

POLITECHNIKA LUBELSKA
Wydział Elektrotechniki i Informatyki
Kierunek Informatyka



INŻYNIERIA SYSTEMÓW BAZ DANYCH

System bazy danych platformy „Netflix”

Autorzy:

Grzegorz Jan Cichocki

Marcin Dominik Cieśla

Piotr Czajka

Marek Krzysztof Cichoń

Lublin 2021

1. Opis projektu

Projekt polega na przygotowaniu bazy danych, która ma imitować funkcjonalność oraz możliwości platformy streamingowej „Netflix”. Netflix istnieje już od wielu lat i jest popularny w wielu krajach zachodnich, w tym w Polsce. Za miesięczną opłatą, usługa pozwala na oglądanie dowolnej liczby filmów i seriali, które dostępne są na platformie. W usłudze zakładamy konto użytkownika, jednak dzieli się ono na profile. Każdy profil może mieć własną nazwę oraz zdjęcie profilowe, których pula jest zdefiniowana w bazie danych. Każdy profil ma również własną historię oglądania filmów i seriali generowaną na podstawie aktualnego oglądania, jak i listy dzieł chcianych do obejrzenia. Profil może wrócić do oglądania danej produkcji od momentu, w którym poprzednio przestał oglądać. Każdemu użytkownikowi można również przypisać preferowany język, w którym ma wyświetlać się interfejs.

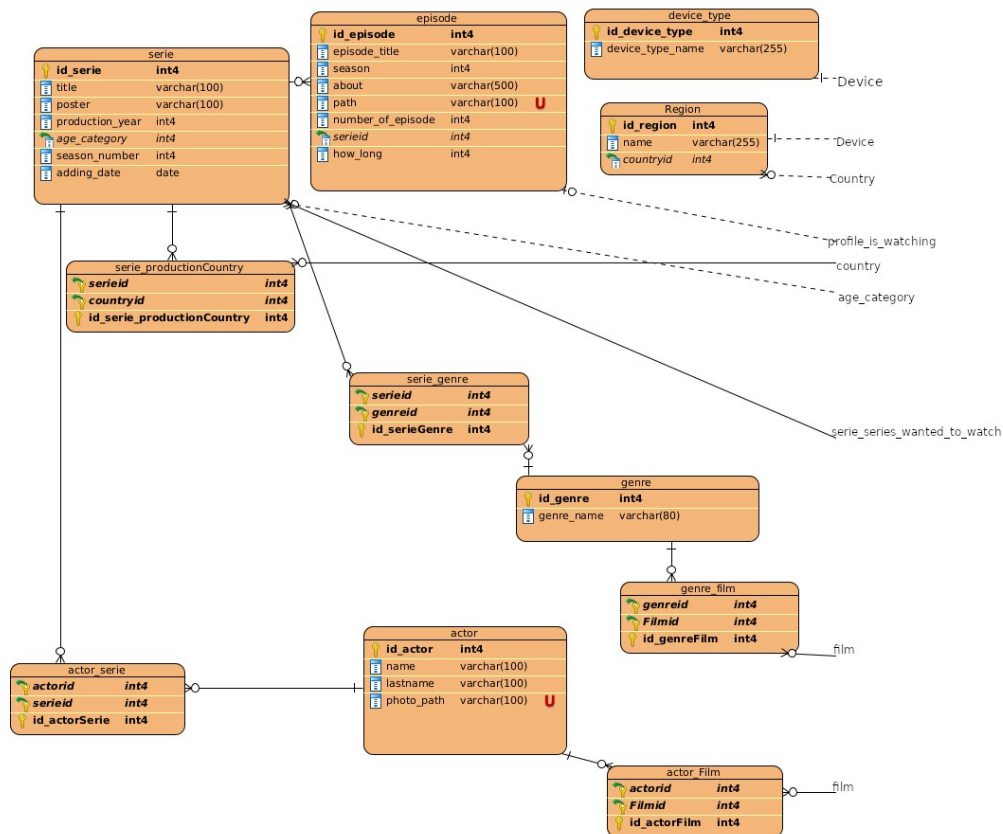
Użytkownik może korzystać z platformy na wielu urządzeniach, których typy zostały wypisane w tabeli słownikowej. Każde urządzenie jest zapisywane w postaci typu urządzenia, jego adresu IP oraz regionu logowania. Użytkownik może również podpiąć kilka metod płatności za subskrypcję – typy subskrypcji zostały wypisane w odpowiedniej tabeli. Zapłacić może za pomocą karty podarunkowej oraz karty kredytowej.

Filmy i seriale są w oddzielnych tabelach, przy czym z serialem połączona jest tabela „Episode”, w której wypisane są wszystkie odcinki danych seriali, w tym, między innymi, ich tytuły, okładki, rodzaje i ścieżki dostępu. Filmy stanowią jedną całość i również one charakteryzują się między innymi takimi polami jak tytuł, rodzaj czy miejsce produkcji. Filmy możemy segregować na podstawie kraju pochodzenia, jak i aktorów w nich występujących. Aktorzy mogą występować w wielu produkcjach, a więc i nawet po ich nazwiskach można wyszukiwać filmy czy seriale. Filmy i seriale mają również swoje własne kategorie wiekowe.

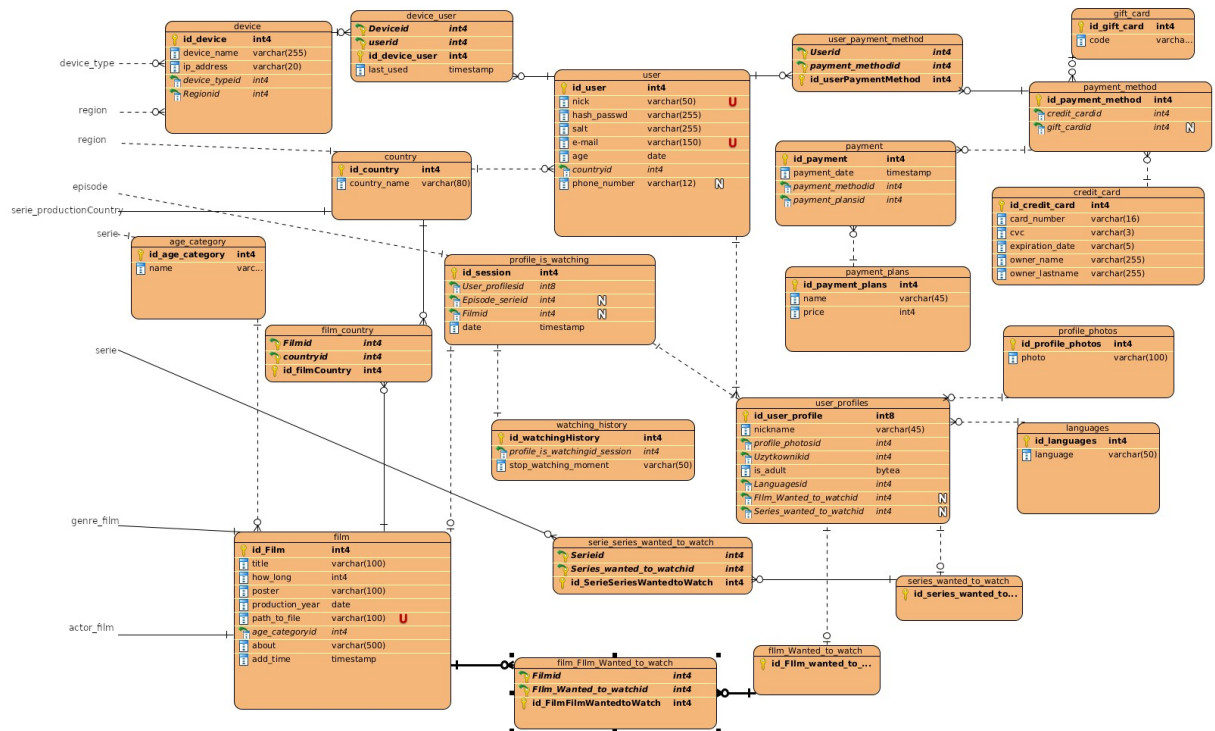
Diagram związków encji został przygotowany w środowisku Visual Paradigm. Wybrany silnikiem bazy danych jest PostgreSQL, a zarządzanie jest możliwe chociażby przez dedykowane oprogramowanie pgAdmin 4. Całość działa na komputerze z systemem operacyjnym GNU/Linux Pop!_OS 20.10.

2. Diagram związków encji zaprojektowanej bazy danych

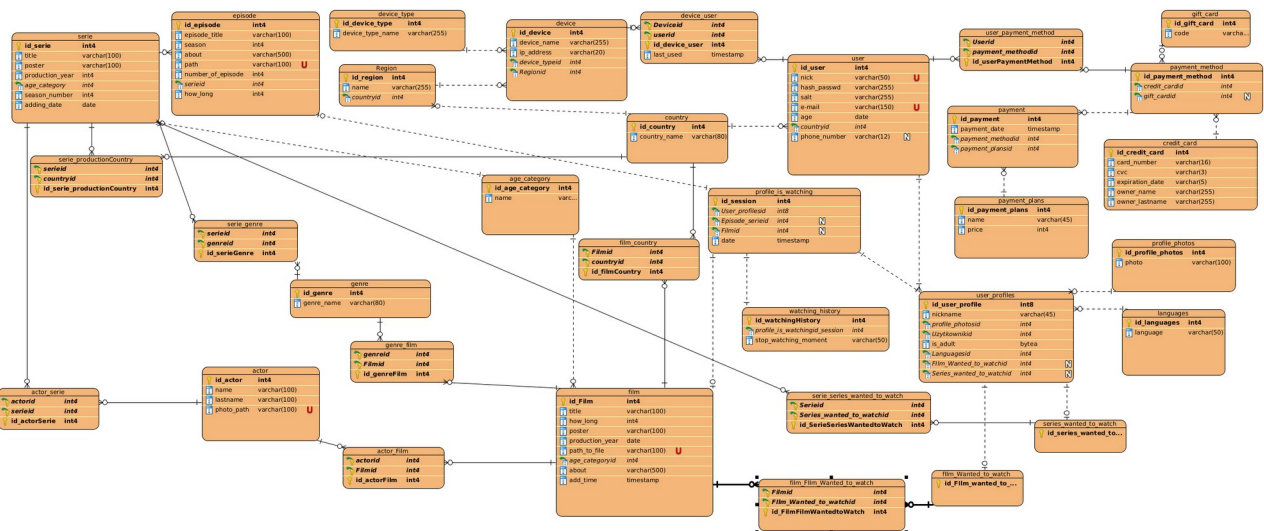
Schemat ERD (diagram związków encji) został podzielony na dwie części, aby zwiększyć czytelność. Przecięte linie relacji zostały odpowiednio podpisane. Diagram związków encji został przygotowany w środowisku Visual Paradigm.



Fot. 1: Lewa część schematu ERD



Fot. 2: Prawa część schematu



Fot. 3: Widok całego schematu ERD

3. Opis tabeli

Tabela 1: Opis tabeli bazy danych

Nazwa tabeli	Typ	Opis
serie	dane	Dane seriali dostępnych na platformie do obejrzenia
episode	dane	Dane danych odcinków danych serialów dostępnych na platformie
actor	dane	Dane aktorów, którzy grają w danych serialach i filmach
genre	walidacja	Tabela słownikowa zawierająca różne gatunki filmów
device	podzbiór	Tabela słownikowa zawierająca typy różnych urządzeń, na których można uruchomić platformę
country	walidacja	Nazwy krajów, z których może pochodzić użytkownik, film lub serial
film	dane	Tabela zawierająca wszystkie filmy możliwe do obejrzenia na platformie wraz z różnymi informacjami
device_user	dane	Tabela łącząca danego użytkownika z urządzeniem oraz oznaczeniem ostatniego użycia
user	dane	Informacje o użytkowniku platformy takie jak nazwa użytkownika, hasło czy email
profile_is_watching	dane	Zawiera informacje o aktualnie oglądanym filmie lub serialu w postaci identyfikatora
watching_history	dane	Zbiór filmów i odcinków seriali odtworzonych przez dany profil użytkownika
payment_plans	podzbiór	Tabela zawierająca wypisane możliwe plany subskrypcji oraz ich ceny
payment	dane	Tabela z informacjami na temat

		dokonanych płatności
user_profiles	dane	Informacje dotyczące profilu użytkownika, w tym nazwa oraz czy jest dorosły
gift_card	dane	Dane kart podarunkowych, użytych do uruchomienia subskrypcji
credit_card	dane	Dane kart płatniczych podanych do uruchomienia i opłacania subskrypcji
profile_photos	podzbiór	Lokalizacje do zdjęć profilowych w postaci ścieżek do pliku
languages	walidacja	Nazwy języków, które mogą być użyte w filmach
series_wanted_to_watch	dane	Lista seriali do obejrzenia przez użytkowników postaci identyfikatora serialu
film_wanted_to_watch	dane	Lista filmów, które użytkownik chciałby obejrzeć w postaci identyfikatora filmu

4. Opis związków między encjami

Tabela 2: Pierwsza część tabeli opisująca związki encji

	serie	actor	genre	film	category_age_	country	episode	user	film_wanted_to_watch	series_wanted_to_watch	profile_is_watching	watching_history	region	user_profiles
serie		1:N	M:N	X	N:1	M:N	1:N	X	X	1:N	X	X	X	X
actor	N:1		X	M:N	X	X	X	X	X	X	X	X	X	X
genre	N:M	X		M:N	X	X	X	X	X	X	X	X	X	X
film	X	N:M	N:M		N:1	M:N	X	X	1:N	X	1:1	X	X	X
age_category	1:N	X	X	1:N		X	X	X	X	X	X	X	X	X
country	N:M	X	X	N:M	X		X	1:N	X	X	X	X	1:N	X
episode	N:1	X	X	X	X	X		X	X	X	1:1	X	X	X
user	X	X	X	X	X	N:1	X		X	X	X	X	X	1:N
film_wanted_to_watch	X	X	X	N:1	X	X	X	X		X	X	X	X	1:1
series_wanted_to_watch	N:1	X	X	X	X	X	X	X	X		X	X	X	1:1
profile_is_watching	X	X	X	1:1	X	X	1:1	X	X	X		1:1	X	1:1
watching_history	X	X	X	X	X	X	X	X	X	X	1:1		X	X
region	X	X	X	X	X	N:1	X	X	X	X	X	X		X
user_profiles	X	X	X	X	X	X	X	N:1	1:1	1:1	1:1	X	X	

Tabela 3: Druga część tabeli opisująca związki encji

[illegible]

5. Opis pól tabeli

Tabela 4: Tabela User

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
id_user	całkowity	TAK	TAK	Wewnętrzny identyfikator użytkownika
nick	znakowy (max. 50)	TAK	TAK	Ciąg znaków określający nazwę użytkownika
hash_password	znakowy (max. 255)	TAK	NIE	Ciąg znaków określający zaszyfrowane hasło
salt	znakowy (max. 255)	TAK	NIE	Ciąg znaków określający sól dodawaną do hasła
e-mail	znakowy (max. 100)	TAK	TAK	Ciąg znaków określający adres email
age	data	TAK	NIE	Data określająca wiek użytkownika
country_id	całkowity	TAK	NIE	Identyfikator odnoszący się do kraju, z którego pochodzi użytkownik
phone_number	znakowy (max.12)	NIE	NIE	Ciąg znaków określający numer telefonu użytkownika

Tabela 5: Tabela User_profiles

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
id_user_	całkowity	TAK	TAK	Identyfikator

profile				profilu użytkownika
nickname	znakowy (max. 45)	TAK	NIE	Ciąg znaków określających nazwę profilu
profile_photosid	całkowity	TAK	NIE	Identyfikator odnoszący się do zdjęcia profilowego profilu
uzytkownikid	całkowity	TAK	NIE	Identyfikator odnoszący się do użytkownika, którego dotyczy profil
is_adult	binarny	TAK	NIE	Znak określający czy użytkownik profilu jest dorosły
languagesid	całkowity	TAK	NIE	Identyfikator odnoszący się do języka przypisanego do profilu
Film_wanted_to_watchid	całkowity	NIE	NIE	Identyfikator odnoszący się do listy filmów do obejrzenia
Series_wanted_to_watchid	całkowity	NIE	NIE	Identyfikator odnoszący się do listy seriali do obejrzenia

Tabela 6: Tabela profile_is_watching

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_session	całkowity	TAK	TAK	Identyfikator sesji
User_ profiles	całkowity	TAK	NIE	Identyfikator odnoszący się do profilu użytkownika
Episode_ serie	całkowity	NIE	NIE	Identyfikator odnoszący się do danego odcinka serialu
Filmid	całkowity	NIE	NIE	Identyfikator

				odnoszący się do danego filmu
date	znakowy czasu	TAK	NIE	Data określająca czas rozpoczęcia oglądania

Tabela 7: Tabela Film

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
id_Film	całkowity	TAK	TAK	Identyfikator filmu
title	znakowy (max. 100)	TAK	NIE	Ciąg znaków określający tytuł filmu
how_long	całkowity	TAK	NIE	Liczba całkowita określająca długość filmu w minutach
poster	znakowy (max. 100)	TAK	NIE	Ciąg znaków określający ścieżkę do pliku z plakatem filmu
production_year	data	TAK	NIE	Data określająca rok produkcji filmu
path_to_file	znakowy (max. 100)	TAK	TAK	Ciąg znaków określający ścieżkę do pliku z filmem
age_categoryid	całkowity	TAK	NIE	Identyfikator odnoszący się do kategorii filmu
about	znakowy (max. 500)	TAK	NIE	Ciąg znaków określający opis filmu
add_time	znakowy czasu	TAK	NIE	Data określająca moment dodania filmu na platformę

Tabela 8: Tabela Credit_card

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_credit_card	całkowity	TAK	TAK	Wewnętrzny

				identyfikator credit_card
Card_number	Znakowe(max. 16)	TAK	TAK	Ciąg określający numer karty
Cvc	Znakowe(max. 3)	TAK	TAK	Ciąg określający Numer cvc danej karty
Expiration_date	Znakowe(max. 5)	TAK	TAK	Ciąg określający datę upływu terminu karty
Owner_name	Znakowe (max. 255)	TAK	TAK	Ciąg określający imię osoby do której należy karta
Owner_lastname	Znakowe (max. 255)	TAK	TAK	Ciąg określający nazwisko osoby do której należy karta

Tabela 9: Tabela Payment_method

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_payment_method	całkowity	TAK	TAK	Wewnętrzny identyfikator payment_method
Credit_cardid	całkowity	TAK	TAK	Liczba całkowita przypisująca do metody płatności daną kartę kredytową
Gift_cardid	całkowity	NIE	TAK	Liczba całkowita przypisująca do metody płatności daną kartę podarunkową

Tabela 10: Tabela Payment

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
------------	----------	-------------------------	-----------------------------	------

Id_payment	całkowity	TAK	TAK	Wewnętrzny identyfikator payment
Payment_date	Znakowy-czasu	TAK	TAK	Data określająca moment płatności
Payment_methodid	całkowity	TAK	TAK	Liczba całkowita przypisująca do płatności daną metodę płatności
Payment_plansid	całkowity	TAK	TAK	Liczba całkowita przypisująca do płatności dany plan płatności

Tabela 11: Tabela Payment_plans

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_payment_plans	całkowity	TAK	TAK	Wewnętrzny identyfikator payment_plans
name	Znakowe (max.45)	TAK	TAK	Ciąg znaków określający nazwę planu
price	całkowity	TAK	TAK	Liczba całkowita określająca cenę planu

Tabela 12: Tabela gift_card

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_gift_card	całkowity	TAK	TAK	Wewnętrzny identyfikator gift_card
code	Znakowe(max.15)	TAK	TAK	Ciąg znaków określający kod karty podarunkowej

Tabela 13: Tabela Serie

Nazwa Pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_serie	int4	tak	tak	Wewnętrzny identyfikator klienta
title	Varchar (100)	Tak	nie	Ciąg znaków określających tytuł filmu
poster	Varchar (100)	tak	nie	Ciąg znaków określających wydawcę
production_year	int4	tak	nie	Liczba określająca date produkcji
age_category	int4	tak	nie	Liczba całkowita będąca kluczem obcym. Odnosi się do tabeli age_category
season_number	int4	tak	nie	Liczba określająca ilość sezonów
adding_date	date	tak	nie	Data określająca datę dodania do platformy

Tabela 14: Tabela Episode

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
id_episode	int4	Tak	Tak	Wewnętrzny identyfikator odcinka
episode_title	Varchar(100)	Tak	Nie	Ciąg znaków określający tytuł odcinka
season	int4	tak	tak	Liczba określająca sezon do którego należy odcinek
about	Varchar(500)	Nie	Nie	Ciąg znaków

				opisujący odcinek
path	Varchar(100)	Tak	Tak	Ciąg znaków określający ścieżkę do pliku odcinka
number_of_episode	int4	Tak	nie	Liczba określająca numer odcinka
serieid	int4	Tak	nie	Liczba całkowita będąca kluczem obcym. Odnosi się do tabeli serie
How_long	int	tak	nie	Liczba określająca długość odcinka w minutach

Tabela 15: Tabela Age_category

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	opis
id_age_category	int4	TAK	TAK	Wewnętrzny identyfikator kategorii wiekowej
name	Varchar(100)	TAK	TAK	Opis kategorii wiekowej

Tabela 16: Tabela Device_type

Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	opis
Id_device_type	int4	TAK	TAK	Wewnętrzny identyfikator typu urządzenia
Device_type_name	Varchar(255)	TAK	Nie	Opis typu urządzenia

Tabela 17: Tabela Device

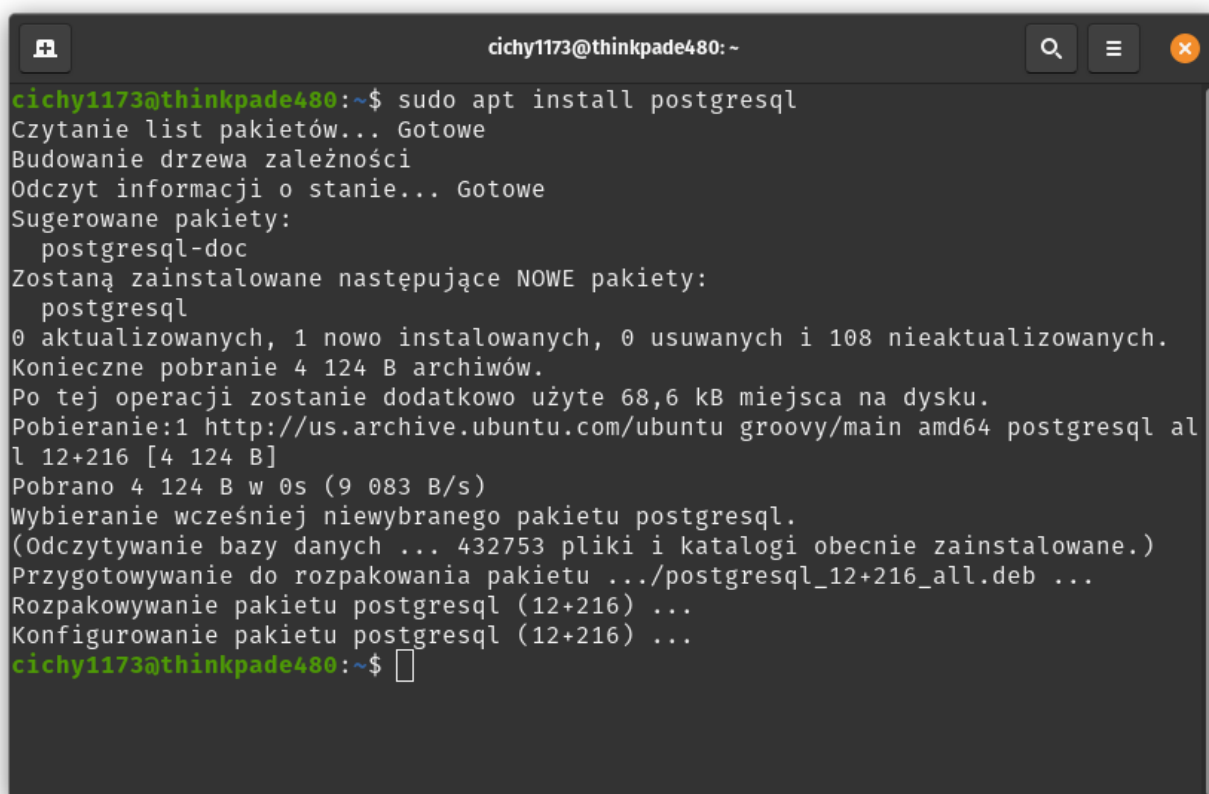
Nazwa pola	Typ pola	Czy pole jest wymagane?	Czy wartość jest unikatowa?	Opis
Id_device	Int4	Tak	Tak	Wewnętrzny identyfikator urządzenia
Device_name	Varchar(255)	Tak	nie	Ciąg znaków określający nazwę urządzenia
Ip_address	Varchar(20)	Tak	nie	Ciąg znaków określający adres ip urządzenia
Device_typeid	int4	tak	nie	Liczba całkowita będąca kluczem obcym. Odnosi się do tabeli device_type
regionid	int4	tak	nie	Liczba całkowita będąca kluczem obcym. Odnosi się do tabeli region

6. Opis instalacji bazy danych

Wybrany silnikiem bazy danych jest **PostgreSQL**. Wybór był jednomyślny i poparty wieloma zaletami. Przede wszystkim PostgreSQL cechuje się wysoką stabilnością i prostotą działania. Ponadto, wykazuje się dobrą wydajnością, zwłaszcza kiedy baza danych jest pełna naprawdę wielu rekordów, a docelowo Netflix faktycznie może mieć bardzo dużo danych. Ponadto jest to projekt Open Source, a więc można go dowolnie wykorzystywać i modyfikować. Finalnie pod uwagę wzięliśmy jeszcze jedną rzecz, a jest to fakt, że sam Netflix wykorzystuje właśnie PostgreSQL.

Baza danych została zainstalowana na systemie GNU/Linux Pop!_OS 20.10 z wykorzystaniem narzędzia z graficznym interfejsem użytkownika pgAdmin 4.

Najpierw musimy zainstalować silnik bazy danych. Najszybszym i najbardziej uniwersalnym sposobem jest wykorzystanie terminala. Należy uruchomić terminal i wpisać komendę `sudo apt install postgresql`.



```
cichy1173@thinkpade480: ~  
cichy1173@thinkpade480:~$ sudo apt install postgresql  
Czytanie list pakietów... Gotowe  
Budowanie drzewa zależności  
Odczyt informacji o stanie... Gotowe  
Sugerowane pakiety:  
  postgresql-doc  
Zostaną zainstalowane następujące NOWE pakiety:  
  postgresql  
0 aktualizowanych, 1 nowo instalowanych, 0 usuwanych i 108 nieaktualizowanych.  
Konieczne pobranie 4 124 B archiwów.  
Po tej operacji zostanie dodatkowo użyte 68,6 kB miejsca na dysku.  
Pobieranie:1 http://us.archive.ubuntu.com/ubuntu groovy/main amd64 postgresql al  
l 12+216 [4 124 B]  
Pobrano 4 124 B w 0s (9 083 B/s)  
Wybieranie wcześniej niewybranego pakietu postgresql.  
(Odczytywanie bazy danych ... 432753 pliki i katalogi obecnie zainstalowane.)  
Przygotowywanie do rozpakowania pakietu ../postgresql_12+216_all.deb ...  
Rozpakowywanie pakietu postgresql (12+216) ...  
Konfigurowanie pakietu postgresql (12+216) ...  
cichy1173@thinkpade480:~$
```

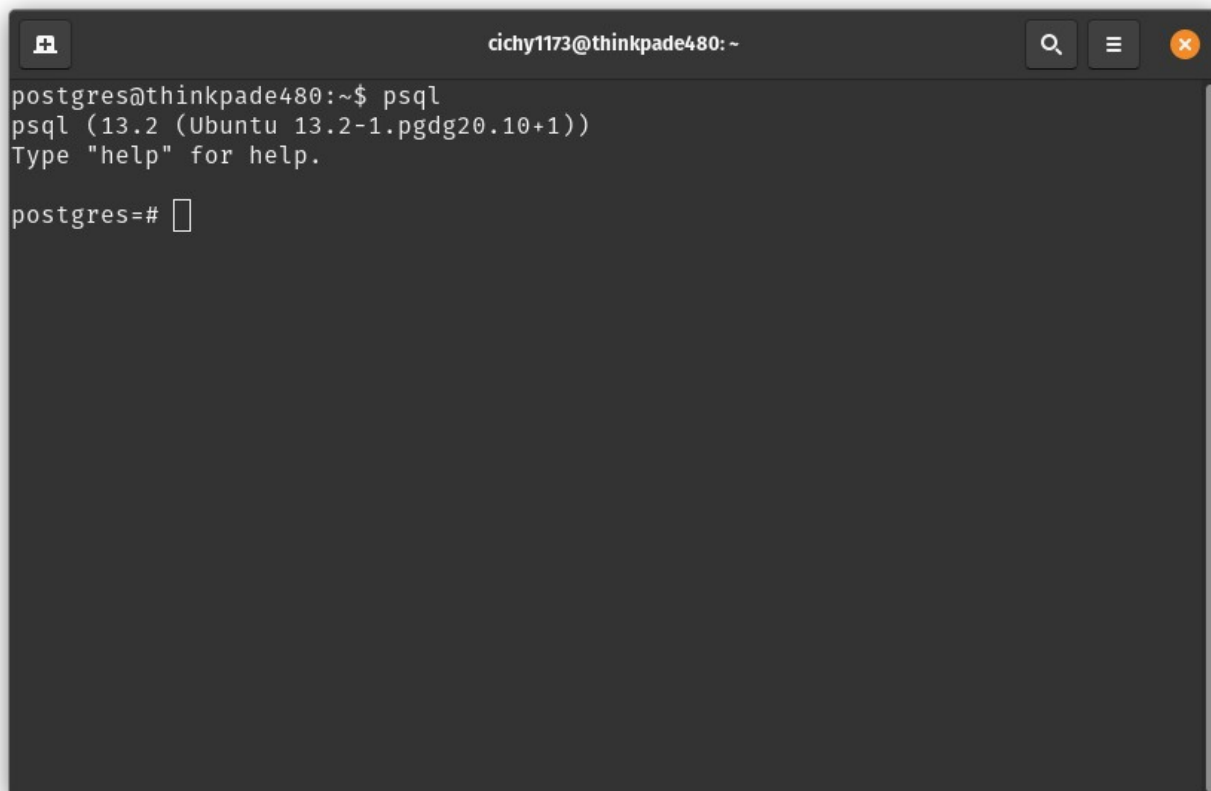
Fot. 4: Instalacja PostgreSQL w terminalu

Następnie należy zainstalować narzędzie pgAdmin 4. Również można wpisać odpowiednie komendy. Najpierw dodajemy repozytorium, a następnie uruchamiamy instalację. Komendy wyglądają następująco:

```
sudo curl https://www.pgadmin.org/static/packages_pgadmin_org.pub | sudo apt-key add  
sudo sh -c 'echo "deb https://ftp.postgresql.org/pub/pgadmin/pgadmin4/apt/$(lsb_release -cs)
```

```
pgadmin4 main" > /etc/apt/sources.list.d/pgadmin4.list && apt update'  
sudo apt install pgadmin4
```

Po tym kroku wystarczy uruchomić PostgreSQL wpisując tylko komendę `sudo -i -u postgres`. Uruchomi się linia poleceń z uprawnieniami użytkownika „postgres”. Wpisując `psql` uzyskujemy dostęp do wiersza poleceń PostgreSQL.

A screenshot of a terminal window with a dark background. The window title is 'cichy1173@thinkpade480: ~'. The terminal shows the following text: 'postgres@thinkpade480:~\$ psql', 'psql (13.2 (Ubuntu 13.2-1.pgdg20.10+1))', 'Type "help" for help.', and 'postgres=#' followed by a cursor. The window has standard Ubuntu window controls (minimize, maximize, close) in the top right corner.

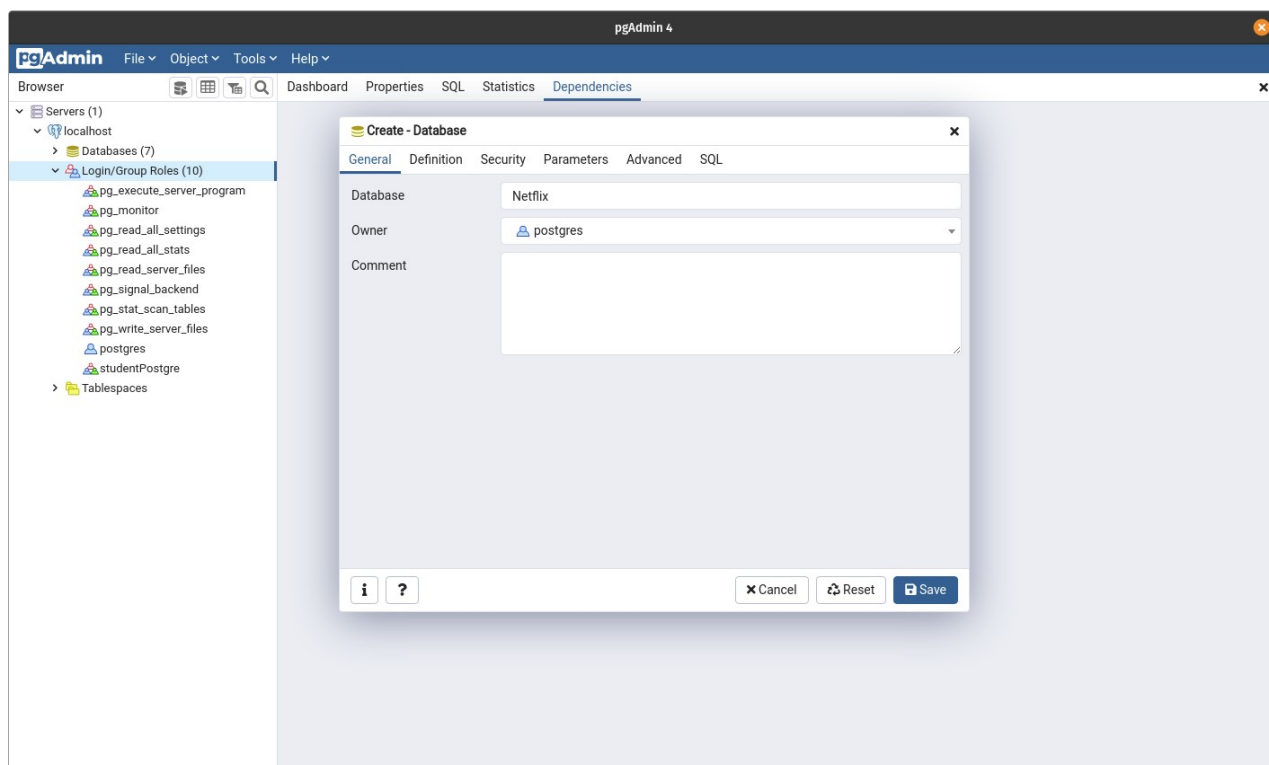
```
postgres@thinkpade480:~$ psql  
psql (13.2 (Ubuntu 13.2-1.pgdg20.10+1))  
Type "help" for help.  
  
postgres=#
```

Fot. 5: Wiersz poleceń PostgreSQL

Za pomocą odpowiednich komend możemy również utworzyć nowego użytkownika.

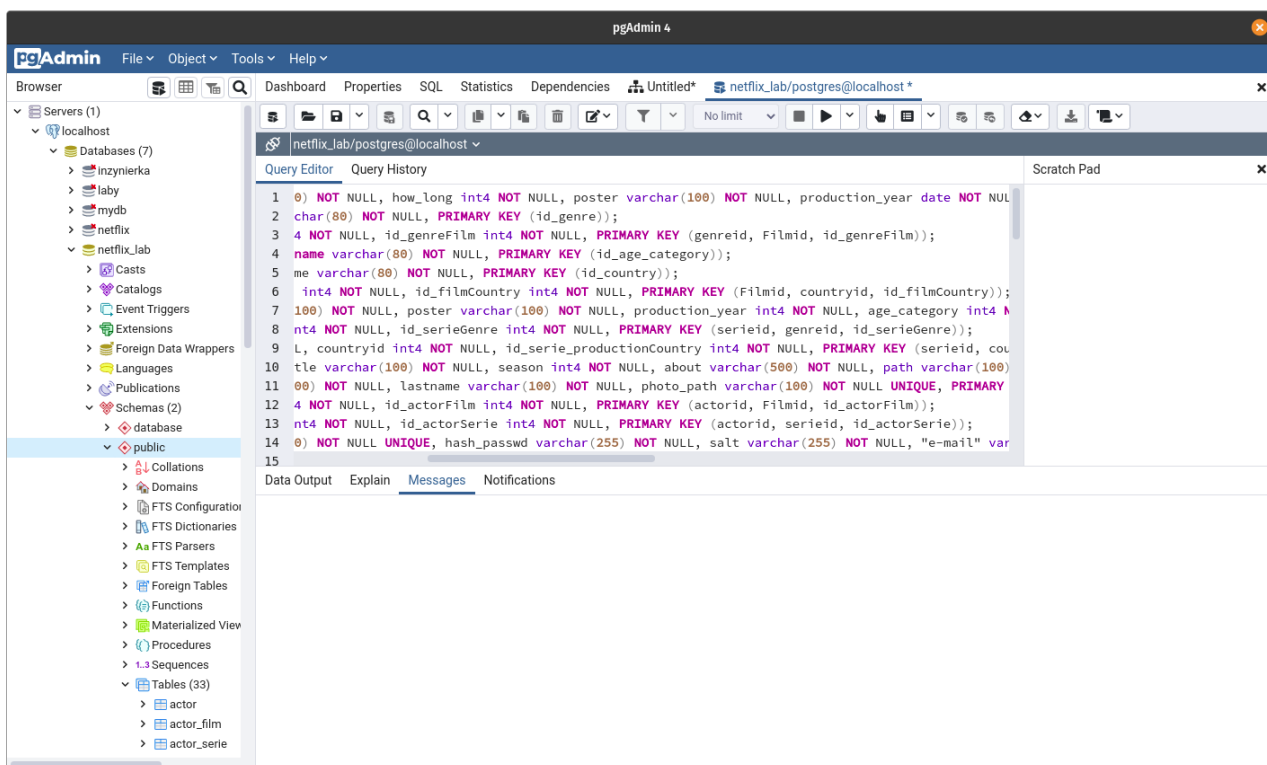
W tym momencie można przejść do narzędzia pgAdmin 4. Aplikacja powinna automatycznie nawiązać połączenie z serwerem. Jeśli nie, to należy ręcznie wpisać ustanowioną nazwę użytkownika oraz hasło. Po chwili powinno pojawić się powiadomienie informujące o połączeniu.

Z drzewa po lewej stronie, należy zaznaczyć „Databases” oraz wybrać „Create -> Database” Należy nadać nazwę bazy danych „Netflix”.

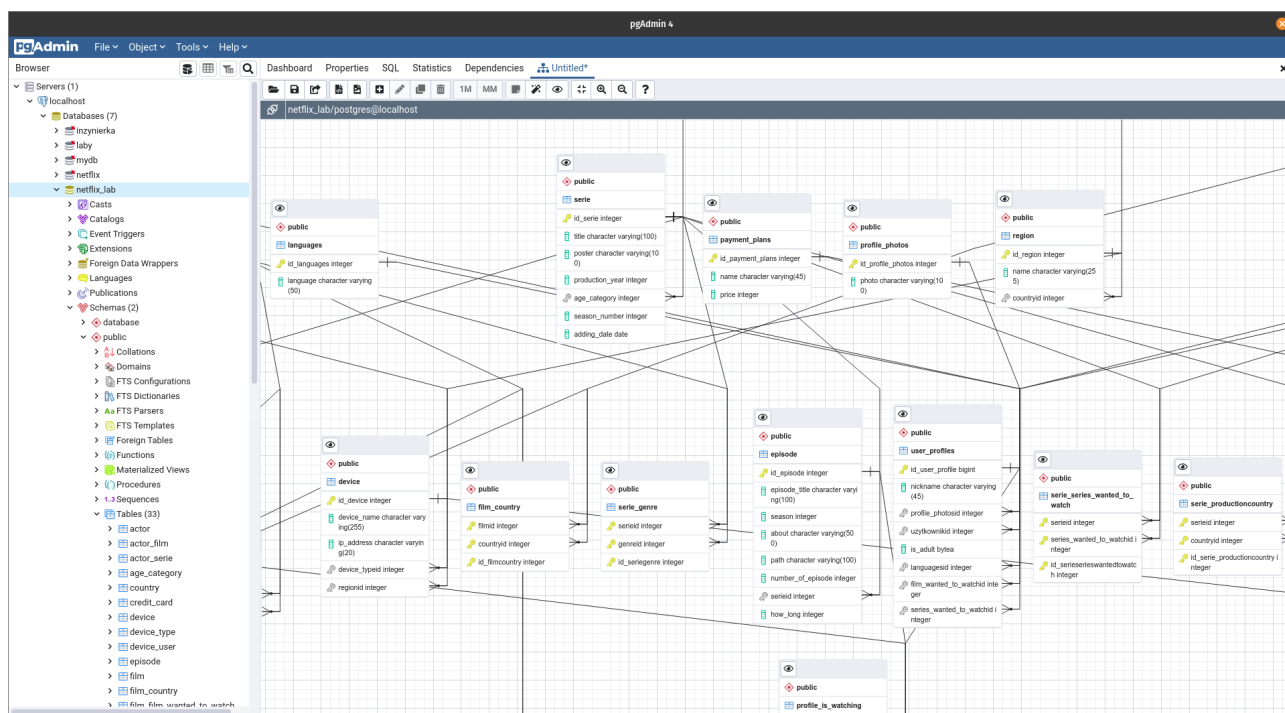


Fot. 6: Tworzenie nowej bazy danych

Następnie trzeba rozwinąć gałąź z nową bazą danych i najechać na schemat *public*. Tutaj należy kliknąć PPM i wybrać „Query Tool”. Uruchomi się linia poleceń. Należy wkleić kod SQL, który tworzy całą strukturę bazy danych. Po chwili powinny utworzyć się wszystkie tabele wraz ze związkami. Można również użyć opcji „Generate ERD”, aby podejrzeć diagram związków encji.



Fot. 7: Tworzenie tabel i związków w bazie danych



Fot. 8: Podgląd wygenerowanego diagramu ERD

Następnie trzeba wykorzystać wcześniej otwarte „Query Tool” do wprowadzenia danych do tabel. Odpowiednia kolejność dodawania została podana poniżej.

7. Kod SQL potrzebny do stworzenia bazy danych

Poniżej znajduje się kod SQL.

```
CREATE TABLE film (id_Film int4 NOT NULL, title varchar(100) NOT NULL, how_long int4 NOT NULL, poster varchar(100) NOT NULL, production_year date NOT NULL, path_to_file varchar(100) NOT NULL UNIQUE, age_categoryid int4 NOT NULL, about varchar(500) NOT NULL, add_time timestamp NOT NULL, PRIMARY KEY (id_Film));
```

```
CREATE TABLE genre (id_genre int4 NOT NULL, genre_name varchar(80) NOT NULL, PRIMARY KEY (id_genre));
```

```
CREATE TABLE genre_film (genreid int4 NOT NULL, Filmid int4 NOT NULL, id_genreFilm int4 NOT NULL, PRIMARY KEY (genreid, Filmid, id_genreFilm));
```

```
CREATE TABLE age_category (id_age_category int4 NOT NULL, name varchar(80) NOT NULL, PRIMARY KEY (id_age_category));
```

```
CREATE TABLE country (id_country int4 NOT NULL, country_name varchar(80) NOT NULL, PRIMARY KEY (id_country));
```

```
CREATE TABLE film_country (Filmid int4 NOT NULL, countryid int4 NOT NULL, id_filmCountry int4 NOT NULL, PRIMARY KEY (Filmid, countryid, id_filmCountry));
```

```
CREATE TABLE serie (id_serie int4 NOT NULL, title varchar(100) NOT NULL, poster varchar(100) NOT NULL, production_year int4 NOT NULL, age_category int4 NOT NULL, season_number int4 NOT NULL, adding_date date NOT NULL, PRIMARY KEY (id_serie));
```

```
CREATE TABLE serie_genre (serieid int4 NOT NULL, genreid int4 NOT NULL, id_serieGenre int4 NOT NULL, PRIMARY KEY (serieid, genreid, id_serieGenre));
```

```
CREATE TABLE serie_productionCountry (serieid int4 NOT NULL, countryid int4 NOT NULL, id_serie_productionCountry int4 NOT NULL, PRIMARY KEY (serieid, countryid, id_serie_productionCountry));
```

```
CREATE TABLE episode (id_episode int4 NOT NULL, episode_title varchar(100) NOT NULL, season int4 NOT NULL, about varchar(500) NOT NULL, path varchar(100) NOT NULL UNIQUE, number_of_episode int4 NOT NULL, serieid int4 NOT NULL, how_long int4 NOT NULL, PRIMARY KEY (id_episode));
```

```
CREATE TABLE actor (id_actor int4 NOT NULL, name varchar(100) NOT NULL, lastname varchar(100) NOT NULL, photo_path varchar(100) NOT NULL UNIQUE, PRIMARY KEY (id_actor));
```

```
CREATE TABLE actor_Film (actorid int4 NOT NULL, Filmid int4 NOT NULL, id_actorFilm int4 NOT NULL, PRIMARY KEY (actorid, Filmid, id_actorFilm));
```

```
CREATE TABLE actor_serie (actorid int4 NOT NULL, serieid int4 NOT NULL, id_actorSerie int4 NOT NULL, PRIMARY KEY (actorid, serieid, id_actorSerie));
```

```
CREATE TABLE "user" (id_user int4 NOT NULL, nick varchar(50) NOT NULL UNIQUE, hash_passwd varchar(255) NOT NULL, salt varchar(255) NOT NULL, "e-mail" varchar(150) NOT NULL UNIQUE, age date NOT NULL, countryid int4 NOT NULL, phone_number varchar(12), PRIMARY KEY (id_user));
```

```
CREATE TABLE credit_card (id_credit_card int4 NOT NULL, card_number varchar(16) NOT NULL, cvc varchar(3) NOT NULL, expiration_date varchar(5) NOT NULL, owner_name varchar(255) NOT NULL, owner_lastname varchar(255) NOT NULL, PRIMARY KEY (id_credit_card));
```

```
CREATE TABLE user_profiles (id_user_profile int8 NOT NULL, nickname varchar(45) NOT NULL, profile_photosid int4 NOT NULL, Uzytkownikid int4 NOT NULL, is_adult bytea NOT NULL, Languagesid int4 NOT NULL, Film_Wanted_to_watchid int4, Series_wanted_to_watchid int4, PRIMARY KEY (id_user_profile));
```

```
CREATE TABLE profile_photos (id_profile_photos int4 NOT NULL, photo varchar(100) NOT NULL, PRIMARY KEY (id_profile_photos));
```

```
CREATE TABLE profile_is_watching (id_session int4 NOT NULL, User_profilesid int8 NOT NULL, Episode_serieid int4, Filmid int4, "date" timestamp NOT NULL, PRIMARY KEY (id_session));
```

```
CREATE TABLE languages (id_languages int4 NOT NULL, language varchar(50) NOT NULL, PRIMARY KEY (id_languages));
```

```
CREATE TABLE payment_plans (id_payment_plans int4 NOT NULL, name varchar(45) NOT NULL, price int4 NOT NULL, PRIMARY KEY (id_payment_plans));
```

```
CREATE TABLE gift_card (id_gift_card int4 NOT NULL, code varchar(15) NOT NULL, PRIMARY KEY (id_gift_card));
```

```
CREATE TABLE payment (id_payment int4 NOT NULL, payment_date timestamp NOT NULL, payment_methodid int4 NOT NULL, payment_plansid int4 NOT NULL, PRIMARY KEY (id_payment));
```

```
CREATE TABLE payment_method (id_payment_method int4 NOT NULL, credit_cardid int4 NOT NULL, gift_cardid int4, PRIMARY KEY (id_payment_method));
```

```
CREATE TABLE user_payment_method (Userid int4 NOT NULL, payment_methodid int4 NOT NULL, id_userPaymentMethod int4 NOT NULL, PRIMARY KEY (Userid, payment_methodid, id_userPaymentMethod));
```

```
CREATE TABLE device (id_device int4 NOT NULL, device_name varchar(255) NOT NULL, ip_address varchar(20) NOT NULL, device_typeid int4 NOT NULL, Regionid int4 NOT NULL, PRIMARY KEY (id_device));
```

```
CREATE TABLE device_user (Deviceid int4 NOT NULL, userid int4 NOT NULL, id_device_user int4 NOT NULL, last_used timestamp NOT NULL, PRIMARY KEY (Deviceid, userid, id_device_user));
```

```
CREATE TABLE device_type (id_device_type int4 NOT NULL, device_type_name varchar(255) NOT NULL, PRIMARY KEY (id_device_type));
```

```
CREATE TABLE Region (id_region int4 NOT NULL, name varchar(255) NOT NULL, countryid int4 NOT NULL, PRIMARY KEY (id_region));
```

```
CREATE TABLE film_Wanted_to_watch (id_Film_wanted_to_watch SERIAL NOT NULL, PRIMARY KEY (id_Film_wanted_to_watch));
```

```
CREATE TABLE series_wanted_to_watch (id_series_wanted_to_watch SERIAL NOT NULL,  
PRIMARY KEY (id_series_wanted_to_watch));
```

```
CREATE TABLE film_Film_Wanted_to_watch (Filmid int4 NOT NULL,  
Film_Wanted_to_watchid int4 NOT NULL, id_FilmFilmWantedtoWatch int4 NOT NULL,  
PRIMARY KEY (Filmid, Film_Wanted_to_watchid, id_FilmFilmWantedtoWatch));
```

```
CREATE TABLE serie_series_wanted_to_watch (Serieid int4 NOT NULL,  
Series_wanted_to_watchid int4 NOT NULL, id_SerieSeriesWantedtoWatch int4 NOT NULL,  
PRIMARY KEY (Serieid, Series_wanted_to_watchid, id_SerieSeriesWantedtoWatch));
```

```
CREATE TABLE watching_history (id_watchingHistory int4 NOT NULL,  
profile_is_watchingid_session int4 NOT NULL, stop_watching_moment varchar(50) NOT NULL,  
PRIMARY KEY (id_watchingHistory));
```

```
ALTER TABLE genre_film ADD CONSTRAINT FKgenre_film71472 FOREIGN KEY (genreid)  
REFERENCES genre (id_genre) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE genre_film ADD CONSTRAINT FKgenre_film139093 FOREIGN KEY (Filmid)  
REFERENCES film (id_Film) ON UPDATE Cascade ON DELETE Set default;
```

```
ALTER TABLE film ADD CONSTRAINT FKfilm23698 FOREIGN KEY (age_categoryid)  
REFERENCES age_category (id_age_category) ON UPDATE Cascade ON DELETE Set default;
```

```
ALTER TABLE film_country ADD CONSTRAINT FKfilm_count414041 FOREIGN KEY  
(Filmid) REFERENCES film (id_Film) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE film_country ADD CONSTRAINT FKfilm_count669570 FOREIGN KEY  
(countryid) REFERENCES country (id_country) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE serie ADD CONSTRAINT FKserie922359 FOREIGN KEY (age_category)  
REFERENCES age_category (id_age_category) ON UPDATE Cascade ON DELETE Set null;
```

```
ALTER TABLE serie_genre ADD CONSTRAINT FKserie_genr558468 FOREIGN KEY (serieid)  
REFERENCES serie (id_serie) ON UPDATE No action ON DELETE No action;
```

```
ALTER TABLE serie_genre ADD CONSTRAINT FKserie_genr703857 FOREIGN KEY (genreid)  
REFERENCES genre (id_genre) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE serie_productionCountry ADD CONSTRAINT FKserie_prod318377 FOREIGN  
KEY (serieid) REFERENCES serie (id_serie) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE serie_productionCountry ADD CONSTRAINT FKserie_prod379201 FOREIGN  
KEY (countryid) REFERENCES country (id_country) ON UPDATE Cascade ON DELETE No  
action;
```

```
ALTER TABLE episode ADD CONSTRAINT FKepisode293403 FOREIGN KEY (serieid)  
REFERENCES serie (id_serie) ON UPDATE Cascade ON DELETE Cascade;
```

```
ALTER TABLE actor_Film ADD CONSTRAINT FKactor_Film524896 FOREIGN KEY (actorid)  
REFERENCES actor (id_actor) ON UPDATE Cascade ON DELETE No action;
```

```
ALTER TABLE actor_Film ADD CONSTRAINT FKactor_Film474332 FOREIGN KEY (Filmid)  
REFERENCES film (id_Film) ON UPDATE Cascade ON DELETE No action;
```

ALTER TABLE actor_serie ADD CONSTRAINT FKactor_seri794137 FOREIGN KEY (actorid) REFERENCES actor (id_actor) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE actor_serie ADD CONSTRAINT FKactor_seri428205 FOREIGN KEY (serieid) REFERENCES serie (id_serie) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE "user" ADD CONSTRAINT FKuser271711 FOREIGN KEY (countryid) REFERENCES country (id_country) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE user_profiles ADD CONSTRAINT FKuser_profi110750 FOREIGN KEY (profile_photosid) REFERENCES profile_photos (id_profile_photos) ON UPDATE No action ON DELETE Set default;

ALTER TABLE user_profiles ADD CONSTRAINT FKuser_profi985153 FOREIGN KEY (Uzytkownikid) REFERENCES "user" (id_user);

ALTER TABLE profile_is_watching ADD CONSTRAINT FKprofile_is79705 FOREIGN KEY (User_profilesid) REFERENCES user_profiles (id_user_profile) ON UPDATE Cascade ON DELETE Cascade;

ALTER TABLE profile_is_watching ADD CONSTRAINT FKprofile_is443338 FOREIGN KEY (Filmid) REFERENCES film (id_Film) ON UPDATE No action ON DELETE No action;

ALTER TABLE profile_is_watching ADD CONSTRAINT FKprofile_is993269 FOREIGN KEY (Episode_serieid) REFERENCES episode (id_episode) ON UPDATE No action ON DELETE No action;

ALTER TABLE user_profiles ADD CONSTRAINT FKuser_profi624622 FOREIGN KEY (Languagesid) REFERENCES languages (id_languages) ON UPDATE No action ON DELETE Set default;

ALTER TABLE payment_method ADD CONSTRAINT FKpayment_me53348 FOREIGN KEY (credit_cardid) REFERENCES credit_card (id_credit_card) ON UPDATE Cascade ON DELETE Set default;

ALTER TABLE payment_method ADD CONSTRAINT FKpayment_me658051 FOREIGN KEY (gift_cardid) REFERENCES gift_card (id_gift_card) ON UPDATE Cascade ON DELETE Cascade;

ALTER TABLE payment ADD CONSTRAINT FKpayment199132 FOREIGN KEY (payment_methodid) REFERENCES payment_method (id_payment_method) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE payment ADD CONSTRAINT FKpayment7431 FOREIGN KEY (payment_plansid) REFERENCES payment_plans (id_payment_plans) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE user_payment_method ADD CONSTRAINT FKuser_payme781594 FOREIGN KEY (Userid) REFERENCES "user" (id_user) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE user_payment_method ADD CONSTRAINT FKuser_payme381628 FOREIGN KEY (payment_methodid) REFERENCES payment_method (id_payment_method) ON UPDATE Cascade ON DELETE No action;


```

ALTER TABLE device_user ADD CONSTRAINT FKdevice_use333048 FOREIGN KEY
(Deviceid) REFERENCES device (id_device) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE device_user ADD CONSTRAINT FKdevice_use421854 FOREIGN KEY (userid)
REFERENCES "user" (id_user) ON UPDATE Cascade ON DELETE Cascade;

ALTER TABLE device ADD CONSTRAINT FKdevice918743 FOREIGN KEY (device_typeid)
REFERENCES device_type (id_device_type) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE Region ADD CONSTRAINT FKRegion257814 FOREIGN KEY (countryid)
REFERENCES country (id_country) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE device ADD CONSTRAINT FKdevice920182 FOREIGN KEY (Regionid)
REFERENCES Region (id_region) ON UPDATE Cascade ON DELETE No action;

ALTER TABLE user_profiles ADD CONSTRAINT FKuser_profi295817 FOREIGN KEY
(Film_Wanted_to_watchid) REFERENCES film_Wanted_to_watch (id_Film_wanted_to_watch)
ON UPDATE Cascade ON DELETE Cascade;

ALTER TABLE user_profiles ADD CONSTRAINT FKuser_profi532880 FOREIGN KEY
(Series_wanted_to_watchid) REFERENCES series_wanted_to_watch (id_series_wanted_to_watch)
ON UPDATE Cascade ON DELETE Set null;

ALTER TABLE film_Film_Wanted_to_watch ADD CONSTRAINT FKfilm_Film_791760
FOREIGN KEY (Filmid) REFERENCES film (id_Film) ON UPDATE Cascade ON DELETE No
action;

ALTER TABLE film_Film_Wanted_to_watch ADD CONSTRAINT FKfilm_Film_844480
FOREIGN KEY (Film_Wanted_to_watchid) REFERENCES film_Wanted_to_watch
(id_Film_wanted_to_watch) ON UPDATE Cascade ON DELETE Cascade;

ALTER TABLE serie_series_wanted_to_watch ADD CONSTRAINT FKserie_seri134189
FOREIGN KEY (Serieid) REFERENCES serie (id_serie) ON UPDATE No action ON DELETE
No action;

ALTER TABLE serie_series_wanted_to_watch ADD CONSTRAINT FKserie_seri73619
FOREIGN KEY (Series_wanted_to_watchid) REFERENCES series_wanted_to_watch
(id_series_wanted_to_watch) ON UPDATE Cascade ON DELETE Cascade;

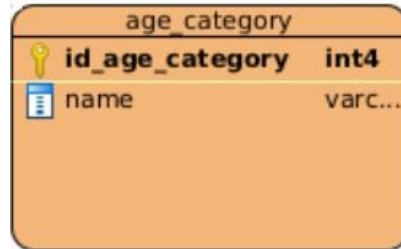
ALTER TABLE watching_history ADD CONSTRAINT FKwatching_h70704 FOREIGN KEY
(profile_is_watchingid_session) REFERENCES profile_is_watching (id_session) ON UPDATE
Cascade ON DELETE No action;

```

8. Wypełnianie tabel danymi

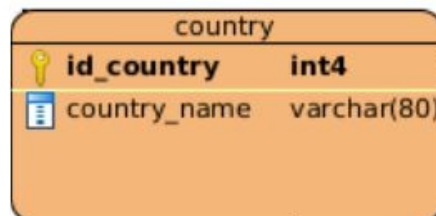
Wypełnianie danych zostało przedstawione w odpowiedniej kolejności, dzięki czemu nie powinny pojawić się żadne problemy.

1. age_category



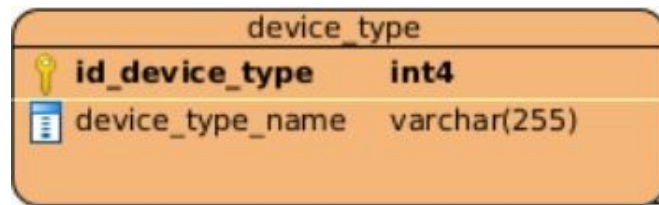
```
INSERT INTO public.age_category(name) VALUES ('all');
INSERT INTO public.age_category(name) VALUES ('7+');
INSERT INTO public.age_category(name) VALUES ('13+');
INSERT INTO public.age_category(name) VALUES ('16+');
INSERT INTO public.age_category(name) VALUES ('18+');
```

2. country



```
INSERT INTO public.country(country_name) VALUES ('Polska');
INSERT INTO public.country(country_name) VALUES ('Anglia');
INSERT INTO public.country(country_name) VALUES ('Hiszpania');
INSERT INTO public.country(country_name) VALUES ('Rosja');
INSERT INTO public.country(country_name) VALUES ('Wlochy');
INSERT INTO public.country(country_name) VALUES ('Niemcy');
INSERT INTO public.country(country_name) VALUES ('Francja');
INSERT INTO public.country(country_name) VALUES ('Grecja');
INSERT INTO public.country(country_name) VALUES ('USA');
INSERT INTO public.country(country_name) VALUES ('Szwecja');
```

3. device_type



```
INSERT INTO public.device_type(device_type_name) VALUES ('smartfon');
INSERT INTO public.device_type(device_type_name) VALUES ('tablet');
INSERT INTO public.device_type(device_type_name) VALUES ('komputer');
INSERT INTO public.device_type(device_type_name) VALUES ('laptop');
INSERT INTO public.device_type(device_type_name) VALUES ('telewizor');
```

4. serie



```
INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Czarna lista','\plakaty\serie\czarna_lista.jpg',2015,4,8,'2020-12-12');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Breaking Bad','\plakaty\serie\breaking_bad.jpg',2013,4,5,'2020-11-11');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('House of Cards','\plakaty\serie\house_of_cards.jpg',2014,4,6,'2021-11-12');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Formula 1: Jazda o zycie','\plakaty\serie\formula1.jpg',2018,4,3,'2019-01-01');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Homeland','\plakaty\serie\homeland.jpg',2011,4,8,'2018-04-04');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Jezioro marzeń','\plakaty\serie\jezioro_marzeń.jpg',2012,3,6,'2018-01-01');

INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('H2O, wystarczy kropla','\plakaty\serie\h2o_wystarczy_kropla.jpg',2013,2,3,'2021-04-04');
```

```
INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Psi Patrol','\plakaty\serie\psi_patrol.jpg',2012,1,3,'2020-03-06');
```

```
INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Peaky Blinders','\plakaty\serie\peaky_blinders.jpg',2015,4,5,'2019-06-06');
```

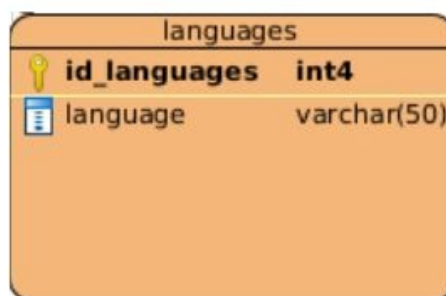
```
INSERT INTO public.serie(title,poster,production_year,age_category,season_number,adding_date)
VALUES ('Riverdale','\plakaty\serie\peaky_blinders.jpg',2019,4,5,'2020-10-10');
```

5. gift_card



```
INSERT INTO public.gift_card(code) VALUES ('9322-0012-6798');
INSERT INTO public.gift_card(code) VALUES ('8254-1860-3617');
INSERT INTO public.gift_card(code) VALUES ('1047-8943-5679');
INSERT INTO public.gift_card(code) VALUES ('5079-1759-6826');
INSERT INTO public.gift_card(code) VALUES ('6748-4870-1285');
INSERT INTO public.gift_card(code) VALUES ('9016-5870-4658');
INSERT INTO public.gift_card(code) VALUES ('6591-7920-1476');
INSERT INTO public.gift_card(code) VALUES ('4769-0175-2347');
INSERT INTO public.gift_card(code) VALUES ('0386-2856-1293');
INSERT INTO public.gift_card(code) VALUES ('9754-8045-1768');
```

6. languages

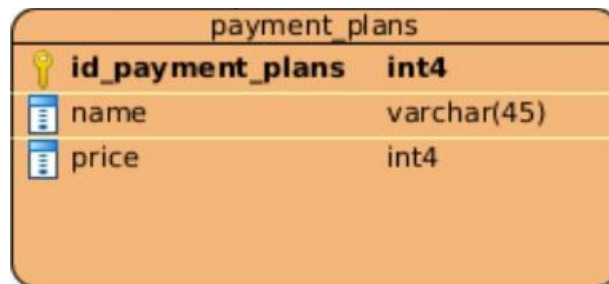


```

INSERT INTO public.languages(language) VALUES ('Polski');
INSERT INTO public.languages(language) VALUES ('Angielski(brytyjski)');
INSERT INTO public.languages(language) VALUES ('Hiszpanski');
INSERT INTO public.languages(language) VALUES ('Rosyjski');
INSERT INTO public.languages(language) VALUES ('Wloski');
INSERT INTO public.languages(language) VALUES ('Niemiecki');
INSERT INTO public.languages(language) VALUES ('Francuski');
INSERT INTO public.languages(language) VALUES ('Grecki');
INSERT INTO public.languages(language) VALUES ('Angielski(amerykanski)');
INSERT INTO public.languages(language) VALUES ('Szwedzki');

```

7. payment_plans

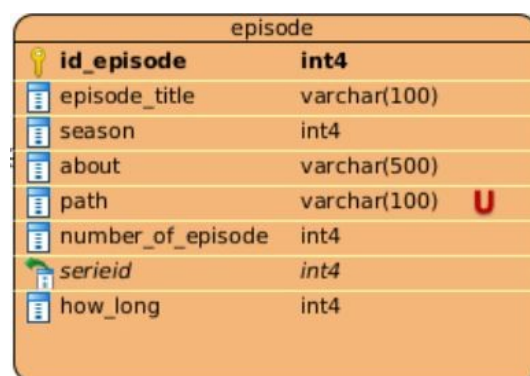


```

INSERT INTO public.payment_plans(name,price) VALUES ('Podstawowy',34);
INSERT INTO public.payment_plans(name,price) VALUES ('Standard',43);
INSERT INTO public.payment_plans(name,price) VALUES ('Premium',52);
INSERT INTO public.payment_plans(name,price) VALUES ('Mobilny',30);
INSERT INTO public.payment_plans(name,price) VALUES ('Ilosciowy',28);

```

8. episode



INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Wolny strzelec(nr 145)',1,'Liz i Red tropią mordercę pozorującego katastrofy w celu
ukrycia swych zabójstw. Departament Sprawiedliwości usilnie stara się zrozumieć motyw
Reda.','serieale\czarna lista\czarnalistaS1E2.mkv',2,1,40,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Wuijing(nr 84)',1,'Liz i Red działają incognito, by unieszkodliwić zabójcę
dziesiątkującego szeregi agentów CIA. Tom pyta Liz o człowieka, który go zaatakował.','serieale\
czarna lista\czarnalistaS1E3.mkv',3,1,42,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Pilot',1,'Nauczyciel chemii w szkole średniej dowiaduje się, że ma nieuleczalnego raka
płuc. Aby zapewnić byt rodzinie, bierze się za produkcję i rozprowadzanie metamfetaminy.','serieale\
breaking bad\sezon1\breakingbadS1E1.mkv',1,2,58,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Ptaszek w klatce',1,'Po pierwszej nieudanej transakcji Walt i Jesse muszą się pozbyć
dwóch trupów. Tymczasem Skyler podejrzewa, że jej mąż coś knuje.','serieale\breaking bad\sezon1\
breakingbadS1E2.mkv',2,2,48,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Rozdział 1',1,'Gdy nowo wybrany prezydent nie dotrzymuje obietnicy, Frank i Claire
postanawiają zapomnieć o lojalności i wszelkich zasadach.','serieale\house of cards\sezon1\
houseofcardsS1E1.mkv',1,3,NULL,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Rozdział 2',1,'Z pomocą młodej dziennikarki, Zoe Barnes, Frank nagłaśnia sprawę, która
wywołuje poruszenie w Białym Domu. Claire przeprowadza drastyczne cięcia w fundacji.','serieale\
houseofcardsS1E1\sezon1\houseofcardsS1E2.mkv',2,3,49,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Ruszamy!',2,'Nowy sezon Formuły 1 otwiera wyścig Grand Prix Australii. Dziesięć
zespołów toczy zjadłą rywalizację o miejsce na podium.','serieale\formula1\sezon2\
formula1S2E1.mkv',1,4,33,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Łaska',1,'Saudyjski książę prawdopodobnie finansuje działania terrorystów, a agentka
działająca pod przykryciem w jego otoczeniu dostarcza Carrie potwierdzające to dowody.','serieale\
homeland\sezon1\homelandS1E1.mkv',1,5,55,NULL);
```

INSERT INTO

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
```

```
VALUES ('Dziecko',1,'Romans Pacea i Tamary zostaje ujawniony. Rodząca Bessie musi skorzystać z pomocy babci Jen.','seriale\jezioromarzen\sezon1\jezioromarzen.mkv',1,6,42,NULL);
```

```
INSERT INTO
```

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Czy to rekiny?',1,'Dziewczyny dowiadują się, że morskie żółwie wpadają w rybackie sieci. Rikki i Emma spieszą więc z pomocą, przez co rybacy sądzą, że atakują ich rekiny.','seriale\h2o\sezon1\h2oS1E3.mkv',3,7,25,NULL);
```

```
INSERT INTO
```

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Pieski ratują sterowiec/Pieski ratują konkurs gotowania',1,'Burmistrz Humdinger uważa, że sterowiec wpływa na pogodę, dlatego postanawia go porwać. Zatoka Przygód i Mglista Osada stają w szranki na pokazie gotowania.','seriale\psipatrol\sezon1\psipatrolS1E1',1,8,23,NULL);
```

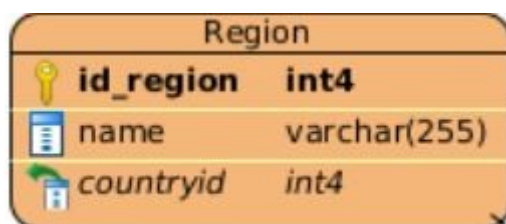
```
INSERT INTO
```

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Odcinek 6',1,'Thomas szykuje się do odsunięcia Kimbera od władzy, ale rodzinne tajemnice, które wychodzą na jaw, oraz owładnięty obsesją Campbell mogą stanąć mu na przeszkodzie.','seriale\peakyblindrs\sezon1\peakyblindrsS1E6',6,9,54,NULL);
```

```
INSERT INTO
```

```
public.episode(episode_title,season,about,path,number_of_episode,serieid,how_long,FIELD8)
VALUES ('Rozdział dwudziesty trzeci: Szkolna dżungla',1,'Kiedy uczniowie ze szkoły Southside zostają przeniesieni do Riverdale High, wybucha konflikt kulturowy. FBI rozmawia z Archiem. Betty poszukuje dawno zaginionego brata.','seriale\riverdale\sezon1\riverdaleS1E10',10,10,42,NULL);
```

9. region



```
INSERT INTO public.Region(name,countryid) VALUES ('województwo świętokrzyskie',1);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('województwo opolskie',1);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('województwo małopolskie',1);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('Andaluzja',3);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('Katalonia',3);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('Saksonia',6);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('Brema',6);
```

```
INSERT INTO public.Region(name,countryid) VALUES ('Sycylia',5);
```

INSERT INTO public.Region(name,countryid) VALUES ('Toskania',5);

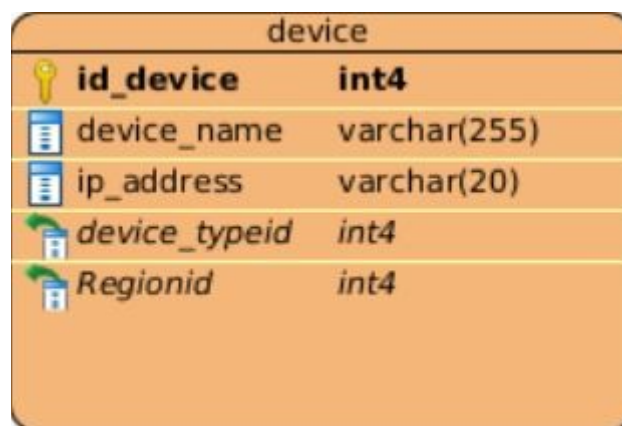
INSERT INTO public.Region(name,countryid) VALUES ('Abruzja',5);

10. profile_photos



INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\wozek.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\pingwin.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\trol.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\balwan.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\maska.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\donut.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\slonce.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\ksiezyc.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\mars.jpg');
INSERT INTO public.profile_photos(photo) VALUES ('\users\profilepictures\ziemia.jpg');

11. Device




```

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Chrome
PC (Cadmium)','83.26.217.57',3,1);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Safari
MAC (Cadmium)','83.28.89.207',4,2);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Apple
iPhone XR','83.28.89.20',1,3);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Samsung
2018 Kant-M2 UHD TV (1.5G) Smart TV','31.0.123.52',5,4);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Philips
2K18 PHTV18AMT1 Smart TV','31.0.123.52',5,5);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Apple
iPhone 11 Pro','89.64.25.20',1,6);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Netflix
Windows App','95.156.244.211',3,7);

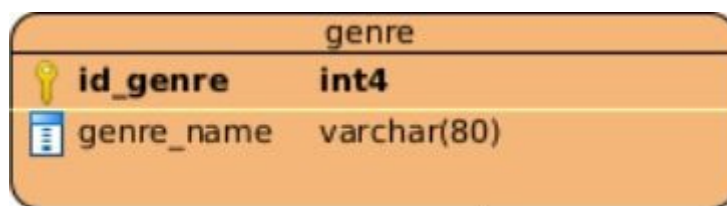
INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('LG 2015
MTK A5LR Smart TV','193.186.206.90',5,8);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES ('Philips
TPV 2K18 MT5802 Smart TV','185.122.99.100',5,9);

INSERT INTO public.device(device_name,ip_address,device_typeid,regionid) VALUES
('DefaultWidevineAndroidPhone','37.248.156.19',1,10);

```

12. genre



```









INSERT INTO public.genre(genre_name) VALUES ('dramat');
INSERT INTO public.genre(genre_name) VALUES ('komedia');
INSERT INTO public.genre(genre_name) VALUES ('thriller');
INSERT INTO public.genre(genre_name) VALUES ('sci-fi');
INSERT INTO public.genre(genre_name) VALUES ('animowany');
INSERT INTO public.genre(genre_name) VALUES ('historyczny');
INSERT INTO public.genre(genre_name) VALUES ('akcji');
INSERT INTO public.genre(genre_name) VALUES ('edukacyjny');

```

```
INSERT INTO public.genre(genre_name) VALUES ('dokumentalny');
```

```
INSERT INTO public.genre(genre_name) VALUES ('biograficzny');
```

13. Film

film	
 id_Film	int4
 title	varchar(100)
 how_long	int4
 poster	varchar(100)
 production_year	date
 path_to_file	varchar(100) U
 age_categoryid	int4
 about	varchar(500)
 add_time	timestamp

```
INSERT INTO
```

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)  
VALUES ('Kingsman: Tajne
```

```
Służby',130,'/plakaty/filmy/kingsman.jpg','2014-05-17','/filmy/kingsman2014.nfv','2','brytyjsko-  
amerykański film szpiegowski','2015-05-14 16:30:00');
```

```
INSERT INTO
```

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)  
VALUES ('Avengers: Koniec
```

```
gry',182,'/plakaty/filmy/avengers.jpg','2019-07-26','/filmy/avengersEndgame.nfv','4','amerykański  
fantastycznonaukowy film akcji z 2019 roku na podstawie serii komiksów o grupie  
superbohaterów','2019-06-25 19:30:00');
```

```
INSERT INTO
```

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)  
VALUES ('X-Men: Mroczna
```

```
Phoenix',114,'/plakaty/filmy/phoenix.jpg','2019-08-10','/filmy/phoenix.nfv','3','amerykański  
fantastycznonaukowy film akcji na podstawie serii komiksów o grupie superbohaterów','2019-08-  
20 18:00:01');
```

```
INSERT INTO
```

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
```

```
VALUES ('Gwiezdne wojny: część II – Atak klonów',142,'/plakaty/filmy/atakklonow.jpg','2002-02-  
10','/filmy/atakklonow.nfv','5','chronologicznie druga część sagi Gwiezdne wojny','2014-11-14  
15:25:00');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Gwiezdne wojny: część I – Mroczne
widmo',136,'plakaty/filmy/mrocznewidmo.jpg','1999-12-06','/filmy/
mrocznewidmo.nfv','3','chronologicznie pierwsza część sagi Gwiezdne wojny','2014-10-26
16:45:10');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Niezgodna',139,'plakaty/filmy/niezgodna.jpg','2014-05-18','/filmy/
niezgodna.nfv','1','amerykański przygodowy film akcji z elementami fantastyki
socjologicznej','2018-08-19 18:30:00');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Więzień labiryntu',113,'plakaty/filmy/labirynt.jpg','2014-05-11','/filmy/
labirynt.nfv','4','amerykański film fantastycznonaukowy wyreżyserowany przez Wesa Balla','2018-
09-23 20:05:00');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Więzień labiryntu: Lek na
śmierć',143,'plakaty/filmy/labirynt_lek.jpg','2018-09-04','/filmy/labirynt_lek.nfv','2','amerykański
film science fiction z 2018 roku w reżyserii Wesa Balla','2021-05-03 20:30:00');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Niezniszczalni',332,'plakaty/niezniszczalni.jpg','2010-04-22','/filmy/
niezniszczalni.nfv','1','The Expendables to amerykańska seria thrillerów akcji','2013-05-12
15:45:00');
```

INSERT INTO

```
public.film(title,how_long,poster,production_year,path_to_file,age_categoryid,about,add_time)
VALUES ('Faceci w czerni',98,'plakaty/filmy/faceciwczerni.jpg','1997-09-20','/filmy/
faceciwczerni.nfv','2','Men in Black to seria amerykańskich komedii akcji science fiction','2012-06-
18 20:30:05');
```

14. serie_genre

serie_genre	
 serieid	int4
 genreid	int4
 id_serieGenre	int4

```
INSERT INTO public.serie_genre(serieid,genreid) VALUES (9,1);
```

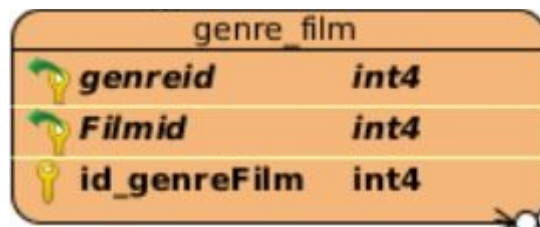
```
INSERT INTO public.serie_genre(serieid,genreid) VALUES (10,2);
```

```

INSERT INTO public.serie_genre(serieid,genreid) VALUES (11,3);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (12,4);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (13,5);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (14,6);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (15,7);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (16,6);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (17,9);
INSERT INTO public.serie_genre(serieid,genreid) VALUES (18,1);

```

15. genre_film

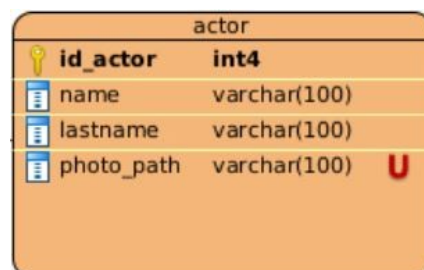


```

INSERT INTO public.genre_film(genreid,Filmid) VALUES (1,1);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (3,5);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (3,2);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (4,3);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (5,4);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (2,4);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (1,6);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (4,5);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (5,10);
INSERT INTO public.genre_film(genreid,Filmid) VALUES (3,8);

```

16. actor



```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('James','Spader','aktorzy\jamespader.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Bryan','Cranston','aktorzy\bryancranston.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Kevin','Spacey','aktorzy\kevinspacey.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Charles','Leclerc','aktorzy\charlesleclerc.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Claire','Danes','aktorzy\clairedanes.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Katie','Holmes','aktorzy\katieholmes.jpg');
```

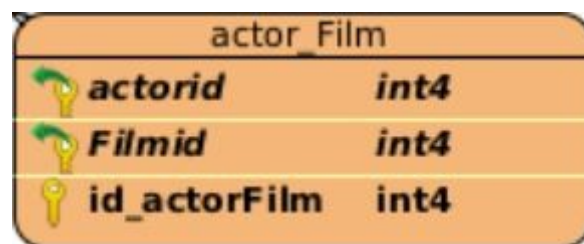
```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Cariba','Heine','aktorzy\caribaheine.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Devan','Cohen','aktorzy\devancohen.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Helen','McCrory','aktorzy\helenmccrory.jpg');
```

```
INSERT INTO public.actor(name,lastname,photo_path) VALUES ('Lili','Reinhart','aktorzy\lilireinhart.jpg');
```

17. actor_film



```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (5,5);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (7,3);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (8,1);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (2,5);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (3,8);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (9,10);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (9,4);
```

```
INSERT INTO public.actor_film(actorId,FilmId) VALUES (2,7);
```

INSERT INTO public.actor_film(actorId,FilmId) VALUES (8,8);

INSERT INTO public.actor_film(actorId,FilmId) VALUES (5,2);

18. actor_serie



The diagram shows the structure of the actor_serie table. It has three columns: actorid, serieid, and id_actorSerie, all of type int4. actorid and serieid are marked as primary keys with yellow key icons.

actor_serie	
 actorid	int4
 serieid	int4
 id_actorSerie	int4

INSERT INTO public.actor_serie(actorid,serieid) VALUES (1,9);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (2,10);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (3,11);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (4,12);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (5,13);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (6,14);

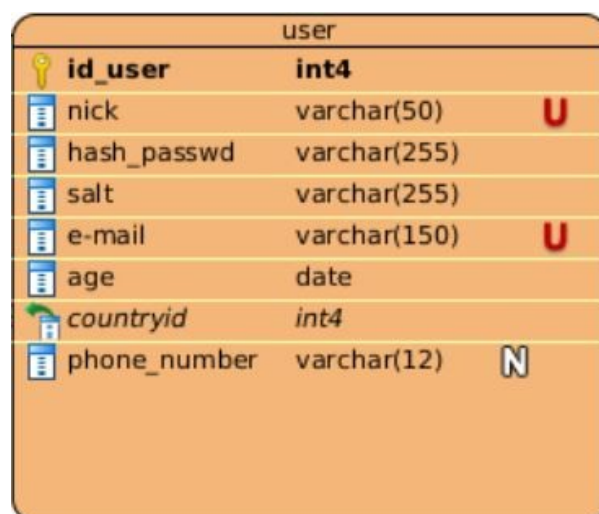
INSERT INTO public.actor_serie(actorid,serieid) VALUES (7,15);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (8,16);









INSERT INTO public.actor_serie(actorid,serieid) VALUES (9,17);

INSERT INTO public.actor_serie(actorid,serieid) VALUES (10,12);

19. user



The diagram shows the structure of the user table. It has eight columns: id_user, nick, hash_passwd, salt, e-mail, age, countryid, and phone_number. id_user is the primary key. nick, hash_passwd, salt, e-mail, and phone_number are unique. countryid is a foreign key to the actor_serie table.

user	
 id_user	int4
 nick	varchar(50) U
 hash_passwd	varchar(255)
 salt	varchar(255)
 e-mail	varchar(150) U
 age	date
 countryid	int4
 phone_number	varchar(12) N

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Stolarski','ABQUyr625190ienURYTNHGBAqweYIm','ERUIJKNW23','Stolarz1999@gmail.com','1997-12-15',1,'982-756-165');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Pingwin16','SUTI5678WEHBNEUIP20396821KCTN2','ENB34JKL12','From0Manchester@gmail.com','2013-05-06',2,'867-147-173');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Roboloxo','URH3JFMN54NGU54INGTI45NGI45NG4','WERUTHNU13','Roboloxer@gmail.com','2015-01-01',3,'209-652-802');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Khadgar','EIUHG8479GH487UGH4587UH4875HG2','BYWEUVBUW2','K0Porebskipe@gmail.com','2007-08-12',4,'056-195-320');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Ragnaros','JG398GJ938WGH389GHJ983GHJ3894','JG39G39G38','W0Banach@gmail.com','1998-11-17',5,'175-486-091');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Anduin','SIDUHFSIDUFHSIFHEW882HF2I8FB28','WEHFBHWBFR','Karolando@gmail.com','2005-07-23',6,'450-184-284');




INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Jaina1976','UWRHN28GBV823VGBN832GHNIK3INHQ','2R2G3H35J3','Trolejcart@gmail.com','1973-04-04',7,'826-687-103');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Uther2012','EGRFH38RGHV8R7EHG3I8HGF29FJ28J','WFNH82F7NB','JacobsPersonalEmail@gmail.com','1940-05-08',8,'856-185-327');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Tyrande56','WIOFJN827HNGVUI3WRNH32874H29GN','GNH2847HG2','LadyHelenpe@gmail.com','1986-12-12',9,'952-587-175');

INSERT INTO public.user(nick,hash_passwd,salt,email,age,countryid,phone_number) VALUES ('Malfurion23','WOY85IDNSJW64JNM1Y3N1JFN2UGM4K','HJTYU2MFU4','[to jest moj_email_PO Czarek@gmail.com](mailto:to_jest_moj_email_PO Czarek@gmail.com)','2006-06-19',10,'732-801-296');

20. serie_productioncountry

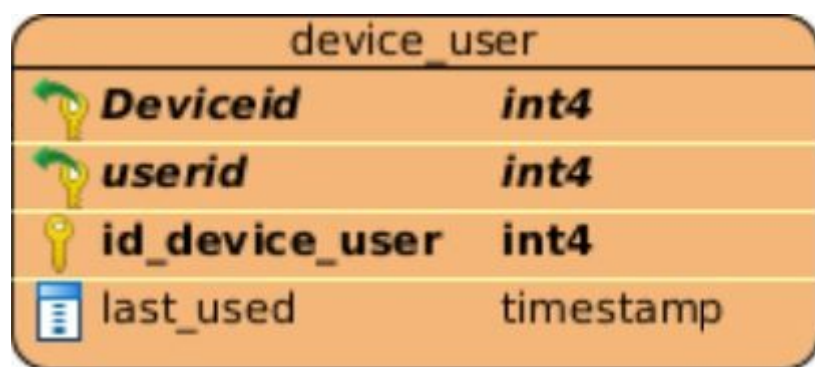
serie_productionCountry	
 serieid	int4
 countryid	int4
 id_serie_productionCountry	int4


```

INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (10,9);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (9,9);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (11,2);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (14,9);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (15,9);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (12,3);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (14,4);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (16,6);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (11,9);
INSERT INTO public.serie_productionCountry(serieid,countryid) VALUES (10,9);

```

21. device_user



```

INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (1,1,'2020-04-26
15:46:57');
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (2,4,'2021-05-14
21:03:29');
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (3,2,'2021-02-21
18:35:05');
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (5,3,'2020-11-08
10:51:18');
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (4,5,'2020-08-12
04:41:22');
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (8,1,'2021-07-14
15:55:01');

```



```
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (7,7,'2020-04-17 07:48:11');
```

```
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (10,9,'2021-01-26 10:39:24');
```

```
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (6,5,'2021-02-23 22:16:42');
```

```
INSERT INTO public.device_user(DeviceId,userId,last_used) VALUES (9,4,'2020-12-30 05:45:13');
```

22. credit_card

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('1234123412341234',123,'01/25','Tomek','Kowalski');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('2323232323232323',234,'02/26','Tymon','Duda');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('3434343434433443',232,'05/28','Karolina','Tusk');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('4567456745674678',964,'12/23','Natalia','Gosiewska');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('5123123412346235',841,'11/24','Monika','Czajka');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('6511879875164174',102,'08/22','Dominik','Godlewski');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('7417181181951565',966,'07/23','Dominik','Krawczyk');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('4984187416354987',852,'12/24','Michał','Kowalczyk');
```

```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('7498498747846579',741,'11/22','Aneta','Stajniak');
```

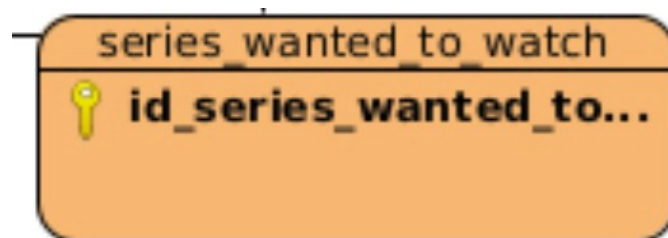
```
INSERT INTO public.credit_card(card_number,cvc,expiration_date,owner_name,owner_lastname) VALUES ('7894561237894568',420,'10/23','Michalina','Nowak');
```

23. film_wanted_to_watch



```
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (1);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (2);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (3);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (4);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (5);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (6);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (7);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (8);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (9);
INSERT INTO public.film_wanted_to_watch(id_Film_wanted_to_watch) VALUES (10);
```

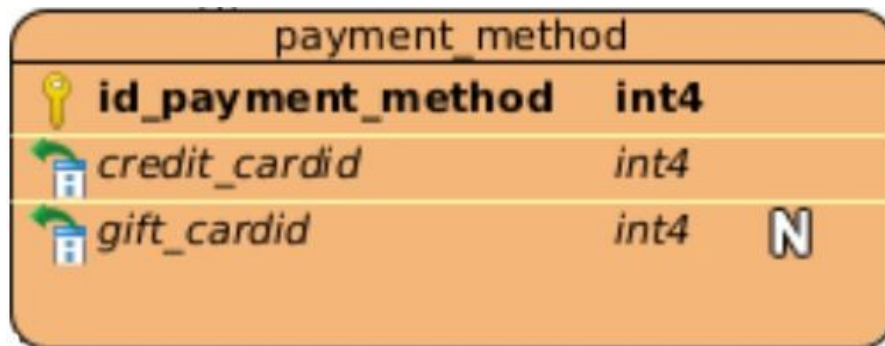
24. `series_wanted_to_watch`



```
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (1);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (2);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (3);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (4);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (5);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (6);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (7);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (8);
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (9);
```

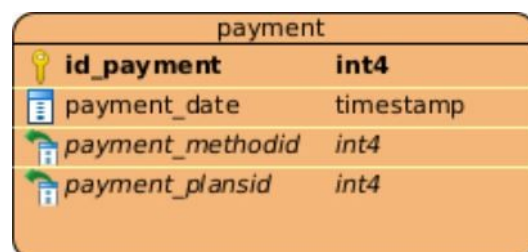
INSERT INTO public.series_wanted_to_watch(id_series_wanted_to_watch) VALUES (10);

25. payment_method



```
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (1,1);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (2,NULL);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (3,2);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (4,NULL);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (5,NULL);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (6,4);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (7,9);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (8,NULL);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (9,NULL);
INSERT INTO public.payment_method(credit_cardid,gift_cardid) VALUES (10,NULL);
```

26. payment



```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES
('2020-11-12 14:51:23',1,1);
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES
('2021-10-13 17:24:13',2,2);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2019-12-21 15:38:56',3,3);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2021-05-06 07:29:41',4,4);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2021-04-01 09:24:35',5,5);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2021-01-01 00:00:00',6,1);
```









```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2020-06-04 22:24:23',7,2);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2018-05-06 14:23:46',8,3);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2017-11-16 19:28:10',9,4);
```

```
INSERT INTO public.payment(payment_date,payment_methodid,payment_plansid) VALUES ('2020-12-14 13:56:23',10,5);
```

27. user_profiles

user_profiles		
	id_user_profile	int8
	nickname	varchar(45)
	profile_photosid	int4
	Uzytkownikid	int4
	is_adult	bytea
	Languagesid	int4
	Film_Wanted_to_watchid	int4 N
	Series_wanted_to_watchid	int4 N

```
INSERT INTO public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Rollermix',1,1,'000',1,1,NULL);
```

```
INSERT INTO public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Durszlak',2,2,'000',2,NULL,NULL);
```

```
INSERT INTO public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Krokiet',3,3,'000',3,NULL,1);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Ruen',3,4,'000',1,NULL,NULL);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Elwi',4,5,'000',1,2,NULL);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Vdr1984',5,6,'000',1,NULL,NULL);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Hallack',6,7,'001',1,NULL,NULL);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Multi',8,2,'001',1,3,NULL);
```






INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Skillplease',1,2,'001',1,NULL,NULL);
```

INSERT INTO

```
public.user_profiles(nickname,profile_photosid,uzytownikid,is_adult,Languagesid,Film_Wanted_to_watchid,Series_wanted_to_watchid) VALUES ('Prysio',2,1,'001',1,NULL,NULL);
```

28. profile_is_watching

profile_is_watching			
	id_session	int4	
	User_profilesid	int8	
	Episode_serieid	int4	N
	Filmid	int4	N
	date	timestamp	

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (14,2,NULL,'2020-06-22 11:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (15,NULL,1,'2021-04-23 12:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (16,NULL,NULL,'2020-03-22 13:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (17,NULL,NULL,'2020-05-22 14:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (18,5,NULL,'2020-02-22 15:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (19,NULL,6,'2021-01-22 16:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (20,7,NULL,'2021-01-22 17:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (21,NULL,8,'2020-01-22 19:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (22,NULL,NULL,'2021-01-22 19:10:25');
```

```
INSERT INTO public.profile_is_watching(user_profilesid,Episode_serieid,Filmid,date) VALUES (23,NULL,NULL,'2021-02-22 12:10:25');
```

29. watching_history

watching_history	
 id_watchingHistory	int4
 profile_is_watchingid_session	int4
 stop_watching_moment	varchar(50)

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (2,34);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (3,54);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (4,12);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (5,15);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (6,20);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (7,13);
```



```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (8,14);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (9,32);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment) VALUES (10,10);
```

```
INSERT INTO public.watching_history(profile_is_watchingid_session,stop_watching_moment)
VALUES (11,12);
```

30. serie_series_wanted_to_watch

serie_series_wanted_to_watch	
 Serieid	int4
 Series_wanted_to_watchid	int4
 id_SerieSeriesWantedtoWatch	int4

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(9,1);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(10,2);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(11,3);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(12,4);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(13,5);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(14,6);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(15,7);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(16,8);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(14,9);
```

```
INSERT INTO public.serie_series_wanted_to_watch(Serieid,Series_wanted_to_watchid) VALUES
(12,10);
```

31. film_film_wanted_to_watch

film_Film_Wanted_to_watch	
 Filmid	int4
 Film_Wanted_to_watchid	int4
 id_FilmFilmWantedtoWatch	int4

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (1,1);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (2,2);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (3,3);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (4,4);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (5,5);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (6,6);

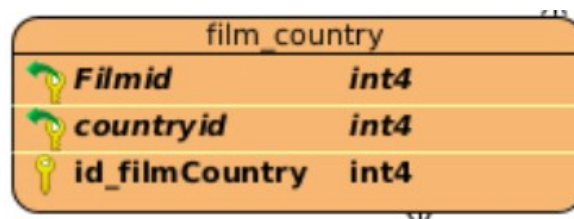
INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (7,7);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (8,8);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (9,9);

INSERT INTO public.film_Film_wanted_to_watch(Filmid,Film_Wanted_to_watchid) VALUES (10,10);

32. film_country



INSERT INTO public.film_country(genreid,Filmid) VALUES (1,1);

INSERT INTO public.film_country(genreid,Filmid) VALUES (2,2);

INSERT INTO public.film_country(genreid,Filmid) VALUES (3,3);

INSERT INTO public.film_country(genreid,Filmid) VALUES (4,4);

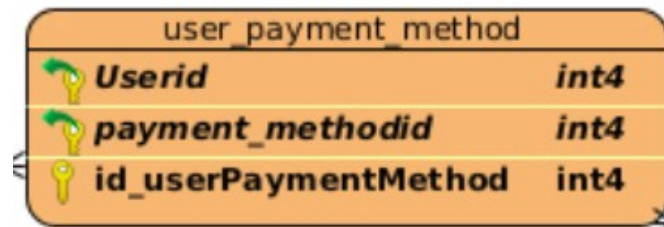
INSERT INTO public.film_country(genreid,Filmid) VALUES (5,5);

INSERT INTO public.film_country(genreid,Filmid) VALUES (6,6);

INSERT INTO public.film_country(genreid,Filmid) VALUES (7,7);


```
INSERT INTO public.film_country(genreid,Filmid) VALUES (8,8);
INSERT INTO public.film_country(genreid,Filmid) VALUES (9,9);
INSERT INTO public.film_country(genreid,Filmid) VALUES (10,10);
```

33. user_payment_method



```
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (1,1);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (2,2);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (3,3);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (4,4);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (5,5);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (6,6);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (7,7);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (8,8);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (9,9);
INSERT INTO public.user_payment_method(userid,payment_methodid) VALUES (10,10);
```