

Dokumentacja - Projekt na Bazy Danych 1

Aplikacja biblioteczna w PostgreSQL

Aleksandra Śliwska, 12.01.24r.

1 Projekt koncepcji

1.1 Temat projektu i założenia

Aplikacja została zaprojektowana z myślą o organizacji biblioteki i obsługi klientów, do użytku przez bibliotekarzy. Pozwala ona na przeszukiwanie i edycję zbioru danych o książkach i czytelnikach, oraz zarządzanie wypożyczeniami, zwrotami i wpłatami.

Aplikacja powinna usprawniać obsługę klienta poprzez pozwalanie na szybkie wyszukiwanie informacji o książkach, autorach, wydawnictwach, gatunkach książek; o dostępności egzemplarzy; o kontach czytelników i ich statusie, długach, nieoddanych wypożyczeniach i upodobaniach (w celu usprawnienia polecenia nowych książek przez bibliotekarza).

Dodatkowo aplikacja powinna zawsze odwzorowywać aktualny stan rzeczy w bibliotece i nie pozwalać na wprowadzanie danych niezgodnych z rzeczywistością, ale równocześnie dać bibliotekarzowi jak największą swobodę edycji każdego rekordu.

Według zasad biblioteki czytelnicy mają czas na zwrócenie wypożyczonych egzemplarzy książek do 30 dni po dacie wypożyczenia, za każdy dodatkowy dzień powinien być naliczany dług w wysokości 1 zł.

Taki dodatkowy dzień będzie w tej dokumentacji od teraz nazywany "przedłużeniem").

Użytkownicy, których ilość przedłużeń w ciągu całej ich historii korzystania z biblioteki po zsumowaniu wyniesie więcej niż 365 dni powinni zostać zbanowani z systemu. Zabrania się wypożyczania książek użytkownikom zadłużonym lub zbanowanym (Zakłada się, że te obostrzenia nie istniały przed powstaniem aplikacji w celu ułatwienia wygenerowania przykładowych danych).

1.2 Podstawowe funkcje

1. Przeszukiwanie bazy danych - użytkownik może przeszukać wszystkie rekordy zawarte w bazie danych oraz filtrować wyniki dzięki odpowiednim formularzom.
2. Wyświetlanie profilu każdego rekordu - dodatkowego okna z informacjami o rekordzie.
3. Podsumowywanie informacji - wiele rekordów powiązanych jest z dodatkowymi danymi o nich, których nie da się bezpośrednio wyciągnąć z tabel bazy danych. Aplikacja wyświetla te informacje w odpowiednich miejscach, zarówno dla indywidualnych rekordów (w profilach i tabelach), jak również dla całej bazy danych (w raportach).
4. Edycja i usuwanie rekordów - każdy rekord może zostać edytowany lub usunięty. Dodatkowo zaimplementowane jest proste oddawanie egzemplarza przez użytkownika.
5. Blokowanie wprowadzania niepoprawnych danych - zapisywane informacje muszą być prawidłowo sformułowane oraz nie kolidować logicznie z resztą danych w bazie. Pilnują tego odpowiednie wyzwalacze.

2 Projekt diagramów (konceptualny)

2.1 Przepływ danych

W tym modelu bazy danych definiujemy tylko jeden typ użytkownika - bibliotekarza. Wszystkie dane przechowywane są na serwerze ElephantSQL, zatem przepływ danych wejścia i wyjścia odbywa się tylko pomiędzy serwerem a bibliotekarzami. Jedyne modyfikacje danych, jakie następują podczas przepływu to normalizacja przez wyzwalacze wielkich i małych liter dla nazw gatunków, imion i nazwisk wpisywanych przez użytkownika. Wszystkie inne dane są przekazywane i odbierane w takiej formie, na jaką wskazują zapytania.

2.2 Encje i ich atrybuty

W bazie danych znajdują się odpowiednie encje:

- książka (książka_id, tytuł, opis, autor_id, wydawnictwo_id) - przedstawiająca abstrakcyjny koncept książki, danego "tytułu a nie fizyczny obiekt
- autor (autor_id, imie, nazwisko) - autor książek
- wydawnictwo (wydawnictwo_id, nazwa, data_zalozenia) - wydawnictwo książek
- gatunek (gatunek_id, gatunek_rodzic_id, nazwa) - gdzie *gatunek_rodzic_id* określa związek rekurencyjny, wskazując na ID gatunku, który jest jego rodzicem (ogólniejszy gatunek, do którego należy). Dany gatunek może mieć tylko jednego rodzica.
- egzemplarz (egzemplarz_id, książka_id) - fizyczny obiekt, jeden z reprezentantów danej książki
- czytelnik (czytelnik_id, imie, nazwisko, email, telefon, data_rejestracji) - użytkownik biblioteki
- wypożyczenie (wypożyczenie_id, egzemplarz_id, czytelnik_id, data_wypożyczenia, data_zwrotu) - wypożyczenie danego egzemplarza przez danego czytelnika
- wpłata (wpłata_id, data_wplaty, kwota, czytelnik_id) - wpłata danej nieujemnej kwoty do biblioteki przez danego czytelnika, w celu spłacenia jego długów

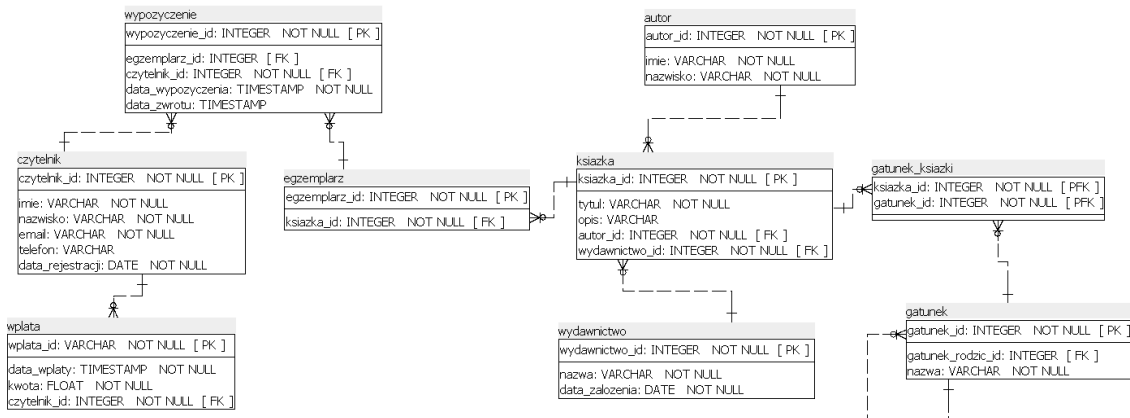
2.3 Relacje pomiędzy encjami

Klucz główny każdej tabeli to ID w postaci liczby całkowitej o nazwie <nazwa tabeli>_id.

Relacja pomiędzy tabelami "książka" i "gatunek" jest typu wiele do wielu, dlatego wstawiono między nie tabelę asocjacyjną "gatunek_książki" o atrybutach (książka_id, gatunek_id), czyli wykorzystującą klucze główne tabel.

Gatunek ma ze sobą relację rekurencyjną, jedno z jego pól wskazuje na ID gatunku, który jest jego rodzicem, czyli szerszą kategorią, do której należy.

Pozostałe relacje są typu 1:N. One, jak również typy atrybutów, klucze główne i obce są pokazane na poniższym diagramie ERD.



Rysunek 1: ERD systemu bibliotecznego użytego w projekcie.

3 Projekt logiczny

3.1 Projektowanie tabel, kluczy, indeksów

Kod w SQL (definicja tabel) zamieszczony [tutaj](#).

Poniżej opisano rozumowanie za nadaniem każdemu atrybutowi odpowiednich własności. Poprawności wpisywanych do systemu danych pilują odpowiednie wyzwalacze.

czytelnik:

- czytelnik_id SERIAL - klucz główny, automatycznie przypisywany
- imie VARCHAR(32) NOT NULL - imię czytelnika, składające się tylko z liter alfabetu, jedynie pierwsza litera jest wielka
- nazwisko VARCHAR(32) NOT NULL - nazwisko czytelnika, składające się tylko z liter alfabetu, jedynie pierwsza litera jest wielka
- email VARCHAR(32) NOT NULL - email czytelnika - składa się on ze znaków alfanumerycznych i kropek przed małąpka, z jednej małąpki, jak również z liter alfabetu i kropek po małąpce
- telefon VARCHAR(32) - numer telefonu czytelnika, opcjonalny - 9-cio znakowe VARCHARy składające się z samych cyfr. Możliwa jest zmiana wyzwalaczy w razie chęci dodania, na przykład, wpisywania kodu kraju.
- data_rejestracji DATE NOT NULL - data rejestracji czytelnika w systemie - dowolna

wplata:

- wplata_id SERIAL - klucz główny, automatycznie przypisywany
- data_wplaty TIMESTAMP NOT NULL - data dokonania wpłaty, dowolna
- kwota NUMERIC(7, 2) NOT NULL CHECK (kwota > 0) - dodatnia wpłacona kwota do biblioteki przez czytelnika
- czytelnik_id INTEGER NOT NULL - czytelnik, który dokonał wpłaty - klucz obcy odnoszący się do czytelnik_id z tabeli czytelnik - jeżeli czytelnik zostanie usunięty, to wpłata kaskadowo też

autor:

- autor_id SERIAL - klucz główny, automatycznie przypisywany

- imie VARCHAR(32) NOT NULL - imię autora, składające się tylko z liter alfabetu, jedynie pierwsza litera jest wielka
- nazwisko VARCHAR(32) NOT NULL - nazwisko autora, składające się tylko z liter alfabetu, jedynie pierwsza litera jest wielka

wydawnictwo:

- wydawnictwo_id SERIAL - klucz główny, automatycznie przypisywany
- nazwa VARCHAR(32) NOT NULL - nazwa wydawnictwa, dowolna
- data_zalozenia DATE NOT NULL - data założenia wydawnictwa, dowolna

ksiazka:

- ksiazka_id SERIAL - klucz główny, automatycznie przypisywany
- tytul VARCHAR(128) NOT NULL - tytuł książki, dowolny
- opis VARCHAR - opis książki, opcjonalny, o dowolnej długości
- autor_id INTEGER NOT NULL - ID autora, który napisał książkę - klucz obcy odnoszący się do autor_id w tabeli autor - jeżeli autor zostanie usunięty, to książka kaskadowo też
- wydawnictwo_id INTEGER NOT NULL - ID wydawnictwa, które wydało książkę - klucz obcy odnoszący się do wydawnictwo_id w tabeli wydawnictwo - jeżeli wydawnictwo zostanie usunięte, to książka kaskadowo też

egzemplarz:

- egzemplarz_id SERIAL - klucz główny, automatycznie przypisywany
- ksiazka_id INTEGER NOT NULL - ID książki, której to jest egzemplarz - klucz obcy odnoszący się do ksiazka_id w tabeli ksiazka - jeżeli książka zostanie usunięta, to egzemplarz kaskadowo też

wypozyczenie:

- wypozyczenie_id SERIAL - klucz główny, automatycznie przypisywany
- data_wypozyczenia TIMESTAMP NOT NULL - data wypożyczenia książki, nie może być wcześniejsza niż data rejestracji danego czytelnika w systemie
- data_zwrotu TIMESTAMP - data zwrotu książki, jeżeli jest NULL, to oznacza, że książka nie została zwrócona
- egzemplarz_id INTEGER - ID egzemplarza, który został wypożyczony - klucz obcy odnoszący się do egzemplarz_id z tabeli egzemplarz - nie można wstawić NULL, ale zostanie on ustawiony NULL, jeżeli dany egzemplarz zostanie usunięty z systemu bibliotecznego
- czytelnik_id INTEGER NOT NULL - ID czytelnika, który wypożyczył egzemplarz - klucz obcy odnoszący się do czytelnik_id z tabeli czytelnik - jeżeli czytelnik zostanie usunięty, to wypożyczenie też
- CHECK (data_wypozyczenia < data_zwrotu) - data wypożyczenia musi być wcześniejsza niż data zwrotu. Dodatkowo sprawdzane jest, czy przedział wypożyczenia danego egzemplarza nie pokrywa się z innym przedziałem czasowym wypożyczenia tego samego egzemplarza.

gatunek:

- gatunek_id SERIAL - klucz główny, automatycznie przypisywany
- gatunek_rodzic_id INTEGER - klucz obcy rekurencyjny - odnosi się do gatunek_id z tabeli gatunek i reprezentuje szerszą kategorię (rodzica), do której należy ten rekord. NULL oznacza, że gatunek jest najszerszą kategorią i nie ma rodzica. Gatunek może mieć tylko jednego rodzica. Usunięcie gatunku, na który wskazuje klucz obcy powoduje ustawienie tu wartości NULL

- nazwa VARCHAR(32) NOT NULL - nazwa gatunku, składająca się tylko z liter alfabetu, jedynie pierwsza litera jest wielka

gatunek_książki: (reprezentacja powiązania tabel książka i gatunek)

- książka_id INTEGER NOT NULL - klucz obcy i główny (wspólnie z gatunek_id) odnoszący się do książka_id z tabeli książka. Usunięcie książki, do której odnosi się klucz obcy powoduje kaskadowe usunięcie rekordu gatunek_książki.
- gatunek_id INTEGER NOT NULL - klucz obcy i główny (wspólnie z książka_id) odnoszący się do gatunek_id z tabeli gatunek, reprezentuje gatunek przypisany danej książce. Usunięcie gatunku, do którego odnosi się klucz obcy powoduje kaskadowe usunięcie rekordu gatunek_książki.

3.2 Analiza zależności funkcyjnych

Spełniona 1NF:

- wszystkie tabele w bazie mają niepodzielne atrybuty
- każdy atrybut niekluczowy jest funkcjonalnie zależny od klucza głównego

Spełniona 2NF:

- jest 1NF
- każdy atrybut niewchodzący w skład klucza głównego jest funkcjonalnie zależny od całości klucza głównego

Spełniona 3NF:

- wszystkie atrybuty niekluczowe są wzajemnie niezależne
- każda tabela ma klucz główny

3.3 Zaprojektowanie operacji na danych

Załączniki z kodem w SQL:

1. [Definicja widoków](#)
2. [Definicja funkcji](#)
3. [Definicja wyzwalaczy](#)
4. [Definicja wyrażeń SELECT, INSERT, UPDATE i DELETE](#)

Dla ułatwienia pisania kwerend, a także dlatego, że wiele zapytań potrzebuje podobnych danych, zdefiniowano odpowiednie widoki:

1. najnowsze_wypożyczenie (wypożyczenie_id, egzemplarz_id, czytelnik_id, data_wypożyczenia, data_zwrotu) - zawiera tylko najnowsze wypożyczenie dla każdego egzemplarza (nie zawiera pozycji z egzemplarzami, które nigdy nie zostały wypożyczone)
2. dostępne_egzemplarze (egzemplarz_id, książka_id) - zawiera wszystkie egzemplarze, które są aktualnie możliwe do wypożyczenia
3. książka_statystyki (książka_id, ilosc_egzemplarzy_w_systemie, ilosc_dostępnych_egzemplarzy) - zawiera wszystkie książki zarejestrowane w systemie, łącznie z ich ilością zarejestrowanych egzemplarzy i ilością egzemplarzy aktualnie dostępnych do wypożyczenia
4. przedłużone_wypożyczenia (wypożyczenie_id, czytelnik_id, dlugosc_wypożyczenia) - zawiera wszystkie wypożyczenia, które trwają/trwały więcej niż 30 dni
5. balans_czytelnika (czytelnik_id, balans) - zawiera aktualny obliczony balans na koncie każdego czytelnika, czyli sumę wpłaconych przez niego pieniędzy, od której odjęta jest suma długów za przetrzymywanie książek (1 zł za każdy dzień ponad limit 30-stu dni)

6. czytelnik_statystyki (czytelnik_id, ilosc_aktualnie_wypożyczonych, ilosc_aktualnie_po_terminie) - zawiera dla każdego czytelnika ilość egzemplarzy książek, które ma aktualnie wypożyczone oraz ile z nich nie zostało zwróconych w terminie 30-stu dni
7. zbanowani_czytelnicy (czytelnik_id, suma_przedluzen) - zawiera ID czytelnika i jego ogólną sumę dni niezwracania książek na czas podczas korzystania z biblioteki, ale tylko, jeżeli ta suma przekroczyła 365 dni
8. status_bana (czytelnik_id, czy_jest_zbanowany) - zawiera ID wszystkich czytelników oraz pole z wartością 'tak' lub 'nie' określającą, czy czytelnik jest zbanowany z systemu bibliotecznego
9. status_wypożyczenia (wypożyczenie_id, dni_nad_limitem) - zawiera wszystkie wypożyczenia w systemie oraz ilość dni ponad limit długości wypożyczenia (30 dni). Jeżeli egzemplarz był oddany w terminie, to wyświetlane jest 0.

Podobnie dla ułatwienia pisania kwerend określono odpowiednie funkcje:

1. wszystkie_gatunki_książki(id_książki INTEGER) RETURNS TABLE(gatunek_id INTEGER) - zwraca wszystkie unikatowe ID gatunków danej książki, łącznie ze wszystkimi gatunkami-rodzicami, które znajdują się rekurencyjnie. W efekcie nawet, jeżeli do książki przypisany był jeden gatunek, to zwracane jest całe jego drzewo do korzenia.
2. czestosc_gatunkow_dla_książek(książki_query VARCHAR) RETURNS TABLE(gatunek_id INTEGER, czestosc INTEGER) - zwraca ilość wystąpień każdego gatunku, sumując wystąpienia pozyskane z wszystkie_gatunki_książki(id), gdzie id to ID każdej książki zwracanej przez wykonanie książki_query
3. znajdz_drzewo_gatunku_wstecz (id_gatunku INTEGER) RETURNS VARCHAR - rysuje drzewo rodzicielstwa dla podanego gatunku (w postaci: NazwaGatunkuBazowego -> NazwaGatunku2 -> NazwaGatunku3 -> NazwaGatunkuPodanego)

Poprawności wpisywanych do systemu danych pilnują wyzwalacze:

1. normalizacja_gatunek - zmienia pierwszą literę wprowadzanej nazwy gatunku na wielką, a resztę na małą
2. normalizacja_autor - sprawdza, czy wprowadzane imię i nazwisko autora składają się wyłącznie z liter alfabetu. Jeżeli nie - odrzuca zmianę. Jeżeli tak - zmienia pierwszą literę imienia i nazwiska na wielką, a resztę na małą.
3. normalizacja_czytelnik - sprawdza, czy wprowadzane imię i nazwisko czytelnika składają się wyłącznie z liter alfabetu. Jeżeli nie - odrzuca zmianę. Jeżeli tak - zmienia pierwszą literę imienia i nazwiska na wielką, a resztę na małą.
4. egzemplarz_nowe_wypożyczenie - nie dopuszcza dodania nowego wypożyczenia, jeżeli:
 - nie podano numeru ID egzemplarza
 - czytelnik wypożyczający egzemplarz jest zadłużony (ma ujemny balans) lub zbanowany
 - data wypożyczenia jest starsza niż data rejestracji czytelnika w systemie
 - data wypożyczenia jest z przyszłości (nowsza niż aktualna)
 - ramy czasowe tego wypożyczenia zachacają o ramy czasowe innego wypożyczenia
5. egzemplarz_edytowane_wypożyczenie - nie dopuszcza do edycji wypożyczenia, jeżeli:
 - data wypożyczenia jest starsza niż data rejestracji czytelnika w systemie
 - data wypożyczenia jest z przyszłości (nowsza niż aktualna)
 - ramy czasowe tego wypożyczenia zachacają o ramy czasowe innego wypożyczenia

6. `poprawnosc_emaila` - sprawdza, czy wprowadzany adres email składa się ze znaków alfanumerycznych lub kropki przed małąpłą, z jednej małąpki oraz z liter alfabetu lub kropek po małąpce. Jeżeli nie, odrzuca zmiany.
7. `poprawnosc_telefonu` - sprawdza, czy wprowadzany numer telefonu czytelnika składa się z 9 cyfr. Jeżeli nie, odrzuca zmiany.

Finalnie zdefiniowano kwerendy SELECT, INSERT, UPDATE i DELETE.

Dla form służących do wyszukiwania i filtrowania wyników wyszukiwania zapytania są budowane po stronie aplikacji, w zależności od tego, które pola formy zostały wypełnione. W wielu przypadkach dozwolone jest wyszukiwanie po atrybutach z innych tabel.

Przykładowe wygenerowane zapytanie przez formę szukającą książek, kiedy w pole 'Nazwa wydawnictwa' wprowadzono wartość 'Zielony Atrament', a w pole 'Nazwisko autora' wpisano '%i':

```
SELECT k.ksiazka_id, k.tytul, k.opis, k.wydawnictwo_id, k.autor_id, ks.
    ilosc_egzemplarzy_w_systemie, ks.ilosc_dostepnych_egzemplarzy
FROM ksiazka k JOIN autor a USING (autor_id) JOIN wydawnictwo w USING (wydawnictwo_id)
    LEFT JOIN egzemplarz e USING (ksiazka_id) LEFT JOIN ksiazka_statystyki ks USING (
        ksiazka_id)
WHERE w.nazwa LIKE 'Zielony Atrament' AND a.nazwisko LIKE '%i' GROUP BY k.ksiazka_id, k.
    tytul, k.opis, k.wydawnictwo_id, k.autor_id, ks.ilosc_egzemplarzy_w_systemie, ks.
    ilosc_dostepnych_egzemplarzy
ORDER BY k.ksiazka_id
```

Przykładowe wygenerowane zapytania dla każdej tabeli można znaleźć w wyżej załączonym pliku SQL ze zdefiniowanymi wyrażeniami SELECT.

Z kolei tak wyglądają zapytania typu INSERT do każdej tabeli:

```
-- czytelnik
INSERT INTO czytelnik (imie, nazwisko, email, telefon, data_rejestracji) VALUES (?, ?, ?,
    ?, NOW()::DATE)

-- egzemplarze - zapytanie wywoływane tyle razy ile uzytkownik sprecyzuje w polu "Ilosc
    egzemplarzy"
INSERT INTO egzemplarz (ksiazka_id) VALUES (?);

-- ksiazka
INSERT INTO ksiazka (tytul, opis, wydawnictwo_id, autor_id) VALUES (?, ?, ?, ?);

-- autor
INSERT INTO autor (imie, nazwisko) VALUES (?, ?);

-- wydawnictwo
INSERT INTO wydawnictwo (nazwa, data_zalozenia) VALUES (?, ?);

-- gatunek
INSERT INTO gatunek (gatunek_rodzic_id, nazwa) VALUES (?, ?);

-- gatunek_ksiazki
INSERT INTO gatunek_ksiazki (ksiazka_id, gatunek_id) VALUES (?, ?);

-- wypozyczenie
INSERT INTO wypozyczenie (egzemplarz_id, czytelnik_id, data_wypozyczenia, data_zwrotu)
    VALUES (?, ?, NOW()::TIMESTAMP, NULL);

-- wplata
INSERT INTO wplata (czytelnik_id, kwota, data_wplaty) VALUES (?, ?, NOW()::TIMESTAMP);
```

Tak wygląda zwracanie egzemplarza przez czytelnika:

```
UPDATE wypozyczenie
SET data_zwrotu = NOW()::TIMESTAMP
WHERE czytelnik_id = ? AND egzemplarz_id = ? AND data_zwrotu IS NULL;
```

Aplikacja pozwala także na edycję każdego atrybutu każdego rekordu (za wyjątkiem kluczy głównych). Zapytania UPDATE są generowane przez aplikację na podstawie pól, które są wypełnione przez użytkownika (nie wszystkie muszą być, pozostawione puste nie są zmieniane). Przykładowe wygenerowane zapytanie dla edycji czytelnika o id = 50 i uzupełnionych jedynie polach 'Email' = 'izielinska56@gmail.com' i 'Telefon' = '123456789':

```
UPDATE czytelnik SET email = 'izielinska56@gmail.com', telefon = '123456789' WHERE
czytelnik_id = 50;
```

Usuwanie w rekordów w aplikacji obsługiwane jest przez proste zapytanie DELETE, analogiczne dla każdej tabeli (zawsze po id):

```
DELETE FROM czytelnik WHERE czytelnik_id = ?;
```

Wyrażenia do raportów o stanie całej bazy danych wyglądały następująco:

- najczęściej pojawiający się w systemie gatunek (łącznie z gatunkami-rodzicami), jeżeli podsumowane zostałyby zbiory gatunków dotyczące każdej książki w systemie:

```
SELECT gatunek_id, nazwa, czestosc
FROM czestosc_gatunkow_dla_ksiazek('SELECT ksiazka_id FROM ksiazka') JOIN gatunek
USING (gatunek_id)
ORDER BY czestosc DESC;
```

- wszystkie wypożyczenia, które nadal nie zostały zwrócone i trwają dłużej niż 30 dni oraz ilość dni, którymi wykroczyły ponad ten limit:

```
SELECT wypozyczenie_id, w.czytelnik_id, w.egzemplarz_id, dlugosc_wypozyczenia-30 AS
dni_nad_limitem
FROM przedluzone_wypozyczenia JOIN wypozyczenie w USING (wypozyczenie_id)
WHERE w.data_zwrotu IS NULL;
```

- wszyscy czytelnicy, którzy są dłużnikami biblioteki:

```
SELECT czytelnik_id, balans
FROM balans_czytelnika
WHERE balans < 0
ORDER BY balans;
```

Dodatkowo każdy rekord posiada swój własny profil, czyli podsumowanie informacji o nim. Wysyłana jest jak najmniejsza ilość zapytań do bazy, więc są one skonsolidowane. Pozyskane dane są odpowiednio formatowane.

- Profil czytelnika:

```
-- ogolne dane, balans, czy jest zbanowany, ilosc aktualnie wypożyczonych ksiazek,
ilosc aktualnie przetrzymywanych ksiazek po terminie, ile ksiazek wypożyczyl w
sumie, ile dni spoznil sie z oddawaniem ksiazek w sumie
```

```
WITH zsumowana_ilosc_wypozycczen AS (
SELECT czytelnik_id, COUNT(*) AS ogolna_ilosc_wypozycczen
FROM wypozycczenie
GROUP BY czytelnik_id
), zsumowana_ilosc_dni_nad_termin AS (
SELECT czytelnik_id, SUM(dlugosc_wypozycczenia-30) as suma_przedluzen
FROM przedluzone_wypozycczenia
GROUP BY czytelnik_id
)
```



```

SELECT imie, nazwisko, email, telefon, data_rejestracji, balans, czy_jest_zbanowany,
       ilosc_aktualnie_wypozyczonych, ilosc_aktualnie_po_terminie,
       ogolna_ilosc_wypozyczen, suma_przedluzen
FROM czytelnik LEFT JOIN balans_czytelnika USING (czytelnik_id) LEFT JOIN status_bana
       USING (czytelnik_id) LEFT JOIN czytelnik_statystyki USING (czytelnik_id) LEFT
       JOIN zsumowana_ilosc_wypozyczen USING (czytelnik_id) LEFT JOIN
       zsumowana_ilosc_dni_nad_termin USING (czytelnik_id)
WHERE czytelnik_id = ?;

```

-- jakie gatunki czyta

```

SELECT nazwa, czestosc
FROM czestosc_gatunkow_dla_ksiazek('SELECT DISTINCT ksiazka_id FROM ksiazka JOIN
       egzemplarz USING (ksiazka_id) JOIN wypozyczenie USING (egzemplarz_id) WHERE
       czytelnik_id = ?') JOIN gatunek USING (gatunek_id)
ORDER BY czestosc DESC;

```

- Profil książki:

-- ogolne dane, ilosc wszystkich i dostepnych egzemplarzy

```

SELECT tytul, opis, wydawnictwo_id, autor_id, ilosc_egzemplarzy_w_systemie,
       ilosc_dostepnych_egzemplarzy
FROM ksiazka LEFT JOIN ksiazka_statystyki USING (ksiazka_id)
WHERE ksiazka_id = ?;

```

-- wszystkie drzewa gatunkow ksiazki

```

SELECT znajdz_drzewo_gatunku_wstecz(gatunek_id) AS drzewo
FROM gatunek_ksiazki
WHERE ksiazka_id = ?;

```

-- wszystkie egzemplarze i ich ostatnie wypozyczenia i dostepnosc

```

SELECT egzemplarz_id, wypozyczenie_id, czytelnik_id, data_wypozyczenia, data_zwrotu,
       CASE WHEN egzemplarz_id IN (SELECT egzemplarz_id FROM dostepne_egzemplarze) THEN
       'tak' ELSE 'nie' END AS czy_egzemplarz_dostepny
FROM egzemplarz LEFT JOIN najnowsze_wypozyczenie USING (egzemplarz_id)
WHERE ksiazka_id = ?;

```

- Profil autora:

-- ogolne dane, ile ksiazek autora, ile egzemplarzy

```

WITH statystyki_ksiazkowe AS (
    SELECT autor_id,
           CASE WHEN COUNT(DISTINCT ksiazka_id) IS NULL THEN 0 ELSE COUNT(DISTINCT
           ksiazka_id) END AS ilosc_ksiazek,
           CASE WHEN COUNT(DISTINCT egzemplarz_id) IS NULL THEN 0 ELSE COUNT(DISTINCT
           egzemplarz_id) END AS ilosc_egzemplarzy
    FROM ksiazka LEFT JOIN egzemplarz USING (ksiazka_id)
    GROUP BY autor_id
)
SELECT imie, nazwisko, ilosc_ksiazek, ilosc_egzemplarzy
FROM autor LEFT JOIN statystyki_ksiazkowe USING (autor_id)
WHERE autor_id = ?;

```

-- gatunki, w ktorych pisze

```

SELECT nazwa, czestosc
FROM czestosc_gatunkow_dla_ksiazek('SELECT ksiazka_id FROM ksiazka WHERE autor_id =
       ?') JOIN gatunek USING (gatunek_id)

```

- ```
ORDER BY czestosc DESC;
```
- Profil wydawnictwa:
 

```
-- ogolne dane, ile ksiazek wydawnictwa jest w systemie

WITH statystyki_ksiazkowe AS (
 SELECT wydawnictwo_id, COUNT(DISTINCT ksiazka_id) AS ilosc_ksiazek
 FROM ksiazka
 GROUP BY wydawnictwo_id
)
SELECT nazwa, data_zalozenia, ilosc_ksiazek
FROM wydawnictwo LEFT JOIN statystyki_ksiazkowe USING (wydawnictwo_id)
WHERE wydawnictwo_id = ?;

-- gatunki wydawane przez wydawnictwo

SELECT nazwa, czestosc
FROM czestosc_gatunkow_dla_ksiazek('SELECT ksiazka_id FROM ksiazka WHERE
 wydawnictwo_id = ?') JOIN gatunek USING (gatunek_id)
ORDER BY czestosc DESC;
```
  - Profil gatunku:
 

```
-- ogolne dane, drzewo gatunku wstecz

SELECT nazwa, gatunek_rodzic_id, znajdz_drzewo_gatunku_wstecz(gatunek_id) AS drzewo
FROM gatunek
WHERE gatunek_id = ?;

-- lista bezposrednich dzieci gatunku

SELECT gatunek_id, nazwa
FROM gatunek
WHERE gatunek_rodzic_id = ?;
```
  - Profil wpłaty:
 

```
-- ogolne dane

SELECT data_wplaty, kwota, czytelnik_id, imie, nazwisko
FROM wplata JOIN czytelnik USING (czytelnik_id)
WHERE wplata_id = ?;
```
  - Profil wypożyczenia:
 

```
-- ogolne dane

SELECT egzemplarz_id, tytul, czytelnik_id, imie, nazwisko, data_wypozyczenia,
 data_zwrotu
FROM wypozyczenie JOIN czytelnik USING (czytelnik_id) LEFT JOIN egzemplarz USING (
 egzemplarz_id) LEFT JOIN ksiazka USING (ksiazka_id)
WHERE wypozyczenie_id = ?;
```
  - Profil egzemplarza:
 

```
-- ogolne dane

SELECT ksiazka_id, tytul
FROM egzemplarz JOIN ksiazka USING (ksiazka_id)
WHERE egzemplarz_id = ?;
```
  - Profil gatunek\_ksiazki:

```
-- ogolne dane

SELECT ksiazka_id, tytul, gatunek_id, nazwa, znajdz_drzewo_gatunku_wstecz(gatunek_id)
AS drzewo
FROM gatunek_ksiazki JOIN ksiazka USING (ksiazka_id) JOIN gatunek USING (gatunek_id)
WHERE ksiazka_id = ? AND gatunek_id = ?;
```

## 4 Projekt funkcjonalny

### 4.1 Interfejsy aplikacji

| ID książki | Tytuł                               | Opis                     | ID wydawnictwa | ID autora | Ilość egzemplarzy... | Ilość dostępnych... |
|------------|-------------------------------------|--------------------------|----------------|-----------|----------------------|---------------------|
| 4          | Pojutrze                            | Podczas podróży pr...    | 4              | 10        | 1                    | 1                   |
| 9          | Filipiada                           | Starożytna legenda o...  | 4              | 7         | 2                    | 2                   |
| 12         | W chmurach                          | Losy trzech obcych s...  | 4              | 5         | 1                    | 1                   |
| 24         | Bezimienne Dusze                    | Głosy z przeszłości n... | 4              | 9         | 3                    | 1                   |
| 32         | Koce i rozmowy                      | W mieście, gdzie iluz... | 4              | 5         | 3                    | 1                   |
| 36         | Jak gotować, żeby nie trafić na SOR | W futurystycznym sp...   | 4              | 5         | 3                    | 2                   |
| 41         | Co jest takiego w motylu            | Gdy gwiazdy zaczyn...    | 4              | 4         | 4                    | 2                   |
| 47         | Ja, pletwa                          | W krainie, gdzie kolo... | 4              | 7         | 2                    | 1                   |
| 58         | Mapy Grecji                         |                          | 4              | 1         | 1                    | 1                   |

Rysunek 2: Wygląd aplikacji - okno wyszukiwania.

Główny panel sterowania aplikacji znajduje się po lewej stronie okna. Poniżej opisane są poszczególne ekrany wyświetlane po wciśnięciu każdego z przycisków.

#### 4.1.1 Przycisk 'Szukaj'

Po wciśnięciu przycisku "Szukaj" w głównym oknie aplikacji wyświetla się panel z 9-ciooma zakładkami.

Rysunek 3: Zakładki do nawigacji pomiędzy formami.

Kliknięcie każdej powoduje wyświetlenie formy, do której można wprowadzić informacje, aby filtrować wyniki wyszukiwania.

|                                       |                      |                 |                                  |                                        |                                            |
|---------------------------------------|----------------------|-----------------|----------------------------------|----------------------------------------|--------------------------------------------|
| ID książki:                           | <input type="text"/> | ID egzemplarza: | <input type="text"/>             | Tytuł:                                 | <input type="text"/>                       |
| Opis:                                 | <input type="text"/> | ID wydawnictwa: | <input type="text"/>             | Nazwa wydawnictwa:                     | <input type="text" value="Pióro i miecz"/> |
| ID autora:                            | <input type="text"/> | Imię autora:    | <input type="text" value="%a%"/> | Nazwisko autora:                       | <input type="text"/>                       |
| <input type="button" value="Szukaj"/> |                      |                 |                                  | <input type="button" value="Wyczyść"/> |                                            |

Rysunek 4: Forma do wprowadzania danych do filtrowania wyszukiwania.

W formie można używać znaku %, który oznacza dowolną ilość dowolnych znaków, tak jak w wyrażeniu LIKE w PostgreSQL. Przycisk "Wyczyść" usuwa wszystkie wprowadzone dane z formy. Przycisk "Szukaj" wysyła zapytanie do bazy danych i wyświetla pod sobą albo informację o błędzie, albo tabelę ze zwróconymi wynikami. Aby tabela została odświeżona, wciśnięcie tego przycisku jest konieczne. Pozostawienie wszystkich pól pustych powoduje zwrócenie tabeli ze wszystkimi jej rekordami.

| ID książki | Tytuł                               | Opis                     | ID wydawnictwa | ID autora | Ilość egzemplarzy... | Ilość dostępnych ... |
|------------|-------------------------------------|--------------------------|----------------|-----------|----------------------|----------------------|
| 4          | Pojutrze                            | Podczas podróży pr...    | 4              | 10        | 1                    | 1                    |
| 9          | Filipiada                           | Starożytna legenda o...  | 4              | 7         | 2                    | 2                    |
| 12         | W chmurach                          | Losy trzech obcych s...  | 4              | 5         | 1                    | 1                    |
| 24         | Bezimienne Dusze                    | Głosy z przeszłości n... | 4              | 9         | 3                    | 1                    |
| 32         | Koce i rozmowy                      | W mieście, gdzie iluz... | 4              | 5         | 3                    | 1                    |
| 36         | Jak gotować, żeby nie trafić na SOR | W futurystycznym sp...   | 4              | 5         | 3                    | 2                    |
| 41         | Co jest takiego w motyłu            | Gdy gwiazdy zaczyn...    | 4              | 4         | 4                    | 2                    |
| 47         | Ja, pletwa                          | W krainie, gdzie kolo... | 4              | 7         | 2                    | 1                    |
| 58         | Mapy Grecji                         |                          | 4              | 1         | 1                    | 1                    |

Rysunek 5: Tabela wynikowa wyszukiwania.

Szerokości kolumn można zmieniać poprzez przeciąganie ich granic w nagłówku kursorem myszy. Na każdy rekord w danej tabeli można podwójnie kliknąć, aby wyświetlić profil tego rekordu. Zawiera on podsumowanie informacji o nim w formie tekstowej.

Biblioteka SQL - profil elementu

Książka

Edytuj

Usuń

# Bezimiennie Dusze

ID: 24  
 Opis: Głosy z przeszłości nawiedzają młodego naukowca, prowadząc go do odkrycia zapomnianych tajemnic i zagubionych miłości.  
 ID wydawnictwa: 4  
 ID autora: 9  
 Ilość egzemplarzy w systemie: 3  
 Ilość dostępnych egzemplarzy: 1  
 Gatunki:  
 Fikcja literacka -> Historyczne  
 Fikcja literacka -> Współczesne  
 Fikcja literacka -> Historyczne -> Marynistyczne -> Pirackie  
 Egzemplarze (i ich najnowsze wypożyczenia):

| ID egzemplarza | ID wypożyczenia | ID czytelnika | Najnowsza data wypoży... | Najnowsza data zwrotu | Czy egzemplarz jest dostępny |
|----------------|-----------------|---------------|--------------------------|-----------------------|------------------------------|
| 61             | 89              | 33            | 2023-12-07 07:00:04      | 2023-12-15 11:28:09   | tak                          |
| 62             | 90              | 11            | 2023-07-30 14:00:25      |                       | nie                          |
| 63             | 91              | 34            | 2022-06-24 01:22:10      |                       | nie                          |

Rysunek 6: Profil rekordu.

W prawym górnym rogu znajdują się przyciski "Edytuj" i "Usuń". Kliknięcie pierwszego przenosi użytkownika do ekranu z formą do edycji danego rekordu.

Biblioteka SQL - profil elementu

Tytuł:

Opis:

ID wydawnictwa:

a

ID autora:

Edytuj

W polach liczbowych muszą zawierać się odpowiednio liczby całkowite lub zmiennoprzecinkowe (z kropką).

Rysunek 7: Forma do edycji rekordu.

Użytkownik może wypełnić dowolne pola nowymi wartościami oraz dowolnie pozostawić puste. Po kliknięciu przycisku "Edytuj", wysyłana jest do zmiany jedynie zawartość pól, które nie są puste. Poniżej przycisku wyświetla się komunikat o sukcesie lub niepowodzeniu operacji.

Z kolei po wciśnięciu przycisku "Usuń" w głównym oknie profilu rekord jest usuwany i wyświetla się komunikat o wykonanej operacji.

## Biblioteka SQL - profil elementu

Usunięto rekord z ID = 63

Rysunek 8: Wynik usunięcia rekordu.

### 4.1.2 Przycisk 'Dodaj'

Rysunek 9: Ekran z formami do dodawania rekordów, ukrytymi w odpowiednich zakładkach.

Wciśnięcie przycisku "Dodaj" w głównym panelu sterowania przenosi użytkownika do ekranu z formami do dodawania rekordów. Nawigacja działa na podobnej zasadzie, jak ta opisana w poprzednim podpunkcie. Na górze okna widoczne jest 7 zakładek (dwie są przeniesione do panelu głównego w celu umożliwienia szybszego dostępu do nich). Kliknięcie na każdą z nich przenosi użytkownika do odpowiedniej formy do dodawania rekordu do danej tabeli. Tym razem wszystkie pola muszą zostać wypełnione, aby zapytanie miało szansę się powieść. Po wciśnięciu przycisku "Dodaj" pod formą wysyłane jest zapytanie do bazy danych, a odpowiedź wyświetlana jest pod formą. Przycisk "Wyczyść" powoduje usunięcie wartości z pól formy.

#### 4.1.3 Przycisk 'Wypożycz/zwróć'

Biblioteka SQL - Aleksandra Śliwska

**Wypożyczanie/zwracanie książki**

ID egzemplarza:  ID czytelnika:

Szukaj

Dodaj

Wypożycz/zwróć

Wpłać

Raporty

Rysunek 10: Forma do wypożyczania i zwracania egzemplarzy przez czytelników.

Kliknięcie przycisku "Wypożycz/zwróć" przenosi użytkownika do formy, która pozwala na zwracanie i wypożyczanie egzemplarzy z biblioteki. Wypożyczanie działa identycznie, jak dodawanie rekordu do bazy opisane w poprzednim podpunkcie. Zwracanie też, jedynie wysyła inne zapytanie i zwraca inną odpowiedź użytkownikowi.

#### 4.1.4 Przycisk 'Wpłać'

The screenshot shows a web application window titled "Biblioteka SQL - Aleksandra Śliwska". On the left is a teal sidebar with five buttons: "Szukaj", "Dodaj", "Wypożycz/zwróć", "Wpłać" (highlighted in blue), and "Raporty". The main area has a light gray header with the title "Dokonywanie płatności". Below the header, there are two input fields: "ID czytelnika:" and "Kwota:". Below the "ID czytelnika:" field is a "Dodaj" button. Below the "Kwota:" field is a "Wyczyść" button.

Rysunek 11: Forma do dokonywania wpłat przez czytelników.

Kliknięcie przycisku "Wpłać" przenosi użytkownika do kolejnej formy do dodawania rekordu do bazy. Działa ona identycznie, jest tu przeniesiona tylko w celu szybkiego dostępu.



#### 4.1.5 Przycisk 'Raporty'

| Raporty                                           |          |
|---------------------------------------------------|----------|
| Najczęściej występujące gatunki                   |          |
| Nieoddane na czas egzemplarze (nadal wypożyczone) |          |
| Dłużnicy                                          |          |
| Odśwież                                           |          |
| ID czytelnika                                     | Balans   |
| 16                                                | -1431.87 |
| 21                                                | -1161.04 |
| 33                                                | -821.84  |
| 31                                                | -751.32  |
| 37                                                | -750.33  |
| 49                                                | -697.45  |
| 47                                                | -577.96  |
| 12                                                | -563.00  |
| 42                                                | -541.29  |
| 34                                                | -510.87  |
| 26                                                | -510.20  |
| 20                                                | -504.00  |
| 25                                                | -385.04  |
| 5                                                 | -325.84  |
| 24                                                | -244.00  |
| 30                                                | -231.21  |
| 11                                                | -117.63  |
| 7                                                 | -95.63   |
| 48                                                | -73.00   |
| 41                                                | -54.00   |

Rysunek 12: Ekran z zakładkami, kryjącymi raporty o całej bazie danych.

Kliknięcie przycisku "Raporty" przenosi użytkownika do ekranu z informacjami ogólnymi o całej bibliotece. Trzy dostępne zakładki posiadają w swojej zawartości przycisk "Odśwież", którego kliknięcie powoduje wypisanie odpowiednich informacji na ekran w postaci tabeli.

## 5 Dokumentacja

Cały kod aplikacji (bez haseł do bazy danych) oraz kod SQL znajduje się w [repozytorium GitHub](#).

### 5.1 Wprowadzanie danych

Manipulacja danymi jest w każdym przypadku ręczna, ale niektóre dane (takie jak niektóre daty) są uzupełniane automatycznie - jeżeli nie istnieje pole do wypełnienia w danej formie do dodawania, to ta wartość jest uzupełniana za użytkownika. Nie istnieją sposoby na import danych.

## 5.2 Krótka instrukcja obsługi aplikacji

| ID książki | Tytuł                               | Opis                     | ID wydawnictwa | ID autora | Ilość egzemplarzy... | Ilość dostępnych ... |
|------------|-------------------------------------|--------------------------|----------------|-----------|----------------------|----------------------|
| 4          | Pojutrze                            | Podczas podróży pr...    | 4              | 10        | 1                    | 1                    |
| 9          | Filipiada                           | Starożytna legenda o...  | 4              | 7         | 2                    | 2                    |
| 12         | W chmurach                          | Losy trzech obcych s...  | 4              | 5         | 1                    | 1                    |
| 24         | Bezimienne Dusze                    | Głosy z przeszłości n... | 4              | 9         | 3                    | 1                    |
| 32         | Koce i rozmowy                      | W mieście, gdzie iluz... | 4              | 5         | 3                    | 1                    |
| 36         | Jak gotować, żeby nie trafić na SOR | W futurystycznym sp...   | 4              | 5         | 3                    | 2                    |
| 41         | Co jest takiego w motylu            | Gdy gwiazdy zaczyn...    | 4              | 4         | 4                    | 2                    |
| 47         | Ja, płetwa                          | W krainie, gdzie kolo... | 4              | 7         | 2                    | 1                    |
| 58         | Mapy Grecji                         |                          | 4              | 1         | 1                    | 1                    |

Rysunek 13: Wygląd aplikacji.

Panel aplikacji znajduje się po lewej. Przycisk "Szukaj" prowadzi do ekranu z zakładkami, w których znajdują się odpowiednie formy do wyszukiwania rekordów w tabelach. Pozostawienie formy pustej zwraca wszystkie rekordy. Przycisk "Wyczyść" czyści pola formy.

Na każdy rekord w tabeli wynikowej można podwójnie kliknąć, aby otworzyć profil rekordu.

Biblioteka SQL - profil elementu

Książka

Edytuj

Usuń

Bezimiennie Dusze

ID: 24

Opis: Głosy z przeszłości nawiedzają młodego naukowca, prowadząc go do odkrycia zapomnianych tajemnic i zagubionych miłości.

ID wydawnictwa: 4

ID autora: 9

Ilość egzemplarzy w systemie: 3

Ilość dostępnych egzemplarzy: 1

Gatunki:

Fikcja literacka -> Historyczne

Fikcja literacka -> Współczesne

Fikcja literacka -> Historyczne -> Marynistyczne -> Pirackie

Egzemplarze (i ich najnowsze wypożyczenia):

| ID egzemplarza | ID wypożyczenia | ID czytelnika | Najnowsza data wypoży... | Najnowsza data zwrotu | Czy egzemplarz jest dostępny |
|----------------|-----------------|---------------|--------------------------|-----------------------|------------------------------|
| 61             | 89              | 33            | 2023-12-07 07:00:04      | 2023-12-15 11:28:09   | tak                          |
| 62             | 90              | 11            | 2023-07-30 14:00:25      |                       | nie                          |
| 63             | 91              | 34            | 2022-06-24 01:22:10      |                       | nie                          |

Rysunek 14: Profil rekordu.

Znajdują się w nim dodatkowe informacje zebrane z całej bazy danych oraz przyciski do edycji i usuwania rekordu. Wypełniając formę do edycji można pozostawić puste pola, których nie chce się zmieniać.

Pozostałe przyciski w głównym panelu sterowania, mianowicie "Dodaj", "Wypożycz/zwróć" i "Wpłać" służą do dodawania rekordów do bazy. Szybkie zwracanie książek jest możliwe w formie wyświetlanej po wcisnięciu przycisku "Wypożycz/zwróć".

Przycisk "Raporty" kryje zakładki z podsumowaniami istotnych informacji z całej bazy danych.

### 5.3 Wykaz literatury

1. [Oficjalna dokumentacja PostgreSQL](#)
2. Łączenie się z bazą danych z aplikacji napisanej w Javie: [PostgreSQL JDBC tutorial](#)
3. Tworzenie GUI dla aplikacji napisanej w Javie: [oficjalne tutoriale do biblioteki Swing](#)
4. Prosta biblioteka upiększająca Swing GUI (funkcjonalność "Look and Feel" biblioteki Swing) - [dokumentacja FlatLaf](#)