1. You are given a database schema representing a fictional online bookstore. Your task is to normalize the given schema using the principles of database normalization. For each step of normalization, provide a brief explanation of the normalization rule being applied and the resulting modified schema.

Schema:
The initial schema is as follows:

Table: Books

BookId	Title	Author	Year
1	Book A	Author X	2005
2	Book B	Author Y	2010
3	Book C	Author X	2018

Table: Customers

CustomerID	Name	Email
1	John Doe	john@example.com
2	Jane Smith	jane@example.com

Table: Orders

OrderID	CustomerID	BookID
1	1	1
2	1	2
3	2	1

Normalization Steps:
Perform the following normalization steps on the given schema:

Step 1: First Normal Form (1NF)
Explain the 1NF rule and modify the schema to comply with it.
Ans:
1NF states that “*The intersection of each row and column contains one and only one value (atomic values): No repeating groups*”
The table is already in 1NF because there are no multiple values for the same column in any of the rows. It would not have been in 1NF if the table looked like the following:

BookId	Title	Author	Year
1	Book A	Author X, Author Y	2005
2	Book B	Author Y	2010
3	Book C	Author X	2018

BookID

Step 2: Second Normal Form (2NF)

Explain the 2NF rule and modify the schema to comply with it.

Ans: The 2NF rule states that "*A relation is said to be in 2NF if it is in 1NF and every non-primary-key attribute is fully functionally dependent on the primary key*." It could also be written as : *2NF = 1NF + No partial dependencies*

After converting to 2NF, the schema would look like below:

Table: Book

BookId	Title	AuthorID	Year
1	Book A	1	2005
2	Book B	2	2010
3	Book C	1	2018

Table: Author

AuthorId	AuthorName
1	Author X
2	Author Y

Table: Customer

CustomerID	Name	Email
1	John Doe	john@example.com
2	Jane Smith	jane@example.com

Table: Order

OrderID	CustomerID	BookID
1	1	1
2	1	2
3	2	1

Step 3: Third Normal Form (3NF)

Explain the 3NF rule and modify the schema to comply with it.

Ans: The 3NF rule states that "*A relation is said to be in 3NF if it is in 2NF and in which no non-primary-key attribute is transitively dependent on the primary key*". It could also be written as "2NF + No transitive dependencies"

To remove transitive dependencies, we have to create a new table to store the values of Orders and Customers.

Table: Order

OrderID	CustomerID
1	1
2	1
3	2

Table: OrderLineItems (One order can have many line items)

OrderID	BookID
1	1
2	2
3	1

2. Consider the following table, which represents customer orders:

OrderID | CustomerName | Product | Quantity | OrderDate

Describe the potential problems or anomalies that can occur with this table structure and explain which normal form(s) it violates. Normalize the table and present the resulting normalized table(s) in the appropriate normal form(s) and provide a brief explanation for each normalization step.

Problems:

Redundancy:

The rows of order with the same customer have multiple occurrences of the name of the customer. Similarly, the name of the product is redundant if there are multiple rows with the same product.

Update Anomalies:

If the name of a customer needs to be updated or the name of a product needs to be updated, we need to update such values in multiple rows which may result in inconsistencies.

Insertion Anomalies:

A new customer cannot be added without adding an order.

Deletion Anomalies:

If an order is deleted, the name of the corresponding customer may also be lost from the system.

Non-atomic values:

If the order is accommodated in the same row with multiple names of the products, then the values are non-atomic.

Unnormalized to 1NF

OrderId	CustomerName	Product	Quantity	OrderDate
1	Yogen	Macbook, Iphone	1	2024-05-10
2	Yogen	Earpod	2	2024-05-11
3	Saurab	Samsung Galaxy	1	2024-05-12

Here the product names “Macbook, Iphone” have multiple values for the product. So, this is unnormalized.

1NF

To convert it to 1NF, we have to make sure the rows are atomic

OrderId	CustomerName	Product	Quantity	OrderDate
1	Yogen	Macbook	1	2024-05-10
1	Yogen	Iphone	1	2024-05-10
3	Yogen	Earpod	2	2024-05-11
4	Saurab	Samsung Galaxy	1	2024-05-12

2NF

To convert the table into 2NF, we need to remove partial dependencies. Here,

OrderId	CustomerId	OrderDate

ProductId -> ProductName

ProductId	ProductName

(OrderId, ProductId) -> Quantity

OrderId	ProductId	Quantity

3NF

The table does not have any transitive dependencies, so it is already in 3NF.

3-) Given the following unnormalized table structure for a university course enrollment system:
StudentID | StudentName | Major | Course1 | Course2 | Course3 | ... | CourseN

Identify the normalization problems in this structure and explain which normal form(s) it violates. Normalize the table and present the resulting normalized table(s) in the appropriate normal form(s) and provide a brief explanation for each normalization step.

The given table violates 1NF because the same row of students could have multiple courses expanded horizontally in column instead of vertical scalability.
To achieve 1NF we can design 3 different tables:
StudentId -> StudentName, Major

StudentId	StudentName	Major

CourseId -> (Course)

CourseId	CourseName

(StudentId, CourseId)

StudentId	CourseId

The table is already in 3NF because there are no transitive dependencies.