LONDON METROPOLITAN UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS4051NI Fundamentals of Computing

Assessment Weightage & Type

60% Individual Coursework

Year and Semester

2021 Autumn

Student Name: Aadarsha Muni Shakya

Group: N1

London Met ID: 20049438

College ID:NP01NT4S210023

Assignment Due Date:

Assignment Submission Date: 10th September, 2021

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 content

List of Figures

List of Tables

1. Introduction

In this project, integrated development environment (IDLE) for python is used to code the library management system project. This project is about creating a library management system which allows the use to borrow multiple books and return a single book. Talking about python, Python is a high-level, general-purpose and a very popular programming language (GeeksforFeeks, 2021). Python has features like allowing programming in Object-Oriented, programs are generally smaller than other programming languages, huge collection of standard libraries which will allow the users to do machine learning, GUI applications, Web frameworks and many more.

Throughout the project the main goal was to create a responsive library management system which will increase the quantity when borrowed and decrease when returned. On top of that, a bill is also generated; when books or book is borrowed the bill will have information like name of the book, name of the customer, date and time when borrowed and total cost of book or books. And when a book is returned the bill will have information like name of the book, name of the customer, date and time when returned and if the book is borrowed for more then 10 days fine is also added.

Similarly, the objectives of the project are to learn basic programming, validation, iteration, data structures, file handling and exception handling. From creating program which add two integers to creating a library management system is covered in this module. In-order to code an advanced program, the basics codes should be learned. Similarly, throughout this module basic coding was taught which finally helped in the overall project. Basic programming like printing strings, mathematical operations and many more. For validation, propre use of if-else like nested if-else can be done. Similarly for iteration, loop like for and while can be used. Also, the value of a 2D list cand be extracted using nested loop. Moreover, data structures can help to store data in an organized way which will allow the values to be accessed and modified effectively.

For file handling, the objective of extracting data from text file, writing data in text file were met. the bill generating part of the project uses file handling. Similarly, exception handling is used to avoid the program from crashing. For illustration, if user assigns string value in a variable which takes integer values the program will crash. to avoid this exception handling is done.

2. Discussion and Analysis

2.1 Algorithm

Step 1: Display welcome message

Step 2: Read data.txt file

Step 3: Extract data from file and store it in a list

Step 4: Display list in a table format

Step 5: Initialize total_cost to 0, book_name to empty string and flag to true

Step 6: Display 1 for Borrow, 2 for return and 3 to exit

Step 7: Loop while flag equals to true

Step 8: Input value 1, 2 or 3

Step 9: If value equals to 1

Step 10: Input name of the customer

Step 11: Loop while flag is equals to True

Step 12: Input bookID of the book

Step 13: If bookID equals to 1

Step 14: If quantity not equals to 0

Step 15: subtract 1 from quantity of book with ID 1

Step 16: Add cost of the book to total_cost

Step 17: Add book name with id 1 to book_name

Step 18: Else, display out of stock message

Step 19: Display y to borrow another book

Step 20: If y, Jump to step 7

Step 21: Else, Create a new text file with unique name

Step 22: Write total_cose, book_name, name, date and time

Step 23: break

Step 24: If bookID equals to 2

Step 25: Check if quantity equals to 0

Step 26: If no, subtract 1 from quantity of book with ID 2

Step 27: Add cost of the book to total_cost

Step 28: Add book name with id 2 to book_name

Step 29: If yes, display out of stock message

Step 30: Display y to borrow another book

Step 31: If y, Jump to step 7

Step 32: Else, Create a new text file with unique name

Step 33: Write total_cose, book_name, name, date and time

Step 34: break

Step 35: If bookID equals to 3

Step 36: Check if quantity equals to 0

Step 37: If no, subtract 1 from quantity of book with ID 3

Step 38: Add cost of the book to total_cost

Step 39: Add book name with id 3 to book_name

Step 40: If yes, display out of stock message

Step 41: Display y to borrow another book

Step 42: If y, Jump to step 7

Step 43: Else, Create a new text file with unique name

Step 44: Write total_cose, book_name, name, dateand time

Step 45: break

Step 46: If bookID equals to 4

Step 47: Check if quantity equals to 0

Step 48: If no, subtract 1 from quantity of book with ID 4

Step 49: Add cost of the book to total_cost

Step 50: Add book name with id 4 to book_name

Step 51: If yes, display out of stock message

Step 52: Display y to borrow another book

Step 53: If y, Jump to step 7

Step 54: Else, Create a new text file with unique name

Step 55: Write total_cose, book_name, name, dateand time

Step 56: break

Step 57: If bookID equals to 5

Step 58: Check if quantity equals to 0

Step 59: If no, subtract 1 from quantity of book with ID 5

Step 60: Add book name with id 5 to book_name

Step 61: Add cost of the book to total_cost

Step 62: If yes, display out of stock message

Step 63: Display y to borrow another book

Step 64: If y, Jump to step 7

Step 65: Else, Create a new text file with unique name

Step 66: Write total_cose, book_name, name, dateand time

Step 67: break

Step 68: If value equals to 2

Step 69: Input the name of the customer

Step 70: Input the book ID

Step 71: Input number of days the book was borrowed

Step 72: If bookID equals to 1

Step 73: Add 1 to quantity of book with ID 1

Step 74: If days greater than 10

Step 75: Fine of $2 per day is added

Step 76: Else fine of $0 is added

Step 77: Create a new text fine with unique name

Step 78: Write customer name, Name of book with bookID 1, fine and date and time

Step 79: If bookID equals to 2

Step 80: Add 1 to quantity of book with ID 2

Step 81: If days greater than 10

Step 82: Fine of $2 per day is added

Step 83: Else fine of $0 is added

Step 84: Create a new text fine with unique name

Step 85: Write customer name, Name of book with bookID 2, fine and date and time

Step 86: If bookID equals to 3

Step 87: Add 1 to quantity of book with ID 3

Step 88: If days greater than 10

Step 89: Fine of $2 per day is added

Step 90: Else fine of $0 is added

Step 91: Create a new text fine with unique name

Step 92: Write customer name, Name of book with bookID 3, fine and date and time

Step 93: If bookID equals to 4

Step 94: Add 1 to quantity of book with ID 4

Step 95: If days greater than 10

Step 96: Fine of $2 per day is added

Step 97: Else fine of $0 is added

Step 98: Create a new text fine with unique name

Step 99: Write customer name, Name of book with bookID 4, fine and date and time

Step 100: If bookID equals to 5

Step 101: Add 1 to quantity of book with ID 5

Step 102: If days greater than 10

Step 103: Fine of $2 per day is added

Step 104: Else fine of $0 is added

Step 105: Create a new text fine with unique name

Step 106: Write customer name, Name of book with bookID 5, fine and date and a time

Step 107: If value equals to 3

Step 108: Exit the program

2.2 Flowchart

Flowchart is a graphical representation of an algorithm (Aggarwal, 2020). This will help the users to visually imagine the program and conduct necessary changes. Checking on algorithm for editing the program is a hectic work of flowchart is used which is less hectic.

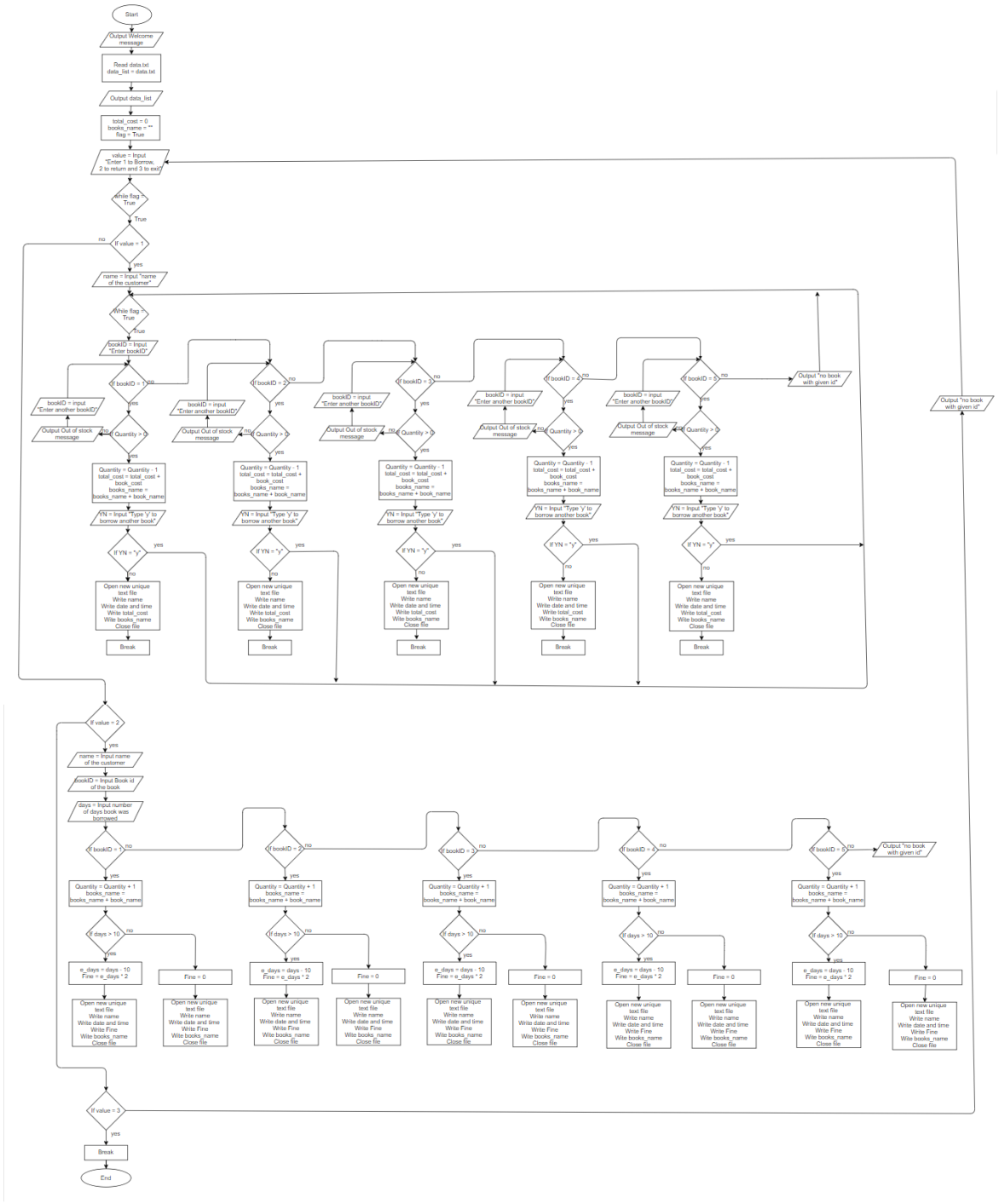

Figure 1: Flowchart

2.3 Pseudocode

MODULE: function

FUNCTION welcome ()

OUTPUT “Hello and Welcome to library management system”

END FUNCTION

FUNCTION valid_value ()

OUTPUT “The entered value is invalid please try again”

END FUNCTION

FUNCTION ty_three ()

OUTPUT “Thankyou for using my library management system”

END FUNCTION

FUNCTION now_borrow ()

OUTPUT “You will now borrow a book”

END FUNCTION

FUNCTION now_return ()

OUTPUT “You will now return a book”

END FUNCTION

FUNCTION error_book ()

OUTPUT “Please provide a valid Book ID !!!” & “Try again with a different book ID”

END FUNCTION

FUNCTION n_avilable_book ()

OUTPUT “Book out of stock !!!”

END FUNCTION

FUNCTION avilable_display ()

OUTPUT “Book is available”

END FUNCTION

FUNCTION ty_alt ()

OUTPUT “Thankyou for borrowing book/books” & “Be sure to return it within 10 days or pay the fine”

END FUNCTION

```
FUNCTION fine ()
    OUTPUT “Thankyou for returning the book” & ” You have returned
    the book late so you will have to pay the fine”
END FUNCTION
FUNCTION thankyou ()
    OUTPUT “Thankyou for returning the book” & “If you want to return
    another book enter 2 down below”
END FUNCTION
FUNCTION invalid_dt_bookid ()
    OUTPUT “ID should be a integer between 1-5”
END FUNCTION
FUNCTION invalid_dt ()
    OUTPUT “Enter a integer value”
END FUNCTION

FUNCTION one_d_list () RETURNS ARRAY
    DECLARE dd_list : ARRAY
    DECLARE data_list : ARRAY
    OPENFILE “data” FOR READ
    dd_list ← lines in data
    FOR i ← 0 TO length(dd_list)
        FOR j ← 0 TO length(dd_list[i])
            data_list ← dd_list[i][j]
        END FOR
    END FOR
    CLOSEFILE “data”
    RETURN data_list
END FUNCTION
FUNCTION display_book(data_list)
    OUTPUT data_list[0] & data_list[1] & data_list[2] & data_list[3] &
    data_list[4]
```

```
        OUTPUT data_list[5] & data_list[6] & data_list[7] & data_list[8] &
        data_list[9]
        OUTPUT data_list[10] & data_list[11] & data_list[12] & data_list[13]
        & data_list[14]
        OUTPUT data_list[15] & data_list[16] & data_list[17] & data_list[18]
        & data_list[19]
        OUTPUT data_list[20] & data_list[21] & data_list[22] & data_list[23]
        & data_list[24]
    END FUNCTION
    FUNCTION continue_borrow (data_list, display_book, total_cost, name,
    booksName) RETURNS INTEGER, STRING
        add ← TRUE
        Book_cost, bookName ← borrow_book(data_list)
        total_cost ← total_cost + real(book_cost)
        booksName ← booksName & bookName
        WHILE add = TRUE
            YN ← INPUT “If tou want to borrow a another book type 'y',
            else type any onthe word or letter :”
            IF YN = “y” THEN
                Book_cost, bookName ← borrow_book(data_list)
                total_cost ← total_cost + real(book_cost)
                booksName ← booksName & bookName
            ELSE
                ty_alt()
                BREAK
            END IF
        END WHILE
        RETURN total_cost, booksName
    END FUNCTION
    FUNCTION borrow_book (data_list) RETURNS INTEGER, STRING
        flag ← TRUE
```

```
BookID ← INPUT "Enter the book Id of the book you want to borrow:"
WHILE flag = TRUE
    IF BookID = 1 THEN
        qnt ← integer(data_list[3])
        IF qnt > 0 THEN
            CALL available_display()
            CALL remove_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            price ← CALL cost(data_list,bookID)
            bookName ← CALL book_name (data_list, bookID)
            BREAK
        ELSE
            n_available_book()
            BookID ← INPUT "Enter the book Id of a another book you want to borrow:"
        END IF
    ELSE IF BookID = 2 THEN
        qnt ← integer(data_list[8])
        IF qnt > 0 THEN
            CALL available_display()
            CALL remove_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            price ← CALL cost(data_list,bookID)
            bookName ← CALL book_name (data_list, bookID)
            BREAK
        ELSE
            n_available_book()
```

```
                BookID ← INPUT “Enter the book Id of a
                another book you want to borrow:”
        END IF
    ELSE IF BookID = 3 THEN
        qnt ← integer(data_list[13])
        IF qnt > 0 THEN
            CALL available_display()
            CALL remove_book (bookID, qnt, data_list)
            CALL display_book(data_list)
            price ← CALL cost(data_list,bookID)
            bookName ← CALL book_name (data_list ,
            bookID)
            BREAK
        ELSE
            n_available_book()
            BookID ← INPUT “Enter the book Id of a
            another book you want to borrow:”
        END IF
    ELSE IF BookID = 4 THEN
        qnt ← integer(data_list[18])
        IF qnt > 0 THEN
            CALL available_display()
            CALL remove_book ( bookID, qnt, data_list)
            CALL display_book(data_list)
            price ← CALL cost(data_list,bookID)
            bookName ← CALL book_name (data_list,
            bookID)
            BREAK
        ELSE
            n_available_book()
```

```
                    BookID ← INPUT "Enter the book Id of a
                    another book you want to borrow:"
                END IF
            ELSE IF BookID = 5 THEN
                qnt ← integer(data_list[23])
                IF qnt > 0 THEN
                    CALL available_display()
                    CALL remove_book(bookID,qnt,data_list)
                    CALL display_book(data_list)
                    price ← CALL cost(data_list,bookID)
                    bookName ← CALL book_name(data_list,
                    bookID)
                    BREAK
                ELSE
                    n_available_book()
                    BookID ← INPUT "Enter the book Id of a
                    another book you want to borrow:"
                END IF
            ELSE
                CALL error_book()
            END IF
        END WHILE
        RETURN price, bookName
    END FUNCTION
    FUNCTION remove_book(bookID, quantity,data_list)
        qnt1 ←data_list[3]
        qnt2 ←data_list[8]
        qnt3 ←data_list[13]
        qnt4 ←data_list[18]
        qnt5 ←data_list[23]
        OPENFILE "data" FOR WRITE
```

```
IF BookID = 1 THEN
    qnt1 ← integer(data_list[3]) - 1
    WRITEFILE "1,Nineteen Eighty-Four,George Orwell," &
    string(qnt1) & ",$9.99\n2,To Kill a Mockingbird,Harper Lee,"
    & qnt2 & ",$17.99\n3,The Catcher in the Rye,J.D.Salinger,"
    & qnt3 & ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
    ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 2 THEN
    qnt2 ← integer(data_list[8]) - 1
    WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
    & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," &
    string(qnt2) & ",$17.99\n3,The Catcher in the
    Rye,J.D.Salinger," & qnt3 & ",$16.99\n4,Beloved,Toni
    Morrison," & qnt4 & ",$16.00\n5,Invinsible Man,Ralph
    Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 3 THEN
    qnt3 ← integer(data_list[13]) - 1
    WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
    & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2 &
    ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," &
    string(qnt3) & ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
    ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 4 THEN
    qnt4 ← integer(data_list[18]) - 1
    WRITEFILE "1,Nineteen Eighty-Four,George Orwell," &
    qnt1)& ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2
    & ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," & qnt3
    & ",$16.99\n4,Beloved,Toni Morrison," & string(qnt4) &
    ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 5 THEN
    qnt5 ← integer(data_list[23]) - 1
```

```
            WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
            & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2 &
            ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," & qnt3 &
            ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
            ",$16.00\n5,Invinsible Man,Ralph Ellison," & string(qnt5) &
            ",$14.00"
        END IF
        CLOSEFILE "data"
        fnl_qnt ← quantity – 1
        IF BookID = 1 THEN
            data_list[3] ← string(fnl_qnt)
        ELSE IF BookID = 2 THEN
            data_list[8] ← string(fnl_qnt)
        ELSE IF BookID = 3 THEN
            data_list[13] ← string(fnl_qnt)
        ELSE IF BookID = 4 THEN
            data_list[18] ← string(fnl_qnt)
        ELSE IF BookID = 5 THEN
            data_list[23] ← string(fnl_qnt)
        END IF
    END FUNCTION
    FUNCTION cost(data_list, bookID) RETURNS STRING
        IF bookID = 1 THEN
            price ← data_list[4]
        ELSE IF bookID = 2 THEN
            price ← data_list[9]
        ELSE IF bookID = 3 THEN
            price ← data_list[14]
        ELSE IF bookID = 4 THEN
            price ← data_list[19]
        ELSE IF bookID = 5 THEN
```

```
            price ← data_list[24]
        END IF
        RETURN price
    END FUNCTION
    FUNCTION book_name(data_list, bookID) RETURNS STRING
        IF bookID = 1 THEN
            bookName ← data_list[1]
        ELSE IF bookID = 2 THEN
            bookName ← data_list[6]
        ELSE IF bookID = 3 THEN
            bookName ← data_list[11]
        ELSE IF bookID = 4 THEN
            bookName ← data_list[16]
        ELSE IF bookID = 5 THEN
            bookName ← data_list[21]
        END IF
        RETURN bookName
    END FUNCTION
    FUNCTION b_bill(total_cost,name,booksName)
        dnt ← Current DateTime
        min ← str(Current Minute)
        sec ← str(Current Second)
        msec ← str(Current Microsecond)
        fileName ← name & min & sec & msec
        OPENFILE fileName FOR WRITE
        WRITEFILE “Name of the Customer: ” & name
        WRITEFILE “Price of the book/books borrowed: $” &
        string(total_cost)
        WRITEFILE “Date and Time of borrow:” & string(dnt)
        WRITEFILE “Name of the book/books borrowed: ” & booksName
        CLOSEFILE filename
```

```
END FUNCTION
FUNCTION return_book(data_list, display_book, name)
    flag ← TRUE
    BookID ← INPUT “Enter the book Id of the book you want to
    borrow:”
    days← INPUT “Enter the number of days you borrowed the book: ”
    CALL fine_yn(days,data_list,name,bookID)
    WHILE flag = TRUE
        IF BookID = 1 THEN
            qnt ← integer(data_list[3])
            CALL add_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            BREAK
        ELSE IF BookID = 2 THEN
            qnt ← integer(data_list[8])
            CALL add_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            BREAK
        ELSE IF BookID = 3 THEN
            qnt ← integer(data_list[13])
            CALL add_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            BREAK
        ELSE IF BookID = 4 THEN
            qnt ← integer(data_list[18])
            CALL add_book(bookID,qnt,data_list)
            CALL display_book(data_list)
            BREAK
        ELSE IF BookID = 5 THEN
            qnt ← integer(data_list[23])
            CALL add_book(bookID,qnt,data_list)
```

```
                    CALL display_book(data_list)
                    BREAK
                ELSE
                    CALL error_book()
                    BREAK
                END IF
            END WHILE
        END FUNCTION

        FUNCTION add_book(bookID, quantity, data_list)
            qnt1 ←data_list[3]
            qnt2 ←data_list[8]
            qnt3 ←data_list[13]
            qnt4 ←data_list[18]
            qnt5 ←data_list[23]
            OPENFILE “data” FOR WRITE
            IF BookID = 1 THEN
                qnt1 ← integer(data_list[3]) + 1
                WRITEFILE "1,Nineteen Eighty-Four,George Orwell," &
                string(qnt1) & ",$9.99\n2,To Kill a Mockingbird,Harper Lee,"
                & qnt2 & ",$17.99\n3,The Catcher in the Rye,J.D.Salinger,"
                & qnt3 & ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
                ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
            ELSE IF BookID = 2 THEN
                qnt2 ← integer(data_list[8]) + 1
                WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
                & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," &
                string(qnt2) & ",$17.99\n3,The Catcher in the
                Rye,J.D.Salinger," & qnt3 & ",$16.99\n4,Beloved,Toni
                Morrison," & qnt4 & ",$16.00\n5,Invinsible Man,Ralph
                Ellison," & qnt5 & ",$14.00"
```

```
ELSE IF BookID = 3 THEN
        qnt3 ← integer(data_list[13]) + 1
        WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
        & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2 &
        ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," &
        string(qnt3) & ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
        ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 4 THEN
        qnt4 ← integer(data_list[18]) + 1
        WRITEFILE "1,Nineteen Eighty-Four,George Orwell," &
        qnt1)& ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2
        & ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," & qnt3
        & ",$16.99\n4,Beloved,Toni Morrison," & string(qnt4) &
        ",$16.00\n5,Invinsible Man,Ralph Ellison," & qnt5 & ",$14.00"
ELSE IF BookID = 5 THEN
        qnt5 ← integer(data_list[23]) + 1
        WRITEFILE "1,Nineteen Eighty-Four,George Orwell," & qnt1
        & ",$9.99\n2,To Kill a Mockingbird,Harper Lee," & qnt2 &
        ",$17.99\n3,The Catcher in the Rye,J.D.Salinger," & qnt3 &
        ",$16.99\n4,Beloved,Toni Morrison," & qnt4 &
        ",$16.00\n5,Invinsible Man,Ralph Ellison," & string(qnt5) &
        ",$14.00"
END IF
CLOSEFILE “data”
fnl_qnt ← quantity + 1
IF BookID = 1 THEN
        data_list[3] ← string(fnl_qnt)
ELSE IF BookID = 2 THEN
        data_list[8] ← string(fnl_qnt)
ELSE IF BookID = 3 THEN
        data_list[13] ← string(fnl_qnt)
```

```
        ELSE IF BookID = 4 THEN
            data_list[18] ← string(fnl_qnt)
        ELSE IF BookID = 5 THEN
            data_list[23] ← string(fnl_qnt)
        END IF
    END FUNCTION
    FUNCTION fine_yn(days, data_list, name, bookID)
        IF days > 10 THEN
            CALL r_f_bill(name,data_list,bookID,days)
            CALL fine()
        ELSE
            CALL r_nf_bill(name,data_list,bookID)
            CALL thankyou()
        END IF
    END FUNCTION
    FUNCTION r_f_bill(name, data_list, bookID, days)
        bookName ← CALL book_name(data_list, bookID)
        fine_ame ←  (days - 10) * 2
        dnt ← Current DateTime
        min ← str(Current Minute)
        sec ← str(Current Second)
        msec ← str(Current Microsecond)
        fileName ← name & min & sec & msec
        OPENFILE fileName FOR WRITE
        WRITEFILE “Name of the Customer: ” & name
        WRITEFILE “fine: $” & string(fine_ame)
        WRITEFILE “Date and Time of borrow:” & string(dnt)
        WRITEFILE “Name of the book borrowed: ” & bookName
        CLOSEFILE filename
    END FUNCTION
    FUNCTION r_nf_bill(name, data_list, bookID)
```

```
            bookName ← CALL book_name(data_list, bookID)
            fine_ame ←  0
            dnt ← Current DateTime
            min ← str(Current Minute)
            sec ← str(Current Second)
            msec ← str(Current Microsecond)
            fileName ← name & min & sec & msec
            OPENFILE fileName FOR WRITE
            WRITEFILE “Name of the Customer: ” & name
            WRITEFILE “fine: $” & string(fine_ame)
            WRITEFILE “Date and Time of borrow:” & string(dnt)
            WRITEFILE “Name of the book borrowed: ” & bookName
            CLOSEFILE filename
        ENDFUNCTION
    ENDMODULE

    MODULE main
        IMPORT function
        CALL welcome()
        D_list ← CALL functions.one_d_list()
        CALL functions.display_book(D_list)
        DECLARE total_cost :  INTEGER
        DECLARE books_name : STRING
        DECLARE flag : BOOLEAN
        total_cost ← 0
        books_name ← ""
        flag ← TRUE
        WHILE flag = TRUE
            OUTPUT “Enter '1' to borrow a book, Enter '2' to return a book,
            Enter '3' to exit”
            value ← INPUT “Please enter a value: ”
```

```
            IF value = 1 THEN
                CALL functions.now_borrow()
                name ← INPUT "Enter the name of the person:"
                t_cost, Book_name = CALL continue_borrow ( D_list,
                functions.display_book, total_cost, name, books_name)
                CALL functions.b_bill(t_cost,name,Book_name)
            ELSE IF value = 2 THEN
                CALL functions.now_return()
                r_name ← INPUT "Enter the name of the person:"
                CALL functions.return_book()
            ELSE IF value = 3 THEN
                flag ← FALSE
                CALL functions.ty_three()
                BREAK
            ELSE
                CALL functions.valid_value()
            END IF
        END WHILE
        END MODULE
```

2.4 Data Structures

Data Structure is a collection of data type which is stored in an organized way which will allow the values to be accessed and modified effectively (McDonnell, 2019). In this project array list is used to store data, int is used to convert string number to integer and float is used to store integer values which are not whole numbers. Similarly, str is used to convert other data type to string. Other data structures like dictionary can also be used but array allows the appending of values which makes the work easy.

In array data structure, each data is assigned at an index of an array and another data is assigned at the next index and so on (Busbee, 2018). In *Figure 2*

given down below the data from text file is extracted and converted into a 1D list. At first, while extracting the data is formed in a 2D list and finally the 2D list is converted to 1D list.

```
def one_d_list():
    """Convert the text file to 1D list"""
    file = open("data.txt","r")
    dd_list = []
    data_list = []
    for line in file:
        line = line.replace("\n","")
        dd_list.append(line.split(","))
    for i in range(len(dd_list)):
        for j in range(len(dd_list[i])):
            data_list.append(dd_list[i][j])
    return data_list
```

Figure 2: Converting text file to 1D list

Using array list the data can be displayed in an effective way. For example, in *Figure 3* down below the 1D list is being used to display the data in a tabular format.

```
def display_book(d_list):
    """Displays values of dictionary in a table. Takes dictionary as prremeter"""
    '''Column name'''
    print("-------------------------------------------------------------------------)
    print("Book Id         Book Name            Author       Quantity   Price")
    print("-------------------------------------------------------------------------")
    '''Data of table'''
    print("   ",d_list[0],"     ",d_list[1]," ",d_list[2],"    ",d_list[3],"   ",d_list[4])
    print("   ",d_list[5],"     ",d_list[6]," ",d_list[7],"      ",d_list[8],"   ",d_list[9])
    print("   ",d_list[10],"    ",d_list[11],"",d_list[12],"     ",d_list[13],"   ",d_list[14])
    print("   ",d_list[15],"          ",d_list[16],"        ",d_list[17],"    ",d_list[18],"   ",d_list[19])
    print("   ",d_list[20],"        ",d_list[21],"    ",d_list[22],"    ",d_list[23],"   ",d_list[24])
    print("-------------------------------------------------------------------------")
```

Figure 3: Display data of list

Also, appending data is very easy. For example, in *Figure 4* down below the variable qnt1 is easily changed. This will allow the programmer to append data whenever they want.

```
    update textfile and data list when borrowed
qnt1 = data_list[3]
qnt2 = data_list[8]
qnt3 = data_list[13]
qnt4 = data_list[18]
qnt5 = data_list[23]
file = open("data.txt","w")
if bookID == 1:
    qnt1 = int(data_list[3]) - 1
    file.write("1,Nineteen Eighty-Four,George Orwell,"+str(qnt1)
```

Figure 4: Appending value of list

For int data type, in this project it is used to convert string numbers to integer value. Down below in *Figure 5* the element of data_list is converted to integer for further arithmetic operations.

```
if bookID == 1:
    qnt = int(data_list[3])
    add_book(bookID,qnt,data_list)
    display_book(data_list)
    break
```

Figure 5: int data type

Similarly, float data type, in the project it is used to calculate total cost (For visual representation look for *Figure 6*). The reason behind using this data type is the cost of the books are not whole numbers. So, in order to add floating-point numbers float data type should be used.

```
if YN == "y":
    book_cost , bookName = borrow_book(D_list)
    total_cost += float(book_cost)
    booksName += bookName+"\n"
else:
    ty_alt()
    break
```

Figure 6: Float data type

For str data type, any other data type can be converted to string. In this project this data type is used to convert date type, float type, integer type to string in order to write in the text file (for visual representation look for *Figure 7*). This is very important because other data type cannot be written in the text file.

```
def b_bill(total_cost,name,booksName):
    """Write bill in a qnique file"""
    dnt = datetime.datetime.now()
    minute = str(datetime.datetime.now().minute)
    second = str(datetime.datetime.now().second)
    microsecond = str(datetime.datetime.now().microsecond)
    randomValue = minute+second+microsecond
    file = open("borrow_bill/b_"+name+randomValue+".txt","w")
    file.write("Name of the Customer: "+name+"\n")
    file.write("Price of the book/books borrowed: $"+str(total_cost)+"\n")
    file.write("Date and Time of borrow: "+str(dnt)+"\n")
    file.write("Name of the book/books borrowed: \n"+booksName)
    file.close()
```

Figure 7: str data type

In other data structure like dictionary, a key is used to uniquely identify the values assigned to the key, the key can be in any order so this is called unordered collection of key-value pairs (Yildirim, 2021). Data dictionary is also known as associative arrays. While using this data structure the values cannot be changed which will cause trouble in this project because quantity must be reduced when borrowed.

3 Program

When the project is runed a welcome message is displayed. Along with the message the details of the books like book ID, book name, author, quantity and price are displayed in a table format. Also, a message is displayed which tells the users to enter ‘1’ to borrow, enter ‘2’ to return and enter ’3’ to exit the program. (Look for *Figure 8*)

```
************************************************
 Hello and Welcome to library management system
************************************************

-------------------------------------------------------------------------
Book Id         Book Name             Author       Quantity   Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      6     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-------------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 8: Welcome message and stock

For borrow, when ‘1’ is entered by the users a note is displayed which says ‘You will now borrow a book’. After this the users are supposed to enter their name. Then the users enter the book ID of the book. After assigning book ID the program checks whether the book is available or not if no error message is displayed. Else, an available message and the updated table of information about the book is displayed. If the users want to borrow another book they can type ‘y’ else type anything they want. (Look for *Figure 9*)

```
Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 1

*****************************
 You will now borrow a book
*****************************

Enter the name of the person: Aadarsha
Enter the book Id of the book you want to borrow:1

************************
 Book is available !!!
************************

-------------------------------------------------------------------------
Book Id         Book Name             Author       Quantity   Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-------------------------------------------------------------------------
If tou want to borrow a another book type 'y', else type any onthe word or letter :y
```

Figure 9: Borrow a book

If the user borrows another book, the users must assign the book ID of the other book they want to borrow. Again, the book available message and updated table is displayed. When any other value except ‘y’ is assigned the borrow process comes to the end and the bill is generated. (Look for *Figure 10*)

```
If tou want to borrow a another book type 'y', else type any onthe word or letter :y
Enter the book Id of the book you want to borrow:3

************************
 Book is available !!!
************************

-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-----------------------------------------------------------------------
If tou want to borrow a another book type 'y', else type any onthe word or letter :n

*********************************************************
            Thankyou for borrowing book/books
 Be sure to return it within 10 days or pay the fine
*********************************************************


Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 10: Borrow multiple books

When a borrow process is complete a bill is generated with a unique name using the costumers name and current minute, second and microsecond. In the bill, borrower name, total cost of books, date and time and book names are written. (Look for *Figure 11*)

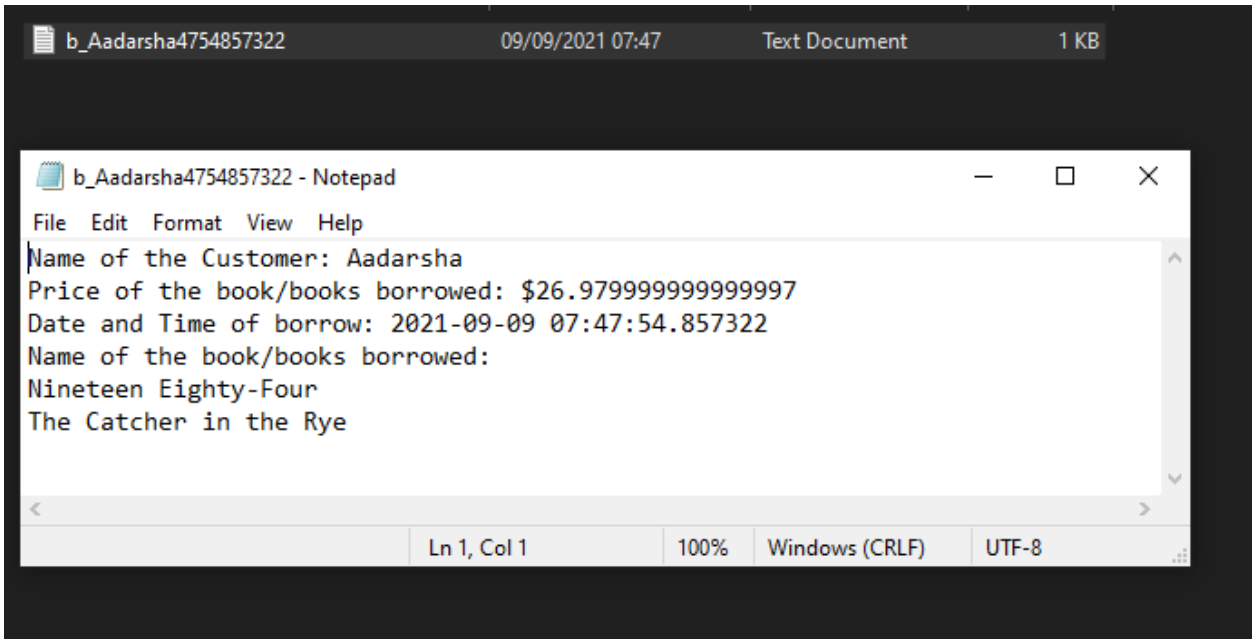

Figure 11: Bill of borrow

For return, when ‘2’ is entered by the users a note is displayed which says ‘You will now return a book’. After this the users are supposed to enter their name. Then the users enter the book ID of the book and the number of days the book was borrowed. After this the program will check if the given book ID is present in the stock or not. If book is not available an error message is displayed. Else the number of days the book was borrowed will be compared to 10. If days is less than 10 a thank you message is displayed and the bill is generated. (Look for *Figure 12*)

```
Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 2

****************************
 You will now return a book
****************************

Enter the name of the person: Aadarsha
Enter the book Id of the book you are going to return:2
Enter the number of days you borrowed the book:5

*********************************************************************************************************
                              Thankyou for returning the book
 If you want to return another book enter 2 down below, enter 1 to borrow and enter 3 to exit
*********************************************************************************************************

-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        6     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-----------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 12: Return with no fine

When the bill is generated after returning the book in time details like name of the customer, fine of $0, date and time and the name of the book is writen. (Look for *Figure 13*)

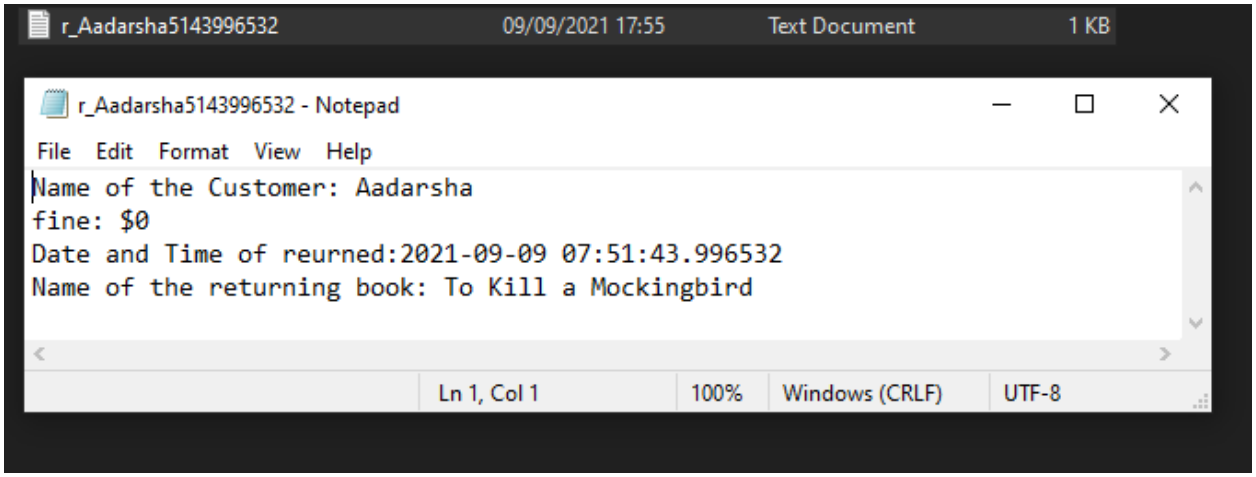

Figure 13: Return bill with no fine

When the book is returned late, thank you message and an additional message saying ‘you have returned the book late so you will have to pay the fine’ is displayed and the bill is generated. (Look for *Figure 14*)

```
Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 2

*****************************
 You will now return a book
*****************************

Enter the name of the person: aad
Enter the book Id of the book you are going to return:3
Enter the number of days you borrowed the book:20

*********************************************************************
                Thankyou for returning the book
 You have returned the book late so you will have to pay the fine
      If you want to return another book enter 2 down below
*********************************************************************

-----------------------------------------------------------------------
Book Id         Book Name             Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        6     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-----------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 14: Return with no fine

When the bill is generated after returning the book late details like name of the customer, fine of $2 per day, date and time and the name of the book is write. (Look for *Figure 15*)

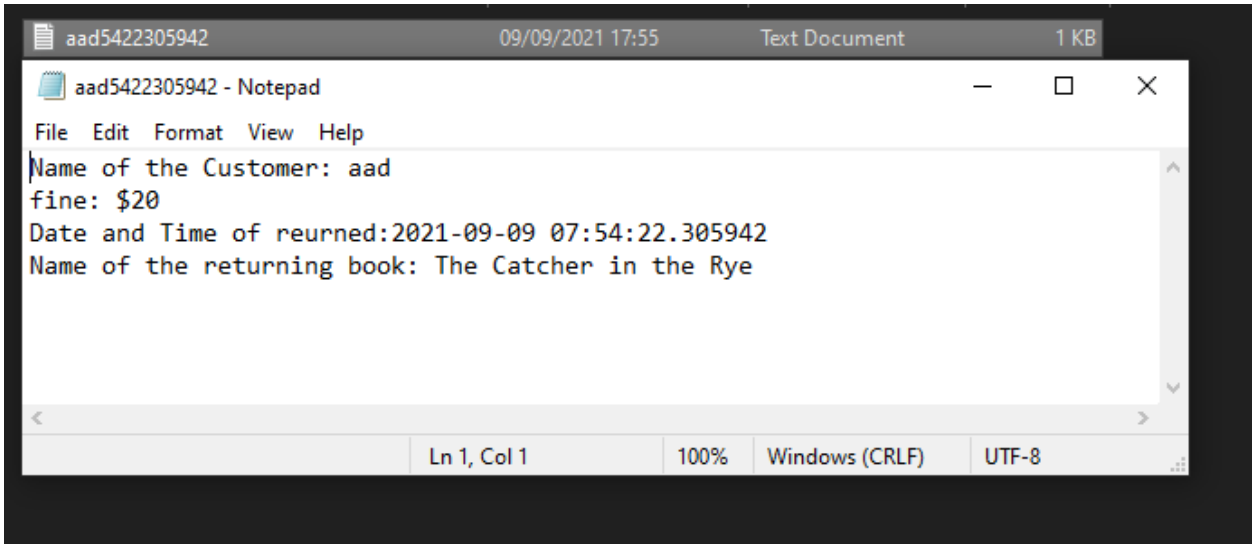

Figure 15: Return bill with no fine

To exit the program the value assigned should be '3'. If yes, a thank you message is displayed. (Look for *Figure 16*)

```
Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 3

**************************************************
 Thankyou for using my library management system
**************************************************

>>>
```

Figure 16:Exit the program

4 Testing

4.1 Test 1

Table 1: Test 1

Objectives	Show implementation of try, exception
Action	Provide invalid input while assigning value Assign “xyz”
Expected Results	Error message saying “Enter a integer value”
Actual Result	“Enter a integer value” is displayed
Conclusion	Test successful.

```
************************************************
 Hello and Welcome to library management system
************************************************

-------------------------------------------------------------------------
Book Id          Book Name             Author        Quantity    Price
-------------------------------------------------------------------------
    1        Nineteen Eighty-Four   George Orwell       6      $9.99
    2        To Kill a Mockingbird   Harper Lee         5      $17.99
    3      The Catcher in the Rye  J.D.Salinger        0      $16.99
    4             Beloved          Toni Morrison       4      $16.00
    5          Invinsible Man      Ralph Ellison       8      $14.00
-------------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: xyz

************************
 Enter a integer value
************************

Please enter a value:
```

Figure 17: Implementation of try, exception

4.2 Test 2

Table 2: Test 2

Objectives	Selection borrow and return option
Action	Provide negative value as input Provide non-existing value as input Assign -100 Assign 6
Expected Results	Error Message saying “The enterer value is invalid please try again” Should be displayed
Actual Result	“The enterer value is invalid please try again” is displayed
Conclusion	Test successful.

```
************************************************
 Hello and Welcome to library management system
************************************************

-------------------------------------------------------------------------
Book Id         Book Name             Author       Quantity    Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      6      $9.99
    2       To Kill a Mockingbird   Harper Lee        5      $17.99
    3      The Catcher in the Rye  J.D.Salinger       0      $16.99
    4             Beloved          Toni Morrison      4      $16.00
    5          Invinsible Man      Ralph Ellison      8      $14.00
-------------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: -100

*********************************************
 The entered value is invalid please try again
*********************************************


Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 18: Assigning negative value as input

```
************************************************
 Hello and Welcome to library management system
************************************************

-------------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      6     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      8     $14.00
-------------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 6

************************************************
 The entered value is invalid please try again
************************************************


Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 19:Assigning non-existing value as input

4.3 Test 3

Table 3: Test 3

Objectives	File generation of borrow
Action	Do a complete borrow process Borrow a book with book ID 1 Show quantity decrease in shell Show new generated txt file
Expected Results	The book should be borrowed successfully
Actual Result	The book was borrowed successfully
Conclusion	Test successful.

```
************************************************
 Hello and Welcome to library management system
************************************************

-------------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      6     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      8     $14.00
-------------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 1

*****************************
 You will now borrow a book
*****************************

Enter the name of the person: Aadarsha
Enter the book Id of the book you want to borrow:1

************************
 Book is available !!!
************************

-------------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-------------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      8     $14.00
-------------------------------------------------------------------------
If tou want to borrow a another book type 'y', else type any onthe word or letter :n
```

Figure 20: Borrow process part 1

```
*******************************************************
           Thankyou for borrowing book/books
 Be sure to return it within 10 days or pay the fine
*******************************************************


Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 21: Borrow process part 2

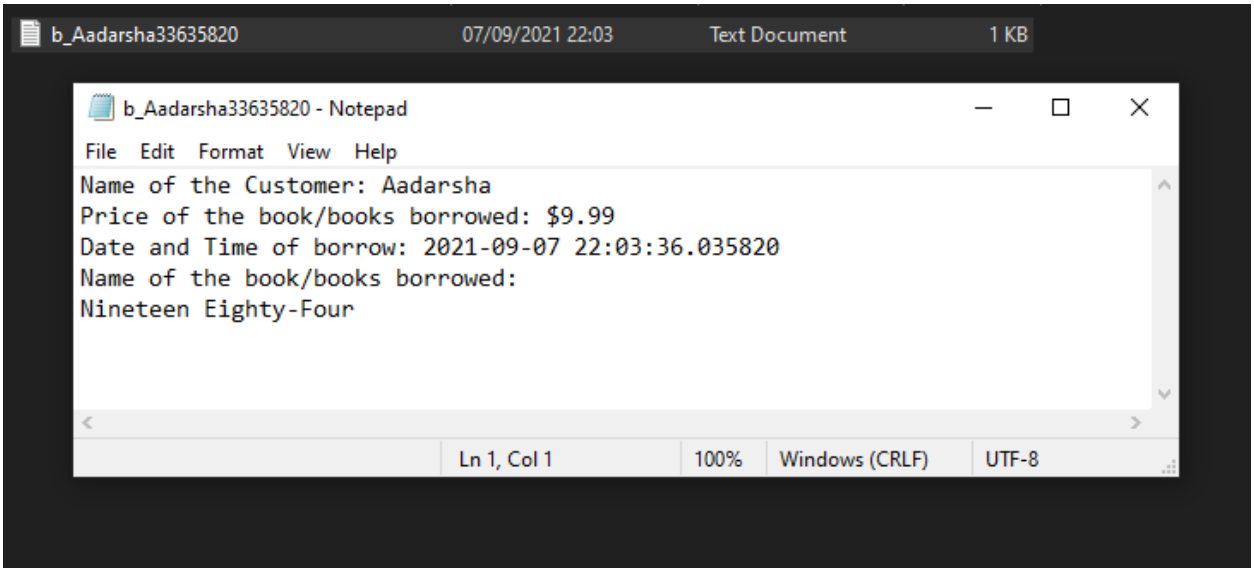

Figure 22: Bill generated after borrow

4.4 Test 4

Table 4: Test 4

Objectives	File generation of return
Action	Do a complete return process Return a book with book ID 3 Show quantity increase in shell Show new generated txt file
Expected Results	The book will be returned successfully
Actual Result	The book was returned successfully
Conclusion	Test successful.

Figure 23: Return process part 1

```
**************************************************
 Hello and Welcome to library management system
**************************************************

-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       0     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      8     $14.00
-----------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 2

*****************************
 You will now return a book
*****************************

Enter the name of the person: Aadarsha
Enter the book Id of the book you are going to return:3
Enter the number of days you borrowed the book:15

**********************************************************************
               Thankyou for returning the book
 You have returned the book late so you will have to pay the fine
      If you want to return another book enter 2 down below
**********************************************************************


-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      8     $14.00
-----------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value:
```

Figure 24: Return process part 2

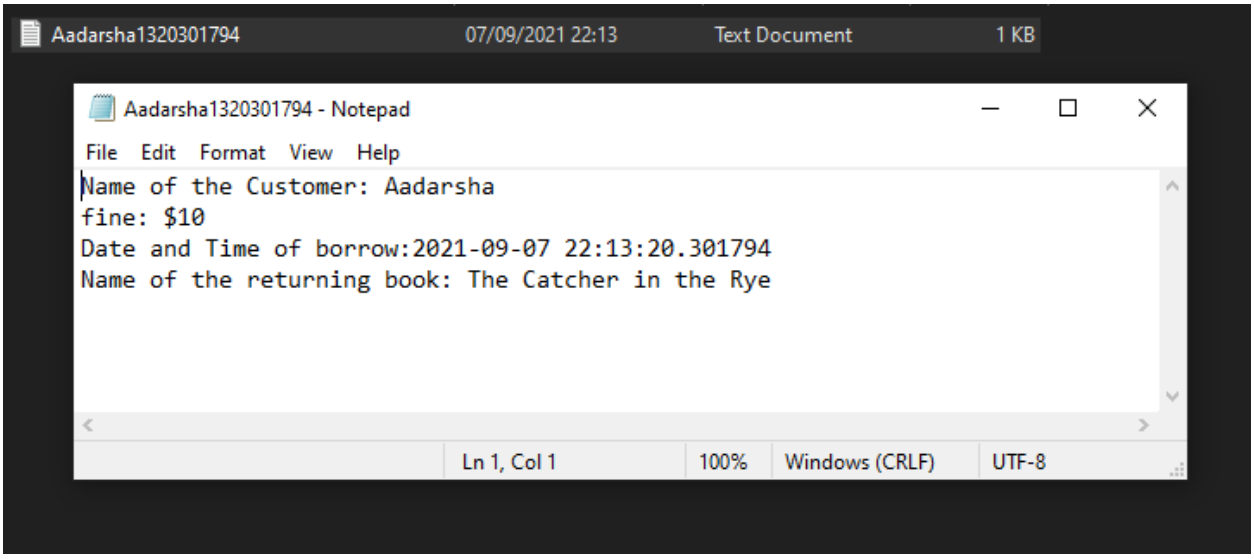

Figure 25: Bill generated after return

4.5 Test 5

Table 5: Test 5

Objectives	Show the update in stock
Action	Show stock before borrowing and returning a book Borrow book with book ID 5 Return book with book ID 1 Show stock after borrowing and returning a book
Expected Results	Quantity must decrease when borrowed and must increase when returned
Actual Result	Quantity is decrease when borrowed and increased when returned
Conclusion	Test successful.

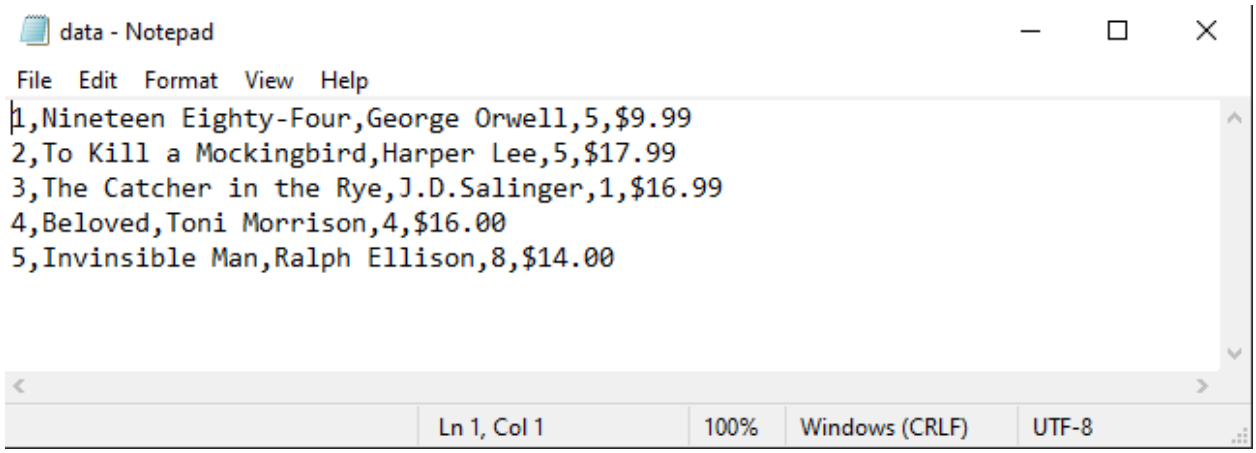

Figure 26: Stock before borrow and return process

```
************************************************
 Hello and Welcome to library management system
************************************************

-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      8     $14.00
-----------------------------------------------------------------------

Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 1

****************************
 You will now borrow a book
****************************

Enter the name of the person: aadarsha
Enter the book Id of the book you want to borrow:5

************************
 Book is available !!!
************************

-----------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-----------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      5     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4            Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man     Ralph Ellison      7     $14.00
-----------------------------------------------------------------------
```

Figure 27: Process of borrowing and returning part 1

```
If tou want to borrow a another book type 'y', else type any onthe word or letter :n

*******************************************************
            Thankyou for borrowing book/books
 Be sure to return it within 10 days or pay the fine
*******************************************************


Enter '1' to borrow a book
Enter '2' to return a book
Enter '3' to exit
Please enter a value: 2

****************************
 You will now return a book
****************************

Enter the name of the person: aadarsha
Enter the book Id of the book you are going to return:1
Enter the number of days you borrowed the book:9

*************************************************************************************************
                              Thankyou for returning the book
 If you want to return another book enter 2 down below, enter 1 to borrow and enter 3 to exit
*************************************************************************************************

-------------------------------------------------------------------
Book Id         Book Name            Author       Quantity   Price
-------------------------------------------------------------------
    1       Nineteen Eighty-Four   George Orwell      6     $9.99
    2       To Kill a Mockingbird   Harper Lee        5     $17.99
    3      The Catcher in the Rye  J.D.Salinger       1     $16.99
    4             Beloved          Toni Morrison      4     $16.00
    5          Invinsible Man      Ralph Ellison      7     $14.00
-------------------------------------------------------------------
```

Figure 28: Process of borrowing and returning part 2

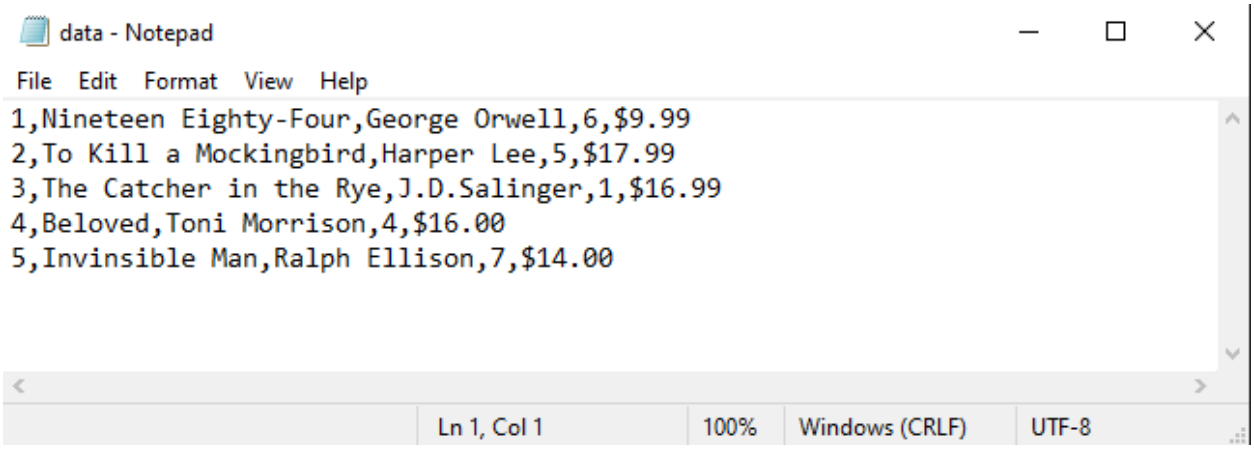

Figure 29: Stock after borrow and return process

5 Conclusion

For conclusion, throughout this module a new programming language python was taught. Other things like writing algorithm, flowchart and pseudocode were taught. In python basic programming, validation, iteration, data structures, file handling and exception handling were covered. These small programming concepts were used to create an advanced and responsive system. Without those concepts building an advanced system is impossible. For illustration, if a programmer is building a program to add two numbers. And if the user assigns a string value the whole program will crash. To avoid the program from crashing the programmer should know the use of exception handling. Which will lead to a proper program.

Moreover, file handling was very important for this project. This is because all work like reading and extracting data from a text file, writing in a new text file with unique name were covered by this topic. If a programmer lacks these skills the extracting and editing of data and bill generation will not work properly. For example, while extracting and editing the quantity the format of other values should be same. If not, the whole text file containing data will be deleted. To avoid this problem programmers should use proper file handling.

Previously, algorithm, flowchart and pseudocode of small basic program was made. But for this project algorithm, flowchart and pseudocode of an advanced system is made. Therefore, this was all for the library management system project.

6 Appendix

Appendix 1: main

```
import functions
functions.welcome() #Calling welcome function
D_list = functions.one_d_list() #Calling one_d_list function and assigning the returned value in D_list
functions.display_book(D_list) #Calling display_book function
total_cost = 0 # Initializing total_cost
books_name = "" # Initializing books_name
flag = True # Initializing flag
while flag == True:
    print("\nEnter '1' to borrow a book\nEnter '2' to return a book\nEnter '3' to exit") #Print statement to make the users aware about the work done by the program
    valid = True
    '''Try except'''
    while valid == True:
        try:
            value = int(input("Please enter a value: ")) #Taking value from the users
            break
        except:
            functions.invalid_dt() #Calling invalid_dt function
    if value == 1:
        '''For borrow'''
        functions.now_borrow() #Calling now_borrow function
        name = input("Enter the name of the person: ") #Taking name as input from users
        t_cost, Book_name = functions.continue_borrow(D_list,functions.display_book,total_cost,name,books_name) #Calling continue_borrow function amd assigning the returned value in t_cost, Book_name
        functions.b_bill(t_cost,name,Book_name) #Calling b_bill function
```

```
    elif value == 2:
        '''For return'''
        functions.now_return() #Calling now_return function
        r_name = input("Enter the name of the person: ") #Taking r_name as input from
users
        functions.return_book(D_list,functions.display_book,r_name) #Calling return_book
function
    elif value == 3:
        '''For exit'''
        flag == False
        functions.ty_three() #Calling ty_three function
        break
    else:
        '''if book ID is not 1,2 or 3'''
        functions.valid_value() #Calling valid_value function
```

Appendix 2: functions

```
        import datetime
def welcome():
    """Function to Dislpay welcome message"""
    print("\n**********************************************\n Hello and Welcome to library
management system\n**********************************************\n") #print statement
def valid_value():
    """Function to Dislpay enter valid value message"""
    print("\n**********************************************\n The entered value is invalid please
try again\n**********************************************\n") #print statement
def ty_three():
    """Function to Dislpay thank you message"""
    print("\n**********************************************\n Thankyou for using my library
management system\n**********************************************\n") #print statement
def now_borrow():
    """Function to Dislpay now you will  borrow message"""
```

```
    print("\n*****************************\n You will now borrow a
book\n*****************************\n") #print statement
def now_return():
    """ Function toDislpay now you will  return message"""
    print("\n*****************************\n You will now return a
book\n*****************************\n") #print statement
def error_book():
    """Function to Display invalid book id"""
    print("\n***********************************\n Please provide a valid Book ID
!!!\n***********************************\n") #print statement
    print("\n***********************************\n Try again with a different book
ID\n***********************************\n") #print statement
def n_avilable_book():
    """Function to Display out of stock"""
    print("\n************************\n Book out of stock !!!\n************************\n") #print
statement
def avilable_display():
    """Function to Dislpay Book is available """
    print("\n************************\n Book is available !!!\n************************\n") #print
statement
def ty_alt():
    """Function to Display thank you message after borrowing a book"""
    print("\n*****************************************************\n           Thankyou for
borrowing book/books\n Be sure to return it within 10 days or pay the
fine\n*****************************************************\n") #print statement
def fine():
    """Function to Display thank you message after returning a book late"""
    print("\n*********************************************************************\n
Thankyou for returning the book\n You have returned the book late so you will have to
pay the fine\n      If you want to return another book enter 2 down
below\n*********************************************************************\n") #print statement
```

```
def thankyou():
    """Function to Display thank you message after returning a book in time"""

print("\n*********************************************************************************************
***\n                    Thankyou for returning the book\n If you want to return another
book enter 2 down below, enter 1 to borrow and enter 3 to
exit\n*********************************************************************************************
\n") #print statement
def invalid_dt_bookid():
    """Function which Displayes ID should be a integer between 1-5 when execption
occer while assigning book ID"""
    print("\n***********************************\n ID should be a integer between 1-
5\n***********************************\n") #print statement
def invalid_dt():
    """Function which Displayes when Enter a integer value when execption occer while
assigning value"""
    print("\n**********************\n Enter a integer value\n**********************\n") #print
statement

def one_d_list():
    """Convert the text file to 1D list"""
    file = open("data.txt","r")
    '''Initializing lists'''
    dd_list = []
    data_list = []
    for line in file:
        line = line.replace("\n","") #Replacing '\n' with empty siring
        dd_list.append(line.split(",")) #Converting to 2D list
    '''converting 2D list to 1D list'''
    for i in range(len(dd_list)):
        for j in range(len(dd_list[i])):
```

```
            data_list.append(dd_list[i][j])
    return data_list
def display_book(d_list):
    """Displays values of dictionary in a table. Takes dictionary as prremeter"""
    '''Column name'''
    print("-------------------------------------------------------------------------") #Border line
    print("Book Id         Book Name            Author       Quantity    Price") #Column name
    print("-------------------------------------------------------------------------")#Border line
    '''Data of table'''
    print("   ",d_list[0],"     ",d_list[1]," ",d_list[2],"    ",d_list[3],"   ",d_list[4]) #details of with
book ID 1
    print("   ",d_list[5],"     ",d_list[6]," ",d_list[7],"      ",d_list[8],"   ",d_list[9]) #details of with
book ID 2
    print("   ",d_list[10],"    ",d_list[11],"",d_list[12],"     ",d_list[13],"   ",d_list[14]) #details of
with book ID 3
    print("   ",d_list[15],"          ",d_list[16],"        ",d_list[17],"    ",d_list[18],"   ",d_list[19])
#details of with book ID 5
    print("   ",d_list[20],"        ",d_list[21],"    ",d_list[22],"    ",d_list[23],"
",d_list[24])#details of with book ID 5
    print("-------------------------------------------------------------------------")#Border line
def continue_borrow(D_list,display_book,total_cost,name,booksName):
    """To calculate total cost and all the books borrowed"""
    add = True #Initializing add
    book_cost , bookName = borrow_book(D_list) #Calling borrow_book function and
assigning the returned value to book_cost and bookName
    total_cost += float(book_cost) #Adding book_cost of borrowed book and assigning to
total_cost
    booksName += bookName+"\n" #Joining bookName of borrowed book and assigning
it to booksName
    while add == True:
```

```
        YN = input("If tou want to borrow a another book type 'y', else type any onthe word or letter :")#Taking YN as input from the users
        if YN == "y":
            book_cost , bookName = borrow_book(D_list) #Calling borrow_book function and assigning the returned value to book_cost and bookName
            total_cost += float(book_cost) #Adding book_cost of borrowed book to total_cost
            booksName += bookName+"\n" #Joining bookName of borrowed book to booksName
        else:
            ty_alt() #alling ty_alt function
            break
    return total_cost, booksName
def borrow_book(data_list):
    """Function which decreases quantity, displays books, returns price and bookName"""
    '''Initializing price, bookName, flag, vld, vld1'''
    price = 0
    bookName = ""
    flag = True
    vld = True
    vld1 = True
    '''Exception handling'''
    while vld == True:
        try:
            bookID = int(input("Enter the book Id of the book you want to borrow:")) #Taking book ID as input from the users
            break
        except:
            invalid_dt_bookid()#Calling invalid_dt function
    while flag == True:
        if bookID == 1:
            qnt = int(data_list[3]) #Assigning the 3 index data_list to qnt
```

```
        if qnt > 0:
            avilable_display() #Calling avalable_display function
            remove_book(bookID,qnt,data_list) #Calling remove_book function
            display_book(data_list) #Calling display_book functin
            price = cost(data_list,bookID) #Calling cost function
            bookName = book_name(data_list,bookID) #Calling book_name function
            break
        else:
            n_avilable_book() #Calling n_avilable_book function
            while vld1 == True:
                try:
                    bookID = int(input("Enter the book Id of a another book you want to
borrow:")) #Taking book ID as input from the users
                    break
                except:
                    invalid_dt_bookid()#Calling invalid_dt function
    elif bookID == 2:
        qnt = int(data_list[8]) #Assigning the 8 index data_list to qnt
        if qnt > 0:
            avilable_display()#Calling avalable_display function
            remove_book(bookID,qnt,data_list) #Calling remove_book function
            display_book(data_list) #Calling display_book functin
            price = cost(data_list,bookID) #Calling cost function
            bookName = book_name(data_list,bookID)#Calling book_name function
            break
        else:
            n_avilable_book() #Calling n_avilable_book function
            while vld1 == True:
                try:
                    bookID = int(input("Enter the book Id of a another book you want to
borrow:"))#Taking book ID as input from the users
```

```
                    break
                except:
                    invalid_dt_bookid() #Calling invalid_dt function
        elif bookID == 3:
            qnt = int(data_list[13])#Assigning the 13 index data_list to qnt
            if qnt > 0:
                avilable_display()#Calling avalable_display function
                remove_book(bookID,qnt,data_list) #Calling remove_book function
                display_book(data_list)#Calling display_book functin
                price = cost(data_list,bookID)#Calling cost function
                bookName = book_name(data_list,bookID)#Calling book_name function
                break
            else:
                n_avilable_book() #Calling n_avilable_book function
                while vld1 == True:
                    try:
                        bookID = int(input("Enter the book Id of a another book you want to
borrow:")) #Taking book ID as input from the users
                        break
                    except:
                        invalid_dt_bookid() #Calling invalid_dt function
        elif bookID == 4:
            qnt = int(data_list[18])#Assigning the 18 index data_list to qnt
            if qnt > 0:
                avilable_display()#Calling avalable_display function
                remove_book(bookID,qnt,data_list)#Calling remove_book function
                display_book(data_list)#Calling display_book functin
                price = cost(data_list,bookID) #Calling cost function
                bookName = book_name(data_list,bookID) #Calling book_name function
                break
            else:
```

```
            n_avilable_book() #Calling n_avilable_book function
            while vld1 == True:
                try:
                    bookID = int(input("Enter the book Id of a another book you want to
borrow:")) #Taking book ID as input from the users
                    break
                except:
                    invalid_dt_bookid() #Calling invalid_dt function
        elif bookID == 5:
            qnt = int(data_list[23])#Assigning the 23 index data_list to qnt
            if qnt > 0:
                avilable_display() #Calling avalable_display function
                remove_book(bookID,qnt,data_list)#Calling remove_book function
                display_book(data_list) #Calling display_book functin
                price = cost(data_list,bookID) #Calling cost function
                bookName = book_name(data_list,bookID)#Calling book_name function
                break
            else:
                n_avilable_book() #Calling n_avilable_book function
                while vld1 == True:
                    try:
                        bookID = int(input("Enter the book Id of a another book you want to
borrow:"))#Taking book ID as input from the users
                        break
                    except:
                        invalid_dt_bookid()#Calling invalid_dt function
        else:
            error_book()
            break
    return price , bookName
```

```
def remove_book(bookID,quantity,data_list):
    """Update textfile and data list when borrowed"""
    '''Initializing qnt1,qnt2,qnt3,qnt4,qnt4'''
    qnt1 = data_list[3]
    qnt2 = data_list[8]
    qnt3 = data_list[13]
    qnt4 = data_list[18]
    qnt5 = data_list[23]
    file = open("data.txt","w")
    if bookID == 1:
        qnt1 = int(data_list[3]) - 1 #Updating variable qnt1
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+str(qnt1)+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 2:
        qnt2 = int(data_list[8]) - 1 #Updating variable qnt2
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+str(qnt2)+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 3:
        qnt3 = int(data_list[13]) - 1 #Updating variable qnt3
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+str(qnt3)+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 4:
```

```
        qnt4 = int(data_list[18]) - 1 #Updating variable qnt4
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+str(qnt4)+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 5:
        qnt5 = int(data_list[23]) - 1 #Updating variable qnt5
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+str(qnt5)+",$14.00")
    file.close()
    fnl_qnt = quantity-1 #Calculating quantity afrer borrow
    if bookID == 1:
        data_list[3] = str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
    elif bookID == 2:
        data_list[8] = str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
    elif bookID == 3:
        data_list[13] =  str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
    elif bookID == 4:
        data_list[18] = str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
    elif bookID == 5:
        data_list[23] = str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
def cost(data_list,bookID):
    """Retuens the price of the book borrowed"""
    if bookID == 1:
        price = data_list[4] #Assigning the 4 index data_list to price
        price = price.replace("$","") #Replacing $ with empty string
    elif bookID == 2:
```

```
        price = data_list[9] #Assigning the 9 index data_list to price
        price = price.replace("$","") #Replacing $ with empty string
    elif bookID == 3:
        price = data_list[14] #Assigning the 14 index data_list to price
        price = price.replace("$","") #Replacing $ with empty string
    elif bookID == 4:
        price = data_list[19]#Assigning the 19 index data_list to price
        price = price.replace("$","") #Replacing $ with empty string
    elif bookID == 5:
        price = data_list[24]#Assigning the 24 index data_list to price
        price = price.replace("$","") #Replacing $ with empty string
    return price
def book_name(data_list,bookID):
    """Returns book name which is borrowed"""
    bookName = "" #Initializing bookName
    if bookID == 1:
        bookName = data_list[1]#Assigning the 1 index data_list to bookName
    elif bookID == 2:
        bookName = data_list[6]#Assigning the 6 index data_list to bookName
    elif bookID == 3:
        bookName = data_list[11]#Assigning the 11 index data_list to bookName
    elif bookID == 4:
        bookName = data_list[16]#Assigning the 16 index data_list to bookName
    elif bookID == 5:
        bookName = data_list[21]#Assigning the 21 index data_list to bookName
    return bookName
def b_bill(total_cost,name,booksName):
    """Write bill in a qnique file"""
    dnt = datetime.datetime.now()#Returns current date and time
    '''Creatimg unique name for file generation'''
    minute = str(datetime.datetime.now().minute)
```

```
    second = str(datetime.datetime.now().second)
    microsecond = str(datetime.datetime.now().microsecond)
    randomValue = minute+second+microsecond
    '''File handling'''
    file = open("borrow_bill/b_"+name+randomValue+".txt","w")
    file.write("Name of the Customer: "+name+"\n")
    file.write("Price of the book/books borrowed: $"+str(total_cost)+"\n")
    file.write("Date and Time of borrow: "+str(dnt)+"\n")
    file.write("Name of the book/books borrowed: \n"+booksName)
    file.close()
def return_book(data_list, display_book,name):
    """Function to take book ID. If valid ID is entered call other functions else call error
function"""
    '''initializing boolean variables'''
    flag = True
    vld = True
    vld1 = True
    '''Exception handling'''
    while vld == True:
        try:
            bookID = int(input("Enter the book Id of the book you are going to return:"))
#Taking bookID as input from users
            break
        except:
            invalid_dt_bookid()#Calling invalid_dt_bookid
    while vld == True:
        try:
            days = int(input("Enter the number of days you borrowed the book:")) #Taking
days as input from users
            break
        except:
```

```
            invalid_dt() #Calling invalid_dt function
    fine_yn(days,data_list,name,bookID)#Calling fine_yn function
    while flag == True:
        if bookID == 1:
            qnt = int(data_list[3]) #Assigning the 3 index data_list to qnt
            add_book(bookID,qnt,data_list) #Calling add_book function
            display_book(data_list) #Calling display_book functin
            break
        elif bookID == 2:
            qnt = int(data_list[8]) #Assigning the 8 index data_list to qnt
            add_book(bookID,qnt,data_list)#Calling add_book function
            display_book(data_list)#Calling display_book functin
            break
        elif bookID == 3:
            qnt = int(data_list[13])#Assigning the 13 index data_list to qnt
            add_book(bookID,qnt,data_list)#Calling add_book function
            display_book(data_list)#Calling display_book functin
            break
        elif bookID == 4:
            qnt = int(data_list[18])#Assigning the 18 index data_list to qnt
            add_book(bookID,qnt,data_list)#Calling add_book function
            display_book(data_list)#Calling display_book functin
            break
        elif bookID == 5:
            qnt = int(data_list[23]) #Assigning the 23 index data_list to qnt
            add_book(bookID,qnt,data_list)#Calling add_book function
            display_book(data_list)#Calling display_book functin
            break
        else:
            error_book()#Calling error_book functin
            break
```

```
def add_book(bookID,quantity,data_list):
    """Function to update textfile and data list when borrowed"""
    '''Initializing qnt1,qnt2,qnt3,qnt4,qnt4'''
    qnt1 = data_list[3]
    qnt2 = data_list[8]
    qnt3 = data_list[13]
    qnt4 = data_list[18]
    qnt5 = data_list[23]
    '''file handling'''
    file = open("data.txt","w")
    if bookID == 1:
        qnt1 = int(data_list[3]) + 1 #Updating variable qnt1
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+str(qnt1)+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 2:
        qnt2 = int(data_list[8]) + 1 #Updating variable qnt2
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+str(qnt2)+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 3:
        qnt3 = int(data_list[13]) + 1 #Updating variable qnt3
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+str(qnt3)+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
```

```
    elif bookID == 4:
        qnt4 = int(data_list[18]) + 1 #Updating variable qnt4
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+str(qnt4)+",$16.00\n5,Invinsible Man,Ralph Ellison,"+qnt5+",$14.00")
    elif bookID == 5:
        qnt5 = int(data_list[23]) + 1 #Updating variable qnt5
        '''Writing updated variable in txt file'''
        file.write("1,Nineteen Eighty-Four,George Orwell,"+qnt1+",$9.99\n2,To Kill a
Mockingbird,Harper Lee,"+qnt2+",$17.99\n3,The Catcher in the
Rye,J.D.Salinger,"+qnt3+",$16.99\n4,Beloved,Toni
Morrison,"+qnt4+",$16.00\n5,Invinsible Man,Ralph Ellison,"+str(qnt5)+",$14.00")
    file.close()
    fnl_qnt = quantity + 1 #Calculating quantity afrer return
    if bookID == 1:
        data_list[3] = str(fnl_qnt)#Assigning fnl_qnt to the 3 index data_list
    elif bookID == 2:
        data_list[8] = str(fnl_qnt)#Assigning fnl_qnt to the 8 index data_list
    elif bookID == 3:
        data_list[13] =  str(fnl_qnt)#Assigning fnl_qnt to the 13 index data_list
    elif bookID == 4:
        data_list[18] = str(fnl_qnt)#Assigning fnl_qnt to the 18 index data_list
    elif bookID == 5:
        data_list[23] = str(fnl_qnt)#Assigning fnl_qnt to the 23 index data_list
def fine_yn(days,data_list,name,bookID):
    """Finction to determine fine needs to be payed or not"""
    if days > 10:
        r_f_bill(name,data_list,bookID,days)#Calling r_f_bill function
        fine()#Calling fine function
```

```
    else:
        r_nf_bill(name,data_list,bookID) #Calling r_nf_bill function
        thankyou() #Calling thankyou function
def r_f_bill(name,data_list,bookID,days):
    """Function to write in the txt file if returned late"""
    bookName = book_name(data_list,bookID) #Calling book_name function
    fine_amt = (days-10) * 2 #calculating fine amount
    dnt = datetime.datetime.now() #Current returns date and time
    '''Creatimg unique name for file generation'''
    minute = str(datetime.datetime.now().minute)
    second = str(datetime.datetime.now().second)
    microsecond = str(datetime.datetime.now().microsecond)
    randomValue = minute+second+microsecond
    '''File handling'''
    file = open("retuen_bill/"+name+randomValue+".txt","w")
    file.write("Name of the Customer: "+name+"\n")
    file.write("fine: $"+str(fine_amt)+"\n")
    file.write("Date and Time of return:"+str(dnt)+"\n")
    file.write("Name of the returning book: "+bookName)
    file.close()
def r_nf_bill(name,data_list,bookID):
    """Function to write in the txt file if returned in time"""
    bookName = book_name(data_list,bookID)#Calling book_name function
    fine_amt = 0 # fine is 0 because book is returned in time
    dnt = datetime.datetime.now() #Current returns date and time
    '''Creatimg unique name for file generation'''
    minute = str(datetime.datetime.now().minute)
    second = str(datetime.datetime.now().second)
    microsecond = str(datetime.datetime.now().microsecond)
    randomValue = minute+second+microsecond
    '''File handling'''
```

```
    file = open("retuen_bill/r_"+name+randomValue+".txt","w")
    file.write("Name of the Customer: "+name+"\n")
    file.write("fine: $"+str(fine_amt)+"\n")
    file.write("Date and Time of return:"+str(dnt)+"\n")
    file.write("Name of the returning book: "+bookName)
    file.close()
```

7 References

Aggarwal, N., 2020. *GeeksforGeeks.* [Online]
Available at: https://www.geeksforgeeks.org/an-introduction-to-flowcharts/
[Accessed 8 September 2021].

Busbee, K., 2018. *Arrays and Lists.* [Online]
Available at: https://press.rebus.community/programmingfundamentals/chapter/arrays-and-lists/
[Accessed 9 September 2021].

GeeksforFeeks, 2021. *Python Programming Language.* [Online]
Available at: https://www.geeksforgeeks.org/python-programming-language/
[Accessed 9 September 2021].

McDonnell, M., 2019. *Data Types and Data Structures.* [Online]
Available at: https://www.integralist.co.uk/posts/data-types-and-data-structures/#data-structures
[Accessed 9 September 2021].

Yildirim, S., 2021. *4 Must-Know Features of Python Dictionaries.* [Online]
Available at: https://towardsdatascience.com/4-must-know-features-of-python-dictionaries-d62af8c22fd2
[Accessed 9 September 2021].