**(40 points)** Write the queries pulling from the database data described in the file librarydb\_queries.txt.

1. Pairs of people (their card numbers, first names and last names) with the same phone number, with no repetition.

Ex1. If Ben Smith with card number 1 has the same phone number as Keith Smith with card number 2, then the following tuple should be in the result of the query:

[1 | Ben | Smith | 2 | Keith | Smith] (or [2 | Keith | Smith | 1 | Ben | Smith])

Tuple [2 | Keith | Smith | 1 | Ben | Smith] should not appear in the result of the query if the above tuple is in the result, as that would denote repetition of information.

Tuple [1 | Keith | Smith | 1 | Keith | Smith] should not appear in the result of the query as it means that member with card number 1 has the same phone number as member with phone number 1, which is obvious.

Ex2. If Ben Smith with card number 1 and Grace Smith with card number 3 have the same phone number as Keith Smith with card number 2, then the following tuples should be in the result of the query:

[1 | Ben | Smith | 2 | Keith | Smith] (or [2 | Keith | Smith | 1 | Ben | Smith])

[1 | Ben | Smith | 3 | Grace | Smith] (or [3 | Grace | Smith | 1 | Ben | Smith])

[3 | Grace | Smith | 2 | Keith | Smith] (or [2 | Keith | Smith | 3 | Grace | Smith])

--------------------------------

πcard\_no,first\_name,last\_name(πm1.phone\_no(σm1.phone\_no = m2.phone\_no ∧ m1.card\_no ≠ m2.card\_no (ρm1 MEMBER ⨯ ρm2 MEMBER))⨝MEMBER)

2. Members (their card numbers, first, middle and last names) who held a book longer than 2 weeks, along with the date they borrowed the book.

--------------------------------

πcard\_no,first\_name,middle\_name,last\_name,date\_borrowed ((πdate\_borrowed,card\_no(σadddate(date\_borrowed, 14) < now() ∧ date\_returned = null BORROW)) ⨝ πcard\_no,first\_name,middle\_name,last\_name MEMBER)

3. Members (their card numbers, first, middle and last names) who didn't ever borrow any book.

--------------------------------

πcard\_no,first\_name,middle\_name,last\_name(πcard\_no(σbarcode = null ∧ date\_borrowed = null ∧ date\_returned = null ∧ renewals\_no = null ∧ paid = null BORROW) ⨝ πcard\_no,first\_name,middle\_name,last\_name MEMBER)

4. ISBNs and titles of books borrowed on 22/9/3.

--------------------------------

πISBN,title((πbarcode(σdate\_borrowed = date('2022-09-03') BORROW) ⨝ πISBN,barcode COPY) ⨝ πtitle,ISBN BOOK)

5. Genres.

--------------------------------

πgenre BOOK

6. ISBNs and titles of books that have more than one author.

--------------------------------

πISBN, title((πba2.ISBN (σba1.ISBN=ba2.ISBN ∧ ba1.author\_id ≠ ba2.author\_id (ρba1 BOOK\_AUTHOR ⨯ ρba2 BOOK\_AUTHOR)))⨝BOOK)

7. ISBN(s) and title(s) of all books written by Thomas Connolly(s), if any.

--------------------------------

πISBN,title((πauthor\_id(σfirst\_name = 'Thomas' ∧ last\_name = 'Connolly'AUTHOR) ⨝ πISBN,author\_id BOOK\_AUTHOR) ⨝ πtitle,ISBN BOOK)

8. ISBN(s) and title(s) of all books borrowed by member 331, if any.

--------------------------------

πISBN,title((πbarcode(σcard\_no = 331 BORROW) ⨝ πISBN,barcode COPY) ⨝ πtitle,ISBN BOOK)

9. Member(s) (their card number) who renewed a book twice and still didn't return it, if any.

--------------------------------

πcard\_no(πcard\_no(σrenewals\_no = 2 ∧ date\_returned = null BORROW) ⨝ πcard\_no MEMBER)

10. Book(s) (their ISBN(s) and title(s)) with the highest edition.

If there is a book with edition 9th, and no other book has a higher edition in the Libary database, then that book should be returned in the result (it's ISBN and title, to be specific).

--------------------------------

πISBN,title(((πb1.edition(σb1.edition ≥ b2.edition (ρb1 BOOK ⨯ ρb2 BOOK)))-(πb1.edition(σb1.edition < b2.edition (ρb1 BOOK ⨯ ρb2 BOOK))))⨝BOOK)

**(4 points)** Suggest 2 improvements/extensions to the Library database and provide justification in each case.

One of the extensions can be adding a relation to find which books are the most popular, which this will help the library to buy or obtain more books in demand.

An improvement or extension would be to add an attribute column that indicates when a book was added to the COPY relation, because this will help the library workers to find which books need an audit. For example, lets say this is the case. There is a book added 50 years ago and a worker needs to find possible bad books to replace. Sure they can find the condition from the database but for books that have not been checked out for a long time, it might not have the condition updated. Now, they can either update the condition or replace it because of mold or other reasons.