

# Web Frameworks

---

JULY 11, 2022

# Learning Objectives


---

At the end of this lesson, you should be able to:

- Define a web framework
- Build a simple Flask webpage
- Create a simple template


# What is a web Framework

---

- A software "framework" supporting the development of dynamic web sites, services and APIs
  - Some examples: Django, Flask, and Bottle to name a few
  - Pros and cons for each, need to select which one to use based on level of support, ability for integration and customization in your project/technology
- 


# Properties of an Ideal Web Framework

---

- Clean – clutter free, URLs
  - Loosely coupled components
  - Designer friendly templates
  - Minimal coding
  - Fast development
- 


# Flask

---

- Easy to use, lightweight, customizable
  - Install on terminal using: *sudo pip3 install Flask*
  - Serves a web page as opposed to normal print statements
- 

# Flask Features

---

- Included development server
  - Support for unit testing
  - RESTful request handling
  - Easy templating
  - Secure cookie handling
- 

# Templating

---

- Templating allows us to create reusable HTML files that allow us to change just certain elements of the file at runtime.
- We use the Jinja2 template language in Flask.  
<https://jinja.palletsprojects.com/en/3.0.x/intro/>
- Jinja2 allows us to define variables, if controls, and loops in our templates.
  - Variables - <https://jinja.palletsprojects.com/en/3.1.x/templates/#variables>
  - IF statements - <https://jinja.palletsprojects.com/en/3.1.x/templates/#tests>
  - FOR loops - <https://jinja.palletsprojects.com/en/3.1.x/templates/#synopsis>
- Brief illustration of Jinja2's uses: <https://youtu.be/4yaG-jFfePc?t=718>

# Flask

---

**DEMO**



# Q&A

---