COE848 Lab 3 Database Design

Note: Primary keys were mentioned for the lab because I forgot to mention it in the previous lab for each tale that was created. A modification was made with one of many to one relationships. I discarded the Amazon entity where the products would be shipped since I needed to ask how the professor exactly how the n-ary relationships because for the primary key I needed to know what exactly the primary key would have. After I am sure with the Amazon entity for my tables it will be included for the future labs. *(Any further modifications or changes will be mentioned in the following labs).*

Table Descriptions

1. **Publisher:** The publisher ID is the primary key of the Publisher table. The publisher table shares one to many relationships with the Book Table. Every Book table will have atleast one to relationship with a publisher entity set. The book will have total participation relationship with the publisher. The primary key of the publisher table will be stored as foreign key in the book table.
2. **Customer:** The customer ID will be the primary key of the customer table. The customer table shares many to many relationships with the book table
3. **Warehouse:** The Warehouse ID is the primary of the Warehouse. The Warehouse table shares one to many relationships with the Book Table. Every entity of the book entity set will have at least one relationship with the Warehouse entity. The book will have total participation with the Warehouse. The primary key of the Warehouse table will be stored as foreign key in the book table.
4. **Author:** Author ID will be the primary key of author table. The author table will share many to many relationships with the book table.
5. **FedEx:** The FedEx ID is the primary key of the FedEx Table. The FedEx shares one to many relationships with the Book Table. The primary key of the FedEx Table is stored as foreign key in the Book Table.
6. **Book:** The Book ID is the primary key of the Book Table. The Book Table shares three foreign keys in its table. These foreign keys are the primary keys of the Warehouse, author and FedEx Table.
7. **Customer Book:** The Customer and Book is having a many to many relationships. The junction table between customer and book is used. This table uses two foreign keys. These keys are primary keys of customer and book respectively. The constraint in the table ensures that no two relationships are repeated again.
8. **Book Author:** The book and author share many to many relationships. This results in use of junction table between the two tables. The primary key of book and the author becomes the two foreign keys used in this table. Also, the constraint ensures that no two relationships are repeated twice.
   The Book and the Author share many to many relationships. This results in the use of the junction table between the two tables. The primary key of the book and the author becomes the two foreign keys used in this Table. The constraint ensures that no two relationships are repeated consecutively.

CREATION OF DUMP OF THE DATABASE

```
PRAGMA foreign_keys=ON;
BEGIN TRANSACTION;
CREATE TABLE BOOK(
BookId INTEGER PRIMARY KEY AUTOINCREMENT,
BookName VARCHAR(50) NOT NULL,
Price INT NOT NULL,
ISBN INT,
Language VARCHAR(50),
PublicationDate VARCHAR(50),
AmazonRating INT,
Genre Char(10),
CONSTRAINT fk_dir FOREIGN KEY(BookId) REFERENCES BOOK (WarehouseId),
CONSTRAINT fk_dir FOREIGN KEY(BookId) REFERENCES Publisher (PublisherId),
CONSTRAINT fk_dir FOREIGN KEY(BookId) REFERENCES FedEx (FedExId)
);
CREATE TABLE Customer(
CustomerID INTEGER PRIMARY KEYT AUTOINCREMENT,
CustomerName VARCHAR(50) NOT NULL,
Price INT NOT NULL,
PhoneNO INT
);
Create TABLE Warehouse(
WarehouseId INTEGER PRIMARY KEY AUTOINCREMENT,
Address VARCHAR(50),
PhoneNO INT
);
Create TABLE Publisher(
PublisherID INTEGER PRIMARY KEY AUTOINCREMENT,
Address VARCHAR(50),
```

```
Website VARCHAR(50)
);
CREATE TABLE Author(
AuthorID INTEGER PRIMARY KEY AUTOINCREMENT,
AuthorName VARCHAR(50) NOT NULL,
Website VARCHAR(50) NOT NULL,
PhoneNo INT
);
CREATE TABLE FedEx(
FedExId INTEGER PRIMARY KEY AUTOINCREMENT,
Address VARCHAR(50),
PhoneNo INT
);
CREATE TABLE CustomerBook(
CustomerId INT NOT NULL,
BookId INT NOT NULL,
CONSTRAINT PK_CustomerBook PRIMARY KEY
(
 CustomerId,
 BookId
),
FOREIGN KEY(CustomerId) REFERENCES Customer(CustomerId),
FOREIGN KEY(BookId) REFERENCES Book(BookId)
);
Create TABLE BookAuthor(
AuthorId INT NOT NULL,
BookId INT NOT NULL,
CONSTRAINT PK_BookAuthor PRIMARY KEY
(
 AuthorId,
 BookId
```

```
),
FOREIGN KEY(AuthorId) REFERENCES Author(AuthorId),
FOREIGN KEY(BookId) REFERENCES Book(BookId)
);
COMMIT;
```

Note: The (+) is the indication of all the attributes being considered.

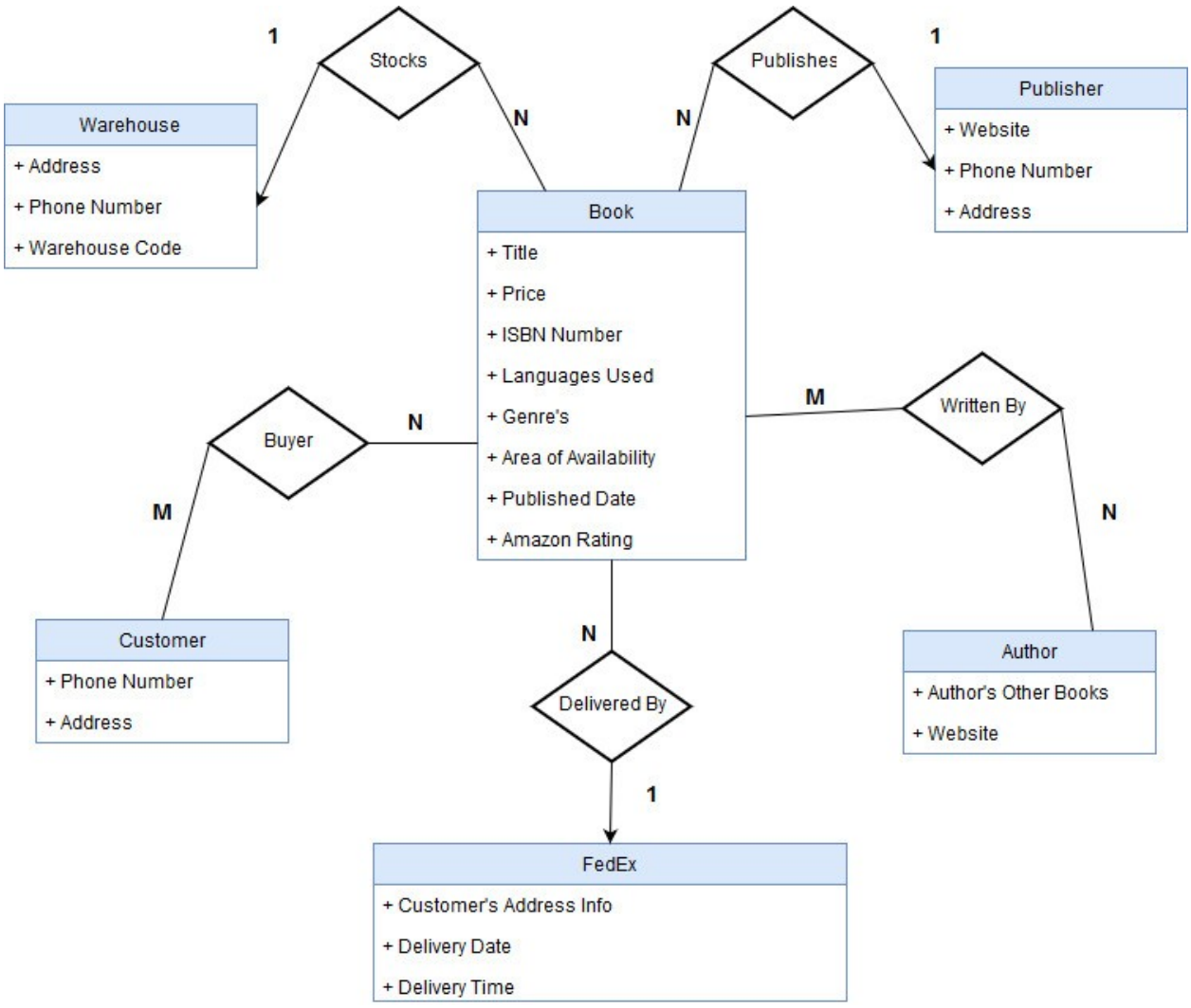

Figure 1: The image indicates the Entity Relation (E-R) Diagram Design.