**Задание**

**Преподаватель:**Алексей Кузьмин, Николай Хащанов, Екатерина Волочаева, Александр Сумовский

Все работы мы проверяем на антиплагиат, поэтому важно выполнять их самостоятельно. Работам, которые не пройдут проверку, преподаватель поставит незачёт.

Если вы сталкиваетесь со сложностями, задавайте вопросы нашим экспертам и аспирантам, они с удовольствием вам помогут. Успехов в решении заданий!

**Модуль 4. Домашнее задание по теме «Углубление в SQL»**

**Цели домашнего задания:**

* научиться проектировать архитектуру таблиц базы данных, работать с ограничениями и атрибутами полей, заполнять таблицы данными;
* закрепить навыки работы с первичными и внешними ключами;
* продемонстрировать знания по модификации таблиц баз данных.

**Перечень заданий**

**Основная часть**  
База данных: если подключение к облачной базе, то создаёте новую схему с префиксом в виде фамилии, название должно быть на латинице в нижнем регистре и таблицы создаете в этой новой схеме, если подключение к локальному серверу, то создаёте новую схему и в ней создаёте таблицы.

Задание 1. Спроектируйте базу данных, содержащую три справочника:  
· язык (английский, французский и т. п.);  
· народность (славяне, англосаксы и т. п.);  
· страны (Россия, Германия и т. п.).  
Две таблицы со связями: язык-народность и народность-страна, отношения многие ко многим. Пример таблицы со связями — film\_actor.  
Требования к таблицам-справочникам:  
· наличие ограничений первичных ключей.  
· идентификатору сущности должен присваиваться автоинкрементом;  
· наименования сущностей не должны содержать null-значения, не должны допускаться дубликаты в названиях сущностей.  
Требования к таблицам со связями:  
· наличие ограничений первичных и внешних ключей.

В качестве ответа на задание пришлите запросы создания таблиц и запросы по добавлению в каждую таблицу по 5 строк с данными.

**Дополнительная часть**

Задание 1. Создайте новую таблицу film\_new со следующими полями:  
· film\_name — название фильма — тип данных varchar(255) и ограничение not null;  
· film\_year — год выпуска фильма — тип данных integer, условие, что значение должно быть больше 0;  
· film\_rental\_rate — стоимость аренды фильма — тип данных numeric(4,2), значение по умолчанию 0.99;  
· film\_duration — длительность фильма в минутах — тип данных integer, ограничение not null и условие, что значение должно быть больше 0.  
Если работаете в облачной базе, то перед названием таблицы задайте наименование вашей схемы.

Задание 2. Заполните таблицу film\_new данными с помощью SQL-запроса, где колонкам соответствуют массивы данных:  
· film\_name — array[The Shawshank Redemption, The Green Mile, Back to the Future, Forrest Gump, Schindler’s List];  
· film\_year — array[1994, 1999, 1985, 1994, 1993];  
· film\_rental\_rate — array[2.99, 0.99, 1.99, 2.99, 3.99];  
· film\_duration — array[142, 189, 116, 142, 195].

Задание 3. Обновите стоимость аренды фильмов в таблице film\_new с учётом информации, что стоимость аренды всех фильмов поднялась на 1.41.

Задание 4. Фильм с названием Back to the Future был снят с аренды, удалите строку с этим фильмом из таблицы film\_new.

Задание 5. Добавьте в таблицу film\_new запись о любом другом новом фильме.

Задание 6. Напишите SQL-запрос, который выведет все колонки из таблицы film\_new, а также новую вычисляемую колонку «длительность фильма в часах», округлённую до десятых.

Задание 7. Удалите таблицу film\_new.

**Результат домашнего задания**  
Заполните ответами шаблон для сдачи ДЗ в формате .sql.

**Критерии оценки**  
Для зачета необходимо правильно выполнить все задания из основной части.

*Преподаватель вправе предложить дополнительные задачи в рамках задания, чтобы подтвердить, что студент разобрался в теме.  
Преподаватель вправе поставить незачет без права пересдачи текущего задания, если студент прислал на проверку результат чужой работы и отказался делать дополнительное задание.*

Изображение выглядит как текст

Автоматически созданное описание