

AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH
(AIUB)

FACULTY OF SCIENCE & TECHNOLOGY



Course Title
INTRODUCTION TO DATABASE

Summer 2022-2023

Section:

TITLE

Name of your Project system

Supervised By

Md. Ariful Islam

Submitted By: Group no: 8

Name	ID
MOHAMMAD SAYEEM KHAN	22-49767-3
MD. FARSEEN ISLAM SATTIQUE	22-48691-3
MD. TOWFIQ BIN HASAN	22-48732-3
NOWRIN ISLAM NISHAT	22-49739-3

TABLE OF CONTENTS

TOPICS	Page no.
Title Page	1
Table of Content	2
1. Introduction	3
2. Case Study	3
3. ER Diagram	4
4. Normalization	5-6
5. Finalization	7
6. Table Creation	8-13
7. Data Insertion	14-19
8. Query Test	30-34
9. Conclusion	35

Online Bookstore Management System

1. Introduction

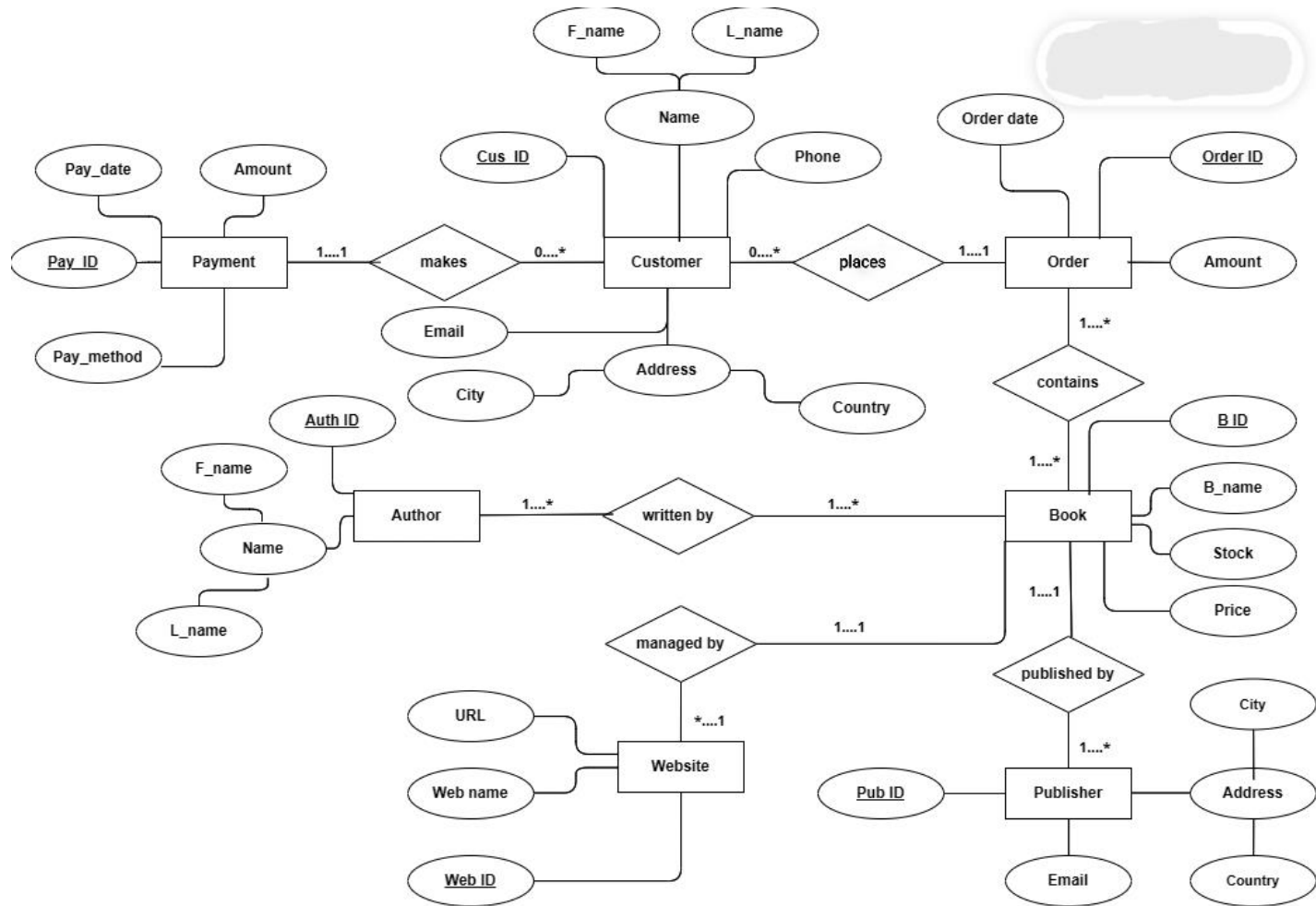
A database system serves as the backbone of modern data management, providing a structured and organized approach to store, manipulate, and access data. Our database system is meticulously designed in Oracle 10g, to meet the specific requirements of the online bookstore. The ER Diagram is drawn through Visual Diagram. The heart of this database system lies in its core tables. Besides, our database system incorporates various querying capabilities through Oracle 10g Database software.

2. Case Study

In an online bookstore management system, multiple interlinked entities and their attributes, along with cardinalities:

Every customer, while registering in the system, gains the capability to place numerous orders. Each order placed by a customer is connected to their unique customer ID, with first name, last name, email, phone, and address, ensuring that all orders are accurately attributed to the respective customers. A customer can make many orders or no orders, while an order can be placed by only one customer, ensuring a one to many relationship. Customers can make multiple payments, and each payment is tied to a particular customer and has its own payment_id, payment_method, payment_date, and amount. A customer can order multiple books. Similarly, a book can be part of various orders placed by different customers, which can be entitled to a many-to-many relationship. Book details, including order_date, order_id, and amount, are managed within each order. Every book has its own unique book_id, price, title, stock, and publish_date. Books often have multiple authors, and an author can be associated with numerous books in a many-to-many way. Books are published by a publisher. The publisher has its own unique publisher_ID, email, publish_date, and address. A publisher can publish many books, while a book is published by one publisher. Author details, such as first names, last names, and author IDs, are maintained in the system as well. A website hosts the books, and its attributes are web_ID, name, and URL. The website manages multiple books, while a book can be managed by one website.

3. ER Diagram



4. Normalization

4.1 Contains Relationship

- ❖ UNF : Order_ID, Order_date, Amount, B_ID, B_name, Stock, Price
- ❖ 1NF : Same as UNF
- ❖ 2NF :

- (i) B_ID(PK), B_name, Stock, Price, Order_ID (FK)
- (ii) Order_ID(PK) , Order_date, Amount
- (iii) B_ID (PK), Order_ID (FK)
- ❖ 3NF : Same as 2NF
- ❖

4.2 Managed by Relationship

- ❖ UNF : B_ID, B_name, Stock, Price, Web_ID, Web_name, URL
- ❖ 1NF : Same as UNF
- ❖ 2NF :
 - (i) B_ID (PK) , B_name, Stock, Price, Web_ID (FK)
 - (ii) Web_ID (PK) , Web_name, URL
- ❖ 3NF : Same as 2NF

4.3 Written by Relationship

- ❖ UNF : Auth_ID, name, F_name, L_name, B_ID, B_name, Stock, Price
- ❖ 1NF : Same as UNF
- ❖ 2NF :
 - (i) Auth_ID (PK), F_name, L_name
 - (ii) B_ID (PK), B_name, Stock, Price
 - (iii) B_ID (PK), Auth_ID (FK)
- ❖ 3NF :
 - (i) Auth_ID (PK), F_name,
 - (ii) B_ID(PK), B_name, Stock, Price
 - (iii) B_ID(PK), Auth_ID (FK)
 - (iv) F_name, L_name

4.4 Payment relationship

- ❖ UNF : Cus_ID, name, F_name, L_name, address, city, country, email, phone, Pay_ID, Pay_method, Pay_Date
- ❖ 1NF : Same as UNF
- ❖ 2NF :
 - (i) Pay_ID (PK), amount, Pay_method, Pay_date
 - (ii) Cus_ID (pk), F_name, L_name, city, country, email, phone, Pay_ID (FK)
- ❖ 3NF :
 - (i) Pay_ID (PK), amount, Pay_method, Pay_date
 - (ii) Cus_ID (pk), F_name, L_name, city, email, phone, Pay_ID (FK)

(iii) City, country

4.5 Published relationship

- ❖ UNF : B_ID, B_name, price, title, stock, Pub_ID, email, address, city, country, Pub_date
- ❖ 1NF : Same as UNF
- ❖ 2NF :
 - (i) B_ID(PK), B_name, price, title, stock
 - (ii) Pub_ID(PK), email, Pub_date, city, country
- ❖ 3NF :
 - (i) Pub_ID(PK), email, Pub_date, city, B_ID(FK)
 - (ii) B_ID(PK), B_name, price, title, stock
 - (iii) City, country

5. Finalization

- ❖ Books:
 - B_ID (PK), B_name, Stock, Price, Order_ID (FK), Web_ID (FK)
- ❖ Orders:

Order_ID (PK), Order_date, Amount

❖ Order_List:

B_ID (PK), Order_ID (FK)

❖ Website:

Web_ID (PK), Web_name, URL

❖ Authors:

Auth_ID (PK), F_name, L_name, B_ID (FK)

❖ Writes:

Auth_ID (PK), B_ID (FK)

❖ Publisher:

Pub_ID (PK), Email, City, Pub_date, B_ID (FK)

❖ Customers:

Cus_ID(PK), F_name, L_name, City, Email, Phone, Pay_ID (FK)

❖ Payments:

Pay_ID (PK), Amount, Pay_method, Pay_date

❖ Address:

City, Country

~~City, Country~~

6. Table creation

6.1 Orders Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Orders (Order_ID number(3) primary key, Order_date Date,  
Amount number(6,2))  
  
describe Orders
```

Object Type **TABLE** Object **ORDERS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
<u>ORDERS</u>	<u>ORDER_ID</u>	Number	-	3	0	1	-	-	-
	<u>ORDER_DATE</u>	Date	7	-	-	-	✓	-	-
	<u>AMOUNT</u>	Number	-	6	2	-	✓	-	-
1 - 3									

Fig 6.1: Orders table creation

6.2 Website Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Website (Web_ID number(6,3) primary key, Web_name  
varchar (30), URL varchar (50))  
  
describe Website
```

Object Type TABLE Object WEBSITE									
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
<u>WEBSITE</u>	<u>WEB_ID</u>	Number	-	6	3	1	-	-	-
	<u>WEB_NAME</u>	Varchar2	30	-	-	-	✓	-	-
	<u>URL</u>	Varchar2	50	-	-	-	✓	-	-
									1 - 3

Fig 6.2: Website table creation

6.3 Books Table

User: BOOKMGT

Home > SQL > **SQL Commands**

☒ Autocommit Display 50 ▼

```
create table Books (B_ID number(3) primary key, B_name varchar(50),
Price number(6,2), Stock number(5), Order_ID number(3),
constraint ofk foreign key(Order_ID) references
Orders (Order_ID),Web_ID number(6,3), constraint wfk foreign key(Web_ID)
references Website(Web_ID))

describe Books
```

Object Type TABLE Object BOOKS									
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
<u>BOOKS</u>	<u>B_ID</u>	Number	-	3	0	1	-	-	-
	<u>B_NAME</u>	Varchar2	50	-	-	-	✓	-	-
	<u>PRICE</u>	Number	-	6	2	-	✓	-	-
	<u>STOCK</u>	Number	-	5	0	-	✓	-	-
	<u>ORDER_ID</u>	Number	-	3	0	-	✓	-	-
	<u>WEB_ID</u>	Number	-	6	3	-	✓	-	-
									1 - 6

Fig 6.3: Books table creation

6.4 Authors Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Authors (Auth_ID number (3), F_name varchar (15), L_name  
varchar (15), B_ID number(3), constraint bfk foreign key (B_ID)  
references Books (B_ID))
```

```
describe Authors
```

Object Type **TABLE** Object **AUTHORS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AUTHORS	AUTH_ID	Number	-	3	0	-	✓	-	-
	F_NAME	Varchar2	15	-	-	-	✓	-	-
	L_NAME	Varchar2	15	-	-	-	✓	-	-
	B_ID	Number	-	3	0	-	✓	-	-
1 - 4									

Fig 6.4: Authors table creation

6.5 Publisher Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Publisher (Pub_ID number(3), Email varchar(30), City varchar(12),  
Pub_date Date, B_ID number(3), constraint bafk foreign key (B_ID)  
references Books(B_ID))
```

```
describe Publisher
```

Object Type **TABLE** Object **PUBLISHER**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PUBLISHER	PUB_ID	Number	-	3	0	-	✓	-	-
	EMAIL	Varchar2	30	-	-	-	✓	-	-
	CITY	Varchar2	12	-	-	-	✓	-	-
	PUB_DATE	Date	7	-	-	-	✓	-	-
	B_ID	Number	-	3	0	-	✓	-	-
1 - 5									

Fig 6.5: Publisher table creation

6.6 Customers Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50

```
create table Customers (Cus_ID number (3) primary key, F_name varchar (15),  
L_name varchar (15), City varchar (12), Email varchar (30),  
Phone varchar (15), Pay_ID number (3),  
constraint pcfk foreign key (Pay_ID) references Payments (Pay_ID))  
  
describe Customers
```

Object Type **TABLE** Object **CUSTOMERS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMERS	CUS_ID	Number	-	3	0	1	-	-	-
	F_NAME	Varchar2	15	-	-	-	✓	-	-
	L_NAME	Varchar2	15	-	-	-	✓	-	-
	CITY	Varchar2	12	-	-	-	✓	-	-
	EMAIL	Varchar2	30	-	-	-	✓	-	-
	PHONE	Varchar2	15	-	-	-	✓	-	-
	PAY_ID	Number	-	3	0	-	✓	-	-

1 - 7

Fig 6.6: Customers table creation

6.7 Payments Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50

```
create table Payments (Pay_ID number (3) primary key, Amount number(8,2),  
Pay_Method varchar (12), Pay_date Date)  
describe Payments
```

Object Type	TABLE	Object	PAYMENTS						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PAYMENTS	PAY_ID	Number	-	3	0	1	-	-	-
	AMOUNT	Number	-	8	2	-	✓	-	-
	PAY_METHOD	Varchar2	12	-	-	-	✓	-	-
	PAY_DATE	Date	7	-	-	-	✓	-	-
									1 - 4

Fig 6.7: Payments table creation

6.8 Address Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Address (City varchar (12) primary key, Country varchar (12))
describe Address
```

Object Type	TABLE	Object	ADDRESS						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADDRESS	CITY	Varchar2	12	-	-	-	✓	-	-
	COUNTRY	Varchar2	12	-	-	-	✓	-	-
									1 - 2

Fig 6.8: Address table creation

6.9 Order_List Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Order_List (B_ID number (3) primary key, Order_ID number (3),
constraint olfk foreign key (Order_ID) references Orders (Order_ID))
describe Order_List
```

Object Type	TABLE Object		ORDER_LIST						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ORDER_LIST	B_ID	Number	-	3	0	1	-	-	-
	ORDER_ID	Number	-	3	0	-	✓	-	-
									1 - 2

Fig 6.9: Order_List table creation

6.10 Writes Table

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 50 ▼

```
create table Writes (Auth_ID number (3) primary key, B_ID number (3),
constraint wafk foreign key (B_ID) references Books(B_ID))
describe Writes
```

Object Type	TABLE Object		WRITES						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
WRITES	AUTH_ID	Number	-	3	0	1	-	-	-
	B_ID	Number	-	3	0	-	✓	-	-
									1 - 2

Fig 6.10: Writes table creation

7. Data Insertion

7.1 Insert Orders table data

User: ONLINE_BOOKSTORE

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
Insert into Orders(Order_ID, Order_Date, Amount) values (101, '12-DEC-23', 530)
Insert into Orders(Order_ID, Order_Date, Amount) values (102, '12-DEC-23', 595)
Insert into Orders(Order_ID, Order_Date, Amount) values (103, '13-DEC-23', 210)
select *
from Orders
```

Results	Explain	Describe	Saved SQL	History
ORDER_ID	ORDER_DATE	AMOUNT		
101	12-DEC-23	530		
102	12-DEC-23	595		
103	13-DEC-23	210		

3 rows returned in 0.00 seconds [CSV Export](#)

Fig 7.1: Orders table data insertion

7.2 Insert Website table data

User: BOOKMGT

Home > SQL > **SQL Commands**

☒ Autocommit Display 100 ▼

```
INSERT INTO Website (Web_ID, Web_name, URL)
VALUES (101, 'Book Paradise', 'https://www.bookparadise.com');
select * from Website
```

WEB_ID	WEB_NAME	URL
101	Book Paradise	https://www.bookparadise.com

Fig 7.2: Website table data insertion

7.3 Insert Books table data

User: BOOKMGT

Home > SQL > **SQL Commands**

☒ Autocommit Display 100 ▼

```
INSERT INTO Books (B_ID, B_name, Price, Stock, Order_ID, Web_ID)
VALUES (1, 'Tale of Glory', 530, 30, 101, 331);

INSERT INTO Books (B_ID, B_name, Price, Stock, Order_ID, Web_ID)
VALUES (2, 'Goals Revealed', 595, 20, 102, 331);

INSERT INTO Books (B_ID, B_name, Price, Stock, Order_ID, Web_ID)
VALUES (3, 'Rise of Rivals', 210, 50, 103, 331);

select * from Website
```

Results Explain Describe Saved SQL History

B_ID	B_NAME	PRICE	STOCK	ORDER_ID	WEB_ID
1	Tale of Glory	530	30	101	331
3	Rise of Rivals	210	50	103	331
2	Goals Revealed	595	20	102	331

3 rows returned in 0.00 seconds

[CSV Export](#)

Fig 7.3: Books table data insertion

7.4 Insert Authors table data

User: ONLINE_BOOKSTORE

Home > SQL > **SQL Commands**

☒ Autocommit Display 10 ▼

```

Insert into Authors(Auth_ID, F_name, L_name, B_ID) values (990, 'Daniel', 'Gibbs', 01)
Insert into Authors(Auth_ID, F_name, L_name, B_ID) values (991, 'Stefany', 'Martin', 02)
Insert into Authors(Auth_ID, F_name, L_name, B_ID) values (992, 'Christopher', 'Sams', 03)

select *
from Authors

```

Results Explain Describe Saved SQL History

AUTH_ID	F_NAME	L_NAME	B_ID
990	Daniel	Gibbs	1
991	Stefany	Martin	2
992	Christopher	Sams	3

3 rows returned in 0.00 seconds [CSV Export](#)

Fig 7.4: Authors table data insertion

7.5 Insert Publisher table data

User: ONLINE_BOOKSTORE

Home > SQL > **SQL Commands**

☒ Autocommit Display 10 ▼

```

Insert into Publisher(Pub_ID, Email, City, Pub_Date, B_ID) values (521, 'abcpublishers@gmail.com', 'Dhaka', '13-MAY-21', 01)
Insert into Publisher(Pub_ID, Email, City, Pub_Date, B_ID) values (524, 'greenpublishers@gmail.com', 'Chittagong', '21-MAR-19', 02)
Insert into Publisher(Pub_ID, Email, City, Pub_Date, B_ID) values (528, 'vfqpublishers@gmail.com', 'Khulna', '10-JAN-20', 03)

select *
from Publisher

```

PUB_ID	EMAIL	CITY	PUB_DATE	B_ID
521	abcpublishers@gmail.com	Dhaka	13-MAY-21	1
524	greenpublishers@gmail.com	Chittagong	21-MAR-19	2
528	vfqpublishers@gmail.com	Khulna	10-JAN-20	3

Fig 7.5: Publisher table data insertion

7.6 Insert Customers table data

User: ONLINE_BOOKSTORE

Home > SQL > **SQL Commands**

☒ Autocommit Display 10 ▼

```

Insert into Customers(Cus_ID, F_name, L_name, City, Email, Phone, Pay_ID) values (11, 'Ujjoyeni', 'Dey', 'Dhaka', 'udey@gmail.com', '01718922131', 221)
Insert into Customers(Cus_ID, F_name, L_name, City, Email, Phone, Pay_ID) values (12, 'Neloy', 'Hasan', 'Chittagong', 'neloy2@gmail.com', '01618290156', 222)
Insert into Customers(Cus_ID, F_name, L_name, City, Email, Phone, Pay_ID) values (13, 'Rizwan', 'Karim', 'Khulna', 'rkarim@gmail.com', '01558920011', 223)

select *
from Customers

```

Results Explain Describe Saved SQL History

CUS_ID	F_NAME	L_NAME	CITY	EMAIL	PHONE	PAY_ID
11	Ujjoyeni	Dey	Dhaka	udey@gmail.com	01718922131	221
12	Neloy	Hasan	Chittagong	neloy2@gmail.com	01618290156	222
13	Rizwan	Karim	Khulna	rkarim@gmail.com	01558920011	223

3 rows returned in 0.00 seconds [CSV Export](#)

Fig 7.6: Customers table data insertion

7.7 Insert Payments table data

User: ONLINE_BOOKSTORE

Home > SQL > **SQL Commands**

☒ Autocommit Display 10 ▼

```

INSERT INTO Payments (Pay_ID, Amount, Pay_Method, Pay_Date) VALUES (221, 530, 'Bkash', '12-DEC-23')
INSERT INTO Payments (Pay_ID, Amount, Pay_Method, Pay_Date) VALUES (222, 595, 'card', '12-DEC-23')
INSERT INTO Payments (Pay_ID, Amount, Pay_Method, Pay_Date) VALUES (223, 210, 'Nagad', '13-DEC-23')

select Pay_ID, Amount, Pay_Method, Pay_Date
from Payments

```

PAY_ID	AMOUNT	PAY_METHOD	PAY_DATE
221	530	Bkash	12-DEC-23
222	595	card	12-DEC-23
223	210	Nagad	13-DEC-23

Fig 7.7: Payments table data insertion

7.8 Insert Address table data

User: ONLINE_BOOKSTORE

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```

INSERT INTO Address(City, Country) values ('Dhaka', 'Bangladesh')
INSERT INTO Address(City, Country) values ('Chittagong', 'Bangladesh')
INSERT INTO Address(City, Country) values ('Khulna', 'Bangladesh')
SELECT City, Country
from Address

```

CITY	COUNTRY
Dhaka	Bangladesh
Chittagong	Bangladesh
Khulna	Bangladesh

Fig 7.8: Address table data insertion

7.9 Insert Order_List table data

User: ONLINE_BOOKSTORE

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```

Insert into Order_List(B_ID, Order_ID) values (01, 101)
Insert into Order_List(B_ID, Order_ID) values (02, 102)
Insert into Order_List(B_ID, Order_ID) values (03, 103)
select *
from Order_List

```

B_ID	ORDER_ID
1	101
2	102
3	103

Fig 7.9: Order_List table data insertion

7.10 Insert Writes table data

User: ONLINE_BOOKSTORE

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
Insert into Writes(Auth_ID, B_ID) values (990, 01)
Insert into Writes(Auth_ID, B_ID) values (991, 02)
Insert into Writes(Auth_ID, B_ID) values (992, 03)
select *
from Writes
```

AUTH_ID	B_ID
990	1
991	2
992	3

Fig 7.10: Writes table data insertion

8. Query Test

8.1 Simple Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT * FROM Orders;
```

ORDER_ID	ORDER_DATE	AMOUNT
1	09-DEC-89	150.5
2	20-FEB-23	200.75
3	10-MAR-23	120.3

Fig 8.1: Simple Query

8.2 Single Row Function Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT B_name, UPPER(B_name) AS Uppercase Title FROM Books;
```

B_NAME	UPPERCASE_TITLE
Tale of Glory	TALE OF GLORY
Goals Revealed	GOALS REVEALED
Rise of Rivals	RISE OF RIVALS

Fig 8.2: Query with a Single Row Function

8.3 Multiple Row Function Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT SUM(Stock) AS Total_Stock FROM Books;
```

TOTAL_STOCK
250

Fig 8.3: Query with Multiple Row Function / Aggregate Function

8.3 Single Row Sub Query and Multiple Row Sub Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT * FROM Books WHERE Price = (SELECT MAX(Price) FROM Books);
```

B_ID	B_NAME	PRICE	STOCK	ORDER_ID	WEB_ID
1	Tale of Glory	29.99	50	201	1

Fig 8.3.1: Single Row Sub Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT Auth_ID FROM Authors WHERE B_ID IN (SELECT B_ID FROM Books WHERE Stock > 100);
```

AUTH_ID
503

Fig 8.3.2: Multiple Row Sub Query

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT B.B_name, A.F_name, A.L_name
FROM Books B
JOIN Authors A ON B.B_ID = A.B_ID;
```

B_NAME	F_NAME	L_NAME
Tale of Glory	Daniel	Gibbs
Goals Revealed	Stefany	Martin
Rise of Rivals	Christopher	Sams

Fig 8.4.1: Query to show Inner Join

User: BOOKMGT

Home > SQL > SQL Commands

☒ Autocommit Display 100 ▼

```
SELECT *
FROM Books, Authors
WHERE Books.B_ID = Authors.B_ID(+);
```

B_ID	B_NAME	PRICE	STOCK	ORDER_ID	WEB_ID	AUTH_ID	F_NAME	L_NAME	B_ID
1	Tale of Glory	29.99	50	201	1	501	Daniel	Gibbs	1
2	Goals Revealed	19.99	80	202	2	502	Stefany	Martin	2
3	Rise of Rivals	14.5	120	203	1	503	Christopher	Sams	3

Fig 8.4.2: Query to show Outer Join

8.5 Simple View and Complex View

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 200 ▼

```
CREATE VIEW BookInfo AS  
SELECT B_ID, B_name, Price, Stock  
FROM Books;
```

```
select * from BookInfo
```

B_ID	B_NAME	PRICE	STOCK
1	Tales of Glory	29.99	50
2	Goals Revealed	19.99	80
3	Rise of Rivals	14.5	120

Fig 8.5.1 Query to show Simple View

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 200 ▼

```
CREATE VIEW Book_Order_Details AS  
SELECT B.B_ID, B.B_name, B.Price, B.Stock, O.Order_ID, O.Order_date, O.Amount, W.Web_name, W.URL  
FROM Books B  
JOIN Orders O ON B.Order_ID = O.Order_ID  
JOIN Website W ON B.Web_ID = W.Web_ID;
```

```
select * from Book_Order_Details
```

B_ID	B_NAME	PRICE	STOCK	ORDER_ID	ORDER_DATE	AMOUNT	WEB_NAME	URL
1	Tales of Glory	29.99	50	101	15-JAN-23	500	Book paradise	www.bookparadise.com
2	Goals Revealed	19.99	80	102	20-FEB-23	800	Readers Heaven	www.readersheaven.com
3	Rise of Rivals	14.5	120	103	10-MAR-23	1200	Book paradise	www.bookparadise.com

Fig 8.5.2 Query to show Complex View

9. Conclusion

In conclusion, the database system developed for the online bookstore management system is a robust and comprehensive solution that streamlines data management, enhances customer experience, and facilitates informed decision-making. As the backbone of the bookstore's operations, the database system lays the foundation for sustainable growth and success in the ever-evolving digital landscape