Task 2b

Write Relational Algebra Expressions For The Following:

1. Retrieving all books of a specific genre with a rating above 4.

Book Table:
Book(bool_id pk, book_title, author, genre, book_condition, price, seller_id Fk → User(user_id), book_content_url, book_description, listing_type, status date)

Review Table:
Review(review_id, book_id FK → Book(book_id), user_id FK → User(user_id), rating, comment, review_date)

To retrieve all books of a specific genre with a rating of 4, we need to join the Book Table with Review Table and then filter out the specific genre with ratings above 4

π(book_id)(σ(genre='Specific_Genre')(Book) ⋈ Review)σ(rating > 4) (Review)

- π (projection): Retrieves the book_id.
- σ (selection): Filters books of a specific genre.
- σ (selection): Filters reviews where rating > 4.
- ⋈ (natural join): Joins Book and Review tables on book_id.

2. Finding all users who have listed more than three books

User Table:
User(user_id pk, name, email Unique, password, phone_number, address, registration_date)

Book Table:
Book(bool_id pk, book_title, author, genre, book_condition, price, seller_id Fk → User(user_id), book_content_url, book_description, listing_type, status date)

To find all users who have listed more than three books, we need to group Books based on user and count books listed per user.

π(user_id, name)(σ(count(book_id) > 3)(γ user_id, count(book_id) → Books))

- γ (grouping): Groups the Book table by user_id and counts the books listed per user.
- σ (selection): Filters users who have listed more than 3 books.
- π (projection): Retrieves user_id and name from the User table.

3. Listing all transactions for books priced above $50

Book Table:
Book(bool_id pk, book_title, author, genre, book_condition, price, seller_id Fk → User(user_id), book_content_url, book_description, listing_type, status date)

Transaction Table:
Transaction(transaction_id PK, book_id FK → Book(book_id), buyer_id FK → User(user_id), seller_id FK → User(user_id), transaction_date, status, payment_method)

To list all transactions for books priced above $50, we will join the Transaction Table and Book table to get the list of books

π(transaction_id, book_id, buyer_id, seller_id, transaction_date, status)
(σ (price > 50)(Book ⋈ Transaction))

- ⋈ (natural join): Joins the Transaction and Book tables on book_id.
- σ (selection): Filters books with price > 50.
- π (projection): Retrieves transaction attributes.