



# **DATABASE SYSTEMS**

**Module Code: AICT005-4-1-DAS**

**Group Assignment**

**Student Name and TP Number:**

1. Faithlin Hoe Wei Xin (TP055347)
2. Koh Tian Xin (TP055789)
3. Tan Chin Yuon (TP056500)
4. Siah Sung Hanns (TP056553)

**Intake Code: UCDF1905ICT(SE)**

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

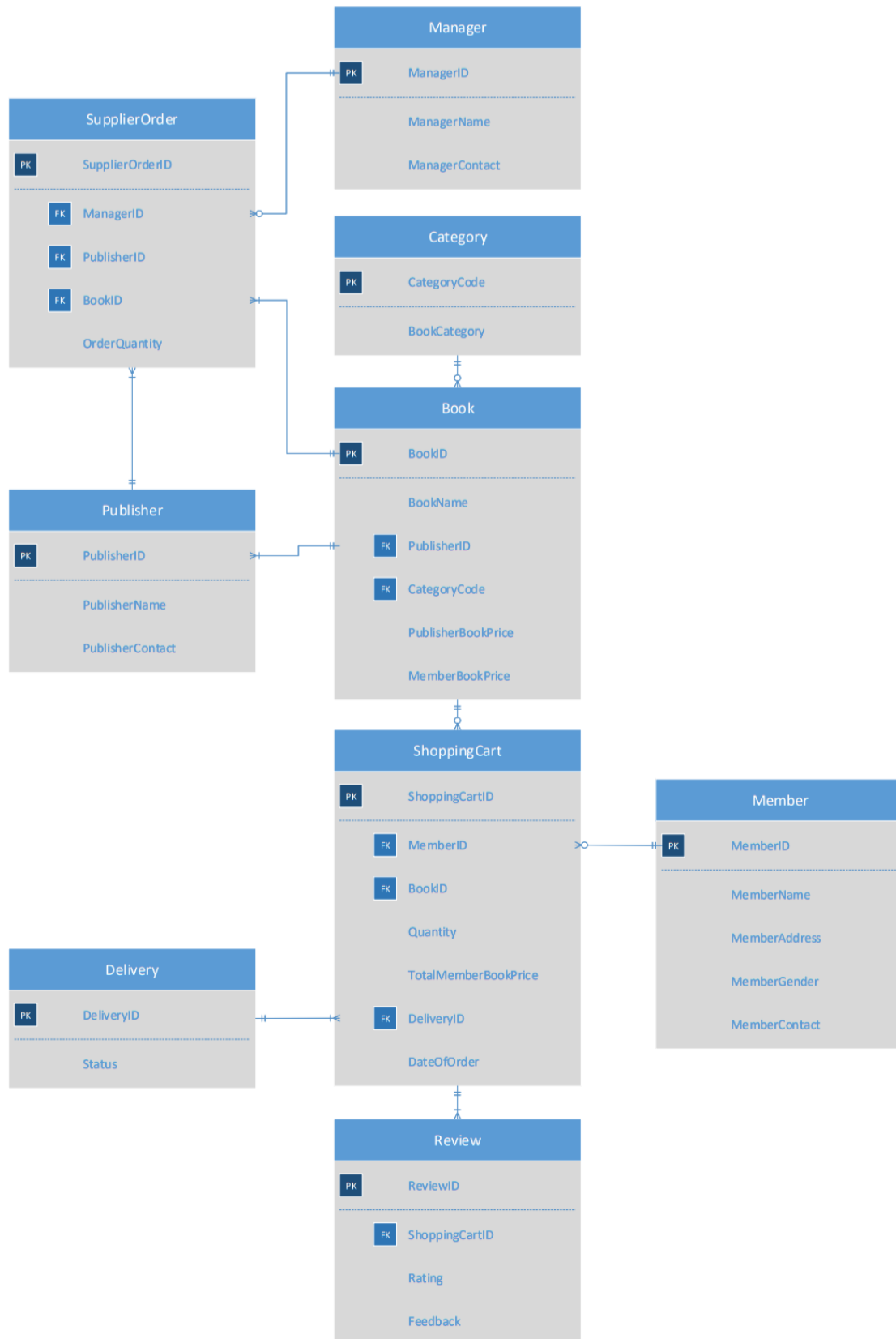
**Hand in date: 7<sup>th</sup> of June 2020**

## **Contents**

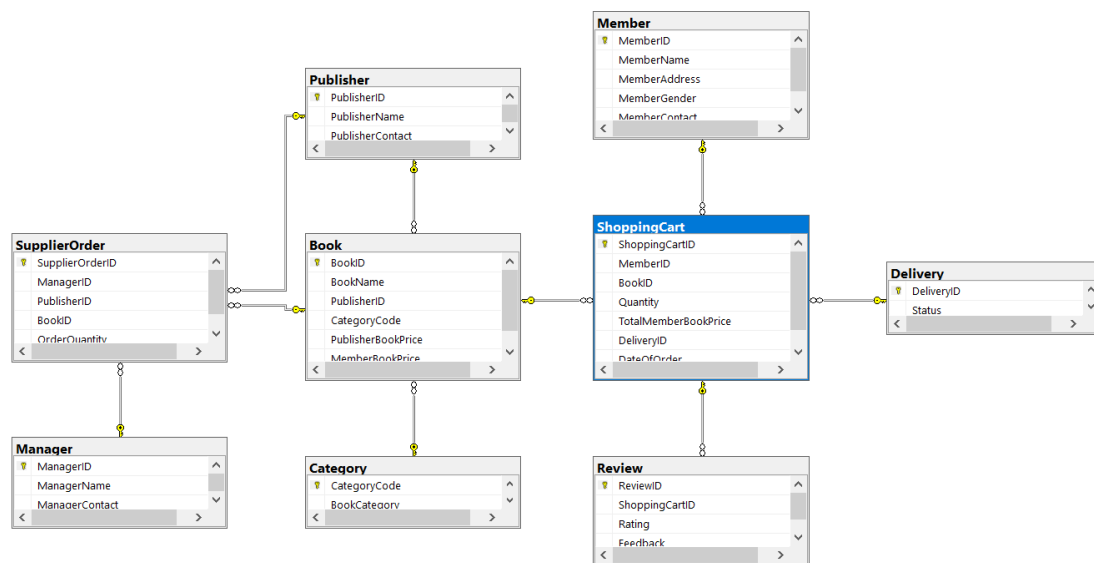
1.0 Database Schema .....	3
1.1 Finalised ERD .....	3
1.2 Database Diagram .....	4
2.0 Data Definition Language.....	5
2.1 Tables .....	5
2.2 Query Statements .....	9
3.0 Data Manipulation Language.....	14
4.0 Workload Matrix.....	24

## 1.0 Database Schema

## 1.1 Finalised ERD



## 1.2 Database Diagram



## 2.0 Data Definition Language

### 2.1 Tables

MSI\WEIXIN07.eBookstore - dbo.Book						
	BookID	BookName	PublisherID	CategoryC...	PublisherB...	MemberBo...
▶	03256	Secret of th...	P003	C002	36	45
	04221	Binary Light	P008	C009	14	20
	04454	Touch of W...	P004	C003	26	30
	04456	Clue of the ...	P001	C002	17	20
	04562	The Last Pic...	P005	C006	33	40
	04856	Zenith of H...	P009	C004	74	80
	04859	The Woman...	P008	C007	25	30
	05859	Made for Evil	P007	C008	66	69
	06565	2105: Refor...	P002	C003	48	55
	09855	Wet Wild W...	P002	C005	42	50
*	NULL	NULL	NULL	NULL	NULL	NULL

Table: Book

MSI\WEIXIN07.eBoo...re - dbo.Category		
	CategoryC...	BookCateg...
▶	C001	Horror
	C002	Thriller
	C003	Mystery
	C004	Engineering
	C005	Finance
	C006	Romance
	C007	Famillyt
	C008	Adventure
	C009	Comedy
	C010	Sci-Fi
*	NULL	NULL

Table: Category

MSI\WEIXIN07.eBoo...ore - dbo.Delivery		
	DeliveryID	Status
▶	1	Delivering
	2	Delivered
	3	Waiting Pick...
	4	Delayed
	5	Preparing
*	NULL	NULL

Table: Delivery

MSI\WEIXIN07.eBo...ore - dbo.Manager			
	ManagerID	ManagerN...	ManagerC...
▶	M001	Dwayne Joh...	174827847
	M002	Jason Stath...	164726499
	M003	Brian Oconn...	174927964
	M004	Vin Diesel	1646394833
	M005	Paul Walker	1664323655
	M006	Alexandra ...	1565567890
	M007	Scarlette J.	1234322650
	M008	Mia Khalifa	1234626590
	M009	Lexi Lore	1258462310
	M010	Milo Moire	1234456990
*	NULL	NULL	NULL

Table: Manager

MSI\WEIXIN07.eBo...tore - dbo.Member					
	MemberID	MemberNa...	MemberAd...	MemberGe...	MemberCo...
▶	TP012345	Siew Chong	Kuala Lump...	Female	132648594
	TP015546	Yousef Baraj...	Kemaman	Male	112563255
	TP016548	Clive Ford	Batu Pahat	Female	199665522
	TP035649	Daniella Par...	Puchong	Female	125486252
	TP041548	Patryk Singl...	Johor Bharu	Male	177885544
	TP045891	Haidar Hib...	Cheras	Female	199584465
	TP046589	Indiana Gates	Bandar Puteri	Female	145889655
	TP065842	Khalid Mcki...	Kampar	Female	144563215
	TP098546	Tyreese San...	Klang	Male	133665548
	TP099654	Cassidy Gilli...	Seremban	Female	115668599
*	NULL	NULL	NULL	NULL	NULL

Table: Member

	PublisherID	PublisherN...	PublisherC...
▶	P001	John Wick	1234567890
	P002	Nicki Minaj	1122334455
	P003	Snoop Dog	1876543210
	P004	Elison Mony...	1234543210
	P005	Elon Musk	1110909090
	P006	Bill Gates	1233645282
	P007	Jack Ma	1428792378
	P008	Mark Zucker...	1280812783
	P009	Tony Stark	1458736892
	P010	Bruce Wayne	1967391873
*	NULL	NULL	NULL

Table: Publisher

	ReviewID	ShoppingC...	Rating	Feedback
▶	001	1	5	Very good!
	002	2	0	N/A
	003	3	0	N/A
	004	4	0	N/A
	005	5	3	The size of t...
	006	6	4	Need more ...
	007	7	0	N/A
	008	8	3	OK
	009	9	0	N/A
	010	10	4	Interesting!
*	NULL	NULL	NULL	NULL

Table: Review

MSI\WEIXIN07.eBoo...dbo.ShoppingCart							
	ShoppingC...	MemberID	BookID	Quantity	TotalMemb...	DeliveryID	DateOfOrd...
▶	1	TP015546	09855	1	50	2	2020-04-27
	10	TP041548	03256	2	90	2	2020-05-25
	2	TP045891	04562	1	40	1	2020-04-28
	3	TP065842	06565	2	110	3	2020-04-29
	4	TP098546	04856	1	80	5	2020-04-29
	5	TP099654	09855	3	150	2	2020-04-30
	6	TP035649	04859	1	30	2	2020-05-02
	7	TP015546	04454	2	60	4	2020-05-02
	8	TP012345	03256	1	45	2	2020-05-05
	9	TP045891	09855	3	150	1	2020-05-10
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Table: ShoppingCart

MSI\WEIXIN07.eBoo...dbo.SupplierOrder					
	SupplierOr...	ManagerID	PublisherID	BookID	OrderQuan...
▶	1	M003	P003	03256	20
	10	M010	P005	04562	20
	2	M003	P002	06565	30
	3	M007	P009	04856	40
	4	M007	P002	09855	30
	5	M009	P005	04562	60
	6	M002	P008	04859	50
	7	M006	P004	04454	20
	8	M006	P003	03256	30
	9	M003	P007	05859	40
*	NULL	NULL	NULL	NULL	NULL

Table: SupplierOrder



## 2.2 Query Statements

---

```
CREATE DATABASE eBookstore

CREATE TABLE Publisher(
PublisherID nvarchar(10) NOT NULL PRIMARY KEY,
PublisherName nvarchar(100) NOT NULL,
PublisherContact int NOT NULL)

CREATE TABLE Manager(
ManagerID nvarchar(10) NOT NULL PRIMARY KEY,
ManagerName nvarchar(100) NOT NULL,
ManagerContact int NOT NULL)

CREATE TABLE Category(
CategoryCode nvarchar(5) NOT NULL PRIMARY KEY,
BookCategory nvarchar(50) NOT NULL)

CREATE TABLE Member(
MemberID nvarchar(8) NOT NULL PRIMARY KEY,
MemberName nvarchar(100) NOT NULL,
MemberAddress nvarchar(150) NOT NULL,
MemberGender nvarchar(6) NOT NULL,
MemberContact int NOT NULL)

CREATE TABLE Delivery(
DeliveryID nvarchar(10) NOT NULL PRIMARY KEY,
Status nvarchar(50) NOT NULL)

CREATE TABLE Book(
BookID nvarchar(10) NOT NULL PRIMARY KEY,
BookName nvarchar(100) NOT NULL,
PublisherID nvarchar(10) FOREIGN KEY REFERENCES Publisher(PublisherID),
CategoryCode nvarchar(5) FOREIGN KEY REFERENCES Category(CategoryCode),
PublisherBookPrice int NOT NULL,
MemberBookPrice int NOT NULL)

CREATE TABLE SupplierOrder(
SupplierOrderID nvarchar(10) NOT NULL PRIMARY KEY,
ManagerID nvarchar(10) FOREIGN KEY REFERENCES Manager(ManagerID),
PublisherID nvarchar(10) FOREIGN KEY REFERENCES Publisher(PublisherID),
BookID nvarchar(10) FOREIGN KEY REFERENCES Book(BookID),
OrderQuantity int NOT NULL)
```

```
CREATE TABLE ShoppingCart(  
ShoppingCartID nvarchar(10) NOT NULL PRIMARY KEY,  
MemberID nvarchar(8) FOREIGN KEY REFERENCES Member(MemberID),  
BookID nvarchar(10) FOREIGN KEY REFERENCES Book(BookID),  
Quantity int NOT NULL,  
TotalMemberBookPrice int NOT NULL,  
DeliveryID nvarchar(10) FOREIGN KEY REFERENCES Delivery(DeliveryID),  
DateOfOrder date NOT NULL)  
  
CREATE TABLE Review(  
ReviewID nvarchar(10) NOT NULL PRIMARY KEY,  
ShoppingCartID nvarchar(10) FOREIGN KEY REFERENCES ShoppingCart(ShoppingCartID),  
Rating int NOT NULL,  
Feedback nvarchar(200))  
  
Alter Authorization on Database::eBookstore To [sa];  
  
INSERT INTO Publisher VALUES  
( 'P001', 'John Wick', '01234567890'),  
( 'P002', 'Nicki Minaj', '01122334455'),  
( 'P003', 'Snoop Dog', '01876543210'),  
( 'P004', 'Elison Monyet', '01234543210'),  
( 'P005', 'Elon Musk', '01110909090'),  
( 'P006', 'Bill Gates', '01233645282'),  
( 'P007', 'Jack Ma', '01428792378'),  
( 'P008', 'Mark Zuckerberg', '01280812783'),  
( 'P009', 'Tony Stark', '01458736892'),  
( 'P010', 'Bruce Wayne', '01967391873')  
  
INSERT INTO Manager VALUES  
( 'M001', 'Dwayne Johnson', '0174827847'),  
( 'M002', 'Jason Statham', '0164726499'),  
( 'M003', 'Brian Oconnor', '0174927964'),  
( 'M004', 'Vin Diesel', '01646394833'),  
( 'M005', 'Paul Walker', '01664323655'),  
( 'M006', 'Alexandra Daddario', '01565567890'),  
( 'M007', 'Scarlette J.', '01234322650'),  
( 'M008', 'Mia Khalifa', '01234626590'),  
( 'M009', 'Lexi Lore', '01258462310'),  
( 'M010', 'Milo Moire', '01234456990')
```

```
INSERT INTO Category VALUES
```

```
('C001', 'Horror'),  
('C002', 'Thriller'),  
('C003', 'Mystery'),  
('C004', 'Engineering'),  
('C005', 'Finance'),  
('C006', 'Romance'),  
('C007', 'Famillyt'),  
('C008', 'Adventure'),  
('C009', 'Comedy'),  
('C010', 'Sci-Fi')
```

```
INSERT INTO Member VALUES
```

```
('TP012345', 'Siew Chong', 'Kuala Lumpur', 'Female', '0132648594'),  
('TP035649', 'Daniella Parkinson', 'Puchong', 'Female', '0125486252'),  
('TP045891', 'Haidar Hibbert', 'Cheras', 'Female', '0199584465'),  
('TP065842', 'Khalid Mckinney', 'Kampar', 'Female', '0144563215'),  
('TP041548', 'Patryk Singleton', 'Johor Bharu', 'Male', '0177885544'),  
('TP016548', 'Clive Ford', 'Batu Pahat', 'Female', '0199665522'),  
('TP098546', 'Tyreese Sanchez', 'Klang', 'Male', '0133665548'),  
('TP015546', 'Yousef Barajas', 'Kemaman', 'Male', '0112563255'),  
('TP046589', 'Indiana Gates', 'Bandar Puteri', 'Female', '0145889655'),  
('TP099654', 'Cassidy Gilliam', 'Seremban', 'Female', '0115668599')
```

```
INSERT INTO Delivery VALUES
```

```
('1', 'Delivering'),  
('2', 'Delivered'),  
('3', 'Waiting Pickup'),  
('4', 'Delayed'),  
('5', 'Preparing')
```

```
INSERT INTO Book VALUES
```

```
('03256', 'Secret of the Vanishing Lodger', 'P003', 'C002', '36', '45'),  
( '06565', '2105: Reformation', 'P002', 'C003', '48', '55'),  
( '09855', 'Wet Wild Winter', 'P002', 'C005', '42', '50'),  
( '04562', 'The Last Pickle', 'P005', 'C006', '33', '40'),  
( '05859', 'Made for Evil', 'P007', 'C008', '66', '69'),  
( '04856', 'Zenith of Hyperion', 'P009', 'C004', '74', '80'),  
( '04221', 'Binary Light', 'P008', 'C009', '14', '20'),  
( '04859', 'The Woman in the Window', 'P008', 'C007', '25', '30'),  
( '04454', 'Touch of Wyverns', 'P004', 'C003', '26', '30'),  
( '04456', 'Clue of the Crooked Staircase', 'P001', 'C002', '17', '20')
```

```
INSERT INTO SupplierOrder VALUES
```

```
('1', 'M003', 'P003', '03256', '20'),  
( '2', 'M003', 'P002', '06565', '30'),  
( '3', 'M007', 'P009', '04856', '40'),  
( '4', 'M007', 'P002', '09855', '30'),  
( '5', 'M009', 'P005', '04562', '60'),  
( '6', 'M002', 'P008', '04859', '50'),  
( '7', 'M006', 'P004', '04454', '20'),  
( '8', 'M006', 'P003', '03256', '30'),  
( '9', 'M003', 'P007', '05859', '40'),  
( '10', 'M010', 'P005', '04562', '20')
```

```
INSERT INTO ShoppingCart VALUES
```

```
('1', 'TP015546', '09855', '1', '50', '2', '27 April 2020'),  
( '2', 'TP045891', '04562', '1', '40', '1', '28 April 2020'),  
( '3', 'TP065842', '06565', '2', '110', '3', '29 April 2020'),  
( '4', 'TP098546', '04856', '1', '80', '5', '29 April 2020'),  
( '5', 'TP099654', '09855', '3', '150', '2', '30 April 2020'),  
( '6', 'TP035649', '04859', '1', '30', '2', '2 May 2020'),  
( '7', 'TP015546', '04454', '2', '60', '4', '2 May 2020'),  
( '8', 'TP012345', '03256', '1', '45', '2', '5 May 2020'),  
( '9', 'TP045891', '09855', '3', '150', '1', '10 May 2020'),  
( '10', 'TP041548', '03256', '2', '90', '2', '25 May 2020')
```

```
INSERT INTO ShoppingCart VALUES
('1','TP015546','09855','1','50','2','27 April 2020'),
('2','TP045891','04562','1','40','1','28 April 2020'),
('3','TP065842','06565','2','110','3','29 April 2020'),
('4','TP098546','04856','1','80','5','29 April 2020'),
('5','TP099654','09855','3','150','2','30 April 2020'),
('6','TP035649','04859','1','30','2','2 May 2020'),
('7','TP015546','04454','2','60','4','2 May 2020'),
('8','TP012345','03256','1','45','2','5 May 2020'),
('9','TP045891','09855','3','150','1','10 May 2020'),
('10','TP041548','03256','2','90','2','25 May 2020')
```

```
INSERT INTO Review VALUES
('001','1','5','Very good!'),
('002','2','0','N/A'),
('003','3','0','N/A'),
('004','4','0','N/A'),
('005','5','3','The size of the words too small'),
('006','6','4','Need more pictures'),
('007','7','0','N/A'),
('008','8','3','OK'),
('009','9','0','N/A'),
('010','10','4','Interesting!')
```

### **3.0 Data Manipulation Language**

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

```
--Q1 Faithlin
SELECT Book.BookID, Book.BookName, AVG(Review.Rating) AS 'Rating'
FROM Book LEFT JOIN ShoppingCart ON Book.BookID=ShoppingCart.BookID
LEFT JOIN Review ON ShoppingCart.ShoppingCartID=Review.ShoppingCartID
GROUP BY Book.BookID, Book.BookName
HAVING AVG(Review.Rating) = (SELECT MAX(R.Average)
FROM (SELECT Book.BookID, Book.BookName, AVG(Review.Rating) AS Average
FROM Book LEFT JOIN ShoppingCart ON Book.BookID=ShoppingCart.BookID
LEFT JOIN Review ON ShoppingCart.ShoppingCartID=Review.ShoppingCartID
GROUP BY Book.BookID, Book.BookName) AS R)
```

81 %

Results Messages

	BookID	BookName	Rating
1	04859	The Woman in the Window	4

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

```
--Q2 Kayson
SELECT MemberName, ShoppingCart.MemberID, COUNT(Review.Feedback) as 'Number of feedback per member'
FROM ShoppingCart LEFT JOIN Review ON ShoppingCart.ShoppingCartID = Review.ShoppingCartID LEFT JOIN Member on ShoppingCart.MemberID = Member.MemberID
WHERE Review.Rating <> 0 GROUP BY MemberName, ShoppingCart.MemberID
```

81 %

Results Messages

	MemberName	MemberID	Number of feedback per member
1	Siew Chong	TP012345	1
2	Yousef Barajas	TP015546	1
3	Daniella Parkinson	TP035649	1
4	Patryk Singleton	TP041548	1
5	Cassidy Gilliam	TP099654	1

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

```
--Q3 Hanns
SELECT Publisher.PublisherID, Publisher.PublisherName, COUNT(Book.BookID) as 'Number of books published'
FROM Publisher LEFT JOIN Book ON Publisher.PublisherID = Book.PublisherID
GROUP BY Publisher.PublisherID, Publisher.PublisherName
```

81 %

Results Messages

	PublisherID	PublisherName	Number of books published
1	P001	John Wick	1
2	P002	Nicki Minaj	2
3	P003	Snoop Dog	1
4	P004	Elison Monyet	1
5	P005	Elon Musk	1
6	P006	Bill Gates	0
7	P007	Jack Ma	1
8	P008	Mark Zuckerberg	2
9	P009	Tony Stark	1
10	P010	Bruce Wayne	0



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

```
--Q4 Remy
SELECT Publisher.PublisherName, SUM(SupplierOrder.OrderQuantity) as 'Total Number Of Books Ordered'
FROM Publisher LEFT JOIN SupplierOrder ON SupplierOrder.PublisherID = Publisher.PublisherID
GROUP BY Publisher.PublisherName
```

81 %

Results Messages

	PublisherName	Total Number Of Books Ordered
1	Bill Gates	NULL
2	Bruce Wayne	NULL
3	Elison Monyet	20
4	Elon Musk	80
5	Jack Ma	40
6	John Wick	NULL
7	Mark Zuckerberg	50
8	Nicki Minaj	60
9	Snoop Dog	50
10	Tony Stark	40

- v. Find the total number of books ordered by each member.

```
SELECT Member.MemberID, Member.MemberName, SUM(ShoppingCart.Quantity) As 'Total Books Ordered'
FROM Member LEFT JOIN ShoppingCart ON Member.MemberID = ShoppingCart.MemberID
GROUP BY Member.MemberID, Member.MemberName;
```

81 %

Results Messages

	MemberID	MemberName	Total Books Ordered
1	TP012345	Siew Chong	1
2	TP015546	Yousef Barajas	3
3	TP016548	Clive Ford	NULL
4	TP035649	Daniella Parkinson	1
5	TP041548	Patryk Singleton	2
6	TP045891	Haidar Hibbert	4
7	TP046589	Indiana Gates	NULL
8	TP065842	Khalid Mckinney	2
9	TP098546	Tyreese Sanchez	1
10	TP099654	Cassidy Gilliam	3

- vi. Find the bestselling book(s).

```
--Q6 Kayson
SELECT Book.BookName, SUM(ShoppingCart.Quantity) as 'Total Sold Quantity'
FROM Book INNER JOIN ShoppingCart ON ShoppingCart.BookID = Book.BookID GROUP BY Book.BookName
HAVING SUM(ShoppingCart.Quantity) = (SELECT MAX(Quant.SumQuant)
FROM (SELECT Book.BookName, SUM(ShoppingCart.Quantity) as SumQuant
FROM Book INNER JOIN ShoppingCart ON ShoppingCart.BookID = Book.BookID GROUP BY Book.BookName) AS Quant)
```

81 %

Results Messages

	BookName	Total Sold Quantity
1	Wet Wild Winter	7

- vii. 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).

```
--Q7 Hanns
SELECT COUNT(MemberID) as 'Total number of members',
COUNT(CASE WHEN MemberGender = 'Male' THEN MemberGender ELSE NULL END) as 'Total male member',
COUNT(CASE WHEN MemberGender = 'Female' THEN MemberGender ELSE NULL END) as 'Total female member' FROM Member
```

81 %

Results Messages

	Total number of members	Total male member	Total female member
1	10	3	7

- viii. 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.

```
--Q8 Remy
SELECT ShoppingCart.MemberID AS 'Member Identification Number', Member.MemberAddress AS 'Member Address', Member.MemberContact AS 'Member Contact Number',
ShoppingCart.BookID AS 'Book Serial Number', Book.BookName AS 'Book Name', ShoppingCart.Quantity AS 'Quantity', ShoppingCart.DateOfOrder as 'Date of Order', Delivery.Status as 'Delivery Status'
FROM ShoppingCart
INNER JOIN Member ON ShoppingCart.MemberID=Member.MemberID
INNER JOIN Delivery ON ShoppingCart.DeliveryID=Delivery.DeliveryID
INNER JOIN Book ON ShoppingCart.BookID=Book.BookID
WHERE Delivery.Status <> 'Delivered'
```

	Member Identification Number	Member Address	Member Contact Number	Book Serial Number	Book Name	Quantity	Date of Order	Delivery Status
1	TP045891	Cheras	199584465	04562	The Last Pickle	1	2020-04-28	Delivering
2	TP065842	Kampar	144563215	06565	2105: Reformation	2	2020-04-29	Waiting Pickup
3	TP098546	Klang	133665548	04856	Zenith of Hyperion	1	2020-04-29	Preparing
4	TP015546	Kemaman	112563255	04454	Touch of Wyverns	2	2020-05-02	Delayed
5	TP045891	Cheras	199584465	09855	Wet Wild Winter	3	2020-05-10	Delivering

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

```
--Q9 Hanns
SELECT Member.MemberID, Member.MemberName, SUM(ShoppingCart.TotalMemberBookPrice) as Total
FROM Member INNER JOIN ShoppingCart ON ShoppingCart.MemberID = Member.MemberID GROUP BY Member.MemberID, Member.MemberName
HAVING SUM(ShoppingCart.TotalMemberBookPrice) = (SELECT MAX(A.Summ)
FROM (SELECT Member.MemberID, Member.MemberName, SUM(ShoppingCart.TotalMemberBookPrice) as Summ
FROM Member INNER JOIN ShoppingCart ON ShoppingCart.MemberID = Member.MemberID GROUP BY Member.MemberID, Member.MemberName) AS A)
```

81 %

Results Messages

	MemberID	MemberName	Total
1	TP045891	Haidar Hibbert	190

- x. Show a list of total books as added by each members in the shopping cart.

```
--Q10 Remy
SELECT Member.MemberID AS 'Member Identification Number', SUM(ShoppingCart.Quantity) AS 'Total Books Added into Shopping Cart'
FROM ShoppingCart
RIGHT JOIN Member ON ShoppingCart.MemberID=Member.MemberID
GROUP BY Member.MemberID
```

81 %

Results Messages

	Member Identification Number	Total Books Added into Shopping Cart
1	TP012345	1
2	TP015546	3
3	TP016548	NULL
4	TP035649	1
5	TP041548	2
6	TP045891	4
7	TP046589	NULL
8	TP065842	2
9	TP098546	1
10	TP099654	3

## 4.0 Workload Matrix

### AICT005-4-1 Database Systems – Workload Matrix

<b>Part</b>	<b>Group Component</b>	<b>Student 1 Name: Faithlin Hoe Wei Xin</b>	<b>Student2 Name: Koh Tian Xin</b>	<b>Student 3 Name: Tan Chin Yuon</b>	<b>Student 4 Name: Siah Sung Hanns</b>	<b>Total</b>
<b>1</b>	a) Database and Database Management System	<b>Checking</b>	<b>Disadvantage of file-based system</b>	<b>Discussion</b>	<b>Advantage and Functions of DBMS</b>	<b>100 %</b>
<b>1</b>	b) Business Rules & Normalization	<b>Business Rule</b>	<b>Normalisation</b>	<b>Business Rule</b>	<b>Normalisation</b>	<b>100 %</b>
<b>1</b>	c) Entity Relationship Diagram	<b>Checking</b>	<b>Checking</b>	<b>Checking</b>	<b>ERD</b>	<b>100 %</b>
<b>2</b>	d) Database Schema	<b>Checking</b>	<b>Checking</b>	<b>Checking</b>	<b>-Database Diagram - ERD</b>	<b>100 %</b>

\*Individual component is omitted because marks will be awarded separately

\*Ignore student 4 column if there are only 3 members

\*Total % of contribution of all members must sum up to 100% (for each row)