INFORMATICS INSTITUTE OF TECHNOLOGY

Foundation Certificate in Higher Education

Module : DOC334 Introduction to Programming in Python – P2

Module Leader : Mr. Sudarshana Welihinda

Assignment Number : 1

Assignment type : Individual Coursework (ICW)

Issue Date : 14th March 2022

Hand–in – Date : 18thApril 2022

Done by :

Student Name	**Student ID**
Ahmed Aamil	20211096

I. Abstract

This report describes a solution for a bookstore facing issues with book management by explaining the constructed source code, test cases, and ultimately screenshots of the working program.

II. Acknowledgement

Completing this project would not be possible without the support and guidance of module lecturers. I would like to extend my sincere thanks to all of them.

I am highly indebted to Mr. Sudarshana Welihinda, Mr. Nishan Saliya, and Ms. Keerthiga Rajenthiram for their guidance and supervision in providing the necessary information and resources to complete the coursework.

Table of Contents

III. List of Figures v

IV. List of Tables vi

1.Problem Statement 1

1.1. Problem Understanding 2

1.2. Solution Outline 2

1.3. Assumptions and distinctive notes 2

1.3.1. External Package notes 3

2.Program Codes 4

3.Test Cases and Screen Shots 24

3.1. Adding Books 24

3.2. Deleting Books 27

3.3. Editing Books 28

3.4. Book Chapter View 30

3.5. Search Books 31

III. List of Figures

Figure 1 Subject Table 2
Figure 2 Adding Book Sample 1 25
Figure 3Adding Book Sample 2 25
Figure 4 Adding Book Sample 3 26
Figure 5 phpMyAdmin Results for Adding Books 26
Figure 6 Deleting Books Sample 1 27
Figure 7 phpMyAdmin Results for Deleting Books 27
Figure 8 Editing Books Sample 1 28
Figure 9 phpMyAdmin Results for Editing Books 1 29
Figure 10 Editing Books Sample 2 29
Figure 11 phpMyAdmin Results for Editing Books 2 29
Figure 12 Book Chapter View sample 1 30
Figure 13 Search Books Sample 1 31
Figure 14 Search Books Sample 2 32
Figure 15 Search Books Sample 3 32
Figure 16 Search Books Sample 4 32

IV. List of Tables

Table 1 Test Cases for Adding Books 25
Table 2 Test Cases for Deleting Books.....27
Table 3 Test Cases for Editing Books.....28
Table 4 Test Cases for Book Chapter View.....30
Table 5 Test Cases for Searching Books 31

1. Problem Statement

You are to create a **console Python 3.x program** which will allow users to demonstrate a small program for a library.
Currently all books available at ABC books store are managed using a manual process. Users should search for books from the racks which is a time taking process. With the increase of software development in every field developing book management software will be helpful task. It is needed to develop a system which is useful for maintain books in a procedure-oriented manner. Users should be able to easily search for books from index page and search for books to find their required book.

Design and develop fully transactional console-based application which has the following functionalities.

Books: Facility to create information related each book in the bookstore.
Input: Book No (Primary Key), Title, Subject Code, Author, Publisher, Price, Location, and any other information you think it is useful to have.

Books Chapters: Facility to create books chapter information. For every book available on the site has to have a clear view to read each chapter using books chapter's view.
Input: Book No, Chapter No, Title, starting page no, Ending page no

Subjects: Facility to create subjects. Books available on the site should sub divided into different category which will help users to easily find required category books.
Input: Subject Code, Name

Books Query: Direct search is needed to find books available on the site. Users must be able to search by book number, Book Title, Author, **or** Publisher.
Add / Edit / Delete Books: Facility should be provided add, edit, or delete information related to a book.

You are to create a Python 3.x program which mimics the above requirements. Your program **must** run in the console. GUI base program will result **ZERO** marks. You can decide how the program menu and the console interface will look like, based on the above requirements. You must clearly state your assumptions if you have any.

1.1. Problem Understanding

The problem stated above is that ABC Bookstore has been managing their books and information by a manual operation. The fact that it is a time-taking process is indicated, on top of that, there would be some issues when inserting, modifying, or removing data related to books. Furthermore, proper management of all of this information is challenging, thus bookstore consumers must manually search every rack for books.

As a result, developing book management software that is procedure-oriented, user-friendly, and capable of recording all accessible data would be beneficial to the bookstore.

1.2. Solution Outline

The solution to the problem statement above stated is an application that would mimic the mentioned operations, such as adding, editing, and deleting books, as well as book query searching by book number, title, author, publisher, and searching books by topic category. Users could also view the desired chapter by using chapter view.

A database will be used to hold all data related to books, book chapters, and book subjects by creating separate relational data sets. Users will be able to access the features through the main menu, which is a console-based interface, which could be used to navigate to any operation as required. Therefore, the proposed solution would increase users' efficiency to manage books with this library system.

1.3. Assumptions and distinctive notes

The following assumptions have been carried out to complete this project in order to evaluate the program's basic functionality.

- ABC bookstore will have a total of six/seven book subjects and subject codes. Other subjects would be categorized as 'OTHER' subject codes. Moreover, to avoid primary key constraints, the Subject category table has been hardcoded in the Subject Table.

SubjectCode	Name
ARTS	ART
ENG	ENGLISH
FAN	FANTASY
HIS	HISTORY
LIFE	LIFESTYLE

Figure 1 Subject Table

- A chapter view of a particular book would also be constructed for testing purposes, based on sample data in the form of a.TXT file.
- Any inputs from users may or may not be validated. Therefore, an assumption has been made that the user input data may be accurate and may contain dummy values for testing purposes (Example: Total Chapters, Location etc.).

To run this program, follow these instructions,

- **Step 01:** Import the library.sql file from the database folder into phpMyAdmin by creating a new database or selecting an existing one. However, it is **recommended** to run the.py files in the database folder in the following order: Book, BookChapter, and Subject.
- **Step 02:** Open the main.py source code to execute the program.

  Due to various line-extension issues and indentation issues that cause the code to jump to the next line, please run the code through the original .py file rather than copying it from the document.

1.3.1. External Package notes

The external package that was used in the development of this program is mysql.connector., which is used to execute MYSQL queries and relate with the database.

To import mysql.connector,

First, ensure that Xampp is installed and that the Apache and MySQL servers are operational with administration. (https://www.apachefriends.org/download.html)

- In Administrator mode, launch the command console (Check that you are connected to the internet)
- Type the below command and press enter key.

  Windows – pip install mysql-connector-python or pip3 install mysql-connector-python

Once installed, mysql.connector is active and ready to use.

2. Program Codes

main.py: This is the main source code file, containing the command line menu program.

```
# START


# Importing Modules and Packages

import mysql.connector

import time

import dbms

### import sys


# Initializing Variables (Global etc.)

option = ''

answer = ''

fo = 0


# Opening a .TXT file

fo = open('ChapterView.txt','r+')


# Input & Process

print ("\n\t\tWELCOME TO ABC BOOK STORE LIBRARY SYSTEM")


# Defining MENU Function for the Console Interface

def menu():
```

```
    "console interface"
    print("————————————————————————————————————————————————————————————————————————————————")
    print('''1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT''')
    print("————————————————————————————————————————————————————————————————————————————————")

    # Read User Input
    option = input("\nEnter Requested Option (1/2/3/4/5) : ")

    #Process
    print()
    if option == '1':
        dbms.search()
    elif option == '2':
        dbms.add()
    elif option == '3':
        dbms.edit()
    elif option == '4':
        bNumber = input("Enter the Book Number : ")
```

```
            dbms.dlt(bNumber)
        elif option == '5':
            dbms.chapview()
        elif option == '6':
            print ("You're about to exit.")
        else:
            print ("\n\nOops!, Wrong Option....Please try again")
            menu()


    # Output
    # Calling the MENU
    menu()


    while True:
        print()
        answer = input("Do You want to EXIT the System ? (YES or NO) ")
        answer = answer.upper()


        if answer == 'YES':
            print()
            time.sleep(1)
            print ("Thanks For using ABC Book Store.")
            break
        elif answer == 'NO':
```

```
        print ("\n\t\tWELCOME BACK TO ABC BOOK STORE LIBRARY SYSTEM")
        menu()
        # User Input Validation
    else:
        print ("\n<<INVALID USER INPUT>>")
        print()
        answer = input("Do You want to EXIT the System ? (YES or NO) ")

# Disconnection(s)
fo.close()

# STOP
```

dbms.py : This module file contains all the operations such as, adding, deleting, updating etc.

```
# START

# Importing Modules and Packages
import mysql.connector
from mysql.connector import Error
import time

# Initializing Variables (Global etc.)
opt = ''
list1 = []
fo = 0

# Opening a .TXT file
fo = open('ChapterView.txt', 'r+')

# Open database connection
conDict = {"host":"localhost",
          "database": "library",
          "user":"root",
          "password":""}

try:
    db = mysql.connector.connect (**conDict)
    if db.is_connected():

# Executing SQL queries by Defining Functions

# Defining Search Function
        def search():
            'This Function is to Search Books'
```

```
            print("\nYou are about to search for Books")
            print ()
            time.sleep(1)
            print ("'Choose an option to start the query
    1. Book Number
    2. Book Title
    3. Author
    4. Publisher
    5. Subject category
    6. Select All Books"')
            print()
            # Query Option INPUT
            opt = input("Choose an option (1/2/3/4/5/6) : ")
            if opt == '1' or opt == '2' or opt == '3' or opt == '4' or opt == '5' or opt == '6'
    :
                if opt == '1':
                    bno =  input("Enter the Book Number : ")
                    cursor = db.cursor()
                    sql = ('SELECT * FROM book WHERE BookNo = %s')
                    data = bno
                    cursor.execute(sql,[data])
                    result = cursor.fetchall()
                    print ('\n',cursor.rowcount," Book Search found")
                    # Output
                    for i in result:
                        print ("————————————————————————————————————————————————")
                        print ('<<<Book Number>>> : ',i[0])
                        print ('Book Title : ', i[1],)
                        print ('Subject Code : ', i[2])
                        print ('Author : ', i[3])
                        print ('Publisher : ', i[4])
                        print ('Price : ', i[5])
                        print ('Location : ', i[6])
```

```
                print ("————————————————————————————————————————————————
")

            elif opt == '2':
                bName = input("Enter the Book Title : ")
                cursor = db.cursor()
                sql = (f"SELECT * FROM book WHERE Title LIKE '%{bName}%'")
                cursor.execute(sql)
                result = cursor.fetchall()
                print ('\n',cursor.rowcount," Book Search found")
                for i in result:
                    print
("_____________________________________________________________________
_____________________________________________________________________
_____________________")
                    print(f"Book Number : {i[0]} | <<<Book Title>>> : {i[1]} | Subject
Code : {i[2]} | Author : {i[3]} | Publisher : {i[4]} | Price : {i[5]} | Location : {i[6]} " )

            elif opt == '3':
                bAut = input("Enter Author Name : ")
                cursor = db.cursor()
                sql = (f"SELECT * FROM book WHERE Author LIKE '%{bAut}%'")
                data = (bAut)
                cursor.execute(sql)
                result = cursor.fetchall()
                print ('\n',cursor.rowcount," Book Search found")
                # Output
                for i in result:
                    print ("————————————————————————————————————————————————
_____________________________________________________________________
_____________________________________________________________________
_____________________")
```

```
                print(f"Book Number : {i[0]} | Book Title : {i[1]} | Subject Code :
{i[2]} | <<<Author>>> : {i[3]} | Publisher : {i[4]} | Price : {i[5]} | Location : {i[6]} " )

            elif opt == '4':
                bPub = input("Enter Publisher Name : ")
                cursor = db.cursor()
                sql = (f"SELECT * FROM book WHERE Publisher LIKE
'%{bPub}%'")

                cursor.execute(sql)
                result = cursor.fetchall()
                print ('\n',cursor.rowcount," Book Search found")
                # Output
                for i in result:
                    print
("_____
_____
_____")
                    print(f"Book Number : {i[0]} | Book Title : {i[1]} | Subject Code :
{i[2]} | Author : {i[3]} | <<<Publisher>>> : {i[4]} | Price : {i[5]} | Location : {i[6]} " )

            elif opt == '5':
                print()
                print ('''1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) Fantasy - FAN
6) OTHER''')
                sName = input("Enter the  subject Code (ARTS/ENG/HIS etc.) : ")
                sName = sName.upper()
                cursor = db.cursor()
                sql = (f"SELECT * FROM book WHERE SubjectCode LIKE
'%{sName}%'")
```

```
                cursor.execute(sql)
                result = cursor.fetchall()
                print ('\n',cursor.rowcount," Book Search found")
                # Output
                for i in result:
                    print
("_____
_____
_____")
                    print(f"Book Number : {i[0]} | Book Title : {i[1]} | <<<Subject
Code>>> : {i[2]} | Author : {i[3]} | Publisher : {i[4]} | Price : {i[5]} | Location : {i[6]} " )

            elif opt == '6':
                cursor = db.cursor()
                cursor.execute('SELECT * FROM book')
                result = cursor.fetchall()
                print ('\n',cursor.rowcount," Book Search found")
                # Output
                for i in result:
                    print
("_____
_____
_____")
                    print(f"Book Number : {i[0]} | Book Title : {i[1]} | Subject Code :
{i[2]} | Author : {i[3]} | Publisher : {i[4]} | Price : {i[5]} | Location : {i[6]} " )

        else:
            print ("\nIncorrect Option....Try Again")
            search()
            return


# Defining ADD Function
    def add():
```

```
        "This Function is to Add Books"
        print ()
        bookNo = int(input("Enter Book Number : "))
        bookTitle = input ("Enter Book Title :  ")
        author = input ("Enter Author Name : ")
        publisher = input ("Enter Publisher : ")
        price = float(input ("Enter the Price : "))
        location = input ("Enter location : ")
        print ("""\nChoose your Subject Name
1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) FANTASY - FAN
6) Programming, Business etc - OTHER """)

        # Subject Details
        Name = input("Choose your Subject Name : ")
        Name = Name.upper()
        if Name == "ART":
            subjectCode = "ARTS"
        elif Name == "LIFESTYLE":
            subjectCode = "LIFE"
        elif Name == "ENGLISH":
            subjectCode = "ENG"
        elif Name == "HISTORY":
            subjectCode = "HIS"
        elif Name == "FANTASY":
            subjectCode = "FAN"
        else:
            subjectCode = "OTHER"

        cursor = db.cursor()
```

```
        data2 = (bookNo, bookTitle, subjectCode, author, publisher, price,
location)
        sql2 = 'INSERT INTO book VALUES(%s, %s, %s, %s, %s, %s, %s)'

        # Process
        cursor.execute(sql2,data2)
        db.commit()
        print () #Chapter Details
        chapter = int(input ("Enter Total chapters "))
        for pg in range (1, chapter+1):
            chapterNo = int(pg)
            title = input(f"Enter the Title for chapter {pg} : ")
            strtPg = int(input(f"Enter Starting page number for chapter {pg} : "))
            endPg = int(input(f"Enter Ending Page number for chapter {pg} : "))
            sql3 = 'INSERT INTO bookchapter VALUES(%s, %s, %s, %s, %s)'
            data = (bookNo, chapterNo,title,strtPg,endPg)

            # Process
            cursor.execute(sql3,data)
        db.commit()
        print ()
        print (cursor.rowcount," Book Added Successfully")
        return


# Defining EDIT Function
    def edit():
        "This Function is to update books"
        print ("'Choose an option to edit
1. Book Details
2. Book Chapter Details"')
        print()
        opt = input("Enter the option (1/2) : ")
        print()
```

```
            bno = int(input("Enter the Book Number : "))
            if opt == '1':
                print("'You're about to change book details...Choose the detail to edit
1. Title
2. Author
3. Publisher
4. Price
5. Location"')
                opt1 = input("Choose the option (1/2/3/4/5) : ")
                print()

                if opt1 == '1':
                    update = input("Type the updated Title information : ")
                    cursor = db.cursor()
                    sql = 'UPDATE book SET Title = %s WHERE BookNO = %s'
                    data = (update, bno)

                    # Process
                    cursor.execute (sql, data)
                    db.commit()
                    print()
                    print (cursor.rowcount," Book Updated Successfully")

                elif opt1 == '2':
                    update = input("Type the updated Author information : ")
                    cursor = db.cursor()
                    sql = 'UPDATE book SET Author = %s WHERE BookNO = %s'
                    data = (update, bno)

                    # Process
                    cursor.execute (sql, data)
                    db.commit()
                    print()
                    print (cursor.rowcount," Book Updated Successfully")
```

```
elif opt1 == '3':
    update = input("Type the updated Publisher information : ")
    cursor = db.cursor()
    sql = 'UPDATE book SET Publisher = %s WHERE BookNO = %s'
    data = (update, bno)

    # Process
    cursor.execute (sql, data)
    db.commit()
    print()
    print (cursor.rowcount," Book Updated Successfully")

elif opt1 == '4':
    update = input("Type the updated Price information : ")
    cursor = db.cursor()
    sql = 'UPDATE book SET Price = %s WHERE BookNO = %s'
    data = (update, bno)

    # Process
    cursor.execute (sql, data)
    db.commit()
    print()
    print (cursor.rowcount," Book Updated Successfully")

elif opt1 == '5':
    update = input("Type the updated Location information : ")
    cursor = db.cursor()
    sql = 'UPDATE book SET Location = %s WHERE BookNO = %s'
    data = (update, bno)

    # Process
    cursor.execute (sql, data)
    db.commit()
```

```
                print()
                print (cursor.rowcount," Book Updated Successfully")

            else:
                print ('Please Try Again')
                edit()
        elif opt == '2':
            print("""You're about to change Chapter detail
1. Chapter Name
2. Starting Page
3. Ending Page""")
            opt2 = input("Enter the option (1/2/3) ")
            if opt2 == "1":
                chno = input("Enter the Chapter Number")
                update = input("Type the updated Chapter Title information : ")
                cursor = db.cursor()
                sql = 'UPDATE BookChapter SET Title = %s WHERE BookNO = %s
AND ChapterNo = %s'
                data = (update, bno, chno)

                # Process
                cursor.execute (sql, data)
                db.commit()
                print()
                print (cursor.rowcount," Book Updated Successfully")
            if opt2 == "2":
                chno = input("Enter the Chapter Number ")
                update = int(input("Type the updated Starting Page information : "))
                cursor = db.cursor()
                sql = 'UPDATE BookChapter SET StartingNo = %s WHERE
BookNO = %s AND ChapterNo = %s'
                data = (update, bno, chno)

                # Process
```

```
                cursor.execute (sql, data)
                db.commit()
                print()
                print (cursor.rowcount," Book Updated Successfully")

            if opt2 == "3":
                chno = input("Enter the Chapter Number : ")
                update = int(input("Type the updated Ending Page information : "))
                cursor = db.cursor()
                sql = 'UPDATE BookChapter SET  EndingNo = %s WHERE BookNO
= %s AND ChapterNo = %s'
                data = (update, bno, chno)

                # Process
                cursor.execute (sql, data)
                db.commit()
                print()
                print (cursor.rowcount," Book Updated Successfully")

            else:
                print ("\nInvalid option to edit. Try Again....")
                edit()
                return


# Defining DLT Function
        def dlt(bno: str):
            "This function is to Delete books"
            cursor = db.cursor()
            sql1 = 'DELETE FROM bookchapter WHERE BOOKNo = %s'
            data = bno

            # Process
            cursor.execute(sql1,[data])
```

```
            db.commit()
            sql2 = 'DELETE FROM book WHERE BookNo = %s'
            cursor.execute(sql2,[data])
            db.commit()
            print()
            print (cursor.rowcount," Book Deleted Successfully")
            return


# Defining CHAPVIEW Function
        def chapview():
            "This function is to View Chapter"
            print ("Chapter information with Respective Book Numbers")
            cursor = db.cursor()
            cursor.execute('SELECT * FROM bookchapter')
            result = cursor.fetchall()

            # Output
            for i in result:
                print
("_____
_____")
                print(f"<<<Book Number>>> : {i[0]} | <<<Chapter No>>> : {i[1]} | Title :
{i[2]} | Starting Page No : {i[3]} | Ending Page No : {i[4]} |" )
            print()
            print ("Note : Chapter View only Available for Testing Purposes with
Sample Data")
            print()

            # Read User Input
            bno = int(input("Enter the Book Number : "))
            chno = int(input("Enter the Chapter Number to View : "))

            cursor = db.cursor()
```

```
            sql = ('SELECT Title, StartingNo, EndingNo FROM bookchapter WHERE
BookNo = %s AND ChapterNo= %s;')
            data = (bno, chno )
            cursor.execute (sql,data)
            result = cursor.fetchall()
            for x in result:
                title = x[0]
                startno = x[1]
                endno= x[2]

            # Reading the Content
            view = fo.readlines()
            time.sleep(1)
            print(f"'\nChapter Title : {title}
Starting Page : {startno}
Ending Page : {endno}"')
            for read in view:
                print(read)
            return

except Error as er:
    print ("Oops!")
    print (er)

# Disconnect from server
db.close

# STOP
```

1. Book.py : A one-time-use code for executing the book table.

```
import mysql.connector
from mysql.connector import Error
# Open database connection
conDict = {"host":"localhost",
          "database": "library",
          "user":"root",
          "password":""}

try:
    db = mysql.connector.connect (**conDict)
    if db.is_connected():
        cursor = db.cursor()
        # Execute SQL query
        cursor.execute ("""CREATE TABLE IF NOT EXISTS Book
                        (BookNo int(50) NOT NULL,
                         Title varchar(150),
                         SubjectCode varchar(60),
                         Author varchar(40),
                         Publisher varchar(50),
                         Price float(25),
                         Location varchar(50),
                         PRIMARY KEY(BookNo));""")
        data = cursor.fetchall()

except Error as e:
    print ("Oops!")
    print (e)

# Disconnect from server
db.close
```

2. BookChapter.py : A one-time-use code for executing the Book chapter table.

```
import mysql.connector
from mysql.connector import Error
# Open database connection
conDict = {"host":"localhost",
          "database": "library",
          "user":"root",
          "password":""}

try:
    db = mysql.connector.connect (**conDict)
    if db.is_connected():
        cursor = db.cursor()
        # Execute SQL query to Create Chapter Table
        cursor.execute ("""CREATE TABLE IF NOT EXISTS BookChapter
                    (BookNo int(50), ChapterNo int(40), Title varchar (250),
StartingNo int(50), EndingNo int(50),
                    PRIMARY KEY(BookNo,ChapterNo),
                    FOREIGN KEY(BookNo) REFERENCES Book(BookNo));""")
        data = cursor.fetchall()

except Error as e:
    print ("Oops!")
    print (e)

# Disconnect from server
db.close
```

3. Subject.py: A one-time-use code for executing the subject category table.

```
import mysql.connector
from mysql.connector import Error
# Open database connection
conDict = {"host":"localhost",
          "database": "library",
          "user":"root",
          "password":""}
try:
    db = mysql.connector.connect (**conDict)
    if db.is_connected():
        # Execute SQL query to Create Subject Table
        cursor = db.cursor()
        cursor.execute ("""CREATE TABLE IF NOT EXISTS Subject
                    (SubjectCode Varchar(60)NOT NULL, Name varchar(70) NOT
NULL,
                    PRIMARY KEY(SubjectCode));""")
        data = cursor.fetchall()
        # Execute Subject Details
        sql = '''INSERT INTO subject (SubjectCode, Name) VALUES
('ARTS', 'ART'),
('ENG', 'ENGLISH'),
('FAN', 'FANTASY'),
('HIS', 'HISTORY'),
('LIFE', 'LIFESTYLE');'''
        cursor.execute(sql)
        db.commit()

except Error as e:
    print ("Oops!")
    print (e)
# Disconnect from server
db.close
```

3. Test Cases and Screen Shots

3.1. Adding Books

Test Case #	User Input	Actual Results	Expected Results	Pass/Fail
1	Enter Book Number : 1 Enter Book Title : The Goldfinch Enter Author Name : Donna Tartt Enter Publisher : Little, Brown Book Group Limited Enter the Price : 10.99 Enter location : United States Choose your Subject Name Choose your Subject Name : English Enter Total chapters 3 Enter the Title for chapter 1 : Boy with a Skull Enter Starting page number for chapter 1 : 10 Enter Ending Page number for chapter 1 : 500 Enter the Title for chapter 2 : The Anatomy Lesson Enter Starting page number for chapter 2 : 502 Enter Ending Page number for chapter 2 : 809 Enter the Title for chapter 3 : Park Avenue Enter Starting page number for chapter 3 : 810 Enter Ending Page number for chapter 3 : 1500	1 Book Added Successfully	1 Book Added Successfully	**Pass**
2	Enter Book Number : 2 Enter Book Title : Learn Python the Hard Way Enter Author Name : Zed Shaw Enter Publisher : Addison-Wesley Professional Enter the Price : 29.99 Enter location : UK Choose your Subject Name : Programming Enter Total chapters 1 Enter the Title for chapter 1 : Basic Development Enter Starting page number for chapter 1 : 6 Enter Ending Page number for chapter 1 : 100	1 Book Added Successfully	1 Book Added Successfully	**Pass**
3	Enter Book Number : 3 Enter Book Title : Harry Potter and the Goblet of Fire Enter Author Name : J. K. Rowling Enter Publisher : Scholastic Enter the Price : 6.64 Enter location : London Choose your Subject Name : Fantasy Enter Total chapters 4 Enter the Title for chapter 1 : The Riddle House	1 Book Added Successfully	1 Book Added Successfully	**Pass**

	Enter Starting page number for chapter 1 : 5 Enter Ending Page number for chapter 1 : 10 Enter the Title for chapter 2 : The Scar Enter Starting page number for chapter 2 : 27 Enter Ending Page number for chapter 2 : 49 Enter the Title for chapter 3 : The Invitation Enter Starting page number for chapter 3 : 53 Enter Ending Page number for chapter 3 : 80 Enter the Title for chapter 4 : Back to the Burrow Enter Starting page number for chapter 4 : 84 Enter Ending Page number for chapter 4 : 100			

Table 1 Test Cases for Adding Books

```
                WELCOME TO ABC BOOK STORE LIBRARY SYSTEM
_____________________________________

1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________

Enter Requested Option (1/2/3/4/5) : 2


Enter Book Number : 1
Enter Book Title :  The Goldfinch
Enter Author Name : Donna Tartt
Enter Publisher : Little, Brown Book Group Limited
Enter the Price : 10.99
Enter location : United States

Choose your Subject Name
1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) Fantasy - FAN
6) Programming, Business etc - OTHER
Choose your Subject Name : English

Enter Total chapters 3
Enter the Title for chapter 1 : Boy with a Skull
Enter Starting page number for chapter 1 : 10
Enter Ending Page number for chapter 1 : 500
Enter the Title for chapter 2 : The Anatomy Lesson
Enter Starting page number for chapter 2 : 502
Enter Ending Page number for chapter 2 : 809
Enter the Title for chapter 3 : Park Avenue
Enter Starting page number for chapter 3 : 810
Enter Ending Page number for chapter 3 : 1500

1  Book Added Successfully

Do You want to EXIT the System ? (YES or NO) yes

Thanks For using ABC Book Store.
```

Figure 3Adding Book Sample 2

```
                WELCOME TO ABC BOOK STORE LIBRARY SYSTEM
_____________________________________

1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________

Enter Requested Option (1/2/3/4/5) : 2


Enter Book Number : 3
Enter Book Title :  Harry Potter and the Goblet of Fire
Enter Author Name : J. K. Rowling
Enter Publisher : Scholastic
Enter the Price : 6.64
Enter location : London

Choose your Subject Name
1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) Fantasy - FAN
6) Programming, Business etc - OTHER
Choose your Subject Name : Fantasy

Enter Total chapters 4
Enter the Title for chapter 1 : The Riddle House
Enter Starting page number for chapter 1 : 5
Enter Ending Page number for chapter 1 : 10
Enter the Title for chapter 2 : The Scar
Enter Starting page number for chapter 2 : 27
Enter Ending Page number for chapter 2 : 49
Enter the Title for chapter 3 : The Invitation
Enter Starting page number for chapter 3 : 53
Enter Ending Page number for chapter 3 : 80
Enter the Title for chapter 4 : Back to the Burrow
Enter Starting page number for chapter 4 : 84
Enter Ending Page number for chapter 4 : 100

1  Book Added Successfully
```

Figure 2 Adding Book Sample 1

```
                WELCOME TO ABC BOOK STORE LIBRARY SYSTEM
_____________________________________________

1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________________

Enter Requested Option (1/2/3/4/5) : 2


Enter Book Number : 2
Enter Book Title :  Learn Python the Hard Way
Enter Author Name : Zed Shaw
Enter Publisher : Addison-Wesley Professional
Enter the Price : 29.99
Enter location : UK

Choose your Subject Name
1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) Fantasy - FAN
6) Programming, Business etc - OTHER
Choose your Subject Name : Programming

Enter Total chapters 1
Enter the Title for chapter 1 : Basic Development
Enter Starting page number for chapter 1 : 6
Enter Ending Page number for chapter 1 : 100

1  Book Added Successfully

Do You want to EXIT the System ? (YES or NO) yes

Thanks For using ABC Book Store.
```

Figure 4 Adding Book Sample 3

	BookNo	Title	SubjectCode	Author	Publisher	Price	Location
Edit Copy Delete	1	The Goldfinch	ENG	Donna Tartt	Little, Brown Book Group Limited	10.99	United States
Edit Copy Delete	2	Learn Python the Hard Way	OTHER	Zed Shaw	Addison-Wesley Professional	29.99	UK
Edit Copy Delete	3	Harry Potter and the Goblet of Fire	FAN	J. K. Rowling	Scholastic	6.64	London
Edit Copy Delete	4	Harry Potter and the Prisoner of Azkaban	FAN	J. K. Rowling	Pottermore Publishing	10.99	UK
Edit Copy Delete	5	The Da Vinci Code: Featuring Robert Langdon	ARTS	Dan Brown	Knopf Doubleday Publishing Group	30.02	United States
Edit Copy Delete	6	The Happiness Project	ENG	Gretchen Rubin	HarperCollins	43	London
Edit Copy Delete	7	Alias Grace	HIS	Margaret Atwood	Little, Brown Book Group Limited	54	New York

Figure 5 phpMyAdmin Results for Adding Books

3.2.Deleting Books

Test Case #	User Input	Actual Results	Expected Results	Pass/Fail
1	Enter the Book Number : 7	1 Book Deleted Successfully	1 Book Deleted Successfully	**Pass**

Table 2 Test Cases for Deleting Books

```
                WELCOME TO ABC BOOK STORE LIBRARY SYSTEM
_____________________________________________

1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________________

Enter Requested Option (1/2/3/4/5) : 4

Enter the Book Number : 7

1  Book Deleted Successfully

Do You want to EXIT the System ? (YES or NO)
```

Figure 6 Deleting Books Sample 1

←T→	BookNo 1	Title	SubjectCode	Author	Publisher	Price	Location
Edit Copy Delete	1	The Goldfinch	ENG	Donna Tartt	Little, Brown Book Group Limited	10.99	United States
Edit Copy Delete	2	Learn Python the Hard Way	OTHER	Zed Shaw	Addison-Wesley Professional	29.99	UK
Edit Copy Delete	3	Harry Potter and the Goblet of Fire	FAN	J. K. Rowling	Scholastic	6.64	London
Edit Copy Delete	4	Harry Potter and the Prisoner of Azkaban	FAN	J. K. Rowling	Pottermore Publishing	10.99	UK
Edit Copy Delete	5	The Da Vinci Code: Featuring Robert Langdon	ARTS	Dan Brown	Knopf Doubleday Publishing Group	30.02	United States
Edit Copy Delete	6	The Happiness Project	ENG	Gretchen Rubin	HarperCollins	43	London

Figure 7 phpMyAdmin Results for Deleting Books

3.3.Editing Books

Test Case #	User Input	Actual Results	Expected Results	Pass/Fail
1	Enter the option (1/2) : 1 Enter the Book Number : 4 You're about to change book details...Choose the detail to edit Choose the option (1/2/3/4/5) : 4 Type the updated Price information : 34.7	1 Book Updated Successfully	1 Book Updated Successfully	**Pass**
2	Enter Requested Option (1/2/3/4/5) : 3 Enter the option (1/2) : 2 Enter the Book Number : 6 You're about to change Chapter detail Enter the option (1/2/3) 1 Enter the Chapter Number1 Type the updated Chapter Title information : One Rainy Evening	1 Book Updated Successfully	1 Book Updated Successfully	**Pass**

Table 3 Test Cases for Editing Books

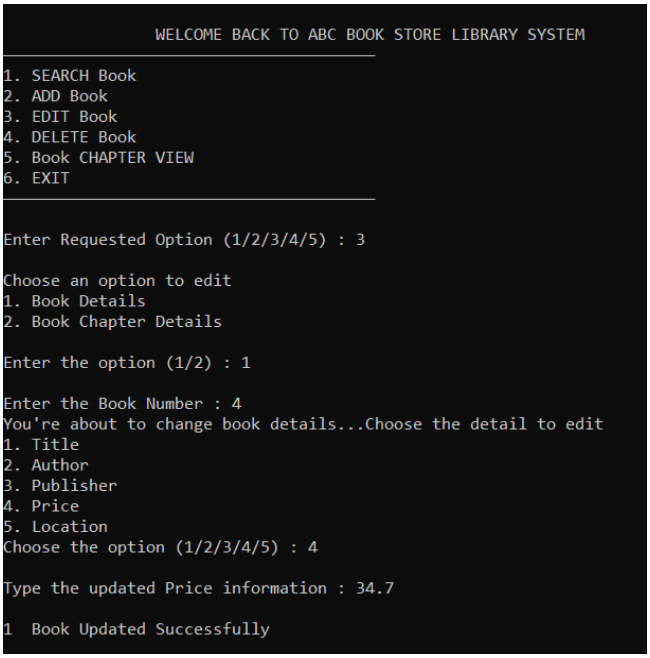

Figure 8 Editing Books Sample 1

	BookNo	Title	SubjectCode	Author	Publisher	Price	Location
Edit Copy Delete	1	The Goldfinch	ENG	Donna Tartt	Little, Brown Book Group Limited	10.99	United States
Edit Copy Delete	2	Learn Python the Hard Way	OTHER	Zed Shaw	Addison-Wesley Professional	29.99	UK
Edit Copy Delete	3	Harry Potter and the Goblet of Fire	FAN	J. K. Rowling	Scholastic	6.64	London
Edit Copy Delete	4	Harry Potter and the Prisoner of Azkaban	FAN	J. K. Rowling	Pottermore Publishing	34.7	UK
Edit Copy Delete	5	The Da Vinci Code: Featuring Robert Langdon	ARTS	Dan Brown	Knopf Doubleday Publishing Group	30.02	United States
Edit Copy Delete	6	The Happiness Project	ENG	Gretchen Rubin	HarperCollins	43	London

Figure 9 phpMyAdmin Results for Editing Books 1

```
1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT

Enter Requested Option (1/2/3/4/5) : 3

Choose an option to edit
1. Book Details
2. Book Chapter Details

Enter the option (1/2) : 2

Enter the Book Number : 6
You're about to change Chapter detail
1. Chapter Name
2. Starting Page
3. Ending Page
Enter the option (1/2/3) 1
Enter the Chapter Number1
Type the updated Chapter Title information : One Rainy Evening

1  Book Updated Successfully
```

Figure 10 Editing Books Sample 2

Edit Copy Delete	6	1	One Rainy Evening	6	40	

Figure 11 phpMyAdmin Results for Editing Books 2

3.4. Book Chapter View

Test Case #	User Input	Actual Results	Expected Results	Pass/Fail
1	Enter Requested Option (1/2/3/4/5) : 5 Enter the Book Number : 3 Enter the Chapter Number to View : 3	Chapter Title : The Invitation Starting Page : 53 Ending Page : 80 Lorem ipsum dolor sit amet, consectetur adipiscing elit….	Chapter Title : The Invitation Starting Page : 53 Ending Page : 80 Lorem ipsum dolor sit amet, consectetur adipiscing elit….	**Pass**

Table 4 Test Cases for Book Chapter View

```
1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________

Enter Requested Option (1/2/3/4/5) : 5

Chapter information with Respective Book Numbers
_____________________________________________________________________________________________________
<<<Book Number>>> : 1 | <<<Chapter No>>> : 1 | Title : Boy with a Skull | Starting Page No : 10 | Ending Page No : 500 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 1 | <<<Chapter No>>> : 2 | Title : The Anatomy Lesson | Starting Page No : 502 | Ending Page No : 809 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 1 | <<<Chapter No>>> : 3 | Title : Park Avenue | Starting Page No : 810 | Ending Page No : 1500 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 2 | <<<Chapter No>>> : 1 | Title : Basic Development | Starting Page No : 6 | Ending Page No : 100 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 3 | <<<Chapter No>>> : 1 | Title : The Riddle House | Starting Page No : 5 | Ending Page No : 10 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 3 | <<<Chapter No>>> : 2 | Title : The Scar | Starting Page No : 27 | Ending Page No : 49 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 3 | <<<Chapter No>>> : 3 | Title : The Invitation | Starting Page No : 53 | Ending Page No : 80 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 3 | <<<Chapter No>>> : 4 | Title : Back to the Burrow | Starting Page No : 84 | Ending Page No : 100 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 4 | <<<Chapter No>>> : 1 | Title : The  Knight | Starting Page No : 5 | Ending Page No : 50 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 5 | <<<Chapter No>>> : 1 | Title : Later that Night | Starting Page No : 10 | Ending Page No : 60 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 5 | <<<Chapter No>>> : 2 | Title : Moments Later | Starting Page No : 65 | Ending Page No : 100 |
_____________________________________________________________________________________________________
<<<Book Number>>> : 6 | <<<Chapter No>>> : 1 | Title : One Rainy Evening | Starting Page No : 6 | Ending Page No : 40 |

Note : Chapter View only Available for Testing Purposes with Sample Data

Enter the Book Number : 3
Enter the Chapter Number to View : 3

Chapter Title : The Invitation
Starting Page : 53
Ending Page : 80
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam hendrerit nisi sed sollicitudin pellentesque. Nunc posuere purus rhoncus pulvinar aliquam.
ttitor venenatis. Donec a dui et dui fringilla consectetur id nec massa. Aliquam erat volutpat. Sed ut dui ut lacus dictum fermentum vel tincidunt neque
 nunc eros, mattis at dui ac, convallis semper risus. In adipiscing ultrices tellus, in suscipit massa vehicula eu.
```

Figure 12 Book Chapter View sample 1

3.5.Search Books

Test Case #	User Input	Actual Results	Expected Results	Pass/Fail
1	Choose an option (1/2/3/4/5/6) : 1 Enter the Book Number : 3	1 Book Search found	1 Book Search found	**Pass**
2	Choose an option (1/2/3/4/5/6) : 2 Enter the Book Title : harry potter	2 Book Search found	2 Book Search found	**Pass**
3	Choose an option (1/2/3/4/5/6) : 5 Enter the subject Code (ARTS/ENG/HIS etc.) : Eng	2 Book Search found	2 Book Search found	**Pass**
4	Choose an option (1/2/3/4/5/6) : 6 6 Book Search found	6 Book Search found	6 Book Search found	**Pass**

Table 5 Test Cases for Searching Books

Figure 13 Search Books Sample 1

```
                WELCOME TO ABC BOOK STORE LIBRARY SYSTEM
_____________________________________________

1. SEARCH Book
2. ADD Book
3. EDIT Book
4. DELETE Book
5. Book CHAPTER VIEW
6. EXIT
_____________________________________________

Enter Requested Option (1/2/3/4/5) : 1


You are about to search for Books

Choose an option to start the query
1. Book Number
2. Book Title
3. Author
4. Publisher
5. Subject category
6. Select All Books

Choose an option (1/2/3/4/5/6) : 1
Enter the Book Number : 3

 1  Book Search found
_____________________________
<<<Book Number>>> :  3
Book Title :  Harry Potter and the Goblet of Fire
Subject Code :  FAN
Author :  J. K. Rowling
Publisher :  Scholastic
Price :  6.64
Location :  London
_____________________________
```

Figure 14 Search Books Sample 2

```
1) ART - ARTS
2) LIFESTYLE - LIFE
3) ENGLISH - ENG
4) HISTORY - HIS
5) Fantasy - FAN
6) OTHER
Enter the  subject Code (ARTS/ENG/HIS etc.) : eng

 2  Book Search found
_____________________________________________________________________________________________________________________________
Book Number : 1 | Book Title : The Goldfinch | <<<Subject Code>>> : ENG | Author : Donna Tartt | Publisher : Little, Brown Book Group Limited | Price : 10.99 | Location : United States
_____________________________________________________________________________________________________________________________
Book Number : 6 | Book Title : The Happiness Project | <<<Subject Code>>> : ENG | Author :  Gretchen Rubin | Publisher : HarperCollins | Price : 43.0 | Location : London
```

Figure 15 Search Books Sample 3

Figure 16 Search Books Sample 4

End of Coursework