Module Code & Module Title

CS4051NI Fundamentals of Computing

Assessment Weightage & Type

60% Individual Coursework

Year and Semester

2020-21 Autumn

Student Name: Kishor Shrestha

Group: C6(computing)

London Met ID: 20048913

College ID: NP01CP4S210161

Assignment Due Date: 2021/09/10

Assignment Submission Date: 2021/09/10

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of Contents

Table of Figures

Table of tables

1.0 Introduction

We designed an individual library management system using python. This project is an individual coursework. The project also describes how the program works on various types of input values. To comprehend the program execution flow, remove the extensive coding of the program using the functions on a desirable number of times, the program is separated into distinct modules.

This project is assigned for Python for the first time. Python 3.8 IDLE (Integrated Development and Learning Environment) completes the coding of the project. Python is a high-level programming language with dynamic semantics which is interpreted, objective-oriented. Its high-level integrated data structures combined with the dynamic typing and dynamic binding require the development of rapid applications and the usage of the existing components together in scripting or adhesive language desirable. The simplicity and ease of syntax of Python underlines readability and decreases software ongoing costs. Python offers modules and packages that promote flexibility of program and reuse of code. The Python interpreter and the large standard library are accessible for all major platforms in source or binary form free of charge and can be disseminated free of charge. Python is popular among programmers since it allows them to work more efficiently. The edit-test-debug cycle is extraordinarily rapid because there is no compilation step. Python scripts are simple to debug: a flaw or improper input will never result in a segmentation fault. Instead, whenever the interpreter finds a mistake, it throws an exception. The interpreter prints a stack trace if the application fails to catch exception. Inspection of local and global variables, execution of random expressions, setting breakpoints, moving through the code one line at a time, and so on are all possible with a source level debugger. The debugger is implemented in Python, demonstrating Python's introspective capabilities (Rossum, 2021).

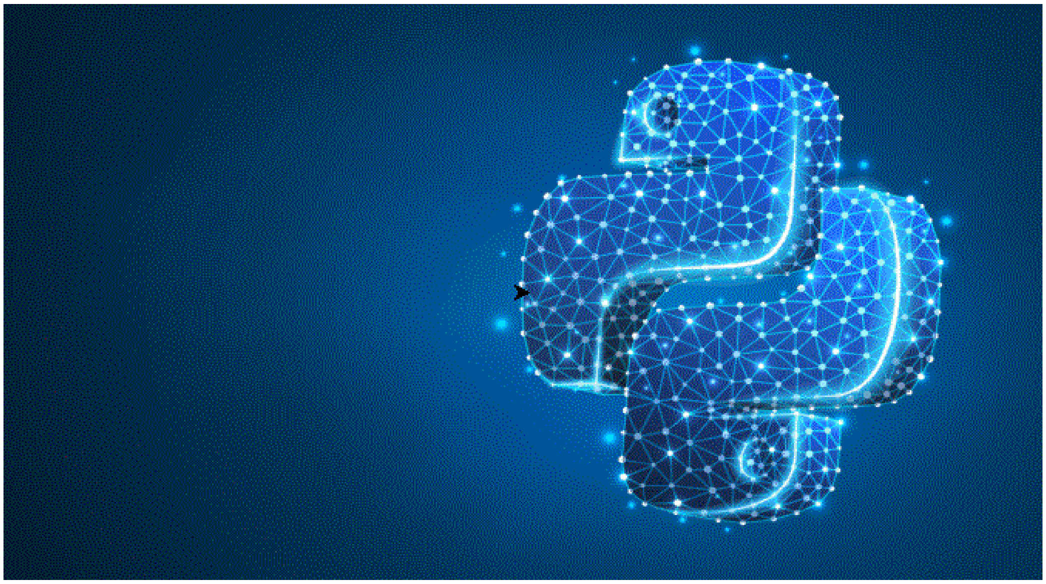


Figure 1: Figure of python program

In this project modified version of library management system is carried out. Each section represents a subsystem consisting of a number of interconnected components. The parties cooperate collaboratively on the acquisition, reprocessing and circulation of library information resources for a library management scheme. The term ‘library management system' is one that librarians and system suppliers most typically use to characterize systems that conduct acquisition, cataloguing, and distribution activities. It has typically supplanted earlier terminology, such as ‘library housekeeping system,' which used to be common in British literature and indicated that this type of system is used to conduct day-to-day transactions in a library. The name change may reflect the fact that these systems also do management reporting, allowing them to enable higher degrees of library administration than transactional subsystems. Library management systems oversee the flow and cataloging of assets and the management of membership. Using library management systems in many businesses, they automate critical housekeeping features. The delivery of data and resources to library clients is expedited. These systems give an online or digital interface which unifies the administrative background, the circulation of libraries and collection of assets. They also offer proprietary portals which enable library employers to obtain or reserve library resources quickly. An example of an information system is a library management system. A computerized information system is a system that displays items within a physical system, e.g. information resources in a library collection (sciencedirect.com, 2021).

1.1 Goals and objectives

A library management system was the major purpose of the project. To explain all the functionality, we had to create a flowchart which illustrated our bit adder program visually. Some of the following points are listed in the use of python and data structures

Figure 2: Figure of goals and objective of python

Pseudo code is also produced to make the code easier to grasp and its features

- Testing component for the functioning and desired outputs is performed
- Tracking book, librarian information, problems etc.
- Added and updated records are upgraded to ensure proper data management
- Management of information of issues
- Monitoring the information and transaction of the issues
- Management of information of books
- Systematic use of python and its components.

2.0 Discussion and analysis

2.1 Algorithm

An algorithm is a sequence of instructions for carrying out a certain activity. This could be a simple process, like two numbers being multiplied or a sophisticated operation, such playing a compressed video clip. Search engines will offer the most relevant results for certain queries using proprietary algorithms from their search index. Algorithms are commonly developed as functions in computer programming. These features can be referenced by a bigger program as mini programs. For example, a program for image viewing can include a library of functions using an individual algorithm to make different image file formats. A picture editing program may include algorithms for processing picture data. E.g., cutting, resizing, sharpness, blurring, red-eye reduction and color improvement are examples for image processing. Programmers are therefore often looking for the most effective algorithms. Developers may make sure their programs execute as quickly as possible and consume minimal system resources by employing highly efficient algorithms (christianson, 2021).

Algorithm steps:

Step 1: Start

Step 2: A stock.txt file will be created and will store information about books inside library.

Step 3: Four options are executed asking user to choose. They are:

To display library books: Register 1

To Borrow library book: Register 2

To return library book: Register 3

To exit: Register 4

Step 4: If integer 1 is entered inside the code all the books stored in stock.txt of library management System will be shown.

Step 5: If int 2 is input, the user is directed to the borrowing book part.

Step 6: It will request the user to insert his first and last names in the appropriate fields.

It will show the library's books as well as ask the user to select from them.

Step 7: If the book is accessible, every quantity of the book, as well as information about the borrowed books' time and date, is logged and presented, along with the fee of the books. A txt file with the user's profile is also generated.

Step 8: The customer will be asked if they wish to borrow more books. If you want to borrow additional N, hit Y. If you don't want to borrow anything more, press N. If Y is selected, then library's books will be presented once more. Otherwise, N will show the first part.

Step 9: If 2 is submitted, it will lead you to the part where you can return.

Step 10: If it is before the return time, this will establish the entire number of books obtained by users with the Fine.

Step 11: The respondent is required to input the first name of the borrower who has previously borrowed in the returning area. The name of the borrower is matched to the name in the borrower's record that will be received. If the two names are identical.

Step 12: It will show all of the books that the user has borrowed, as well as the information and price of the books that have been recorded. Otherwise, the user would be warned that the borrower's name submitted is wrong.

Step 13: If the borrower's name is true, the system would then question if the book was handed on time or not. It will show up. Is the book getting returned after the due date? If you want to say yes, click y, and if you want to say no, click N. If yes, it will calculate the number of days passed the deadline and apply the fine to the number of days.

Step 14: The user will be directed to the first panel, which welcomes them to the library administration system.

Step 15: If the user inputs 4, the library management system is closed.

Step 16: End.

3.0 Pseudocode

Pseudocode is an unofficial method to describe programming that requires no rigorous syntax of the programming language and technology. It is used to create a contour or an approximate draft of a program. Pseudocode illustrates the flow of a program, but does not provide any information. System designers produce pseudo code to help programmers comprehend and align code correctly to the needs of a software project. The original programming language is not the pseudocode. It can therefore not be compiled into an executable. It employs short phrases to write code for programs before it is transformed into a particular programming language. This is designed to check flow faults of the highest level and to understand the data flows of the program the final program will employ. It obviously saves time when conceptual problems have previously been fixed during practical programming. First of all, a program description and functionality are collected, followed by pseudo code for creating declarations to obtain the desired program results (mandavia, 2021).

3.1 Pseudo code of borrowing book:

```
Function borrow book()
Input Enter the first name of the borrower.
    Print "Please enter letters ranging from A to Z
Input Enter the last name of the borrower.
    Print "Please enter letters ranging from A to Z"
     create borrow.txt file and store the borrower details in that file
 else
    print "invalid name"
END
If
Success = false
    Print "Please choose one of the following options."

 IF
Initialize list split quantity
    Print "Book is available here"
```

```
While
Loop= true
    Input “"Do you want to borrow more books? However you cannot borrow same book twice. Press y for yes and n for no.”
        Print “Please choose an option from the list below..”
Else
Loop= false
Break
Elif
Choice upper()=N
    Print “Thank you for borrowing books from our library.”


Else:
    Print “Please choose an option as instructed”
Else:
    Print “Book is not available here”


Else:
    Print “Book is not available here”
If success= false
Expect Index error:
 Print “Please choose any book according to their given number.”
```

```
Expect value error:

    Print “Please choose the number as suggested.”

END
```

3.2 Pseudo code of returning books:

```
Function return book():

Input “Enter the name of borrower here:”

Try:

Open with (a and R) as f:

Lines= readlines()

Lines= a.strip ("$") for a lines

With open (a “r”) as far:

Data= f.read()

Print data

Expect:

    Print “The borrower name entered is incorrect.”

End IF

    Display the borrow.text file of given name.

Return txt file and update the stock file.

While

Expired=true
```

```
Add fine
    Fine=2*days
End while
    Add total fine and cost
Display text file with the total cost after returning
```

3.3 Pseudo code of Listsplit:

```
Function ListSplit():
Global book name
Global author name
Global quantity
Global cost
For I in range (lenlines)
    Int = 0
If
    Int=o
    Book will be append
Elif
    Int=1
    Author name will append
```

```
Elif
    Int=2
    Quantity will append
Elif
    Int=3
    Cost will append
END
```

3.4 Pseudo code of dateandtime:

```
Function get date()
Import datetime
    Now= date time.datetime.now
Print : date and time"
    Return str(now) and date:


Function gettime ():
Import date time
Now= date time.datetime.now
    Print : date and time"
Return str(now) and date:
```

3.5 Pseudo code of Main :

```
Function start():
While = true
    Print "Hello and thank you for visiting the library management system."
    Print "Register 1. to display"
    Print "Register 2. to borrow a book"
    Print "Register3. to return a book"
    Print "Register4. to exit"
Try:
    a= Enter the correct choice  from 1 to 4:
if a =1
    books, text file will append
if a =2
    list of borrow book will opened
if a= 3
    list of return book will opened
if a= 4
    print "Thank you for making use of the library management system."
else:
    print "Enter the correct choice  from 1 to 4"
```

```
expect value error:
        print "Please enter the information as suggested below"
Start()
end
```

4.0 Flowchart

A flowchart is the visual portrayal of the process or system structure, method or solution to the problem step by step. The flowchart describes the data stream by means of IT systems and flow elements. The flow is a series of logical operations which satisfy specific demands. A flux diagram shows how the process work can be enhanced, enables the important process parts to be identified and separated the phases which are not essential or excessive. The Flowcharts are particularly useful for programmers, drawing them in the early phase of creation of computers. The Flowcharts help to write complicated programs in every high level of language. The Flowcharts enables developers to simply understand the logic of the complicated issues and to convey their solution to others and allow participants to share the process. The group talks are particularly useful in the creation of Flowcharts because participants cannot fully understand the details of the whole process. The Flowcharts allow you to view the team members' inputs and resources. The areas for monitoring, improvement or enhanced efficiency must be identified (odessa, 2021).

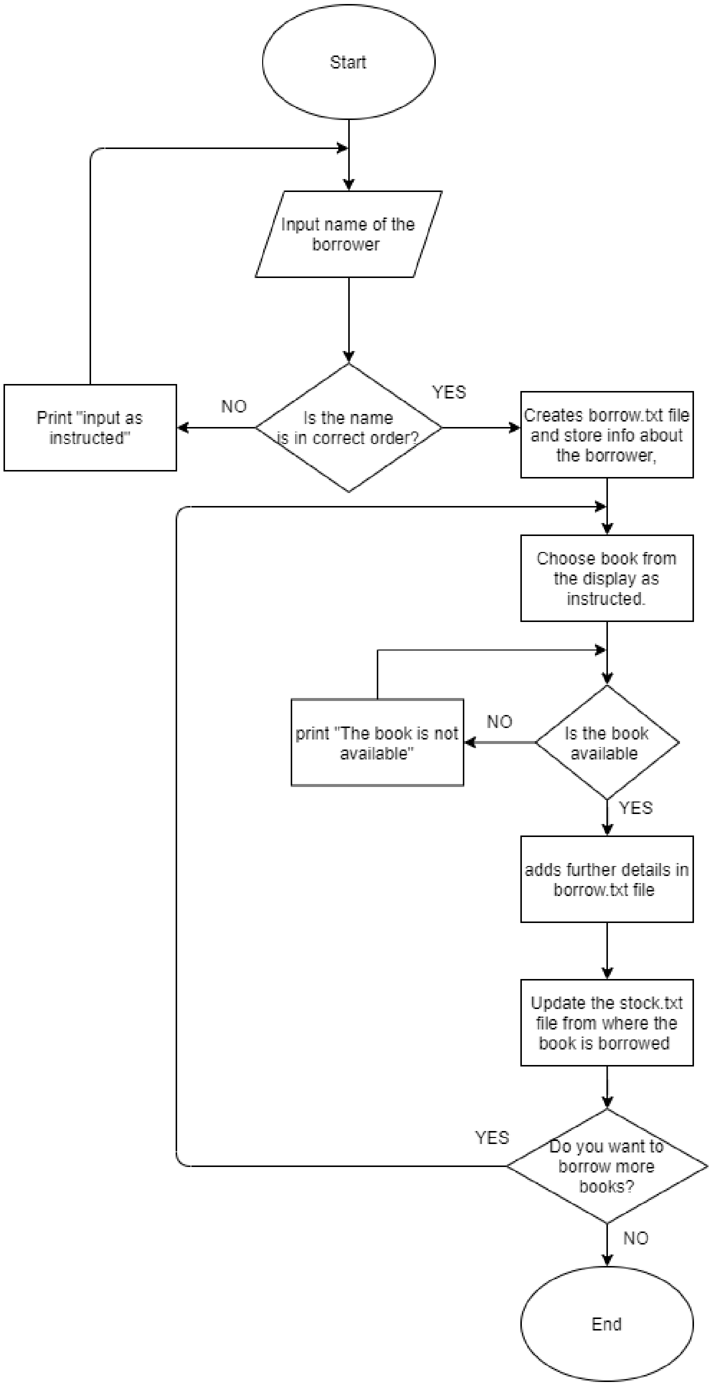

Start
Input name of the borrower
Is the name is in correct order?
NO
Print "input as instructed"
YES
Creates borrow.txt file and store info about the borrower,
Choose book from the display as instructed.
Is the book available
NO
print "The book is not available"
YES
adds further details in borrow.txt file
Update the stock.txt file from where the book is borrowed
Do you want to borrow more books?
YES
NO
End

Figure 3: Flowchart of borrow

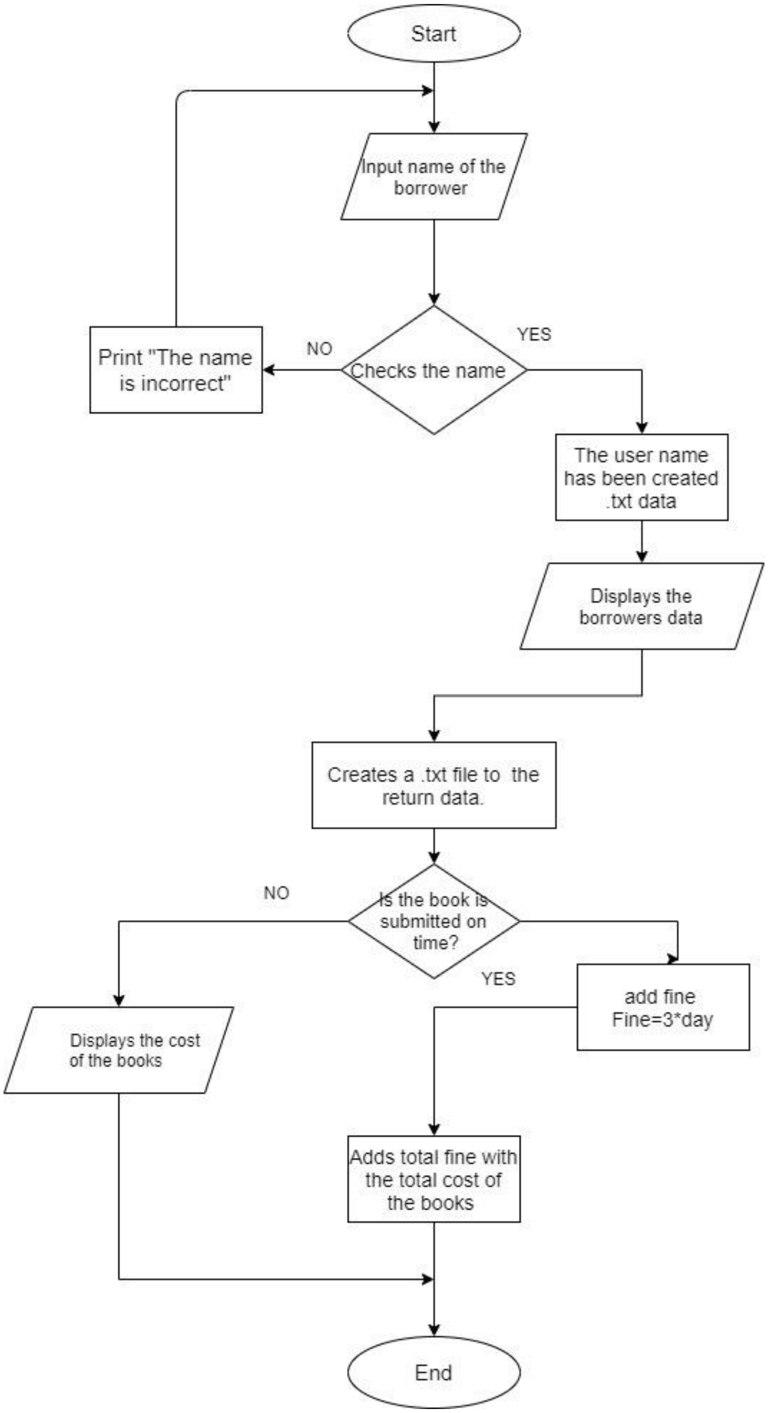

Start
Input name of the borrower
Checks the name
NO
YES
Print "The name is incorrect"
The user name has been created .txt data
Displays the borrowers data
Creates a .txt file to the return data.
Is the book is submitted on time?
NO
YES
add fine
Fine=3*day
Displays the cost of the books
Adds total fine with the total cost of the books
End

Figure 4: Flowchart of return

5.0 Data structure

Data structure can be defined as the set of data pieces that allow for efficient use of data in the computer in order to be efficiently stored and organized. For instance arrays, linked list, stack, queue, etc. some data structures. In practically every part of computer science, data structures are often used, i.e. the operating system, compiler design, artificial intelligence, graphics and much more. Data structures are the main core of many IT methods, which allow programmers to efficiently deal with data.

5.1 Python Built in Data structure

It plays a key role to improve a software or program's performance, as the fundamental function of the software is the quickest possible store and retrieval of user data. The data structures that allow you to store and access data, are implicitly supported in Python. The structures are known as list, dictionary, tuples and set (jaiswal, 2021).

❖ **List**

Lists will be used to sequentially store data of various forms of data. Each entry of the list, called the index, has addresses assigned. The index value begins with zero and continues until the last element termed the positive index.

For Example:

List = [Avenger, Tony Starc]

Print(list)

- List is changeable therefore it can be altered and manipulated.

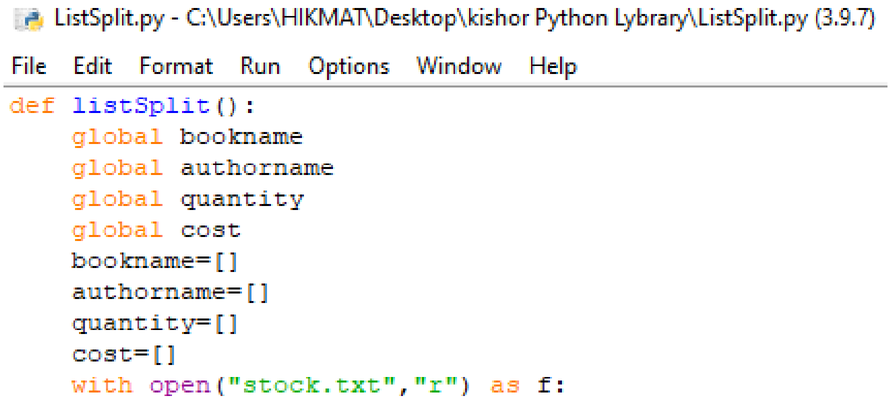
Example:

List = [Avenger, Tony Starc]

List[1] = Will Smith

Now the above list will be [Avenger, Will Smith]

- Negative indexing as well as ordering are supported in the list. This indexes either negatively and positively. Every index list begins at zero.


ListSplit.py - C:\Users\HIKMAT\Desktop\kishor Python Lybrary\ListSplit.py (3.9.7)
File Edit Format Run Options Window Help

```
def listSplit():
    global bookname
    global authorname
    global quantity
    global cost
    bookname=[]
    authorname=[]
    quantity=[]
    cost=[]
    with open("stock.txt","r") as f:
```

Figure 5: Screenshot of list

❖ Dictionary

Dictionary are used to store **key-value** pairs. To understand better, think of a phone directory where hundreds and thousands of names and their corresponding numbers have been added. Now the constant values here are Name and the Phone Numbers which are called as the keys.

For Example:

python = {1, “Tesla”,”elon musk”,32,$1 }

print(python([1])

It would print book name Tesla

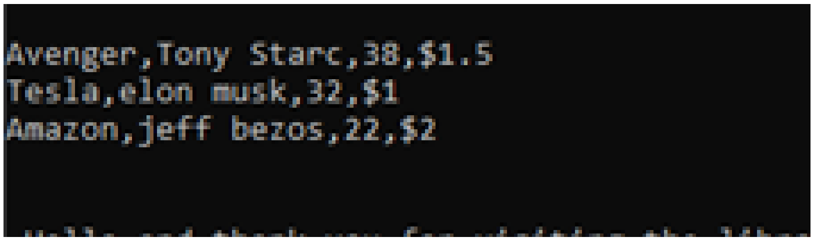
- For accessing the values of dictionary, square brackets along with the the key is used.

Dictionary can be updated by adding key value pairs

For Example:

dict = {1, “Tesla”,”elon musk”,32,$1 }

dict["books"] = "Tesla"



```
Avenger,Tony Starc,38,$1.5
Tesla,elon musk,32,$1
Amazon,jeff bezos,22,$2
```

Figure 6: Screenshot of Dictionary

❖ Tuples

Tuples are same like lists, except the data cannot be changed regardless of what once typed into the tuple. Only then may the tuple data be altered when the data inside a tuple is mutable. The example software helps you to better understand.

Example:

value = () ///it is an empty tuple

value = (1, “Tesla”,”elon musk”,32,$1) ///tuple with 4 elements

Indexing a tuple is same as in a lists in python,

For Example:

value = (1, “Tesla”,”elon musk”,32,$1)

print(value[0]) ///this will print us 1.

Print ([-3]) ///this will print us elon musk.

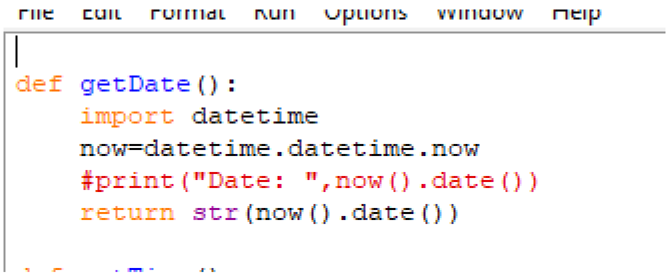
Iterating through tuples

value = (1, “Tesla”,”elon musk”,32,$1)

for B in range(len(value)):

print(value[B])

Once a tuple is created, you cannot change its values. Tuples are unchangeable.



```
def getDate():
    import datetime
    now=datetime.datetime.now
    #print("Date: ",now().date())
    return str(now().date())
```

Figure 7: Screenshot of tuples

❖ Sets

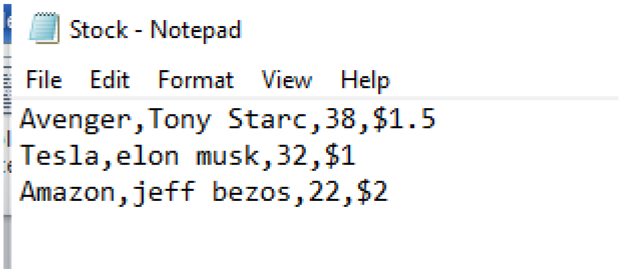
Sets are a special set of unordered items. So, it would only be input once, even if the data was repeated more than once. It seems like the mathematical sets you have learned (edureka.com, 2021).

For Example

Set = {1, “Tesla”,”elon musk”,32,$1 }

Print(Set)

- It will print all the index of set.



```
Avenger,Tony Starc,38,$1.5
Tesla,elon musk,32,$1
Amazon,jeff bezos,22,$2
```

Figure 8: Screenshot of sets

❖ String

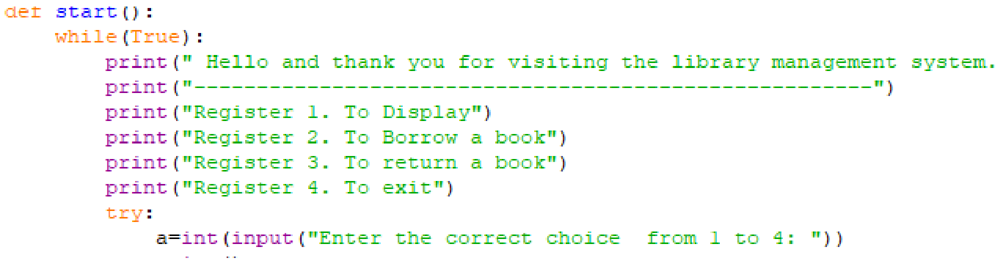
A string is a data type similar to an integer and a floating point unit that is used to represent text rather than figures in programming. It is made up of a series of characters that may include gaps and numerals. The words "burger" and "I ate 4 hamburgers," for example, are both strings. If provided properly, even "12345" may be deemed a string. To be identified as a string rather than a number or variables value, programmers usually surround strings in quotation marks.

For Example:

for i in range(len(ListSplit.bookname)):

print("Enter", i, "to borrow book", ListSplit.bookname[i])

- Words that are written inside double quotation are strings.



```
def start():
    while(True):
        print(" Hello and thank you for visiting the library management system.
        print("-----------------------------------------------------")
        print("Register 1. To Display")
        print("Register 2. To Borrow a book")
        print("Register 3. To return a book")
        print("Register 4. To exit")
        try:
            a=int(input("Enter the correct choice  from 1 to 4: "))
```

Figure 9: screenshot of string

5.2 Python user-defined data structures

These data structures have their own set of features and are created by modifying built-in data structures. Based on these characteristics, they are used in appropriate situations. These can be divided down into the following categories:

- **Stacks**

  These are data structures that operate on the LIFO principle (Last In First Out). The objects all go to the same node, which is called top. The elements leave the scene. The top node is the sole way to get to the final entry element.

- **Queues**

  These data structures are built on the first-in, first-out (FIFO) principle (First In First Out). The elements enter from one end and exit from the other. The lines we see at ATMs, the mess, and other places are all the same.

- **Linked lists**

  linear data structures that serve as indicators for the values of other elements. A node has two values: one is the data, and the other is the pointer to the next node. Linked lists, as opposed to lists, can be scaled up.

- **Graphs**

  These data structures are also non-linear, allowing nodes and edges to collaborate effectively. There are various nodes that connect to other nodes to form a structure that resembles a word map. It's used in apps like Google Maps to choose the shortest route between two locations, as well as to find friends in social networking apps.

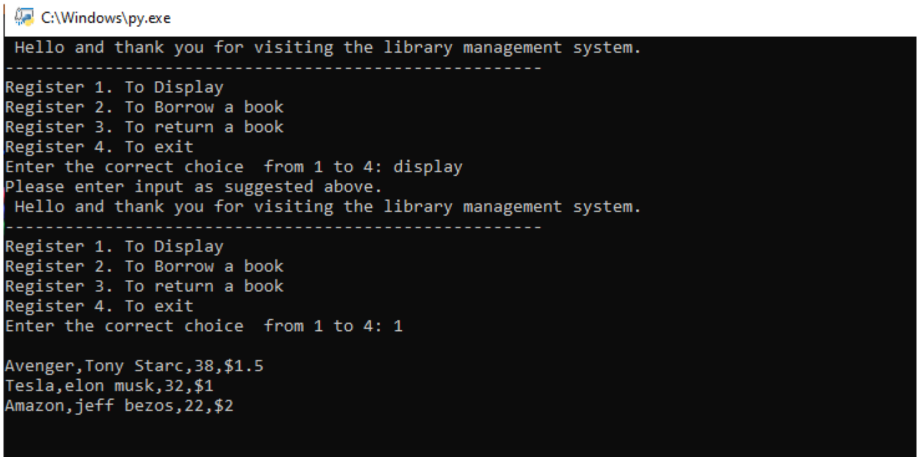
7.0 Testing

Python's most popular testing framework is pytest. You may use pytest to test anything from simple Python scripts to databases, APIs, and user interfaces. Despite the fact that pytest is primarily used for API testing, we will simply cover the fundamentals of pytest in this post. Installation: Using the command, you may install pytest from PyPI. Use: In the project source, use the following command to invoke the pytesttest runner: Unlike nose2, pytest searches for test files throughout the project directory.

7.1 Test 1

Action	To verify whether it's capable of handling exceptions.
Expected Output	It must be capable of managing large amount of user data.
Actual Output	The program has been capable to manage even incorrect input from the user.
Test Result	Unless the user leaves the program, everything continues within the flow.

Table 1: Table of test 1



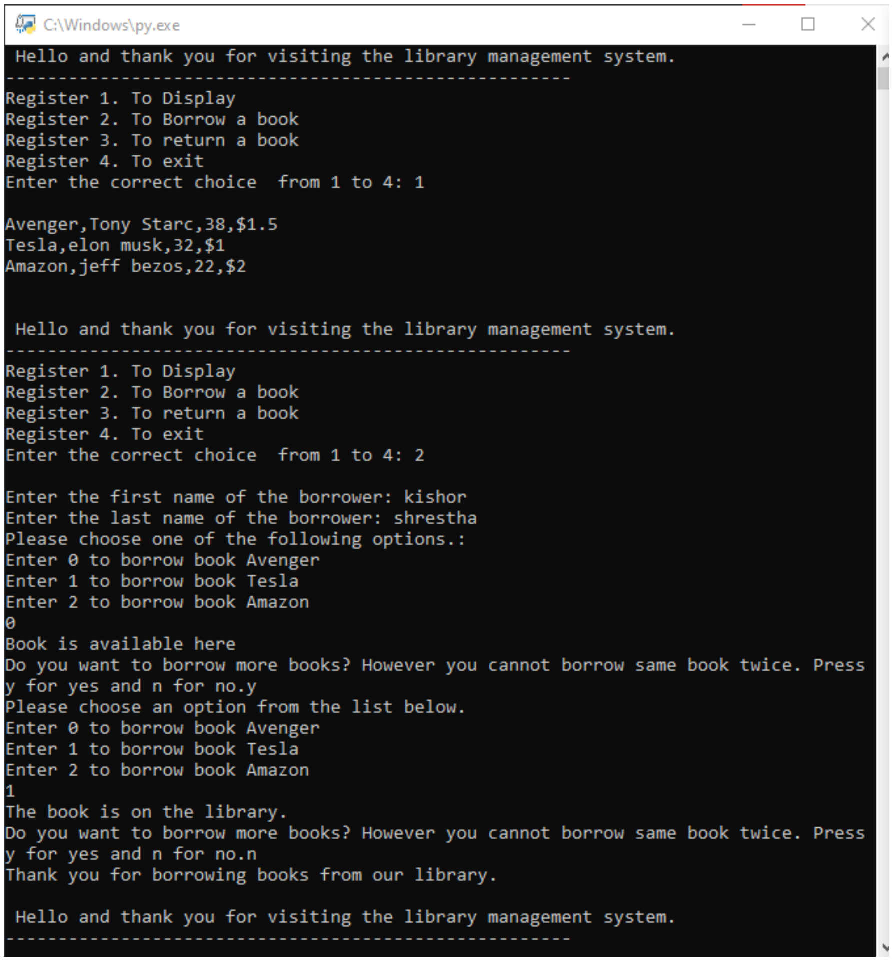
```
 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: display
Please enter input as suggested above.
 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 1

Avenger,Tony Starc,38,$1.5
Tesla,elon musk,32,$1
Amazon,jeff bezos,22,$2
```

Figure 10: Figure of test 1

7.2 Test 2

Action	To verify whether the program is capable of borrowing multiple books.
Expected Output	It must be able to inform the users about whether or not they want to borrow additional books
Actual Output	The program was able to inform the users about whether or not they want to borrow additional books.
Test Result	Test was successful.

Table 2: Table of test 2



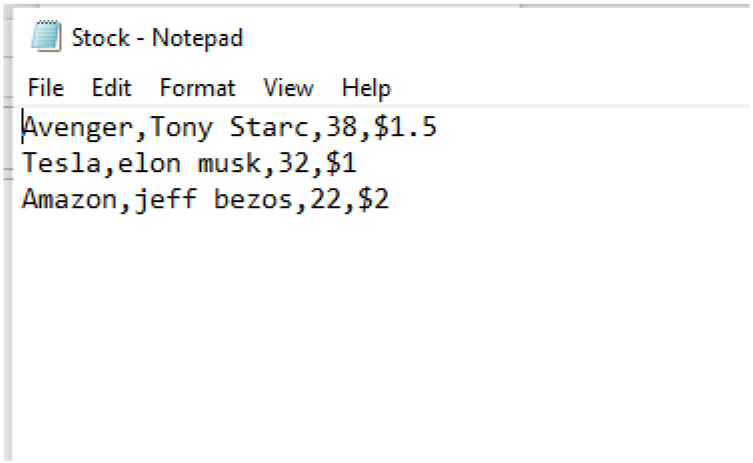
```
 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 1

Avenger,Tony Starc,38,$1.5
Tesla,elon musk,32,$1
Amazon,jeff bezos,22,$2


 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 2

Enter the first name of the borrower: kishor
Enter the last name of the borrower: shrestha
Please choose one of the following options.:
Enter 0 to borrow book Avenger
Enter 1 to borrow book Tesla
Enter 2 to borrow book Amazon
0
Book is available here
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.y
Please choose an option from the list below.
Enter 0 to borrow book Avenger
Enter 1 to borrow book Tesla
Enter 2 to borrow book Amazon
1
The book is on the library.
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.n
Thank you for borrowing books from our library.

 Hello and thank you for visiting the library management system.
------------------------------------------------------
```

Figure 11: Figure of test 2

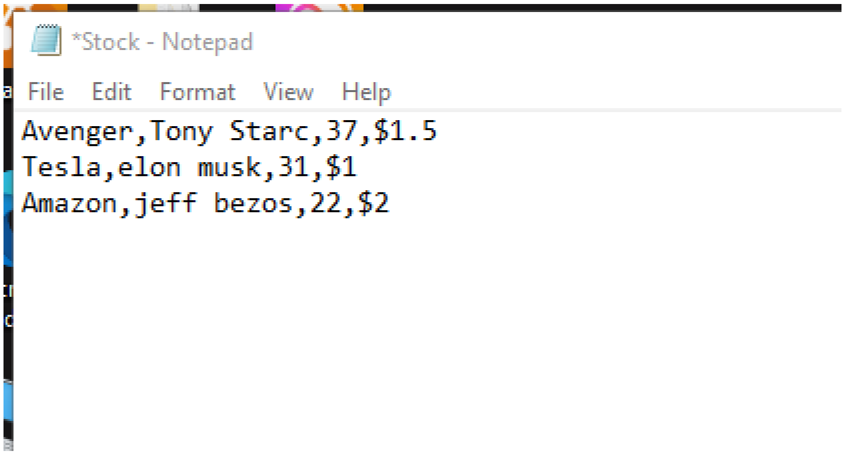
7.3 Test 3

Action	To verify if the number of books inside the store file has been lowered by one when borrowing
Expected Output	It should be able to lower the number of books in circulation by 1.
Actual Output	The program was able to lower the number of books in circulation by 1.
Test Result	Test was successful.

Table 3: Table of test 3



```
Avenger,Tony Starc,38,$1.5
Tesla,elon musk,32,$1
Amazon,jeff bezos,22,$2
```



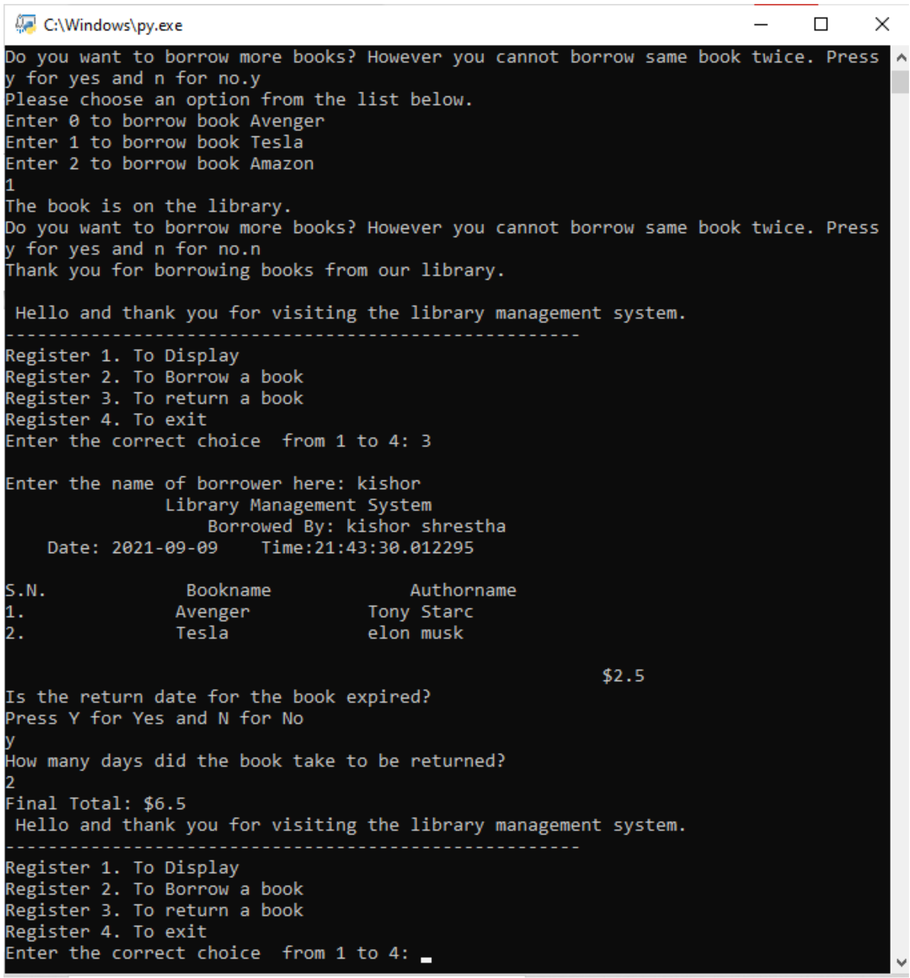
```
Avenger,Tony Starc,37,$1.5
Tesla,elon musk,31,$1
Amazon,jeff bezos,22,$2
```

Figure 12: Figure of test 3

7.4 Test 4

Action	To find if the fine is implemented after the book is returned.
Expected Output	It must be able to show the overall cost of the returned book.
Actual Output	The program was be able to show the overall cost of the returned book.
Test Result	Test was successful.

Table 4: Table of test 4



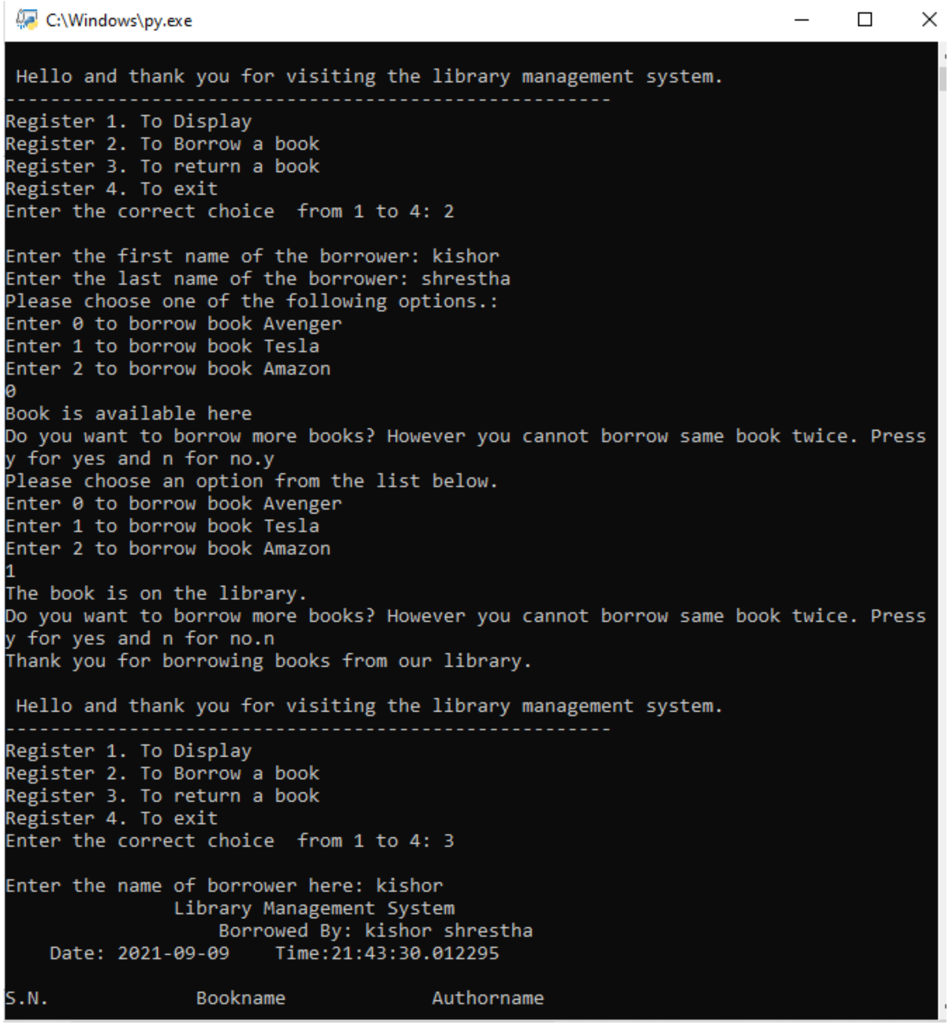
```
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.y
Please choose an option from the list below.
Enter 0 to borrow book Avenger
Enter 1 to borrow book Tesla
Enter 2 to borrow book Amazon
1
The book is on the library.
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.n
Thank you for borrowing books from our library.

 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 3

Enter the name of borrower here: kishor
               Library Management System
                   Borrowed By: kishor shrestha
    Date: 2021-09-09    Time:21:43:30.012295

S.N.             Bookname             Authorname
1.              Avenger           Tony Starc
2.              Tesla             elon musk

                                                        $2.5
Is the return date for the book expired?
Press Y for Yes and N for No
y
How many days did the book take to be returned?
2
Final Total: $6.5
 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: _
```

Figure 13: Figure of test 4

7.5 Test 5

Action	To see if the data has been saved in a text file or not.
Expected Output	It should be able to store data in text file.
Actual Output	The program was able to store data in text file.
Test Result	Test was successful.

Table 5: Table of test 5



```
 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 2

Enter the first name of the borrower: kishor
Enter the last name of the borrower: shrestha
Please choose one of the following options.:
Enter 0 to borrow book Avenger
Enter 1 to borrow book Tesla
Enter 2 to borrow book Amazon
0
Book is available here
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.y
Please choose an option from the list below.
Enter 0 to borrow book Avenger
Enter 1 to borrow book Tesla
Enter 2 to borrow book Amazon
1
The book is on the library.
Do you want to borrow more books? However you cannot borrow same book twice. Press
y for yes and n for no.n
Thank you for borrowing books from our library.

 Hello and thank you for visiting the library management system.
------------------------------------------------------
Register 1. To Display
Register 2. To Borrow a book
Register 3. To return a book
Register 4. To exit
Enter the correct choice  from 1 to 4: 3

Enter the name of borrower here: kishor
               Library Management System
                   Borrowed By: kishor shrestha
    Date: 2021-09-09    Time:21:43:30.012295

S.N.             Bookname            Authorname
```

Figure 14: Figure of test 5

8.0 Conclusion

This course work was the first task of Python to us and, while it was evident, difficult, I nevertheless manage to overcome certain hurdles. The main purpose of this project was to develop and learn about the many components of data primitives and the use of codes and about the use of python. WAS CREATED Python to a library manager system. IN, The MySQL Database system for this python Library implemented data storage. The different tables for maintaining the library data are included in this database.

All of them are self-contained Python data structures that may be built in the same way as simple data types like integer and so on. Some of these can be classified as follows: This database provides the various tables to keep library data. The study was supported by overcoming several types of hurdles and problems and presenting various facets of the subject. A thorough study was therefore required to provide the program a distinct concept, Above all, Several features such as Add Books were used to construct the library management system: We can add LMS system books with book details such as the title, author, publisher, price, book numbers with this capability. We could also include a Member Name, Mobile Number, Email Address, Class Name and Member Address to the Add Member function. You can acquire details about your prior Book by using Update Book. Likewise the Member Details will be updated using Update Member function and the details of the published book are recorded by means of the issue book feature. The details of Book ID, Member ID, issue dates, and return dates are stored in this feature. Using Return Book function book transactions, the members shall update and return them. Search bar books available in the system and reports based on books, editors and member searches are performed using Report Menu in a similar fashion. It was entirely implemented today and works wonderfully since I had difficulties with various programs. I also had a little flowchart difficulty when developing several forms to illustrate the functionality of this software, but now it has also been solved.

I have experienced many challenges during my coursework because I was unfamiliar with such courses. I received assistance both from my lecturer and the elderly, which made my work somewhat easier. While doing this course I have worked hard and made my 100 percent effort. Overall, this course work was incredibly fun since I had something fresh to study.

9.0 Appendix

9.1 Appendix for borrow

```
import dt

import ListSplit


def borrowBook():

    success=False

    while(True):

        firstName=input("Enter the first name of the borrower: ")

        if firstName.isalpha():

            break

        print("Please enter letters ranging from A to Z")

    while(True):

        lastName=input("Enter the last name of the borrower: ")

        if lastName.isalpha():

            break

        print("Please enter letters ranging from A to Z")


    t="Borrow-"+firstName+".txt"

    with open(t,"w+") as f:

        f.write("               Library Management System  \n")

        f.write("                   Borrowed By: "+ firstName+" "+lastName+"\n")
        f.write("    Date: " + dt.getDate()+"    Time:"+ dt.getTime()+"\n\n")
        f.write("S.N. \t\t Bookname \t      Authorname \n" )

    while success==False:
        print("Please choose one of the following options:")
        for i in range(len(ListSplit.bookname)):
            print("Enter", i, "to borrow book", ListSplit.bookname[i])
        try:
            a=int(input())
            try:
                if(int(ListSplit.quantity[a])>0):
                    print("Book is available here")
                    with open(t,"a") as f:
                        f.write("1. \t\t"+ ListSplit.bookname[a]+"\t\t
"+ListSplit.authorname[a]+"\n")

                    ListSplit.quantity[a]=int(ListSplit.quantity[a])-1
                    with open("Stock.txt","w+") as f:
                        for i in range(3):

f.write(ListSplit.bookname[i]+","+ListSplit.authorname[i]+","+str(ListSplit.quantity[i])+","+"
$"+ListSplit.cost[i]+"\n")
```

```
#multiple book borrowing code
loop=True
count=1
while loop==True:
    choice=str(input("Do you want to borrow more books? However you cannot borrow same book twice. Press y for yes and n for no."))
    if(choice.upper()=="Y"):
        count=count+1
        print("Please choose an option from the list below.")
        for i in range(len(ListSplit.bookname)):
            print("Enter", i, "to borrow book", ListSplit.bookname[i])
        a=int(input())
        if(int(ListSplit.quantity[a])>0):
            print("The book is on the library.")
            with open(t,"a") as f:
                f.write(str(count) +". \t\t"+ ListSplit.bookname[a]+"\t\t "+ListSplit.authorname[a]+"\n")

            ListSplit.quantity[a]=int(ListSplit.quantity[a])-1
            with open("Stock.txt","w+") as f:
                for i in range(3):
```

```
f.write(ListSplit.bookname[i]+","+ListSplit.authorname[i]+","+str(ListSplit.quantity[i])+","+"
$"+ListSplit.cost[i]+"\n")
                                        success=False
                            else:
                                loop=False
                                break
                        elif (choice.upper()=="N"):
                            print ("Thank you for borrowing books from our library. ")
                            print("")
                            loop=False
                            success=True
                        else:
                            print("Please choose an option as instructed")


                else:
                    print("Book is not available here")
                    borrowBook()
                    success=False
            except IndexError:
                print("")
                print("Please choose any book according to their given number.")
```

```
        except ValueError:
            print("")
            print("Please choose the number as suggested.")
```

9.2 Appendix for Main

```
import Return
import ListSplit
import dt
import Borrow


def start():
    while(True):
        print(" Hello and thank you for visiting the library management system.    ")
        print("----------------------------------------------------")
        print("Register 1. To Display")
        print("Register 2. To Borrow a book")
        print("Register 3. To return a book")
        print("Register 4. To exit")
        try:
            a=int(input("Enter the correct choice  from 1 to 4: "))
            print()
            if(a==1):
                with open("stock.txt","r") as f:
                    lines=f.read()
                    print(lines)
                    print ()

            elif(a==2):
                ListSplit.listSplit()
                Borrow.borrowBook()
            elif(a==3):
                ListSplit.listSplit()
                Return.returnBook()
            elif(a==4):
                print("Thank you for making use of the library management system.")
                break
            else:
                print("Enter the correct choice  from 1 to 4")
        except ValueError:
            print("Please enter input as suggested above.")
start()
```

9.3 Appendix for return

```
import ListSplit

import dt

def returnBook():

    name=input("Enter the name of borrower here: ")

    a="Borrow-"+name+".txt"

    try:

        with open(a,"r") as f:

            lines=f.readlines()

            lines=[a.strip("$") for a in lines]


        with open(a,"r") as f:

            data=f.read()

            print(data)

    except:

        print("The borrower name entered is incorrect")

        returnBook()


    b="Return-"+name+".txt"

    with open(b,"w+")as f:

        f.write("                Library Management System \n")
        f.write("                    Returned By: "+ name+"\n")
        f.write("    Date: " + dt.getDate()+"    Time:"+ dt.getTime()+"\n\n")
        f.write("S.N.\t\tBookname\t\tCost\n")

    total=0.0
    for i in range(3):
        if ListSplit.bookname[i] in data:
            with open(b,"a") as f:
                f.write(str(i+1)+"\t\t"+ListSplit.bookname[i]+"\t\t$"+ListSplit.cost[i]+"\n")
                ListSplit.quantity[i]=int(ListSplit.quantity[i])+1
            total+=float(ListSplit.cost[i])

    print("\t\t\t\t\t\t\t"+"$"+str(total))
    print("Is the return date for the book expired?")
    print("Press Y for Yes and N for No")
    stat=input()
    if(stat.upper()=="Y"):
        print("How many days did the book take to be returned?")
        day=int(input())
        fine=2*day
        with open(b,"a")as f:
            f.write("\t\t\t\t\tFine: $"+ str(fine)+"\n")
        total=total+fine


    print("Final Total: "+ "$"+str(total))
    with open(b,"a")as f:
        f.write("\t\t\t\t\tTotal: $"+ str(total))


    with open("Stock.txt","w+") as f:
            for i in range(3):
f.write(ListSplit.bookname[i]+","+ListSplit.authorname[i]+","+str(ListSplit.quantity[i])+","+" $"+ListSplit.cost[i]+"\n")
```

10.0 Bibliography

christianson, p. (2021, 9 1). *techterms.com*. Retrieved from techterms: https://techterms.com/definition/algorithm

edureka.com. (2021, 9 2). Retrieved from edureka: https://www.edureka.co/blog/data-structures-in-python/

jaiswal, S. (2021, 9 2). *javatpoint.com*. Retrieved from javatpoint: https://www.javatpoint.com/data-structure-introduction

mandavia, M. (2021, 9 1). *economictimes.indiatimes.com*. Retrieved from indiatimes: https://economictimes.indiatimes.com/definition/pseudocode

odessa, C. (2021, 9 2). *conceptdraw.com*. Retrieved from conceptdraw: https://www.conceptdraw.com/How-To-Guide/flowchart-definition

Rossum, G. v. (2021, 9 1). *python.org*. Retrieved from python: https://www.python.org/doc/essays/blurb/

sciencedirect.com. (2021, 9 1). Retrieved from sciencedirect: https://www.sciencedirect.com/topics/computer-science/library-management-system