«Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №42 «Криптология и кибербезопасность»

Отчёт по лабораторной работе №1

"Безопасность базы данных"

Студент: Нурматов С.Р.

Группа: Б17-565

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2020

База данных сериалов для открытого интернет - ресурса “Cinema-Online”

1. Описание предметной области.

Данный открытый интернет - ресурс предоставляет возможность бесплатно рассматривать различные сериалы online. Интернет – ресурс Cinema-Online позволяет не только смотреть сериалы, но и просматривать информацию о них. А именно, пользователь может

1. Прочитать аннотацию, для того, чтобы понять, стоит ли смотреть данный сериал или нет.
2. Узнать в какой стране выпущен данный сериал.
3. Дата выхода.
4. На каких телеканалах транслируется данный сериал.
5. Жанр
6. Статус (Открытый или закрытый).
7. Посмотреть список главных ролей.
8. А так же узнать какие режиссеры работали над данным сериалом.

Таблицы

Для построения данной базы данных необходимо 10 таблиц.

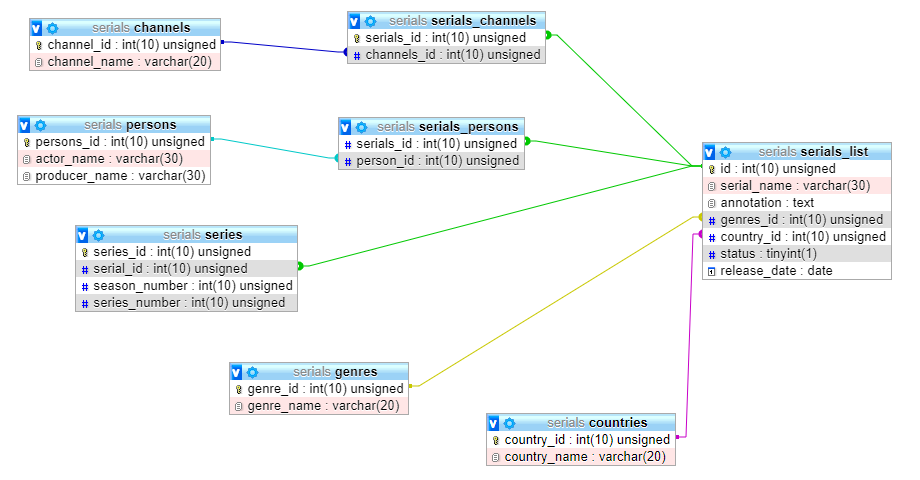
Таблица “serials\_list”, которая отвечает за хранение списка сериалов. В таблице присутствуют поля для некоторой информации: описание сериала (annotaion), статус сериала (status) (1 – продолжается, 0 – завершен) , дата выхода (release\_date). Некоторые характеристики сериалов выведены в отдельные таблицы, а именно жанры выведены в таблицу “genres”, страны выведены в таблицу “countries”.

Таблица “channels” содержит информацию о том, на каких каналах транслируется определенный сериал. Так как таких каналов несколько, то необходимо использовать связь “многие ко многим”, следовательно для реализации такой связи используется смежная таблица “serials\_channels”, которая хранит в себе id сериала и id канала.

В таблице “persons” хранится список актеров и продюсеров каждого сериала. Поле “it\_is\_producer” типа BOOLEAN определяет кем является персона, продюсером или актёром, 1 – продюсер, 0 – актёр. Так как у одного сериала может быть несколько продюсеров и всегда много актеров, а продюсеры и актеры могут участвовать в нескольких сериалах, то тут также реализуется связь “многие ко многим”. Смежная таблица “serials\_persons” содержит id сериала и id персоны. По этим данным можно узнать принадлежность персоны к созданию сериалов.

Список серий, а также номера сезонов, которым принадлежат серии хранится в отдельной таблице “series”.

1. Диаграмма отношения сущностей.



1. Заключение

В данной работе была выбрана предметная область для построения модели данных и реализована на языке SQL. Так же была построена диаграмма отношения сущностей, которая показывает содержание таблиц и отношения между ними. Данная модель отвечает третьей нормальной форме, что говорит о зависимости первичного ключа.

В результате данной работы, в приложении А изложена программа реализации модели.

Приложение А

CREATE TABLE `channels` (

`channel\_id` int(10) UNSIGNED NOT NULL,

`channel\_name` varchar(20) NOT NULL

);

CREATE TABLE `countries` (

`country\_id` int(10) UNSIGNED NOT NULL,

`country\_name` varchar(20) NOT NULL

);

CREATE TABLE `genres` (

`genre\_id` int(10) UNSIGNED NOT NULL,

`genre\_name` varchar(20) NOT NULL

);

CREATE TABLE `persons` (

`persons\_id` int(10) UNSIGNED NOT NULL,

`name` varchar(30) NOT NULL,

`It\_is\_producer` tinyint(1) NOT NULL

);

CREATE TABLE `serials\_channels` (

`serials\_id` int(10) UNSIGNED NOT NULL,

`channels\_id` int(10) UNSIGNED NOT NULL

);

CREATE TABLE `serials\_list` (

`id` int(10) UNSIGNED NOT NULL,

`serial\_name` varchar(30) NOT NULL,

`annotation` text NOT NULL,

`genres\_id` int(10) UNSIGNED NOT NULL,

`country\_id` int(10) UNSIGNED NOT NULL,

`status` tinyint(1) NOT NULL,

`release\_date` date NOT NULL

);

CREATE TABLE `serials\_persons` (

`serials\_id` int(10) UNSIGNED NOT NULL,

`person\_id` int(10) UNSIGNED NOT NULL

);

CREATE TABLE `series` (

`series\_id` int(10) UNSIGNED NOT NULL,

`serial\_id` int(10) UNSIGNED NOT NULL,

`season\_number` int(10) UNSIGNED NOT NULL,

`series\_number` int(10) UNSIGNED NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

ALTER TABLE `channels`

ADD PRIMARY KEY (`channel\_id`);

ALTER TABLE `countries`

ADD PRIMARY KEY (`country\_id`);

ALTER TABLE `genres`

ADD PRIMARY KEY (`genre\_id`);

ALTER TABLE `persons`

ADD PRIMARY KEY (`persons\_id`);

ALTER TABLE `serials\_channels`

ADD PRIMARY KEY (`serials\_id`),

ADD KEY `channels\_id` (`channels\_id`);

ALTER TABLE `serials\_list`

ADD PRIMARY KEY (`id`),

ADD KEY `genres\_id` (`genres\_id`,`country\_id`),

ADD KEY `country\_id` (`country\_id`);

ALTER TABLE `serials\_persons`

ADD KEY `serials\_id` (`serials\_id`),

ADD KEY `person\_id` (`person\_id`);

ALTER TABLE `series`

ADD PRIMARY KEY (`series\_id`),

ADD KEY `serial\_id` (`serial\_id`);

ALTER TABLE `channels`

MODIFY `channel\_id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=3;

ALTER TABLE `countries`

MODIFY `country\_id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=3;

ALTER TABLE `genres`

MODIFY `genre\_id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=3;

ALTER TABLE `persons`

MODIFY `persons\_id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=13;

ALTER TABLE `serials\_list`

MODIFY `id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT;

ALTER TABLE `series`

MODIFY `series\_id` int(10) UNSIGNED NOT NULL AUTO\_INCREMENT;

ALTER TABLE `serials\_channels`

ADD CONSTRAINT `serials\_channels\_ibfk\_1` FOREIGN KEY (`serials\_id`) REFERENCES `serials\_list` (`id`),

ADD CONSTRAINT `serials\_channels\_ibfk\_2` FOREIGN KEY (`channels\_id`) REFERENCES `channels` (`channel\_id`);

ALTER TABLE `serials\_list`

ADD CONSTRAINT `serials\_list\_ibfk\_1` FOREIGN KEY (`genres\_id`) REFERENCES `genres` (`genre\_id`),

ADD CONSTRAINT `serials\_list\_ibfk\_2` FOREIGN KEY (`country\_id`) REFERENCES `countries` (`country\_id`);

ALTER TABLE `serials\_persons`

ADD CONSTRAINT `serials\_persons\_ibfk\_1` FOREIGN KEY (`serials\_id`) REFERENCES `serials\_list` (`id`),

ADD CONSTRAINT `serials\_persons\_ibfk\_2` FOREIGN KEY (`person\_id`) REFERENCES `persons` (`persons\_id`);

ALTER TABLE `series`

ADD CONSTRAINT `series\_ibfk\_1` FOREIGN KEY (`serial\_id`) REFERENCES `serials\_list` (`id`);

COMMIT;