



Chap1 Programming & Python

第1章 程序设计和Python语言

Department of Computer Science and Technology
Department of University Basic Computer Teaching
Nanjing University

- 教材：
 - 《Python程序设计》，张莉主编，高等教育出版社，2022年第二版
 - 《Python程序设计实验指导》，张莉、陶烨编，高等教育出版社，2022年第一版
- 课程网站
 - <https://yiqinnju.github.io/course/python/python.html>
- 联系信息：
 - yiqincs at nju.edu.cn
 - 仙林校区计算机系楼501

- 课程时间：
 - 理论课：周一5-6节（每周）+周三5-6节（单周）
 - 上机课：周一7-8节（每周）
- 成绩评定：
 - 平时成绩（40%）
 - 课后作业：类似于签到，只要按时缴纳作业即可拿满分
 - 课程项目：期中/期末会各有一次中等规模的程序设计项目，可组成团队
 - 期末考试成绩（60%）：开卷考试，可以携带任何纸质版本材料入场

1.1

程序设计概述

- **计算机中的程序**是指按特定的算法用某种计算机语言描述完成指定任务的处理过程。

- **程序语言分类**

**按对机器依赖性、
接近自然语言的程度、
语言的发展过程**

{ 机器语言
汇编语言
高级语言

- CPU能理解且能直接执行的指令集合
- 用**机器语言**编写的程序形式：

10110000

00001101

00101100

00000101

11110100

依赖于具体计算机

难记忆

易出错

阅读理解困难

编程者需了解特定计算机的硬件及指令系统

- 汇编语言是一种符号化的机器语言，用助记符号代替指令操作码、操作数。
- 用**汇编语言**编写的程序形式：

MOV AL,13D

SUB AL,5D

HLT



汇编语言与机器语言的异同？

- 高级语言接近于自然语言和数学语言，在一定程度上与具体计算机无关的符号化语言。
- 用高级语言编写的程序形式（以C语言为例）：

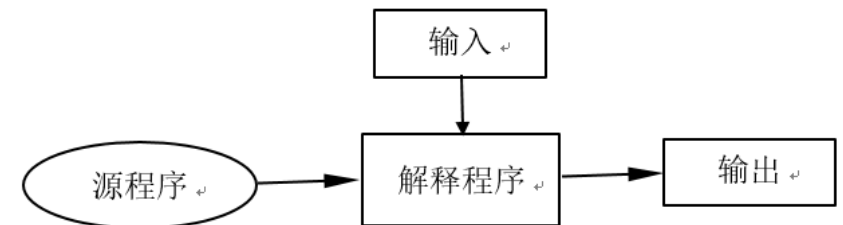
```
#include <stdio.h>
int main(void)
{
    printf("%d\n", 13-5);
    return 0;
}
```


易学易用
代码易理解
易调试、修改和移植，并与机器无关

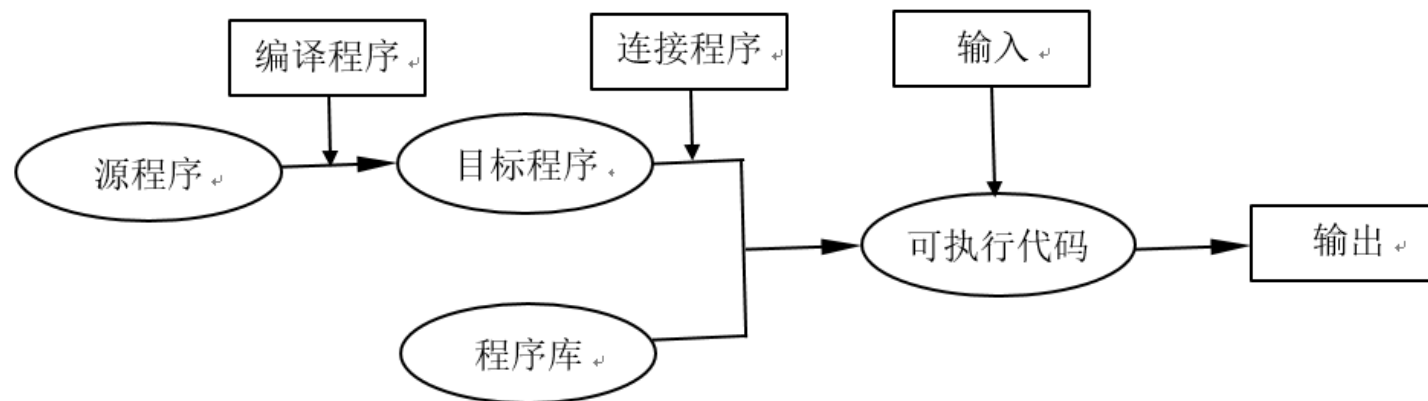
不支持对硬件的直接操作
代码需要翻译指令序列后才能执行

- 高级语言的翻译方式

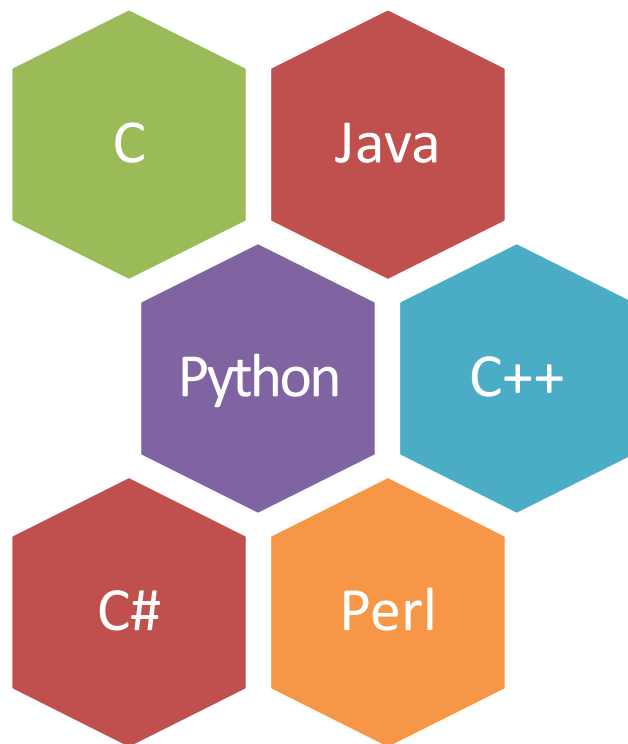
- 解释方式 (Interpreter 解释程序)



- 编译方式 (Compiler 编译程序或称编译器)



程序设计语言



- 就程序本质而言
 - 刻画计算（数值计算与非数值计算，与处理同义）过程的处理对象（数据）与处理规则（算法）。
- 瑞士计算机科学家 Niklaus Wirth认为：
 - **程序 = 算法 + 数据结构**

程序 = 算法 + 数据结构

- **算法**是解决问题的步骤和方法，是指令的有限序列
- **数据结构**则是数据的组织形式，涉及操作对象以及它们之间的关系和操作

程序 = 算法 + 数据结构

13

计算 $1+2+\dots+100$ 的和，算法和数据结构设计：

S1: $\text{sum} \leftarrow 0$

S2: $i \leftarrow 1$

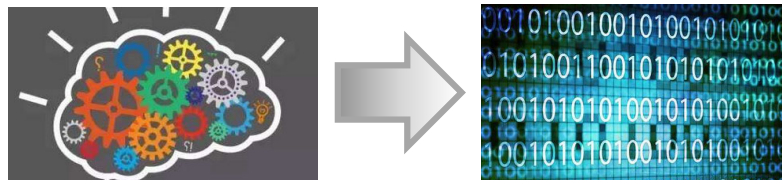
S3: 如果 $i \leq 100$ 执行 $\text{sum} \leftarrow \text{sum} + i$ ，如果 $i > 100$ 转S5

S4: $i \leftarrow i + 1$ ，转S3

S5: 输出sum

程序 = 数据结构 + 算法

- 算法是程序的**灵魂**，在程序设计中具有**核心**地位
- 实际应用中用户首先关心的是自己的数据能够被处理，其次才关心如何处理，用什么工具处理数据，所以从应用角度出发，程序设计的**重心**已转移到**数据**上了



- 结构化程序设计
 - 自顶向下 (top-down) , 逐步细化(stepwise refinement), 由抽象到具体的功能分解过程, 使用三种控制结构即顺序、选择和循环控制构造程序

- 面向对象程序设计
 - 是目前计算机开发中使用的主流方法，它是一种对现实世界理解并抽象的方法，将对象作为程序的基本单元，并将程序和数据封装在内，以提高软件的重用性、灵活性和扩展性。


```
类 狗狗
{
    过程 跑步;
    过程 打滚;
    .....
}
```

```
类 吉娃娃 继承自 狗狗
{
    过程 卖萌;
    过程 咬人;
    .....
}
```

一只叫
旺财的
吉娃娃

1.2

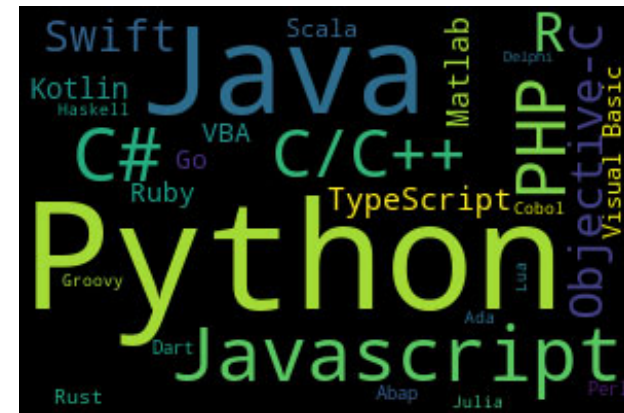
PYTHON简介

Python指数——PYPL

19

Worldwide, Feb 2022 compared to a year ago:


Rank	Change	Language	Share	Trend
1		Python	28.52 %	-1.7 %
2		Java	18.12 %	+1.2 %
3		JavaScript	8.9 %	+0.4 %
4	↑	C/C++	7.62 %	+1.1 %
5	↓	C#	7.39 %	+0.6 %
6		PHP	5.81 %	-0.3 %
7		R	4.04 %	+0.2 %
8		Objective-C	2.46 %	-1.1 %
9		Swift	2.03 %	+0.0 %
10		TypeScript	1.94 %	+0.2 %



编程语言流行指数 ([PyPL](#))
Feb 2022

Python指数——TIOBE

20

Feb 2022	Feb 2021	Change	Programming Language		Ratings	Change
1	3	▲		Python	15.33%	+4.47%
2	1	▼		C	14.08%	-2.26%
3	2	▼		Java	12.13%	+0.84%
4	4			C++	8.01%	+1.13%
5	5			C#	5.37%	+0.93%
6	6			Visual Basic	5.23%	+0.90%
7	7			JavaScript	1.83%	-0.45%
8	8			PHP	1.79%	+0.04%
9	10	▲		Assembly language	1.60%	-0.06%
10	9	▼		SQL	1.55%	-0.18%

Feb 2022

Python应用实例

21



1.2.1 PYTHON的历史和特性

python 的诞生

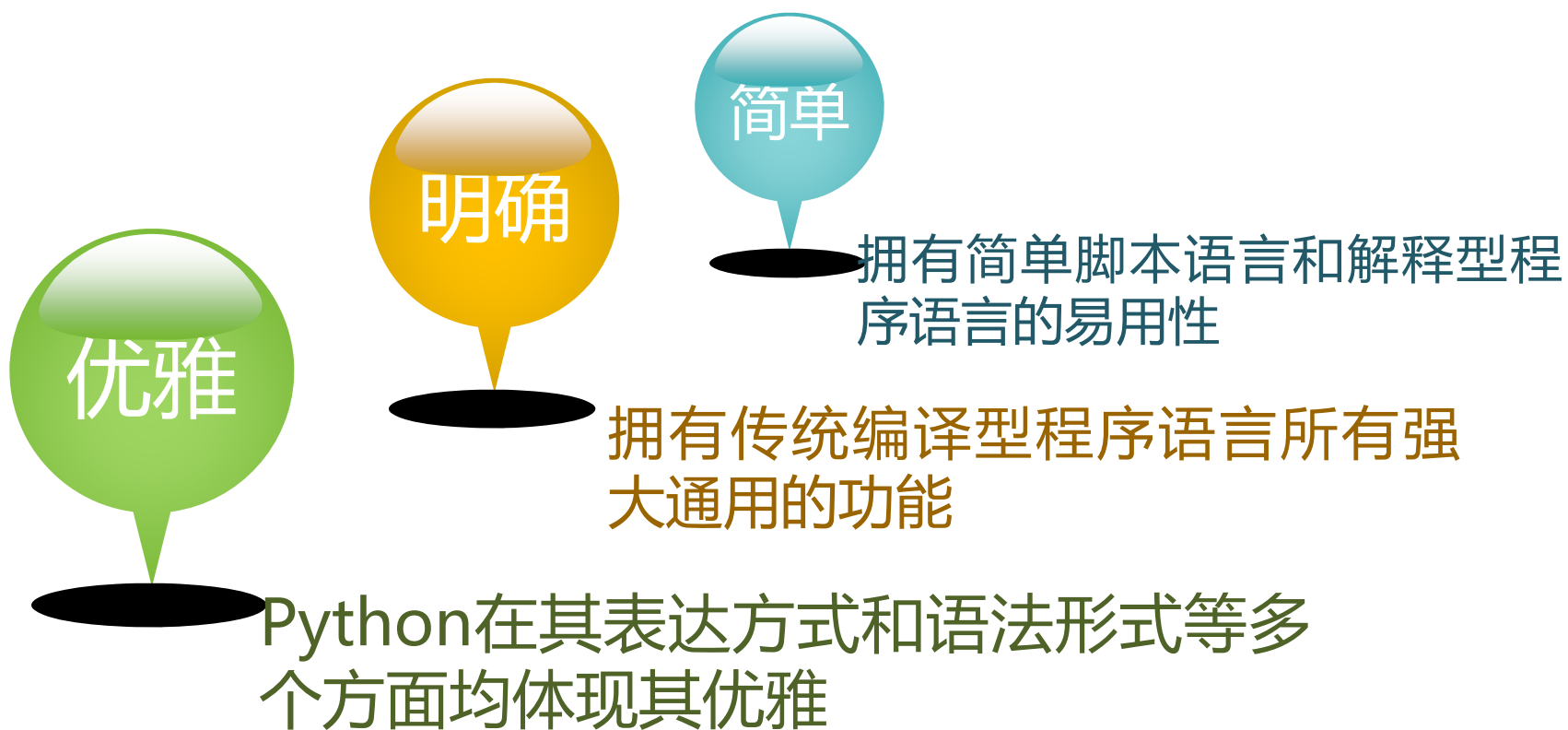


Guido van Rossum

- 第1个Python编译器/解释器于1991年诞生
- Python名称来自Guido挚爱的电视剧Monty Python's Flying Circus
- Python介于C和Shell之间、功能全面、易学易用、可扩展

Python的哲学

24



- 胶水语言 Glue Language

- 很容易和其他著名的程序语言连接 (C/C++)，集成封装

- 脚本语言 Script Language

- 高级脚本语言，比脚本语言只能处理简单任务强大

- 面向对象语言 Object-Oriented Language

- 完全支持继承、重载、派生、多继承

Python的库——具有强大吸引力

26



Python是一门通用程序语言，有众多程序库支持，可用于基础性、前瞻性的科研工作

Python提供丰富的库，解决人工智能领域中大量数据处理和科学计算



Python的库——具有强大吸引力

27



Python提供各种库，具有极其便捷强大的数据处理和统计功能

Python提供的re模块能支持正则表达式



1.2.2 PYTHON的版本

Python的2个版本

29



Python 2.x



Python 3.x



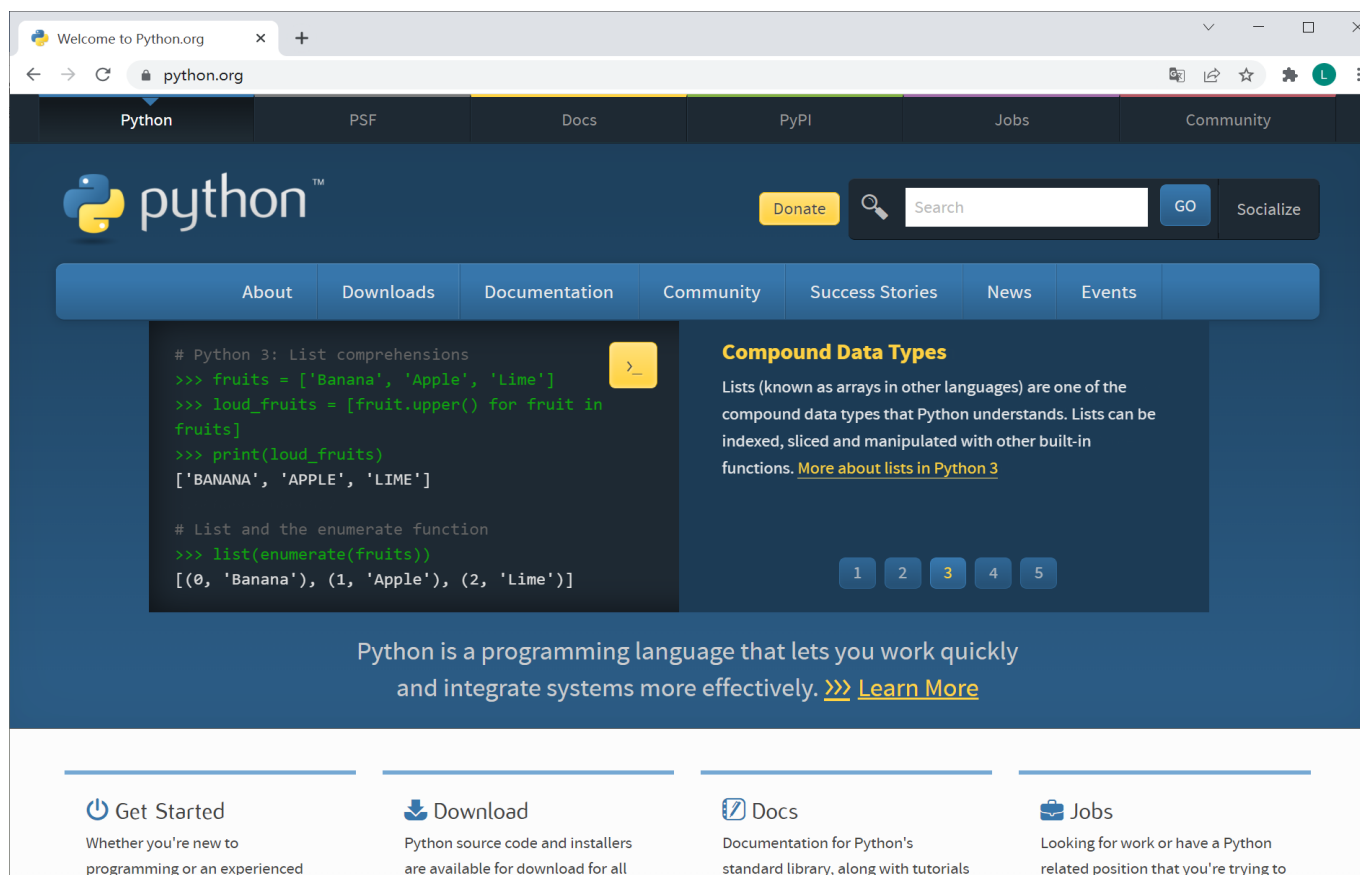
互不兼容

本书为Python3.x



Python官方网站

30



Python 格言

The Zen of Python

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!

```
>>> import this
```

by Tim Peters

1.3

PYTHON开发 环境和运行方式

经典 Hello World

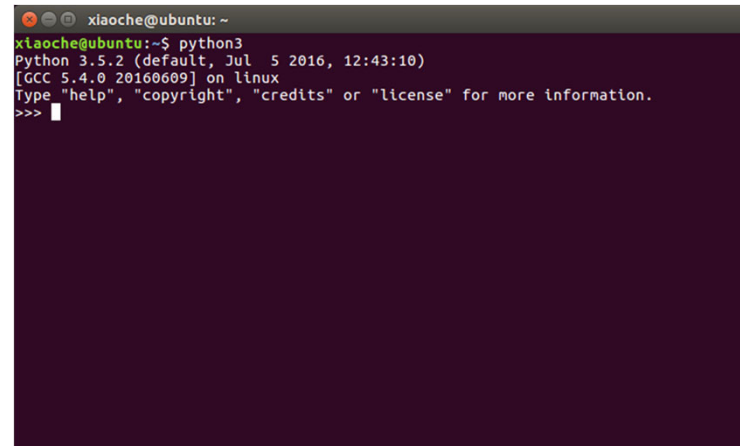
33

```
myString = 'Hello, World!'
print(myString)
```

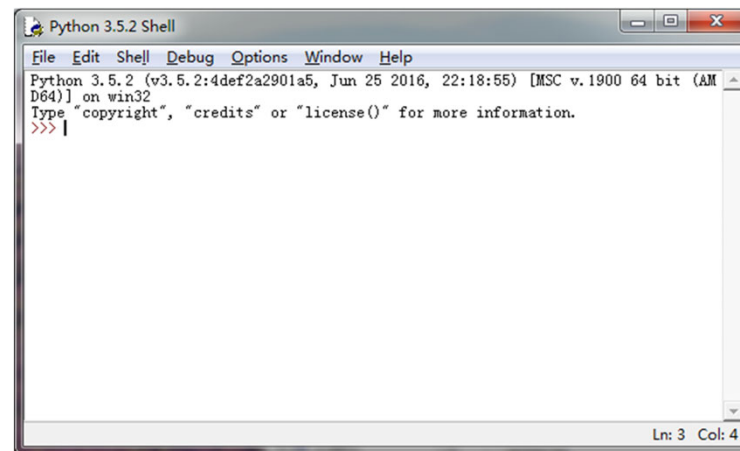
1.3.1 PYTHON开发环境

Python开发环境1

- Mac OS & Linux
 - \$ python
 - \$ python3
- Python内置IDE
 - IDLE



```
xiaoche@ubuntu: ~  
xiaoche@ubuntu:~$ python3  
Python 3.5.2 (default, Jul 5 2016, 12:43:10)  
[GCC 5.4.0 20160609] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> |
```



```
Python 3.5.2 Shell  
File Edit Shell Debug Options Window Help  
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit (AMD64)] on win32  
Type "copyright", "credits" or "license()" for more information.  
>>> |
```

Python 集成开发环境 (IDE)

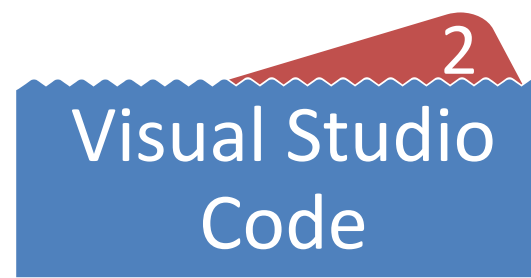
36



Python编辑器

IDE

- 自动补全和智能提示等编辑和调试功能
- 小巧，定制方便，但功能相对简单



Python 集成开发环境 (IDE)

38

- 包含编辑器，一般还具有项目管理和版本控制等功能
- 通常还预装了完成某些任务的扩展库、文档和教程，给程序开发带来很大便利。



Anaconda

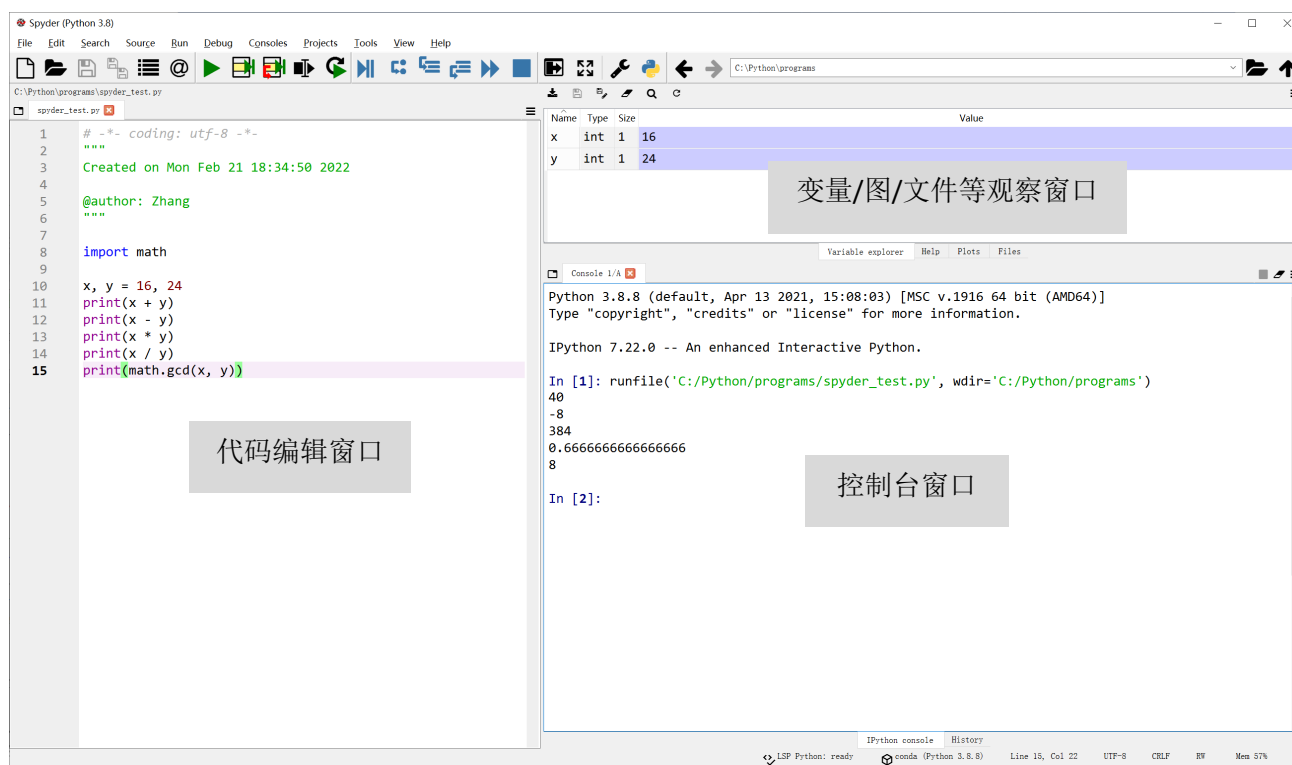
- 安装
- download the installer

官网或清华镜像站点

– install and use

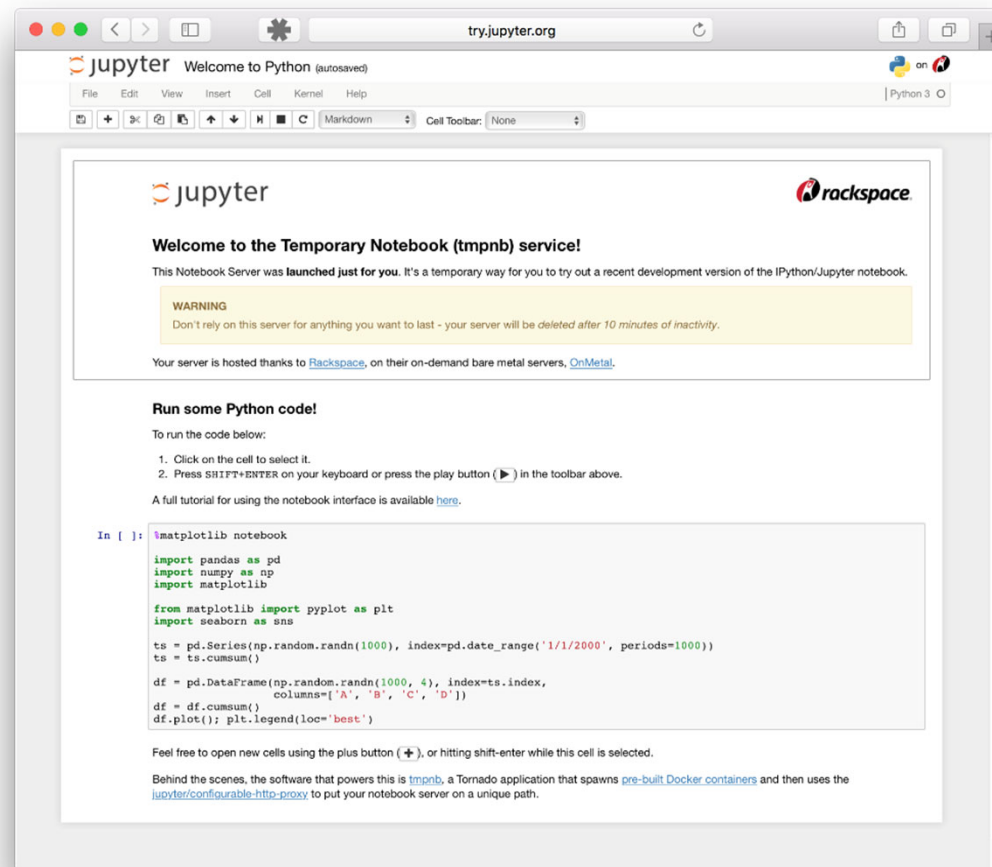
勾选 “Add Anaconda to my PATH environment variable”

安装后一般使用Spyder Python解释器编辑和执行程序



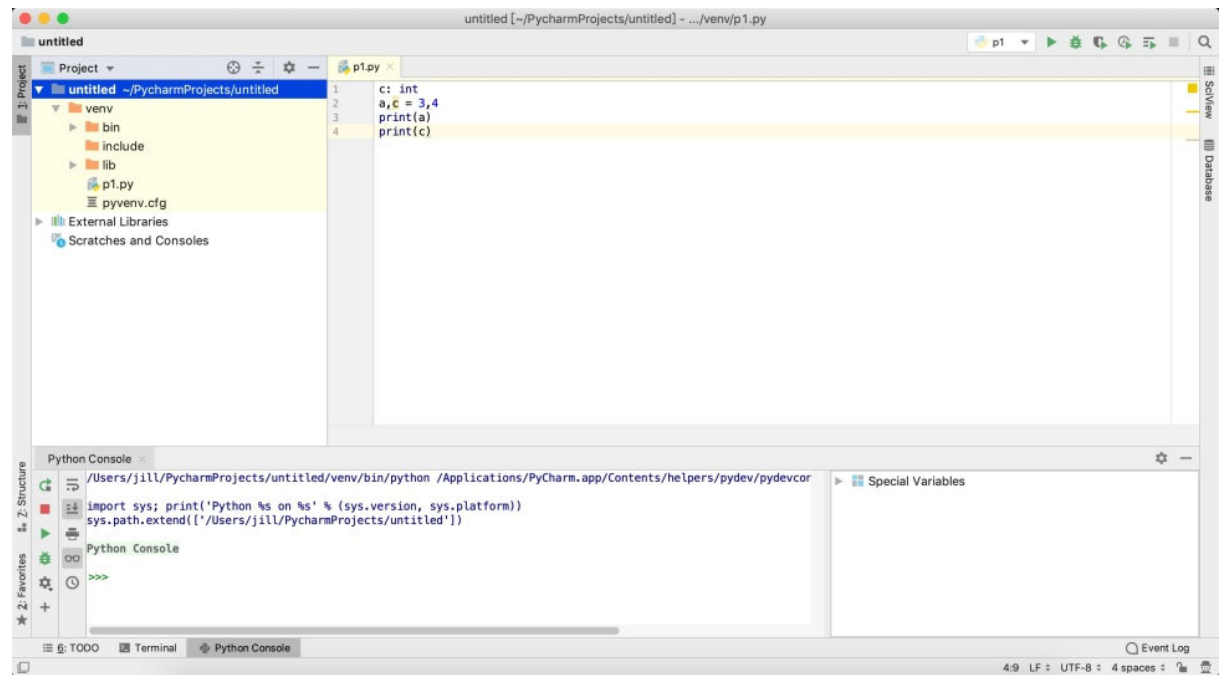
Jupyter Notebook

40




Python IDE

PyCharm



- 用pip/conda在**操作系统终端**安装第三方包，在官网<https://pypi.org/>页面搜索，例如：



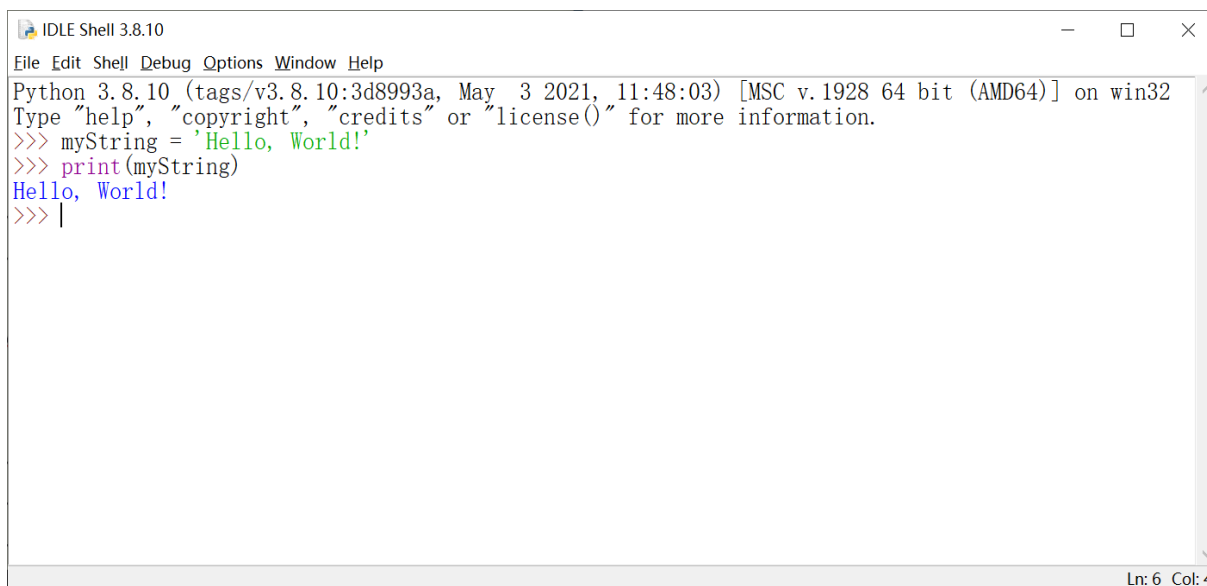
```
imac — pip install atx — 80x24
Last login: Wed Apr 10 09:55:39 on console
iMacdeiMac:~ imac$ pip install atx
Collecting atx
  Downloading https://files.pythonhosted.org/packages/86/83/38e68ae1a950aef87375
4ab039bf4da5baa6029ae396de1d6645f9aa23df/atx-1.2.0-py3-none-any.whl (2.2MB)
    6% |███| 133kB 12kB/s eta 0:02:43
```

1.3.2 PYTHON运行方式

Python的运行方式（一）

44

Shell方式

A screenshot of the IDLE Shell 3.8.10 window. The window has a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area shows the following text:

```
Python 3.8.10 (tags/v3.8.10:3d8993a, May 3 2021, 11:48:03) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> myString = 'Hello, World!'
>>> print(myString)
Hello, World!
>>> |
```

The status bar at the bottom right indicates 'Ln: 6 Col: 4'.

- Shell是交互式的解释器
- 输入一行命令，解释器就解释运行出相应结果

Python的运行方式 (二)

45

文件执行方式

- 在Python的IDE环境中，创建一个以py为扩展名的文件
- 用Python解释器在Shell中运行出结果



经典 Hello World



```
>>> myString = 'Hello, World!'
>>> print(myString)
Hello, World!
>>> myString
'Hello, World!'
```



```
# Filename: helloworld.py
myString = 'Hello, World!'
print(myString)
```

- 都是解释运行方式
 - 如果整个代码段较短，则优先考虑选择用Shell交互方式
 - 如果代码段较长，则建议使用文件执行方式



1.4

小结

- 程序设计语言
- 程序设计
- Python简介
- Python开发环境
- Python运行方式

