

Dokumentacja Projektu – Kinowej Bazy Danych

Wykonał: **Bartłomiej Siwak**

Prowadzący: **mgr inż. Józef Woźniak**

Grupa: **WCY20IY5S1**

1. Analiza biznesowa projektowanej rzeczywistości

Prezentowana baza danych przystosowana jest do obsługi pojedynczego kina, w którym znajdują się sale z określoną liczbą miejsc, w których będą miały miejsce seanse.

Klienci mogą dokonywać zakupów, w ramach jednego zakupu można kupić jeden lub wiele biletów.

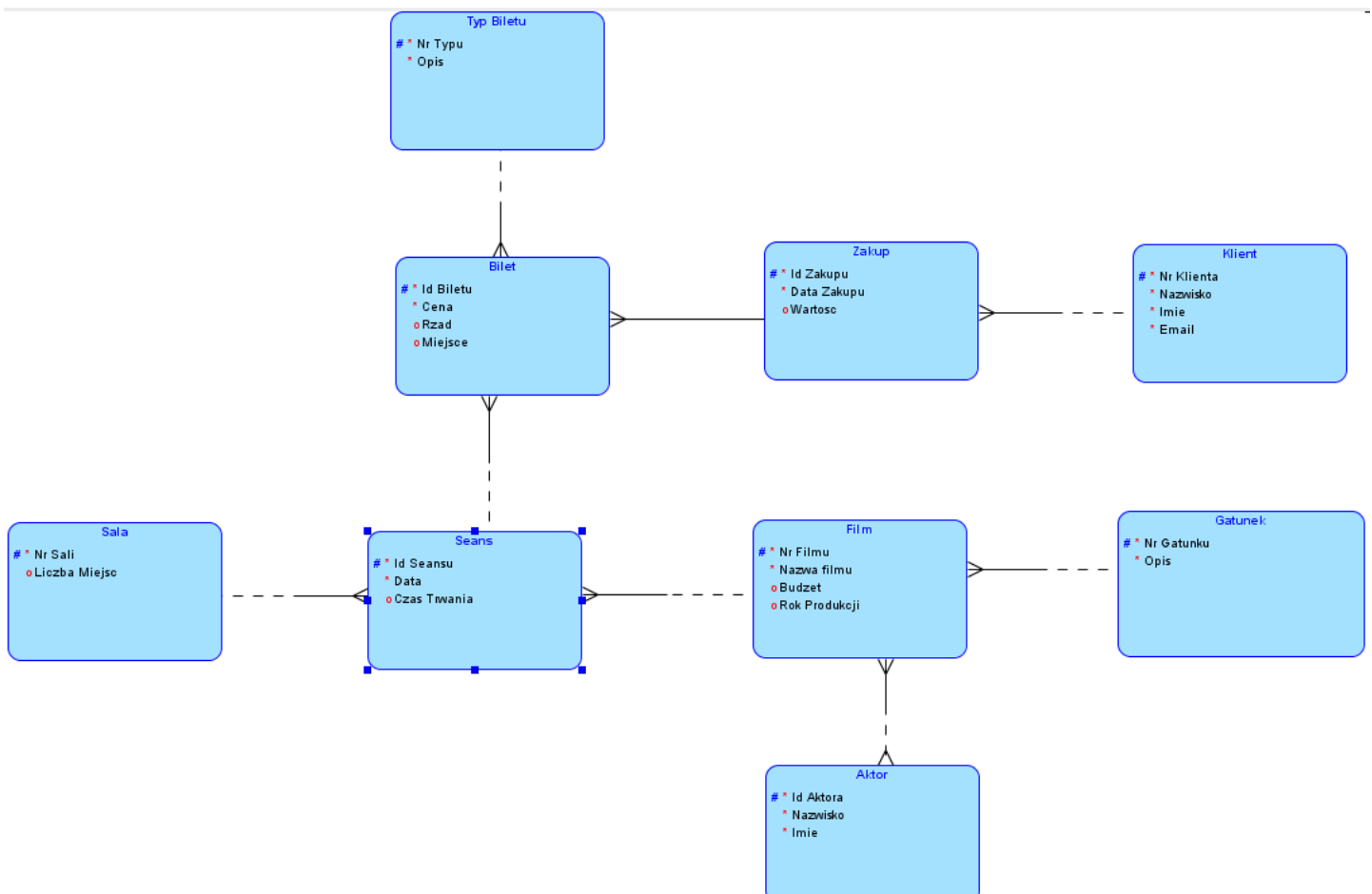
Bilety sprzedawane są na konkretny seans, na którym odtwarzany jest konkretny film.

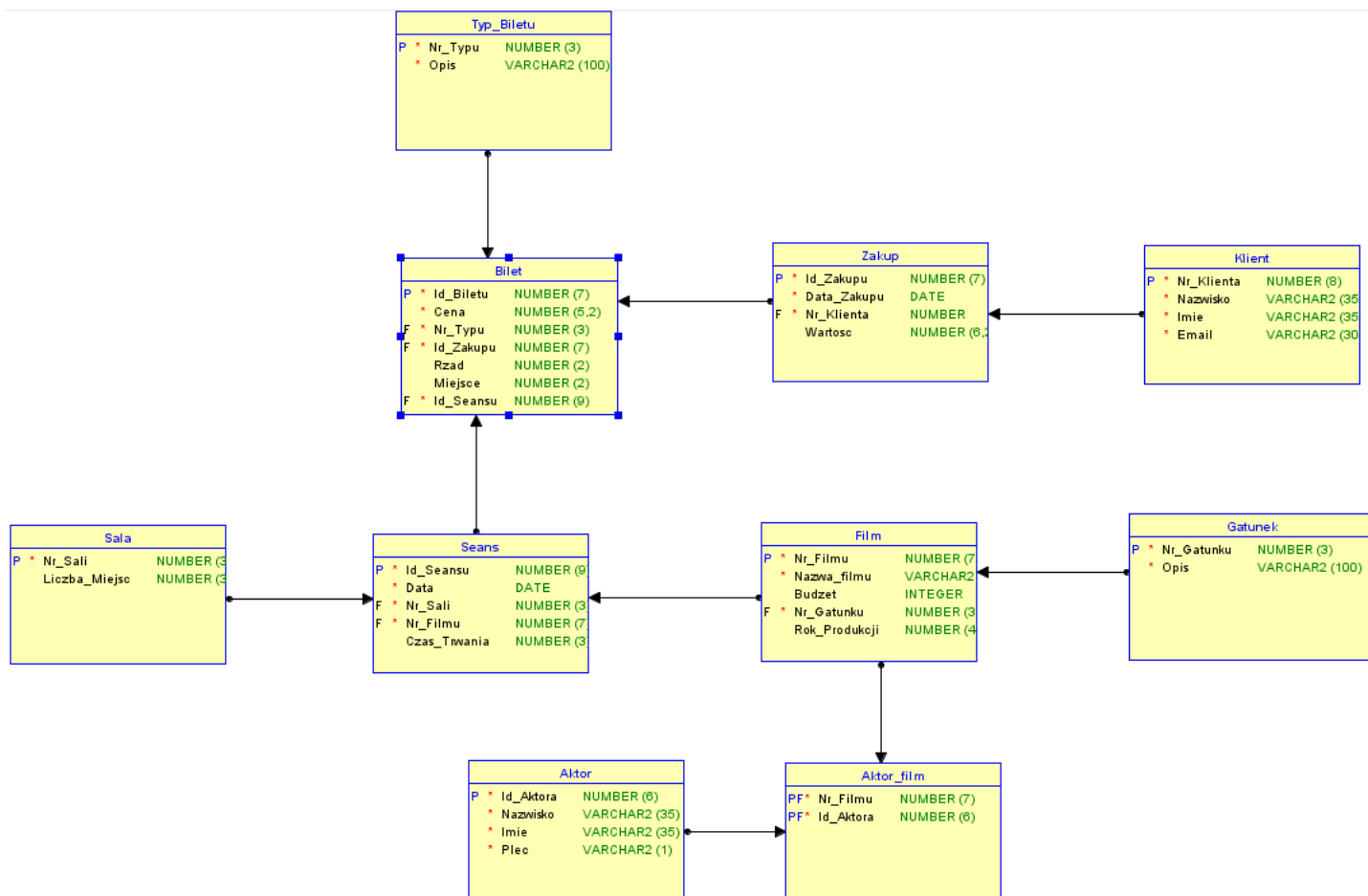
W bazie przechowywane są również informacje o aktorach oraz ich powiązaniach z konkretnymi filmami. Jeden aktor może grać w wielu filmach, w jednym filmie może grać wielu aktorów, co oznacza, że jest to relacja M:N, w związku z czym, przy generacji z modelu logicznego do relacyjnego, otrzymujemy tabelę intersekcji Film_Aktor.

Każdy film musi mieć dokładnie jeden określony gatunek.

Każdy bilet musi mieć dokładnie jeden określony typ biletu (normalny, ulgowy, itp...)

2. Model logiczny i relacyjny bazy danych





3. Oprogramowanie tworzące bazę danych

Oracle SQL Developer;

Oracle Data Modeler.

4. Skrypty wdrożeniowe instalujące i deinstalujące zrealizowany projekt

Skrypt instalujący (PROJEKT_INSTALACJA.sql):

```

CREATE TABLE aktor (
    id_aktora NUMBER(6) NOT NULL,
    nazwisko VARCHAR2(35) NOT NULL,
    imie VARCHAR2(35) NOT NULL,
    plec VARCHAR2(1) NOT NULL
);

ALTER TABLE aktor ADD CONSTRAINT aktor_pk PRIMARY KEY ( id_aktora );

CREATE TABLE aktor_film (
    nr_filmu NUMBER(7) NOT NULL,
    id_aktora NUMBER(6) NOT NULL
);

ALTER TABLE aktor_film ADD CONSTRAINT relation_18_pk PRIMARY KEY (
    nr_filmu,
    id_aktora );

CREATE TABLE bilet (

```

```
id_biletu NUMBER(7) NOT NULL,  
cena    NUMBER(5, 2) NOT NULL,  
nr_tytu  NUMBER(3) NOT NULL,  
id_zakupu NUMBER(7) NOT NULL,  
rzad    NUMBER(2),  
miejsce NUMBER(2),  
id_seansu NUMBER(9) NOT NULL  
);
```

```
ALTER TABLE bilet ADD CONSTRAINT bilet_pk PRIMARY KEY ( id_biletu );
```

```
CREATE TABLE film (  
    nr_filmu    NUMBER(7) NOT NULL,  
    nazwa_filmu VARCHAR2(35) NOT NULL,  
    budzet     INTEGER,  
    nr_gatunku  NUMBER(3) NOT NULL,  
    rok_produkcji NUMBER(4)  
);
```

```
ALTER TABLE film ADD CONSTRAINT film_pk PRIMARY KEY ( nr_filmu );
```

```
CREATE TABLE gatunek (  
    nr_gatunku NUMBER(3) NOT NULL,  
    opis      VARCHAR2(100) NOT NULL  
);
```

```
ALTER TABLE gatunek ADD CONSTRAINT gatunek_pk PRIMARY KEY ( nr_gatunku );
```

```
CREATE TABLE klient (  
    nr_klienta NUMBER(8) NOT NULL,  
    nazwisko   VARCHAR2(35) NOT NULL,  
    imie       VARCHAR2(35) NOT NULL,  
    email      VARCHAR2(30) NOT NULL  
);
```

```
ALTER TABLE klient ADD CONSTRAINT klient_pk PRIMARY KEY ( nr_klienta );
```

```
CREATE TABLE sala (  
    nr_Sali    NUMBER(3) NOT NULL,  
    liczba_miejsc NUMBER(3)  
);
```

```
ALTER TABLE sala ADD CONSTRAINT sala_pk PRIMARY KEY ( nr_Sali );
```

```
CREATE TABLE seans (  
    id_seansu  NUMBER(9) NOT NULL,  
    data       DATE NOT NULL,  
    nr_Sali    NUMBER(3) NOT NULL,  
    nr_filmu   NUMBER(7) NOT NULL,  
    czas_trwania NUMBER(3)  
);
```

```

COMMENT ON COLUMN seans.czas_trwania IS
    'Czas trwania seansu określony w minutach.';

ALTER TABLE seans ADD CONSTRAINT seans_pk PRIMARY KEY ( id_seansu );

CREATE TABLE typ_biletu (
    nr_typu NUMBER(3) NOT NULL,
    opis VARCHAR2(100) NOT NULL
);

ALTER TABLE typ_biletu ADD CONSTRAINT typ_biletu_pk PRIMARY KEY ( nr_typu );

CREATE TABLE zakup (
    id_zakupu NUMBER(7) NOT NULL,
    data_zakupu DATE NOT NULL,
    nr_klienta NUMBER NOT NULL,
    wartosc NUMBER(6, 2)
);

ALTER TABLE zakup ADD CONSTRAINT zakup_pk PRIMARY KEY ( id_zakupu );

ALTER TABLE aktor_film
    ADD CONSTRAINT aktor_fk FOREIGN KEY ( id_aktora )
    REFERENCES aktor ( id_aktora );

ALTER TABLE aktor_film
    ADD CONSTRAINT film_fk FOREIGN KEY ( nr_filmu )
    REFERENCES film ( nr_filmu );

ALTER TABLE seans
    ADD CONSTRAINT film_fkv1 FOREIGN KEY ( nr_filmu )
    REFERENCES film ( nr_filmu );

ALTER TABLE film
    ADD CONSTRAINT gatunek_fk FOREIGN KEY ( nr_gatunku )
    REFERENCES gatunek ( nr_gatunku );

ALTER TABLE zakup
    ADD CONSTRAINT klient_fk FOREIGN KEY ( nr_klienta )
    REFERENCES klient ( nr_klienta );

ALTER TABLE seans
    ADD CONSTRAINT sala_fk FOREIGN KEY ( nr_Sali )
    REFERENCES sala ( nr_Sali );

ALTER TABLE bilet
    ADD CONSTRAINT seans_fk FOREIGN KEY ( id_seansu )
    REFERENCES seans ( id_seansu );

ALTER TABLE bilet
    ADD CONSTRAINT typ_biletu_fk FOREIGN KEY ( nr_typu )
    REFERENCES typ_biletu ( nr_typu );

```

```

ALTER TABLE bilet
  ADD CONSTRAINT zakup_fk FOREIGN KEY ( id_zakupu )
    REFERENCES zakup ( id_zakupu );

create sequence gatunek_seq;

insert into gatunek
values(gatunek_seq.nextval, 'Fabularny');

insert into gatunek
values(gatunek_seq.nextval, 'Kryminalny');

insert into gatunek
values(gatunek_seq.nextval, 'Dramat');

insert into gatunek
values(gatunek_seq.nextval, 'Komedia');

insert into gatunek
values(gatunek_seq.nextval, 'Horror');

--////////////////////////////////////

create sequence film_seq
start with 1000000
increment by 1
nocycle;

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Pewnego razu na Dzikim Zachodzie', 1, 1000000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, '2001: Odyseja kosmiczna', 3, 75000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Czas Apokalipsy', 2, 62500000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Lot nad kukułczym gniazdem', 1, 55500000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Ojciec chrzestny', 1, 25000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Siedmiu samurajów', 1, 37500000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Harakiri', 3, 49000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)

```

```

values(film_seq.nextval, 'Pulp Fiction', 3, 48000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Ojciec chrzestny II', 2, 32000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Dobry, zły i brzydki', 4, 71000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Psychoza', 4, 83000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Dwunastu gniewnych ludzi', 4, 97000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Mechaniczna pomarańcza', 4, 8000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Męczeństwo Joanny dArc', 5, 62000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Dawno temu w Ameryce', 5, 25000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Obywatel Kane', 2, 30000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Rashomon', 2, 47000000);

insert into film (nr_filmu, nazwa_filmu, nr_gatunku, budzet)
values(film_seq.nextval, 'Łowca androidów', 1, 33000000);

--////////////////////////////////////

create sequence sala_seq;

insert into sala
values(sala_seq.nextval, 90);
insert into sala
values(sala_seq.nextval, 80);
insert into sala
values(sala_seq.nextval, 85);
insert into sala
values(sala_seq.nextval, 70);
insert into sala
values(sala_seq.nextval, 95);
insert into sala
values(sala_seq.nextval, 100);
insert into sala
values(sala_seq.nextval, 85);

--////////////////////////////////////

```

```
alter table aktor
add constraint CHK_PLEC check (plec in ('M', 'K'));

create sequence aktor_seq;

insert into aktor
values(aktor_seq.nextval, 'Morgan', 'Freeman', 'M');

insert into aktor
values(aktor_seq.nextval, 'Bradd', 'Pitt', 'M');

insert into aktor
values(aktor_seq.nextval, 'Leonardo', 'DiCaprio', 'M');

insert into aktor
values(aktor_seq.nextval, 'Robert', 'De Niro', 'M');

insert into aktor
values(aktor_seq.nextval, 'Matt', 'Damon', 'M');

insert into aktor
values(aktor_seq.nextval, 'Michael', 'Caine', 'M');

insert into aktor
values(aktor_seq.nextval, 'Christian', 'Bale', 'M');

insert into aktor
values(aktor_seq.nextval, 'Tom', 'Hanks', 'M');

insert into aktor
values(aktor_seq.nextval, 'Gary', 'Oldman', 'M');

insert into aktor
values(aktor_seq.nextval, 'Al', 'Pacino', 'M');

insert into aktor
values(aktor_seq.nextval, 'Natalie', 'Portman', 'K');

insert into aktor
values(aktor_seq.nextval, 'Anne', 'Hathaway', 'K');

insert into aktor
values(aktor_seq.nextval, 'Talía', 'Shire', 'K');

insert into aktor
values(aktor_seq.nextval, 'Jennifer', 'Lawrence', 'K');

insert into aktor
values(aktor_seq.nextval, 'Julianne', 'Moore', 'K');

insert into aktor
```

```
values(aktor_seq.nextval, 'Diane', 'Keaton', 'K');
```

```
insert into aktor  
values(aktor_seq.nextval, 'Lea', 'Seydoux', 'K');
```

```
insert into aktor  
values(aktor_seq.nextval, 'Robin', 'Wright', 'K');
```

```
--////////////////////////////////////
```

```
insert into aktor_film  
values(1000002, 1);
```

```
insert into aktor_film  
values(1000002, 7);
```

```
insert into aktor_film  
values(1000002, 9);
```

```
insert into aktor_film  
values(1000004, 5);
```

```
insert into aktor_film  
values(1000011, 2);
```

```
insert into aktor_film  
values(1000013, 2);
```

```
insert into aktor_film  
values(1000015, 2);
```

```
insert into aktor_film  
values(1000016, 3);
```

```
insert into aktor_film  
values(1000002, 4);
```

```
insert into aktor_film  
values(1000010, 10);
```

```
insert into aktor_film  
values(1000011, 10);
```

```
insert into aktor_film  
values(1000009, 8);
```

```
insert into aktor_film  
values(1000009, 3);
```



```
insert into aktor_film  
values(1000008, 5);
```

```
insert into aktor_film  
values(1000007, 5);
```

```
insert into aktor_film  
values(1000005, 9);
```

```
insert into aktor_film  
values(1000016, 9);
```

```
insert into aktor_film  
values(1000002, 11);
```

```
insert into aktor_film  
values(1000003, 12);
```

```
insert into aktor_film  
values(1000005, 13);
```

```
insert into aktor_film  
values(1000007, 12);
```

```
insert into aktor_film  
values(1000013, 18);
```

```
insert into aktor_film  
values(1000014, 17);
```

```
insert into aktor_film  
values(1000016, 15);
```

```
insert into aktor_film  
values(1000015, 18);
```

```
--////////////////////////////////////
```

```
insert into typ_biletu  
values(1, 'Normalny');
```

```
insert into typ_biletu  
values(2, 'Ulgowy');
```

```
insert into typ_biletu  
values(3, 'Student');
```

```
insert into typ_biletu  
values(4, 'Senior');
```

```
insert into typ_biletu  
values(5, 'Weteran');
```

```
insert into typ_biletu  
values(6, 'Rodzinny');
```

```
--////////////////////////////////////  
create sequence klient_seq;
```

```
insert into klient  
values(klient_seq.nextval, 'Abacki', 'Andrzej', 'a.abacki@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Nowak', 'Halina', 'h.nowak@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Kowalski', 'Mikołaj', 'm.kowalski@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Siwak', 'Bartek', 'b.siwak@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Biernat', 'Kamil', 'k.biernat@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Ciach', 'Malgorzata', 'm.ciach@gmail.com');
```

```
insert into klient  
values(klient_seq.nextval, 'Balcerzak', 'Emil', 'e.balcerzak@gmail.com');
```

```
--////////////////////////////////////
```

```
create sequence seans_seq;
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)  
values(seans_seq.nextval, sysdate - 4, 1, 1000002);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)  
values(seans_seq.nextval, sysdate - 4, 1, 1000004);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)  
values(seans_seq.nextval, sysdate - 3, 1, 1000004);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate - 1, 3, 1000005);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate, 5, 1000014);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate + 4, 3, 1000015);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate + 2, 7, 1000015);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate + 2, 6, 1000016);
```

```
insert into seans(id_seansu, data, nr_Sali, nr_filmu)
values(seans_seq.nextval, sysdate + 10, 7, 1000017);
```

```
--////////////////////////////////////
```

```
create sequence zakup_seq;
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate - 10, 1);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate - 7, 1);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate - 5, 2);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate - 2, 3);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate - 1, 4);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate, 6);
```

```
insert into zakup (id_zakup, data_zakup, nr_klienta)
values(zakup_seq.nextval, sysdate, 6);
```

```
--////////////////////////////////////
```

```
create sequence bilet_seq;
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 25.99, 1, 1, 1);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 26.99, 1, 4, 2);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 24.99, 1, 3, 2);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 12.99, 2, 4, 4);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 18.99, 3, 3, 7);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 18.99, 3, 3, 1);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 17.99, 3, 2, 7);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 13.99, 4, 6, 6);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 17.99, 6, 5, 6);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 17.99, 6, 5, 6);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 17.99, 6, 5, 4);
```

```
insert into bilet(id_biletu, cena, nr_typu, id_zakupu, id_seansu)
values(bilet_seq.nextval, 17.99, 6, 7, 3);
```

```
--////////////////////////////////////
```

```
--Bierzemy pod uwage filmy, na ktore sa zaplanowane seanse, ale na ktore mogly
sie nie sprzedac zadne bilety
```

```
create or replace view Przychod_filmu_ostatnie_trzy_miesiace
as
select nazwa_filmu, sum(nvl(cena, 0)) przychod
from film f join seans s on f.nr_filmu = s.nr_filmu
left join bilet b on b.Id_Seansu = s.Id_seansu
where s.data between sysdate - 90 and sysdate
group by f.nr_filmu, nazwa_filmu;
```

```
create or replace view Ilosc_Biletow_Na_Gatunek
as
```

```

select opis „Gatunek”, count(*) „Ilość zakupionych biletów”
from film f join gatunek g on f.nr_gatunku = g.nr_gatunku
      join seans s on f.nr_filmu = s.nr_filmu
      join bilet b on b.id_seansu = s.id_seansu
group by f.nr_gatunku, opis
having count(*) > 0.8 * (select avg(count(*))
      from film f join gatunek g on f.nr_gatunku = g.nr_gatunku
      join seans s on f.nr_filmu = s.nr_filmu
      join bilet b on b.id_seansu = s.id_seansu
      group by f.nr_gatunku, opis);

create or replace view Najpopularniejszy_aktor_oraz_aktorka
as
select nazwisko || ' ' || imie „Aktor”, count(*) „Ilość zakupionych biletów”
from aktor a join aktor_film af on a.id_aktora = af.id_aktora
      join film f on af.nr_filmu = f.nr_filmu
      join seans s on s.nr_filmu = f.nr_filmu
      join bilet b on b.id_seansu = s.id_seansu
where plec = 'M'
group by a.id_aktora, nazwisko, imie
having count(*) = (select max(count(*))
      from aktor a1 join aktor_film af1 on a1.id_aktora = af1.id_aktora
      join film f1 on af1.nr_filmu = f1.nr_filmu
      join seans s1 on s1.nr_filmu = f1.nr_filmu
      join bilet b1 on b1.id_seansu = s1.id_seansu
      where plec = 'M'
      group by a1.id_aktora, nazwisko, imie)

union
select nazwisko || ' ' || imie „Aktor”, count(*) „Ilość zakupionych biletów”
from aktor a join aktor_film af on a.id_aktora = af.id_aktora
      join film f on af.nr_filmu = f.nr_filmu
      join seans s on s.nr_filmu = f.nr_filmu
      join bilet b on b.id_seansu = s.id_seansu
where plec = 'K'
group by a.id_aktora, nazwisko, imie
having count(*) = (select max(count(*))
      from aktor a1 join aktor_film af1 on a1.id_aktora = af1.id_aktora
      join film f1 on af1.nr_filmu = f1.nr_filmu
      join seans s1 on s1.nr_filmu = f1.nr_filmu
      join bilet b1 on b1.id_seansu = s1.id_seansu
      where plec = 'K'
      group by a1.id_aktora, nazwisko, imie);

create or replace view Filmy_budzet_powyzej_sredniej_dla_gatunku
as
select nazwa_filmu, opis, budzet
from film f join gatunek g on f.nr_gatunku = g.nr_gatunku
where budzet > (select avg(budzet)
      from film f1 join gatunek g1 on f1.nr_gatunku = g1.nr_gatunku
      where f.nr_gatunku = f1.nr_gatunku);

create or replace view Najczesciej_ogladany_film

```

```
as
select *
from (select nazwa_filmu, count(*) "Ilość zakupionych biletów"
      from film f join seans s on s.nr_filmu = f.nr_filmu
      join bilet b on b.id_seansu = s.id_seansu
      group by f.nr_filmu, nazwa_filmu
      order by 2 desc)
where rownum <= 1;
```

Skrypt deinstalujący (PROJEKT_DEINSTALACJA.sql):

```
drop table aktor cascade constraints;

drop table aktor_film cascade constraints;

drop table bilet cascade constraints;

drop table film cascade constraints;

drop table gatunek cascade constraints;

drop table klient cascade constraints;

drop table sala cascade constraints;

drop table seans cascade constraints;

drop table typ_biletu cascade constraints;

drop table zakup cascade constraints;

drop view FILMY_BUDZET_POWYZEJ_SREDNIEJ_DLA_GATUNKU;

drop view ILOSC_BILETOW_NA_GATUNEK;

drop view NAJPOPULARNIEJSZY_AKTOR_ORAZ_AKTORKA;
```

```
drop view PRZYCHOD_FILMU_OSTATNIE_TRZY_MIESIACE;  
  
drop view NAJCZESCIEJ_OGLADANY_FILM;  
  
drop sequence aktor_seq;  
  
drop sequence bilet_seq;  
  
drop sequence film_seq;  
  
drop sequence gatunek_seq;  
  
drop sequence klient_seq;  
  
drop sequence sala_seq;  
  
drop sequence seans_seq;  
  
drop sequence zakup_seq;
```

5. Instrukcję instalacji projektu i sprawdzenia jego poprawności

Instrukcja instalacji:

Plik PROJEKT_INSTALACJA.sql umieścić w folderze temp na dysku C, a następnie w Oracle SQL Developer uruchomić go za pomocą komendy @"C:\temp\PROJEKT_INSTALACJA.sql".

Poprawność można sprawdzić w widoku lewej strony SQL Developera, po odświeżeniu, będą widoczne tabele, widoki oraz sekwensery.

Sprawdzić można również spójność bazy danych poprzez próbę wprowadzenia np. biletu na seans, który nie istnieje lub filmu, dla gatunku, który nie istnieje. Te przykładowe operacje nie powiodą się, gdyż zdefiniowane są klucze obce, zapewniające wspomnianą spójność.