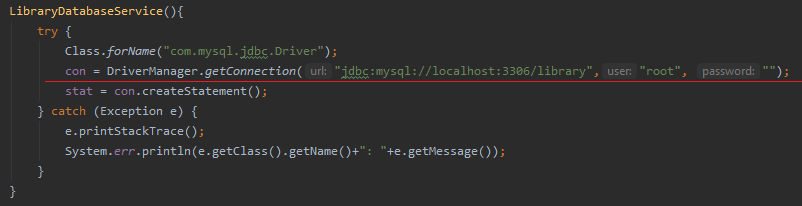
# Sposób uruchomienia

Przed uruchomieniem aplikacji należy wgrać na serwer bazę umieszczoną w repozytorium danego projektu o nazwie „library”. Baza została zrobiona w MySQL.

Następnie należy zmienić dane w konstruktorze klasy LibraryDatabaseService, by połączenie z bazą zaczęło działać (dopasować port, nazwę użytkownika, hasło).

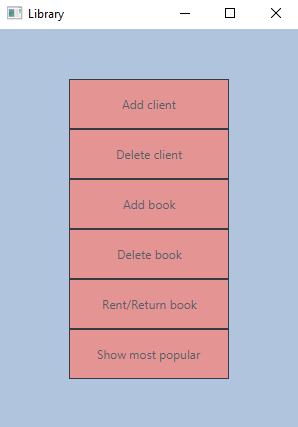
Aplikacja korzysta z JavaFX. By upewnić się że aplikacja zadziała, trzeba również wgrać odpowiednie biblioteki javaFX oraz sprawdzić, czy została zainstalowana wtyczka javaFX runtime. (Dokładny opis znajduje się pod tym linkiem <https://www.jetbrains.com/help/idea/javafx.html#troubleshoot>.)

Następnie należy skompilować i odpalić z IntelliJ IDEA.

# Sposób działania

Aplikacja ma na celu ułatwienie pracy na danych w przykładowej bibliotece. Zapisuje, usuwa rekordy, oraz wyświetla najczęściej pożyczane książki według 2 kryteriów.

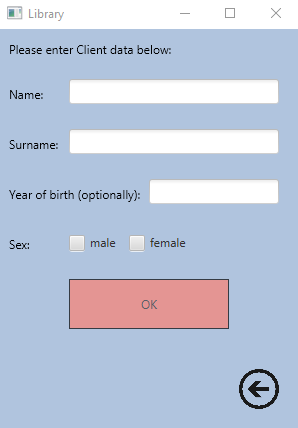
Po odpaleniu aplikacji użytkownik zobaczy następujące okienko:



Wciśniecie danego klawisza pozwala mu na wykonanie konkretnej czynności.

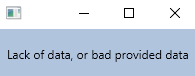
## Dodawanie klienta

Po naciśnięciu pierwszego klawisza „Add client” okno wygląda w dany sposób:



By dodać klienta należy wpisać imię, nazwisko oraz zaznaczyć płeć. Opcjonalnie można podać rok urodzenia klienta biblioteki.

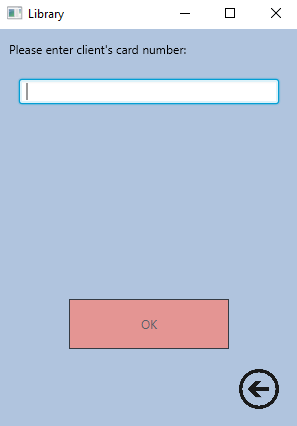
W przypadku nie podania istotnych danych, bądź złego formatu podanych danych wyświetla się okienko z komunikatem. Założono że imię i nazwisko rozpoczyna się z wielkiej litery i zawiera tylko litery, rok zaś to 4 cyfry, w tym pierwsza to 1 lub 2, użyte w tym celu zostały klasy Pattern i Matcher.



Gdy udaje się wprowadzić prawidłowe dane, zostaje wykonana jedna z metod addClient(…) z klasy LibraryDatabaseService. Różnią się one liczbą argumentów. W przypadku podania roku urodzenia są to 4 , w przypadku pozostawieniu tego pola pustego 3. Obie metody dodają klienta do bazy za pomocą komendy SQL Insert into. Do komendy zostają wpisane odpowiednie argumenty podane przy wywoływaniu funkcji. Następnie, w przypadku poprawnego dodania klienta, w metodzie sprawdzany jest maksymalny indeks (numer karty). Metoda zwraca Stringa w postaci komunikatu który pojawia się w wyskakującym oknie. Jeśli wszystko zostało wykonane poprawnie, treść komunikatu informuje o sukcesie i o unikalnym numerze karty klienta, który został mu automatycznie nadany przy dodawaniu go do bazy.

## Usuwanie klienta

By usunąć klienta należy wcisnąć guzik „Delete client”. Okno wygląda następująco:



Gdy wpiszemy odpowiednie dane wyświetli nam się okienko komunikatu. Informuje nas o tym czy usunięto klienta lub czy w ogóle był on w bazie. Wraz z usuwaniem klienta, usunięta zostaje również historia jego wypożyczeń. Po wciśnięciu „Ok”, sprawdzane zostaje czy podano prawidłowe dane(postać liczbowa). Jeżeli tak zostaje wykonana metoda deleteClient(int cardNumber) z klasy LibraryDatabaseService. W metodzie usuwane zostają wszystkie wypożyczenia klienta o podanej numerze karty(gdybyśmy od tego nie zaczęli, naruszylibyśmy założone więzy). Następnie zostaje usunięty sam klient. Gdy liczba usuniętych rekordów jest równa 0, metoda zwraca Stringa z stosowną informacją.

## Dodawanie książki

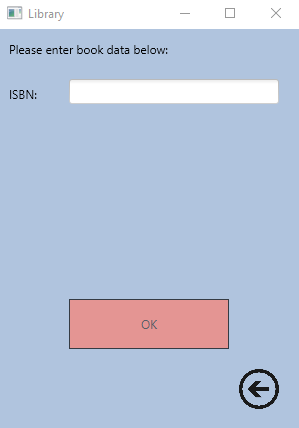
By dodać książkę należy wcisnąć guzik „Add book”. Ukaże się nam taka zawartość okna:



Wszystkie pola są wymagane i nie mogą pozostać puste. W przypadku nie wypełnienia ich, pojawi się stosowny komunikat. Gdy poprawnie wypełnimy pola, zostaje wykonana metoda z klasy LibraryDatabaseService addBook(…). Za pomocą komendy INSERT INTO , którą wypełniamy argumentami podanymi w metodzie, dodajemy książkę do bazy. Metoda zwraca String z komunikatem który pojawia się w wyskakującym okienku. Przy próbie dodania książki o tym samym isbn, metoda zwróci komunikat z treścią błędu.

## Usuwanie książki

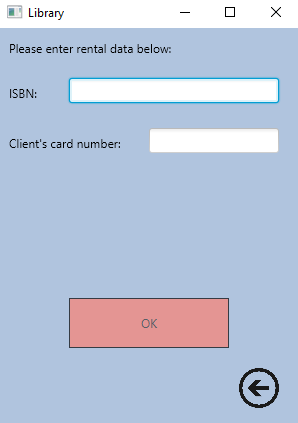
By usunąć książkę z bazy należy wcisnąć guzik „Delete book”. Zobaczymy następujące okno:



Należy poprawnie podać isbn usuwanej książki. Po wciśnięciu klawisza ok, zostaje wykonana metoda deleteBook(isbn), która zwraca nam String z treścią komunikatu, który ukazuje się w wyskakującym oknie. Metoda działa analogicznie, jak metoda usuwania klienta. Wraz z usunięciem książki, usuwane są wypożyczenia związane z nią.

## Wypożyczenie/Oddanie książki

By wypożyczyć książkę klientowi bądź poinformować system o jej zwróceniu należy wcisnąć guzik „Rent/Return book”. Ukaże nam się podana zawartość okna:



Należy wpisać poprawnie isbn oddawanej/wypożyczanej książki oraz numer karty klienta. Po wciśnięciu „Ok” zostaje wywołana metoda z klasy LibraryDatabaseService addUdateRental(isbn, cardNumber). Sprawdza ona najpierw czy podany isbn i numer kart znajdują się w bazie. Jeśli nie zwraca String z odpowiednim komunikatem. Następnie sprawdza zapytaniem select, czy dla danego klienta wypożyczającego tę książkę, istnieje rekord w którym data oddania równa by była NULL. Jeśli tak oznaczałoby, iż klient chce oddać daną książkę. Rekord zostaje wówczas zupdateowany – w miejsce pustego pola dateOfReturn pojawia się obecna data. Jeśli nie istnieje taki rekord, zakładamy że klient chce pożyczyć daną książkę. Metoda sprawdza czy dana książka nie została już komuś pożyczona i czy posiada puste pole dateOfReturn. Oznaczałoby to że w danym momencie, ktoś inny jest w posiadaniu danej książki i nie możemy ją wypożyczyć. Jeśli nie istnieją takie rekordy, wstawiamy nowe wypożyczenie, wpisując podany isbn, numer karty klienta oraz obecną date jako dateOfRent. Metoda zwraca Stringa z komunikatem, który pojawia się na wyskakującym okienku.

## Przegląd najpopularniejszych książek

By przejrzeć najczęściej wypożyczane książki należy wcisnąć guzik „Show most popular”. Pokaże nam się dana zawartość okna:



W zależności jakie wyszukiwanie chcemy zastosować, zaznaczamy interesujące nas pola, wypełniamy obok danymi a następnie klikamy „Show”. W zależności od tego jakie kryteria zostały uwzględnione, zostaje wywołana dana wersja metody getPopularBookList() z klasy LibraryDatabaseService. Zwraca ona ObservableList<Book>, następnie zwracaną listą zostaje wypełniona tabela. W metodzie zostają przekazane konkretne wypełnione przez użytkownika kryteria. Następnie zostają one wprowadzone do komendy SELECT. W przypadku podania wieku czytelnika, zostaje obliczony rok jego urodzenia, a select szuka książek wypożyczone przez czytelników pomiędzy rocznikami 3 lata młodszymi i 3 lata starszymi. Za pomocą COUNT(\*) , liczone zostają ilości wypożyczeń danej książki. Wybrane zostaje 10 rekordów.