



POLITECHNIKA LUBELSKA
WYDZIAŁ ELEKTROTECHNIKI I
INFORMATYKI

KIERUNEK STUDIÓW
INFORMATYKA

Przedmiot: Zarządzanie bazami SQL i NoSQL

Raport z wykonania projektu pt.

System biblioteczny - MongoDB

Autorzy:

Wiktoria Matacz

Agnieszka Marzęda

Ewelina Musińska

Lublin, 2025

Spis treści

Opis bazy danych.....	3
Motywacje wykonania bazy danych	3
Kolekcje w systemie bibliotecznym – opisy	4
Kolekcja: Czytelnik	5
Kolekcja: Pozycja.....	6
Kolekcja: Recenzja	8
Kolekcja: Wypożyczenie	9
Wypełnianie bazy danymi	10
Kolekcja Czytelnik	10
Kolekcja Pracownik	10
Kolekcja Pozycja.....	11
Kolekcja Wypożyczenie	11
Kolekcja Recenzja	12
Prezentacja zapytań pobierających dane z bazy	12
Prezentacja zapytań edytujących i usuwających dane	31
Edytujące.....	31
Usuwanie.....	39
Użytkownicy	42
Eksport, import i kopia zapasowa	46

Opis bazy danych

Baza danych Biblioteka została zaprojektowana z myślą o kompleksowym zarządzaniu funkcjonowaniem biblioteki, uwzględniając obsługę czytelników, pracowników, wypożyczeń, dostępnych pozycji oraz recenzji. Wykorzystuje model dokumentowy MongoDB, który pozwala przechowywać dane w postaci dokumentów JSON w ramach odpowiednich kolekcji.

Kolekcja Czytelnik przechowuje dane osób korzystających z usług biblioteki, takie jak imię i nazwisko. Czytelnicy mogą wypożyczać pozycje dostępne w zbiorach oraz dodawać recenzje przeczytanych książek. Z kolei kolekcja Pracownik zawiera informacje o zatrudnionych osobach, takie jak imię, nazwisko, stanowisko, adres e-mail oraz data zatrudnienia. Pracownicy odpowiadają za obsługę wypożyczeń.

Kolekcja Pozycja opisuje wszystkie materiały dostępne w bibliotece, w tym książki, audiobooki lub e-booki. Dane te obejmują tytuł, autora, gatunek, rok wydania oraz liczbę dostępnych egzemplarzy. Każda pozycja może być wypożyczana i recenzowana przez czytelników. Informacje o wypożyczeniach przechowywane są w kolekcji Wypożyczenie, która zawiera odniesienia do czytelnika, pozycji, daty wypożyczenia i zwrotu oraz aktualny status.

Opinie czytelników na temat konkretnych pozycji znajdują się w kolekcji Recenzja. Każda recenzja zawiera ocenę, komentarz oraz odniesienia do czytelnika i recenzowanej pozycji. Relacje między kolekcjami realizowane są za pomocą identyfikatorów, co umożliwia efektywne powiązanie danych i łatwe śledzenie interakcji użytkowników z zasobami biblioteki. Cała struktura bazy wspiera funkcjonowanie biblioteki oraz zapewnia możliwość łatwego wyszukiwania, analizy i zarządzania danymi.

Motywacje wykonania bazy danych

Główną motywacją stworzenia tej bazy danych jest potrzeba zautomatyzowania oraz usprawnienia zarządzania biblioteką w sposób nowoczesny, wydajny i przyjazny dla użytkownika. Tradycyjne systemy oparte na dokumentach papierowych lub prostych arkuszach kalkulacyjnych nie są skalowalne ani wystarczająco funkcjonalne w przypadku dużej liczby pozycji i użytkowników.

Dzięki zaprojektowanej bazie danych możliwe jest:

- szybkie wyszukiwanie pozycji i informacji o ich dostępności,
- pełna kontrola nad historią wypożyczeń i opóźnieniami,
- zarządzanie pracownikami,
- zapewnienie lepszej obsługi użytkowników i zwiększenie ich satysfakcji z korzystania z biblioteki.

Kolekcje w systemie bibliotecznym – opisy

System biblioteczny zbudowany w oparciu o kolekcje: czytelnik, pozycja, pracownik, recenzja oraz wypożyczenie to kompleksowe rozwiązanie do zarządzania biblioteką. Umożliwia przechowywanie danych o czytelnikach, ich rezerwacjach, wypożyczeniach i recenzjach, co pozwala na pełny wgląd w aktywność użytkowników. Kolekcja pracowników wspiera organizację pracy personelu biblioteki, umożliwiając zarządzanie ich zadaniami, urlopami i zamówieniami książek. Dane o pozycjach bibliotecznych, takie jak książki i e-booki, są szczegółowo opisane i zorganizowane, co usprawnia dostępność oraz ich wyszukiwanie. System integruje informacje o statusach wypożyczeń, dostępności egzemplarzy oraz opiniach czytelników, co wspiera zarządzanie zasobami oraz poprawia doświadczenie użytkowników. Dzięki tej strukturze system biblioteczny jest nowoczesnym narzędziem wspomagającym zarówno użytkowników, jak i bibliotekarzy, zapewniając przejrzystość, efektywność oraz łatwość obsługi.



Rysunek 1. Struktura bazy Biblioteka - podział na kolekcje

Kolekcja: Czytelnik

Kolekcja przechowuje dane o osobach korzystających z systemu bibliotecznego, znajdują się tam informacje takie jak: dane osobowe, dane adresowe, dane logowania oraz informacje związane z rezerwacjami i wypożyczeniami książek.

```
{
  "_id": { "$oid": "unikalne_czytelnik_id" },
  "imie": "string",
  "nazwisko": "string",
  "pesel": "string",
  "data_ur": "YYYY-MM-DD",
  "plec": "string",
  "miejscowosc": "string",
  "ulica": "string",
  "nr_domu": "string",
  "nr_mieszkania": "string",
  "dane_logowania": {
    "login": "string",
    "haslo": "zahaszowane haslo"
  },
  "recenzja_id": { "$oid": "unikalne_recenzja_id" },
  "idWypozyczenie": { "$oid": "unikalne_idwypozyczenie_id" },
  "rezerwacja": {
    "nr_w_kolejce": number_or_null,
    "status": "string_or_null",
    "data_anulowania": "YYYY-MM-DD_or_null",
    "data_utworzenia": "YYYY-MM-DD_or_null",
    "data_realizacji": "YYYY-MM-DD_or_null",
    "idCzytelnik": { "$oid": "czytelnik_id" },
    "idPozycja": { "$oid": "item_id_or_null" }
  }
}
```

Rysunek 2. Struktura kolekcji Czytelnik

_id – unikalny identyfikator dla czytelnika

imie - imię czytelnika

nazwisko – nazwisko czytelnika

pesel- unikalny numer identyfikacyjny czytelnika

data_ur – data urodzenia czytelnika (format YYYY-MM-DD)

plec- płeć czytelnika ("K"-kobieta, "M"- mężczyzna)

miejscowosc- miejsce zamieszkania

ulica-nazwa ulicy zamieszkania

nr_domu- numer domu zamieszkania

nr_mieszkania-numer mieszkania (opcjonalnie, jeśli brak -można ustawić jako null)

dane_logowania:

login- login używany do logowania do systemu

haslo- zaszyfrowane hasło do konta czytelnika

recenzja_id – identyfikator recenzji napisanej przez czytelnika (referencja do kolekcji recenzja)

idWypozyczenie – Identyfikator ostatniego wypożyczenia czytelnika (referencja do kolekcji wypożyczenie)

Kolekcja: Pozycja

Kolekcja przechowuje informacje o wszystkich zasobach dostępnych w bibliotece, takich jak: książki, e-booki oraz audiobooki. Pozwala zarządzać fizycznymi egzemplarzami jak i cyfrowymi, uwzględniając ich dostępność, lokalizację oraz szczegóły bibliograficzne.

```
{
  "_id": { "$oid": "ObjectId" },
  "tytul": "String",
  "autor": "String",
  "opis": "String",
  "rok_wydania": "Date",
  "jezyk": "String",
  "typ": "String",
  "isbn": "String",
  "wydawnictwo": "String",
  "kategorie": ["String"],
  "egzemplarze": [
    {
      "_id": { "$oid": "ObjectId" },
      "numer_inwentarzowy": "String",
      "status": "String",
      "lokalizacja": {
        "filia": "String",
        "regal": "String",
        "polka": "String"
      },
      "data_dodania": "Date"
    }
  ]
}
```

Rysunek 3. Struktura kolekcji Pozycja

_id – unikalny identyfikator pozycji

tytul- tytuł pozycji (np. książki) używany do identyfikacji i wyszukiwania

autor- imię i nazwisko autora danej pozycji

opis- krótki opis pozycji, umożliwiający zapoznanie się z jej zawartością

rok_wydania- data wydania pozycji, przydatna do określenia jej aktualności

jezyk- język, w jakim została napisana pozycja

typ- rodzaj pozycji np. Książka, ebook, audiobook

isbn- międzynarodowy numer książki, umożliwiający jednoznaczną identyfikację publikacji

wydawnictwo- nazwa wydawnictwa, które wydało daną pozycję

kategorie-lista kategorii lub gatunków, do których należy pozycja

egzemplarze:

_id- unikalny identyfikator egzemplarza

numer_inwentarzowy- unikalny numer przypisany do egzemplarza w systemie bibliotecznym

status- aktualny status egzemplarza np. dostępny lub wypożyczony

lokalizacja:

filia- nazwa filii

regal-numer regału

polka-numer półki

data_dodania: data dodania egzemplarza do systemu bibliotecznego

Kolekcja: Pracownik

Kolekcja zawiera dane dotyczące pracowników bibliotek, w tym ich dane personalne, przypisanie do filii, historię zamówień, urlopów, a także szczegóły dotyczące zatrudnienia. Pozwala na efektywne zarządzanie personelem w strukturze bibliotecznej.

```
{
  "_id": { "$oid": "ObjectId" },
  "imie": "String",
  "nazwisko": "String",
  "filia": {
    "nazwa": "String",
    "ulica": "String",
    "nr_domu": "String",
    "miasto": "String",
    "kod_pocztowy": "String",
    "telefon": "String"
  },
  "zamowienia": [
    {
      "nazwa": "String",
      "data_zamowienia": "Date",
      "dostawcy": ["String"],
      "wartosc": "Number",
      "towary": [
        {
          "nazwa": "String",
          "ilosc": "Number",
          "cena": "Number"
        }
      ]
    }
  ],
  "urlopy": [
    {
      "data_poczkowa": "Date",
      "data_koncowa": "Date",
      "typ_urlopu": "String"
    }
  ],
  "stanowisko": "String",
  "data_zatrudnienia": "Date",
  "status": "String",
  "kontakt": {
    "telefon": "String",
    "email": "String"
  }
}
```

Rysunek 4. Struktura kolekcji Pracownik

_id- unikalny identyfikator pracownika w systemie

imie- imię pracownika

nazwisko- nazwisko pracownika

filia:

nazwa- nazwa filii

ulica- nazwa ulicy, na której znajduje się filia

nr_domu- numer budynku filii

miasto- miasto w którym znajduje się filia

kod_pocztowy- kod pocztowy filii

telefon- numer telefonu do filii

zamowienia:

nazwa- nazwa zamówienia

data_zamowienia- data złożenia zamówienia

dostawcy- lista dostawców zamówienia

wartosc- łączna wartość zamówienia

towary:

nazwa- nazwa towaru

ilosc – liczba zamówionych egzemplarzy

cena – cena jednostkowa towaru

urlopy:

data_początkowa- data rozpoczęcia urlopu

data_koncowa – data zakończenia urlopu

typ_urlopu – rodzaj urlopu (wypoczynkowy, na zadanie itd.)

stanowisko- stanowisko zajmowane przez pracownika (bibliotekarz kierownik itd.)

data_zatrudnienia-data rozpoczęcia pracy przez pracownika

status – status pracownika (aktywny, nieaktywny)

kontakt:

telefon-numer telefonu pracownika

email- adres email pracownika

Kolekcja: Recenzja

Kolekcja przechowuje opinie czytelników na temat książek, e-booków lub audiobooków. Umożliwia to dzielenie się wrażeniami i ocenami. Każda recenzja jest powiązana z konkretną pozycją biblioteczną oraz czytelnikiem, który ją dodał.

```
{
  "_id": { "$oid": "" },
  "ocena": {"numer"},
  "komentarz": {"string"},
  "idPozycja": { "$oid": "" },
  "idCzytelnik": { "$oid": "" }
}
```

Rysunek 5. Struktura kolekcji Recenzja

_id- unikalny identyfikator recenzji

ocena- ocena wystawiona przez czytelnika dla pozycji np. książki, gdzie wyższa ocena oznacza wyższą ocenę

komentarz- tekstowy komentarz czytelnika opisujący jego wrażenia z danej pozycji

idPozycja- referencja do identyfikatora pozycji w kolekcji Pozycja, której dotyczy recenzja

idCzytelnika- referencja do identyfikatora czytelnika w kolekcji Czytelnik, który dodał recenzję

Kolekcja: Wypożyczenie

Kolekcja przechowuje informacje o procesach wypożyczenia książek przez czytelników w bibliotece. Rejestruje wszystkie szczegóły związane z wypożyczeniami, takie jak terminy zwrotów, statusu oraz ewentualne kary.

```
{
  "_id": { "$oid": "" },
  "czytelnik_id": { "$oid": "" },
  "egzemplarz_id": { "$oid": "" },
  "data_wypozyczenia": "",
  "data_zwrotu": "",
  "status": "",
  "po_terminie": ,
  "kary": [
    {
      "_id": { "$oid": "" },
      "kwota": ,
      "data_nalozenia": "",
      "status": "",
      "uwagi": ""
    }
  ]
}
```

Rysunek 6. Struktura kolekcji Wypożyczenie

_id- unikalny identyfikator wypożyczenia

czytelnik_id- referencja do identyfikatora czytelnika wypożyczającego egzemplarz

egzemplarz_id- referencja do identyfikatora egzemplarza wypożyczonego przez czytelnika

data_wypozyczenia-data wypożyczenia książki

data_zwrotu- data zwrotu książki

status- status wypożyczenia (oddane, wypożyczone)

po_terminie- informacja czy zwrot był po terminie

kary:

_id- unikalny identyfikator kary

kwota- kwota nałożonej kary

data_nalozenia-data nałożenia kary

status- status kary np opłacona

uwagi- dodatkowe informacje

Wypełnianie bazy danymi

Kolekcja Czytelnik

admin > Biblioteka > Czytelnik

Documents 10 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) ⚡ Explain

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#) 25 1 – 10 of 10

```
_id: ObjectId('607f1f77bcf86cd799431001')
imie: "Anna"
nazwisko: "Nowak"
pesel: "90010112345"
data_ur: "1990-01-01"
plec: "K"
miejscowosc: "Lublin"
ulica: "Krańcowa"
nr_domu: "15A"
nr_mieszkania: "7"
▶ dane_logowania: Object
  recenzja_id: ObjectId('607f1f77bcf86cd799432001')
  idWypozyczenie: ObjectId('507f1f77bcf86cd799439011')
▶ rezerwacja: Object
▶ adresy_alternatywne: Array (1)
▶ kontakt: Object
```

```
_id: ObjectId('607f1f77bcf86cd799431002')
```

Rysunek 7. Kolekcja Czytelnik

Kolekcja Pracownik

admin > Biblioteka > Pracownik

Documents 10 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) ⚡ Explain

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#) 25 1 – 10 of 10

```
_id: ObjectId('644b10b56f62a6dbec999999')
imie: "Anna"
nazwisko: "Nowak"
▶ filia: Object
▶ zamowienia: Array (1)
▶ urlopy: Array (4)
  stanowisko: "Bibliotekarz"
  data_zatrudnienia: "2022-01-15"
  status: "Aktywny"
▶ kontakt: Object
```

```
_id: ObjectId('644b10b56f62a6dbec999998')
imie: "Jan"
nazwisko: "Kowalski"
▶ filia: Object
▶ zamowienia: Array (1)
▶ urlopy: Array (1)
  stanowisko: "Kierownik"
  data_zatrudnienia: "2020-06-01"
  status: "Aktywny"
▶ kontakt: Object
```

Rysunek 8. Kolekcja Pracownik

Kolekcja Pozycja

admin > Biblioteka > Pozycja

Documents 9 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

Explain

ADD DATA

EXPORT DATA

UPDATE

DELETE

25

1 - 9 of 9

```
_id: ObjectId('607f1f77bcf86cd799436001')
tytuł : "Lalka"
autor : "Bolesław Prus"
opis : "Powieść społeczno-obyczajowa z okresu pozytywizmu."
rok_wydania : "1890-01-01"
jezyk : "polski"
typ : "ksiazka"
isbn : "9788373271280"
wydawnictwo : "Wydawnictwo Literackie"
kategorie : Array (2)
egzemplarze : Array (3)
```

```
_id: ObjectId('607f1f77bcf86cd799436002')
tytuł : "Lalka"
autor : "Bolesław Prus"
opis : "Powieść społeczno-obyczajowa z okresu pozytywizmu."
rok_wydania : "1890-01-01"
jezyk : "polski"
typ : "ebook"
```

Rysunek 9. Kolekcja Pozycja

Kolekcja Wypożyczenie

admin > Biblioteka > Wypożyczenie

Documents 10 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

Explain

ADD DATA

EXPORT DATA

UPDATE

DELETE

25

1 - 10 of 10

```
_id: ObjectId('507f1f77bcf86cd799439011')
czytelnik_id : ObjectId('607f1f77bcf86cd799431001')
egzemplarz_id : ObjectId('707f1f77bcf86cd799432001')
data_wypozyczenia : "2024-01-15"
data_zwrotu : "2024-01-30"
status : "oddane"
po_terminie : false
kary : Array (empty)
```

```
_id: ObjectId('507f1f77bcf86cd799439012')
czytelnik_id : ObjectId('607f1f77bcf86cd799431002')
egzemplarz_id : ObjectId('707f1f77bcf86cd799432002')
data_wypozyczenia : "2024-02-05"
data_zwrotu : "2024-02-20"
status : "oddane"
po_terminie : true
kary : Array (1)
```

Rysunek 10. Kolekcja Wypożyczenie

Kolekcja Recenzja

admin > Biblioteka > Recenzja

Documents 11 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

Explain

ADD DATA

EXPORT DATA

UPDATE

DELETE

25

1 - 11 of 11

```
{
  "_id": ObjectId('607f1f77bcf86cd799434001'),
  "ocena": 5,
  "komentarz": "Świetna książka! Polecam każdemu miłośnikowi gatunku.",
  "idPozycja": ObjectId('607f1f77bcf86cd799436001'),
  "idCzytelnik": ObjectId('607f1f77bcf86cd799431001')
}
```

```
{
  "_id": ObjectId('607f1f77bcf86cd799434002'),
  "ocena": 4,
  "komentarz": "Dobra lektura, choć końcówka mogła być lepiej rozwinięta.",
  "idPozycja": ObjectId('607f1f77bcf86cd799436002'),
  "idCzytelnik": ObjectId('607f1f77bcf86cd799431002')
}
```

```
{
  "_id": ObjectId('607f1f77bcf86cd799434003'),
  "ocena": 3,
  "komentarz": "Średnia książka, niektóre wątki były naciągane.",
  "idPozycja": ObjectId('607f1f77bcf86cd799436003'),
  "idCzytelnik": ObjectId('607f1f77bcf86cd799431003')
}
```

Rysunek 11. Kolekcja Recenzja

Prezentacja zapytań pobierających dane z bazy

1. Pobranie wszystkich książek z dostępnych egzemplarzy w konkretnej filii.

```
> db.Pozycja.aggregate([
  { $unwind: "$egzemplarze" },
  { $match: { "egzemplarze.lokalizacja.filia": "Filia nr 2", "egzemplarze.status": "dostępny" } },
  {
    $project: {
      tytuł: 1,
      opis: 1,
      "egzemplarze.numer_inwentarzowy": 1,
      "egzemplarze.lokalizacja": 1
    }
  }
]);
< {
  "_id": ObjectId('607f1f77bcf86cd799436003'),
  "tytuł": 'Zbrodnia i kara',
  "opis": 'Klasyczna powieść psychologiczna rosyjskiego pisarza.',
  "egzemplarze": {
    "numer_inwentarzowy": 'INV-201',
    "lokalizacja": {
      "filia": 'Filia nr 2',
      "regal": 'R-4',
      "półka": 'P-2'
    }
  }
}
```

Rysunek 12. Zapytanie pobierające dostępne egzemplarze

2. Pobranie listy wszystkich książek z ich egzemplarzami, recenzjami i liczbą wypożyczeń.

```
> db.Pozycja.aggregate([
  {
    $lookup: {
      from: "Recenzja",
      localField: "_id",
      foreignField: "idPozycja",
      as: "recenzje"
    }
  },
  {
    $lookup: {
      from: "wypozyczenie",
      localField: "egzemplarze._id",
      foreignField: "egzemplarz_id",
      as: "wypozyczenia"
    }
  },
  {
    $addFields: {
      liczba_wypozyczen: { $size: "$wypozyczenia" }
    }
  },
  {
    $project: {
      tytul: 1,
      autor: 1,
      rok_wydania: 1,
      jezyk: 1,
      egzemplarze: 1,
      recenzje: { ocena: 1, komentarz: 1 },
      liczba_wypozyczen: 1
    }
  }
]);
```

Rysunek 13. Zapytanie pobierające listę książek

```

    }
  ],
  liczba_wypozycczen: 0
}
{
  _id: ObjectId('607f1f77bcf86cd799436009'),
  tytul: 'Solaris',
  autor: 'Stanisław Lem',
  rok_wydania: '1961-01-01',
  jezyk: 'polski',
  recenzje: [
    {
      ocena: 3,
      komentarz: 'Może być. Nie zachwyca, ale też nie rozczarowuje.'
    }
  ],
  liczba_wypozycczen: 0
}
{
  _id: ObjectId('607f1f77bcf86cd799436010'),
  tytul: 'Czarne oceany',
  autor: 'Jacek Dukaj',
  rok_wydania: '2001-01-01',
  jezyk: 'polski',
  recenzje: [
    {
      ocena: 4,
      komentarz: 'Dobry styl autora, wciągająca historia.'
    }
  ],
  liczba_wypozycczen: 0
}

```

Rysunek 14. Efekt zapytania pobierającego listę książek

3. Pobranie listy czytelników, którzy mają aktualne wypożyczenia lub zaległości finansowe.

```
> db.Czytelnik.aggregate([
  {
    $lookup: {
      from: "Wypozyczenie",
      localField: "_id",
      foreignField: "czytelnik_id",
      as: "wypozyczenia"
    }
  },
  {
    $match: {
      $or: [
        { "wypozyczenia.status": { $ne: "oddane" } },
        { "wypozyczenia.kary.status": "nieopłacona" }
      ]
    }
  },
  {
    $project: {
      imie: 1,
      nazwisko: 1,
      miejscowość: 1,
      "wypozyczenia.data_wypozyczenia": 1,
      "wypozyczenia.data_zwrotu": 1,
      "wypozyczenia.kary": 1
    }
  }
]);
```

Rysunek 15. Pobranie listy czytelników

```

{
  _id: ObjectId('607f1f77bcf86cd799431005'),
  imie: 'Katarzyna',
  nazwisko: 'Lewandowska',
  'miejscowość': 'Poznań',
  wypozyczenia: [
    {
      data_wypozyczenia: '2024-04-05',
      data_zwrotu: '2024-04-18',
      kary: [
        {
          _id: ObjectId('807f1f77bcf86cd799433002'),
          kwota: 5,
          data_nalozenia: '2024-04-20',
          status: 'nieopłacona',
          uwagi: 'Lekkie uszkodzenie okładki'
        },
        {
          _id: ObjectId('807f1f77bcf86cd799433003'),
          kwota: 5,
          data_nalozenia: '2024-04-27',
          status: 'nieopłacona',
          uwagi: 'Dodatkowa kara za nieterminowe uregulowanie'
        }
      ]
    }
  ]
}
]
}
{
  _id: ObjectId('607f1f77bcf86cd799431008'),
  imie: 'Tomasz',
  nazwisko: 'Kamiński',
  'miejscowość': 'Szczecin',
  wypozyczenia: [
    {

```

Rysunek 16. Efekt zapytania o pobranie listy czytelników

4. Lista pracowników z historią zamówień i szczegółami zamówionych towarów.

```
db.Pracownik.aggregate([
  { $unwind: "$zamowienia" },
  { $unwind: "$zamowienia.towary" },
  {
    $group: {
      _id: "$_id",
      imie: { $first: "$imie" },
      nazwisko: { $first: "$nazwisko" },
      zamowienia: {
        $push: {
          nazwa: "$zamowienia.nazwa",
          data_zamowienia: "$zamowienia.data_zamowienia",
          towar: "$zamowienia.towary.nazwa",
          ilosc: "$zamowienia.towary.ilosc",
          wartosc: { $multiply: ["$zamowienia.towary.cena", "$zamowienia.towary.ilosc"] }
        }
      }
    }
  },
  { $project: { imie: 1, nazwisko: 1, zamowienia: 1 } }
]);
```

Rysunek 17. Lista pracowników

```

{
  _id: ObjectId('644b10b56f62a6dbec999992'),
  imie: 'Marek',
  nazwisko: 'Wójcik',
  zamówienia: [
    {
      nazwa: 'Audiobooki',
      data_zamowienia: '2025-01-10',
      towar: 'Kryminał',
      ilosc: 25,
      wartosc: 1250
    },
    {
      nazwa: 'Audiobooki',
      data_zamowienia: '2025-01-10',
      towar: 'Fantastyka',
      ilosc: 15,
      wartosc: 1050
    }
  ]
}
{
  _id: ObjectId('644b10b56f62a6dbec999991'),
  imie: 'Agnieszka',
  nazwisko: 'Kaczmarek',
  zamówienia: [
    {
      nazwa: 'Czasopisma',
      data_zamowienia: '2025-02-28',
      towar: 'National Geographic',
      ilosc: 12,
      wartosc: 300
    },
    {
      nazwa: 'Czasopisma',

```

Rysunek 18. Efekt zapytania o listę pracowników

5. Pobranie statystyk dotyczących recenzji (średnia ocena, liczba recenzji) dla każdej książki.

```
db.Pozycja.aggregate([
  {
    $lookup: {
      from: "Recenzja",
      localField: "_id",
      foreignField: "idPozycja",
      as: "recenzje"
    }
  },
  {
    $addFields: {
      srednia_ocena: { $avg: "$recenzje.ocena" },
      liczba_recenzji: { $size: "$recenzje" }
    }
  },
  {
    $project: {
      tytul: 1,
      autor: 1,
      typ: 1,
      srednia_ocena: 1,
      liczba_recenzji: 1
    }
  }
]);
```

Rysunek 19. Pobranie statystyk

```

{
  _id: ObjectId('607f1f77bcf86cd799436001'),
  tytuł: 'Lalka',
  autor: 'Bolesław Prus',
  typ: 'ksiazka',
  srednia_ocena: 5,
  liczba_recenzji: 1
}
{
  _id: ObjectId('607f1f77bcf86cd799436002'),
  tytuł: 'Lalka',
  autor: 'Bolesław Prus',
  typ: 'ebook',
  srednia_ocena: 4,
  liczba_recenzji: 1
}
{
  _id: ObjectId('607f1f77bcf86cd799436003'),
  tytuł: 'Zbrodnia i kara',
  autor: 'Fiodor Dostojewski',
  typ: 'ksiazka',
  srednia_ocena: 3,
  liczba_recenzji: 1
}
{
  _id: ObjectId('607f1f77bcf86cd799436004'),
  tytuł: 'Zbrodnia i kara',
  autor: 'Fiodor Dostojewski',
  typ: 'audiobook',
  srednia_ocena: 5,
  liczba_recenzji: 1
}

```

Rysunek 20. Efekt zapytania o statystyki

6. Suma kar na każdego czytelnika - Zwraca ID czytelnika, łączną kwotę kar i liczbę kar – posortowane malejąco wg sumy.

```
db.Wypozyczenie.aggregate([
  { $unwind: "$kary" },
  {
    $group: {
      _id: "$czytelnik_id",
      suma_kar: { $sum: "$kary.kwota" },
      liczba_kar: { $sum: 1 }
    }
  },
  {
    $lookup: {
      from: "Czytelnik",
      localField: "_id",
      foreignField: "_id",
      as: "czytelnik"
    }
  },
  { $unwind: "$czytelnik" },
  {
    $project: {
      imie: "$czytelnik.imie",
      nazwisko: "$czytelnik.nazwisko",
      suma_kar: 1,
      liczba_kar: 1
    }
  },
  { $sort: { suma_kar: -1 } }
]);
```

Rysunek 21. Zapytanie o sumę kar

```
{
  _id: ObjectId('607f1f77bcf86cd799431010'),
  suma_kar: 24,
  liczba_kar: 2,
  imie: 'Marcin',
  nazwisko: 'Zajac'
}
{
  _id: ObjectId('607f1f77bcf86cd799431007'),
  suma_kar: 20,
  liczba_kar: 1,
  imie: 'Magdalena',
  nazwisko: 'Wojcik'
}
{
  _id: ObjectId('607f1f77bcf86cd799431002'),
  suma_kar: 10.5,
  liczba_kar: 1,
  imie: 'Piotr',
  nazwisko: 'Nowak'
}
{
  _id: ObjectId('607f1f77bcf86cd799431005'),
  suma_kar: 10,
  liczba_kar: 2,
  imie: 'Katarzyna',
  nazwisko: 'Lewandowska'
}
```

Rysunek 22. Efekt zapytania o sumę kar

7. Łączna wartość zamówień na pracownika - Zgrupowane wg pracownika, pokazuje łączną wartość zamówień i ich liczbę.

```
db.Pracownik.aggregate([
  { $unwind: "$zamowienia" },
  {
    $group: {
      _id: "$_id",
      imie: { $first: "$imie" },
      nazwisko: { $first: "$nazwisko" },
      laczna_wartosc: { $sum: "$zamowienia.wartosc" },
      liczba_zamowien: { $sum: 1 }
    }
  },
  { $sort: { laczna_wartosc: -1 } }
]);
```

Rysunek 23. Łączna wartość zamówień pracownika

```
{
  _id: ObjectId('644b10b56f62a6dbee999996'),
  imie: 'Piotr',
  nazwisko: 'Zieliński',
  laczna_wartosc: 5000,
  liczba_zamowien: 2
}
{
  _id: ObjectId('644b10b56f62a6dbee999992'),
  imie: 'Marek',
  nazwisko: 'Wójcik',
  laczna_wartosc: 1800,
  liczba_zamowien: 1
}
{
  _id: ObjectId('644b10b56f62a6dbee999997'),
  imie: 'Ewa',
  nazwisko: 'Wiśniewska',
  laczna_wartosc: 1500,
  liczba_zamowien: 1
}
```

Rysunek 24. Efekt zapytania o łączną wartość zamówień

8. Średni czas wypożyczeń - Oblicza średni czas wypożyczeń w dniach.

```
db.Wypozyczenie.aggregate([
  { $match: { status: "oddane" } },
  { $project: {
    czytelnik_id: "$czytelnik_id",
    dni: {
      $divide: [
        { $subtract: [
          { $toDate: "$data_zwrotu" },
          { $toDate: "$data_wypozyczenia" }
        ]
        }, 1000 * 60 * 60 * 24
      ]
    }
  }
},
{ $group: {
  _id: "$czytelnik_id",
  srednia_dni: { $avg: "$dni" }
}
},
{ $lookup: {
  from: "Czytelnik",
  localField: "_id",
  foreignField: "_id",
  as: "czytelnik"
}
},
{ $unwind: "$czytelnik" },
{ $project: {
  imie: "$czytelnik.imie",
  nazwisko: "$czytelnik.nazwisko",
  srednia_dni: 1
}
},
{ $sort: { srednia_dni: -1 } }
]);
```

Rysunek 25. Średni czas wypożyczeń


```
{
  _id: ObjectId('607f1f77bcf86cd799431001'),
  srednia_dni: 15,
  imie: 'Anna',
  nazwisko: 'Kowalska'
}
{
  _id: ObjectId('607f1f77bcf86cd799431007'),
  srednia_dni: 15,
  imie: 'Magdalena',
  nazwisko: 'Wójcik'
}
{
  _id: ObjectId('607f1f77bcf86cd799431002'),
  srednia_dni: 15,
  imie: 'Piotr',
  nazwisko: 'Nowak'
}
{
  _id: ObjectId('607f1f77bcf86cd799431009'),
  srednia_dni: 14,
  imie: 'Agnieszka',
  nazwisko: 'Kaczmarek'
}
{
  _id: ObjectId('607f1f77bcf86cd799431003'),
  srednia_dni: 14,
  imie: 'Maria',
  nazwisko: 'Wiśniewska'
}
```

Rysunek 26. Efekt zapytania o średni czas wypożyczeń

9. TOP 3 pozycje z najwyższą średnią oceną - Zwraca ID i tytuł pozycji z najwyższą średnią oceną i liczbę ocen.

```
db.Recenzja.aggregate([
  {
    $group: {
      _id: "$idPozycja",
      srednia_ocena: { $avg: "$ocena" },
      liczba_ocen: { $sum: 1 }
    }
  },
  {
    $lookup: {
      from: "Pozycja",
      localField: "_id",
      foreignField: "_id",
      as: "pozycja"
    }
  },
  { $unwind: "$pozycja" },
  {
    $project: {
      tytul: "$pozycja.tytul",
      srednia_ocena: 1,
      liczba_ocen: 1
    }
  },
  { $sort: { srednia_ocena: -1 } },
  { $limit: 3 }
]);
```

Rysunek 27. TOP 3 pozycje

```
{
  _id: ObjectId('607f1f77bcf86cd799436004'),
  srednia_ocena: 5,
  liczba_ocen: 1,
  tytul: 'Zbrodnia i kara'
}
{
  _id: ObjectId('607f1f77bcf86cd799436001'),
  srednia_ocena: 5,
  liczba_ocen: 1,
  tytul: 'Lalka'
}
{
  _id: ObjectId('607f1f77bcf86cd799436007'),
  srednia_ocena: 5,
  liczba_ocen: 1,
  tytul: 'Solaris'
}
```

Rysunek 28. Efekt zapytania o TOP 3 pozycje

10. Liczba egzemplarzy każdej książki - filtruje tylko książki, zlicza ich egzemplarze (jeśli istnieją), i sortuje malejąco wg liczby.

```
db.Pozycja.aggregate([
  {
    $match: { typ: "ksiazka" }
  },
  {
    $project: {
      tytul: 1,
      typ: 1,
      liczba_egzemplarzy: {
        $cond: {
          if: { $isArray: "$egzemplarze" },
          then: { $size: "$egzemplarze" },
          else: 0
        }
      }
    }
  },
  { $sort: { liczba_egzemplarzy: -1 } }
]);
```

Rysunek 29. Liczba egzemplarzy

```
{
  _id: ObjectId('607f1f77bcf86cd799436001'),
  tytul: 'Lalka',
  typ: 'ksiazka',
  liczba_egzemplarzy: 2
}
{
  _id: ObjectId('607f1f77bcf86cd799436005'),
  tytul: 'Wiedźmin: Ostatnie życzenie',
  typ: 'ksiazka',
  liczba_egzemplarzy: 2
}
{
  _id: ObjectId('607f1f77bcf86cd799436003'),
  tytul: 'Zbrodnia i kara',
  typ: 'ksiazka',
  liczba_egzemplarzy: 1
}
{
  _id: ObjectId('607f1f77bcf86cd799436007'),
  tytul: 'Solaris',
  typ: 'ksiazka',
  liczba_egzemplarzy: 1
}
```

Rysunek 30. Efekt zapytania o liczbę egzemplarzy

11. Lista książek danego autora z liczbą egzemplarzy w systemie.

```
db.Pozycja.aggregate([
  { $match: { autor: "Andrzej Sapkowski", typ: "ksiazka" } },
  {
    $project: {
      tytul: 1,
      liczba_egzemplarzy: {
        $cond: {
          if: { $isArray: "$egzemplarze" },
          then: { $size: "$egzemplarze" },
          else: 0
        }
      }
    }
  }
]);
{
  _id: ObjectId('607f1f77bcf86cd799436005'),
  tytul: 'Wiedźmin: Ostatnie życzenie',
  liczba_egzemplarzy: 2
}
```

Rysunek 31. Liczba książek danego autora

12. Czytelnicy, którzy anulowali rezerwację – z datą anulowania i książką.

```
db.Czytelnik.aggregate([
  { $match: { "rezerwacja.status": "anulowana" } },
  {
    $lookup: {
      from: "Pozycja",
      localField: "rezerwacja.idPozycja",
      foreignField: "_id",
      as: "pozycja"
    }
  },
  { $unwind: { path: "$pozycja", preserveNullAndEmptyArrays: true } },
  {
    $project: {
      imie: 1,
      nazwisko: 1,
      data_anulowania: "$rezerwacja.data_anulowania",
      tytul: "$pozycja.tytul"
    }
  }
]);
{
  _id: ObjectId('607f1f77bcf86cd799431005'),
  imie: 'Katarzyna',
  nazwisko: 'Lewandowska',
  data_anulowania: '2024-01-20',
  tytul: 'Zbrodnia i kara'
}
```

Rysunek 32. Czytelnicy, którzy anulowali rezerwację

13. Lista czytelników, którzy wystawili oceny niższe niż 3.

```
db.Recenzja.aggregate([
  { $match: { ocena: { $lt: 3 } } },
  {
    $lookup: {
      from: "Czytelnik",
      localField: "idCzytelnik",
      foreignField: "_id",
      as: "czytelnik"
    }
  },
  { $unwind: "$czytelnik" },
  {
    $project: {
      imie: "$czytelnik.imie",
      nazwisko: "$czytelnik.nazwisko",
      ocena: 1,
      komentarz: 1
    }
  }
]);
```

Rysunek 33. Lista czytelników

```
{
  _id: ObjectId('607f1f77bcf86cd799434005'),
  ocena: 2,
  komentarz: 'Rozczarowanie. Fabuła przewidywalna, postaci płaskie.',
  imie: 'Katarzyna',
  nazwisko: 'Lewandowska'
}
{
  _id: ObjectId('607f1f77bcf86cd799434008'),
  ocena: 1,
  komentarz: 'Strasznie nudna. Nie mogłem doczytać do końca.',
  imie: 'Tomasz',
  nazwisko: 'Kamiński'
}
```

Rysunek 34. Efekt zapytania o listę czytelników

14. Lista e-booków i audiobooków z linkiem dostępu oraz typem.

```
db.Pozycja.find(
  { typ: { $in: ["ebook", "audiobook"] } },
  {
    _id: 0,
    tytul: 1,
    autor: 1,
    typ: 1,
    link_dostepu: 1
  }
);
{
  tytul: 'Lalka',
  autor: 'Bolesław Prus',
  typ: 'ebook',
  link_dostepu: 'https://biblioteka.pl/ebooki/lalka.epub'
}
{
  tytul: 'Zbrodnia i kara',
  autor: 'Fiodor Dostojewski',
  typ: 'audiobook',
  link_dostepu: 'https://biblioteka.pl/audiobooki/zbrodnia_i_kara.mp3'
}
```

Rysunek 35. Lista ebooków i audiobooków

15. Najczęściej zamawiane towary – zliczenie wystąpień każdego towaru.

```
db.Pracownik.aggregate([
  { $unwind: "$zamowienia" },
  { $unwind: "$zamowienia.towary" },
  {
    $group: {
      _id: "$zamowienia.towary.nazwa",
      laczna_ilosc: { $sum: "$zamowienia.towary.ilosc" }
    }
  },
  { $sort: { laczna_ilosc: -1 } }
]);
```

Rysunek 36. Najczęściej zamawiane towary

```

{
  _id: 'Kryminał',
  łączna_ilosc: 25
}
{
  _id: 'Powieść 1',
  łączna_ilosc: 20
}
{
  _id: 'Bajki',
  łączna_ilosc: 20
}
{
  _id: 'Fantastyka',
  łączna_ilosc: 15
}

```

Rysunek 37. Efekt zapytania o najczęściej zamawiane towary

Prezentacja zapytań edytujących i usuwających dane

Edytujące

1. Aktualizacja statusu wszystkich egzemplarzy danej pozycji książkowej. - Zapytanie zmienia status wszystkich egzemplarzy na "uszkodzony".

```

db.Pozycja.updateMany(
  { _id: ObjectId("607f1f77bcf86cd799436001") }, // ID książki "Lałka"
  { $set: { "egzemplarze.$.status": "uszkodzony" } }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

Rysunek 38. Aktualizacja statusu egzemplarzy

```

db.Pozycja.find(
  { _id: ObjectId("607f1f77bcf86cd799436001") });
{
  _id: ObjectId('607f1f77bcf86cd799436001'),
  tytuł: 'Lalka',
  autor: 'Bolesław Prus',
  opis: 'Powieść społeczno-obyczajowa z okresu po',
  rok_wydania: '1890-01-01',
  język: 'polski',
  typ: 'książka',
  isbn: '9788373271280',
  wydawnictwo: 'Wydawnictwo Literackie',
  kategorie: [
    'literatura piękna',
    'klasyka'
  ],
  egzemplarze: [
    {
      _id: ObjectId('707f1f77bcf86cd799437001'),
      numer_inwentarzowy: 'INV-101',
      status: 'uszkodzony',
      lokalizacja: {
        filia: 'Filia nr 1',
        regał: 'R-2',
        'półka': 'P-3'
      },
      data_dodania: '2023-02-15'
    },
  ],
}

```

Rysunek 39. Egzemplarz ze zmienionym statusem

2. Dodanie nowego egzemplarza do istniejącej książki. - Zapytanie dodaje nowy egzemplarz do pola egzemplarze o określonym id książki.

```
db.Pozycja.updateOne(
  { _id: ObjectId("607f1f77bcf86cd799436001") },
  {
    $push: {
      egzemplarze: {
        _id: ObjectId(),
        numer_inwentarzowy: "INV-103",
        status: "dostępny",
        lokalizacja: { filia: "Filia nr 4", regał: "R-6", półka: "P-3" },
        data_dodania: new Date("2023-09-15")
      }
    }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Rysunek 40. Dodanie nowego egzemplarza

```
db.Pozycja.find(
  { _id: ObjectId("607f1f77bcf86cd799436001")
});
{
  _id: ObjectId('607f1f77bcf86cd799436001'),
  tytuł: 'Lałka',
  autor: 'Bolesław Prus',

```

Rysunek 41. Nowy egzemplarz

```
{
  _id: ObjectId('6818afb70c5c7e63423b4e41'),
  numer_inwentarzowy: 'INV-103',
  status: 'dostępny',
  lokalizacja: {
    filia: 'Filia nr 4',
    regał: 'R-6',
    'półka': 'P-3'
  },
  data_dodania: 2023-09-15T00:00:00.000Z
}
```

Rysunek 42. Nowy egzemplarz

3. Zmiana danych czytelnika i dodanie nowego adresu. - Zapytanie aktualizuje dane czytelnika i dodaje alternatywny adres kontaktowy.

```
db.Czytelnik.updateOne(
  { _id: ObjectId("607f1f77bcf86cd799431001") }, // ID czytelnika
  {
    $set: { nazwisko: "Nowak", "kontakt.email": "anna.nowak@example.com" },
    $push: { adresy_alternatywne: { miasto: "Warszawa", ulica: "Marszałkowska", nr_domu: "12", kod_pocztowy: "00-001" } }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Rysunek 43. Zmiana danych

```
db.Czytelnik.find({
  _id: ObjectId("607f1f77bcf86cd799431001")
});
{
  _id: ObjectId('607f1f77bcf86cd799431001'),
  imie: 'Anna',
  nazwisko: 'Nowak',
  pesel: '90010112345',
  data_ur: '1990-01-01',
  plec: 'K',
  'miejscowość': 'Lublin',
  ulica: 'Krańcowa',
  nr_domu: '15A',
  nr_mieszkania: '7',
  dane_logowania: {
    login: 'akowalska',
    haslo: '$2a$10$N9qo8uL0ickgx2ZMRZoMy...'
  },
  recenzja_id: ObjectId('607f1f77bcf86cd799431001'),
  idWypozyczenie: ObjectId('507f1f77bcf86cd799431001'),
  rezerwacja: {
    nr_w_kolejce: null,
  }
}
```

Rysunek 44. Efekt zmiany danych

```

    status: null,
    data_anulowania: null,
    data_utworzenia: null,
    data_realizacji: null,
    idCzytelnik: ObjectId('607f1f77bcf86cd7994310'),
    idPozycja: null
  },
  adresy_alternatywne: [
    {
      miasto: 'Warszawa',
      ulica: 'Marszałkowska',
      nr_domu: '12',
      kod_pocztowy: '00-001'
    }
  ],
  kontakt: {
    email: 'anna.nowak@example.com'
  }
}

```

Rysunek 45. Efekt zmiany danych

4. Zwiększa o 10% wartość wszystkich zamówień pracowników z filii „Filia nr 2”.

```

db.Pracownik.updateMany(
  { "filia.nazwa": "Filia nr 2" },
  {
    $mul: { "zamowienia.$.wartosc": 1.1 }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

Rysunek 46. Zwiększenie wartości zamówień

```

db.Pracownik.find({
  "filia.nazwa": "Filia nr 2"
});
{
  _id: ObjectId('644b10b56f62a6dbee999999',
  imie: 'Anna',
  nazwisko: 'Nowak',
  filia: {
    nazwa: 'Filia nr 2',
    ulica: 'Szkołna',
    nr_domu: '10',
    miasto: 'Kraków',
    kod_pocztowy: '30-001',
    telefon: '987-654-321'
  },
  zamowienia: [
    {
      nazwa: 'Podręczniki matematyczne',
      data_zamowienia: '2025-04-01',
      dostawcy: [
        'Dostawca A',
        'Dostawca B'
      ],
      wartosc: 1100,

```

Rysunek 47. Efekt zwiększenia wartości zamówień

5. Zmieniono numer telefonu do kontaktu pracownika „Ewa Wiśniewska”.

```

db.Pracownik.updateOne(
  { imie: "Ewa", nazwisko: "Wiśniewska" },
  {
    $set: { "kontakt.telefon": "222-123-123" }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

Rysunek 48. Zmiana numeru telefonu

```

stanowisko: 'Asystent',
data_zatrudnienia: '2021-03-10',
status: 'Na urlopie',
kontakt: {
  telefon: '222-123-123',
  email: 'ewa.wisniewska@biblioteka.pl'
}

```

Rysunek 49. Efekt zmiany numeru telefonu

6. Dodano nowy wpis urlopu typu „Na żądanie” dla pracownika „Anna Nowak”.

```
db.Pracownik.updateOne(
  { imie: "Anna", nazwisko: "Nowak" },
  {
    $push: {
      urlopy: {
        data_początkowa: "2025-10-01",
        data_koncowa: "2025-10-03",
        typ_urlopu: "Na żądanie"
      }
    }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Rysunek 50. Nowy wpis o urlopie

```
{
  data_początkowa: '2025-10-01',
  data_koncowa: '2025-10-03',
  typ_urlopu: 'Na żądanie'
},
{
  stanowisko: 'Bibliotekarz',
  data_zatrudnienia: '2022-01-15',
  status: 'Aktywny',
  kontakt: {
    telefon: '123-456-789',
    email: 'anna.nowak@biblioteka.pl'
  }
}
```

Rysunek 51. Nowy wpis o urlopie

7. Dodanie nowej recenzji do książki z konkretnym ID oraz przypisanie do konkretnego czytelnika.

```
db.Recenzja.insertOne({
  ocena: 5,
  komentarz: "Wyjątkowa książka, warta każdej minuty!",
  idPozycja: ObjectId("607f1f77bcf86cd799436002"),
  idCzytelnik: ObjectId("607f1f77bcf86cd799431002")
});
{
  acknowledged: true,
  insertedId: ObjectId('6818b3eb0c5c7e63423b4e42')
}
```

Rysunek 52. Dodanie recenzji

```

db.Recenzja.find({
  _id: ObjectId('6818b3eb0c5c7e63423b4e42')
});
{
  _id: ObjectId('6818b3eb0c5c7e63423b4e42'),
  ocena: 5,
  komentarz: 'Wyjątkowa książka, warta każdej minuty!',
  idPozycja: ObjectId('607f1f77bcf86cd799436002'),
  idCzytelnik: ObjectId('607f1f77bcf86cd799431002')
}

```

Rysunek 53. Efekt dodania recenzji

8. Przeniesienie pracownika do innej filii (z Filii nr 1 do Filii nr 4).

```

db.Pracownik.updateOne(
  { imie: "Jan", nazwisko: "Kowalski" },
  {
    $set: {
      "filia.nazwa": "Filia nr 4",
      "filia.ulica": "Długa",
      "filia.nr_domu": "8",
      "filia.miasto": "Gdańsk",
      "filia.kod_pocztowy": "80-001",
      "filia.telefon": "666-777-888"
    }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

Rysunek 54. Przeniesienie pracownika

```

db.Pracownik.find({
  imie: "Jan", nazwisko: "Kowalski"
});
{
  _id: ObjectId('644b10b56f62a6dbec999998'),
  imie: 'Jan',
  nazwisko: 'Kowalski',
  filia: {
    nazwa: 'Filia nr 4',
    ulica: 'Długa',
    nr_domu: '8',
    miasto: 'Gdańsk',
    kod_pocztowy: '80-001',
    telefon: '666-777-888'
  },
}

```

Rysunek 55. Efekt przeniesienia pracownika

9. Zmiana języka książki na angielski.

```
db.Pozycja.updateOne(
  { _id: ObjectId("607f1f77bcf86cd799436007") },
  { $set: { jezyk: "angielski" } }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Rysunek 56. Zmiana języka

```
db.Pozycja.find({
  _id: ObjectId("607f1f77bcf86cd799436007")
});
{
  _id: ObjectId('607f1f77bcf86cd799436007'),
  tytul: 'Solaris',
  autor: 'Stanisław Lem',
  opis: 'Klasyka science fiction o kontakcie z
  rok_wydania: '1961-01-01',
  jezyk: 'angielski',
}
```

Rysunek 57. Efekt zmiany języka

Usuwanie

1. Zapytanie realizuje usuwanie książki o określonym id wraz ze wszystkimi powiązаныmi recenzjami z tą książką.

```
> // Pobranie ID książki
const bookId = ObjectId("607f1f77bcf86cd799436010");

// Usunięcie książki
db.Pozycja.deleteOne({ _id: bookId });

// Usuwanie recenzji powiązanej z tą książką
db.Recenzje.deleteMany({ idPozycja: bookId });
< {
  acknowledged: true,
  deletedCount: 0
}
```

Rysunek 58. Usunięcie książki

```

_id: ObjectId('607f1f77bcf86cd799436008')
tytuł: "Solaris"
autor: "Stanisław Lem"
opis: "Klasyka science fiction o kontakcie z obcą formą in
rok_wydania: "1961-01-01"
jezyk: "angielski"
typ: "ksiazka"
isbn: "9780156027601"
wydawnictwo: "Mariner Books"
▶ kategorie: Array (2)
▶ egzemplarze: Array (1)

```

```

_id: ObjectId('607f1f77bcf86cd799436009')
tytuł: "Solaris"
autor: "Stanisław Lem"
opis: "Klasyka science fiction o kontakcie z obcą formą in
rok_wydania: "1961-01-01"
jezyk: "polski"
typ: "audiobook"
format: "MP3"
lektor: "Janusz Gajos"
czas_trwania: "8:45:00"
link_dostepu: "https://biblioteka.pl/audiobooki/solaris.mp
dostepnosc: "dostepny"
data_dodania: "2023-10-20"

```

Rysunek 59. Efekt usunięcia książki

2. Usuń wszystkich czytelników, którzy nie mają przypisanego żadnego wypożyczenia ani aktywnej rezerwacji - używane do czyszczenia „martwych kont”, które nic nie wypożyczyły i nie mają aktywności.

```

db.Czytelnik.deleteMany({
  idWypozyczenie: null,
  $or: [
    { "rezerwacja.status": null },
    { "rezerwacja.status": { $exists: false } }
  ]
});
{
  acknowledged: true,
  deletedCount: 0
}

```

Rysunek 60. Usunięcie nieaktywnych czytelników

Baza nie posiada „martwych kont”.

3. Usunięcie ostatniego zamówienia pracownika z Filii nr 3.

```
db.Pracownik.updateOne(
  { "filia.nazwa": "Filia nr 3" },
  {
    $pop: { zamowienia: 1 }
  }
);
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Rysunek 61. Usunięcie zamówienia

```
> db.Pracownik.find({
  "filia.nazwa": "Filia nr 3"
});
< {
  _id: ObjectId('644b10b56f62a6dbec999'),
  imie: 'Ewa',
  nazwisko: 'Wiśniewska',
  filia: {
    nazwa: 'Filia nr 3',
    ulica: 'Jana Pawła II',
    nr_domu: '20',
    miasto: 'Wrocław',
    kod_pocztowy: '50-001',
    telefon: '444-555-666'
  },
  zamowienia: [],
}
```

Rysunek 62. Efekt usunięcia zamówienia

Użytkownicy

User ⓘ	Description	Authentication Method ⓘ	MongoDB Roles
admin	admin	SCRAM	atlasAdmin@admin
czytelnik	czytelnik	SCRAM	readAnyDatabase@admin
pracownik	pracownik	SCRAM	readWriteAnyDatabase@admin

Rysunek 63. Lista użytkowników bazy Biblioteka

1. Użytkownik z rolą "Czytelnik"

- **Uprawnienia:** Odczyt danych z kolekcji pozycje (książki, dostępność), wypożyczenie (informacje o swoich wypożyczeniach).
- **Przykład:** Może odczytywać książki i swoje wypożyczenia, ale nie może dodawać, usuwać ani edytować danych.

2. Użytkownik z rolą "Pracownik"

- **Uprawnienia:** Odczyt i zapis w kolekcjach wypożyczenie, czytelnicy, pozycje. Może dodawać nowe książki, wypożyczać książki, edytować dane czytelników, ale nie ma dostępu do administracji systemem.
- **Przykład:** Może zarządzać wypożyczeniami i książkami, ale nie może tworzyć nowych użytkowników ani zarządzać uprawnieniami.

3. Użytkownik z rolą "Admin"

- **Uprawnienia:** Pełna kontrola nad systemem, w tym zarządzanie użytkownikami, książkami, wypożyczeniami oraz pracownikami.
- **Przykład:** Może tworzyć i usuwać użytkowników, zmieniać uprawnienia, zarządzać wszystkimi danymi w systemie.

Testowanie zapytań dla użytkownika z rolą “Czytelnik”:

```
> db.pozycja.find({ kategorie: "literatura piękna" })
< {
  _id: ObjectId('607f1f77bcf86cd799436001'),
  tytuł: 'Lalka',
  autor: 'Bolesław Prus',
  opis: 'Powieść społeczno-obyczajowa z okresu pozytywizmu.',
  rok_wydania: '1890-01-01',
  język: 'polski',
  typ: 'książka',
  isbn: '9788373271280',
  wydawnictwo: 'Wydawnictwo Literackie',
  kategorie: [
    'literatura piękna',
    'klasyka'
  ],
  egzemplarze: [
    {
      _id: ObjectId('707f1f77bcf86cd799437001'),
      numer_inwentarzowy: 'INV-101',
      status: 'dostępny',
      lokalizacja: {
        filia: 'Filia nr 1',
        regał: 'R-2',
        półka: 'P-3'
      },
      data_dodania: '2023-02-15'
    },
    {
      _id: ObjectId('707f1f77bcf86cd799437002'),
      numer_inwentarzowy: 'INV-102',
      status: 'wypożyczony',
      lokalizacja: {
        filia: 'Filia nr 3',
        regał: 'R-1',
        półka: 'P-5'
      },
      data_dodania: '2023-05-20'
    }
  ]
}
```

Atlas atlas-x1xomf-shard-0 [primary] Biblioteka>

Rysunek 64. Zapytanie pobierające książki z kategorii literatura piękna

```
> db.pozycja.insertOne({
  tytuł: "Nowa Książka",
  autor: "Nowy Autor",
  rok_wydania: "2025",
  isbn: "9781234567890",
  wydawnictwo: "Testowe Wydawnictwo"
})
MongoServerError[AtlasError]: user is not allowed to do action [insert] on [Biblioteka.pozycja]
Atlas atlas-x1xomf-shard-0 [primary] Biblioteka>
```

Rysunek 65. Zapytanie wstawiające nową pozycję

Testowanie zapytań dla użytkownika z rolą “Pracownik”:

```
> db.pozycja.insertOne({
  tytuł: "Nowa Książka",
  autor: "Autor Testowy",
  opis: "Opis książki",
  rok_wydania: "2025-01-01",
  język: "polski",
  typ: "książka",
  isbn: "9781234567890",
  wydawnictwo: "Test",
  kategorie: ["literatura polska"]
})
< {
  acknowledged: true,
  insertedId: ObjectId('681929261c349e53aa689d20')
}
Atlas atlas-x1xomf-shard-0 [primary] test>
```

Rysunek 66. Zapytanie realizujące wstawianie nowej książki

```
> db.createUser({
  user: "czytelnik_anna",
  pwd: "hasło_czytelnika",
  roles: [
    { role: "read", db: "biblioteka" }
  ]
})
❗ ▶ MongoServerError[AtlasError]: CMD_NOT_ALLOWED: createUser
Atlas atlas-x1xomf-shard-0 [primary] test>
```

Rysunek 67. Zapytanie tworzące nowego użytkownika

Testowanie zapytań dla użytkownika z rolą “Admin”:

```
> db.createUser({
  user: "czytelnik_anna",
  pwd: "hasło_czytelnika",
  roles: [
    { role: "read", db: "biblioteka" }
  ]
})
< { ok: 1 }
test>
```

Rysunek 68. Zapytanie tworzące nowego użytkownika

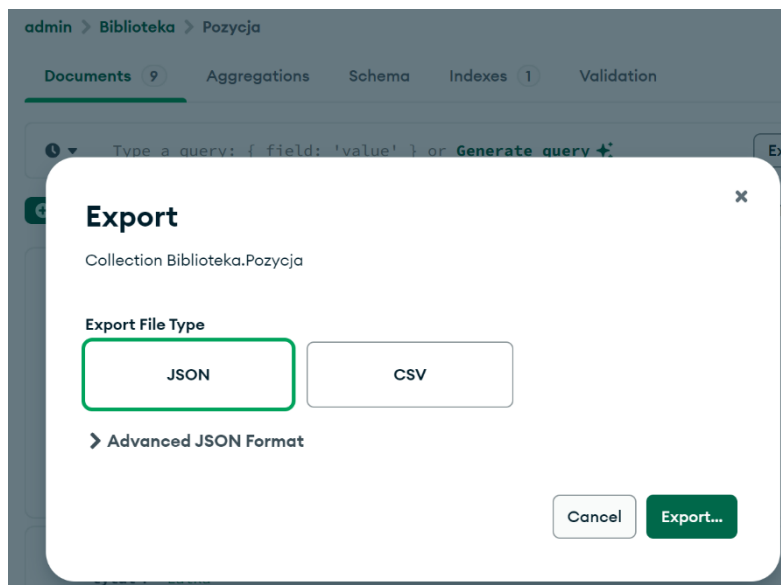
```
> db.dropUser("czytelnik_anna")
< { ok: 1 }
```

Rysunek 69. Zapytanie usuwające użytkownika

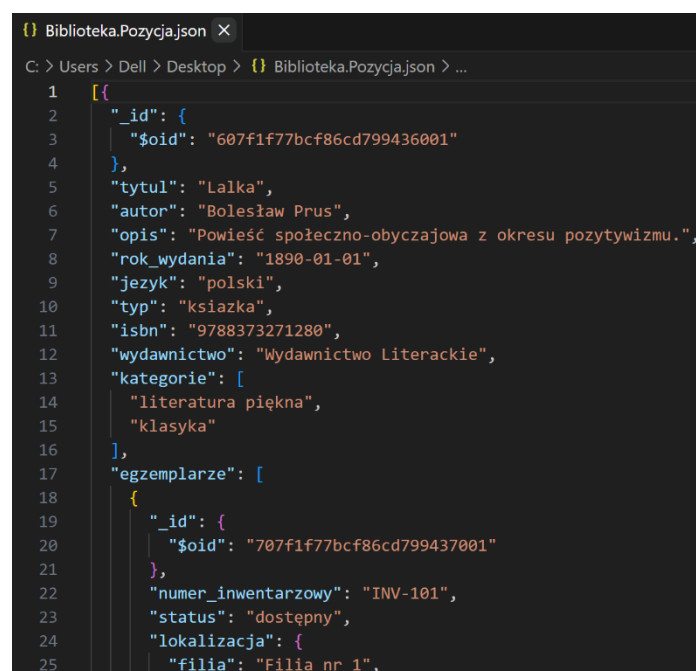
```
> db.pracownik.insertOne({
  imie: "Tomasz",
  nazwisko: "Nowak",
  stanowisko: "Bibliotekarz",
  filia: { nazwa: "Filia nr 2", miasto: "Kraków" },
  data_zatrudnienia: new Date("2023-01-01"),
  status: "Aktywny",
  kontakt: { telefon: "123-456-789", email: "tomasz.nowak@biblioteka.pl" }
})
< {
  acknowledged: true,
  insertedId: ObjectId('68192a6bfd79b633b4cfbcf6')
}
test>
```

Rysunek 70. Zapytanie wstawiające nowego pracownika

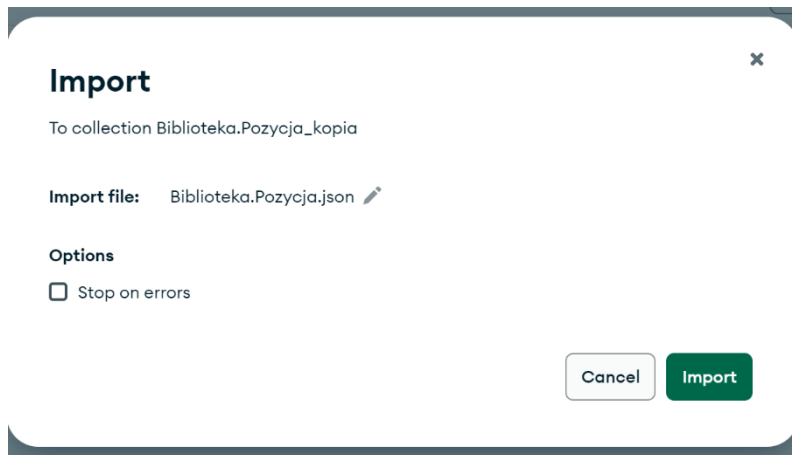
Eksport, import i kopia zapasowa



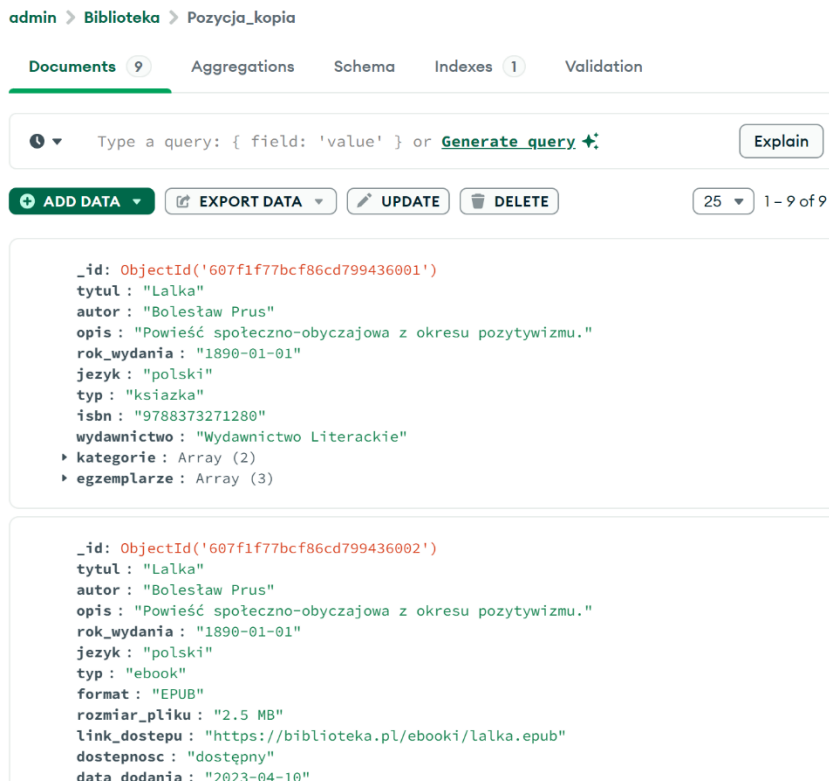
Rysunek 71. Eksport kolekcji Pozycja



Rysunek 72. Widok wyeksportowanych danych z kolekcji Pozycja w formacie JSON



Rysunek 73. Import kolekcji Pozycja do kolekcji Pozycja_kopia



Rysunek 74. Widok zaimportowanych danych w kolekcji Pozycja_kopia

```
PS C:\Program Files\MongoDB\Server\8.0\bin\bin> .\mongodump.exe --db Biblioteka --out "C:\Users\Dell\Desktop\MongoCopy"
2025-05-05T23:00:37.993+0200 writing Biblioteka.Recenzja to C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Recenzja.bson
2025-05-05T23:00:38.001+0200 writing Biblioteka.Pracownik to C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pracownik.bson
2025-05-05T23:00:38.002+0200 writing Biblioteka.Czytelnik to C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Czytelnik.bson
2025-05-05T23:00:38.002+0200 writing Biblioteka.Wypozyczenie to C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Wypozyczenie.bson
2025-05-05T23:00:38.003+0200 done dumping Biblioteka.Pracownik (10 documents)
2025-05-05T23:00:38.005+0200 writing Biblioteka.Pozycja to C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pozycja.bson
2025-05-05T23:00:38.006+0200 done dumping Biblioteka.Pozycja (9 documents)
2025-05-05T23:00:38.007+0200 done dumping Biblioteka.Recenzja (11 documents)
2025-05-05T23:00:38.007+0200 done dumping Biblioteka.Wypozyczenie (10 documents)
2025-05-05T23:00:38.008+0200 done dumping Biblioteka.Pozycja (9 documents)
```

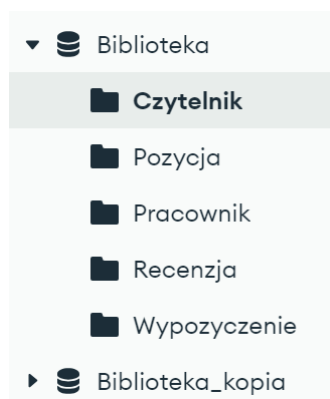
Rysunek 75. Utworzenie kopii zapasowej bazy Biblioteka

```

PS C:\Program Files\MongoDB\Server\8.0\bin\bin> .\mongorestore.exe -d Biblioteka_kopia "C:\Users\Dell\Desktop\MongoCopy\Biblioteka"
2025-05-05T23:03:53.306+0200 The --db and --collection flags are deprecated for this use-case; please use --nsInclude instead, i
.e. with --nsInclude=${DATABASE}.${COLLECTION}
2025-05-05T23:03:53.315+0200 building a list of collections to restore from C:\Users\Dell\Desktop\MongoCopy\Biblioteka dir
2025-05-05T23:03:53.316+0200 don't know what to do with file "C:\Users\Dell\Desktop\MongoCopy\Biblioteka\prelude.json", skipping
...
2025-05-05T23:03:53.317+0200 reading metadata for Biblioteka_kopia.Czytelnik from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Czy
telnik.metadata.json
2025-05-05T23:03:53.317+0200 reading metadata for Biblioteka_kopia.Pozycja from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pozyc
ja.metadata.json
2025-05-05T23:03:53.317+0200 reading metadata for Biblioteka_kopia.Pracownik from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pra
cownik.metadata.json
2025-05-05T23:03:53.318+0200 reading metadata for Biblioteka_kopia.Recenzja from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Rece
nzja.metadata.json
2025-05-05T23:03:53.320+0200 reading metadata for Biblioteka_kopia.Wypozyczenie from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\
Wypozyczenie.metadata.json
2025-05-05T23:03:53.361+0200 restoring Biblioteka_kopia.Pracownik from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pracownik.bson
2025-05-05T23:03:53.366+0200 restoring Biblioteka_kopia.Czytelnik from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Czytelnik.bson
2025-05-05T23:03:53.376+0200 restoring Biblioteka_kopia.Pozycja from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Pozycja.bson
2025-05-05T23:03:53.379+0200 finished restoring Biblioteka_kopia.Pracownik (10 documents, 0 failures)
2025-05-05T23:03:53.381+0200 restoring Biblioteka_kopia.Wypozyczenie from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Wypozyчени
e.bson
2025-05-05T23:03:53.385+0200 finished restoring Biblioteka_kopia.Czytelnik (10 documents, 0 failures)
2025-05-05T23:03:53.389+0200 finished restoring Biblioteka_kopia.Pozycja (9 documents, 0 failures)
2025-05-05T23:03:53.395+0200 finished restoring Biblioteka_kopia.Wypozyczenie (10 documents, 0 failures)
2025-05-05T23:03:53.395+0200 restoring Biblioteka_kopia.Recenzja from C:\Users\Dell\Desktop\MongoCopy\Biblioteka\Recenzja.bson
2025-05-05T23:03:53.410+0200 finished restoring Biblioteka_kopia.Recenzja (11 documents, 0 failures)
2025-05-05T23:03:53.410+0200 no indexes to restore for collection Biblioteka_kopia.Recenzja
2025-05-05T23:03:53.410+0200 no indexes to restore for collection Biblioteka_kopia.Wypozyczenie
2025-05-05T23:03:53.410+0200 no indexes to restore for collection Biblioteka_kopia.Czytelnik
2025-05-05T23:03:53.410+0200 no indexes to restore for collection Biblioteka_kopia.Pozycja
2025-05-05T23:03:53.411+0200 no indexes to restore for collection Biblioteka_kopia.Pracownik
2025-05-05T23:03:53.411+0200 50 document(s) restored successfully. 0 document(s) failed to restore.

```

Rysunek 76. Załadowanie kopii bazy Biblioteka jako Biblioteka_kopia



Rysunek 77. Widok w MongoDB