

The Relational Model

Name: Arjun Bista

Student ID: 918462199

Course: Database System (CSC675)

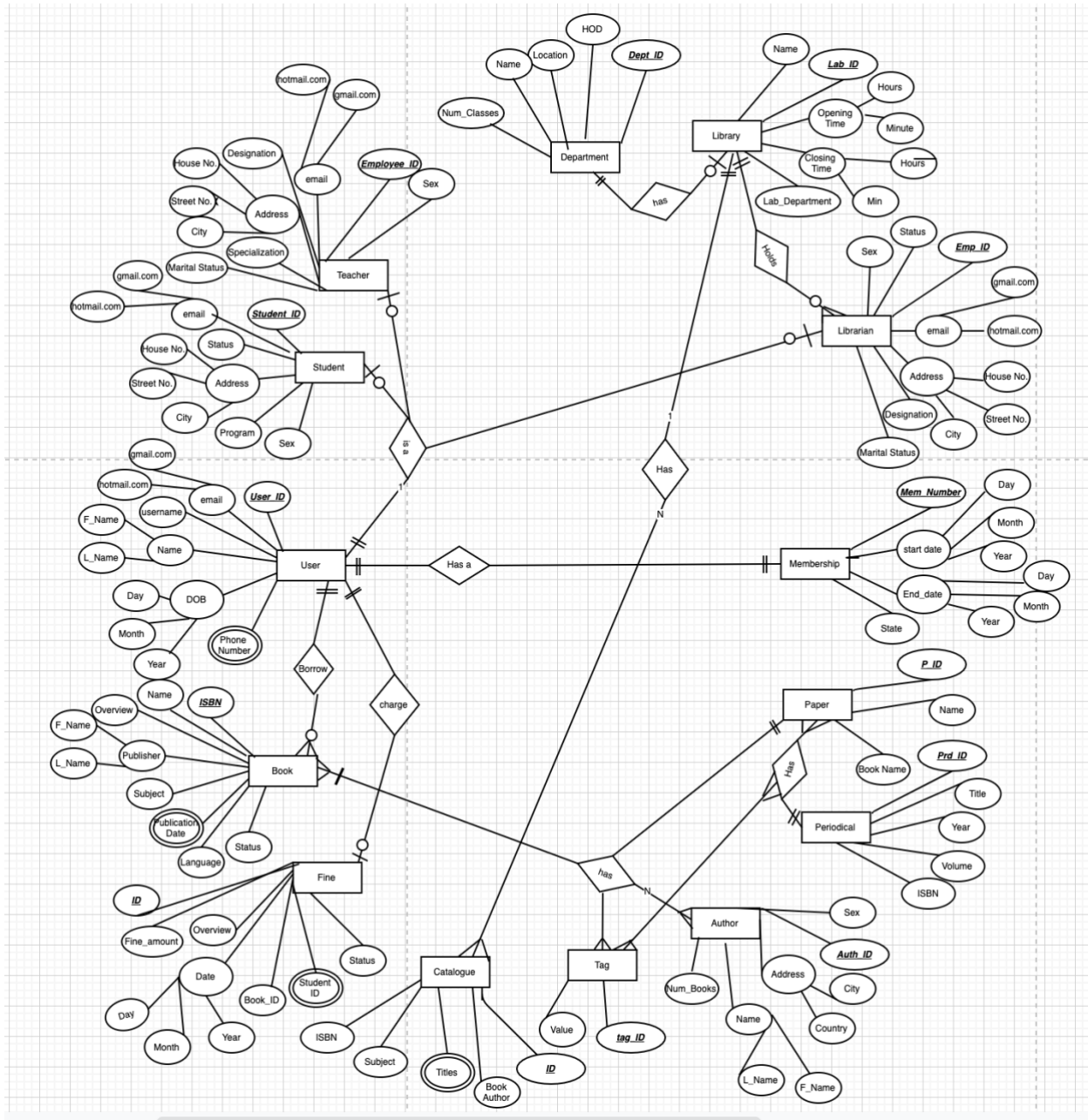
Professor: Jose Ortiz Costa

Project: HW2

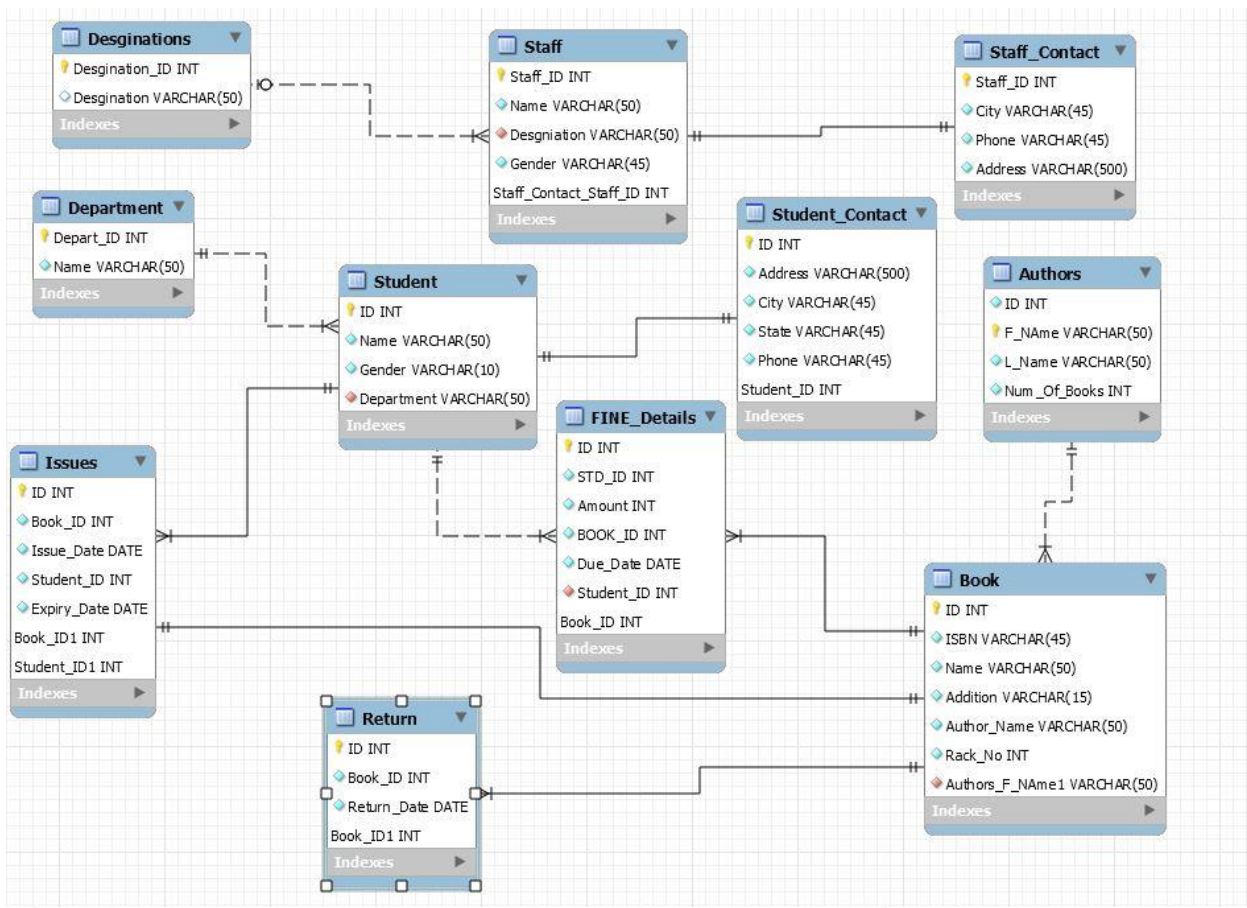
Table of Content

	Page No.
1. Section I (Final ERD Version)	3
2. Section II (Database Model)	4
3. Section III (Forward Engineering)	4-6
4. Section IV (Inserting Data)	6-7
5. Section V (Testing)	8
6. Section VI (Testing Table)	9-10

Section I: Final ERD Version



Section II: Database Model



Section III: Forward Engineering

```
CREATE DATABASE Library_System
```

```
CREATE TABLE Desgnations(
Desg_ID INT IDENTITY(1,1) PRIMARY KEY,
Desgnation VARCHAR(50) not null UNIQUE
);
```

```
CREATE TABLE Staff(
Staff_ID INT IDENTITY(1,1) PRIMARY KEY,
Name VARCHAR(50) not null,
Desgnation VARCHAR(50) not null ,
Gender VARCHAR(10) not null CHECK(Gender='male' or Gender='Female'),
FOREIGN KEY (Desgnation) REFERENCES Desgnations(Desgnation) ON UPDATE CASCADE
);
```

```

CREATE TABLE Staff_Contact(
Staff_ID INT PRIMARY KEY REFERENCES Staff(Staff_ID),
Address VARCHAR(500) not null,
City VARCHAR(50) not null,
Phone VARCHAR(20) not null,
CONSTRAINT ID_STF UNIQUE (Staff_ID)
);

```

```

CREATE TABLE Department(
Dpt_ID INT IDENTITY(1,1) PRIMARY KEY,
Dpt_Name VARCHAR(50) not null UNIQUE
);

```

```

CREATE TABLE Student(
Std_ID INT IDENTITY(1,1) PRIMARY KEY,
Name VARCHAR(50) not null,
Gender VARCHAR(10) not null check(Gender='male' or Gender='Female'),
Student_Dpt int not null,
FOREIGN KEY (Student_Dpt) REFERENCES Department(Dpt_ID)
);

```

```

CREATE TABLE Student_Contact(
Std_ID INT PRIMARY KEY REFERENCES Student(Std_ID),
[Address] VARCHAR(500) ,
City VARCHAR(50) not null,
[State] VARCHAR(50) not null,
Phone VARCHAR(20) not null,
CONSTRAINT Student_ID UNIQUE (Std_ID)
);

```

```

CREATE TABLE Authors(
Author_ID INT IDENTITY(1,1) PRIMARY KEY,
F_Name VARCHAR(100) not null UNIQUE,
L_Name VARCHAR(50) not null,
Num_Of_books INT
);

```

```

CREATE TABLE Book(
Book_ID INT IDENTITY(1,1) PRIMARY KEY,
ISBN VARCHAR(50) not null,
Book_Name VARCHAR(100) not null unique,
Edition VARCHAR(5) not null,
Author_Name VARCHAR(100) not null,
Rack_No INT not null,
FOREIGN KEY (Author_Name) REFERENCES Authors(F_Name)
);

```

```

CREATE TABLE Issues(
Issue_ID INT IDENTITY(1,1) PRIMARY KEY,
Book_Name VARCHAR(100) not null,
Book_ID INT not null REFERENCES Book(Book_ID),
Std_ID INT not null REFERENCES Student(Std_ID),
issue_Date Date not null ,
Expiry_Date Date not null,
FOREIGN KEY (Book_Name) REFERENCES Book(Book_Name),
CONSTRAINT B_ID UNIQUE (Book_ID),
CONSTRAINT S_ID UNIQUE (Std_ID)
);

```

```

CREATE TABLE FINE_Details(
F_ID INT IDENTITY(1,1) PRIMARY KEY,
Std_ID INT not null REFERENCES Student(Std_ID) ,
Amount INT not null,
Due_Date Date not null,
Book_ID INT not null REFERENCES Book(Book_ID),
CONSTRAINT FStd_ID UNIQUE (Std_ID) ,
CONSTRAINT FB_ID UNIQUE (Book_ID)
);

```

Section IV: Inserting Data

```

INSERT INTO Designations VALUES( 'Adimin'); /* Auto implemented Desg_ID */
INSERT INTO Designations VALUES( 'Librarian'); /* Auto implemented Desg_ID */
INSERT INTO Designations VALUES( 'Clerk'); /* Auto implemented Desg_ID */
SELECT * FROM Designations;

```

```

INSERT INTO Staff VALUES( 'Ashraf', 'Adimin', 'Male'); /* Auto implemented Staff_ID */
INSERT INTO Staff VALUES( 'Ehsan', 'Librarian', 'Male'); /* Auto implemented Staff_ID */
INSERT INTO Staff VALUES( 'Urva', 'Clerk', 'Female'); /* Auto implemented Staff_ID */
SELECT * FROM Staff;

```

```

INSERT INTO Staff_Contact VALUES( 1, 'hjhs jsHVX JHVK', 'Checago', '+283498230');
INSERT INTO Staff_Contact VALUES( 2, 'hjhs jsHVX JHVK', 'Checago', '+576889223');
INSERT INTO Staff_Contact VALUES( 3, 'nbmkuf hgcvukjv ', 'London', '+3456787334');
SELECT * FROM Staff_Contact;

```

```

INSERT INTO Department VALUES( 'Physics'); /* Auto implemented Dpt_ID */
INSERT INTO Department VALUES( 'Chemistry'); /* Auto implemented Dpt_ID */
INSERT INTO Department VALUES( 'Computer Science'); /* Auto implemented Dpt_ID */
SELECT * FROM Department

```

```

INSERT INTO Student VALUES( 'Albart ', 'Male', 1); /* Auto implemented Std_ID */
INSERT INTO Student VALUES( 'Ainstine ', 'Male', 2); /* Auto implemented Std_ID */
INSERT INTO Student VALUES( 'Ehsan ', 'Male', 3); /* Auto implemented Std_ID */
SELECT * FROM Student

```

```

INSERT INTO Student_Contact VALUES( 5, 'xfgahg ujcsVZ ', 'London', 'United state',
'+7687968709');
INSERT INTO Student_Contact VALUES( 6, 'qwujbfdkj jka ', 'Checago', 'United Kingdom',
'+235678992');
INSERT INTO Student_Contact VALUES( 7, 'xfgahg ujcsVZ ', 'Lahore', 'United state',
'+0972737121');
SELECT * FROM Student_Contact

INSERT INTO Authors VALUES( 'Albart', 'Ainstine', 5); /* Auto implemented Author_ID */
INSERT INTO Authors VALUES( 'Ehsan', 'Tariq', 1); /* Auto implemented Author_ID */
INSERT INTO Authors VALUES( 'Ahmad', 'Faza', 3); /* Auto implemented Author_ID */
SELECT * FROM Authors

INSERT INTO Book VALUES( 'ISB12', 'Book1', 3, 'Albart', 1); /* Auto implemented Book_ID
*/
INSERT INTO Book VALUES( 'ISB32', 'Book2', 6, 'Ehsan', 4); /* Auto implemented Book_ID
*/
INSERT INTO Book VALUES( 'ISB42', 'Book3', 4, 'Ahmad', 2); /* Auto implemented Book_ID
*/
SELECT * FROM Book

INSERT INTO Issues VALUES( 'Book1', 1,1, '01-06-2019', '01-11-2019');/* Auto implemented
Issue_ID */
INSERT INTO Issues VALUES( 'Book2', 2,2, '01-01-2020', '05-03-2020');/* Auto implemented
Issue_ID */
INSERT INTO Issues VALUES( 'Book3', 3,3, '04-04-2020', '10-11-2020'); /* Auto implemented
Issue_ID */
SELECT * FROM Issues

CREATE TABLE [Return](
Return_ID INT IDENTITY(1,1) PRIMARY KEY,
Book_ID INT not null REFERENCES Book(Book_ID),
Return_Date Date not null,
CONSTRAINT BK_ID UNIQUE (Book_ID)
)
INSERT INTO [Return] VALUES( 1, '01-11-2019'); /* Auto implemented Return_ID */
INSERT INTO [Return] VALUES( 2, '05-03-2020'); /* Auto implemented Return_ID */
INSERT INTO [Return] VALUES( 3, '10-11-2020'); /* Auto implemented Return_ID */
SELECT * FROM [Return]

INSERT INTO FINE_Details VALUES( 5, 300, '04-04-2020', 1); /* Auto implemented F_ID */
INSERT INTO FINE_Details VALUES( 6, 200, '12-04-2020', 2); /* Auto implemented F_ID */
INSERT INTO FINE_Details VALUES( 7, 300, '01-05-2020', 3); /* Auto implemented F_ID */
SELECT * FROM FINE_Details

```

Section V: Testing

```
DELETE FROM Designations WHERE Desgination='Admin';
UPDATE Designations SET Desgination='Checker' WHERE Desgination= 'Clerk';
SELECT Desgination FROM Designations;

DELETE FROM Staff WHERE Desgination='Admin';
UPDATE Staff SET Desgination='Checker' WHERE Desgination= 'Clerk';
SELECT * FROM Staff;

DELETE FROM Staff_Contact WHERE City='London';
UPDATE Staff_Contact SET City='Lahore' WHERE City= 'Checago';
SELECT * FROM Staff_Contact where City = 'Checago';

DELETE FROM Department WHERE Dpt_Name='Physics'; /* Cannot delete due to reference*/
UPDATE Department SET Dpt_Name='Chemistry' WHERE Dpt_Name= 'Physicology';
SELECT * FROM Department where Dpt_Name= 'Computer Science';

DELETE FROM Student; /* Cannot delete due to reference*/
UPDATE Student SET Name='Ehsan' WHERE Name= 'Usman';
SELECT Name,Student_Dpt FROM Student;

DELETE FROM Student_Contact WHERE City='Lahore';
UPDATE Student_Contact SET City='London' WHERE Name= 'Karachi';
SELECT City FROM Student_Contact where City='Karachi';

DELETE FROM Authors WHERE F_Name='Albart'; /* Cannot delete due to reference*/
UPDATE Authors SET F_Name='Usman' WHERE Name= 'Ehsan';
SELECT F_Name, L_Name FROM Authors;

DELETE FROM Book WHERE Book_Name='Book1'; /* Cannot delete due to reference*/
UPDATE Book SET Book_Name='Book2' WHERE Name= 'Book4';
SELECT * FROM Book where Book_Name='Book3';

DELETE FROM Issues WHERE Book_Name='Book1';
UPDATE Issues SET Book_Name='Book2' WHERE Name= 'Book4';
SELECT * FROM Issues;

DELETE FROM [Return] WHERE Book_ID=2;
UPDATE [Return] SET Return_Date='10-11-2020' WHERE Return_Date= '23-12-2020';
SELECT * FROM [Return];

DELETE FROM FINE_Details WHERE Std_ID=6;
UPDATE FINE_Details SET Due_Date='01-05-2020' WHERE Due_Date= '23-12-2020';
SELECT * FROM FINE_Details where Due_Date < '01-05-2020';
```


Section VI: Testing Table

Test	Statement	Entity	Pass/Fails	Error Description
1	Delete	Designations	Fails	Designation cannot be deleted because of reference key issue.
2	Update	Designations	Pass	None
3	Delete	Staff	Pass	None
4	Update	Staff	Pass	None
5	Delete	Staff_Contact	Pass	None
6	Update	Staff_Contact	Pass	None
7	Delete	Department	Fails	Department Name cannot be deleted because of reference key issue.
8	Update	Department	Pass	None
9	Delete	Student	Fails	Student Record cannot be deleted because of reference key issue.
10	Update	Student	Pass	None
11	Delete	Student_Contact	Pass	None
12	Update	Student_Contact	Pass	None
13	Delete	Authors	Fails	Author's First Name cannot be deleted because of reference key issue.

Test	Statement	Entity	Pass/Fails	Error Description
14	Update	Authors	Pass	None
15	Delete	Book	Fails	Book Name cannot be deleted because of reference key issue.
16	Update	Book	Pass	None
17	Delete	Issues	Pass	None
18	Update	Issues	Pass	None
	Delete	Return	Pass	None
	Update	Return	Pass	None
	Delete	FINE_Details	Pass	None
	Update	FINE_Details	Pass	None