## 实验七、文件和数据格式化

实验目的：掌握文件的读写等操作；掌握PIL、json库的使用

实验要求：

1、能对文件进行打开、关闭、读、写等基本操作。

2、运用CSV和json格式对多维数据进行存储和读写。

3、能使用PIL库对图像文件进行常见的修图操作。

实验学时：4学时（实验1-2：2学时；实验3：2学时）

实验内容：

1. 程序练习题7.1

说明：此题难度较大，为降低难度，我们不要求生成后的文件能够被python解释器正确执行。

提示：python中有33个保留字，可以用以下代码片段获取：

import keyword

kws = keyword.kwlist

2. 程序练习题7.2

提示：比较教材表7.7中的方法Image.thumbnail(size)和表7.8中的Image.resize(size)两个函数在调整图像大小应用上的差异。

3. 程序练习题7.5-7.6

参考代码：

import os

def userOperateInterface():

print("\n请选择词典功能")

print("i: 添加单词")

print("s: 查询单词")

print("Q: 退出词典")

print("请选择功能：")

return input()

def addWord(wordDict:dict, fileName):

#添加自己的代码

def selectWord(wordDict:dict):

#添加自己的代码

def main():

wordDict = {}

if 字典库文件存在:

装载字典库到wordDict

else:

字典库文件不存在则创建一个文件

print("\*\*\*\*\*\*欢迎使用简明英汉词典\*\*\*\*\*\*")

while True:

op = userOperateInterface()

if op == 'i':

addWord(wordDict, 'dict.txt')

elif op == 's':

selectWord(wordDict)

elif op == 'Q':

break

else:

print("输入有误\n")

main()

实验代码

7.1

import keyword

stopwords = '

\

t

\

n

\

r: ()'

functionwords = '.('

word = []

output = ''

lastAvailable = ['from', 'import']

last = False

def readFile(path):

file = open(path,'r',encoding = 'utf

-

8')

string = file.read()

return string[1:]

def parse(string):

global word

global output

for i in string:

if i in stopwords:

wd = '

'.join(word)

res = isKeyWord(wd)

if res == False:

if i not in functionwords and last == False:

wd = wd.upper()

if wd in lastAvailable:

last = True

else:

last = False

output += wd

output += i

word = []

else:

word.append(i)

def isKeyWord(string):

if string in keyword.kwlist:

return True

return False

def outPutFile():

file = open('D:

\

\

12.8.py','w',encoding = 'utf

-

8')

file.write(output)

string = readFile('D:

\

\

12.7.py')

parse(string)

outPutFile()

7.2

from PIL import Image

import os

import math

def getsize(path):

return os.stat(path).st\_size

def compress(path):

size = getsize(path)

ratio = math.sqrt(10\*1024/size)

im = Image.open(path)

height = im.height;

width = im.width;

m\_height = int(ratio\*height)

m\_width = int(ratio\*width)

ph = im.resize((m\_width, m\_height))

ph.save('test\_compressed.jpg',)

compress('D:

\

\

timg.jpg')

7.5

dict = {}

digits = '0123456789'

path = 'dict.txt'

def readFile(path, arg):

try:

file = open(path,arg,encoding = 'GBK')

except:

file = open(path,'w',encoding

= 'GBK')

return file

def readWords():

file = readFile(path, 'r')

while True:

line = file.readline()

if not line:

break

word = line.split(' ',2)

dict[word[0]] = word[1][:

-

1]

file.close()

def writeFile(word,dsp):

file = readFile(path, 'a

')

file.write('{} {}

\

n'.format(word, dsp))

file.close()

def modifyFile(word, dsp):

file = readFile(path, 'r')

line = file.readlines()

flen = len(line)

-

1

for i in range(flen):

if word in line[i]:

file.close()

line[i] = '{} {

}

\

n'.format(word, dsp)

file = readFile(path, 'w')

file.writelines(line)

break

file.close()

def editMode():

print('\*'\*50)

print('\*'\*50)

while True:

word = input("(

按数字键退出

)

请输入您想添加或修改的单词

:")

if word in digits:

print('\*'\*50)

print('\*'\*50)

return

try:

print('

该单词已存在于单词库

,

当前解释是：

{}'.format(dict[word]))

except:

print('

您添加的是一个新单词

')

print('

---------------------------------

')

description = input('

请输入您的解释

:

\

n')

try:

dict[word] += ',%s'%description

modifyFile(word, dict[word])

except KeyError:

dict[word] = '%s'%description

writeFile(word, dict[word])

print('

----------

添加完成

--------------

')

def searchMode():

print('\*'\*50)

print('\*'\*50)

while True:

word = input("(

按数字键退出

)

想查的单词

:")

if word in digits:

print('\*'\*50

)

print('\*'\*50)

return

print('

---------------------------------

')

try:

print(dict[word])

except KeyError:

print('

对不起，这个单词没有收录

')

print('

---------------------------------

')

def interface():

readWords()

def switch(option):

funcdic

= {

1:lambda: searchMode(),

2:lambda: editMode(),

3:lambda: exit()

}

return funcdic[option]()

while True:

print('

----------

欢迎使用英汉词典

----------

')

print('1.

查询单词

\

n2.

添加单词

\

n3.

退出

\

n')

option = int(input('

请输入您的选择：

'))

switch(option)

interf

ace()

7.6

dict = {}

file = None

digits = '0123456789'

def readWords():

global

file

file = open('C:

\

\

Users

\

\

GM

\

\

Desktop

\

\

补充单

词

.txt','r',encoding = 'GBK')

string = file.read()

def editMode():

print('\*'\*50)

print('\*'\*50)

while True:

word = input("(

按数字键退出

)

请输入您想添加或修改的单词

:")

if word in digits:

print('\*'\*50)

print('\*'\*50)

return

print('

---------------------------------

')

description = input('

请输入您的解释

:

\

n')

try:

dict[word] += ',%s'%description

except KeyError:

dict[word] = '%s'%description

print('

----------

添加完成

--------------

')

def searchMode():

print('\*'\*50)

print('\*'\*50)

while True:

word = input("(

按数字键退出

)

想查的单词

:")

if word in digits:

print('\*'\*50)

print('\*'\*50)

return

print('

---------------------------------

')

try:

print(dict[word])

except KeyError:

print('

对不起，这个单词没有收录

')

print('

---------------------------------

')

def interface():

def switch(option):

funcdic = {

1:lambda: searchMode(),

2:lambda: editMode(),

3:lambda: exit()

}

return funcdic[option]()

while True:

print('

----------

欢迎使用英汉词典

----------

')

print('1.

查询单词

\

n2.

添加单词

\

n3.

退出

\

n')

option = int(input('

请输入您的选择：

'))

switch(option)

interface()