



**POLITECHNIKA LUBELSKA
WYDZIAŁ ELEKTROTECHNIKI I
INFORMATYKI**

**KIERUNEK STUDIÓW
INFORMATYKA**

Przedmiot: Zarządzanie bazami SQL i NoSQL

Raport z wykonania projektu pt.

System biblioteczny

Autorzy:

*Wiktoria Matacz
Agnieszka Marzeda
Ewelina Musińska*

Lublin, 2025

ROZDZIAŁ 1. OPIS ORAZ WSKAZANIE PROBLEMU, KTÓRY ZOSTANIE ROZWIAZANY PRZY POMOCY SYSTEMU BIBLIOTECZNEGO.

Współczesne biblioteki, zarówno publiczne, jak i akademickie, stoją przed wieloma wyzwaniami związanymi z zarządzaniem zbiorami, obsługą czytelników oraz optymalizacją procesów bibliotecznych. Tradycyjne metody zarządzania, oparte na ręcznej ewidencji i papierowych kartotekach, są czasochłonne, podatne na błędy oraz niewystarczające w dobie cyfryzacji i rosnących oczekiwania użytkowników. W związku z tym istnieje pilna potrzeba wdrożenia nowoczesnego systemu bibliotecznego, który zautomatyzuje procesy, usprawni zarządzanie zbiorami oraz poprawi komfort korzystania z usług bibliotecznych.

W tradycyjnych bibliotekach proces wypożyczania i zwrotu książek jest często prowadzony ręcznie, co prowadzi do wielu problemów. Przede wszystkim, ręczna ewidencja wypożyczeń i zwrotów jest podatna na błędy, takie jak nieprawidłowe odnotowanie daty zwrotu lub zgubienie informacji o wypożyczonych egzemplarzach. Brak automatyzacji powoduje, że pracownicy poświęcają dużo czasu na żmudne zadania administracyjne, zamiast skupiać się na obsłudze czytelników. To nie tylko obniża efektywność pracy, ale również wpływa na jakość obsługi klienta.

Kolejnym problemem są trudności w zarządzaniu zbiorami. Bez systemu informatycznego trudno jest śledzić lokalizację egzemplarzy, ich dostępność oraz stan (np. czy książka jest dostępna, wypożyczona czy zarezerwowana). Brak centralnej bazy danych utrudnia wyszukiwanie pozycji przez czytelników oraz zarządzanie katalogami. Pracownicy muszą ręcznie przeszukiwać karty katalogowe lub listy, co jest czasochłonne i mało efektywne.

Problemy z rezerwacjami również stanowią wyzwanie. W tradycyjnym systemie rezerwacja książek jest często prowadzona telefonicznie lub osobiście, co jest niewygodne dla czytelników i czasochłonne dla pracowników. Brak automatycznego powiadomiania o dostępności zarezerwowanych pozycji prowadzi do frustracji użytkowników, którzy nie są informowani na czas o możliwości odbioru książki.

Nieskuteczne zarządzanie karami to kolejny istotny problem. W przypadku przetrzymania książek kary są naliczane ręcznie, co może prowadzić do błędów i nieporozumień z czytelnikami. Brak systemu automatycznego naliczania kar utrudnia egzekwowanie zasad, a także sprawia, że proces ten jest mało przejrzysty dla użytkowników.

Brak dostępu do informacji online to kolejna bolączka tradycyjnych bibliotek. Czytelnicy nie mają możliwości sprawdzenia dostępności książek, zarezerwowania pozycji lub przeglądania katalogu online. Ogranicza to dostępność usług bibliotecznych, szczególnie dla osób, które nie mogą odwiedzić biblioteki osobiście. W dobie cyfryzacji i rosnących oczekiwania użytkowników, brak takiej funkcjonalności znacząco obniża atrakcyjność biblioteki.

System biblioteczny

W przypadku bibliotek z wieloma filiami trudno jest koordynować pracę pracowników oraz zarządzać zbiorami rozproszonymi w różnych lokalizacjach. Brak systemu informatycznego utrudnia wymianę informacji między filiami, co prowadzi do nieefektywnego zarządzania zasobami. Pracownicy muszą ręcznie aktualizować informacje o dostępności książek w poszczególnych filiach, co jest czasochłonne i podatne na błędy.

Wdrożenie nowoczesnego systemu bibliotecznego przynosi wiele korzyści zarówno dla biblioteki, jak i dla jej użytkowników. Przede wszystkim, system automatyzuje procesy wypożyczeń i zwrotów, co eliminuje błędy związane z ręczną ewidencją. Dzięki temu pracownicy mogą skupić się na obsłudze czytelników, zamiast poświęcać czas na żmudne zadania administracyjne. Czytelnicy z kolei zyskują możliwość samodzielnego wypożyczania i zwracania książek za pomocą samoobsługowych terminali, co znacznie przyspiesza proces i zwiększa wygodę.

Kolejną korzyścią jest efektywne zarządzanie zbiorami. System śledzi lokalizację każdego egzemplarza, jego dostępność oraz stan, co ułatwia pracownikom zarządzanie pozycjami. Czytelnicy mają natomiast dostęp do aktualnego katalogu online, co pozwala im na szybkie wyszukiwanie interesujących ich książek. Dzięki temu proces wyszukiwania i rezerwacji staje się prostszy i bardziej intuicyjny.

Rezerwacje książek również zostają uproszczone. Czytelnicy mogą rezerwować książki online, a system automatycznie powiadamia ich o dostępności zarezerwowanych pozycji. Rezerwacje są przypisywane do konkretnych czytelników, co eliminuje nieporozumienia i zwiększa przejrzystość procesu. Dzięki temu czytelnicy nie muszą już dzwonić lub przychodzić osobiście, aby zarezerwować książkę.

Automatyczne naliczanie kar to kolejna istotna korzyść. System automatycznie nalicza kary za przetrzymanie książek i przypisuje je do konta czytelnika. Czytelnicy są powiadamiani o naliczonych karach, co zwiększa przejrzystość i uczciwość systemu. Dzięki temu biblioteka może skuteczniej egzekwować zasady, a czytelnicy mają jasność co do obowiązujących regulaminów.

Dostęp do informacji online to kolejna znacząca zaleta systemu. Czytelnicy mają dostęp do katalogu bibliotecznego online, mogą sprawdzać dostępność książek, e-booków oraz audiobooków, przeglądać recenzje oraz zarządzać swoimi kontami. System umożliwia również zdalne rezerbowanie i przedłużanie terminów zwrotu, co znacznie zwiększa wygodę korzystania z usług bibliotecznych. Dzięki temu biblioteka staje się bardziej dostępna dla osób, które nie mogą odwiedzić jej osobiście.

Lepsze zarządzanie pracownikami i filiami to kolejna korzyść wynikająca z wdrożenia systemu. System umożliwia koordynację pracy między filiami, śledzenie dostępności zbiorów w różnych lokalizacjach oraz efektywne zarządzanie pracownikami. Pracownicy mają dostęp do centralnej bazy danych, co ułatwia wymianę informacji i zwiększa efektywność pracy. Dzięki temu biblioteka może lepiej zarządzać swoimi zasobami i zapewnić spójną obsługę we wszystkich filiach.

System biblioteczny

Ostatnią, ale nie mniej ważną korzyścią jest poprawa komfortu czytelników. System oferuje intuicyjny interfejs, umożliwiający łatwe wyszukiwanie książek, rezerwowanie pozycji oraz zarządzanie kontem. Czytelnicy otrzymują powiadomienia o zbliżających się terminach zwrotu oraz dostępności zarezerwowanych książek, co zwiększa ich satysfakcję z korzystania z usług bibliotecznych.

Wdrożenie systemu bibliotecznego rozwiązuje kluczowe problemy związane z zarządzaniem biblioteką, takie jak ręczna ewidencja, trudności w zarządzaniu zbiorami, nieskuteczne rezerwacje oraz brak dostępu do informacji online. System ten nie tylko usprawnia pracę biblioteki, ale również znacznie poprawia komfort korzystania z usług bibliotecznych dla czytelników. Dzięki automatyzacji procesów, biblioteka może skupić się na swojej misji – promowaniu czytelnictwa i zapewnianiu dostępu do wiedzy.

ROZDZIAŁ 2. CEL BUDOWY SYSTEMU INFORMATYCZNEGO ORAZ PREZENTACJA WYMAGAŃ FUNKCJONALNYCH.

Cel budowy systemu informatycznego:

Główym celem budowy systemu informatycznego jest stworzenie nowoczesnej, zintegrowanej platformy bibliotecznej, która usprawni zarządzanie zbiorami, procesy wypożyczeń oraz obsługę czytelników w bibliotekach miejskich. System ma umożliwić użytkownikom wygodny dostęp do zasobów bibliotecznych online, a pracownikom i administratorom – efektywne zarządzanie zbiorami i procesami bibliotecznymi.

Wymagania funkcjonalne dla czytelnika

1. Założenie konta w systemie
2. Przeglądanie listy książek, audiobooków oraz e-booków
3. Rezerwacja wybranej pozycji
4. Wypożyczenie pozycji
5. Wystawianie recenzji
6. System powiadomień
7. Zarządzanie kontem
8. Możliwość opłaty kary fizycznie oraz online
9. Statystyki i analizy

Wymagania funkcjonalne dla pracownika filii

1. Wydawanie i przyjmowanie książek
2. Rozliczanie kar
3. Porządkowanie zbiorów.
4. Generowanie raportów.
5. Przyjmowanie zamówień
6. Aktualizowanie stanu książek

Wymagania funkcjonalne dla administratora

1. Zarządzanie książkami
2. Zarządzanie użytkownikami
3. Zarządzanie filiami
4. Składanie zamówień
5. Monitorowanie systemu
6. Zarządzanie kategoriami i autorami

ROZDZIAŁ 3. SZCZEGÓLOWY OPIS PROCESÓW, ZALEŻNOŚCI MIĘDZY NIMI ORAZ OBIEKTÓW.

1. Charakterystyka poszczególnych encji:

Nazwa encji	Opis encji
Książka	Reprezentuje ogólne informacje o książkach, takie jak identyfikator czy cecha fizyczna, np. obecność twardej okładki.
Czytelnik	Encja czytelnik reprezentuje osoby korzystające z usług biblioteki. Zawiera szczegółowe dane kontaktowe i personalne, takie jak imię, nazwisko, numer PESEL, datę urodzenia czy adres zamieszkania. Dzięki kolumnie płeć możliwe jest także przechowywanie informacji o płci czytelnika, co może być przydatne w celach statystycznych.
Pozycja	Przechowuje szczegółowe informacje o publikacjach dostępnych w bibliotece. Zawiera kluczowe dane, takie jak tytuł, opis, rok wydania czy język. Dzięki kolumnom Wydawnictwo_idWydawnictwo i Filia_idFili możliwe jest powiązanie pozycji z konkretnym wydawnictwem oraz filią biblioteki. Tabela ta jest sercem katalogu biblioteki, ponieważ łączy informacje o książkach, ich autorach (poprzez tabelę pozycja_autor) oraz kategoriach tematycznych.
Kara	Służy do przechowywania informacji o nałożonych karach na czytelników za opóźnienia w zwrocie wypożyczonych książek lub inne naruszenia regulaminu. Każda kara ma określoną wysokość (kwotę), status (np. uregulowana, nieuregulowana) oraz datę przyznania.
Kategoria	Pozwala na grupowanie książek według tematyki, gatunku lub przeznaczenia, na przykład fantasy, kryminał, literatura popularnonaukowa. Każda kategoria ma unikalny identyfikator i nazwę.
Egzemplarz	Zawiera informacje o poszczególnych egzemplarzach książek dostępnych w bibliotece. Każdy egzemplarz ma przypisany unikalny numer (nr_egzemplarza) i status, który określa, czy egzemplarz jest dostępny do wypożyczenia, zarezerwowany
Pozycja_autor	Ta tabela stanowi realizację relacji wiele-do-wielu między tabelą pozycja (książki) a tabelą

System biblioteczny

	autor. Każda pozycja może mieć wielu autorów, a każdy autor może być przypisany do wielu pozycji.
Filia	Encja filia przechowuje informacje o lokalizacjach poszczególnych oddziałów biblioteki, takich jak adres (ulica, numer domu). Jest kluczowa w przypadku systemów bibliotecznych obejmujących wiele placówek, ponieważ pozwala na przypisanie książek, egzemplarzy lub wypożyczeń do konkretnych filii.
Urlop	Encja zawiera dokładne informacje o urlopie pracowników filii. Przechowuje datę początkową i końcową oraz typ urlopu.
Zamówienie	Służy do zamawiania nowych książek do biblioteki. Przechowuje informacje o zamawianych towarzach, pracowniku, który je zamówił oraz określa wartość zamówienia i jego datę.
Lokalizacja	Zawiera informacje o dokładnym położeniu każdej książki w wybranej filii. Dzięki temu łatwiej będzie odszukać książce w bibliotece.
Egzemplarz_w_wypożyczeniu	Encja pośrednicząca łączy dwie główne encje (Wypożyczenie i Egzemplarz) w relacji wiele do wielu. Każdy rekord w tej tabeli wskazuje na konkretne wypożyczenie i przypisany do niego egzemplarz.
Rezerwacja	Służy do rezerwacji niedostępnych książek, które zostały wypożyczone. Przechowują informacje o zarezerwowanej książce, użytkowniku który zarezerwował książkę oraz jego numer w kolejce do wypożyczenia książki.
Ebook	Encja zawiera informacje o rozmiarze pliku, formacie (np.mobi, epub) oraz kod pod którym znajduję się dostęp do wypożyczonego ebooka.
Autor	Encja zawiera informacje o autorze lub autorach książki, takie jak imię, nazwisko oraz kraj pochodzenia.
Dane_logowania	Encja pozwala na śledzenie i zarządzanie dostępem do systemu, umożliwiając bezpieczne uwierzytelnianie i ochronę danych osobowych czytelników.
Recenzja	Encja pozwala na gromadzenie informacji zwrotnej od czytelników dotyczącej ocen oraz komentarzy do poszczególnych książek, co może być wykorzystane do analizy popularności, rekomendacji, ulepszania oferty biblioteki lub dostarczania informacji zwrotnej autorom, pomagania innym

System biblioteczny

	czytelnikom w dokonywaniu wyborów oraz zwiększania społeczności czytelniczej biblioteki.
Towary	Encja zawiera takie informacje jak nazwa książki, jej cena, liczba sztuk oraz suma ich wartości, co jest potrzebne do składania zamówień.
Audiobook	Zawiera informacje o książce w formacie audiobook, tj. narrator, rozmiar pliku, czas trwania i kod dostępu, który jest przekazywany czytelnikowi wypożyczającemu audiobook.
Pensja	Przechowuje informacje potrzebne do wypłacenia wynagrodzenia pracownikom filii, tzn. kwota podstawowa, premia i data wypłaty.
Wydawnictwo	Encja przechowuje informacje o wydawnictwie, takie jak nazwa, adres czy numery NIP, REGON i KRS.
Wypożyczenie	Zawiera informacje o danym wypożyczeniu, jak na przykład data wypożyczenia oraz data zwrotu.
Pracownik filii	Encja zawiera najważniejsze informacje o pracownikach filii, takie jak imię czy nazwisko.
Pozycja_kategoria	Encja pośrednicząca łączy dwie encje pozycja oraz kategoria w relacji wiele do wielu. Każdy rekord w tabeli zawiera informacje o danych pozycjach oraz ich kategoriach.

2. Charakterystyka związków między encjami:

1. **Czytelnik i kara** – Relacja jeden do wielu. Czytelnik może mieć wiele kar, jednak nie musi też ich mieć wcale. Każda kara jest przypisana do konkretnego czytelnika.
2. **Czytelnik i rezerwacja** - Relacja jeden do wielu. Czytelnik może zarezerwować wiele pozycji albo nie musi rezerwować ich wcale, jeśli nie chce wypożyczać, jednak każda rezerwacja jest przypisana do konkretnego czytelnika.
3. **Czytelnik i recenzja** - Relacja jeden do wielu. Czytelnik może napisać wiele recenzji jednak każda recenzja jest przypisana do jednego czytelnika.
4. **Czytelnik i dane logowania** - Relacja jeden do jeden. Każdy czytelnik ma dokładnie jeden zestaw danych logowania, a każdy zestaw danych logowania jest przypisany do jednego czytelnika.
5. **Kara i egzemplarz w wypożyczeniu** - Relacja jeden do jeden. Każda kara jest związana z jednym konkretnym wypożyczeniem, a każde wypożyczenie może mieć przypisaną co najwyżej jedną karę.
6. **Filia i pracownik filii** - Relacja jeden do wielu. Filia musi mieć wielu pracowników, ale każdy pracownik musi być przypisany do konkretnej filii. Tabela Pracownik filii przechowuje informacje o tym, w jakiej filii pracuje dany pracownik.
7. **Filia i pozycja** – Relacja jeden do wielu. Jedna filia może przechowywać wiele pozycji, ale każda pozycja jest przypisana do jednej konkretnej filii.
8. **Egzemplarz i egzemplarz w wypożyczeniu** - Relacja jeden do jeden. Jeden egzemplarz może być w danej chwili wypożyczony tylko raz, i każde wypożyczenie dotyczy konkretnego egzemplarza. Egzemplarz jest nośnikiem pozycji, który może być wypożyczany przez czytelników.
9. **Pozycja i egzemplarz** – Relacja jeden do wielu. Pozycja musi mieć przynajmniej jeden egzemplarz.
10. **Egzemplarz i lokalizacja** – Relacja jeden do jeden. Jeden egzemplarz ma przypisaną jedną konkretną lokalizację, a jedna lokalizacja może być przypisana do jednego egzemplarza. Lokalizacja określa, gdzie w bibliotece znajduje się egzemplarz (dział, regał, półka)

11. **Pozycja i pozycja_kategoria** – Relacja jeden do wielu. Pozycja musi należeć do jednej lub więcej kategorii.
12. **Kategoria i pozycja_kategoria** – Relacja jeden do wielu. Do jednej kategorii może należeć wiele pozycji.
13. **Urlap i pracownik filii** – Relacja jeden do wielu. Pracownik może mieć wiele urlopów.
14. **Pozycja i ebook** – Relacja jeden do wielu. Jedna pozycja może mieć wiele ebooków. Każdy ebook może należeć tylko do jednej pozycji.
15. **Autor i pozycja_autor** – Relacja jeden do wielu. Jeden autor może mieć wiele napisanych pozycji (książek).
16. **Pozycja i pozycja_autor** – Relacja jeden do wielu. Jedna pozycja może być napisana przez wielu autorów, ale musi być napisana przez przynajmniej jednego.
17. **Rezerwacja i pozycja** – Relacja jeden do wielu. Osoba dokonująca rezerwacji może zarezerwować kilka różnych pozycji. Pozycja jest przypisana do jednej konkretnej rezerwacji.
18. **Zamówienie i towary** – Relacja jeden do wielu. Jedno zamówienie może zawierać wiele towarów.
19. **Zamówienie i pracownik filii** – Relacja jeden do wielu. Jeden pracownik może złożyć wiele zamówień.
20. **Wypożyczenie i egzemplarz w wypożyczeniu** – Relacja jeden do wielu. Jeden egzemplarz wypożyczonej książki może być częścią wielu różnych wypożyczeń w różnych okresach, ale każde wypożyczenie dotyczy przynajmniej jednego egzemplarza.
21. **Pozycja i audiobook** – Relacja jeden do wielu. Jedna pozycja może mieć wiele wersji audiobooków, ale każdy audiobook dotyczy tylko jednej konkretnej pozycji.
22. **Pensja i pracownik filii** – Relacja jeden do jednego. Jeden pracownik musi mieć tylko jedną pensję.

23. **Wydawnictwo i pozycja** – Relacja jeden do wielu. Jedno wydawnictwo może mieć wiele pozycji, ale każda pozycja może mieć tylko jedno wydawnictwo.
24. **Wypożyczenie i czytelnik** – Relacja jeden do wielu. Jeden czytelnik może mieć wiele wypożyczeń, ale każde wypożyczenie jest przypisane tylko do jednego czytelnika.
25. **Pozycja i recenzja** – Relacja jeden do wielu. Jedna pozycja może mieć wiele recenzji, ale każda recenzja dotyczy tylko jednej pozycji.

3. Charakterystyka poszczególnych atrybutów encji.

Specyfikacja atrybutów encji dane logowania

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idDane_logowania	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację danych logowania
login	VARCHAR(45)	TAK	TAK	Unikalny atrybut służy jako identyfikator w procesie logowania
haslo	CHAR(60)	TAK	NIE	Atrybut przechowuje zaszyfrowane hasło które pomaga w zachowaniu bezpieczeństw a przed nieautoryzowanym dostępem

Specyfikacja atrybutów encji czytelnik

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idCzytelnik	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację czytelników
imie	VARCHAR(40)	TAK	NIE	Atrybut, który służy do wspomagania weryfikacji użytkownika

System biblioteczny

nazwisko	VARCHAR(50)	TAK	NIE	Pomaga weryfikować czytelnika
pesel	CHAR(11)	TAK	TAK	Służy do pełnego potwierdzenia tożsamości
data_ur	DATE	TAK	NIE	Sprawdza czy czytelnik może sam wypożyczać wszystkie książki
plec	ENUM('M', 'K')	TAK	NIE	Służy do identyfikacji płci czytelnika np. w statystykach
miejscowosc	VARCHAR(80)	TAK	NIE	Kontakt z czytelnikiem w przypadku nieopłacenia kary
ulica	VARCHAR(50)	NIE	NIE	Dokładny adres w przypadku wysłania listu
nr_domu	VARCHAR(10)	NIE	NIE	Dokładny adres w przypadku wysłania listu
nr_mieszkania	VARCHAR(10)	NIE	NIE	Dokładny adres w przypadku wysłania listu

Specyfikacja atrybutów encji kara

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idKara	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację kary
kwota	DECIMAL(10,2)	TAK	NIE	Kara jaką musi uiścić czytelnik. Z powodu przetrzymania bądź uszkodzenia książki
status	ENUM ('niezaplacona', 'zaplacona')	TAK	NIE	Atrybut przechowuje dane o tym czy czytelnik zapłacił już nałożoną karę
data_nalozenia	DATETIME	TAK	NIE	Data od której jest naliczana przez system kara

Specyfikacja atrybutów encji egzemplarz

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idEgzempkarz	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację egzemplarza
nr_egzemplarza	VARCHAR(45)	TAK	NIE	Numer egzemplarza który pozwoli na dokładna identyfikację wypożyczonej pozycji w danej filii
status	ENUM ('w wypożyczeniu', 'wolny', 'w rezerwacji')	TAK	NIE	Atrybut przechowuje dane o tym czy czytelnik ma możliwość wypożyczenia danego egzemplarza

Specyfikacja atrybutów encji ksiazka

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idKsiazka	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację książki
Twarda_okladka	BOOLEN	TAK	NIE	Pozwala na wybranie książki zgodnej z preferencjami czytelnika

Specyfikacja atrybutów encji pozycja

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idPozycja	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną wypożyczonej pozycji
tytuł	VARCHAR(45)	TAK	NIE	Pozwala na znalezienie interesującego czytelnika tytułu
opis	LONGTEXT	TAK	NIE	Atrybut przechowujący opis książki/audiobooka bądź e-booka do wypożyczenia
rok_wydania	DATE	TAK	NIE	Data pozwalająca na wyszukanie pozycji z interesującego czytelnika roku
jezyk	VARCHAR(45)	TAK	NIE	Atrybut przechowujący informację o tym w jakim języku jest pozycja do wypożyczenia
typ	ENUM ('ksiazka', 'e-book', 'audiobook')	TAK	NIE	Typ definiuje formę wypożyczenia egzemplarza

Specyfikacja atrybutów encji Filia

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idFilia	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację filii
ulica	VARCHAR(45)	TAK	NIE	Pozwala na wybranie zidentyfikowanie miejsca wypożyczonej książki
nr_domu	VARCHAR(45)	TAK	NIE	Pozwala na wybranie zidentyfikowanie miejsca wypożyczonej książki

Specyfikacja atrybutów encji kategoria

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
idKategoria	INT	TAK	TAK	Unikalny identyfikator umożliwiający jednoznaczną identyfikację kategorii
nazwa	VARCHAR(40)	TAK	NIE	Atrybut pozwalający ustalić z jakiej kategorii pochodzi egzemplarz

Specyfikacja atrybutów encji autorzy

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
Id_autora	INT	Tak	Tak	Unikalny identyfikator autora.
Imię	VARCHAR(40)	Tak	Nie	Imię autora.
Nazwisko	VARCHAR(50)	Tak	Nie	Nazwisko autora.
Kraj	VARCHAR(80)	Tak	Nie	Kraj pochodzenia autora.

Specyfikacja atrybutów encji urlop

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdUrlop	INT	Tak	Tak	Unikalny identyfikator urlopu.
Data_poczatkowa	DATE	Nie	Nie	Data pierwszego dnia urlopu pracownika.
Data_koncowa	DATE	Nie	Nie	Data ostatniego dnia urlopu pracownika.
Typ	ENUM	Nie	Nie	Rodzaj urlopu.
Pracownik_filii_idPracownik_fili	INT	Tak	Nie	Klucz obcy łączący urlop z tabelą pracowników
Pracownik_filii_Filia_idFilia	INT	Tak	Nie	Klucz obcy łączący urlop z tabelą filii.

Specyfikacja atrybutów encji zamówienia

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdZamowienie	INT	TAK	Tak	Unikalny identyfikator zamówienia.
Nazwa	VARCHAR(45)	Tak	Nie	Nazwa zamówienia.
Data_zamówienia	DATE	Tak	Nie	Data złożenia zamówienia.
Dostawcy	VARCHAR(60)	Tak	Nie	Nazwa dostawcy.
Wartość	DECIMAL(10,2)	Tak	Nie	Wartość zamówienia.
Pracownik_filiu_idPracownik_filiu	INT	Tak	Nie	Klucz obcy łączący urlop z tabelą pracowników.
Pracownik_filiu_Filia_idFilia	INT	Tak	Nie	Klucz obcy łączący urlop z tabelą filii.

Specyfikacja atrybutów encji lokalizacja

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdLokalizacja	INT	Tak	Tak	Unikalny identyfikator lokalizacji.
Dział	VARCHAR(40)	Nie	Nie	Nazwa działu.
Regał	INT	Nie	Nie	Numer regału.
Półka	INT	Nie	Nie	Numer półki.
Pozycja	INT	Nie	Nie	Numer pozycji.

Specyfikacja atrybutów encji egzemplarz w wypożyczeniu

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdEgzemplarz_w_Wypożyczeniu	INT	Tak	Tak	Unikalny identyfikator egzemplarza w wypożyczeniu.
Wypożyczenie_idWypożyczenie	INT	Tak	Nie	Klucz obcy łączący egzemplarz w wypożyczeniu z tabelą wypożyczenie.
Egzemplarz_idEgzemplarz	INT	Tak	Nie	Klucz obcy łączący egzemplarz w wypożyczeniu z tabelą egzemplarz.

Specyfikacja atrybutów encji rezerwacja

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdRezerwacja	INT	Tak	Tak	Unikalny identyfikator rezerwacji.
Nr_w_kolejce	INT	Nie	Nie	Numer kolejności rezerwacji.
Status	ENUM	Tak	Nie	Status rezerwacji.
Data_anulowania	DATETIME	Nie	Nie	Data anulowania rezerwacji.
Data_utworzenia	DATETIME	Tak	Nie	Data utworzenia rezerwacji.
Data_realizacji	DATETIME	Nie	Nie	Data realizacji rezerwacji.
Czytelnik_idCzytelnik	INT	Tak	Nie	Klucz obcy łączący rezerwacje z tabelą czytelnicy.
Pozycja_idPozycja	INT	Tak	Nie	Klucz obcy łączący rezerwacje z tabelą pozycja.
Pozycja_Filia_idFilia	INT	Tak	Nie	Klucz obcy łączący rezerwacją z kluczem obcym tabeli pozycja.

Specyfikacja atrybutów encji ebook

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdEbook	INT	Tak	Tak	Unikalny identyfikator ebooka.

System biblioteczny

Rozmiar_pliku	BIGINT(20)	Tak	Nie	Rozmiar pliku zawierający ebook.
Format	VARCHAR(10)	Tak	Nie	Format pliku.
Kod	VARCHAR(45)	Tak	Nie	Kod dostępu do ebooka.
Pozycja_idPozycja	INT	Tak	Nie	Klucz obcy łączący ebook z tabelą pozycja.

Specyfikacja atrybutów encji dane logowania

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdDane_logowania	INT	Tak	Tak	Unikalny identyfikator danych logowania.
Login	VARCHAR(45)	Tak	Tak	Login czytelnika.
Hasło	CHAR(60)	Tak	Nie	Szyfrowane hasło czytelnika.
Czytelnik_idCzytelnik	INT	Tak	Tak	Klucz obcy łączący dane logowania z tabelą czytelnicy.

Specyfikacja atrybutów encji recenzja

Nazwa atrybutu	Typ danych(maksymalny rozmiar)	Wartość wymagana (TAK/NIE)	Wartość unikatowa (TAK/NIE)	Opis atrybutu
IdRecenzja	INT(11)	Tak	Tak	Atrybut umożliwiający sprawdzenie miedzy innymi ilość recenzji dla danej książki
ocena	ENUM('1', '2', '3', '4', '5')	Nie	Nie	Przechowuje wartość liczbową określającą poziom

System biblioteczny

				zadowolenia czytelnika z danej książki.
komentarz	TEXT	Nie	Nie	Przechowuje treść komentarza w którym czytelnik może wyrazić opinie i uwagi dotyczącej książki.
Pozycja_idPozycja	INT(11)	Tak	Tak	Klucz obcy łączący tabelę recenzja z tabelą pozycja.

Specyfikacja atrybutów encji towary

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdTowary	INT(11)	Tak	Tak	Identyfikator towaru.
nazwa	VARCHAR(45)	Tak	Nie	Nazwa towaru.
ilosc	INT(11)	Tak	Nie	Informacja o liczbie książek. Jest ona potrzebna do złożenia zamówienia.
cena	DECIMAL(10, 2)	Tak	Nie	Cena, którą trzeba zapłacić za daną książkę.
wartosc	DECIMAL(10, 2)	Tak	Nie	Suma, którą należy

System biblioteczny

				zapłacić za podaną liczbę sztuk danej książki.
Zamowienie_idZamowienie	INT(11)	Tak	Tak	Klucz obcy łączący towary z tabelą zamówienie.

Specyfikacja atrybutów encji audiobook

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdAudiobook	INT(11)	Tak	Tak	Identyfikator audiobooka.
narrator	VARCHAR(45)	Tak	Nie	Imię i nazwisko narratora.
Rozmiar_pliku	DECIMAL(5, 2)	Tak	Nie	Rozmiar pliku w MB.
Czas_trwania	TIME	Tak	Nie	Czas trwania pliku w formacie HH:MM:SS.
kod	VARCHAR(45)	Tak	Tak	Kod dostępu do pliku.
Pozycja_idPozycja	INT(11)	Tak	Tak	Klucz obcy łączący tabelę audiobook z tabelą pozycja.

Specyfikacja atrybutów encji pensja

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdPensja	INT(11)	Tak	Tak	Identyfikator pensji.
Kwota_podstawa	DECIMAL(10, 0)	Tak	Nie	Kwota, którą pracownik otrzymuje co miesiąc, bez doliczonych premii.
Data_wyplaty	DATE	Tak	Nie	Dzień, w którym pracownik powinientrzymać wynagrodzenie.
premia	DECIMAL(10, 0)	Nie	Nie	Ewentualna kwota premii doliczana do wynagrodzenia pracownika.

Specyfikacja atrybutów encji wydawnictwo

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdWydawnictwo	INT(11)	Tak	Tak	Identyfikator wydawnictwa.
nazwa	VARCHAR(45)	Tak	Tak	Nazwa wydawnictwa.
Data_zalozenia	DATE	Tak	Nie	Informacja, kiedy zostało

System biblioteczny

				założone wydawnictwo.
NIP	BIGINT(20)	Tak	Tak	Numer NIP wydawnictwa.
REGON	BIGINT(20)	Tak	Tak	Numer REGON wydawnictwa.
KRS	BIGINT(20)	Tak	Tak	Numer KRS wydawnictwa.
adres	VARCHAR(45)	Tak	Nie	Adres siedziby wydawnictwa.

Specyfikacja atrybutów encji wypożyczenie

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdWypozyczenie	INT(11)	Tak	Tak	Identyfikator wypożyczenia.
Data_wypozyczenia	DATE	Tak	Nie	Data wypożyczenia książki.
Data_zwrotu	DATE	Nie	Nie	Data oddania książki.
Czytelnik_idCzytelnik	INT(11)	Tak	Tak	Klucz obcy łączący tabelę wypożyczenie z tabelą czytelnik.

Specyfikacja atrybutów encji pracownik_filia

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagana (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdPracownik_filia	INT(11)	Tak	Tak	Identyfikator pracownika filii.
imie	VARCHAR(45)	Tak	Nie	Imię pracownika.
nazwisko	VARCHAR(45)	Tak	Nie	Nazwisko pracownika.
Filia_idFilia	INT(11)	Tak	Tak	Klucz obcy łączący tabelę pracownik filii z tabelą filia.
Pensja_idPensja	INT(11)	Tak	Tak	Klucz obcy łączący tabelę pracownik filii z tabelą pensja.

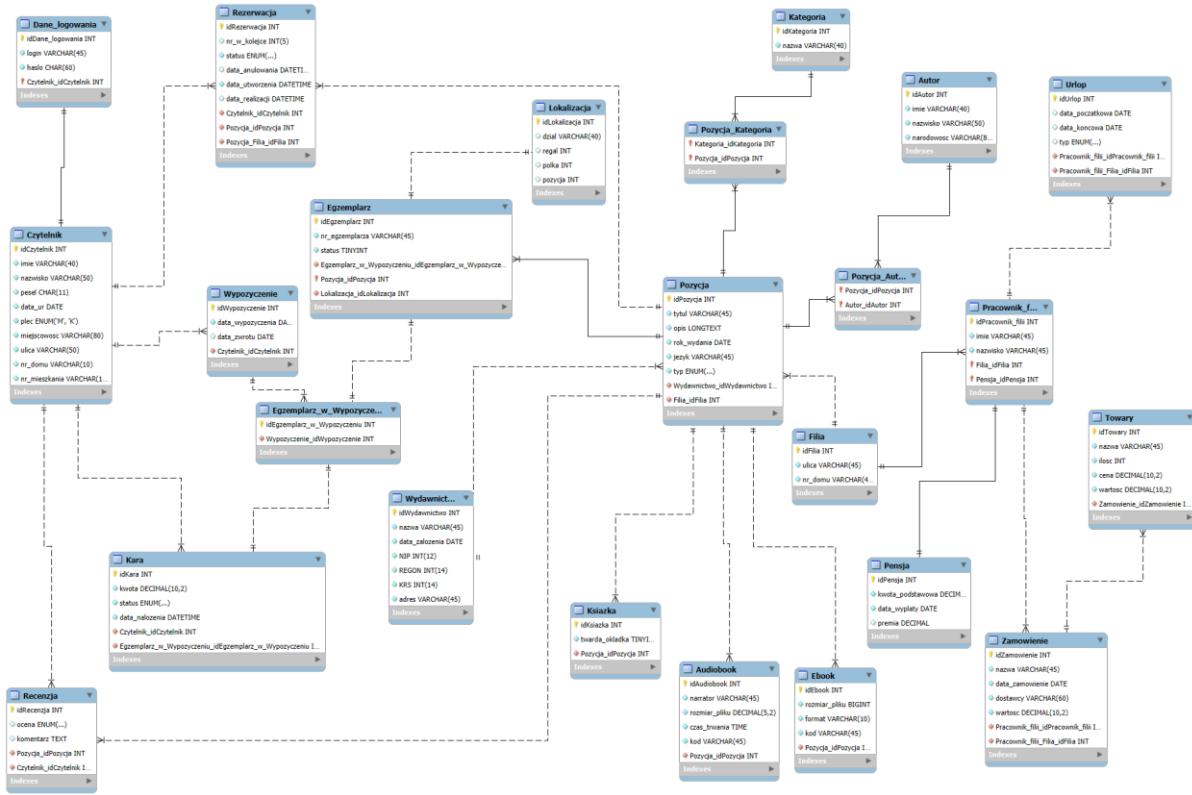
Specyfikacja atrybutów encji pozycja_kategoria

Nazwa atrybutu	Typ danych i maksymalny rozmiar	Wartość wymagan a (Tak/Nie)	Wartość unikatowa (Tak/Nie)	Opis atrybutu
IdPracownik_filia	INT(11)	Tak	Tak	Identyfikator pracownika filii.
imie	VARCHAR(45)	Tak	Nie	Imię pracownika.
nazwisko	VARCHAR(45)	Tak	Nie	Nazwisko pracownika.
Filia_idFilia	INT(11)	Tak	Tak	Klucz obcy łączący tabelę

System biblioteczny

				pracownik filii z tabelą filia.
Pensja_idPensja	INT(11)	Tak	Tak	Klucz obcy łączący tabelę pracownik filii z tabelą pensja.

ROZDZIAŁ 4. MODEL STRUKTURY BAZY DANYCH.



Rysunek 1 Model struktury bazy systemu bibliotecznego

ROZDZIAŁ 5. WSTAWIANIE REKORDÓW DO BAZY DANYCH.

Tabela Audiobook

```
INSERT INTO `audiobook` (`idAudiobook`, `narrator`, `rozmiar_pliku`, `czas_trwania`, `kod`, `Pozycja_idPozycja`) VALUES  
(1, 'Jan Kowalski', 350.45, '10:15:32', 'AUDIO123', 1),  
(2, 'Anna Nowak', 200.78, '07:20:10', 'AUDIO124', 2),  
(3, 'Piotr Wiśniewski', 450.50, '12:45:00', 'AUDIO125', 3),  
(4, 'Katarzyna Lewandowska', 300.00, '09:30:45', 'AUDIO126', 4),  
(5, 'Michał Zieliński', 150.25, '05:10:15', 'AUDIO127', 5),  
(6, 'Ewa Kamińska', 275.00, '08:50:20', 'AUDIO128', 6),  
(7, 'Tomasz Nowicki', 400.30, '11:05:50', 'AUDIO129', 7),  
(8, 'Magdalena Wojciechowska', 180.15, '06:25:33', 'AUDIO130', 8),  
(9, 'Rafał Kwiatkowski', 360.10, '10:55:45', 'AUDIO131', 9),  
(10, 'Joanna Szymańska', 225.75, '07:40:00', 'AUDIO132', 10);
```

Tabela Autor

```
INSERT INTO `autor` (`idAutor`, `imie`, `nazwisko`, `narodowosc`) VALUES  
(1, 'Adam', 'Mickiewicz', 'Polska'),  
(2, 'J.K.', 'Rowling', 'Wielka Brytania'),  
(3, 'George', 'Orwell', 'Wielka Brytania'),  
(4, 'Maria', 'Konopnicka', 'Polska'),  
(5, 'Franz', 'Kafka', 'Czechy'),  
(6, 'Gabriel', 'Garcia Marquez', 'Kolumbia'),  
(7, 'Albert', 'Camus', 'Francja'),  
(8, 'Haruki', 'Murakami', 'Japonia'),  
(9, 'Homer', '', 'Grecja'),  
(10, 'Jane', 'Austen', 'Wielka Brytania');
```

Tabela Dane Logowania

```
INSERT INTO `dane_logowania` (`idDane_logowania`, `login`, `haslo`, `Czytelnik_idCzytelnik`) VALUES  
(1, 'jankowalski', '$2y$10$C4p/9pFgLs3X/Wvv7vU8iJ07kh0qewq5hzV0myv9FwF8MQF5noY2i', 1),  
(2, 'annanowak', '$2y$10$P5aZn/Dy9v6l0fQldPjsQwq9Iwdcz0m.zkzqhd5C7r9uAc5AV/kS', 2),  
(3, 'piotrwisniewski', '$2y$10$QpTfOgk2Nq7Ys9c0jlCUy79t.FYZjwHF4lXbPyIRu9D80x1.I5y9S', 3),  
(4, 'katarzynazielinska', '$2y$10$HySxY1JKLQ7KzK7tb/AbIxSwN6M9hMw14g.b7sLbfj1OYX7KNOSw2', 4),  
(5, 'michallewandowski', '$2y$10$gG7Xp5zXqZD9PiWxR0z6jdsfUGFvvACa9b6rpwINxMffj4oa.ZwWe', 5),
```

System biblioteczny

```
(6, 'ewakowalczyk',
 '$2y$10$4sUbjEqw7Hs.kOX0jzMPvQIU8NjDQOhVhd4IghZpwdB2.XhhDUuAi', 6),
(7, 'tomaszzawadzki',
 '$2y$10$5qNKQKtB6IQmLzGTeFQL2Q2nfyQ0ZMGsQkFmV3YbSz6J2CTm7T8u', 7),
(8, 'magdalenaszymanska',
 '$2y$10$LSXQFiXczPfS0PqUyDqAi8RwiYF87E9v.ZfgkCG5qV9m7P0a5lJ30', 8),
(9, 'robertmazur',
 '$2y$10$g6owIkLxKLwwMwI8W4FPHXVfBqAywfQ.EJcHkOEnZ5gdJccmPR7b6', 9),
(10, 'joannadabrowska',
 '$2y$10$M10Hz9K43HzbIOL1w02d/UdWmdGbQsUjp/Hi6tyVnPvqtgnwJX1K', 10);
```

Tabela Czytelnik

```
INSERT INTO `czytelnik`(`idCzytelnik`, `imie`, `nazwisko`, `pesel`,
`data_ur`, `plec`, `miejscowosc`, `ulica`, `nr_domu`, `nr_mieszkania`) VALUES
(1, 'Jan', 'Kowalski', '92031412345', '1992-03-14', 'M', 'Warszawa',
'Kwiatowa', '15', '2'),
(2, 'Anna', 'Nowak', '85051254321', '1985-05-12', 'K', 'Kraków', 'Lipowa',
'10', '4A'),
(3, 'Piotr', 'Wiśniewski', '96010167890', '1996-01-01', 'M', 'Poznań',
'Długa', '7', '3B'),
(4, 'Katarzyna', 'Zielińska', '88122398765', '1988-12-23', 'K', 'Wróclaw',
'Słoneczna', '22', '1'),
(5, 'Michał', 'Lewandowski', '93111445678', '1993-11-14', 'M', 'Gdańsk',
'Kolejowa', '5', '6'),
(6, 'Ewa', 'Kowalczyk', '79040723456', '1979-04-07', 'K', 'Lublin',
'Warszawska', '12', '7'),
(7, 'Tomasz', 'Zawadzki', '85090567890', '1985-09-05', 'M', 'Łódź', 'Zielona',
'3', '1'),
(8, 'Magdalena', 'Szymańska', '91021654321', '1991-02-16', 'K', 'Szczecin',
'Leśna', '18', '3'),
(9, 'Robert', 'Mazur', '80120812345', '1980-12-08', 'M', 'Bydgoszcz', 'Cicha',
'21', '5'),
(10, 'Joanna', 'Dąbrowska', '93071045678', '1993-07-10', 'K', 'Katowice',
'Grunwaldzka', '9', '8');
```

Tabela E-book

```
INSERT INTO `ebook`(`idEbook`, `rozmiar_pliku`, `format`, `kod`,
`Pozycja_idPozycja`) VALUES
(1, 150000000, 'pdf', 'A1B2C3D4E5', 1),
(2, 200000000, 'epub', 'F6G7H8I9J0', 2),
(3, 120000000, 'pdf', 'K1L2M3N4O5', 3),
(4, 180000000, 'epub', 'P6Q7R8S9T0', 4),
(5, 250000000, 'mobi', 'U1V2W3X4Y5', 5),
(6, 100000000, 'pdf', 'Z6A7B8C9D0', 6),
(7, 300000000, 'epub', 'E1F2G3H4I5', 7),
(8, 500000000, 'pdf', 'J6K7L8M9N0', 8),
(9, 700000000, 'mobi', 'O1P2Q3R4S5', 9),
```

System biblioteczny

```
(10, 450000000, 'epub', 'T6U7V8W9X0', 10);
```

Tabela **Egzemplarz**

```
INSERT INTO `egzemplarz` (`idEgzemplarz`, `nr_egzemplarza`, `status`,  
`Pozycja_idPozycja`, `Lokalizacja_idLokalizacja`) VALUES  
(1, 'EGZ-001', 1, 1, 1),  
(2, 'EGZ-002', 0, 1, 2),  
(3, 'EGZ-003', 1, 2, 3),  
(4, 'EGZ-004', 0, 3, 4),  
(5, 'EGZ-005', 1, 4, 5),  
(6, 'EGZ-006', 0, 5, 6),  
(7, 'EGZ-007', 1, 6, 7),  
(8, 'EGZ-008', 0, 7, 8),  
(9, 'EGZ-009', 1, 8, 9),  
(10, 'EGZ-010', 0, 9, 10);
```

Tabela **Egzemplarz w wypożyczeniu**

```
INSERT INTO `egzemplarz_w_wypożyczeniu` (`idEgzemplarz_w_Wypożyczeniu`,  
`Wypożyczenie_idWypożyczenie`, `Egzemplarz_idEgzemplarz`) VALUES  
(1, 1, 1),  
(2, 1, 2),  
(3, 2, 3),  
(4, 2, 4),  
(5, 3, 5),  
(6, 3, 6),  
(7, 4, 7),  
(8, 4, 8),  
(9, 5, 9),  
(10, 5, 10);
```

Tabela **Filia**

```
INSERT INTO `filia` (`idFilia`, `ulica`, `nr_domu`) VALUES  
(1, 'ul. Przykładowa', '10'),  
(2, 'ul. Nowa', '5A'),  
(3, 'ul. Kwiatowa', '12B'),  
(4, 'ul. Stara', '3'),  
(5, 'ul. Morska', '7'),  
(6, 'ul. Zielona', '10A'),  
(7, 'ul. Jasna', '15'),  
(8, 'ul. Wesoła', '8'),  
(9, 'ul. Długa', '4'),  
(10, 'ul. Krótka', '20');
```

System biblioteczny

Tabela Kara

```
INSERT INTO `kara` (`idKara`, `kwota`, `status`, `data_nalozenia`,  
`Czytelnik_idCzytelnik`, `Wypozyzczenie_Egzemplarz_idWypozyzczenie_Egzemplarz`)  
VALUES  
(1, 10.00, 'niezapłacona', '2025-03-07 08:30:00', 1, 1),  
(2, 15.50, 'zapłacona', '2025-03-06 14:20:00', 2, 2),  
(3, 5.00, 'niezapłacona', '2025-03-05 10:15:00', 3, 3),  
(4, 20.00, 'zapłacona', '2025-03-04 16:45:00', 4, 4),  
(5, 25.00, 'niezapłacona', '2025-03-03 09:00:00', 5, 5),  
(6, 12.75, 'zapłacona', '2025-03-02 18:30:00', 6, 6),  
(7, 8.50, 'niezapłacona', '2025-03-01 11:25:00', 7, 7),  
(8, 30.00, 'zapłacona', '2025-02-28 13:40:00', 8, 8),  
(9, 18.25, 'niezapłacona', '2025-02-27 17:00:00', 9, 9),  
(10, 22.50, 'zapłacona', '2025-02-26 15:10:00', 10, 10);
```

Tabela Kategoria

```
INSERT INTO `kategoria` (`idKategoria`, `nazwa`) VALUES  
(1, 'Fikcja'),  
(2, 'Nauka'),  
(3, 'Historia'),  
(4, 'Podróże'),  
(5, 'Biografia'),  
(6, 'Kryminał'),  
(7, 'Fantastyka'),  
(8, 'Poradniki'),  
(9, 'Psychologia'),  
(10, 'Dziecięca');
```

Tabela Książka

```
INSERT INTO `ksiazka` (`idKsiazka`, `twarda_okladka`, `Pozycja_idPozycja`)  
VALUES  
(1, 1, 1),  
(2, 0, 2),  
(3, 1, 3),  
(4, 0, 4),  
(5, 1, 5),  
(6, 0, 6),  
(7, 1, 7),  
(8, 0, 8),  
(9, 1, 9),  
(10, 0, 10);
```

Tabela **Lokalizacja**

```
INSERT INTO `lokalizacja` (`idLokalizacja`, `dzial`, `regal`, `polka`, `pozycja`) VALUES
(1, 'Literatura piękna', 1, 2, 3),
(2, 'Historia', 2, 3, 4),
(3, 'Nauki ścisłe', 3, 1, 5),
(4, 'Technologia', 4, 2, 6),
(5, 'Sztuka', 5, 1, 2),
(6, 'Dziecięca', 6, 3, 1),
(7, 'Podręczniki', 7, 4, 5),
(8, 'Fantastyka', 8, 2, 3),
(9, 'Biografie', 9, 3, 4),
(10, 'Kuchnia', 10, 1, 6);
```

Tabela **Pensja**

```
INSERT INTO `pensja` (`idPensja`, `kwota_podstawowa`, `data_wyplaty`, `premia`) VALUES
(1, 3000, '2025-03-07', 500),
(2, 3200, '2025-02-28', 450),
(3, 2800, '2025-01-31', 400),
(4, 3500, '2025-03-05', 600),
(5, 2900, '2025-02-15', 300),
(6, 3100, '2025-03-01', 550),
(7, 3300, '2025-02-20', 650),
(8, 3400, '2025-03-10', 700),
(9, 3000, '2025-01-15', 350),
(10, 3250, '2025-02-05', 500);
```

Tabela **Pozycja**

```
INSERT INTO `pozycja` (`idPozycja`, `tytul`, `opis`, `rok_wydania`, `jazyk`, `typ`, `Wydawnictwo_idWydawnictwo`, `Filia_idFilia`) VALUES
(1, 'Wielka Przygoda', 'Opis książki o wielkiej przygodzie.', '2022-05-01', 'Polski', 'ksiazka', 1, 1),
(2, 'Nauka Programowania', 'Podstawy programowania w różnych językach.', '2021-10-15', 'Polski', 'ksiazka', 2, 2),
(3, 'Wielki Wojownik', 'Opowieść o wojowniku walczącym z mrocznymi siłami.', '2020-07-30', 'Polski', 'audiobook', 3, 3),
(4, 'Zielony Las', 'Relaksująca książka o życiu w lesie.', '2019-08-05', 'Polski', 'e-book', 4, 4),
(5, 'Matematyka dla Każdego', 'Nauka matematyki od podstaw.', '2021-11-21', 'Polski', 'ksiazka', 5, 5),
(6, 'Historie Miłosne', 'Zbiór opowiadań miłosnych.', '2018-12-12', 'Polski', 'audiobook', 6, 6),
(7, 'Sekrety Kosmosu', 'Przewodnik po tajemnicach kosmosu.', '2020-04-18', 'angielski', 'ksiazka', 7, 7),
(8, 'Skrzynia Skarbów', 'Przygody dzieci w poszukiwaniu skarbów.', '2019-03-11', 'Polski', 'e-book', 8, 8),
```

System biblioteczny

```
(9, 'Mistrzostwo w Sporcie', 'Jak osiągnąć sukces w różnych dyscyplinach sportowych.', '2022-01-20', 'Polski', 'ksiazka', 9, 9),
(10, 'Tajemnice Starożytnego Egiptu', 'Opis najważniejszych odkryć archeologicznych w Egipcie.', '2017-09-05', 'Angielski', 'audiobook', 10, 10);
```

Tabela Pozycja Autor

```
INSERT INTO `pozycja_autor` (`Pozycja_idPozycja`, `Autor_idAutor`) VALUES
(1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10);
```

Tabela Pozycja kategoria

```
INSERT INTO `pozycja_kategoria` (`Kategoria_idKategoria`, `Pozycja_idPozycja`)
VALUES
(1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10);
```

Tabela Pracownik filii

```
INSERT INTO `pracownik_fili` (`idPracownik_fili`, `imie`, `nazwisko`,
`Filia_idFilia`, `Pensja_idPensja`) VALUES
(1, 'Jan', 'Kowalski', 1, 1),
(2, 'Anna', 'Nowak', 2, 2),
(3, 'Piotr', 'Wójcik', 3, 3),
(4, 'Maria', 'Kaczmarek', 1, 4),
(5, 'Krzysztof', 'Lewandowski', 2, 5),
(6, 'Ewa', 'Zielińska', 3, 6),
(7, 'Marek', 'Szymański', 1, 7),
(8, 'Katarzyna', 'Jankowska', 2, 8),
(9, 'Paweł', 'Witkowski', 3, 9),
(10, 'Magdalena', 'Kwiatkowska', 1, 10);
```

Tabela **Recenzja**

```
INSERT INTO `recenzja` (`idRecenzja`, `ocena`, `komentarz`, `Pozycja_idPozycja`) VALUES
(1, '5', 'Niezwykle wciągająca pozycja, zdecydowanie polecam!', 1),
(2, '4', 'Dobra książka, choć momentami nieco przewidywalna.', 2),
(3, '3', 'Średnia fabuła, ale dobrze napisana.', 3),
(4, '5', 'Rewelacyjny audiobook, narrator sprawił, że historia ożyła.', 4),
(5, '2', 'Zbyt długi, trudno było przez niego przebrnąć.', 5),
(6, '1', 'Nie polecam, zmarnowany potencjał.', 6),
(7, '4', 'E-book bardzo wygodny w użytkowaniu, dobrze zredagowany.', 7),
(8, '5', 'Najlepsza książka, jaką czytałem w tym roku!', 8),
(9, '3', 'Przeciętna historia, ale zakończenie było dobre.', 9),
(10, '4', 'Bardzo interesująca, choć momentami nudząca.', 10);
```

Tabela **Rezerwacja**

```
INSERT INTO `rezerwacja` (`idRezerwacja`, `nr_w_kolejce`, `status`, `data_anulowania`, `data_utworzenia`, `data_realizacji`, `Czytelnik_idCzytelnik`, `Pozycja_idPozycja`, `Pozycja_Filia_idFilia`) VALUES
(41, 1, 'oczekujaca', NULL, '2025-03-01 00:00:00', NULL, 1, 1, 1),
(42, 2, 'zrealizowana', NULL, '2025-03-02 00:00:00', '2025-03-05 00:00:00', 2, 2, 2),
(43, 3, 'oczekujaca', NULL, '2025-03-03 00:00:00', NULL, 3, 3, 3),
(44, 4, 'anulowana', '2025-03-04 00:00:00', '2025-03-04 00:00:00', NULL, 4, 4, 4),
(45, 5, 'zrealizowana', NULL, '2025-03-05 00:00:00', '2025-03-07 00:00:00', 5, 5, 5),
(46, 6, 'oczekujaca', NULL, '2025-03-06 00:00:00', NULL, 6, 6, 6),
(47, 7, 'zrealizowana', NULL, '2025-03-07 00:00:00', '2025-03-10 00:00:00', 7, 7, 7),
(48, 8, 'anulowana', '2025-03-08 22:00:00', '2025-03-08 00:00:00', NULL, 8, 8, 8),
(49, 9, 'oczekujaca', NULL, '2025-03-09 00:00:00', NULL, 9, 9, 9),
(50, 10, 'zrealizowana', NULL, '2025-03-10 00:00:00', '2025-03-12 00:00:00', 10, 10, 10);
```

Tabela **Towary**

```
INSERT INTO `towary` (`idTowary`, `nazwa`, `ilosc`, `cena`, `wartosc`, `Zamowienie_idZamowienie`) VALUES
(1, 'Towar A', 10, 15.50, 155.00, 1),
(2, 'Towar B', 5, 25.00, 125.00, 2),
(3, 'Towar C', 20, 10.00, 200.00, 3),
(4, 'Towar D', 3, 30.00, 90.00, 4),
(5, 'Towar E', 8, 20.75, 166.00, 5),
(6, 'Towar F', 12, 12.50, 150.00, 6),
(7, 'Towar G', 6, 40.00, 240.00, 7),
(8, 'Towar H', 15, 18.60, 279.00, 8),
(9, 'Towar I', 10, 22.30, 223.00, 9),
```

```
(10, 'Towar J', 7, 35.00, 245.00, 10);
```

Tabela **Urlop**

```
INSERT INTO `urlop` (`idUrlop`, `data_poczatkowa`, `data_koncowa`, `typ`, `Pracownik_filii_idPracownik_filii`, `Pracownik_filii_Filia_idFilia`) VALUES  
(1, '2025-05-01', '2025-05-14', 'wypoczynkowy', 1, 1),  
(2, '2025-06-01', '2025-06-10', 'chorobowy', 2, 2),  
(3, '2025-07-05', '2025-07-12', 'macierzynski', 3, 3),  
(4, '2025-08-01', '2025-08-15', 'tacierzynski', 4, 1),  
(5, '2025-09-01', '2025-09-07', 'wypoczynkowy', 5, 2),  
(6, '2025-10-05', '2025-10-20', 'wypoczynkowy', 6, 3),  
(7, '2025-11-10', '2025-11-25', 'chorobowy', 7, 1),  
(8, '2025-12-01', '2025-12-10', 'macierzynski', 8, 2),  
(9, '2026-01-01', '2026-01-15', 'tacierzynski', 9, 3),  
(10, '2026-02-01', '2026-02-10', 'innny', 10, 1);
```

Tabela **Wydawnictwo**

```
INSERT INTO `wydawnictwo` (`idWydawnictwo`, `nazwa`, `data_zalozenia`, `NIP`, `REGON`, `KRS`, `adres`) VALUES  
(1, 'Wydawnictwo ABC', '2005-03-10', 1234567890, 9876543210, 1122334455, 'ul. Przykładowa 1, Warszawa'),  
(2, 'Wydawnictwo XYZ', '2010-06-15', 2345678901, 8765432109, 2233445566, 'ul. Nowa 5, Kraków'),  
(3, 'Wydawnictwo DEF', '2015-09-22', 3456789012, 7654321098, 3344556677, 'ul. Kwiatowa 12, Wrocław'),  
(4, 'Wydawnictwo GHI', '2000-01-01', 4567890123, 6543210987, 4455667788, 'ul. Stara 3, Gdańsk'),  
(5, 'Wydawnictwo JKL', '2018-11-05', 5678901234, 5432109876, 5566778899, 'ul. Morska 7, Poznań'),  
(6, 'Wydawnictwo MNO', '2012-02-18', 6789012345, 4321098765, 6677889900, 'ul. Zielona 10, Łódź'),  
(7, 'Wydawnictwo PQR', '2017-07-30', 7890123456, 3210987654, 7788990011, 'ul. Jasna 15, Lublin'),  
(8, 'Wydawnictwo STU', '2003-08-22', 8901234567, 2109876543, 8899001122, 'ul. Wesoła 8, Gdynia'),  
(9, 'Wydawnictwo VWX', '2014-10-10', 9012345678, 1098765432, 9900112233, 'ul. Długa 4, Białystok'),  
(10, 'Wydawnictwo YZA', '2020-05-15', 1123456789, 8765432101, 1010101010, 'ul. Krótka 20, Szczecin');
```

Tabela **Wypożyczenie**

```
INSERT INTO `wypożyczenie` (`idWypożyczenie`, `data_wypożyczenia`,  
`data_zwrotu`, `Czytelnik_idCzytelnik`) VALUES  
(1, '2025-02-01', '2025-02-15', 1),  
(2, '2025-02-02', '2025-02-16', 2),  
(3, '2025-02-05', '2025-02-20', 3),  
(4, '2025-02-10', '2025-02-24', 4),  
(5, '2025-02-12', '2025-02-26', 5),  
(6, '2025-02-14', '2025-02-28', 6),  
(7, '2025-02-16', '2025-03-02', 7),  
(8, '2025-02-18', '2025-03-03', 8),  
(9, '2025-02-20', '2025-03-05', 9),  
(10, '2025-02-22', '2025-03-07', 10);
```

Tabela **Zamówienie**

```
INSERT INTO `zamówienie` (`idZamówienie`, `nazwa`, `data_zamówienie`,  
`dostawcy`, `wartosc`, `Pracownik_filiu_idPracownik_filiu`,  
`Pracownik_filiu_Filia_idFilia`) VALUES  
(1, 'Zamówienie A', '2025-03-01', 'Dostawca 1', 150.50, 1, 1),  
(2, 'Zamówienie B', '2025-03-02', 'Dostawca 2', 250.75, 2, 2),  
(3, 'Zamówienie C', '2025-03-03', 'Dostawca 3', 120.30, 3, 3),  
(4, 'Zamówienie D', '2025-03-04', 'Dostawca 4', 185.60, 4, 1),  
(5, 'Zamówienie E', '2025-03-05', 'Dostawca 5', 300.00, 5, 2),  
(6, 'Zamówienie F', '2025-03-06', 'Dostawca 6', 220.90, 6, 3),  
(7, 'Zamówienie G', '2025-03-07', 'Dostawca 7', 400.40, 7, 1),  
(8, 'Zamówienie H', '2025-03-08', 'Dostawca 8', 150.20, 8, 2),  
(9, 'Zamówienie I', '2025-03-09', 'Dostawca 9', 180.10, 9, 3),  
(10, 'Zamówienie J', '2025-03-10', 'Dostawca 10', 320.00, 10, 1);
```

System biblioteczny

ROZDZIAŁ 8. PREZENTACJA ZAPYTAŃ POBIERAJĄCYCH DANE Z BAZY.

1. Ile autorów pochodzi z każdego kraju.

```
1 •  SELECT narodosc, COUNT(*) AS liczba_autorow
2   FROM autor
3   GROUP BY narodosc
4   ORDER BY liczba_autorow DESC;
```

narodosc	liczba_autorow
Wielka Brytania	3
Polska	2
Czechy	1
Kolumbia	1
Francja	1
Japonia	1
Grecja	1

Rysunek 2 Kwerenda: Ile autorów pochodzi z każdego kraju

2. Średni wiek czytelników dla każdego miasta.

```
1 •  SELECT miejscowosc, ROUND(AVG(wiek)) AS sredni_wiek
2   FROM (
3     SELECT miejscowosc, TIMESTAMPDIFF(YEAR, data_ur, CURDATE()) AS wiek
4     FROM czytelnik
5   ) AS subquery
6   GROUP BY miejscowosc;
```

miejscowosc	sredni_wiek
Bydgoszcz	44
Gdańsk	31
Katowice	31
Kraków	39
Lublin	45
Poznań	29
Szczecin	34
Warszawa	32
Wrocław	36
Łódź	39

Rysunek 3 Kwerenda: Średni wiek czytelników dla każdego miasta

3. Miasta, w których mieszka mniej niż 2 czytelników.

```
1 •  SELECT miejscowosc, COUNT(*) AS liczba_czytelnikow
2   FROM czytelnik
3   GROUP BY miejscowosc
4   HAVING COUNT(*) < 2;
```

miejscowosc	liczba_czytelnikow
Bydgoszcz	1
Gdańsk	1
Katowice	1
Kraków	1
Lublin	1
Poznań	1
Szczecin	1
Warszawa	1
Wrocław	1
Łódź	1

Rysunek 4 Kwerenda: Miasta, w których mieszka mniej niż 2 czytelników

System biblioteczny

4. Pobranie wszystkich autorów oraz czytelników w jednej liście.

```
1 •  SELECT imie, nazwisko, 'Autor' AS typ FROM autor
2     UNION
3     SELECT imie, nazwisko, 'Czytelnik' AS typ FROM czytelnik;
```

The screenshot shows a 'Result Grid' window from MySQL Workbench. The grid has three columns: 'imie', 'nazwisko', and 'typ'. The data consists of two groups of rows. The first group, under 'typ' 'Autor', contains 15 rows of author names. The second group, under 'typ' 'Czytelnik', contains 15 rows of reader names. The rows are listed in a single column, separated by a vertical line between the two types.

imie	nazwisko	typ
Adam	Mickiewicz	Autor
J.K.	Rowling	Autor
George	Orwell	Autor
Maria	Konopnicka	Autor
Franz	Kafka	Autor
Gabriel	Garcia Marquez	Autor
Albert	Camus	Autor
Haruki	Murakami	Autor
Homer		Autor
Jane	Austen	Autor
Jan	Kowalski	Czyt...
Anna	Nowak	Czyt...
Piotr	Wiśniewski	Czyt...
Katarz...	Zielńska	Czyt...
Michał	Lewandowski	Czyt...
Ewa	Kowalczyk	Czyt...
Tomasz	Zawadzki	Czyt...
Magda...	Szymańska	Czyt...
Robert	Mazur	Czyt...
Joanna	Dąbrowska	Czyt...

Rysunek 5 Kwerenda: Pobranie wszystkich autorów oraz czytelników w jednej liście

5. Autorzy, którzy mają mniej niż 3 książki.

```
1 •  SELECT a.imie, a.nazwisko
2     FROM autor a
3     WHERE (SELECT COUNT(*) FROM pozycja_autor pa WHERE pa.Autor_idAutor = a.idAutor) < 3;
```

The screenshot shows a 'Result Grid' window from MySQL Workbench. The grid has two columns: 'imie' and 'nazwisko'. It displays 7 rows of author names, each consisting of a first name and a last name. These authors have fewer than 3 books in the database.

imie	nazwisko
Adam	Mickiewicz
J.K.	Rowling
George	Orwell
Maria	Konopnicka
Franz	Kafka
Gabriel	Garcia Marquez
Albert	Camus
Haruki	Murakami
Jane	Austen

Rysunek 6 Kwerenda: Autorzy, którzy mają mniej niż 3 książki

6. Średnia ocena dla każdej kategorii książek z przynajmniej jedną recenzją.

```
1 •  SELECT k.nazwa AS Kategoria, AVG(r.ocena) AS Srednia_Ocena
2     FROM kategoria k
3     JOIN pozycja_kategoria pk ON k.idKategoria = pk.Kategoria_idKategoria
4     JOIN pozycja p ON pk.Pozycja_idPozycja = p.idPozycja
5     JOIN recenzja r ON p.idPozycja = r.Pozycja_idPozycja
6     GROUP BY k.nazwa
7     HAVING COUNT(r.idRecenzja) >= 1;
```

The screenshot shows a 'Result Grid' window from MySQL Workbench. The grid has two columns: 'Kategoria' and 'Srednia_Ocena'. It lists 10 categories of books along with their average review scores. The categories include Biografia, Dziecięca, Fantastyka, Fikcja, Historia, Kryminał, Nauka, Podróże, Poradniki, and Psychologia. The average scores range from 1.0000 to 5.0000.

Kategoria	Srednia_Ocena
Biografia	2.0000
Dziecięca	4.0000
Fantastyka	4.0000
Fikcja	5.0000
Historia	3.0000
Kryminał	1.0000
Nauka	4.0000
Podróże	5.0000
Poradniki	5.0000
Psychologia	3.0000

Rysunek 7 Kwerenda: Średnia ocena dla każdej kategorii książek z przynajmniej jedną recenzją

System biblioteczny

7. Średnia ocena książek i audiobooków.

```
1 •  SELECT p.typ, AVG(r.ocena) AS Srednia_Ocena
2   FROM pozycja p
3   INNER JOIN recenzja r ON p.idPozycja = r.Pozycja_idPozycja
4   WHERE p.typ IN ('ksiazka', 'audiobook')
5   GROUP BY p.typ;
```

typ	Srednia_Ocena	
ksiazka	3.6000	
audiobook	2.6667	

Rysunek 8 Kwerenda: Średnia ocena książek i audiobooków

8. Czytelnicy z największą liczbą wypożyczeń.

```
1 •  SELECT c.imie, c.nazwisko, COUNT(w.idWypozyczenie) AS Liczba_Wypozyczen
2   FROM czytelnik c
3   JOIN wypozyczenie w ON c.idCzytelnik = w.Czytelnik_idCzytelnik
4   GROUP BY c.imie, c.nazwisko
5   HAVING COUNT(w.idWypozyczenie) = (
6       SELECT MAX(liczba_wypozyczen)
7       FROM (
8           SELECT COUNT(idWypozyczenie) AS liczba_wypozyczen
9           FROM wypozyczenie
10          GROUP BY Czytelnik_idCzytelnik
11      ) AS max_wypozyczenia
12  );
```

imie	nazwisko	Liczba_Wypozyczen
Anna	Nowak	1
Ewa	Kowalczyk	1
Jan	Kowalski	1
Joanna	Dąbrowska	1
Katarzyna	Zielńska	1
Magdalena	Szymańska	1
Michał	Lewandowski	1
Piotr	Wiśniewski	1
Robert	Mazur	1
Tomasz	Zawadzki	1

Rysunek 9 Kwerenda: Czytelnicy z największą liczbą wypożyczeń

9. Suma nałożonych kar na czytelników w każdej miejscowości.

```
1 •  SELECT c.miejscowosc, SUM(k.kwota) AS Suma_Kar
2   FROM czytelnik c
3   INNER JOIN kara k ON c.idCzytelnik = k.Czytelnik_idCzytelnik
4   GROUP BY c.miejscowosc;
```

miejscowosc	Suma_Kar
Bydgoszcz	18.25
Gdańsk	25.00
Katowice	22.50
Kraków	15.50
Lublin	12.75
Poznań	5.00
Szczecin	30.00
Warszawa	10.00
Wrocław	20.00
Łódź	8.50

Rysunek 10 Kwerenda: Suma nałożonych kar na czytelników w każdej miejscowości

System biblioteczny

10. Liczba książek z największą liczbą rezerwacji.

```
1 •   SELECT p.tytul, COUNT(r.idRezerwacja) AS Liczba_Rezerwacji
2     FROM pozycja p
3     INNER JOIN rezerwacja r ON p.idPozycja = r.Pozycja_idPozycja
4     GROUP BY p.tytul
5     HAVING COUNT(r.idRezerwacja) = (
6         SELECT MAX(liczba_rezerwacji)
7         FROM (
8             SELECT COUNT(idRezerwacja) AS liczba_rezerwacji
9                 FROM rezerwacja
10                GROUP BY Pozycja_idPozycja
11            ) AS max_rezerwacje
12        ))
```

Result Grid	
Filter Rows: <input type="text"/>	
Export: <input type="button"/> Wrap Cell Content: <input type="checkbox"/>	
tytul	Liczba_Rezerwacji
Historie Młosne	1
Matematyka dla Każdego	1
Mistrzostwo w Sporcie	1
Nauka Programowania	1
Sekrety Kosmosu	1
Skrzynia Skarbów	1
Tajemnice Starożytne Egiptu	1
Wielka Przygoda	1
Wielki Wojownik	1
Zielony Las	1

Rysunek 11 Kwerenda: Liczba książek z największą liczbą rezerwacji

11. Liczba książek przypisanych do każdego autora.

```
1 •   SELECT a.imie, a.nazwisko, COUNT(pa.Pozycja_idPozycja) AS liczba_ksiazek
2     FROM autor a
3     LEFT JOIN pozycja_autor pa ON a.idAutor = pa.Autor_idAutor
4     GROUP BY a.imie, a.nazwisko
5     ORDER BY liczba_ksiazek DESC;
```

Result Grid		
Filter Rows: <input type="text"/>		
Export: <input type="button"/> Wrap Cell Content: <input type="checkbox"/>		
imie	nazwisko	liczba_ksiazek
Adam	Mickiewicz	1
Homer		1
J.K.	Rowling	1
Jane	Austen	1
George	Orwell	1
Maria	Konopnicka	1
Franz	Kafka	1
Gabriel	Garcia Marquez	1
Albert	Camus	1
Haruki	Murakami	1

Rysunek 12 Kwerenda: Liczba książek przypisanych do każdego autora

12. Średnia długość audiobooków dla każdego narratora.

```
1 •   SELECT narrator, SEC_TO_TIME(AVG(TIME_TO_SEC(czas_trwania))) AS sredni_czas
2     FROM audiobook
3     GROUP BY narrator;
```

Result Grid	
Filter Rows: <input type="text"/>	
Export: <input type="button"/> Wrap Cell Content: <input type="checkbox"/>	
narrator	sredni_czas
Anna Nowak	07:20:10.0000
Ewa Kamińska	08:50:20.0000
Jan Kowalski	10:15:32.0000
Joanna Szymańska	07:40:00.0000
Katarzyna Lewandowska	09:30:45.0000
Małgorzata Wojciechowska	06:25:33.0000
Michał Zieliński	05:10:15.0000
Piotr Wiśniewski	12:45:00.0000
Rafał Kwiatkowski	10:55:45.0000
Tomasz Nowicki	11:05:50.0000

Rysunek 13 Kwerenda: Średnia długość audiobooków dla każdego narratora

System biblioteczny

13. Książki, które są zarówno dostępne jako e-book, jak i audiobook.

```
1 •  SELECT p.tytul
2   FROM pozycja p
3   JOIN ebook e ON p.idPozycja = e.Pozycja_idPozycja
4   JOIN audiobook a ON p.idPozycja = a.Pozycja_idPozycja;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	tytul			
▶	Wielka Przygoda			
	Nauka Programowania			
	Wielki Wojownik			
	Zielony Las			
	Matematyka dla Każdego			
	Historie Miłosne			
	Sekrety Kosmosu			
	Skrzynia Skarbów			
	Mistrzostwo w Sporcie			
	Tajemnice Starożytnego Egiptu			

Rysunek 14 Kwerenda: Książki, które są zarówno dostępne jako e-book, jak i audiobook

14. Czytelnicy, którzy wypożyczyli książki z różnych filii.

```
1 •  SELECT c.imie, c.nazwisko, COUNT(DISTINCT f.idFilia) AS liczba_fili
2   FROM czytelnik c
3   JOIN wypozczenie w ON c.idCzytelnik = w.Czytelnik_idCzytelnik
4   JOIN egzemplarz_w_wypozczeniu ew ON w.idWypozczenie = ew.Wypozczenie_idWypozczenie
5   JOIN egzemplarz e ON ew.Egzemplarz_idEgzemplarz = e.idEgzemplarz
6   JOIN filia f ON e.Lokalizacja_idLokalizacja = f.idFilia
7   GROUP BY c.imie, c.nazwisko
8   HAVING liczba_fili > 1
9   ORDER BY liczba_fili DESC;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	imie	nazwisko	liczba_fili		
▶	Piotr	Wiśniewski	2		
	Anna	Nowak	2		
	Jan	Kowalski	2		
	Katarzyna	Zielńska	2		
	Michał	Lewandowski	2		

Rysunek 15 Kwerenda: Czytelnicy, którzy wypożyczyli książki z różnych filii

15. Najczęściej wypożyczane kategorie książek.

```
1 •  SELECT k.nazwa AS kategoria, COUNT(w.idWypozczenie) AS liczba_wypozczen
2   FROM kategoria k
3   JOIN pozycja_kategoria pk ON k.idKategoria = pk.Kategoria_idKategoria
4   JOIN pozycja p ON pk.Pozycja_idPozycja = p.idPozycja
5   JOIN egzemplarz e ON p.idPozycja = e.Pozycja_idPozycja
6   JOIN egzemplarz_w_wypozczeniu ew ON e.idEgzemplarz = ew.Egzemplarz_idEgzemplarz
7   JOIN wypozczenie w ON ew.Wypozczenie_idWypozczenie = w.idWypozczenie
8   GROUP BY k.nazwa
9   ORDER BY liczba_wypozczen DESC
10  LIMIT 5;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
	kategoria	liczba_wypozczen			
▶	Fikcja	2			
	Historia	1			
	Biografia	1			
	Fantastyka	1			
	Psychologia	1			

Rysunek 16 Kwerenda: Najczęściej wypożyczane kategorie książek

ROZDZIAŁ 9. PREZENTACJA ZAPYTAŃ EDYTUJĄCYCH I USUWAJĄCYCH DANE.

Zapytanie edytujące dane:

Zwiększenie kwoty podstawowej o 10% dla pracowników, którzy pracują w filii przy ul. Nowa

```
UPDATE pensja
SET kwota_podstawowa = kwota_podstawowa * 1.10
WHERE idPensja IN (
    SELECT Pensja_idPensja
    FROM pracownik_filii
    WHERE Filia_idFilia =
        (SELECT idFilia FROM filia WHERE ulica = 'ul. Nowa')
)
;|
```

Rysunek 17 Kwerenda edytujące dane

Zapytanie usuwające dane:

Usunięcie wydawnictw, które nie wydały żadnych książek

```
DELETE FROM wydawnictwo
WHERE idWydawnictwo NOT IN (
    SELECT DISTINCT Wydawnictwo_idWydawnictwo
    FROM pozycja
    WHERE Wydawnictwo_idWydawnictwo IS NOT NULL
)
;|
```

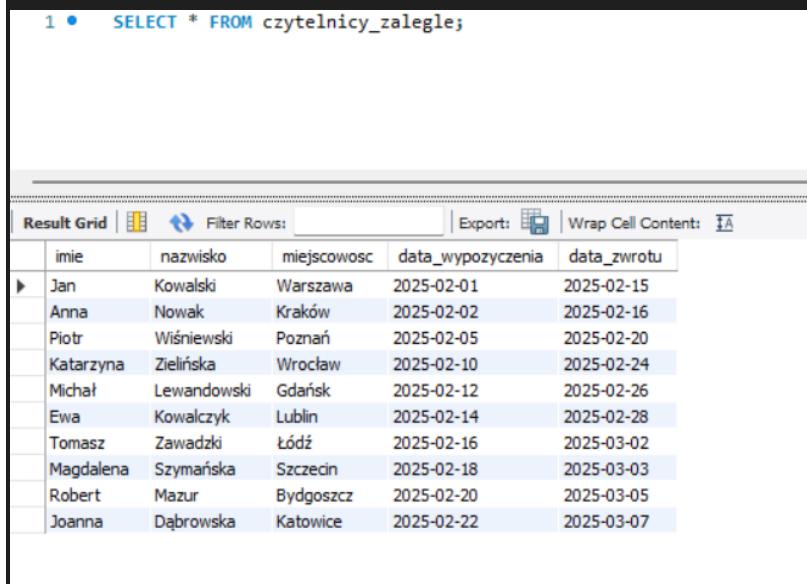
Rysunek 18 Kwerenda usuwające dane

ROZDZIAŁ 10. WIDOKI, WYZWALACZE I PROCEDURY SKŁADOWE.

Widoki w SQL to wirtualne tabele, które są tworzone na podstawie wyników zapytań do bazy danych. Nie przechowują one danych samodzielnie, lecz działają jak zapisane zapytania, które generują dane dynamicznie w momencie ich użycia. Poniżej przedstawiamy widoki, które ilustrują sposób organizacji i przetwarzania danych w systemie bibliotecznym.

1. Widok – Czytelnicy, którzy mają zaległe książki.

```
CREATE VIEW Czytelnicy_Zalegle AS
SELECT
    c.imie,
    c.nazwisko,
    c.miejscowosc,
    w.data_wypozyczenia,
    w.data_zwrotu
FROM czytelnik c
JOIN wypozyczenie w ON c.idCzytelnik = w.Czytelnik_idCzytelnik
WHERE w.data_zwrotu IS NULL OR w.data_zwrotu < CURDATE();
```



The screenshot shows a database interface with a SQL editor and a results grid. The SQL editor contains the code for creating a view named 'Czytelnicy_Zalegle'. Below it, a command '1 • SELECT * FROM czytelnicy_zalegle;' is shown. The results grid displays ten rows of data, each representing a reader with outstanding loans. The columns are labeled: imie, nazwisko, miejscowosc, data_wypozyczenia, and data_zwrotu. The data includes names like Jan Kowalski, Anna Nowak, Piotr Wiśniewski, etc., along with their loan details.

	imie	nazwisko	miejscowosc	data_wypozyczenia	data_zwrotu
▶	Jan	Kowalski	Warszawa	2025-02-01	2025-02-15
	Anna	Nowak	Kraków	2025-02-02	2025-02-16
	Piotr	Wiśniewski	Poznań	2025-02-05	2025-02-20
	Katarzyna	Zielńska	Wrocław	2025-02-10	2025-02-24
	Michał	Lewandowski	Gdańsk	2025-02-12	2025-02-26
	Ewa	Kowalczyk	Lublin	2025-02-14	2025-02-28
	Tomasz	Zawadzki	Łódź	2025-02-16	2025-03-02
	Magdalena	Szymańska	Szczecin	2025-02-18	2025-03-03
	Robert	Mazur	Bydgoszcz	2025-02-20	2025-03-05
	Joanna	Dąbrowska	Katowice	2025-02-22	2025-03-07

Rysunek 19 Widok - Czytelnicy, którzy mają zaległe książki

2. Widok - Książki i audiobooki, które mają ocenę powyżej 4.

```

CREATE VIEW dobrze_oceniane AS
SELECT p.tytul, 'ksiazka' AS Typ, AVG(r.ocena) AS Srednia_Ocena
FROM pozycja p
INNER JOIN recenzja r ON p.idPozycja = r.Pozycja_idPozycja
WHERE p.typ = 'ksiazka'
GROUP BY p.tytul
HAVING AVG(r.ocena) > 4

UNION

SELECT p.tytul, 'audiobook' AS Typ, AVG(r.ocena) AS Srednia_Ocena
FROM pozycja p
INNER JOIN recenzja r ON p.idPozycja = r.Pozycja_idPozycja
WHERE p.typ = 'audiobook'
GROUP BY p.tytul
HAVING AVG(r.ocena) > 4;

```

1 • select * FROM `dobrze_oceniane`;
2

tytul	Typ	Srednia_Ocena
Wielka Przygoda	ksiazka	5.0000

Rysunek 20 Widok - Książki i audiobooki, które mają ocenę powyżej 4

3. Widok - Liczba zamówień podzielone na poszczególne miesiące.

```

CREATE VIEW `ZamowieniaMiesiecznie` AS
SELECT
    YEAR(data_zamowienie) AS Rok,
    MONTH(data_zamowienie) AS Miesiac,
    COUNT(*) AS IloscZamowien
FROM
    zamowienie
GROUP BY
    YEAR(data_zamowienie),
    MONTH(data_zamowienie)
ORDER BY
    Rok DESC, Miesiac DESC;

```

System biblioteczny

```
1  SELECT * FROM zamowieniamiesiecznie;
2
```

The screenshot shows a MySQL query result grid. At the top, there are buttons for 'Result Grid' (selected), 'Edit', 'Export', and 'Wrap Cell Content'. Below the header, there are three columns: 'Rok' (Year), 'Miesiac' (Month), and 'IloscZamowien' (Number of Orders). A single row is displayed: 2025, 3, 10.

Rok	Miesiac	IloscZamowien
2025	3	10

Rysunek 21 Widok - Liczba zamówień podzielone na poszczególne miesiące

Wyzwalacze w SQL to specjalne procedury, które są automatycznie wykonywane w odpowiedzi na określone zdarzenia zachodzące w bazie danych, takie jak dodanie, modyfikacja lub usunięcie rekordu. Wyzwalacze są powiązane z konkretnymi tabelami i działają na poziomie wiersza lub tabeli.

1. Wyzwalacz – Aktualizacja liczby egzemplarzy po dodaniu nowej pozycji

```
DELIMITER //

CREATE TRIGGER after_insert_pozycja
AFTER INSERT ON pozycja
FOR EACH ROW
BEGIN
    INSERT INTO egzemplarz (nr_egzemplarza, status, Pozycja_idPozycja,
Lokalizacja_idLokalizacja)
    VALUES (CONCAT('EGZ-', NEW.idPozycja), 1, NEW.idPozycja, 1); -- Przykładowa lokalizacja (idLokalizacja = 1)
END //

DELIMITER ;
```

```
INSERT INTO pozycja (idPozycja, tytul, opis, rok_wydania, jezyk, typ,
Wydawnictwo_idWydawnictwo, Filia_idFiliala)
VALUES (11, 'Nowa Testowa Książka', 'Opis nowej testowej książki',
'2023-10-01', 'polski', 'ksiazka', 1, 1);
```

The screenshot shows a MySQL query result grid. At the top, there are buttons for 'Result Grid' (selected), 'Edit', 'Export/Import', and 'Wrap Cell Content'. Below the header, there are six columns: 'idEgzemplarz', 'nr_egzemplarza', 'status', 'Pozycja_idPozycja', and 'Lokalizacja_idLokalizacja'. A single row is displayed: 0, EGZ-11, 1, 11, 1.

idEgzemplarz	nr_egzemplarza	status	Pozycja_idPozycja	Lokalizacja_idLokalizacja
0	EGZ-11	1	11	1

Rysunek 22 Wyzwalacz – Aktualizacja liczby egzemplarzy po dodaniu nowej pozycji

2. Wyzwalacz - Sprawdzanie, czy nowa wartość premii nie przekracza 25% kwoty podstawowej

```
DELIMITER //
CREATE TRIGGER `before_update_pensja`
BEFORE UPDATE ON `pensja`
FOR EACH ROW
BEGIN
    IF NEW.premia > NEW.kwota_podstawowa * 0.25 THEN
        SET NEW.premia = NEW.kwota_podstawowa * 0.25;
    END IF;
END //

DELIMITER ;
```

```
UPDATE `pensja`
SET `premia` = 5000
WHERE `idPensja` = 1;
```

The screenshot shows a MySQL Workbench interface. In the top-left pane, there is a text area containing two SQL statements: an UPDATE statement and a SELECT statement. The SELECT statement is highlighted with a red dot and the number '1'. In the bottom-right pane, there is a 'Result Grid' showing the results of the SELECT query.

	idPensja	kwota_podstawowa	data_wyplaty	premia
▶	1	3000	2025-03-07	750
*	NULL	NULL	NULL	NULL

Rysunek 23 Wyzwalacz - Sprawdzanie, czy nowa wartość premii nie przekracza 25% kwoty podstawowej

3. Wyzwalacz - Zablokowanie dodania ebooka w nieobsługiwany formacie.

```
DELIMITER //

CREATE TRIGGER format_ebooka
BEFORE UPDATE ON ebook
FOR EACH ROW
BEGIN
    IF NEW.format NOT IN ('pdf', 'epub', 'mobi') THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Nieobsługiwany format e-booka.';
    END IF;
END //

DELIMITER ;
```

The screenshot shows the MySQL Workbench interface. On the left, there is a code editor window containing the SQL script for creating a trigger named 'format_ebooka'. The trigger checks if the 'format' column of an updated row is not one of 'pdf', 'epub', or 'mobi'. If it's not, it signals an error with state '45000' and sets a message text. The script ends with a delimiter change and a final semicolon. On the right, there is an 'Output' tab showing the results of an 'INSERT' query. The query inserts a new row into the 'ebook' table with id 12, size 12345678, format 'docx', and position 'EBOOK124'. The output shows a warning message: 'Error Code: 1644. Nieobsługiwany format e-booka.' (Unsupported file format).

```
1 •  INSERT INTO ebook (idEbook, rozmiar_pliku, format, kod, Pozycja_idPozycja)
2   VALUES (12, 12345678, 'docx', 'EBOOK124', 2);
3

Output:-----
Action Output
# Time Action Message
1 23:21:14 INSERT INTO ebook (idEbook, rozmiar_pliku, format, kod, Pozycja... Error Code: 1644. Nieobsługiwany format e-booka.
```

Rysunek 24 Wyzwalacz - Zablokowanie dodania ebooka w nieobsługiwany formacie

Procedury składowane w SQL to zdefiniowane zestawy instrukcji SQL, które są zapisywane i przechowywane bezpośrednio w bazie danych. Umożliwiają one realizację bardziej złożonych operacji w sposób uporządkowany i wielokrotnie powtarzalny, co czyni je ważnym elementem zaawansowanego zarządzania danymi.

1. Procedura składowa - Przyjmuje parametr ID czytelnika i zwraca listę wypożyczonych książek wraz z terminem zwrotu.

```
CREATE DEFINER=`root`@`localhost` PROCEDURE
`GetWypozyczeniaCzytelnika`(IN czytelnik_id INT)
BEGIN
    SELECT
        p.tytul AS Tytul,
        w.data_wypozyczenia AS Data_Wypozyczenia,
        w.data_zwrotu AS Data_Zwrotu
    FROM wypozyczenie w
    JOIN egzemplarz_w_wypozyczeniu ew ON w.idWypozyczenie =
ew.Wypozyczenie_idWypozyczenie
        JOIN egzemplarz e ON ew.Egzemplarz_idEgzemplarz = e.idEgzemplarz
        JOIN pozycja p ON e.Pozycja_idPozycja = p.idPozycja
        WHERE w.Czytelnik_idCzytelnik = czytelnik_id;
END
```

1 • CALL GetWypozyczeniaCzytelnika(2);

Result Grid			
	Tytul	Data_Wypozyczenia	Data_Zwrotu
▶	Nauka Programowania	2025-02-02	2025-02-16
	Wielki Wojownik	2025-02-02	2025-02-16

Rysunek 25 Procedura składowa - Przyjmuje parametr ID czytelnika i zwraca listę wypożyczonych książek wraz z terminem zwrotu

2. Procedura składowa - Pobiera dane o pensji dla określonego idPensja.

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `pobierz_pensja`(IN p_idPensja INT)
BEGIN
    SELECT * FROM pensja WHERE idPensja = p_idPensja;
END
```

```
1 • CALL pobierz_pensja(1);
2
```

	idPensja	kwota_podstawowa	data_wyplaty	premia
▶	1	3000	2025-03-07	500

Rysunek 26 Procedura składowa - Pobiera dane o pensji dla określonego idPensja

3. Procedura składowa - Zwraca łączną kwotę niezapłaconych kar dla wskazanego czytelnika.

```
DELIMITER //
CREATE PROCEDURE suma_niezaplaconych_kar(
IN p_id_czytelnik INT
)
BEGIN
    SELECT SUM(kwota) AS suma_kar
FROM kara
WHERE Czytelnik_idCzytelnik = p_id_czytelnik AND status =
`niezapłacona`;
END //
DELIMETER ;
```

```
1 • CALL suma_niezaplaconych_kar(1);
```

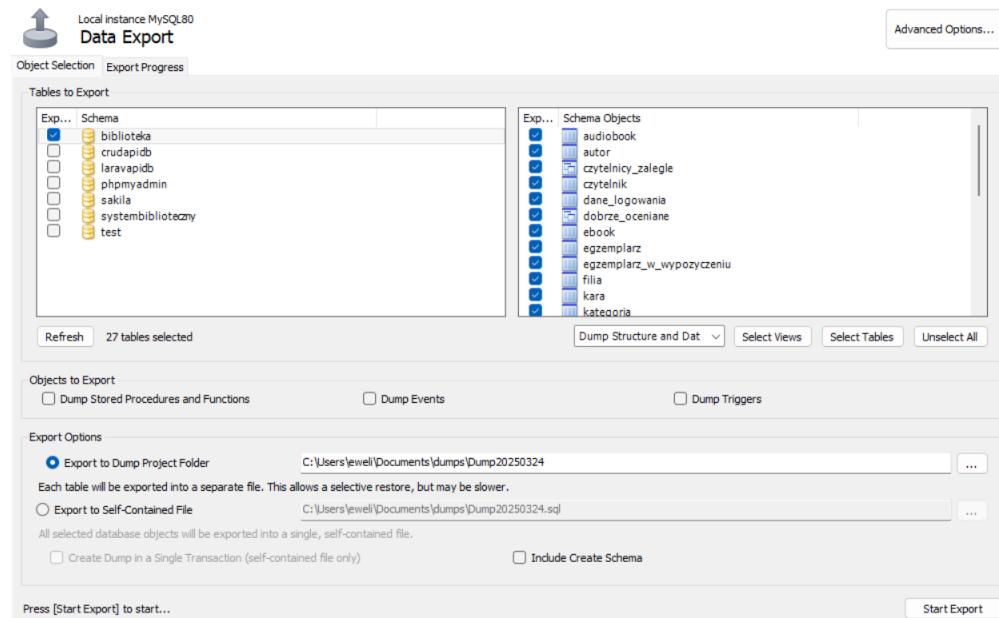
	suma_kar
▶	10.00

Rysunek 27 Procedura składowa - Zwraca łączną kwotę niezapłaconych kar dla wskazanego czytelnika

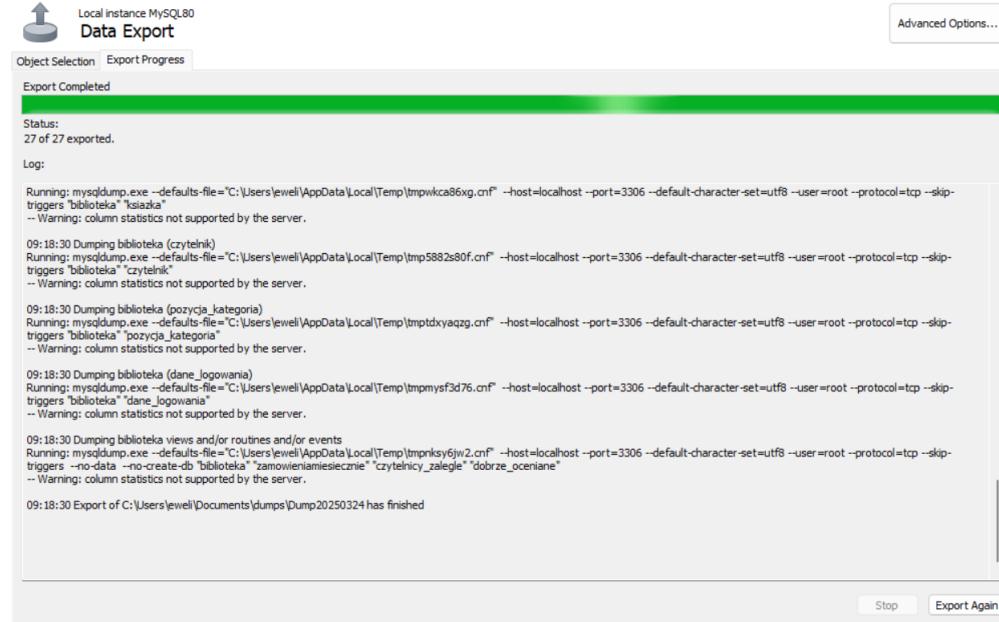
System biblioteczny

ROZDZIAŁ 11. KOPIA ZAPASOWA, IMPORT ORAZ EKSPORT.

Eksportowanie danych:



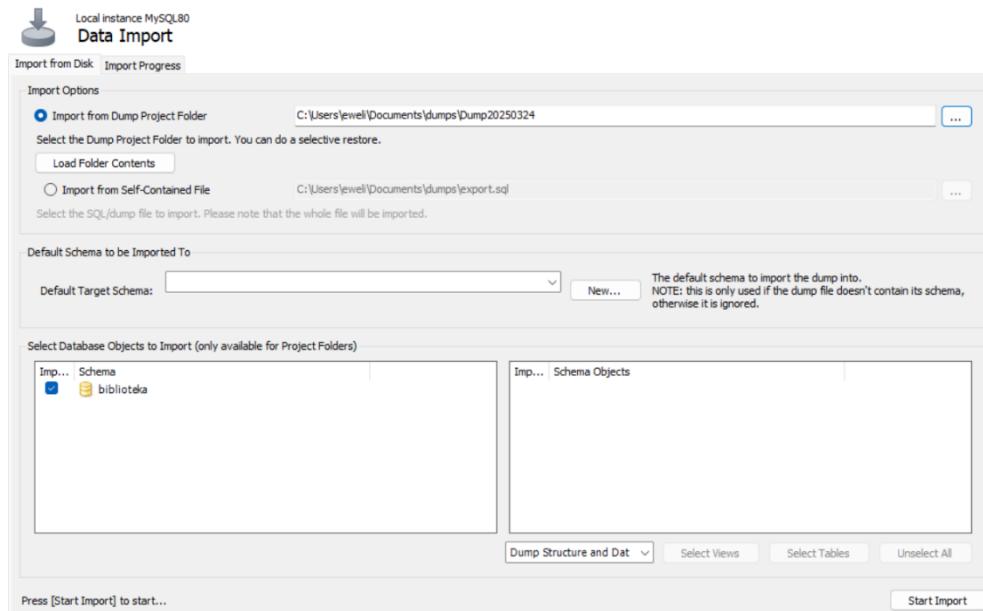
Rysunek 28 Wybór bazy do eksportu do oddzielnych plików



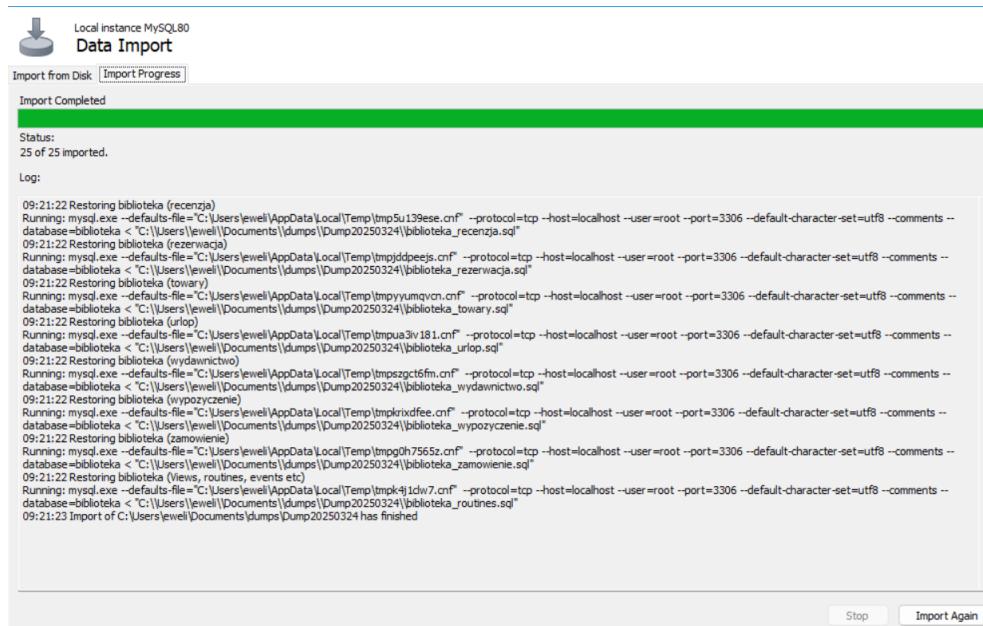
Rysunek 29 Komunikat udanego eksportu danych

System biblioteczny

Import bazy danych:



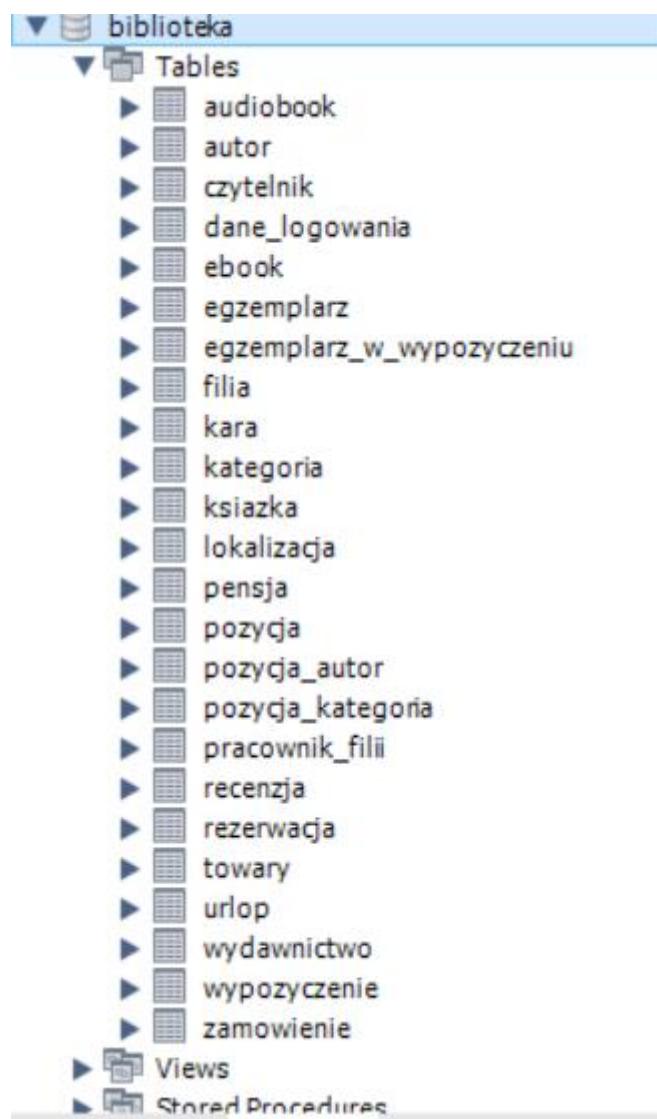
Rysunek 30 Wybór bazy do importu



Rysunek 31 Komunikat o sukcesie importu

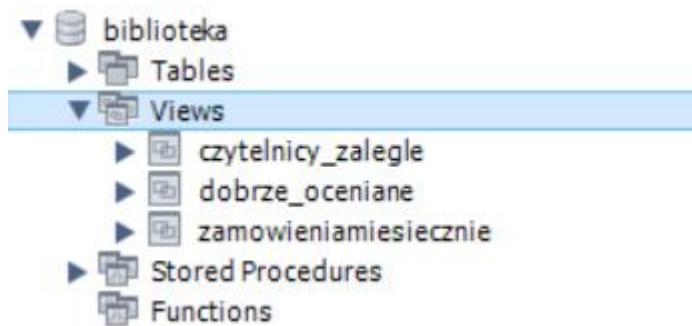
System biblioteczny

Sprawdzenie czy baza danych została poprawnie zimportowana:

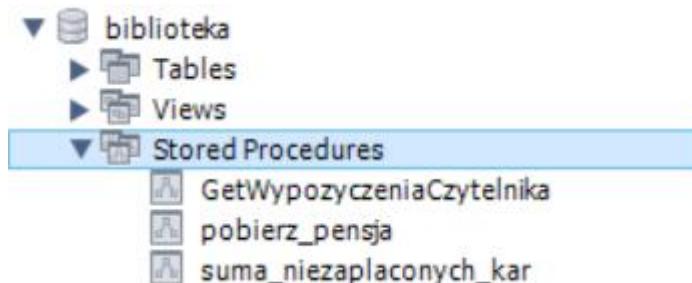


Rysunek 32 Widok na wszystkie tabele z bazy

System biblioteczny



Rysunek 33 Widok na wszystkie widoki z bazy



Rysunek 34 Widok na wszystkie procedury z bazy

ROZDZIAŁ 12. ZARZĄDZANIE UŻYWKOWNIKAMI.

1. Stażysta

Jego uprawnienia będą ograniczone, aby uniknąć wykonania niepożądanych operacji. Hasło do konta to: N3J[H1@FOIU3Z/BY

The screenshot shows the 'Administration - Users and Privileges' window in MySQL Workbench. On the left, a list of existing users is shown. On the right, a form for creating a new user account 'newuser@%'. The 'Login' tab is selected. The 'Login Name' field contains 'stazysta'. The 'Authentication Type' dropdown is set to 'Standard'. The 'Limit to Hosts Matching' field contains '%'. The 'Password' and 'Confirm Password' fields both contain '*****'. Buttons at the bottom include 'Add Account', 'Delete', 'Refresh', 'Revert', and 'Apply'.

Rysunek 35 Tworzenie użytkownika o nazwie stazysta

System biblioteczny

Local instance MySQL80
Users and Privileges

User Accounts

User	From Host
(!) <anonymous>	%
pma	localhost
root	127.0.0.1
root	::1
sakila1	localhost
sakila2	%
stazysta	%

Details for account stazysta@%

Schema Schema Privileges
systembiblioteczny CREATE TEMPORARY TABLES, GRANT OPTION, LOCK TABLES, SELECT, SHOW VIEW, TRIGGER,

Schema and Host fields may use % and _ wildcards.
The server will match specific entries before wildcarded ones.

The user 'stazysta'@'%' will have the following access rights to the schema 'systembiblioteczny':

Object Rights DDL Rights Other Rights

SELECT CREATE
 INSERT ALTER
 UPDATE REFERENCES
 DELETE INDEX
 EXECUTE CREATE VIEW
 SHOW VIEW CREATE ROUTINE
 ALTER ROUTINE
 EVENT
 DROP
 TRIGGER

GRANT OPTION
 CREATE TEMPORARY TABLES
 LOCK TABLES

Add Account Delete Refresh Revert Apply

Rysunek 36 Wybór uprawnień dla stażysty

Tworzę nowe połączenie w którym dostęp do bazy ma stażysta nie mający uprawnień do tworzenia u usuwania tabel.

Setup New Connection

Connection Name: Biblioteka dla stazysty

Connection Method: Standard (TCP/IP)

Parameters SSL Advanced

Hostname: 127.0.0.1

Username: stazysta

Password: Store in Vault ...

Default Schema:

Connect to MySQL Server

Please enter password for the following service:

Service: Mysql@127.0.0.1:3306

User: stazysta

Password: *****

Save password in vault

OK Cancel

Configure Server Management... Test Connection Cancel OK

Rysunek 37 Tworzenie nowego połączenia z dostępem dla stażysty

System biblioteczny

Sprawdzam aktualnie zalogowanego użytkownika:

```
1 •  SELECT USER();
```

USER()	stazysta@localhost
USER()	stazysta@localhost

Rysunek 38 Aktualnie zalogowany użytkownik

Przy próbie usunięcia tabeli widać, że nie można tego zrobić:

```
1 •  DROP TABLE `biblioteka`.`dane_logowania`;
```

#	Time	Action	Message
1	21:33:50	DROP TABLE `biblioteka`.`dane_logowania`	Error Code: 1142. DROP command denied to user 'stazysta'@'localhost' for table 'biblioteka`.`dane_logowania'

Rysunek 39 Próba usunięcia tabeli z poziomu stażysty

Stażysta nie ma również uprawnień do tworzenia nowych tabel:

```
1 •  CREATE TABLE `biblioteka`.`kosz`;
```

#	Time	Action	Message
1	21:33:50	DROP TABLE `biblioteka`.`dane_logowania`	Error Code: 1142. DROP command denied to user 'stazysta'@'localhost' for table 'biblioteka`.`dane_logowania'
2	21:34:52	DROP TABLE `dane_logowania`	Error Code: 1046. No database selected Select the default DB to be used by double-clicking its name in the SCHEM
3	21:35:34	DROP TABLE `biblioteka`.`dane_logowania`	Error Code: 1142. DROP command denied to user 'stazysta'@'localhost' for table 'biblioteka`.`dane_logowania'
4	21:49:25	CREATE TABLE `biblioteka`.`kosz`	Error Code: 1142. CREATE command denied to user 'stazysta'@'localhost' for table 'biblioteka`.`kosz'

Rysunek 40 Próba utworzenia tabeli z poziomu stażysty

Ma jedynie możliwość do zapytań typu SELECT, UPDATE, TRIGGER oraz SHOW VIEW

```
1 •  SELECT * FROM `biblioteka`.`kategoria`;
```

idKategoria	nazwa
1	Fikcja
2	Nauka
3	Historia
4	Podróże
5	Biografia
6	Kryminał
7	Fantastyka
8	Poradniki
9	Psychologia
10	Dziecięca
*	NULL
*	NULL

Rysunek 41 Kwerenda typu SELECT z poziomu stażysty

System biblioteczny

2. Root

Posiada wszystkie możliwe uprawnienia. To konto nie posiada wymaganego hasła.

The screenshot shows the 'Details for account root@localhost' screen in MySQL Workbench. The 'Schema Privileges' tab is selected. On the left, a list of roles is shown with checkboxes, all of which are checked: DBA, MaintenanceAdmin, ProcessAdmin, UserAdmin, SecurityAdmin, MonitorAdmin, DBManager, DBDesigner, ReplicationAdmin, BackupAdmin, and Custom. On the right, a list of global privileges is shown, also with all checkboxes checked: ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TABLESPACE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, FILE, GRANT OPTION, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, and RELOAD.

Rysunek 42 Utworzenie konta root ze wszystkimi uprawnieniami

Korzystanie z zapytań typu SELECT:

```
1 •   SELECT * FROM ebook;
```

The screenshot shows the 'Result Grid' window in MySQL Workbench displaying the output of the 'SELECT * FROM ebook;' query. The table has columns: idEbook, rozmiar_pliku, format, kod, and Pozycja_idPozycja. The data consists of 10 rows of eBook entries, each with a unique ID, file size, format, identifier, and position information. The last row is a footer row with all columns labeled 'NULL'.

	idEbook	rozmiar_pliku	format	kod	Pozycja_idPozycja
▶	1	150000000	pdf	A1B2C3D4E5	1
	2	200000000	epub	F6G7H8I9J0	2
	3	120000000	pdf	K1L2M3N4O5	3
	4	180000000	epub	P6Q7R8S9T0	4
	5	250000000	mobi	U1V2W3X4Y5	5
	6	100000000	pdf	Z6A7B8C9D0	6
	7	300000000	epub	E1F2G3H4I5	7
	8	500000000	pdf	J6K7L8M9N0	8
	9	700000000	mobi	O1P2Q3R4S5	9
*	10	450000000	epub	T6U7V8W9X0	10
*	NULL	NULL	NULL	NULL	NULL

Rysunek 43 Kwerenda typu SELECT z poziomu roota

System biblioteczny

Tworzenie nowych tabel:

```
1 CREATE TABLE `przyklad` (
2     `id` INT AUTO_INCREMENT PRIMARY KEY,
3     `nazwa` VARCHAR(100) NOT NULL,
4     `data_utworzenia` TIMESTAMP DEFAULT CURRENT_TIMESTAMP
5 );
6

Output
Action Output
# Time Action Message
1 23:03:13 SELECT * FROM ebook LIMIT 0, 1000 10 row(s) returned
2 23:08:01 CREATE TABLE `przyklad` Error Code: 1113. A table must have at least 1 column
3 23:08:49 CREATE TABLE `przyklad` ( `id` INT AUTO_INCREMENT PRIMARY KEY, `nazwa` VARCHAR(10... 0 row(s) affected
```

Rysunek 44 Tworzenie tabeli z poziomu roota

Wstawianie wartości do tabeli:

```
1 • INSERT INTO `przyklad` (`nazwa`, `data_utworzenia`)
2 VALUES
3 ('Przykład 1', '2025-03-23 10:00:00'),
4 ('Przykład 2', '2025-03-23 11:00:00'),
5 ('Przykład 3', '2025-03-23 12:00:00');
6
7

Output
Action Output
# Time Action Message
1 23:03:13 SELECT * FROM ebook LIMIT 0, 1000 10 row(s) returned
2 23:08:01 CREATE TABLE `przyklad` Error Code: 1113. A table must have at least 1 column
3 23:08:49 CREATE TABLE `przyklad` ( `id` INT AUTO_INCREMENT PRIMARY KEY, `nazwa` VARCHAR(10... 0 row(s) affected
4 23:09:45 INSERT INTO `przyklad` (`nazwa`, `data_utworzenia`) VALUES ('Przykład 1', '2025-03-23 10:00:00'), ('... 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
```

Rysunek 45 Wstawianie wartości do tabeli z poziomu roota

Usuwanie tabeli:

```
1 • DROP TABLE `przyklad`;
2
3

Output
Action Output
# Time Action Message
1 23:03:13 SELECT * FROM ebook LIMIT 0, 1000 10 row(s) returned
2 23:08:01 CREATE TABLE `przyklad` Error Code: 1113. A table must have at least 1 column
3 23:08:49 CREATE TABLE `przyklad` ( `id` INT AUTO_INCREMENT PRIMARY KEY, `nazwa` VARCHAR(10... 0 row(s) affected
4 23:09:45 INSERT INTO `przyklad` (`nazwa`, `data_utworzenia`) VALUES ('Przykład 1', '2025-03-23 10:00:00'), ('... 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
5 23:10:54 DROP TABLE `przyklad` 0 row(s) affected
```

Rysunek 46 Usuwanie tabeli z poziomu roota

System biblioteczny

3. Bambik

Użytkownik o tej nazwie będzie miał możliwość tylko do używania zapytań typu SELECT. Hasło do konta tego użytkownika to: admin

The screenshot shows the 'Details for account Bambik@%' dialog in MySQL Workbench. The 'Schema Privileges' tab is selected. In the 'Role' section, 'Custom' is checked. In the 'Global Privileges' section, only 'SELECT' is checked, while other options like 'EVENT', 'EXECUTE', and 'UPDATE' are unchecked.

Rysunek 47 Tworzenie użytkownika o nazwie bambik z możliwością wykonywania zapytań typu SELECT

Użytkownik nie może usuwać tabel:

The screenshot shows the MySQL Workbench interface. A query 'DROP TABLE `audiobook`;' is run. The output window shows the command and its execution time. An error message indicates that the 'DROP' command is denied to user 'Bambik'@localhost for table 'biblioteka`.`audiobook'.

Rysunek 48 Próba usuwania tabeli z poziomu bambika

Użytkownik nie może tworzyć nowych tabel:

The screenshot shows the MySQL Workbench interface. Two queries are run: 'CREATE TABLE `przyklad` (' and 'CREATE TABLE `przyklad` (`id` INT AUTO_INCREMENT PRIMARY KEY, `nazwa` VARCHAR(100) NOT NULL, `data_utworzenia` TIMESTAMP DEFAULT CURRENT_TIMESTAMP);'. Both queries result in errors indicating that the 'CREATE' command is denied to user 'Bambik'@localhost for table 'biblioteka`.`przyklad'.

Rysunek 49 Próba tworzenia tabeli z poziomu bambika

System biblioteczny

Nie może wstawiać wartości do tabeli:

```
1 •  INSERT INTO `przyklad` (`nazwa`, `data_utworzenia`)
2   VALUES
3   ('Przykład 1', '2025-03-23 10:00:00'),
4   ('Przykład 2', '2025-03-23 11:00:00'),
5   ('Przykład 3', '2025-03-23 12:00:00');
6

Output
Action Output
# Time Action Message
① 1 23:44:40 DROP TABLE `audiobook`' Error Code: 1142. DROP command denied to user 'Bambik'@localhost' for table 'biblioteka`.`audiobook'
② 2 23:45:28 CREATE TABLE `przyklad` ( `id` INT ... Error Code: 1142. CREATE command denied to user 'Bambik'@localhost' for table 'biblioteka`.`przyklad'
③ 3 23:47:42 INSERT INTO `przyklad` ('nazwa', 'data... Error Code: 1142. INSERT command denied to user 'Bambik'@localhost' for table 'biblioteka`.`przyklad'
```

Rysunek 50 Wstawianie wartości do tabeli z poziomu bambika

Użytkownik z tymi uprawnieniami może tylko korzystać z zapytań typu SELECT w każdej bazie danych:

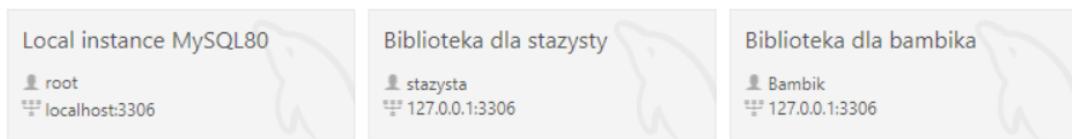
```
1 •  SELECT * FROM filia;
2
```

	idFilia	ulica	nr_domu
▶	1	ul. Przykładowa	10
	2	ul. Nowa	5A
	3	ul. Kwiatowa	12B
	4	ul. Stara	3
	5	ul. Morska	7
	6	ul. Zielona	10A
	7	ul. Jasna	15
	8	ul. Wesoła	8
	9	ul. Długa	4
*	10	ul. Krótka	20
*	NULL	NULL	NULL

Rysunek 51 Zapytanie typu SELECT z poziomu bambika

Podsumowując widzimy 3 połączenia z czego dostęp do nich mają różni użytkownicy:

MySQL Connections



Rysunek 52 Widok na utworzone połączenia

ROZDZIAŁ 13. MIGRACJA DO SQL SERVER.

W tym rozdziale przedstawiamy sposób korzystania z MySQL bezpośrednio z poziomu konsoli. Aby rozpocząć, należy otworzyć wiersz poleceń (CMD) i przejść do katalogu:

C:\Program Files\MySQL Server 9.2\bin.

Następnie, aby uruchomić klienta MySQL i połączyć się z serwerem, wymagane jest podanie nazwy użytkownika, hasła oraz portu.

```
PS C:\Program Files\MySQL\MySQL Server 9.2\bin> .\mysql -u root -p -P 3306
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 9.2.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> |
```

Rysunek 53 Połaczenie z MySQL Server

Następnie tworzę nową bazę danych.

```
mysql> create database BibliotekaNaSerwerze;
Query OK, 1 row affected (0.01 sec)

mysql> use BibliotekaNaSerwerze;
Database changed
mysql> |
```

Rysunek 54 Tworzenie nowej bazy danych

Kolejnym krokiem jest zimportowanie wcześniej wyeksportowanej bazy danych do nowo utworzonej.

```
mysql> source C:\Users\eweli\Downloads\mydb (5).sql
Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

Rysunek 55 Import systemu bibliotecznego

System biblioteczny

Aby upewnić się, że baza danych została pomyślnie utworzona, można sprawdzić jej obecność, a dokładniej tabele jakie zostały do niej dodane.

Tables_in_bibliotekanaserwerze
audiobook
autor
czytelnik
dane_logowania
ebook
egzemplarz
egzemplarz_w_wypozyczeniu
filia
kara
kategoria
ksiazka
lokalizacja
pensja
pozycja
pozycja_autor
pozycja_kategoria
pracownik_filii
recenzja
rezerwacja
towary
urlop
wydawnictwo
wypozyczenie
zamowienie

Rysunek 56 Widok na tabele w bazie danych

System biblioteczny

d poprawności danych w bazie.

mysql> select * from pensja;				
idPensja	kwota_podstawowa	data_wyplaty	premia	
1	3000	2025-03-07	500	
2	3200	2025-02-28	450	
3	2800	2025-01-31	400	
4	3500	2025-03-05	600	
5	2900	2025-02-15	300	
6	3100	2025-03-01	550	
7	3300	2025-02-20	650	
8	3400	2025-03-10	700	
9	3000	2025-01-15	350	
10	3250	2025-02-05	500	

10 rows in set (0.00 sec)

Rysunek 57 Zapytanie typu SELECT w MySQL Server