

Python程序设计

王晶 计算机学院（国家示范性软件学院）





内容

- 课程介绍
- Python语言介绍
- 开发环境准备
- Python代码编写规范



Python程序设计：教学内容（一）

基础知识

教学内容	参考课时
课程概述和相关环境准备	1
基本数据类型、表达式和输入输出	3
程序控制结构	2
组合数据类型	4
函数	2
文件	2
面向对象设计方法	2
异常处理	2

作业：在线编程练习

oj系统地址（校园网内登录）：
<http://10.91.145.2/>
登录账号：个人学号



Python程序设计：教学内容（二）

应用第三方库进行项目开发

教学内容	参考课时
数据获取	4
数据计算	2
数据预处理	4
数据可视化	4

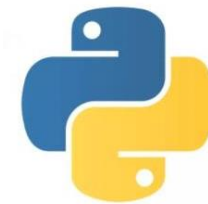
作业：完成项目单元
开发及集成开发

平台：
爱课堂



Python程序设计：考核方式

- 在线作业（30分）：由计算机自动判题，按照通过用例数的比例给分
- 小作业（30分）：使用第三方工具，完成数据的爬取、处理和展示
- 大作业（40分）：综合使用数据处理工具，爬取网站信息，并对数据进行处理、分析和可视化展示













内容

- 课程介绍
- Python语言介绍
- 开发环境准备
- Python代码编写规范

TIOBE编程语言排名 (2022年8月)



<https://www.tiobe.com/tiobe-index/>

Aug 2022	Aug 2021	Change	Programming Language		Ratings	Change
1	2	▲		Python	15.42%	+3.56%
2	1	▼		C	14.59%	+2.03%
3	3			Java	12.40%	+1.96%
4	4			C++	10.17%	+2.81%
5	5			C#	5.59%	+0.45%
6	6			Visual Basic	4.99%	+0.33%
7	7			JavaScript	2.33%	-0.61%
8	9	▲		Assembly language	2.17%	+0.14%
9	10	▲		SQL	1.70%	+0.23%
10	8	▼		PHP	1.39%	-0.80%

Python在“圈外”



有人说，**学习让人焦虑。**

但果真如此吗？

在这个碎片化知识的时代，
我们还相信学习的力量吗？

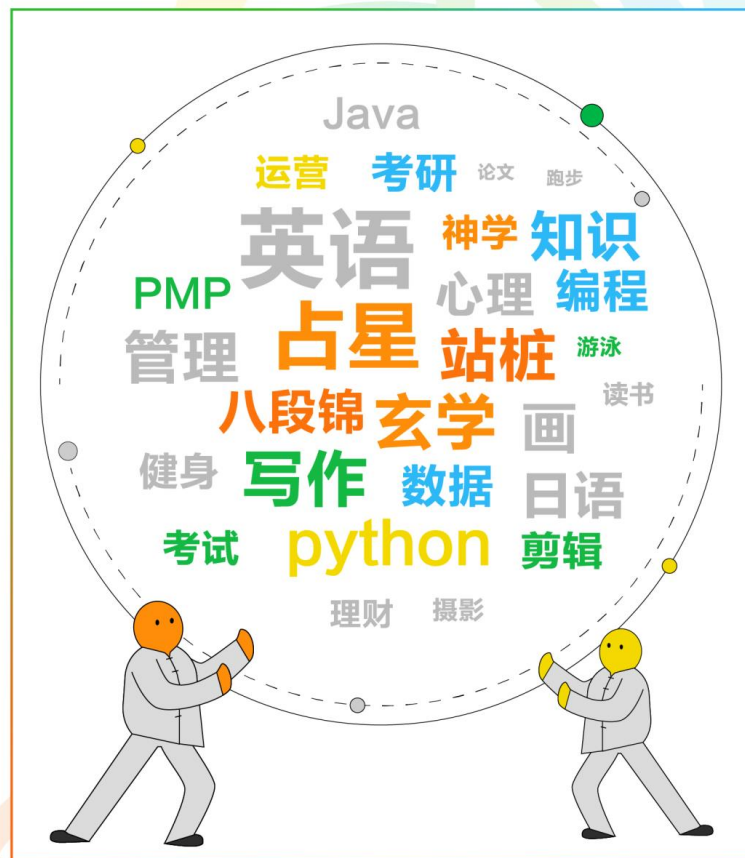
2012-2022

「印象笔记」已经完整走过十年
在这十年里国民学习力又是如何变迁的？
请查收这份《2022全民学习力报告》。

卷是真的卷，摆烂？骗你的！



惊！你的邻居竟然在学……





Python语言简介



- Python语言拥有者是Python Software Foundation(PSF)
- PSF是非盈利组织，致力于保护Python语言开放、开源和发展



Monty Python组合

Python的作者

- Guido Von Rossum, 荷兰人
- 出生于1956年1月31日
- 1982年, Guido从阿姆斯特丹大学 (University of Amsterdam)毕业, 获得数学和计算机科学硕士学位
- 2005年12月, 加入Google
- 2019年10月, 宣布退休



Guido von Rossum



Python的历史

- 1991年, Python雏形
- 1994年, Python 1.0
- 2000 年, Python 2.0
- 2008年, Python 3.0

最后一个Python2. x: Python 2.7.18

最新版本: Python 3.10.6 (Aug. 2, 2022)





Python语言的基本特点

Python是一种动态的、解释性的脚本语言，语法简洁易读

- 通用、跨平台
- 简单易用，编程模式非常符合人类的思维方式和习惯
- 结合其开源、免费的特性，拥有大量的几乎支持所有领域应用开发的成熟扩展库（第三方库），形成了庞大的计算生态
- 具有良好的“粘合性”，可以把使用多种不同语言编写的程序融合在一起，满足不同应用领域的需求



Python语言的优势

语法简洁，是C代码量的10%

```
#include <stdio.h>

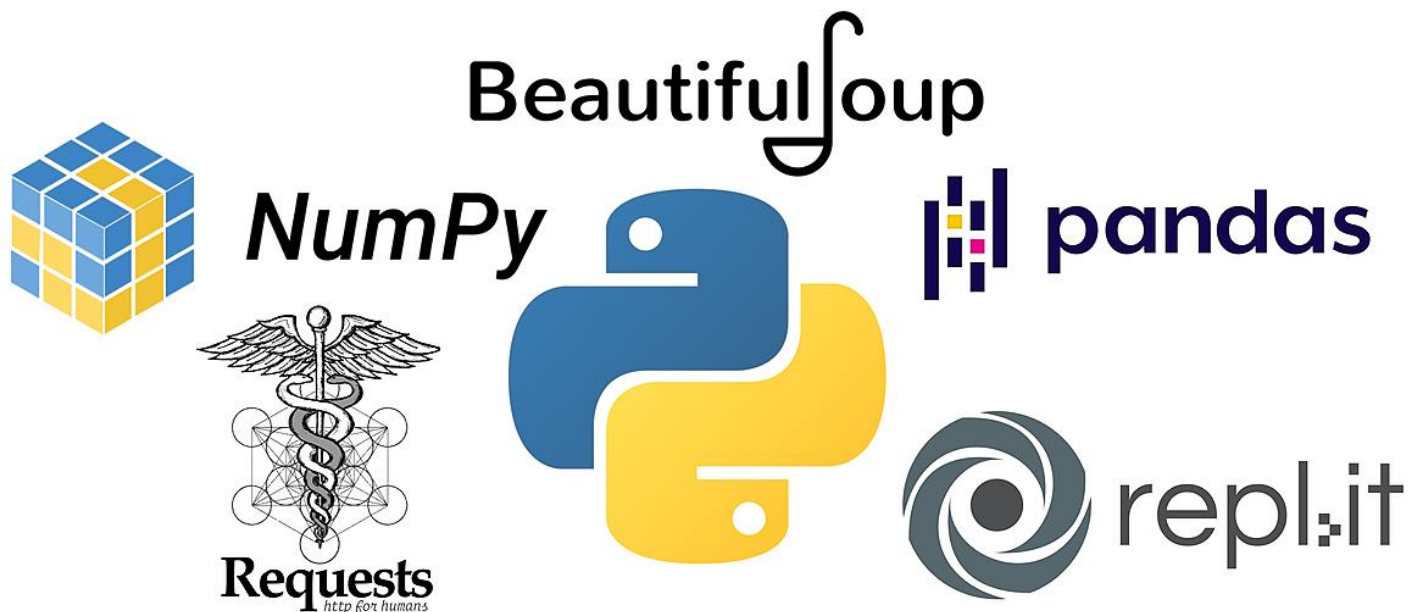
int main()
{
    printf("Hello world!");
}
```

```
print("Hello world!")
```



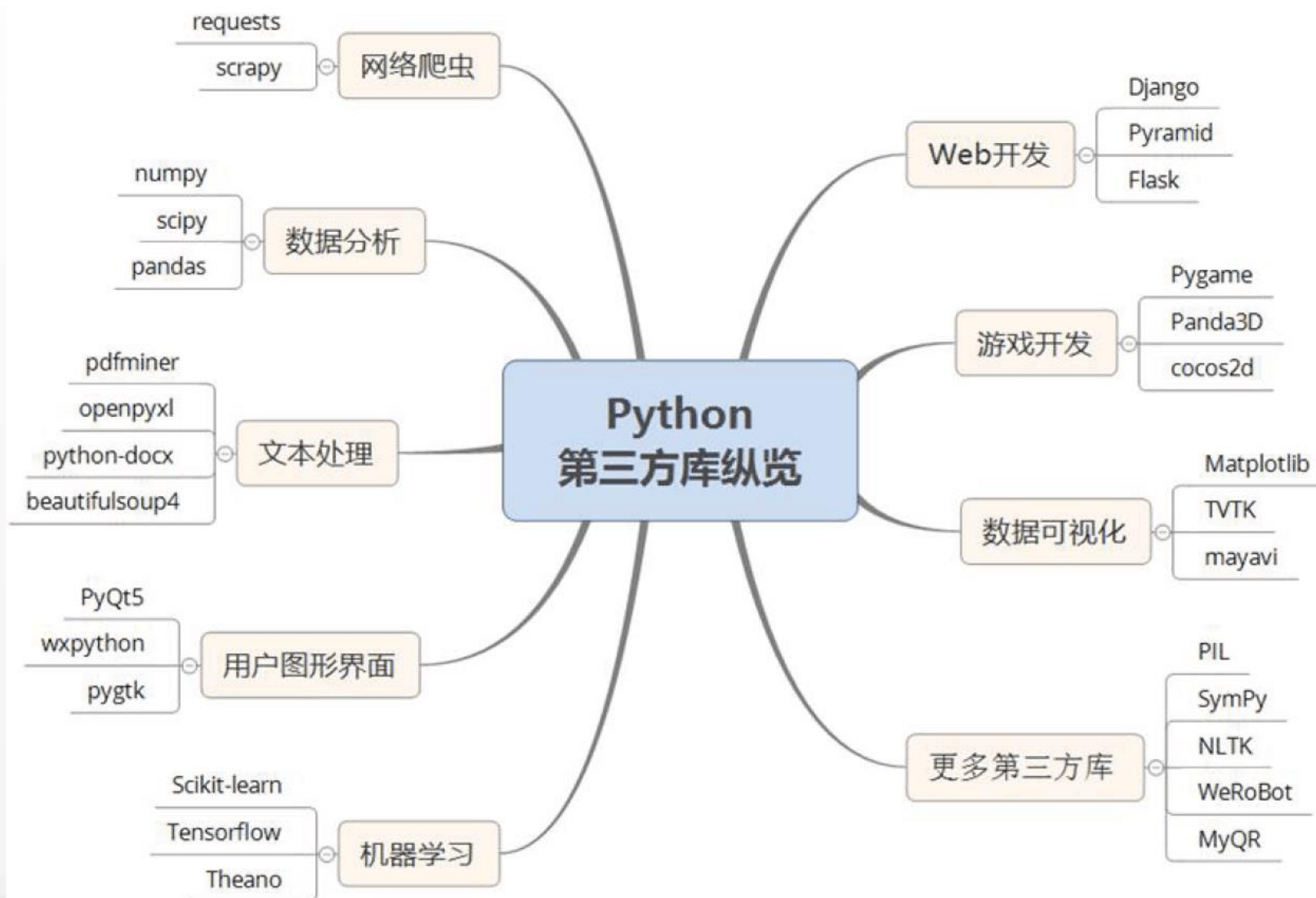

Python语言的优势

具有庞大的计算生态，可以很容易利用已有功能，避免重复“造轮子”





Python语言的优势





Python语言的缺点

- 运行速度慢
- 源代码不能加密
- GIL锁限制多线程并发



内容

- 课程介绍
- Python语言介绍
- 开发环境准备
- Python代码编写规范



Python下载

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

The screenshot shows the Python.org website's download page. At the top, there's a navigation bar with links for Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a search bar with a 'GO' button and a 'Socialize' link. A secondary navigation bar includes links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large heading 'Download the latest version for Windows' and a prominent yellow button labeled 'Download Python 3.10.6'. Below the button, there are links for other operating systems (Windows, Linux/UNIX, macOS, Other), a link for pre-releases and Docker images, and a note about Python 2.7. To the right of the text, there is an illustration of two parachutes carrying boxes, symbolizing the delivery of the software.

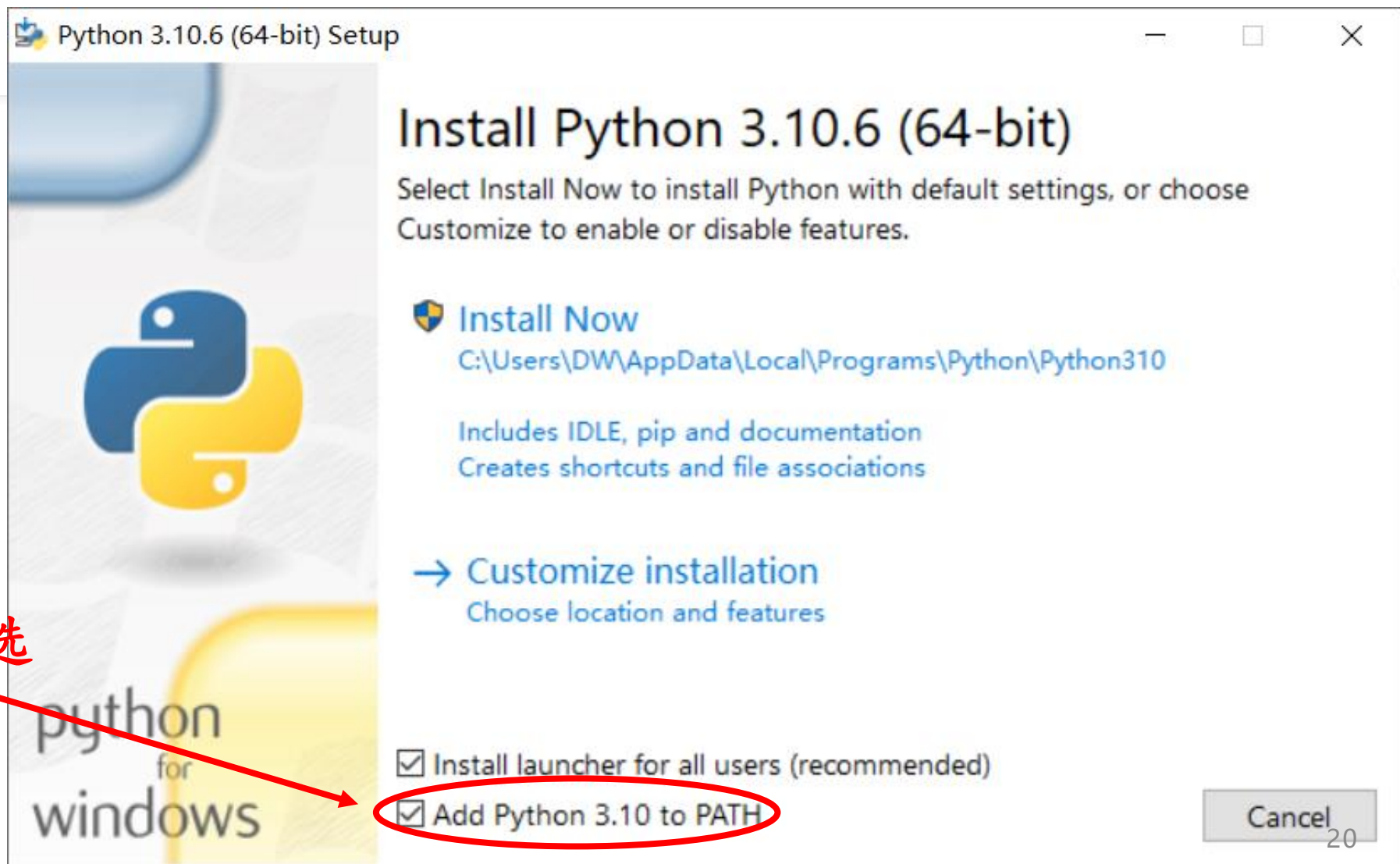
Active Python Releases

For more information visit the [Python Developer's Guide](#).

Python version	Maintenance status	First released	End of support	Release schedule
3.10	bugfix	2021-10-04	2026-10	PEP 619



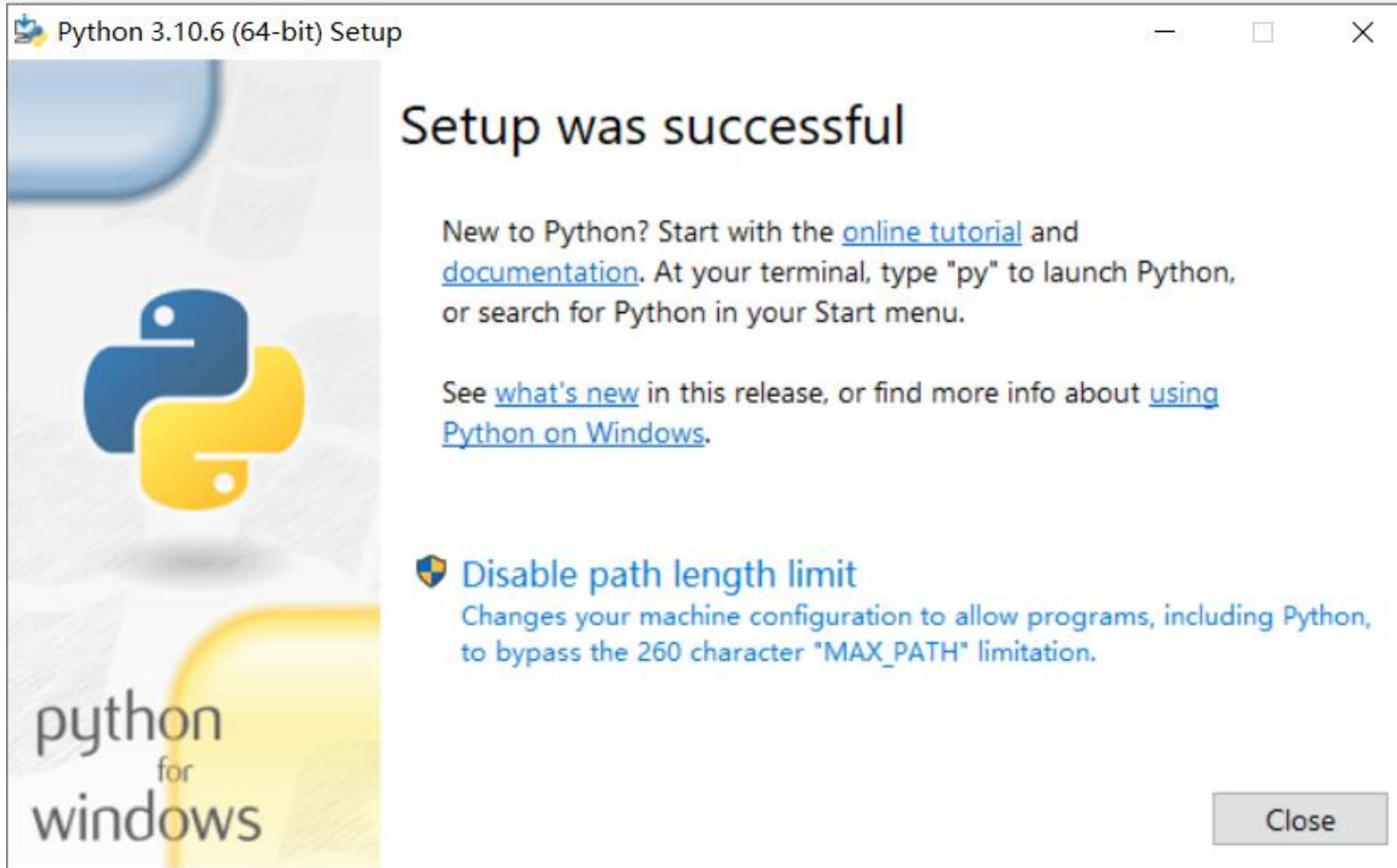
Python安装



这里需要手动勾选



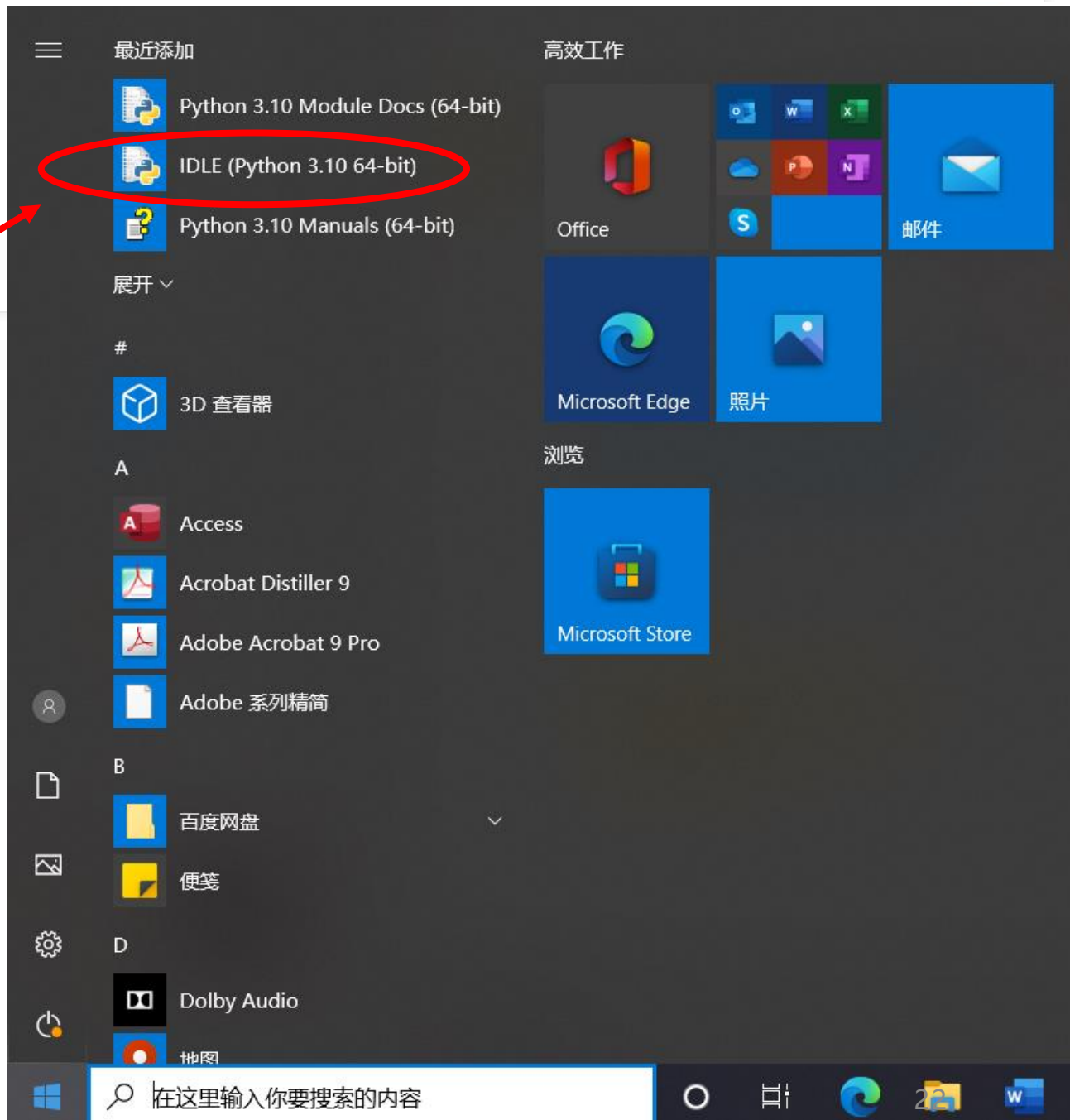
Python安装成功



Python安装成功

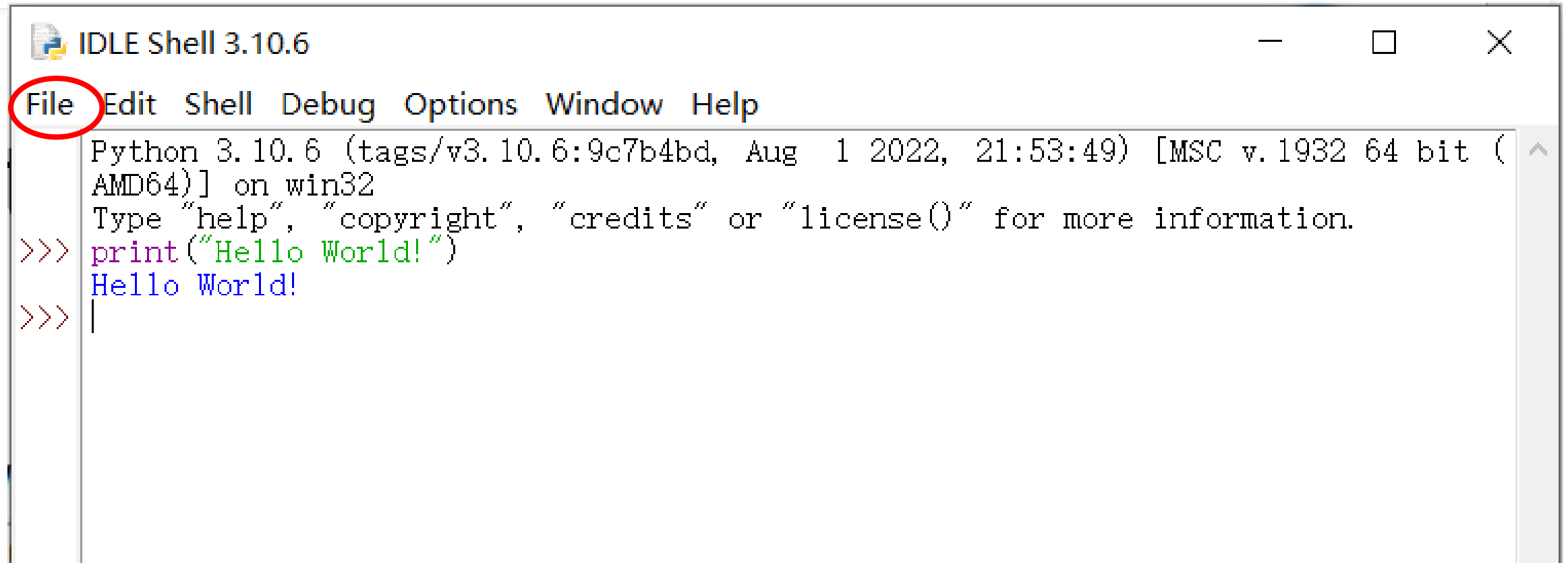
IDLE: Python自带的集成开发环境*

集成开发环境: *IDE*
(*Integrated Development Environment*)





在IDLE中执行Python语句



```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello World!")
Hello World!
>>> |
```



在IDLE中编辑及运行Python程序

turtle2.py - F:\学校\2022工作\Python程序设计.2022\代码\turtle2.py (3.10.6)

File Edit Format **Run** Options Window Help

```
import turtle
```

```
t = turtle.Pen()
```

```
t.color('blue')
```

```
t.shape('turtle')
```

```
for angle in range(0, 361, 20):
```

```
    t.forward(100)
```

```
    t.stamp()
```

```
    t.home()
```

```
    t.seth(angle)
```




命令行方式运行Python程序

- Win + R键
- 运行cmd
- 通过cd操作进入python文件所在目录
- `python XX.y`



Python集成开发环境



简化较大规模、专业领域项目开发



推荐使用：Pycharm集成开发环境

- 社区版免费
- 简单，集成度高
- 适合较大、较复杂工程

[PyCharm: Python IDE for Professional Developers by JetBrains](#)



Version: 2022.2.1
Build: 222.3739.56
17 August 2022

[System requirements](#)

[Installation instructions](#)

[Other versions](#)

[Third-party software](#)

Download PyCharm

[Windows](#)

[macOS](#)

[Linux](#)

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)

Free 30-day trial available

Community

For pure Python development

[Download](#)

Free, built on open-source



Get the Toolbox App to download PyCharm and its future updates with ease

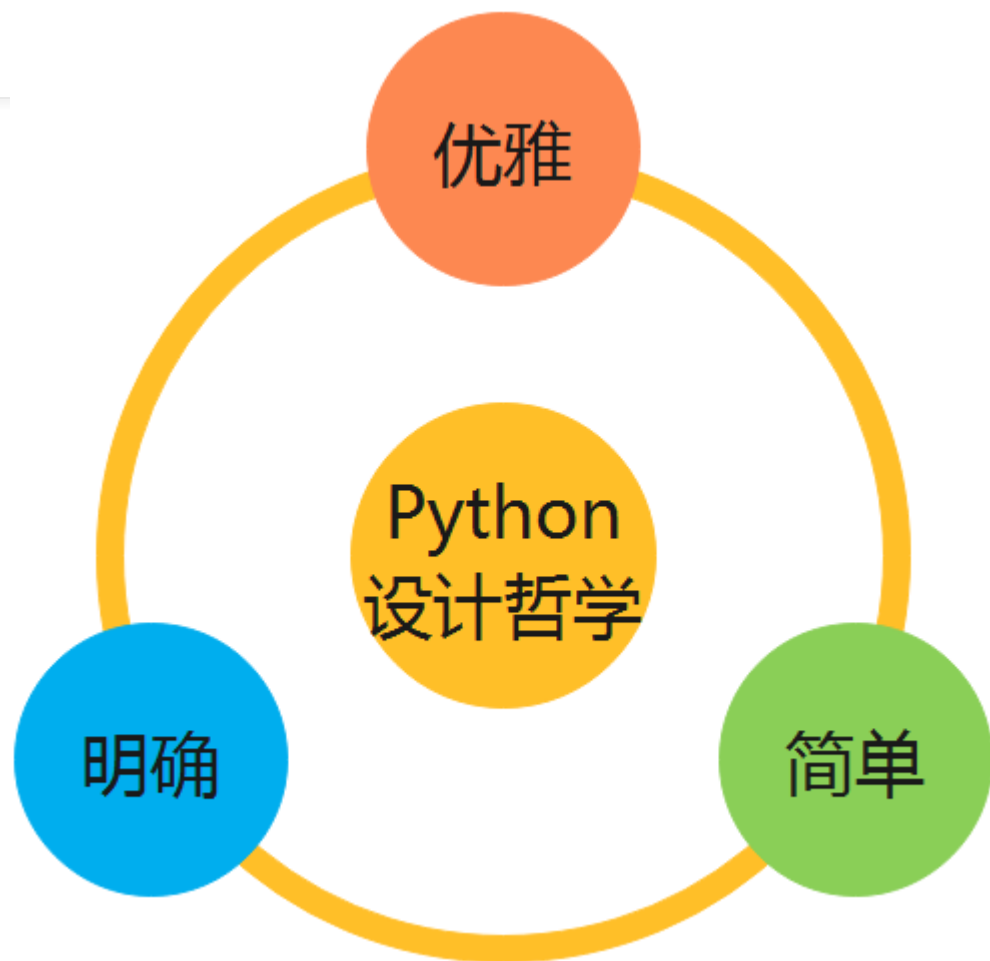


内容

- 课程介绍
- Python语言介绍
- 开发环境准备
- Python代码编写规范



Python的设计哲学



Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

... ..

The Zen of Python, by Tim Peters



Python代码编写规范

PEP 8 -- Style Guide for Python Code

- 分号：不要在行尾加分号，一行一个语句
- 行长度：每行不超过80个字符
- 括号：宁缺毋滥使用括号
- 缩进：缩进体现逻辑关系
- 空行：适当使用空行增强程序的可读性
- 空格：按照标准排版格式使用空格
- 注释：对模块、函数、行内注释使用正确风格
- 命名：应避免的名称及命名约定



Python代码编写规范： 缩进和空格

缩进是Python语法规则的一部分

Python采用代码缩进和冒号来区分代码之间的层次，作用等于其它语言中的{}或者begin...end

```
if True:
    print("Hello girl!")
else:
    print("Hello boy!")
```

语句行首不能出现多余空格



Python代码编写规范： 注释

单行注释：使用#表示。作为单独一行放在被注释代码之上，或放在语句或表达式之后

```
# 这是单行注释
```

多行注释：使用三个单引号或三个双引号表示。注释内容过多时，采用多行注释

```
'''  
这是使用三个单引号的多行注释  
'''  
  
"""  
这是使用三个双引号的多行注释  
"""
```




Python代码编写规范： 行长度

利用隐式行连接将长语句分行

```
foo_bar(self, width, height, color='black', design=None, x='foo',  
        emphasis=None, highlight=0)
```

```
if (width == 0 and height == 0 and  
    color == 'red' and emphasis == 'strong'): print("ok")
```



Python代码编写规范：宁缺毋滥使用括号

```
if x:  
    bar()  
while x:  
    bar()  
if x and y:  
    bar()  
if not x:  
    bar()  
return foo
```



```
if (x):  
    bar()  
while (x):  
    bar()  
if (x and y):  
    bar()  
if not(x):  
    bar()  
return (foo)
```





Python命名规则：应避免的名称

- 单字符名称, 除了计数器和迭代器
- 包/模块名中的连字符(-)
- 双下划线开头并结尾的名称(Python保留, 例如__init__)



Python命名规则：命名约定

- 所谓"内部(Internal)"表示仅模块内可用, 或者, 在类内是保护或私有的
- 用单下划线(_)开头表示模块变量或函数是protected的
- 对类名使用大写字母开头的单词(如CapWords), 但是模块名应该用小写加下划线的方式(如lower_with_under.py).



Python编码规范使用原则

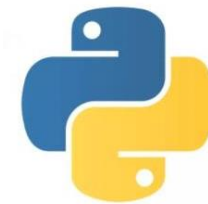
编码规范很繁杂

不同人使用的规范常略有不同
难以在各种情况坚持使用规范

编码规范的利弊权衡

有利：程序可读性或者正确性增加
不利：繁琐增加成本

基本规则：可读性强，便于理解和调试（避免不恰当的“炫技”）



课程概述：小结

- Python是当前最热门的编程语言
- Python是一个开放、免费的动态、解释性的脚本语言
- Python拥有庞大的计算生态，可以方便地利用各种第三方库实现各专业领域的项目开发
- Python编程哲学：优雅，明确，简单