

**WOJSKOWA AKADEMIA TECHNICZNA**

im. Jarosława Dąbrowskiego

---

**WYDZIAŁ CYBERNETYKI**



# PROJEKT ZALICZENIOWY

Temat: **IMPLEMENTACJA MODELU BAZY DANYCH –  
WYPOŻYCZALNIA KSIĄŻEK**

Autor:

**Marcin KOZŁOWSKI**

---

Warszawa 2023

# 1. Analiza biznesowa

Prezentowany jest uproszczony model wypożyczalni książek. Dostępne są wypożyczenia książek w formie papierowej (materialnej) lub w formie cyfrowej z wyróżnieniem ebooka lub audiobooka. Prowadzony jest spis książek aktualnie dostępnych oraz tych które kiedyś znajdowały się lub będą się znajdować w wypożyczalni. Prowadzona jest dokumentacja zawierająca autorów tych książek oraz gatunki, do których należą tak by łatwiej można było polecić klientom wybrane tytuły z uwagi na ich zainteresowania. Ponadto zadeklarowani klienci mogą otrzymać powiadomienia o książkach z ich kategorii gatunkowych lub z ich ulubionych twórców literackich. Przyjęto uproszczenie w postaci nakładanych kar za nieterminowy zwrot tytułu w formie elektronicznej jako karę za nieterminowe zapłacenie za miesięczny abonament dostępu do wybranego tytułu.

Model umożliwia znalezienie książek o odpowiednim gatunku literackim, które zostały napisane przez określoną osobę lub osoby, które były współautorami.

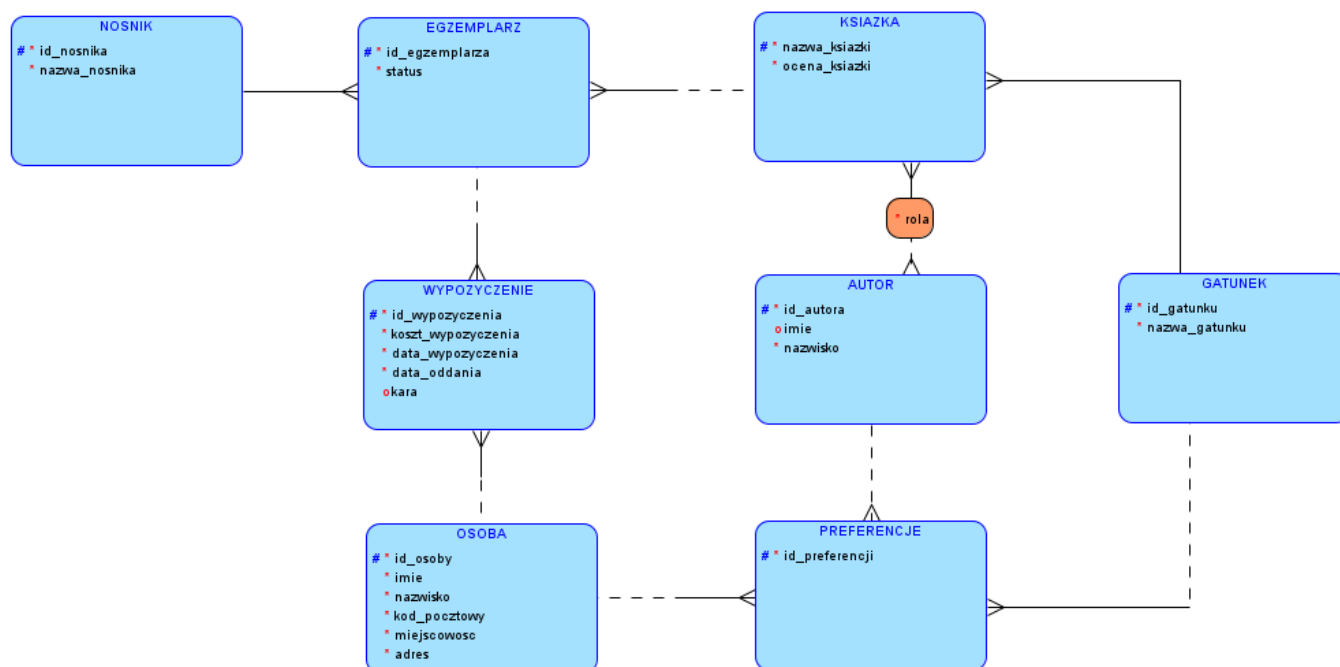
Model umożliwia również znalezienie odpowiednich klientów ze względu na wypożyczone tytuły literackie, ze względu na nośniki które wypożyczyli oraz wgląd do dat wypożyczeń i terminów zwrotów. Z założeniami przyjęto, że gdy dana osoba zwróci książkę (w dowolnej formie wypożyczenia) rekord o wypożyczeniu jest kasowany z bazy danych, natomiast w przypadku przekroczenia terminu oddania każdego dnia baza danych powinna naliczać karę do momentu zwrotu książki oraz zapłacenia kary.

Dostępne są poszczególne widoki prezentujące ilość wypożyczonych aktualnie książek w podziałach na nazwy książek oraz rodzaj nośnika, tak by można było dopasować model do najbardziej preferowanej opcji przez klientów oraz ich wyborów gatunków.

Dostępny jest widok prezentujący osoby z nieterminowym oddaniem książki, widok poleceń który może zostać wykorzystany przez pracowników wypożyczalni by dostosować tytuł do upodobań klienta oraz widok aktualnie wypożyczonych tytułów.

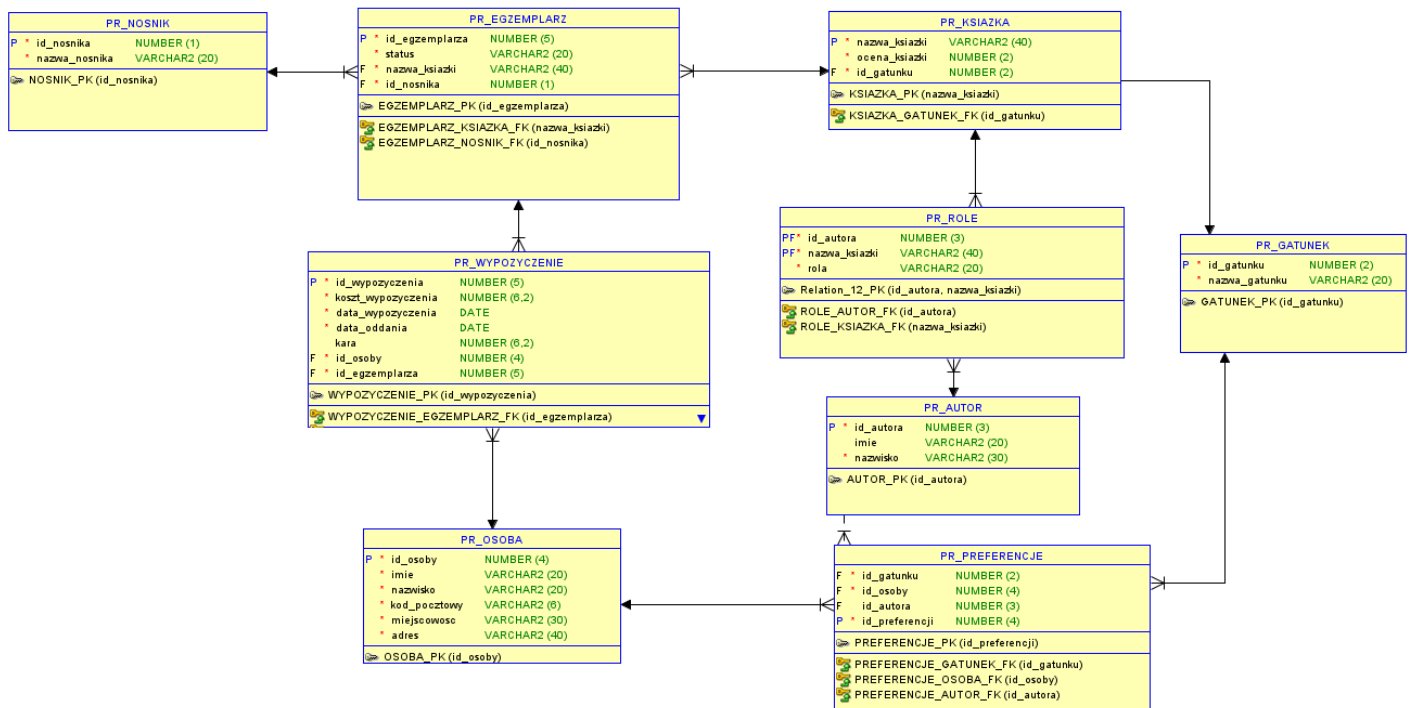
W załączonym pliku dostępne są poszczególne perspektywy oraz inne zapytania np. do wglądu w zyski miesięczne oraz roczne ze względu na prowadzoną działalność.

## 2. Model logiczny bazy danych



Obraz 1. Przedstawienie modelu logicznego bazy danych.

### 3. Model relacyjny bazy danych



Obraz 2. Przedstawienie modelu relacyjnego bazy danych.

## 4. Oprogramowanie tworzące bazę danych

W skład oprogramowania tworzącego bazę danych wchodzi:

- Oracle SQL Developer Tool,
- Oracle SQL Developer Data Modeler.

Oracle SQL Developer Tool umożliwia operacje na bazie danych takie jak:

- wprowadzanie nowych rekordów (np. rejestracja nowych wypożyczeń, nowych użytkowników, dostępnych książek, itp.),
- usuwanie zbędnych lub nieaktualnych rekordów (np. książek, które nie znajdują się już na stanie wypożyczalni, klientów, którzy zadeklarowali wypisanie się z wypożyczalni, wypożyczonych książek, gdy zostaną zwrócone),
- modyfikację rekordów (np. nałożenie kar za nieterminowy zwrot wypożyczonych pozycji, zmianę informacji o aktualnie zarejestrowanych użytkownikach oraz autorów, aktualizacje preferencji wybranych klientów wypożyczalni),
- zakładanie nowych tabel (w przypadku chęci poszerzenia działalności operacyjnej lub jej uszczegółowienia),
- usuwanie aktualnych tabel (w przypadku podjęcia decyzji, że część tabel jest zbędna w aktualnym systemie),
- modyfikację aktualnych tabel (np. dodanie nowych kolumn, umożliwiających przechowywanie większej ilości potrzebnych danych).

Oracle SQL Developer Tool jest narzędziem, które również umożliwia zakładanie, usuwanie oraz modyfikację tabel. Jest on natomiast preferowany ze względu na łatwiejsze, szybsze operacje na wielu tabelach oraz widoczność całego modelu bazy danych.

## 5. Skrypty wdrożeniowe

### A) Skrypt instalujący:

```
-- Generated by Oracle SQL Developer Data Modeler 22.2.0.165.1149
--   at:          2023-01-12 20:04:14 CET
--   site:        Oracle Database 11g
--   type:        Oracle Database 11g

-- predefined type, no DDL - MDSYS.SDO_GEOMETRY

-- predefined type, no DDL - XMLTYPE

CREATE TABLE pr_autor (
    id_autora NUMBER(3) NOT NULL,
    imie      VARCHAR2(20),
    nazwisko  VARCHAR2(30) NOT NULL
);

ALTER TABLE pr_autor ADD CONSTRAINT autor_pk PRIMARY KEY ( id_autora );

CREATE TABLE pr_egzemplarz (
    id_egzemplarza NUMBER(5) NOT NULL,
    status          VARCHAR2(20) NOT NULL,
    nazwa_ksiazki   VARCHAR2(40) NOT NULL,
    id_nosnika      NUMBER(1) NOT NULL
);

ALTER TABLE pr_egzemplarz ADD CONSTRAINT egzemplarz_pk PRIMARY KEY (
id_egzemplarza );

CREATE TABLE pr_gatunek (
    id_gatunku  NUMBER(2) NOT NULL,
    nazwa_gatunku VARCHAR2(20) NOT NULL
);

ALTER TABLE pr_gatunek ADD CONSTRAINT gatunek_pk PRIMARY KEY ( id_gatunku );

CREATE TABLE pr_ksiazka (
    nazwa_ksiazki VARCHAR2(40) NOT NULL,
    ocena_ksiazki  NUMBER(2) NOT NULL,
    id_gatunku     NUMBER(2) NOT NULL
);
```

```

ALTER TABLE pr_ksiazka ADD CONSTRAINT ksiazka_pk PRIMARY KEY ( nazwa_ksiazki
);

CREATE TABLE pr_nosnik (
    id_nosnika    NUMBER(1) NOT NULL,
    nazwa_nosnika VARCHAR2(20) NOT NULL
);

ALTER TABLE pr_nosnik ADD CONSTRAINT nosnik_pk PRIMARY KEY ( id_nosnika );

CREATE TABLE pr_osoba (
    id_osoby      NUMBER(4) NOT NULL,
    imie          VARCHAR2(20) NOT NULL,
    nazwisko      VARCHAR2(20) NOT NULL,
    kod_pocztowy  VARCHAR2(6) NOT NULL,
    miejscowosc   VARCHAR2(30) NOT NULL,
    adres         VARCHAR2(40) NOT NULL
);

ALTER TABLE pr_osoba ADD CONSTRAINT osoba_pk PRIMARY KEY ( id_osoby );

CREATE TABLE pr_preferencje (
    id_gatunku    NUMBER(2) NOT NULL,
    id_osoby      NUMBER(4) NOT NULL,
    id_autora     NUMBER(3),
    id_preferencji NUMBER(4) NOT NULL
);

ALTER TABLE pr_preferencje ADD CONSTRAINT preferencje_pk PRIMARY KEY (
id_preferencji );

CREATE TABLE pr_role (
    id_autora     NUMBER(3) NOT NULL,
    nazwa_ksiazki VARCHAR2(40) NOT NULL,
    rola         VARCHAR2(20) NOT NULL
);

ALTER TABLE pr_role ADD CONSTRAINT relation_12_pk PRIMARY KEY ( id_autora,
                                                                nazwa_ksiazki
);

CREATE TABLE pr_wypozyczenie (
    id_wypozyczenia    NUMBER(5) NOT NULL,
    koszt_wypozyczenia NUMBER(8, 2) NOT NULL,
    data_wypozyczenia  DATE NOT NULL,
    data_oddania       DATE NOT NULL,
    kara               NUMBER(8, 2),
    id_osoby           NUMBER(4) NOT NULL,
    id_egzemplarza     NUMBER(5) NOT NULL

```

```

);

ALTER TABLE pr_wypozyczenie ADD CONSTRAINT wypozyczenie_pk PRIMARY KEY (
id_wypozyczenia );

ALTER TABLE pr_egzemplarz
  ADD CONSTRAINT egzemplarz_ksiazka_fk FOREIGN KEY ( nazwa_ksiazki )
    REFERENCES pr_ksiazka ( nazwa_ksiazki );

ALTER TABLE pr_egzemplarz
  ADD CONSTRAINT egzemplarz_nosnik_fk FOREIGN KEY ( id_nosnika )
    REFERENCES pr_nosnik ( id_nosnika );

ALTER TABLE pr_ksiazka
  ADD CONSTRAINT ksiazka_gatunek_fk FOREIGN KEY ( id_gatunku )
    REFERENCES pr_gatunek ( id_gatunku );

ALTER TABLE pr_preferencje
  ADD CONSTRAINT preferencje_autor_fk FOREIGN KEY ( id_autora )
    REFERENCES pr_autor ( id_autora );

ALTER TABLE pr_preferencje
  ADD CONSTRAINT preferencje_gatunek_fk FOREIGN KEY ( id_gatunku )
    REFERENCES pr_gatunek ( id_gatunku );

ALTER TABLE pr_preferencje
  ADD CONSTRAINT preferencje_osoba_fk FOREIGN KEY ( id_osoby )
    REFERENCES pr_osoba ( id_osoby );

ALTER TABLE pr_role
  ADD CONSTRAINT role_autor_fk FOREIGN KEY ( id_autora )
    REFERENCES pr_autor ( id_autora );

ALTER TABLE pr_role
  ADD CONSTRAINT role_ksiazka_fk FOREIGN KEY ( nazwa_ksiazki )
    REFERENCES pr_ksiazka ( nazwa_ksiazki );

ALTER TABLE pr_wypozyczenie
  ADD CONSTRAINT wypozyczenie_egzemplarz_fk FOREIGN KEY ( id_egzemplarza )
    REFERENCES pr_egzemplarz ( id_egzemplarza );

ALTER TABLE pr_wypozyczenie
  ADD CONSTRAINT wypozyczenie_osoba_fk FOREIGN KEY ( id_osoby )
    REFERENCES pr_osoba ( id_osoby );

-- Oracle SQL Developer Data Modeler Summary Report:
--

```



```

-- CREATE TABLE          9
-- CREATE INDEX           0
-- ALTER TABLE          19
-- CREATE VIEW            0
-- ALTER VIEW             0
-- CREATE PACKAGE          0
-- CREATE PACKAGE BODY    0
-- CREATE PROCEDURE        0
-- CREATE FUNCTION         0
-- CREATE TRIGGER          0
-- ALTER TRIGGER           0
-- CREATE COLLECTION TYPE  0
-- CREATE STRUCTURED TYPE  0
-- CREATE STRUCTURED TYPE BODY 0
-- CREATE CLUSTER          0
-- CREATE CONTEXT          0
-- CREATE DATABASE         0
-- CREATE DIMENSION        0
-- CREATE DIRECTORY        0
-- CREATE DISK GROUP        0
-- CREATE ROLE              0
-- CREATE ROLLBACK SEGMENT  0
-- CREATE SEQUENCE         0
-- CREATE MATERIALIZED VIEW 0
-- CREATE MATERIALIZED VIEW LOG 0
-- CREATE SYNONYM          0
-- CREATE TABLESPACE      0
-- CREATE USER              0
--
-- DROP TABLESPACE        0
-- DROP DATABASE           0
--
-- REDACTION POLICY        0
--
-- ORDS DROP SCHEMA        0
-- ORDS ENABLE SCHEMA      0
-- ORDS ENABLE OBJECT      0
--
-- ERRORS                  0
-- WARNINGS                0

```

### A) Skrypt deinstalujący:

```
DELETE PR_WYPOZYCZENIE;  
DELETE PR_EGZEMPLARZ;  
DELETE PR_NOSNIK;  
DELETE PR_ROLE;  
DELETE PR_KSIAZKA;  
DELETE PR_PREFERENCJE;  
DELETE PR_GATUNEK;  
DELETE PR_AUTOR;  
DELETE PR_OSOBA;  
  
DROP TABLE PR_WYPOZYCZENIE;  
DROP TABLE PR_EGZEMPLARZ;  
DROP TABLE PR_NOSNIK;  
DROP TABLE PR_ROLE;  
DROP TABLE PR_KSIAZKA;  
DROP TABLE PR_PREFERENCJE;  
DROP TABLE PR_GATUNEK;  
DROP TABLE PR_AUTOR;  
DROP TABLE PR_OSOBA;
```

Skrypty wdrożeniowe, jak i skrypt wypełniający bazę danych przykładowymi rekordami zostały dołączone do sprawozdania w oddzielnych plikach.

## 6. Skrypty wdrożeniowe

### A) Instrukcja instalacji:

1. Otworzyć narzędzie Oracle SQL Developer.
2. Połączyć się z dowolnym serwerem Baz Danych (w którym mamy uprawnienia tworzenia, kasowania, modyfikacji tabel oraz wprowadzania do nich danych)
3. Uruchomić z poziomu bazy danych plik (skrypt) PR\_CREATE\_DB.sql lub otworzyć go manualnie i nacisnąć CTRL+A, a zawartość wprowadzić do Worksheet'a za pomocą CTRL+V i nacisnąć Run Statement lub CTRL+ENTER.
4. W przypadku błędów, sprawdzić czy plik nie został przypadkiem zmodyfikowany, gdyż skrypt wdrożeniowy został przetestowany wcześniej.
5. Identycznie jak w pkt 3. postąpić z plikiem (skryptem) PR\_INSERT\_ALL\_DB.sql .
6. W przypadku błędów, sprawdzić czy plik nie został przypadkiem zmodyfikowany, gdyż skrypt wdrożeniowy został przetestowany wcześniej.
7. Zgodność danych potwierdzić z wywołanym na końcu zapytaniem w skrypcie (który pokazuje ilość danych we wszystkich tabelach) a obrazem Ilosc\_danych.PNG .

'PR_AUTOR'	COUNT (*)
PR_AUTOR	10
PR_EGZEMPLARZ	80
PR_GATUNEK	9
PR_KSIAZKA	20
PR_NOSNIK	3
PR_OSOBA	30
PR_PREFERENCJE	19
PR_ROLE	27
PR_WYPOZYCZENIE	50

9 rows selected.

Obraz 3. Sprawdzenie poprawności tabel oraz wprowadzonych danych.

## **B) Instrukcja deinstalacji:**

W przypadku chęci usunięcia danych i tabel wcześniej zainstalowanych, uruchomić z poziomu bazy danych plik (skrypt) PR\_DELETE\_DB.sql lub otworzyć go manualnie i nacisnąć CTRL+A, a zawartość wprowadzić do Worksheet'a za pomocą CTRL+V i nacisnąć Run Statement lub CTRL+ENTER. W przypadku błędów upewnić się, że skrypt nie został zmodyfikowany oraz że wcześniej wykonano instalacje tabel na serwerze bazodanowym.

Instrukcje instalacji jak i deinstalacji zostały załączone również w oddzielnych plikach.

## 7. Przykładowe perspektywy oraz instrukcje typu SELECT

```
CREATE VIEW PO_TERMINIE AS
(SELECT wyp.id_wypozyczenia AS "ID wypożyczenia", os.imie AS "Imie",
os.nazwisko AS "Nazwisko", round((sysdate- TO_DATE(wyp.data_oddania,
'YY/MM/DD')),0) AS "Dni po terminie", ksk.nazwa_ksiazki AS "Nazwa książki",
wyp.kara AS "Kara"
FROM pr_wypozyczenie wyp
JOIN pr_osoba os ON wyp.id_osoby = os.id_osoby
JOIN pr_egzemplarz egz ON wyp.id_egzemplarza = egz.id_egzemplarza
JOIN pr_ksiazka ksk ON egz.nazwa_ksiazki = ksk.nazwa_ksiazki
WHERE kara IS NOT NULL

union

SELECT null, null, null, null, 'Razem', sum(nvl(kara,0)) FROM
pr_wypozyczenie);
```

	ID wypożyczenia	Imie	Nazwisko	Dni po terminie	Nazwa książki	Kara
1	26	Tasha	Vega	138	Fish And Giants	10,2
2	27	Blake	Banks	132	Victory Of The Banished	1,69
3	28	Elvira	Hammond	187	Completion Of Next Year	2,18
4	29	Bartholomew	Curtis	157	Dead At The Castle	10,12
5	30	Hanley	Hodgson	114	Criminal With Sins	11,68
6	31	Elvira	Hammond	185	Turtle Without Direction	14,41
7	32	Lillian	Bailey	145	Guardians Of Greatness	20,46
8	33	Eugenia	Bowen	125	Butchers With Determination	19,06
9	36	Virgil	Montgomery	130	Victory Of Perfection	9,57
10	37	Eugenia	Bowen	157	Death In My Dreams	10,29
11	38	Elmer	Welch	114	Enter The Hunter	4,68
12	39	Patrick	Dean	171	King Of The Stockades	19,57
13	40	Tasha	Vega	125	Blacksmith With Gold	10,29
14	44	Erin	Sharp	130	Butchers With Gold	5,55
15	45	Hanley	Hodgson	95	Witches And Slaves	21,33
16	46	Elmer	Welch	122	Fish And Giants	8,37
17	47	Walter	Cross	104	Victory Of The Banished	2,18
18	48	Lillian	Bailey	149	Completion Of Next Year	9,13
19	49	Hanley	Hodgson	92	Dead At The Castle	2,54
20	50	Wallace	Hardy	46	Criminal With Sins	10,5
21	(null)	(null)	(null)	(null)	Razem	203,8

Obraz 4. Wynik perspektywy PO\_TERMINIE

```
CREATE VIEW ILOSC_WYPOZYCZEN AS
(SELECT COUNT(wyp.id_egzemplarza) AS "Ilość wypożyczeń", ksk.nazwa_książki AS
"Nazwa książki"
FROM pr_wypozyczenie wyp
JOIN pr_osoba os ON wyp.id_osoby = os.id_osoby
JOIN pr_egzemplarz egz ON wyp.id_egzemplarza = egz.id_egzemplarza
JOIN pr_książka ksk ON egz.nazwa_książki = ksk.nazwa_książki
GROUP BY ksk.nazwa_książki);
```

	↕ Ilość wypożyczeń	↕ Nazwa książki
1	2	Origin Of Joy
2	2	Victory Of Perfection
3	3	Criminal Of The Plague
4	3	Butchers With Gold
5	3	Witches And Slaves
6	2	Guardians Of Greatness
7	2	Hunters And Guardians
8	2	King Of The Stockades
9	2	Blacksmith With Gold
10	2	Turtle Without Direction
11	2	Butchers With Determination
12	3	Pirate Of The South
13	3	Victory Of The Banished
14	2	Death In My Dreams
15	3	Bandits Of The World
16	3	Fish And Giants
17	2	Enter The Hunter
18	3	Completion Of Next Year
19	3	Dead At The Castle
20	3	Criminal With Sins

Obraz 5. Wynik perspektywy ILOSC\_WYPOZYCZEN

```
CREATE VIEW WYPOZYCZENIA_PODZIAL AS
(SELECT nsk.nazwa_nosnika AS "Nazwa nośnika", COUNT(egz.id_nosnika) AS "Ilość
wypożyczeń", ksk.nazwa_książki AS "Nazwa książki"
FROM pr_egzemplarz egz
JOIN pr_książka ksk ON egz.nazwa_książki = ksk.nazwa_książki
JOIN pr_nosnik nsk ON egz.id_nosnika = nsk.id_nosnika
WHERE egz.status = 'BORROWED'
GROUP BY nsk.nazwa_nosnika, ksk.nazwa_książki);
```

	Nazwa nośnika	Ilość wypożyczeń	Nazwa książki
1	AUDIOBOOK	1	Witches And Slaves
2	PAPER	1	Witches And Slaves
3	PAPER	1	Criminal With Sins
4	AUDIOBOOK	1	Criminal With Sins
5	AUDIOBOOK	1	Guardians Of Greatness
6	AUDIOBOOK	1	Hunters And Guardians
7	AUDIOBOOK	1	Victory Of Perfection
8	PAPER	1	King Of The Stockades
9	PAPER	1	Blacksmith With Gold
10	EBOOK	1	Bandits Of The World
11	EBOOK	1	Criminal With Sins
12	AUDIOBOOK	1	Fish And Giants
13	PAPER	1	Hunters And Guardians
14	AUDIOBOOK	1	Victory Of The Banished
15	AUDIOBOOK	1	Completion Of Next Year
16	AUDIOBOOK	1	Dead At The Castle
17	AUDIOBOOK	1	Turtle Without Direction
18	AUDIOBOOK	1	Origin Of Joy
19	PAPER	1	Butchers With Gold
20	PAPER	1	Dead At The Castle
21	EBOOK	1	Blacksmith With Gold
22	EBOOK	1	Butchers With Gold
23	EBOOK	1	Dead At The Castle
24	AUDIOBOOK	1	Butchers With Gold
25	PAPER	1	Turtle Without Direction
26	PAPER	1	Butchers With Determination
27	EBOOK	1	King Of The Stockades
28	EBOOK	1	Witches And Slaves
29	PAPER	1	Criminal Of The Plague
30	PAPER	1	Completion Of Next Year
31	PAPER	1	Victory Of Perfection
32	PAPER	1	Death In My Dreams

Obraz 6. Częściowy wynik perspektywy ILOSC\_WYPOZYCZEN

```
CREATE VIEW POLECANIE_KSIAZEK AS
(SELECT os.imie AS "Imie klienta", os.nazwisko AS "Nazwisko klienta",
gat.nazwa_gatunku AS "Ulubione gatunki", aut.nazwisko AS "Ulubieni autorzy"
FROM pr_osoba os
JOIN pr_preferencje prf ON os.id_osoby = prf.id_osoby
LEFT OUTER JOIN pr_autor aut ON prf.id_autora = aut.id_autora
LEFT OUTER JOIN pr_gatunek gat ON prf.id_gatunku = gat.id_gatunku);
```

	Imie klienta	Nazwisko klienta	Ulubione gatunki	Ulubieni autorzy
1	Erin	Sharp	SCI-FI	Warren
2	Isaiah	Salvage	SCI-FI	Warren
3	Scout	Kinsman	SCI-FI	Warren
4	Isaiah	Salvage	THRILLER	Hayes
5	Scout	Kinsman	THRILLER	Hayes
6	Elvira	Hammond	THRILLER	Hayes
7	Hadwin	Vasquez	HORROR	Cobbett
8	Trina	Robson	FANTASY	Carpenter
9	Tracy	Foster	ROMANCE	Johnston
10	Hanley	Hodgson	HISTORICAL	Vega
11	Tasha	Vega	BIOGRAPHY	Buckley
12	Walter	Cross	DOCUMENTAL	Fairbank
13	Wallace	Hardy	CULINARY	Arnold
14	Echo	Rodgers	CULINARY	Arnold
15	Lillian	Bailey	DOCUMENTAL	Fairbank
16	Elvira	Hammond	BIOGRAPHY	Buckley
17	Isaiah	Salvage	BIOGRAPHY	Buckley
18	Raymond	Andrus	HISTORICAL	Lyons
19	Wallace	Hardy	HISTORICAL	Vega

*Obraz 7. Wynik perspektywy POLECANIE\_KSIAZEK*



```
SELECT id_osoby, COUNT(kara)
FROM pr_wypozyczenie
GROUP BY id_osoby
HAVING COUNT(kara) > 1
```

	ID_OSOBY	COUNT(KARA)
1	22	2
2	25	2
3	21	2
4	20	3
5	5	2
6	18	2

Obraz 8. Wynik instrukcji typu SELECT, pokazujący osoby z więcej niż jedną nałożoną karą.

```
SELECT EXTRACT(YEAR FROM data_wypozyczenia) AS "Rok",
to_char(to_date(data_wypozyczenia, 'YY/MM/DD'), 'Month') AS "Miesiąc",
COUNT(id_wypozyczenia) AS "Ilość wypożyczeń"
FROM PR_WYPOZYCZENIE
GROUP BY EXTRACT(YEAR FROM data_wypozyczenia),
to_char(to_date(data_wypozyczenia, 'YY/MM/DD'), 'Month');
```

	Rok	Miesiąc	Ilość wypożyczeń
1	2022	Wrzesień	6
2	2022	Sierpień	12
3	2022	Czerwiec	13
4	2022	Lipiec	13
5	2022	Październik	6

Obraz 9. Wynik instrukcji typu SELECT, pokazujący ilość wypożyczeń z podziałem na miesiąc i rok.

```
SELECT EXTRACT(YEAR FROM data_wypozyczenia) AS "Rok", EXTRACT(MONTH FROM
data_wypozyczenia) AS "Miesiąc", SUM(koszt_wypozyczenia) AS "Zysk miesięczny",
SUM(nvl(kara,0)) as "Kary"
FROM PR_WYPOZYCZENIE
GROUP BY EXTRACT(YEAR FROM data_wypozyczenia), EXTRACT(MONTH FROM
data_wypozyczenia);
```

	Rok	Miesiąc	Zysk miesięczny	Kary
1	2022	8	873,74	59,6
2	2022	9	366,23	0
3	2022	10	206,06	0
4	2022	6	969,42	89,8
5	2022	7	924,93	54,4

Obraz 10. Wynik instrukcji typu SELECT, pokazujący zyski z działalności operacyjnej jak i nałożone kary z podziałem na miesiące i lata.

```
SELECT id_wypozyczenia, imie, nazwisko
FROM pr_wypozyczenie wyp
JOIN pr_osoba os ON wyp.id_osoby = os.id_osoby
WHERE id_egzemplarza IN (SELECT id_egzemplarza FROM pr_egzemplarz egz
JOIN pr_ksiazka ksk ON egz.nazwa_ksiazki = ksk.nazwa_ksiazki
JOIN pr_role rol ON ksk.nazwa_ksiazki = rol.nazwa_ksiazki
JOIN pr_autor aut ON rol.id_autora = aut.id_autora
WHERE aut.nazwisko = 'Warren' AND rol.rola = 'AUTHOR');
```

	ID_WYPOZYCZENIA	IMIE	NAZWISKO
1	1	Rosa	Maldonado
2	2	Rosa	Maldonado
3	3	Elvira	Hammond
4	21	Zachary	Glisson
5	22	Violet	Long
6	23	Echo	Rodgers
7	41	Bartholomew	Curtis
8	42	Erin	Sharp
9	43	Wallace	Hardy

Obraz 11. Wynik instrukcji typu SELECT, pokazujący osoby z wypożyczonymi egzemplarzami książek, gdzie Warren jest autorem książki.

```

SELECT ROWNUM, nazwa_ksiazki
FROM (SELECT nazwa_ksiazki, COUNT(status)
FROM pr_egzemplarz egz
WHERE egz.status = 'BORROWED'
GROUP BY nazwa_ksiazki
ORDER BY COUNT(status))
WHERE ROWNUM < 6;

```

ROWNUM	NAZWA_KSIAZKI
1	1 Origin Of Joy
2	2 Blacksmith With Gold
3	3 King Of The Stockades
4	4 Hunters And Guardians
5	5 Guardians Of Greatness

Obraz 12. Wynik instrukcji typu *SELECT*, pokazujący TOP 5 wypożyczonych aktualnie książek (z uwagi na ilość wypożyczeń)