



Python与金融数据挖掘

文欣秀

wenxinxiu@ecust.edu.cn



学生信息



教学与考试安排

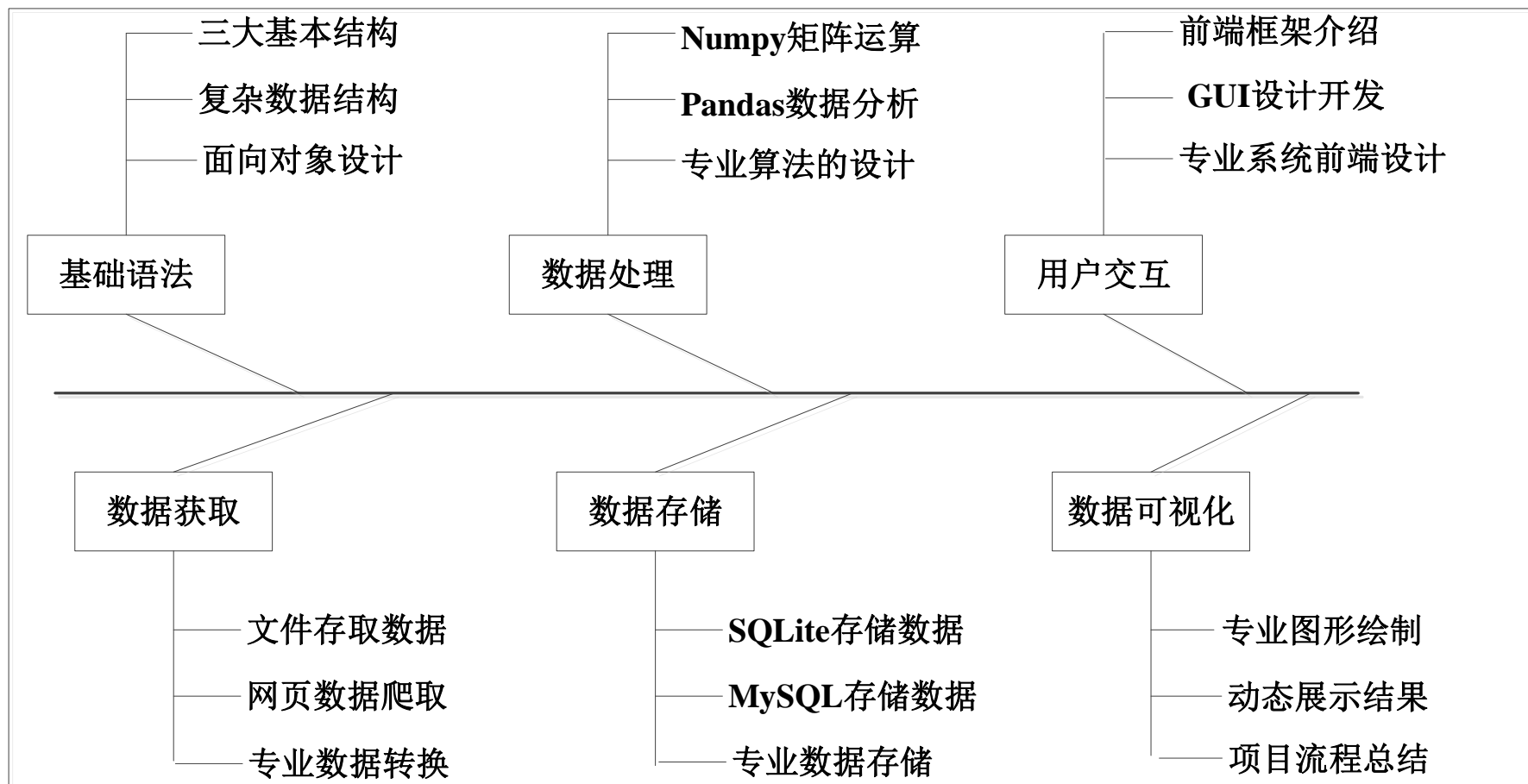
选用教材：Python程序设计基础(第二版) 李东方等

课时安排：48学时，1-18双周周四3、4节信息楼211

成绩评定：平时成绩(30%)+期末成绩(70%)

平时成绩：课后作业、线上讨论、课堂测试、三类实验

课程知识模块








问卷调查

问 题

以下哪种语言是目前最流行的编程语言？



世界编程语言排行榜

Feb 2023	Feb 2022	Change	Programming Language	
1	1			Python
2	2			C
3	4	▲		C++
4	3	▼		Java
5	5			C#

高级程序设计语言

编译程序： 在执行程序前，将程序源代码编译链接生成可执行程序，可脱离环境执行，**效率较高**。一旦修改，必须重新编译链接。



解释程序： 一边由解释器翻译，一边执行，执行**效率较低**，但是**修改方便**，可以随时修改随时运行。



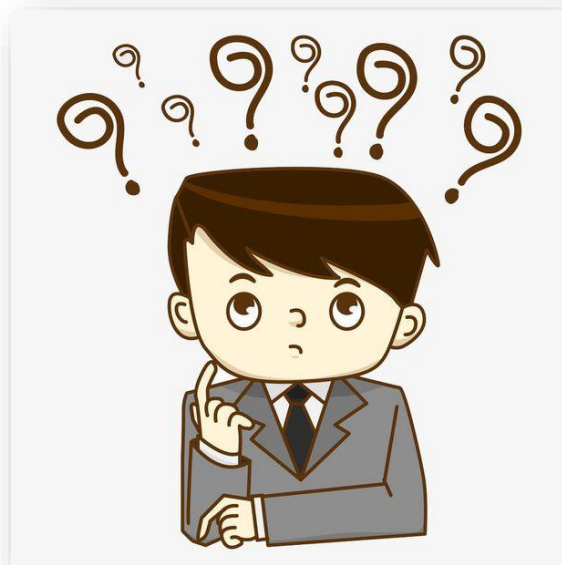
Python定义

Python是一种**面向对象**、**解释式**的计算机程序设计语言。它的语法简洁而清晰(C代码量的10%)，有丰富的**标准库**和强大的**第三方库**。



问题

Python的标准库有哪些？请尝试举例。



Python库简介

Python有强大的标准库和丰富的第三方库

标 准 库: <https://docs.python.org/3.8/library/>

如: math、os、time、calendar等

Python标准库

```
>>> import math
```

```
>>> dir(math)
```

```
['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh',  
'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'comb', 'copysign', 'cos', 'cosh',  
'degrees', 'dist', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod',  
'frexp', 'fsum', 'gamma', 'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan',  
'isqrt', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'perm', 'pi',  
'pow', 'prod', 'radians', 'remainder', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'trunc']
```

```
>>> help(math.sqrt)
```

Python标准库

```
>>> import math
```

```
>>> math.pow(3,2)
```

```
>>> math.sqrt(9)
```

```
>>> math.factorial(5)
```

```
>>> math.pi
```

```
>>> math.sin(math.pi/2)
```

Python标准库

```
>>> import calendar
```

```
>>> dir(calendar)
```

```
['Calendar', 'EPOCH', 'FRIDAY', 'February', 'HTMLCalendar', 'IllegalMonthError',  
'IllegalWeekdayError', 'January', 'LocaleHTMLCalendar', 'LocaleTextCalendar',  
'MONDAY', 'SATURDAY', 'SUNDAY', 'THURSDAY', 'TUESDAY', 'TextCalendar',  
'WEDNESDAY', '_EPOCH_ORD', '__all__', '__builtins__', '__cached__', '__doc__',  
'__file__', '__loader__', '__name__', '__package__', '__spec__', '_colwidth', '_locale',  
'_localized_day', '_localized_month', '_monthlen', '_nextmonth', '_prevmonth',  
'_spacing', 'c', 'calendar', 'datetime', 'day_abbr', 'day_name', 'different_locale', 'error',  
'firstweekday', 'format', 'formatstring', 'isleap', 'leapdays', 'main', 'mdays', 'month',  
'month_abbr', 'month_name', 'monthcalendar', 'monthrange', 'prcal', 'prmonth',  
'prweek', 'repeat', 'setfirstweekday', 'sys', 'timegm', 'week', 'weekday', 'weekheader']
```

```
>>> help(calendar.isleap)
```

Python标准库

```
>>> import calendar
```

```
>>> calendar.isleap(2023)
```

```
>>> import calendar as ca
```

```
>>> ca.isleap(2023)
```

```
>>> from calendar import *
```

```
>>> isleap(2023)
```


Python标准库

```
import calendar as cl
year = int(input("请输入年份"))
data= cl.calendar(year)
print(data)
```

请输入年份2023

2023						
January						
Mo	Tu	We	Th	Fr	Sa	Su
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					
February						
Mo	Tu	We	Th	Fr	Sa	Su
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28					
March						
Mo	Tu	We	Th	Fr	Sa	Su
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		
April						
Mo	Tu	We	Th	Fr	Sa	Su
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
May						
Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				
June						
Mo	Tu	We	Th	Fr	Sa	Su
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

Python标准库

```
import calendar as cl
year = int(input("请输入年份"))
month = int(input("请输入月份"))
data=cl.month(year,month)
print(data)
```

```
请输入年份2023
请输入月份2
      February 2023
Mo Tu We Th Fr Sa Su
      1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28
```

Python标准库

```
>>> import webbrowser as wb
```

```
>>> dir(wb)
```

```
['BackgroundBrowser', 'BaseBrowser', 'Chrome', 'Chromium', 'Elinks', 'Error',  
'Galeon', 'GenericBrowser', 'Grail', 'Konqueror', 'Mozilla', 'Netscape', 'Opera',  
'UnixBrowser', 'WindowsDefault', '__all__', '__builtins__', '__cached__', '__doc__',  
'__file__', '__loader__', '__name__', '__package__', '__spec__', '_browsers', '_lock',  
'_os_preferred_browser', '_synthesize', '_tryorder', 'get', 'main', 'open', 'open_new',  
'open_new_tab', 'os', 'register', 'register_X_browsers', 'register_standard_browsers',  
'shlex', 'shutil', 'subprocess', 'sys', 'threading']
```

```
>>> help(wb.open)
```

Python标准库

```
>>> import time
```

```
>>> dir(time)
```

```
['_STRUCT_TM_ITEMS', '__doc__', '__loader__', '__name__', '__package__',  
 '__spec__', 'altzone', 'asctime', 'ctime', 'daylight', 'get_clock_info', 'gmtime', 'localtime',  
 'mktime', 'monotonic', 'monotonic_ns', 'perf_counter', 'perf_counter_ns',  
 'process_time', 'process_time_ns', 'sleep', 'strptime', 'strftime', 'struct_time',  
 'thread_time', 'thread_time_ns', 'time', 'time_ns', 'timezone', 'tzname']
```

```
>>> help(time.sleep)
```

Python标准库

```
import time
import webbrowser as wb
time.sleep(2)
wb.open("http://jwc.ecust.edu.cn")
```



Python标准库

```
>>> import os
```

```
>>> dir(os)
```

```
[...'chdir', 'chmod', 'close', 'closerange', 'cpu_count', 'curdir', 'defpath',  
'device_encoding', 'devnull', 'dup', 'dup2', 'environ', 'error', 'execl', 'execle', 'execlp',  
'execlpe', 'execv', 'execve', 'execvp', 'execvpe', 'extsep', 'fdopen', 'fsdecode', 'fsencode',  
'fspath', 'fstat', 'fsync', 'ftruncate', 'get_exec_path', 'get_handle_inheritable',  
'get_inheritable', 'get_terminal_size', 'getcwd', 'getcwdb', 'getenv', 'getlogin', 'getpid',  
'getppid', 'isatty', 'kill', 'linesep', 'link', 'listdir', 'lseek', 'lstat', 'makedirs', 'mkdir', 'name',  
'open', 'pardir', 'path', 'pathsep', 'pipe', 'popen', 'putenv', 'read', 'readlink', 'remove',  
'removedirs', 'rename', 'renames', 'replace', 'rmdir', 'scandir', 'sep',  
'set_handle_inheritable', 'set_inheritable', 'spawnl', 'spawnle', 'spawnv', 'spawnve', 'st',  
'startfile', 'stat', 'write' ...]
```

```
>>> help(os.popen)
```

Python标准库

```
import os  
seconds=input("您希望多少秒之后关机: ")  
os.popen("shutdown -s -t %s"%seconds)
```

您希望多少秒之后关机: 20

Python库简介

Python有强大的标准库和丰富的第三方库

标准库: <https://docs.python.org/3.8/library/>

如: math、os、time、calendar等

第三方库: <https://pypi.org/>

如: numpy、pandas等, 需要安装

Python应用领域

科学计算：Numpy、SciPy...

WEB开发：Django、Flask...

数据分析：Pandas、Matplotlib...

人工智能：Scikit-Learn、Keras...

游戏开发：Pyglet、Pygame ...

网络爬虫：Requests、Scrapy ...

...



爬图片案例



0.jpg



1.jpg



2.jpg



3.jpg



9.jpg



10.jpg



11.jpg



12.jpg



18.jpg



19.jpg



20.jpg



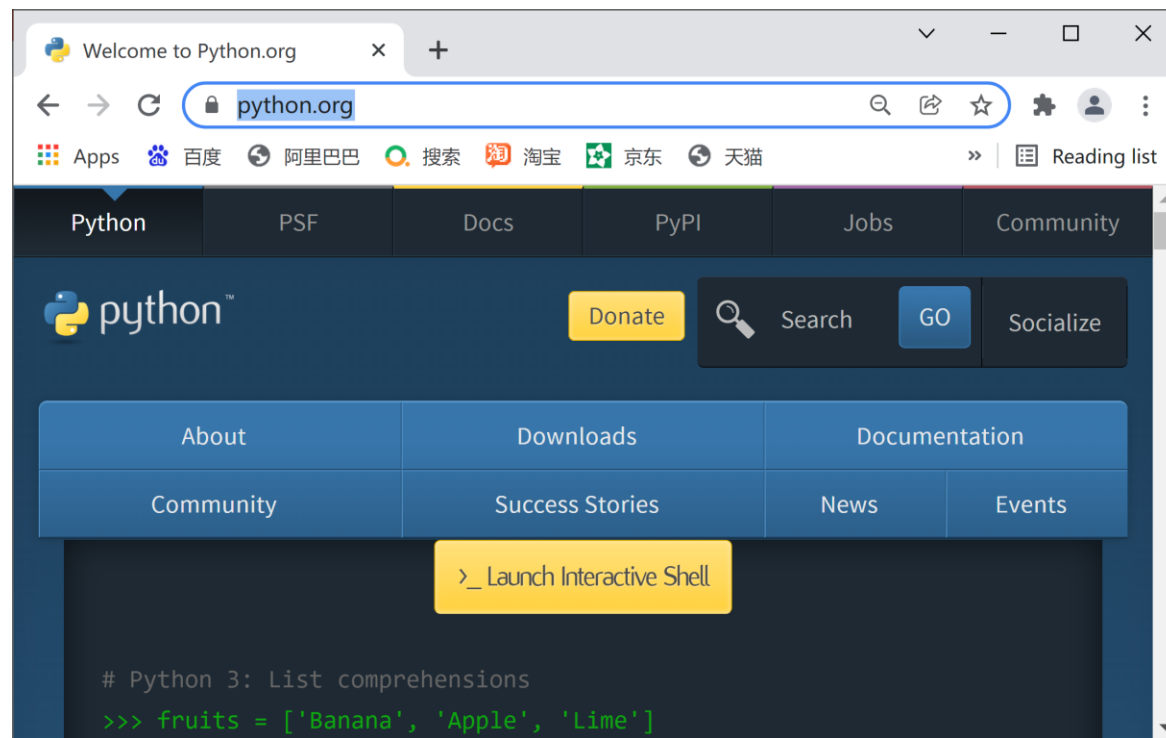
21.jpg

新闻舆情评分系统案例

1. 大公司 | 阿里巴巴投资生物育种,先正达、敦煌种业、中垦玉种业...
阿里巴巴该条新闻舆情评分为-5
<https://www.163.com/dy/article/HU78CE6A0512CUPE.html>
2. 清华大学深入研究乡村振兴,《阿里巴巴乡村振兴模式研究报告》在京...
阿里巴巴该条新闻舆情评分为-5
<https://3g.163.com/dy/article/HU72SUJI0514BL38.html>
3. 通用型生物育种平台博瑞迪获A2轮超亿元融资,阿里巴巴领投
阿里巴巴该条新闻舆情评分为-5
<https://www.163.com/dy/article/HU715BIR051480G7.html>
4. ATOX港股:阿里巴巴业绩周四出炉,股价能否再获惊喜?
阿里巴巴该条新闻舆情评分为-10
<https://finance.ifeng.com/c/8NcI1Izx00h>
5. 传阿里巴巴数千万美元投资以色列风投公司JVP
阿里巴巴该条新闻舆情评分为-15
<https://www.jiemian.com/article/251276.html>
6. ... 中小企业 | 外贸 | 市场规模 | 新能源 | 汽车配件 | 阿里国际站 | 阿里巴巴...
阿里巴巴该条新闻舆情评分为-5
<https://www.163.com/dy/article/HU6KH1DT0511A6N9.html>
7. 阿里巴巴天猫目标在未来三年实现线上交易规模翻一番
阿里巴巴该条新闻舆情评分为-5
<https://www.jiemian.com/article/3075677.html>
8. ... 专利 | 云计算 | 吴翰清 | 微软 | 科技巨头 | 阿里云 | 阿里巴巴集团_手机...
阿里巴巴该条新闻舆情评分为-5

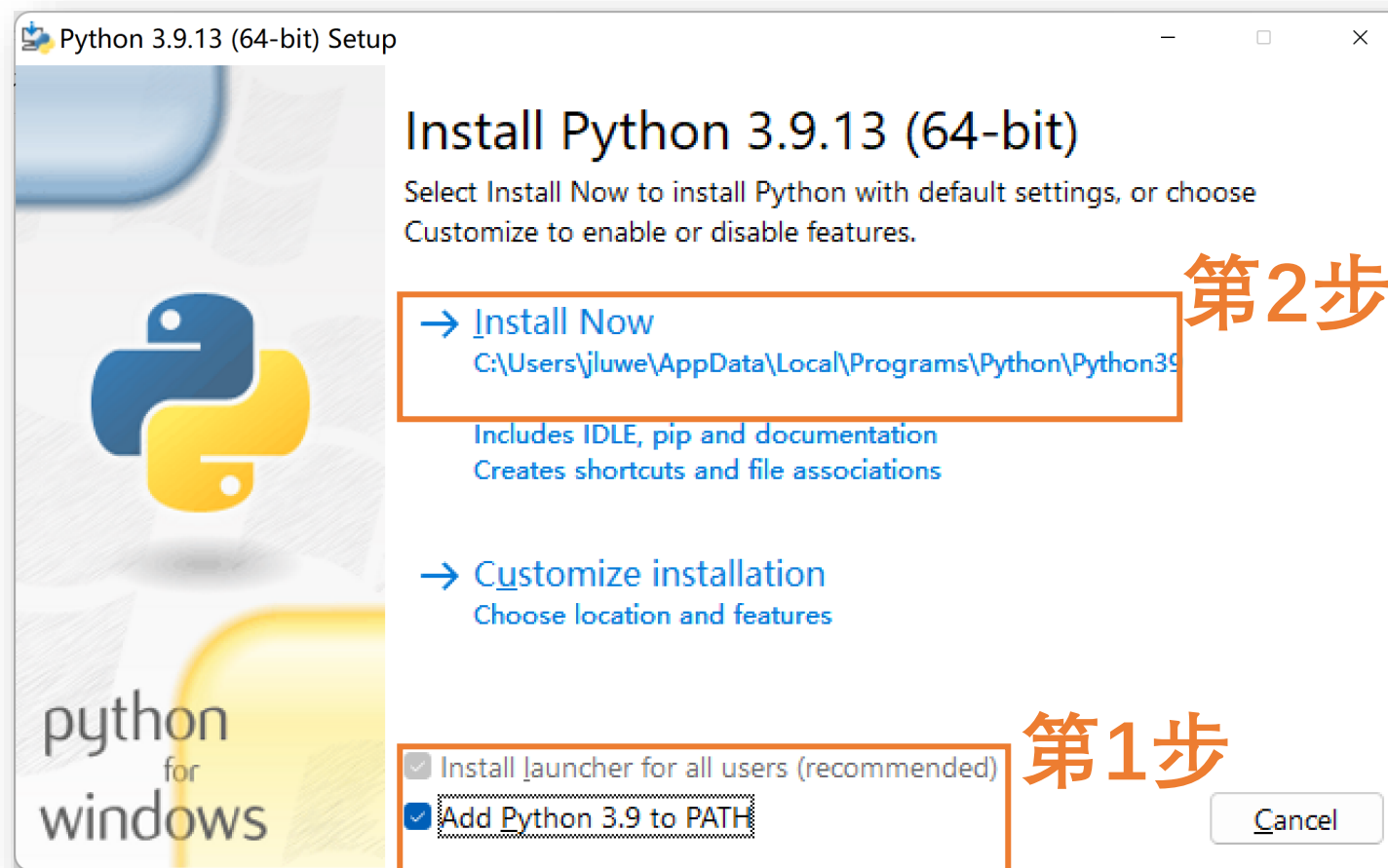
软件安装

➤ IDLE: <https://www.python.org/>



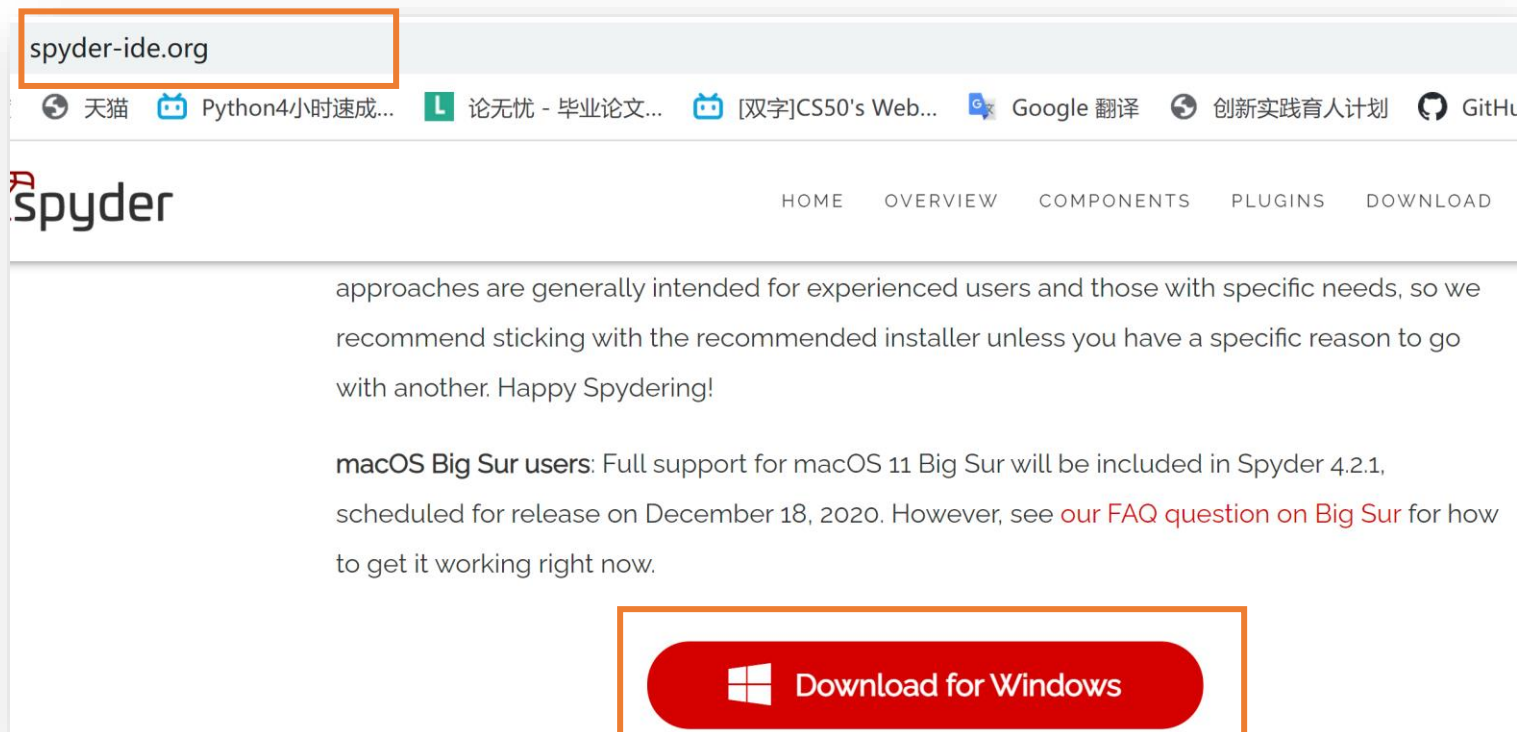
软件安装

➤ IDLE: <https://www.python.org/>



软件安装

➤ Spyder: <https://www.spyder-ide.org/>



第三方库安装

```
C:\Users\jluwe>pip install matplotlib
```

```
Collecting matplotlib
```

```
  Downloading matplotlib-3.5.1-cp39-cp39-win_amd64.whl (7.2 MB)
```

```
| 7.2 MB 156 kB/s
```

```
WARNING: You are using pip version 21.2.4; however, version 22.0.0 is available.
```

```
You should consider upgrading via the 'C:\Users\jluwe\AppData\Local\Programs\Python\Python39\python.exe -m pip install --upgrade pip' command.
```

```
C:\Users\jluwe>python.exe -m pip install --upgrade pip
```

```
C:\Users\jluwe>pip install matplotlib
```

```
Collecting matplotlib
```

```
  Downloading matplotlib-3.5.1-cp39-cp39-win_amd64.whl (7.2 MB)
```

```
| 7.2 MB 156 kB/s
```


程序设计方法



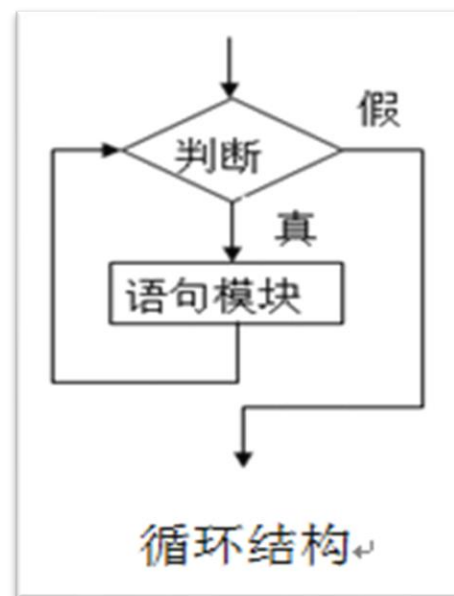
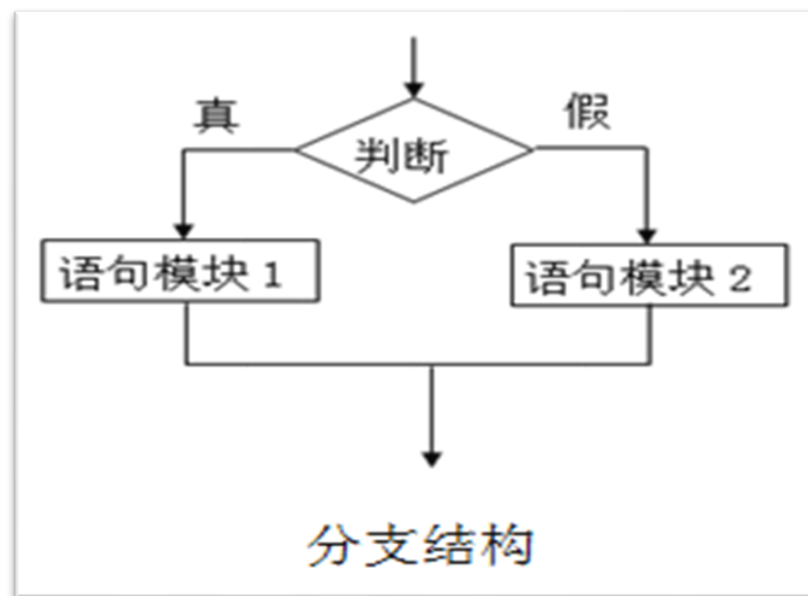
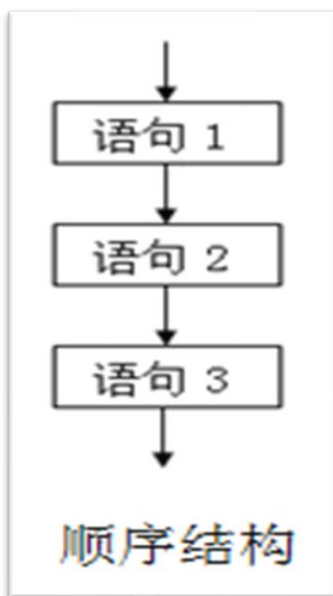
图灵奖获得者沃思

算法+数据结构=程序

Algorithm+Data Structures=Programs

算法：对操作的描述，即要求计算机进行操作的步骤。

三种基本结构



程序设计方法



图灵奖获得者沃思

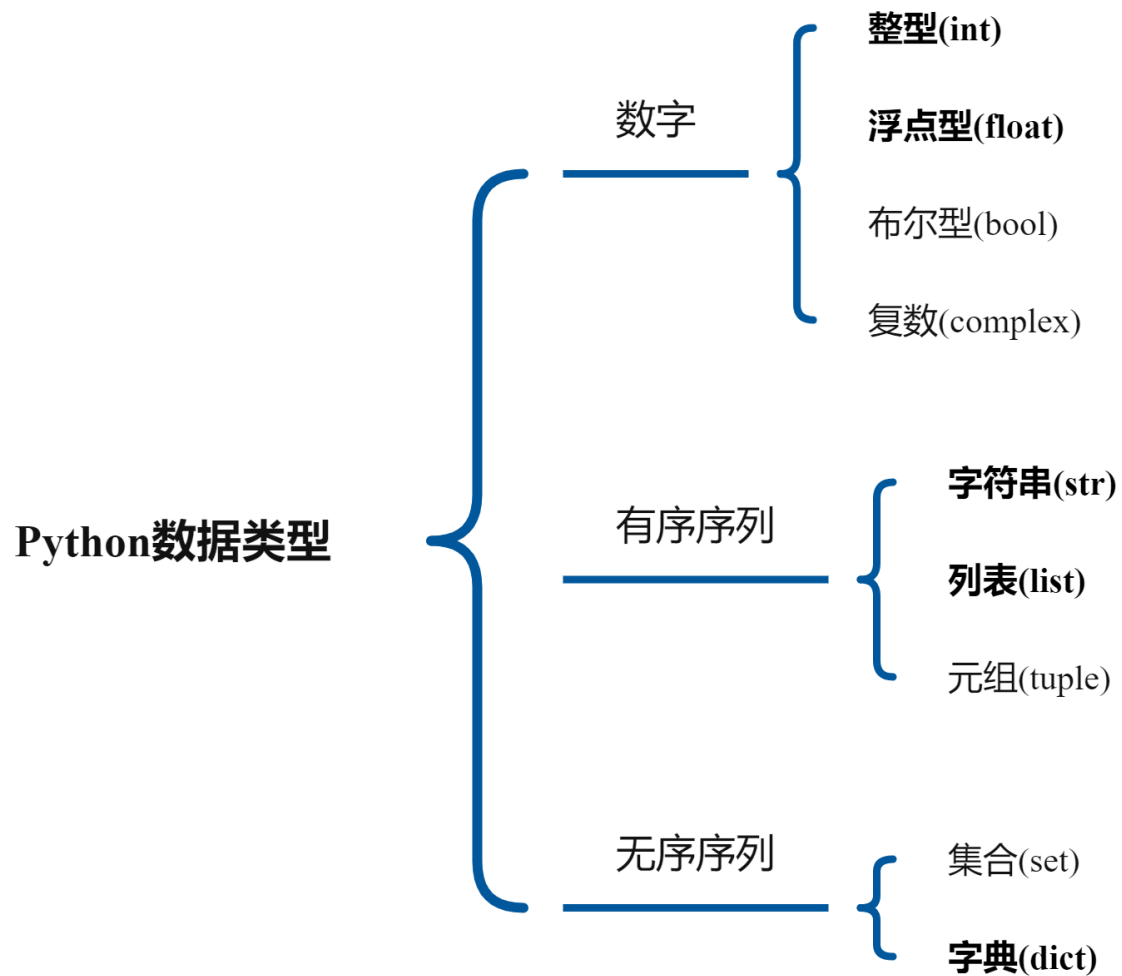
算法+数据结构=程序

Algorithm+Data Structures=Programs

算法：对操作的描述，即要求计算机进行操作的步骤。

数据结构：对数据的描述，程序中用到数据的类型及数据的组织形式。

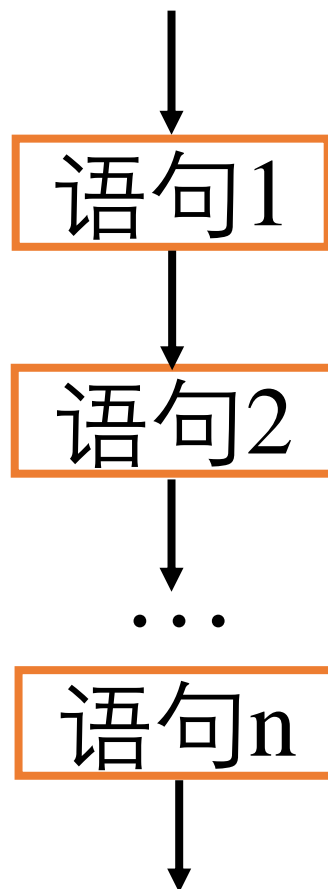
Python数据类型



银行贷款本息问题

编写程序，输入贷款金额money，贷期year和贷款年利息rate，计算贷款到期时的本息合计total并输出。到期还款本息的计算公式为 $\text{total} = \text{money} * (1 + \text{rate})^{\text{year}}$ 。

顺序结构



变量命名规则

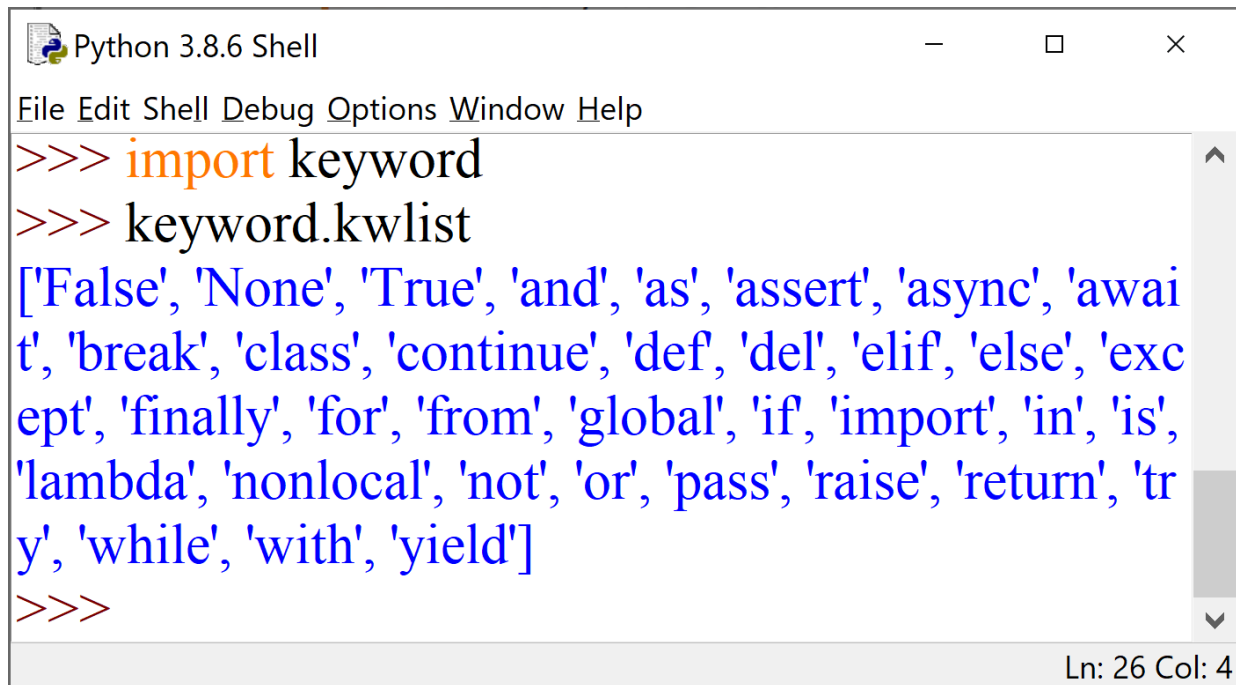
变量：指在运行过程中值可以被修改的量

- 由**字母**或**下划线**开始
- 其它字符可以是**数字**、**字母**或**下划线**
- 区分**大小**写
- 尽量**见名知义**
- 不能使用**关键字**

Python关键字

>>> **import keyword**

>>> **keyword.kwlist**

A screenshot of a Python 3.8.6 Shell window. The window has a title bar with the text 'Python 3.8.6 Shell' and standard window controls. Below the title bar is a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area shows the following code:

```
>>> import keyword
>>> keyword.kwlist
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
>>>
```

The output is displayed in blue text. At the bottom right of the window, the status bar shows 'Ln: 26 Col: 4'.

问题

下面哪个选项属于Python合法的变量名?

(A) **if3**

(B) 3if

(C) if

(D) #if

简单输入示例一

```
>>> myName = input("请输入姓名: ")
```

input(): 功能是读取用户输入的字符串

简单输入示例二

```
>>> myAge = int(input("请输入年龄: "))
```

int() : 将字符串转换为整数

```
>>> myScore = float(input("请输入成绩: "))
```

float() : 将字符串转换为单精度数

标准算术运算符

运 算 符	描 述
+	加法
-	减法
*	乘法
/	浮点除法
//	整除运算，返回商
%	整除运算，返回余数，也叫取模
**	幂运算

简单输出示例

```
>>> myString = "我的名字是张三"
```

```
>>> print(myString)
```

print(): 输出程序运行结果

银行贷款本息问题答案一

```
import math
money=float(input("请输入贷款金额: "))
rate=float(input("请输入贷款年利息: "))
year=int(input("请输入贷款年数: "))
total=money*pow((1+rate),year)
print("贷款总金额: ",total)
```

```
请输入贷款金额: 10000
请输入贷款年利息: 0.05
请输入贷款年数: 10
贷款总金额: 16288.94626777442
```

银行贷款本息问题答案二

```
money=float(input("请输入贷款金额: "))  
rate=float(input("请输入贷款年利息: "))  
year=int(input("请输入贷款年数: "))  
total=money*((1+rate)**year)  
print("贷款总金额: ",total)
```

```
请输入贷款金额: 10000  
请输入贷款年利息: 0.05  
请输入贷款年数: 10  
贷款总金额: 16288.94626777442
```

思考题

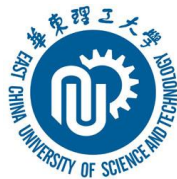
顺序结构可以解决专业或生活中的哪些问题？
请举例说明：输入、算法、输出。



欢迎二维码

```
import qrcode  
text="文欣秀热烈欢迎你\n我们即将开始Python之旅\n你准备好了吗？"  
img =qrcode.make(text)  
img.save('welcome.jpg')
```





谢 谢