



数据可视化 Python Flask框架

李春芳 副教授

理工学部 计算机学院

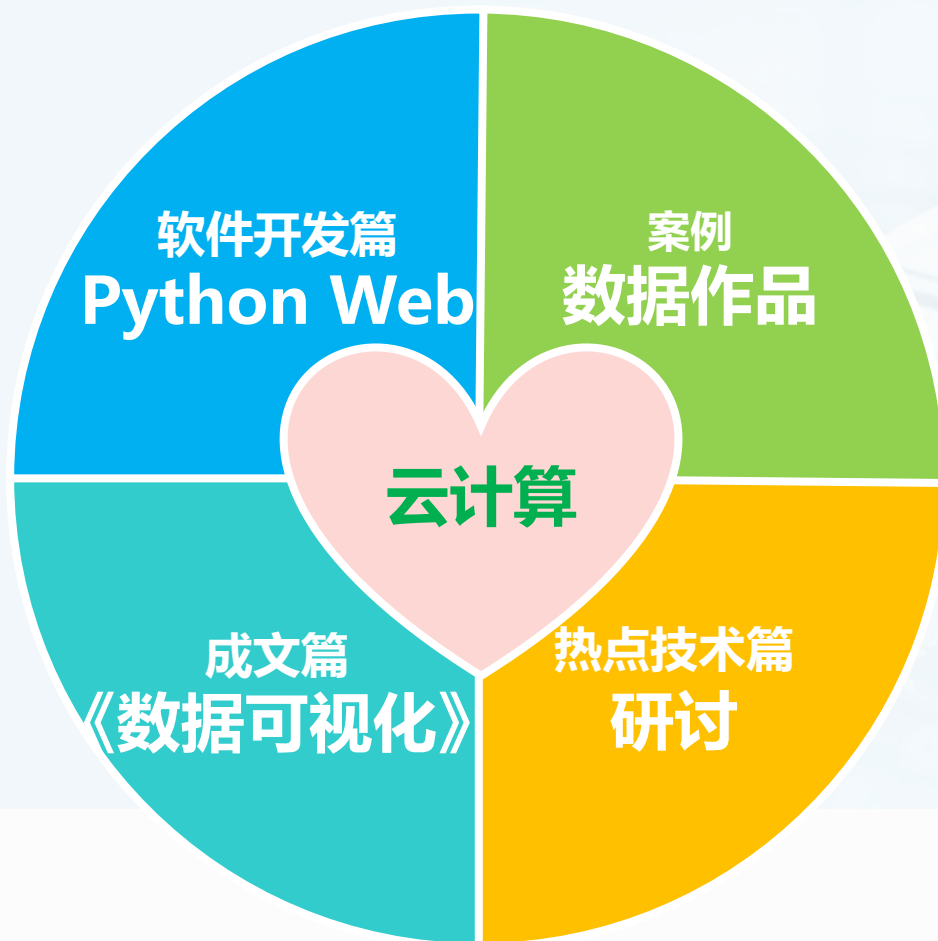
2017-7-3



中国传媒大学
COMMUNICATION UNIVERSITY OF CHINA

实践课程概述

课程目标：软件开发中的数据可视化



Python Flask

Web开发、数据分析、可视化、门槛低

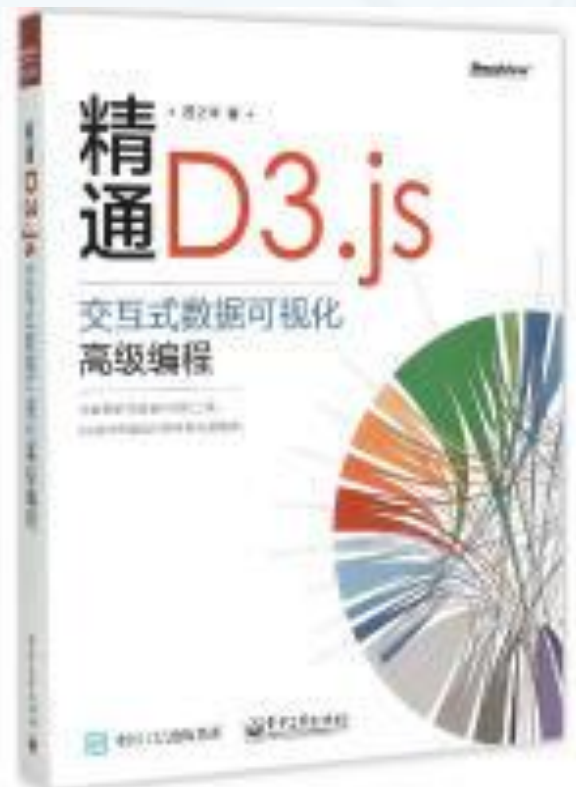
为什么学习Python Web开发?

- ◆ 学了很多语言，发现都是语法
- ◆ 学了软件工程，没开发过软件
- ◆ 学了数据库，原来数据库不是单独使用的
- ◆ 学了可视化，只能做一张张的图
- ◆ 大创的设计开发软件如何实现
- ◆ 毕设不会开发软件
- ◆

教材&电子书

链接: <http://pan.baidu.com/s/1co2MEM> 密码: bq4x

链接: <http://pan.baidu.com/s/1kVJsBzD> 密码: pqu6



Flask是什么？

- ◆ 支持各种关系数据库
- ◆ 支持图数据库Neo4J
- ◆ 扩展功能交由第三方开发
- ◆ 教材代码Git可下载
 - ◆ <https://github.com/miguelgrinberg/flask>
- ◆ 官方文档：flask.pocoo.org



[overview](#) // [docs](#) // [community](#) // [snippets](#) // [extensions](#)

Flask is a microframework for Python based on Werkzeug, Jinja 2 and good intentions. And before you ask: It's [BSD licensed](#)!

Flask is Fun

Latest Version: [0.12.2](#)

```
from flask import Flask
app = Flask(__name__)

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

And Easy to Setup

https://github.com/miguelgrinberg/flasky

https://github.com/miguelgrinberg/flask



This repository Search

Pull requests Issues Marketplace Gist



miguelgrinberg / flask
forked from pallets/flask

Watch ▾

4

Star

12

Fork

8,885

Code

Pull requests 0

Projects 0

Wiki

Insights ▾

A microframework based on Werkzeug, Jinja2 and good intentions <http://flask.pocoo.org/>

3,023 commits

17 branches

16 releases

433 contributors

BSD-3-Clause

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

This branch is 28 commits behind pallets:master.

Pull request Compare



davidism committed on GitHub Merge pull request #2374 from astralblue/json_encode_non_utc_datetimes ...

Latest commit d625d41 17 days ago

.github

update development resources

a month ago

artwork

Added lineart logo

2 years ago



安装Git客户端

> <http://git-scm.com>

 <https://git-scm.com>



 Search entire site...

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.



Learn Git in your browser for free with **Try Git**.



About

The advantages of Git compared to other source control systems.



Documentation

Command reference pages, Pro Git book content, videos and other material.

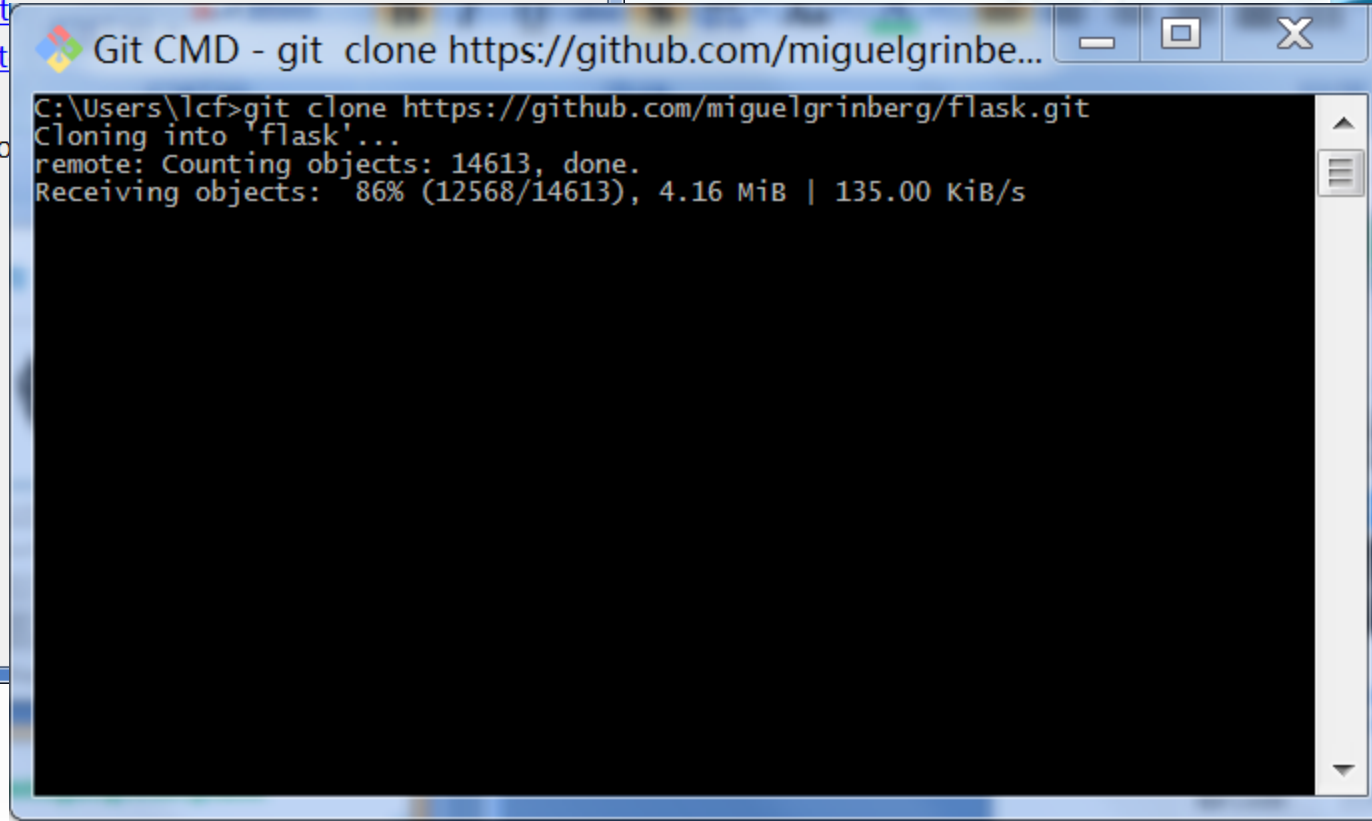
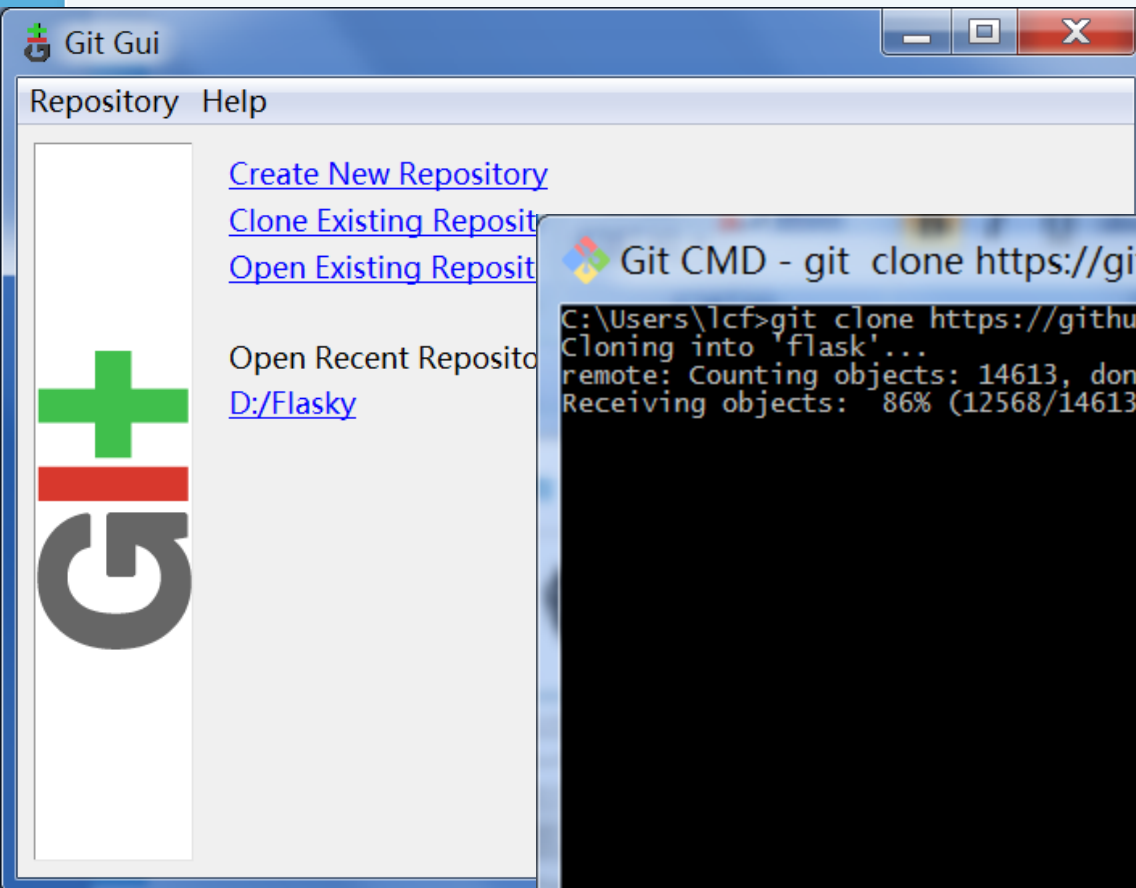


<http://git-scm.com/about/free-and-open-source>

Community

使用Git

› **Git clone <https://github.com/miguelgrinberg/flasky.git>**



TIPS: 命令行||GUI

◆ 下载到当前目录下，创建flasky文件夹

`Git clone https://github.com/miguelgrinberg/flasky.git`

◆ 迁出一个1a版本

`Git checkout 1a`

◆ 迁出其他历史版本之前，要恢复到原始状态

`git reset --hard`（恢复之前的.git，修改会丢失，请到别的文件夹修改保存）

› `Git fetch --all`

› `Git fetch --tags`

› `Git reset --hard origin/master`

查看两个版本之间的差别

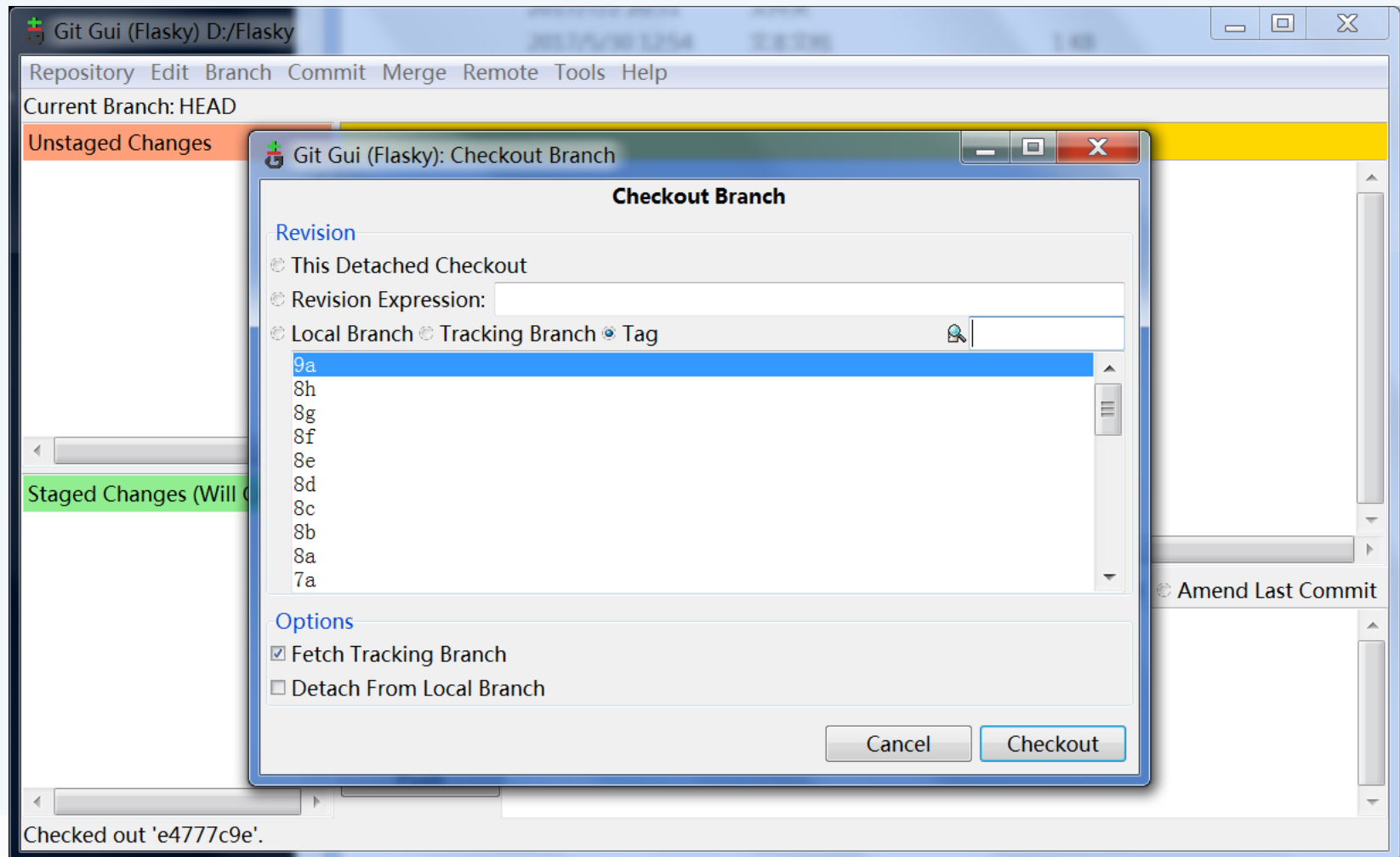
› Git diff 2a 2b

```
D:\flasky2017\flasky>git diff 2a 2b
diff --git a/hello.py b/hello.py
index 062cbc2..2bf2aed 100644
--- a/hello.py
+++ b/hello.py
@@ -7,5 +7,10 @@ def index():
     return '<h1>Hello World!</h1>'

+@app.route('/user/<name>')
+def user(name):
+    return '<h1>Hello, %s!</h1>' % name
+
+
if __name__ == '__main__':
    app.run(debug=True)

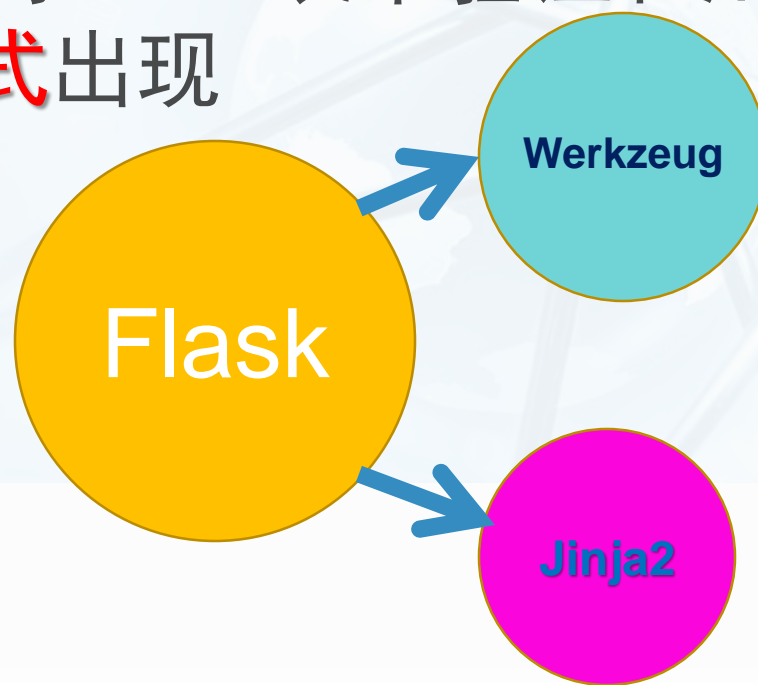
D:\flasky2017\flasky>
```

Git之GUI



Flask框架

- › 可扩展的框架
- › 主要的依赖：
 - Werkzeug: 路由、调试和Web服务器网关接口
 - Jinja2: 模板
- › 不原生支持数据库访问、Web表单验证和用户认证等，以**扩展形式**出现



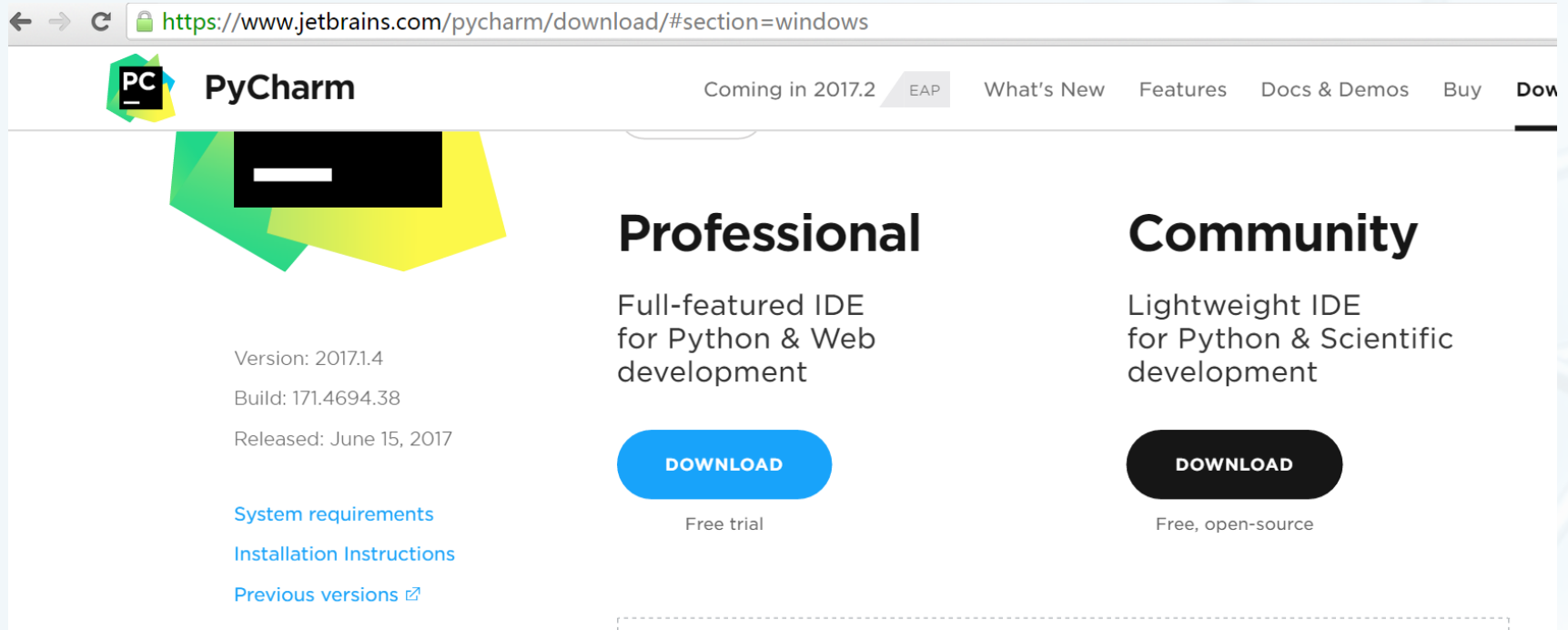
Python3.5安装&Flask

› Pip install flask



Pycharm IDE


› Pycharm社区版，无需注册



The screenshot shows the PyCharm download page. The browser address bar displays the URL: <https://www.jetbrains.com/pycharm/download/#section=windows>. The page header includes the PyCharm logo, the text "Coming in 2017.2 EAP", and navigation links: "What's New", "Features", "Docs & Demos", "Buy", and "Download". The main content area is divided into two columns. The left column features a large graphic of overlapping green and yellow shapes with a black rectangle containing a white horizontal line. Below this graphic, the text reads: "Version: 2017.1.4", "Build: 171.4694.38", and "Released: June 15, 2017". At the bottom of this column are three links: "System requirements", "Installation Instructions", and "Previous versions" with an external link icon. The right column has two main sections. The first section is titled "Professional" and describes it as a "Full-featured IDE for Python & Web development". Below this is a blue "DOWNLOAD" button and the text "Free trial". The second section is titled "Community" and describes it as a "Lightweight IDE for Python & Scientific development". Below this is a black "DOWNLOAD" button and the text "Free, open-source".

← → ↻ <https://www.jetbrains.com/pycharm/download/#section=windows>

PyCharm Coming in 2017.2 EAP [What's New](#) [Features](#) [Docs & Demos](#) [Buy](#) [Download](#)



Version: 2017.1.4
Build: 171.4694.38
Released: June 15, 2017

[System requirements](#)
[Installation Instructions](#)
[Previous versions](#) ↗

Professional

Full-featured IDE
for Python & Web
development

DOWNLOAD

Free trial

Community

Lightweight IDE
for Python & Scientific
development

DOWNLOAD

Free, open-source

Pycharm使用Flask

PC Settings

Project: ruyi2017 > Project Interpreter For current project

Project Interpreter: 3.5.2 (C:\Users\lcf\AppData\Local\Programs\Python\Python35\python.exe)


Package	Version	Latest
Flask	0.12.1	→ 0.12.2
Flask-Bootstrap	3.3.7.1	3.3.7.1
Flask-Bootstrap3	3.1.1.3	3.1.1.3
Flask-Login	0.4.0	0.4.0

Available Packages

Q flask

Package	Description
Flask	A microframework based on Werkzeug, Jinja2 and good intentions
Flask-ACL	
Flask-API	
Flask-API-Utils	

Version: 0.12.2



Flask安装到的目录

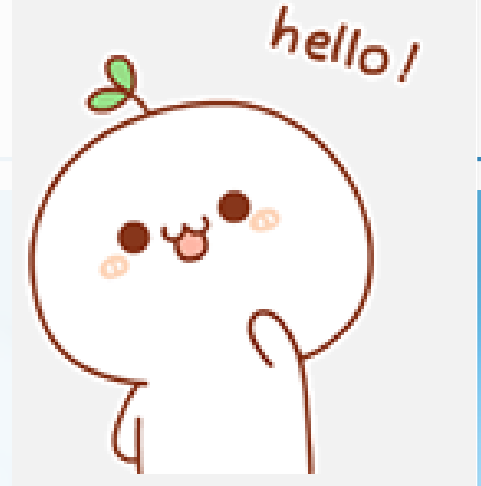
└─ lcf └─ AppData └─ Local └─ Programs └─ Python └─ Python35 └─ Lib └─ site-packages └─

刻录 新建文件夹

名称	修改日期	类型
__pycache__	2017/6/22 11:27	文件夹
blinker	2017/6/22 11:27	文件夹
blinker-1.4-py3.5.egg-info	2017/6/22 11:27	文件夹
click	2017/4/26 22:17	文件夹
click-6.7.dist-info	2017/4/26 22:17	文件夹
flask	2017/4/26 22:17	文件夹
flask_login	2017/6/22 11:26	文件夹
Flask_Login-0.4.0.dist-info	2017/6/22 11:26	文件夹
Flask-Mail-0.9.1-py3.5.egg-info	2017/6/22 11:27	文件夹

Hello.py

- ◆ `from flask import Flask`
- ◆ `app = Flask(__name__)`
- ◆ `@app.route('/')`
- ◆ `def index():`
 - ◆ `return '<h1>Hello World!</h1>'`
- ◆ `if __name__ == '__main__':`
 - ◆ `app.run(debug=True)`



```
D:\Flasky>python hello.py
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger pin code: 150-044-645
127.0.0.1 - - [02/Jul/2017 21:37:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Jul/2017 21:37:21] "GET /favicon.ico HTTP/1.1" 404 -
```

← → ↻ 127.0.0.1:5000

Hello World!

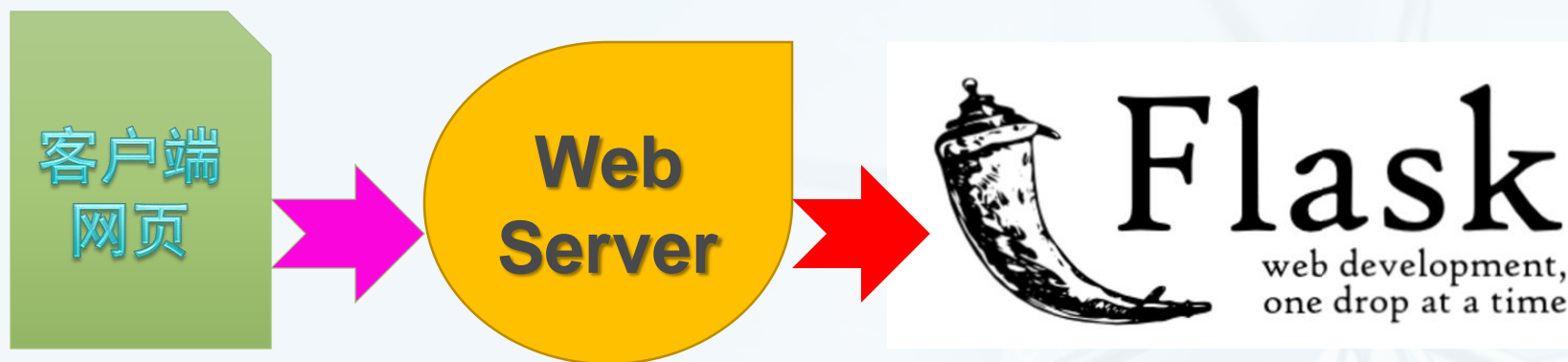
Flask初始化

- › Web 服务器使用 Web Server Gateway Interface, WSGI, 把客户端请求转交给Flask的对象处理
- ◆ `from flask import Flask`
- ◆ `app = Flask(__name__)`
- ◆ Flask类的构造函数只有一个参数，即程序主模块或包的名字，这个**参数决定程序的根目录**

Flask路由和视图函数

领导就是路由器

- ◆ 路由：处理URL和函数之间关系的程序



- ◆ `@app.route('/')`
- ◆ `def index():` 视图函数
- ◆ `return '<h1>Hello World!</h1>'`

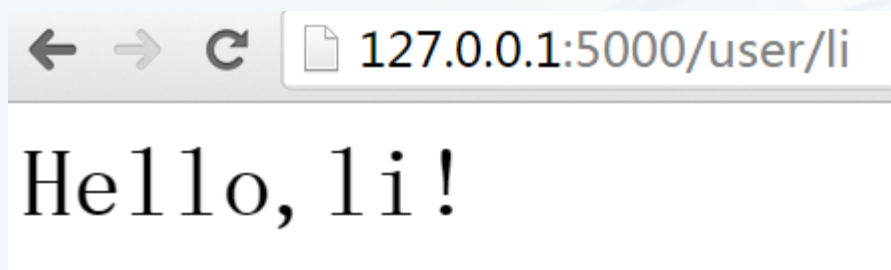
迁出：Git checkout 2a

动态参数路由器

- ◆ `@app.route('/user/<name>')`
- ◆ `def user(name):`
- ◆ `return '<h1>Hello ,%s!</h1>' % name`

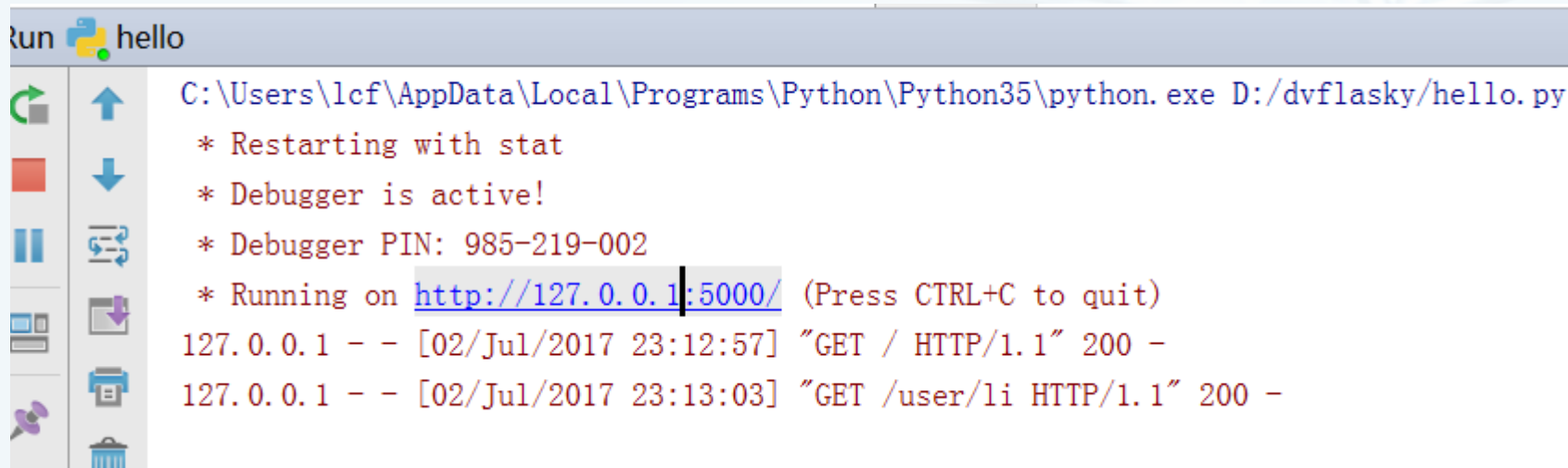
◆ [Git checkout 2b](#)

- ◆ 路由 `/user/<int:id>` 匹配 `id` 为整数的 URL



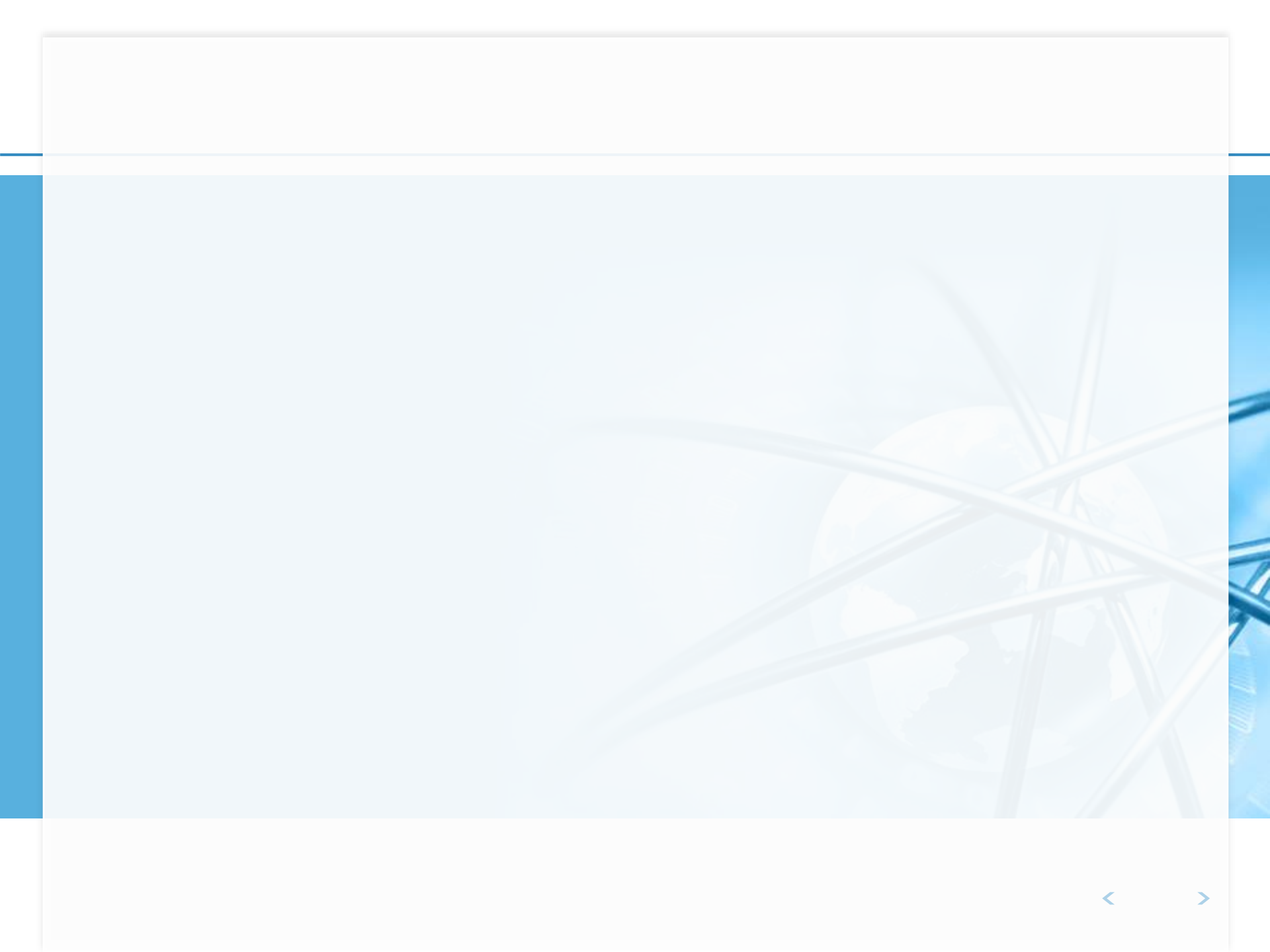
启动服务器

- ◆ 程序用run方法启动Flask集成的开发Web服务器
- ◆ `if __name__ == '__main__':`
- ◆ `app.run(debug=True)`

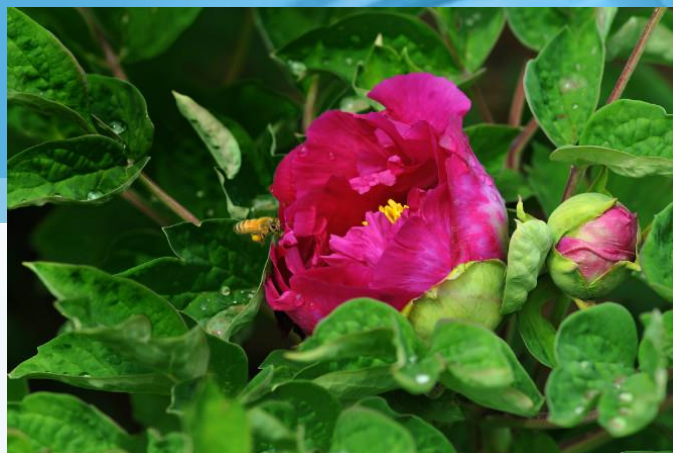


The screenshot shows a terminal window titled "Run hello" with a Python icon. The terminal output is as follows:

```
C:\Users\lcf\AppData\Local\Programs\Python\Python35\python.exe D:/dvflasky/hello.py
* Restarting with stat
* Debugger is active!
* Debugger PIN: 985-219-002
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [02/Jul/2017 23:12:57] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Jul/2017 23:13:03] "GET /user/li HTTP/1.1" 200 -
```



谢谢



期待各位的佳作