

# python

The Python logo, consisting of two interlocking snakes, one blue and one yellow, is positioned below the word "python".

```
import turtle
turtle.setup(650,350,200,200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")

for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
    turtle.circle(40, 80/2)
    turtle.fd(40)
    turtle.circle(16, 180)
    turtle.fd(40 * 2/3)
```

Python语言程序设计

# 从Web解析到网络空间

---



嵩 天  
北京理工大学





# 单元开篇

# 从Web解析到网络空间



- Python库之网络爬虫
- Python库之Web信息提取
- Python库之Web网站开发
- Python库之网络应用开发





# Python库之网络爬虫

# Python库之网络爬虫

**Requests: 最友好的网络爬虫功能库**

- 提供了简单易用的类HTTP协议网络爬虫功能
- 支持连接池、SSL、Cookies、HTTP(S)代理等
- Python最主要的页面级网络爬虫功能库

# Python库之网络爬虫

## Requests: 最友好的网络爬虫功能库

```
import requests  
  
r = requests.get('https://api.github.com/user',\  
                 auth=('user', 'pass'))  
  
r.status_code  
r.headers['content-type']  
r.encoding  
r.text
```



<http://www.python-requests.org/>

# Python库之网络爬虫

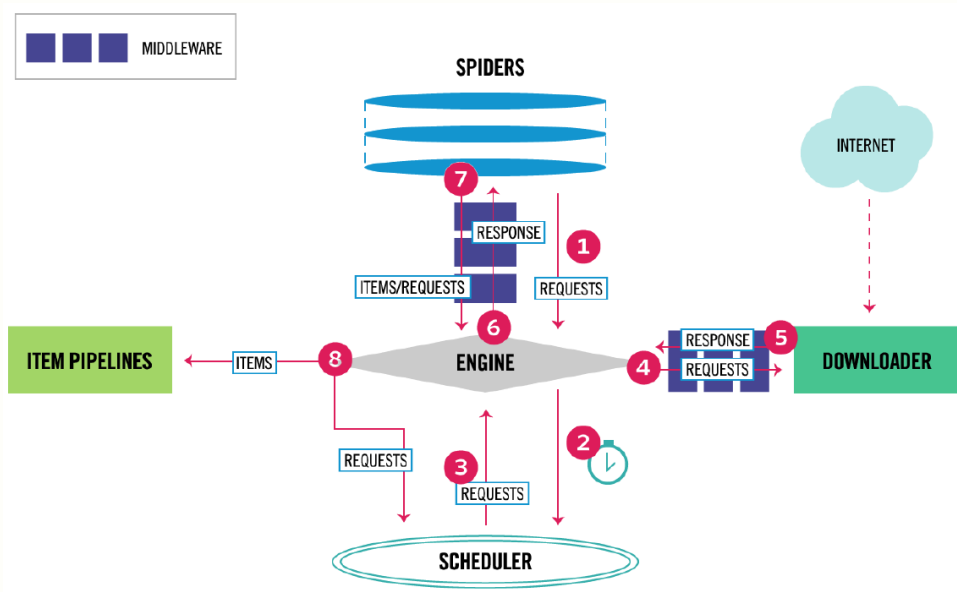
## **Scrapy:** 优秀的网络爬虫框架

- 提供了构建网络爬虫系统的框架功能，功能半成品
- 支持批量和定时网页爬取、提供数据处理流程等
- Python最主要且最专业的网络爬虫框架



# Python库之网络爬虫

**Scrapy:** Python数据分析高层次应用库



<https://scrapy.org>

# Python库之网络爬虫

**pyspider: 强大的Web页面爬取系统**

- 提供了完整的网页爬取系统构建功能
- 支持数据库后端、消息队列、优先级、分布式架构等
- Python重要的网络爬虫类第三方库

# Python库之网络爬虫

## pyspider: 强大的Web页面爬取系统

pyspider > js\_test\_sciencedirect Quickstart 脚本编写指南

```
{
  "fetch": {
    "fetch_type": "js"
  },
  "process": {
    "callback": "detail_page"
  },
  "project": "js_test_sciencedirect",
  "taskId": "091b162322318ebba208ad0feb01d6ed",
  "url": "http://www.sciencedirect.com/science/article/pii/S0261560602000463"
}
```

run

```
#!/usr/bin/env python
# -*- encoding: utf-8 -*-
# vim: set et sw=4 ts=4 sts=4 ff=unix fenc=utf8:
# Created on 2014-10-31 13:05:52

import re
from libs.base_handler import *

class Handler(BaseHandler):
    """
    this is a sample handler
    """
    def on_start(self):
        self.crawl('http://www.sciencedirect.com/science/article/pii/S1568494612005741',
                   callback=self.detail_page)

    def index_page(self, response):
        for each in response.doc('a').items():
            if re.match('http://www.sciencedirect.com/science/article/pii/\w+/$',
                        each.attr.href):
                self.crawl(each.attr.href, callback=self.detail_page)

    @config(fetch_type="js")
    def detail_page(self, response):
        self.index_page(response)
        self.crawl(response.doc('HTML>BODY>DIV#page-
area>DIV#rightPane>DIV#rightOuter>DIV#rightInner>DIV.InnerPadding>DIV#recommend_rela
ted_articles>OL#relArtList>LI>A.viewMoreArticles<.Link').attr.href,
                   callback=self.index_page)

    return {
        "url": response.url,
        "title": response.doc('HTML>BODY>DIV#page-
area>DIV#centerPane>DIV#centerContent>DIV#centerInner>DIV#frag_1>H1.svTitle').text(),
        "authors": [{"name": x.text(), "url": x.attr.href} for x in
response.doc('HTML>BODY>DIV#page-
area>DIV#centerPane>DIV#centerContent>DIV#centerInner>DIV#frag_1>UL.authorGroup.noCo
llab>LI.smh5A.authorName').items()],
        "abstract": response.doc('HTML>BODY>DIV#page-
area>DIV#centerPane>DIV#centerContent>DIV#centerInner>DIV#frag_2>DIV.abstract.svAbst-
ract>P').text(),
        "keywords": [x.text() for x in response.doc('HTML>BODY>DIV#page-
area>DIV#centerPane>DIV#centerContent>DIV#centerInner>DIV#frag_2>UL.keyword>LI.svKey
word').items()],
        "url": 'http://www.sciencedirect.com/science/article/pii/S0261560602000463'
    }

    'keywords': [],
    'title': 'Editorial',
    'url': 'http://www.sciencedirect.com/science/article/pii/S0261560602000463'
}
```

ScienceDirect Journals Books Shopping cart

Purchase Export Search ScienceDirect Advanced search

Journal of International Money and Finance  
Volume 21, Issue 6, November 2002, Pages 693  
International Financial Integration

Editorial  
J.R Lothian  
Show more

Choose an option to locate/access this article:

Check if you have access through your login credentials or your institution

Purchase \$35.95 Get Full Text Elsewhere

Check access

enable css selector helper web html follow 21 messages

<http://docs.pyspider.org>



# Python库之Web信息提取

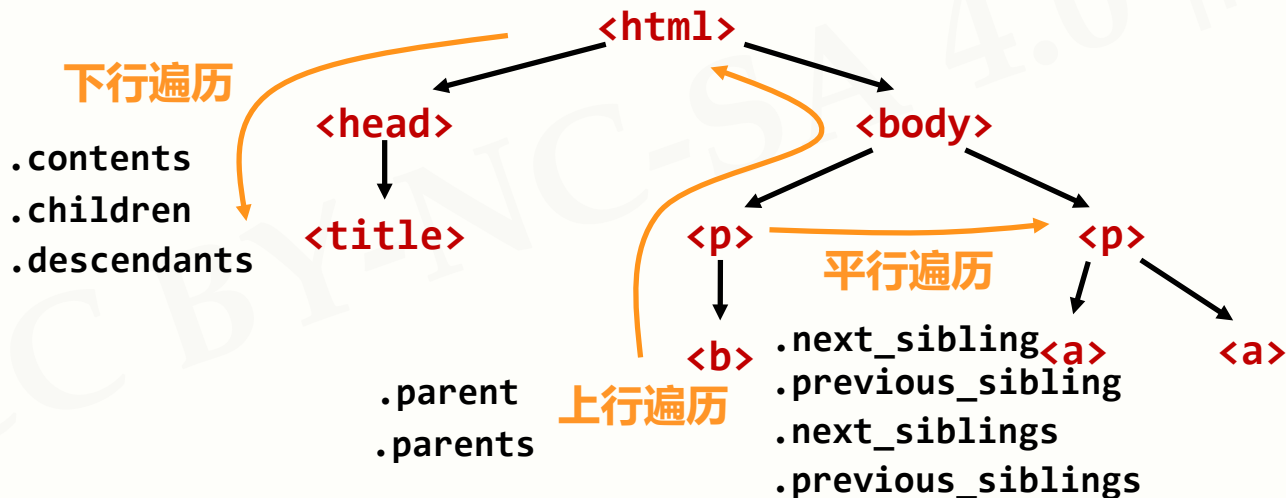
# Python库之Web信息提取

## **Beautiful Soup: HTML和XML的解析库**

- 提供了解析HTML和XML等Web信息的功能
- 又名beautifulsoup4或bs4，可以加载多种解析引擎
- 常与网络爬虫库搭配使用，如Scrapy、requests等

# Python库之Web信息提取

## Beautiful Soup: HTML和XML的解析库



<https://www.crummy.com/software/BeautifulSoup/bs4>

# Python库之Web信息提取

**Re: 正则表达式解析和处理功能库**

- 提供了定义和解析正则表达式的一批通用功能
- 可用于各类场景，包括定点的Web信息提取
- Python最主要的标准库之一，无需安装

# Python库之Web信息提取

## Re: 正则表达式解析和处理功能库

`re.search()`

`re.split()`

`re.match()`

`r'\d{3}-\d{8}|\d{4}-\d{7}'`

`re.finditer()`

`re.findall()`

`re.sub()`

<https://docs.python.org/3.6/library/re.html>



# Python库之Web信息提取

**Python-Goose:** 提取文章类型Web页面的功能库

- 提供了对Web页面中文章信息/视频等元数据的提取功能
- 针对特定类型Web页面，应用覆盖面较广
- Python最主要的Web信息提取库

# Python库之Web信息提取

## Python-Goose: 提取文章类型Web页面的功能库

```
from goose import Goose

url = 'http://www.elmundo.es/elmundo/2012/10/28/espana/1351388909.html'

g = Goose({'use_meta_language': False, 'target_language': 'es'})

article = g.extract(url=url)

article.cleaned_text[:150]
```

<https://github.com/grangier/python-goose>



# Python库之Web网站开发

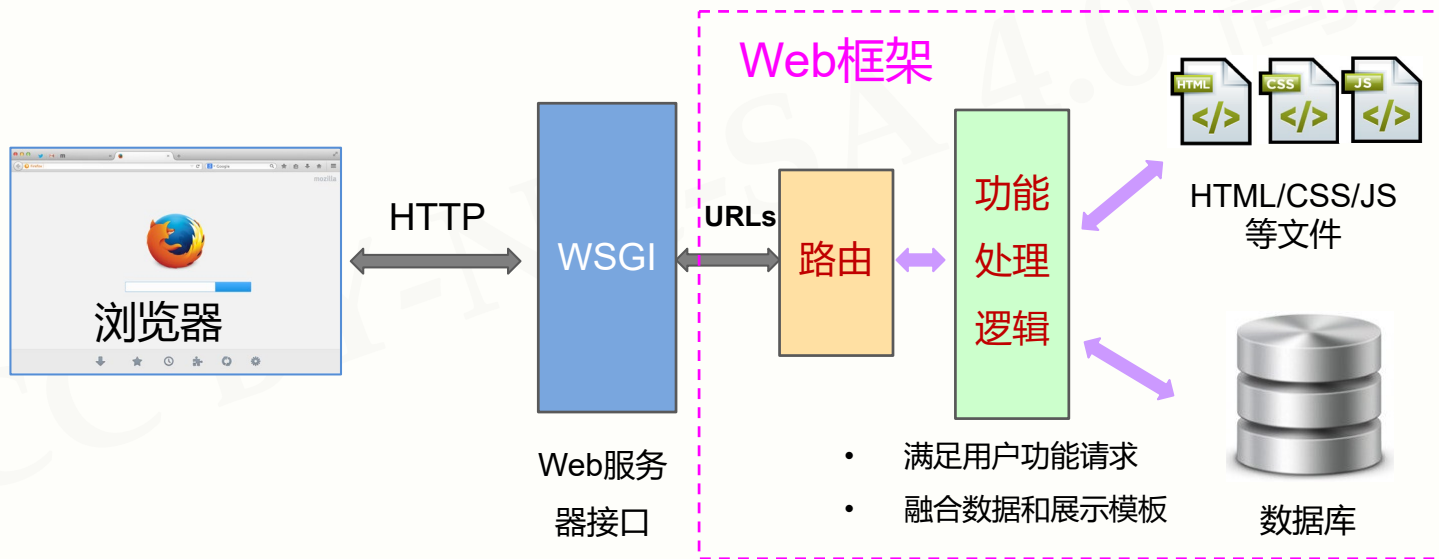
# Python库之Web网站开发

**Django: 最流行的Web应用框架**

- 提供了构建Web系统的基本应用框架
- MTV模式：模型(model)、模板(Template)、视图(Views)
- Python最重要的Web应用框架，略微复杂的应用框架

# Python库之Web网站开发

## Django: 最流行的Web应用框架



<https://www.djangoproject.com>

# Python库之Web网站开发

## **Pyramid:** 规模适中的Web应用框架

- 提供了简单方便构建Web系统的应用框架
- 不大不小，规模适中，适合快速构建并适度扩展类应用
- Python产品级Web应用框架，起步简单可扩展性好

# Python库之Web网站开发

## Pyramid: 规模适中的Web应用框架

```
from wsgiref.simple_server import make_server
from pyramid.config import Configurator
from pyramid.response import Response
def hello_world(request):
    return Response('Hello World!')
if __name__ == '__main__':
    with Configurator() as config:
        config.add_route('hello', '/')
        config.add_view(hello_world, route_name='hello')
        app = config.make_wsgi_app()
    server = make_server('0.0.0.0', 6543, app)
    server.serve_forever()
```

- 10行左右Hello Word程序

<https://trypyramid.com/>

# Python库之Web网站开发

## Flask: Web应用开发微框架

- 提供了最简单构建Web系统的应用框架
- 特点是：简单、规模小、快速
- Django > Pyramid > Flask



# Python库之Web网站开发

## Flask: Web应用开发微框架

```
from flask import Flask  
  
app = Flask(__name__)  
  
@app.route('/')  
def hello_world():  
    return 'Hello, World!'
```



<http://flask.pocoo.org>



# Python库之网络应用开发

# Python库之网络应用开发

## **WeRoBot:** 微信公众号开发框架

- 提供了解析微信服务器消息及反馈消息的功能
- 建立微信机器人的重要技术手段

# Python库之Web网站开发

## WeRoBot: 微信公众号开发框架

```
import werobot
```

```
robot = werobot.WeRoBot(token='tokenhere')
```

```
@robot.handler
```

```
def hello(message):
```

```
    return 'Hello World!'
```

- 对微信每个消息反馈一个Hello World

<https://github.com/offu/WeRoBot>

# Python库之网络应用开发

**aip: 百度AI开放平台接口**

- 提供了访问百度AI服务的Python功能接口
- 语音、人脸、OCR、NLP、知识图谱、图像搜索等领域
- Python百度AI应用的最主要方式

# Python库之Web网站开发

**aip:** 百度AI开放平台接口

	百度语音		文字识别		人脸识别
	自然语言处理		图像审核		知识图谱

<https://github.com/Baidu-AIP/python-sdk>

# Python库之网络应用开发

## **MyQR: 二维码生成第三方库**

- 提供了生成二维码的系列功能
- 基本二维码、艺术二维码和动态二维码


# Python库之Web网站开发

**MyQR:** 二维码生成第三方库



<https://github.com/sylnsfar/qrcode>





# 单元小结

# 从Web解析到网络空间

- Requests、Scrapy、pyspider
- Beautiful Soup、Re、Python-Goose
- Django、Pyramid、Flask
- WeRobot、aip、MyQR





# 小花絮

# 资深程序员如何看待编程工具？

PyCharm? No! Visual Studio? No! Anaconda? No!

- 编程工具琳琅满目，资深程序员都在用什么？如何选择？
- 资深程序员**更**理解逻辑、**更**期待效率、**更**重视简洁，因此：
  - 资深程序员都**不用**集成开发环境，**不用**复杂调试工具，**不用**复杂图形界面工具
  - 资深程序员都喜欢用**编辑器**类型的开发工具，小巧、灵活、可定制
  - 建议：Visual Studio Code (VSCode) 、Notepad++、Vim，足矣！

请初学者"老老实实"用IDLE，这个工具足够了



