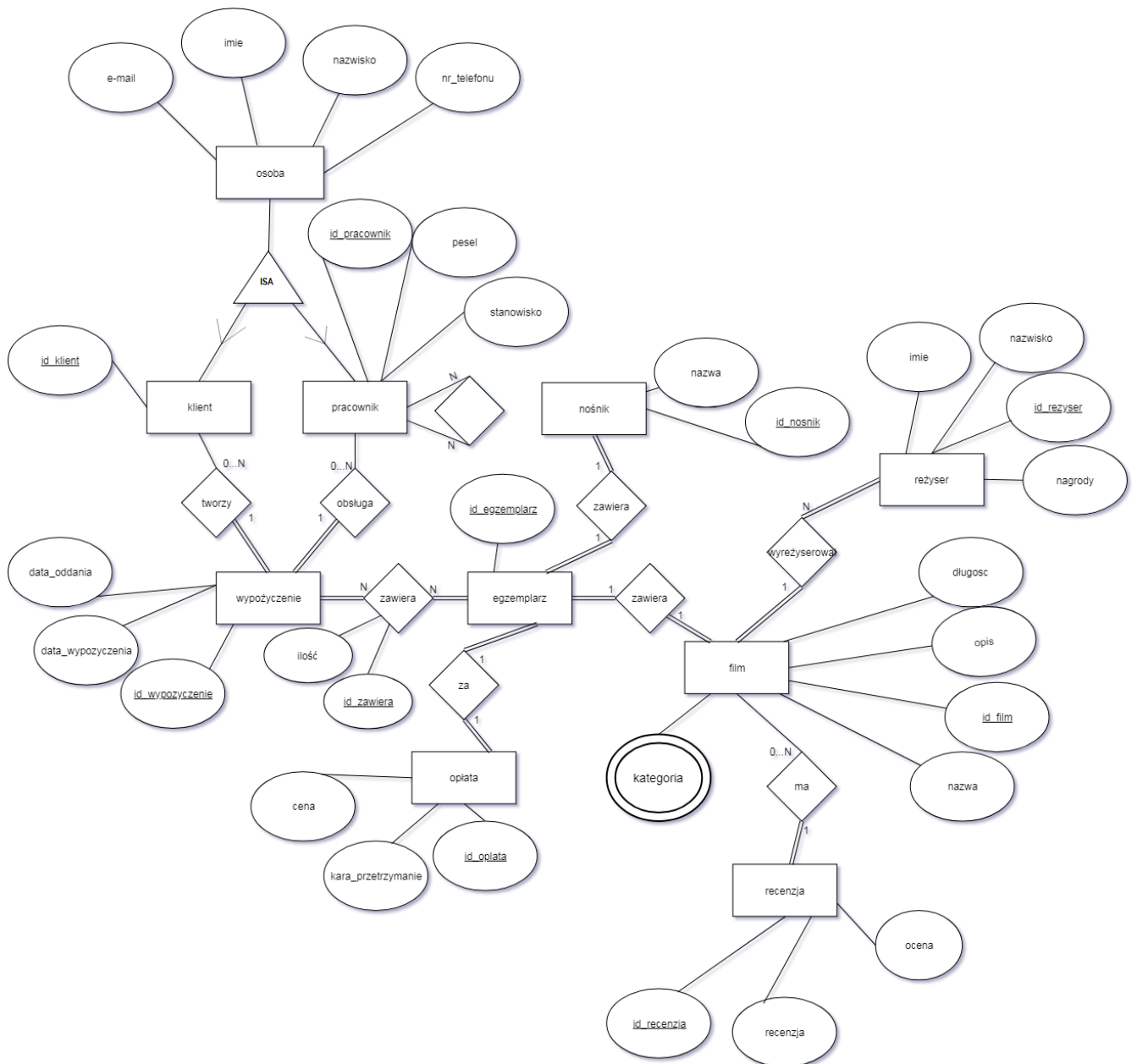


MATEUSZ KUCAJ GRUPA 4 110953

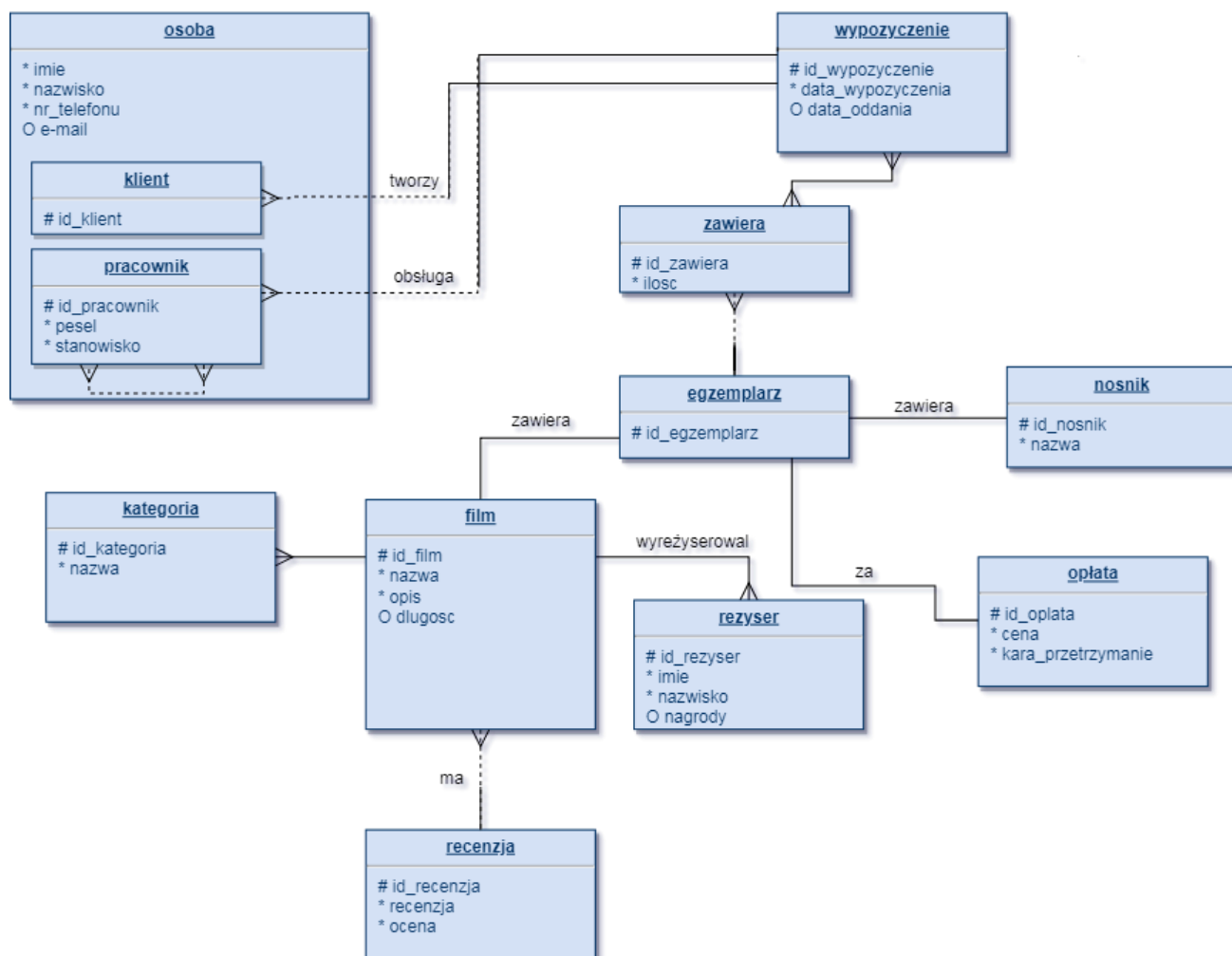
PROJEKT BAZY DANYCH

WYPOŻYCZALNIA DVD

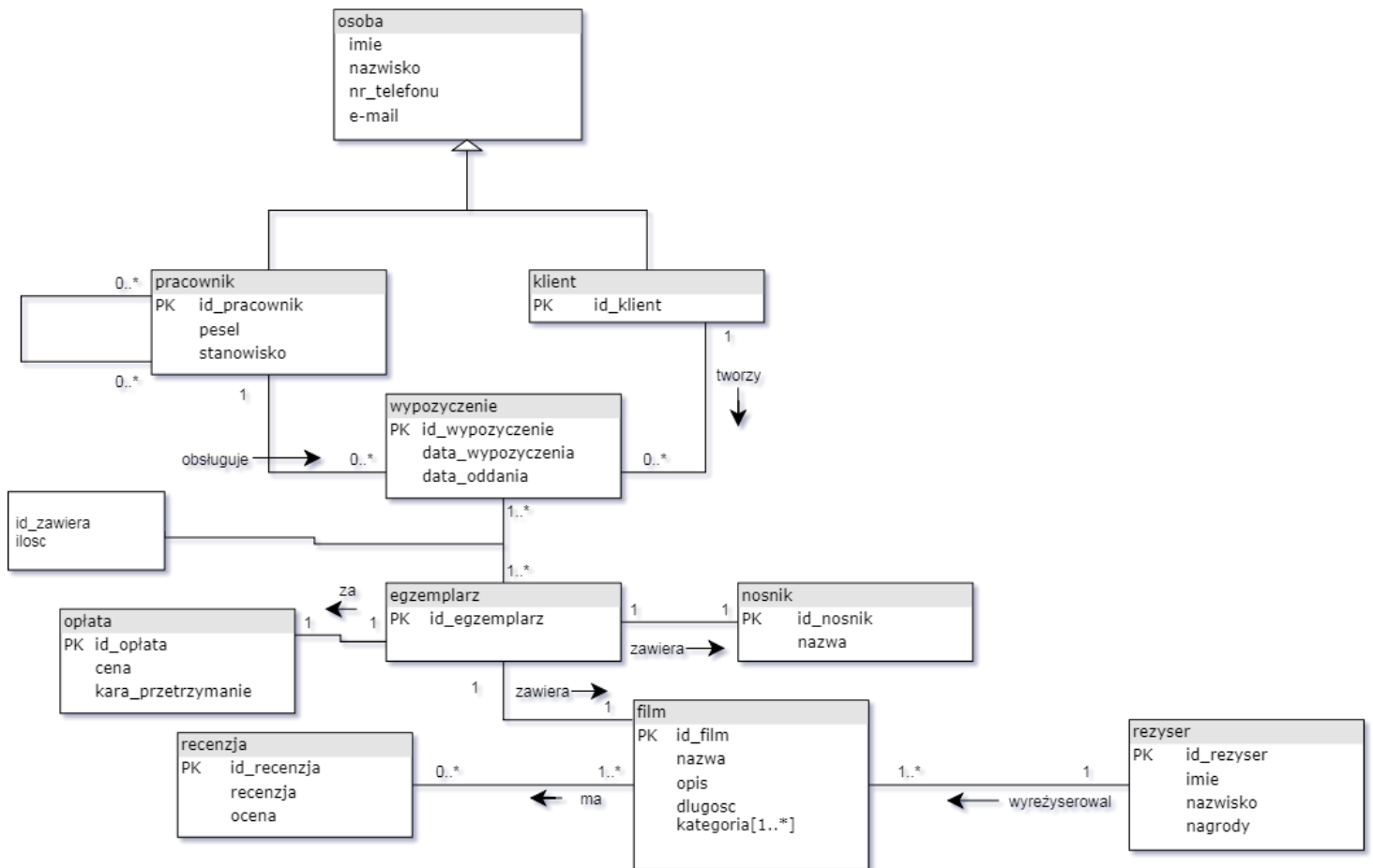
NOTACJA CHENA

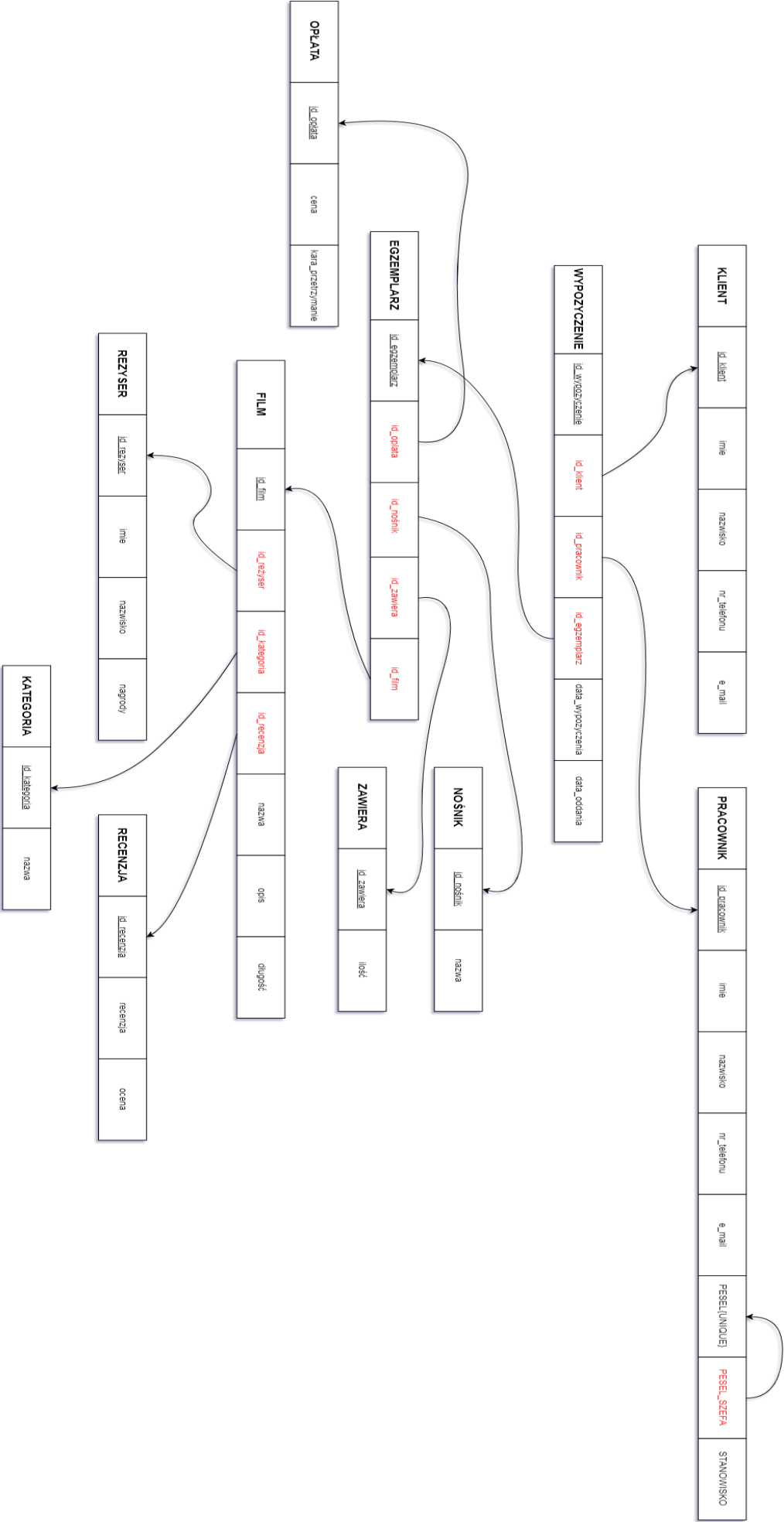


NOTACJA BARKER



NOTACJA UML





Transformacja hierarchii encji schemat 2 – utworzenie dwóch tabel z atrybutami encji podklasy i z atrybutami encji nadklasy (atrybuty encji nadklasy występują w dwóch tabelach). W tym przypadku utworzenie tabeli Pracownik i Klient z atrybutami encji nadklasy.

Związek unarny w modelu relacyjnym – PESEL_SZEFA jest kluczem obcym w tabeli pracownik a PESEL jest niepowtarzalny, pracownik może być przełożonym lub podwładnym

Atrybut wielowartościowy „kategoria” – tworzona jest nowa tabela o nazwie kategoria, jest w relacji z tabelą film, film może mieć kilka kategorii

Encja wypożyczenie jest w relacji „zawiera” (z atrybutami) z encją egzemplarz – tworzona jest nowa tabela o nazwie „zawiera”, która posiada kolumny id_zawiera i ilość, jest ona w relacji z tabelą egzemplarz.

Tworzenie tabel:

EGZEMPLARZ [\[open in SQL editor\]](#)

```
CREATE TABLE EGZEMPLARZ
(
  ID_EGZEMPLARZ integer NOT NULL,
  ID_FILM integer NOT NULL,
  ID_ZAWIERA integer NOT NULL,
  ID_NOSNIK integer NOT NULL,
  ID_OPLATA integer,
  CONSTRAINT PK_EGZEMPLARZ_0 PRIMARY KEY (ID_EGZEMPLARZ)
);

ALTER TABLE EGZEMPLARZ ADD CONSTRAINT FK_EGZEMPLARZ_0
  FOREIGN KEY (ID_FILM) REFERENCES FILM (ID_FILM) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE EGZEMPLARZ ADD CONSTRAINT FK_EGZEMPLARZ_1
  FOREIGN KEY (ID_ZAWIERA) REFERENCES ZAWIERA (ID_ZAWIERA) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE EGZEMPLARZ ADD CONSTRAINT FK_EGZEMPLARZ_2
  FOREIGN KEY (ID_NOSNIK) REFERENCES NOSNIK (ID_NOSNIK) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE EGZEMPLARZ ADD CONSTRAINT FK_EGZEMPLARZ_3
  FOREIGN KEY (ID_OPLATA) REFERENCES OPLATA (ID_OPLATA) ON UPDATE NO ACTION ON DELETE NO ACTION;
GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
  ON EGZEMPLARZ TO SYSDBA WITH GRANT OPTION;
```

FILM

```
CREATE TABLE FILM
(
  ID_FILM integer NOT NULL,
  ID_REZYSER integer NOT NULL,
  ID_KATEGORIA integer NOT NULL,
  ID_RECENZJA integer NOT NULL,
  NAZWA varchar(100),
  OPIS varchar(500),
  DLUGOSC varchar(30),
  CONSTRAINT PK_FILM_0 PRIMARY KEY (ID_FILM)
);

ALTER TABLE FILM ADD CONSTRAINT FK_FILM_0
  FOREIGN KEY (ID_REZYSER) REFERENCES REZYSER (ID_REZYSER) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE FILM ADD CONSTRAINT FK_FILM_1
  FOREIGN KEY (ID_KATEGORIA) REFERENCES KATEGORIA (ID_KATEGORIA) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE FILM ADD CONSTRAINT FK_FILM_2
  FOREIGN KEY (ID_RECENZJA) REFERENCES RECENZJA (ID_RECENZJA) ON UPDATE NO ACTION ON DELETE NO ACTION;
GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
  ON FILM TO SYSDBA WITH GRANT OPTION;
```

```
CREATE TABLE KATEGORIA
(
  ID_KATEGORIA integer NOT NULL,
  NAZWA varchar(100),
  CONSTRAINT PK_KATEGORIA_0 PRIMARY KEY (ID_KATEGORIA)
);
```

```
CREATE TABLE KLIENT
(
  IMIE varchar(30) NOT NULL,
  NAZWISKO varchar(30) NOT NULL,
  NR_TELEFONU varchar(30) NOT NULL,
  E_MAIL varchar(100),
  ID_KLIENT integer NOT NULL,
  CONSTRAINT PK_KLIENT_0 PRIMARY KEY (ID_KLIENT)
);
```

```
CREATE TABLE NOSNIK
(
  ID_NOSNIK integer NOT NULL,
  NAZWA varchar(10),
  CONSTRAINT PK_NOSNIK_0 PRIMARY KEY (ID_NOSNIK)
);
```

```
CREATE TABLE OPLATA
(
  ID_OPLATA integer NOT NULL,
  CENA double precision,
  KARA_PRZETRZYMANIE double precision,
  CONSTRAINT PK_OPLATA_0 PRIMARY KEY (ID_OPLATA)
);

GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
ON OPLATA TO SYSDBA WITH GRANT OPTION;
```

```

CREATE TABLE PRACOWNIK
(
    ID_PRACOWNIK integer NOT NULL,
    IMIE varchar(30),
    NAZWISKO varchar(50),
    NR_TELEFONU varchar(15),
    E_MAIL varchar(50),
    PESEL varchar(11),
    PESELSZEFA varchar(11),
    STANOWISKO varchar(50),
    CONSTRAINT PK_PRACOWNIK_0 PRIMARY KEY (ID_PRACOWNIK),
    CONSTRAINT FK_PRA_U UNIQUE (PESEL)
);

ALTER TABLE PRACOWNIK ADD CONSTRAINT FK_PRACOWNIK_0
    FOREIGN KEY (PESELSZEFA) REFERENCES PRACOWNIK (PESEL) ON UPDATE NO ACTION ON DELETE NO ACTION;
GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
    ON PRACOWNIK TO SYSDBA WITH GRANT OPTION;

```

```

CREATE TABLE PRACOWNIK
(
    ID_PRACOWNIK integer NOT NULL,
    IMIE varchar(30),
    NAZWISKO varchar(50),
    NR_TELEFONU varchar(15),
    E_MAIL varchar(50),
    PESEL varchar(11),
    PESELSZEFA varchar(11),
    STANOWISKO varchar(50),
    CONSTRAINT PK_PRACOWNIK_0 PRIMARY KEY (ID_PRACOWNIK),
    CONSTRAINT FK_PRA_U UNIQUE (PESEL)
);

ALTER TABLE PRACOWNIK ADD CONSTRAINT FK_PRACOWNIK_0
    FOREIGN KEY (PESELSZEFA) REFERENCES PRACOWNIK (PESEL) ON UPDATE NO ACTION ON DELETE NO ACTION;
GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
    ON PRACOWNIK TO SYSDBA WITH GRANT OPTION;

```

```

CREATE TABLE RECENZJA
(
    ID_RECENZJA integer NOT NULL,
    RECENZJA varchar(500),
    OCENA varchar(10),
    CONSTRAINT PK_RECENZJA_0 PRIMARY KEY (ID_RECENZJA)
);

```

```

CREATE TABLE REZYSER
(
    ID_REZYSER integer NOT NULL,
    IMIE varchar(50),
    NAZWISKO varchar(50),
    NAGRODY integer,
    CONSTRAINT PK_REZYSER_0 PRIMARY KEY (ID_REZYSER)
);

```

```

CREATE TABLE WYPOZYCZENIE
(
    ID_WYPOZYCZENIE integer NOT NULL,
    ID_PRACOWNIK integer NOT NULL,
    ID_KLIENT integer NOT NULL,
    ID_EGZEMPLARZ integer NOT NULL,
    DATA_WYPOZYCZENIA date,
    DATA_ODDANIA date,
    CONSTRAINT PK_WYPOZYCZENIE_0 PRIMARY KEY (ID_WYPOZYCZENIE)
);

ALTER TABLE WYPOZYCZENIE ADD CONSTRAINT FK_WYPOZYCZENIE_0
    FOREIGN KEY (ID_PRACOWNIK) REFERENCES PRACOWNIK (ID_PRACOWNIK) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE WYPOZYCZENIE ADD CONSTRAINT FK_WYPOZYCZENIE_1
    FOREIGN KEY (ID_KLIENT) REFERENCES KLIENT (ID_KLIENT) ON UPDATE NO ACTION ON DELETE NO ACTION;
ALTER TABLE WYPOZYCZENIE ADD CONSTRAINT FK_WYPOZYCZENIE_2
    FOREIGN KEY (ID_EGZEMPLARZ) REFERENCES EGZEMPLARZ (ID_EGZEMPLARZ) ON UPDATE NO ACTION ON DELETE NO ACTION;
GRANT DELETE, INSERT, REFERENCES, SELECT, UPDATE
    ON WYPOZYCZENIE TO SYSDBA WITH GRANT OPTION;

```

```

CREATE TABLE ZAWIERA
(
    ID_ZAWIERA integer NOT NULL,
    ILOSC integer,
    CONSTRAINT PK_ZAWIERA_0 PRIMARY KEY (ID_ZAWIERA)
);

```


PRZYKŁADOWE DOPISYWANIE DANYCH DO BAZY:

```
SQL> INSERT INTO egzemplarz VALUES(16,13,1,3,5);  
SQL>
```

```
SQL> INSERT INTO film(id_film,id_rezyser,id_kategoria,id_recenzja,nazwa,opis,dlu  
gosc) VALUES(16,1,1,2,'Bekarty wojny','W okupowanej przez nazistów Francji oddzi  
ał złożony z Amerykanów żydowskiego pochodzenia planuje zamach na Hitlera.','2h3  
3min');  
SQL>
```

```
SQL> INSERT INTO klient values(16,'Mateusz','Kowalski','689232901','matkow@gmail  
.com');  
SQL>
```

```
SQL> UPDATE klient SET imie='Nowak' WHERE id_klient=16;  
SQL>
```

PRZYKŁADOWE ZAPYTANIA MODYFIKUJĄCE I USUWAJĄCE DANE Z BAZY:

```
SQL> UPDATE egzemplarz SET id_film=16 where id_egzemplarz=16;  
SQL>
```

```
SQL> UPDATE film SET id_recenzja=3 where id_film=16;  
SQL>
```

```
SQL> DELETE FROM egzemplarz where id_film=16;  
SQL>
```

```
SQL> DELETE FROM film where id_film=16;  
SQL>
```

POBIERANIE DANYCH Z SYSTEMU BAZODANOWEGO

```
SELECT r.ID_EGZEMPLARZ, r.ID_FILM, r.ID_ZAWIERA, r.ID_NOSNIK, r.ID_OPLATA
FROM EGZEMPLARZ r
```

Data					
ID_EGZEMPLARZ	ID_FILM	ID_ZAWIERA	ID_NOSNIK	ID_OPLATA	
1	2	2	2	2	14
2	3	3	3	3	13
3	4	4	4	4	12
4	5	5	5	5	11
5	6	6	6	6	10
6	7	7	7	7	9
7	8	8	8	8	8
8	9	9	9	9	7
9	10	10	10	10	6
0	11	11	11	11	5
1	12	12	12	12	4
2	13	13	13	13	3
3	14	14	14	14	2
4	15	15	1	15	1
5	16	1	1	1	15
6	17	3	4	5	6
7	1	1	1	1	1
8	19	1	1	3	1
9	20	1	1	3	1
0	21	1	1	3	2

```
SELECT r.ID_FILM, r.ID_REZYSER, r.ID_KATEGORIA, r.ID_RECENZJA, r.NAZWA, r.OPIS,
       r.DLUGOSC
FROM FILM r
```

Data					
ID_FILM	ID_REZYSER	ID_KATEGORIA	ID_RECENZJA	NAZWA	OPIS
1	1	1	1	1 Pulp Fiction	Przemoc i odkupienie w opowieści o dwóch patnych mordercach pracujących na zlecenie mafii, lonie gangstera, bokserze i parze okradającej ludzi w restauracji.
2	2	2	2	2 Peny Magazynek	Joker, mody rekrut, który pracuje jako korespondent wojenny, na własne oczy przekonuje się, czym jest wojna w Wietnamie.
3	3	3	3	3 Lot nad kukuczym gniazdem	Historia zrodzieja, szulera i chuligana, który, by uniknąć więzienia, udaje nie poczytalność. Trafia do szpitala dla umysłowo chorych, gdzie twarde rękawice rzadzi się.
4	4	4	4	4 Incepcja	Czasz, gdy technologia pozwala na wchodzenie w świat snów. Zrodziej Cobb ma za zadanie wszczepić myśl do piątego umysłu.
5	5	5	5	5 Dracula	Rumuński ksiądz, a w rzeczywistości wampir Vlad Dracula wyjeżdża do Londynu, gdzie mieszka kobieta przypominająca mu jego tragicznie utraconą ukochaną.
6	6	6	6	6 Dobry, zły i brzydki	Dzięki Zachd. Brzydki i Dobry usiłują dotrzeć do ukrytego przez wojsko skarbu. Po piątach deptać im ży.
7	7	7	7	7 Wysnione milosci	Francis i Marie są dobrymi przyjaciółmi. Pewnej nocy spotykają Nicolasa, w którym oboje się zakochują.
8	8	8	8	8 The Social Network	Historia powstania Facebooka. Komputerowy geniusz z Harvardu zakłada stronę thefacebook.com, która nieoczekiwanie bije rekordy popularności.
9	9	9	9	9 Meskie Opowiesci	Zrealizowane z zamachem, pene dynamicznej akcji kino rozrywkowe w najlepszym wydaniu. Trzy niezależne nowele filmowe nakręcone przez znakomitych reżyserów.
10	10	10	10	10 Braveheart Waleczne Serce	W XIII-wiecznej Szkocji William Wallace, po zabójstwie swojej żony, zwołuje rodaków do walki ze znieprawdzoną angielską monarchią.
11	11	11	11	11 Hugo i jego wynalazek	Mieszkający na paryskim dworcu Hugo odkrywa pozostawiony przez ojca sekret i rusza na spotkanie tajemniczej przygody.
12	12	12	12	12 Grand Budapest Hotel	Konsjerf synnego hotelu zawiera znajomość z modym pracownikiem, który zostaje jego zaufanym protegowanym.
13	13	13	13	13 Dzikosc serca	Modzi kochankowie, Sailor i Lula, uciekają przed bandą zbirów, których matka dziewczyny wynajęła do zabicia jej towarzysza.
14	14	14	14	14 Kler	Życie trzech księży ulega zmianie, kiedy ich drogi krzyżują się ponownie.
15	15	15	15	15 P'noc p'nocny zach'd	Pracownik agencji reklamowej, który omykowo za agenta sulb specjalnych USA, trafia w sam rodek afery.

```
SELECT r.ID_KATEGORIA, r.NAZWA
FROM KATEGORIA r
```

!!!		
cs	Data	
	ID_KATEGORIA	NAZWA
1	1	Gangsterski
2	2	Dramat
3	3	Psychologiczny
4	4	Surrealistyczny
5	5	Horror
6	6	Western
7	7	Melodramat
8	8	Biograficzny
9	9	Nowele filmowe
0	10	Dramat historyczny
1	11	Familijny
2	12	Komedia
3	13	Romans
4	14	Obyczajowy
5	15	Thriller

```
SELECT r.ID_NOSNIK, r.NAZWA
```

!!!		
cs	Data	
	ID_NOSNIK	NAZWA
1	1	VHS
2	2	CD
3	3	DVD
4	4	Bluray
5	5	DVD-R
6	6	DVD+R
7	7	DVD+RW
8	8	DVD-RW
9	9	DVD-RAM
0	10	CD-ROM
1	11	CD-R
2	12	CD-RW
3	13	DVD-ROM
4	14	DVD-R DL
5	15	DVD+R DL

```
SELECT r.IMIE, r.NAZWISKO, r.NR_TELEFONU, r.E_MAIL, r.ID_KLIENT
FROM KLIENT r
```

!!!					
cs	Data				
	IMIE	NAZWISKO	NR_TELEFONU	E_MAIL	ID_KLIENT
1	Bozydar	Wieczorek	792120538	BozydarWieczorek@dayrep.com	1
2	Zosia	Szymanski	796392737	ZosiaSymanska@jourrapide.com	2
3	Gracja	Pawłowska	674906162	GracjaPawlowska@teleworm.us	3
4	Stanisława	Kalinowska	663577969	StanisławaKalinowska@teleworm.us	4
5	Blanka	Pawlak	796373897		5
6	Kuba	Zawadzki	661634211	KubaZawadzki@jourrapide.com	6
7	Lechosława	Woźniak	798815351	LechosławaWozniak@armyspy.com	7
8	Tobiasz	Majewski	518739378	TobiaszMahewsju@dayrep.com	8
9	Krystian	Pawlak*	796920778	KrystianaPawlak@dayrep.com	9
0	Irena*	Tomaszewska	783397376	IrenkaTomaszewska@jourrapide.com	10
1	Radomił	Jablonski	799908071	RadomiłJablonski@rhyta.com	11
2	Władysława	Zajac	538339927		12
3	Krzysztof	Kowalski	517955123	Krzs@armyspy.com	13
4	Zosia	Wieczorek	602615084	ZosiaWieczorek@dayrep.com	14
5	Klaudiusz	Dabrowski	792466711	KlaudiuszDabrowski@dayrep.com	15
6	mati	najs	722336690	mnor@gmail.com	16

```
SELECT r.ID_OPLATA, r.CENA, r.KARA_PRZETRZYMANIE
FROM OPLATA r
```

	Data		
	ID_OPLATA	CENA	KARA_PRZETRZYMANIE
1	1	11.000000	0.500000
2	2	10.000000	0.400000
3	3	13.000000	0.600000
4	4	14.000000	0.700000
5	5	12.000000	0.400000
6	6	15.000000	0.500000
7	7	9.000000	0.300000
8	8	8.000000	0.500000
9	9	19.000000	0.600000
0	10	21.000000	0.500000
1	11	17.000000	1.000000
2	12	20.000000	1.100000
3	13	18.000000	0.600000
4	14	7.000000	0.700000
5	15	16.000000	0.600000

```
SELECT r.ID_RECENZJA, r.RECENZJA, r.OCENA
FROM RECENZJA r
```

	Data		
	ID_RECENZJA	RECENZJA	OCENA
1	1	[null]	8,4
2	2	[null]	8
3	3	[null]	8,5
4	4	[null]	8,3
5	5	[null]	7,5
6	6	[null]	8,2
7	7	[null]	7,3
8	8	[null]	7
9	9	[null]	7,2
10	10	[null]	8,1
11	11	[null]	7,0
12	12	[null]	7,8
13	13	[null]	7,6
14	14	[null]	7,3
15	15	[null]	7,5

```
SELECT r.ID_PRACOWNIK, r.IMIE, r.NAZWISKO, r.NR_TELEFONU, r.E_MAIL, r.PESEL,
r.PESELSZEFA, r.STANOWISKO
FROM PRACOWNIK r
```

	Data							
	ID_PRACOWNIK	IMIE	NAZWISKO	NR_TELEFONU	E_MAIL	PESEL	PESELSZEFA	STANOWISKO
1	1	Ela	Czarnecka	797307728	ElaCzarnecia@armyspy.com	42030753925	97103100535	Sprzedawca
2	2	Aureliusz	Symanski	602086904	AurSym@rhyta.com	77012000810	97103100535	Sprzedawca
3	3	Egidiusz	Sawicki	691957580	egisaw@rhyta.com	70120850856	97103100535	Sprzedawca
4	4	Jacek	Nowak	667372792	jacnow@teleworm.us	35052809413	97103100535	Sprzedawca
5	5	Jacek	Adamski	535320490	jacAdam@teleworm.us	69061903077	97103100535	Sprzedawca
6	6	Hipolit	Walczak	609471920	hipolitwalczak@jourrapide.com	91050984394	97103100535	Doradca Klienta
7	7	J'zeffa	Wo«niak	883722253	jozefaWozniak@jourrapide.com	65081295224	97103100535	Doradca Klienta
8	8	Alicja	Adamska	721030352	AlicjaAdamska@armyspy.com	72112408708	97103100535	Doradca Klienta
9	9	Dobromi	Adamski	888609693	DobromilAdamski@jourrapide.com	69031289792	97103100535	Doradca Klienta
10	10	Bronislaw	Chmielewski	603813438	BronslawChmielewski@rhyta.com	91032845552	97103100535	Doradca Klienta
11	11	Katarzyna	Kowalczyk	795644166	KatarzynaKowalczyk@rhyta.com	50031073483	97103100535	Sprzedawca
12	12	Gertruda	Jaworska	692733924	GertrudaJaworska@armspy.com	71040215846	97103100535	Sprzedawca
13	13	Tymon	Dudek	883591267	TymonDudek@teleworm.us	93090508059	96121002353	Zastepca Kierownika
14	14	Jan	Bucaj	733991030	JanBuc@teleworm.us	97103100535	96121002353	Kierownik
15	15	Laura	Gierek	971368232	LauGie@o2.com	96121002353	96121002353	Szef

```
SELECT r.ID_REZYSER, r.IMIE, r.NAZWISKO, r.NAGRODY
FROM REZYSER r
```

II				
ics	Data			
	ID_REZYSER	IMIE	NAZWISKO	NAGRODY
1	1	Quentin	Tarantino	[null]
2	2	Stanley	Kubrick	[null]
3	3	Milos	Forman	[null]
4	4	Christopher	Nolan	[null]
5	5	Francis	Coppola	[null]
6	6	Sergio	Leone	[null]
7	7	Xavier	Dolan	[null]
8	8	David	Fincher	[null]
9	9	Frank	Darabont	[null]
10	10	Mel	Gibson	[null]
11	11	Martin	Scorsese	[null]
12	12	Wes	Anderson	[null]
13	13	David	Lynch	[null]
14	14	Wojciech	Smarzowski	[null]
15	15	Alfred	Hitchcock	[null]

```
SELECT r.ID_ZAWIERA, r.ILOSC
FROM ZAWIERA r
```

cs		
	Data	
	ID_ZAWIERA	ILOSC
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15

```
SELECT r.ID_WYPOZYCZENIE, r.ID_PRACOWNIK, r.ID_KLIENT, r.ID_EGZEMPLARZ,
       r.DATA_WYPOZYCZENIA, r.DATA_ODDANIA
FROM WYPOZYCZENIE r
```

III						
ics	Data					
	ID_WYPOZYCZENIE	ID_PRACOWNIK	ID_KLIENT	ID_EGZEMPLARZ	DATA_WYPOZYCZENIA	DATA_ODDANIA
1	1	1	9	1	15.11.2018	30.11.2018
2	2	2	2	1	16.11.2018	[null]
3	3	3	3	1	17.11.2018	[null]
4	4	4	4	2	18.11.2018	[null]
5	5	5	5	1	20.11.2018	[null]
6	6	6	6	1	20.11.2018	[null]
7	7	7	7	1	20.11.2018	25.11.2018
8	8	8	8	3	20.11.2018	26.11.2018
9	9	9	9	4	21.11.2018	27.11.2018
10	10	10	10	5	22.11.2018	28.11.2018
11	11	11	11	6	23.11.2018	29.11.2018
12	12	12	12	7	26.11.2018	[null]
13	13	1	13	8	27.11.2018	[null]
14	14	5	14	9	28.11.2018	[null]
15	15	9	15	10	29.11.2018	[null]
16	16	1	16	2	02.12.2018	[null]

PROCEDURY:

WYPOZYCZ3 [\[open in SQL editor\]](#)

```
SET TERM ^ ;
CREATE PROCEDURE WYPOZYCZ3 (
    I_IMIE varchar(50),
    I_NAZWISKO varchar(50),
    I_NR_TELEFONU varchar(50),
    I_E_MAIL varchar(100),
    I_ID_PRACOWNIK integer,
    I_ID_EGZ integer,
    I_DATA date )
AS
declare kID int;
BEGIN
INSERT INTO klient(imie,nazwisko,nr_telefonu,e_mail) values(:i_imie,:i_nazwisko,:i_nr_telefonu,:i_e_mail);
SELECT id_klient FROM klient WHERE imie=:i_imie AND nazwisko=:i_nazwisko AND nr_telefonu=:i_nr_telefonu AND e_mail=:i_e_mail INTO :kID;
INSERT INTO wypozyczenie(id_klient,id_pracownik,id_egzemplarz,data_wypozyczenia) values(:kID,:i_id_pracownik,:i_id_egz,:i_data);
END^
SET TERM ; ^
```

Procedura do tworzenia wypożyczenia, dodaje klienta do tabeli klient oraz wypożyczenie do tabeli wypożyczenie - po wywołaniu z odpowiednimi parametrami.

EGZEM [\[open in SQL editor\]](#)

```
SET TERM ^ ;
CREATE PROCEDURE EGZEM (
    IFILM varchar(100),
    IZAWIERA integer,
    INOSNIK varchar(10),
    IOPLATA integer )
AS
declare fID int;
declare nID int;
declare oID int;
BEGIN
SELECT id_film FROM film where nazwa=:iFILM INTO :fID;
SELECT id_oplata FROM oplata where cena=:iOPLATA INTO :oID;
SELECT id_nosnik FROM nosnik where nazwa=:iNOSNIK INTO :nID;
IF (:fID is NULL) THEN exception pusty;
IF (:nID is NULL) THEN exception nosn;
IF (:oID is NULL) THEN exception op;
INSERT INTO egzemplarz(id_oplata,id_nosnik,id_zawiera,id_film)
values(:oID,:nID,:izawiera,:fID);
END^
SET TERM ; ^

GRANT EXECUTE
ON PROCEDURE EGZEM TO SYSDBA WITH GRANT OPTION;
```

Procedura tworząca egzemplarz, dodaje rekord do tabeli egzemplarz – po wywołaniu z odpowiednimi parametrami.

WYZWALACZE:

TRIG_SPRAWDZ_OCENE [\[open in SQL editor\]](#)

```
SET TERM ^ ;
CREATE TRIGGER TRIG_SPRAWDZ_OCENE FOR RECENZJA ACTIVE
BEFORE insert OR update POSITION 0
AS
BEGIN
    if (new.ocena>10) then exception name;
END^
SET TERM ; ^
```

Wyzwalacz sprawdzający czy kolumna ocena nie jest większa od 10 - przy dodawaniu rekordu do tabeli recenzja

```
SET TERM ^ ;
CREATE TRIGGER TRIG_SPRAWDZ_STANOWISKO FOR PRACOWNIK ACTIVE
AFTER delete POSITION 0
AS
BEGIN
    if (old.stanowisko='Szef') then exception stanowisko;
END^
SET TERM ; ^
```

Wyzwalacz sprawdzający czy chcemy usunąć szefa - przy usuwaniu rekordu z tabeli pracownik

```
1 SET TERM ^ ;
2 CREATE TRIGGER TRIG_SPRAWDZ_KATEGORIE FOR KATEGORIA ACTIVE
3 BEFORE insert POSITION 0
4 AS
5 DECLARE
6 istnieje1 int;
7 BEGIN
8 SELECT COUNT(nazwa) FROM KATEGORIA where nazwa=new.nazwa INTO istnieje1;
9 if (istnieje1>0) then exception istnieje;
10 END^
11 SET TERM ; ^
```

Wyzwalacz sprawdzający czy kategoria, którą chcemy dodać już istnieje

ZAPYTANIA SELECT

```
SQL> SELECT p.id_pracownik, w.id_wypozyczenie FROM wypozyczenie AS w INNER JOIN
pracownik AS p ON w.id_pracownik=p.id_pracownik;
```

ID_PRACOWNIK	ID_WYPOZYCZENIE
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
1	13
5	14
9	15
1	16

```
SQL>
```

```
SQL> SELECT f.nazwa,k.nazwa FROM film as f LEFT OUTER JOIN kategoria as k ON k.i
d_kategoria=f.id_kategoria;
```

NAZWA	NAZWA
Pulp Fiction	Gangsterski
Pełny Magazynek	Dramat
Lot nad kukułczym gniazdem	Psychologiczny
Incepcja	Surrealistyczny
Dracula	Horror
Dobry, zły i brzydki	Western
Wysnione miłości	Melodramat
The Social Network	Biograficzny
Meskie Opowieści	Nowele filmowe
Braveheart Waleczne Serce	Dramat historyczny
Hugo i jego wynalazek	Familijny
Grand Budapest Hotel	Komedia
Dzikosc serca	Romans
Kler	Obyczajowy
Północ północny zachód	Thriller


```
SQL> SELECT nazwa FROM film WHERE id_film>10 GROUP BY nazwa;
```

NAZWA

```
=====
Dzikosc serca
Grand Budapest Hotel
Hugo i jego wynalazek
Kler
Północ północny zachód
```

```
SQL> SELECT imie,nazwisko,stanowisko FROM pracownik WHERE stanowisko='Sprzedawca' ORDER BY imie ASC;
```

IMIE	NAZWISKO	STANOWISKO
Aureliusz	Symanski	Sprzedawca
Egidiusz	Sawicki	Sprzedawca
Ela	Czarnecka	Sprzedawca
Gertruda	Jaworska	Sprzedawca
Jacek	Nowak	Sprzedawca
Jacek	Adamski	Sprzedawca
Katarzyna	Kowalczyk	Sprzedawca

```
SQL>
```

(SKORELOWANE) – czyli podzapytanie bezpośrednio powiązane z zapytaniem nadrzędnym

```
SQL> SELECT p.id_pracownik,(SELECT COUNT(id_klient) FROM wypozyczenie as w WHERE p.id_pracownik=w.id_pracownik) as LiczbaKlientow FROM pracownik as p;
```

ID_PRACOWNIK	LICZBAKLIENTOW
1	3
2	1
3	1
4	1
5	2
6	1
7	1
8	1
9	2
10	1
11	1
12	1
13	0
14	0
15	0

```
SQL> SELECT count(id_wypozyczenie),data_wypozyczenia,data_oddania FROM wypozyczenie GROUP BY data_oddania,data_wypozyczenia HAVING data_oddania is null;
```

COUNT	DATA_WYPOZYCZENIA	DATA_ODDANIA
1	2018-11-16	<null>
1	2018-11-17	<null>
1	2018-11-18	<null>
2	2018-11-20	<null>
1	2018-11-26	<null>
1	2018-11-27	<null>
1	2018-11-28	<null>
1	2018-11-29	<null>
1	2018-12-02	<null>

```
SQL>
```

```
SQL> SELECT nazwa FROM film WHERE id_kategoria IN ('1','3','4');
```

NAZWA

```
=====
Pulp Fiction
Lot nad kukułczym gniazdem
Incepcja
```

```
SQL>
```

```
SQL> SELECT nazwa FROM film WHERE id_film=ANY(SELECT id_film FROM egzemplarz WHERE id_oplata=6);
```

NAZWA

```
=====
Lot nad kukułczym gniazdem
Braveheart Waleczne Serce
```

```
SQL> SELECT nazwisko FROM rezyser WHERE id_rezyser=ALL(SELECT id_rezyser FROM film WHERE id_film=3);
NAZWISKO
=====
Forman
SQL>
```

```
SQL> SELECT nazwa FROM nosnik WHERE EXISTS (SELECT id_nosnik FROM egzemplarz WHERE egzemplarz.id_nosnik=nosnik.id_nosnik AND id_egzemplarz=3);
NAZWA
=====
DUD
SQL>
```

```
SQL> SELECT imie, nazwisko FROM klient WHERE imie LIKE 'S%a';
IMIE                                NAZWISKO
=====
Stanislawa                        Kalinowska
SQL>
```

```
SQL> SELECT p.imie,p.nazwisko,w.id_wypozyczenie FROM pracownik as p INNER JOIN wypozyczenie as w ON p.id_pracownik=w.id_pracownik WHERE p.imie LIKE 'a%';
IMIE                                NAZWISKO                                ID_WYPOZYCZENIE
=====
Ela                                Czarnecka                                1
Ela                                Czarnecka                                13
Ela                                Czarnecka                                16
Józefa                            Woźniak                                   7
Alicja                            Adamska                                   8
Katarzyna                        Kowalczyk                                11
Gertruda                         Jaworska                                  12
SQL>
```

```
SQL> SELECT COUNT(id_egzemplarz) FROM egzemplarz WHERE id_nosnik=3;
COUNT
=====
4
SQL>
```

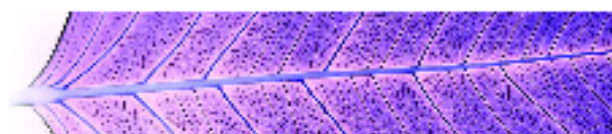
```
SQL> SELECT film.id_film, oplata.cena FROM film INNER JOIN egzemplarz ON egzemplarz.id_film=film.id_film INNER JOIN oplata ON egzemplarz.id_oplata=opлата.id_oplata WHERE oplata.cena>10;
ID_FILM                                CENA
=====
15                                11.000000000000000
1                                11.000000000000000
1                                11.000000000000000
1                                11.000000000000000
13                                13.000000000000000
12                                14.000000000000000
11                                12.000000000000000
10                                15.000000000000000
3                                15.000000000000000
7                                19.000000000000000
6                                21.000000000000000
5                                17.000000000000000
4                                20.000000000000000
3                                18.000000000000000
1                                16.000000000000000
SQL>
```

```
SQL> SELECT id_egzemplarz, (SELECT COUNT(id_wypozyczenie) FROM wypozyczenie WHERE egzemplarz.id_egzemplarz=wypozyczenie.id_egzemplarz) AS iloscWypozyczen FROM egzemplarz;
ID_EGZEMPLARZ                                ILOSCWYPOZYCZEN
=====
2                                2
3                                1
4                                1
5                                1
7                                1
8                                1
9                                1
10                               1
11                               0
12                               0
13                               0
14                               0
15                               0
16                               0
17                               0
18                               6
19                               0
20                               0
21                               0
SQL>
```

RAPORTY

Pracownicy

Wypożyczalnia DVD



IMIE	NAZWISKO	NR_TELEFONU	E_MAIL
Sprzedawca			
Ela	Czarnecka	797307728	ElaCzarnecka@armyspy.
Aureliusz	Symanski	602086904	AurSym@rhyta.com
Egidiusz	Sawicki	691957580	egisaw@rhyta.com
Jacek	Nowak	667372792	ja.cnw@teleworm.us
Jacek	Adamski	535320490	ja.cAdam@teleworm.us

Doradca Klienta			
Hipolit	Walczak	609471920	hipolitwalczak@jourrapid
Józefa	Woźniak	883722253	jzefaWozniak@jourrapid
Alicja	Adamska	721030352	AlicjaAdamska@armyspy
Dobromi	Adamski	888609693	DobromilAdamski@jourra
Bronisław	Chmielewski	603813438	BronslawChmielewski@r
Katarzyna	Kowalczyk	795644166	KatarzynaKowalczyk@rh

Zastępca Kierownika			
Gertruda	Jaworska	692733924	GertrudaJaworska@arms

Kierownik			
Tymon	Dudek	883591267	TymonDudek@teleworm.
Jan	Bucaj	733991030	JanBuc@teleworm.us

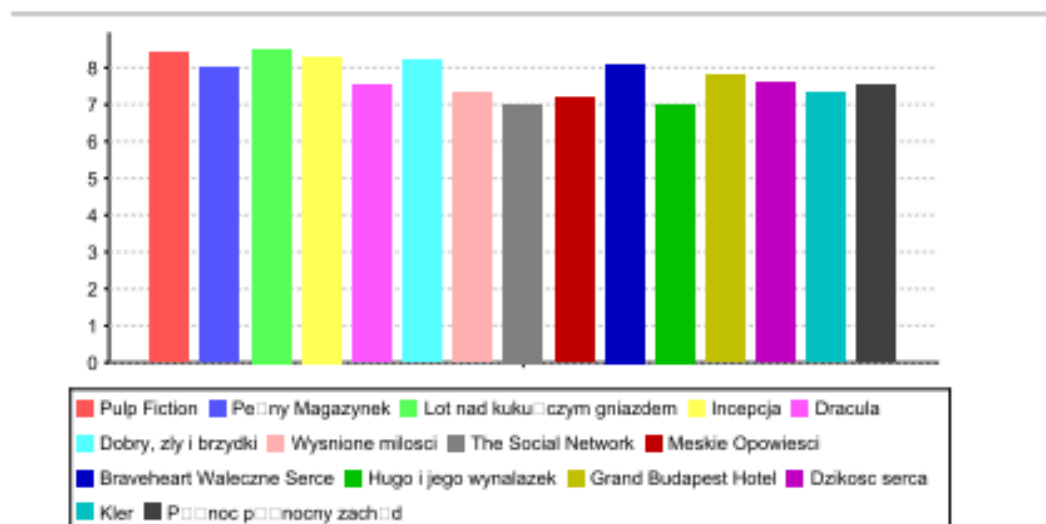
Szef			
Laura	Gierek	971368232	LauGie@o2.com

Oceny Filmów

Wypożyczalnia DVD



NAZWA	DLUGOSC	OCENA
Pulp Fiction	2h34min	8.4
Peny Magazynek	1h56min	8
Lot nad kukuczym	2h13min	8.5
Incepcja	2h28min	8.3
Dracula	2h8min	7.5
Dobry, zły i brzydki	2h59min	8.2
Wysnione miłości	1h35min	7.3
The Social Network	2h0min	7
Meskie Opowieści	1h30min	7.2
Braveheart Waleczne	2h57min	8.1
Hugo i jego wynalazek	2h7min	7.0
Grand Budapest Hotel	1h40min	7.8
Dzikosc serca	2h4min	7.6
Kler	2h13min	7.3
Pnocy pnoćny	2h16min	7.5



Formularz Wypożyczenia

Wypożyczalnia DVD

Imie i Nazwisko:	Krystian Pawlak*
Nazwa filmu:	Pulp Fiction
Nosnik:	VHS
Cena:	11.0
Kara za przetrzymanie: (1 dzien)	0.5
Data Wypożyczenia:	15.11.18

Wypożyczenia

Wypożyczalnia DVD

IMIE	NAZWISKO	DATA_WYPOZYCZENIA	DATA_ODDANIA
------	----------	-------------------	--------------

Wysnione milosci

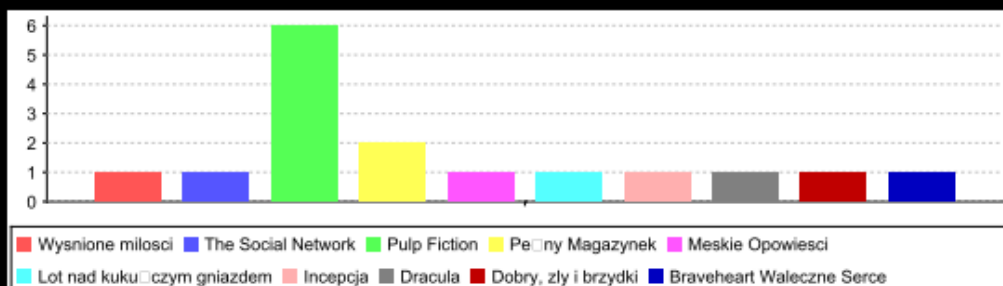
Wadysawa	Zajc	26.11.18 00:00	null
----------	------	----------------	------

The Social Network

Krzysztof	Kowalski	27.11.18 00:00	null
-----------	----------	----------------	------

Pulp Fiction

Krystian	Pawlak*	15.11.18 00:00	30.11.18 00:00
Zosia	Szymanski	16.11.18 00:00	null
Gracja	Pawowska	17.11.18 00:00	null
Blanka	Pawlak	20.11.18 00:00	null
Kuba	Zawadzki	20.11.18 00:00	null
Lechosawa	Woniak	20.11.18 00:00	25.11.18 00:00



niedziela 13 stycznia 2019

Page 1 of 3

IMIE	NAZWISKO	DATA_WYPOZYCZENIA	DATA_ODDANIA
------	----------	-------------------	--------------

Peny Magazynek

Stanisława	Kalinowska	18.11.18 00:00	null
mati	najs	02.12.18 00:00	null

Meskie Opowiesci

Zosia	Wieczorek	28.11.18 00:00	null
-------	-----------	----------------	------

Lot nad kukułczym gniazdem

Tobiasz	Majewski	20.11.18 00:00	26.11.18 00:00
---------	----------	----------------	----------------

Incepcja

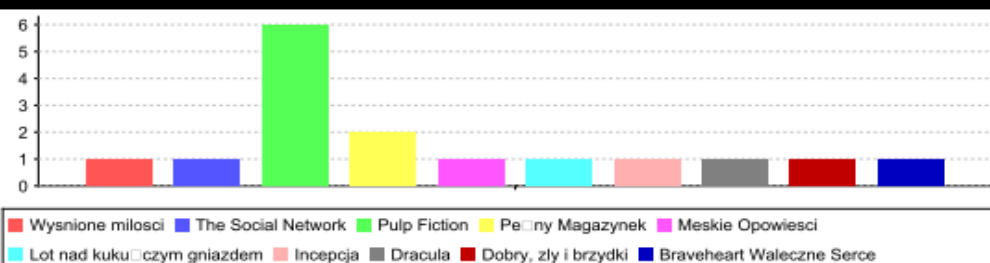
Krystian	Pawlak*	21.11.18 00:00	27.11.18 00:00
----------	---------	----------------	----------------

Dracula

Irena*	Tomaszewska	22.11.18 00:00	28.11.18 00:00
--------	-------------	----------------	----------------

Dobry, zły i brzydki

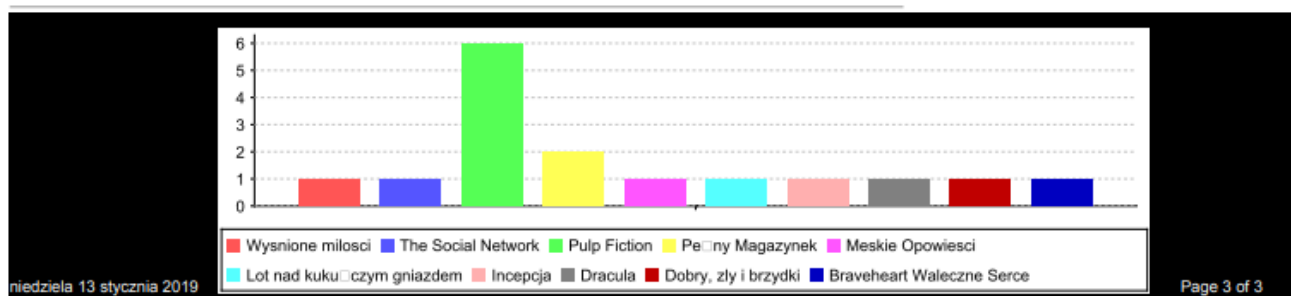
Radomi	Jabonski	23.11.18 00:00	29.11.18 00:00
--------	----------	----------------	----------------



niedziela 13 stycznia 2019

Page 2 of 3

IMIE	NAZWISKO	DATA_WYPOZYCZENIA	DATA_ODDANIA
Braveheart Waleczne Serce			
Klaudiusz	Dabrowski	29.11.18 00:00	null



APLIKACJA:

Raport Ocena Filmów

Domyslna

Generuj

Raport Wypozyczenia

Domyslna

Generuj

Raport Pracownicy

Domyslna

Generuj

Formularz wypożyczenia

Imie i Nazwisko

Nazwa Filmu

Nosnik

Cena

Kara za przetrzymanie

Data Wypozyczenia

Generuj

Mateusz Kucaj GRUPA 4 110953

Dane do połączenia z bazą danych:

```
String url = "jdbc:firebirdsql://localhost:3050/C:/Users/Mati/Desktop/BAZY PROJEKT/db/DB.FDB";
String user = "SYSDBA";
String pass = "masterkey";
```

Kod do obsługi przycisku „GENERUJ”

```
public void actionPerformed(ActionEvent e) {
    try {
        Class.forName("org.firebirdsql.jdbc.FBDriver"); //sterowniki do jdbc (odpowiednie dla firebird)
        Connection con = DriverManager.getConnection(url, user, pass); //połączenie z baza
        JasperDesign jd = JRXmlLoader
            .load("C:/Users/Mati/JaspersoftWorkspace/MyReports/Leaf_Grey_1.jrxml"); //ładowanie raportu
        JasperReport jr = JasperCompileManager.compileReport(jd); //skompilowanie raportu
        HashMap p = new HashMap();
        if (comboBox.getSelectedItem().toString()
            .equals("Sortuj alfabetycznie filmy")) //jeżeli z listy rozwijanej zostało wybrane "Sortuj alfabetycznie filmy"
            p.put("SORT", "nazwa"); //dodanie do HashMapy 'nazwa' z kluczem 'SORT'
        else if (comboBox.getSelectedItem().toString()
            .equals("Sortuj po ocenie rosnąco")) //jeżeli z listy rozwijanej zostało wybrane "Sortuj po ocenie rosnąco"
            p.put("SORT", "ocena"); //dodanie do HashMapy 'ocena' z kluczem 'SORT'
        else
            p.put("SORT", "dlugosc"); //domyślnie sortuje po długości filmu
        JasperPrint jp = JasperFillManager.fillReport(jr, p, con); //wypełnienie raportu parametrami
        JasperViewer viewer = new JasperViewer(jp, false); //wyświetlenie raportu w programie JasperViewer
        viewer.setTitle("Wypożyczalnia - filmy");
        viewer.setVisible(true);
    } catch (SQLException ex) {
        if (ex.getErrorCode() == 1045)
            System.out.println("błąd SQL - " + "Błąd logowania");
        else
            System.out.println("błąd SQL - " + ex);
    } catch (JRException ex) {
        System.out.println(ex);
    } catch (ClassNotFoundException e1) {
        e1.printStackTrace();
    }
}
```