



Group Assignment

AICT005 - DAS

Database Systems

UCDF2005(1)ICT(DI)

Lecturer: Miss Seetha Letchumy

Weightage: 50%

Hand out date: 19 May 2021

Hand in date: 11 June 2021

Group name: Group 1

Part: Part 2

Group member:

No.	Name	TP No.
1	Cheong Sheng Kui	TP060652
2	Lai Kai Yong	TP059040
3	Lim Wye Yee	TP059371

Workload Matrix

Name / TP Number	Responsibilities
Lai Kai Yong (TP059040)	Responsible for Final Entity Relationship Diagram, Relational Schema, Normalization Process, Data Dictionary for Tables, DBMS Database Diagram, SQL-Data Definition Language (DDL), Insert Statement, Data in Tables and SQL-Data Manipulation Language (DML) – Q1, Q2, Q3, Q4.
Cheong Sheng Kui (TP060652)	Responsible for Final Entity Relationship Diagram, Relational Schema, Normalization Process, Data Dictionary for Tables, DBMS Database Diagram, SQL-Data Definition Language (DDL), Insert Statement, Data in Tables and SQL-Data Manipulation Language (DML) – Q5, Q6, Q7.
Lim Wye Yee (TP059371)	Responsible for Final Entity Relationship Diagram, Relational Schema, Normalization Process, Data Dictionary for Tables, DBMS Database Diagram, SQL-Data Definition Language (DDL), Insert Statement, Data in Tables and SQL-Data Manipulation Language (DML) – Q8, Q9, Q10.

Introduction

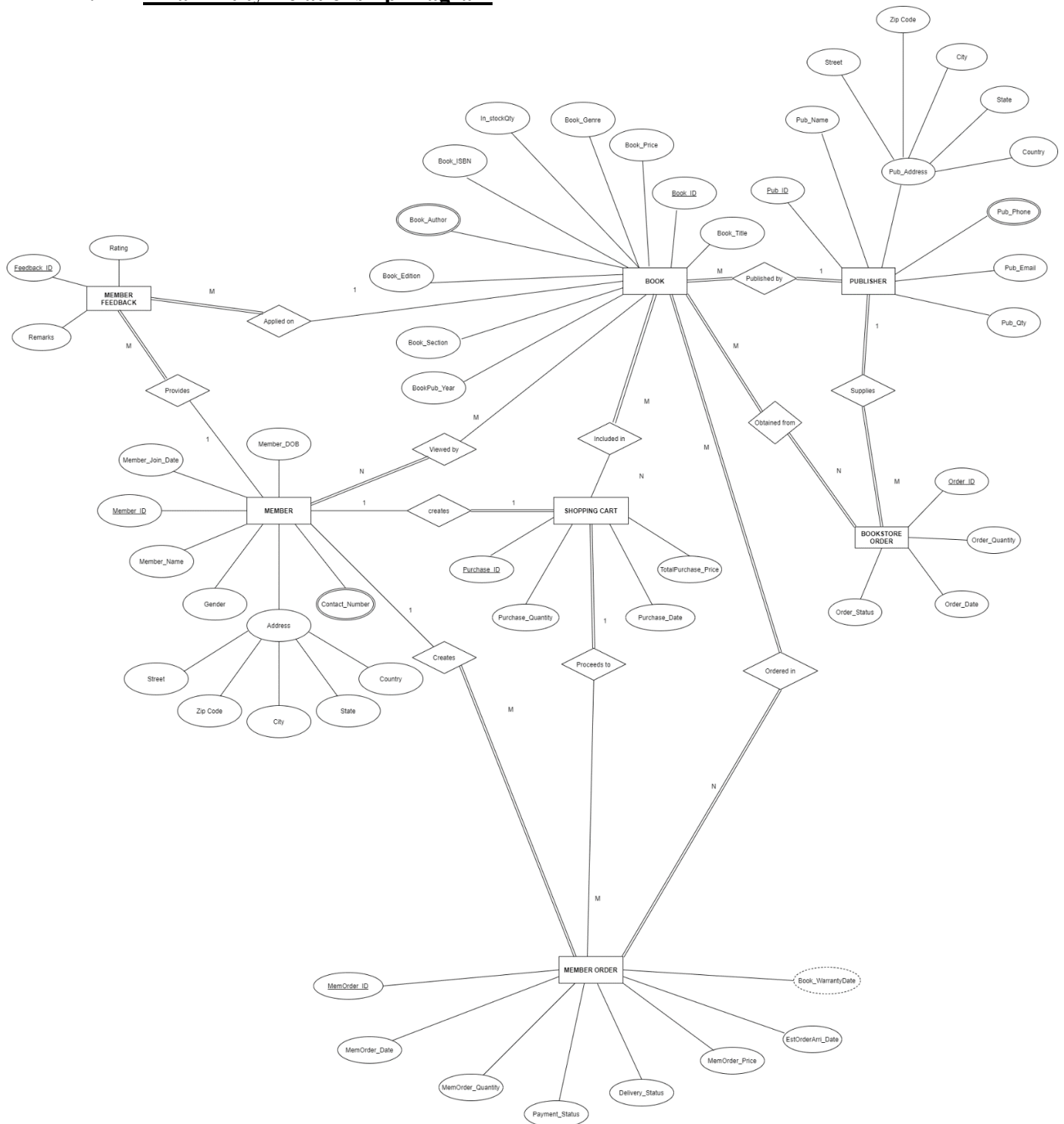
This documentation aims to utilize fundamental knowledge on implementing database (DBMS) design using data dictionary as reference for appropriate data types and constraint. Relating to the scenario, students as the APU E-bookstore project team are required to build a DBMS in Microsoft SQL Server Management Studio (SSMS). The database diagram is constructed based on a finalized Entity-Relationship Diagram (ERD) with minor modifications. Corresponding relational schema is derived from the diagram after normalizing the data to 3NF by remove repeating groups, reducing data redundancy, determining full dependencies and transitive dependencies. A data dictionary table has to be generated to act as guidance for data types and constraints needed when creating the DBMS in the Microsoft SSMS. The manipulation of SQL statement is implemented for the creation of DBMS and queries manipulation which are Data Definition Language (DDL) and Data Manipulation Language (DML). DDL is used to construct the structure of the APU E-bookstore in terms of data type for each column in the tables and insertion of required data. DML is implemented to solve ten queries based on specific conditions and requirements to display a better view on the database.

Table of Contents

I.	Database Schema	1
A.	Final Entity Relationship Diagram	1
B.	Corresponding Relational Schema.....	3
C.	Normalization Process	4
D.	Relations in 3NF	9
E.	Data Dictionary for Tables.....	10
F.	DBMS Database Diagram.....	18
II.	SQL-Data Definition Language (DDL).....	22
1.	Insert Statement.....	25
2.	Data in Tables.....	36
III.	SQL-Data Manipulation Language (DML)	48
IV.	Conclusion	56
V.	References	57

I. Database Schema

A. Final Entity Relationship Diagram



ERD Final Modification:

Due to some issues identified in the visualization of the database, some amendment on the Entity Relationship Diagram had been implemented.

- **Book_WarrantyDate** had been set to derived attribute.
- Relation **BOOK_MEMBER_ORDER** had been added.

B. Corresponding Relational Schema

BOOK (Book_ID, Book_ISBN, Book_Price, Book_Section, Book_Title, Book_Genre, Book_Edition, BookPub_Year, In_stockQty, Pub_ID (FK))

BOOK_AUTHOR (Book_ID (FK), First_Author, Second_Author, Third_Author)

MEMBER (Member_ID, Member_Name, Gender, Member_DOB, Member_Join_Date, Street, Zip Code, City, State, Country)

MEMBER_CONTACT_NUMBER (Member_ID (FK), Personal_Phone, Work_Phone)

PUBLISHER (Pub_ID, Pub_Name, Pub_Email, Pub_Qty, Street, Zip Code, City, State, Country)

PUB_PHONE (Pub_ID (FK), Headquarter_phoneNo, Branch_phoneNo)

BOOKSTORE ORDER (Order_ID, Order_Date, Order_Status, Pub_ID (FK))

SHOPPING CART (Purchase_ID, Purchase_Date, TotalPurchase_Price, Member_ID (FK))

MEMBER ORDER (MemOrder_ID, MemOrder_Date, Payment_Status, Delivery_Status, MemOrder_Price, EstOrderArri_date, Purchase_ID (FK), Member_ID(FK))

MEMBER FEEDBACK (Feedback_ID, Rating, Remarks, Member_ID(FK), Book_ID(FK))

BOOK_MEMBER (Book_ID (FK), Member_ID (FK))

BOOK_SHOPPING_CART (Book_ID(FK), Purchase_ID(FK), Purchase_Quantity)

BOOK_BOOKSTORE_ORDER(Book_ID(FK), Order_ID(FK), Order_Quantity)

BOOK_MEMBER_ORDER(Book_ID(FK), MemOrder_ID(FK), MemOrder_Quantity)

C. Normalization Process

1. **BOOK** (Book_ID, Book_ISBN, Book_Price, Book_Section, Book_Title, Book_Genre, Book_Edition, BookPub_Year, In_stockQty, Pub_ID (FK))

Functional dependencies

Book_ID → Book_ISBN, Book_Price, Book_Section, Book_Title, Book_Genre, Book_Edition, BookPub_Year, In_stockQty, Pub_ID (full dependency)

Book ISBN → Book_Title, Book_Genre, Book_Edition, BookPub_Year (transitive dependency)

3NF

BOOK (Book_ID, Book_ISBN, Book_Price, Book_Section, In_stockQty, Pub_ID (FK))

UNIVERSAL_BOOK (Book_ISBN, Book_Title, Book_Genre, Book_Edition, BookPub_Year)

2. **BOOK_AUTHOR** (Book_ID (FK), First_Author, Second_Author, Third_Author)

Functional dependencies

Book_ID → First_Author, Second_Author, Third_Author (full dependency)

Since there are no partial or transitive dependencies, the BOOK_AUTHOR relation above is already in 3NF.

3. **MEMBER** (Member_ID, Member_Name, Gender, Member_DOB, Member_Join_Date, Street, Zip Code, City, State, Country)

Functional dependencies

Member_ID → Member_Name, Gender, Member_DOB, Member_Join_Date, Street, Zip Code, City, State, Country (full dependency)

Since there are no partial or transitive dependencies, the MEMBER relation above is already in 3NF.

4. **MEMBER_CONTACT_NUMBER** (Member_ID (FK), Personal_Phone, Work_Phone)

Functional dependencies

Member_ID → Personal_Phone, Work_Phone (full dependency)

Since there are no partial or transitive dependencies, the MEMBER_CONTACT_NUMBER relation above is already in 3NF.

5. **PUBLISHER** (Pub_ID, Pub_Name, Pub_Email, Pub_Qty, Street, Zip Code, City, State, Country)

Functional dependencies

Pub_ID → Pub_Name, Pub_Email, Pub_Qty, Street, Zip Code, City, State, Country (full dependency)

Since there are no partial or transitive dependencies, the PUBLISHER relation above is already in 3NF.

6. **PUB_PHONE** (Pub_ID (FK), Headquarter_phoneNo, Branch_phoneNo)

Functional dependencies

Pub_ID → Headquarter_phoneNo, Branch_phoneNo (full dependency)

Since there are no partial or transitive dependencies, the PUB_PHONE relation above is already in 3NF.

7. **BOOKSTORE ORDER** (Order_ID, Order_Date, Order_Status, Pub_ID (FK))

Functional dependencies

Order_ID → Order_Date, Order_Quantity, Order_Status, Pub_ID (full dependency)

Since there are no partial or transitive dependencies, the BOOKSTORE ORDER relation above is already in 3NF.

8. **SHOPPING CART** (Purchase_ID, Purchase_Date, TotalPurchase_Price, Member_ID (FK))

Functional dependencies

Purchase_ID → Purchase_Date, TotalPurchase_Price, Member_ID (full dependency)

Since there are no partial or transitive dependencies, the SHOPPING CART relation above is already in 3NF.

9. **MEMBER ORDER** (MemOrder_ID, MemOrder_Date, Payment_Status, Delivery_Status, MemOrder_Price, EstOrderArri_date, Purchase_ID (FK), Member_ID(FK))

Functional dependencies

MemOrder_ID → MemOrder_Date, Payment_Status, Delivery_Status, MemOrder_Price, EstOrderArri_date, Purchase_ID, Member_ID (full dependency)

Since there are no partial or transitive dependencies, the MEMBER ORDER relation above is already in 3NF.

10. **MEMBER FEEDBACK** (Feedback_ID, Rating, Remarks, Member_ID(FK), Book_ID(FK))

Functional dependencies

Feedback_ID → Rating, Remarks, Member_ID, Book_ID (full dependency)

Since there are no partial or transitive dependencies, the MEMBER FEEDBACK relation above is already in 3NF.

11. **BOOK_MEMBER** (Book_ID (FK), Member_ID (FK))

Since there are no non-key attributes in the BOOK-MEMBER relation, it is already in 3NF.

12. **BOOK_SHOPPING_CART** (Book_ID(FK), Purchase_ID(FK), Purchase_Quantity)

Functional dependencies

Book_ID, Purchase_ID → Purchase_Quantity (full dependency)

Since there are no partial or transitive dependencies, the BOOK_SHOPPING_CART relation above is already in 3NF.

13. **BOOK_BOOKSTORE_ORDER**(Book_ID(FK), Order_ID(FK), Order_Quantity)

Functional dependencies

Book_ID, Order_ID → Order_Quantity (full dependency)

Since there are no partial or transitive dependencies, the BOOK_BOOKSTORE_ORDER relation above is already in 3NF.

14. **BOOK_MEMBER_ORDER**(Book_ID(FK), MemOrder_ID(FK), MemOrder_Quantity)

Functional dependencies

Book_ID, MemOrder_ID \rightarrow MemOrder_Quantity (full dependency)

Since there are no partial or transitive dependencies, the BOOK_MEMBER_ORDER relation above is already in 3NF.

D. Relations in 3NF

BOOK (Book_ID, Book_ISBN, Book_Price, Book_Section, In_stockQty, Pub_ID (FK))

UNIVERSAL_BOOK (Book_ISBN, Book_Title, Book_Genre, Book_Edition, BookPub_Year)

BOOK_AUTHOR (Book_ID (FK), First_Author, Second_Author, Third_Author)

MEMBER (Member_ID, Member_Name, Gender, Member_DOB, Member_Join_Date, Street, Zip Code, City, State, Country)

MEMBER_CONTACT_NUMBER (Member_ID (FK), Personal_Phone, Work_Phone)

PUBLISHER (Pub_ID, Pub_Name, Street, Zip Code, City, State, Country, Pub_Email, Pub_Qty)

PUB_PHONE (Pub_ID (FK), Headquarter_phoneNo, Branch_phoneNo)

BOOKSTORE_ORDER (Order_ID, Order_Date, Order_Status, Pub_ID (FK))

SHOPPING_CART (Purchase_ID, Purchase_Date, TotalPurchase_Price, Member_ID (FK))

MEMBER_ORDER (MemOrder_ID, MemOrder_Date, Payment_Status, Delivery_Status, MemOrder_Price, EstOrderArri_date, Purchase_ID (FK), Member_ID(FK))

MEMBER_FEEDBACK (Feedback_ID, Rating, Remarks, Member_ID(FK), Book_ID(FK))

BOOK_MEMBER (Book_ID (FK), Member_ID (FK))

BOOK_SHOPPING_CART (Book_ID(FK), Purchase_ID(FK), Purchase_Quantity)

BOOK_BOOKSTORE_ORDER(Book_ID(FK), Order_ID(FK), Order_Quantity)

BOOK_MEMBER_ORDER(Book_ID(FK), MemOrder_ID(FK), MemOrder_Quantity)

E. Data Dictionary for Tables

Table Name	Attribute Name	Contents	Type	Format	Range	Required	PK or FK	FK Referenced Table
BOOK	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999	Y	PK	UNIVERSAL_BOOK
	Book_ISBN	Book ISBN	CHAR(13)	9999999999999	00000000000000 - 99999999999999	Y	FK	
	Book_Price	Book Price	SMALLMONEY	RM99999	RM1 – RM99999	Y		
	Book_Section	Book Stored Section	TINYINT	9	1 - 9	Y		
	In_stockQty	Book in Stock Quantity	INT	99999	1 - 99999	Y		
	Pub_ID	Publisher ID	CHAR(8)	“PU”999999	PU000001 - PU999999	Y	FK	PUBLISHER
UNIVERSAL_BOOK	Book_ISBN	Book ISBN	CHAR(13)	9999999999999	00000000000000 - 99999999999999	Y	PK	
	Book_Title	Book Title / Name	VARCHAR(60)	Xxxxxxxx		Y		
	Book_Genre	Book Genre / Type	VARCHAR(25)	Xxxxxxxx		Y		
	Book_Edition	Book Edition	TINYINT	99	1 - 99	Y		

	BookPub_Year	Book Published Year	SMALLINT	9999	0000 - 9999	Y		
BOOK_AUTHOR	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999	Y	FK	BOOK
	First_Author	Book First Author	VARCHAR(30)	XXXXXXXXXX		Y		
	Second_Author	Book Second Author	VARCHAR(30)	XXXXXXXXXX				
	Third_Author	Book Third Author	VARCHAR(30)	XXXXXXXXXX				
MEMBER	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 - ME999999	Y	PK	
	Member_Name	Member Name	VARCHAR(30)	XXXXXXXXXX		Y		
	Gender	Member Gender	CHAR(1)	X	F or M	Y		
	Member_DOB	Member Date of Birth	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31	Y		
	Member_Join_Date	Member Join Date	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31	Y		
	Street	Member Address Street	VARCHAR(60)	XXXXXXXXXX		Y		
	Zip Code	Member Address Zip Code	CHAR(5)	99999	00000 - 99999	Y		

	City	Member Address City	VARCHAR(60)	XXXXXXXXXX		Y		
	State	Member Address State	VARCHAR(60)	XXXXXXXXXX		Y		
	Country	Member Address Country	VARCHAR(60)	XXXXXXXXXX		Y		
MEMBER_CONTACT_NUMBER	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 – ME999999	Y	FK	MEMBER
	Personal_Phone	Member Personal Phone	CHAR(11)	999-999999999	000-0000000 – 999-9999999			
	Work_Phone	Member Work Phone	CHAR(11)	99-999999999	00-00000000 – 99-99999999			
PUBLISHER	Pub_ID	Publisher ID	CHAR(8)	“PU”999999	PU000001 – PU999999	Y	PK	
	Pub_Name	Publisher Name	VARCHAR(50)	XXXXXXXXXX		Y		
	Pub_Email	Publisher E-mail	VARCHAR(40)	Xxx@xxx		Y		
	Pub_Qty	Publisher Book Supplied Quantity	INT	99999		Y		
	Street	Publisher Address Street	VARCHAR(60)	XXXXXXXXXX		Y		
	Zip Code	Publisher Address Zip Code	CHAR(5)	99999	00000 - 99999	Y		

	City	Publisher Address City	VARCHAR(60)	XXXXXXXXXX		Y		
	State	Publisher Address State	VARCHAR(60)	XXXXXXXXXX		Y		
	Country	Publisher Address Country	VARCHAR(60)	XXXXXXXXXX		Y		
PUB_PHONE	Pub_ID	Publisher ID	CHAR(8)	“PU”999999	PU000001 – PU999999	Y	FK	PUBLISHER
	Headquarter_phoneNo	Publisher Headquarter Phone Number	CHAR(11)	99-99999999	00-00000000 – 99-99999999	Y		
	Branch_phoneNo	Publisher Branch Phone Number	CHAR(11)	99-99999999	00-00000000 – 99-99999999			
BOOKSTORE ORDER	Order_ID	Bookstore Order ID	CHAR(8)	“OR”999999	OR000001 – OR999999	Y	PK	
	Order_Date	Bookstore Order Date	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31	Y		
	Order_Status	Bookstore Order Status	VARCHAR(15)	XXXXXXXXX		Y		
	Pub_ID	Publisher ID	CHAR(8)	“PU”999999	PU000001 – PU999999	Y	FK	PUBLISHER
SHOPPING CART	Purchase_ID	Shopping Cart	CHAR(8)	“PA”999999	PA000001 – PA999999	Y	PK	

	Purchase_Date	Purchase ID Shopping Cart Purchase Date	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31	Y		
	TotalPurchase_Price	Shopping Cart Total Purchase Price	MONEY	RM999999	RM1 – RM999999	Y		
	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 – ME999999	Y	FK	MEMBER
MEMBER ORDER	MemOrder_ID	Member Order ID	CHAR(8)	“MO”999999	MO000001 – MO999999	Y	PK	
	MemOrder_Date	Member Order Date	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31	Y		
	Payment_Status	Member Order Payment Status	VARCHAR(15)	Xxxxxxxx		Y		
	Delivery_Status	Member Order Delivery Status	VARCHAR(15)	Xxxxxxxx				
	MemOrder_Price	Member Order Price	MONEY	RM999999	RM1 – RM999999	Y		
	EstOrderArri_date	Member Order Estimate Arrival Date	DATE	YYYY-MM-DD	0001 – 01 - 01 to 9999 – 12 - 31			

	Purchase_ID	Shopping Cart Purchase ID	CHAR(8)	“PA”999999	PA000001 – PA999999	Y	FK	SHOPPING CART
	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 – ME999999	Y	FK	MEMBER
MEMBER FEEDBACK	Feedback_ID	Member Feedback ID	CHAR(8)	“FI”999999	FI000001 – FI999999	Y	PK	
	Rating	Member Feedback Rating	INT	99	1-10	Y		
	Remarks	Member Feedback Remarks	VARCHAR(60)	XXXXXXXXxx				
	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 – ME999999	Y	FK	MEMBER
	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999	Y	FK	BOOK
BOOK_MEMBER	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999	Y	FK	BOOK
	Member_ID	Member ID	CHAR(8)	“ME”999999	ME000001 – ME999999	Y	FK	MEMBER
BOOK_SHOPPING_CART	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999	Y	FK	BOOK
	Purchase_ID	Shopping Cart Purchase ID	CHAR(8)	“PA”999999	PA000001 – PA999999	Y	FK	SHOPPING CART
	Purchase_Quantity	Shopping Cart Purchase Quantity	INT	999999	1 - 999999	Y		

BOOK_BOOKSTORE_ORDER	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999		Y	FK	BOOK
	Order_ID	Bookstore Order ID	CHAR(8)	“OR”999999	OR000001 – OR999999		Y	FK	BOOKSTORE_ORDER
	Order_Quantity	Bookstore Order Quantity	INT	999999	1 - 999999	Y			
BOOK_MEMBER_ORDER	Book_ID	Book ID	CHAR(8)	“BO”999999	BO000001 - BO999999		Y	FK	BOOK
	MemOrder_ID	MemOrder_ID	CHAR(8)	“MO”999999	MO000001 – MO999999		Y	FK	MEMBER
	MemOrder_Quantity	Member Order Quantity	INT	999999	1 - 999999		Y		ORDER

PK- Primary Key uniquely identifies every record in a table. Each table can only consist of one primary key which is a unique value and cannot be NULL value.

FK- Foreign Key or referencing key, links two tables together. Foreign Key is a column or combination of columns where the value matches the Primary Key in another table.

CHAR- fixed length string which contains letters, numbers, and special characters. The size ranges from 0 to 255 lengths in character, where 1 is the default value assigned.

VARCHAR- variable length string which contains letters, numbers, and special characters. The size ranges from 0 to 65535 lengths in character.

MONEY- monetary data from ranging from -922,337,203,685,477.5808 to 922,337,203,685,477.5807.

SMALLMONEY- monetary data ranging from -214,748.3648 to 214,748.3647.

INT- Enables whole numbers from -2,147,483,648 to 2,147,483,647 to be inserted.

TINYINT- Enables whole numbers from 0 to 255 to be inserted.

SMALLINT- Enables whole numbers from -32,768 to 32,767 to be inserted.

DATE- Date has a format of YYYY-MM-DD and supports date ranging from '1000-01-01' to '9999-12-31'

(W3Schools, 2021)

Note:

BookPub_Year cannot be stored as YEAR value since Microsoft SQL does not supports YEAR value. Hence, **BookPub_Year** is stored as SMALLINT to remain it as a 4-value data. The reason that DATE is not selected is because the value will be stored in the format of YYYY-MM-DD while **BookPub_Year** only consists of YYYY format (Peter, 2009).

F. DBMS Database Diagram

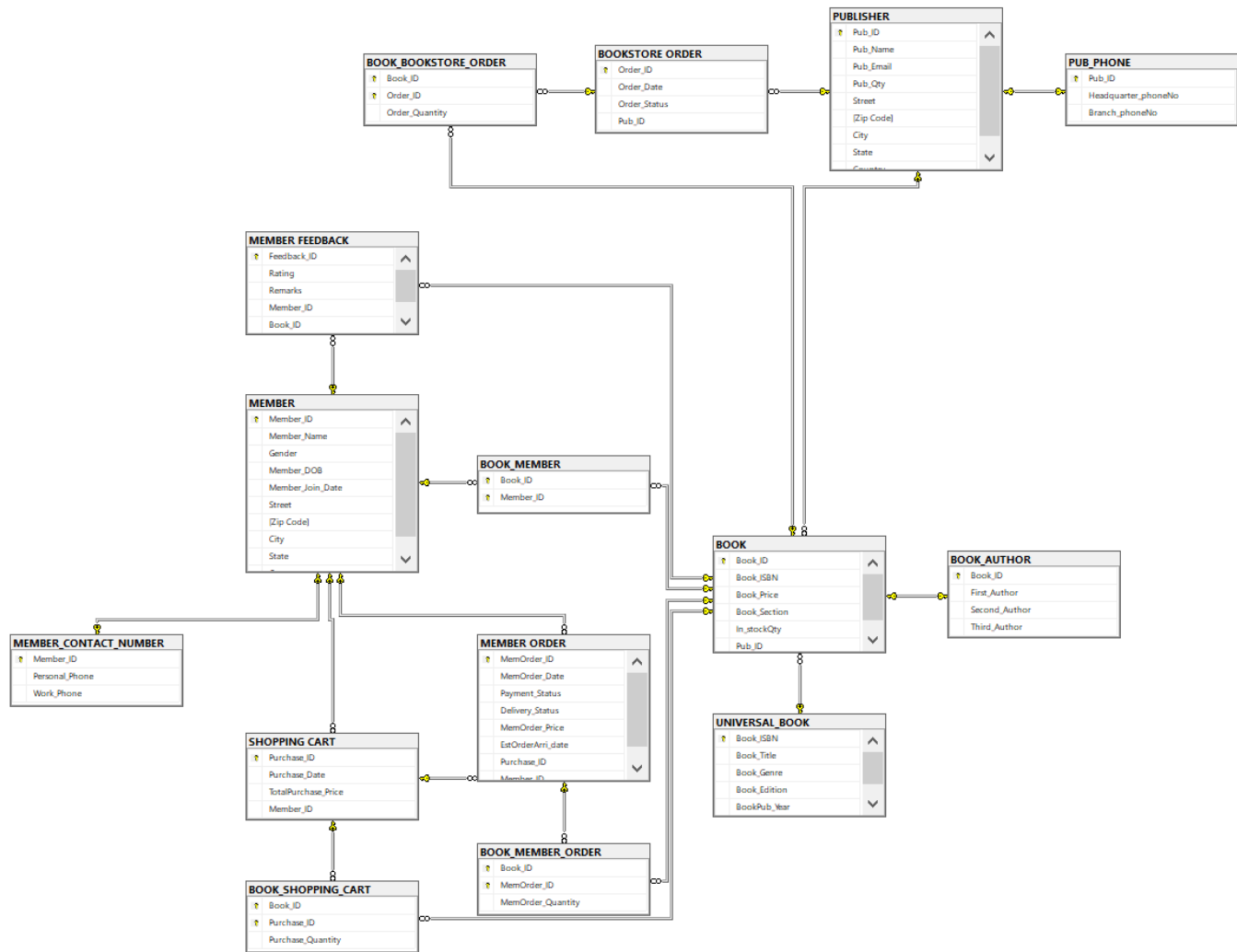


Figure I.1: DBMS Database Diagram full figure

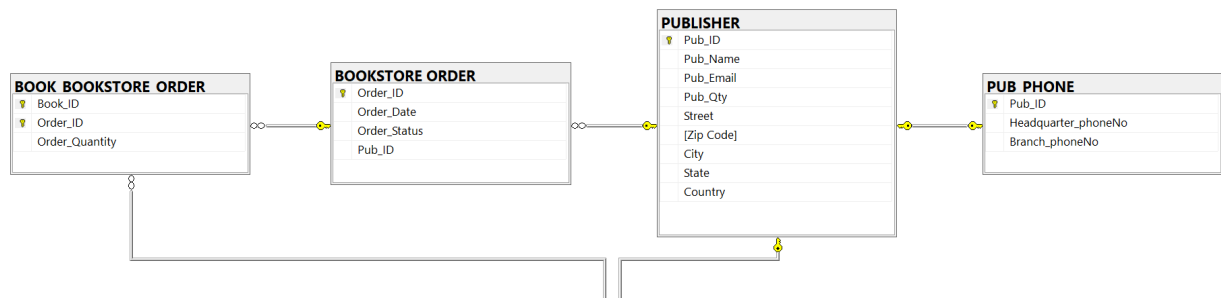


Figure I.2: Bookstore order and Publisher part of the Database Diagram

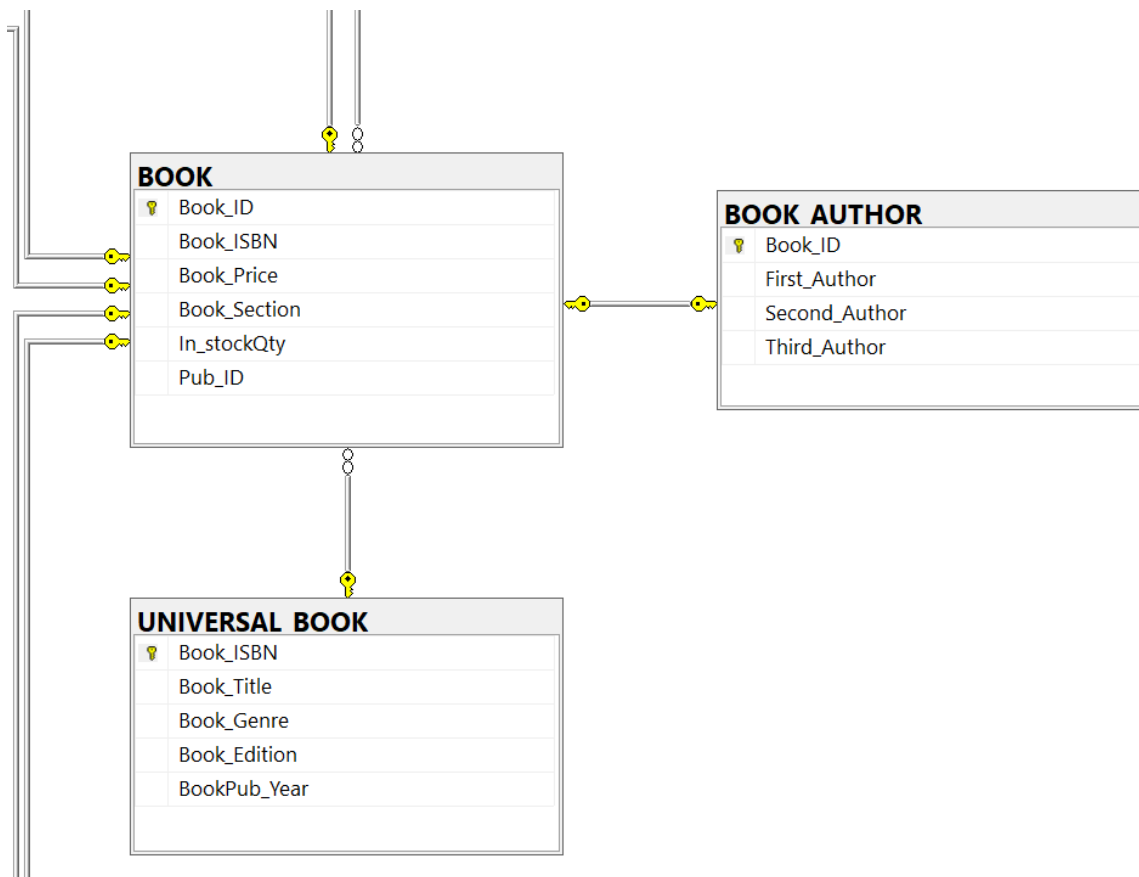


Figure I.3: Book part of the Database Diagram

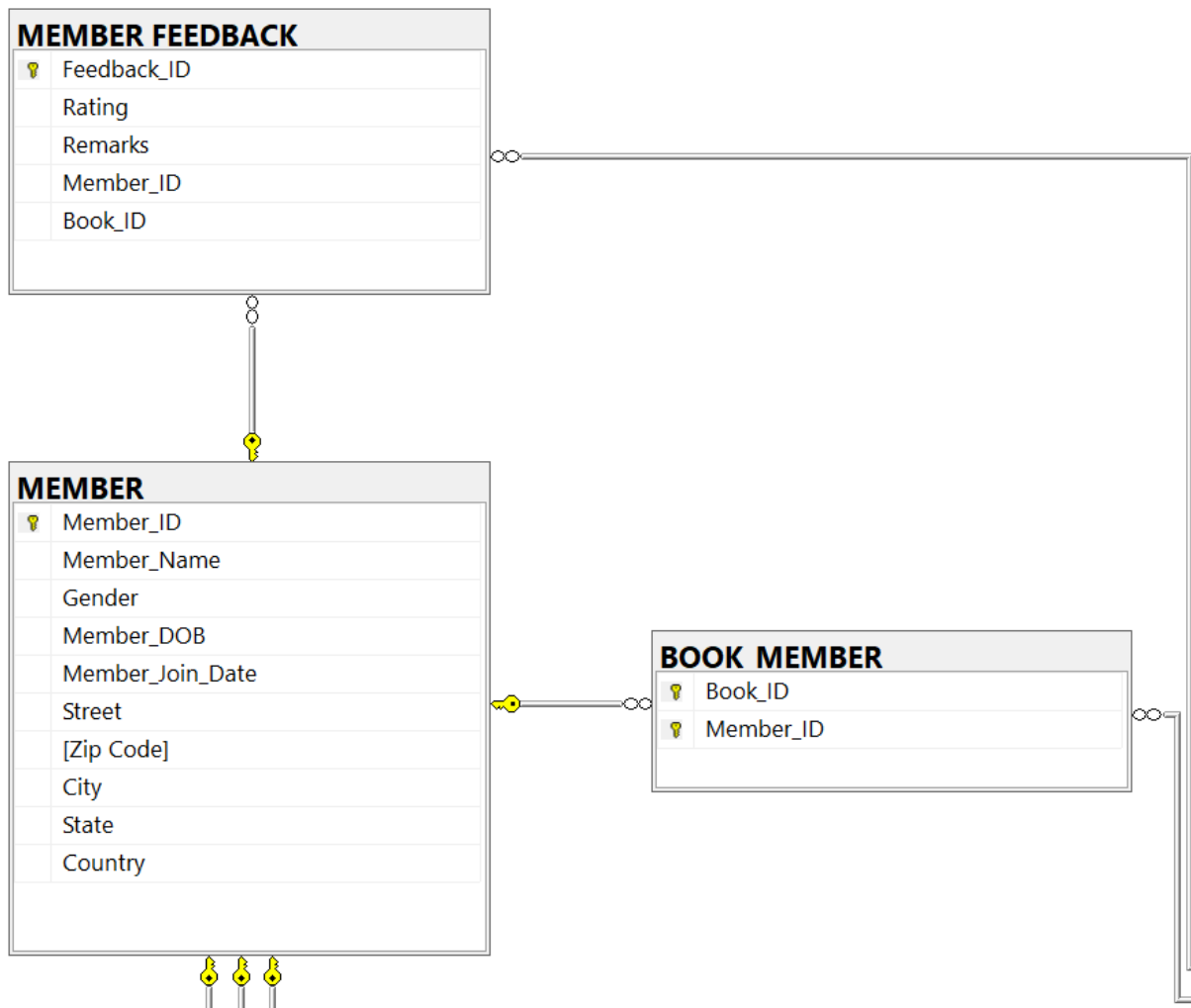


Figure I.4: Member part of the Database Diagram

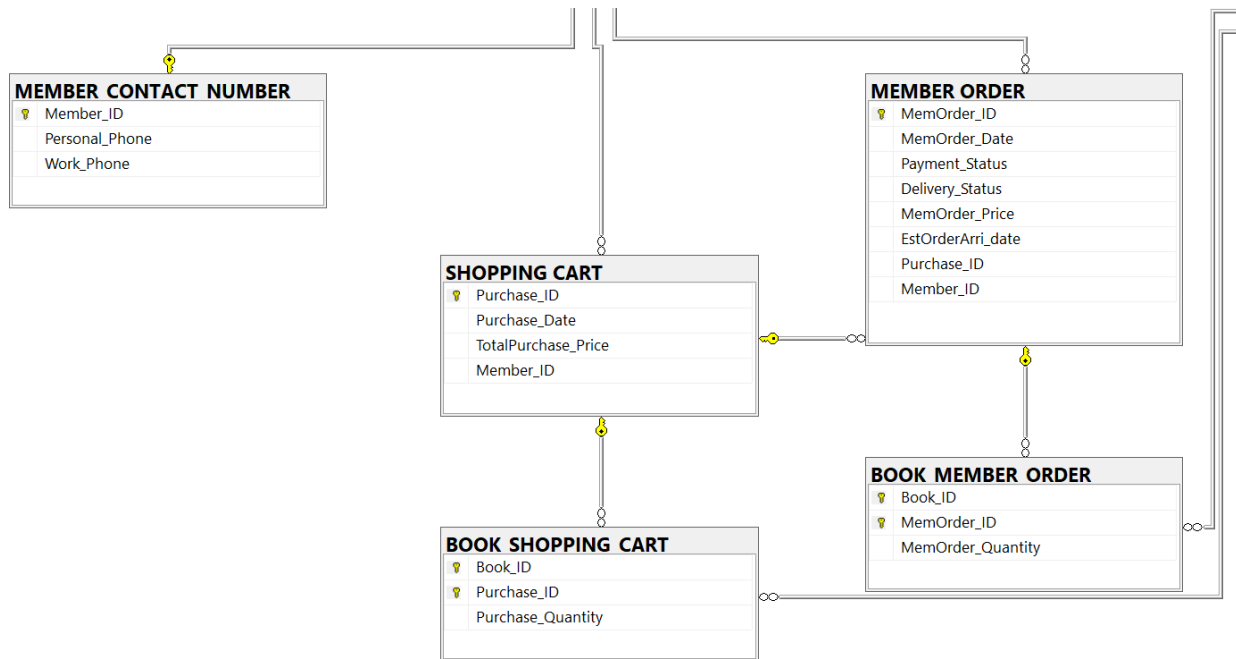


Figure I.5: Member / Shopping cart and member order part of the Database Diagram

II. SQL-Data Definition Language (DDL)

```
CREATE DATABASE APUBookstore;
```

```
USE APUBookstore;
```

```
CREATE TABLE BOOK (  
    Book_ID CHAR(8) NOT NULL PRIMARY KEY,  
    Book_ISBN CHAR(13) NOT NULL FOREIGN KEY  
        REFERENCES UNIVERSAL_BOOK(Book_ISBN),  
    Book_Price SMALLMONEY NOT NULL,  
    Book_Section TINYINT NOT NULL,  
    In_stockQty INT NOT NULL,  
    Pub_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES PUBLISHER(Pub_ID)  
);
```

```
CREATE TABLE UNIVERSAL_BOOK (  
    Book_ISBN CHAR(13) NOT NULL PRIMARY KEY,  
    Book_Title VARCHAR(60) NOT NULL,  
    Book_Genre VARCHAR(25) NOT NULL,  
    Book_Edition TINYINT NOT NULL,  
    BookPub_Year SMALLINT NOT NULL  
);
```

```
CREATE TABLE BOOK_AUTHOR (  
    Book_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES BOOK(Book_ID),  
    First_Author VARCHAR(30) NOT NULL,  
    Second_Author VARCHAR(30),  
    Third_Author VARCHAR(30),  
    PRIMARY KEY(Book_ID)  
);
```

```
CREATE TABLE MEMBER (  
    Member_ID CHAR(8) NOT NULL PRIMARY KEY,  
    Member_Name VARCHAR(30) NOT NULL,  
    Gender CHAR(1) NOT NULL,  
    Member_DOB DATE NOT NULL,  
    Member_Join_Date DATE NOT NULL,  
    Street VARCHAR(60) NOT NULL,  
    [Zip Code] CHAR(5) NOT NULL,  
    City VARCHAR(60) NOT NULL,  
    State VARCHAR(60) NOT NULL,  
    Country VARCHAR(60) NOT NULL  
);
```

```
CREATE TABLE MEMBER_CONTACT_NUMBER (  
    Member_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES MEMBER(Member_ID),  
    Personal_Phone CHAR(11),  
    Work_Phone CHAR(11),  
    PRIMARY KEY(Member_ID)  
);
```

```
CREATE TABLE PUBLISHER (  

```

```
Pub_ID CHAR(8) NOT NULL PRIMARY KEY,
Pub_Name VARCHAR(50) NOT NULL,
Pub_Email VARCHAR(40) NOT NULL,
Pub_Qty INT NOT NULL,
Street VARCHAR(60) NOT NULL,
[Zip Code] CHAR(5) NOT NULL,
City VARCHAR(60) NOT NULL,
State VARCHAR(60) NOT NULL,
Country VARCHAR(60) NOT NULL
);

CREATE TABLE PUB_PHONE (
    Pub_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES PUBLISHER(Pub_ID),
    Headquarter_phoneNo CHAR(11) NOT NULL,
    Branch_phoneNo CHAR(11),
    PRIMARY KEY(Pub_ID)
);

CREATE TABLE [BOOKSTORE ORDER](
    Order_ID CHAR(8) NOT NULL PRIMARY KEY,
    Order_Date DATE NOT NULL,
    Order_Status VARCHAR(15) NOT NULL,
    Pub_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES PUBLISHER(Pub_ID)
);

CREATE TABLE [SHOPPING CART](
    Purchase_ID CHAR(8) NOT NULL PRIMARY KEY,
    Purchase_Date DATE NOT NULL,
    TotalPurchase_Price MONEY NOT NULL,
    Member_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES MEMBER(Member_ID)
);

CREATE TABLE [MEMBER ORDER] (
    MemOrder_ID CHAR(8) NOT NULL PRIMARY KEY,
    MemOrder_Date DATE NOT NULL,
    Payment_Status VARCHAR(15) NOT NULL,
    Delivery_Status VARCHAR(15),
    MemOrder_Price MONEY NOT NULL,
    EstOrderArri_date DATE,
    Purchase_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES [SHOPPING CART](Purchase_ID),
    Member_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES Member(Member_ID)
);

CREATE TABLE [MEMBER FEEDBACK](
    Feedback_ID CHAR(8) NOT NULL PRIMARY KEY,
    Rating INT NOT NULL,
    Remarks VARCHAR(60),
    Member_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES Member(Member_ID),
    Book_ID CHAR(8) NOT NULL FOREIGN KEY
        REFERENCES BOOK(Book_ID),
);
```

```
CREATE TABLE BOOK_MEMBER (  
    Book_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES BOOK(Book_ID),  
    Member_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES Member(Member_ID),  
    PRIMARY KEY(Book_ID, Member_ID)  
);  
  
CREATE TABLE BOOK_SHOPPING_CART (  
    Book_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES BOOK(Book_ID),  
    Purchase_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES [SHOPPING CART](Purchase_ID),  
    Purchase_Quantity INT NOT NULL,  
    PRIMARY KEY(Book_ID, Purchase_ID)  
);  
  
CREATE TABLE BOOK_BOOKSTORE_ORDER (  
    Book_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES BOOK(Book_ID),  
    Order_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES [BOOKSTORE ORDER](Order_ID),  
    Order_Quantity INT NOT NULL,  
    PRIMARY KEY(Book_ID, Order_ID)  
);  
  
CREATE TABLE BOOK_MEMBER_ORDER (  
    Book_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES BOOK(Book_ID),  
    MemOrder_ID CHAR(8) NOT NULL FOREIGN KEY  
        REFERENCES [MEMBER ORDER](MemOrder_ID),  
    MemOrder_Quantity INT NOT NULL,  
    PRIMARY KEY(Book_ID, MemOrder_ID)  
);
```

1. Insert Statement

INSERT INTO BOOK

(Book_ID, Book_ISBN, Book_Price, Book_Section, In_stockQty, Pub_ID)

VALUES

```
( 'B0000001', '9087685942124', 20, 1, 18, 'PU000002' ),
( 'B0000002', '2437586901245', 30, 2, 35, 'PU000001' ),
( 'B0000003', '2789640145679', 19, 2, 34, 'PU000004' ),
( 'B0000004', '2365789013467', 24, 2, 35, 'PU000001' ),
( 'B0000005', '3568901348693', 10, 3, 43, 'PU000003' ),
( 'B0000006', '4678901345557', 20, 1, 53, 'PU000005' ),
( 'B0000007', '1855265474852', 23, 4, 45, 'PU000003' ),
( 'B0000008', '8541253698542', 8, 5, 64, 'PU000004' ),
( 'B0000009', '7854876521458', 9, 2, 52, 'PU000004' ),
( 'B0000010', '7854587859654', 12, 4, 14, 'PU000002' ),
( 'B0000011', '1021458796523', 13, 3, 56, 'PU000001' ),
( 'B0000012', '7698158154656', 15, 4, 38, 'PU000004' ),
( 'B0000013', '5646218418935', 14, 5, 47, 'PU000001' ),
( 'B0000014', '9451589264945', 17, 2, 44, 'PU000003' ),
( 'B0000015', '8984987112649', 18, 8, 15, 'PU000005' ),
( 'B0000016', '9848465164889', 21, 2, 19, 'PU000005' ),
( 'B0000017', '4623627864613', 22, 3, 25, 'PU000002' ),
( 'B0000018', '5636563005405', 25, 4, 42, 'PU000003' ),
( 'B0000019', '1155220011445', 27, 5, 45, 'PU000004' ),
( 'B0000020', '4932165999653', 28, 3, 17, 'PU000005' );
```

INSERT INTO UNIVERSAL_BOOK

(Book_ISBN, Book_Title, Book_Genre, Book_Edition, BookPub_Year)

VALUES

```
( '9087685942124', 'Jungle Book', 'Fantasy', 3, 2009 ),
( '2437586901245', 'Python Bible', 'Coding', 5, 2015 ),
( '2789640145679', 'Java Fundamentals', 'Coding', 6, 2003 ),
( '2365789013467', 'C++ Computer', 'Coding', 2, 2000 ),
( '3568901348693', 'Alien Story', 'Sci-Fi', 9, 1983 ),
( '4678901345557', 'Tales of Charkolol', 'Fantasy', 1, 2021 ),
( '1855265474852', 'Dora the explorer', 'Adventure', 4, 2000 ),
( '8541253698542', 'Vandyck's Lost Love', 'Romance', 2, 1999 ),
( '7854876521458', 'Colin's Recipe of Success', 'Cooking', 8, 2001 ),
( '7854587859654', 'Gaming Essentials with Desmond', 'Gaming', 7, 2019 ),
( '1021458796523', 'Phoebe Lifestyle', 'Lifestyle', 2, 2020 ),
( '7698158154656', 'Sleeping hacks by Cheong', 'Lifestyle', 6, 2008 ),
( '5646218418935', 'Drink Fiesta', 'Romance', 4, 2002 ),
( '9451589264945', 'War of the Crown', 'Fantasy', 7, 2013 ),
( '8984987112649', 'Web Development for Dummies', 'Coding', 8, 2010 ),
( '9848465164889', 'Japanese Language Fundamentals', 'Language', 2, 2018 ),
( '4623627864613', 'Time Management Guide', 'Lifestyle', 6, 2021 ),
( '5636563005405', 'Architecture ideas', 'Design', 4, 2003 ),
( '1155220011445', 'Culture Club', 'Romance', 1, 1988 ),
( '4932165999653', 'Good Food in Malaysia', 'Cooking', 5, 2021 );
```

```

INSERT INTO BOOK_AUTHOR
(Book_ID, First_Author, Second_Author, Third_Author)
VALUES
('B0000001', 'Josh', 'Jason', NULL),
('B0000002', 'Muiller', NULL, NULL),
('B0000003', 'Manhatton', 'Meledy', NULL),
('B0000004', 'Cheryl', 'Luna', 'Wee Yi'),
('B0000005', 'Cheong', NULL, NULL),
('B0000006', 'Lunaxi', 'Qiqi', NULL),
('B0000007', 'Hamilton', 'Lisa', 'Mona'),
('B0000008', 'Hanif', 'Bennett', NULL),
('B0000009', 'Josh', NULL, NULL),
('B0000010', 'Mully', 'Noelle', NULL),
('B0000011', 'Yang', 'Evelynn', 'Gwen'),
('B0000012', 'Teoh', 'Irelia', 'Benjamin'),
('B0000013', 'Kok', 'Gragas', NULL),
('B0000014', 'Vandyck', 'Shaco', NULL),
('B0000015', 'Phoebe', NULL, NULL),
('B0000016', 'Desmond', NULL, NULL),
('B0000017', 'Zhong', 'Zilean', NULL),
('B0000018', 'Xiao', 'Ekko', NULL),
('B0000019', 'Eula', 'Leona', 'Morgana'),
('B0000020', 'Paimon', 'Caitlyn', NULL);

```

```

INSERT INTO MEMBER
(Member_ID, Member_Name, Street, [Zip Code], City, State, Country, Gender, Member_DOB,
Member_Join_Date )
VALUES
('ME000001', 'Kim Jong', 'Sejongno Street', '64179', 'Jongno-gu', 'Seoul', 'South Korea',
'F', '20020410', '20201111'),
('ME000002', 'Ji Eun', 'Jalan Teknologi Park', '50300', 'Bukit Jalil', 'Kuala Lumpur',
'Malaysia', 'F', '19900530', '20201203'),
('ME000003', 'Wayne Bryan', 'Yada Street', '51109', 'Higashikata', 'Kuwana', 'Japan',
'M', '19980120', '20210117'),
('ME000004', 'Donny Tom', 'Ekkamai Rd', '33637', 'Watthana', 'Bangkok', 'Thailand', 'M',
'20051215', '20210321'),
('ME000005', 'Johnny Jhin', 'Bianmao Rd', '40812', 'ZengFuXiang', 'Chongqing', 'China',
'M', '19971119', '20210204'),
('ME000006', 'Ibrahim Mustafa', 'Qilla Kohna', '60101', 'Kuhna Qasim Bagh', 'Multan',
'Pakistan', 'M', '19890204', '20210116'),
('ME000007', 'Janna', 'High Street', '23452', 'Manchester', 'London', 'United Kingdom',
'F', '19930923', '20201116'),
('ME000008', 'Taric', 'Victoria Road', '34566', 'Newport', 'Wales', 'United Kingdom',
'M', '20021217', '20201231'),
('ME000009', 'Qiyana', 'Jalan Bukit Bintang', '52100', 'Kuala Lumpur', 'Kuala Lumpur',
'Malaysia', 'F', '20020108', '20201112'),
('ME000010', 'Rell', 'Place de Miremont', '92390', 'Villeneuve', 'Ile-de-France',
'France', 'F', '20000911', '20210526'),
('ME000011', 'Jeremy', 'Kung Yip Street', '23482', 'Kwai Tsing District', 'Hong Kong',
'Hong Kong', 'M', '19960415', '20210429'),
('ME000012', 'Tris Jenner', 'Soi Phyanark Phetburi Road', '10400', 'Phayathai', 'Bangkok',
'Thailand', 'F', '20030708', '20210517'),
('ME000013', 'Kayle Lim', 'Levent', '38473', 'Besiktas', 'Istanbul', 'Turkey', 'F',
'19990429', '20200623'),
('ME000014', 'Sivades', 'Raffles Place', '48623', 'Singapore', 'Singapore', 'Singapore',
'M', '20010420', '20210124'),

```

```
('ME000015', 'Fiora Yu', 'Proctor Place', '20222', 'Mangere Bridge', 'Manukau', 'New Zealand', 'F', '20020831', '20210228');
```

```
INSERT INTO MEMBER_CONTACT_NUMBER
(Member_ID, Personal_Phone, Work_Phone)
VALUES
('ME000001', '016-5323990', '03-85475652'),
('ME000002', NULL, '03-78527213'),
('ME000003', '017-5245948', NULL),
('ME000004', '016-8256821', NULL),
('ME000005', NULL, '03-87544964'),
('ME000006', '017-8545821', NULL),
('ME000007', '014-8547854', NULL),
('ME000008', '015-4426875', NULL),
('ME000009', NULL, '03-15875689'),
('ME000010', '017-5478214', '03-45877542'),
('ME000011', '016-4521425', NULL),
('ME000012', '013-2154845', NULL),
('ME000013', '012-4587542', NULL),
('ME000014', '018-1829754', '03-78541358'),
('ME000015', '017-5682468', NULL);
```

```
INSERT INTO PUBLISHER
(Pub_ID, Pub_Name, Pub_Email, Pub_Qty, Street, [Zip Code], City, State, Country)
VALUES
('PU000001', 'Oxford Publication', 'oxford@gmail.com', 5, 'Jalan Bukit Bintang', '52100', 'Kuala Lumpur', 'Federal Territory of Kuala Lumpur', 'Malaysia'),
('PU000002', 'Ladybird Publication', 'ladybird@gmail.com', 4, 'Rogue Street', '34567', 'Althorne', 'Essex', 'United Kingdom'),
('PU000003', 'Miracle Publication', 'miracle@outlook.com', 3, 'North Bound', '32456', 'Thorone', 'Texas', 'United States'),
('PU000004', 'Miyuki', 'miyukipub@gmail.com', 5, 'Takeshita Street', '87895', 'Harajuku', 'Tokyo', 'Japan'),
('PU000005', 'Gemilang Publishers', 'gemilang@yahoo.com', 7, 'Jalan Helang Hindik', '52100', 'Kuala Lumpur', 'Federal Territory of Kuala Lumpur', 'Malaysia'),
('PU000006', 'Cambridge Publication', 'cambridge@gmail.com', 8, 'Elliott Street', '20419', 'Washington', 'District of Columbia', 'United States'),
('PU000007', 'Popular Publication', 'popular@outlook.com', 3, 'Locust Court', '90808', 'Long Beach', 'California', 'United States'),
('PU000008', 'National Geographic Publication', 'nationalgeo@yahoo.com', 4, 'Fasanenstrasse', '22587', 'Hamburg Osdorf', 'Hamburg', 'Germany'),
('PU000009', 'Oxbridge Publications', 'oxbridge@gmail.com', 2, 'Taman Baru', '68000', 'Sitiawan', 'Perak', 'Malaysia'),
('PU000010', 'IEEE Publications', 'ieee@yahoo..com', 7, 'Jalan Jelutong', '70753', 'Dungun', 'Terengganu', 'Malaysia');
```

```
INSERT INTO PUB_PHONE
(Pub_ID, Headquarter_phoneNo, Branch_phoneNo)
VALUES
('PU000001', '03-23463746', NULL),
('PU000002', '02-48562495', '03-34857462'),
('PU000003', '04-58694274', NULL),
('PU000004', '06-69774526', '03-58674732'),
('PU000005', '09-05674743', NULL);
```

```
( 'PU000006', '03-56855975', '03-85495481'),
( 'PU000007', '05-15844582', '03-78464845'),
( 'PU000008', '09-85888554', NULL),
( 'PU000009', '03-55198268', '05-54857895'),
( 'PU000010', '04-58912654', NULL);
```

INSERT INTO [BOOKSTORE ORDER]

(Order_ID, Order_Date, Order_Status, Pub_ID)

VALUES

```
( 'OR000001', '20200312', 'Completed', 'PU000001'),
( 'OR000002', '20200703', 'Completed', 'PU000002'),
( 'OR000003', '20200920', 'Completed', 'PU000003'),
( 'OR000004', '20201231', 'Completed', 'PU000004'),
( 'OR000005', '20210104', 'Completed', 'PU000005'),
( 'OR000006', '20210115', 'Completed', 'PU000001'),
( 'OR000007', '20210121', 'Completed', 'PU000001'),
( 'OR000008', '20210202', 'Completed', 'PU000002'),
( 'OR000009', '20210216', 'Pending', 'PU000003'),
( 'OR000010', '20210304', 'Pending', 'PU000004'),
( 'OR000011', '20210305', 'Pending', 'PU000005'),
( 'OR000012', '20210408', 'Pending', 'PU000003'),
( 'OR000013', '20210421', 'Pending', 'PU000002'),
( 'OR000014', '20210503', 'Pending', 'PU000001'),
( 'OR000015', '20210521', 'Pending', 'PU000003');
```

INSERT INTO [SHOPPING CART]

(Purchase_ID, Purchase_Date, TotalPurchase_Price, Member_ID)

VALUES

```
( 'PA000001', '20210424', 133, 'ME000001'),
( 'PA000002', '20210424', 119, 'ME000002'),
( 'PA000003', '20210424', 153, 'ME000003'),
( 'PA000004', '20210424', 171, 'ME000004'),
( 'PA000005', '20210426', 160, 'ME000005'),
( 'PA000006', '20210501', 72, 'ME000006'),
( 'PA000007', '20210505', 97, 'ME000007'),
( 'PA000008', '20210515', 130, 'ME000008'),
( 'PA000009', '20210518', 115, 'ME000009'),
( 'PA000010', '20210523', 54, 'ME000010'),
( 'PA000011', '20210528', 49, 'ME000011'),
( 'PA000012', '20210603', 34, 'ME000012'),
( 'PA000013', '20210606', 73, 'ME000013'),
( 'PA000014', '20210610', 122, 'ME000014'),
( 'PA000015', '20210616', 76, 'ME000015');
```

INSERT INTO [MEMBER ORDER]

(MemOrder_ID, MemOrder_Date, Payment_Status, Delivery_Status, MemOrder_Price,
EstOrderArri_date, Purchase_ID, Member_ID)

VALUES

```
( 'MO000001', '20210420', 'Pending', NULL, 84, NULL, 'PA000001', 'ME000001'),
( 'MO000002', '20210424', 'Completed', 'DELIVERED', 86, NULL, 'PA000005', 'ME000006'),
( 'MO000003', '20210423', 'Completed', 'DELIVERED', 40, NULL, 'PA000002', 'ME000002'),
( 'MO000004', '20210425', 'Pending', NULL, 90, NULL, 'PA000003', 'ME000003'),
( 'MO000005', '20210423', 'Pending', NULL, 110, NULL, 'PA000003', 'ME000003'),
( 'MO000006', '20210425', 'Completed', 'DELIVERED', 29, NULL, 'PA000001', 'ME000001'),
```



```
( 'MO000007', '20210427', 'Completed', 'DELIVERING', 130, '20210502', 'PA000004',
'ME000005' ),
( 'MO000008', '20210428', 'Completed', 'DELIVERED', 79, NULL, 'PA000002', 'ME000002' ),
( 'MO000009', '20210428', 'Pending', NULL, 110, NULL, 'PA000003', 'ME000003' ),
( 'MO000010', '20210429', 'Completed', 'DELIVERING', 19, '20210504', 'PA000005',
'ME000006' ),
( 'MO000011', '20210520', 'Completed', 'DELIVERED', 36, NULL, 'PA000009', 'ME000009' ),
( 'MO000012', '20210525', 'Completed', 'DELIVERED', 53, NULL, 'PA000010', 'ME000010' ),
( 'MO000013', '20210530', 'Completed', 'DELIVERED', 34, NULL, 'PA000007', 'ME000007' ),
( 'MO000014', '20210601', 'Completed', 'DELIVERED', 51, NULL, 'PA000009', 'ME000009' ),
( 'MO000015', '20210605', 'Completed', 'DELIVERING', 24, '20210609', 'PA000011',
'ME000011' ),
( 'MO000016', '20210612', 'Pending', NULL, 64, NULL, 'PA000014', 'ME000014' ),
( 'MO000017', '20210703', 'Pending', NULL, 27, NULL, 'PA000006', 'ME000006' ),
( 'MO000018', '20210707', 'Completed', 'DELIVERING', 30, '20210710', 'PA000008',
'ME000008' ),
( 'MO000019', '20210709', 'Completed', 'DELIVERING', 25, '20210711', 'PA000011',
'ME000011' ),
( 'MO000020', '20210710', 'Pending', NULL, 72, NULL, 'PA000006', 'ME000006' );
```

```
INSERT INTO [MEMBER FEEDBACK]
```

```
(Feedback_ID, Rating, Remarks, Member_ID, Book_ID)
```

```
VALUES
```

```
( 'FI000001', 9, 'Very fun plot and story', 'ME000002', 'B0000006' ),
( 'FI000002', 8, 'Nice story', 'ME000003', 'B0000001' ),
( 'FI000003', 6, NULL, 'ME000003', 'B0000003' ),
( 'FI000004', 9, 'Very informative book', 'ME000006', 'B0000003' ),
( 'FI000005', 4, 'Very hard to understand', 'ME000001', 'B0000004' ),
( 'FI000006', 8, 'This book is simple and understandable', 'ME000005', 'B0000002' ),
( 'FI000007', 7, 'Interesting story', 'ME000006', 'B0000001' ),
( 'FI000008', 5, 'Information not sufficient', 'ME000001', 'B0000003' ),
( 'FI000009', 8, 'Very touching love story', 'ME000009', 'B0000013' ),
( 'FI000010', 9, 'Good time management tips', 'ME000009', 'B0000017' ),
( 'FI000011', 8, NULL, 'ME000007', 'B0000011' ),
( 'FI000012', 9, 'Had great laughs', 'ME000009', 'B0000006' ),
( 'FI000013', 7, 'Decent recipes', 'ME000009', 'B0000009' ),
( 'FI000014', 6, 'The story feels rushed', 'ME000010', 'B0000005' ),
( 'FI000015', 8, 'A childhood classic', 'ME000010', 'B0000007' ),
( 'FI000016', 8, 'Easy to understand', 'ME000010', 'B0000016' ),
( 'FI000017', 7, NULL, 'ME000007', 'B0000007' ),
( 'FI000018', 6, NULL, 'ME000009', 'B0000017' ),
( 'FI000019', 4, 'Dull storyline', 'ME000002', 'B0000005' ),
( 'FI000020', 8, 'Cute story', 'ME000002', 'B0000001' );
```

```

INSERT INTO BOOK_SHOPPING_CART
(Book_ID, Purchase_ID, Purchase_Quantity)
VALUES
('B0000003', 'PA000001', 1),
('B0000002', 'PA000001', 2),
('B0000001', 'PA000001', 1),
('B0000004', 'PA000001', 1),
('B0000005', 'PA000001', 1),
('B0000006', 'PA000002', 2),
('B0000002', 'PA000002', 1),
('B0000001', 'PA000002', 1),
('B0000003', 'PA000002', 1),
('B0000005', 'PA000002', 1),
('B0000001', 'PA000003', 1),
('B0000003', 'PA000003', 1),
('B0000002', 'PA000003', 3),
('B0000004', 'PA000003', 1),
('B0000006', 'PA000004', 1),
('B0000003', 'PA000004', 3),
('B0000002', 'PA000004', 1),
('B0000004', 'PA000004', 1),
('B0000001', 'PA000004', 2),
('B0000002', 'PA000005', 1),
('B0000004', 'PA000005', 1),
('B0000003', 'PA000005', 4),
('B0000001', 'PA000005', 1),
('B0000005', 'PA000005', 1),
('B0000010', 'PA000006', 1),
('B0000012', 'PA000006', 1),
('B0000015', 'PA000006', 1),
('B0000019', 'PA000006', 1),
('B0000020', 'PA000007', 1),
('B0000011', 'PA000007', 1),
('B0000007', 'PA000007', 1),
('B0000009', 'PA000007', 1),
('B0000004', 'PA000007', 1),
('B0000008', 'PA000008', 1),
('B0000006', 'PA000008', 1),
('B0000001', 'PA000008', 1),
('B0000018', 'PA000008', 1),
('B0000019', 'PA000008', 1),
('B0000002', 'PA000008', 1),
('B0000003', 'PA000009', 1),
('B0000007', 'PA000009', 1),
('B0000009', 'PA000009', 1),
('B0000013', 'PA000009', 1),
('B0000017', 'PA000009', 1),
('B0000008', 'PA000009', 1),
('B0000006', 'PA000009', 1),
('B0000005', 'PA000010', 1),
('B0000007', 'PA000010', 1),
('B0000016', 'PA000010', 1),
('B0000018', 'PA000011', 1),
('B0000004', 'PA000011', 1),
('B0000009', 'PA000012', 1),
('B0000010', 'PA000012', 1),
('B0000011', 'PA000012', 1),
('B0000001', 'PA000013', 1),

```

```

('B0000004', 'PA000013', 1),
('B0000006', 'PA000013', 1),
('B0000009', 'PA000013', 1),
('B0000015', 'PA000014', 1),
('B0000019', 'PA000014', 1),
('B0000011', 'PA000014', 1),
('B0000013', 'PA000014', 1),
('B0000017', 'PA000014', 1),
('B0000020', 'PA000014', 1),
('B0000008', 'PA000015', 1),
('B0000009', 'PA000015', 1),
('B0000004', 'PA000015', 1),
('B0000014', 'PA000015', 1),
('B0000015', 'PA000015', 1);

```

```
INSERT INTO BOOK_BOOKSTORE_ORDER
(Book_ID, Order_ID, Order_Quantity)
VALUES
('B0000001', 'OR000002', 15),
('B0000002', 'OR000001', 50),
('B0000003', 'OR000004', 30),
('B0000004', 'OR000006', 45),
('B0000005', 'OR000003', 55),
('B0000006', 'OR000005', 16),
('B0000007', 'OR000009', 90),
('B0000008', 'OR000010', 25),
('B0000009', 'OR000011', 15),
('B0000010', 'OR000008', 75),
('B0000011', 'OR000007', 34),
('B0000012', 'OR000004', 30),
('B0000013', 'OR000014', 24),
('B0000014', 'OR000012', 20),
('B0000015', 'OR000013', 15),
('B0000016', 'OR000005', 30),
('B0000017', 'OR000002', 15),
('B0000018', 'OR000015', 31),
('B0000019', 'OR000010', 31),
('B0000020', 'OR000005', 30);
```

```
INSERT INTO BOOK_MEMBER
(Member_ID,Book_ID)
```

```
VALUES
```

```
('ME000001', 'B0000001'),
('ME000001', 'B0000002'),
('ME000001', 'B0000003'),
('ME000001', 'B0000004'),
('ME000001', 'B0000005'),
('ME000001', 'B0000006'),
('ME000001', 'B0000007'),
('ME000001', 'B0000008'),
('ME000001', 'B0000009'),
('ME000001', 'B0000010'),
('ME000001', 'B0000011'),
('ME000001', 'B0000012'),
('ME000001', 'B0000013'),
('ME000001', 'B0000014'),
('ME000001', 'B0000015'),
('ME000001', 'B0000016'),
('ME000001', 'B0000017'),
('ME000001', 'B0000018'),
('ME000001', 'B0000019'),
('ME000001', 'B0000020'),
('ME000002', 'B0000001'),
('ME000002', 'B0000002'),
('ME000002', 'B0000003'),
('ME000002', 'B0000004'),
('ME000002', 'B0000005'),
('ME000002', 'B0000006'),
('ME000002', 'B0000007'),
('ME000002', 'B0000008'),
('ME000002', 'B0000009'),
('ME000002', 'B0000010'),
('ME000002', 'B0000011'),
('ME000002', 'B0000012'),
('ME000002', 'B0000013'),
('ME000002', 'B0000014'),
('ME000002', 'B0000015'),
('ME000002', 'B0000016'),
('ME000002', 'B0000017'),
('ME000002', 'B0000018'),
('ME000002', 'B0000019'),
('ME000002', 'B0000020'),
('ME000003', 'B0000001'),
('ME000003', 'B0000002'),
('ME000003', 'B0000003'),
('ME000003', 'B0000004'),
('ME000003', 'B0000005'),
('ME000003', 'B0000006'),
('ME000003', 'B0000007'),
('ME000003', 'B0000008'),
('ME000003', 'B0000009'),
('ME000003', 'B0000010'),
('ME000003', 'B0000011'),
('ME000003', 'B0000012'),
('ME000003', 'B0000013'),
('ME000003', 'B0000014'),
('ME000003', 'B0000015'),
```

```
('ME000003', 'B0000016'),
('ME000003', 'B0000017'),
('ME000003', 'B0000018'),
('ME000003', 'B0000019'),
('ME000003', 'B0000020'),
('ME000004', 'B0000001'),
('ME000004', 'B0000002'),
('ME000004', 'B0000003'),
('ME000004', 'B0000004'),
('ME000004', 'B0000005'),
('ME000004', 'B0000006'),
('ME000004', 'B0000007'),
('ME000004', 'B0000008'),
('ME000004', 'B0000009'),
('ME000004', 'B0000010'),
('ME000004', 'B0000011'),
('ME000004', 'B0000012'),
('ME000004', 'B0000013'),
('ME000004', 'B0000014'),
('ME000004', 'B0000015'),
('ME000004', 'B0000016'),
('ME000004', 'B0000017'),
('ME000004', 'B0000018'),
('ME000004', 'B0000019'),
('ME000004', 'B0000020'),
('ME000005', 'B0000001'),
('ME000005', 'B0000002'),
('ME000005', 'B0000003'),
('ME000005', 'B0000004'),
('ME000005', 'B0000005'),
('ME000005', 'B0000006'),
('ME000005', 'B0000007'),
('ME000005', 'B0000008'),
('ME000005', 'B0000009'),
('ME000005', 'B0000010'),
('ME000005', 'B0000011'),
('ME000005', 'B0000012'),
('ME000005', 'B0000013'),
('ME000005', 'B0000014'),
('ME000005', 'B0000015'),
('ME000005', 'B0000016'),
('ME000005', 'B0000017'),
('ME000005', 'B0000018'),
('ME000005', 'B0000019'),
('ME000005', 'B0000020'),
('ME000006', 'B0000001'),
('ME000006', 'B0000002'),
('ME000006', 'B0000003'),
('ME000006', 'B0000004'),
('ME000006', 'B0000005'),
('ME000006', 'B0000006'),
('ME000006', 'B0000007'),
('ME000006', 'B0000008'),
('ME000006', 'B0000009'),
('ME000006', 'B0000010'),
('ME000006', 'B0000011'),
('ME000006', 'B0000012'),
('ME000006', 'B0000013'),
```

33

```

('ME000012', 'B0000010'),
('ME000012', 'B0000011'),
('ME000012', 'B0000012'),
('ME000012', 'B0000013'),
('ME000012', 'B0000014'),
('ME000012', 'B0000015'),
('ME000012', 'B0000016'),
('ME000012', 'B0000017'),
('ME000012', 'B0000018'),
('ME000012', 'B0000019'),
('ME000012', 'B0000020'),
('ME000013', 'B0000001'),
('ME000013', 'B0000002'),
('ME000013', 'B0000003'),
('ME000013', 'B0000004'),
('ME000013', 'B0000005'),
('ME000013', 'B0000006'),
('ME000013', 'B0000007'),
('ME000013', 'B0000008'),
('ME000013', 'B0000009'),
('ME000013', 'B0000010'),
('ME000013', 'B0000011'),
('ME000013', 'B0000012'),
('ME000013', 'B0000013'),
('ME000013', 'B0000014'),
('ME000013', 'B0000015'),
('ME000013', 'B0000016'),
('ME000013', 'B0000017'),
('ME000013', 'B0000018'),
('ME000013', 'B0000019'),
('ME000013', 'B0000020'),
('ME000014', 'B0000001'),
('ME000014', 'B0000002'),
('ME000014', 'B0000003'),
('ME000014', 'B0000004'),
('ME000014', 'B0000005'),
('ME000014', 'B0000006'),
('ME000014', 'B0000007'),
('ME000014', 'B0000008'),
('ME000014', 'B0000009'),
('ME000014', 'B0000010'),
('ME000014', 'B0000011'),
('ME000014', 'B0000012'),
('ME000014', 'B0000013'),
('ME000014', 'B0000014'),
('ME000014', 'B0000015'),
('ME000014', 'B0000016'),
('ME000014', 'B0000017'),
('ME000014', 'B0000018'),
('ME000014', 'B0000019'),
('ME000014', 'B0000020'),
('ME000015', 'B0000001'),
('ME000015', 'B0000002'),
('ME000015', 'B0000003'),
('ME000015', 'B0000004'),
('ME000015', 'B0000005'),
('ME000015', 'B0000006'),
('ME000015', 'B0000007'),
('ME000015', 'B0000008'),
('ME000015', 'B0000009'),
('ME000015', 'B0000010'),
('ME000015', 'B0000011'),
('ME000015', 'B0000012'),
('ME000015', 'B0000013'),
('ME000015', 'B0000014'),
('ME000015', 'B0000015'),
('ME000015', 'B0000016'),
('ME000015', 'B0000017'),
('ME000015', 'B0000018'),
('ME000015', 'B0000019'),
('ME000015', 'B0000020');

```

```
INSERT INTO BOOK_MEMBER_ORDER
(MemOrder_ID, Book_ID, MemOrder_Quantity)
VALUES
('MO000001', 'B0000004', 1),
('MO000001', 'B0000002', 2),
('MO000002', 'B0000005', 1),
('MO000002', 'B0000003', 4),
('MO000003', 'B0000006', 2),
('MO000004', 'B0000002', 3),
('MO000005', 'B0000001', 1),
('MO000005', 'B0000002', 3),
('MO000006', 'B0000003', 1),
('MO000006', 'B0000005', 1),
('MO000007', 'B0000001', 2),
('MO000007', 'B0000003', 3),
('MO000008', 'B0000001', 1),
('MO000008', 'B0000002', 1),
('MO000008', 'B0000003', 1),
('MO000008', 'B0000005', 1),
('MO000009', 'B0000001', 1),
('MO000009', 'B0000002', 3),
('MO000010', 'B0000003', 1),
('MO000011', 'B0000013', 1),
('MO000011', 'B0000017', 1),
('MO000012', 'B0000005', 1),
('MO000012', 'B0000006', 1),
('MO000012', 'B0000007', 1),
('MO000013', 'B0000007', 1),
('MO000013', 'B0000011', 1),
('MO000014', 'B0000003', 1),
('MO000014', 'B0000007', 1),
('MO000014', 'B0000009', 1),
('MO000015', 'B0000004', 1),
('MO000016', 'B0000013', 1),
('MO000016', 'B0000017', 1),
('MO000016', 'B0000020', 1),
('MO000017', 'B0000019', 1),
('MO000018', 'B0000002', 1),
('MO000019', 'B0000018', 1),
('MO000020', 'B0000010', 1),
('MO000020', 'B0000012', 1),
('MO000020', 'B0000015', 1),
('MO000020', 'B0000019', 1);
```

2. Data in Tables

1. BOOK Table

	Book_ID	Book_ISBN	Book_Price	Book_Section	In_stockQty	Pub_ID
1	BO000001	9087685942124	20.00	1	18	PU000002
2	BO000002	2437586901245	30.00	2	35	PU000001
3	BO000003	2789640145679	19.00	2	34	PU000004
4	BO000004	2365789013467	24.00	2	35	PU000001
5	BO000005	3568901348693	10.00	3	43	PU000003
6	BO000006	4678901345557	20.00	1	53	PU000005
7	BO000007	1855265474852	23.00	4	45	PU000003
8	BO000008	8541253698542	8.00	5	64	PU000004
9	BO000009	7854876521458	9.00	2	52	PU000004
10	BO000010	7854587859654	12.00	4	14	PU000002
11	BO000011	1021458796523	13.00	3	56	PU000001
12	BO000012	7698158154656	15.00	4	38	PU000004
13	BO000013	5646218418935	14.00	5	47	PU000001
14	BO000014	9451589264945	17.00	2	44	PU000003
15	BO000015	8984987112649	18.00	8	15	PU000005
16	BO000016	9848465164889	21.00	2	19	PU000005
17	BO000017	4623627864613	22.00	3	25	PU000002
18	BO000018	5636563005405	25.00	4	42	PU000003
19	BO000019	1155220011445	27.00	5	45	PU000004
20	BO000020	4932165999653	28.00	3	17	PU000005

Table II.1: Book table

2. UNIVERSAL_BOOK Table

	Book_ISBN	Book_Title	Book_Genre	Book_Edition	BookPub_Year
1	1021458796523	Phoebe Lifestyle	Lifestyle	2	2020
2	1155220011445	Culture Club	Romance	1	1988
3	1855265474852	Dora the explorer	Adventure	4	2000
4	2365789013467	C++ Computer	Coding	2	2000
5	2437586901245	Python Bible	Coding	5	2015
6	2789640145679	Java Fundamentals	Coding	6	2003
7	3568901348693	Alien Story	Sci-Fi	9	1983
8	4623627864613	Time Management Guide	Lifestyle	6	2021
9	4678901345557	Tales of Charkolol	Fantasy	1	2021
10	4932165999653	Good Food in Malaysia	Cooking	5	2021
11	5636563005405	Architecture ideas	Design	4	2003
12	5646218418935	Drink Fiesta	Romance	4	2002
13	7698158154656	Sleeping hacks by Cheong	Lifestyle	6	2008
14	7854587859654	Gaming Essentials with Desmond	Gaming	7	2019
15	7854876521458	Colin's Recipe of Success	Cooking	8	2001
16	8541253698542	Vandyck's Lost Love	Romance	2	1999
17	8984987112649	Web Development for Dummies	Coding	8	2010
18	9087685942124	Jungle Book	Fantasy	3	2009
19	9451589264945	War of the Crown	Fantasy	7	2013
20	9848465164889	Japanese Language Fundamen...	Language	2	2018

Table II.2: Universal book table

3. BOOK_AUTHOR Table

	Book_ID	First_Author	Second_Author	Third_Author
1	BO000001	Josh	Jason	NULL
2	BO000002	Muiller	NULL	NULL
3	BO000003	Manhatton	Meledy	NULL
4	BO000004	Cheryl	Luna	Wee Yi
5	BO000005	Cheong	NULL	NULL
6	BO000006	Lunaxi	Qiqi	NULL
7	BO000007	Hamilton	Lisa	Mona
8	BO000008	Hanif	Bennett	NULL
9	BO000009	Josh	NULL	NULL
10	BO000010	Mully	Noelle	NULL
11	BO000011	Yang	Evelynn	Gwen
12	BO000012	Teoh	Irelia	Benjamin
13	BO000013	Kok	Gragas	NULL
14	BO000014	Vandyck	Shaco	NULL
15	BO000015	Phoebe	NULL	NULL
16	BO000016	Desmond	NULL	NULL
17	BO000017	Zhong	Zilean	NULL
18	BO000018	Xiao	Ekko	NULL
19	BO000019	Eula	Leona	Morgana
20	BO000020	Paimon	Caitlyn	NULL

Table II.3: Book author table

4. MEMBER Table

	Member_ID	Member_Name	Gender	Member_DOB	Member_Join_Date	Street	Zip Code	City	State	Country
1	ME000001	Kim Jong	F	2002-04-10	2020-11-11	Sejongno Street	64179	Jongno-gu	Seoul	South Korea
2	ME000002	Ji Eun	F	1990-05-30	2020-12-03	Jalan Teknologi Park	50300	Bukit Jalil	Kuala Lumpur	Malaysia
3	ME000003	Wayne Bryan	M	1998-01-20	2021-01-17	Yada Street	51109	Higashikata	Kuwana	Japan
4	ME000004	Donny Tom	M	2005-12-15	2021-03-21	Ekkamai Rd	33637	Watthana	Bangkok	Thailand
5	ME000005	Johnny Jhin	M	1997-11-19	2021-02-04	Bianmao Rd	40812	ZengFuXiang	Chongqing	China
6	ME000006	Ibrahim Mustafa	M	1989-02-04	2021-01-16	Qilla Kohna	60101	Kuhna Qasim Bagh	Multan	Pakistan
7	ME000007	Janna	F	1993-09-23	2020-11-16	High Street	23452	Manchester	London	United Kingdom
8	ME000008	Taric	M	2002-12-17	2020-12-31	Victoria Road	34566	Newport	Wales	United Kingdom
9	ME000009	Qiyana	F	2002-01-08	2020-11-12	Jalan Bukit Bintang	52100	Kuala Lumpur	Kuala Lumpur	Malaysia
10	ME000010	Rell	F	2000-09-11	2021-05-26	Place de Miremont	92390	Villeneuve	Ile-de-France	France
11	ME000011	Jeremy	M	1996-04-15	2021-04-29	Kung Yip Street	23482	Kwai Tsing District	Hong Kong	Hong Kong
12	ME000012	Tris Jenner	F	2003-07-08	2021-05-17	Soi Phyanark Phetburi Road	10400	Phayathai	Bangkok	Thailand
13	ME000013	Kayle Lim	F	1999-04-29	2020-06-23	Levent	38473	Besiktas	Istanbul	Turkey
14	ME000014	Sivades	M	2001-04-20	2021-01-24	Raffles Place	48623	Singapore	Singapore	Singapore
15	ME000015	Fiora Yu	F	2002-08-31	2021-02-28	Proctor Place	20222	Mangere Bridge	Manukau	New Zealand

Table II.4: Member table

5. MEMBER_CONTACT_NUMBER Table

	Member_ID	Personal_Phone	Work_Phone
1	ME000001	016-5323990	03-85475652
2	ME000002	NULL	03-78527213
3	ME000003	017-5245948	NULL
4	ME000004	016-8256821	NULL
5	ME000005	NULL	03-87544964
6	ME000006	017-8545821	NULL
7	ME000007	014-8547854	NULL
8	ME000008	015-4426875	NULL
9	ME000009	NULL	03-15875689
10	ME000010	017-5478214	03-45877542
11	ME000011	016-4521425	NULL
12	ME000012	013-2154845	NULL
13	ME000013	012-4587542	NULL
14	ME000014	018-1829754	03-78541358
15	ME000015	017-5682468	NULL

Table II.5: Member contact number table

6. PUBLISHER Table

	Pub_ID	Pub_Name	Pub_Email	Pub_Qty	Street	Zip Code	City	State	Country
1	PU000001	Oxford Publication	oxford@gmail.com	5	Jalan Bukit Bintang	52100	Kuala Lumpur	Federal Territory of Kuala Lumpur	Malaysia
2	PU000002	Ladybird Publication	ladybird@gmail.com	4	Rogue Street	34567	Althorne	Essex	United Kingdom
3	PU000003	Miracle Publication	miracle@outlook.com	3	North Bound	32456	Thorone	Texas	United States
4	PU000004	Miyuki	miyukipub@gmail.com	5	Takeshita Street	87895	Harajuku	Tokyo	Japan
5	PU000005	Gemilang Publishers	gemilang@yahoo.com	7	Jalan Helang Hindik	52100	Kuala Lumpur	Federal Territory of Kuala Lumpur	Malaysia
6	PU000006	Cambridge Publication	cambridge@gmail.com	8	Elliott Street	20419	Washington	District of Columbia	United States
7	PU000007	Popular Publication	popular@outlook.com	3	Locust Court	90808	Long Beach	California	United States
8	PU000008	National Geographic Publication	nationalgeo@yahoo.com	4	Fasanenstrasse	22587	Hamburg Osdorf	Hamburg	Germany
9	PU000009	Oxbridge Publications	oxbridge@gmail.com	2	Taman Baru	68000	Sitiawan	Perak	Malaysia
10	PU000010	IEEE Publications	ieee@yahoo.com	7	Jalan Jelutong	70753	Dungun	Terengganu	Malaysia

Table II.6: Publisher table

7. PUB_PHONE Table

	Pub_ID	Headquarter_phoneNo	Branch_phoneNo
1	PU000001	03-23463746	NULL
2	PU000002	02-48562495	03-34857462
3	PU000003	04-58694274	NULL
4	PU000004	06-69774526	03-58674732
5	PU000005	09-05674743	NULL
6	PU000006	03-56855975	03-85495481
7	PU000007	05-15844582	03-78464845
8	PU000008	09-85888554	NULL
9	PU000009	03-55198268	05-54857895
10	PU000010	04-58912654	NULL

Table II.7: Publisher phone number table

8. BOOKSTORE ORDER Table

	Order_ID	Order_Date	Order_Status	Pub_ID
1	OR000001	2020-03-12	Completed	PU000001
2	OR000002	2020-07-03	Completed	PU000002
3	OR000003	2020-09-20	Completed	PU000003
4	OR000004	2020-12-31	Completed	PU000004
5	OR000005	2021-01-04	Completed	PU000005
6	OR000006	2021-01-15	Completed	PU000001
7	OR000007	2021-01-21	Completed	PU000001
8	OR000008	2021-02-02	Completed	PU000002
9	OR000009	2021-02-16	Pending	PU000003
10	OR000010	2021-03-04	Pending	PU000004
11	OR000011	2021-03-05	Pending	PU000005
12	OR000012	2021-04-08	Pending	PU000003
13	OR000013	2021-04-21	Pending	PU000002
14	OR000014	2021-05-03	Pending	PU000001
15	OR000015	2021-05-21	Pending	PU000003

Table II.8: Bookstore order table

9. SHOPPING CART Table

	Purchase_ID	Purchase_Date	TotalPurchase_Price	Member_ID
1	PA000001	2021-04-24	133.00	ME000001
2	PA000002	2021-04-24	119.00	ME000002
3	PA000003	2021-04-24	153.00	ME000003
4	PA000004	2021-04-24	171.00	ME000004
5	PA000005	2021-04-26	160.00	ME000005
6	PA000006	2021-05-01	72.00	ME000006
7	PA000007	2021-05-05	97.00	ME000007
8	PA000008	2021-05-15	130.00	ME000008
9	PA000009	2021-05-18	115.00	ME000009
10	PA000010	2021-05-23	54.00	ME000010
11	PA000011	2021-05-28	49.00	ME000011
12	PA000012	2021-06-03	34.00	ME000012
13	PA000013	2021-06-06	73.00	ME000013
14	PA000014	2021-06-10	122.00	ME000014
15	PA000015	2021-06-16	76.00	ME000015

Table II.9: Shopping cart table

10. MEMBER ORDER Table

	MemOrder_ID	MemOrder_Date	Payment_Status	Delivery_Status	MemOrder_Price	EstOrderArri_date	Purchase_ID	Member_ID
1	MO000001	2021-04-20	Pending	NULL	84.00	NULL	PA000001	ME000001
2	MO000002	2021-04-24	Completed	DELIVERED	86.00	NULL	PA000005	ME000006
3	MO000003	2021-04-23	Completed	DELIVERED	40.00	NULL	PA000002	ME000002
4	MO000004	2021-04-25	Pending	NULL	90.00	NULL	PA000003	ME000003
5	MO000005	2021-04-23	Pending	NULL	110.00	NULL	PA000003	ME000003
6	MO000006	2021-04-25	Completed	DELIVERED	29.00	NULL	PA000001	ME000001
7	MO000007	2021-04-27	Completed	DELIVERING	130.00	2021-05-02	PA000004	ME000005
8	MO000008	2021-04-28	Completed	DELIVERED	79.00	NULL	PA000002	ME000002
9	MO000009	2021-04-28	Pending	NULL	110.00	NULL	PA000003	ME000003
10	MO000010	2021-04-29	Completed	DELIVERING	19.00	2021-05-04	PA000005	ME000006
11	MO000011	2021-05-20	Completed	DELIVERED	36.00	NULL	PA000009	ME000009
12	MO000012	2021-05-25	Completed	DELIVERED	53.00	NULL	PA000010	ME000010
13	MO000013	2021-05-30	Completed	DELIVERED	34.00	NULL	PA000007	ME000007
14	MO000014	2021-06-01	Completed	DELIVERED	51.00	NULL	PA000009	ME000009
15	MO000015	2021-06-05	Completed	DELIVERING	24.00	2021-06-09	PA000011	ME000011
16	MO000016	2021-06-12	Pending	NULL	64.00	NULL	PA000014	ME000014
17	MO000017	2021-07-03	Pending	NULL	27.00	NULL	PA000006	ME000006
18	MO000018	2021-07-07	Completed	DELIVERING	30.00	2021-07-10	PA000008	ME000008
19	MO000019	2021-07-09	Completed	DELIVERING	25.00	2021-07-11	PA000011	ME000011
20	MO000020	2021-07-10	Pending	NULL	72.00	NULL	PA000006	ME000006

Table II.10: Member order table

11. MEMBER FEEDBACK Table

	Feedback_ID	Rating	Remarks	Member_ID	Book_ID
1	FI000001	9	Very fun plot and story	ME000002	BO000006
2	FI000002	8	Nice story	ME000003	BO000001
3	FI000003	6	NULL	ME000003	BO000003
4	FI000004	9	Very informative book	ME000006	BO000003
5	FI000005	4	Very hard to understand	ME000001	BO000004
6	FI000006	8	This book is simple and understandable	ME000005	BO000002
7	FI000007	7	Interesting story	ME000006	BO000001
8	FI000008	5	Information not sufficient	ME000001	BO000003
9	FI000009	8	Very touching love story	ME000009	BO000013
10	FI000010	9	Good time management tips	ME000009	BO000017
11	FI000011	8	NULL	ME000007	BO000011
12	FI000012	9	Had great laughs	ME000009	BO000006
13	FI000013	7	Decent recipes	ME000009	BO000009
14	FI000014	6	The story feels rushed	ME000010	BO000005
15	FI000015	8	A childhood classic	ME000010	BO000007
16	FI000016	8	Easy to understand	ME000010	BO000016
17	FI000017	7	NULL	ME000007	BO000007
18	FI000018	6	NULL	ME000009	BO000017
19	FI000019	4	Dull storyline	ME000002	BO000005
20	FI000020	8	Cute story	ME000002	BO000001

Table II.11: Member feedback table

12. BOOK_SHOPPING_CART Table

	Book_ID	Purchase_ID	Purchase_Quantity
1	BO000003	PA000001	1
2	BO000002	PA000001	2
3	BO000001	PA000001	1
4	BO000004	PA000001	1
5	BO000005	PA000001	1
6	BO000006	PA000002	2
7	BO000002	PA000002	1
8	BO000001	PA000002	1
9	BO000003	PA000002	1
10	BO000005	PA000002	1
11	BO000001	PA000003	1
12	BO000003	PA000003	1
13	BO000002	PA000003	3
14	BO000004	PA000003	1
15	BO000006	PA000004	1
16	BO000003	PA000004	3
17	BO000002	PA000004	1
18	BO000004	PA000004	1
19	BO000001	PA000004	2
20	BO000002	PA000005	1
21	BO000004	PA000005	1
22	BO000003	PA000005	4
23	BO000001	PA000005	1
24	BO000005	PA000005	1
25	BO000010	PA000006	1
26	BO000012	PA000006	1
27	BO000015	PA000006	1
28	BO000019	PA000006	1
29	BO000020	PA000007	1
30	BO000011	PA000007	1
31	BO000007	PA000007	1
32	BO000009	PA000007	1
33	BO000004	PA000007	1
34	BO000008	PA000008	1
35	BO000006	PA000008	1
36	BO000001	PA000008	1
37	BO000018	PA000008	1
38	BO000019	PA000008	1
39	BO000002	PA000008	1
40	BO000003	PA000009	1
41	BO000007	PA000009	1
42	BO000009	PA000009	1
43	BO000013	PA000009	1
44	BO000017	PA000009	1
45	BO000008	PA000009	1
46	BO000006	PA000009	1
47	BO000005	PA000010	1
48	BO000007	PA000010	1
49	BO000016	PA000010	1
50	BO000018	PA000011	1
51	BO000004	PA000011	1
52	BO000009	PA000012	1
53	BO000010	PA000012	1
54	BO000011	PA000012	1
55	BO000001	PA000013	1
56	BO000004	PA000013	1
57	BO000006	PA000013	1
58	BO000009	PA000013	1
59	BO000015	PA000014	1
60	BO000019	PA000014	1
61	BO000011	PA000014	1
62	BO000013	PA000014	1
63	BO000017	PA000014	1
64	BO000020	PA000014	1
65	BO000008	PA000015	1
66	BO000009	PA000015	1
67	BO000004	PA000015	1
68	BO000014	PA000015	1
69	BO000015	PA000015	1

Table II.12: Book to shopping cart table

13. BOOK_BOOKSTORE_ORDER Table

	Book_ID	Order_ID	Order_Quantity
1	BO000001	OR000002	15
2	BO000002	OR000001	50
3	BO000003	OR000004	30
4	BO000004	OR000006	45
5	BO000005	OR000003	55
6	BO000006	OR000005	16
7	BO000007	OR000009	90
8	BO000008	OR000010	25
9	BO000009	OR000011	15
10	BO000010	OR000008	75
11	BO000011	OR000007	34
12	BO000012	OR000004	30
13	BO000013	OR000014	24
14	BO000014	OR000012	20
15	BO000015	OR000013	15
16	BO000016	OR000005	30
17	BO000017	OR000002	15
18	BO000018	OR000015	31
19	BO000019	OR000010	31
20	BO000020	OR000005	30

Table II.13: Book to bookstore order Table

14. BOOK_MEMBER Table

	Book_ID	Member_ID		Book_ID	Member_ID		Book_ID	Member_ID
1	BO000001	ME000001	44	BO000004	ME000003	87	BO000007	ME000005
2	BO000002	ME000001	45	BO000005	ME000003	88	BO000008	ME000005
3	BO000003	ME000001	46	BO000006	ME000003	89	BO000009	ME000005
4	BO000004	ME000001	47	BO000007	ME000003	90	BO000010	ME000005
5	BO000005	ME000001	48	BO000008	ME000003	91	BO000011	ME000005
6	BO000006	ME000001	49	BO000009	ME000003	92	BO000012	ME000005
7	BO000007	ME000001	50	BO000010	ME000003	93	BO000013	ME000005
8	BO000008	ME000001	51	BO000011	ME000003	94	BO000014	ME000005
9	BO000009	ME000001	52	BO000012	ME000003	95	BO000015	ME000005
10	BO000010	ME000001	53	BO000013	ME000003	96	BO000016	ME000005
11	BO000011	ME000001	54	BO000014	ME000003	97	BO000017	ME000005
12	BO000012	ME000001	55	BO000015	ME000003	98	BO000018	ME000005
13	BO000013	ME000001	56	BO000016	ME000003	99	BO000019	ME000005
14	BO000014	ME000001	57	BO000017	ME000003	100	BO000020	ME000005
15	BO000015	ME000001	58	BO000018	ME000003	101	BO000001	ME000006
16	BO000016	ME000001	59	BO000019	ME000003	102	BO000002	ME000006
17	BO000017	ME000001	60	BO000020	ME000003	103	BO000003	ME000006
18	BO000018	ME000001	61	BO000001	ME000004	104	BO000004	ME000006
19	BO000019	ME000001	62	BO000002	ME000004	105	BO000005	ME000006
20	BO000020	ME000001	63	BO000003	ME000004	106	BO000006	ME000006
21	BO000001	ME000002	64	BO000004	ME000004	107	BO000007	ME000006
22	BO000002	ME000002	65	BO000005	ME000004	108	BO000008	ME000006
23	BO000003	ME000002	66	BO000006	ME000004	109	BO000009	ME000006
24	BO000004	ME000002	67	BO000007	ME000004	110	BO000010	ME000006
25	BO000005	ME000002	68	BO000008	ME000004	111	BO000011	ME000006
26	BO000006	ME000002	69	BO000009	ME000004	112	BO000012	ME000006
27	BO000007	ME000002	70	BO000010	ME000004	113	BO000013	ME000006
28	BO000008	ME000002	71	BO000011	ME000004	114	BO000014	ME000006
29	BO000009	ME000002	72	BO000012	ME000004	115	BO000015	ME000006
30	BO000010	ME000002	73	BO000013	ME000004	116	BO000016	ME000006
31	BO000011	ME000002	74	BO000014	ME000004	117	BO000017	ME000006
32	BO000012	ME000002	75	BO000015	ME000004	118	BO000018	ME000006
33	BO000013	ME000002	76	BO000016	ME000004	119	BO000019	ME000006
34	BO000014	ME000002	77	BO000017	ME000004	120	BO000020	ME000006
35	BO000015	ME000002	78	BO000018	ME000004	121	BO000001	ME000007
36	BO000016	ME000002	79	BO000019	ME000004	122	BO000002	ME000007
37	BO000017	ME000002	80	BO000020	ME000004	123	BO000003	ME000007
38	BO000018	ME000002	81	BO000001	ME000005	124	BO000004	ME000007
39	BO000019	ME000002	82	BO000002	ME000005	125	BO000005	ME000007
40	BO000020	ME000002	83	BO000003	ME000005	126	BO000006	ME000007
41	BO000001	ME000003	84	BO000004	ME000005	127	BO000007	ME000007
42	BO000002	ME000003	85	BO000005	ME000005	128	BO000008	ME000007
43	BO000003	ME000003	86	BO000006	ME000005	129	BO000009	ME000007

	Book_ID	Member_ID		Book_ID	Member_ID		Book_ID	Member_ID
130	BO000010	ME000007	173	BO000013	ME000009	216	BO000016	ME000011
131	BO000011	ME000007	174	BO000014	ME000009	217	BO000017	ME000011
132	BO000012	ME000007	175	BO000015	ME000009	218	BO000018	ME000011
133	BO000013	ME000007	176	BO000016	ME000009	219	BO000019	ME000011
134	BO000014	ME000007	177	BO000017	ME000009	220	BO000020	ME000011
135	BO000015	ME000007	178	BO000018	ME000009	221	BO000001	ME000012
136	BO000016	ME000007	179	BO000019	ME000009	222	BO000002	ME000012
137	BO000017	ME000007	180	BO000020	ME000009	223	BO000003	ME000012
138	BO000018	ME000007	181	BO000001	ME000010	224	BO000004	ME000012
139	BO000019	ME000007	182	BO000002	ME000010	225	BO000005	ME000012
140	BO000020	ME000007	183	BO000003	ME000010	226	BO000006	ME000012
141	BO000001	ME000008	184	BO000004	ME000010	227	BO000007	ME000012
142	BO000002	ME000008	185	BO000005	ME000010	228	BO000008	ME000012
143	BO000003	ME000008	186	BO000006	ME000010	229	BO000009	ME000012
144	BO000004	ME000008	187	BO000007	ME000010	230	BO000010	ME000012
145	BO000005	ME000008	188	BO000008	ME000010	231	BO000011	ME000012
146	BO000006	ME000008	189	BO000009	ME000010	232	BO000012	ME000012
147	BO000007	ME000008	190	BO000010	ME000010	233	BO000013	ME000012
148	BO000008	ME000008	191	BO000011	ME000010	234	BO000014	ME000012
149	BO000009	ME000008	192	BO000012	ME000010	235	BO000015	ME000012
150	BO000010	ME000008	193	BO000013	ME000010	236	BO000016	ME000012
151	BO000011	ME000008	194	BO000014	ME000010	237	BO000017	ME000012
152	BO000012	ME000008	195	BO000015	ME000010	238	BO000018	ME000012
153	BO000013	ME000008	196	BO000016	ME000010	239	BO000019	ME000012
154	BO000014	ME000008	197	BO000017	ME000010	240	BO000020	ME000012
155	BO000015	ME000008	198	BO000018	ME000010	241	BO000001	ME000013
156	BO000016	ME000008	199	BO000019	ME000010	242	BO000002	ME000013
157	BO000017	ME000008	200	BO000020	ME000010	243	BO000003	ME000013
158	BO000018	ME000008	201	BO000001	ME000011	244	BO000004	ME000013
159	BO000019	ME000008	202	BO000002	ME000011	245	BO000005	ME000013
160	BO000020	ME000008	203	BO000003	ME000011	246	BO000006	ME000013
161	BO000001	ME000009	204	BO000004	ME000011	247	BO000007	ME000013
162	BO000002	ME000009	205	BO000005	ME000011	248	BO000008	ME000013
163	BO000003	ME000009	206	BO000006	ME000011	249	BO000009	ME000013
164	BO000004	ME000009	207	BO000007	ME000011	250	BO000010	ME000013
165	BO000005	ME000009	208	BO000008	ME000011	251	BO000011	ME000013
166	BO000006	ME000009	209	BO000009	ME000011	252	BO000012	ME000013
167	BO000007	ME000009	210	BO000010	ME000011	253	BO000013	ME000013
168	BO000008	ME000009	211	BO000011	ME000011	254	BO000014	ME000013
169	BO000009	ME000009	212	BO000012	ME000011	255	BO000015	ME000013
170	BO000010	ME000009	213	BO000013	ME000011	256	BO000016	ME000013
171	BO000011	ME000009	214	BO000014	ME000011	257	BO000017	ME000013
172	BO000012	ME000009	215	BO000015	ME000011	258	BO000018	ME000013

259	BO000019	ME000013
260	BO000020	ME000013
261	BO000001	ME000014
262	BO000002	ME000014
263	BO000003	ME000014
264	BO000004	ME000014
265	BO000005	ME000014
266	BO000006	ME000014
267	BO000007	ME000014
268	BO000008	ME000014
269	BO000009	ME000014
270	BO000010	ME000014
271	BO000011	ME000014
272	BO000012	ME000014
273	BO000013	ME000014
274	BO000014	ME000014
275	BO000015	ME000014
276	BO000016	ME000014
277	BO000017	ME000014
278	BO000018	ME000014
279	BO000019	ME000014
280	BO000020	ME000014
281	BO000001	ME000015
282	BO000002	ME000015
283	BO000003	ME000015
284	BO000004	ME000015
285	BO000005	ME000015
286	BO000006	ME000015
287	BO000007	ME000015
288	BO000008	ME000015
289	BO000009	ME000015
290	BO000010	ME000015
291	BO000011	ME000015
292	BO000012	ME000015
293	BO000013	ME000015
294	BO000014	ME000015
295	BO000015	ME000015
296	BO000016	ME000015
297	BO000017	ME000015
298	BO000018	ME000015
299	BO000019	ME000015
300	BO000020	ME000015

Table II.14: Book to member table

15. BOOK_MEMBER_ORDER Table

	Book_ID	MemOrder_ID	MemOrder_Quantity
1	BO000004	MO000001	1
2	BO000002	MO000001	2
3	BO000005	MO000002	1
4	BO000003	MO000002	4
5	BO000006	MO000003	2
6	BO000002	MO000004	3
7	BO000001	MO000005	1
8	BO000002	MO000005	3
9	BO000003	MO000006	1
10	BO000005	MO000006	1
11	BO000001	MO000007	2
12	BO000003	MO000007	3
13	BO000001	MO000008	1
14	BO000002	MO000008	1
15	BO000003	MO000008	1
16	BO000005	MO000008	1
17	BO000001	MO000009	1
18	BO000002	MO000009	3
19	BO000003	MO000010	1
20	BO000013	MO000011	1
21	BO000017	MO000011	1
22	BO000005	MO000012	1
23	BO000006	MO000012	1
24	BO000007	MO000012	1
25	BO000007	MO000013	1
26	BO000011	MO000013	1
27	BO000003	MO000014	1
28	BO000007	MO000014	1
29	BO000009	MO000014	1
30	BO000004	MO000015	1
31	BO000013	MO000016	1
32	BO000017	MO000016	1
33	BO000020	MO000016	1
34	BO000019	MO000017	1
35	BO000002	MO000018	1
36	BO000018	MO000019	1
37	BO000010	MO000020	1
38	BO000012	MO000020	1
39	BO000015	MO000020	1
40	BO000019	MO000020	1

Table II.15: Book to member order table

III. SQL-Data Manipulation Language (DML)

Name: Lai Kai Yong

1. List the book(s) which has the highest rating. Show book id, book name, and the rating.

SQL Code:

```
SELECT TOP 1 B.Book_ID, UB.Book_Title, AVG(MF.Rating) AS Rating
FROM BOOK B
INNER JOIN [MEMBER FEEDBACK] MF
ON B.Book_ID = MF.Book_ID
INNER JOIN UNIVERSAL_BOOK UB
ON B.Book_ISBN = UB.Book_ISBN
GROUP BY B.Book_ID, UB.Book_Title
ORDER BY AVG(MF.Rating) DESC;
```

-- Or

```
CREATE VIEW Rating AS
SELECT B.Book_ID, UB.Book_Title, AVG(MF.Rating) AS Rating
FROM BOOK B
INNER JOIN [MEMBER FEEDBACK] MF
ON B.Book_ID = MF.Book_ID
INNER JOIN UNIVERSAL_BOOK UB
ON B.Book_ISBN = UB.Book_ISBN
GROUP BY B.Book_ID, UB.Book_Title;

SELECT DISTINCT Book_ID, Book_Title, Rating
FROM Rating
WHERE Rating = (SELECT MAX(Rating) FROM Rating);
```

Output:

	Book_ID	Book_Title	Rating
1	BO000006	Tales of Charkolol	9

Table III.1: DML Question 1 Output

2. Find the total number of feedback per member. Show member id, member name, and total number of feedback per member.

SQL Code:

```
SELECT M.Member_ID, M.Member_Name, COUNT(MF.Feedback_ID) AS 'Total Feedback'  
FROM MEMBER M  
INNER JOIN [MEMBER FEEDBACK] MF  
ON M.Member_ID = MF.Member_ID  
GROUP BY M.Member_ID, M.Member_Name;
```

Output:

	Member_ID	Member_Name	Total Feedback
1	ME000001	Kim Jong	2
2	ME000002	Ji Eun	3
3	ME000003	Wayne Bryan	2
4	ME000005	Johnny Jhin	1
5	ME000006	Ibrahim Mustafa	2
6	ME000007	Janna	2
7	ME000009	Qiyana	5
8	ME000010	Rell	3

Table III.2: DML Question 2 Output

3. Find the total number of book published by each publisher. Show publisher id, publisher name, and number of book published.

SQL Code:

```
SELECT P.Pub_ID, P.Pub_Name, P.Pub_Qty
FROM PUBLISHER P;

-- Or

-- Count all books in bookstore published by publisher
SELECT P.Pub_ID, P.Pub_Name, COUNT(B.Book_ID) AS 'Total book published'
FROM BOOK B
INNER JOIN PUBLISHER P
ON B.Pub_ID = P.Pub_ID
GROUP BY P.Pub_ID, P.Pub_Name
```

Output:

	Pub_ID	Pub_Name	Pub_Qty
1	PU000001	Oxford Publication	5
2	PU000002	Ladybird Publication	4
3	PU000003	Miracle Publication	3
4	PU000004	Miyuki	5
5	PU000005	Gemilang Publishers	7
6	PU000006	Cambridge Publication	8
7	PU000007	Popular Publication	3
8	PU000008	National Geographic Publication	4
9	PU000009	Oxbridge Publications	2
10	PU000010	IEEE Publications	7

OR

	Pub_ID	Pub_Name	Total book published
1	PU000001	Oxford Publication	4
2	PU000002	Ladybird Publication	3
3	PU000003	Miracle Publication	4
4	PU000004	Miyuki	5
5	PU000005	Gemilang Publishers	4

Table III.3: DML Question 3 Output

4. Find the total number of books ordered by store manager from each publisher.

SQL Code:

```
SELECT P.Pub_ID, P.Pub_Name, SUM(BMO.Order_Quantity) AS 'Total Books Ordered'
FROM [BOOKSTORE ORDER] BO
INNER JOIN PUBLISHER P
ON BO.Pub_ID = P.Pub_ID
INNER JOIN BOOK_BOOKSTORE_ORDER BMO
ON BO.Order_ID = BMO.Order_ID
GROUP BY P.Pub_ID, P.Pub_Name;
```

Output:

	Pub_ID	Pub_Name	Total Books Ordered
1	PU000001	Oxford Publication	153
2	PU000002	Ladybird Publication	120
3	PU000003	Miracle Publication	196
4	PU000004	Miyuki	116
5	PU000005	Gemilang Publishers	91

Table III.4: DML Question 4 Output

Name: Cheong Sheng Kui

5. Find the total number of books ordered by each member.

SQL Code:

```
SELECT MO.Member_ID, SUM(BMO.MemOrder_Quantity) AS 'Total Books Ordered'  
FROM [MEMBER ORDER] MO INNER JOIN BOOK_MEMBER_ORDER BMO ON MO.MemOrder_ID =  
BMO.MemOrder_ID  
GROUP BY MO.Member_ID
```

Output:

	Member_ID	Total Books Ordered
1	ME000001	5
2	ME000002	6
3	ME000003	11
4	ME000005	5
5	ME000006	11
6	ME000007	2
7	ME000008	1
8	ME000009	5
9	ME000010	3
10	ME000011	2
11	ME000014	3

Table III.5 DML Question 5 Output

6. Find the bestselling book(s).

SQL Code:

```
SELECT TOP 5 BMO.Book_ID, UB.Book_Title, SUM(BMO.MemOrder_Quantity) AS 'Sales'
FROM BOOK_MEMBER_ORDER BMO INNER JOIN BOOK B ON BMO.Book_ID = B.Book_ID
INNER JOIN UNIVERSAL_BOOK UB ON UB.Book_ISBN = B.Book_ISBN
INNER JOIN [MEMBER ORDER] MO ON BMO.MemOrder_ID = MO.MemOrder_ID
WHERE Payment_Status = 'Completed'
GROUP BY BMO.Book_ID, UB.Book_Title ORDER BY SUM(BMO.MemOrder_Quantity) DESC
```

Output:

	Book_ID	Book_Title	Sales
1	BO000003	Java Fundamentals	11
2	BO000005	Alien Story	4
3	BO000007	Dora the explorer	3
4	BO000001	Jungle Book	3
5	BO000006	Tales of Charkolol	3

Table III.6 DML Question 6 Output

7. Show list of total customers based on gender who are registered as members in APU E-Bookstore. The list should show total number of registered members and total number of gender (male and female).

SQL Code:

```
SELECT COUNT(Member_ID) AS 'Total Registered Member',
(SELECT COUNT(Member_ID) FROM MEMBER WHERE Gender = 'M') AS 'Male Member',
(SELECT COUNT(Member_ID) FROM MEMBER WHERE Gender = 'F') AS 'Female Member'
FROM MEMBER
```

Output:

	Total Registered Member	Male Member	Female Member
1	15	7	8

Table III.7 DML Question 7 Output

Name: Lim Wye Yee

8. Show a list of purchased books that **have not been delivered** to members. The list should show member identification number, address, contact number, book serial number, book title, quantity, date and status of delivery.

SQL Code:

```
SELECT DISTINCT M.Member_ID, CONCAT(M.Street, ', ', M.[Zip Code], ', ', M.City, ', ', M.State, ', ', M.Country) AS 'Member_Address', MCN.Personal_Phone, MCN.Work_Phone, B.Book_ID, UB.Book_Title, BMO.MemOrder_Quantity, MO.MemOrder_Date, MO.Delivery_Status
FROM MEMBER M INNER JOIN MEMBER_CONTACT_NUMBER MCN ON M.Member_ID = MCN.Member_ID
INNER JOIN [MEMBER ORDER] MO ON MO.Member_ID = M.Member_ID
INNER JOIN BOOK_MEMBER_ORDER BMO ON BMO.MemOrder_ID = MO.MemOrder_ID
INNER JOIN BOOK B ON B.Book_ID = BMO.Book_ID
INNER JOIN UNIVERSAL_BOOK UB ON UB.Book_ISBN = B.Book_ISBN
WHERE MO.Delivery_Status != 'Delivered';
```

Output:

	Member_ID	Member_Address	Personal_Phone	Work_Phone	Book_ID	Book_Title	MemOrder_Quantity	MemOrder_Date	Delivery_Status
1	ME000005	Bianmao Rd, 40812, ZengFuXiang, Chongqing, China	NULL	03-87544964	B0000001	Jungle Book	2	2021-04-27	DELIVERING
2	ME000005	Bianmao Rd, 40812, ZengFuXiang, Chongqing, China	NULL	03-87544964	B0000003	Java Fundamentals	3	2021-04-27	DELIVERING
3	ME000006	Qilla Kohna, 60101, Kuhna Qasim Bagh, Multan, Pa...	017-8545821	NULL	B0000003	Java Fundamentals	1	2021-04-29	DELIVERING
4	ME000008	Victoria Road, 34566, Newport, Wales, United Kingd...	015-4426875	NULL	B0000002	Python Bible	1	2021-07-07	DELIVERING
5	ME000011	Kung Yip Street, 23482, Kwai Tsing District, Hong Ko...	016-4521425	NULL	B0000004	C++ Computer	1	2021-05-06	DELIVERING
6	ME000011	Kung Yip Street, 23482, Kwai Tsing District, Hong Ko...	016-4521425	NULL	B0000018	Architecture ideas	1	2021-09-07	DELIVERING

Table III.8: DML Question 8 Output

9. Show the member who spent most on buying books. Show member id, member name and total expenditure.

SQL Code:

```
SELECT TOP 1 M.Member_ID, M.Member_Name, SUM(MO.MemOrder_Price) AS 'Total expenditure'
FROM MEMBER M INNER JOIN [MEMBER ORDER] MO ON M.Member_ID = MO.Member_ID
GROUP BY M.Member_ID, M.Member_Name
ORDER BY SUM(MO.MemOrder_Price) DESC;
```

Output:

	Member_ID	Member_Name	Total expenditure
1	ME000003	Wayne Bryan	310.00

Table III.9: DML Question 9 Output

10. Show a list of total books as added by each member in the shopping cart.

SQL Code:

```
SELECT SC.Member_ID, M.Member_Name, SUM(BSC.Purchase_Quantity) AS 'Total books purchased'
FROM [SHOPPING CART] SC INNER JOIN BOOK_SHOPPING_CART BSC ON SC.Purchase_ID =
BSC.Purchase_ID
INNER JOIN MEMBER M ON M.Member_ID = SC.Member_ID
INNER JOIN BOOK B ON B.Book_ID = BSC.Book_ID
GROUP BY SC.Member_ID, M.Member_Name;
```

Output:

	Member_ID	Member_Name	Total books purchased
1	ME000001	Kim Jong	6
2	ME000002	Ji Eun	6
3	ME000003	Wayne Bryan	6
4	ME000004	Donny Tom	8
5	ME000005	Johnny Jhin	8
6	ME000006	Ibrahim Mustafa	4
7	ME000007	Janna	5
8	ME000008	Taric	6
9	ME000009	Qiyana	7
10	ME000010	Rell	3
11	ME000011	Jeremy	2
12	ME000012	Tris Jenner	3
13	ME000013	Kayle Lim	4
14	ME000014	Sivades	6
15	ME000015	Fiora Yu	5

Table III.10: DML Question 10 Output

IV. Conclusion

In conclusion, creating a database requires ERD with corresponding relational schema, normalization, data dictionary and implementation of SQL statements. ERD displays the relationship between entities and the attributes residing under each of the tables. Relational schema is listed based on the ERD where relationship of entities can be identified by cardinalities and relationship between entities. Normalization is conducted on the relational schema up to 3NF to draft a database with high data integrity. Data dictionary is then generated to define the data type and data size used for all attributes and the relations between tables. SQL statements are then used to implement design on database and fulfill queries on the database. DDL is used to create table and insert data into the database while DML is used to query the database for the display of records regarding specific conditions. DML allows referencing through several table by joining several tables relating to each other together with same key values. The project team has completed all process above to implement APU E-Bookstore's DBMS using Microsoft SQL Server Management Studio (SSMS). The implementation of DBMS will cater the need of members within the organisation as it boosts efficiency and effectiveness of APU Bookstore employees' workforce.

V. References

Peter, 2009. *SQL DataType - How to store a year?*. [Online]
Available at: <https://stackoverflow.com/questions/696506/sql-datatype-how-to-store-a-year>
[Accessed 26 May 2021].

W3Schools, 2021. *SQL Data Types for MySQL, SQL Server, and MS Access*. [Online]
Available at: https://www.w3schools.com/sql/sql_datatypes.asp
[Accessed 26 5 2021].