Introduction to Databases

Module Code : **(CT042-3-1-IDB)**

Intake Code : **APU1F2303CS(IS)**

Lecturer Name : **Ms. Lai Chew Ping**

Hand In Date : **20 November 2023**

Group No. : **Group 22**

No	Name	TP Number	Signature
1	Tay Jun Long	TP074666	
2	Angelina Leanore	TP072929	
3	Terence Lim Dao Liang	TP073243	
4	Eraliev Suimonkul	TP068888	

Table of Contents

1. Database Schema

Entity Relationship diagram

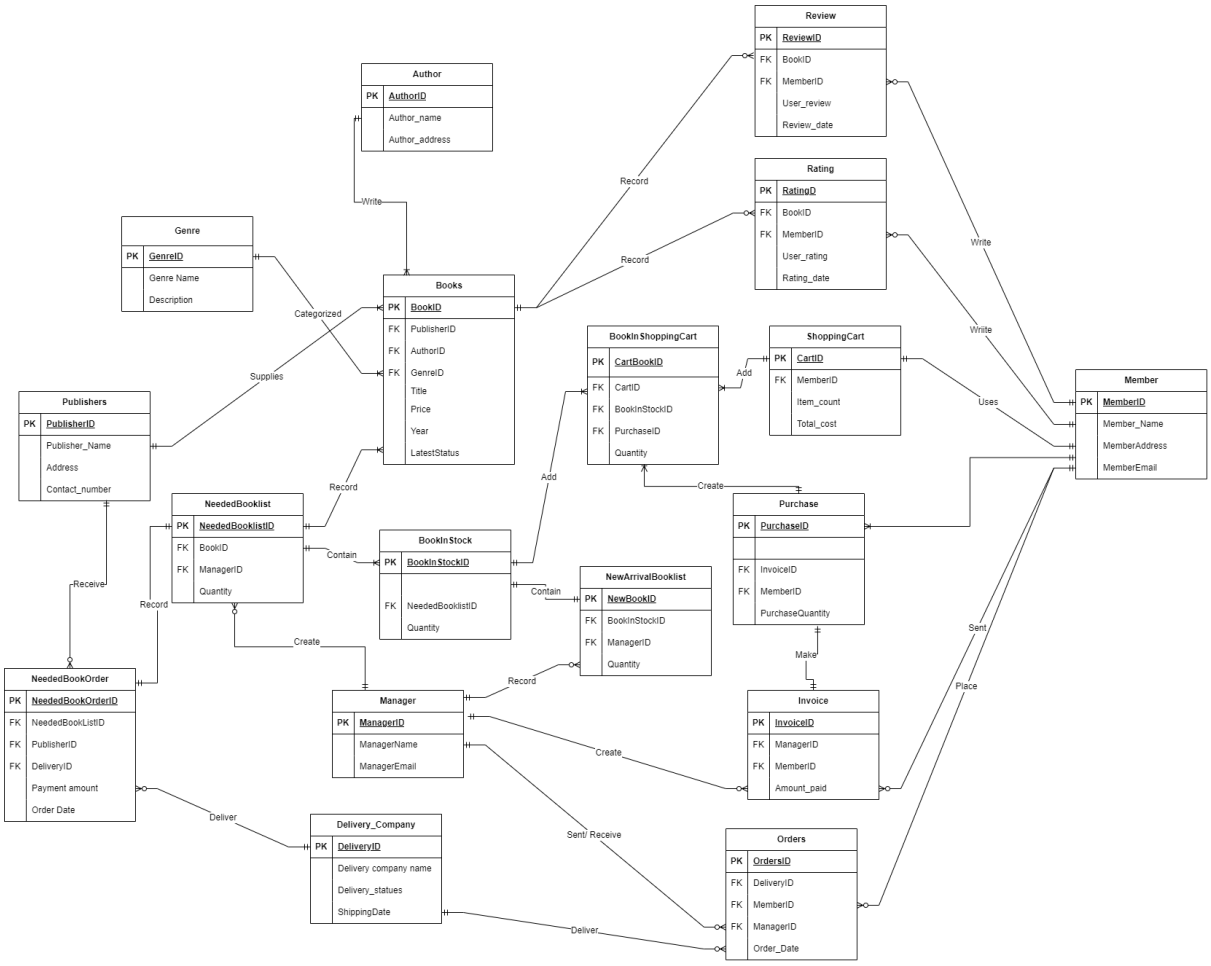

Diagram 1.1 Entity Relationship Diagram done in Draw.io

Database diagram

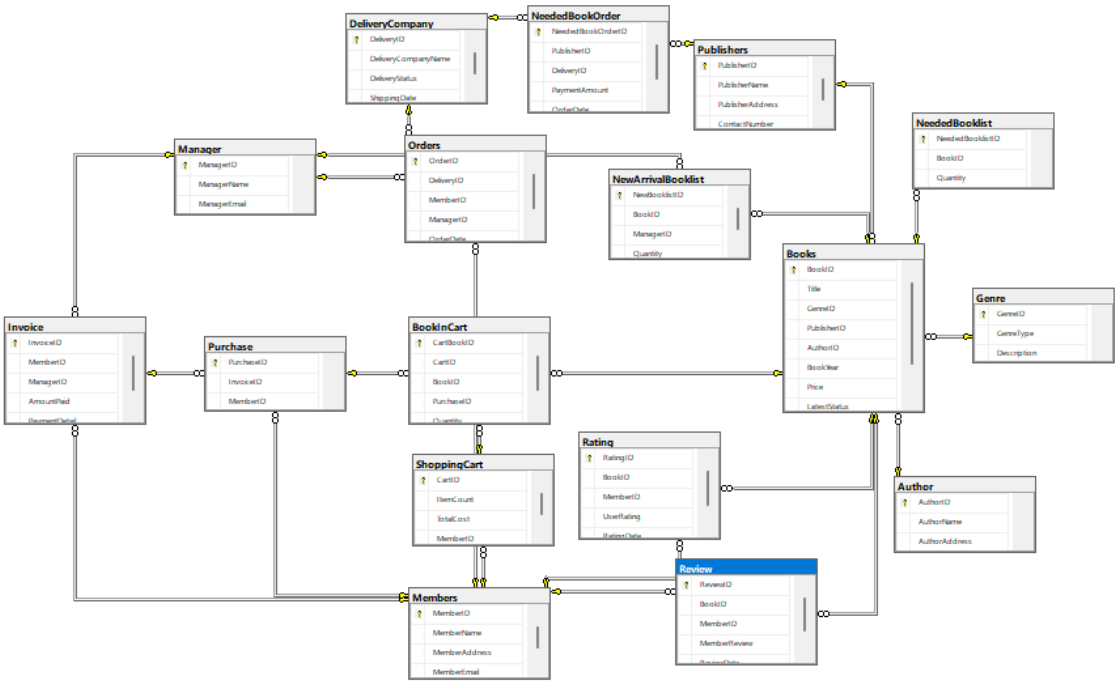

Diagram 1.2 Database diagram generated in SQL.

2. SQL Data Definition Language (DDL)

Table Members

```
CREATE TABLE Members (
MemberID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
MemberName varchar(100),
MemberAddress varchar(100),
MemberEmail varchar(100));

insert into Members values
('Alice', '123 Bukit Jalil, Kuala Lumpur', 'Alice@mail.com'),
('John', '345 Sri Petaling, Kuala Lumpur', 'John@mail.com'),
('Emily', '567 Bukit Bintang, Kuala Lumpur', 'Emily@mail.com'),
('Bob', '789 Main Street, Kuala Lumpurr', 'Bob@mail.com'),
('Charlie', '901 Oak Street, Kuala Lumpur', 'Charlie@mail.com');
```

Table Author

```
CREATE TABLE Author (
AuthorID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
AuthorName varchar(100),
AuthorAddress varchar(100));

insert into Author values
('Danielle Steel', '342 Steel, New York'),
('Dan Brown', '182 Brown, France'),
('John Green', '623 Green, London'),
('Nora Roberts', '346 Roberts, Japan'),
('Paulo Coelho', '678 Coelho, Indonesia'),
('Delia Owens', '787 Delia, Kuala Lumpur');
```

Table Publishers

```
CREATE TABLE Publishers (
PublisherID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
PublisherName varchar(100),
PublisherAddress varchar(100),
ContactNumber varchar(15));

insert into Publishers values
('HarperCollins', '123 Harper, London', '657-8392'),
('Springer Nature', '234 Springer, France', '089-1233'),
('Scholastic', '456 Scholastic, Kuala Lumpur', '647-2740'),
('McGraw-Hill', '678 GrawHill, New York', '097-2374'),
('Cengage', '890 Cengage, Indonesia', '768-7234'),
('Hachette Book', '425 Hachette, Japan', '628-3489');
```

Table NeededBookList

```
CREATE TABLE NeededBooklist (
NeededBooklistID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
ManagerID int FOREIGN KEY REFERENCES Manager(ManagerID),
BookID int FOREIGN KEY REFERENCES Books(BookID),
Quantity int);

insert into NeededBooklist values
(1,4, 100),
(1, 5, 150),
(1, 6, 200),
(2, 7, 125),
(3, 8, 175);
```

Table Review

```
CREATE TABLE Review (
ReviewID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
BookID int FOREIGN KEY REFERENCES Books(BookID),
MemberID int FOREIGN KEY REFERENCES Members(MemberID),
MemberReview varchar(500),
ReviewDate date);

insert into Review values
(1, 3,  'Great Book', '2023-01-15'),
(2, 5,  'Good Book', '2023-01-16'),
(3, 4,  'Exciting Book', '2023-01-17'),
(4, 4,  'Informative', '2023-01-18'),
(5, 1,  'Well Written', '2023-01-19'),
(6, 3,  'Well done', '2023-01-20'),
(7, 2,  'Loved the Character', '2023-01-21'),
(8, 3,  'Best Book Ever', '2023-01-22'),
(9, 5,  'Nice Plot twist', '2023-01-23'),
(10, 1, 'Good Plot', '2023-01-24');
```

Table Rating

```
CREATE TABLE Rating(
RatingID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
BookID int FOREIGN KEY REFERENCES Books(BookID),
MemberID int FOREIGN KEY REFERENCES Members(MemberID),
UserRating decimal(4,2),
RatingDate date);

insert into Rating values
(1, 3,  8, '2023-01-15'),
(2, 5,  7.5, '2023-01-16'),
(3, 4,  8, '2023-01-17'),
(4, 4,  10, '2023-01-18'),
(5, 1,  9, '2023-01-19'),
(6, 3,  8.5, '2023-01-20'),
(7, 2,  7, '2023-01-21') ,
(8, 3,  9.5, '2023-01-22'),
(9, 5,  7.5, '2023-01-23'),
(10, 1, 10, '2023-01-24');
```

Table BookInCart

```
CREATE TABLE BookInCart (
CartBookID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
CartID int FOREIGN KEY REFERENCES ShoppingCart(CartID),
BookInStockID int FOREIGN KEY REFERENCES BookInStock(BookInStockID),
PurchaseID int FOREIGN KEY REFERENCES Purchase(PurchaseID),
Quantity int);

insert into BookInCart values
(1, 1, 1, 3),
(2, 2, 2, 4),
(3, 3, 3, 1),
(4, 4, 4, 2),
(5, 5, 5, 5);
```

Table DeliveryCompany

```
CREATE TABLE DeliveryCompany (
DeliveryID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
DeliveryCompanyName varchar(100),
DeliveryStatus int,
ShippingDate date);

insert into DeliveryCompany values
('FedEx', 1, '2023-11-15'),
('Jnt', 1, '2023-11-16'),
('SiCepat', 0, '2023-11-16'),
('Xpress', 0, '2023-11-14'),
('DHL', 1, '2023-11-17');
```

Table NewArrivalBookList

```
CREATE TABLE NewArrivalBooklist (
NewBooklistID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
BookInStockID int FOREIGN KEY REFERENCES BookInStock(BookInStockID),
ManagerID int FOREIGN KEY REFERENCES Manager(ManagerID),
Quantity int);

insert into NewArrivalBooklist
Values
(1, 3, 100),
(2, 3, 150),
(3, 3, 200),
(4, 4, 125),
(5, 4, 175);
```

Table Invoice

```
CREATE TABLE Invoice (
InvoiceID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
MemberID int FOREIGN KEY REFERENCES Members(MemberID),
ManagerID int FOREIGN KEY REFERENCES Manager(ManagerID),
AmountPaid decimal (10,2),
PaymentDetail decimal (10,2));

insert into Invoice values
(1, 1, 45.98, 50),
(2, 2, 19.99, 20),
(3, 3, 89.97, 90),
(4, 4, 25.55, 25.6),
(1, 1, 76.48, 76.5);
```

Table Orders

```
CREATE TABLE Orders (
OrderID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
MemberID int FOREIGN KEY REFERENCES Members(MemberID),
ManagerID int FOREIGN KEY REFERENCES Manager(ManagerID),
DeliveryID int FOREIGN KEY REFERENCES DeliveryCompany(DeliveryID),
OrderDate date);


insert into Orders values
(1, 1, 1, '2023-11-10'),
(2, 2, 2, '2023-11-12'),
(3, 3, 3, '2023-11-16'),
(4, 4, 4, '2023-11-9'),
(4, 3, 5, '2023-11-14');
```

Table Manager

```
CREATE TABLE Manager (
ManagerID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
ManagerName varchar(100),
ManagerEmail varchar(100));

insert into Manager Values
('Lea', 'Lea@mail.com'),
('Terence', 'Terence@mail.com'),
('Jun', 'Jun@mail.com'),
('Sam', 'Sam@mail.com');
```

Table Genre

```
Create Database IDBAssignment;

use IDBAssignment;

CREATE TABLE Genre (
GenreID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
GenreName varchar(100),
Description varchar(300));

insert into Genre
values
('Drama', 'Works written to be performed on stage'),
('History', 'Records of human events'),
('Fiction', 'prose works of imagination, typically with  narrative and characters'),
('Non-Fiction', 'prose works that present information, facts, and ideas'),
('Mystery', 'Involves solving a crime or uncovering secrets');
```

Table BookInStock

```
CREATE TABLE BookInStock (
BookInStockID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
NeededBooklistID int FOREIGN KEY REFERENCES NeededBooklist(NeededBooklistID),
Quantity int);


Insert into BookInStock values
(1, 100),
(2, 150),
(3, 200),
(4, 125),
(5, 175);
```

Table Book

```
CREATE TABLE Books (
BookID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
Title varchar(100),
GenreID int FOREIGN KEY REFERENCES Genre(GenreID),
PublisherID int FOREIGN KEY REFERENCES Publishers(PublisherID),
AuthorID int FOREIGN KEY REFERENCES Author(AuthorID),
BookYear int,
Price decimal(10,2),
LatestStatus int);


insert into Books Values
('Identity', 5, 3, 4, 2023, 25.99, 1),
('Eleven Minutes', 3, 1, 5, 2003, 19.99, 0),
('Where The Crawdads Sings', 3, 4, 6, 2023, 20, 0),
('Origin', 2, 6, 2, 2017, 30.10, 1),
('The Fault in Our Stars', 1, 4, 3, 2023, 23.13, 1),
('The Obsession', 1, 1, 4, 2017, 36.87, 1),
('Bride', 4, 5, 5, 1990, 28.99, 1),
('The Promise', 1, 2, 1, 2000, 15.99, 1),
('The Alchemist', 2, 6, 5, 1988, 18, 0),
('Looking For Alaska', 5, 3, 3, 2023, 21.50, 1);
```

Table NeededBookOrder

```
CREATE TABLE NeededBookOrder (
NeededBookOrderID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
PublisherID int FOREIGN KEY REFERENCES Publishers(PublisherID),
DeliveryID int FOREIGN KEY REFERENCES DeliveryCompany(DeliveryID),
PaymentAmount decimal (10,2),
OrderDate date);

insert into NeededBookOrder Values
(2, 1, 49.99, '2023-11-10'),
(3, 2, 29.99, '2023-11-12'),
(1, 3, 39.99, '2023-11-16'),
(3, 4, 25.50, '2023-11-9'),
(2, 5, 45.50, '2023-11-14');
```

Table ShoppingCart

```
CREATE TABLE ShoppingCart
(CartID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
ItemCount int,
TotalCost decimal(10,2),
MemberID int FOREIGN KEY (MemberID) REFERENCES Members(MemberID));

insert into ShoppingCart values
(3, 77.97, 3),
(4, 79.96, 5),
(1, 20, 4),
(2, 60.2, 4),
(5, 115.65, 1),
(2, 73.74, 3),
(3, 110.61, 2),
(6, 95.94, 3),
(2, 36, 5),
(5, 107.5, 1);
```

Table Purchase

```
CREATE TABLE Purchase (
PurchaseID int IDENTITY (1,1) NOT NULL PRIMARY KEY,
InvoiceID int FOREIGN KEY REFERENCES Invoice(InvoiceID),
MemberID int FOREIGN KEY REFERENCES Members(MemberID),
PurchaseQuantity int);

insert into Purchase values
(1, 1, 1),
(2, 2, 1),
(3, 3, 2),
(4, 4, 1),
(5, 1, 2);
```

3. SQL Data Manipulation Language (DML)

Question 1

```
--Q1.
SELECT p.PublisherID, p.PublisherName, COUNT(b.BookID) as TotalBooks FROM Books b
JOIN Publishers p ON b.PublisherID = p.PublisherID GROUP BY p.PublisherID, p.PublisherName;
```

119 %

Results | Messages

	PublisherID	PublisherName	TotalBooks
1	1	HarperCollins	2
2	2	Springer Nature	1
3	3	Scholastic	2
4	4	McGraw-Hill	2
5	5	Cengage	1
6	6	Hachette Book	2

Question 2

```
--Q2.
SELECT m.MemberID, m.MemberName, b.Title FROM ShoppingCart sc
JOIN Members m ON sc.MemberID = m.MemberID
JOIN BookInCart bic ON sc.CartID = bic.CartID
JOIN BookInStock bis ON bic.BookInStockID = bis.BookInStockID
JOIN NeededBooklist nbl ON nbl.NeededBooklistID = bis.NeededBooklistID
JOIN Books b ON nbl.BookID = b.BookID
ORDER BY m.MemberName;
```

119 %

Results | Messages

	MemberID	MemberName	Title
1	1	Alice	The Promise
2	4	Bob	The Obsession
3	4	Bob	Bride
4	5	Charlie	The Fault in Our Stars
5	3	Emily	Origin

Question 3

```
--Q3.
SELECT b.BookID, b.Title, AVG(r.UserRating) as Rating FROM Books b
JOIN Rating r ON b.BookID = r.BookID
GROUP BY b.BookID, b.Title
HAVING AVG(r.UserRating) > 9.5
ORDER BY Rating DESC;
```

119 %

Results | Messages

	BookID	Title	Rating
1	4	Origin	10.000000
2	10	Looking For Alaska	10.000000

Question 4

```sql
--Q4
select
M.MemberID,
M.MemberName,
M.MemberAddress,
M.MemberEmail,
count(R.ReviewID) as NumberOfReview
from Members M Join Review R on M.MemberID=R.MemberID
Group by M.MemberID, M.MemberName, M.MemberAddress,M.MemberEmail
order by NumberOfReview ASC;
```

119 %

Results Messages

	MemberID	MemberName	MemberAddress	MemberEmail	NumberOfReview
1	2	John	345 Sri Petaling, Kuala Lumpur	John@mail.com	1
2	4	Bob	789 Main Street, Kuala Lumpurr	Bob@mail.com	2
3	5	Charlie	901 Oak Street, Kuala Lumpur	Charlie@mail.com	2
4	1	Alice	123 Bukit Jalil, Kuala Lumpur	Alice@mail.com	2
5	3	Emily	567 Bukit Bintang, Kuala Lumpur	Emily@mail.com	3

Question 5

```sql
--Q5
SELECT
P.PublisherID,
P.PublisherName,
P.PublisherAddress,
P.ContactNumber,
COUNT(B.BookID) AS NumberOfBooksPublished
FROM Publishers P JOIN Books B ON P.PublisherID = B.PublisherID
GROUP BY P.PublisherID, P.PublisherName, P.PublisherAddress, P.ContactNumber
ORDER BY NumberOfBooksPublished Desc;
```

119 %

Results Messages

	PublisherID	PublisherName	PublisherAddress	ContactNumber	NumberOfBooksPublished
1	1	HarperCollins	123 Harper, London	657-8392	2
2	3	Scholastic	456 Scholastic, Kuala Lumpur	647-2740	2
3	4	McGraw-Hill	678 GrawHill, New York	097-2374	2
4	6	Hachette Book	425 Hachette, Japan	628-3489	2
5	5	Cengage	890 Cengage, Indonesia	768-7234	1
6	2	Springer Nature	234 Springer, France	089-1233	1

Question 6

```sql
--Q6
SELECT
M.ManagerID,
M.ManagerName,
P.PublisherID,
P.PublisherName,
SUM(NBL.Quantity) AS TotalBooksOrdered
FROM
Manager M
JOIN
NeededBooklist NBL ON M.ManagerID = NBL.ManagerID
JOIN
Books B ON NBL.BookID = B.BookID
JOIN
Publishers P ON B.PublisherID = P.PublisherID
WHERE M.ManagerID < 2
GROUP BY
M.ManagerID, M.ManagerName, P.PublisherID, P.PublisherName
ORDER BY
TotalBooksOrdered, PublisherID ASC, ManagerID ASC;
```

98 %

Results | Messages

	ManagerID	ManagerName	PublisherID	PublisherName	TotalBooksOrdered
1	1	Lea	6	Hachette Book	100
2	1	Lea	4	McGraw-Hill	150
3	1	Lea	1	HarperCollins	200

Question 7

```sql
--Q7.
SELECT g.GenreName, SUM(bis.Quantity) as Total FROM Genre g
JOIN Books b ON g.GenreID = b.GenreID
JOIN NeededBooklist nbl ON b.BookID = nbl.BookID
JOIN BookInStock bis ON nbl.NeededBooklistID = bis.NeededBooklistID
GROUP BY g.GenreName
ORDER BY Total DESC;
```

119 %

Results | Messages

	GenreName	Total
1	Drama	525
2	Non-Fiction	125
3	History	100

Question 8

```
--Q8.
SELECT b.BookID, b.Title, bic.Quantity FROM Books b
JOIN NeededBooklist nbl ON b.BookID = nbl.BookID
JOIN BookInStock bis ON nbl.NeededBooklistID = bis.NeededBooklistID
JOIN BookInCart bic ON bis.BookInStockID = bic.BookInStockID
JOIN Purchase p ON bic.PurchaseID = p.PurchaseID
ORDER BY bic.Quantity DESC;
```

119 %

Results | Messages

	BookID	Title	Quantity
1	8	The Promise	5
2	5	The Fault in Our Stars	4
3	4	Origin	3
4	7	Bride	2
5	6	The Obsession	1

Question 9

```
--Q9.
SELECT TOP 1 m.MemberID, m.MemberName, SUM(i.AmountPaid) as TotalSpent FROM Members m
JOIN Invoice i ON m.MemberID = i.MemberID
GROUP BY m.MemberID, m.MemberName
ORDER BY TotalSpent DESC;
```

119 %

Results | Messages

	MemberID	MemberName	TotalSpent
1	1	Alice	122.46

Question 10

```
--Q10
Select * from Members where MemberID Not in (Select MemberID from Orders);
```

119 %

Results | Messages

	MemberID	MemberName	MemberAddress	MemberEmail
1	5	Charlie	901 Oak Street, Kuala Lumpur	Charlie@mail.com

Question 11

```
--Q11.
SELECT b.Title, p.PurchaseQuantity FROM Books b
JOIN NeededBooklist nbl ON b.BookID = nbl.BookID
JOIN BookInStock bis ON nbl.NeededBooklistID = bis.NeededBooklistID
JOIN BookInCart bic ON bis.BookInStockID = bic.BookInStockID
JOIN Purchase p ON bic.PurchaseID = p.PurchaseID
JOIN Invoice i ON p.InvoiceID = i.InvoiceID
JOIN Manager m ON i.ManagerID = m.ManagerID
JOIN Orders o ON m.ManagerID = o.ManagerID
JOIN DeliveryCompany dc ON o.DeliveryID = dc.DeliveryID
WHERE dc.DeliveryStatus = 0;
```

119 %

Results | Messages

	Title	PurchaseQuantity
1	The Obsession	2
2	Bride	1

Question 12

```
--q12
Select * from Members where MemberID in
(Select MemberID from Orders Group by MemberID having Count(MemberID) >=2);
```

119 %

Results | Messages

	MemberID	MemberName	MemberAddress	MemberEmail
1	4	Bob	789 Main Street, Kuala Lumpurr	Bob@mail.com

Assignment Workload Matrix

Part	Component	Terence Lim Dao Liang	Angelina Leanore	Tay Jun Long	Eraliev Suimonkul	Total
1	a) Database and Database Management System	**25%**	**25%**	**25%**	**25%**	**100%**
1	b) Business Rules & Normalization	**25%**	**25%**	**25%**	**25%**	**100%**
1	c) Entity Relationship Diagram	**25%**	**25%**	**25%**	**25%**	**100%**

Part	Component	Terence Lim Dao Liang	Angelina Leanore	Tay Jun Long	Eraliev Suimonkul	Total
2	d) Database Schema	**25%**	**25%**	**25%**	**25%**	**100%**
2	e) DDL	**25%**	**25%**	**25%**	**25%**	**100%**
2	f) DML	**25%**	**25%**	**25%**	**25%**	**100%**