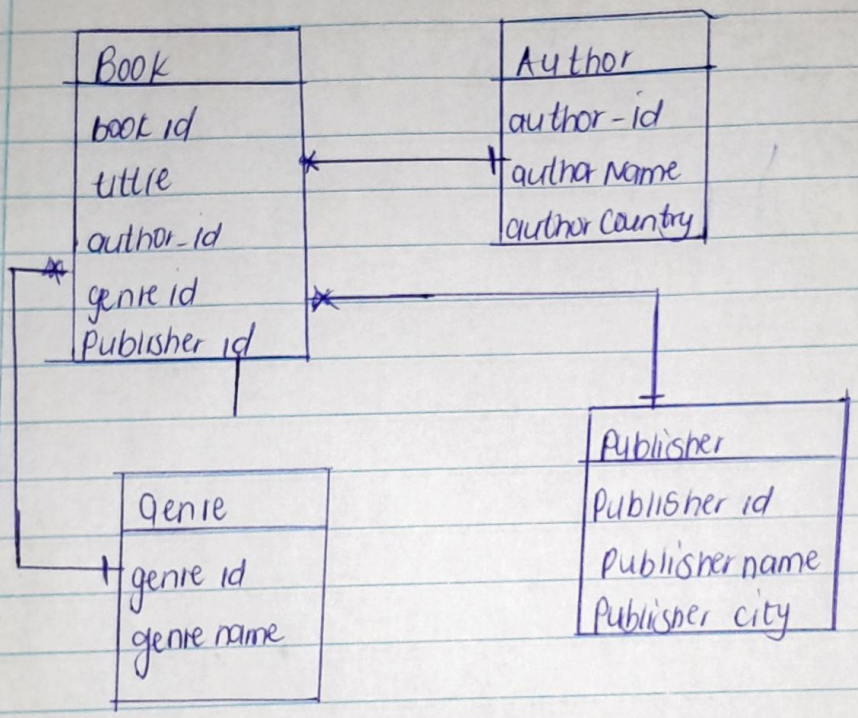
TASK 1 SENARIO 1

1. Author (author-id , authorName , author-Country)
   Publisher (publisher-id , publisherNAME , publisher-city)
   Genre (genre-id, genre-name)
   Book (Book-id, title , author-id, genre id , publisher-id)

2. ER diagram



3. Justification:

1NF → the table Book is initially in 1NF because the columns a having atomic values.

2NF → There are no partial key dependencies so the table is in 2NF.

3NF → There were transitive dependencies, attributes such as Author name ~~were~~ and author country are depending on each other and both are not primary keys. Therefore table author, Genre and Publisher were created.

Scenario 2

1. Customer (customer_id , Customer_name , customer phone , ~~order id~~)
   Product (product_id , product name , product price)
   Order (order_id , order date , customer_id , payment id)
   Payments (payment id , quantity , total_price)

2. customer name ⟶ customer phone
   Product name -> product price
   Order id -> customer name , total price

3. Normalization prevents update unomalies by organising data into multiple related tables to eliminate redundancy and dependency issues. In this case the design ensure that each customer phone is stored once and each product price is stored once.

Senario 3