

Python Web

Flask初识

上海育创网络科技有限公司

主讲人：刘老师(GerryLiu)

课程内容

- Web基础概述
- Flask概述
- Flask基本开发
- 项目案例：基于Flask实现一个登陆注册功能

Web概述

- Web开发可以理解为网站开发，客户可以直接通过Web浏览器使用系统功能，主要包含前端、后端和数据库三个方面。
 - Web前端主要就是负责页面的展示、数据的可视化等功能，主要技术就是HTML、CSS和JavaScript等。
 - Web后端主要就是处理请求的相关业务逻辑代码，会根据前端传入的url请求从数据库中获取对应的数据并反馈，主要技术有：.Net、Java、PHP、Python等。
 - 数据库主要就是存储业务数据用的，常用的数据库有：MySQL、Oracle、Redis、MongoDB等

Flask概述

- Flask是基于Python语言的一种Web微框架，是一种快速搭建Web系统的方式。
- 官网：<http://flask.pocoo.org/>
- 文档：
 - <http://docs.jinkan.org/docs/flask/>
 - <http://www.pythondoc.com/flask-mega-tutorial/index.html>
- 安装方式：
 - pip安装命令：`pip install Flask==0.12`
 - conda安装命令(建议)：`conda install -c anaconda flask==0.12`

Flask安装

```
C:\Windows\system32>pip install Flask==0.12
Collecting Flask==0.12
  Downloading https://files.pythonhosted.org/packages/0e/e9/37ee66dde483dceefe45bb5e92b387f990d4f097df40c400cf816dcebaa4/Flask-0.12-py2.py3-none-any.whl (82kB)
    100% |#####| 92kB 115kB/s
Requirement already satisfied: click>=2.0 in c:\anaconda3\lib\site-packages (from Flask==0.12) (6.7)
Requirement already satisfied: itsdangerous>=0.21 in c:\anaconda3\lib\site-packages (from Flask==0.12) (0.24)
Requirement already satisfied: Werkzeug>=0.7 in c:\anaconda3\lib\site-packages (from Flask==0.12) (0.14.1)
Requirement already satisfied: Jinja2>=2.4 in c:\anaconda3\lib\site-packages (from Flask==0.12) (2.9.4)
Requirement already satisfied: MarkupSafe>=0.23 in c:\anaconda3\lib\site-packages (from Jinja2>=2.4->Flask==0.12) (0.23)
Installing collected packages: Flask
Successfully installed Flask-0.12

C:\Windows\system32>python
Python 3.6.0 |Anaconda 4.3.1 (64-bit)| (default, Dec 23 2016, 11:57:41) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import flask
>>> flask.__version__
'0.12'
```

Flask安装

```
C:\Windows\system32>conda install -c anaconda flask==0.12
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.5.11
  latest version: 4.5.12

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Anaconda3

  added / updated specs:
    - flask==0.12

The following packages will be downloaded:
```

package	build	
qt-5.6.2	vc14h6f8c307_12	55.8 MB
bzip2-1.0.6	hfa6e2cd_5	145 KB
icu-58.2	ha66f8fd_1	21.9 MB
ca-certificates-2018.03.07	0	155 KB
curl-7.62.0	h2a8f88b_0	122 KB
vc-14.1	h21ff451_3	5 KB
libcurl-7.62.0	h2a8f88b_0	257 KB
libprotobuf-3.6.1	h7bd577a_0	1.9 MB
freetype-2.8	h51f8f2c_1	468 KB
tk-8.6.8	hfa6e2cd_0	3.8 MB
openssl-1.0.2p	hfa6e2cd_0	5.4 MB
pycurl-7.43.0.2	py36b74b6da3_0	58 KB

```
freetype: 2.8.1-vc14_0 conda-forge [vc14] -->
openssl: 1.0.2p-hfa6e2cd_1001 conda-forge --> 1.0.2p
pillow: 5.2.0-py36_0 conda-forge --> 5.1.0-p

Proceed ([y]/n)? y

Downloading and Extracting Packages
qt-5.6.2 55.8 MB #####
bzip2-1.0.6 145 KB #####
icu-58.2 21.9 MB #####
ca-certificates-2018 155 KB #####
curl-7.62.0 122 KB #####
vc-14.1 5 KB #####
libcurl-7.62.0 257 KB #####
libprotobuf-3.6.1 1.9 MB #####
freetype-2.8 468 KB #####
tk-8.6.8 3.8 MB #####
openssl-1.0.2p 5.4 MB #####
pycurl-7.43.0.2 58 KB #####
zlib-1.2.11 128 KB #####
python-3.6.7 20.9 MB #####
libpng-1.6.34 1.3 MB #####
protobuf-3.6.1 516 KB #####
flask-0.12 145 KB #####
libssh2-1.8.0 184 KB #####
hdf5-1.8.20 26.7 MB #####
pillow-5.1.0 648 KB #####
sqlite-3.25.3 937 KB #####
libtiff-4.0.9 829 KB #####
jpeg-9b 313 KB #####
vs2015_runtime-15.5. 2.2 MB #####
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

C:\Windows\system32>
C:\Windows\system32>python
Python 3.6.7 |Anaconda 4.3.1 (64-bit)| (default, Oct 28 2018, 19:44)
Type "help", "copyright", "credits" or "license()" for more
>>> import flask
>>> flask.__version__
'0.12'
>>>
```

Flask基本开发

- NOTE: Flask会动态加载所有的修改过的文件。
- Flask应用启动
 - 通过全局Flask对象的run方法来启动web应用，主要需要给定监听IP地址和监听端口号。
- Flask路由给定
 - 通过全局Flask对象的route注解给定对用API的请求路由，也就是给定对应资源的获取url路径。
 - Flask的路由支持普通url路径以及Restful类型的路径。

Flask基本开发

- Flask请求参数获取
 - 通过request.method获取请求方式(一般为GET或者POST)
 - 通过request.args获取GET请求时候url中所携带的请求参数所组成的字典对象。
 - 通过request.form获取POST请求时候所提交的参数所组成的字典对象。
 - 通过request.files获取POST请求过程中，所提交的所有文件数据对象所组成的字典对象。

Flask基本开发

- Flask请求结果返回
 - 如果给定一个字符串等基本数据类型，那么直接返回该数据
 - 可以通过jsonify将一个字典对象作为json格式数据返回(非常常用)
 - 可以通过render_template API返回一个对应的HTML
 - 可以通过url_for找到一个资源对应的url，并且可以使用redirect基于这个url跳转到对应的url资源上。

Flask项目案例(扩展)

- 利用Html + Flask + MySQL实现一个登陆注册功能的小系统。
 - 需求一：实现登陆注册功能
 - 需求二：当用户登陆失败3次后，1分钟之内不允许再登陆
 - 需求三：在用户登陆失败后的任何登陆行为都需要进行验证码识别。
 - 需求四：在所有操作过程中加入必要的提示信息。

Sign In

Register Account Number

Email	<input type="text" value="Email address"/>
Job number	<input type="text" value="Job number"/>
Password	<input type="password" value="Password"/>
Retry Password	<input type="password" value="Password"/>

- HTML、CSS、Bootstrap相关参考网站：
 - <http://www.w3school.com.cn/>
 - <http://www.bootcss.com/>
 - <https://v2.bootcss.com/components.html>
 - <http://getbootstrap.com/2.3.2/getting-started.html>

