## Objective

In this project, we will design and implement a database for a university library.

## Requirements

- Each *book* has a call no., ISBN, a title, an author, an amount (amount of copies available for borrowing), a location (where the book is placed).
- For every *student*, we capture her/his student no., name, gender, and major.
- When a book is successfully borrowed by a student, a *borrow record* should be added to the database. Each *borrow record* contains at least the following information: a borrower, a book, a borrow date and a due date (the due date must be behind the borrow date. The borrow period is four weeks by default). Once the book is returned by the student, the corresponding *borrow record* will be removed.
- For each book borrowed, it may be renewed at most one time. When a renewal is happened, the due date of its corresponding borrow record will be postponed for two weeks.
- Each student may *reserve* a book if the book is not available for the moment. A student can reserve at most one book at the same time.

## Required Functionalities

- **Book search:** Given a search with ISBN as input, you should tell the availability of the book. If the book is available, display the title, author, amount, and location.
- **Book borrow/return:** Given student A and book B, **s**tudent A can borrow book B if the following conditions are satisfied:
  - The available amount of B is larger than 0.
  - The amount of books student A haven't returned yet is smaller than 5.
  - None of the books student A borrowed is overdue.
  - Book B is not reserved by any students, or Student A is the first one who reserved B (in this case the corresponding reservation request should be removed after the book is borrowed).

  The corresponding book and borrow records should be updated upon a successful book borrow/return.
- **Book renew**: Given student A and book B, student A can renew book B if the following conditions are satisfied:
  - None of the books student A borrowed is overdue.
  - Student A hasn't renewed book B after he borrowed it.
  - This renewal is allowed only during the 2nd half of B's borrow period.
  - Book B is not reserved by any students.

  The corresponding borrow record should be updated upon success.
- **Book reserve:** Given student A and book B, student A can reserve book B if the following conditions are satisfied:
  - The available amount of B is 0.
  - Book B is not borrowed by Student A.
  - Student A doesn't hold any other reservation request.

  The corresponding reserve record should be updated upon success.

## Required Trigger

- A book borrow/return will trigger the automatic update of the amount of copies available for borrowing.

## System Implementation

- Write a Java program (command line interface or GUI) to implement the library database application and its required functions.
- Use JDBC Driver to access the Oracle database from the Java program.
- Use PL/SQL to implement the require trigger.

## Project Schedule

The project should be carried out in groups of 3-4 members, and includes two phases.

- **Phase 1: Group forming**

  Each group should register online (http://www.comp.hkbu.edu.hk/~comp2016/2025/proj/reg.html), by March 12, 2025.

  Successful registration will earn each member 5 marks (out of 100). Those who fail to register their groups by the deadline will be grouped *by random* by the instructor.

  Each member of the same group will receive the same mark, so you need to distribute the work among yourselves evenly. In exceptional cases, if the members of the same group wish to receive different grades, you should attach one separate page in the documentation, justifying the reason and identifying individual contributions (in percentage).

- **Phase 2: Database design & implementation**

  You will design an ER diagram, and convert the diagram to relational tables.

  All groups must implement a database according to the functional requirements. This database should support the required functionalities, including book search, borrow/return, renew and reserve. Also, you need to implement some triggers to maintain the integrity and consistency of the data.

  You are expected to submit a project report and source codes of your developed system, and demonstrate the functionalities of your implementation. The final report and demonstration occupy 95 marks.

## Submission (due by 11:59PM April 23, 2025)

- Each group should submit a compressed file named **groupX.zip**.

The groupX.zip file should include the following items:

1. **groupX_source.zip** (the Java project source files)
2. **groupX_dbinsert.sql** (the SQL command file for creating your tables, triggers, constraints and inserting sample data)
3. **groupX_dbdrop.sql** (the SQL command file for dropping your tables, constraints and triggers)
4. **groupX_report.pdf** (the project report should include: 1. group members, including student IDs and names; 2. ER diagram, table schemes, normalization if any, and the corresponding explanation for helping the instructor understand your design; 3. source codes, including comments, of all your implemented functions/triggers and SQL commands)

**Note**: Replace '**X**' with your **group no.** in the above mentioned files.

- Submit your compressed file **groupX.zip** to HKBUMoodle: https://buelearning.hkbu.edu.hk/course/view.php?id=116864.

- Only ONE submission is required for each group, for multiple submissions, only the latest one will be collected.

## Notes

- **Plagiarism (complete/partial copying of other people's work or sharing your own work with other groups) will get a zero mark.**

- **Late submission will not be entertained.**