SVKM's NARSEE MONJEE INSTITUTE OF MANAGEMENT STUDIES

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT AND ENGINEERING

A REPORT ON

Python Project (Library Management System)

BY:

Narendra Raju Mali (B222)
M. Pratyush (B229)
Swaraj Borkar (B232)
COURSE: B(TECH.) COMPUTER ENGINEERING

Under the supervision of

Prof. Sachin Bhandari Sir

Abstract:

The Library Management System In Python is a simple project developed using Python as frontend. This is a simple console-base project which is very easy to understand and use. Library management system is a project which aims in developing a computerized system to maintain all the daily work of library.

Every facet of a library's operations is managed by a library management system. It helps librarians keep a database of freshly published books and books that students have borrowed, among other things such as return book. Also, this project makes it easy for the library to keep the records of books, borrowing and returning of books in a digital way.

Introduction:

A library management system is made to oversee every aspect of a library's operations. Along with other capabilities, it aids librarians in maintaining a database of newly released books and books that students have borrowed. Each and every activity in your library is fully automated by this system. It has following functionalities:

For Admin:

1) Add Admin-It will add new admin
2) Add New Book in library

For Student:

1) List of available books in library- In this feature user will be able to see the list of books available in the library.
2) Borrow the book- In this feature user will be able to borrow the books from library.
3) Return the book- In this feature user will be able to return the books within the 30 days.
4) Exit- In this feature user can exit the library after done with his work.

Aim:

"Our aim is to develop a project for the users in which a user can see the available books, can request the books, can return the books."

Objective:

The main objective of "Library Management System" is to facilitate a user friendly environment for all users and reduces the manual effort. The primary objective of library system is to collect, store, organize, retrieve and make available the information sources to the information users.

Modules:

Driver Code:

```
# driver code
if __name__ == "__main__":
    print("--------------Welcome To Library--------------")
    while (True):
        user_type = '''
        1)Admin
        2)Student
        3)Exit
        '''
        print(user_type)
        user = int(input("Enter Type Of User: "))
        if user == 1:
            user_name = input("Enter Username: ")
            pass_word = int(input("Enter Password: "))
            user_check(user_name, pass_word)
        elif user == 2:
            log_sign=int(input("1)Log in\n2)Sign up"))
            if log_sign==1:
                user_name = input("Enter Username: ")
                pass_word = int(input("Enter Password: "))
                student_check(user_name, pass_word)
            elif log_sign==2:
                new_user = input("Enter Username: ")
                new_pass = int(input("Enter Password: "))
                student_info[new_user]=new_pass
                print(f"Your account has been created. Have a great day!")
            else:
```

```
        elif user == 3:
            exit()
        else:
            print("Enter Proper Choice.")
```

Admin Modules:

```
def user_check(username, password):
    if username in user_info.keys():
        if password == user_info[username]:
            print("You are log in.\n")
            while (True):
                print("1)Add Admin\n2)Add New Books To Library\n3)exit\n")
                choice = int(input("Enter Your Choice: "))
                if choice == 1:
                    add_admin()
                elif choice == 2:
                    add_new_books()
                elif choice == 3:
                    break
                else:
                    print("Enter Proper Choice.")
    else:
        print("Invalid Credentials.")
        re_user_check()


def re_user_check():
    re_user_name = input("Enter Username: ")
    re_pass_word = int(input("Enter Password: "))
    user_check(re_user_name, re_pass_word)


def add_admin():
    print(user_info)
    username = input("Enter Username: ")
    password = int(input("Enter Password: "))
    user_info[username] = password
    print(f"{username} has been added as admin.\n")
```

```
def add_new_books():
    branch = int(input("1)Engineering\n2)Pharmacy\n3)Agriculture"))
    book = input("Enter Name of Book: ")
    quantity = int(input("Enter Quantity of Book: "))
    if branch == 1:
        Engineering[book] = quantity
        print(f"{book} is added in Engineering Branch.\n")
    elif branch == 2:
        Pharmacy[book] = quantity
        print(f"{book} is added in Pharmacy Branch.\n")
    elif branch == 3:
        Agriculture[book] = quantity
        print(f"{book} is added in Agriculture Branch.\n")
    else:
        print("Enter Proper Choice.\n")
```

Student Modules:

```
def student_check(username, password):
    if username in student_info.keys():
        if password == student_info[username]:
            print("You are log in.\n")
            while (True):
                print("1)Books Available\n2)Borrow Book\n3)Return Book\n4)Exit\n")
                choice = int(input("Enter Your Choice: "))
                if choice == 1:
                    books_available()
                elif choice == 2:
                    borrow_book()
                elif choice == 3:
                    return_book()
                elif choice == 4:
                    break
                else:
                    print("Enter Proper Choice.")
    else:
        print("Invalid Credentials.")
        re_student_check()


def re_student_check():
    re_user_name = input("Enter Username: ")
    re_pass_word = int(input("Enter Password: "))
    student_check(re_user_name, re_pass_word)


def books_available():
    branch = int(input("1)Engineering\n2)Pharmacy\n3)Agriculture\n"))
    if branch == 1:
        for k in Engineering:
            print(f"{k}={Engineering[k]}\n")
    elif branch == 2:
        for k in Pharmacy:
            print(f"{k}={Pharmacy[k]}\n")
    elif branch == 3:
        for k in Agriculture:
            print(f"{k}={Agriculture[k]}\n")
    else:
        print("Enter Proper Choice.\n")
```

```
def borrow_book():
    branch = int(input("1)Engineering\n2)Pharmacy\n3)Agriculture\n"))
    book = input("Enter Name Of Book: ")
    if branch == 1:
        if book in Engineering:
            if Engineering[book] > 0:
                print(f"You have issued the {book} Please keep book neat and clean and return within 30 days.\n")
                Engineering[book] = Engineering[book] - 1
                curent_date = datetime.datetime.now()
                with open("history.txt", "a") as f:
                    f.write(f"Username:{user_name}\nBranch:Engineering\nBook:{book}\nIssue Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                    f.write("------------\n")
            else:
                print("Currently book is not available.\n")
        else:
            print("Book is not in library.\n")

    elif branch == 2:
        if book in Pharmacy:
            if Pharmacy[book] > 0:
                print(f"You have issued the {book} Please keep book neat and clean and return within 30 days.\n")
                Pharmacy[book] = Pharmacy[book] - 1
                curent_date = datetime.datetime.now()
                with open("history.txt", "a") as f:
                    f.write(f"Username:{user_name}\nBranch:Pharmacy\nBook:{book}\nIssue Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                    f.write("------------\n")
            else:
                print("Currently book is not available.\n")
        else:
            print("Book is not in library.\n")
    elif branch == 3:
        if book in Agriculture:
            if Agriculture[book] > 0:
                print(f"You have issued the {book} Please keep book neat and clean and return within 30 days.\n")
                Agriculture[book] = Agriculture[book] - 1
                curent_date = datetime.datetime.now()
                with open("history.txt", "a") as f:
                    f.write(f"Username:{user_name}\nBranch:Agriculture\nBook:{book}\nIssue Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                    f.write("------------\n")
            else:
                print("Currently book is not available.\n")
        else:
            print("Currently book is not available.\n")
    else:
        print("Enter Proper Choice.\n")
def return_book():
    branch = int(input("1)Engineering\n2)Pharmacy\n3)Agriculture\n"))
    book = input("Enter Name Of Book: ")
    if branch == 1:
        if book in Engineering:
            Engineering[book] = Engineering[book] + 1
            print(f"You have successfully returned {book}. Have A Great Day!\n")
            curent_date = datetime.datetime.now()
            with open("return_book.txt", "a") as f:
                f.write(f"Username:{user_name}\nBranch:Engineering\nBook:{book}\nReturn Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                f.write("------------\n")
        else:
            print("You are returning book in wrong Branch.\n")
    elif branch == 2:
        if book in Pharmacy:
            Pharmacy[book] = Pharmacy[book] + 1
            print(f"You have successfully returned {book}. Have A Great Day!\n")
            curent_date = datetime.datetime.now()
            with open("return_book.txt", "a") as f:
                f.write(f"Username:{user_name}\nBranch:Pharmacy\nBook:{book}\nIssue Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                f.write("------------\n")
        else:
            print("You are returning book in wrong Branch.\n")
    elif branch == 3:
        if book in Agriculture:
            Agriculture[book] = Agriculture[book] + 1
            print(f"You have successfully returned {book}. Have A Great Day!\n")
            curent_date = datetime.datetime.now()
            with open("return_book.txt", "a") as f:
                f.write(f"Username:{user_name}\nBranch:Agriculture\nBook:{book}\nIssue Date and Time:{curent_date.day}/{curent_date.month}/{curent_date.year} {curent_date.time()}\n")
                f.write("------------\n")
        else:
            print("You are returning book in wrong Branch.\n")
    else:
        print("Enter Proper Choice.\n")
```

Conclusion :

Library management system is a system which will benefit the student as well as the staff. This system has the feature which will provide the student and staff with the feature of admin and student login, In the admin login staff can add new books and admins, in the student login student can login or sign-up after which the list of the books and the stock is displayed then student can borrow or return the book. There is a future scope for this system where we can further modify it by adding features such as Teacher notes, video lectures can be added by teachers as well as online assignments submission facility, a feature of group chat where students can discuss various issues of engineering as well.

GitHub Link:

1. https://github.com/malinarendra/labrary_management_system

References :

1. https://www.w3schools.com/python/
2. https://www.geeksforgeeks.org/python-functions/