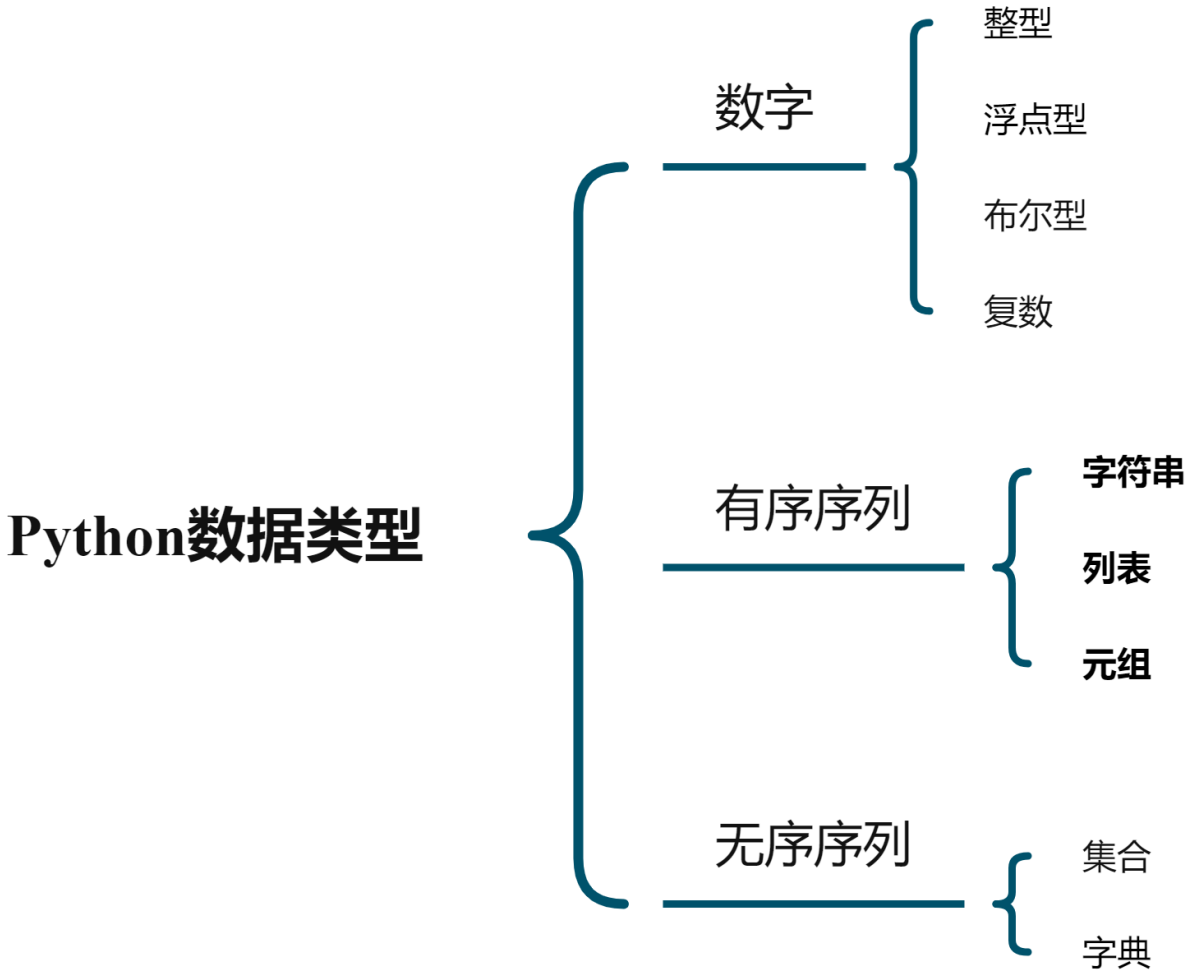
Python与金融数据挖掘(3)

文欣秀

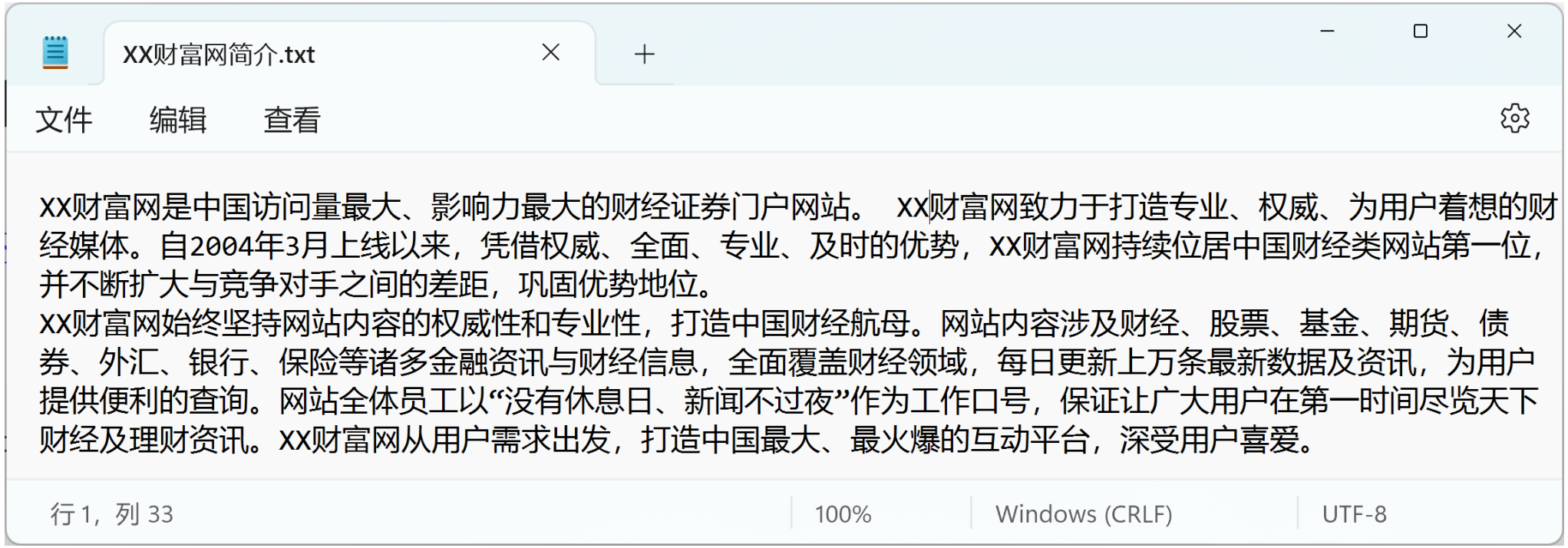
wenxinxiu@ecust.edu.cn

数据类型



拓展问题

编写程序，实现文件“xx财富网简介.txt”中的XX全部替换。



常用字符串方法

s. upper()： 将字符串都转换成大写字母

s. lower()： 将字符串都转换成小写字母

s. split()： 实现字符串的分割

s. replace()： 用第二个子串替代第一个字串

s. strip()： 消除字符串两端的空格及符号

文章单词替换问题

```
fobj=open("XX财富网简介.txt","r",encoding="utf-8")
paper=fobj.read()
fobj.close()
before=input("被替代字：")
after=input("替代为字：")
result=paper.replace(before,after)
print(result)
```

文章单词替换问题

```
with open("XX财富网简介.txt","r",encoding="utf-8") as fobj:
    paper=fobj.read()
    before=input("被替代字：")
    after=input("替代为字：")
    result=paper.replace(before,after)
    print(result)
```

如何将替换后的结果存放到新的文件中？

文件写操作

write()： 将一个**字符串**写入文件中

```
with open("XX财富网简介.txt","r",encoding="utf-8") as fobj:
    paper=fobj.read()
    before=input("被替代字：")
    after=input("替代为字：")
    result=paper.replace(before,after)
with open("result.txt", "w") as f2:
    f2.write(result)
```

营收额排序问题

编写程序，自动模拟产生某公司10个月的营收额（1000万-5000万之间的随机数），实现营收额的从小到大排序。

```
排序前的营收额：
3325 4361 3680 3763 2969 3112 2636 3034 4265 1221

排序后的营收额：
1221 2636 2969 3034 3112 3325 3680 3763 4265 4361
```

列表定义

- 能保存**任意数量任意类型**的Python 对象
- 列表元素用中括号 **[ ]**包裹，元素用**逗号**分隔
- 第一个元素索引为 **0**，最后一个元素索引为**-1**
- 切片运算符[i : j]得到从下标**i**到下标**j-1**的子集
- 元素的个数及元素的**值可以改变**

列表示例

```
>>> aList=[] 或 aList=list()        #定义一个空列表
>>> aList=[1, 2, 3, 4]
>>> aList[-1]                       #获取最后一个元素
>>> aList[2:]                       #获取一个子列表
>>> aList[1]=5                      #修改一个元素
```

列表函数

>>> **len(L)：**返回列表L的长度，即元素的数量

>>> max(L)：返回列表L中的最大元素

>>> sum(L)：返回列表L中所有元素(数字)总和

>>> **sorted(L)：**对任意列表L进行排序

列表方法

t. append(x)： 在列表的末尾添加元素，列表的长度增加1

t. sort(reverse=False)： 将列表中元素进行从小到大排序

t. remove(x)： 删除列表中的第一个值为x的元素

t. count(x)： 返回列表中x出现的次数，若不包含x，返回0

列表方法示例

```
>>> aList=[1, 20, 5, 8]
>>> aList. append(17)
>>> aList. remove(8)
>>> aList. count(5)
>>> aList. sort()
```

营收额排序问题答案

```
import random
myList=[]
for i in range(10):
    number=random. randint(1000,5000)
    myList. append(number)
print("\n\n排序前的营收额：  ")
for i in myList:
    print("{}".format(i),end=" ")
myList. sort()
print("\n\n排序后的营收额：  ")
for i in myList:
    print("{}".format(i), end=" ")
```

拓展问题

编写程序，从文件“price.csv中”读入某公司最近20天的收盘价，输出最高收盘价。

	A
1	181.77
2	128.24
3	136.74
4	167.51
5	127.3
6	131.21
7	140.96
8	114.53
9	154.99
10	120.32
11	141.58
12	102.65
13	175.81
14	125.56
15	198.64
16	145.85
17	127.72
18	134.08
19	178.97
20	167.18

文件读操作二

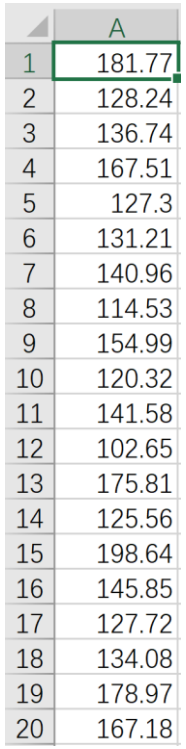
handle. readline()：从文件读一行数据到字符串

例：handle. readline()

handle. readlines()：读取整个文件并创建列表

例：handle. readlines()

读文件示例



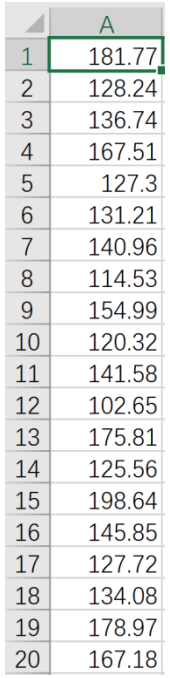
	A
1	181.77
2	128.24
3	136.74
4	167.51
5	127.3
6	131.21
7	140.96
8	114.53
9	154.99
10	120.32
11	141.58
12	102.65
13	175.81
14	125.56
15	198.64
16	145.85
17	127.72
18	134.08
19	178.97
20	167.18

```
income=[]
with open("price.csv","r") as fobj:
    money=fobj. readlines()
    for i in money:
        i=i.strip()
        i=float(i)
        income. append(i)
print("最高收盘价为：{:.2f}".format(max(income)))
```

CSV: 以逗号分隔的文本文件，xlsx文件可以另存为csv文件

文件读操作三

- 直接在文件对象上循环读取内容



	A
1	181.77
2	128.24
3	136.74
4	167.51
5	127.3
6	131.21
7	140.96
8	114.53
9	154.99
10	120.32
11	141.58
12	102.65
13	175.81
14	125.56
15	198.64
16	145.85
17	127.72
18	134.08
19	178.97
20	167.18

```
income=[]
with open("price.csv","r") as fobj:
    for i in fobj:
        i=i. strip()
        i=float(i)
        income. append(i)
print("最高收盘价为：  {:.2f}".format(max(income)))
```

股票代码和名称合并问题

已知部分股票代码和股票名称分别存在两个列表中，编写程序，将股票代码和股票名称合并在一起存到列表中。

代码	名称
000001	上证指数
399001	深证成指
899050	北证50
000300	沪深300
399005	中小100
399006	创业板指

```
代码        名称
('000001', '上证指数')
('399001', '深证成指')
('899050', '北证50')
('00300', '沪深300')
('399005', '中小100')
```

元组定义

- 能保存**任意数量任意类型**的Python 对象
- 元组元素用小括号 **( )**包裹
- 元素的个数及元素的值**不可以改变**
- 索引运算符[i]得到下标为i的元素
- 切片运算符[i : j]得到从下标i到下标j-1的子集

元组示例

```
>>> aTuple = ('robots', 77, 93, 'try')

>>> aTuple[0]

>>> aTuple[1:3]

>>> aTuple[::2]
```

课堂练习

表达式 (1, 2)+(3, 4) 的值为（ ）

A、(1, 2, 3, 4)

B、 (4, 6)

C、10

D、 (16,)

元组创建方法

```
>>> t1=()                          #创建一个空元组
>>> t2=(1,2,3)                     #创建包含三个元素的元组
>>> t3=(1,)                        #创建一个包含一个元素的元组
>>> t4=tuple([1,2,3])                    #将列表转换为元组
>>> t5=1,2,3                             #元组打包
>>> name, age=("Jack",28)          #将元组解包
```

元组应用范围

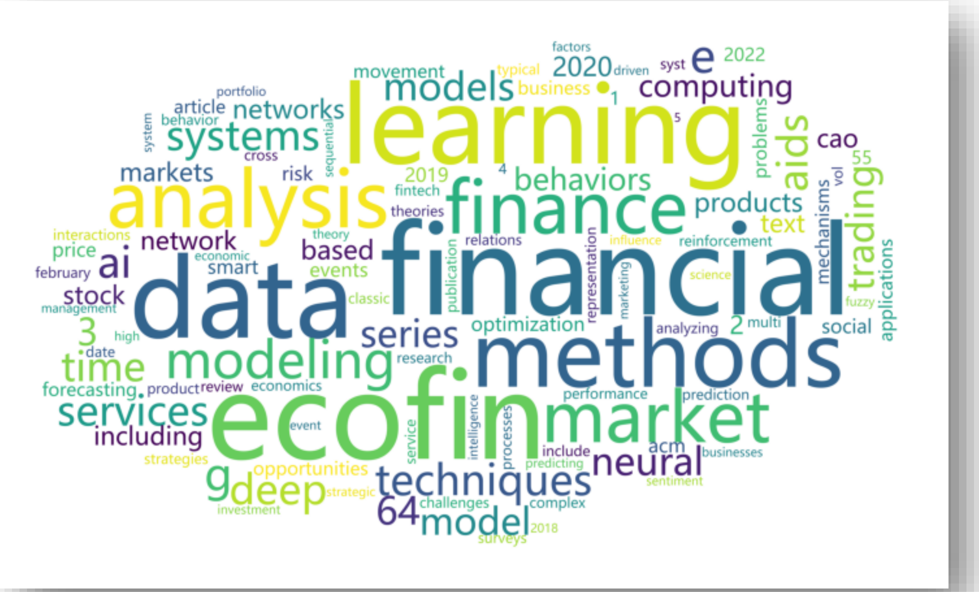
- 不能修改、增加或删除元组中的元素
- del 可删除整个元组，但不能删除元素
- 对元组的访问和处理速度要快于列表
- 可用于函数参数传递，避免参数被修改

股票代码和名称合并答案

```
merge=[]
code=["000001","399001","899050","00300","399005"]
name=["上证指数","深证成指","北证50","沪深300","中小100"]
for i in range(len(code)):
    merge. append((code[i],name[i]))
print("   代码       名称")
for e in merge:
    print(e)
```

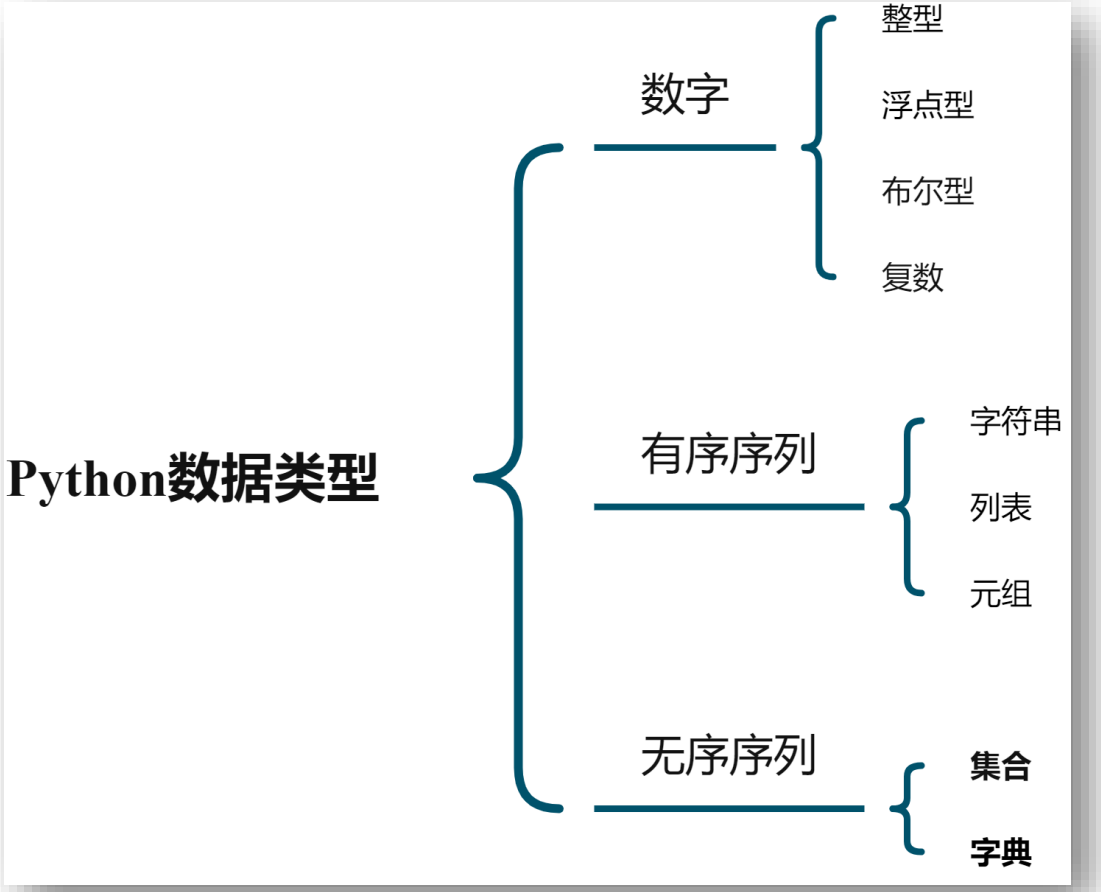
如何将合并后的结果存放到新的文件中？

如何制作词云？


learning
analysis
financial
data
methods
ecofin
market
finance
modeling
techniques
deep
neural
model
services
time
series
systems
models
computing
behaviors
products
trading
aids
networks
network
markets
stock
ai
including
2020
64
optimization
social
text
based
events
movement
cao


朱林涛
李雨杉
魏子勋
王铭成
林晓蝶
王大睿
姚润东
曾思凡
滕佳慈
邓雨婷
葛睿喆
裴思洋
王绘雯
孟哲安
庄楠
王鑫
王予嘉
刘佳恒
贺煜博
徐昊博
李枝阳
沈倪嘉
张为尧
杨世宇
张凯璇
冯金广
管宸昊
欧阳超前
罗健
吴林浩
李金彦
张予嘉
史海云
任辰宇
郑宜轩
胡文波
赵帅
李佳
顾亿
万兴海
张宇皓
武雪燕
袁洲力

数据类型



代码	名称
000001	上证指数
399001	深证成指
899050	北证50
000300	沪深300
399005	中小100
399006	创业板指

字典定义

- 是Python 中的映射数据类型，用**{ }**包裹
- 由**键-值**对构成，键值对使用冒号**:**分隔
- 键必须唯一，必须是不可变数据类型
- 一般以**数字、字符串、元组**等不可变对象作为键
- 值可以是**任意类型**的Python 对象

字典示例

```
>>> test = {} 或test=dict()                #创建一个空字典
>>> info= {'000001': '上证数', '399001' : '深证成指'}
>>> info['000300']= '沪深300'              #新增元素
>>> info['000001']= '上证指数'             #修改元素值
>>> del info['399001']                     #删除元素值
```

课堂练习

正确定义一个字典的是（ ）

A、 a=["A": 10, "B": 20, "C": 30]

B、 a=("A": 10, "B": 20, "C": 30)

C、 a={A:10, B: 20, C: 30}

D、 a={"A":10, "B":20, "C": 30}

常用字典方法

di.keys()： 返回包含字典所有**键**的列表

di.values()： 返回包含字典所有**值**的列表

di.items()： 返回包含所有(**键**、**值**)项列表

di.get(key,[default])： 返回健**key对应的值**，若key不存在，则返回default

di.update(a)： 将字典a中的键值对添加到di中

课堂练习

若dic1 = {'甲' :3, '乙' :1, '丙':5 , '丁' :8}，则执行print(dic1.get('乙', '未找到'))的结果是（ ）

A、未找到

B、1

C、报错

D、输出空值

词云相关库

matplotlib： 用于绘图的第三方库

wordcloud： 用于词云展示的第三方库

imageio： 读取和写入各种图像的第三方库

爱心词云

	A	B
1	学号	姓名
2	22011828	张紫涵
3	22011829	胡嘉欣
4	22011830	秦钰菲
5	22011831	吕盈萱
6	22011832	周宣彤
7	22011833	许之悦
8	22011834	张玥
9	22011835	刘蔼萱
10	22011837	赵一超

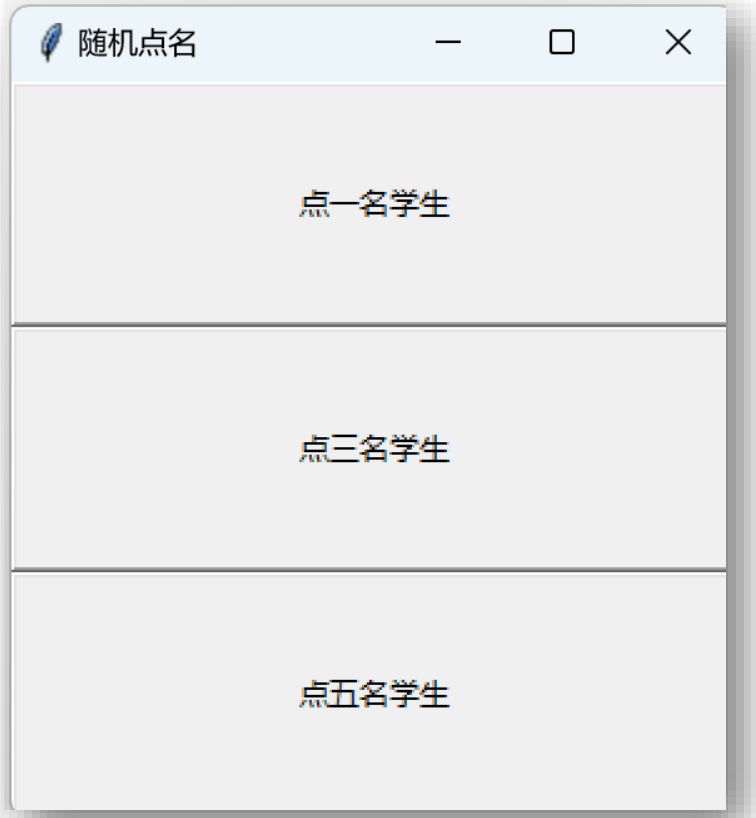

```
from random import *
counts={}#创建一个空字典
with open("student.csv", 'r') as fobj:
    for i in fobj:
        if i[:2]=="学号":
            continue
        i=i. strip()
        code, name=i. split(",")
        counts[name]=randint(30,100)
import matplotlib. pyplot as plt
from wordcloud import WordCloud
from imageio.v2 import imread
```

爱心词云

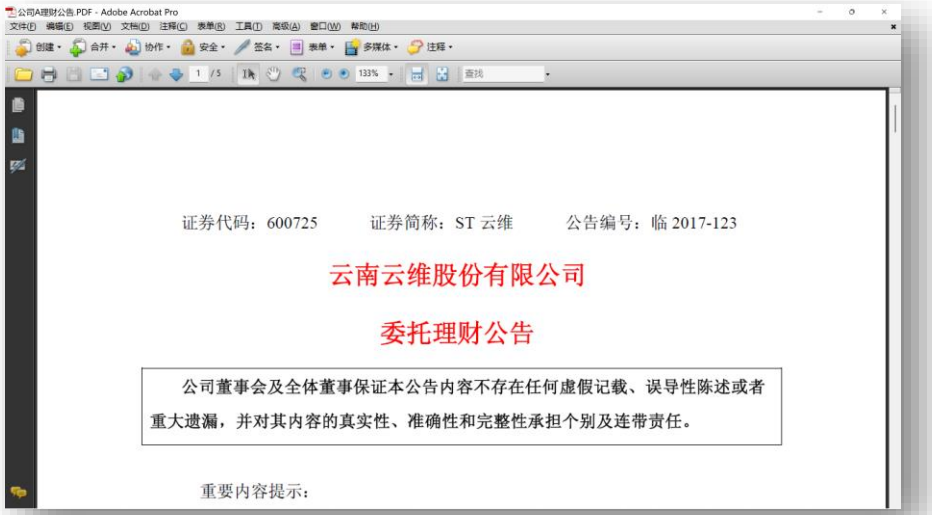
```
pic = imread('love.png')
wc=WordCloud(mask=pic,font_path='msyh.ttc',  #中文字体
             repeat=False, #内容是否可以重复
             background_color='white', #设置背景颜色
             max_words=100,          #设置最大词数
             max_font_size=120,     #设置字体最大值
             min_font_size=10,     #设置字体最小值
             random_state=50,      #设置有配色方案
             scale=1)                    #按照比例进行放大画布
wc.generate_from_frequencies(counts)
plt.imshow(wc)
plt.show()
```

案例分析

	A	B
1	学号	姓名
2	22011828	张紫涵
3	22011829	胡嘉欣
4	22011830	秦钰菲
5	22011831	吕盈萱
6	22011832	周宣彤
7	22011833	许之悦
8	22011834	张玥
9	22011835	刘蔼萱
10	22011837	赵一超



案例分析



证券代码：600725　　证券简称：ST 云维　　公告编号：临 2017-123

云南云维股份有限公司

委托理财公告

公司董事会及全体董事保证本公告内容不存在任何虚假记载、误导性陈述或者重大遗漏，并对其内容的真实性、准确性和完整性承担个别及连带责任。

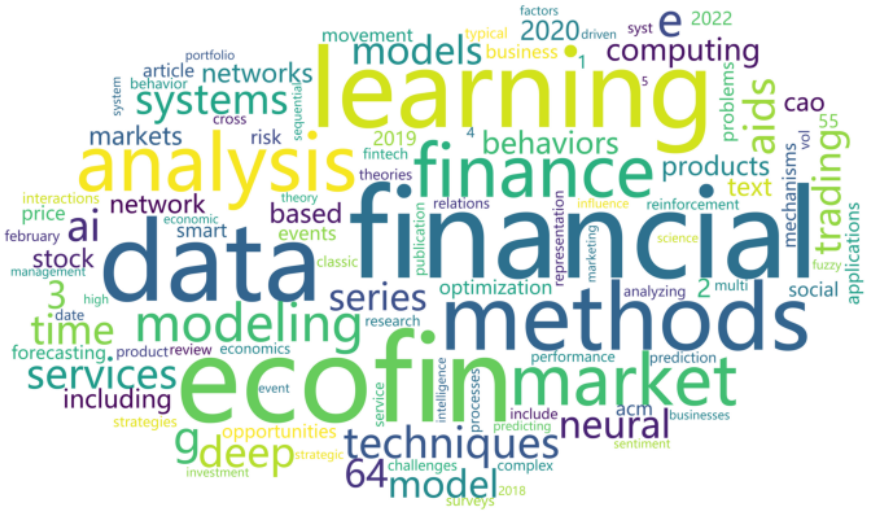
重要内容提示：

案例分析

AI in Finance: Challenges, Techniques, and Opportunities

LONGBING CAO, University of Technology Sydney, Australia

AI in finance refers to the applications of AI techniques in financial businesses. This area has attracted attention for decades, with both classic and modern AI techniques applied to increasingly broader areas of finance, economy, and society. In contrast to reviews on discussing the problems, aspects, and opportunities of finance benefited from specific or some new-generation AI and data science (AIDS) techniques or the progress of applying specific techniques to resolving certain financial problems, this review offers a comprehensive and dense landscape of the overwhelming challenges, techniques, and opportunities of AIDS research in finance over the past decades. The challenges of financial businesses and data are first outlined, followed by a comprehensive categorization and a dense overview of the decades of AIDS research in finance. We then structure and illustrate the data-driven analytics and learning of financial businesses and data. A comparison, criticism, and discussion of classic versus modern AIDS techniques for finance follows. Finally, the open issues and opportunities to address future AIDS-empowered finance and finance-motivated AIDS research are discussed.



PDF文件读取

```
import pdfplumber
pdf = pdfplumber.open('公司A理财公告.PDF')
pages = pdf. pages
text_all = []
for page in pages:                    # 遍历pages中每一页的信息
    text = page. extract_text()       # 提取当页的文本内容
    text_all. append(text)            # 汇总每一页内容
text_all = ''. join(text_all)         # 把列表转换成字符串
print(text_all)                       # 打印全部文本内容
pdf. close()
```

案例分析


中国共产党第二十次全国代表大会


社会主义
发展
坚持
人民
中国
全面
我们
推进
建设
国家
现代化
体系
战略
时代
制度
重大
加强
推动
安全
伟大
实现
特色
实施
文化
政治
加快
完善
贯彻
世界
社会
民主
中国共产党
中华民族
保障
复兴
精神

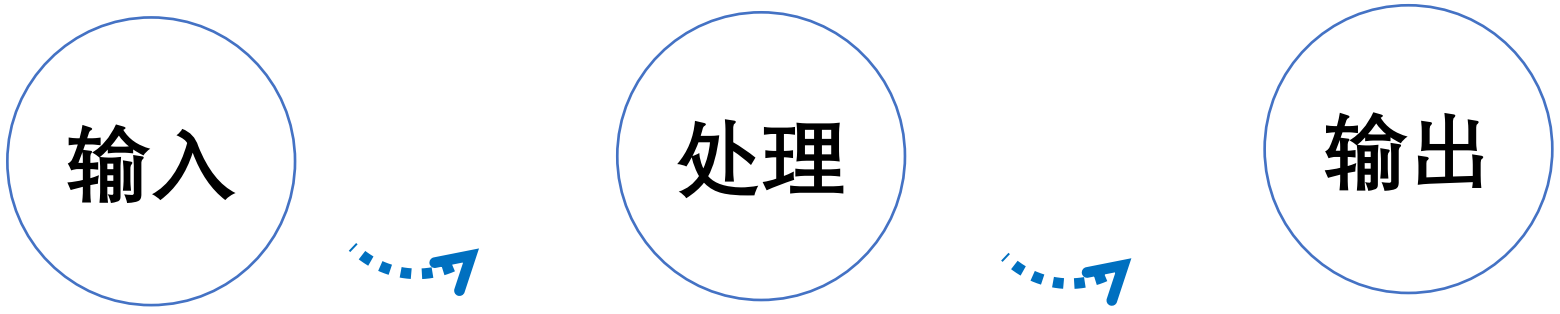
关于文本词频统计

词频统计的内涵：累加问题，即对文档中的每个词设计一个计数器，词语出现一次，计算器加1，词和次数是一对出现，构成

<单词>：<出现次数>

键值对：字典

词频统计问题的IPO描述



从文件中读取一篇待分析的文章

采用字典数据结构统计词语出现的频率

根据词频进行图形绘制或统计高频词语

jieba库分词原理

- 提供中文词库 pip install jieba
- 将待分词的内容与分词词库进行比对
- 通过**图结构和动态规划方法**找到最大概率词组
- 增加自定义中文单词的功能

jieba库三种分词模式

精确模式：将句子最精确地切开，适合文本分析

```
>>>import jieba

>>>jieba. lcut("中华人民共和国是一个伟大的国家")
```

jieba库三种分词模式

全 模 式： 把句子中所有可以成词的词语都扫描出来，速度非常快，但不能消除歧义

```
>>>import jieba

>>>jieba. lcut ("中华人民共和国是一个伟大的国家" ,cut_all=True)
```

jieba库三种分词模式

搜索引擎模式： 在精确模式基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词

```
>>>import jieba

>>>jieba. lcut_for_search("中华人民共和国是一个伟大的国家")
```

添加专属名词

```
>>>import jieba
>>>jieba.lcut("习大大希望中国的老百姓有更好的生活")
>>>jieba.add_word("习大大")
>>>jieba.lcut("习大大希望中国的老百姓有更好的生活")
```

谢　谢