INDUS VALLEY PUBLIC SCHOOL, NOIDA

SYNOPSIS ON LIBRARY MANAGEMENT SYSTEM

PREPARED BY:

JAHNAVI ANAND

SUBMITTED TO: DEEPALI MA'AM

ACKNOWLEDGEMENT

First of all, I would like to thank my inspiring and supportive Computer Science teacher Deepali Ma'am to assign me the integrated Python and SQL project on our chosen topic of 'Library Management'. Without her guidance, both me and this assignment would not be futile, but an incomplete set of data, visuals and aptitude.

Next, I would like to acknowledge my project partner Ms. Nupur Sharma for co-creating and programming with me.

Lastly, I would like to express my gratitude to https://in.pinterest.com/ for furnishing me with the images for my synopsis.

Regards
Jahnavi Anand

TABLE OF CONTENT

RESOURCES UTILIZED

PYTHON

- Python is an interpreted high-level general-purpose programming language.
- Its design philosophy emphasizes code readability with its use of significant indentation.
- Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.
- Python Programming Language was developed by Guido Van Rossum in February 1991.

MYSQL

- MySQL database system refers to the combination of a MySQL server instance and a MySQL database.
- MySQL operates using client/server architecture in which the server runs on the machine containing the databases and clients connect to the server over a network.
- MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).
- MySQL provides set of features that support a secure environment for storing, maintaining and accessing data.

INTRODUCTION OF THE PROJECT

PROJECT OBJECTIVE

1. To allow the authorized users and customers to access the information available in the system.
2. Locate any book wanted by the customer.
3. Reduced labour-intensive manual work for storing various records.
4. Provide greater speed and reduce time consumption.

PROJECT BENEFITS

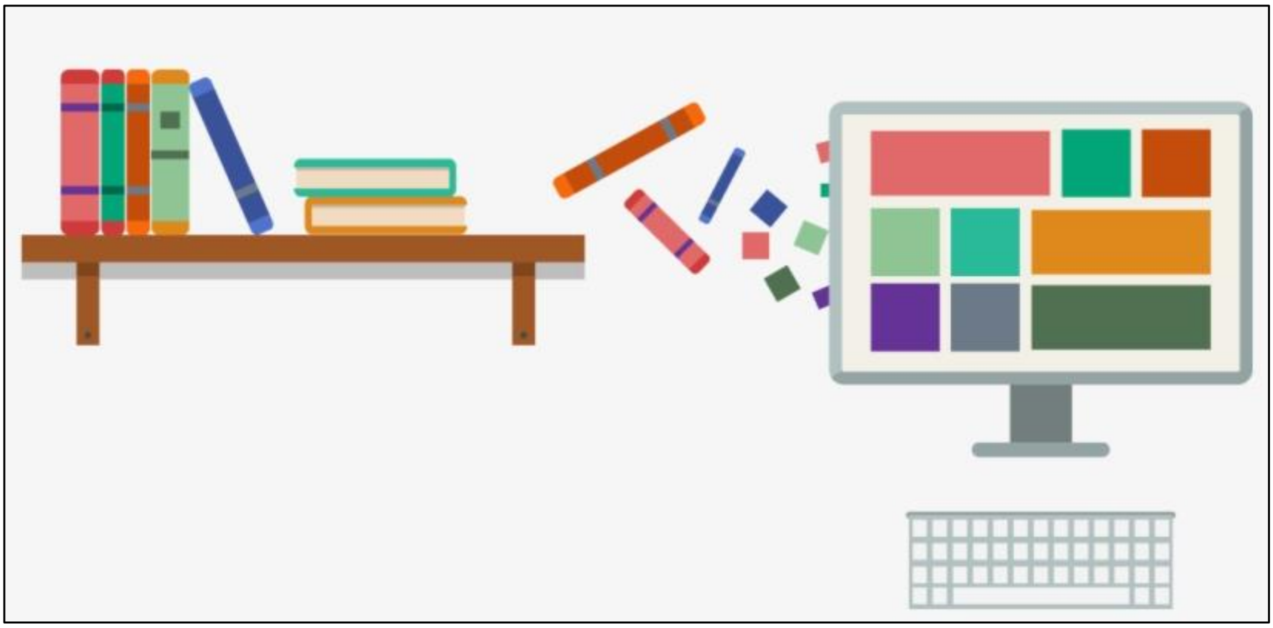
1. Customer service portal for customers to access books.
2. Staff portal for library employees to update the data in the library records.
3. Efficient and effortless storing of vast library data.

PROJECT SCOPE

A library is a collection of materials, books or media that are easily accessible for use. A public library is a library that is accessible by the general public and is usually funded from public sources, such as taxes.

With the extensive collection of books distributed under many genres and a massive community utilizing the facility, keeping hand-written manual records can be a little inconvenient and stressful.

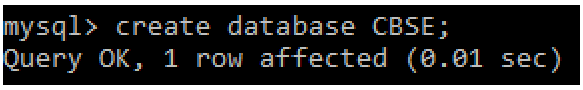
Efficient Library Management System will help in maintaining accurate and professional documentation of both the books and the customers of the library.



THE PROJECT CODE

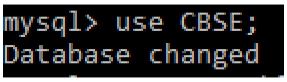
SQL CODE

create database CBSE;



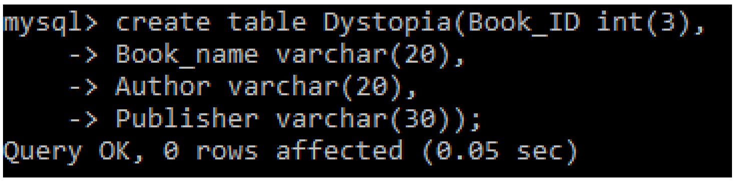
```
mysql> create database CBSE;
Query OK, 1 row affected (0.01 sec)
```

use CBSE;



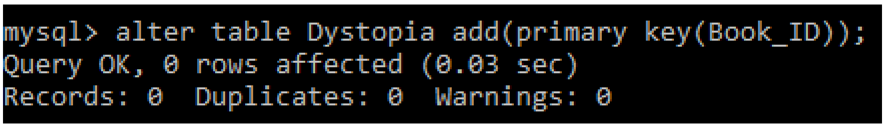
```
mysql> use CBSE;
Database changed
```

create table Dystopia (Book_ID int(3),
Book_name varchar (20),
Author varchar(20),
Publisher varchar(30));



```
mysql> create table Dystopia(Book_ID int(3),
    -> Book_name varchar(20),
    -> Author varchar(20),
    -> Publisher varchar(30));
Query OK, 0 rows affected (0.05 sec)
```

alter table Dystopia add(primary key(Book_ID));



```
mysql> alter table Dystopia add(primary key(Book_ID));
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

insert into Dystopia values(001,'Divergent',’Veronica Roth’,'Katherine Tegen Books');

insert into Dystopia values(002,'The Hunger Games',’Suzanne Collins’,'Scholastic Corporation');

insert into Dystopia values(003,'Shatter Me',’Tahereh Mafi’,'Harper Collins');

insert into Dystopia values(004,'The maze Runner',’James Dashner’,'Dell Publishing');

insert into Dystopia values(005,'The Fifth Wave',’Rick Yancey’,'GP Putnams Sons');

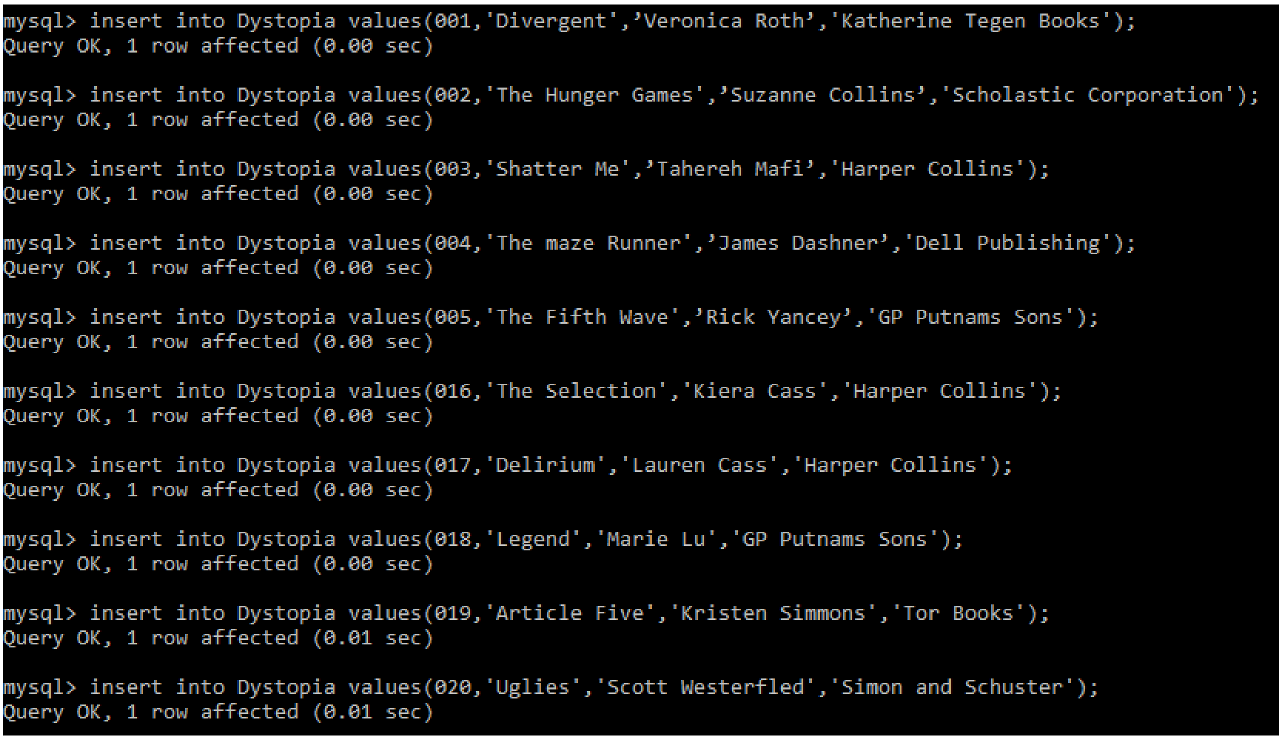
insert into Dystopia values(016,'The Selection','Kiera Cass','Harper Collins');

insert into Dystopia values(017,'Delirium','Lauren Cass','Harper Collins');

insert into Dystopia values(018,'Legend','Marie Lu','GP Putnams Sons');

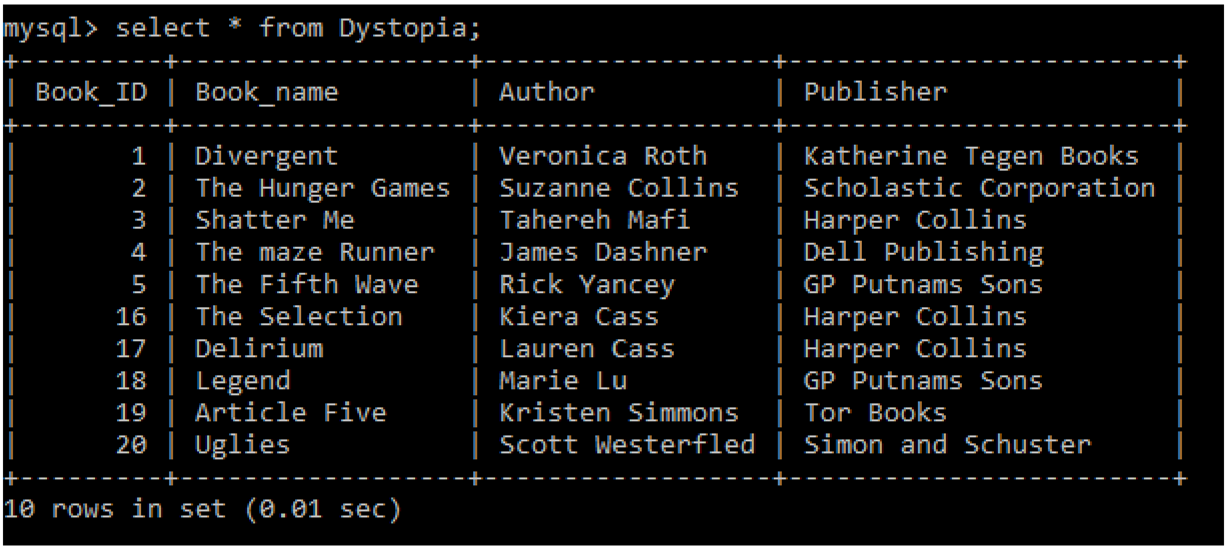
insert into Dystopia values(019,'Article Five','Kristen Simmons','Tor Books');

insert into Dystopia values(020,'Uglies','Scott Westerfled','Simon and Schuster');



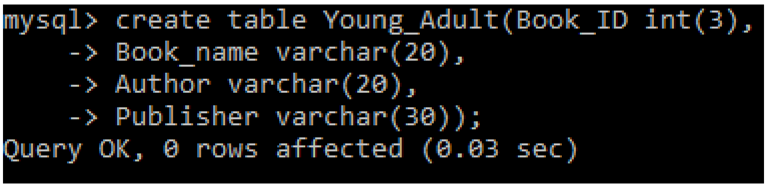
```
mysql> insert into Dystopia values(001,'Divergent',’Veronica Roth’,'Katherine Tegen Books');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(002,'The Hunger Games',’Suzanne Collins’,'Scholastic Corporation');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(003,'Shatter Me',’Tahereh Mafi’,'Harper Collins');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(004,'The maze Runner',’James Dashner’,'Dell Publishing');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(005,'The Fifth Wave',’Rick Yancey’,'GP Putnams Sons');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(016,'The Selection','Kiera Cass','Harper Collins');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(017,'Delirium','Lauren Cass','Harper Collins');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(018,'Legend','Marie Lu','GP Putnams Sons');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(019,'Article Five','Kristen Simmons','Tor Books');
Query OK, 1 row affected (0.01 sec)

mysql> insert into Dystopia values(020,'Uglies','Scott Westerfled','Simon and Schuster');
Query OK, 1 row affected (0.01 sec)
```

select * from Dystopia;



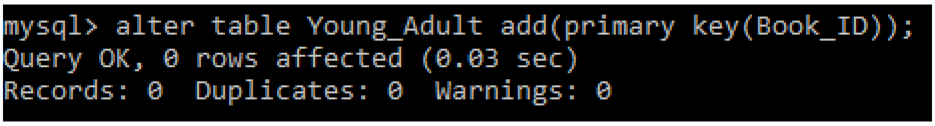
```
mysql> select * from Dystopia;
+---------+------------------+------------------+------------------------+
| Book_ID | Book_name        | Author           | Publisher              |
+---------+------------------+------------------+------------------------+
|       1 | Divergent        | Veronica Roth    | Katherine Tegen Books  |
|       2 | The Hunger Games | Suzanne Collins  | Scholastic Corporation |
|       3 | Shatter Me       | Tahereh Mafi     | Harper Collins         |
|       4 | The maze Runner  | James Dashner    | Dell Publishing        |
|       5 | The Fifth Wave   | Rick Yancey      | GP Putnams Sons        |
|      16 | The Selection    | Kiera Cass       | Harper Collins         |
|      17 | Delirium         | Lauren Cass      | Harper Collins         |
|      18 | Legend           | Marie Lu         | GP Putnams Sons        |
|      19 | Article Five     | Kristen Simmons  | Tor Books              |
|      20 | Uglies           | Scott Westerfled | Simon and Schuster     |
+---------+------------------+------------------+------------------------+
10 rows in set (0.01 sec)
```

create table Young_Adult(Book_ID int(3),
Book_name varchar(20),
Author varchar(20),
Publisher varchar(30));



```
mysql> create table Young_Adult(Book_ID int(3),
    -> Book_name varchar(20),
    -> Author varchar(20),
    -> Publisher varchar(30));
Query OK, 0 rows affected (0.03 sec)
```

alter table Young_Adult add(primary key(Book_ID));



```
mysql> alter table Young_Adult add(primary key(Book_ID));
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

insert into Young_Adult values(006,'Before I Fall',’Lauren Oliver’,'Harper Collins');

insert into Young_Adult values(007,'Fangirl',’Rainbow Rowell’,'St Martins Press');

insert into Young_Adult values(008,'If I Stay',’Gayle Forman’,'EP Dutton’);

insert into Young_Adult values(009,'One of Us',’Karen McManus’,'Dela corte Press');

insert into Young_Adult values(010,'Paper Towns',’John Green’,'EP Dutton’);

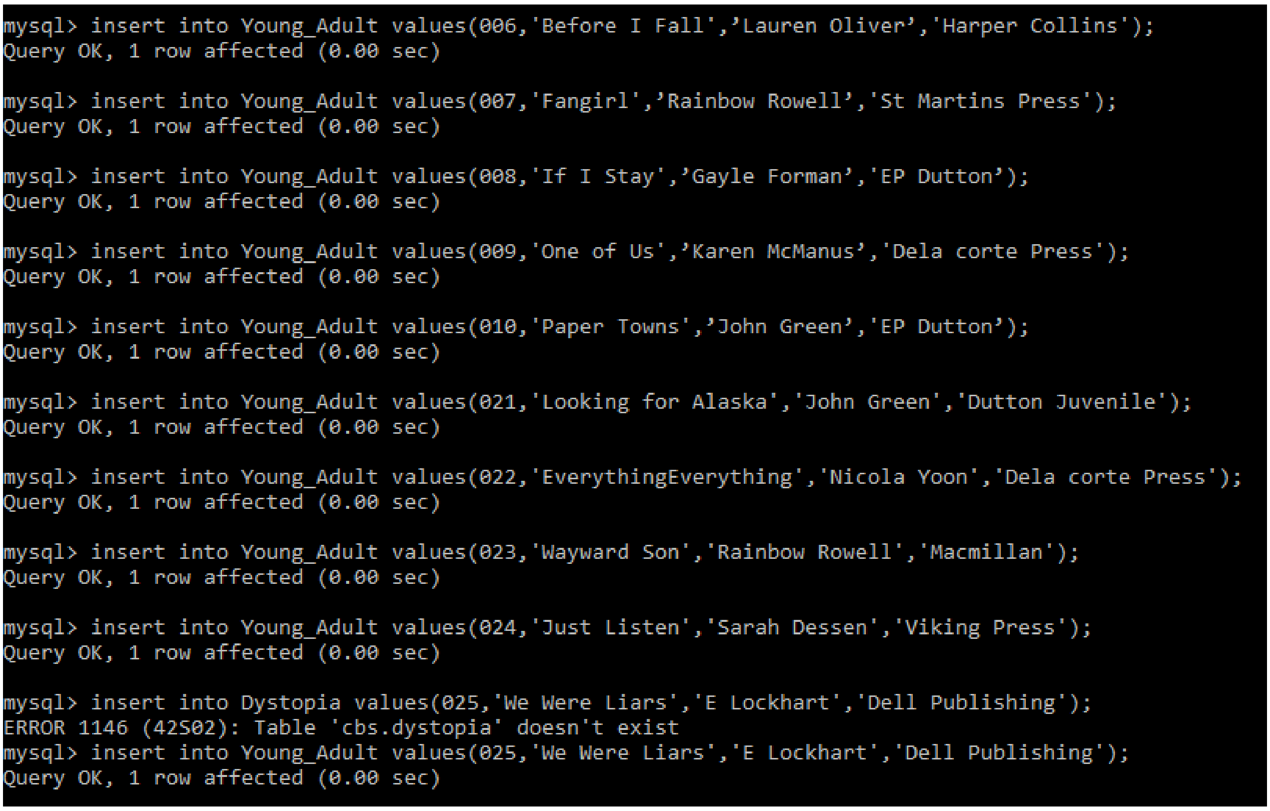
insert into Young_Adult values(021,'Looking for Alaska','John Green','Dutton Juvenile');

insert into Young_Adult values (022,'EverythingEverything','Nicola Yoon','Dela corte Press');

insert into Young_Adult values(023,'Wayward Son','Rainbow Rowell','Macmillan');

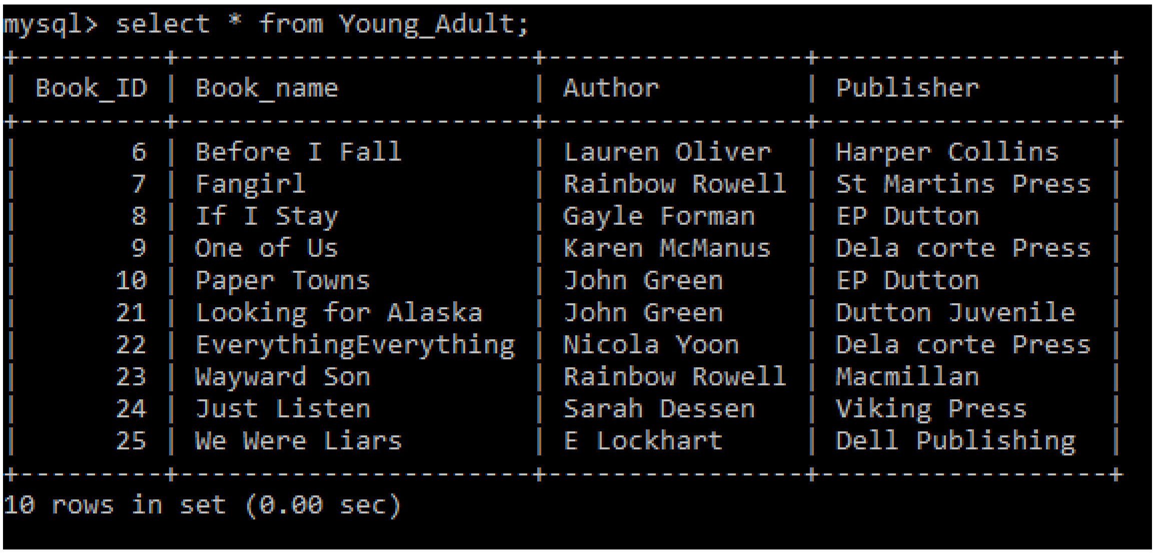
insert into Young_Adult values(024,'Just Listen','Sarah Dessen','Viking Press');

insert into Young_Adult values(025,'We Were Liars','E Lockhart','Dell Publishing');



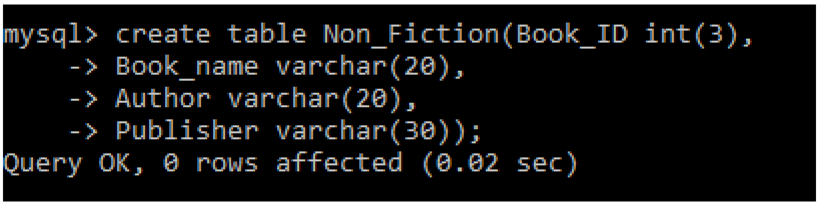
```
mysql> insert into Young_Adult values(006,'Before I Fall',’Lauren Oliver’,'Harper Collins');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(007,'Fangirl',’Rainbow Rowell’,'St Martins Press');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(008,'If I Stay',’Gayle Forman’,'EP Dutton’);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(009,'One of Us',’Karen McManus’,'Dela corte Press');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(010,'Paper Towns',’John Green’,'EP Dutton’);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(021,'Looking for Alaska','John Green','Dutton Juvenile');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(022,'EverythingEverything','Nicola Yoon','Dela corte Press');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(023,'Wayward Son','Rainbow Rowell','Macmillan');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Young_Adult values(024,'Just Listen','Sarah Dessen','Viking Press');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Dystopia values(025,'We Were Liars','E Lockhart','Dell Publishing');
ERROR 1146 (42S02): Table 'cbs.dystopia' doesn't exist
mysql> insert into Young_Adult values(025,'We Were Liars','E Lockhart','Dell Publishing');
Query OK, 1 row affected (0.00 sec)
```

select * from Young_Adult;



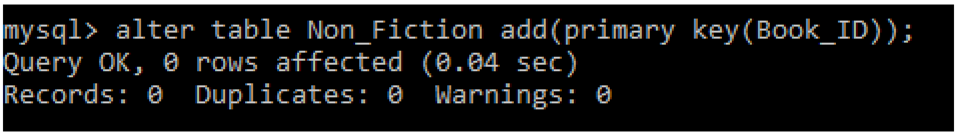
```
mysql> select * from Young_Adult;
+---------+---------------------+---------------+------------------+
| Book_ID | Book_name           | Author        | Publisher        |
+---------+---------------------+---------------+------------------+
|       6 | Before I Fall       | Lauren Oliver | Harper Collins   |
|       7 | Fangirl             | Rainbow Rowell| St Martins Press |
|       8 | If I Stay           | Gayle Forman  | EP Dutton        |
|       9 | One of Us           | Karen McManus | Dela corte Press |
|      10 | Paper Towns         | John Green    | EP Dutton        |
|      21 | Looking for Alaska  | John Green    | Dutton Juvenile  |
|      22 | EverythingEverything| Nicola Yoon   | Dela corte Press |
|      23 | Wayward Son         | Rainbow Rowell| Macmillan        |
|      24 | Just Listen         | Sarah Dessen  | Viking Press     |
|      25 | We Were Liars       | E Lockhart    | Dell Publishing  |
+---------+---------------------+---------------+------------------+
10 rows in set (0.00 sec)
```

create table Non_Fiction(Book_ID int(3),
Book_name varchar(20),
Author varchar(20),
Publisher varchar(30));



```
mysql> create table Non_Fiction(Book_ID int(3),
    -> Book_name varchar(20),
    -> Author varchar(20),
    -> Publisher varchar(30));
Query OK, 0 rows affected (0.02 sec)
```

alter table Non_Fiction add(primary key(Book_ID));



```
mysql> alter table Non_Fiction add(primary key(Book_ID));
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

insert into Non_Fiction values(011,'Rich Dad Poor Dad',’Robert Kiyosaki’,’Plata Publishing’);

insert into Non_Fiction values(012,'The Five AM Club',’Robin Sharma’,’A Jaico Book’);

insert into Non_Fiction values(013,'Turning Point',’APJ Abdul Kalam’,’Harper Collins’);

insert into Non_Fiction values(014,'Think and Grow Rich',’Napolean Hill’,’FingerPrint’);

insert into Non_Fiction values(015,'Educated',’Tara Westover’,’Penguin Random House LCC’);

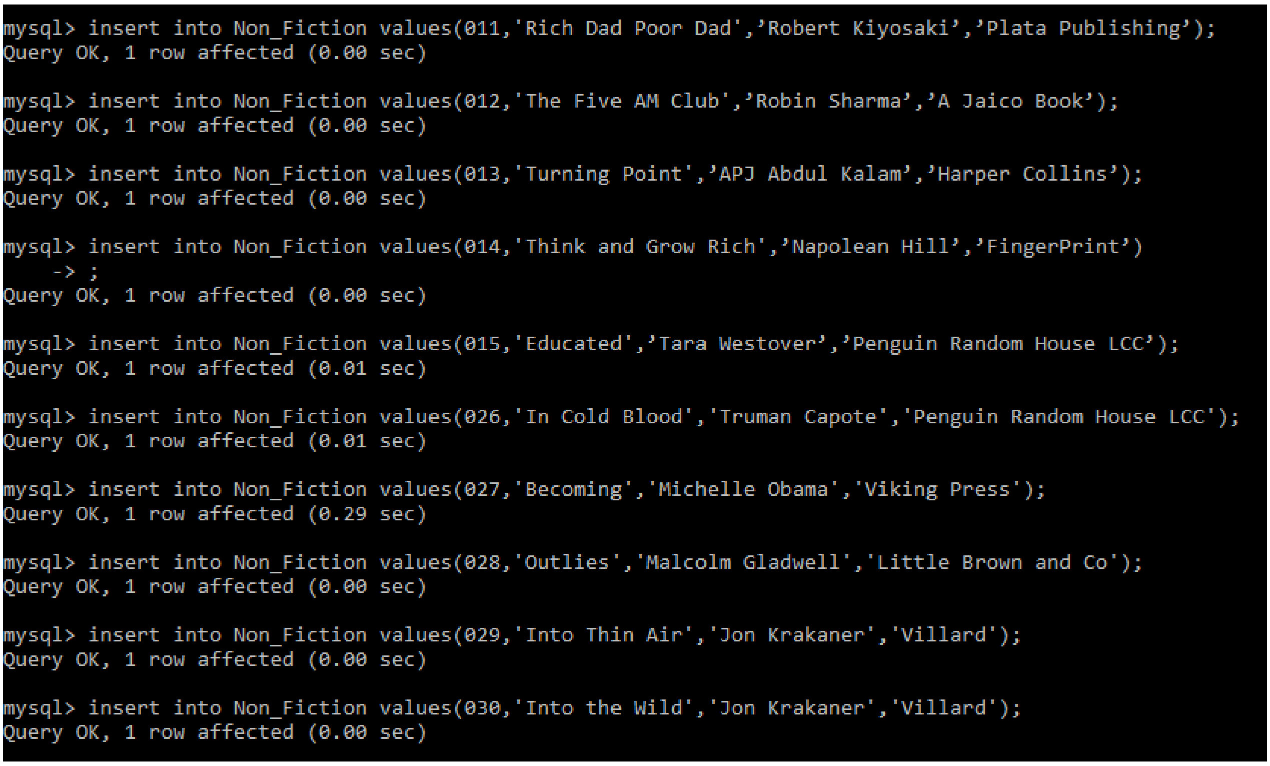
insert into Non_Fiction values(026,'In Cold Blood','Truman Capote','Penguin Random House LCC');

insert into Non_Fiction values(027,'Becoming','Michelle Obama','Viking Press');

insert into Non_Fiction values(028,'Outlies','Malcolm Gladwell','Little Brown and Co');

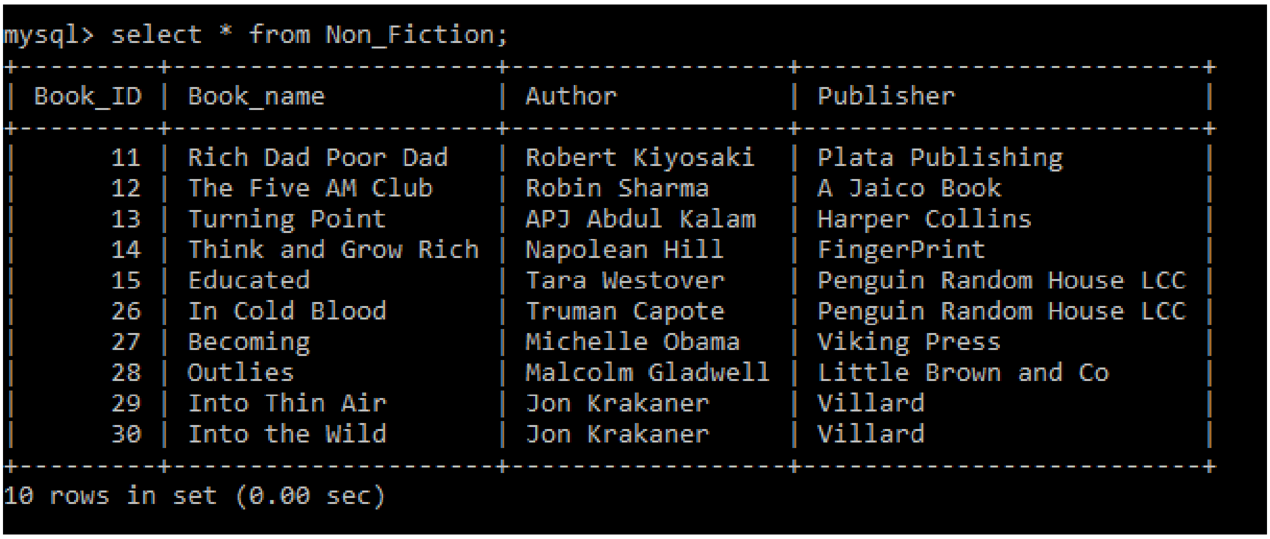
insert into Non_Fiction values(029,'Into Thin Air','Jon Krakaner','Villard');

insert into Non_Fiction values(030,'Into the Wild','Jon Krakaner','Villard');



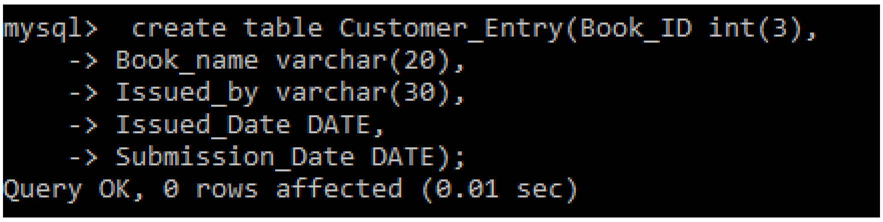
```
mysql> insert into Non_Fiction values(011,'Rich Dad Poor Dad',’Robert Kiyosaki’,’Plata Publishing’);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(012,'The Five AM Club',’Robin Sharma’,’A Jaico Book’);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(013,'Turning Point',’APJ Abdul Kalam’,’Harper Collins’);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(014,'Think and Grow Rich',’Napolean Hill’,’FingerPrint’)
    -> ;
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(015,'Educated',’Tara Westover’,’Penguin Random House LCC’);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Non_Fiction values(026,'In Cold Blood','Truman Capote','Penguin Random House LCC');
Query OK, 1 row affected (0.01 sec)

mysql> insert into Non_Fiction values(027,'Becoming','Michelle Obama','Viking Press');
Query OK, 1 row affected (0.29 sec)

mysql> insert into Non_Fiction values(028,'Outlies','Malcolm Gladwell','Little Brown and Co');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(029,'Into Thin Air','Jon Krakaner','Villard');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Non_Fiction values(030,'Into the Wild','Jon Krakaner','Villard');
Query OK, 1 row affected (0.00 sec)
```

select * from Non_Fiction;



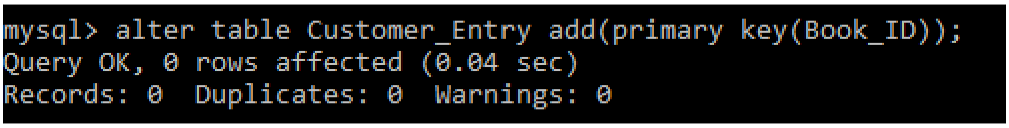
```
mysql> select * from Non_Fiction;
+---------+---------------------+-----------------+-------------------------+
| Book_ID | Book_name           | Author          | Publisher               |
+---------+---------------------+-----------------+-------------------------+
|      11 | Rich Dad Poor Dad   | Robert Kiyosaki | Plata Publishing        |
|      12 | The Five AM Club    | Robin Sharma    | A Jaico Book            |
|      13 | Turning Point       | APJ Abdul Kalam | Harper Collins          |
|      14 | Think and Grow Rich | Napolean Hill   | FingerPrint             |
|      15 | Educated            | Tara Westover   | Penguin Random House LCC |
|      26 | In Cold Blood       | Truman Capote   | Penguin Random House LCC |
|      27 | Becoming            | Michelle Obama  | Viking Press            |
|      28 | Outlies             | Malcolm Gladwell | Little Brown and Co    |
|      29 | Into Thin Air       | Jon Krakaner    | Villard                 |
|      30 | Into the Wild       | Jon Krakaner    | Villard                 |
+---------+---------------------+-----------------+-------------------------+
10 rows in set (0.00 sec)
```

create table Customer_Entry(Book_ID int(3),
Book_name varchar(20),
Issued_by varchar(30),
Issued_Date DATE,
Submission_Date DATE);



```
mysql>  create table Customer_Entry(Book_ID int(3),
    -> Book_name varchar(20),
    -> Issued_by varchar(30),
    -> Issued_Date DATE,
    -> Submission_Date DATE);
Query OK, 0 rows affected (0.01 sec)
```

alter table Customer_Entry add(primary key(Book_ID));



```
mysql> alter table Customer_Entry add(primary key(Book_ID));
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

insert into customer_entry values(017,'Delirium','Kartik Garg',20210706,20210712);

insert into customer_entry values(002,'The Hunger Games','Ruchi Kundra',20210709,NULL);

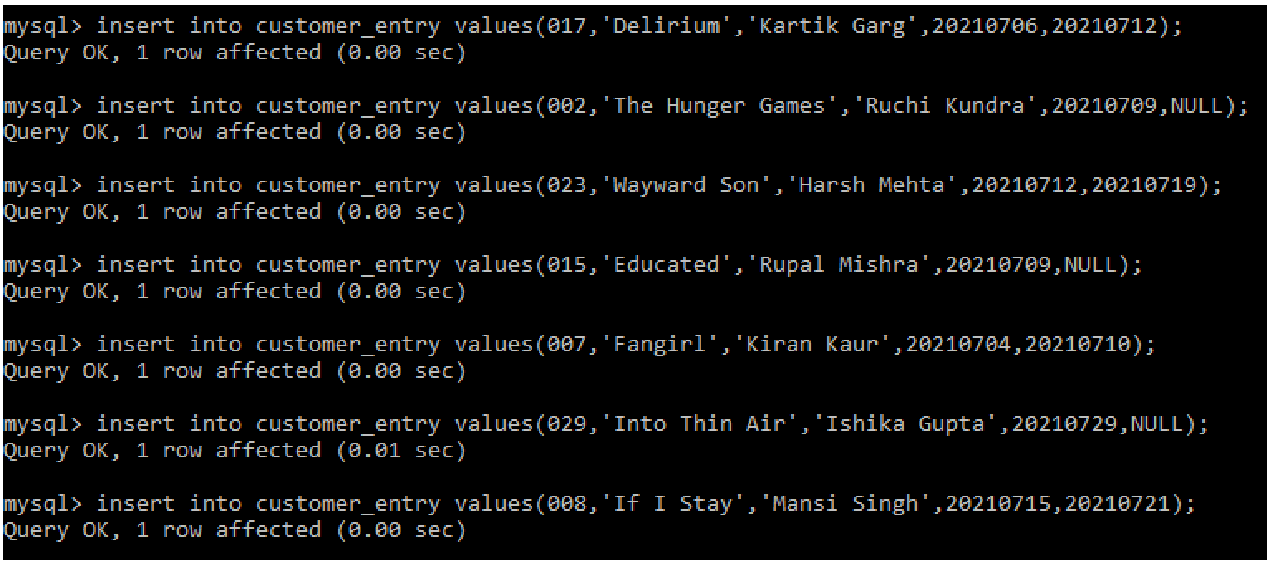
insert into customer_entry values(023,'Wayward Son','Harsh Mehta',20210712,20210719);

insert into customer_entry values(015,'Educated','Rupal Mishra',20210709,NULL);

insert into customer_entry values(007,'Fangirl','Kiran Kaur',20210704,20210710);

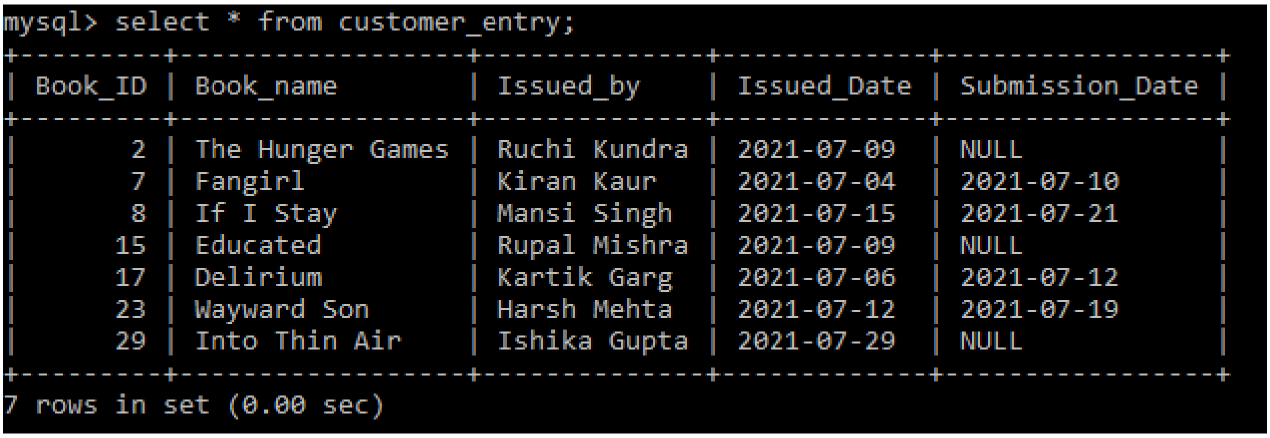
insert into customer_entry values(029,'Into Thin Air','Ishika Gupta',20210729,NULL);

insert into customer_entry values(008,'If I Stay','Mansi Singh',20210715,20210721);



```
mysql> insert into customer_entry values(017,'Delirium','Kartik Garg',20210706,20210712);
Query OK, 1 row affected (0.00 sec)

mysql> insert into customer_entry values(002,'The Hunger Games','Ruchi Kundra',20210709,NULL);
Query OK, 1 row affected (0.00 sec)

mysql> insert into customer_entry values(023,'Wayward Son','Harsh Mehta',20210712,20210719);
Query OK, 1 row affected (0.00 sec)

mysql> insert into customer_entry values(015,'Educated','Rupal Mishra',20210709,NULL);
Query OK, 1 row affected (0.00 sec)

mysql> insert into customer_entry values(007,'Fangirl','Kiran Kaur',20210704,20210710);
Query OK, 1 row affected (0.00 sec)

mysql> insert into customer_entry values(029,'Into Thin Air','Ishika Gupta',20210729,NULL);
Query OK, 1 row affected (0.01 sec)

mysql> insert into customer_entry values(008,'If I Stay','Mansi Singh',20210715,20210721);
Query OK, 1 row affected (0.00 sec)
```

select * from customer_entry;



```
mysql> select * from customer_entry;
+---------+------------------+--------------+-------------+-----------------+
| Book_ID | Book_name        | Issued_by    | Issued_Date | Submission_Date |
+---------+------------------+--------------+-------------+-----------------+
|       2 | The Hunger Games | Ruchi Kundra | 2021-07-09  | NULL            |
|       7 | Fangirl          | Kiran Kaur   | 2021-07-04  | 2021-07-10      |
|       8 | If I Stay        | Mansi Singh  | 2021-07-15  | 2021-07-21      |
|      15 | Educated         | Rupal Mishra | 2021-07-09  | NULL            |
|      17 | Delirium         | Kartik Garg  | 2021-07-06  | 2021-07-12      |
|      23 | Wayward Son      | Harsh Mehta  | 2021-07-12  | 2021-07-19      |
|      29 | Into Thin Air    | Ishika Gupta | 2021-07-29  | NULL            |
+---------+------------------+--------------+-------------+-----------------+
7 rows in set (0.00 sec)
```

PYTHON CODE

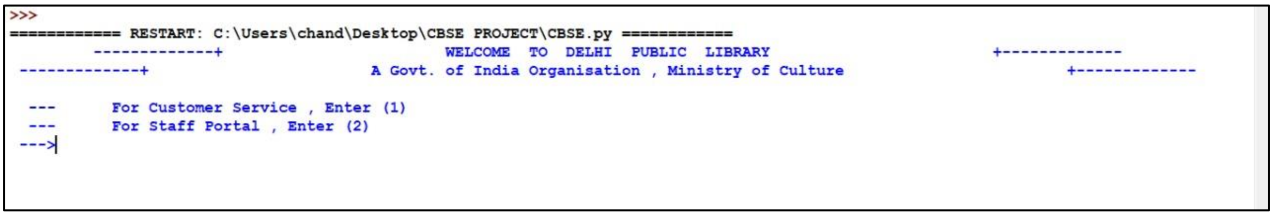
```
import time
from tabulate import tabulate
import mysql.connector
mydb=mysql.connector.connect(host="localhost",use
r="####",passwd="*****",database="cbse")
mycur=mydb.cursor()

#FUNCTIONS

def addbook():
    tab=input("Add Books to which data ? \n--->")
    while tab!="dystopia" and tab!="non fiction"
and tab!="young adult":
        print("ERROR (Last Try) Enter a valid
        response.")
        time.sleep(1)
        c=input("Which genre data ? \n --->")
        break
    if tab=="dystopia":
        n=int(input("How many entries to insert?
"))
        for j in range(n):
            ids=int(input("Enter Book ID --> "))
            g=input("Enter Book name ")
            h=input("Enter Author name ")
            i=input("Enter Publisher ")
            mycur.execute("insert into dystopia
values({},'{}','{}','{}')".format(ids,g,h,i))
            mydb.commit()
    elif tab=="non fiction":
        n=int(input("How many entries to insert ?
"))
        for j in range(n):
            ids=int(input("Enter Book ID --> "))
            g=input("Enter Book name ")
            h=input("Enter Author name ")
            i=input("Enter Publisher ")
            mycur.execute("insert into
non_fiction
values({},'{}','{}','{}')".format(ids,g,h,i))
            mydb.commit()
    elif tab=="young adult":
        n=int(input("How many entries to insert ?
"))
        for j in range(n):
            ids=int(input("Enter Book ID --> "))
            g=input("Enter Book name ")
            h=input("Enter Author name ")
            i=input("Enter Publisher ")
            mycur.execute("insert into
young_adult
values({},'{}','{}','{}')".format(ids,g,h,i))
            mydb.commit()
            time.sleep(0.5)
    print("Success")

def delete():
    n=input("Enter Book Name -->")
    mycur.execute("select * from dystopia")
    myrecords=mycur.fetchall()
    for r in myrecords:
        if n in r:
            mycur.execute("delete from dystopia
where book_name='"+n+"'")
            print("Success")
        else:
            time.sleep(0.1)
    mydb.commit()
    mycur.execute("select * from non_fiction")
    myrecords2=mycur.fetchall()
    for s in myrecords2:
        if n in s:
            mycur.execute("delete from
non_fiction where book_name='"+n+"'")
            print("Success")
        else:
            time.sleep(0.1)
    mydb.commit()
    mycur.execute("select * from young_adult")
    myrecords3=mycur.fetchall()
    for t in myrecords3:
        if n in t:
            mycur.execute("delete from
young_adult where book_name='"+n+"'")
            print("----Success----")
        else:
            time.sleep(0.2)
    mydb.commit()

def showall():
    mycur.execute("select * from Dystopia union
select * from Young_Adult union select * from
Non_Fiction order by book_id")
    result = mycur.fetchall()
    mydata=[[[result]]]
    head=["BOOK ID","BOOK
NAME","AUTHOR","PUBLISHER"]

print(tabulate(result,headers=head,tablefmt="fanc
y_grid"))

def showissue():
    mycur.execute("select * from customer_entry
order by book_id")
    result = mycur.fetchall()
    mydata=[[[result]]]
    head=["BOOK ID","BOOK NAME","ISSUED
BY","ISSUED DATE","SUBMISSION DATE"]

print(tabulate(result,headers=head,tablefmt="fanc
y_grid"))

def addissue():
    n=int(input("How many entries to insert ? "))
    for j in range(n):
        a1=int(input("Enter Book ID --> "))
        b1=input("Enter Book name ")
        c1=input("Enter Issuer's Name ")
        d1=input("Enter Issued Date (Format :
YYYY-MM-DD)  ")
        e1=input("Enter Submission Date (Format :
YYYY-MM-DD) ")
        mycur.execute("insert into customer_entry
values({},'{}','{}','{}','{}')".format(a1,b1,c1,d
1,e1))
        time.sleep(0.5)
        print("Success")
        mydb.commit()

def updatedata():
    mycur=mydb.cursor()
    print("Available Genres : \n 1.) dystopia \n
2.) non fiction \n 3.) young adult")
    an=input("Select Genre \n --->")
    while an!="dystopia" and an!="non fiction"
and an!="young adult":
        print("ERROR (Last Try) Enter a valid
response.")
        time.sleep(1)
        break
    if an=="dystopia":
        n=input("Which Book would you like to
Update ? (Enter Book Name) \n ---> ")
        r=int(input("Enter New Book ID ---> "))
        s=input("Enter New Book Name ---> ")
        t=input("Enter New Author Name ---> ")
        u=input("Enter New Publisher Name ---> ")
        sql="UPDATE non_fiction SET book_id=%s,
book_name=%s, author = %s, publisher=%s WHERE
book_name = %s"
        val=(r,s,t,u,n)
        mycur.execute(sql,val)
        mydb.commit()
    elif an=="non fiction":
        n=input("Which Book would you like to
Update ? (Enter Book Name) \n ---> ")
        r=int(input("Enter New Book ID ---> "))
        s=input("Enter New Book Name ---> ")
        t=input("Enter New Author Name ---> ")
        u=input("Enter New Publisher Name ---> ")
        sql="UPDATE non_fiction SET book_id=%s,
book_name=%s, author = %s, publisher=%s WHERE
book_name = %s"
        val=(r,s,t,u,n)
        mycur.execute(sql,val)
        mydb.commit()
    elif an=="young adult":
        n=input("Which Book would you like to
Update? (Enter Book Name) \n ---> ")
        r=int(input("Enter New Book ID ---> "))
        s=input("Enter New Book Name ---> ")
        t=input("Enter New Author Name ---> ")
        u=input("Enter New Publisher Name ---> ")
        sql="UPDATE non_fiction SET book_id=%s,
book_name=%s, author = %s, publisher=%s WHERE
book_name = %s"
        val=(r,s,t,u,n)
        mycur.execute(sql,val)
        mydb.commit()
    time.sleep(0.5)
    print("----Success----")

def ifbook():
    mycur.execute("select book_name from Dystopia
union select book_name from Young_Adult  union
select book_name from Non_Fiction")
    rows = mycur.fetchall()
    xy=input("Which book would like to Find ? \n
---->")
    for row in rows:
        if xy in row:
            time.sleep(0.5)
            print("Yes --",xy,"-- Book is
Available")
        elif xy not in rows:
            pass
```

```
def ifpublisher():
    mycur.execute("select publisher from Dystopia
union select publisher from Young_Adult  union
select publisher from Non_Fiction")
    rows = mycur.fetchall()
    xy=input("Which book would like to Find? \n -
--->")
    for row in rows:
        if xy in row:
            time.sleep(0.5)
            print("Yes --",xy,"-- Publications is
Available")
        elif xy not in rows:
            pass

#Entrance page

time.sleep(1)
a="-------------+                        WELCOME
TO  DELHI  PUBLIC  LIBRARY
+-------------"
b="-------------+                        A Govt.
of India Organisation , Ministry of Culture
+-------------"
print(a.center(130),"\n",b.center(100))

time.sleep(1)

print("\n"," ---      For Customer Service ,
Enter (1)","\n"," ---      For Staff Portal ,
Enter (2)")
c=int(input(" --->"))
time.sleep(0.5)

while c!=1 and c!=2:
    print("ERROR (Last Try) Enter a valid
response.")
    time.sleep(0.5)
    c=int(input(" --->"))
    break
```

```
#Customer service

if c==1:
    d="----------+       WELCOME TO CUSTOMER
SERVICE PORTAL        ----------+"
    print(d.center(100))
    time.sleep(0.5)
    mydata=[["Find Books by Names","(1)"],["Find
Books of Publications","(2)"]]
    head=["Features to Do","Code Number"]

print(tabulate(mydata,headers=head,tablefmt="fanc
y_grid"))
    print("Please Enter a Code feature to
Perform")

    f=int(input("--->"))
    while f!=1 and f!=2:
        print("ERROR (Last Try) Enter a valid
response.")
        time.sleep(1)
        f=int(input(" --->"))
        break
    if f==1:
        ifbook()
    elif f==2:
        ifpublisher()

    wh=input("Would you like to continue ?
(yes/no) \n --->")
    while wh=="yes":
        print("Please Enter a Code feature to
Perform")
        f=int(input("--->"))
        if f==1:
            ifbook()
        elif f==2:
            ifpublisher()
        wh=input("Would you like to continue ?
(yes/no) \n --->")

#staff can select out books by name or
publisher,insert more books,delete discarded
books,update any detail

elif c==2:
    e="-----------+     WELCOME TO STAFF PORTAL
-----------+"
    print(e.center(100))
    time.sleep(0.5)
    mydata=[["To Show Issued Book
History","(1)"],["To Add New Books
Issued","(2)"],["To Delete Discarded
Books","(3)"],["To Update a Book
Detail","(4)"],["To Display All the Books
Record","(5)"],["To Add Books To Library
Data","(6)"]]
    head=["Features to Do","Code Number"]

print(tabulate(mydata,headers=head,tablefmt="fanc
y_grid"))
    print("Please Enter a Code feature to
Perform")
    f=int(input("--->"))

    while f!=1 and f!=2 and f!=3 and f!=4 and
f!=5 and f!=6:
        print("ERROR (Last Try) Enter a valid
response.")
        time.sleep(1)
        f=int(input(" --->"))
        break

    if f==6:
        print("Available Genres : \n 1. dystopia
\n 2. non fiction \n 3. young adult")
        addbook()
    elif f==3:
        delete()
    elif f==5:
        showall()
    elif f==1:
        showissue()
    elif f==2:
        addissue()
    elif f==4:
        updatedata()

    wh=input("Would you like to continue ?
(yes/no) \n --->")
    while wh=="yes":
        print("Please Enter a Code feature to
Perform")
        f=int(input("--->"))
        if f==6:
            print("Available Genres : \n 1.
dystopia \n 2. non fiction \n 3. young adult")
            addbook()
        elif f==3:
            delete()
        elif f==5:
            showall()
        elif f==1:
            showissue()
        elif f==2:
            addissue()
        elif f==4:
            updatedata()
        wh=input("Would you like to continue ?
(Yes/No) \n --->")
```

ANALYZING THE CODE

As soon as the user runs the program, they are provided by two preferences to select from, based on the individuality of the user which are as follows:



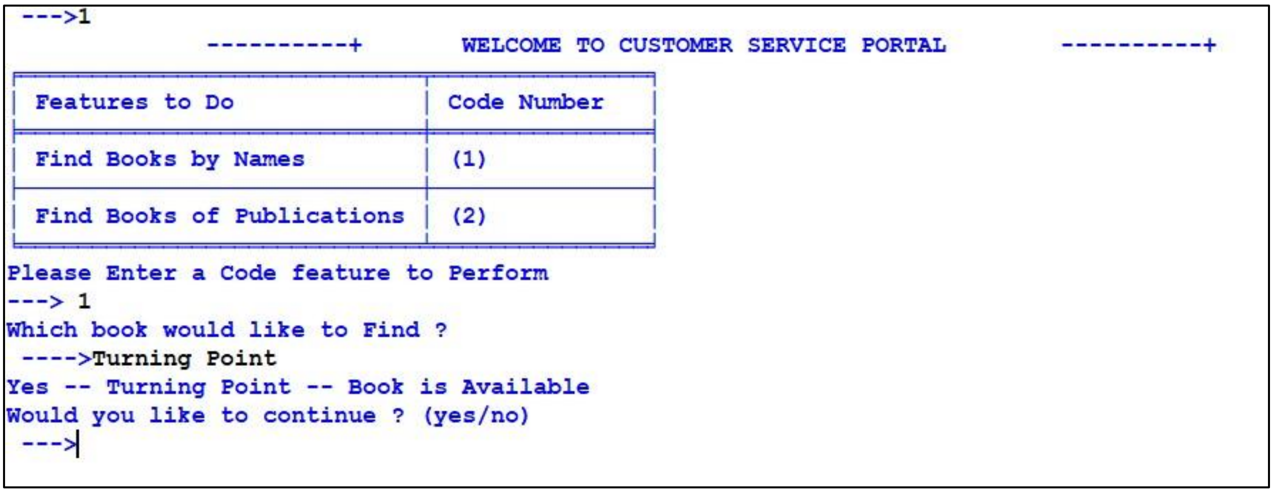
```
>>>
============ RESTART: C:\Users\chand\Desktop\CBSE PROJECT\CBSE.py ============
        -------------+                      WELCOME  TO  DELHI  PUBLIC  LIBRARY                 +-------------
 -------------+                      A Govt. of India Organisation , Ministry of Culture                 +-------------

  ---      For Customer Service , Enter (1)
  ---      For Staff Portal , Enter (2)
 --->
```

CUSTOMER SERVICE

When the user picks **'For Customer Service'**, they are offered two sub options:

FIND BOOKS BY NAME:

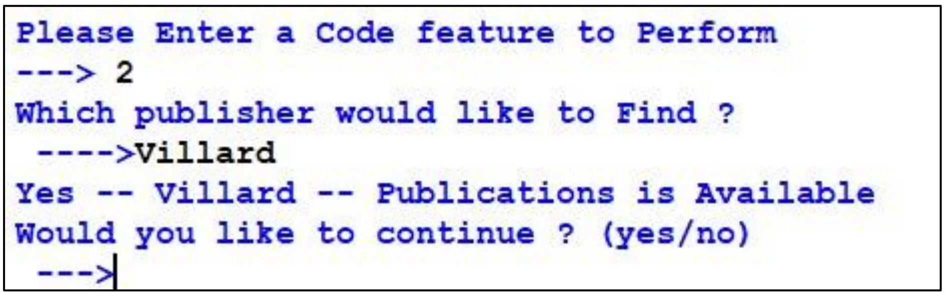
The customer can look for the availability of a specific book in the library by its name through this feature.



```
 --->1
             ----------+       WELCOME TO CUSTOMER SERVICE PORTAL        ----------+

| Features to Do                  | Code Number  |
| Find Books by Names             |  (1)         |
| Find Books of Publications      |  (2)         |

Please Enter a Code feature to Perform
---> 1
Which book would like to Find ?
 ---->Turning Point
Yes -- Turning Point -- Book is Available
Would you like to continue ? (yes/no)
 --->
```

FIND BOOKS BY PUBLICATIONS:

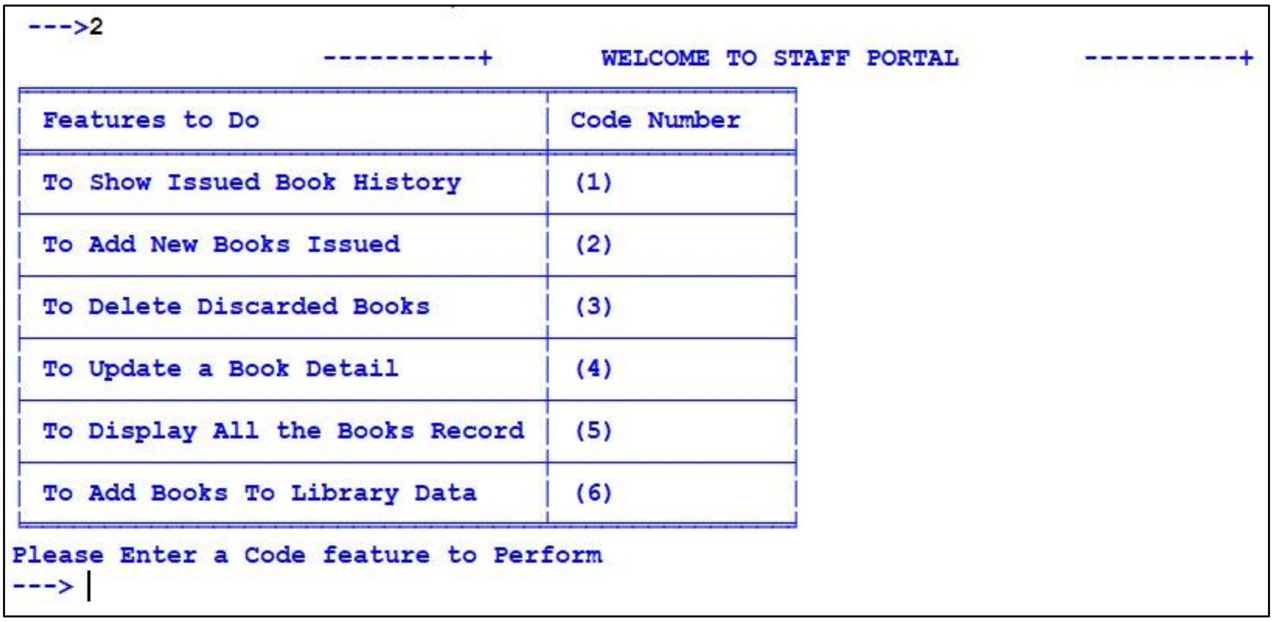
The customer can look for the availability of books in the library by the name of its publication through this feature.



```
Please Enter a Code feature to Perform
---> 2
Which publisher would like to Find ?
 ---->Villard
Yes -- Villard -- Publications is Available
Would you like to continue ? (yes/no)
 --->
```

STAFF PORTAL

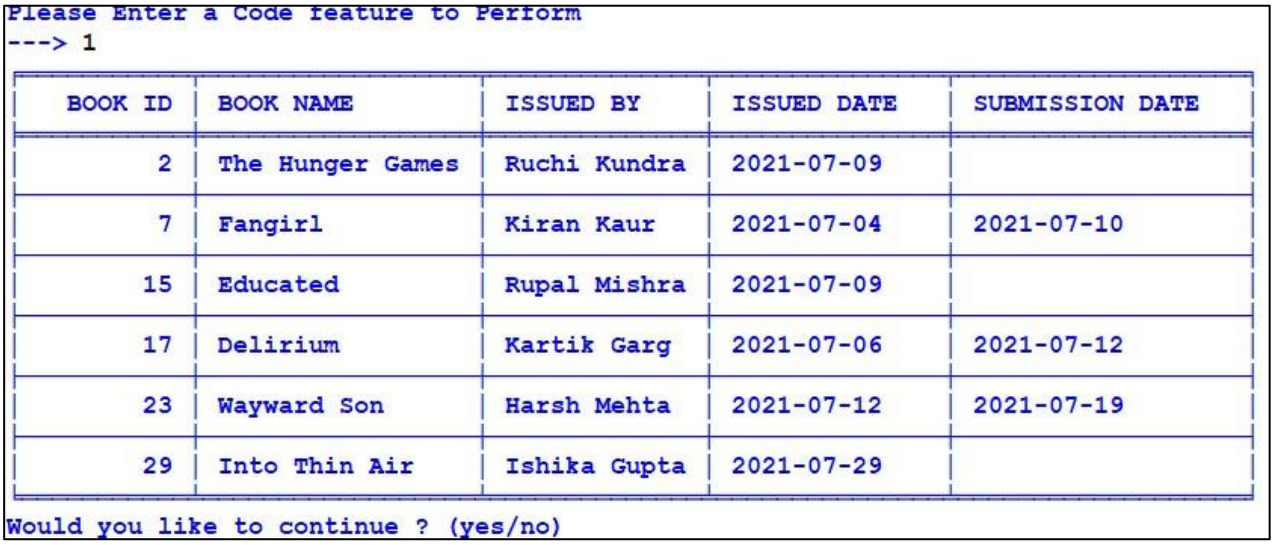
When the user picks **'For Staff Portal'**, they are offered six sub options:



```
 --->2
                    ----------+       WELCOME TO STAFF PORTAL       ----------+
+-----------------------------------+----------------+
| Features to Do                    | Code Number    |
+-----------------------------------+----------------+
| To Show Issued Book History       | (1)            |
+-----------------------------------+----------------+
| To Add New Books Issued           | (2)            |
+-----------------------------------+----------------+
| To Delete Discarded Books         | (3)            |
+-----------------------------------+----------------+
| To Update a Book Detail           | (4)            |
+-----------------------------------+----------------+
| To Display All the Books Record   | (5)            |
+-----------------------------------+----------------+
| To Add Books To Library Data      | (6)            |
+-----------------------------------+----------------+
Please Enter a Code feature to Perform
--->
```

To Show Issued Book History:

With the help of this feature, the library personnel can access the records stored about the issued books from the library data.



```
Please Enter a Code feature to Perform
---> 1
+------------+-------------------+--------------+----------------+-----------------+
|  BOOK ID   | BOOK NAME         | ISSUED BY    | ISSUED DATE    | SUBMISSION DATE |
+------------+-------------------+--------------+----------------+-----------------+
|          2 | The Hunger Games  | Ruchi Kundra | 2021-07-09     |                 |
+------------+-------------------+--------------+----------------+-----------------+
|          7 | Fangirl           | Kiran Kaur   | 2021-07-04     | 2021-07-10      |
+------------+-------------------+--------------+----------------+-----------------+
|         15 | Educated          | Rupal Mishra | 2021-07-09     |                 |
+------------+-------------------+--------------+----------------+-----------------+
|         17 | Delirium          | Kartik Garg  | 2021-07-06     | 2021-07-12      |
+------------+-------------------+--------------+----------------+-----------------+
|         23 | Wayward Son       | Harsh Mehta  | 2021-07-12     | 2021-07-19      |
+------------+-------------------+--------------+----------------+-----------------+
|         29 | Into Thin Air     | Ishika Gupta | 2021-07-29     |                 |
+------------+-------------------+--------------+----------------+-----------------+
Would you like to continue ? (yes/no)
```

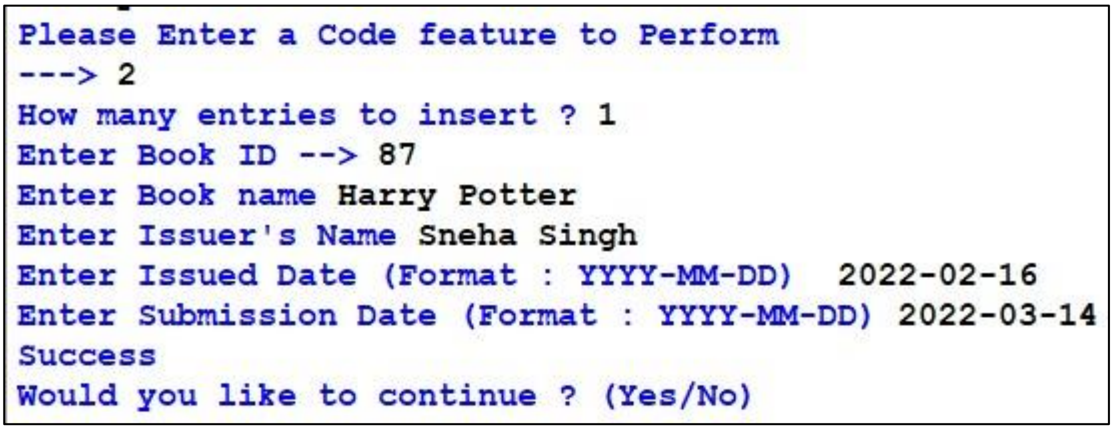
To Add New Book Issued:

This feature enables the library staff to update the issued book history record.

When this option enabled, it asks the user for the following data:

- The total number of entries to be inserted
- Book ID
- Book Name
- Name of the issuer
- Date of issuing
- Date of submission

When the information is rightly inserted, the program exhibits a ‘Success’ message.

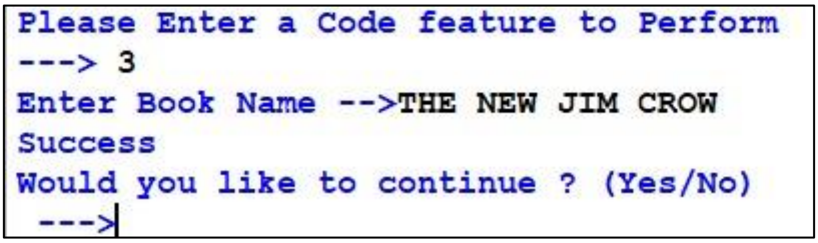


```
Please Enter a Code feature to Perform
---> 2
How many entries to insert ? 1
Enter Book ID --> 87
Enter Book name Harry Potter
Enter Issuer's Name Sneha Singh
Enter Issued Date (Format : YYYY-MM-DD)  2022-02-16
Enter Submission Date (Format : YYYY-MM-DD) 2022-03-14
Success
Would you like to continue ? (Yes/No)
```

TO DELETE DISCARDED BOOKS:

The following feature allows the library staff to delete a particular book record from the library data.

When the command is effectively executed, the program displays a **'Success'** message.



```
Please Enter a Code feature to Perform
---> 3
Enter Book Name -->THE NEW JIM CROW
Success
Would you like to continue ? (Yes/No)
 --->
```

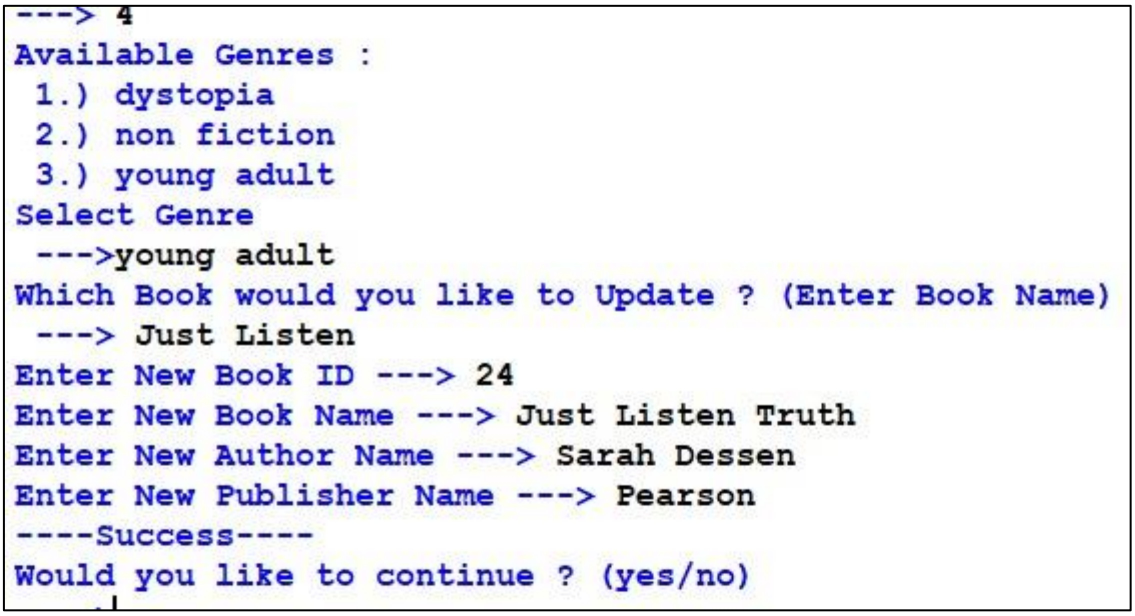
To Update a Book Detail:

This option when selected, lets the library personnel to update an existing book entry from the library data.

As the program proceeds, the staff is asked for the following details:

- Genre of the Book to be updated
- Name of the book to be updated
- New Book ID
- New Book name
- New Author name
- New Publisher name

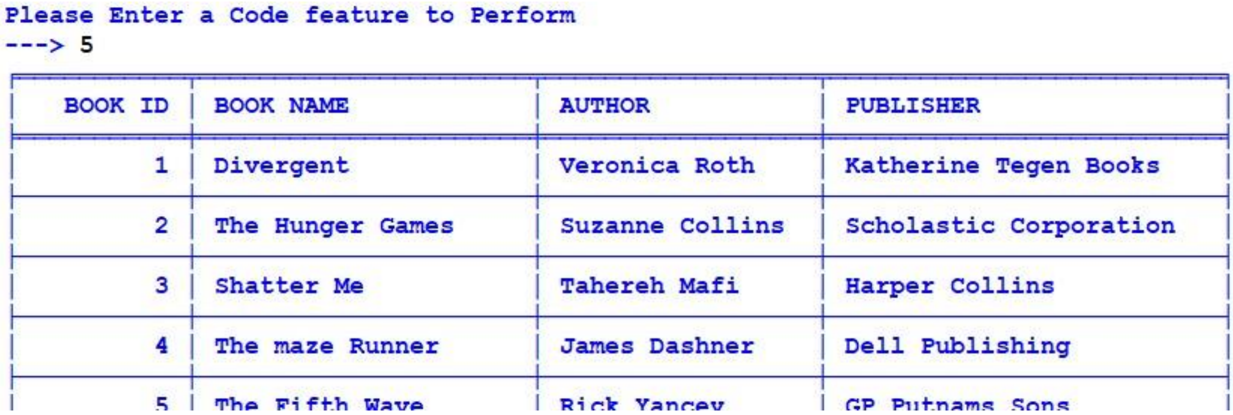
When the information is efficaciously updated, the program shows a **‘Success’** message.



```
---> 4
Available Genres :
 1.) dystopia
 2.) non fiction
 3.) young adult
Select Genre
 --->young adult
Which Book would you like to Update ? (Enter Book Name)
 ---> Just Listen
Enter New Book ID ---> 24
Enter New Book Name ---> Just Listen Truth
Enter New Author Name ---> Sarah Dessen
Enter New Publisher Name ---> Pearson
----Success----
Would you like to continue ? (yes/no)
```

TO DISPLAY THE RECORD TABLE:

This feature facilitates the library staff to print data about all the books from the library system in a tabular form.



```
Please Enter a Code feature to Perform
---> 5
+---------+--------------------+------------------+-----------------------+
|  BOOK ID| BOOK NAME          | AUTHOR           | PUBLISHER             |
+---------+--------------------+------------------+-----------------------+
|        1| Divergent          | Veronica Roth    | Katherine Tegen Books |
+---------+--------------------+------------------+-----------------------+
|        2| The Hunger Games   | Suzanne Collins  | Scholastic Corporation|
+---------+--------------------+------------------+-----------------------+
|        3| Shatter Me         | Tahereh Mafi     | Harper Collins        |
+---------+--------------------+------------------+-----------------------+
|        4| The maze Runner    | James Dashner    | Dell Publishing       |
+---------+--------------------+------------------+-----------------------+
|        5| The Fifth Wave     | Rick Yancey      | GP Putnams Sons       |
```

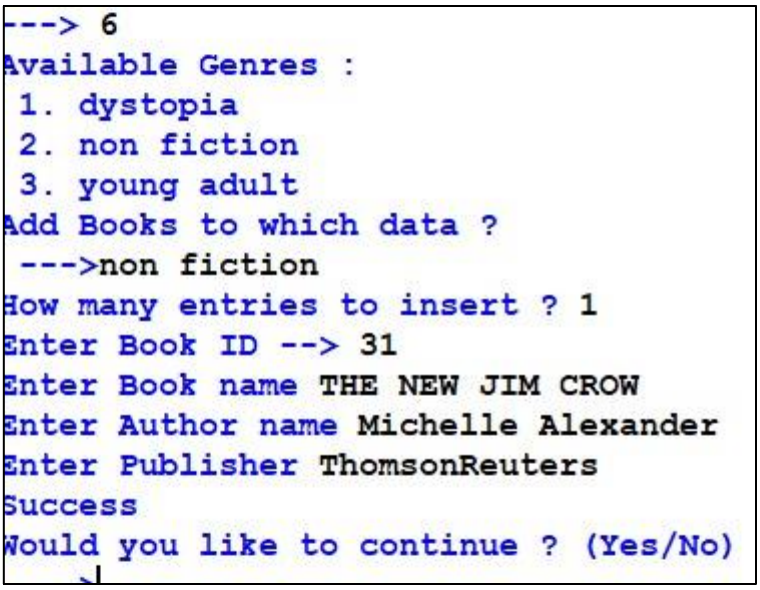
TO ADD BOOKS TO LIBRARY DATA:

The following option permits the library personnel to add new book entries in the library data according to their genre.

To add new book entry in the data, following details are queried:

- Genre of the book
- Total number of fresh entries
- Book ID
- Book name
- Author name
- Publisher name

As the data is effectively added, the program displays a **'Success'** message.



```
---> 6
Available Genres :
 1. dystopia
 2. non fiction
 3. young adult
Add Books to which data ?
 --->non fiction
How many entries to insert ? 1
Enter Book ID --> 31
Enter Book name THE NEW JIM CROW
Enter Author name Michelle Alexander
Enter Publisher ThomsonReuters
Success
Would you like to continue ? (Yes/No)
```

ELEMENTS OF THE CODE:

LIBRARIES USED

- time
- mysql.connector
- tabulate

USER DEFINED MODULES/FUNCTIONS USED

- addbook
- delete
- showall
- showissue
- addissue
- updatedata
- ifbook
- ifpublisher

BUILT IN MODULES/FUNCTIONS USED

- mysql.connector
- mysql.connector.connect
- time.sleep()
- mydb.cursor()
- mycur.execute()
- mydb.commit()
- mycur.fetchall()
- input()
- print()
- tabulate.tabulate()

DATABASE AND TABLES USED

- DATABASE USED: CBSE
- TABLES USED:
  - Dystopia
  - Non_Fiction
  - Young_Adult
  - Customer_Entry

SOFTWARE AND HARDWARE SPECIFIC INFORMATION REQUIRED

PYTHON

Python Version: 3.7.4

Python 3.7.4 was the fourth bug fix release of Python 3.7.

IDLE Version: 3.7.4

Release Date: July 8, 2019

MYSQL

Version:	**5.5.62**
Created:	Tue Nov 13 22:19:38 UTC 2018
Last modified:	Sat May 15 09:20:03 UTC 2021