Flask Pre-reqs

Hello Worl

Hello World

Basic Example Closer Look Variable Routing

Liet Sort

Demo

Structure

Communication Putting It All

Deep Dive

Conclusio

Resource: Contact

An Introduction to Web Development with Flask



Prudhvi Boyapalli

Rice Computer Science Club Hack & Learn Workshop January 13, 2017

Flask Pre-regs

Pre-regs

Hello World

Basic Example Closer Look Variable Routin

List So

Structure Communication Putting It All Together

Deep Div

Summary

1 Introduction

Flask

Pre-reqs

Setup

2 Hello World

(Open up your editor and follow along!)

Basic Example

Closer Look

Variable Routing

3 List Sort

(Don't try to code this up right now.)

Demo

Structure

Communication

Putting It All Together

- 4 Deep Dive
- **6** Conclusion

Summary

Resources

Contact



Flask Pre-reqs

Hello World

Basic Example

Closer Look Variable Routing

List Sort

Demo

Structure Communication Putting It All Together

Deen Dive

Summary Resources Contact

Introduction

How does a web app work? What does Flask do?

Flask is a Python microframework for building web applications.

Introduction

Flask Pre-reqs

Hello Worl

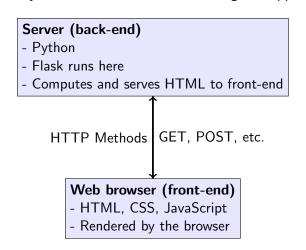
Basic Example Closer Look Variable Routin

List So

Structure Communication Putting It All Together

Deep Dive

Summary Resources



Necessary prior knowledge

ntroduction

Pre-reqs

Pre-reqs Setup

Hello Worl

Basic Example Closer Look Variable Routing

List Sor

Structure Communication Putting It All Together

Deep Dive

Summary Resources Basic familiarity with Python

- If you've taken COMP140 you're in good shape
- Basic familiarity with HTML and JavaScript, CSS
 - Be able to read HTML and JavaScript syntax
 - CSS is optional

Necessary prior knowledge

Introduction

Pre-reqs

Hello World

Hello Worl

Basic Example Closer Look Variable Routing

Demo
Structure
Communication
Putting It All

Deep Dive

Summary

- Basic familiarity with Python
 - If you've taken COMP140 you're in good shape
- Basic familiarity with HTML and JavaScript, CSS
 - Be able to read HTML and JavaScript syntax
 - CSS is optional

Install Python & Flask

Introduction

Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communication Putting It All Together

Deep Div

Summary

Install Python if you haven't already: python.org

• Either Python 2.7.9+ or 3.3+

Install Flask using PIP

 Pip Installs Packages from the Python Package Index a.k.a. PyPI

pip is included in the Python install

Flask Installation

> sudo pip install Flask

Editing

An IDE is overkill for this workshop. Use a programmer's editor that you're comfortable with:

- Vim (All platforms)
- Atom (All platforms)
- Visual Studio Code (All platforms)
- Sublime Text (All platforms)
- Notepad++ (Windows)
- TextMate (OSX)
- Whatever floats your boat

Flask

Setup Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communication

Deep Di

Conclusion

Resource Contact

Debugging

You'll be testing and debugging your code from the browser, so use one with good developer tools:

- Chrome
- Firefox

Other browsers can be used, but Chrome or Firefox will minimize the chance of running into browser specific issues.

Introduction Flask

Pre-reqs Setup

Basic Example Closer Look Variable Routing

List Sort

Demo
Structure
Communication
Putting It All

Deep Dive

Summary

Hello World

Basic Example Closer Look Variable Routing

Putting It All

Summary Resources Contact

Hello World (Open up your editor and follow along!)

Super basic hello world example

Basic Example Closer Look Variable Routing This is the bare minimum code needed on the back-end to serve a web page

```
./code/hello_world_example/hello_world.py
```

```
from flask import Flask
app = Flask(_-name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

Running the web app

Once we've written the flask code, we can run the server locally and view the results in our web browser

Execute these commands to run the code:

- > cd /directory/with/project
- > export FLASK_APP=hello_world.py
- > flask run

Then you'll see something like:

- * Serving Flask app "hello_world"
- * Running on http://127.0.0.1:5000/
 (Press CTRL+C to quit)

Setup

Hello World

Basic Example

Closer Look
Variable Routing

. . . .

Demo Structure Communication

Communication Putting It All Together

Deep Div

Summary

Running the web app

Once we've written the flask code, we can run the server locally and view the results in our web browser

Execute these commands to run the code:

- > cd /directory/with/project
- > export FLASK_APP=hello_world.py
- > flask run

- * Serving Flask app "hello_world"
- * Running on http://127.0.0.1:5000/

- Basic Example Variable Routing

Running the web app

Once we've written the flask code, we can run the server locally and view the results in our web browser

Execute these commands to run the code:

- > cd /directory/with/project
- > export FLASK_APP=hello_world.py
- > flask run

Then you'll see something like:

- * Serving Flask app "hello_world"
- * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

Basic Example Variable Routing

Introduction

Flask Pre-reqs

Hello World

Basic Example Closer Look

Closer Look Variable Routing

Demo Structure Communication Putting It All Together

Deep Dive

Summary Resources Contact • We started up a local flask server which is listening at IP address 127.0.0.1, port 5000

Introduction

Flask Pre-reqs

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communication Putting It All

Deep Dive

Summary Resources Contact

- 1 We started up a local flask server which is listening at IP address 127.0.0.1, port 5000
- We visited this page and the browser sent a request to our flask server

Introduction

Flask Pre-reqs

Hello World

Basic Example Closer Look Variable Routing

List Sort Demo Structure Communicatio Putting It All Together

Deep Dive

Summary

- 1 We started up a local flask server which is listening at IP address 127.0.0.1, port 5000
- We visited this page and the browser sent a request to our flask server
- 3 Because we visited 127.0.0.1:5000/ (emphasis on the "/"), the method bound to "/" was invoked

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communicatio Putting It All Together

Deep Dive

Summary

• We started up a local flask server which is listening at IP address 127.0.0.1, port 5000

- We visited this page and the browser sent a request to our flask server
- 3 Because we visited 127.0.0.1:5000/ (emphasis on the "/"), the method bound to "/" was invoked
- The hello() method in hello_world.py was executed and the return value is sent to the browser as HTML

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo
Structure
Communication
Putting It All
Together

Deep Dive

Summary Resources

- 1 We started up a local flask server which is listening at IP address 127.0.0.1, port 5000
- We visited this page and the browser sent a request to our flask server
- 3 Because we visited 127.0.0.1:5000/ (emphasis on the "/"), the method bound to "/" was invoked
- 4 The hello() method in hello_world.py was executed and the return value is sent to the browser as HTML
- **5** The browser received the HTML and rendered the webpage

Interactive hello world with variable routing

Now that we have a basic idea of how flask works, we can do some cool stuff with routing:

```
./code/hello_world_example/hello_user.py
```

Basic Example Closer Look Variable Routing

```
from flask import Flask
app = Flask(\_name\_\_)
@app.route("/world")
def hello():
    return "Hello World!"
@app.route("/user/<name>")
def hello_user(name):
    return "Hello %s, how are you today?" % name
if __name__ == "__main__":
    app.run()
```

Running the new and improved web app

Introduction

Flask Pre-reqs

Hello Worl

Tiello vvori

Closer Look

Variable Routing

Demo Structure

Communication
Putting It All
Together

Deep Dive

Summary

As before:

- > export FLASK_APP=hello_user.py
- > flask run
- * Serving Flask app "hello_user"
- * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

Hello World

http://127.0.0.1:5000/world

Hello <user>

http://127.0.0.1:5000/user/Foo http://127.0.0.1:5000/user/Bar

Running the new and improved web app

Introduction

Flask Pre-regs

Hello World

Basic Example Closer Look

Variable Routing

Demo Structure Communicatio Putting It All Together

Deep Dive

Summary Resources

As before:

- > export FLASK_APP=hello_user.py
- > flask run
- * Serving Flask app "hello_user"
- * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

Hello World

http://127.0.0.1:5000/world

Hello <user>

http://127.0.0.1:5000/user/Foo http://127.0.0.1:5000/user/Bar

Flask Pre-regs

Pre-reqs

Hello World

Basic Example Closer Look Variable Routing

List Sort

Demo Structure Communication Putting It All Together

Deep Dive

Conclusio

Summary Resources Contact

List Sort (Don't try to code this up right now.)

Quick Demo

Closer Look Variable Routing

Demo

Putting It All

Summary Resources Contact

List Sorting Example

Enter a comma seperated list of strings (no spaces):

list,of,items

Descending Ascending

Sort

Project Structure

As we build more complex web applications, we need to organize our files in a sensible way and maintain compatibility with flask

List Sort Project Structure

Directories indicated in bold.

- list_sort_example
 - static
 - scripts (JavaScript files)
 - styles (CSS files)
 - templates
 - index.htm
 - list_sort.py

Basic Example Closer Look Variable Routing

List Sort

Demo Structure

Communication Putting It All Together

Deep DIV

Conclusio

Resource

Project Structure

As we build more complex web applications, we need to organize our files in a sensible way and maintain compatibility with flask

List Sort Project Structure

Directories indicated in hold.

- list_sort_example
 - static
 - scripts (JavaScript files)
 - styles (CSS files)
 - templates
 - index.html

Basic Example

Variable Routing

Structure

Project Structure

As we build more complex web applications, we need to organize our files in a sensible way and maintain compatibility with flask

List Sort Project Structure

Directories indicated in hold.

- list_sort_example
 - static
 - scripts (JavaScript files)
 - styles (CSS files)
 - templates
 - index.html
 - list_sort.py

Basic Example Variable Routing

Structure

Hypertext Transfer Protocol

The data transfer protocol used on the World Wide Web

GET The most common HTTP method, used to get files from a server. Multiple GET requests are made by your browser every time you visit a webpage to retrieve all the necessary resources needed to render that page, such as HTML files, CSS and JavaScript files, images, etc.

Basic Example

Communication

Hypertext Transfer Protocol

The data transfer protocol used on the World Wide Web

GET The most common HTTP method, used to get files from a server. Multiple GET requests are made by your browser every time you visit a webpage to retrieve all the necessary resources needed to render that page, such as HTML files, CSS and JavaScript files, images, etc.

POST This method is used to send data to a server.

Commonly used to submit form data or any other user input.

These are the most common HTTP request methods and the ones used by the demo, others exist:

developer.mozilla.org/en-US/docs/Web/HTTP/Methods

Introduction

Flask Pre-regs

Hello Worl

Basic Example Closer Look Variable Routin

List Sort Demo Structure

Communication Putting It All Together

Deep Di

Conclusion

Resource Contact

Hypertext Transfer Protocol

The data transfer protocol used on the World Wide Web

GET The most common HTTP method, used to get files from a server. Multiple GET requests are made by your browser every time you visit a webpage to retrieve all the necessary resources needed to render that page, such as HTML files, CSS and JavaScript files, images, etc.

POST This method is used to send data to a server.

Commonly used to submit form data or any other user input.

These are the most common HTTP request methods and the ones used by the demo, others exist:

developer.mozilla.org/en-US/docs/Web/HTTP/Methods

Introduction

Flask Pre-reqs

Hello World

Basic Example Closer Look Variable Routin

Demo Structure Communication Putting It All Together

Deep Div

Conclusion

Resources

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

List Sort
Demo
Structure
Communicat

Putting It All Together

Deep Dive

Summary Resources Contact Retrieving resources and rendering the webpage

1 User enters the address into their browser

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communication Putting It All Together

Deep Dive

Summary Resources Retrieving resources and rendering the webpage

- User enters the address into their browser
- 2 Browser sends a GET request to the address, in this case 127.0.0.1:5000/

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example
Closer Look

Closer Look Variable Routing

Structure Communication Putting It All Together

Deep Div

Summary

Retrieving resources and rendering the webpage

- 1 User enters the address into their browser
- ② Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- 3 The flask method bound to "/" is invoked and returns index.html

Retrieving resources and rendering the webpage

- User enters the address into their browser
- ② Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- 3 The flask method bound to "/" is invoked and returns index.html
- 4 Browser processes the HTML file, which instructs it to load CSS and JavaScript files

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example

Closer Look

Variable Routing

Demo Structure Communication Putting It All Together

Deep Dive

Summary

Retrieving resources and rendering the webpage

- User enters the address into their browser
- ② Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- 3 The flask method bound to "/" is invoked and returns index.html
- 4 Browser processes the HTML file, which instructs it to load CSS and JavaScript files
- **6** Browser sends additional GET requests to the location indicated within the HTML files

Flask

Hello World
Basic Example

Basic Example Closer Look Variable Routing List Sort

Structure Communication Putting It All Together

Deep Div

Summary Resources

Introduction Flask

Pre-reqs Setup

Basic Example Closer Look Variable Routing

Demo Structure Communication Putting It All Together

Deep Dive

Summary Resources Retrieving resources and rendering the webpage

- User enters the address into their browser
- 2 Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- 3 The flask method bound to "/" is invoked and returns index.html
- 4 Browser processes the HTML file, which instructs it to load CSS and JavaScript files
- **5** Browser sends additional GET requests to the location indicated within the HTML files
- 6 Flask serves the static CSS and JavaScript files

Retrieving resources and rendering the webpage

- User enters the address into their browser
- ② Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- The flask method bound to "/" is invoked and returns index.html
- 4 Browser processes the HTML file, which instructs it to load CSS and JavaScript files
- 6 Browser sends additional GET requests to the location indicated within the HTML files
- 6 Flask serves the static CSS and JavaScript files
- **7** Browser now has all necessary files and renders the webpage

- Introduction Flask
- Hello World Basic Example
- Closer Look
 Variable Routing
- Demo Structure Communication Putting It All Together
- Deep Div
- Summary Resources Contact

Introduction Flask

Hello World

Basic Example
Closer Look
Variable Routing

List Sort
Demo
Structure
Communication
Putting It All

Together

Deep Dive

Summary Resources Contact Retrieving resources and rendering the webpage

- 1 User enters the address into their browser
- ② Browser sends a GET request to the address, in this case 127.0.0.1:5000/
- The flask method bound to "/" is invoked and returns index.html
- 4 Browser processes the HTML file, which instructs it to load CSS and JavaScript files
- 6 Browser sends additional GET requests to the location indicated within the HTML files
- 6 Flask serves the static CSS and JavaScript files
- 7 Browser now has all necessary files and renders the webpage

This all happens within milliseconds!

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

List Sort
Demo
Structure
Communicatio
Putting It All
Together

Deep Dive

Summary Resources Contact Send data to the server, back-end computation, return results

User enters information into the input fields and clicks the Sort button

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo
Structure
Communicatio
Putting It All
Together

Deep Div

Summary

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked

Send data to the server, back-end computation, return results

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- The input information is read from the input fields and stored in an object (JavaScript Object Notation)

Flask Pre-regs Setup

Hello World

Basic Example Closer Look Variable Routing

List Sort
Demo
Structure
Communication
Putting It All

Together

Deep Dive

Summary

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routin

Demo Structure Communication Putting It All Together

Deep Dive

Summary

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"

- 1 User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- A PUT request is created and sent to "/sort"
- 6 Flask server receives the request
 - The method bound to "/sort" is invoked

- Putting It All Together

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"
- **5** Flask server receives the request
 - The method bound to "/sort" is invoked
 - 2 The data from the request object is accessed

- Introduction
- Flask Pre-regs Setup
- Hello World Basic Example
- Closer Look Variable Routing
- Demo Structure Communication
- Together

 Deep Dive
- Summary Resources

Send data to the server, back-end computation, return results

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"
- **5** Flask server receives the request
 - 1 The method bound to "/sort" is invoked
 - 2 The data from the request object is accessed
 - 3 The list is sorted

- Introduction
- Flask Pre-regs Setup
- Hello World

 Basic Example

 Closer Look
- Demo Structure
- Putting It All Together

Deep Div

Summary Resources Contact

Send data to the server, back-end computation, return results

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"
- **5** Flask server receives the request
 - 1 The method bound to "/sort" is invoked
 - 2 The data from the request object is accessed
 - 3 The list is sorted
 - 4 A response object is created with the results in JSON format and returned

Introduction

Flask Pre-reqs Setup

Basic Example

Closer Look
Variable Routin

Structure Communication Putting It All Together

Deep Div

Summary Resources Contact



Send data to the server, back-end computation, return results

- 1 User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"
- **5** Flask server receives the request
 - The method bound to "/sort" is invoked
 - The data from the request object is accessed
 - 3 The list is sorted
 - 4 A response object is created with the results in JSON format and returned
- 6 Back in the JavaScript function, the response is parsed

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routin

Demo Structure

Putting It All Together

Deep Div

Resources Contact

Send data to the server, back-end computation, return results

- User enters information into the input fields and clicks the Sort button
- 2 The JavaScript function bound to that button is invoked
- 3 The input information is read from the input fields and stored in an object (JavaScript Object Notation)
- 4 A PUT request is created and sent to "/sort"
- **5** Flask server receives the request
 - The method bound to "/sort" is invoked
 - 2 The data from the request object is accessed
 - 3 The list is sorted
 - 4 A response object is created with the results in JSON format and returned
- 6 Back in the JavaScript function, the response is parsed
- **7** The necessary HTML to display the sorted list is created and inserted into page

- Introduction
- Flask Pre-regs Setup
- Hello World Basic Example
- Closer Look Variable Routing
- Demo Structure
- Putting It All Together

Doop Div

Summary Resources Contact



Introduction

Flask

Pre-reqs

Hello World

Basic Example Closer Look Variable Routing

List Sort

Demo Structure Communication Putting It All Together

Deep Dive

Summary Resources Contact

Deep Dive

Pre-regs

Basic Example Closer Look Variable Routing

Putting It All Together

Conclusion

Resources Contact

Conclusion

Topics Covered

Introduction

Flask Pre-reqs

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communication

Deep Dive

Conclusio

Summary Resources

- Setting up a development environment for Flask
- Basic & and variable routing
- GET and POST requests
- Accessing request data
- Overview of in-browser developer tools
- Building and serving a "complex" web app with HTML, CSS, JavaScript and Python

Next Steps

Variable Routing

Summary

URL Building

More Flask Features

- Additional HTTP Methods
- HTML templates with placeholders, complex templating language
- File uploads
- Reading and writing cookies
- Redirects and error handling
- Managing sessions
- And lots of other stuff

The Flask website and documentation is really good! Lots of great tutorials exist, just search around.

Deploying your web app

Introduction

Flask Pre-reqs Setup

Hello World

Basic Example Closer Look Variable Routing

Demo Structure Communicati Putting It All

Deep Dive

Summary

Resources

Once you've developed and tested you app locally, you can deploy your code and make it publicly accessible. There lots of tutorials on how to do this for various cloud providers (AWS, Azure, DigitalOcean, etc).

Resources

Basic Example Variable Routing

Putting It All

Contact

Resources

- This presentation & and all of the code github.com/hack-rice/flask-tutorial
- Flask website & awesome documentation flask.pocoo.org

Introduction

Flask Pre-reqs

Hello Worl

Basic Example Closer Look

Variable Routin

Demo Structure Communication

Deep Div

Conclusio

Resources

Prudhvi Boyapalli

prb2 AT rice.edu

HackRice

- Fall 2017
- Annual hackathon right here on campus
- Great way to hone your skills and make something cool

RiceApps

- Work on a team of students to build useful applications
- Great way to gain hands on development experience
- http://riceapps.org

Rice CS Facebook Group

• https://facebook.com/groups/365985373415471

CS Club Facebook Page

https://facebook.com/ricecsclub



Introduction

Flask Pre-regs

Hello Worl

Basic Example Closer Look Variable Routin

List Sort

Demo Structure Communication Putting It All Together

Deep Div

Conclusi

Resources Contact

Prudhvi Boyapalli

• prb2 AT rice.edu

HackRice

- Fall 2017
- Annual hackathon right here on campus
- Great way to hone your skills and make something cool

RiceApps

- Work on a team of students to build useful applications
- Great way to gain hands on development experience
- http://riceapps.org

Rice CS Facebook Group

https://facebook.com/groups/365985373415471

CS Club Facebook Page

https://facebook.com/ricecsclub



Introduction

Flask Pre-regs

Hello Worl

Basic Example Closer Look Variable Routin

List Sort

Demo
Structure
Communicatio
Putting It All
Together

Deep Di

Conclusi

Resources Contact

Prudhvi Boyapalli

• prb2 AT rice.edu

HackRice

- Fall 2017
- Annual hackathon right here on campus
- Great way to hone your skills and make something cool

RiceApps

- Work on a team of students to build useful applications
- Great way to gain hands on development experience
- http://riceapps.org

Rice CS Facebook Group

• https://facebook.com/groups/365985373415471

CS Club Facebook Page

https://facebook.com/ricecsclub



Introduction

Flask Pre-reqs Setup

Hello World

Basic Example

Closer Look

List Sort

Demo
Structure
Communicatio
Putting It All

Deep Di

Summary Resources Contact

Prudhvi Boyapalli

prb2 AT rice.edu

HackRice

- Fall 2017
- Annual hackathon right here on campus
- Great way to hone your skills and make something cool

RiceApps

- Work on a team of students to build useful applications
- Great way to gain hands on development experience
- http://riceapps.org

Rice CS Facebook Group

https://facebook.com/groups/365985373415471

CS Club Facebook Page

• https://facebook.com/ricecsclub