

python语言入门与实践

文件和异常

主讲人：吴陈炜 博士

复旦大学智能机器人研究院 工程博士

北京大学信息科学与技术学院 理学硕士

浙江大学信息与电子工程学系 工学学士

网易（杭州）网络有限公司 项目经理

美国项目管理协会 PMP（项目管理专业人士）

之江实验室 科研主管

2020 年 6 月 2日

一道LeetCode习题综合运用初级班的知识

题目：字母异位词分组（LeetCode49）

给定一个字符串数组，将字母异位词组合在一起。字母异位词指字母相同，但排列不同的字符串。

示例：

输入: ["eat", "tea", "tan", "ate", "nat", "bat"]

输出: [["ate","eat","tea"], ["nat","tan"], ["bat"]]

说明：所有输入均为小写字母；不考虑答案输出的顺序。

解题思路

- 这道题让我们群组给定字符串集中所有的错位词（异位词），所谓的错位词就是两个字符串中字母出现的次数都一样，只是位置不同，比如abc, bac, cba等它们就互为错位词，那么我们如何判断两者是否是错位词呢，我们发现如果把错位词的字符顺序重新排列，那么会得到相同的结果，所以重新排序是判断是否互为错位词的方法，由于错位词重新排序后都会得到相同的字符串，我们以此作为key，将所有错位词都保存到字符串数组中，建立key和字符串数组之间的映射，最后再存入结果res中即可。

参考解决方案

```
1 lst = ["eat", "tea", "tan", "ate", "nat", "bat"]
2 dct = {}
3 for i in lst:
4     key = str(sorted(i))
5     if key not in dct.keys():
6         dct[key] = [i]
7     else:
8         dct[key].append(i)
9 res = [val for val in dct.values()]
10 print(res)
```

```
[['eat', 'tea', 'ate'], ['tan', 'nat'], ['bat']]
[Finished in 0.2s]
```

涉及知识点

- 列表的创建
- 列表生成式
- 字典的创建
- 字典的keys和values方法
- 字典的赋值
- for循环
- sorted函数
- if-else判断
- in和not关键字
- print函数

本节课的目标

- 掌握文件的概念的操作方法
- 掌握python新手常见的错误和对应异常
- 掌握python中异常的处理方法（try-except-else语句）
- 综合运用文件和异常处理方法解决实际问题

文件的概念

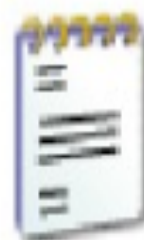
计算机中的文件计算机是信息处理的一种电子工具,信息是以数据的形式在磁盘中保存的,在计算机中的一段数据就是一个文件,它可以是一段声音、一篇文章或者一段动画等。这些文件也可以存放在不同的文件夹中。文件就是我们起好了名字并存储在磁盘上的信息的集合。



压缩文件.zip
WinRAR ZIP 压缩文件
1 KB



图形文件.bmp
1024 x 768
ACDSee BMP 图像



文本文档.txt
文本文档
0 KB



视频文件.MPG
MPG 文件
51,020 KB



音频文件.mp3
清华大学校园歌手

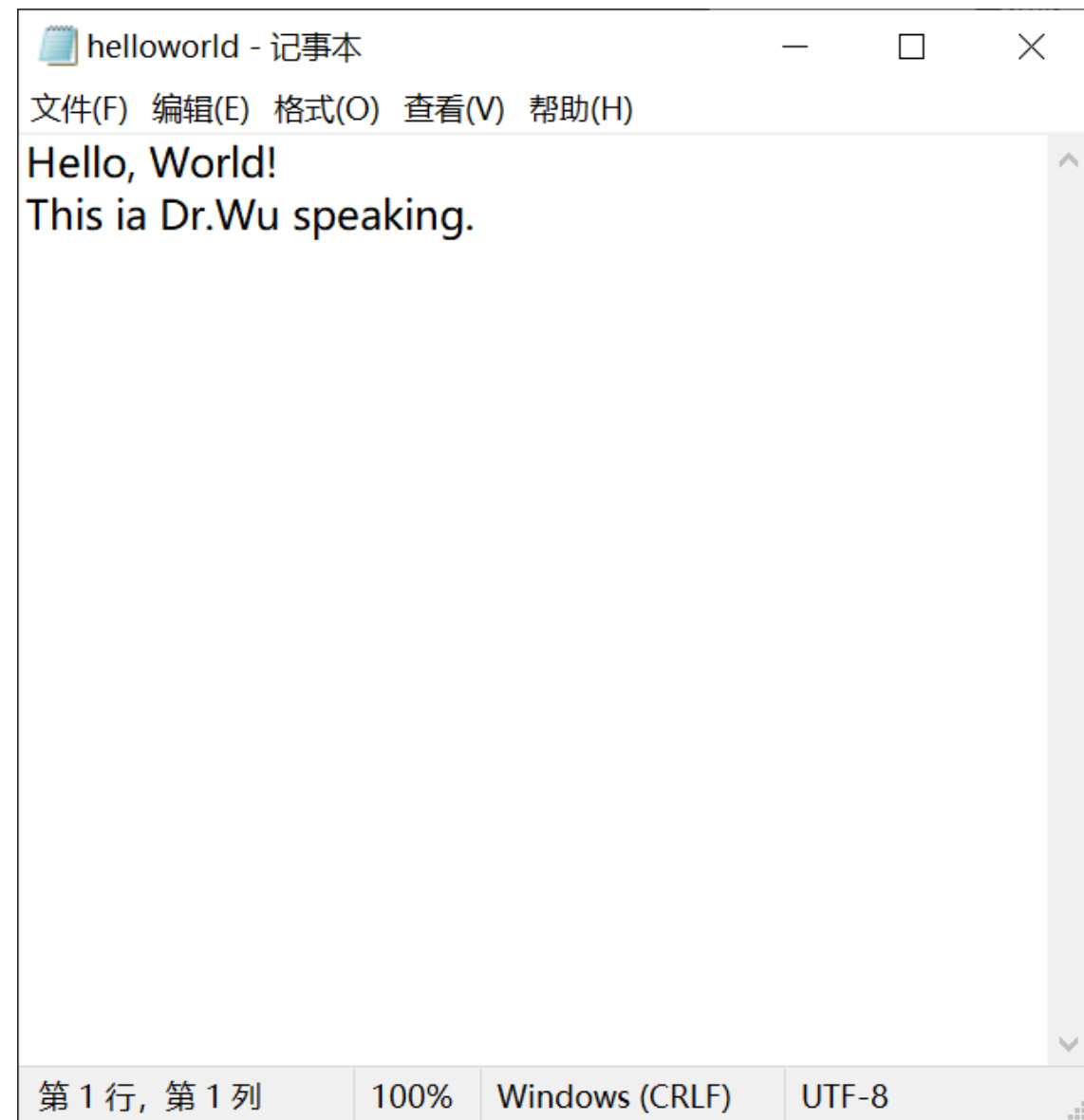


可执行文件.exe

计算机中常见的文件类型

文件操作的基本流程

- 第一步：打开文件，得到编辑窗口
- 第二步：通过窗口，读写修改文件
- 第三步：关闭窗口，保存文件内容



用python读取文件

知识点:

- 使用open函数来打开一个文件
- 使用文件对象的readlines()方法

或read()方法读取文件中的内容

- 使用文件对象的close()方法来
关闭文件

感受一下read和readlines方法的区别:

字符串v.s.字符串列表

```
1 f = open('./helloworld.txt', 'r')
2 for line in f.readlines():
3     print(line)
4 f.close()
```

```
Hello, World!
```

```
This is Dr.Wu speaking.
[Finished in 0.1s]
```

More pythonic way

- with open大法: python帮你自动关闭文件和处理异常

```
with open('./helloworld.txt','r') as file:  
    for content in file.readlines():  
        print(content)
```

```
Hello, World!
```

```
This is Dr.Wu speaking.  
[Finished in 0.1s]
```

用python写文件

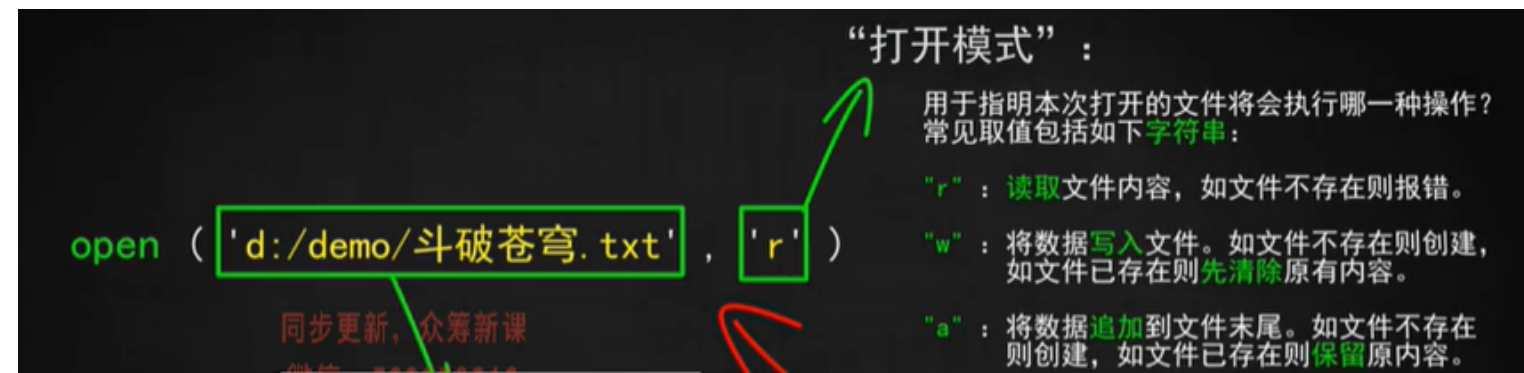
- 写入内容覆盖原来的文件

```
with open('./helloworld.txt','w') as file:  
    file.write("I love python.\n")
```

- 写入内容附在原文件的后面

```
with open('./helloworld.txt','a') as file:  
    file.write("I love python.\n")
```

更多文件的打开方式



```
f = open("test.txt", 'r') # 读模式
f = open("test.txt", 'w') # 写模式
f = open("img.bmp", 'r+') # 可读可写
f = open("img.bmp", 'w+') # 可读可写
f = open("img.bmp", 'rb') # 二进制读取
```

r	文本模式	读取
rb	二进制模式	读取
w	文本模式	写入
wb	二进制模式	写入
a	文本模式	追加
ab	二进制模式	追加
+		可读可写

python中的异常

- 初学者使用python的时候经常会犯一些，比如：
 - 使用中文括号
 - 同时使用空格和tab进行缩进
 - 在代码最后遗漏了括号或者冒号
 - ...
- 对于这些常见错误，python会抛出（raise）错误并终止程序

python中的异常机制

- Python使用被称为异常的特殊对象来管理程序执行期间发生的错误。每当发生让Python不知所措的错误时，它都会创建一个异常对象。如果你编写了处理该异常的代码，程序将继续运行；如果你未对异常进行处理，程序将停止，并显示一个traceback，其中包含有关异常的报告。

常见的报错类别

异常	描述
<code>NameError</code>	尝试访问一个没有声明的变量
<code>ZeroDivisionError</code>	除数为0
<code>SyntaxError</code>	语法错误
<code>IndexError</code>	索引超出序列范围
<code>KeyError</code>	请求一个不存在的字典关键字
<code>IOError</code>	输入输出错误（比如你要读的文件不存在）
<code>AttributeError</code>	尝试访问未知的对象属性
<code>ValueError</code>	传给函数的参数类型不正确，比如给 <code>int()</code> 函数传入字符串形

此外还有等`TypeError`、`IndentationError`等等，可百度关键词“python标准异常”获取更多关于异常的信息。

SyntaxError

```
28 print('hello')
File "C:\Users\wuchenwei\Desktop\py
File "C:\Users\wuchenwei\Desktop\pythoncras
hcourse\D9\testday9.py", line 28
    print('hello')
           ^
SyntaxError: invalid character in identifier
[Finished in 0.1s with exit code 1]
[shell_cmd: python -u "C:\Users\wuchenwei\Des
ktop\pythoncrashcourse\D9\testday9.py"]
[dir: C:\Users\wuchenwei\Desktop\pythoncrashc
ourse\D9]
[path: C:\windows\system32;C:\windows;C:\wind
ows\System32\Wbem;C:\windows\System32\Windows
PowerShell\v1.0\;C:\windows\System32\OpenSSH\
```

SyntaxError: invalid character in identifier

SyntaxError

```
29 name = 'lilei'
30 if name == 'lilei'
```

File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py", line 30

```
31     print('hello')
```

File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py", line 30

```
    if name == 'lilei'|
                        ^
```

SyntaxError: invalid syntax

[Finished in 0.3s with exit code 1]

[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py"]

[dir:

C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]

[path: C:\windows\system32;C:\windows;C:\windows\S

SyntaxError: invalid syntax

IndentationError

```
29 str1 = "hello world!"  
30 print(str1)
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py", line 30
```

```
print(str1)  
^
```

IndentationError: unexpected indent

[Finished in 0.2s with exit code 1]

[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py"]

[dir:

C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]

IndentationError: unexpected indent

IndentationError

```
29 str1 = ['jack', 'rose', 'lilei']  
30 for name in str1:  
31 print(name)
```

File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py", line 31

```
print(name)  
    ^
```

IndentationError: expected an indented block

[Finished in 0.2s with exit code 1]

[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py"]

[dir:

C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]

IndentationError: expected an indented block

IndentationError

```
35 def func1(x):  
36     y = 2  
37     print(y)
```

File "C:\Users\wuchenwei\Desktop\pythoncrashc

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9  
\testday9.py", line 37  
    print(y)  
      ^
```

IndentationError: unindent does not match any outer
indentation level

[Finished in 0.1s with exit code 1]

[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pytho
ncrashcourse\D9\testday9.py"]

[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]

IndentationError: unindent does not match any outer indentation level

TypeError

```
39 age = 18
40 str1 = '我今年' + age + '喜欢音乐'
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashc
```

```
Traceback (most recent call last):
  File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py", line 40, in <module>
    str1 = '我今年' + age + '喜欢音乐'
TypeError: must be str, not int
[Finished in 0.2s with exit code 1]
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pytho
ncrashcourse\D9\testday9.py"]
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
[path: C:\windows\system32;C:\windows;C:\windows\System
32\Wbem;C:\windows\System32\WindowsPowerShell\v1.0\;C:\
windows\System32\OpenSSH\;C:\Program Files (
```

TypeError: must be str, not int

TypeError

```
42 square = 3
43 square(3)
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashc
```

```
Traceback (most recent call last):
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py", line 43, in <module>
    square(3)
```

```
TypeError: 'int' object is not callable
```

```
[Finished in 0.2s with exit code 1]
```

```
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py"]
```

```
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

```
[path: C:\windows\system32;C:\windows;C:\windows\System32\Wbem;C:\windows\System32\WindowsPowerShell\v1.0\;C:\windows\System32\OpenSSH\;C:\Program Files (
```

TypeError: 'int' object is not callable

TypeError

```
day10.py x testday9.py
45 def addnumber(a,b):
46     c = a+b
47     return(c)
48 addnumber(6)

File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\
D9\testday9.py", line 49, in <module>
    addnumber(6)
TypeError: addnumber() missing 1 required
positional argument: 'b'
[Finished in 0.2s with exit code 1]
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\py
thoncrashcourse\D9\testday9.py"]
[dir:
C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

TypeError: addnumber() missing 1 required positional argument: 'b'

TypeError

```
day10.py x testday9.py x
45 def addnumber(a,b):
46     c = a+b
47     return(c)
48 addnumber(6,3,3)

File "C:\Users\wuchenwei\Desktop\pythoncrashco

Traceback (most recent call last):
  File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\t
estday9.py", line 48, in <module>
    addnumber(6,3,3)
TypeError: addnumber() takes 2 positional arguments but
3 were given
[Finished in 0.2s with exit code 1]
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythonc
rashcourse\D9\testday9.py"]
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
[path: C:\windows\system32;C:\windows;C:\windows\System32
```

TypeError: addnumber() takes 2 positional arguments but 3 were given

IndexError

```
50 name = ['jack', 'rose', 'lilei']  
51 print(name[3])
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashco
```

```
Traceback (most recent call last):
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py", line 51, in <module>
```

```
    print(name[3])
```

```
IndexError: list index out of range
```

```
[Finished in 0.2s with exit code 1]
```

```
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py"]
```

```
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

IndexError: list index out of range

NameError

```
52 y = x + 3
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashco
```

```
53 print(y)
```

```
Traceback (most recent call last):
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py", line 52, in <module>
```

```
y = x + 3
```

```
NameError: name 'x' is not defined
```

```
[Finished in 0.1s with exit code 1]
```

```
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py"]
```

```
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

```
[path: C:\windows\system32;C:\windows;C:\windows\System32
```

NameError: name 'x' is not defined

ValueError

```
81 string = 'd'
82 print(int(string))
```

File "C:\Users\wuchenwei\Desktop\pythoncr

```
Traceback (most recent call last):
  File "C:\Users\wuchenwei\Desktop\pythoncrashcours
e\D9\testday9.py", line 82, in <module>
    print(int(string))
ValueError: invalid literal for int() with base
10: 'd'
[Finished in 0.1s with exit code 1]
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\p
ythoncrashcourse\D9\testday9.py"]
[dir:
C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

ValueError: invalid literal for int() with base 10: 'd'

FileNotFoundError

```
74 with open('pythontxt.txt', 'r') as file:
```

```
File "C:\Users\wuchenwei\Desktop\pythoncr
```

```
75     pass
```

```
76
```

Traceback (most recent call last):

File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py", line 74, in <module>

with open('pythontxt.txt', 'r') as file:

FileNotFoundError: [Errno 2] No such file or directory: 'pythontxt.txt'

[Finished in 0.2s with exit code 1]

[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\testday9.py"]

[dir:

C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]

FileNotFoundError: [Errno 2] No such file or directory: 'pythontxt.txt'

ZeroDivisionError

```
55 print(5/0)
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashco
```

```
Traceback (most recent call last):
```

```
File "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py", line 55, in <module>  
    print(5/0)
```

```
ZeroDivisionError: division by zero
```

```
[Finished in 0.2s with exit code 1]
```

```
[shell_cmd: python -u "C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9\testday9.py"]
```

```
[dir: C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9]
```

```
[path: C:\windows\system32;C:\windows;C:\windows\System32
```

ZeroDivisionError: division by zero

try-except语句来自定义处理异常

- 我们可以通过try-except语句来处理可能发生的异常，让python尝试运行一些代码，并告诉它如果这些代码引发了指定的异常，该怎么办。

```
54  try:
55      print(5/0)
56  except ZeroDivisionError:
57      print("You can't divide a number by zero!")
```

```
You can't divide a number by zero!
[Finished in 0.2s]
```


try-except-else语句

```
print('Give me two numbers, I will divide them.')
print('Enter q to quit.')

while True:
    first_number = input('\nPlease input the first number:')
    if first_number == 'q':
        break
    second_number = input('\nPlease input the second number:')
    if second_number == 'q':
        break
    try:
        answer = int(first_number)/int(second_number)
    except ZeroDivisionError:
        print('You cannot divide by 0!')
    except ValueError:
        print("You should print numbers or lower number 'q'")
    else:
        print(answer)
```

```
Give me two numbers, I will divide them.
Enter q to quit.

Please input the first number:5
Please input the second number:2
2.5

Please input the first number:5
Please input the second number:0
You cannot divide by 0!

Please input the first number:3
Please input the second number:a
You should print numbers or lower number 'q'

Please input the first number:3
Please input the second number:q

C:\Users\wuchenwei\Desktop\pythoncrashcourse\D9>
```

体会Try-except语句的作用

- 程序崩溃可不是什么好事，让用户看到traceback也不是好主意。不懂技术的用户会被它们搞糊涂，而且如果用户怀有恶意，他会通过traceback获悉你不希望他知道的信息。例如，他将知道你的程序文件的名称，还将看到部分不能正确运行的代码。有时候，训练有素的攻击者可根据这些信息判断出可对你的代码发起什么样的攻击。
- 通过预测可能发生错误的代码，可编写健壮的程序，它们即便面临无效数据或缺少资源，也能继续运行，从而能够抵御无意的用户错误和恶意攻击。

资源分享：Gutenberg (<http://gutenberg.org/>)

- 可以分析包含整本书的文本文件。很多经典文学作品都是以简单文本文件的方式提供的，因为它们不受版权限制。本节使用的文本来自项目Gutenberg，这个项目提供了一系列不受版权限制的文学作品，如果你要在编程项目中使用文学文本，这是一个很不错的资源。



The screenshot shows the Project Gutenberg website. At the top, the URL 'gutenberg.org' is visible in the browser's address bar. Below the header, the main heading reads 'Free eBooks - Project Gutenberg'. A navigation bar contains links for 'Book search', 'Book categories', 'Browse catalog', 'Mobile site', and 'Report error'. A section titled 'Some of the Latest eBooks' displays a row of book covers, including 'Le secret du faucon', 'LE TRÉSOR', 'BILLY TOMORROW', 'The COLLECTOR'S GUIDE', 'KACHINA DOLLS', 'The art and the science of Indian doll-making', and 'APPLES at appleby night'. Below this, a 'Welcome' section provides an overview of the library, stating it contains over 60,000 free eBooks. It also includes a 'Looking for something to read?' section with advice on how to browse the collection and a note about a new website available for testing at <https://dev.gutenberg.org>.

gutenberg.org

Free eBooks - Project Gutenberg

[Book search](#) · [Book categories](#) · [Browse catalog](#) · [Mobile site](#) · [Report error](#)

Some of the Latest eBooks



Welcome

Project Gutenberg is a library of over 60,000 free eBooks. Choose among them to download them or read them online. You will find the world's great literary works for which U.S. copyright has expired. Thousands of volunteers digitize eBooks, for enjoyment and education.

Looking for something to read? Project Gutenberg eBooks are mostly old, published before 1924, with some published in the decades after. Use one of our [search engines](#), or try using the [Bookshelves](#) to browse by genre, age group, and top authors.

New website available for testing. Visit <https://dev.gutenberg.org> (or [https://dev.gutenberg.org](#))

复习一下：如何统计一本书的字数

```
77 contents = "Alice was beginning  
78 words = contents.split()  
79 print(len(words))
```

```
57  
[Finished in 0.1s]
```

思路，把小说内容放在一个content字符串中，然后使用split方法将其分割为字符串列表，再使用len函数来统计字符串的长度就可以得到字数的数值。

提升一下，如何统计多本书的字数

- 编写函数（重构代码）方便调用进行批处理
- 将多本书书名存在列表中便于遍历
- 注意可能会出现报错类型：
 - **FileNotFoundError**
 - **UnicodeDecodeError**

具体的代码

```
1  def count_words(filename):
2      try:
3          with open(filename, 'r', encoding='utf-8') as f:
4              contents = f.read()
5          except FileNotFoundError:
6              return(f"The file {filename} doesn't exist. Please check!")
7          except UnicodeDecodeError:
8              return(f"Please encode the file with utf-8 format.")
9      words = contents.split()
10     return len(words)
11
12  book_list = ['alice.txt', 'princess.txt', 'siddhartha.txt', 'moby.txt']
13  for book in book_list:
14      print(count_words(book))
15
```

26446

69392

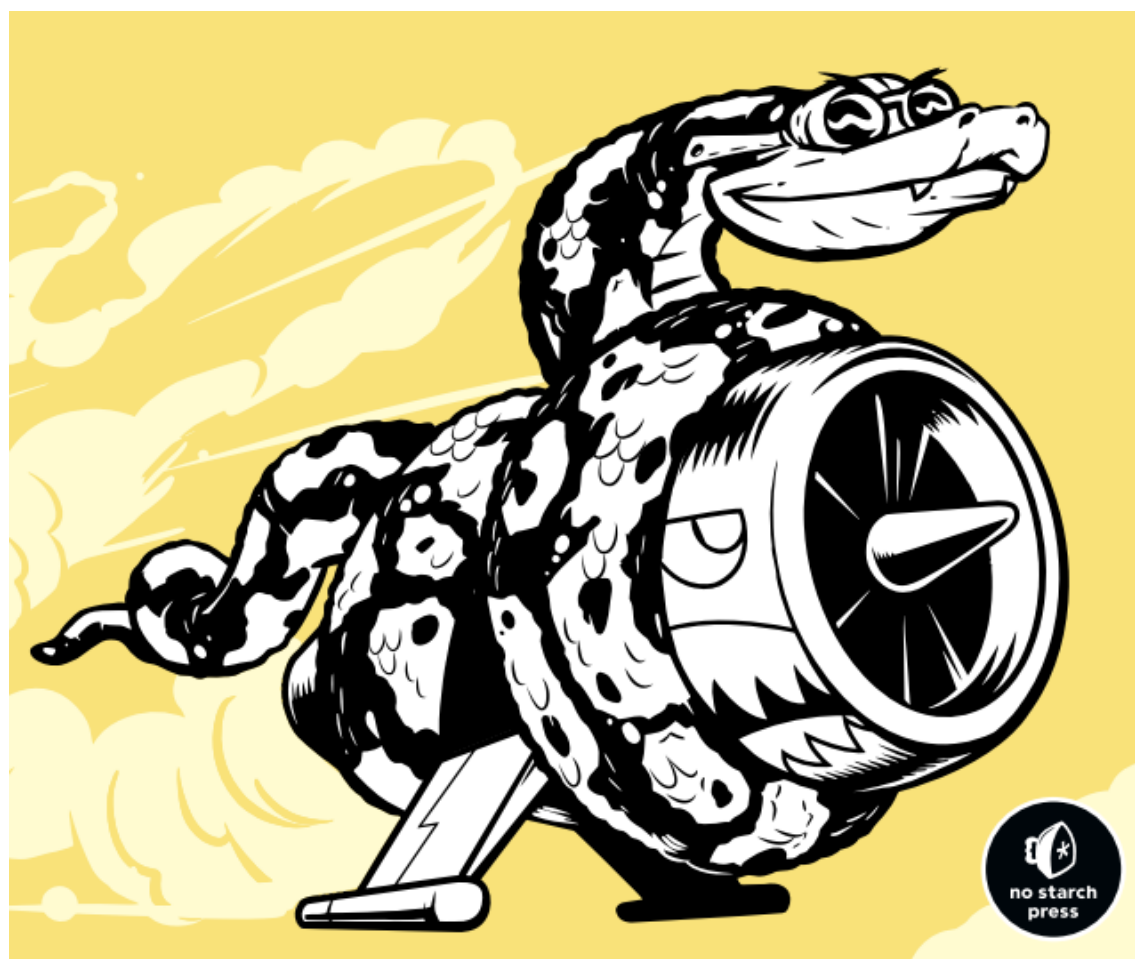
The file siddhartha.txt doesn't exist. Please check!

212169

[Finished in 0.2s]

作业：

- 阅读Python Crash Course第十章10.1-10.3小节
- 完成练习题10.1-10.10



10	189
FILES AND EXCEPTIONS	
Reading from a File	190
Reading an Entire File	190
File Paths	191
Reading Line by Line	193
Making a List of Lines from a File	194
Working with a File's Contents	194
Large Files: One Million Digits	195
Is Your Birthday Contained in Pi?	196
Exercise 10-1: Learning Python	197
Exercise 10-2: Learning C	197
Writing to a File.	197
Writing to an Empty File	197
Writing Multiple Lines	198
Appending to a File.	199
Exercise 10-3: Guest	199
Exercise 10-4: Guest Book	199
Exercise 10-5: Programming Poll.	199
Exceptions.	200
Handling the ZeroDivisionError Exception	200
Using try-except Blocks	200
Using Exceptions to Prevent Crashes	201

综合训练

- ## ■ 通过读文件的方式读取托福100篇阅读理解，统计高频词汇



下节课老师介绍

■ 常老师

- 中国科学院 高能物理研究所 助理研究员
- 北京航空航天大学 能源与动力工程 博士
- 发表SCI和EI论文多篇，核心论文与会议论文若干。对Python有非常丰富的实战经验