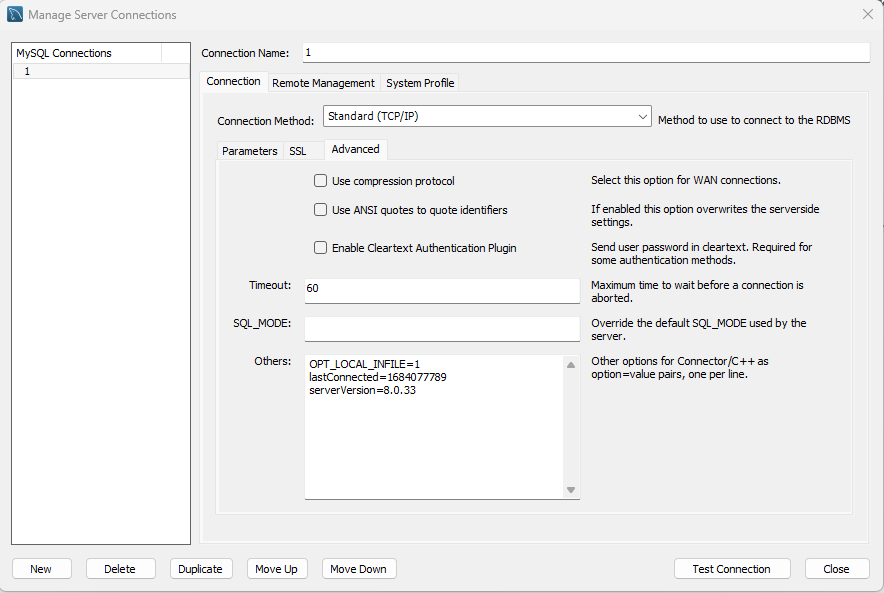
**Все задачи выполняются в MySql.**

1. Создать предварительно две таблицы: промежуточную **stg\_movie** и основную **movie**. Используя скрипты [DDL](https://drive.google.com/drive/folders/1oAu2i5UIs2BMl3ooJwBLm1_SqS8LL-TZ).
2. Загрузить данные из папки [Data](https://drive.google.com/drive/folders/1kj9vXJEMEJ3u2C7SJZkNfGHL-k23Imb9) в таблицу stg\_movie. Используя [Script for load.sql](https://drive.google.com/file/d/1TEU6FOI5OPLVE9MI-FW7JrWkRiadrGOf/view?usp=share_link).   
   При загрузке может возникать ошибка *Error Code: 2068. LOAD DATA LOCAL INFILE file request rejected due to restrictions on access.*

Для её устранения добавляем параметр **OPT\_LOCAL\_INFILE=1** на вкладке подключения.



1. Создать хранимую процедуру [usp\_parseMovies.sql](https://drive.google.com/file/d/177cWfC8KHLe5T92E415k1k7ka3zIBZoq/view?usp=share_link) и вызвать. С её помощью данные из промежуточной таблицы перепишутся в основную.

При запуске может появится ошибка *Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column. To disable safe mode, toggle the option in Preferences -> SQL Editor and reconnect.*

Исправление:

**SET SQL\_SAFE\_UPDATES = 0;**

call usp\_parseMovies ();

Также может появится ошибка *Truncated Incorrect Double Value.* В моём случае она является warning и не мешает загрузке данных. Если у вас это error, то вероятнее всего нужно *turn off strict mode MySql, the error should turn into a warning.* Однако я не проверял, так как у меня это всего лишь Warning.



Хранимая процедура должна гарантированно работать.

Задачи для самостоятельной работы(для извлечения информации о фильмах не используем таблицу stg\_movie):

1. Вывести все уникальные жанры, которые встречаются в фильмах. Отдельно найти все фильмы у которых жанр не указан.
2. Создать таблицы и загрузить в них данные из [Data](https://drive.google.com/drive/folders/1kj9vXJEMEJ3u2C7SJZkNfGHL-k23Imb9) ratings.csv и tags.csv.
3. Найти фильмы, которые имеют средний рейтинг >= 4.0. Посчитать кол-во оценок, которые были получены этими фильмами.  
   Найти фильмы, которые имеют средний рейтинг <= 2.0. Посчитать кол-во оценок, которые были получены этими фильмами.

Вывести title, releaseYear, avgRate, countMark.

1. Найти фильмы, у которых имеется тег **family.** Узнать средний рейтинг для фильмов с таким тегом.   
   Вывести title, releaseYear, avgRate. Найти фильмы, которые содержат в теге слово **bad** или **stupid.**

Вывести title, releaseYear.

1. Найти пользователя(ей), который поставил больше всего оценок фильмам, посчитать среднее значение его оценок.  
   Вывести userId, countMark, avgMark.
2. Написать хранимую процедуру, которая будет принимать в качестве параметров **startYear int, finishYear int, titleTemplate text, genresTemplate** и находить фильмы согласно этим параметрам.   
   То есть фильмы:

* с годом выпуска **between startYear and finishYear**
* у которых в **title** содержится **titleTemplate** (регистром пренебречь)
* у которых в жанрах содержатся все жанры из **genresTemplate**

**genresTemplate** имеет формат: g1|g2|g3 или g1(если жанр один)

Пример:   
**call usp\_getRequireFilms (2010, 2015, "love", "Drama|Romance")**

****

Пояснение: Stuck in Love 2012 имеет жанры **Comedy, Drama, Romance,** следовательно все жанры из **genresTemplate** встречаются.

Love & Mercy 2014 имеет только жанр **Drama**, следовательно не все жанры из **genresTemplate** встречаются. Поэтому фильм не включён в выборку.

Написать хранимую процедуру, которая принимает в качестве параметра **title** и считает рейтинг для этого фильма.   
 Пример:   
**call usp\_getRequireFilms ("Stuck in Love").**

\*\*Дополнительный балл:

1. Определить **Bad Data.** Bad Data считаются те строки, которые не подходят под общую модель обработки и в итоговой таблице имеют ошибки в атрибутах. Например, сохранился неверный год.