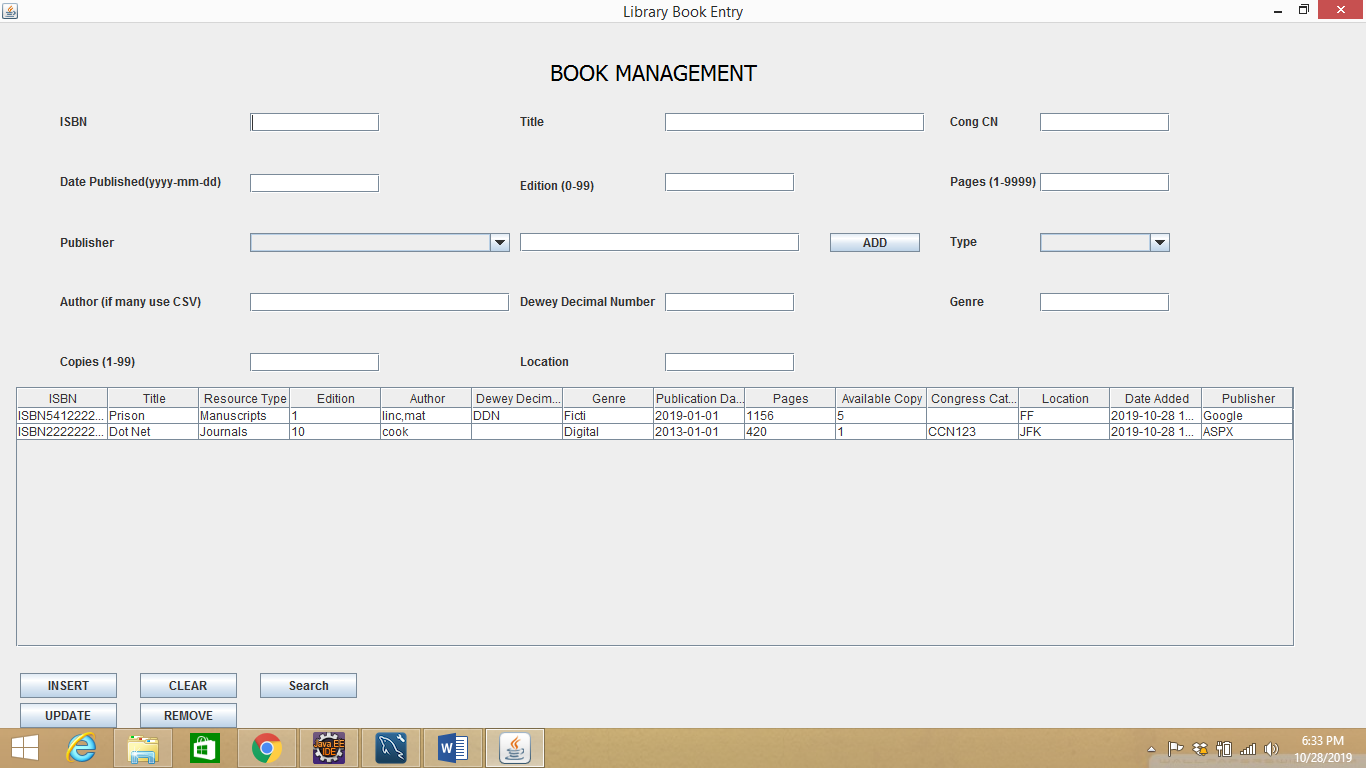
GUI screen looks like



* The attachment has DB design document which is used here
* 5 functionalities are provided – Insert, Clear, Search, Update, Remove
* Clickable table fills in all the above-mentioned fields
* MySQL DB is used with schema name as `library`, `root` as username, `admin` as password. Also, mysql-connector-java-8.0.18.jar should be added in build path.
* Execute the SQL script before starting up the application. It consists of stored procedures that are utilized to fetch results apart from table creation. The application is self sufficient to start without any external guidance.
* All Constraints have been set in JAVA as well as on DB side.
* Certain validations are set on UI side like Edition don’t accept alphabets, author name doesn’t accept numbers. This is implemented via Key press event capture.
* If the publisher is not present it is inserted into the table. Also, the same is applied for author, genre.
* ISBN should be 10 or 13 characters and non-nullable, for title maximum characters is 64(set both on UI and DB side) and non-nullable, Congress Catalogue Number non-nullable, Date Published in specific format and less than present date and non-nullable, edition nullable, pages non-nullable, publisher non-nullable, type non-nullable, authors accepted in CSV format, DDN is null only for fiction, e-Books don’t have location
* Insert button validates the data and insert into data into BookResourceTable, composer(if not present), publisher(if not present), resourcecomposer(which links author to book), Genre(if not present)
* Clear button reset all the fields.
* Search takes any data from the above entered fields to give result and populates into below UI table. All fields are optional. Title and author use partial match. Pages and copies set the lower limit of the book say when page = 150, then books whose page>=150 are displayed. Rest of the fields use exact match.
* Remove button takes input only ISBN to delete a book record. Rest of the fields are ignored.
* Update button validates the data before changing its values in DB.
* Once an ISBN is entered for a book it cannot be changed. The only possible way is to remove and re-insert the ISBN. Several checks have been applied to ensure this.
* All success and error messages are displayed next to buttons.
* SQL Injection is prevented using SP and prepared statements.