**Practical No. 06**

1. Basic file handling operations.

Code:

file = open("Books.txt","w+")

l = ["This file is for books. \n","Books are good source of knowlegde.\n"]

file.write("Writing something in file.\n")

file.writelines(l)

file.close()

file = open("Books.txt","r+")

print(file.read())

file.seek(0)

print(file.readline())

file.seek(0)

print(file.read(9))

file.seek(0)

print(file.readline(9))

file.seek(0)

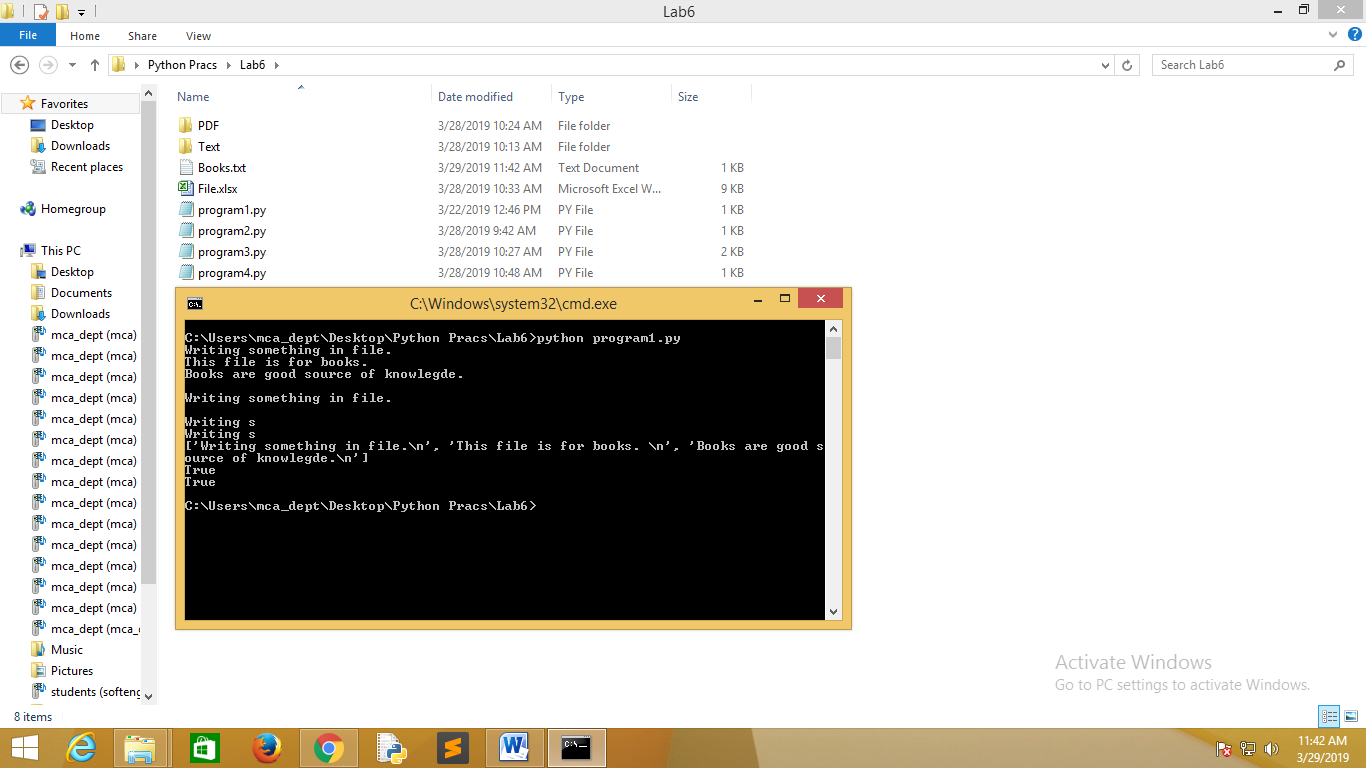
print(file.readlines())

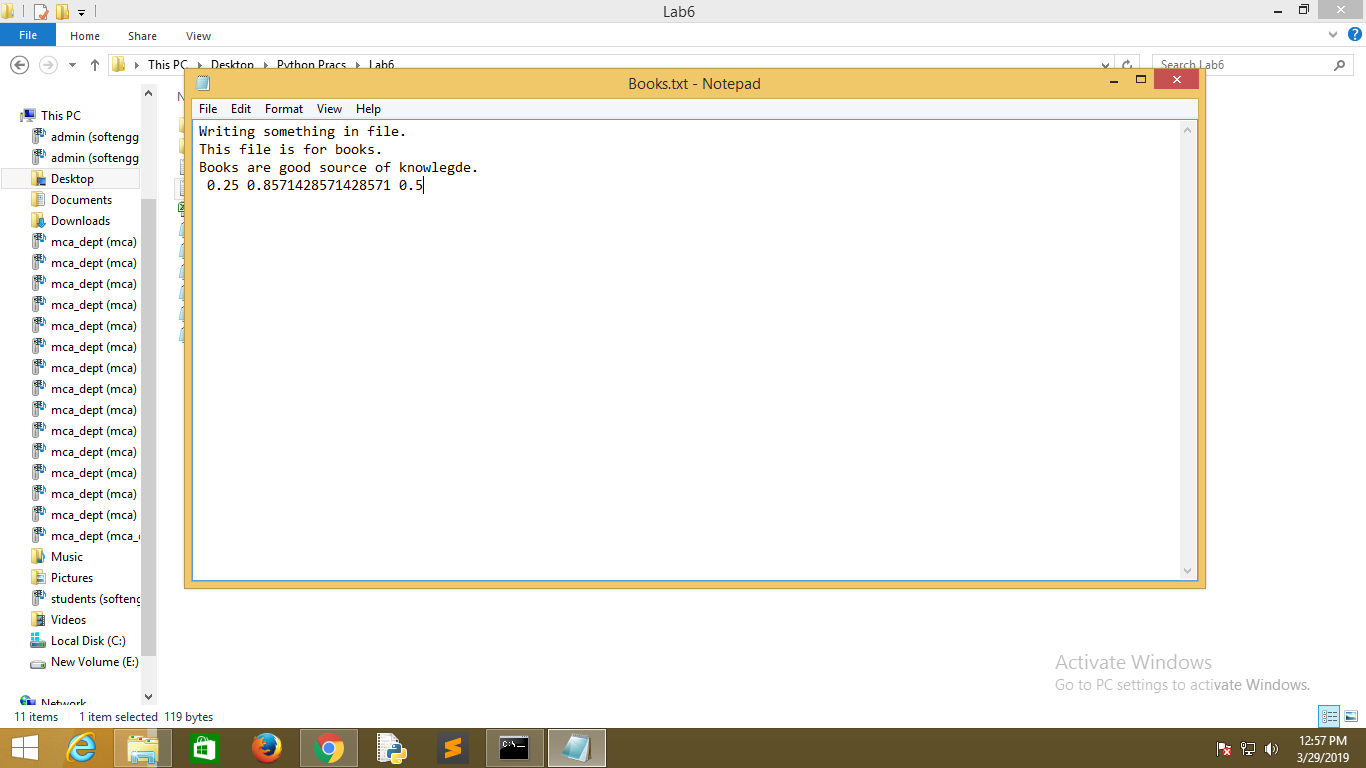
print(file.readable())

print(file.writable())

file.close()

Output:





1. To take a sentence and split and display each word one by one.

Code:

filePath = "Books.txt"

wordList = []

wordCount = 0

#Read lines into a list

file = open(filePath, 'r')

for line in file:

for word in line.split():

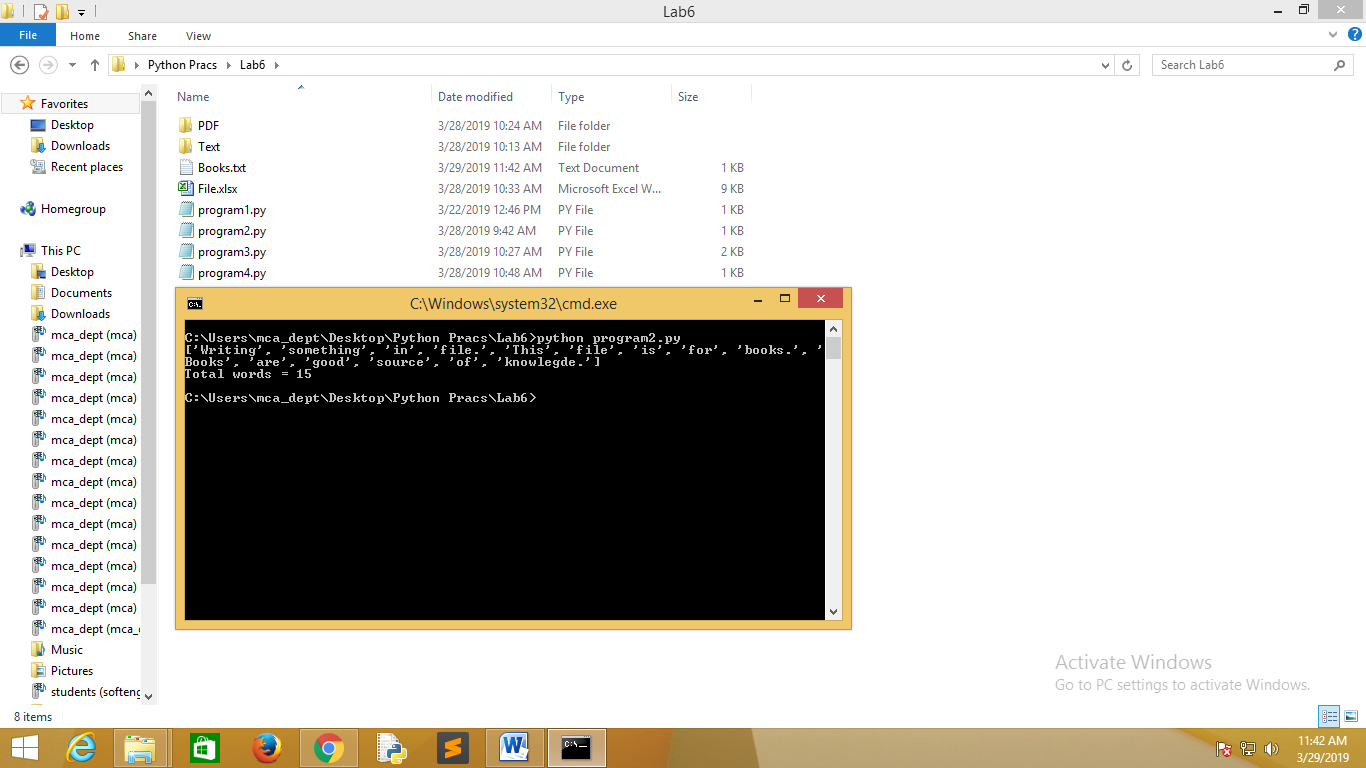
wordList.append(word)

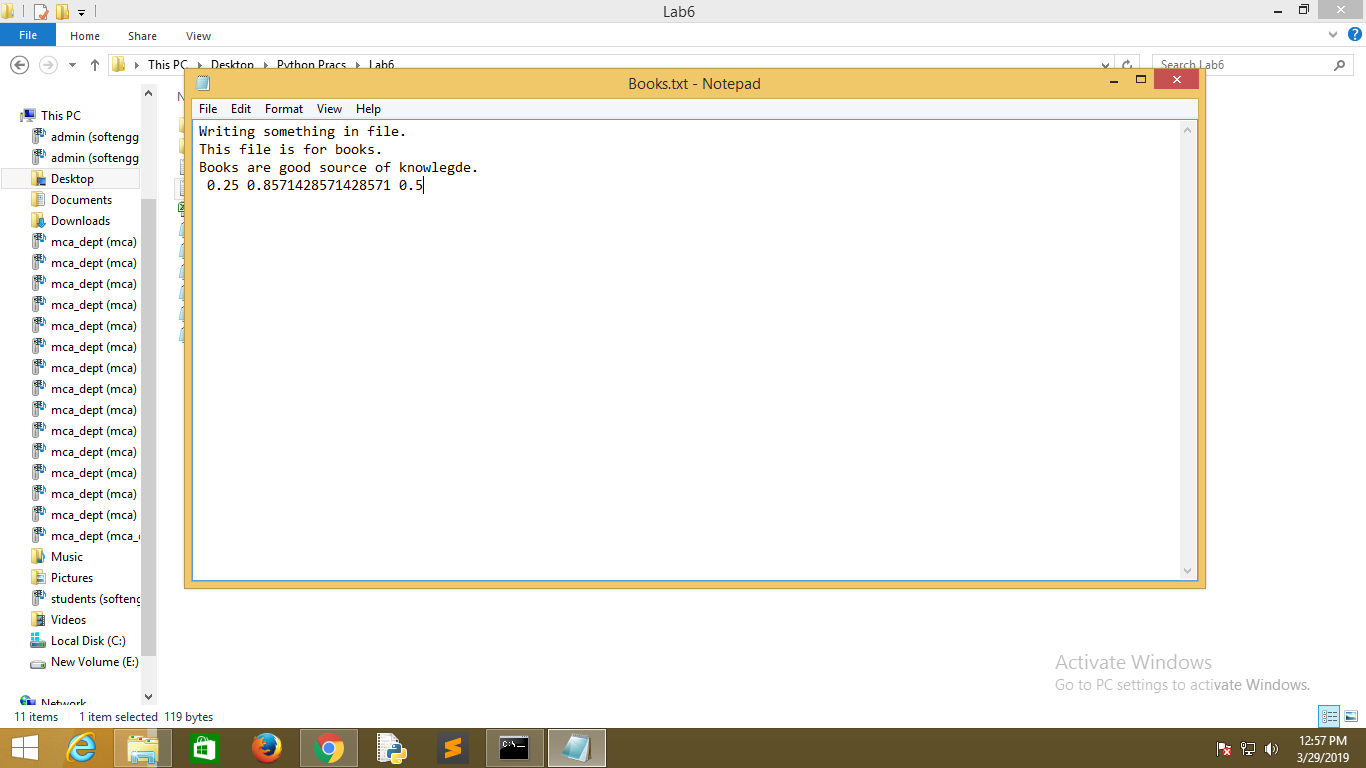
wordCount += 1

print(wordList)

print("Total words = %d" % wordCount)

Output:





1. To rearrange the files of different extension and group them accordingly into their folder.

Code:

import os,shutil

fullpath = os.path.join

start\_directory = "."

text\_files = "./Text"

pdf\_files = "./PDF"

html\_files = "./HTML"

jpeg\_files = "./JPEG"

def main():

for dirname, dirnames, filenames in os.walk(start\_directory):

for filename in filenames:

source = fullpath(dirname, filename)

if filename.endswith("txt"):

if not os.path.exists(text\_files):

os.mkdir(text\_files)

shutil.move(source, fullpath(text\_files, filename))

elif filename.endswith("pdf"):

if not os.path.exists(pdf\_files):

os.mkdir(pdf\_files)

shutil.move(source, fullpath(pdf\_files, filename))

elif filename.endswith("html"):

if not os.path.exists(html\_files):

os.mkdir(html\_files)

shutil.move(source, fullpath(html\_files, filename))

elif filename.endswith("jpeg"):

if not os.path.exists(jpeg\_files):

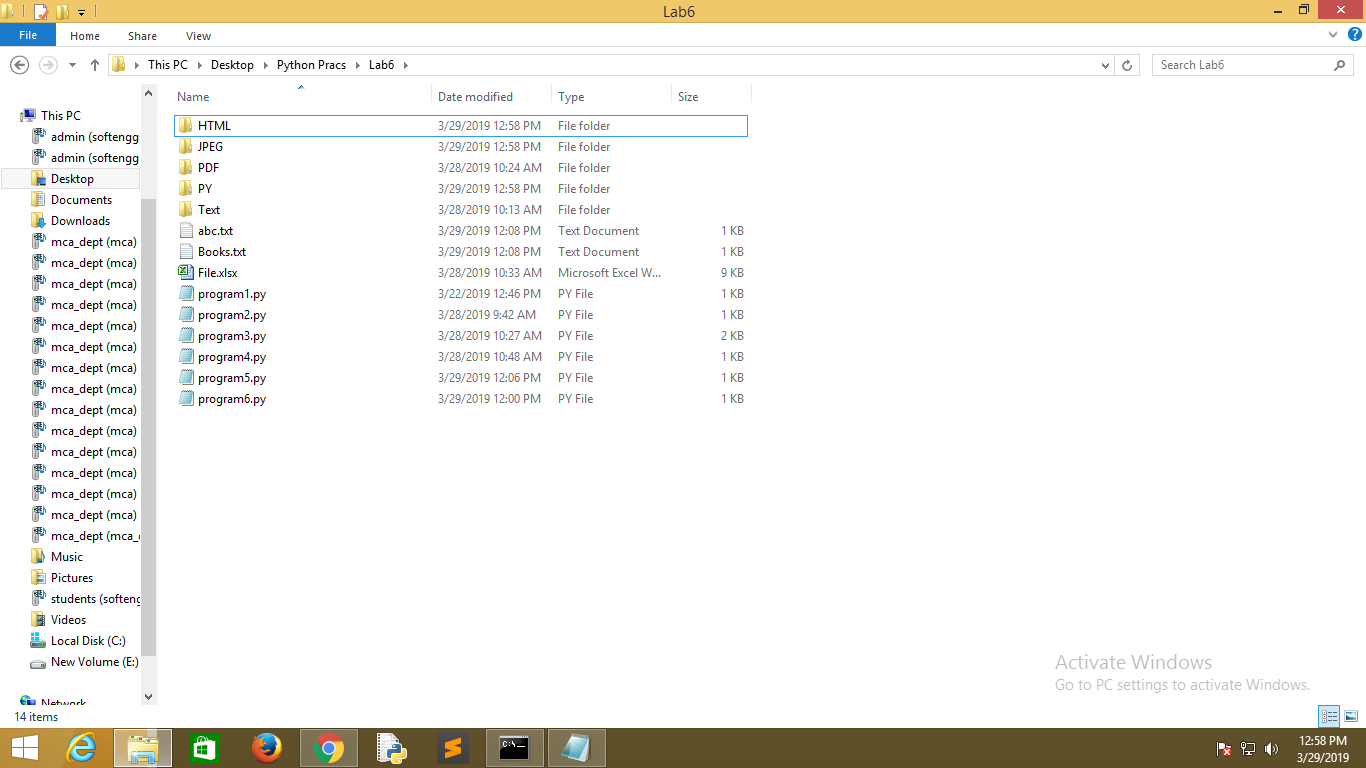
os.mkdir(jpeg\_files)

shutil.move(source, fullpath(jpeg\_files, filename))

if \_\_name\_\_ == "\_\_main\_\_":

main()

Output:



1. Reading an excel file.

Code:

import xlrd

file=xlrd.open\_workbook("File.xlsx");

worksheet=file.sheet\_by\_index(0);

len\_of\_row=worksheet.nrows;

len\_of\_col=worksheet.ncols;

for i in range(0,len\_of\_row):

for j in range(0,len\_of\_col):

print(worksheet.cell\_value(i,j),end=' ');

print("");

print("\n");

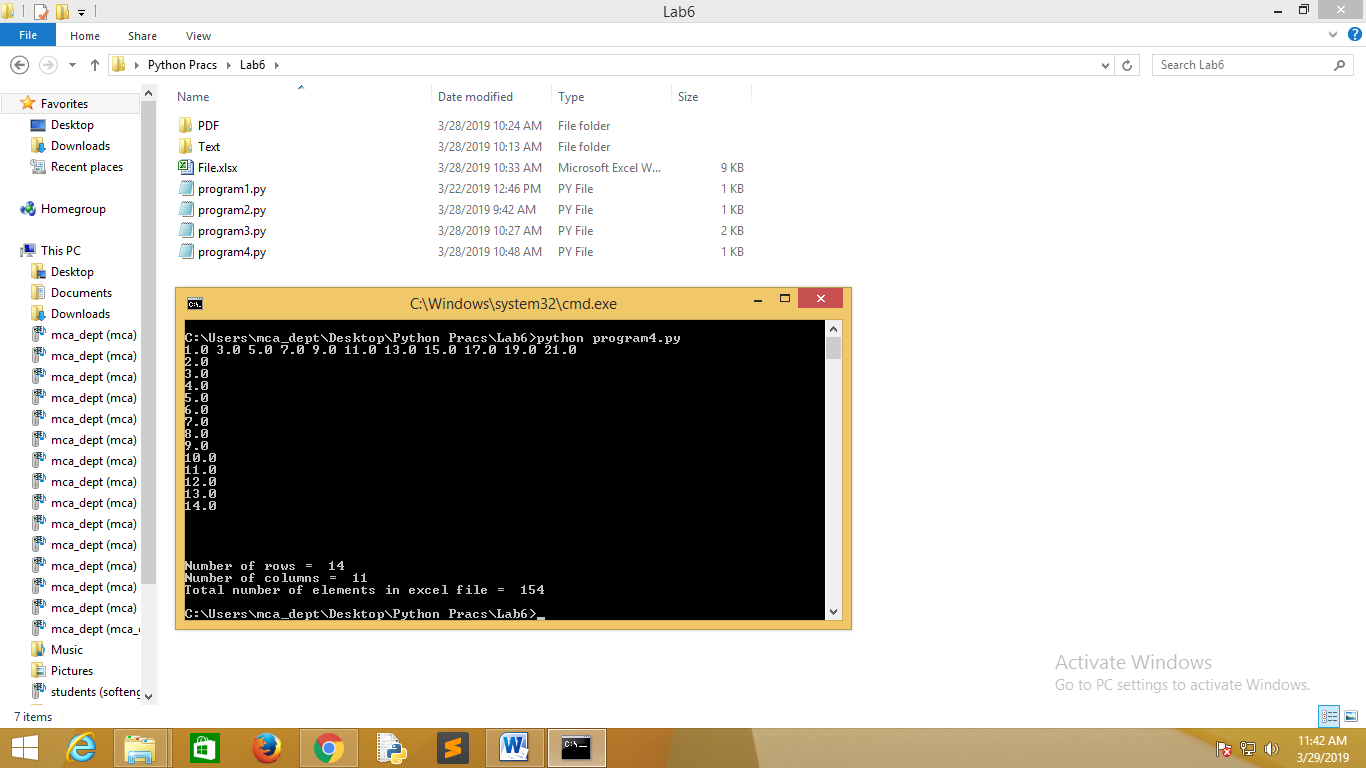
print("\n");

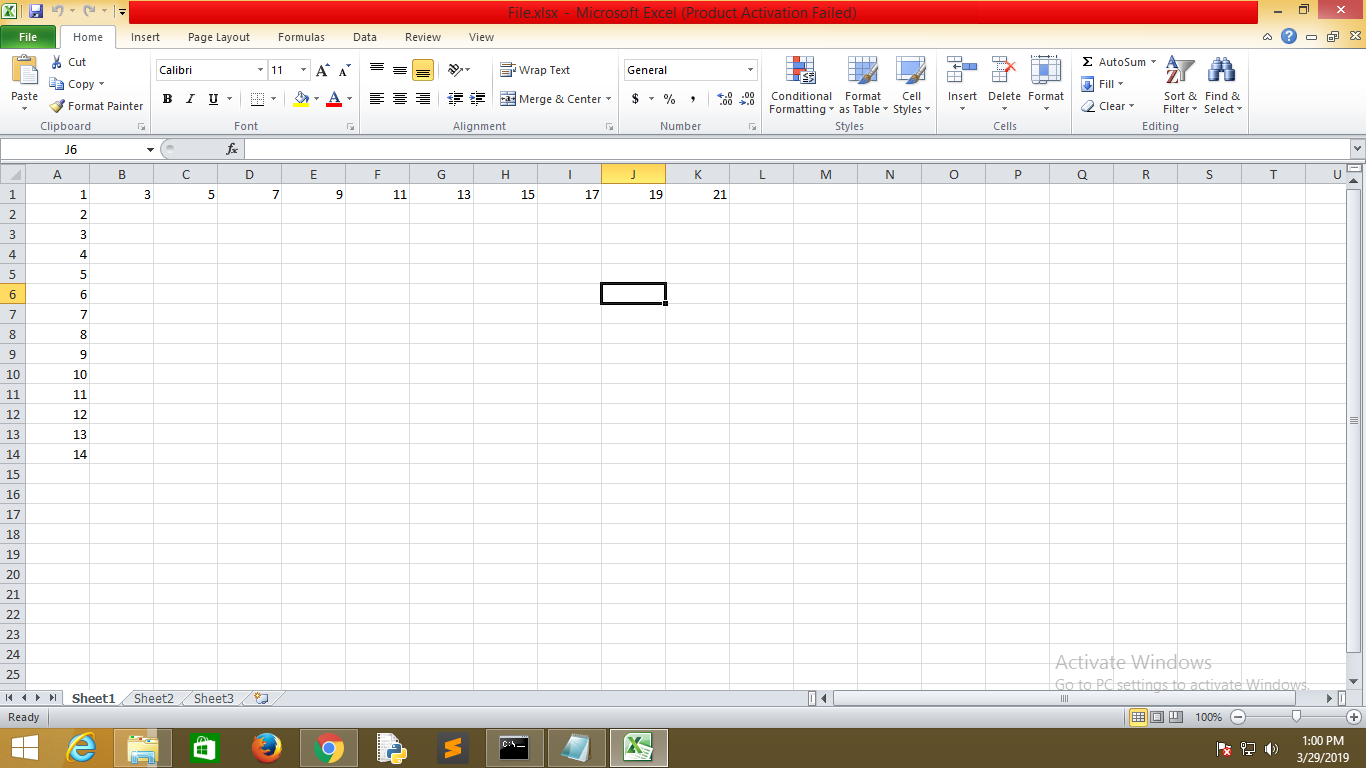
print("Number of rows = ",len\_of\_row);

print("Number of columns = ",len\_of\_col);

print("Total number of elements in excel file = ",len\_of\_row\*len\_of\_col);

Output:





1. Demonstrate inbuilt exception handling in python.

Code:

try:

import sys;

except ImportError:

print("Could not import sys module \n");

try:

file=open(input("Enter file name to open\n"),"rt");

print("File contents \n");

print(file.read());

except FileNotFoundError:

print("File was not found, system exiting\n");

sys.exit(0);

except IOError:

print("Could not read \n");

sys.exit(0);

try:

print("Enter two numbers whose division you want to save in a file \n");

a=int(input());

b=int(input());

file.close();

file=open(input("Enter file name to write \n"),"a");

file.write(" "+str(a/b));

file.close();

except FileNotFoundError:

print("Given file was not found ");

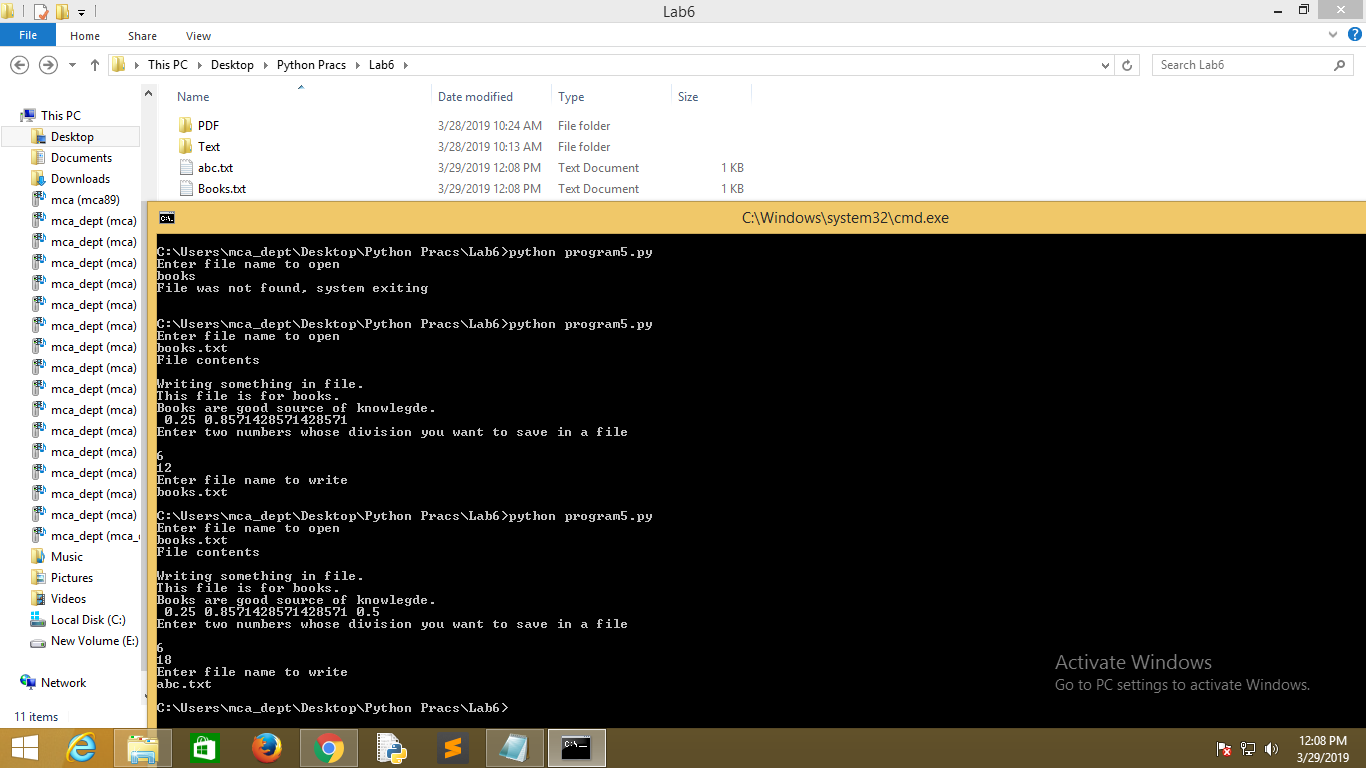
except IOError:

print("File cannot be opened ");

except ArithmeticError:

print("Illegal number entered ");

Output:



1. Demonstrate custom exception handling in python.

Code:

class err(Exception):

def \_\_init\_\_(self,error):

self.error\_name=error;

def get\_error(self):

return "{}".format(self.error\_name);

try:

name=input("Enter name ");

if not name:

raise(err("String cannot be empty"));

email=input("Enter email ");

if not email:

raise(err("Email cannot be empty"));

if "@" not in email or "." not in email:

raise(err("Invalid email address"));

age=int(input("Enter age "));

if age < 18:

raise(err("Age cannot be less than 18"));

print("The entered information is valid ");

print("Name = ",name);

print("Age = ",age);

print("Email = ",email);

except err as e:

print(e.get\_error());

Output:

