

# 人工智能系列教程

# Python

# 关于我



张宏伦

上海交通大学

电子系直博在读

自然语言理解、机器学习、数据挖掘



我的个人网站 <http://zhanghonglun.cn/>

爱好数据、热衷技术

忙碌于各大数据比赛，混迹于知乎、掘金、简书



轮到你们



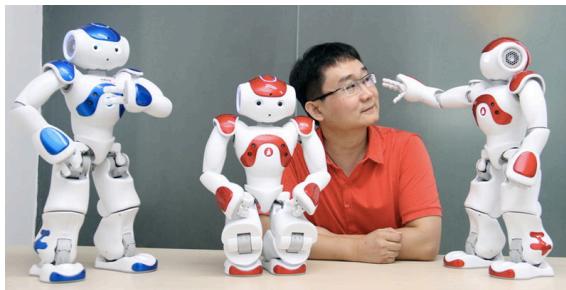
姓名、年龄、学校、兴趣爱好

# 什么是人工智能

媒体  
报道的



公众  
理解的



公司  
宣传的

```
1 #!/usr/bin/python
2 # coding: utf8
3
4 import sys
5 import json
6 import os
7 import urllib2
8 import time
9 import re
10 import pprint
11
12 headers = {}
13 headers['User-Agent'] = "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_4) AppleWebKit/537.36 (KHTML, like
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
```

背后的  
技术

# ■ 什么是人工智能

## 数据 + 人工智能 + 应用

没有数据和应用场景，空谈人工智能是 没有意义 的！

人工智能只有和具体的应用场景相结合，才能发挥出其价值

### 游戏



### 新闻

腾讯新闻 财经 RSS

首页 滚动 国际 社会 图片 视频 军事 评论

2017年08月31日 星期四 上海 26°C

没时间吃饭吃饼干 习近平出访时忙到日程以分钟计  
·《大国外交第一集：大道之行》  
·习近平一声令下，中国首次出动军舰撤侨！

财子第二日续命指南 新闻 1 追击

### 金融



### 医疗



# ■ 什么是人工智能

通过 编程，让 机器 完成 人类 的工作



包括 软件 和 硬件



硬件：看得见、摸得着，例如电脑的键盘、显示屏

软件：看不见、摸不着，例如电脑上安装的游戏

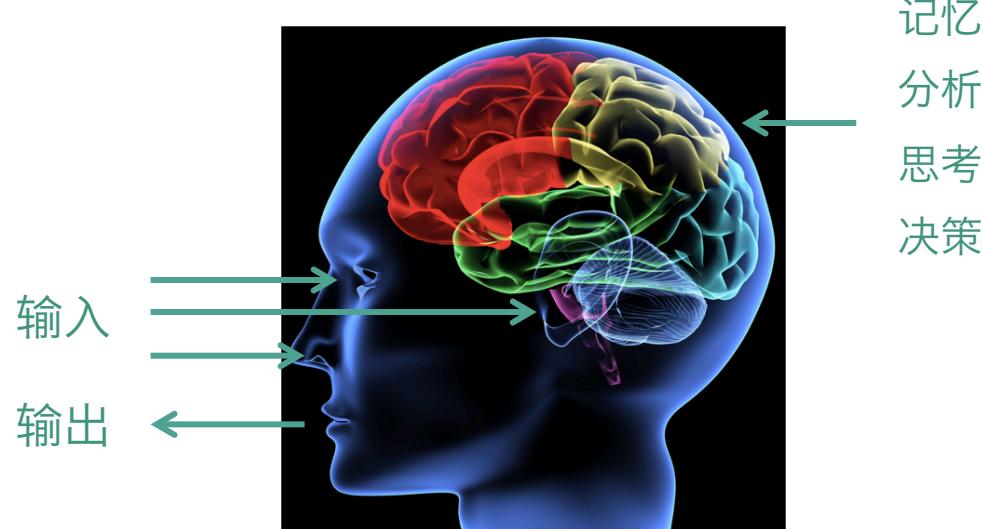


# 为什么要编程

编程，即用 代码 编写 程序

那么，为什么要编程

人就像一台机器一样去处理事情  
想一想，榨汁机是如何工作的



# 为什么要编程

显示器



输出



CPU

大脑中的计算单元

键盘



输入



内存

大脑中的记忆单元

硬盘



带在身上的本子

# 为什么要编程

编程，是为了 控制 计算机  
让计算机替我们 干活！

代码：向计算机发出的一条条命令（去洗碗！）

程序：由多条命令组成的一项任务（给我做饭！）

学习编程：学习如何让计算机替我们干活

Computing!



```
def isPassed(score):
    passed = False
    if score >= 60:
        passed = True
    else:
        passed = False
    return passed
result = isPassed(90)
print(result)
```



# 为什么要编程

你很快会发现，计算机很 **笨** (dump)，但是非常 **听话** (obedient)

它会完全按照你吩咐的来做，即使并非如你希望的那样

学习编程可以帮助我们了解到

**清晰表述** (clarity of expression) 的重要性



# 为什么要编程

吃饭的时候，一人说去 方便 一下

老外不解，旁人告诉他 方便 就是上厕所

敬酒时，另一人对老外说，希望下次出国时能给予 方便，老外纳闷不敢问

酒席上，电视台美女主持人提出，在她 方便 的时候会安排老外做专访

老外愕然：怎么能在你 方便 的时候？

美女主持人说，那在你 方便 时，我请你吃饭



# 编程包括

1. 让计算机听懂，用 编程语言 说话 (Python)
2. 学会编写 算法，用算法解决问题 (Algorithm)

小明有五位好朋友，他们年龄分别是 8 岁、6 岁、10 岁、12 岁、5 岁  
那么，他们之中年龄最大的是多少岁呢

如果你能够将某件事情 教会其他人，说明你真的掌握这件事情了

同样，如果你能够 教会计算机 做某件事情，说明你真的掌握这件事情了

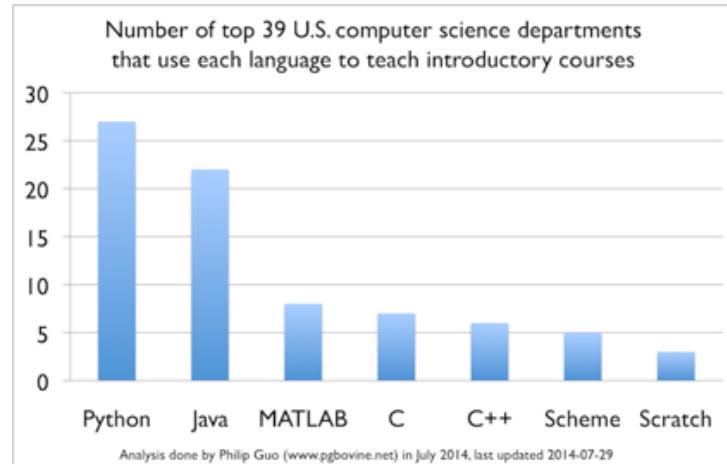
学习编程，让我们学会去思考如何 解决问题，通过编程将我们的 想法 变为现实

# 为什么学Python

计算机编程语言的 **种类**，就如同世界上的 **语言** 一样多

Python、R、C、C++、Java、Go、PHP、Matlab.....

- ✓ 在大量编程语言排名中，都 **稳居前列**
- ✓ 简单易学好用，作为 **第一门** 编程语言再好不过
- ✓ 世界上绝大多数 **大学** 选择的教学编程语言



# 为什么学Python

举个例子：打印字符串

Python

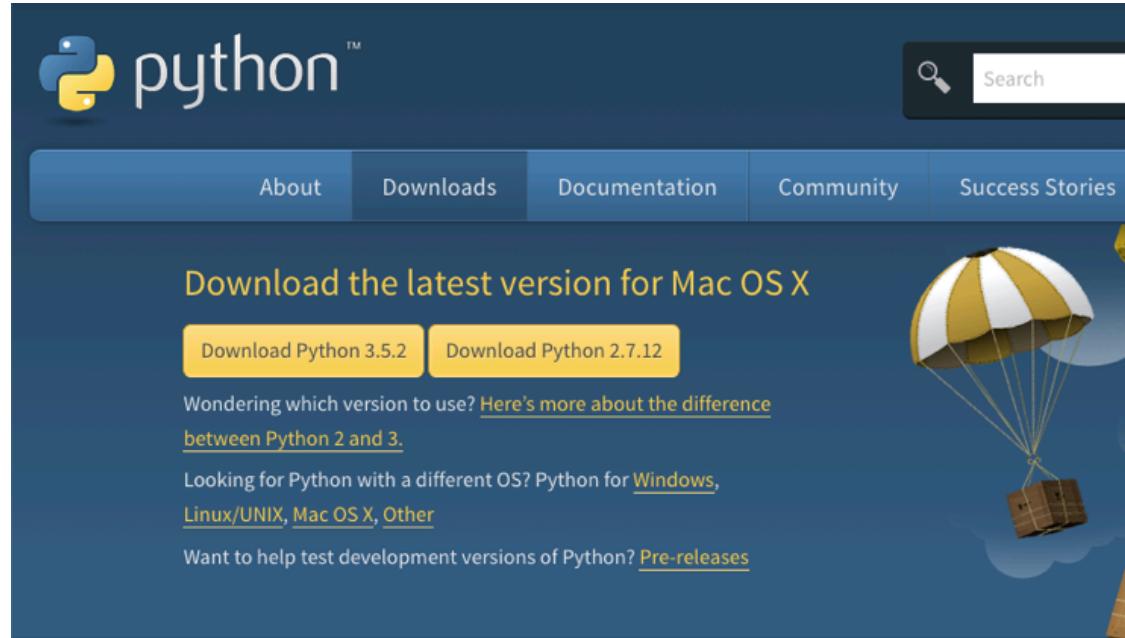
```
print("Hello World!")
```

Java

```
class HelloWorldApp {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

# ■ 安裝Python

<https://www.python.org/downloads/>



# 使用Python

交互式

```
>>> print(123)
123
>>> print(2.5)
2.5
>>> print("Hello World!")
Hello World!
```

脚本式

```
def isPassed(score):
    passed = False
    if score >= 60:
        passed = True
    else:
        passed = False
    return passed

result = isPassed(90)
print(result)
```

# 处理错误

在学习编程的过程中，我们很容易犯一些 错误

不要急，Python会 指出来 我们的错误

```
>>> print 123
```

```
SyntaxError: Missing parentheses in call to 'print'
```

不断 改正错误，不断 学习，我们会变得更加优秀

计算机虽然算得很快，但是却 不会学习

因此，让我们学习Python，比让计算机学习中文，要 简单得多

# 需要帮助

1. <https://docs.python.org/3/index.html>

The screenshot shows the Python 3.5.2 documentation homepage. At the top, it says "Python 3.5.2 documentation". Below that, a welcome message states "Welcome! This is the documentation for Python 3.5.2, last updated Jul 30, 2016." A section titled "Parts of the documentation:" lists various documentation types:

- What's new in Python 3.5?**  
or all "What's new" documents since 2.0
- Tutorial**  
start here
- Library Reference**  
keep this under your pillow
- Language Reference**  
describes syntax and language elements
- Python Setup and Usage**  
how to use Python on different platforms
- Python HOWTOs**  
in-depth documents on specific topics
- Installing Python Modules**  
installing from the Python Package Index & other sources
- Distributing Python Modules**  
publishing modules for installation by others
- Extending and Embedding**  
tutorial for C/C++ programmers
- Python/C API**  
reference for C/C++ programmers
- FAQs**  
frequently asked questions (with answers!)

2. 使用 Google、Bing、Baidu 等搜索引擎寻找解决方法

# 课程考核

平时的 课后作业 + 最后的 编程比赛

积极参与课堂，回答问题 + 上台演示

表现好的同学，会有 小礼物 哟～

本次课后作业：

1. 使用Python计算  $1 + 1$  并打印
2. 使用Python打印出自己的名字

```
>>> print(123)
123
>>> print(2.5)
2.5
>>> print("Hello World!")
Hello World!
```



# Python

去实现你想做的任何事情