CSE 5330 – Database Systems – 005

Project 2 – Part 2

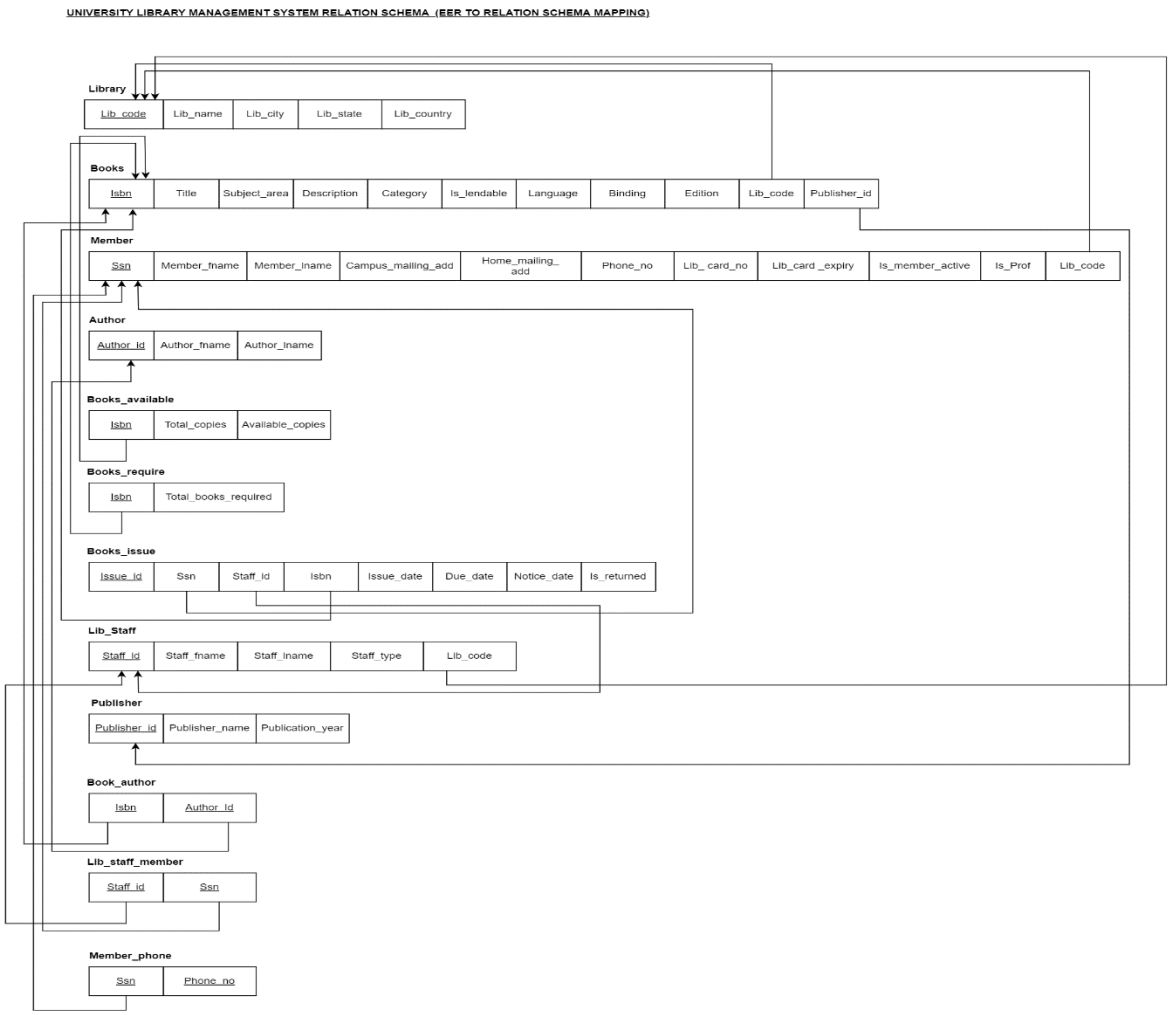
Team–19: Preethi Subramanian – 1002059233
Shruti Devchandbhai Devani –

Overview:

This file contains information about Database System Project 2, Part 2 which was assigned to design and implement a database for keeping track of members, books, the catalog, and the borrowing activity of a university library.

Part – 2: Mapping of EER Schema Diagram for the Library database to Relation Schema:

a) Mapping of Relation Schema from EER Diagram generated in Project 2 Part 1 phase:



b) Reason of Choices made while Mapping from EER Diagram to Relation Schema

ER to Relational Mapping is done based on the following steps and the reasons of choices are defined below:

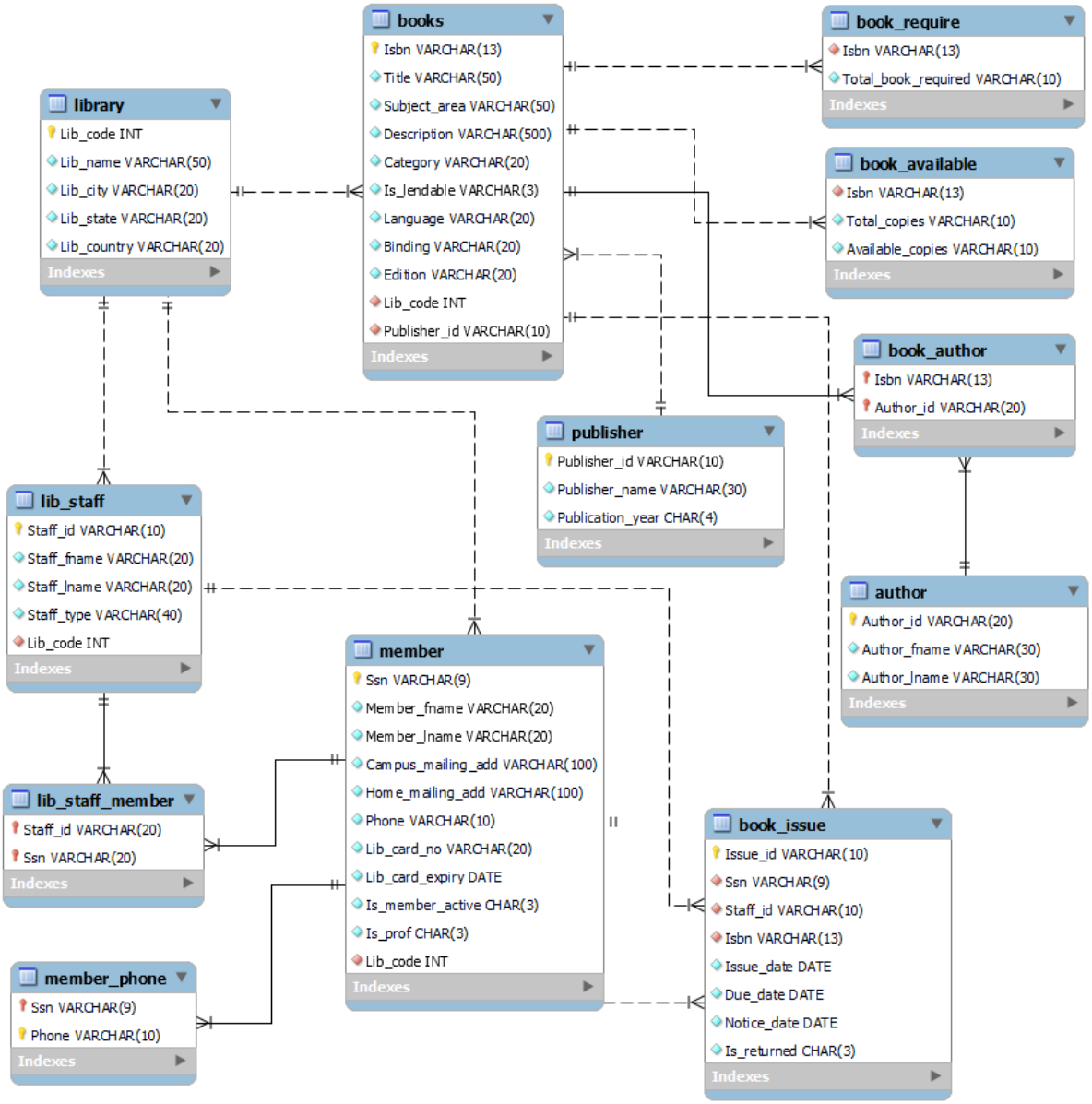
- Mapping of Relations based on the Strong entities identified in the EER Diagram generated as Part 1 of Project 2.
  - The following relations are generated as the below identified as strong entities– Library, Books, Author, Publisher, Book_require, Book_available, Lib_staff, and Member
- Mapping of Relations based on Mapping of Binary 1:1 Relationship between entities
  - Library is maintained by Library staff. Thus a one-to-one mapping is identified between Library and Lib_staff.
  - Lib_staff relation includes Lib_code as the foreign key which references the Lib_code attribute from the Library table.
- Mapping of Relations based on Mapping of Binary 1:N Relationship between entities
  - Library has/contains many books. Thus one-to-many relationship is identified between Library and Books.
  - Books relation includes Lib_code as a foreign key that references the Lib_code attribute from the Library table.
  - One Publisher can publish many books but a book is published by only one publisher. Thus a one-to-many relationship is identified between Publisher and Books.
  - Books relation includes Publisher_id as a foreign key that references the Publisher_id attribute from the Publisher table.
- Mapping of Relations based on Mapping of Binary M:N Relationship between entities
  - A book can be written by many authors and an author can write many different books. Thus, a many-to-many relationship is identified between Books and Authors entities.
  - Book_Author relation is created where it includes the Primary key of Book relation and the Primary key of Author relation
  - Library staff can keep track of multiple members of the library and Members can request multiple library staff for the issuance of books. Thus a many-to-many relationship is identified between Lib_staff who keeps track of members.
  - Lib_staff__member relation is created with Primary key attribute from Lib_staff and primary key attribute from member relation
- Mapping of Relations based on Mapping of Multi valued attribute
  - Member_phone relation is created as the Phone number is a multi-valued attribute. Thus, the entity that has multivalued attributes and the primary key of the entity are joined together and formed as a Member_phone relation. This multi-valued attribute and the primary key of the entity that has a multi-valued attribute (Ssn, Phone_no) are the primary key attribute of the relation member_phone.
- Mapping of Relations based on Mapping of N-ary Relationships between entities
  - Ternary relationship as there exists a relation between three entities Books_available, Lib_staff and Members.

- Members can request the library staff for the loan or borrowing activity. The library staff checks for the availability of books that is been requested by the library member and issues the book to the member. The books that are issued to the members of the library are identified by the Issue_id uniquely to keep track of books and members who loaned/borrowed the book. Thus a book_issue relation is formed with the primary key as Issue_id and attributes that are the primary key of the participating entities are added as a foreign key in the book_issue relation.

➢ Mapping of Specialization, Generalization (Super class and Subclasses)

- Mapping of Book_require and Book_available as Book_require and book_available are subclasses of parent class Book. The relation includes the primary key of the parent class (Book) and the attributes if the relation.

c) Mapping of Relation Schema from EER Diagram generated from Mysql Workbench



d) Tables Creation for the above generated Library Management System Relational Schema

```
use pxs9233;

#Library Table Creation
CREATE TABLE Library (
Lib_code int NOT NULL,
Lib_name varchar(50) NOT NULL,
Lib_city varchar(20) NOT NULL,
Lib_state varchar (20) NOT NULL,
Lib_country varchar(20) NOT NULL,
PRIMARY KEY (Lib_code)
);
commit;

#Publisher Table Creation
CREATE TABLE PUBLISHER (
Publisher_id varchar (10) NOT NULL,
Publisher_name varchar (30)NOT NULL,
Publication_year char(4) NOT NULL,
PRIMARY KEY (Publisher_id)
);
commit;

#Books Table Creation
CREATE TABLE Books (
Isbn varchar(13) NOT NULL,
Title varchar(50) NOT NULL,
Author_id varchar(50)NOT NULL,
Subject_area varchar(50) NOT NULL,
Description varchar(500) NOT NULL,
Category varchar(20) NOT NULL,
Is_lendable varchar(3) NOT NULL,
Language varchar(20) NOT NULL,
Binding varchar(20) NOT NULL,
Edition varchar(20) NOT NULL,
Lib_code int NOT NULL,
Publisher_id varchar (10) NOT NULL,
PRIMARY KEY (Isbn),
FOREIGN KEY (Lib_code) REFERENCES Library (Lib_code),
FOREIGN KEY (Publisher_id) REFERENCES Publisher(Publisher_id)
);
```

```
commit;

#Member Table Creation
CREATE TABLE Member (
Ssn varchar(9) NOT NULL,
Member_fname varchar(20) NOT NULL,
Member_lname varchar(20) NOT NULL,
Campus_mailing_add varchar(100) NOT NULL,
Home_mailing_add varchar(100) NOT NULL,
Phone varchar(10) NOT NULL,
Lib_card_no varchar(20) NOT NULL,
Lib_card_expiry date NOT NULL,
Is_member_active char(3) NOT NULL,
Is_prof char(3) NOT NULL,
Lib_code int NOT NULL,
PRIMARY KEY(Ssn),
FOREIGN KEY (Lib_code) REFERENCES Library(Lib_code)
);
commit;

#Author Table Creation
CREATE TABLE Author (
Author_id varchar (20) NOT NULL,
Author_fname varchar (30) NOT NULL,
Author_lname varchar (30) NOT NULL,
PRIMARY KEY (Author_id)
);
commit;

#Book_Author Table Creation
CREATE TABLE BOOK_AUTHOR(
Isbn varchar(13) NOT NULL,
Author_id varchar (20) NOT NULL,
Author_fname varchar (30) NOT NULL,
Author_lname varchar (30) NOT NULL,
PRIMARY KEY (Isbn, Author_id),
FOREIGN KEY (Isbn) REFERENCES Books(Isbn),
FOREIGN KEY (Author_id) REFERENCES Author(Author_id)
);
commit;

#Book_available Table Creation
CREATE TABLE Book_available (
Isbn varchar(13) NOT NULL,
```

```
Total_copies varchar(10) NOT NULL,
Available_copies varchar(10) NOT NULL,
FOREIGN KEY (Isbn) REFERENCES Books(Isbn)
);
commit;
```

#Book_require Table Creation

```
CREATE TABLE Book_require (
Isbn varchar(13) NOT NULL,
Total_book_required varchar(10) NOT NULL,
FOREIGN KEY (Isbn) REFERENCES Books(Isbn)
);
commit;
```

#Lib_staff Table Creation

```
CREATE TABLE Lib_staff (
Staff_id varchar(10) NOT NULL,
Staff_fname varchar(20) NOT NULL,
Staff_lname varchar(20) NOT NULL,
Staff_type varchar(40) NOT NULL,
Lib_code int NOT NULL,
PRIMARY KEY(Staff_id),
FOREIGN KEY (Lib_code) REFERENCES Library(Lib_code)
);
commit;
```

#Book_issue Table Creation

```
CREATE TABLE Book_issue (
Issue_id varchar(10),
Ssn varchar(9) NOT NULL,
Staff_id varchar(10) NOT NULL,
Isbn varchar(13) NOT NULL,
Issue_date date NOT NULL,
Due_date date NOT NULL,
Notice_date date NOT NULL,
Is_returned char(3) NOT NULL,
PRIMARY KEY (Issue_id),
FOREIGN KEY (Ssn) REFERENCES Member(Ssn),
FOREIGN KEY (Staff_id) REFERENCES Lib_staff(Staff_id),
FOREIGN KEY (Isbn) REFERENCES Books(Isbn)
);
commit;
```

#Library_member Table Creation

```
CREATE TABLE Lib_staff_member(
Staff_id varchar(20) NOT NULL,
Ssn varchar (20) NOT NULL,
PRIMARY KEY (Staff_id, Ssn),
FOREIGN KEY (Staff_id) REFERENCES Lib_staff(Staff_id),
FOREIGN KEY  (Ssn) REFERENCES Member(Ssn)
);
commit;
```

#Member_phone Table Creation

```
CREATE TABLE Member_phone (
Ssn varchar (9) NOT NULL,
Phone varchar(10) NOT NULL,
PRIMARY KEY (Ssn, Phone),
FOREIGN KEY  (Ssn) REFERENCES Member(Ssn)
);
commit;
```