ENGL 516X: Methods of Formal Linguistic Analysis

Semester: Spring '16

Instructor: Sowmya Vajjala

Iowa State University, USA

31 March 2016

What is a web application?

First things first: What according to you is a web-application? What should it do?

What is a web application?

First things first: What according to you is a web-application?
What should it do?
What should we know to develop programs that work over the web in python?

Terminology

1. Server - a computer device or program that provides some functionality to several programs called clients. In web programming, clients are typically browsers on your users' machines.
2. Client - devices or software that access the services of a server.
3. Server side programming: programs that you write to make your server do its job.
4. Client side programming: programs you write on the clients side, so that the client-server communication happens, and so that the client does not go to server for things that can be done there itself.
5. In this class, we will talk about server side web programming.

Web application development in Python

- There are some collections of packages and modules in Python, that allow us to write web application code. These are called "Web Frameworks".
- They take care of low level details and allow us to focus only on our application.
- Several web frameworks exist in python, and their primary differences lie in the diversity of functionalities they provide.
- For a detailed overview: `https://wiki.python.org/moin/WebFrameworks/`
- For a general idea about using python in the web: `https://docs.python.org/2/howto/webservers.html`

Bottle: Introduction

- Bottle is a light weight, easy to setup web framework for Python. It is distributed as a single module, and does not have any external dependencies.
- Provides support for basic data access, file uploads, display of web pages and so on.
- installation:
  1. Download bottle.py from `http://bottlepy.org/bottle.py` and save it into your programming folder. Then, just import bottle in your program, and you are ready to go.
  2. "install" bottle the way you did for BeautifulSoup.

A Hello World Web Application

```
from bottle import route, run, template

@route("/")
def whatever():
    return ’Hello world!’

run(host=’localhost’, port=8080)
```

Terminology

1. route: route in Bottle decides the paths associated with individual functions in the code.
2. get: this method is used to get some information from the server to the client browser.
3. post: This request method is used typically in cases where we ”submit” some data to the server, and the server receives whatever we sent (does some processing if needed)
4. @route, @get, @post - these descriptions before a function definition tell us what the purpose of the function is.

HTML

- HTML - Hyper Text Markup Language. It is a markup used to display formatted data on webpages. We can think of it as a protocol between the browser and the webpage author on how the content should be displayed.
- HTML - is full of various tags. So, key to writing good HTML is to know what these tags are and what functionalities they have.
- You can do a lot of things with HTML, but we don't have to know all that for your final projects.

HTML Tutorial

- HTML is a collection of tags. Most the tags also have an end tag. For example, a tag XX starts with $< XX >$ and ends with $< /XX >$.
- Two top level tags are : head and body. Head contains information about the title of the page and other meta data. Body contains the actual body of your html page.
- Inside body, you can have ”forms” where the user can enter some input (choose from a list of options, enter some text, enter passwords etc.) and submit it to your server to do some additional processing.

A basic HTML page

```
<html>
<head><title>A HTML Page </title></head>
<body> This is an example html page </body>
</html>
```

Some text formatting tags: b is for bold, i is for italics, $< br >$ is for a new line (without an end tag), $< p >$ and $< /p >$ cover a paragraph..and so on.

Having text areas and submit buttons

```
 <form>
  First name: <input type="text" name="first"> <br>
  Last name:  <input type="text" name="last"><br>
  Tell me something about yourself:
<textarea name="message" rows="10" cols="30"> </textarea>
<br><input type="submit" value="Submit">
</form>
```

That is not enough in real-life

1. What should happen after submit?

That is not enough in real-life

1. What should happen after submit?
2. Our data should go somewhere, something has to happen. What is the point otherwise?

That is not enough in real-life

1. What should happen after submit?
2. Our data should go somewhere, something has to happen. What is the point otherwise?
3. So, we use "action" and "method" attributes in form. "action" tells us which function gets activated after submitting the form. "method" tells us if it is a get or post request.

Example HTML with action and method

```
 <textarea form ="formid" name="taname" cols="35">
</textarea>
 <form action="/tokenize" method="post" id="formid">
  <input value="Tokenize" type="submit" />
        </form>
```

To learn more about HTML

`http://www.w3schools.com/html/default.asp`

Some Example Web applications using Bottle

1. Tokenizer application
2. Login page application
3. Uploading a file

For more details on using Bottle

Look at the tutorial and other additional resources in:
`http://bottlepy.org/docs/dev/index.html`

Next Week

How to work create (and update) databases in Python. This will help you finish Assignment 6, along with what you learnt today.