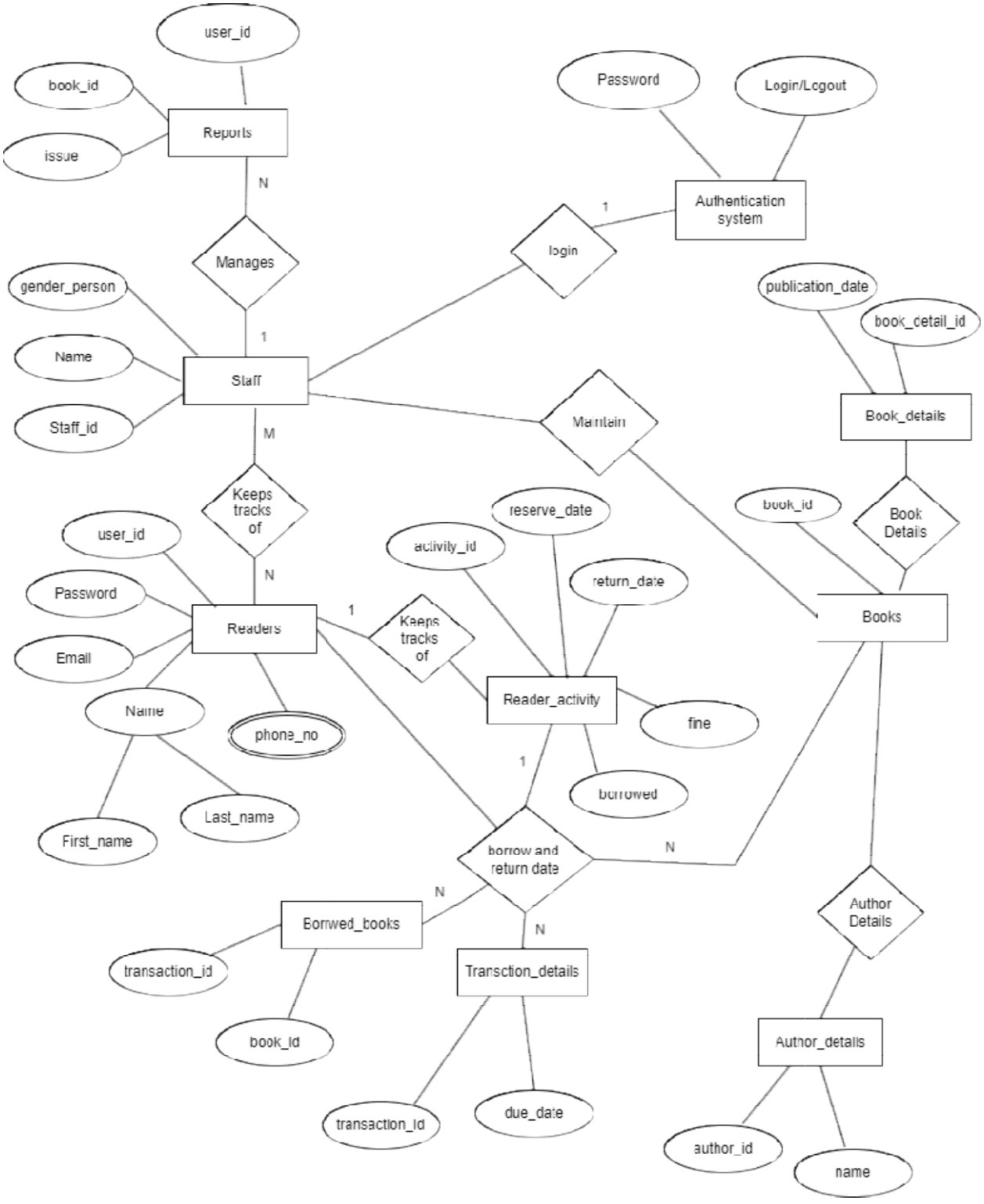
Assignment 1: Analyze a given business scenario and create an ER diagram that includes entities, relationships, attributes, and cardinality. Ensure that the diagram reflects proper normalization up to the third normal form.

ER - diagram for library management system:



Here,

Books,

- book_id (PK)

Authors,

- author_id (PK)
- name

Book_Details,

- book_id (PK)
- publication_date

Transaction,

- transaction_id (PK)
- book_id
- due_date

First Normal Form (1NF):
In the First Normal Form, each table should have a primary key, and there should be no repeating groups or arrays in any column. Our initial tables already satisfy 1NF because they have primary keys, and each cell contains a single value.

Second Normal Form (2NF):
To achieve 2NF, we need to ensure that non-key attributes depend on the entire primary key. In our initial tables, the Books table has partial dependency because Title and Book_details and Author depend only on book_id, not the entire primary key. We'll split the Books table into three tables:

Books (2NF):
- book_id (PK)
- author_id (FK)
- book_details_id (FK)

BookDetails (2NF):

- book_details_id (PK)
- book_title
- publication_date

author (2NF):

- author_id (PK)

- author_name

Now, each table has attributes that depend on the entire primary key.

Third Normal Form (3NF):
In 3NF, we eliminate transitive dependencies. The Transactions table has a transitive dependency on Books through the BookID. To remove this dependency, we'll create a new table for transactions:

Transactions (3NF):
- transaction_id (PK)
- due_date

borrowed_books (3NF):
- id (PK)
- transaction_id (FK)
- book_id (FK)

Now, The BorrowedBooks table manages the relationship between transactions and books.

Our normalized tables in 3NF are as follows:

Books (2NF):
- book_id (PK)
- author_id (FK)
- book_details_id (FK)

BookDetails (2NF):

- book_details_id (PK)
- book_title
- publication_date

author (2NF):

- author_id (PK)
- author_name

Transactions (3NF):
- transaction_id (PK)
- due_date

borrowed_books (3NF):
- id (PK)
- transaction_id (FK)
- book_id (FK)

By following the normalization process to 3NF, we've improved data integrity, reduced redundancy, and eliminated undesirable dependencies, resulting in a well-structured and normalized library system database.

Relationship:

- A reader can reserve N books but one book can be reserved by only one reader. The relationship 1:N.
- A publisher can publish many books but a book is published by only one publisher. The relationship 1:N.
- Staff keeps track of readers. The relationship is M:N.
- Staff maintains multiple reports. The relationship 1:N.
- Staff maintains multiple Books. The relationship 1:N.
- Authentication system provides login to multiple staffs. The relation is 1:N.