Python -
Bottle Framework for Web App Development

Eli Etherton - August 27, 2024

What is Bottle

- Web App Micro Framework
  - No MVC or ORM like Django
- python3 -m pip install bottle
- https://bottlepy.org/docs/dev/
- Dirt Simple Framework that runs on script execution

Basic Architecture

- Bottle for Web App functionality
- Python for code
- Static Files - Data Store for Files
- Templates - HTML Templates with Language
- Database for Data

Put Into Production???

- Python Web Server
- NGINX
- Ports - 8080

Bottle Basics

- route
- run(host='localhost', port=8080)
- Make sure script is running
- Changes do not take effect until script is restarted
- Example = bottle-route.py

Dynamic Routes

- /<value>
- Filters - int, float, path
  - /<value:int>
- Example = bottle-value.py

Redirect

- Redirect to another web page
- Example = bottle-redirect.py

POST

- Post - Process HTML form data
- From bottle import post
- @post('/process')
- name = request.forms.get('name')
- Example - bottle-post.py

GET

- Get - Process values from URL
- from bottle import get
- Red
- @get('/')
- color = request.query.get('color')
- Example - bottle-get.py

Templates

- From bottle import template
- Create "views" folder in root directory
- Template files end with .tpl
- Template files allow you to create dynamic web pages
- return template('hello_template')
- return template('hello_var_template', name=name)

Static Files

- Any Static Files such as Images or CSS Style Sheet
- For Security File Server should be separate server
- File/ Folder Permissions
- Create "static" folder in Root Directory
- Example = bottle-static.py

Labs

- lab-simple.py
  - Create a very basic web app
- lab-note.py
  - Create a Note Taking App
- lab-ping.py
  - Create a web app that automatically detects if a host is reachable