

Заказчик: Стриминговый сервис для просмотра сериалов и фильмов.

Задача: Реализовать базу данных, хранящую информацию о пользователях сервиса и фильмах, представленных на нем.

- 1) Данные об аккаунтах пользователей (id пользователя на сервисе, email, ФИО, пароль, номер телефона, идентификатор подписки)
- 2) Действия пользователей (фильмы и сериалы, которые они посмотрели/не посмотрели/планируют посмотреть, комментарии к фильмам/сериалам)
- 3) Информация о фильмах/сериалах, представленных на сервисе (год производства, страна, жанр, режиссер, список актеров)

Реализация: используется формат реляционной БД. Реализуются следующие 13 таблиц (11 для сущностей и 2 для связей):

- 1) User – информация о пользователях
- 2) Subscription – информация о подписке
- 3) Comments – комментарии, оставляемые пользователем
- 4) Ratings – оценки, поставленные фильмам
- 5) Usage history – какой фильм пользователь посмотрел/не посмотрел
- 6) Status of video – фильм просмотрен/не просмотрен
- 7) Video – информация об имеющемся контенте
- 8) Video types – жанр и тип (сериал, фильм или другое)
- 9) Countries – страна-производитель
- 10) Actors - информация об актерах
- 11) Directors – информация о режиссерах
- 12) Shoot – идентификатор режиссера для каждого фильма
- 13) Filming – идентификатор актера для каждого фильма

Такая декомпозиция базы данных обеспечивает, что **каждая таблица находится в 3NF.**

Описание сущностей:

Название сущности	Название атрибута	Тип данных	Описание	Первичный(PK)/ Внешний ключи (FK)
Actors	Id_actor	int	Идентификатор актера	PK
	First_name	varchar(60)	Имя	
	Last_Name	varchar(60)	Фамилия	
	Date_of_birth	date	Дата рождения	
	age	int	возраст	
Comments	comments	varchar (400)	Текст комментария к фильму	
	Id_user	int	Идентификатор пользователя	FK
	Id_video	int	Идентификатор видео	FK
Directors	Id_director	int	Идентификатор режиссера	PK
	First_name	varchar(60)	имя	
	Last_name	varchar(60)	фамилия	
	Date_of_birth	date	Дата рождения	
	age	int	возраст	

Ratings	score	Int	Выставленная оценка	
	Id_user	varchar(120)	Идентификатор пользователя	FK
	Id_video	varchar(120)	Идентификатор видео	FK
Status_of_video	Name_of_status	int	Статус фильма по просмотрам	
	Id_status	varchar(50)	Идентификатор статуса	PK
Subscription	Id_subscription	int	идентификатор подписки	PK
	type	Varchar(60)	Тип подписки	
	cost	int	стоимость подписки	
	duration	int	Дата истечения подписки	
Usage_history	Id_of_video	int	Идентификатор видео	PK(FK)
	Id_of_users	int	Идентификатор пользователя	PK(FK)
	Id_of_status	int	Идентификатор статуса контента	PK(FK)
User	Id_user	int	идентификатор пользователя	PK
	email	Varchar(60)	Электронная почта	
	First_name	Varchar(60)	имя	
	Last_name	Varchar(60)	фамилия	
	Second_name	Varchar(60)	отчество	
	password	Varchar(60)	Пароль пользователя	
	Phone_number	Varchar(60)	Номер телефона	
	Id_subscription	int	Идентификатор подписки	FK
Video	Id_video	int	Идентификатор видео	PK
	title	Varchar(60)	Название фильма	
	Age_limit	int	Возрастное ограничение	
	year	date	Год выпуска	
	Id_video_type	int	Идентификатор типа видео	FK
	Id_country	int	Идентификатор страны	FK
Video_types	Id_video_type	int	Идентификатор типа видео	PK
	genre	Varchar(60)	Жанр видео	
	Video_type	Varchar(60)	Название типа видео	
Countries	Id_country	int	Идентификатор страны	PK
	Name_of_country	Varchar(60)	Название страны	

Ограничения: Предполагается, что для корректной работы базы данных значения каждого поля каждой сущности обязаны быть ненулевыми.

- 1) Уникальные идентификаторы (id) по умолчанию обладают этим свойством.
- 2) Для всех неидентификационных полей, относящихся к сущностям фильмов, режиссеров и актеров (например, имя режиссера, год выпуска фильма, возраст актера и т.д.) выполнение условия очевидно (данная информация находится в открытом доступе).
- 3) Для пользовательской информации (email, пароль и т.д.) «notnull-ность» должна обеспечиваться на этапе регистрации аккаунта (то есть в случае БД — на этапе insert'a).
- 4) Для сущностей Ratings и Comments: если запись о рейтинге/комментарии существует, то очевидно, что оценка/комментарий — не пустое поле.
- 5) Для таблиц, описывающих связи (shoot, filming) выполнение свойства тривиально (их поля — id'шники => ненулевые из п.1)