Question Set (Text File)

Set-1

Q. 1, Write a function display_oddLines() to display odd number of lines from text file - friends.txt

Ans:
```
def display_oddLines():
    f = open("friends.txt", ___)          #statement 1
    L = f.____________                    #statement 2
    n = ____________                      #statement 3
    print("Displaying odd number of lines:")
    for i in range(1,n, 2):
        print(____)                       #statement 4
                                          #statement 5
    ____________

display_oddlines()
```

OR, Q. 2, Write a program in python to show word with maximum length from a text file - "Demo.txt"

Ans:
```
def display_MaxWord():
    file= open("Demo.txt",___)            #statement 1
    str = file.____________               #statement 2
    WL = ____________                     #statement 3
    max = len(WL[0])
    max_word = WL[0]
    w = "" / w = 0
    for w in WL:
        if ____________                   #statement 4
            max = len(W)
            max_word = w
                                          #statement 5
    ____________
    print("Maximum length word :", max_word)
    print("The length of max word :", max)

display_MaxWord()
```

MCQ Questions – Question Set (Text File)

set-1

MCQ:

Q.1. Fill in the blank and write mode of opening file in statement 1?

(a) a (b) ab (c) w (d) r

(ii) fill in the blank in statement 2 to read data as a line of line / line by line from file

(a) F.readline() , (b) F.Readlines() , (c) f.readlines()?

(d) readlines()

(iii) fill in the blank in statement 3 which calculates number of lines from file

(a) len(L) (b) Len(L) (c) Len(line) (d) len(L)

(iv) fill in the blank in statement 4 which displays odd lines from file

(a) print(line) , (b) print(L) , (c) print(L[i]),

(d) print(file)

(v) fill in the blank in statement 5 to close the file

(a) F.close() (b) f.close() (c) f.Close() (d) close()

MCQ: set-1

Or, Q.2, fill in the blank in statement 1 and write mode of

(i) opening file

(a) w (b) a (c) rb (d) r (e) rt

(ii) fill in blank in statement 2 to read data from the file

(a) File.Read(), (b) File.read(), (c) file.read(),

(d) readlines()

(iii) fill in blank in statement 3 to read data word by word

(a) str.split(), (b) str.split(), (c) Str.split()

(d) split.word()

(iv) fill in blank in statement 4 which select the word of maximum length,

(a) len(w)>=max, (b) len(w)<=Max , (c) len(w)>max,

(d) len()>max

(v) file in blank in statement 5 the file?

(a) file.close(), (b) file.Close(), (c) File.close(), end()

set-2

Q.1. Write a function display() in python that opens a file "DIARY.TXT" and display those lines which starts with the alphabet 'p' or 'P'

Ans:

```
def display():
    file= open("DIARY.TXT", ____)          # statement 1
    L =file.____                           # statement 2
    line = "" / line=0
    for line in L:
        if line[0]=="p" or ____:           # statement 3
            print ____                     # statement 4
                                           # statement 5
____

display()
```

Or, Q.2. Write a function in Python that counts the number of "Me" or "My" words present in text file "STORY.TXT"
output : The Number of Me/My words : 4

Ans.

```
def CountMeMy():
    file= open("STORY.TXT", ____)          # statement 1
    count = 0
    str =file.____                         # statement 2
    L = ____                               # statement 3
    w = "" / w=0
    for w in L:
        if ____ or ____:                   # statement 4
            count += 1
                                           # statement 5
____
    print("The Number of Me/My words :", count)

CountMeMy()
```

Set-2

MCQ:

Q.1 (i) fill in the blank in statement1 to write the mode to open file

(a) ab, (b) w, (c) r, (d) a

(ii) fill in the blank in statement2 to read lines of file/ to read lines of file as list,

(a) file.readline(), (b) file.readlines(), (c) readlines(), (d) file.Readlines()

(iii) fill in the blank in statement 3 to display the lines which is started with P/p - letter,

(a) line[0] == 'P', (b) line[0] >= 'P', (c) line[0] = 'P', (d) line[0] == 'p'

(iv) fill in the blank in statement 4 to print line of file

(a) print(Line), (b) print(line), (c) Print(line) (d) Print(Line)

(v) fill in the blank in statement5 to close the file

(a) file.close(), (b) file.close(), (c) close(), (d) end()

Or, Q.2 MCQ: Set-2

(i) fill in the blank in statement1 to write mode of opening the file

(a) a (b) ab (c) w (d) r

(ii) fill in the blank in statement 2 to read the data from the file

(a) file.Read(), (b) file.read(), (c) file.Read(), (d) read.word()

(iii) fill in the blank in statement 3 to read data word by word,

(a) Str. (b) str.Split(), (c) str.split(), (d) split.read()

(iv) fill in the blank in statement 4 which will count total number of word 'He' or 'Hy'

(a) w == 'He', w == 'Hy' (b) w == 'ine', w == 'Hy', (c) w == 'He', w == 'Hy' (d) w == 'He', w == 'Hy'

MCQ:

(v) Fill in the blanks for statement 5 to close the file (3)

(a) file.close() , (b) file.close(), (c) file.Close(), (d) close(),

Set-3

Q.1, Same as Set-2, Q-OR, Q.2,

Or

Q.2, Write a function AMCount() in Python, which should read each character of a text file STORY.TXT, should count and display the occurrence of alphabets A and M (including small cases a and m too)

output should be: A or a: 4
M or m: 2

Ans:
```
def AMCount():
    file=open("STORY.TXT", ____)          # statement 1
    count1 = count2 = 0
    str = file.____                       # statement 2
    ch="" / ch=0
    for ch in str:
        if ch == 'A' or ____              # statement 3
            count1 = Count1 + 1
        if ____ or ch == 'm':             # statement 4
            count2 = count2 + 1

                                          # statement 5
    ____
    print("A or a:", count1)
    print("M or m:", count2)
AMCount()
```

Set-4

Q.1, Write a program to count the number of upper-case alphabet present in text file "PYTHON.TXT"

Ans:
```
def countUpper():
    ch="" / ch=0
    file=open("PYTHON.TXT", ____)         # statement 1
    count = 0                             # statement 2
    str = file.____
    for ch in str:
        if ____:                          # statement 3
            count += 1
                                          # statement 4
    ____
    print("Number of upper-case alphabet:", count)
countUpper()
```

Set-3

MCQ:

Q.1, (i) fill in the blank in statement 1 which specify mode of opening file

(a) a (b) w (c) rb (d) r

(ii) fill in the blank in statement 2 to read data from file

(a) file.Read(), (b) file.read(), (c) file.readlines(), (d) file.readlines(), (e) readlines(), (f) readline, (g) read()

(iii) fill in the blank in statement 3 which counts the number of character 'A' in file

(a) ch = 'a', (b) ch == 'A', (c) ch >= 'a' (d) ch == 'a'

(iv) fill in the blank in statement 4 which counts the number of character 'M' in file

(a) ch = 'M' (b) ch >= 'M', (c) ch == 'M', (d) cH == 'M'

(v) fill in the blank in statement 5 to close the file

(a) file.close(), (b) file.close(), (c) end(), (d) close()

Set-4

MCQ:

Q.1, (i) Write mode of opening file in statement 1?

(a) a (b) r (c) w (d) ab (e) rb (f) wb (g) rt

(ii) fill in the blank in statement 2 to read data from the file

(a) file.Read(), (b) File.read(), (c) file.read, (d) read.word()

(iii) fill in the blank in statement 3 which will create a condition to count

(a) ch.Isupper() == true (b) ch.isupper() == False

(c) ch.isupper() = True, (d) ch.isupper()

(e) ch.islower() == False

(iv) Write in blank in statement 4 to close file

(a) file.end(), (b) file.Close() (c) File.close,

(d) file.close()

OR

Set-4

(4)

Q.2, Write a function which takes an input and output file. It copies all lines which starts with vowels from input file to output file,

Ans:

```
def CopyLineVowel (infile, outfile):
    fin = open ( infile, ____ )          # statement 1
    fout= open ( outfile, ____ )         # statement 2

    L = fin.____                         # statement 3
    line = "" / line=0
    for line in L:
        if ____________ :                # statement 4
            fout.write (line)

                                         # statement 5
                                         # statement 6

CopyLineVowel ("story.txt", "story2.txt")
```

Set-5

Q.1, Write a function LongLine() that accepts a file name and prints the file's longest line with its length.

Ans:

```
def LongLine (fileName) :        # fileName = STORY.TXT
    file=open (fileName, ____ )  # statement 1
    str = file.____              # statement 2
    max = len (str [0])
    max_line = str [0]
    line = "" / line = 0
    for line in str:
        if ____________ :        # statement 3
            max = len (line)
            max_line = line

                                 # statement 4

    print (" The longest line: ", max_line)
    print (" The length of maximum
                length line: ", max)

LongLine ("STORY.TXT")
```

MCQ Set-4

Q2. (i) Write the mode of opening file in statement 1 ? (4)

(a) ab (b) rt (c) r (d) w (e) wt (f) rb

(ii) Write the mode of opening file in statement 2 ?

(a) w (b) r (c) a (d) wt (e) wb (f) rt

(iii) fill in the blanks in statement 3 which reads lines/ which reads lines as list from the file,

(a) fin.readline (), (b) fin.Readlines (),

(c) fin.readlines (), (d) fout.readlines()

(iv) Write in the blank of statement 5 to close the file – infile

(a) fout.close (), (b) fin.Close (), (c) fin.close (),

(d) fin.Close (), (e) close (), (f) end ()

(vi) fill in the blank in statement 6 to close the file – outfile

(a) fout.Close (), (b) fin.close (), (c) fout.close (),

(d) close (), (e) end ()

(iv) fill in the blank in statement 4 that will display all the line starting with vowel Character.

(a) Line[0] == "AEIOUaeiou", (b) line[0] In "AEIOUaeiou"

(c) line[0] in "AEIOUaeiou" (d) line[0] = "AEIOUaeiou"

Set-5

MCQ:

Q.1. (i) Write mode of opening file in statement 1

(a) r , (b) wb , (c) ab , (d) a , (e) rt , (f) wt , (g) rb

(ii) fill in the blank in statement 2 which will read lines of file / which read lines as list from file

(a) file. Readlines (), (b) File. readlines (),
(c) file. readlines (), (d) fil. readline (),
(e) File. readline

(iii) fill in the blank in statement 3 which will select line of maximum length from file,

(a) Len (line) > max (b) len (Line) >= max
(c) len (Line) >= max (d) len (line) > max

(iv) fill in the blank in statement 4 to close the file

(a) file. end () , (b) file. Close (), (c) file. Close (),
(d) file. close () , (e) close (), (d) end ()

Set-5

Or, (5)

Q.2, Write a function rem_Lower() that accepts two filenames and copies all lines that do not start with a lowercase letter from first file into second.

Ans:

```
def rem_Lower (infile, outfile):
    fin = open (infile, ____)                    # statement 1
    fout = open (outfile, ____)                  # statement 2
    L = fin.____                                 # statement 3
    line = " " / line = 0
    for line in L:
        if line [0] . islower == False:
            fout.____                            # statement 4
            ____                                 # statement 5
            ____                                 # statement 6
rem_Lower ("DIARY.TXT", "DIARY2.TXT")
```

Set-6

Q.1, Write a program to display all the records in a file "python.txt" along with line/record number.

Ans:

```
def DisplayLine()
    f = open ("python.txt", ____)                # statement 1
    L = f.____                                   # statement 2
    for i in range (len(L)):
        print ("Line/Record Number:", i+1,
               ":", ____)                        # statement 3
        ____                                     # statement 4
DisplayLine()
```

Set-5

Or, 8, 2, MCQ: (5)

(i) Write mode of opening the file in statement1 ?

(a) r (b) w (c) a (d) rt (e) ab (f) wb (g) rb

(ii) write mode of opening the second file in statement2 ?

(a) a (b) w (c) r (d) wb (e) ab (f) wt

(iii) Fill in the blanks in statement3 to read lines from file/ to read lines of file as list,

(a) fin.readlines() , (b) fin.Readlines(),
(c) fin.readlines(), (d) fin.readline()
(e) fin.read() , (f) fout.readlines()

(iv) Fill in the blank to write line in second file in statement 4

(a) Fout.Write(line) , (b) fout.write(line),
(c) fin.write(line) , (d) fout.write(Line)

(v) fill in the blank in statement5 to close the first file

(a) fout.close() , (b) fin.close() , (c) fin.Close(),
(d) close() , (e) fout.end() , (f) end()

(vi) fill in the blank in statement 6 to close the second file

(a) fin.close(), (b) fout.Close(), (c) Fout.Close(),
(d) fout.close(), (e) close(), (f) end()

MCQ: Set-6

Q.1, (i) Write mode of opening file in statement 1

(a) r , (b) w , (c) a , (d) wb , (e) rt , (f) ab

(ii) fill in the blank in statement 2 to read lines fro file/ to read lines of file as list

(a) fin.Readlines() , (b) fin. Read lines (),
(c) fin. readlines (), (d) fin. read ()
(e) readlines () , (f) fin. readline ()

(iii) fill in the blank in statement 3 to print odd line from file

(a) L[i] , (b) L[i] , (c) L[1] , (d) L(i)

(iv) fill in the blank in statement 4 to close the file

(a) F. close() , (b) F. close (), (c) f. close (),
(d) f. close (), (e) close(), (f) end ()

Or, Q.2 A text file "PYTHON.TXT" contains alphanumeric text. Write a program that reads this text file and prints only the numbers or digits in the file

Ans:

Method-1:

```
def DisplayDigit():
    f = open("PYTHON.TXT", ____)          # statement 1
    str = f.____                          # statement 2
    wl = ____                             # statement 3
    print("Displaying only numbers/digits of file:")
    w=" "/w=0
    for w in wl:
        for ch in w:
            if ____:                      # statement 4
                print(ch)
                                          # statement 5
DisplayDigit()
```

Method-2:

```
def DisplayDigit():
    f = open("PYTHON.TXT", "r")
    str = f.read()
    print("Displaying only number/digits of file:")
    ch=" "/ch=0
    for ch in str:
        if ch.isdigit() == True:
            print(ch)

    f.close()
DisplayDigit()
```

MCQ: Set-6

Or, 8, 2, method-1 — program —

(i) Write mode of opening file in statement 1

(a) r , (b) a , (c) w , (d) rb, (e) rt , (f) ab, (g) wt

(ii) Fill in the blank in statement 2 to read data from file

(a) file.Read (), (b) File.read (), (c) file.read (),
(d) file.readline (), (e) file.readlines ()

(iii) fill in the blank in statement 3 to read data word by word from file,

(a) Str.Split (), (b) str.Split () , (c) str.split (),
(d) split.word ()

(iv) fill in the blank in statement 4 which will print digit from file

(a) w.digit () == True , (b) w.isdigit ()
(c) w.isdigit = True , (d) w.isdigit () == True
(e) w.isdigit = false

(v) All in the blank in statement 5 to close the file

(a) F.close () , (b) f.Close (), (c) f.close (),
(d) close (), (e) end ()

Set-7

Q.1. Write a function that takes in two text files and copies all lines from first file to second, baring the lines starting with "a".

Ans:

```
def CopyLine (infile, outfile):
    fin = open (infile, ____)                    # statement 1
    fout = open (outfile, ____)                  # statement 2
    L = fin.____                                 # statement 3
    line = "" / line = 0
    for line in L:
        if line[0] == "A" or ____ : #statement 4
            fout.____                            # statement 5
        ____                                     # statement 6
        ____                                     # statement 7

CopyLine ("STORY1.TXT", "STORY2.TXT")
```

or

Q.2, Write a program to count the words "to" and "the" present in a text file "python.txt"

Ans:

```
def CountWord ():
    w = " " / w = 0
    file = open ("python.txt", ____)             # statement 1
    str = file.____                              # statement 2
    wl = ____                                    # statement 3
    c1 = c2 = 0
    for w in wl:
        if ____ or ____ : #statement 4
            c1 = c1 + 1
        if ____ or ____ : #statement 5
            c2 = c2 + 1
        ____                                     # statement 6
    print ("Number of word to/To:", c1)
    print ("Number of word the/The:", c2)

CountWord ()
```

MCQ: set-7

8.1, (i) Write the mode of opening the first file - infile in statement1

(a) a (b) r (c) w (d) rt (e) ab (f) wt (g) wb (h) at

(ii) Write the mode of opening second file - outfile in statement2

(a) w (b) wt (c) a (d) r (e) rt (f) rb (g) wb (h) ab

(iii) fill in the blank in statement3 - which read lines from the first file / - which read lines as list from file

(a) Fin.Readlines(), (b) fin.readlines(),
(c) fin.Readlines(), (d) fin.readline(),
(e) fin.readlines(), (f) fout.readlines()

(iv) fill in the blank in statement4 which displays lines starting with 'a' from file

(a) Line[0] == 'a', (b) line[0] = a,
(c) line[0] == 'a', (d) line[0] == 'A'
(e) line[1] == 'a'

(v) Fill in the blank to write line starting with 'A'/'a' from first - infile to second file - outfile in statement5

(a) Fout.write(line), (b) fout.Write(Line)
(c) fin.write(line), (d) fout.write(line)
(d) write(line)

(vi) fill in the blank in statement6 to close first file - infile

(a) Fout.close(), (b) fout.close(),
(c) fin.Close(), (d) fin.close(),
(e) fin.close(), (f) fin.end(),
(g) close()

(vii) fill in the blank in statement7 to close the second file - outfile

(a) Fout.Close(), (b) fout.close(),
(c) fin.close(), (d) close(), (e) end()

Or, Q.2 MCQ: Set-7 (8)

(i) Write mode at opening file in statement 1

(a) a (b) r (c) rt (d) ab (e) wb (f) at (g) wt

(ii) fill in the blank in statement 2 to read data from file

(a) File.Read(), (b) file.Read(), (c) file.read(),
(d) file.readline(), (e) file.readlines()

(iii) Fill in the blank in statement 3 to read the data word by word

(a) Str.split(), (b) Str.split(), (c) str.Split(),
(d) str.split(), (e) split.read()

(iv) Fill in the blank in statement 4 which will count the number of word 'To'/'to' in the file

(a) W=='To', W=='to', (b) w='To', w='to',
(c) w=='To', w=='to' (d) w=='to', w=='To'
(e) w='To', w=='to'

(v) Fill in the blank in statement 5 which will count the number of word 'The'/'the' in the file

(a) W=='The', W=='to', (b) w='The', w='the'
(c) w>='The', w=='the', (d) w=='The', w=='the',
(e) w=='the', w=='The'

(vi) Fill in the blank in statement 6 to close the file

(a) file.Close(), (b) file.close(), (c) file.Close(),
(d) file.close(), (e) close(), (f) end()

Set - 8

Q.1. Write a function word4() in python that displays 4 letter words present in a text file "myfile.txt"

Ans:
```
def word4():
    file=open("myfile.txt", ____)          # statement 1
    w="" * / w=0
    str = file.____                        # statement 2
    wl = ____                              # statement 3
    for w in wl:
        if ____:                           # statement 4
            print(w)                       # statement 5
____
word4()
```

Q. 2 → Write a python function word3() that displays 3 letter words in a text file "myfile.txt")
↳ same as Q.1.

Q. 3 → Write a program filterWord5() in python that count and displays 5 letter words present in a text file "story.txt"

Ans:
```
def filterWord5():
    f = open("story.txt", ____)            #statement 1
    w = "" / w = 0
    count = 0
    str = f.____                           # statement 2
    wl = ____                              # statement 3
    for w in wl:
        if ____:                           # statement 4
            print(w)
            count +=1                      # statement 5
____
print("Number of 5 letter words:", count)
filterWord5()
```

MCQ: set-8

Q.1 (i) Write mode of opening file in statement 1

(a) r (b) rb (c) a (d) w (e) wt (f) rt (g) at, (h) wb

(ii) fill in the blank in statement 2 which will read data from file

(a) File.Read() , (b) File.read() , (c) file.read(), (d) file.Read() , (e) file.readline(), (f) file.readlines()

(iii) fill in the blank in statement 3 to read data of file word by word

(a) Str.Split() , (b) str.Split() , (c) Str.split(), (d) str.split() , (e) split.word()

(iv) fill in the blank in statement 4 which will display the word having 4 charecters,

(a) len(w) = 4 , (b) Len(w) >= 4 , (c) Len(w) <= 4, (d) Len(w) == 4, (e) Len(w) == 4

(v) fill in the blank in statement 5 to close the file

(a) File.Close() , (b) File.close() , (c) file.Close() , (d) file.close() , (e) close() , (f) file.end()

MCQ: set-8

Or, Q.3, (i) Write mode of opening file in statement 1

(a) w , (b) a , (c) r , (d) wb , (e) at , (f) rt , (g) ab

(ii) fill in the blank in statement 2 to read data from file

(a) F.Read() , (b) F.read() , (c) f.Read() , (d) f.read() , (e) f.readline() , (f) f.readlines()

(iii) fill in the blank in statement 3 to read data word by word

(a) Str.Split() , (b) str.Split() , (c) str.split() , (d) split.word()

(iv) fill in the blank in statement 4 which displays the word having characters

(a) len(W) == 3 , (b) len(W) = 3 , (c) Len(w) == 3, (d) len(w) >= 3 , (e) len() == 3 , (f) Len(w) == 3

(v) fill in the blank in statement 5 to close the file (10)

(a) F. Close (), (b) f. Close (), (c) f. elose (),
(d) f. end (), (e) close (), (f) end ()

Set -10

Q.1. Your teacher has given you a method/function filterWords() in python which read lines from text files NewsLetter.TXT, and displays those words, which are lesser than 4 characters

Ans:

```
def filterWords():
    c=0
    file = open( NewsLetter.TXT , ____ )      #statement1
    line = file.____                          #statement2
    word = ____                               #statement3
    for e in word:
        if ____:                              #statement4
            print(c)                          #statement5
____
filterWords()
```

MCQ: Set-10

Q, 1, (i) Write mode of opening file in statement1.

(a) a (b) ab (c) w (d) r

(ii) fill in the blank in statement2 to read data from the file

(a) File.Read(), (b) file.read(), (c) read.lines(), (d) readlines()

(iii) fill in the blank in statement3 to read data word by word

(a) Line.Split(), (b) Line.split, (c) line.split(), (d) split.word()

(iv) Fill in the blank in statement4 which display the word having lesser than 4 characters

(a) len(c) == 4, (b) len(c) >= 4, (c) len(c) <= 4, (d) len( ) == 4