Introduction to Python Programming Lab (BPLCK105B/205B)

I/II Semester

(common to all branches)

LAB MANUAL

MIRZA ARIF BAIG
Asst Professor Dept of CSE
K.C.T ENGINEERING COLLEGE
GULBARGA

Contents

Setting up the Python Environment **iii**

1 Question 1 **1**

1.1 Student Details 1

1.2 Senior Citizen Check 2

2 Question 2 **4**

2.1 Fibonacci Sequence 4

2.2 Binomial Coefficient 5

3 Mean, Variance and Standard Deviation **6**

4 Digit Frequency **8**

5 Word Frequency **9**

6 Sort File Contents **11**

7 Backup Directory into Zip **13**

8 Assertions and Exceptions Demo **14**

9 Complex Class Demo **16**

10 Student Class Demo **19**

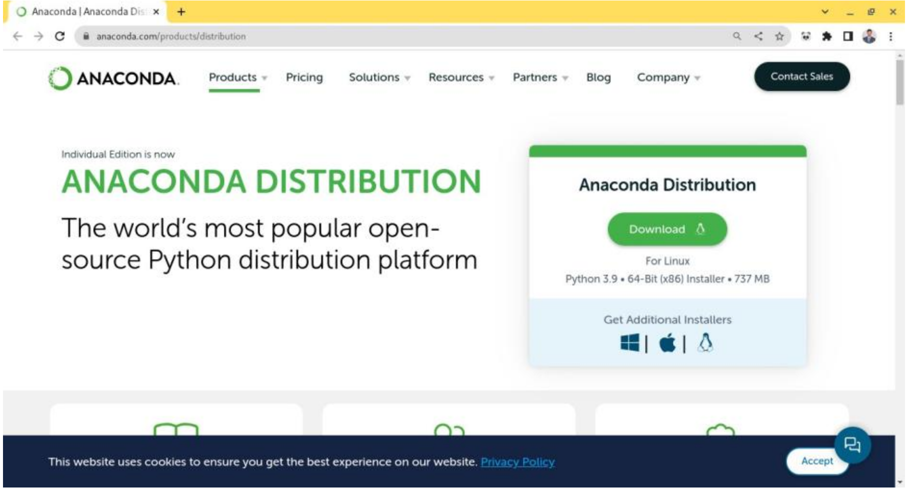
Setting up Python Programming Environment

Anaconda Distribution

Anaconda Distribution equips individuals to easily search and install thousands of Python/R packages and access a vast library of community content and support. Especially with Python being very popular and defacto platform for Data Science. Also it provides good graphical IDEs like **Spyder** and **Jupyter Notebook**.

Download Anaconda Distribution

If you want to install Anaconda Distribution on your system, you can download it from its official webpage.
https://www.anaconda.com/products/distribution
It is available for Linux, Mac, Windows platforms. Download the suitable version based on your operating system.



Individual Edition is now

ANACONDA DISTRIBUTION

The world's most popular open-source Python distribution platform

Anaconda Distribution

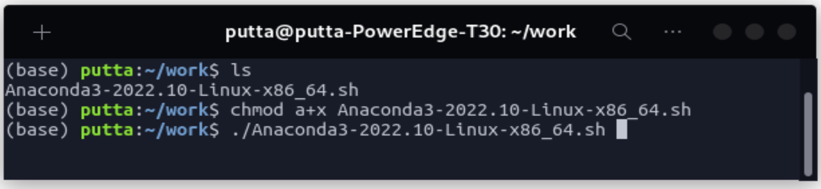
For Linux

Python 3.9 • 64-Bit (x86) Installer • 737 MB

This website uses cookies to ensure you get the best experience on our website. Privacy Policy

Install Anaconda Distribution

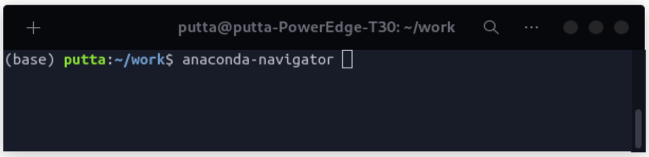
Headover to the folder where it is was downloaded. Open a terminal and navigate to the folder. Run the 3 commands as shown in the diagram below.



```
(base) putta:~/work$ ls
Anaconda3-2022.10-Linux-x86_64.sh
(base) putta:~/work$ chmod a+x Anaconda3-2022.10-Linux-x86_64.sh
(base) putta:~/work$ ./Anaconda3-2022.10-Linux-x86_64.sh
```

The installer proceeds to setup the Anaconda distribution. Choose to perform the default action yes whenever prompted.

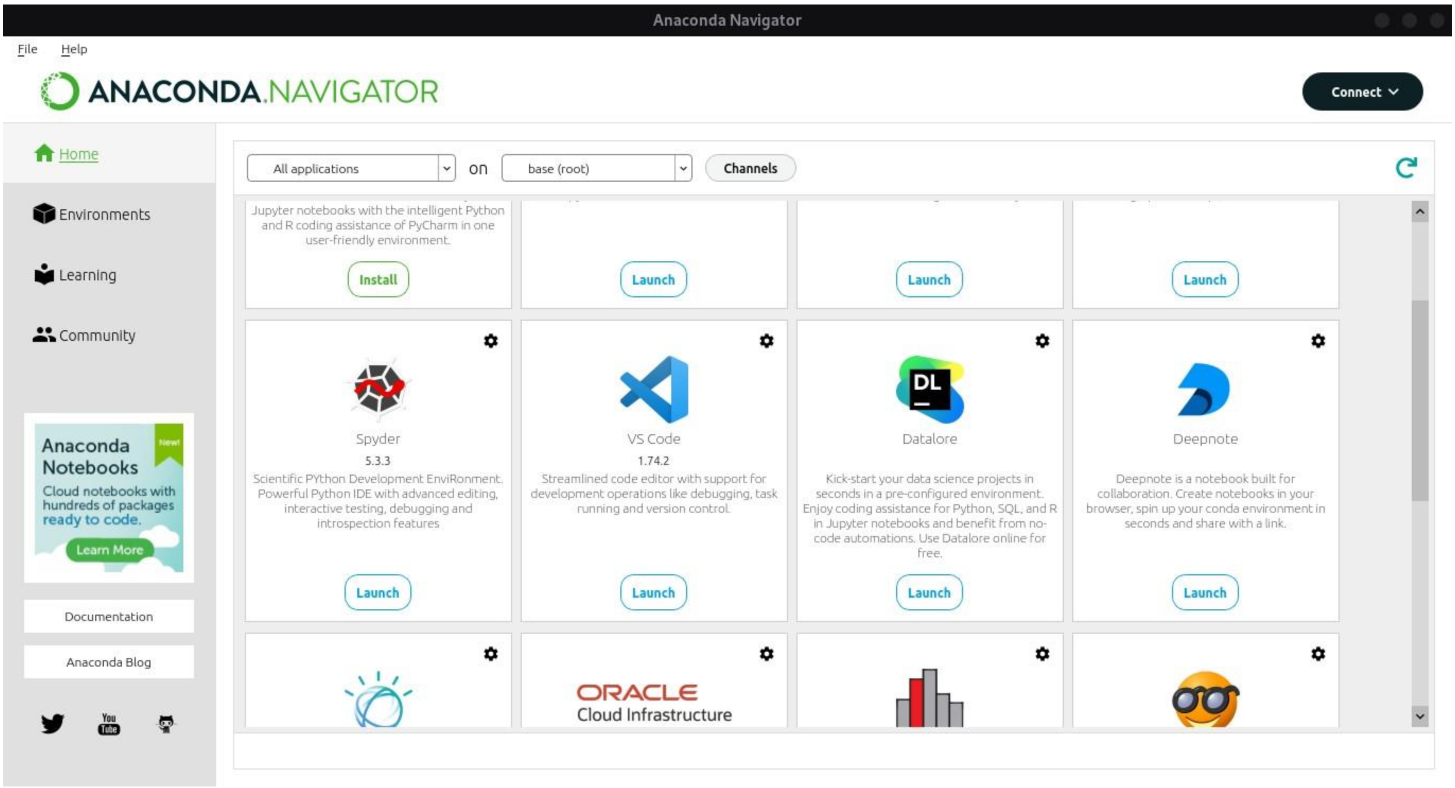
Launch Anaconda Distribution



```
(base) putta:~/work$ anaconda-navigator
```

Once installed open a terminal and run the following command to launch Anaconda Navigator.

Here in Anaconda Navigator choose Spyder IDE. Click on the launch button below it to start the IDE.



Jupyter notebooks with the intelligent Python and R coding assistance of PyCharm in one user-friendly environment.

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

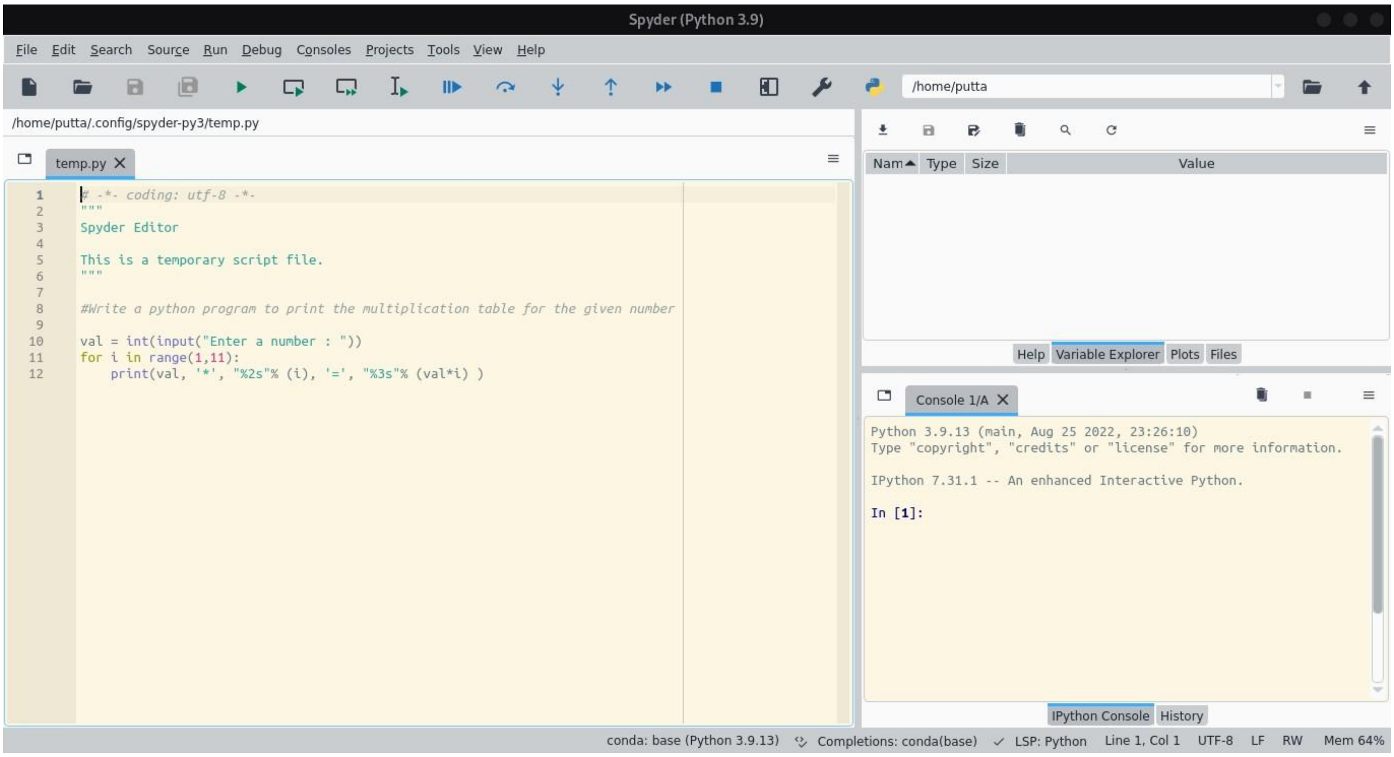
Streamlined code editor with support for development operations like debugging, task running and version control.

Kick-start your data science projects in seconds in a pre-configured environment. Enjoy coding assistance for Python, SQL, and R in Jupyter notebooks and benefit from no-code automations. Use Datalore online for free.

Deepnote is a notebook built for collaboration. Create notebooks in your browser, spin up your conda environment in seconds and share with a link.

Spyder IDE

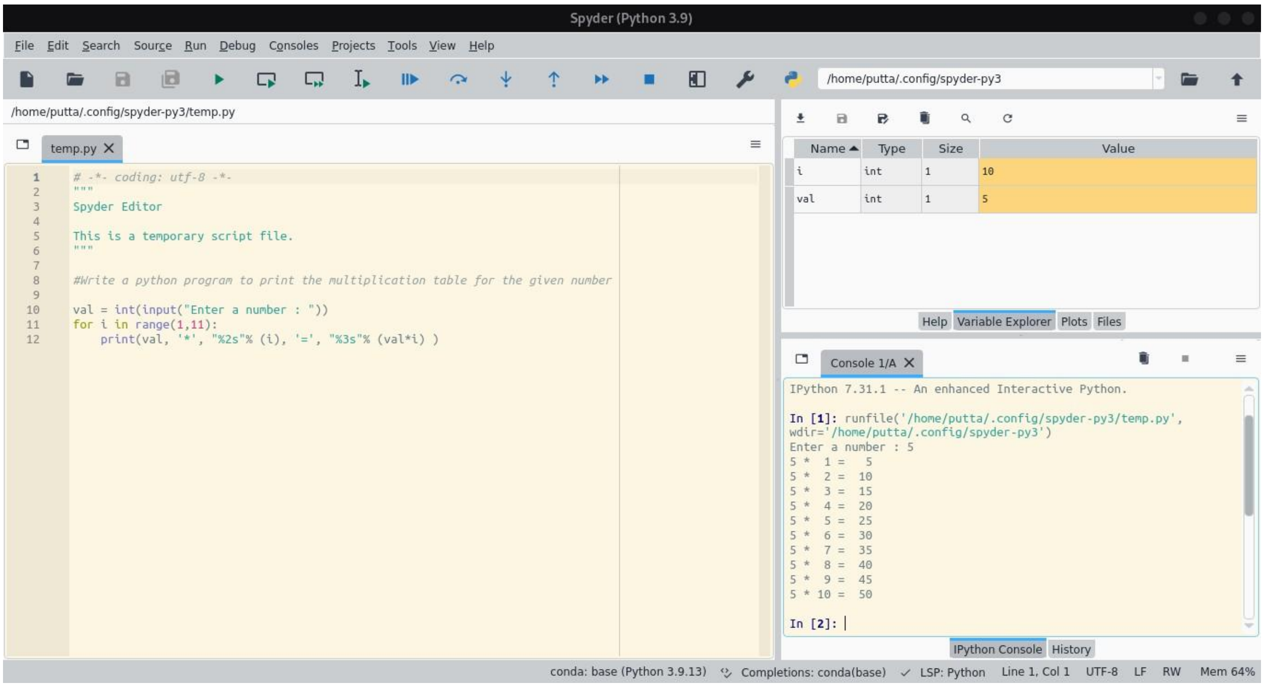
Type a Python program and save it with .py extension. A sample program is shown below.



```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""

#Write a python program to print the multiplication table for the given number

val = int(input("Enter a number : "))
for i in range(1,11):
    print(val, '*', "%2s"% (i), '=', "%3s"% (val*i) )
```

```
Python 3.9.13 (main, Aug 25 2022, 23:26:10)
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]:
```

Then click on Run button (green triangle button) to execute the program. If the program has any errors they are shown on console on the bottom right section. Otherwise the program starts executing and output is shown in the console sub window. A sample is shown below.



Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
/home/putta/.config/spyder-py3
/home/putta/.config/spyder-py3/temp.py
temp.py
Help Variable Explorer Plots Files
Console 1/A
IPython Console History

```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""

#Write a python program to print the multiplication table for the given number

val = int(input("Enter a number : "))
for i in range(1,11):
    print(val, '*', "%2s"% (i), '=', "%3s"% (val*i) )
```

Name ▲	Type	Size	Value
i	int	1	10
val	int	1	5

```
IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runfile('/home/putta/.config/spyder-py3/temp.py',
wdir='/home/putta/.config/spyder-py3')
Enter a number : 5
5 *  1 =   5
5 *  2 =  10
5 *  3 =  15
5 *  4 =  20
5 *  5 =  25
5 *  6 =  30
5 *  7 =  35
5 *  8 =  40
5 *  9 =  45
5 * 10 =  50

In [2]: 
```

conda: base (Python 3.9.13) Completions: conda(base) ✓ LSP: Python Line 1, Col 1 UTF-8 LF RW Mem 64%

Chapter 1

Question 1

1.1 Student Details

Question

Develop a program to read the student details like Name, USN, and Marks in three subjects. Display the student details, total marks and percentage with suitable messages.

Python Code

```python


stName = input("Enter the name of the student : ")
stUSN = input("Enter the USN of the student : ")
stMarks1 = int(input("Enter marks in Subject 1 : "))
stMarks2 = int(input("Enter marks in Subject 2 : "))
stMarks3 = int(input("Enter marks in Subject 3 : "))


print("Student Details\n=========================")
print("%12s"%("Name :"), stName)
print("%12s"%("USN :"), stUSN)
print("%12s"%("Marks 1 :"), stMarks1)
print("%12s"%("Marks 2 :"), stMarks2)
print("%12s"%("Marks 3 :"), stMarks3)
print("%12s"%("Total :"), stMarks1+stMarks2+stMarks3)
print("%12s"%("Percent :"), "%.2f"%((stMarks1+stMarks2+stMarks3)/3))
print("=========================")
```

Listing 1.1: 01AStudDetails

Output

======================================

```


Enter the name of the student : SYED
Enter the USN of the student : 3KC22EC002
Enter marks in Subject 1 : 87
Enter marks in Subject 2 : 78
Enter marks in Subject 3 : 65
Student Details
=========================
      Name : SYED
       USN : 3KC22EC002
   Marks 1 : 87
   Marks 2 : 78
   Marks 3 : 65
     Total : 230
   Percent : 76.67
=========================
```

Listing 1.2: OUTPUT

1.2 Senior Citizen Check

Question

Develop a program to read the name and year of birth of a person. Display whether the person is a senior citizen or not.

Python Code

```python
from datetime import date


perName = input("Enter the name of the person : ")
perDOB = int(input("Enter his year of birth : "))

curYear = date.today().year
perAge = curYear - perDOB

if (perAge > 60):
    print(perName, "aged", perAge, "years is a Senior Citizen.")
else:
    print(perName, "aged", perAge, "years is not a Senior Citizen.")
```

Listing 1.3: 01BChkSnrCitzn.PY

Output

```
=====================================


Enter the name of the person : Akbar Khan
Enter his year of birth : 1978
Akbar Khan aged 44 years is not a Senior Citizen.



Enter the name of the person : UMME
Enter his year of birth : 1957
UMME Best aged 65 years is a Senior Citizen.
```

Listing 1.4: OUTPUT

Chapter 2

Question 2

2.1 Fibonacci Sequence

Question

Develop a program to generate Fibonacci sequence of length (N). Read N from the console.

Python Code

```python
"""

num = int(input("Enter the Fibonacci sequence length to be generated : "))

firstTerm = 0
secondTerm = 1
print("The Fibonacci series with", num, "terms is :")
print(firstTerm, secondTerm, end=" ")
for i in range(2,num):
    curTerm = firstTerm + secondTerm
    print(curTerm, end=" ")
    firstTerm = secondTerm
    secondTerm = curTerm
print()
```

Listing 2.1: 02AFibonacci.py

Output

```


Enter the Fibonacci sequence length to be generated: 8
The Fibonacci series with 8 terms is:
 0 1 1 2 3 5 8 13


```

```

Enter the Fibonacci sequence length to be generated : 5
The Fibonacci series with 5 terms is :
0 1 1 2 3
```

Listing 2.2: OUTPUT

2.2 Binomial Coefficient

Question

Write a function to calculate factorial of a number. Develop a program to compute binomial coefficient (Given N and R).

Python Code

```python
"""

def fact(num):
    if num == 0:
        return 1
    else:
        return num * fact(num-1)
n = int(input("Enter the value of N : "))
r = int(input("Enter the value of R (R cannot be negative or greater than N): "))
nCr = fact(n)/(fact(r)*fact(n-r))
print(n,′C′,r," = ","%d"%nCr,sep="")
```

Listing 2.3: 02BFactNCR.py

Output

```

Enter the value of N : 7
Enter the value of R (R cannot be negative or greater than N): 5
7C5 = 21

Enter the value of N : 5
Enter the value of R (R cannot be negative or greater than N): 5
5C5 = 1

Enter the value of N : 3
Enter the value of R (R cannot be negative or greater than N): 1
3C1 = 3

Enter the value of N : 8
Enter the value of R (R cannot be negative or greater than N): 0
8C0 = 1
```

Listing 2.4: OUTPUT

Chapter 3

Mean, Variance and Standard Deviation

Question

Read N numbers from the console and create a list. Develop a program to print mean, variance and standard deviation with suitable messages.

Python Code

```
"""

from math import sqrt

myList = []

num = int(input("Enter the number of elements in your list : "))

for i in range(num):
    val = int(input("Enter the element : "))
    myList.append(val)

print('The length of list1 is', len(myList))

print('List Contents', myList)


total = 0
for elem in myList:
    total += elem

mean = total / num

total = 0
for elem in myList:
    total += (elem - mean) * (elem - mean)

```

```
variance = total / num

stdDev = sqrt(variance)

print("Mean =", mean)
print("Variance =", variance)
print("Standard Deviation =", "%.2f"%stdDev)
```

Listing 3.1: 03MeanVarSD.py

Output

```


Enter the number of elements in your list : 5
Enter the element : 45
Enter the element : 34
Enter the element : 86
Enter the element : 92
Enter the element : 35

The length of list1 is 5
List Contents [45, 34, 86, 92, 35]

Mean = 58.4
Variance = 642.64
Standard Deviation = 25.35
```

Listing 3.2: OUTPUT

Chapter 4

Digit Frequency

Question

Read a multi-digit number (as chars) from the console. Develop a program to print the frequency of each digit with suitable message.

Python Code

```python

"""
num = input("Enter a number : ")
print("The number entered is :", num)
uniqDig = set(num)
for elem in uniqDig:
    print(elem, "occurs", num.count(elem), "times")
```

Listing 4.1: 04DigitFreq.py

Output

```

Enter a number : 234939
The number entered is : 234939
4 occurs 1 times
9 occurs 2 times
3 occurs 2 times
2 occurs 1 times

Enter a number : 7843338
The number entered is : 7843338
7 occurs 1 times
4 occurs 1 times
3 occurs 3 times
8 occurs 2 times
```

Listing 4.2: OUTPUT

Chapter 5

Word Frequency

Question

Develop a program to print 10 most frequently appearing words in a text file. [Hint: Use dictionary with distinct words and their frequency of occurrences. Sort the dictionary in the reverse order of frequency and display dictionary slice of first 10 items]

Python Code

```
"""
import sys
import string
import os.path

fname = input("Enter the filename : ")      #sample file text.txt also provided

if not os.path.isfile(fname):
    print("File", fname, "doesn’t exists")
    sys.exit(0)

infile = open(fname, "r")

filecontents = ""

for line in infile:
    for ch in line:
        if ch not in string.punctuation:
            filecontents = filecontents + ch
        else:
            filecontents = filecontents + ’ ’    #replace punctuations and \n with
   space

wordFreq = {}
wordList = filecontents.split()
```

```

#Calculate word Frequency

for word in wordList:
    if word not in wordFreq.keys():
        wordFreq[word] = 1
    else:
        wordFreq[word] += 1

#Sort Dictionary based on values in descending order
sortedWordFreq = sorted(wordFreq.items(), key=lambda x:x[1], reverse=True )

#Display 10 most frequently appearing words with their count

print("\n===================================================")
print("10 most frequently appearing words with their count")
print("===================================================")
for i in range(10):
    print(sortedWordFreq[i][0], "occurs", sortedWordFreq[i][1], "times")
```

Listing 5.1: 05WordFreq.py

Output

```


Enter the filename : text.txt

===================================================
10 most frequently appearing words with their count
===================================================
the occurs 45 times
of occurs 24 times
party occurs 12 times
part occurs 12 times
a occurs 9 times
and occurs 8 times
second occurs 7 times
to occurs 6 times
shall occurs 6 times
first occurs 5 times
```

Listing 5.2: OUTPUT

Chapter 6

Sort File Contents

Question

Develop a program to sort the contents of a text file and write the sorted contents into a separate text file. [Hint: Use string methods strip(), len(), list methods sort(), append(), and file methods open(), readlines(), and write()].

Python Code

```
"""

import os.path
import sys

fname = input("Enter the filename whose contents are to be sorted : ")      #sample file unsorted.txt also provided

if not os.path.isfile(fname):
    print("File", fname, "doesn't exists")
    sys.exit(0)

infile = open(fname, "r")

myList = infile.readlines()
# print(myList)

#Remove trailing \n characters
lineList = []
for line in myList:
    lineList.append(line.strip())

lineList.sort()

```

```
#Write sorted contents to new file sorted.txt

outfile = open("sorted.txt","w")


for line in lineList:
     outfile.write(line + "\n")

infile.close()  # Close the input file
outfile.close() # Close the output file


if os.path.isfile("sorted.txt"):
     print("\nFile containing sorted content sorted.txt created successfully")
     print("sorted.txt contains", len(lineList), "lines")
     print("Contents of sorted.txt")
     print("=======================================================================")
     rdFile = open("sorted.txt","r")
     for line in rdFile:
         print(line, end="")
```

Listing 6.1: 06SortLinesFile.py

Output

```

Enter the filename whose contents are to be sorted : unsorted.txt

File containing sorted content sorted.txt created successfully
sorted.txt contains 15 lines
Contents of sorted.txt
=================================================================
A deep C diva.
All the troubles you have will pass away very quickly.
Beware of a tall black man with one blond shoe.
Don't read everything you believe.
Exercise caution in your daily affairs.
He changes the domain.
How does a hacker fix a function which doesn't work for all of the elements in its domain?
Lay on, MacDuff, and curs'd be him who first cries, "Hold, enough!".
People are beginning to notice you.  Try dressing before you leave the house.
The surest protection against temptation is cowardice.
To be or not to be.
Tuesday is the Wednesday of the rest of your life.
What is the square root of 4b^2?
You display the wonderful traits of charm and courtesy.
You may be recognized soon.  Hide.
```

Listing 6.2: OUTPUT

Chapter 7

Backup Directory into Zip

Question

Develop a program to backing Up a given Folder (Folder in a current working directory) into a ZIP File by using relevant modules and suitable methods.

Python Code

```

"""
import os
import sys
 import pathlib
 import zipfile

dirName = input("Enter Directory name that you want to backup : ")
if not os.path.isdir(dirName):
    print("Directory", dirName, "doesn't exists")
    sys.exit(0)
curDirectory = pathlib.Path(dirName)
with zipfile.ZipFile("myZip.zip", mode="w") as archive:
    for file_path in curDirectory.rglob("*"):
        archive.write(file_path, arcname=file_path.relative_to(curDirectory))
if os.path.isfile("myZip.zip"):
    print("Archive", "myZip.zip", "created successfully")
else:
    print("Error in creating zip archive")
```

Listing 7.1: 07BackFolderZip.py

Output

```

Enter Directory name that you want to backup : zipDemo
Archive myZip.zip created successfully
```

Listing 7.2: OUTPUT

Chapter 8

Assertions and Exceptions Demo

Question

Write a function named DivExp which takes TWO parameters a, b and returns a value c (c=a/b). Write suitable assertion for a greater than 0 in function DivExp and raise an exception for when b=0. Develop a suitable program which reads two values from the console and calls a function DivExp.

Python Code

```python
"""
import sys

def DivExp(a,b):
    assert a>0, "a should be greater than 0"
    try:
        c = a/b
    except ZeroDivisionError:
        print("Value of b cannot be zero")
        sys.exit(0)
    else:
        return c

val1 = int(input("Enter a value for a : "))
val2 = int(input("Enter a value for b : "))

val3 = DivExp(val1, val2)

print(val1, "/", val2, "=", val3)
```

Listing 8.1: 08AssertExceptDemo.py

Output

```
Enter a value for a : 7
Enter a value for b : 6
7 / 6 = 1.1666666666666667


Enter a value for a : 0
Enter a value for b : 5
AssertionError: a should be greater than 0


Enter a value for a : -3
Enter a value for b : 6
AssertionError: a should be greater than 0


Enter a value for a : 6
Enter a value for b : 0
Value of b cannot be zero
```

Listing 8.2: OUTPUT

Chapter 9

Complex Class Demo

Question

Define a function which takes 2 objects representing complex numbers and returns new complex number with a addition of two complex numbers. Define a suitable class Complex to represent the complex number. Develop a program to read N (N greater than 2) complex numbers and to compute the addition of N complex numbers.

Python Code

```
"""

class Complex:
    def __init__(self, realp = 0, imagp=0):
        self.realp = realp
        self.imagp = imagp

    def setComplex(self, realp, imagp):
        self.realp = realp
        self.imagp = imagp

    def readComplex(self):
        self.realp = int(input("Enter the real part : "))
        self.imagp = int(input("Enter the real part : "))

    def showComplex(self):
        print(’(’,self.realp,’)’,’+i’,’(’,self.imagp,’)’,sep="")

    def addComplex(self, c2):
        c3 = Complex()
        c3.realp = self.realp + c2.realp
        c3.imagp = self.imagp + c2.imagp
        return c3

```

```python
def add2Complex(a,b):
    c = a.addComplex(b)
    return c

def main():
    c1 = Complex(3,5)
    c2 = Complex(6,4)

    print("Complex Number 1")
    c1.showComplex()
    print("Complex Number 2")
    c2.showComplex()

    c3 = add2Complex(c1, c2)

    print("Sum of two Complex Numbers")
    c3.showComplex()

#Addition of N (N >=2) complex numbers

    compList = []

    num = int(input("\nEnter the value for N : "))

    for i in range(num):
        print("Object", i+1)
        obj = Complex()
        obj.readComplex()
        compList.append(obj)

    print("\nEntered Complex numbers are : ")
    for obj in compList:
        obj.showComplex()

    sumObj = Complex()
    for obj in compList:
        sumObj = add2Complex(sumObj, obj)


    print("\nSum of N complex numbers is", end = " ")
    sumObj.showComplex()

main()
```

Listing 9.1: 09ComplexClass.py

Output

```
Complex Number 1
(3)+i(5)
Complex Number 2
(6)+i(4)
Sum of two Complex Numbers
(9)+i(9)

Enter the value for N : 5
Object 1
Enter the real part : 1
Enter the real part : 9
Object 2
Enter the real part : 2
Enter the real part : 8
```

```
Object 3
Enter the real part : 3
Enter the real part : 7
Object 4
Enter the real part : 4
Enter the real part : 6
Object 5
Enter the real part : 5
Enter the real part : 5

Entered Complex numbers are :
(1)+i(9)
(2)+i(8)
(3)+i(7)
(4)+i(6)
(5)+i(5)

Sum of N complex numbers is (15)+i(35)
```

Listing 9.2: OUTPUT

Chapter 10

Student Class Demo

Question

Develop a program that uses class Student which prompts the user to enter marks in three subjects and calculates total marks, percentage and displays the score card details. [Hint: Use list to store the marks in three subjects and total marks. Use init method to initialize name, USN and the lists to store marks and total, Use getMarks() method to read marks into the list, and display() method to display the score card details.]

Python Code

```python
"""

class Student:
    def __init__(self, name = "", usn = "", score = [0,0,0,0]):
        self.name = name
        self.usn = usn
        self.score = score

    def getMarks(self):
        self.name = input("Enter student Name : ")
        self.usn = input("Enter student USN : ")
        self.score[0] = int(input("Enter marks in Subject 1 : "))
        self.score[1] = int(input("Enter marks in Subject 2 : "))
        self.score[2] = int(input("Enter marks in Subject 3 : "))
        self.score[3] = self.score[0] + self.score[1] + self.score[2]

    def display(self):
        percentage = self.score[3]/3
        spcstr = "=" * 81
        print(spcstr)
        print("SCORE CARD DETAILS".center(81))
        print(spcstr)
        print("%15s"%("NAME"), "%12s"%("USN"), "%8s"%"MARKS1","%8s"%"MARKS2","%8s"%"MARKS3","%8s"%"TOTAL","%12s"%("PERCENTAGE"))
        print(spcstr)
        print("%15s"%self.name, "%12s"%self.usn, "%8d"%self.score[0],"%8d"%self.score[1],"%8d"%self.score[2],"%8d"%self.score[3],"%12.2f"%percentage)
        print(spcstr)


def main():
    s1 = Student()
    s1.getMarks()
    s1.display()

main()
```

Listing 10.1: 10StudentClass.py

Output

```

Enter student Name : SYED
Enter student USN : 3KC22EC002
Enter marks in Subject 1 : 87
Enter marks in Subject 2 : 79
Enter marks in Subject 3 : 92
=================================================================================
                                SCORE CARD DETAILS
=================================================================================
           NAME          USN   MARKS1   MARKS2   MARKS3    TOTAL   PERCENTAGE
=================================================================================
       SYED       3KC22EC002       87       79       92      258          86.00
=================================================================================
```

Listing 10.2: OUTPUT