



# AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

Dept. of Computer Science  
Faculty of Science and Technology

## CSC2210: OBJECT ORIENTED PROGRAMMING 2

Summer 2023-2024

Section: [F]

Group No: 09

Project Report On

*Library Management System*

Supervised By

Abrar Fahad

Submitted By:

Name			ID		
Avoy, Tasin Islam			22-48818-3		
Sara, Sabrina Sultana			22-48765-3		
Rabbani, Farin Sara			22-48772-3		
Rahman, Sayed Saifur			22-48817-3		
Sadia, Halima Tus			22-48810-3		
Obtained Marks for CO2 and CO3 (Description given in the following page)					
Assessment Criteria	Not Attended/ Incorrect (0)	Inadequate (1-2)	Average (3)	Good (4)	Excellent (5)
Evaluation Criteria (CO2)		Total =	Evaluation Criteria (CO3)		Total =
Requirement fulfillment			Organization of the application		
Validation			Representation and Integration of Database		
Verification			Graphical User Interface		

**CO2:** Display and verify the mean of a real-life Project using the concepts of C# Graphical User Interface based environment with database integration to depict a desktop-based application.

Assessment Criteria	Not Attended/ Incorrect (0)	Inadequate (1-2)	Average (3)	Good (4)	Excellent (5)
Evaluation Criteria	Evaluation Definition				
Requirement fulfillment	Fails to demonstrate any understanding of real-life scenario-based project development or functional requirement identification. There is no attempt to depict a project or identify functional requirements accurately.	Demonstrates limited understanding of real-life scenario-based project development and functional requirement identification. The project depicted lacks coherence or relevance to real-life scenarios, and functional requirements are inaccurately identified or insufficiently described.	Presents a basic depiction of a real-life scenario-based project and identifies some functional requirements. However, the project lacks depth or complexity, and some functional requirements may be vaguely defined or missing key details.	Effectively demonstrates a realistic scenario-based project and accurately identifies most functional requirements. The project is well-developed with appropriate complexity, and functional requirements are clearly articulated with relevant details.	Exhibits an exceptional understanding of real-life scenario-based project development and accurately identifies all functional requirements. The project is meticulously developed with thorough attention to detail, reflecting a comprehensive understanding of Object-Oriented Programming project development activities.
Validation	Fails to demonstrate any understanding or implementation of validation forms in their system. There is no attempt to deal with data validation, and validation requirements are completely ignored or incorrectly applied.	Demonstrates limited understanding of validation forms and data validation techniques. While some attempt may be made to implement validation, it is incomplete or poorly executed, leading to inadequate handling of data validation.	Shows a basic understanding of validation forms and data validation techniques. They attempt to implement validation, but some aspects may be missing or incorrectly implemented, resulting in partial or inconsistent handling of data validation.	Effectively demonstrates the use of validation forms and implements data validation techniques. Validation is mostly accurate and comprehensive, ensuring the proper handling of data input and verification in the system.	Exhibits an exceptional understanding and implementation of validation forms and data validation techniques. Validation is meticulously implemented with thorough attention to detail, ensuring robust data validation procedures and contributing to the overall reliability and integrity of the system.
Verification	Fails to demonstrate any attempt to verify the system data or functional requirements. There is no evidence of	Demonstrates limited understanding of verification processes and data flow in the system. Verification	Shows a basic understanding of verification processes and attempts to verify system data. However, verification	Identifies and verifies system data, ensuring proper functional requirements are met. Verification efforts are mostly accurate and	Exhibits an exceptional understanding of verification processes and meticulously verifies system data. Verification

	understanding or implementation of verification processes, and data flow is not considered.	attempts are incomplete or inaccurate, and there is insufficient consideration given to ensuring data integrity and functionality.	efforts may be inconsistent or lack thoroughness, and there may be gaps in ensuring proper functional requirements and data flow.	thorough, with attention to ensuring data integrity and appropriate data flow within the system.	efforts are comprehensive and precise, with a keen focus on ensuring all functional requirements are met and maintaining proper data flow throughout the system.
--	---------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------

**CO3:** Prepare and Explain a real life desktop based application synthesizing several component of C# along with development tools to adhere the given requirements.

Assessment Criteria	Not Attended/ Incorrect (0)	Inadequate (1-2)	Average (3)	Good (4)	Excellent (5)
Evaluation Criteria	Evaluation Definition				
Organization of the application	Fails to identify any suitable real time application or requirements for project development activities related to OOP.	Limited understanding about the project scopes and scenarios or identification of functional requirements.	Lacks depth or relevance to OOP project development activities and may contain inaccuracies. Real-life scenarios are mentioned, but the discussion lacks depth or clarity.	Consider and integrate the idea of several core aspects of the project along with relevance to real-life scenarios. Demonstrating a solid understanding of the application presentation.	Generalize and exhibits an exceptional understanding of project preparation according to a to real-life scenarios. Also contains proper and insightful identification of the system which is comprehensive and precise.
Representation and Integration of Database	Fails to identify and present any understanding or implementation of database. Also failed to integrate the data with the project itself.	Limited understanding of the database concepts or their proper way of using in a real time project. While some attempt may be made to implement but it is incomplete or poorly executed, leading to inadequate design.	Lacks depth or relevance to database integration with the application. Shows a basic understanding but some aspects may be missing or incorrectly implemented, resulting in partial or inconsistency. May lack proper normalization.	Integrate the database with the forms properly and implements it with proper validation which is mostly accurate and comprehensive, ensuring the proper handling of data input and verification along with general normalization.	Exhibits an exceptional understanding and implementation of database ensuring attention to detail, and robust data manipulation procedures and contributing to the overall clarity.
Graphical User Interface	Fails to present or prepare GUI based application interfaces. There is no evidence of creating or integrating such things according to their usefulness.	Limited understanding of graphical user interfaces. Lack of design knowledge. Very poor attempt to make such things which are currently obsolete or can't be identified as coherent.	Shows a basic understanding of creating user interfaces. Most of them are interconnected but maybe some of them lack it. However, most of it can be described as user friendly.	Effectively identifies and meet the consider the simplicity. Design related works are mostly accurate and taken proper attention to ensuring a user-friendly coherent system.	Exhibits an exceptional work design following a high standard of simple and elegant work. Several controls and mechanism has been organized in a preferred way according to the coherent usage .

<b>Table of content</b>	<b><u>Pages</u></b>
I.Introduction-----	5
II. User Story: -----	5-7
III. Er-diagram-----	7
IV. SQL Query-----	7-8
V. Screenshot-----	8-17
VI. GitHub Link-----	18
VII. Contribution Table-----	18

- **Introduction:**

The Library Management System is a desktop-based application developed to enhance library operations using the C# programming language. It has three user roles, namely Admin, Librarian and Student. Upon entering the system through a secure login page, the users get access to their comprehensive Dashboards. The dashboard consists of features which are accessible by dedicated users. These features help the users to interact with each other in the Library Management System.

- **User story:**

The functional descriptions of the three users used in the project are provided below:

- **Admin:**

1. Upon entering the system through a secure login page as Admin, the user gets access to the comprehensive Admin Dashboard.

2. The dashboard consists of four primary features: Books, Librarian, Student, and Log out.

- In the Books section, the admin can:
  - Add new books
  - Quickly view the library's book collection
- The Librarian segment allows:
  - The appointing of Librarians
  - Viewing of Librarian information
- The Student category allows various functions, such as:
  - Addition of student information
  - Viewing student information
  - Issue of books
  - Return of books
  - Details of issue and return information
- Lastly, the Logout feature ensures:
  - A secure logout, returning users to the login page.

- **Librarian:**

1. When a Librarian logs into the system, they are presented with a Librarian dashboard that includes the following features: Books, Student, Accounts, and Logout.

2. The “Books” section allows the Librarian to:

- View and manage the library's collection
- Add new books

3. In the “Student” section, the Librarian can:

- Manage student information
- Issue and return books
- Oversee all student-related book transactions
- Assign and manage fines

3. The “Accounts” feature enables the Librarian to:

- Manage purchases for new books

4. The “Logout” option ensures:

- A secure exit from the system, returning them to the login page.

- **Student:**

When a Student logs in, they access a simplified Student dashboard with options for View Books, Borrow History, and Logout.

1. The View Books section lets the student:

- Browse the library's collection
- No ability to update or remove books

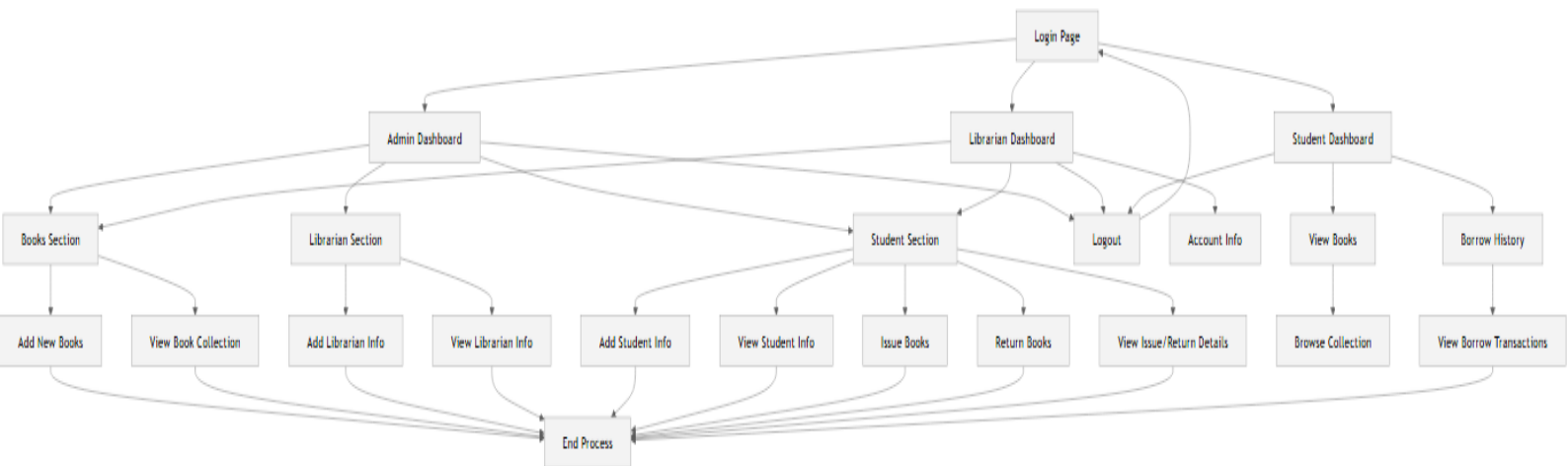
2. In Borrow History, the student can:

- See details of books they have borrowed
- View current and past transactions
- View assigned fines

3. The Logout feature allows the student to:

- Securely log out of the system
- Return to the login page.

## ER-Diagram



### • SQL Query:

```
create database Library_Management_System;
```

```
use Library_Management_System;
```

```
CREATE TABLE Users (  
    Id INT PRIMARY KEY IDENTITY(1,1),  
    UserName VARCHAR(100) NOT NULL,  
    Password VARCHAR(255) NOT NULL,  
    Enroll NVARCHAR(100) NOT NULL,  
    Contact NVARCHAR(15),  
    Email NVARCHAR(100) NOT NULL,  
    Address NVARCHAR(255),  
    DOB DATE NOT NULL,  
    Gender VARCHAR(100) NOT NULL,  
    UserType INT NOT NULL  
);
```

```
CREATE TABLE Librarian (  
    LibrarianSL INT PRIMARY KEY IDENTITY(1,1),  
    Id INT NOT NULL,  
    Salary DECIMAL(10, 2) NOT NULL,  
    FOREIGN KEY (Id) REFERENCES Users(Id)
```

```
);
```

```
CREATE TABLE [dbo].[Book](
    [BookID] [int] IDENTITY(1,1) NOT NULL,
    [Name] [varchar](50) NOT NULL,
    [Author] [varchar](50) NOT NULL,
    [Publication] [varchar](50) NOT NULL,
    [ISBN] [varchar](50) NOT NULL,
    [Quantity] [int] NOT NULL,
    [Price] [int] NULL,
    CONSTRAINT [PK_Book] PRIMARY KEY CLUSTERED
(
    [BookID] ASC
));
```

```
CREATE TABLE [dbo].[IRBook](
    [BID] [int] IDENTITY(1,1) NOT NULL,
    [Stu_enroll] [varchar](100) NOT NULL,
    [Stu_name] [varchar](100) NOT NULL,
    [PhoneNo] [NVARCHAR] (100) NOT NULL,
    [Email] [varchar](100) NOT NULL,
    [Address] [varchar](100) NOT NULL,
    [Book_name] [varchar](100) NOT NULL,
    [Book_issue_date] [varchar](100) NOT NULL,
    [Book_return_date] [varchar](100) NULL,
    [Fine] [int] default(0),

    PRIMARY KEY CLUSTERED
(
    [BID] ASC
));
```

```
CREATE TABLE [dbo].[Accounts](
    [CustomerID] [int] IDENTITY(1,1) NOT NULL,
    [Stu_Name] [varchar](50) NOT NULL,
    [PhoneNo] [NVARCHAR](50) NOT NULL,
    [email] [varchar](100) NOT NULL,
    [Address] [varchar](100) NOT NULL,
    [Book] [varchar] (1000) NOT NULL,
    [Price] [int] default(0),

    CONSTRAINT [PK_Accounts] PRIMARY KEY CLUSTERED
(
    [CustomerID] ASC
));
```

```
SELECT U.*, L.Salary FROM Users U INNER JOIN Librarian L ON U.Id = L.Id WHERE
U.UserType = 2;
```

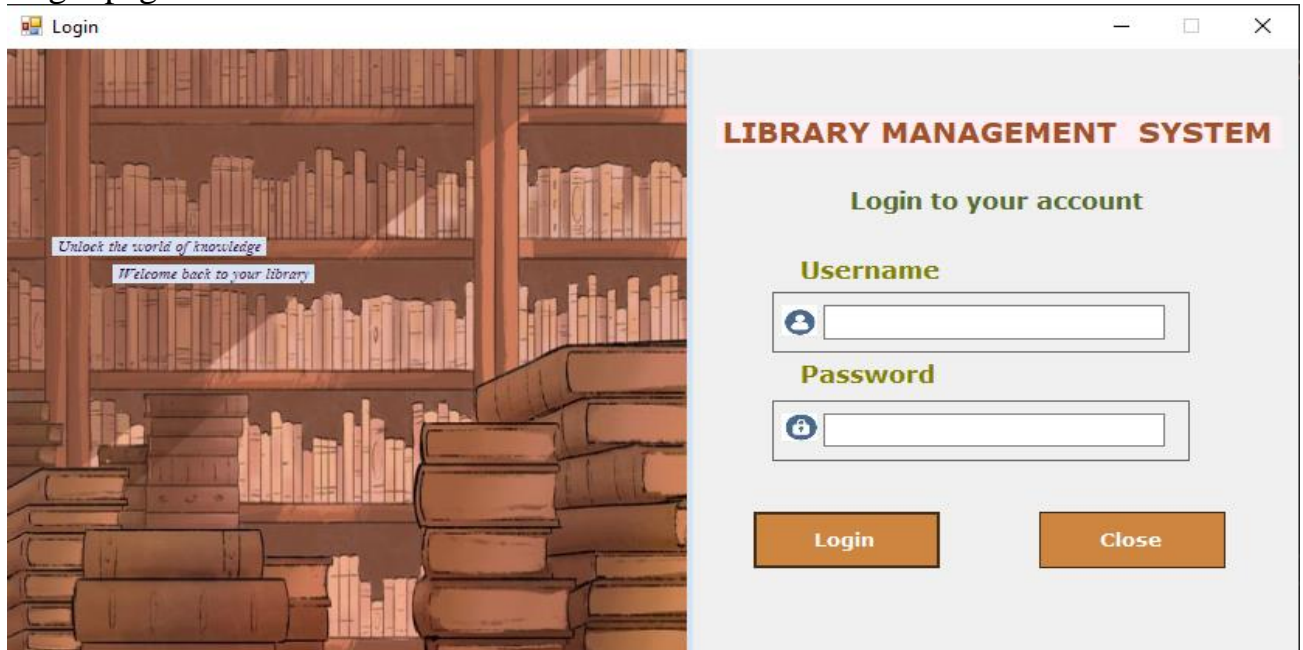
```
SELECT U.*, L.Salary FROM Users U INNER JOIN Librarian L ON U.Id = L.Id WHERE
U.UserType = 2 AND U.Id=5;
```

```
SELECT U.*, L.Salary FROM Users U INNER JOIN Librarian L ON U.Id = L.Id WHERE
U.UserType = 2 AND UserName LIKE ' ' + txtSearchViewLibrarian.Text + '%';
```



- Screenshot:

Login page:



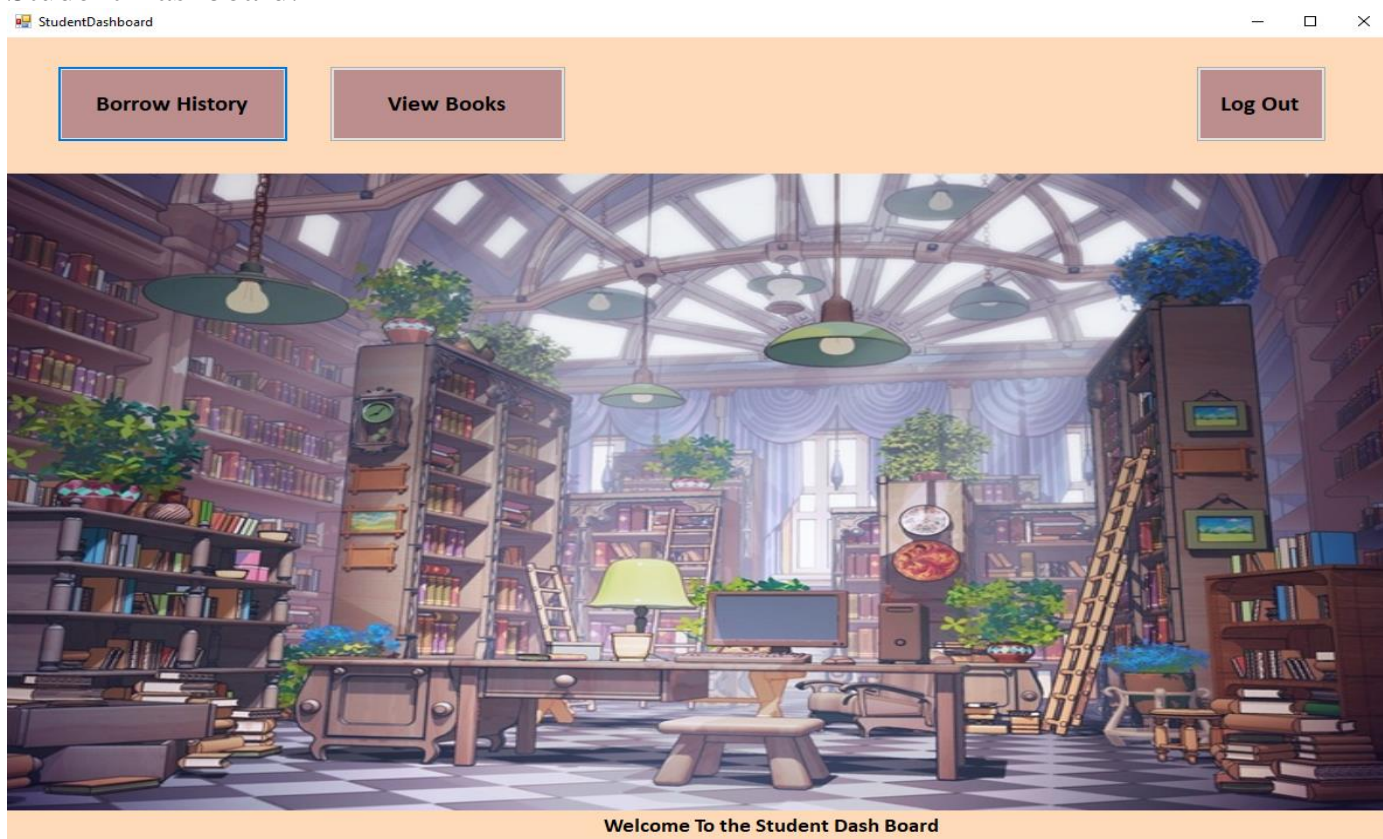
AdminDashboard:



## Librarian Dashboard:



## Student Dashboard:





## Accounts:

 Accounts



# Accounts



Enter Enrollment Number

**Search Student**

**Refresh** **Exit**

Student Name

Phone Number

Email


Address


Books Name

Price


**Buy** **Print**

## Add Books:

 AddBooks



# Add Books



Book Name

Book Author

Book Publication

ISBN No.

Book Price


Book Quantity

**Save** **Cancel**

## View Book Info:

ViewBookInfo

Welcome..  
N.B. If you want to do Update and  
Remove operation..  
This is the place for you..



please select any cell which rows informations you  
want to update and remove and then perform those  
operation if you want....

Book Name

Refresh

	BookID	Name	Author	Publication	ISBN	Quantity	Price
	2	Introduction to AI...	Thomas H. Comen	MIT Press	100	20	2000
	3	Computer Networ...	Andrew S. Tanen...	Pearson	101	25	1000
	4	perating System ...	Abraham Silbersc...	Wiley	102	30	2500
	5	Database System...	Abraham Silbersc...	McGraw-Hill	103	40	1500
	6	Clean Code	Robert C. Martin	Prentice Hall	104	20	3000
▶	9	To Kill a Mocking...	Harper Lee	J.B. Lippincott	105	30	1000

Book Name

To Kill a Mockingbird

Book Author

Harper Lee

Book Publication

J.B. Lippincott

ISBN No.

105

Book Price

1000

Book Quantity

30


Update

Remove

Cancel

## Add Librarian:

AddLibrarian



Add Librarian

User Name

Password

Enroll

Email

Address

Phone Number

Date of Birth

Saturday , October 5, 2024

Gender

Please Select

Salary

Save


Print

Clear

Cancel

## View Librarian Info:

Welcome..  
N.B. If you want to do Update and Remove operation..  
This is the place for you..



please select any cell which rows informations you want to update and remove and then perform those operation if you want....

User ID

**Refresh**

	Id	UserName	Password	Enroll	Contact	Email	Address	DOB	Gender	UserType	Salary
▶	2	user1	1234	li-001	01598389	user1@g...	Badda	11/1/2002	Female	2	10000.00
	12	user2	1212	li-002	45645654	user2@g...	Mirpur	10/4/2001	Male	2	10000.00
	13	Sara	1212	li-003	0498854	waw@gm...	Dhaka	11/7/2004	Female	2	30000.00
	14	user4	2121	li-004	65656565	user4@g...	Mirpur	10/4/2001	Female	2	20000.00

Password

Email

Phone Number

Date of Birth

Salary

Address

Enroll

Gender

**Update**

**Remove**

**Cancel**

## Add Student Info:



### Add Student Info



Student Name

Password

Enroll No

Phone Number

Date of Birth

Email



Address

Gender

**Save** **Print** **Clear**

**Cancel**

## View Student Info:



Welcome..  
N.B. If you want to do Update and Remove operation..  
This is the place for you..

please select any cell which rows informations you want to update and remove and then perform those operation if you want....

**Name**

**Refresh**

	Id	UserName	Password	Enroll	Contact	Email	Address
▶	6	Avoy	1234	en-001	954869	avoy@gmail.com	Mirpur
	8	Farin	5678	en-002	98547896	farin@gmail.com	Badda
	15	Tasin	1234	en-003	894758	tasin@gmail.com	Mirpur
*							

**Student Name**

**Email**

**Enroll No**

**Address**

**Phone Number**

**Date of Birth**

**Password**


**Gender**

**Update**


**Remove**

**Cancel**

## Issue Books:



### Issue Books



**Enter Enrollment Number**

**Search Student**

**Refresh**

**Exit**

**Student Name**

**Phone Number**

**Email**

**Address**


**Books Name**


**Issue Date**

**Issue Book**

[\*\*Maximum 3 books can be issued to a Student\*\*]

## Return Book:






Enter Enrollment Number

	BID	Stu_enroll	Stu_name	PhoneNo	Email	Address	Book_name	Book_issue_date	Book_return_date	Fine
▶	7	en-001	Avoy	456456	avoy@gmail.com	Dhaka	Introduction to AI...	Tuesday, Octobe...		0

## Issue Return Book Info:

Issue Books Info										
	BID	Stu_enroll	Stu_Name	PhoneNo	Email	Address	Book_Name	Book_issue_date	Book_return_date	
▶	1	en-001	Avoy	456456	avoy@gmail.com	Dhaka	Introduction to AI...	Saturday, Septe...	Saturday, Septe...	
	2	en-006	Tasin	2345564	tasin@gmail.com	Turag	To Kill a Mocking...	Monday, Septem...	Monday, Septem...	
	3	en-005	Sara	857875	sara@gmail.com	Barishall	To Kill a Mocking...	Tuesday, Octobe...	Thursday, Octob...	
	4	en-005	Sara	857875	sara@gmail.com	Barishall	To Kill a Mocking...	Tuesday, Octobe...	Thursday, Octob...	
	5	en-004	Farin	8989889	Farin@gmail.com	Cumilla	Database System...	Tuesday, Octobe...	Thursday, Octob...	
	6	en-002	Tus	456785	tus@gmail.com	Rajshahi	Clean Code	Tuesday, Octobe...	Thursday, Octob...	
Return Books Info										
	BID	Stu_enroll	Stu_name	PhoneNo	Email	Address	Book_name	Book_issue_date	Book_return_date	Fine
▶	1	en-001	Avoy	456456	avoy@gmail.com	Dhaka	Introduction to AI...	Saturday, Septe...	Saturday, Septe...	0
	2	en-006	Tasin	2345564	tasin@gmail.com	Turag	To Kill a Mocking...	Monday, Septem...	Monday, Septem...	0
	3	en-005	Sara	857875	sara@gmail.com	Barishall	To Kill a Mocking...	Tuesday, Octobe...	Thursday, Octob...	40
	4	en-005	Sara	857875	sara@gmail.com	Barishall	To Kill a Mocking...	Tuesday, Octobe...	Thursday, Octob...	40
	5	en-004	Farin	8989889	Farin@gmail.com	Cumilla	Database System...	Tuesday, Octobe...	Thursday, Octob...	40
	6	en-002	Tus	456785	tus@gmail.com	Rajshahi	Clean Code	Tuesday, Octobe...	Thursday, Octob...	40

## Borrow History:

 BorrowHistory

**Borrow History**

**Name:**

Sara


**Enroll:**








en-005

	Book_Name	Book_issue_date	Book_return_date	Fine
▶	To Kill a Mockingbird	Tuesday, October 1, 2024	Thursday, October 10, 2024	40
	To Kill a Mockingbird	Tuesday, October 1, 2024	Thursday, October 10, 2024	40
*				

Exit

## Print in Librarian:

 Print preview



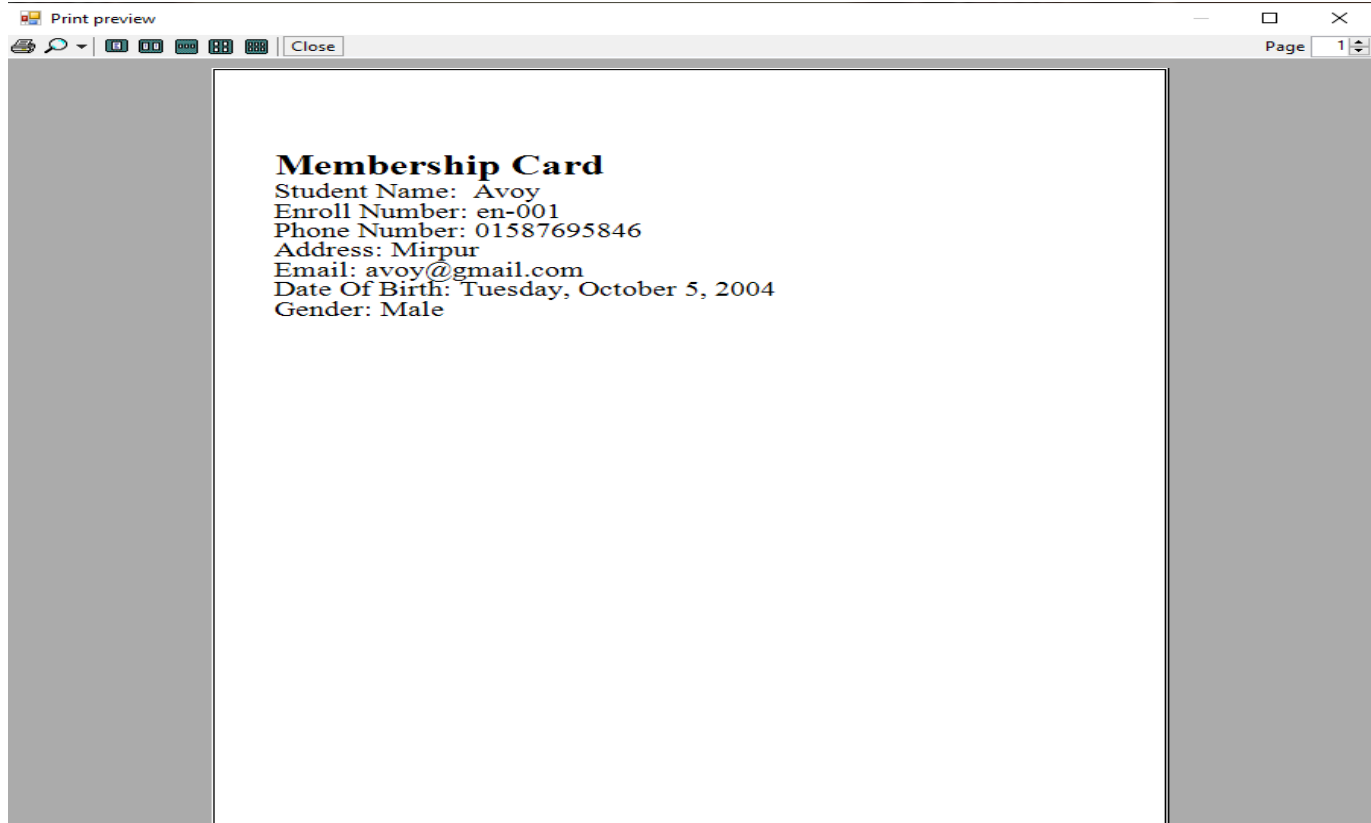
Close

Page 1

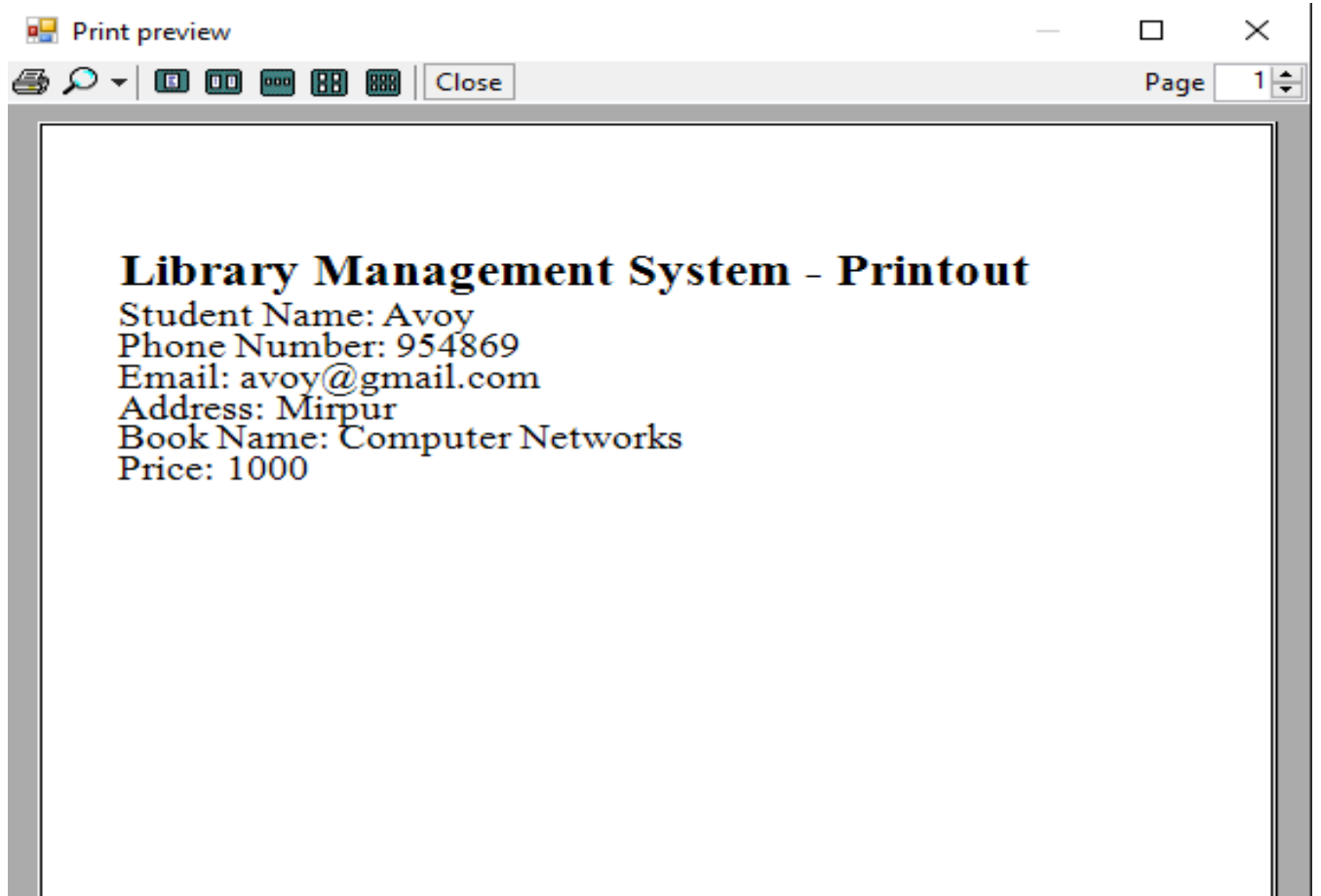
**Congratilation!!!!!! You have been apointed**  
Librarian Name: Sara  
Phone Number: 01398467584  
Email: sara@gmail.com  
Dath Of Birth: Tuesday, October 5, 2004  
Gender: Male  
Address: Mirpur  
Salary: 20000



### Print in Student:



### Print in Accounts:



**Contribution Table:**

<b>Name</b>	<b>ID</b>	<b>Contribution</b>
<b>Avoy, Tasin Islam</b>	<b>22-48818-3</b>	<b>Login, AddStudentInfo, Accounts</b>
<b>Sara, Sabrina Sultana</b>	<b>22-48765-3</b>	<b>AddLibrarian, ViewLibrarianInfo, BorrowHistory</b>
<b>Rabbani, Farin Sara</b>	<b>22-48772-3</b>	<b>IssueBooks, ViewStudentInfo, AdminDashboard</b>
<b>Rahman, Sayed Saifur</b>	<b>22-48817-3</b>	<b>ViewBooksInfo, IssueReturnBookInfo, StudentDashboard</b>
<b>Sadia, Halima Tus</b>	<b>22-48810-3</b>	<b>AddBooks, ReturnBook, LibrarianDashboard,</b>

**GitHub Link:**

**[https://github.com/Avoy55548/Library\\_Management\\_System.git](https://github.com/Avoy55548/Library_Management_System.git)**