ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ

УНИВЕРСИТЕТ имени академика С. П. КОРОЛЕВА»

Факультет информатики

Кафедра технической кибернетики

Пояснительная записка к курсовому проекту

по курсу «Базы данных»

Выполнили Кравченко К.А.

Елкин Д.А.

Группа 6408

Преподаватель Логанова Л.В.

САМАРА 2014

Оглавление

[Постановка задачи 3](#_Toc381475155)

[ER-модель 4](#_Toc381475156)

# Постановка задачи

В данном курсовом проекте была поставлена задача разработки базы данных под названием “Каталог фильмов”. В качестве предметной области была выбрана киноиндустрия. Полученная база данных может быть использована для подбора фильмов для просмотра, применяться в качестве домашнего каталога фильмов и т.д. Для обеспечения доступа пользователя к данным была разработана система, исполняющая следующие функции:

* просмотр, редактирование, добавление и удаление данных;
* выполнение запросов пользователя;
* вывод отчетов по всем имеющимся фильмам.

Разработанная база данных содержит полную информацию о фильмах, режиссерах, актерах. Предоставляется также информация о видеоносителях, на которых выходили фильмы, и наградах, которые данные фильмы завоевали.

Разработка данного приложения велась в среде Oracle.

# ER-модель

ER-модель данных приведена на рисунке 1. Рассмотрим таблицы входящие в базу данных:

1. **Movie**  
   Содержит информацию о фильмах. Атрибуты:
   * *id* – идентификатор фильма;
   * *name* – название;
   * *release\_date* – дата выпуска фильма;
   * *restriction* – возрастные ограничения;
   * *duration* – длительность; ПОМЕНЯЙ НА КАРТИНКЕ
   * *world\_box* – кассовые сборы фильма в мире;
   * *budget* – бюджет фильма;
   * *site* – официальный сайт;
   * *description –* информация о фильме;
   * *poster* – постер фильма;
   * *rating –* пользовательский рейтинг фильма.
2. **Director**  
   Предоставляет информацию о режиссерах. Атрибуты:
   * *id* – идентификатор режиссера;
   * *name* – фамилия, имя;
   * *birth\_date* – дата рождения;
   * *photo* – фотография; ПОМЕНЯЙ НА КАРТИНКЕ
   * *birth\_country –* страна, в которой родился режиссер.
3. **Actor**  
   Содержит информацию об актерах. Атрибуты:
   * *id* – идентификатор актера;
   * *name* – фамилия, имя;
   * *birth\_date* – дата рождения;
   * *photo* – фотография; ПОМЕНЯЙ НА КАРТИНКЕ
   * *birth\_country –* страна, в которой родился актер.
4. **Genre**Задает известные жанры. Атрибуты:
   * *name* – название жанра;
   * *description* – информация о жанре.
5. **Award**Содержит информацию о наградах (премиях). Атрибуты:
   * *id* – идентификатор награды;
   * *name* – название;
   * *year* – год проведения церемонии награждения;
   * *location* – место проведения церемонии награждения.
6. **Nomination**Содержит информацию о номинациях различных наград. Атрибуты:
   * *id* – идентификатор номинации;
   * *award\_id* – идентификатор награды;
   * *name* – название номинации.
7. **MovieNominationsConnector**Служит для реализации связи многие ко многим между таблицами **Movie** и  
   **Nominations**. Атрибуты:
   * *movie\_id –* идентификатор фильма
   * *nomination\_id –* идентификатор номинации
   * *status –* описывает, является данный фильм лауреатом или просто номинантом.
8. **Country**Содержит названия стран. Атрибуты:
   * *id –* идентификатор страны;
   * *name –* название страны.
9. **User**Содержит информацию о пользователе сервиса. Атрибуты:
   * *id –* идентификатор пользователя;
   * *login –* идентификатор пользователя на сервисе;
   * *password –* пароль пользователя на сервисе;
   * *birth\_date –* дата рождения пользователя;
   * *registration\_date –* дата регистрации пользователя в системе;
   * *name –* ФИО пользователя.
10. **Rate**Содержит информацию об оценках, которые пользователи выставляют фильмам. Атрибуты:
    * *movie\_id –* идентификатор фильма, для которого пользователь (*user\_id)* выставил оценку;
    * *user\_id –* идентификатор пользователя, который выставил оценку фильму (*movie\_id)*;
    * *rate –* оценка пользователя (*user\_id*) для фильма (*movie\_id*);
11. **Character**Содержит информацию о герое, которого играл определенный актер в определенном фильме. Атрибуты:
    * *actor\_id –* идентификатор актера, который играет данного героя в фильме (*movie\_id*);
    * *movie\_id –* идентификатор фильма, в котором актер (*actor\_id*) играет данного героя;
    * *char\_name –* имя героя, которого играет актер (*actor\_id*) в фильме (*movie\_id*);
    * *description –* описание героя, которого играет актер (*actor\_id*) в фильме (*movie\_id*).
12. **Review**Содержит информацию о рецензии к фильму. Атрибуты:
    * *review\_id* – идентификатор рецензии;
    * *movie\_id –* идентификатор фильма, на который написана рецензия;
    * *user\_id* – идентификатор пользователя, который написал рецензию;
    * *text –* текст рецензии;
    * *review\_date –* дата публикации рецензии;
    * *rating –* характер рецензии.

Характер рецензии, определяет в каком контексте написана рецензия (положительный, отрицательный, нейтральный).

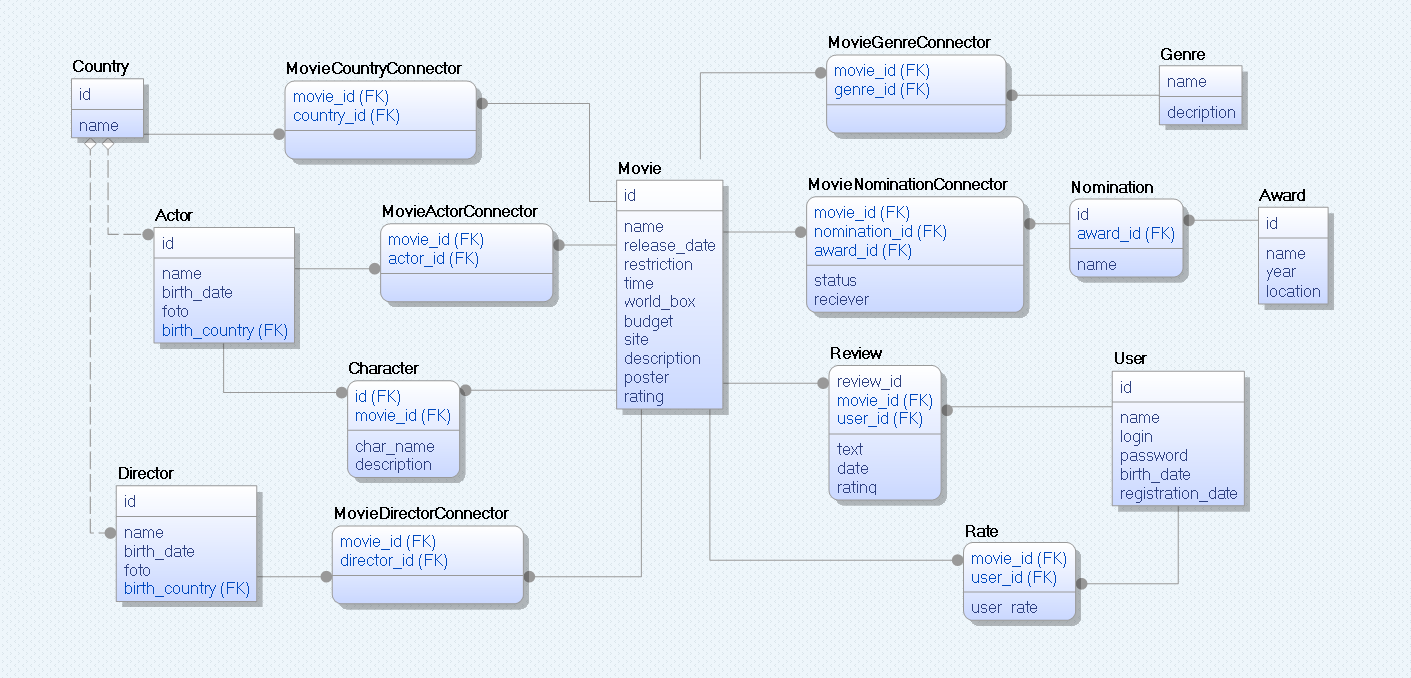
Также в базе данных присутствуют дополнительные таблицы, обеспечивающие связи многие ко многим.

В данной базе данных реализованы следующие связи:

1. **Movie – Director**Тип связи: многие ко многим. Для реализации данного типа связи используется дополнительная таблица **MovieDirectorConnector.** Предполагается, что один фильм могут снимать несколько режиссеров. С другой стороны один режиссер за свою карьеру снимает не менее одного фильма.
2. **Movie – Actor**Тип связи: многие ко многим. Для реализации данного типа связи используется дополнительная таблица **MovieActorConnector.** Очевидно, что в фильме могут сниматься более одного актера. Актеры также часто снимаются во многих фильмах.
3. **Movie – Genre**Тип связи: многие ко многим. Для реализации данного типа связи используется дополнительная таблица **MovieGenreConnector.** При проектировании базы данных учитывалось, что современные фильмы отличаются тем, что отнести их к определенному жанру достаточно проблематично, поэтому была введена связь один ко многим между таблицей фильмов и таблицей жанров. Но с другой стороны к любому жанру можно отнести заведомо больше одного фильма.
4. **Movie – Country**

Тип связи: многие ко многим. Для реализации данного типа связи используется дополнительная таблица **MovieCountryConnector.** При проектировании базы данных учитывалось, что современные фильмы отличаются тем, что они могли быть сняты в разных странах. Очевидно, что в одной стране снимали ни один фильм.

1. **Movie – Nomination**  
   Тип связи: многие ко многим. Для реализации данного типа связи используется дополнительная таблица **MovieNominationConnector.** Обычно фильм номинируется сразу в нескольких категориях, а весь смысл номинации в том, что есть несколько фильмов, которые могут выиграть данную награду. В данном случае, следует отметить интересную особенность таблицы **MovieNominationConnector.** Кроме того, что она исполняет функцию таблицы на связь, в ней также присутствуют дополнительные атрибуты, позволяющие определить является ли фильм лауреатом и кто из людей, принимавших участие в создании фильма, номинирован.



*Рисунок 1. ER-модель данных.*

# Ограничения целостности

В данной базе данных введены следующие семантические ограничения: год выпуска фильма не может быть меньше 1800, длительность фильма положительна, также следующие поля не должны быть пустыми: названия фильма, имя режиссёра, имя актера, название награды, название номинации, имя/логин/пароль пользователя, рейтинг, имя героя, статус номинации фильма. Введены также ограничения ссылочной целостности для связанных таблиц. Так, например, при удалении записи из таблицы **Movie** удаляются все связанные записи в дочерних таблицах. В случае если пользователь введет неправильные значения атрибутов, выведется информативное сообщение об ошибке.