

Национальный исследовательский ядерный университет «МИФИ»

(Московский Инженерно-Физический Институт)

Кафедра №42 «Криптология и кибербезопасность»

Лабораторная работа №1

«Построение модели данных»

Тимин Александр Б21-515 (2024г.)

1 Неформальное описание предметной области

Предметная область данной работы охватывает базу данных телевизионных сериалов. Компания, управляющая веб-сайтом, предоставляет информацию о различных телевизионных шоу для своих посетителей. Сведения о сериалах включают название, год выпуска, бюджет производства, краткую аннотацию, списки актеров и эпизодов, а также отзывы критиков и другие дополнительные данные.

2 Спецификация таблиц

Таблица Shows (Сериалы):

1. ShowID (Идентификатор сериала): Уникальный идентификатор сериала.
2. Title (Название): Название сериала.
3. Year (Год выпуска): Год выпуска сериала.
4. Budget (Бюджет): Бюджет производства сериала.
5. Synopsis (Аннотация): Краткое описание сюжета сериала.

Таблица Actors (Актеры):

1. ActorID (Идентификатор актера): Уникальный идентификатор актера.
2. Name (Имя): Имя актера.
3. DateOfBirth (Дата рождения): Дата рождения актера.
4. Biography (Биография): Биографическая информация об актере.

Таблица Episodes (Эпизоды):

1. EpisodeID (Идентификатор эпизода): Уникальный идентификатор эпизода.
2. Title (Название эпизода): Название эпизода.
3. SeasonNumber (Номер сезона): Номер сезона, к которому относится эпизод.
4. EpisodeNumber (Номер эпизода): Номер эпизода в рамках сезона.
5. AirDate (Дата выхода): Дата выхода эпизода в эфир.

6. ShowID (Идентификатор сериала): Ссылка на идентификатор сериала, к которому относится эпизод.

Таблица Reviews (Отзывы):

1. ReviewID (Идентификатор отзыва): Уникальный идентификатор отзыва.
2. ShowID (Идентификатор сериала): Ссылка на идентификатор сериала, к которому относится отзыв.
3. Reviewer (Рецензент): Имя или никнейм рецензента.
4. ReviewText (Текст отзыва): Текстовое содержание отзыва.
5. Rating (Оценка): Оценка, выставленная сериалу.

Таблица Cast (Актёрский состав):

1. ActorID (Идентификатор актера): Ссылка на идентификатор актера.
2. ShowID (Идентификатор сериала): Ссылка на идентификатор сериала, в котором актер сыграл роль.
3. Role (Роль): Роль, сыгранная актером в сериале.

3 Соответствия требованиям третьей нормальной формы

- Все таблицы имеют первичные ключи (PK).
- В таблице Episodes поле ShowID является внешним ключом (FK), ссылается на первичный ключ таблицы Shows. В таблице Reviews поле ShowID также является внешним ключом (FK), ссылается на первичный ключ таблицы Shows.
- Таблица Cast имеет составной первичный ключ (ActorID, ShowID), гарантируя уникальность связи между актером и сериалом.
- Нет транзитивных зависимостей, каждая таблица содержит только данные, прямо относящиеся к своей сущности.

Таким образом, предложенная структура базы данных соответствует требованиям третьей нормальной формы.

Заключение

В ходе работы была выбрана предметная область – «База данных сериалов», составлено неформальное описание предметной области, спроектирована схема базы данных для хранения информации о сериалах, по спроектированной схеме была создана база данных. Также в ходе работы предоставлено доказательство соответствия базы данных третьей нормальной форме.

На защиту

На рисунке 1 – скриншот сообщения с заданиями на защиту.

Re: БСБД Лаба 1

•  Михаил Куприяшин 28 мая, 2:25
Кому: вам     ...

Мне пришлось вручную рендерить ERD, потому что отчёт ведёт на GitHub, а там -- сырые исходники..
Это край, товарищ)

На защиту: (1) каждое телешоу разбито на сезоны, которые когда-то выходят в эфир, содержат определённые эпизоды и, иногда, как-то называются (2) есть реестр иноагентов, куда иногда добавляют, а иногда убирают оттуда актёров. Кого-то, возможно, добавляли уже дважды. Доработайте схему данных, чтобы учесть эту информацию

Рисунок 1 – задания на защиту.

В приложении теперь находится ссылка на ERD в формате картинки.

На рисунке 2 изображена актуальная схема зависимостей таблиц базы данных.

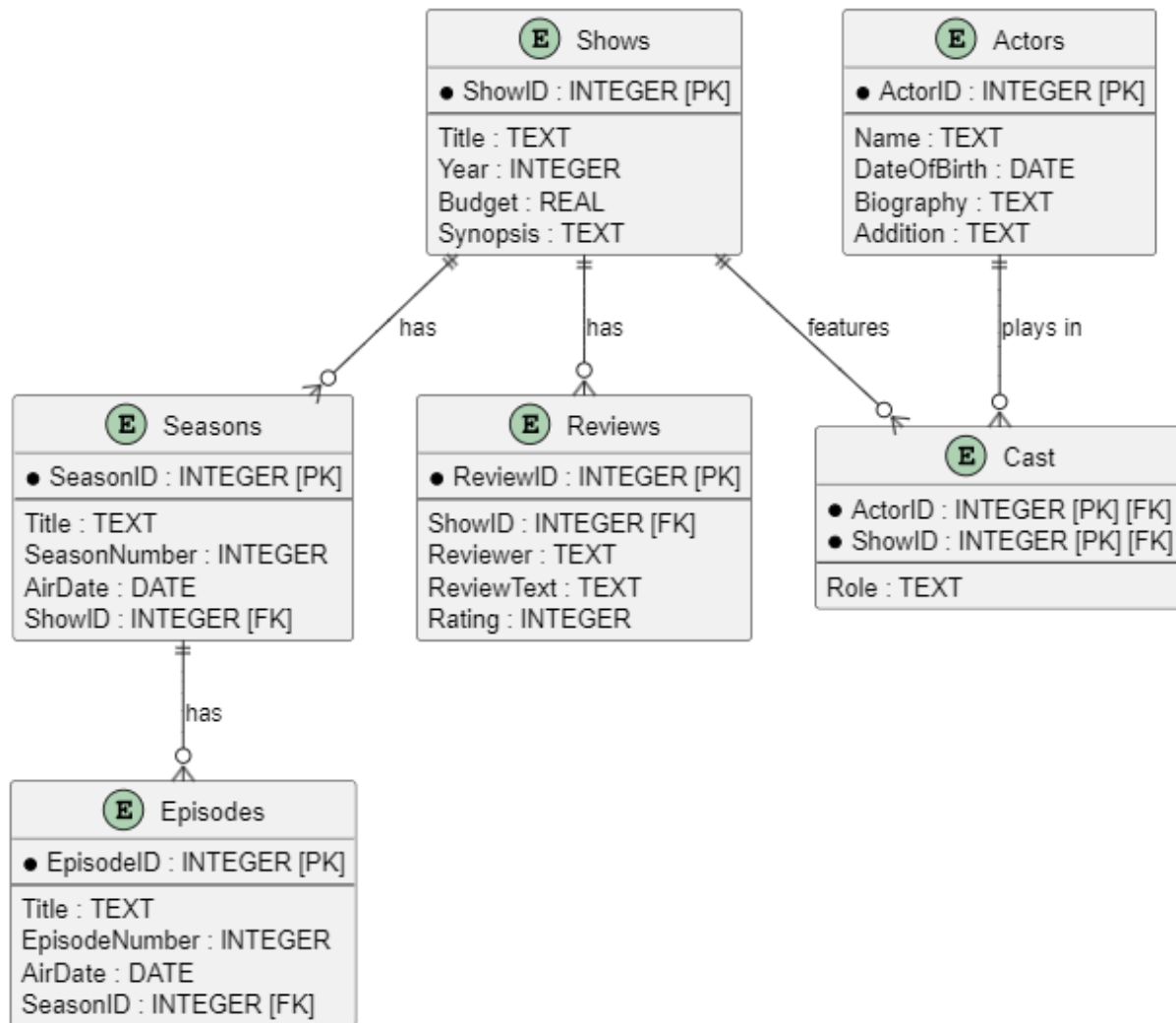


Рисунок 2 - схема зависимостей таблиц базы данных.

По сравнению с предыдущей схемой присутствует несколько различий:

1. *Сезоны*. Добавлена таблица Seasons, в которой будет храниться информация о каждом конкретном сезоне:
 - *SeasonID (INTEGER)* – первичный ключ;
 - *Title (TEXT)* – название сезона (при наличии);
 - *SeasonNumber (INTEGER)* – порядковый номер сезона;
 - *AirDate (DATE)* – дата выхода сезона;
 - *ShowID (INTEGER)* – внешний ключ (ссылается на первичный ключ в таблице Shows).
2. *Эпизоды*. Изменена таблица Episodes:

- *SeasonNumber* – удалена колонка, обозначающая порядковый номер сезона;
- *ShowID* – удалена колонка с внешним ключом (ссылавшимся на первичный ключ таблицы Shows);
- *SeasonID (INTEGER)* – добавлена колонка с внешним ключом (ссылающимся на первичный ключ таблицы Seasons).

3. *Актеры*. Добавлена колонка *Addition (TEXT)*, хранящая дополнительную информацию об актере (например, о статусе иноагента).

Для внедрения изменений необходимо выполнить несколько SQL-запросов (ссылка на листинг в разделе приложения).

На рисунке 3 приведен скриншот из GUI SQLiteStudio, на котором видны все отредактированные таблицы.

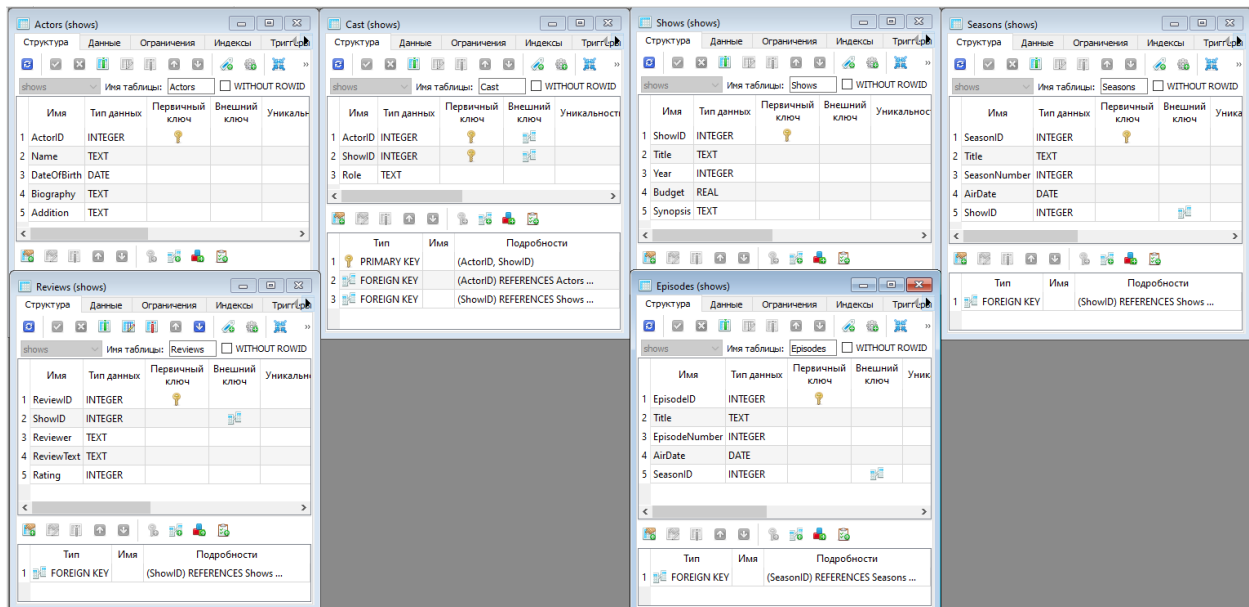


Рисунок 3 - отредактированные таблицы.

Приложение

- [обновленная UML-схема базы данных \(png\)](#);
- [SQL-скрипт редактирования таблиц](#);
- [созданная база данных \(SQLite\)](#);
- [отредактированный отчет \(docx\)](#);
- [отредактированный отчет \(pdf\)](#).