ECOM 5416

Parallel and Distributed Systems

Introduction to Flask framework

There are many modules or frameworks which allow building your webpage using python like a bottle, Django, Flask, etc. But the real popular ones are Flask and Django. Django is easy to use as compared to Flask but Flask provides you with the versatility to program with.

Flask is a web application framework written in Python, that provides the basics for URL routing and page rendering.

Flask offers suggestions, but doesn't enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easy.

Installation

We recommend using the latest version of Python. Flask supports Python 3.7 and newer.

use the following command to install Flask:

```
$ pip install Flask
```

Introduction to Flask framework

A Simple Example:

```
# Importing flask module in the project is mandatory
# An object of Flask class is our WSGI application.
from flask import Flask

# Flask constructor takes the name of
# current module (__name__) as argument.
app = Flask(__name__)
```

Flask Documentation

https://flask.palletsprojects.com/en/2.2.x/

Routing

Modern web applications use meaningful URLs to help users.
Use the route() decorator to bind a function to a URL.

```
@app.route('/')
def index():
    return 'Index Page'

@app.route('/hello')
def hello():
    return 'Hello, World'
```

Rendering Templates

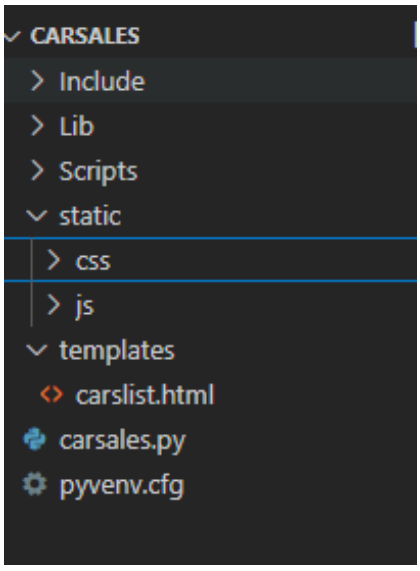
To render a template you can use the render_template() method. All you have to do is provide the name of the template and the variables you want to pass to the template engine as keyword arguments. Here's a simple example of how to render a template:

```
from flask import render_template

@app.route('/hello/')
@app.route('/hello/<name>')
def hello(name=None):
    return render_template('hello.html', name=name)
```

Flask will look for templates in the templates folder

```
@app.route('/home')
def home():
  return render_template("home.html")
```



Run Flask app

$flask run

```
* Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
```

Connecting to a MySQL database

Python module must be installed — PyMySQL — and a username and password must be included in the connection string (even an empty password has a place).

```
from flask import Flask, render_template
import pymysql

app = Flask(__name__)

class Database:
  def __int__(self):
    host = "http://127.0.0.1:81/"
    user = "root"
    password = ''
    db = "flaskmysql"
    self.con = mysql.connect(host=host, user=user, password=password, db=db)
    self.cur = self.con.cursor()

  def list_cars(self):
    self.cur.execute("SELECT * FROM cars")
    result = self.cur.fetchall()
    return result
```

```
@app.route('/home')
def home():
   return render_template("home.html")

@app.route('/product')
def product():
   def db_query():
      db= Database()
      products = db.list_cars()
      return products

   res = db_query()
   return render_template("product.html", result = res)
```

$flask run