask, render_template

app = Flask(__name__)

# Define a route for the root URL ("/") and render an HTML template
@app.route('/')
def home():
    return render_template('index.html')

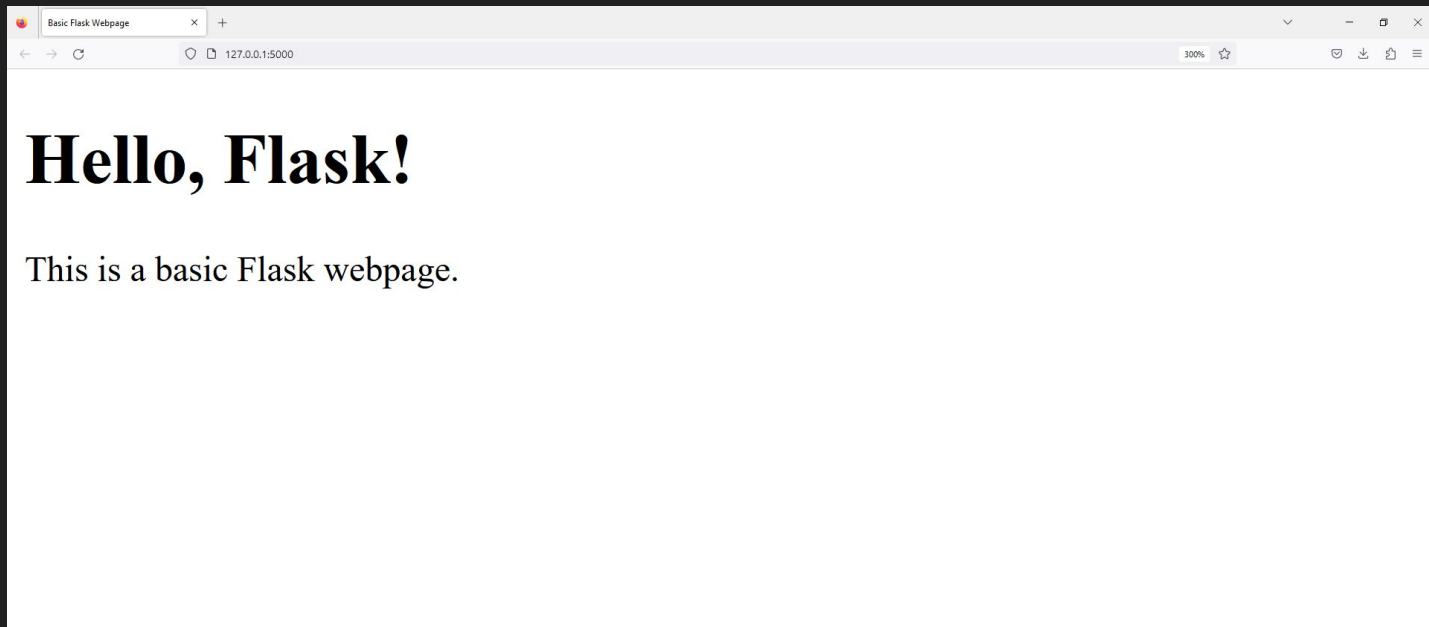
if __name__ == '__main__':
    app.run(debug=True)
```

index.html



```
<!DOCTYPE html>
<html>
<head>
  <title>Basic Flask Webpage</title>
</head>
<body>
  <h1>Hello, World!</h1>
  <p>This is a basic Flask webpage.</p>
</body>
</html>
```

Run it!



styles.css



```
/* styles.css */  
body {  
    background-color: lightgray;  
}  
  
h1 {  
    color: blue;  
}
```

Adding Logic

Run it!



Running code in HTML

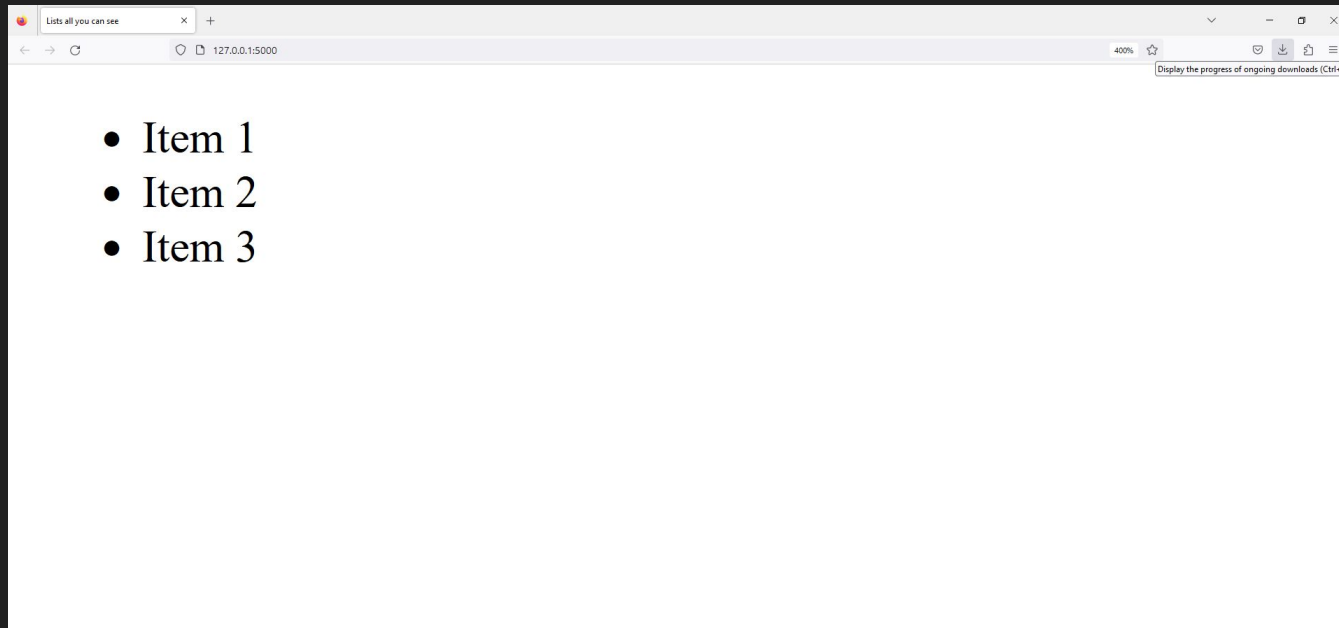


```
def home():  
    items = ["Item 1", "Item 2", "Item 3"]  
    return render_template('items.html', items=items)
```




```
<h1>List of Items</h1>  
    <ul>  
        {% for item in items %}  
            <li>{{ item }}</li>  
        {% endfor %}  
    </ul>
```

Run it!



Using Forms in Flask



```
<form method="POST" action="{{ url_for('generate_story') }}">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required><br>

  <label for="age">Age:</label>
  <input type="number" id="age" name="age" required><br>

  <label for="job_title">Future Job Title:</label>
  <input type="text" id="job_title" name="job_title" required><br>

  <input type="submit" value="Generate Story">
</form>
```

Using Forms in Flask



```
@app.route('/generate_story', methods=['POST'])
def generate_story():
    name = request.form.get('name')
    age = request.form.get('age')
    job_title = request.form.get('job_title')

    story = f"Once upon a time, {name}, at the age of {age}, dreamed of becoming a {job_title}."


    return render_template('story.html', story=story)
```

Updating Page Info



```
<h1>Search Page</h1>
  <form method="GET" action="{{ url_for('search_results') }}">
    <input type="text" name="query" placeholder="Enter your search query" required>
    <button type="submit">Search</button>
  </form>
  {% if results %}
  <div id="results">
    <h2>Search Results</h2>
    <ul>
      {% for result in results %}
        <li>{{ result }}</li>
      {% endfor %}
    </ul>
  </div>
  {% endif %}
```

Updating a webpage



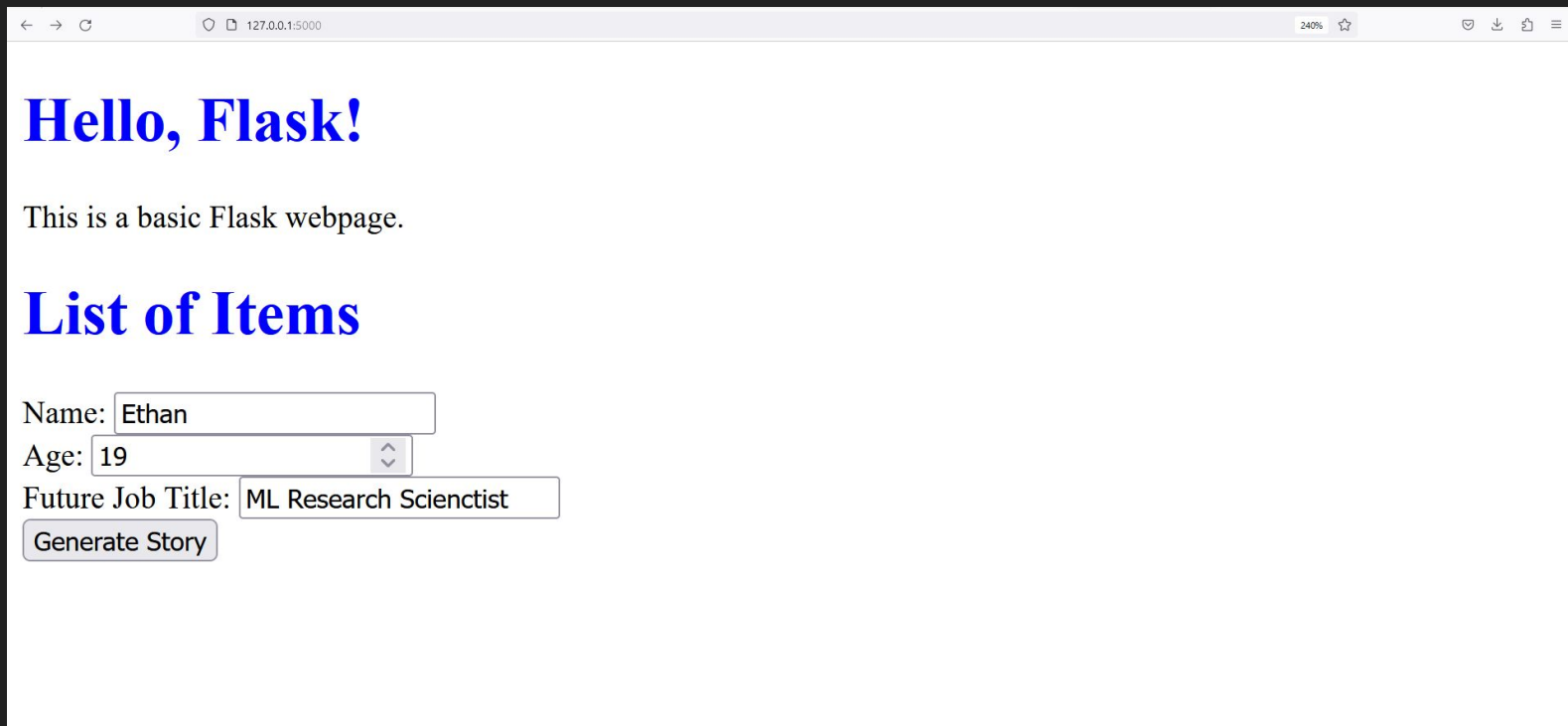
```
@app.route('/search')
def search():
    return render_template('search.html', results=None)

@app.route('/search_results', methods=['GET'])
def search_results():
    query = request.args.get('query')

    # Perform your search logic here and populate the 'results' variable with search results.
    # For this example, we'll assume 'results' is a list of search results.
    results = ['Result 1', 'Result 2', 'Result 3']
    results.append(str(query))# Replace with your actual search results.

    return render_template('search.html', results=results)
```

Results



A screenshot of a web browser window displaying a Flask application. The browser's address bar shows the URL `127.0.0.1:5000`. The page content includes a large blue heading "Hello, Flask!", a paragraph "This is a basic Flask webpage.", and another large blue heading "List of Items". Below these, there is a form with three input fields: "Name:" with the value "Ethan", "Age:" with the value "19" and a dropdown arrow, and "Future Job Title:" with the value "ML Research Scientist". At the bottom of the form is a button labeled "Generate Story".

← → ↻ 127.0.0.1:5000 240% ☆

Hello, Flask!

This is a basic Flask webpage.

List of Items

Name:

Age: ▾

Future Job Title:

Your Story

Once upon a time, Ethan, at the age of 19, dreamed of becoming a ML Research Scientist.

Projects

[Geeks for Geeks](#)

[Pythonista](#)

Try to using GPT-4V to copy web pages

Upcoming Hackathons @ Charlotte!



CENTER FOR
ENTREPRENEURSHIP
AND INNOVATION

GOLDEN HACK **IMPACT WEEKEND**

OCT 20-22

ENTREPRENEURSHIP.CHARLOTTE.EDU