**Projekt Kino**

Mateusz Pilszczek 4G ZEG

Spis treści

[Dokumentacja 2](#_Toc134908499)

[Analiza środowiska kina 2](#_Toc134908500)

[Zapytania 4](#_Toc134908501)

[Tworzenie bazy danych i importowanie danych. 5](#_Toc134908502)

[Kwerendy z tworzenia tabeli 6](#_Toc134908503)

[Tabela admin 6](#_Toc134908504)

[Tabela bilety 6](#_Toc134908505)

[Tabela film 7](#_Toc134908506)

[Tabela klient 7](#_Toc134908507)

[Tabela rezerwacje 8](#_Toc134908508)

[Tabela sala 8](#_Toc134908509)

[Tabela seans 8](#_Toc134908510)

[Kwerendy z uzupełniania tabel danymi 9](#_Toc134908511)

[Tabela bilety 10](#_Toc134908512)

[Tabela admin 10](#_Toc134908513)

[Tabela film 10](#_Toc134908514)

[Tabela sala 12](#_Toc134908515)

[Tabela klient 13](#_Toc134908516)

[Tabela seans 13](#_Toc134908517)

[Screen z diagramu ER 14](#_Toc134908518)

[Pliki bazy danych i WWW 14](#_Toc134908519)

# Dokumentacja

## Analiza środowiska kina

opis co się tam dzieje – Potencjalni klienci przychodzą obejrzeć seans na wielkim ekranie projektora. Zazwyczaj do filmu towarzyszy im przekąska, którą kupują przed wejściem do sali kinowej.

na czym firma zarabia – Firma zarabia na klientach, którzy przychodzą obejrzeć na film, na przekąskach, które umilają seans widzom i na reklamach puszczanych tuż przed kinematografią Firma zarabia dodatkowo na udostępnianiu sali innym zainteresowanym firmą.

jakie ma koszty – koszty utrzymania, sprzątania, media, opłaty rządowe, opłaty za filmy, opłaty za hosting strony kina, opłaty za zatrudnionych pracowników, druk biletów.

możliwe zdarzenia:

niezadowolony klient żądający zwrotu pieniędzy,

klient żądający zwrotu pieniędzy bo mu przekąska nie smakuje,

Baza danych składa się z:

* Admin
  + idadmin - INT NOT NULL AUTO\_INCREMENT
  + Imie – VARCHAR(45) NOT NULL
  + Nazwisko – VARCHAR(45) NOT NULL
  + Stanowisko – VARCHAR(45) NOT NULL
* Bilety
  + Idbilety – INT NOT NULL AUTO\_INCREMENT
  + Rodzaj – VARCHAR(45) NOT NULL
  + Cena – INT NOT NULL
  + Seans\_idseans - INT NULL
* Film
  + Idfilm – int(11) NOT NULL AUT\_INCREMENT
  + Title – varchar(45) NOT NULL
  + Rezyser – varchar(45) NULL
  + Rok\_prod – int(4) NULL
  + Czas\_trwania – varchar(10) NULL
  + Kraj – varchar(45) NULL
  + Obsada – longtext NULL
  + Ogr\_wiek – varchar(3) NULL
  + Gatunek – varchar(45) NULL
  + Opis - longtext
* Klient
  + Idklient - INT NOT NULL
  + Imie - VARCHAR(45) NOT NULL
  + Nazwisko - VARCHAR(45) NOT NULL
  + Email - VARCHAR(45) NOT NULL
  + Telefon - INT NOT NULL
* Rezerwacje
  + Idrezerwacje - INT NOT NULL AUTO\_INCREMENT
  + Seans\_idseans - INT NULL
  + Klient\_idklient - INT NULL
* Sala
  + Idsala - INT NOT NULL AUTO\_INCREMENT
  + Nazwa\_Sali - VARCHAR(45) NOT NULL
  + Miejsca - INT UNSIGNED NOT NULL
* Seans
  + Idseans - INT NOT NULL AUTO\_INCREMENT
  + Data - DATE NOT NULL
  + Godzina - VARCHAR(5) NOT NULL
  + Film\_idfilm - INT NULL
  + Sala\_idsala - INT NULL

## Zapytania

1. Podaj tytuł i czas trwania filmu o nazwie Fernando:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) title, czas\_trwania FROM film WHERE title = "Fernando";

1. Wybierz wszystkie filmy gatunku fantasy:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM film WHERE gatunek = 'Fantasy';

1. Podaj wszystkie tytuły filmów bez ograniczeń wiekowych:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) title FROM film WHERE ogr\_wiek = "BO";

1. Wyświetl salę o największej liczbie miejsc:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nazwa\_sali, miejsca FROM sala ORDER BY miejsca DESC LIMIT 1;

1. Wypisz imię, nazwisko i nr telefonu klienta o id 5

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) imie, nazwisko, telefon FROM klient WHERE klient.idklient = 5;

1. Wyświetl sumę miejsc we wszystkich salach:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [SUM](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_sum)(miejsca) FROM sala;

1. Podaj rok produkcji filmu „Atak paniki”.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) title, rok\_prod FROM film WHERE title = "Atak paniki";

1. Wyświetl imię nazwisko i id rezerwacji klientów:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) rezerwacje.idrezerwacje, klient.imie, klient.nazwisko FROM rezerwacje JOIN klient ON rezerwacje.klient\_idklient = klient.idklient;

1. Wyświetl czas trwania filmu „Narzeczony na niby”.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) title, czas\_trwania FROM film WHERE title = "Narzeczony na niby";

1. Wyświetl jakiego gatunku jest film „Plan B”.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) title, gatunek FROM film WHERE title = "Plan B";

1. Podaj ile miejsca jest w Sali numer 3.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) idsala, miejsca FROM sala WHERE sala.idsala = "3";

1. Wypisz nazwy wszystkich dostępnych sal.
2. [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nazwa\_sali FROM sala;
3. Wyświetl wszystkie filmy, których reżyserem jest Rian Johnson.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM film WHERE rezyser = "Rian Johnson";

1. Podaj imię i nazwisko managera.

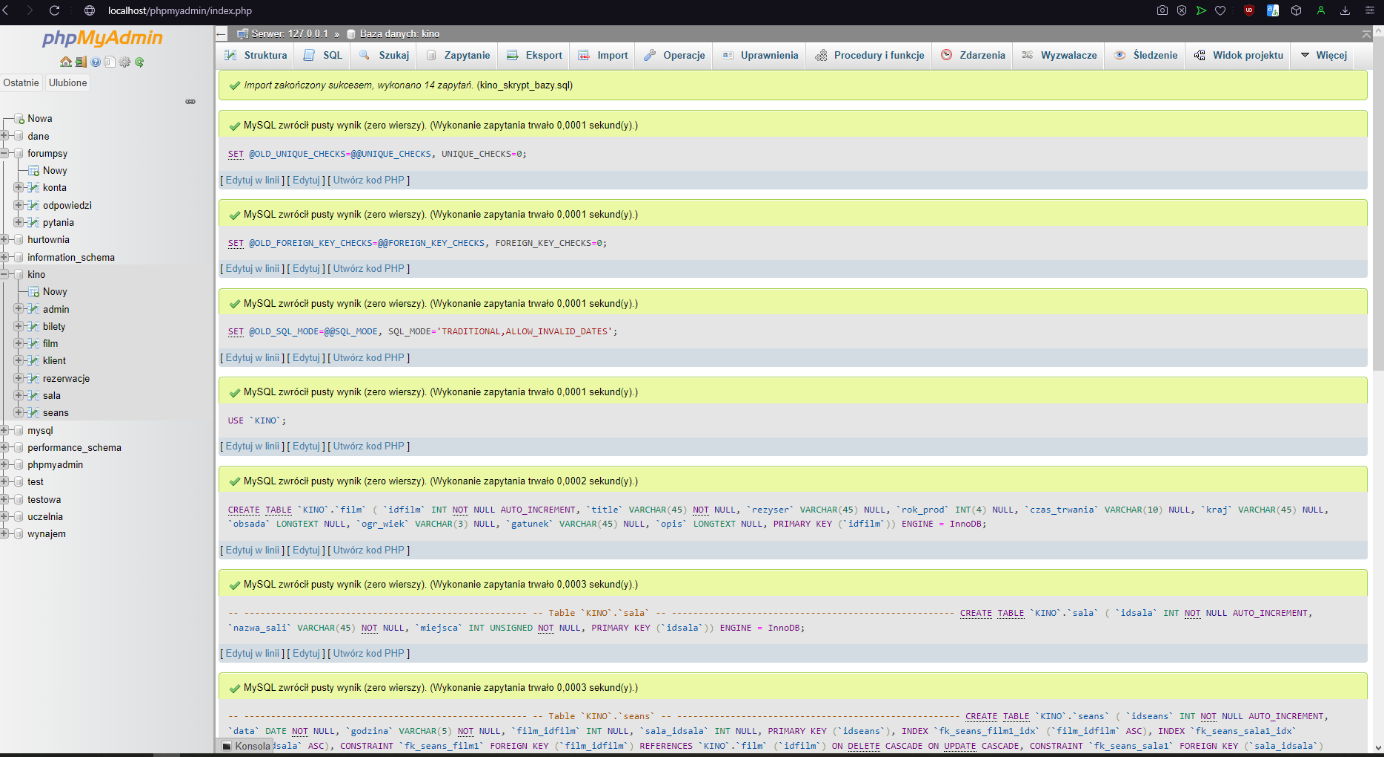
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) imie, nazwisko FROM admin WHERE stanowisko = "manager";

1. Wyświetl nazwiska i nr tel wszystkich klientów

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) nazwisko, telefon FROM klient;

## Tworzenie bazy danych i importowanie danych.

Zrzut ekranu z importu bazy danych:



### Kwerendy z tworzenia tabeli

#### Tabela admin

CREATE TABLE `admin` (

`idadmin` int(11) NOT NULL,

`imie` varchar(45) NOT NULL,

`nazwisko` varchar(45) NOT NULL,

`stanowisko` varchar(45) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `admin`

ADD PRIMARY KEY (`idadmin`);

MODIFY `idadmin` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=4;

#### Tabela bilety

CREATE TABLE `bilety` (

`idbilety` int(11) NOT NULL,

`rodzaj` varchar(45) NOT NULL,

`cena` int(11) NOT NULL,

`seans\_idseans` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `bilety`

ADD PRIMARY KEY (`idbilety`),

ADD KEY `fk\_bilety\_seans1\_idx` (`seans\_idseans`);

ADD CONSTRAINT `fk\_bilety\_seans1` FOREIGN KEY (`seans\_idseans`) REFERENCES `seans` (`idseans`) ON DELETE CASCADE ON UPDATE CASCADE;

MODIFY `idbilety` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=3;

#### Tabela film

CREATE TABLE `film` (

`idfilm` int(11) NOT NULL,

`title` varchar(45) NOT NULL,

`rezyser` varchar(45) DEFAULT NULL,

`rok\_prod` int(4) DEFAULT NULL,

`czas\_trwania` varchar(10) DEFAULT NULL,

`kraj` varchar(45) DEFAULT NULL,

`obsada` longtext DEFAULT NULL,

`ogr\_wiek` varchar(3) DEFAULT NULL,

`gatunek` varchar(45) DEFAULT NULL,

`opis` longtext DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `film`

ADD PRIMARY KEY (`idfilm`);

MODIFY `idfilm` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

#### Tabela klient

CREATE TABLE `klient` (

`idklient` int(11) NOT NULL,

`imie` varchar(45) NOT NULL,

`nazwisko` varchar(45) NOT NULL,

`email` varchar(45) NOT NULL,

`telefon` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `klient`

ADD PRIMARY KEY (`idklient`);

#### Tabela rezerwacje

CREATE TABLE `rezerwacje` (

`idrezerwacje` int(11) NOT NULL,

`seans\_idseans` int(11) DEFAULT NULL,

`klient\_idklient` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `rezerwacje`

ADD PRIMARY KEY (`idrezerwacje`),

ADD KEY `fk\_rezerwacje\_seans1\_idx` (`seans\_idseans`),

ADD KEY `fk\_rezerwacje\_klient1\_idx` (`klient\_idklient`);

ADD CONSTRAINT `fk\_rezerwacje\_klient1` FOREIGN KEY (`klient\_idklient`) REFERENCES `klient` (`idklient`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `fk\_rezerwacje\_seans1` FOREIGN KEY (`seans\_idseans`) REFERENCES `seans` (`idseans`) ON DELETE CASCADE ON UPDATE CASCADE;

MODIFY `idrezerwacje` int(11) NOT NULL AUTO\_INCREMENT;

#### Tabela sala

CREATE TABLE `sala` (

`idsala` int(11) NOT NULL,

`nazwa\_sali` varchar(45) NOT NULL,

`miejsca` int(10) UNSIGNED NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `sala`

ADD PRIMARY KEY (`idsala`);

MODIFY `idsala` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=5;

#### Tabela seans

CREATE TABLE `seans` (

`idseans` int(11) NOT NULL,

`data` date NOT NULL,

`godzina` varchar(5) NOT NULL,

`film\_idfilm` int(11) DEFAULT NULL,

`sala\_idsala` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

ALTER TABLE `seans`

ADD PRIMARY KEY (`idseans`),

ADD KEY `fk\_seans\_film1\_idx` (`film\_idfilm`),

ADD KEY `fk\_seans\_sala1\_idx` (`sala\_idsala`);

ADD CONSTRAINT `fk\_seans\_film1` FOREIGN KEY (`film\_idfilm`) REFERENCES `film` (`idfilm`) ON DELETE CASCADE ON UPDATE CASCADE,

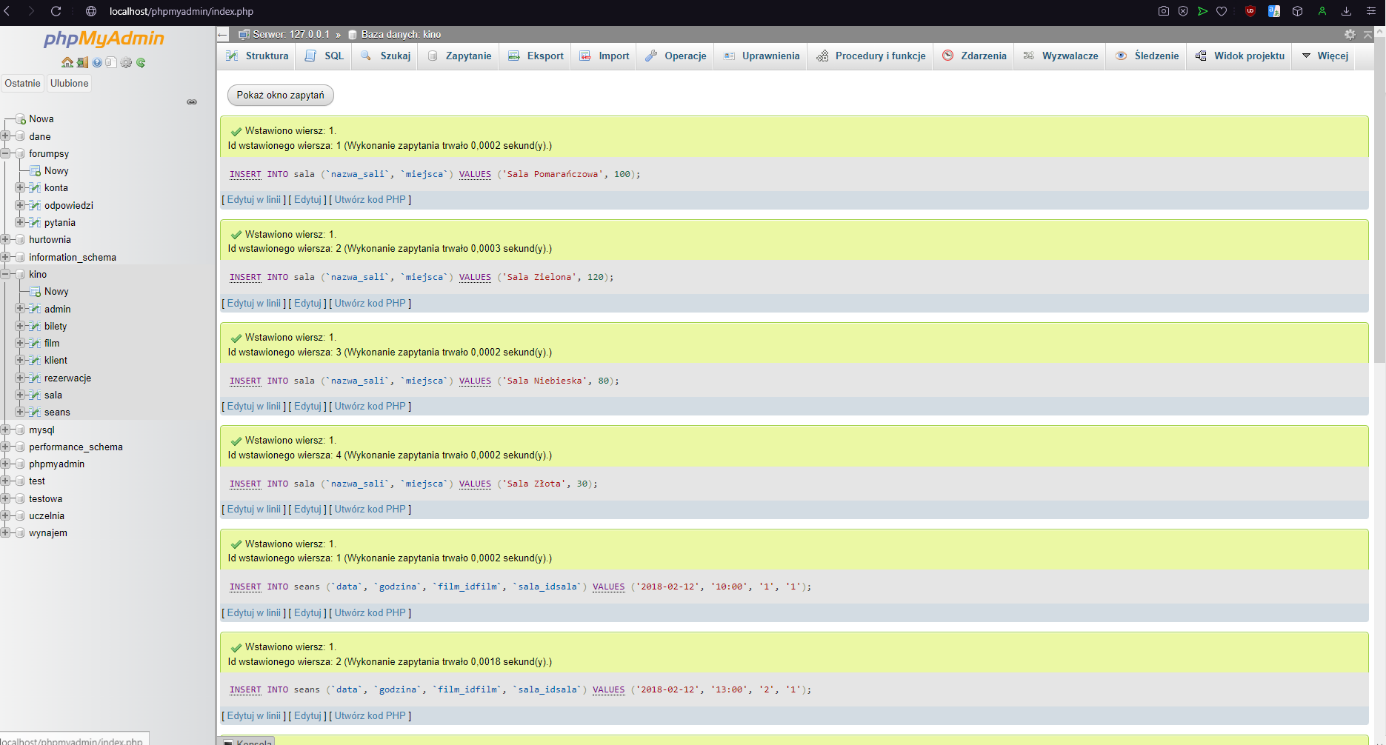
ADD CONSTRAINT `fk\_seans\_sala1` FOREIGN KEY (`sala\_idsala`) REFERENCES `sala` (`idsala`) ON DELETE NO ACTION ON UPDATE NO ACTION;

COMMIT;

MODIFY `idseans` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

### Kwerendy z uzupełniania tabel danymi

Wszystkie kwerendy z wpisanymi danymi znajdują się poniżej;



#### Tabela bilety

INSERT INTO bilety (`rodzaj`, `cena`) VALUES ('Normalny', 25);

INSERT INTO bilety (`rodzaj`, `cena`) VALUES ('Ulgowy', 20);

#### Tabela admin

INSERT INTO admin (`imie`, `nazwisko`, `stanowisko`) VALUES ('Piotr', 'Kowalski', 'manager');

INSERT INTO admin (`imie`, `nazwisko`, `stanowisko`) VALUES ('Jan', 'Nowak', 'kasjer');

INSERT INTO admin (`imie`, `nazwisko`, `stanowisko`) VALUES ('Alicja', 'Piotrowska', 'kasjer');

#### Tabela film

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Plan B', 'Kinga Dębska', 2017, '86 min', 'Polska', 'Kinga Preis, Marcin Dorociński, Edyta Olszówka, Roma Gąsiorowska, Krzysztof Stelmaszyk', '15+', 'Dramat, Komedia', 'Film o miłości - pełen emocji, humoru i życiowej prawdy. Bohaterów poznajemy przed walentynkami w chwili, gdy w ich życiu wydarza się coś całkiem nieoczekiwanego - coś, co wywraca je do góry nogami. Sytuacje, z którymi się konfrontują, prowadzą do zaskakujących rozwiązań. Dają nadzieję na to, że trudne początki nieraz prowadzą do odnalezienia prawdziwych uczuć i autentycznych więzi, a w życiu zawsze jest jakiś plan B!');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Pszczółka Maja: Miodowe Igrzyska', 'Noel Cleary, Alexis Stadermann', 2018, '85 min', 'Niemcy', '','BO', 'Komedia, Animowany, Familijny', 'Druga część szalonych i zabawnych perypetii najsłynniejszej pszczółki świata, która żadnej przygody się nie boi! Tym razem Maja i Gucio wezmą udział w międzyłąkowych igrzyskach sportowych, których stawka okaże się wyjątkowo wysoka. W ulu aż huczy od sensacyjnych wieści! W pszczelej stolicy odbyć się mają Miodowe Igrzyska. Maja marzy, by wziąć w nich udział. Niestety mieszkańcy ula nie zostają na nie zaproszeni. Na domiar złego Cesarzowa żąda oddania połowy miodowych zbiorów na potrzeby sportowców. Królowa gotowa jest się pogodzić z poleceniami przełożonej, ale Maja od dziecka jest buntowniczką. A nic nie złości jej bardziej niż jawna niesprawiedliwość. Wyrusza więc do stolicy, by powiedzieć Cesarzowej, co myśli o niej i jej nakazach. I nawet Guciowi nie uda się jej powstrzymać przed wpakowaniem się w kłopoty. Zuchwałość Mai zostanie ukarana: pszczółka musi wystartować w zawodach, ale jeśli przegra jej ul straci całe zapasy na zimę… Cesarzowa nie podejrzewa nawet na co stać tę małą rozrabiakę i paczkę jej wiernych przyjaciół.');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Winchester: Dom duchów', 'Michael Spierig, Peter Spierig', 2018, '99 min', 'Australia, USA', 'Helen Mirren, Jason Clarke', '15+', 'Horror', 'Mrożący krew w żyłach horror inspirowany prawdziwymi wydarzeniami, jakie miały miejsce w jednym z najsłynniejszych nawiedzonych domów w historii. Laureatka Oscara Helen Mirren i Jason Clarke w opowieści o wdowie, która płaci przerażającą cenę za grzechy swego męża…lub tak jej się tylko wydaje. Po tragicznej śmierci męża i syna, Sarah Winchester pogrąża się w żałobie i oddala od codziennego życia. Jest przekonana, że jej rodzinę nawiedzają duchy osób, które zostały zabite z broni skonstruowanej przez jej męża, czyli ze słynnego karabinu Winchester. By bronić się przed demonami Sarah zleca budowę niezwykłej posiadłości. Składający się z setek pokoi, skonstruowany jak labirynt, pełen prowadzących donikąd schodów i ślepych korytarzy dom ma być pułapką na duchy. Rodzina Winchester, zaniepokojona stanem zdrowia wdowy wzywa na konsultację słynnego psychiatrę, który ma odwieść kobietę od jej szalonych planów. Wkrótce okazuje się jednak, że Sarah Winchester wcale nie oszalała…');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Kształt wody', 'Guillermo del Toro', 2017, '119 min', 'USA', 'Sally Hawkins, Michael Shannon, Richard Jenkins, Doug Jones, Michael Stuhlbarg, Octavia Spencer', '12+', 'Fantasy', 'Wyróżniony 7 nominacjami do Złotych Globów i Złotym Lwem na festiwalu w Wenecji 2017 najnowszy film wizjonerskiego reżysera Guillermo del Toro, twórcy nagrodzonego 3 Oscarami „Labiryntu fauna”. Baśń dla dorosłych, której akcja rozgrywa się u szczytu zimnej wojny, w Stanach Zjednoczonych około roku 1962. Elisa (nominowana do Oscara Sally Hawkins) wiedzie monotonną, samotną egzystencję, a całą noc pracuje w pilnie strzeżonym, sekretnym laboratorium rządowym. Jej życie zmienia się na zawsze, gdy wraz z koleżanką z pracy, Zeldą (nagrodzona Oscarem Octavia Spencer), odkrywa, że w laboratorium przeprowadzany jest otoczony ścisłą tajemnicą eksperyment, który zaważyć może na przyszłych losach świata. W pozostałych rolach: nominowany do Oscara Michael Shannon, nominowany do Oscara Richard Jenkins, Michael Stuhlbarg i Doug Jones.');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Podatek od miłości', 'Bartłomiej Ignaciuk', 2017, '102 min', 'Polska', 'Aleksandra Domańska, Grzegorz Damięcki, Roma Gąsiorowska, Weronika Rosati, Magdalena Popławska', '15+', 'Romantyczny, Komedia', '„Podatek od miłości” to nowa produkcja kinowa TVN. W rolach głównych plejada gwiazd: Grzegorz Damięcki, Aleksandra Domańska, Roma Gąsiorowska, Magdalena Różczka, Magdalena Popławska, Anna Smołowik, Weronika Rosati, Ewa Gawryluk, Grażyna Wolszczak, Michał Czernecki, Zbigniew Zamachowski oraz Tomasz Włosok. „Podatek od miłości” to inteligentna i przewrotna komedia romantyczna o miłości, którą można spotkać wszędzie, nawet… w Urzędzie Skarbowym. Surowa inspektor Klara (Aleksandra Domańska) ściga domniemanego oszusta podatkowego – cynicznego, acz pełnego uroku Mariana (Grzegorz Damięcki), który przez niefrasobliwość i dobre serce popadł w kłopoty podatkowe. Kiedy los „ściganego” wydaje się przesądzony, nieoczekiwanie i wbrew wszelkim regułom Klara i Marian zakochują się. Uruchomi to ciąg komicznych wydarzeń, które odmienią życie bohaterów. On dla dobra sprawy może zostać facetem do towarzystwa. Ona może trafić do więzienia. Czy tak się stanie?');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Gwiezdne Wojny: Ostatni Jedi', 'Rian Johnson', 2017, '150 min', 'USA', 'Daisy Ridley, John Boyega, Mark Hamill, Oscar Isaac, Carrie Fisher', '12+', 'Sci-Fi, Przygodowy', 'W najnowszej produkcji Lucasfilm „Gwiezdne Wojny: Ostatni Jedi" bohaterowie "Przebudzenia Mocy" wraz z legendarnymi postaciami gwiezdnego uniwersum odkrywają zaskakujące sekrety przeszłości i niezgłębione dotąd tajemnice Mocy…');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Narzeczony na niby', 'Bartosz Prokopowicz', 2018, '112 min', 'Polska', 'ulia Kamińska, Piotr Stramowski, Sonia Bohosiewicz, Piotr Adamczyk, Barbara Kurdej-Szatan, Tomasz Karolak, Dorota Kolak', '12+', 'Romantyczny, Komedia', 'Czy małe, niewinne kłamstewko może wymknąć się spod kontroli i doprowadzić do totalnej katastrofy? A może dopiero jak wszystko się w życiu rozpada, możliwe staje się poukładanie wszystkiego od nowa? Karina próbuje z całych sił stworzyć pierwszy (tym razem) szczęśliwy związek. Niestety mężczyzna, któremu oddała swoje serce, wydaje się być bardziej zainteresowany sobą, niż nią. Pewnego dnia, jedno nieprzewidziane zdarzenie i jedno małe kłamstewko zamieni jej uporządkowane życie w prawdziwy emocjonalny rollercoster. Szybko okaże się też, że każdy z bohaterów ma tak naprawdę coś do ukrycia. Gdy prawda wyjdzie na jaw, wszyscy będą musieli zrobić w końcu porządek w swoich związkach i w swoich sercach. „Narzeczonego na niby” pokochają ci, którzy choć raz zakochali się w niewłaściwej osobie oraz ci, którzy wierzą, że o prawdziwą miłość warto walczyć, nawet jeśli zaczyna się mocno niefortunnie.');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Atak paniki', 'Paweł Maślona', 2017, '99 min', 'Polska', 'Dorota Segda, Artur Żmijewski, Grzegorz Damięcki, Magdalena Popławska, Mirosław Haniszewsi, Andrzej Konopka', '15+', 'Dramat, Komedia', 'Wielowątkowa historia pozornie niezwiązanych ze sobą osób, które w pewnym momencie doświadczają... ataku paniki. Opowieść o tym jak z pozoru nieistotne w życiu zdarzenia mogą doprowadzić do ogromnych konsekwencji, metaforycznych końców świata.');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Cudowny chłopak', 'Stephen Chbosky', 2017, '113 min', 'Hongkong, USA', 'Jacob Tremblay, Julia Roberts, Owen Wilson', '7+', 'Dramat, Komedia', 'Julia Roberts, Owen Wilson i Jacob Tremblay, młody gwiazdor znakomitego „POKOJU” w porywającej opowieści o tym, że niełatwo być innym. Ekranizacja międzynarodowego bestsellera, który wzruszył i rozbawił miliony czytelników na całym świecie. Każdy chce się wyróżniać. Nikt nie chce być inny. Dla 10-letniego Auggiego nie ma niczego bardziej niezwykłego niż „zwykły dzień w szkole”. Urodzony z licznymi deformacjami twarzy, dotąd uczył się w domu pod okiem mamy. Teraz jednak zaczynając 5 klasę w normalnej szkole, ma nadzieję, że koledzy potraktują go jak zwyczajnego chłopca. Jego wygląd sprawia jednak, że staje się szkolną sensacją, a dla wielu wręcz „dziwadłem”. Jedni się z niego śmieją, inni wytykają palcami, ale nikt tak naprawdę niczego o nim nie wie. Wszystko zacznie się zmieniać, gdy w szkole pojawi się nowa uczennica. Urocza Summer potrafi dostrzec w Auggiem coś więcej niż nietypowy wygląd. Wystarczy jeden przyjazny gest, jedna pomocna dłoń, by obudzić w Auggiem siłę, dzięki której będzie potrafił udowodnić, że nie tylko twarz czyni go kimś niezwykłym.');

insert into film (title, rezyser, rok\_prod, czas\_trwania, kraj, obsada, ogr\_wiek, gatunek, opis)

values ('Fernando', 'Carlos Saldanha', 2017, '108 min', 'USA', '', 'BO', 'Komedia, Animowany, Familijny', '“Fernando” to przygodowa komedia animowana, inspirowana słynną książką dla dzieci o byczku Fernando, autorstwa Munro Leafa i Roberta Lawsona, a zrealizowana w studio Blue Sky, w którym powstały m.in.: ”Epoka lodowcowa” i ”Rio”. Tytułowy Fernando to wielki byczek o równie wielkim sercu. Omyłkowo wzięty za groźną bestię zostaje schwytany i uprowadzony z rodzinnego domu w hiszpańskim miasteczku. Gotowy na wszystko, by powrócić na łono rodziny, Fernando staje na czele grupy niezwykłych sojuszników i wyrusza na wielką wyprawę, podczas której przeżyje przygodę swego życia. W polskiej wersji językowej Fernando przemówi głosem Marcina Dorocińskiego.');

#### Tabela sala

INSERT INTO sala (`nazwa\_sali`, `miejsca`) VALUES ('Sala Pomarańczowa', 100);

INSERT INTO sala (`nazwa\_sali`, `miejsca`) VALUES ('Sala Zielona', 120);

INSERT INTO sala (`nazwa\_sali`, `miejsca`) VALUES ('Sala Niebieska', 80);

INSERT INTO sala (`nazwa\_sali`, `miejsca`) VALUES ('Sala Złota', 30);

#### Tabela klient

INSERT INTO klient (idklient, imie, nazwisko, email, telefon)

VALUES

(1, 'Jan', 'Kowalski', 'jan.kowalski@example.com', 123456789),

(2, 'Anna', 'Nowak', 'anna.nowak@example.com', 987654321),

(3, 'Michał', 'Wiśniewski', 'michal.wisniewski@example.com', 555666777),

(4, 'Karolina', 'Kwiatkowska', 'karolina.kwiatkowska@example.com', 111222333);

(5, 'Katarzyna', 'Nowicka', 'katarzyna.nowicka@example.com', 444555666);

#### Tabela seans

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '10:00', '1', '1');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '13:00', '2', '1');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '17:00', '3', '2');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '21:00', '4', '2');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '11:00', '5', '3');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '14:00', '6', '3');

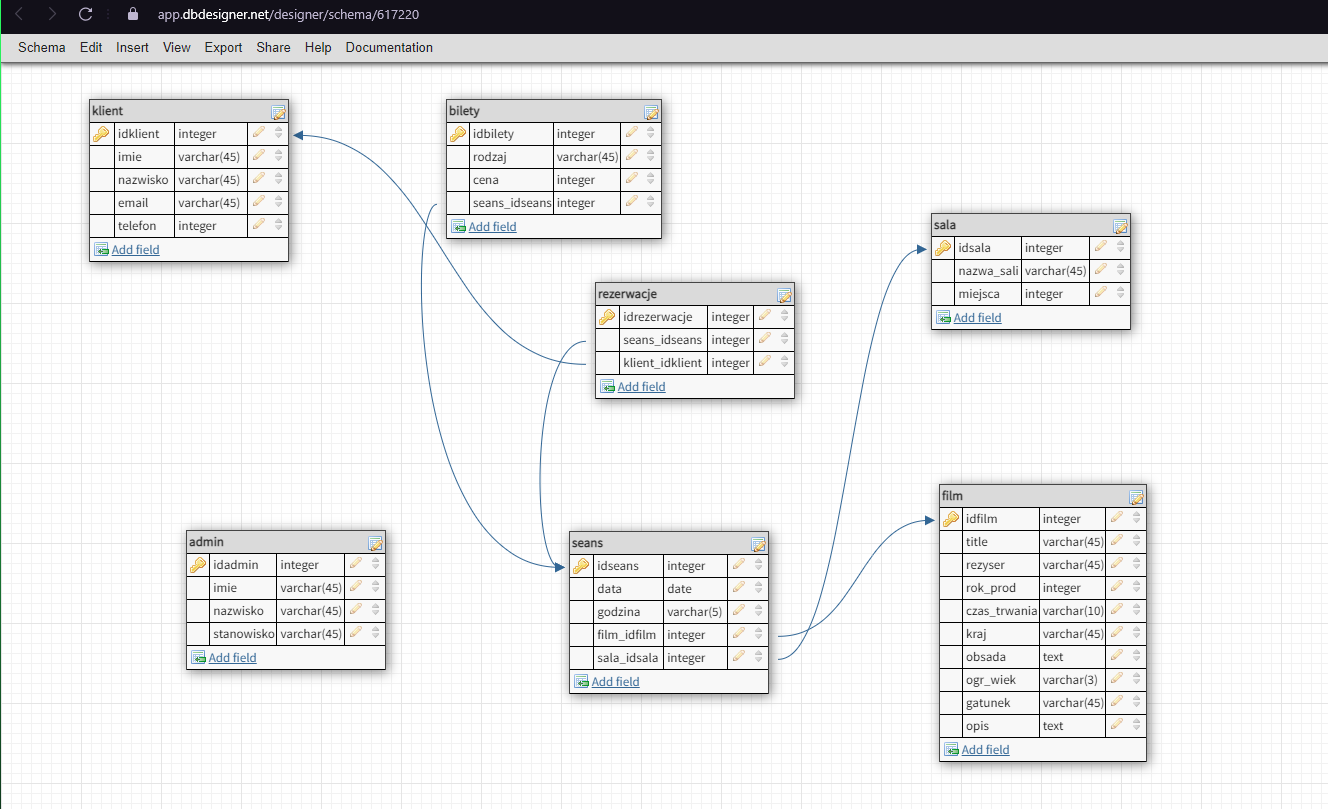
INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '18:30', '7', '4');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '22:00', '8', '4');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '19:00', '9', '3');

INSERT INTO seans (`data`, `godzina`, `film\_idfilm`, `sala\_idsala`) VALUES ('2018-02-12', '22:00', '10', '1');

# Screen z diagramu ER



Link: <https://dbdesigner.page.link/maAGbZjEkiW7voG48>

# Pliki bazy danych i WWW

Wszelkie pliki są dostępna na:

<https://szkolnastrona123.000webhostapp.com/kino-PROJEKT/kino.php>

kod źródłowy oraz wszelkie pliki na githubie:

<https://github.com/Mejteoo/ProjektKino-bazydanych.git>

Proszę wykorzystać plik o nazwie „lista kwerend.txt”

<https://github.com/Mejteoo/ProjektKino-bazydanych/blob/main/lista%20kwerend.txt>