

声明：本课程版权归华算科技所有，仅限个人学习，严禁任何形式的录制、传播和账号分享。一经发现，平台将依法保留追究权，情节严重者将承担法律责任。

Python与机器学习

——Python简介

华算科技 黄老师
2022年2月21日



课程简介

这是一个数据爆炸的时代

近30年来，人类生产的信息已超过过去5000年信息生产的总和。

COD Home

Home
What's new?

Accessing COD Data

Browse
Search
Search by structural formula

Add Your Data

Deposit your data
Manage depositions

National Institute of Standards and Technology
Computational Chemistry Comparison and Benchmark Database

Release 21 August 2020
NIST Standard Reference Database 101

- I Introduction
- II Experimental data
- III Calculated data
- IV Data comparisons
- V Cost comparisons
- VI Input and output files
- VII Tutorials and Units
- VIII Links to other sites
- IX Feedback
- X Older CCCDB versions
- XI Geometries
- XII Vibrations
- XIII Reaction data
- XIV Entropy data
- XV Bibliographic data
- XVII Ion data
- XVIII Bad calculations
- XX Index of properties
- XXI H-bond dimers
- XXII Oddities

[NIST policy on privacy, security, and accessibility.](#)

© 2013 copyright by the U.S. Secretary of Commerce on behalf of the United States of America. All rights reserved.

Precomputed vibrational scaling factors

The following tables list the vibrational frequency scaling factor and its uncertainty (1 σ) as determined from data in the CCCBDB. Click on an entry for the list of molecules and frequencies used to compute the vibrational scaling factor.

Methods with predefined basis sets		
semi-empirical	AM1	0.954 ± 0.059
	PM3	0.974 ± 0.077
	PM6	1.062 ± 0.105
molecular mechanics OREINDING		0.936 ± 0.170

Methods with standard basis sets					
hartree fock	STO-3G	0.817 ± 0.048	0.906 ± 0.044	0.903 ± 0.032	0.903 ± 0.032
	HF	242m 256m	264m 260m	267m 268m	268m 268m
	ROHF		0.907 ± 0.178		
	LSDA	0.896 ± 0.082	0.984 ± 0.068	0.982 ± 0.065	0.980 ± 0.065
	BLYP	0.925 ± 0.050	0.995 ± 0.050	0.994 ± 0.044	0.992 ± 0.044
	B1B95	0.883 ± 0.049	0.957 ± 0.039	0.955 ± 0.031	0.954 ± 0.031
B3LYP	0.892 ± 0.051	0.965 ± 0.043	0.962 ± 0.035	0.962 ± 0.035	

DNA Search Options:

Polymer

- All
- DNA Only
- Protein DNA Complexes
- Drug DNA Complexes
- Hybrids and Chimera
- Peptide Nucleic Acid / Mimetics

Protein Function

- All
- Enzymes
- Structural
- Regulatory

Structural Features

- All
- Single Stranded
- A DNA
- B DNA
- Z DNA
- Other Double Helical Structures
- Triple helices
- Quadruple helices

Experimental Method

- All
- XRAY
- NMR

Text Search

Filter results by text search

Use this option to narrow your results down

Polymer Type: All + Protein Function: All + Structural Features: All

Results: 8021 [Download results as an excel file](#)

NDB ID:	PDB ID:	Title:
7D8T	7D8T	MITF bHLH
Release: 2021-10-13		Classification: TRANSCRIPTION FACTOR Authors: Liu, Z., Chen Pan, L., Fang Citation: Targeting the To Be Published Experiment: X-RAY DIFFRACTION Resolution: 2.79 Å R work: 0.263 R free: 0.321
7F2F	7F2F	The complete set of human
Release: 2021-10-13		Classification: TRANSCRIPTION FACTOR Authors: Guo, W., Xue Citation: Crystal structure of Actin Crystallin Experiment: X-RAY DIFFRACTION Resolution: 2.79 Å R work: 0.263 R free: 0.321
7N8S	7N8S	LINE-1 endonuclease domain complex with DNA
Release: 2021-10-13		Classification: HYDROLASE/DNA Authors: Miller, I., Totrov, M., Korotchkina, L., Kazyulkin, D.N., Gudkov, A.V., Korotchkina, L. Citation: Structural dissection of sequence recognition and catalytic mechanism Nucleic Acids Res. pp. - 2021 Experiment: X-RAY DIFFRACTION Resolution: 2.79 Å R work: 0.263 R free: 0.321
7N94	7N94	LINE-1 endonuclease domain complex with DNA
Release: 2021-10-13		Classification: HYDROLASE/DNA Authors: Miller, I., Totrov, M., Korotchkina, L., Kazyulkin, D.N., Gudkov, A.V., Korotchkina, L. Citation: Structural dissection of sequence recognition and catalytic mechanism Nucleic Acids Res. pp. - 2021 Experiment: X-RAY DIFFRACTION Resolution: 2.79 Å R work: 0.263 R free: 0.321

您实验室的研究中使用数据科学最大的障碍是什么？（单选）

A. 了解可用的工具

B. 缺乏数据科学的专业知识

C. 该领域变化过于迅速

D. 优先级不够

E. 其它

您实验室的研究中使用数据科学最大的障碍是什么？（单选）

A. 了解可用的工具	34%
B. 缺乏数据科学的专业知识	24%
C. 该领域变化过于迅速	14%
D. 优先级不够	17%
E. 其它	11%

数据来源：DASSAULT SYSTÈMES

课程安排

2.21

- 上午：Python简介，编写Python程序，变量，数据结构，条件语句
- 下午：循环语句，函数，Python处理数据，文件读写

2.22

- 上午：numpy库，pandas库，谱数据平滑，matplotlibs库
- 下午：机器学习简介，sklearn库，最小二乘原理，线性回归

2.23

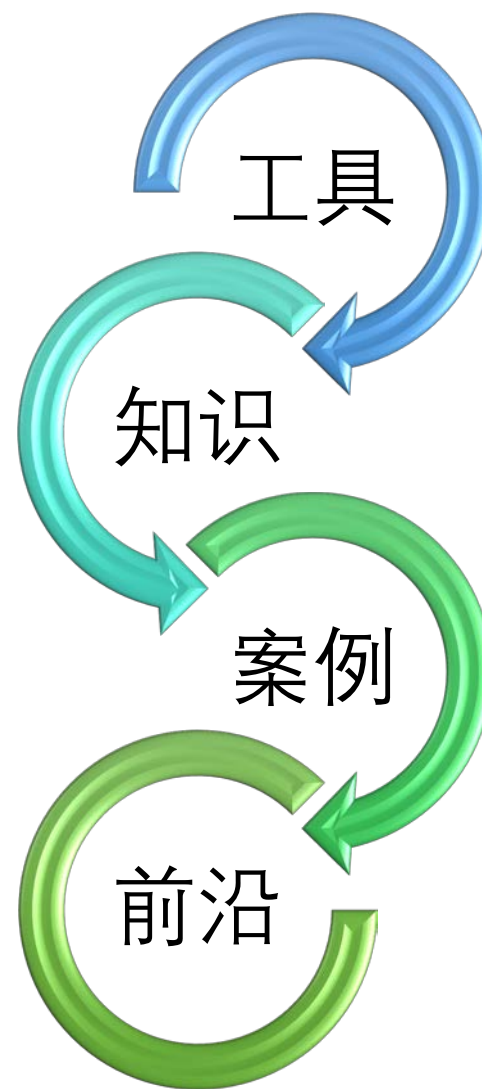
- 上午：交叉验证，决策树分类，支持向量机算法

2.24

- 上午：案例1——预测d带中心
- 下午：高通量筛选，matminer库

2.25

- 上午：案例2——预测体模量，机器学习前沿



1. Python介绍
2. Python与其它语言
3. 编写Python程序

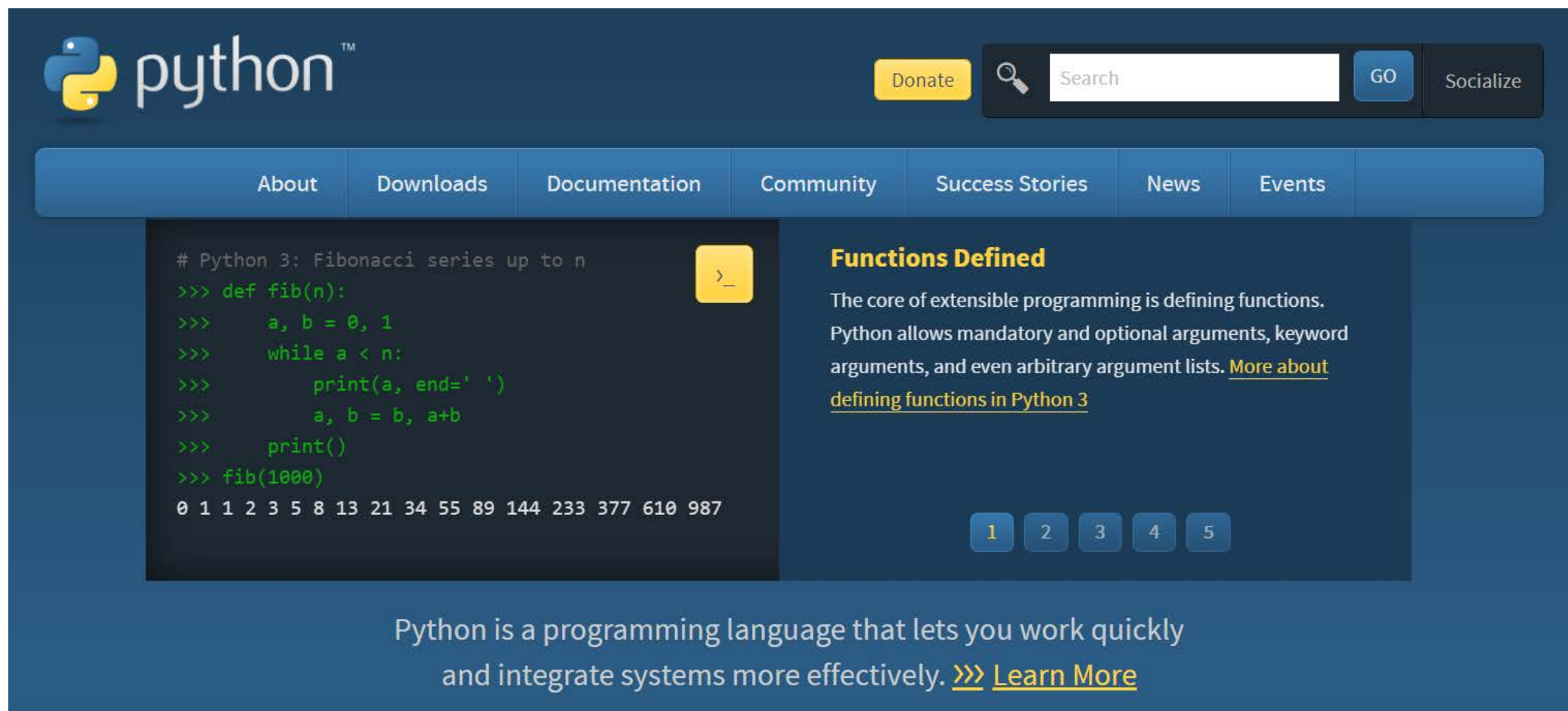
1. Python介绍

2. Python与其它语言

3. 编写Python程序

Python

Python 是一种编程语言，可以加快我们的工作并提高系统的效率。



The image is a screenshot of the Python.org homepage. At the top left is the Python logo and the word "python" with a trademark symbol. To the right are a "Donate" button, a search bar with a magnifying glass icon and a "GO" button, and a "Socialize" button. Below this is a navigation bar with links: "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events". The main content area is split into two columns. The left column contains a code block with a yellow terminal icon on the right. The code is a Python 3 script for a Fibonacci series. The right column has the heading "Functions Defined" and a paragraph about defining functions, with a link "More about defining functions in Python 3". Below the paragraph are five numbered buttons (1-5). At the bottom of the page is a footer with the text: "Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)".

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

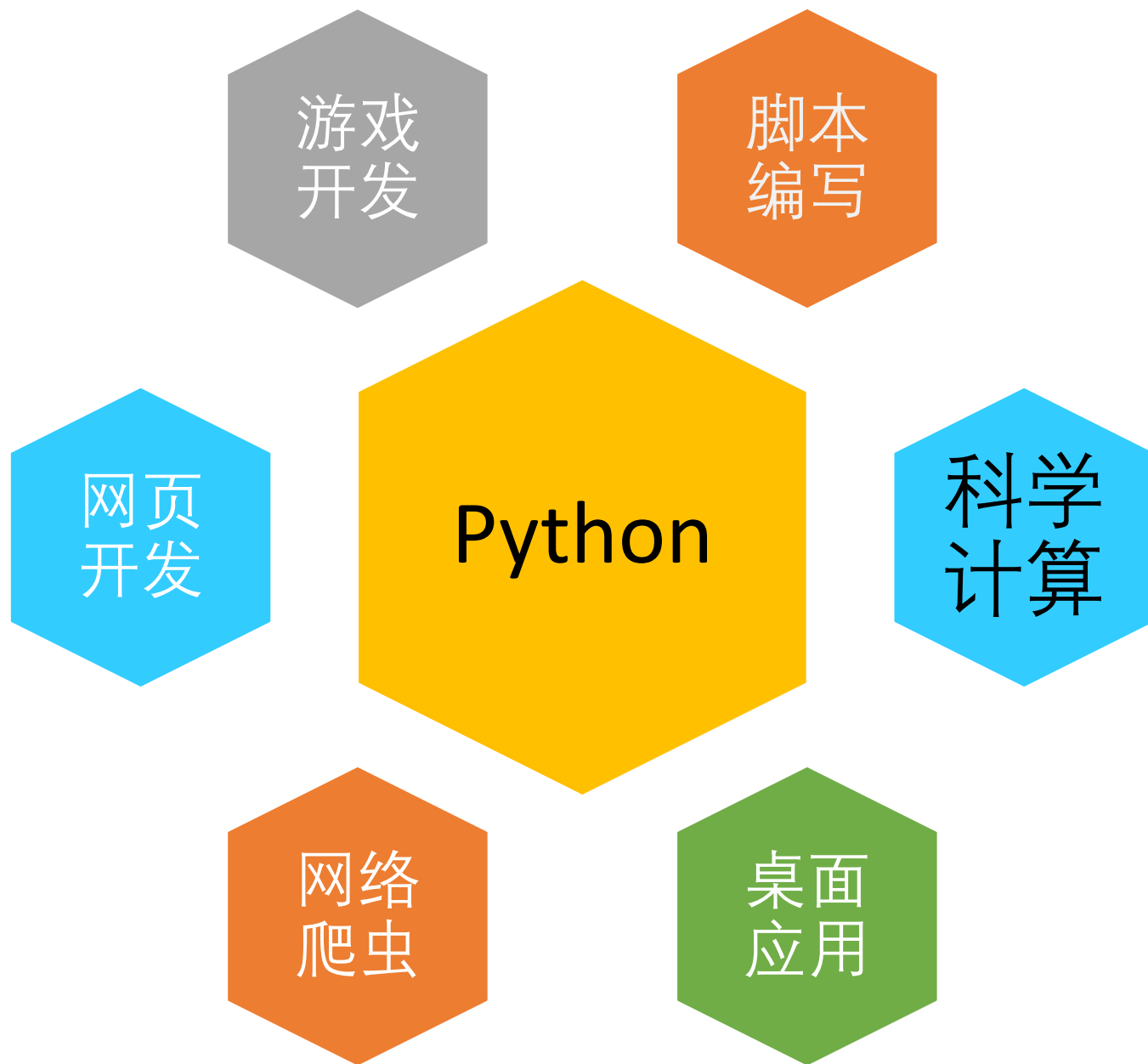
Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

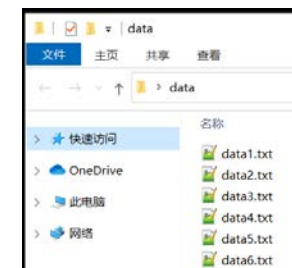
Python功能



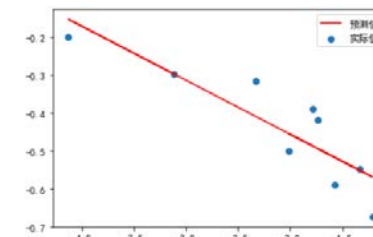
• 数据处理

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

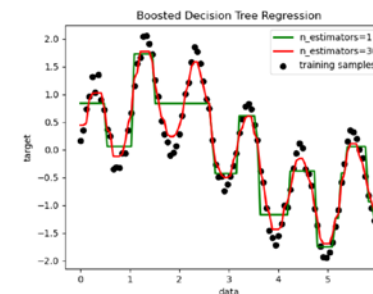
• 文件批处理



• 可视化



• 机器学习



为什么叫Python?

python 的翻译

名词

蟒蛇 python, boa

蟒 python, boa

精 essence, extract, demon, daemon, fiend, python



Python



Apple



天猫

Anaconda: 免费开源的Python和R语言的发行版本



Anaconda (中文名: 大蟒蛇)

为什么叫Python?



吉多·范罗苏姆(1956-)

1982年 阿姆斯特丹大学
数学和计算机科学硕士

1991年 Python

2005年 Google

2013年 Dropbox 2019年 退休

2020年 Microsoft



Monty Python's Flying Circus(1969-1974)

Python版本

1991年，第一个Python解释器诞生，由C语言实现

1994年1月，Python1.0版本发布

2000年10月，Python2.0发布

2008年12月，Python3.0发布

注意，Python3并不向后兼容Python2。

2020年1月1日，Python2停止更新和维护。

简洁
优美
容易使用

1. Python介绍

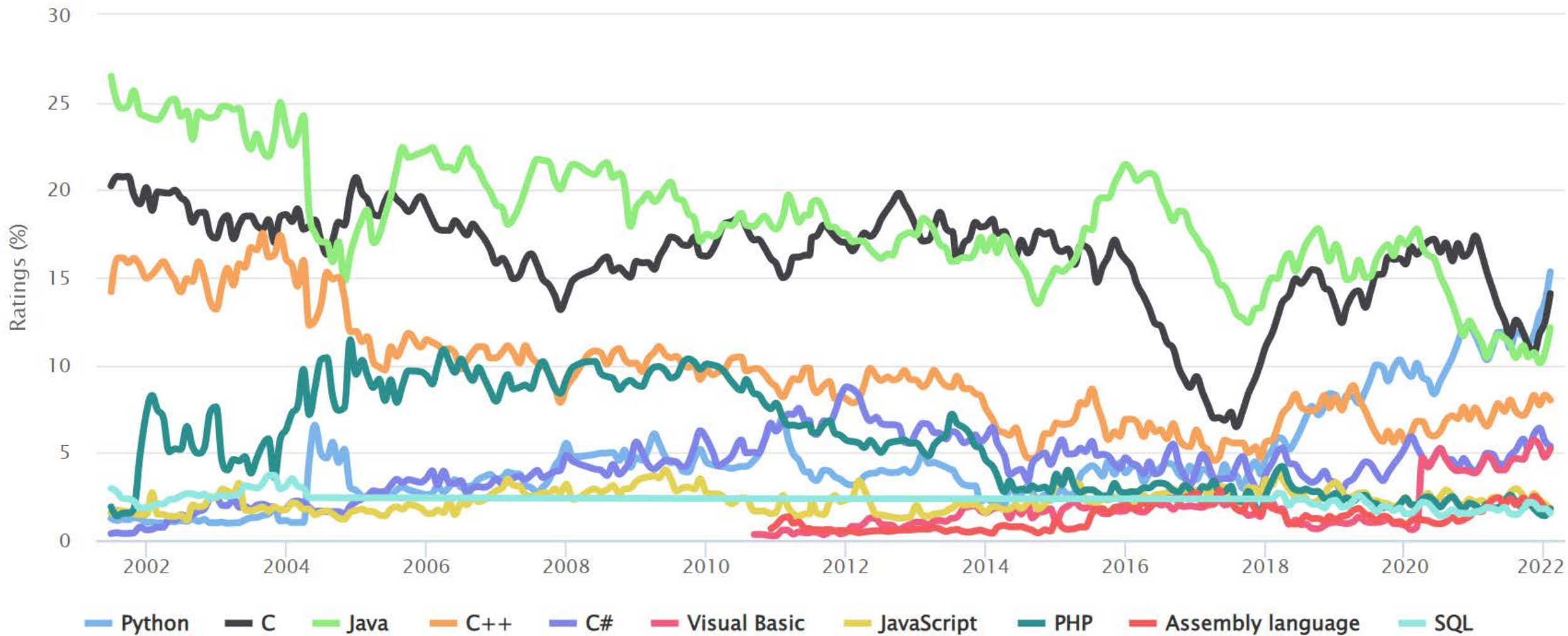
2. Python与其它语言

3. 编写Python程序

2022年2月榜单







TIOBE Programming Community Index

Source: www.tiobe.com



TIOBE Index for January 2022

January Headline: Python Programming Language of the Year 2021

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%

2021年年度编程语言

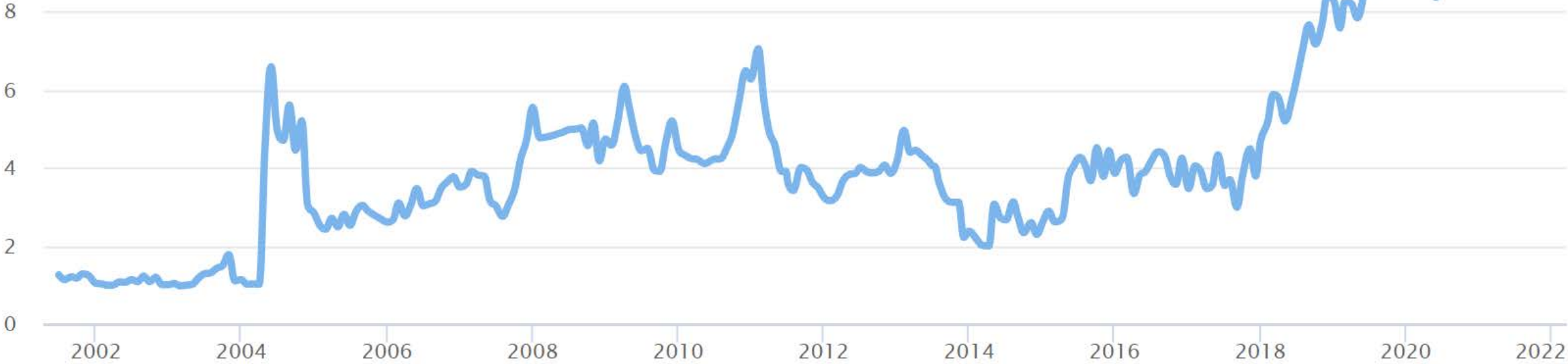
TIOBE Index for Python

占有率13.58%，排名第1

Source: www.tiobe.com

Year	Winner
2021	🏆 Python
2020	🏆 Python
2019	🏆 C
2018	🏆 Python
2017	🏆 C

Ratings (%)



For the first time in more than 20 years we have a **new leader** of the pack: the Python programming language. The long-standing hegemony of Java and C is over. Python, which started as a **simple** scripting language, as an alternative to Perl, has become mature. Its **ease of learning**, its **huge amount of libraries**, and its **widespread use** in all kinds of domains, has made it the **most popular programming language** of today. Congratulations Guido van Rossum! Proficiat!

-- Paul Jansen CEO TIOBE Software

Python与其它编程语言



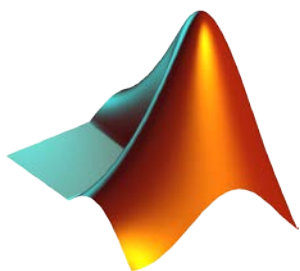
Python



C / C++



Fortran



MATLAB

占有率	15.33%	14.08% / 8.01%	0.58%	1.03%
排行	1	2 / 4	23	14
是否免费	免费	免费	免费	收费
难易程度	★	★★★★★	★★★	★
平台大小	★	★	★	★★★★★
编写效率	★★★★★	★	★★★	★★★★★

Python优势——编写效率

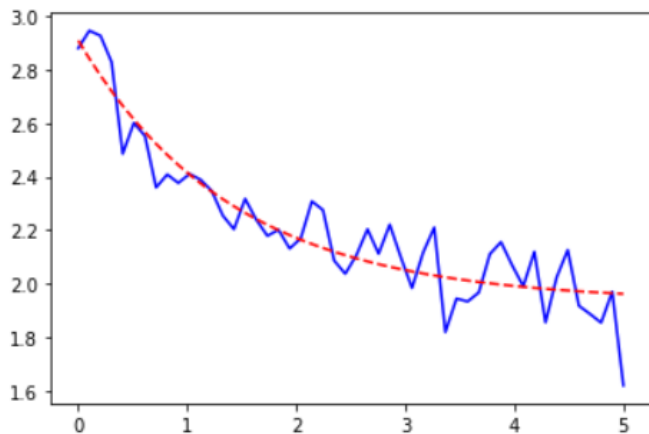
Python

```
In [5]: from scipy.optimize import curve_fit
import matplotlib.pyplot as plt
import numpy as np

def func(x, a, b, c):
    return a * np.exp(-b * x) + c

x_val = np.linspace(0, 5, 50)
y_val = func(x_val, 1, 1, 2) + 0.1 * np.random.normal(size = len(x_val))
plt.plot(x_val, y_val, 'b-')
pfit, pcov = curve_fit(func, x_val, y_val)
y_fit = func(x_val, pfit[0], pfit[1], pfit[2]) for i in x_val
plt.plot(x_val, y_fit, 'r--')
```

Out[5]: [



C++

```
void Iz::fit_GN(){
    cout<<setiosflags(ios::fixed)<<setprecision(4);

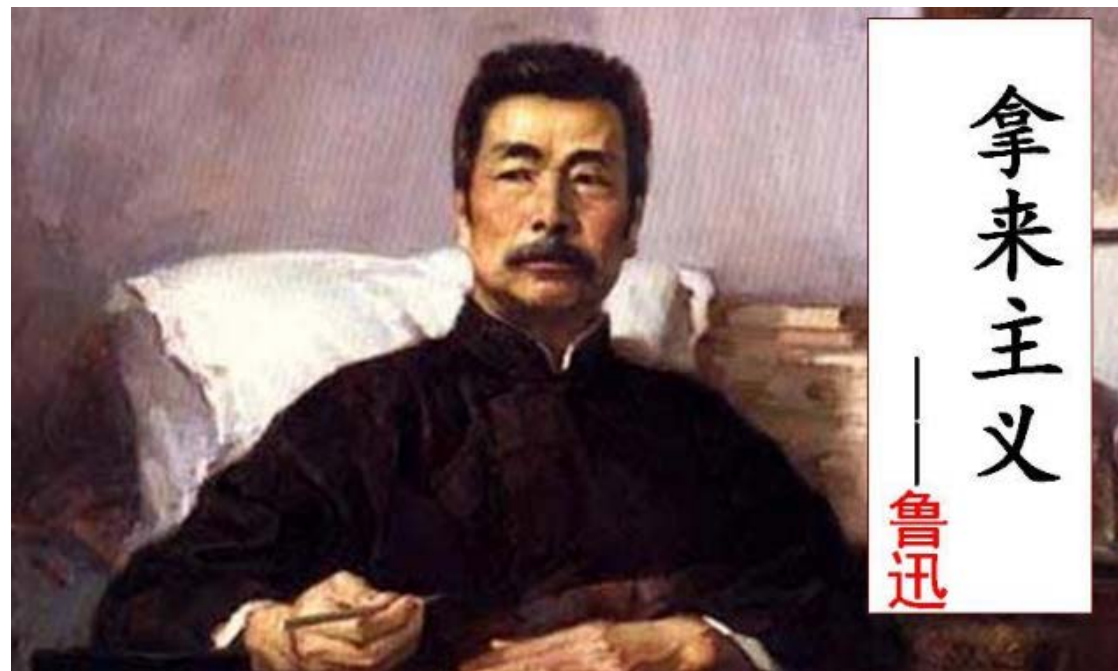
    para = para_initial(xvals, yvals);
    Vector2d para_d(0.0, 0.0);
    VectorXd r(xvals.size());
    double cod = cal_cod(xvals, yvals, para(0), para(1));
    cout<<"LOOP:"<<endl;
    cout<<"n\tA\tt\tcod"<<endl;
    cout<<"0\t"<<para(0)<<"\t"<<para(1)<<"\t"<<cod<<endl;
    for(int i = 0; i < 50; i++){
        for(int j = 0; j < xvals.size(); j++){
            r(j) = f_err(para(0), para(1), xvals[j], yvals[j]);
        }
        MatrixXd jac = cal_jac(para(0), para(1), xvals);
        para_d = - (jac.transpose() * jac).inverse() * (jac.transpose() * r);
        para = para + para_d;
        cod = cal_cod(xvals, yvals, para(0), para(1));
        cout<<i + 1<<"\t"<<para(0)<<"\t"<<para(1)<<"\t"<<cod<<endl;
        if(cod < 0){
            cout<<"Bad cod value..."<<endl;
            break;
        }
        if(abs(para_d(0)) < 0.0001 && abs(para_d(1)) < 0.0001){
            cout<<"Accuracy achieved, leave the loop..."<<endl;
            break;
        }
        if(i == 49){
            cout<<"Unable to converge..."<<endl;
            break;
        }
    }
    if(cod > cod_opt){
        para_opt(0) = para(0);
        para_opt(1) = para(1);
        step_opt = step;
        cod_opt = cod;
    }

    cout<<resetiosflags(ios::fixed);
}
```

Python优势——库

Python开发效率高的重要原因：有非常强大的第三方库

```
C:\Windows\system32>pip3 list
Package                               Version
-----
alabaster                             0.7.12
anaconda-client                       1.7.2
anaconda-navigator                   2.0.3
anaconda-project                     0.9.1
anyio                                 2.2.0
appdirs                              1.4.4
argh                                  0.26.2
argon2-cffi                          20.1.0
asn1crypto                           1.4.0
astroid                              2.5
astropy                              4.2.1
async-generator                       1.10
atomicwrites                         1.4.0
attrs                                20.3.0
autopep8                             1.5.6
Babel                                2.9.0
backcall                             0.2.0
backports.functools-lru-cache        1.6.4
backports.shutil-get-terminal-size  1.0.0
backports.tempfile                   1.0
backports.weakref                    1.0.post1
bcrypt                               3.2.0
beautifulsoup4                       4.9.3
bitarray                             1.9.2
```



1. Python介绍

2. Python与其它语言

3. 编写Python程序

开始使用Python



集成了Python编译器



Python编写、学习平台

环境变量：在操作系统中用来指定操作系统运行环境的一些参数



命令模式 (蓝色)

- Enter: 切换到编辑模式
- A: 在代码块前插入空白代码块
- B: 在代码块后插入空白代码块
- X: 剪切当前代码块
- C: 复制当前代码块
- V: 粘贴当前代码块
- DD: 删除代码块
- Z: 取消删除代码块

编辑模式 (绿色)

- Ctrl + Enter: 运行当前代码块
- Shift + Enter: 运行当前代码块并选定下一代码块
- Alt + Enter: 运行当前代码块并在后面插入新代码块

第一个Python程序

```
In [1]: print('Hello World!')
```

Hello World!

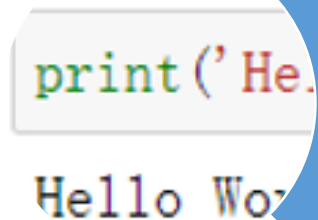
Python自带的函数，用于输出到界面

表示中间内容为字符串

Python中'...'与"..."一致

print('Hello World!')

输出的内容



```
print('Hello World')
```

Hello World

Jupyter

适合课程学习，交流



```
python
Microsoft Windows [版本 10.0.17134.1]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Windows\system32\python
Python 3.8.8 (default, Apr 13 2021)
Warning:
This Python interpreter is in a conda environment, but the
base environment has not been activated. Libraries may
not be available. Please see https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html
for more information.

> print('Hello World')
Hello World
```

命令提示符(cmd)

不推荐



```
1 print('Hello World')
```

集成开发环境(IDE)

适合用于开发，做项目

错误提示

程序编写过程中，报错是非常常见的，可根据报错提示对代码进行修改

```
In [2]: print(Hello World)
```

```
File "C:\Users\26093\AppData\Local\Temp\ipykernel_8484\4293340409.py", line 1  
    print(Hello World)  
          ^
```

```
SyntaxError: invalid syntax
```

错误位置

错误说明

Hello World → 'Hello World'

最常见错误：拼写错误

避免方法：注意关键词高亮提示

多个字符串输出

```
In [3]: print('Hello!', 'Machine', 'Learning!')
```

Hello! Machine Learning!

print('Hello!', 'Machine', 'Learning! ')

Hello! Machine Learning!

转义字符

输出: I'm fine.

```
In [4]: print('I'm fine.')
```

```
File "<ipython-input-4-470215fd291d>", line 1  
    print('I'm fine.')  
          ^
```

```
SyntaxError: invalid syntax
```

```
In [5]: print('I\'m fine.')
```

```
I'm fine.
```



常用转义字符

`\'` → `'`

`\''` → `''`

`\\` → `\`

`\t` → 横向制表符

`\b` → 退格

`\r` → 回车

`\n` → 换行