|  |
| --- |
| Close-up image showing the leaf-sides of two oversized books side-by-side on a bookshelf, with additional books in soft focus background |
| LIBRARY MANAGEMENT SYSYTEM |
| |  |  |  | | --- | --- | --- | | SHREYANSH JAIN |  | 12 A1 | |

**DELHI PUBLIC SCHOOL GHAZIABAD**



**ACADEMIC YEAR: 2020-21**

**PROJECT REPORT ON**

**LIBRARY MANAGEMENT SYSTEM**

**NAME : SHREYANSH JAIN**

**CLASS : XII A1**

**SUBJECT : COMPUTER SCIENCE**

**SUB CODE : 083**

**PROJECT GUIDE: Mrs Madhu Singh**

**PGT (CS)**

## DELHI PUBLIC SCHOOL GHAZIABAD



# **CERTIFICATE**

This is to certify that **SHREYANSH JAIN** has successfully completed the Project Work entitled **LIBRARY MANAGEMENT SYSTEM** in the subject Computer Science (083) laid down in the regulations of CBSE for the purpose of Practical Examination in Class XII for the academic year 2020-2021 under guidance of **Mrs. MADHU SINGH**

**SIGNATURE**

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS [ T O C ]** | | |
| **S.No,** | **DESCRIPTION** | **PAGE NO** |
| 01 | ACKNOWLEDGEMENT | **04** |
| 02 | INTRODUCTION | **05** |
| 03 | SOURCE CODE | **08** |
| 04 | OUTPUT | **18** |
| 05 | SOFTWRAE REQUIRED | **26** |
| 06 | BIBLIOGRAPHY | **26** |

**ACKNOWLEDGEMENT**

Apart from the efforts of me, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I express my heartfelt gratitude to my parents for constant encouragement while carrying out this project.

My sincere thanks to **MADHU SINGH MA’AM**, A guide, Mentor ,who critically reviewed my project and helped in solving each and every problem, occurred during implementation of the project.

The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the project. I am grateful for their constant support and help.

**PROJECT ON LIBRARY MANAGEMENT SYSTEM**

**INTRODUCTION**

This programme helps the users to manage and store books details. The system helps both students and library manager to keep a constant track of all the books available in the library. It allows both the admin and the student to search for the desired book. This programme can be used only if the Student Library ID and the Password is correct by doing so, we enforce security and restriction of unauthorised access.

­­­

**OBJECTIVES OF THE PROJECT**

The objective of this project is to let the students apply the programming knowledge into a real- world situation/problem and exposed the students how programming skills helps in developing a good software.

1. Write programs utilizing modern software tools.
2. Apply object oriented programming principles effectively when developing small to medium sized projects.
3. Write effective procedural code to solve small to medium sized problems.
4. Students will demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.
5. Students will demonstrate ability to conduct a research or applied Computer Science project, requiring writing and presentation skills which exemplify scholarly style in computer science.

**PROPOSED SYSTEM**

Today one cannot afford to rely on the fallible human beings of be really wants to stand against today’s merciless competition where not to wise saying **“to err is human”** no longer valid, it’s out-dated to rationalize your mistake. So, to keep pace with time, to bring about the best result without malfunctioning and greater efficiency so to replace the unending heaps of flies with much sophisticated hard disk of the computer.

One has to use the data management software. Software has been an ascent in atomization various organisations. Many software products working are now in markets, which have helped in making the organizations work easier and efficiently. Data management initially had to maintain a lot of ledgers and a lot of paperwork has to be done but now software production this organization has made their work faster and easier. Now only this software has to be loaded on the computer and work can be done.

This prevents a lot of time and money. The work becomes fully automated and any information regarding the organization can be obtained by clicking the button. Moreover, now it’s an age of computers of and automating such an organization gives the better look.

**MODULES USED**

* admin\_mm(): Admin Main Menu
* stud\_mm(libid,name): Student Main Menu
* admlogin(): Admin Login
* stulogin(): Student Login
* bsearch(): To Search a Book in Library
* Addbook(): To Add a Book in Library
* Student\_detail(): To Add New Student Details
* Addissue(): To Issue a Book From Library
* UpdateStu(): To Update any Student Detail
* UpdateBook(): To Update any Book Detail
* Search(): To Search for Issued Book
* return\_book(): To Return a Issued Book
* OutputBook(): To Display List of Books
* OutputStu(): To Display List of Student
* OutputIssue(): To Display List of Issued Books

**SOURCE CODE**

'''

Default admin password is admin

Default student password is its LibID

LibId 101 Sakshi has borrowed bookid 1 computer science with python (For testing purposes)

This program uses Tabulate package, open cmd, type 'pip install tabulate'

change mysql password according to your device

'''

import mysql.connector as mysql

import datetime

mycon=mysql.connect(host="localhost",user="root",passwd="",database="library")

cursor=mycon.cursor()

cursor.execute("drop database library;")

cursor.execute("create database library;")

cursor.execute("use library;")

from tabulate import tabulate

'''Main Menu'''

#admin

def admin\_mm():

print(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

print("| |")

print("|\*\*\*\*\*\*\*\* MAIN MENU \*\*\*\*\*\*\*\*\*\*|")

print("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|")

rep=''

while (True):

print("Select")

print("1) Add Books")

print("2) To Enter Student Details")

print("3) To Issue book ")

print("4) To return Book ")

print("5) To Search Book ")

print("6) To Update Student Records")

print("7) To Update Books record")

print("8) Display Book List")

print("9) Display Issue List")

print("10)Display Student List")

ch=int(input("enter your choice "))

if ch==1:

Addbook()

OutputBook()

if ch==2:

Student\_detail()

OutputStu()

if ch==3:

Addissue()

OutputIssue()

if ch==4:

return\_book()

OutputIssue()

if ch==5:

Search()

if ch==6:

UpdateStu()

OutputStu()

if ch==7:

UpdateBook()

OutputBook()

if ch==8:

OutputBook()

if ch==9:

OutputIssue()

if ch==10:

OutputStu()

rep=input('Want To Continue (y/n)')

if(rep!='y'):

break

print("THANK YOU")

print("DO VISIT AGAIN")

#student

def stud\_mm(libid,name):

while True:

print(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

print("| |")

print("|\*\*\*\*\*\*\*\* MAIN MENU \*\*\*\*\*\*\*\*\*\*|")

print("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|")

print("LibID: ",libid," Name: ",name,'\n')

issued="select \* from issue where libid=%s;"

date=datetime.datetime.now().date()

cursor.execute(issued,[libid])

found=cursor.fetchall()

if found:

print("Books currently borrowed: \n")

headers=['Name','Issue date','Due date']

val=[]

v=[]

alert=0

for i in found:

cursor.execute("select \* from booklist where bookid=%s;",[i[1]])

book=cursor.fetchall()

tup=(book[0][1],i[2],i[3])

if (i[3]-date).days<=3:

t=(book[0][1],(i[3]-date).days,i[3])

v.append(t)

alert=1

val.append(tup)

print(tabulate(val,headers))

if alert==1:

print("\nReturn Date approaching soon for these books: \n")

print(tabulate(v,['Name','Days left','Due date']))

else:

print("You currently have no books issued. Borrow one and start reading!!")

print("\n\n1)Search Books\n2)Change pwd\n3)Exit(other options to be added)")

ch=int(input("\nEnter your choice : "))

if ch==1:

bsearch()

if ch==2:

while True:

npwd=input("Enter new password: ")

if npwd==input("Confirm password: "):

update="update student set pwd=%s where libid=%s;"

cursor.execute(update,[npwd,libid])

print("Successfully Updated!!")

break

else:

print("passwords don't match!!")

if ch==3:

break

''' Login module'''

#admin

def admlogin():

while True:

try:

unm=input("Enter username - ")

pwd=input("Enter password - ")

find=("select \* from users where uname=%s and pwd=%s;")

cursor.execute(find,[unm,pwd])

result=cursor.fetchall()

if result:

for i in result:

print("WELCOME ",i[2])

admin\_mm()

return(1)

else:

print("Username or password is incorrect!!")

retry=input("Do you want to retry? (y/n)")

if retry.lower()=='n':

break

except:

print("ERROR!!")

retry=input("Do you want to retry? (y/n)")

if retry.lower()=='n':

break

#student

def stulogin():

while True:

try:

unm=input("Enter LibID - ")

pwd=input("Enter password - ")

find=("select \* from student where LibID=%s and pwd=%s;")

cursor.execute(find,[unm,pwd])

result=cursor.fetchall()

if result:

for i in result:

print("WELCOME ",i[1])

stud\_mm(i[0],i[1])

break

else:

print("Username or password is incorrect!!")

retry=input("Do you want to retry? (y/n)")

if retry.lower()=='n':

break

except:

print("ERROR!!")

retry=input("Do you want to retry? (y/n)")

if retry.lower()=='n':

break

# Book search for Student

def bsearch():

while True:

bname='%'+input("Enter book to be searched: ")+'%'

query="select \* from booklist where bookname like %s;"

cursor.execute(query,[bname])

find=cursor.fetchall()

status=""

a="Issued by someone"

b="Available"

if find:

val=[]

for i in find:

bid=i[0]

query="select \* from issue where bookid=%s;"

cursor.execute(query,[int(bid)])

issued=cursor.fetchall()

if issued:

status=a

else:

status=b

tup=(i[1],i[2],i[3],status)

val.append(tup)

print("Search Results:-\n")

print(tabulate(val,['Name','Author','Publisher','Status']),'\n')

ch=input("Search again?? (y/n)")

if ch.lower()=='n':

break

else:

ch=input("NOT FOUND!! Retry with a different keyword? (y/n)")

if ch.lower()=='n':

break

#login table

cursor.execute("create table users(uname varchar(20) primary key,pwd varchar(20),name varchar(50));")

cursor.execute("insert into users values('admin','admin','ADMIN');")

#table booklist

cursor.execute("create table booklist(Bookid int primary key,Bookname varchar(50),Author varchar(50),Publisher varchar(50),Subject varchar(10),Copies int);")

s="insert into booklist values(%s,%s,%s,%s,%s,%s)"

books=[(1,'Computer Science With Python','Sumitra Arora','Dhanpat Rai Publisher','CS',20),(2, "Concept of Physics 1", "HC Verma","Bharti Bhawan Publisher","Physics",15),(3, "Concept of Physics 2", "HC Verma","Bharti Bhawan Publisher","Physics",15),(4, "Mathematics XII", "RD Sharma","Dhanpat Rai Publisher","Maths",20),(5, "Fault In our Stars", "John Green","Penguin Publishers","Psychology",10)]

cursor.executemany(s,books)

mycon.commit()

#table student

cursor.execute("create table Student(libid int primary key,name varchar(50),class varchar(50),adm int,pwd varchar(20));")

s="insert into student values(%s,%s,%s,%s,%s)"

stu=[(100,'Raghav','12 A3',12331,'100'),(101,"Sakshi","12 C1",21322,'101'),(102,"Amar","12 H2",19241,'102'),(103,"Amit","12 A5",19532,'103'),(104,"Nia","12 A6",12231,'104')]

cursor.executemany(s,stu)

mycon.commit()

#table issue

cursor.execute("create table Issue(libid int references student(libid),bookid int references booklist(bookid),issue\_date date,return\_date date);")

s="insert into issue values(%s,%s,%s,%s);"

stu=[101,1,'2020-08-26','2020-08-29']

cursor.execute(s,stu)

mycon.commit()

#update book list

def Addbook():

bid=int(input("ENTER BOOK ID "))

bnm=input("Enter book name ")

author=input("Enter author ")

publ=input("Enter publisher ")

sub=input("Enter Subject ")

copy=int(input("No. of copies "))

query="insert into booklist values({},'{}','{}','{}','{}',{})".format(bid,bnm,author,publ,sub,copy)

cursor.execute(query)

mycon.commit()

#update student details before issuing

def Student\_detail():

libid=int(input("Enter Student ID: "))

name=input("Enter Student Name: ")

clas=input("Enter Student Class: ")

Adm=int(input("Enter Student Addmission: "))

pwd=input("Enter temporary password: ")

query="insert into student values({},'{}','{}',{},'{}')".format(libid,name,clas,Adm,pwd)

cursor.execute(query)

mycon.commit()

#issued book details

def Addissue():

libid=int(input("Enter Student ID "))

bid=int(input("Enter Book ID "))

Idate=datetime.datetime.now().date()

Rdate=input("Enter Return date(YYYY-MM-DD) ")

query="insert into issue values({},{},'{}','{}')".format(libid,bid,Idate,Rdate)

cursor.execute(query)

mycon.commit()

def UpdateStu():

c=int(input("Enter the student id whose detail are to be updated "))

print("what do you want to be update?")

print("1)Name")

print("2)Class")

print("3)Admission number")

ch=int(input("Enter Choice "))

if(ch==1):

nm=input("Enter The Name")

s="Update student SET name=%s where libid=%s"

data=(nm,c)

cursor.execute(s,data)

mycon.commit()

if(ch==2):

cl=input("Enter The Class")

s="Update student SET class=%s where libid=%s"

data=(cl,c)

cursor.execute(s,data)

mycon.commit()

if(ch==3):

nm=int(input("Enter The Admission Number"))

s="Update student SET adm=%s where libid=%s"

data=(nm,c)

cursor.execute(s,data)

mycon.commit()

def UpdateBook():

c=int(input("Enter the book id whose detail are to be updated "))

print("what do you want to be update?")

print("1)BOOK Name")

print("2)Author")

print("3)No. of copies")

ch=int(input("Enter Choice "))

if(ch==1):

nm=input("Enter The Name")

s="Update booklist SET bookname=%s where bookid=%s"

data=(nm,c)

cursor.execute(s,data)

mycon.commit()

if(ch==2):

cl=input("Enter The Author Name")

s="Update booklist SET author=%s where bookid=%s"

data=(cl,c)

cursor.execute(s,data)

mycon.commit()

if(ch==3):

nm=int(input("Enter The Copies Available"))

s="Update booklist SET copies=%s where bookid=%s"

data=(nm,c)

cursor.execute(s,data)

mycon.commit()

def Search():

while(True):

print("1)To search for a book")

print("2)To search of an issued book")

ch=int(input("Enter your choice"))

if ch==1:

c=int(input("Enter the book id "))

s="select \* from booklist where bookid={}".format(c)

cursor.execute(s)

r=cursor.fetchone()

if r==():

print("book not present")

else:

print(r)

if ch==2:

c=int(input("Enter the book id "))

s="select \* from issue where bookid={}".format(c)

cursor.execute(s)

r=cursor.fetchone()

if r==():

print("book not present")

else:

print(r)

rep=input("want to search more books")

if rep!='y':

break

def return\_book():

c=int(input("Enter the book id "))

s="delete from issue where bookid={}".format(c)

cursor.execute(s)

mycon.commit()

def OutputBook():

cursor.execute("select \* from booklist")

data=cursor.fetchall()

head=['Bookid','Bookname','Author','Publisher','Subject','Copies']

print(tabulate(data,head))

def OutputStu():

cursor.execute("select \* from Student")

data=cursor.fetchall()

print(tabulate(data,['libid','name','class','adm','pwd']))

def OutputIssue():

cursor.execute("select \* from issue")

data=cursor.fetchall()

print(tabulate(data,['libid','bookid','issue\_date','return\_date']))

#Main

while True:

print(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

print("| |")

print("|\*\*\*\*\*\*\*\* LOGIN \*\*\*\*\*\*\*\*\*\*|")

print("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n")

print("1) Student Login 2) Admin Login\n3) Exit \n")

choice=int(input("Enter your choice: "))

if choice==1:

stulogin()

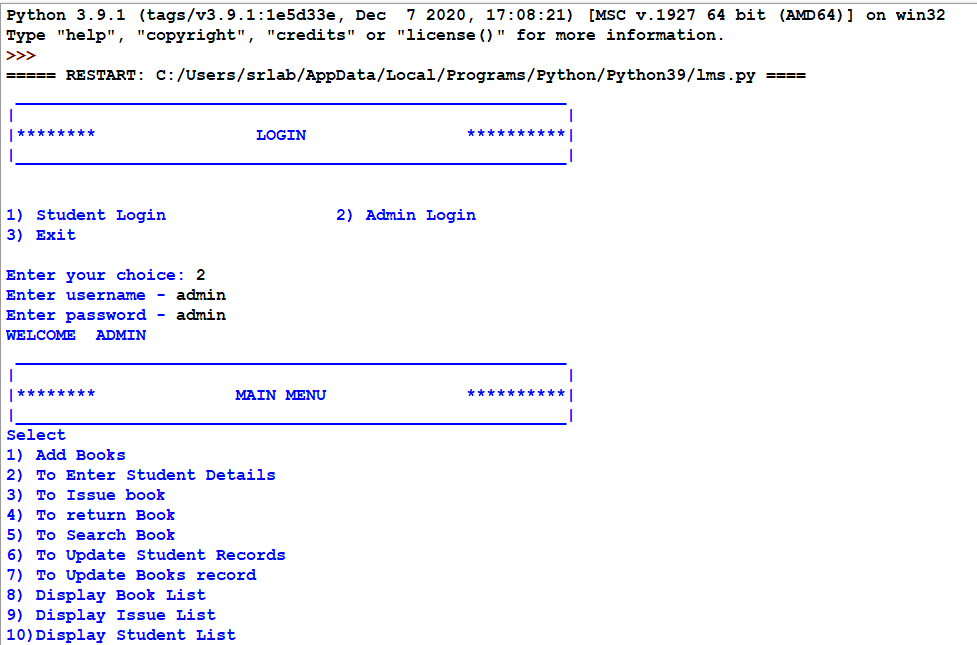
if choice==2:

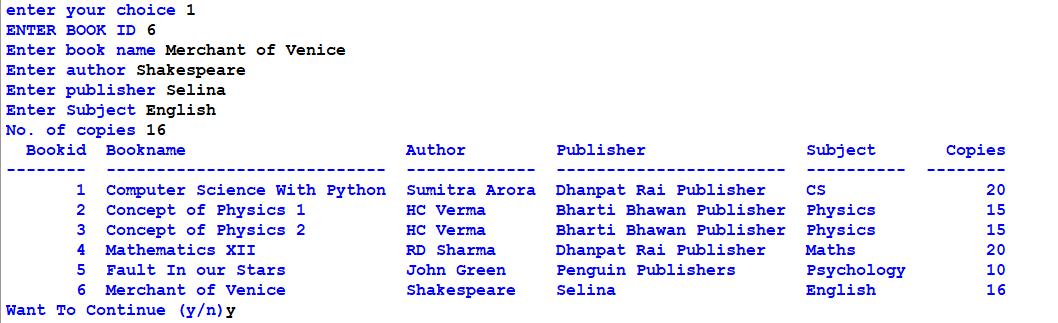
admlogin()

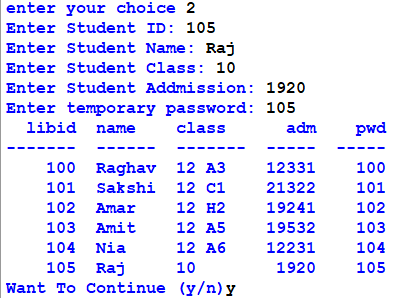
if choice==3:

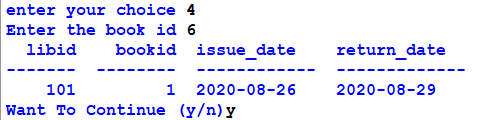
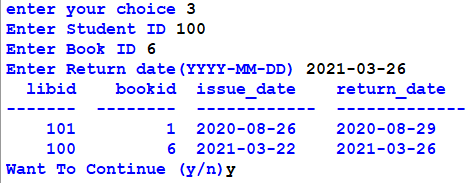
break

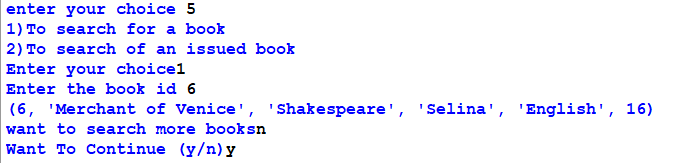
**OUTPUT**

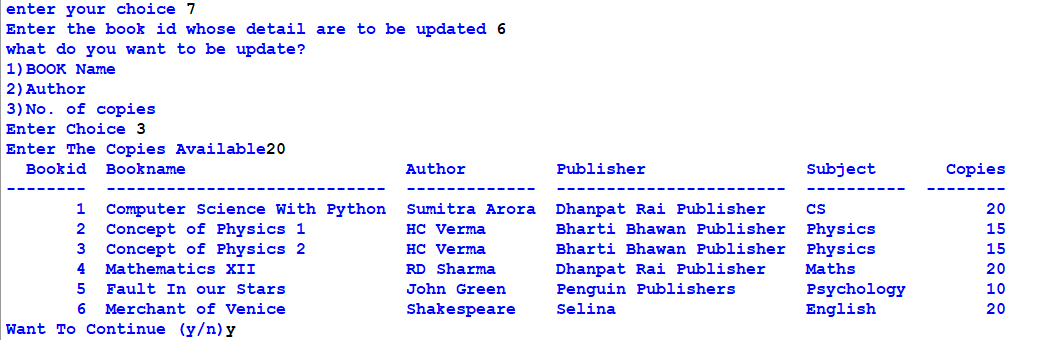
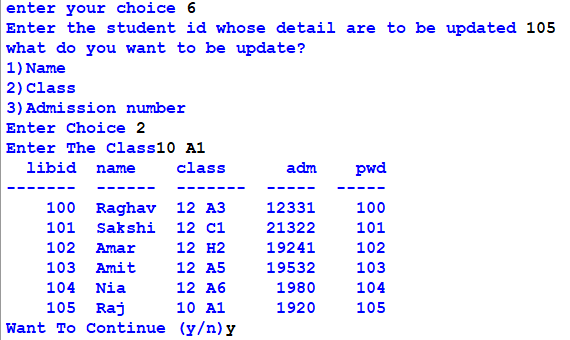


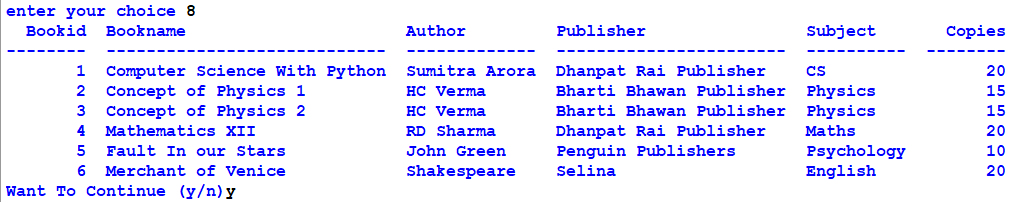


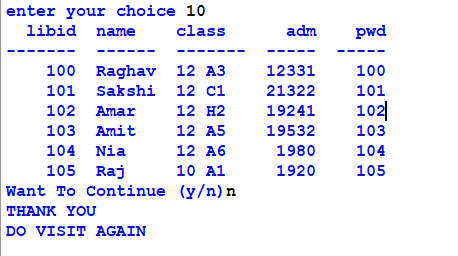
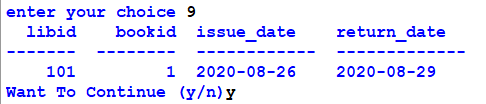


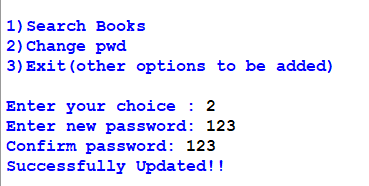
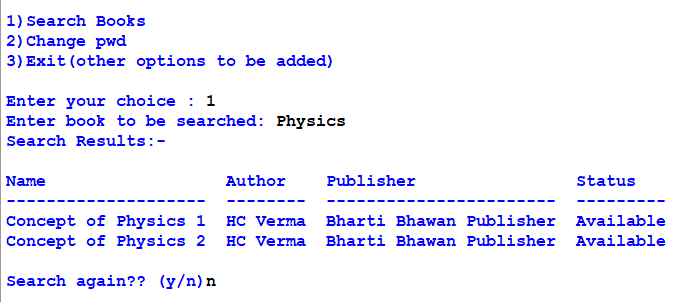
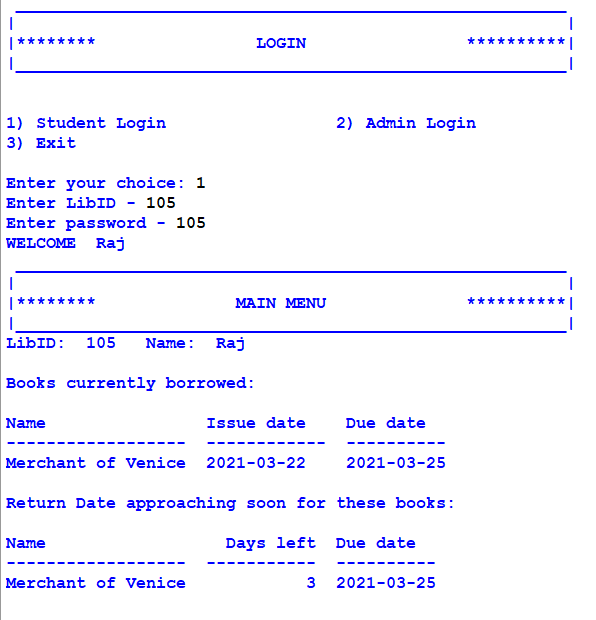


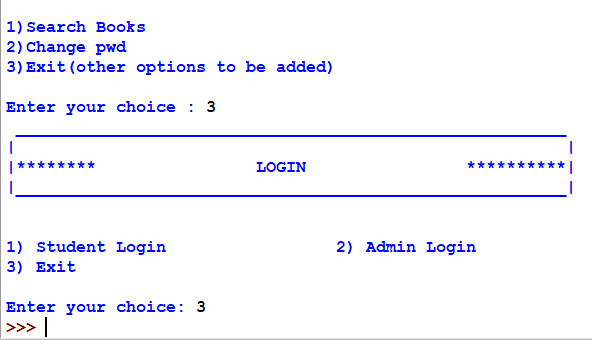












**TESTING**

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test , with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

It can also be stated as the process of validating and verifying that a software program/application/product meets the business and technical requirements that guided its design and development, so that it works as expected and can be implemented with the same characteristics. Software Testing, depending on the testing method employed, can be implemented at any time in the development process, however the most test effort is employed after the requirements have been defined and coding process has been completed.

**HARDWARE AND SOFTWARE REQUIREMENTS**

I.OPERATING SYSTEM : WINDOWS 7 AND ABOVE

II. PROCESSOR : PENTIUM(ANY) OR AMD

ATHALON(3800+- 4200+ DUALCORE)

III. MOTHERBOARD : 1.845 OR 915,995 FOR PENTIUM 0R MSI

K9MM-V VIAK8M800+8237R PLUS CHIPSET FOR AMD ATHALON

IV. RAM : 512MB+

V. Hard disk : SATA 40 GB OR ABOVE

VI. CD/DVD r/w multi drive combo: (If back up required)

VII. FLOPPY DRIVE 1.44 MB : (If Backup required)

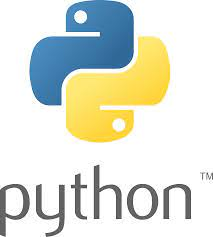
VIII. MONITOR : 14.1 or 15 -17 inch

IX. Keyboard and mouse

X. Printer : (if print is required – [Hard copy])

**SOFTWARE REQUIREMENTS:**

1. Windows OS
2. Python
3. MySQL

**BIBLIOGRAPHY**

**1.**Computer science With Python - Class XII

By : PREETI Arora

**2**.NCERT class 12

**3**.www.google.com

**FUTURE SCOPE**

In any project , the present satisfaction is important but visualising the future scope is most important . The above source code is the output of our effort in the provided limited time and thus may have some shortcomings which are needed to be overcome for the complete enforcement and usage of this project therefore any suggestions are most welcome .

