Database Management Group Assignment Part 2

E-Library Management System

Module Code:

AICT015-4-1-DBM

Prepared by:

Group 8

1. Yap Xin Ling (L) TP077224

2. Yeo Pei Wen TP077057

3. Sim Tian TP077056

Module Lecturer:

Miss Tan Li June

Table of Content

1.0 Database Schema

Database schema is crucial in managing data, for instance, data integration, data validation, or data analysis (Frozza et al., 2021). Define database schema are vital to ensure data interoperability as data volumes are vary. Additionally, database schema creates a structured framework that can handle various data types and enable sophisticated queries for efficient information retrieval (Diène et al., 2020).

1.1 Entity Relationship Diagram

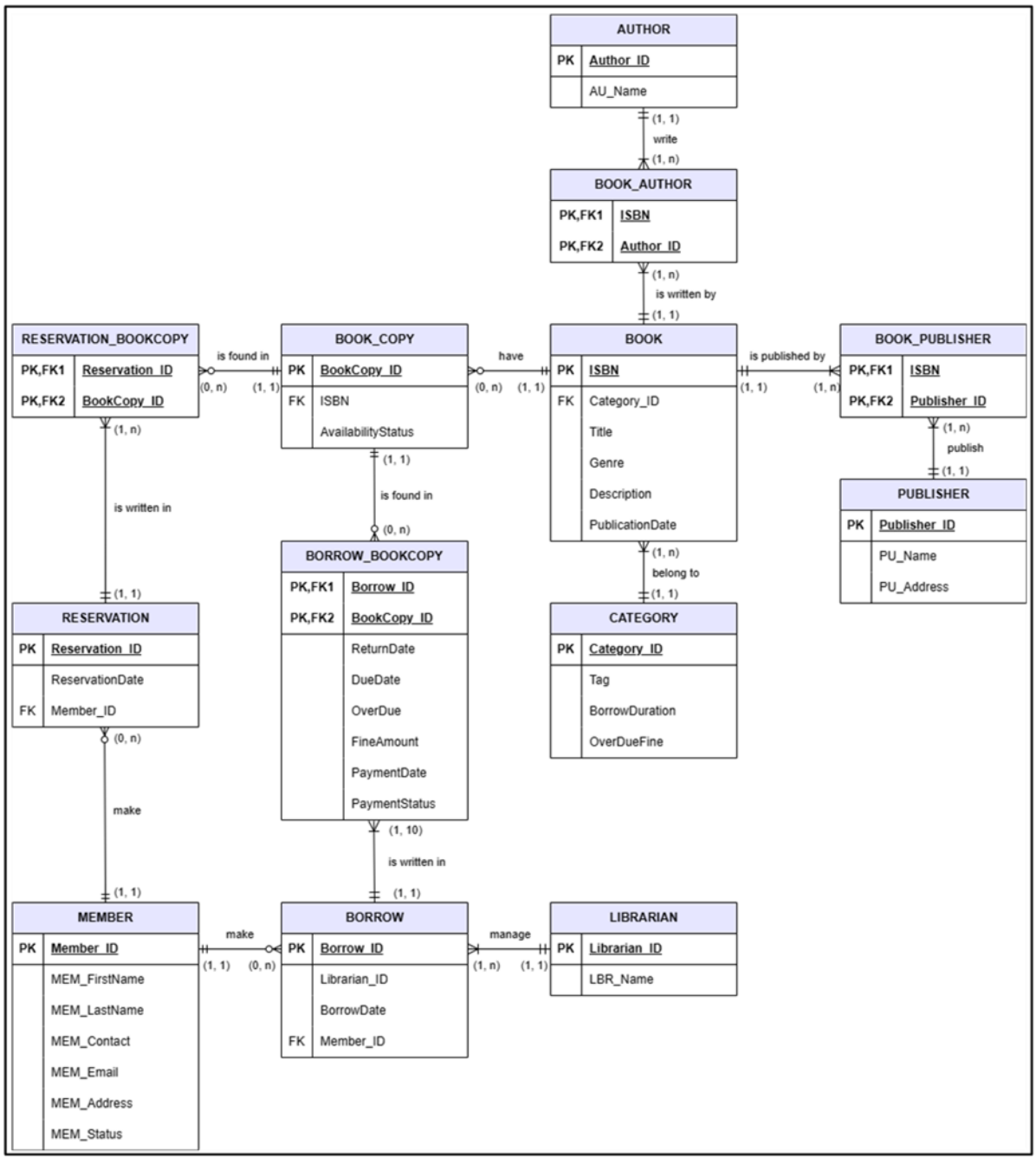

Figure 1: Entity Relationship Diagram (ERD) of the E-Library Management System

Figure 1 shows the ERD for the E-library Database Management System that clearly represents the complicated relationship between entities and attributes. Cardinality and connectivity are assigned to every relationship of all the entity. Entity-relationship diagram (ER Diagram) is used to visualise the interactions of components and relationships within a system are designed in the database design phases (Idrissov et al., 2020).

1.2 Database Diagram

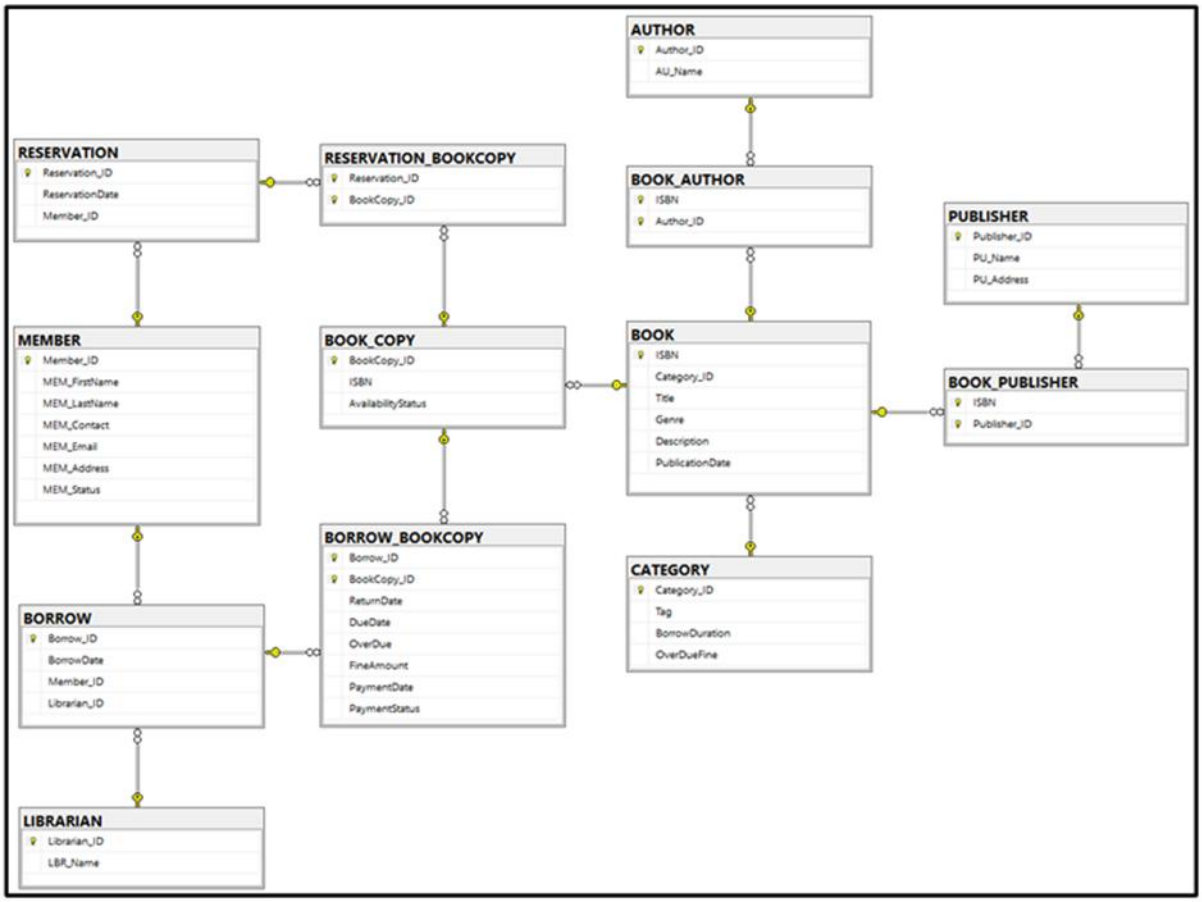

Figure 2: Database Diagram of the E-Library Management System

Figure 2 illustrates the database diagram for the E-library Database Management System. The database diagram should be defined in accordance with the requirements of the library as well as establish constraints to control how the fields in the table relates to one another (Eyada et al., 2020).

2.0 SQL – Data Definition Language (DDL)

Data Definition Language (DDL) is an essential component of Structured Query Language (SQL) as it enables users to create database itself and database objects, such as tables, indexes, and view (Rockoff, 2021). For instance, DDL provides ALTER commands that enable users to modify the design of tables in the database. Moreover, DDL help to define access rights to those database objects.

2.1 Create Database

```
CREATE DATABASE LIBRARY_MANAGEMENT_SYSTEM
```

Figure 3: Query for Creating the LIBRARY_MANAGEMENT_SYSTEM Database

2.2 Create Tables

2.2.1 MEMBER Table

```
CREATE TABLE MEMBER (
Member_ID nvarchar(50) PRIMARY KEY,
MEM_FirstName nvarchar(50),
MEM_LastName nvarchar(50),
MEM_Contact nvarchar(15),
MEM_Email nvarchar(100),
MEM_Address nvarchar(200),
MEM_Status varchar(10)
)
```

Figure 4: Query for Creating the MEMBER Table

2.2.2 LIBRARIAN Table

```
CREATE TABLE LIBRARIAN (
Librarian_ID nvarchar(50) PRIMARY KEY,
LBR_Name nvarchar(50)
)
```

Figure 5: Query for Creating the LIBRARIAN Table

2.2.3 AUTHOR Table

```
CREATE TABLE AUTHOR (
Author_ID nvarchar(50) PRIMARY KEY,
AU_Name nvarchar(50)
)
```

Figure 6: Query for Creating the AUTHOR Table

2.2.4 PUBLISHER Table

```
CREATE TABLE PUBLISHER(
Publisher_ID nvarchar(50) PRIMARY KEY,
PU_Name nvarchar(50),
PU_Address nvarchar(150)
)
```

Figure 7: Query for Creating the PUBLISHER Table

2.2.5 CATEGORY Table

```
CREATE TABLE CATEGORY(
Category_ID nvarchar(50) PRIMARY KEY,
Tag nvarchar(50),
BorrowDuration int,
OverDueFine decimal(10,2)
)
```

Figure 8: Query for Creating the CATEGORY Table

2.2.6 BOOK Table

```
CREATE TABLE BOOK(
ISBN nvarchar(50) PRIMARY KEY,
Category_ID nvarchar(50) FOREIGN KEY REFERENCES Category(Category_ID),
Title nvarchar(100),
Genre nvarchar(50),
Description nvarchar(800),
PublicationDate date
)
```

Figure 9: Query for Creating the BOOK Table

2.2.7 BOOK_COPY Table

```
CREATE TABLE BOOK_COPY (
BookCopy_ID nvarchar(50) PRIMARY KEY,
ISBN nvarchar(50) FOREIGN KEY REFERENCES BOOK(ISBN),
AvailabilityStatus nvarchar(10)
)
```

Figure 10: Query for Creating the BOOK_COPY Table

2.2.8 BORROW Table

```
CREATE TABLE BORROW (
Borrow_ID nvarchar(50) PRIMARY KEY,
BorrowDate date,
Member_ID nvarchar(50) FOREIGN KEY REFERENCES MEMBER(Member_ID),
Librarian_ID nvarchar(50) FOREIGN KEY REFERENCES LIBRARIAN(Librarian_ID)
)
```

Figure 11: Query for Creating the BORROW Table

2.2.9 BORROW_BOOKCOPY Table

```
CREATE TABLE BORROW_BOOKCOPY (
Borrow_ID nvarchar(50) FOREIGN KEY REFERENCES BORROW(Borrow_ID),
BookCopy_ID nvarchar(50) FOREIGN KEY REFERENCES BOOK_COPY(BookCopy_ID),
ReturnDate date,
DueDate date,
OverDue int,
FineAmount decimal(10, 2),
PaymentDate date,
PaymentStatus nvarchar(10),
PRIMARY KEY (Borrow_ID, BookCopy_ID)
)
```

Figure 12: Query for Creating the BORROW_BOOKCOPY Table

2.2.10 RESERVATION Table

```
CREATE TABLE RESERVATION (
Reservation_ID nvarchar(50) PRIMARY KEY,
ReservationDate date,
Member_ID nvarchar(50) FOREIGN KEY REFERENCES MEMBER(Member_ID)
)
```

Figure 13: Query for Creating the RESERVATION Table

2.2.11 RESERVATION_BOOKCOPY Table

```
CREATE TABLE RESERVATION_BOOKCOPY (
Reservation_ID nvarchar(50) FOREIGN KEY REFERENCES RESERVATION(Reservation_ID),
BookCopy_ID nvarchar(50) FOREIGN KEY REFERENCES BOOK_COPY(BookCopy_ID),
PRIMARY KEY (Reservation_ID, BookCopy_ID)
)
```

Figure 14: Query for Creating the RESERVATION_BOOKCOPY Table

2.2.12 BOOK_PUBLISHER Table

```
CREATE TABLE BOOK_PUBLISHER (
ISBN nvarchar(50) FOREIGN KEY REFERENCES BOOK(ISBN),
Publisher_ID nvarchar(50) FOREIGN KEY REFERENCES PUBLISHER(Publisher_ID),
PRIMARY KEY (ISBN, Publisher_ID)
)
```

Figure 15: Query for Creating the BOOK_PUBLISHER Table

2.2.13 BOOK_AUTHOR Table

```
CREATE TABLE BOOK_PUBLISHER (
ISBN nvarchar(50) FOREIGN KEY REFERENCES BOOK(ISBN),
Publisher_ID nvarchar(50) FOREIGN KEY REFERENCES PUBLISHER(Publisher_ID),
PRIMARY KEY (ISBN, Publisher_ID)
)
```

Figure 16: Query for Creating the BOOK_AUTHOR Table

3.0 SQL – Data Manipulation Language (DML)

Data Manipulation Language (DML) is also an indispensable component of the SQL. The reason for this is that DML allows users to retrieve, update, add, or delete data in a database (Rockoff, 2021). DML includes commands to obtain and modify data within the database.

3.1 Inserting Data into Table

3.1.1 MEMBER Table

```
INSERT INTO MEMBER
(Member_ID, MEM_FirstName, MEM_LastName, MEM_Contact, MEM_Email, MEM_Address, MEM_Status)
VALUES
('M001', 'Andy', 'Lau', '0123456789', 'andy.lau3@gmail.com', 'Unit A-5-03, Taman Sri Endah Apartment, 57000 Sri Petaling, Kuala Lumpur', 'Active'),
('M002', 'Angelina', 'Jolie', '0112345678', 'angelina.jolie45@gmail.com', 'No.29, Taman Limbongan Indah, 75260 Melaka', 'Active'),
('M003', 'Beyonce', 'Knowles', '0198765432', 'beyonce789@gmail.com', 'Unit B-28-18 Jalan Kuching, Tuju Residences, 51200 Kuala Lumpur', 'Active'),
('M004', 'Jennie', 'Kim', '0134567890', 'jennierubyjane@gmail.com', 'No.6, Taman Kingfisher, 88400 Kota Kinabalu, Sabah', 'Active'),
('M005', 'Lalisa', 'Manobal', '0145678901', 'lalalalisa_m@gmail.com', 'No.8, Taman Desa Impian 2, Alma, 14000 Bukit Mertajam, Penang', 'Active'),
('M006', 'Jisoo', 'Kim', '0167890123', 'sooyaaa@gmail.com', 'No.31, Jalan Harmonium, Taman Desa Tebrau, 81100 Tebrau, Johor', 'Active'),
('M007', 'Rose', 'Park', '0178901234', 'roses_are_rosie@gmail.com', 'Unit F-9-10, Pangsapuri Sutera, Jalan Sutera Dahlia, 80542 Johor Bahru, Johor', 'Active'),
('M008', 'Taylor', 'Swift', '0189012345', 'taylor_swift@gmail.com', 'No.54, Jalan Harmoni, 50450 Kuala Lumpur', 'Active'),
('M009', 'Jack', 'Ma', '0156789012', 'jack.ma404@gmail.com', 'No.67, Taman Alfa Bator Jelawat, 16020 Bachok, Kelantan', 'Active'),
('M010', 'Stephen', 'Chow', '0109876543', 'stephen.chow62@gmail.com', 'No.51, Taman Rambai Utama, Jalan Raja 18, 75450 Melaka Tengah, Melaka', 'Active'),
```

Figure 17: Query for Inserting Data into MEMBER Table

	Member_ID	MEM_FirstName	MEM_LastName	MEM_Contact	MEM_Email	MEM_Address	MEM_Status
1	M001	Andy	Lau	0123456789	andy.lau3@gmail.com	Unit A-5-03, Taman Sri Endah Apartment, 57000 Sri Pe...	Active
2	M002	Angelina	Jolie	0112345678	angelina.jolie45@gmail.com	No.29, Taman Limbongan Indah, 75260 Melaka	Active
3	M003	Beyonce	Knowles	0198765432	beyonce789@gmail.com	Unit B-28-18 Jalan Kuching, Tuju Residences, 51200 K...	Active
4	M004	Jennie	Kim	0134567890	jennierubyjane@gmail.com	No.6, Taman Kingfisher, 88400 Kota Kinabalu, Sabah	Active
5	M005	Lalisa	Manobal	0145678901	lalalalisa_m@gmail.com	No.8, Taman Desa Impian 2, Alma, 14000 Bukit Mertaj...	Active
6	M006	Jisoo	Kim	0167890123	sooyaaa@gmail.com	No.31, Jalan Harmonium, Taman Desa Tebrau, 81100 ...	Active
7	M007	Rose	Park	0178901234	roses_are_rosie@gmail.com	Unit F-9-10, Pangsapuri Sutera, Jalan Sutera Dahlia, 8...	Active
8	M008	Taylor	Swift	0189012345	taylor_swift@gmail.com	No.54, Jalan Harmoni, 50450 Kuala Lumpur	Active
9	M009	Jack	Ma	0156789012	jack.ma404@gmail.com	No.67, Taman Alfa Bator Jelawat, 16020 Bachok, Kelan...	Active
10	M010	Stephen	Chow	0109876543	stephen.chow62@gmail.com	No.51, Taman Rambai Utama, Jalan Raja 18, 75450 ...	Active

Table 1: MEMBER

3.1.2 LIBRARIAN Table

```
INSERT INTO LIBRARIAN
(Librarian_ID, LBR_Name)
VALUES
('L_001', 'Eleanor Vance'),
('L_002', 'Mateo Rivera'),
('L_003', 'Evelyn Wright'),
('L_004', 'Yao Shi Yi'),
('L_005', 'Noah Green')
```

Figure 18: Query for Inserting Data into LIBRARIAN Table

	Librarian_ID	LBR_Name
1	L_001	Eleanor Vance
2	L_002	Mateo Rivera
3	L_003	Evelyn Wright
4	L_004	Yao Shi Yi
5	L_005	Noah Green

Table 2: LIBRARIAN

3.1.3 AUTHOR Table

```
INSERT INTO AUTHOR
(Author_ID, AU_Name)
VALUES
('A001', 'J.K. Rowling'),
('A002', 'Stephen Meyer'),
('A003', 'Suzanne Collins'),
('A004', 'Roald Dahl'),
('A005', 'Rick Riordan'),
('A006', 'Veronica Roth'),
('A007', 'Ransom Riggs'),
('A008', 'Richard Howard'),
('A009', 'Sophie Lark'),
('A010', 'Charles Addams'),
```

Figure 19: Query for Inserting Data into AUTHOR Table

	Author_ID	AU_Name
1	A001	J.K. Rowling
2	A002	Stephen Meyer
3	A003	Suzanne Collins
4	A004	Roald Dahl
5	A005	Rick Riordan
6	A006	Veronica Roth
7	A007	Ransom Riggs
8	A008	Richard Howard
9	A009	Sophie Lark
10	A010	Charles Addams

Table 3: AUTHOR

3.1.4 PUBLISHER Table

```
INSERT INTO PUBLISHER
(Publisher_ID, PU_Name, PU_Address)
VALUES
('P001', 'Company Bloomsbury', '4, Jalan Pemaju 3, Hicom-glenmarie Industrial Park, 40150 Shah Alam, Selangor'),
('P002', 'Company Woodsland', '12, Jalan DBP, Dolomite Business Park, 68100 Batu Caves, Selangor'),
('P003', 'Company Murcott', 'No. 6, Petaling Jaya, Malaysia, Jalan 51/217, Seksyen 51, 46050 Petaling Jaya, Selangor'),
('P004', 'Company Willington', '63, Lrg Maarof, Bangsar, 59000 Kuala Lumpur, Wilayah Persekutuan Kuala Lumpur'),
('P005', 'Company Osteria', 'No 7-2, Tingkat, 2, Jln Seri Rejang 1, Taman Sri Rampai, 53300 Kuala Lumpur, Federal Territory of Kuala Lumpur'),
('P006', 'Asia Pacific University', 'Jalan Teknologi 5, Taman Teknologi Malaysia, 57000 Kuala Lumpur, Wilayah Persekutuan Kuala Lumpur')
```

Figure 20: Query for Inserting Data into PUBLISHER Table

	Publisher_ID	PU_Name	PU_Address
1	P001	Company Bloomsbury	4, Jalan Pemaju 3, Hicom-glenmarie Industrial Park...
2	P002	Company Woodsland	12, Jalan DBP, Dolomite Business Park, 68100 Batu...
3	P003	Company Murcott	No. 6, Petaling Jaya, Malaysia, Jalan 51/217, Seksy...
4	P004	Company Willington	63, Lrg Maarof, Bangsar, 59000 Kuala Lumpur, Wila...
5	P005	Company Osteria	No 7-2, Tingkat, 2, Jln Seri Rejang 1, Taman Sri Ra...
6	P006	Asia Pacific University	Jalan Teknologi 5, Taman Teknologi Malaysia, 570...

Table 4: PUBLISHER

3.1.5 CATEGORY Table

```
INSERT INTO CATEGORY
(Category_ID, Tag, BorrowDuration, OverDueFine)
VALUES
('C01', 'Yellow', 3, 1),
('C02', 'Red', 1, 3),
('C03', 'Green', 15, 0.5),
('C04', 'Black', 0, 0)
```

Figure 21: Query for Inserting Data into CATEGORY Table

	Category_ID	Tag	BorrowDuration	OverDueFine
1	C01	Yellow	3	1.00
2	C02	Red	1	3.00
3	C03	Green	15	0.50
4	C04	Black	0	0.00

Table 5: CATEGORY

3.1.6 BOOK Table

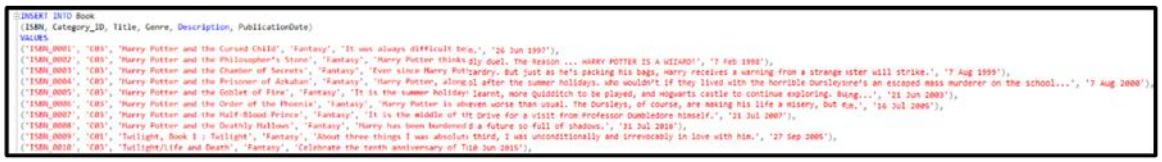

Figure 22: Query for Inserting Data into BOOK Table

	ISBN	Category_ID	Title	Genre	Description	PublicationDate
1	ISBN_0001	C03	Harry Potter and the Cursed Child	Fantasy	It was always difficult being Harry Potter, and it isn't much...	1997-06-26
2	ISBN_0002	C03	Harry Potter and the Philosopher's Stone	Fantasy	Harry Potter thinks he is an ordinary boy - until he is rescu...	1998-02-07
3	ISBN_0003	C03	Harry Potter and the Chamber of Secrets	Fantasy	Ever since Harry Potter had come home for the summer, ...	1999-08-07
4	ISBN_0004	C03	Harry Potter and the Prisoner of Azkaban	Fantasy	Harry Potter, along with his best friends, Ron and Hermio...	2000-08-07
5	ISBN_0005	C03	Harry Potter and the Goblet of Fire	Fantasy	It is the summer holidays and soon Harry Potter will be st...	2003-06-21
6	ISBN_0006	C03	Harry Potter and the Order of the Phoenix	Fantasy	Harry Potter is about to start his fifth year at Hogwarts Sch...	2005-07-16
7	ISBN_0007	C03	Harry Potter and the Half-Blood Prince	Fantasy	It is the middle of the summer, but there is an unseasona...	2007-07-21
8	ISBN_0008	C03	Harry Potter and the Deathly Hallows	Fantasy	Harry has been burdened with a dark, dangerous and se...	2016-07-31
9	ISBN_0009	C01	Twilight, Book 1 : Twilight	Fantasy	About three things I was absolutely positive. First, Edward...	2005-09-27
10	ISBN_0010	C03	Twilight/Life and Death	Fantasy	Celebrate the tenth anniversary of Twilight! This special d...	2015-06-10

Table 6: BOOK

3.1.7 BOOK_COPY Table

```
INSERT INTO BOOK_COPY
(BookCopy_ID, ISBN, AvailabilityStatus)
VALUES
('BC_0001', 'ISBN_0001', 'Available'),
('BC_0002', 'ISBN_0001', 'Available'),
('BC_0003', 'ISBN_0001', 'Available'),
('BC_0004', 'ISBN_0002', 'Available'),
('BC_0005', 'ISBN_0002', 'Available'),
('BC_0006', 'ISBN_0002', 'On Loan'),
('BC_0007', 'ISBN_0003', 'On Loan'),
('BC_0008', 'ISBN_0003', 'Available'),
('BC_0009', 'ISBN_0003', 'On Loan'),
('BC_0010', 'ISBN_0004', 'Available'),
```

Figure 23: Query for Inserting Data into BOOK_COPY Table

	BookCopy_ID	ISBN	AvailabilityStatus
1	BC_0001	ISBN_0001	Available
2	BC_0002	ISBN_0001	Available
3	BC_0003	ISBN_0001	Available
4	BC_0004	ISBN_0002	Available
5	BC_0005	ISBN_0002	Available
6	BC_0006	ISBN_0002	On Loan
7	BC_0007	ISBN_0003	On Loan
8	BC_0008	ISBN_0003	Available
9	BC_0009	ISBN_0003	On Loan
10	BC_0010	ISBN_0004	Available

Table 7: BOOK_COPY

3.1.8 BORROW Table

```
INSERT INTO BORROW
(Borrow_ID, BorrowDate, Member_ID, Librarian_ID)
VALUES
('B001', '20 Oct 2023', 'M001', 'L_002'),
('B002', '20 Oct 2023', 'M002', 'L_002'),
('B003', '20 Oct 2023', 'M010', 'L_003'),
('B004', '20 Oct 2023', 'M005', 'L_002'),
('B005', '26 Oct 2023', 'M048', 'L_002'),
('B006', '26 Oct 2023', 'M033', 'L_001'),
('B007', '26 Oct 2023', 'M028', 'L_001'),
('B008', '30 Oct 2023', 'M003', 'L_003'),
('B009', '30 Oct 2023', 'M021', 'L_002'),
('B010', '30 Oct 2023', 'M024', 'L_001'),
```

Figure 24: Query for Inserting Data into BORROW Table

	Borrow_ID	BorrowDate	Member_ID	Librarian_ID
1	B001	2023-10-20	M001	L_002
2	B002	2023-10-20	M002	L_002
3	B003	2023-10-20	M010	L_003
4	B004	2023-10-20	M005	L_002
5	B005	2023-10-26	M048	L_002
6	B006	2023-10-26	M033	L_001
7	B007	2023-10-26	M028	L_001
8	B008	2023-10-30	M003	L_003
9	B009	2023-10-30	M021	L_002
10	B010	2023-10-30	M024	L_001

Table 8: BORROW

3.1.9 BORROW_BOOKCOPY Table

```
INSERT INTO BORROW_BOOKCOPY
(Borrow_ID, BookCopy_ID, ReturnDate, DueDate, OverDue, FineAmount, PaymentDate, PaymentStatus)
VALUES
('B001', 'BC_0001', '2023-11-05', '2023-11-04', 1,  0.5, '2023-11-05', 'Paid'),
('B001', 'BC_0004', '2023-11-05', '2023-11-04', 1,  0.5, '2023-11-05', 'Paid'),
('B001', 'BC_0018', '2023-11-05', '2023-11-04', 1,  0.5, '2023-11-05', 'Paid'),
('B002', 'BC_0008', '2023-11-04', '2023-11-04', 0,  0, NULL, 'No Fee'),
('B002', 'BC_0022', '2023-11-04', '2023-11-04', 0,  0, NULL, 'No Fee'),
('B002', 'BC_0038', '2023-11-04', '2023-11-04', 0,  0, NULL, 'No Fee'),
('B003', 'BC_0051', '2023-11-04', '2023-11-04', 0,  0, NULL, 'No Fee'),
('B003', 'BC_0042', '2023-11-05', '2023-11-04', 1,  0.5, '2023-11-05', 'Paid'),
('B004', 'BC_0029', '2023-11-04', '2023-11-04', 0,  0, NULL, 'No Fee'),
('B004', 'BC_0035', '2023-11-05', '2023-11-04', 1,  0.5, '2023-11-05', 'Paid'),
```

Figure 25: Query for Inserting Data into BORROW_BOOCOPY Table

	Borrow_ID	BookCopy_ID	ReturnDate	DueDate	OverDue	FineAmount	PaymentDate	PaymentStatus
1	B001	BC_0001	2023-11-05	2023-11-04	1	0.50	2023-11-05	Paid
2	B001	BC_0004	2023-11-05	2023-11-04	1	0.50	2023-11-05	Paid
3	B001	BC_0018	2023-11-05	2023-11-04	1	0.50	2023-11-05	Paid
4	B002	BC_0008	2023-11-04	2023-11-04	0	0.00	NULL	No Fee
5	B002	BC_0022	2023-11-04	2023-11-04	0	0.00	NULL	No Fee
6	B002	BC_0038	2023-11-04	2023-11-04	0	0.00	NULL	No Fee
7	B003	BC_0042	2023-11-05	2023-11-04	1	0.50	2023-11-05	Paid
8	B003	BC_0051	2023-11-04	2023-11-04	0	0.00	NULL	No Fee
9	B004	BC_0029	2023-11-04	2023-11-04	0	0.00	NULL	No Fee
10	B004	BC_0035	2023-11-05	2023-11-04	1	0.50	2023-11-05	Paid

Table 9: BORROW_BOOKCOPY

3.1.10 RESERVATION Table

```
INSERT INTO RESERVATION
(Reservation_ID, ReservationDate, Member_ID)
VALUES
('R001', '28 Nov 2023', 'M032'),
('R002', '3 Dec 2023', 'M044'),
('R003', '23 Feb 2024', 'M025'),
('R004', '8 Mar 2024', 'M045'),
('R005', '10 Mar 2024', 'M037'),
('R006', '11 Mar 2024', 'M047')
```

Figure 26: Query for Inserting Data into RESERVATION Table

	Reservation_ID	ReservationDate	Member_ID
1	R001	2023-11-28	M032
2	R002	2023-12-03	M044
3	R003	2024-02-23	M025
4	R004	2024-03-08	M045
5	R005	2024-03-10	M037
6	R006	2024-03-11	M047

Table 10: RESERVATION

3.1.11 RESERVATION_BOOKCOPY

```
INSERT INTO RESERVATION_BOOKCOPY
(Reservation_ID, BookCopy_ID)
VALUES
('R001', 'BC_0009'),
('R002', 'BC_0014'),
('R003', 'BC_0034'),
('R004', 'BC_0063'),
('R005', 'BC_0098'),
('R006', 'BC_0007')
```

Figure 27: Query for Inserting Data into RESERVATION_BOOKCOPY Table

	Reservation_ID	BookCopy_ID
1	R001	BC_0009
2	R002	BC_0014
3	R003	BC_0034
4	R004	BC_0063
5	R005	BC_0098
6	R006	BC_0007

Table 11: RESERVATION_BOOKCOPY

3.1.12 BOOK_PUBLISHER Table

```
INSERT INTO BOOK_PUBLISHER
(ISBN, Publisher_ID)
VALUES
('ISBN_0001', 'P001'),
('ISBN_0002', 'P001'),
('ISBN_0003', 'P001'),
('ISBN_0004', 'P001'),
('ISBN_0005', 'P001'),
('ISBN_0006', 'P001'),
('ISBN_0007', 'P001'),
('ISBN_0008', 'P001'),
('ISBN_0009', 'P002'),
('ISBN_0010', 'P002'),
```

Figure 28: Query for Inserting Data into BOOK_PUBLISHER Table

	ISBN	Publisher_ID
1	ISBN_0001	P001
2	ISBN_0002	P001
3	ISBN_0003	P001
4	ISBN_0004	P001
5	ISBN_0005	P001
6	ISBN_0006	P001
7	ISBN_0007	P001
8	ISBN_0008	P001
9	ISBN_0009	P002
10	ISBN_0010	P002

Table 12: BOOK_PUBLISHER

3.1.13 BOOK_AUTHOR Table

```
INSERT INTO BOOK_AUTHOR
(ISBN, Author_ID)
VALUES
('ISBN_0001', 'A001'),
('ISBN_0002', 'A001'),
('ISBN_0003', 'A001'),
('ISBN_0004', 'A001'),
('ISBN_0005', 'A001'),
('ISBN_0006', 'A001'),
('ISBN_0007', 'A001'),
('ISBN_0008', 'A001'),
('ISBN_0009', 'A002'),
('ISBN_0010', 'A002');
```

Figure 29: Query for Inserting Data into BOOK_AUTHOR Table

	ISBN	Author_ID
1	ISBN_0001	A001
2	ISBN_0002	A001
3	ISBN_0003	A001
4	ISBN_0004	A001
5	ISBN_0005	A001
6	ISBN_0006	A001
7	ISBN_0007	A001
8	ISBN_0008	A001
9	ISBN_0009	A002
10	ISBN_0010	A002

Table 13: BOOK_AUTHOR

3.2 Create Queries

1. List all the members' first name, last name, and total number of books that are currently on loan. The results are sorted in alphabetical order of members' first name.

Query:

```
SELECT MEM_FirstName, MEM_LastName, COUNT(*) AS TotalOnLoanBook FROM Borrow
JOIN MEMBER ON MEMBER.Member_ID = BORROW.Member_ID
JOIN BORROW_BOOKCOPY ON BORROW_BOOKCOPY.Borrow_ID = BORROW.Borrow_ID
WHERE ReturnDate IS NULL
GROUP BY MEM_FirstName, MEM_LastName
ORDER BY MEM_FirstName ASC
```

Result Table:

Results Messages

	MEM_FirstName	MEM_LastName	TotalOnLoanBook
1	Andy	Lau	1
2	Anna	Nordstrom	1
3	Elsa	Nordstrom	1
4	Lalisa	Manobal	1
5	Tom	Holland	1
6	Victoria	Song	1
7	Will	Smith	1
8	Young Bae	Dong	1

2. Display the active members and list of books they have borrowed in year 2023. The result must consist of members' first name, last name, book name, and the borrow date. Sort the result by borrow date in ascending order.

Query:

```
SELECT MEM_FirstName, MEM_LastName, Title, BorrowDate FROM BORROW
JOIN MEMBER ON BORROW.Member_ID = MEMBER.Member_ID
JOIN BORROW_BOOKCOPY ON BORROW.Borrow_ID = BORROW_BOOKCOPY.Borrow_ID
JOIN BOOK_COPY ON BORROW_BOOKCOPY.BookCopy_ID = BOOK_COPY.BookCopy_ID
JOIN BOOK ON BOOK_COPY.ISBN = BOOK.ISBN
WHERE MEM_Status = 'Active' AND YEAR(BorrowDate) = '2023'
ORDER BY BorrowDate ASC
```

Result Table:

Results Messages

	MEM_FirstName	MEM_LastName	Title	BorrowDate
1	Andy	Lau	Harry Potter and the Cursed Child	2023-10-20
2	Andy	Lau	Harry Potter and the Philosopher's Stone	2023-10-20
3	Andy	Lau	Harry Potter and the Order of the Phoenix	2023-10-20
4	Angelina	Jolie	Harry Potter and the Chamber of Secrets	2023-10-20
5	Angelina	Jolie	Harry Potter and the Deathly Hallows	2023-10-20
6	Angelina	Jolie	The Ballad of Songbirds and Snakes	2023-10-20
7	Stephen	Chow	Catching Fire	2023-10-20
8	Stephen	Chow	The Titan's Curse	2023-10-20
9	Lalisa	Manobal	New Moon	2023-10-20
10	Lalisa	Manobal	Breaking Dawn	2023-10-20

3. Show members who had paid overdue more than 2 times. The results must consist of their first name, last name, contact number, number of overdue, and total due amount.

Query:

```
SELECT MEM_FirstName, MEM_LastName, MEM_Contact, COUNT(*) AS NumOfOverdue, SUM(FineAmount) AS TotalFineAmount FROM BORROW_BOOKCOPY
JOIN BORROW ON BORROW.Borrow_ID = BORROW_BOOKCOPY.Borrow_ID
JOIN MEMBER ON MEMBER.Member_ID = BORROW.Member_ID
WHERE OverDue IS NOT NULL AND OverDue > 0 AND PaymentStatus = 'Paid'
GROUP BY MEM_FirstName, MEM_LastName, MEM_Contact
HAVING COUNT(*) > 2
```

Result Table:

Results | Messages

	MEM_FirstName	MEM_LastName	MEM_Contact	NumOfOverdue	TotalFineAmount
1	Andy	Lau	0123456789	3	1.50
2	Simon Dominic	Jung	0189012345	4	3.50

4. Display the name and description of the book borrowed most in each tag colour. Sort the result by tag colour in descending order.

Query:

```
WITH YellowTagBookBorrowCount AS(
SELECT TOP 1 Title, Description, Tag, COUNT(*) AS BorrowCount FROM BORROW_BOOKCOPY
JOIN BOOK_COPY ON BORROW_BOOKCOPY.BookCopy_ID = BOOK_COPY.BookCopy_ID
JOIN BOOK ON BOOK_COPY.ISBN = BOOK.ISBN
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
GROUP BY BOOK.Title, BOOK.Description, Tag
HAVING Tag = 'Yellow'
ORDER BY BorrowCount DESC
),
GreenTagBookBorrowCount AS(
SELECT TOP 1 Title, Description, Tag, COUNT(*) AS BorrowCount FROM BORROW_BOOKCOPY
JOIN BOOK_COPY ON BORROW_BOOKCOPY.BookCopy_ID = BOOK_COPY.BookCopy_ID
JOIN BOOK ON BOOK_COPY.ISBN = BOOK.ISBN
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
GROUP BY BOOK.Title, BOOK.Description, Tag
HAVING Tag = 'Green'
ORDER BY BorrowCount DESC
),
RedTagBookBorrowCount AS(
SELECT TOP 1 Title, Description, Tag, COUNT(*) AS BorrowCount FROM BORROW_BOOKCOPY
JOIN BOOK_COPY ON BORROW_BOOKCOPY.BookCopy_ID = BOOK_COPY.BookCopy_ID
JOIN BOOK ON BOOK_COPY.ISBN = BOOK.ISBN
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
GROUP BY BOOK.Title, BOOK.Description, Tag
HAVING Tag = 'Red'
ORDER BY BorrowCount DESC
)
SELECT * FROM YellowTagBookBorrowCount
UNION
SELECT * FROM GreenTagBookBorrowCount
UNION
SELECT * FROM RedTagBookBorrowCount
ORDER BY Tag DESC
```

Result Table:

Results | Messages

	Title	Description	Tag	BorrowCount
1	A Promised Land	A riveting, deeply personal account of history in the ...	Yellow	3
2	Einstein: His Life and Universe	Einstein was a rebel and nonconformist from boyho...	Red	3
3	Harry Potter and the Deathly Hallows	Harry has been burdened with a dark, dangerous a...	Green	4

5. Show the percentage of total books on loan by month in year 2023.

Query:

```
WITH MonthLoan2023 AS(
SELECT MONTH(BorrowDate)AS Month,
COUNT(*) AS MonthLoan FROM BORROW
JOIN BORROW_BOOKCOPY ON BORROW.Borrow_ID = BORROW_BOOKCOPY.Borrow_ID
WHERE YEAR(BorrowDate) = '2023'
GROUP BY Month(BorrowDate)
)

SELECT Month, MonthLoan, (MonthLoan * 100.0/(SELECT COUNT(*) AS TotalLoan FROM BORROW_BOOKCOPY
JOIN BORROW ON BORROW_BOOKCOPY.Borrow_ID = BORROW.Borrow_ID
WHERE YEAR(BorrowDate) = '2023')) AS PercentageOfTotalLoan2023
FROM MonthLoan2023
```

Result Table:

Results | Messages

	Month	MonthLoan	PercentageOfTotalLoan2023
1	10	21	40.384615384615
2	11	20	38.461538461538
3	12	11	21.153846153846

6. Show the total number of books reserved by month in year 2023.

Query:

```
SELECT YEAR(ReservationDate)AS Year, MONTH(ReservationDate)AS Month, COUNT(*) AS TotalReservationIn2023 FROM RESERVATION
GROUP BY YEAR(ReservationDate), MONTH(ReservationDate)
HAVING YEAR(ReservationDate) = '2023'
ORDER BY YEAR(ReservationDate), MONTH(ReservationDate)
```

Result Table:

Results | Messages

	Year	Month	TotalReservationIn2023
1	2023	11	1
2	2023	12	1

7. Show the publisher whose books are least in quantity for yellow tagged books.

Query:

```
SELECT TOP 1 PUBLISHER.Publisher_ID, PU_Name, PU_Address, COUNT(*) AS NumOfYellowBook FROM BOOK_PUBLISHER
JOIN PUBLISHER ON BOOK_PUBLISHER.Publisher_ID = PUBLISHER.Publisher_ID
JOIN BOOK ON BOOK_PUBLISHER.ISBN = BOOK.ISBN
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
WHERE Tag = 'Yellow'
GROUP BY PUBLISHER.Publisher_ID, PU_Name, PU_Address
ORDER BY NumOfYellowBook ASC
```

Result Table:

Results | Messages

	Publisher_ID	PU_Name	PU_Address	NumOfYellowBook
1	P001	Company Bloomsbury	4, Jalan Pemaju 3, Hicom-glenmarie Industrial Pa...	1

8. Find the number(s) of books written by each author. Show the according to tag colour.

Query:

```
SELECT AU_Name, Tag, COUNT(*) AS TotalBooksWritten FROM BOOK
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
JOIN BOOK_AUTHOR ON BOOK.ISBN = BOOK_AUTHOR.ISBN
JOIN AUTHOR ON BOOK_AUTHOR.Author_ID = AUTHOR.Author_ID
GROUP BY AU_Name, Tag
ORDER BY Tag
```

Result Table:

Results | Messages

	AU_Name	Tag	TotalBooksWritten
1	Dino	Black	1
2	Hoshi	Black	1
3	Jeonghan	Black	1
4	Joshua	Black	1
5	S.coups	Black	1
6	Shanazar Valiyev	Black	1
7	Woozi	Black	1
8	J.K. Rowling	Green	8
9	Louise Heal Kawai	Green	1
10	Ransom Riggs	Green	1

9. Display the name and contact number of member(s) who had at least borrowed book under genre fantasy once.

Query:

```
SELECT MEM_FirstName, MEM_LastName, MEM_Contact, Genre, COUNT(*) AS NumOfFantasyBookBorrowed FROM MEMBER
JOIN BORROW ON BORROW.Member_ID = MEMBER.Member_ID
JOIN BORROW_BOOKCOPY ON BORROW.Borrow_ID = BORROW_BOOKCOPY.Borrow_ID
JOIN BOOK_COPY ON BORROW_BOOKCOPY.BookCopy_ID = BOOK_COPY.BookCopy_ID
JOIN BOOK ON BOOK_COPY.ISBN = BOOK.ISBN
JOIN CATEGORY ON BOOK.Category_ID = CATEGORY.Category_ID
GROUP BY MEM_FirstName, MEM_LastName, MEM_Contact, Genre
HAVING Genre = 'Fantasy' AND COUNT(*) >= 1
```

Result Table:

Results | Messages

	MEM_FirstName	MEM_LastName	MEM_Contact	Genre	NumOfFantasyBookBorrowed
1	Andy	Lau	0123456789	Fantasy	3
2	Angelababy	Yang	0167890123	Fantasy	2
3	Angelina	Jolie	0112345678	Fantasy	2
4	Beyonce	Knowles	0198765432	Fantasy	1
5	Chris	Evans	0156789012	Fantasy	2
6	Eddie	Peng	0109876543	Fantasy	1
7	Elsa	Nordstrom	0134567890	Fantasy	1
8	Emma	Watson	0123456789	Fantasy	1
9	Jack	Ma	0156789012	Fantasy	1
10	Jay	Park	0178901234	Fantasy	1

10. Display the name and address of publisher(s) who had published more than 4 books.

Query:

```
SELECT PU_Name, PU_Address, COUNT(*)AS NumberofPublication FROM Publisher
JOIN BOOK_PUBLISHER ON PUBLISHER.Publisher_ID=BOOK_PUBLISHER.Publisher_ID
GROUP BY PU_Name, PU_Address
HAVING COUNT(*) > 4
```

Result Table:

Results | Messages

	PU_Name	PU_Address	NumberofPublication
1	Company Woodsland	12, Jalan DBP, Dolomite Business Park, 68100 Batu...	13
2	Company Bloomsbury	4, Jalan Pemaju 3, Hicom-glenmarie Industrial Park...	14
3	Company Willington	63, Lrg Maarof, Bangsar, 59000 Kuala Lumpur, Wila...	6
4	Asia Pacific University	Jalan Teknologi 5, Taman Teknologi Malaysia, 570...	7
5	Company Osteria	No 7-2, Tingkat, 2, Jln Seri Rejang 1, Taman Sri Ra...	5
6	Company Murcott	No. 6, Petaling Jaya, Malaysia, Jalan 51/217, Seksy...	5

References

Diène, B., Rodrigues, J. J. P. C., Diallo, O., Ndoye, E. H. M., & Korotaev, V. V. (2020). Data management techniques for Internet of Things. *Mechanical Systems and Signal Processing*, *138*, 106564. https://doi.org/10.1016/J.YMSSP.2019.106564

Eyada, M. M., Saber, W., El Genidy, M. M., & Amer, F. (2020). Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments. *IEEE Access*, 8, 110656–110668. https://doi.org/10.1109/ACCESS.2020.3002164

Frozza, A. A., Defreyn, E. D., Dos, R., & Mello, S. (2021). An Approach for Schema Extraction of NoSQL Columnar Databases: the HBase Case Study. *Journal of Information and Data Management*, 12(5). https://doi.org/10.5753/JIDM.2021.1966

Idrissov, A., Škec, S., & Maier, A. M. (2020). Visualising systems: mapping system features and interactive information visualisations in design. In *Proceedings of the Design Society: DESIGN Conference 1*, 2295-2304. Cambridge University Press. https://doi.org/10.1017/dsd.2020.138

Rockoff, L. (2021). *The language of SQL*. Addison-Wesley Professional. https://books.google.com/books?hl=en&lr=&id=0NnPEAAAQBAJ&oi=fnd&pg=PT17&dq=sql+data+definition+language+create+database&ots=gD4O2yP0Nj&sig=tzbbqiws8R_QRqjacdXTm2pATPw

Submission status	Submitted for grading
Grading status	Not graded
Last modified	Sunday, 7 April 2024, 4:51 PM
File submissions	DBM Group Assignment Part 2 - Group 8.docx7 April 2024, 4:51 PM Turnitin ID: 2342056165 6%