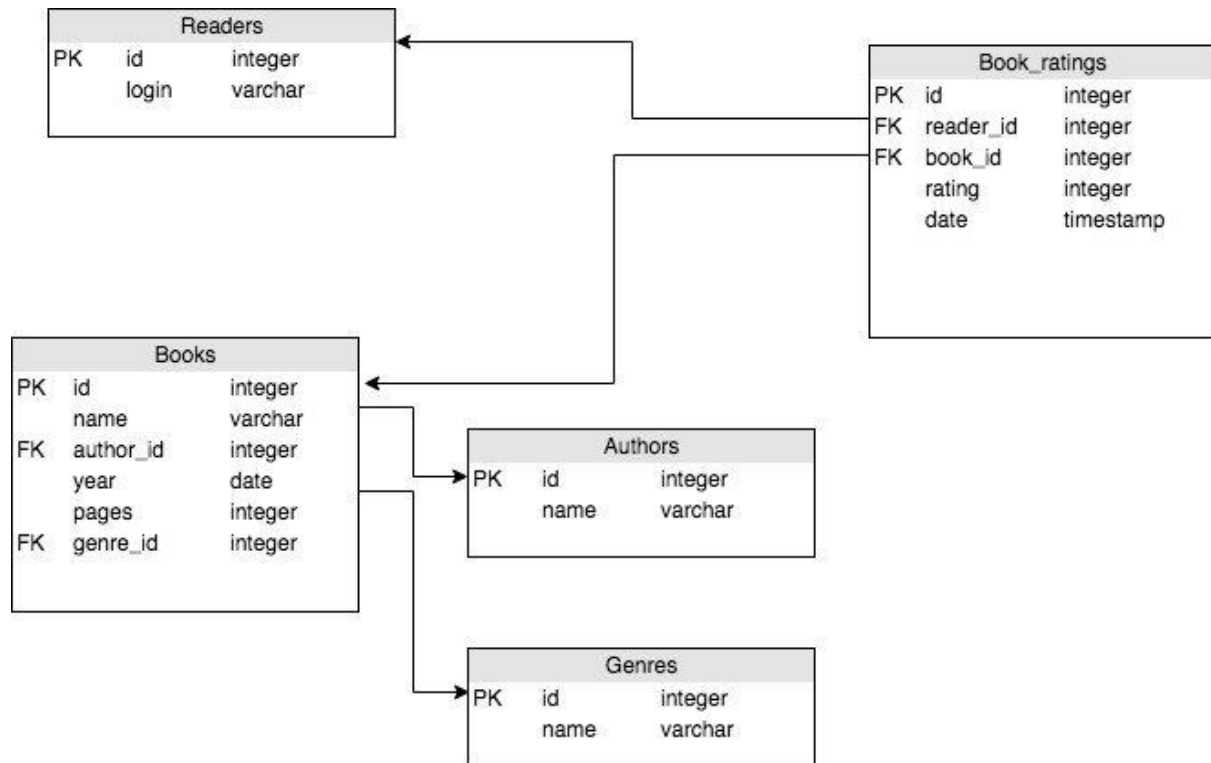


Описание БД

В работе реализована БД для электронной библиотеки (аналог <http://readly.ru/> и подобных ресурсов). Читатели могут выбрать книги определенных авторов и жанров, ознакомиться с информацией по книге и поставить ей оценку, а также посмотреть другие оценки.

БД содержит таблицы с данными читателей, книг, списком авторов и жанров, а также историю оценок книг, поставленных читателями.

Схема бд



Описание таблиц

1. Таблица "Readers" - читатели библиотеки. Содержит

- id (уникальный), числовой, первичный ключ - идентификатор читателя
- login, текст - имя читателя

2. Таблица "Books" - книги. Содержит

- id (уникальный), числовой, первичный ключ - идентификатор книги
- name, текст - название книги
- author_id, числовой, внешний ключ - идентификатор автора из таблицы "Authors"
- year, числовой - год издания
- pages, числовой - количество страниц
- genre_id, числовой, внешний ключ - идентификатор жанра из таблицы "Genres"

3. Таблица "Authors" - список авторов книг. Содержит

- id (уникальный), числовой, первичный ключ - идентификатор автора
- name, текст - ФИО автора

4. Таблица "Genres" - список жанров книг. Содержит

- id (уникальный), числовой, первичный ключ - идентификатор жанра
- name, текст - название жанра

5. Таблица "Book_ratings" - оценки книг, поставленных читателями. Содержит

- id (уникальный), числовой, первичный ключ - идентификатор оценки
- reader_id, числовой, внешний ключ - идентификатор читателя из таблицы "Readers"
- book_id, числовой, внешний ключ - идентификатор книги из таблицы "Books"
- rating, числовой - оценка, поставленная книге, от 0 до 5
- date, дата_время - дата и время выставления оценки

Запросы для создания и наполнения БД

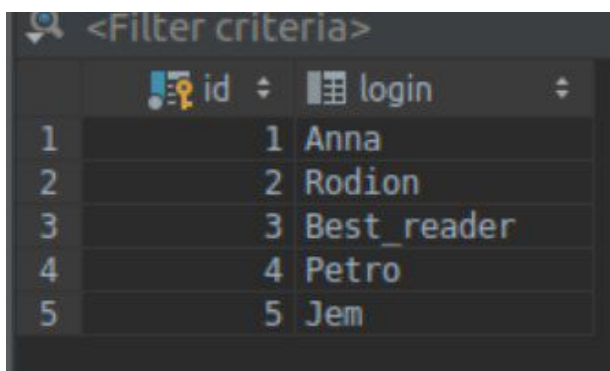
(https://github.com/Jemcha/SQL/blob/master/tables_final.sql):

```
create table Readers (id integer primary key, login varchar);
create table Authors (id integer primary key, name varchar);
create table Genres (id integer primary key, name varchar);

create table Books (id integer primary key, name varchar, author_id integer, genre_id integer,
year integer, pages integer);
alter table Books add foreign key (author_id) references Authors(id);
alter table Books add foreign key (genre_id) references Genres(id);

create table Book_ratings (id integer primary key, reader_id integer, book_id integer, rating
integer, date timestamp);
alter table Book_ratings add foreign key (reader_id) references Readers(id);
alter table Book_ratings add foreign key (book_id) references Books(id);

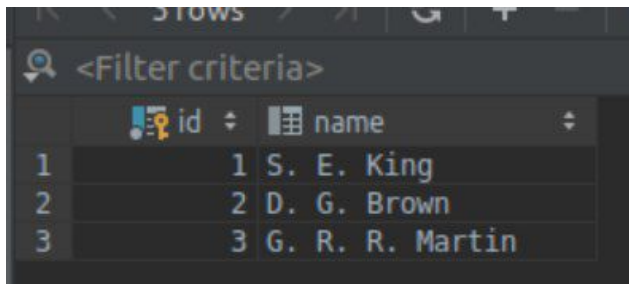
insert into Readers values
('1', 'Anna'),
('2', 'Rodion'),
('3', 'Best_reader'),
('4', 'Petro'),
('5', 'Jem');
```



The screenshot shows a database interface with a table named 'Readers'. The table has two columns: 'id' and 'login'. The data is as follows:

	id	login
1	1	Anna
2	2	Rodion
3	3	Best_reader
4	4	Petro
5	5	Jem

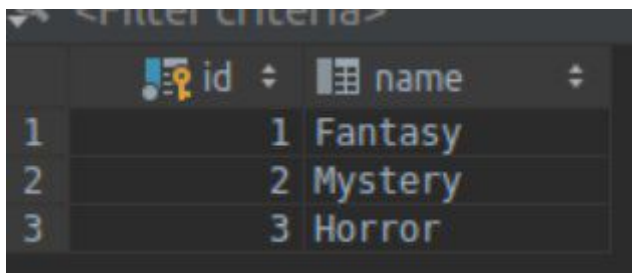
```
insert into Authors values
('1', 'S. E. King'),
('2', 'D. G. Brown'),
('3', 'G. R. R. Martin');
```



The screenshot shows a database table with the title "<Filter criteria>". The table has two columns: 'id' and 'name'. The data is as follows:

	id	name
1	1	S. E. King
2	2	D. G. Brown
3	3	G. R. R. Martin

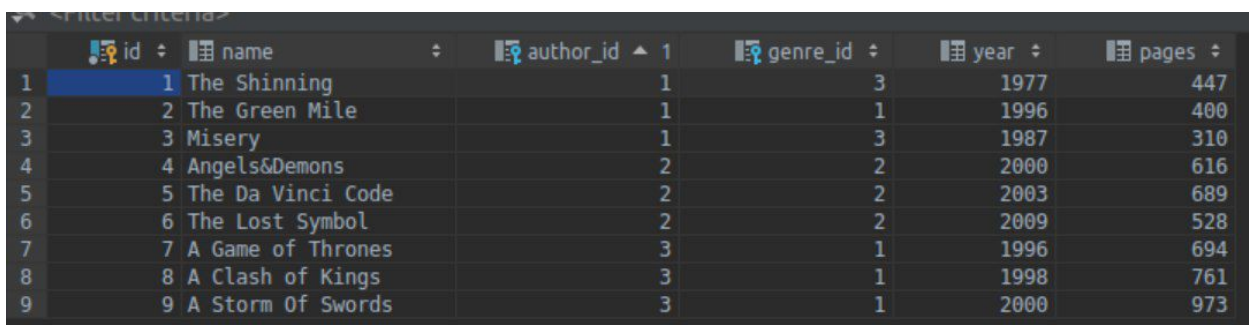
```
insert into Genres values
('1', 'Fantasy'),
('2', 'Mystery'),
('3', 'Horror');
```



The screenshot shows a database table with the title "<Filter criteria>". The table has two columns: 'id' and 'name'. The data is as follows:

	id	name
1	1	Fantasy
2	2	Mystery
3	3	Horror

```
insert into Books values
('1', 'The Shinning', '1', '3', '1977', '447'),
('2', 'The Green Mile', '1', '1', '1996', '400'),
('3', 'Misery', '1', '3', '1987', '310'),
('4', 'Angels&Demons', '2', '2', '2000', '616'),
('5', 'The Da Vinci Code', '2', '2', '2003', '689'),
('6', 'The Lost Symbol', '2', '2', '2009', '528'),
('7', 'A Game of Thrones', '3', '1', '1996', '694'),
('8', 'A Clash of Kings', '3', '1', '1998', '761'),
('9', 'A Storm Of Swords', '3', '1', '2000', '973');
```



The screenshot shows a database table with the title "<Filter criteria>". The table has six columns: 'id', 'name', 'author_id', 'genre_id', 'year', and 'pages'. The data is as follows:

	id	name	author_id	genre_id	year	pages
1	1	The Shinning	1	3	1977	447
2	2	The Green Mile	1	1	1996	400
3	3	Misery	1	3	1987	310
4	4	Angels&Demons	2	2	2000	616
5	5	The Da Vinci Code	2	2	2003	689
6	6	The Lost Symbol	2	2	2009	528
7	7	A Game of Thrones	3	1	1996	694
8	8	A Clash of Kings	3	1	1998	761
9	9	A Storm Of Swords	3	1	2000	973

```
insert into Book_ratings values
('1', '1', '1', '4', '22-02-2019 12:54:55'),
('2', '1', '2', '3', '23-02-2019 17:42:50'),
```

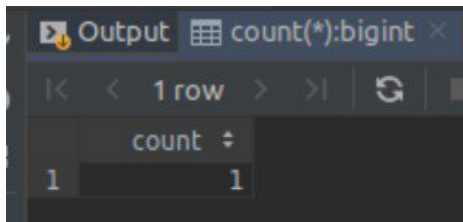
```
( '3', '1', '4', '5', '12-02-2019 13:30:55'),
( '4', '1', '8', '4', '23-02-2019 18:51:31'),
( '5', '2', '1', '5', '11-02-2019 14:54:45'),
( '6', '2', '3', '3', '05-02-2019 05:58:35'),
( '7', '3', '5', '4', '21-02-2019 10:11:22'),
( '8', '3', '7', '5', '22-02-2019 12:23:55'),
( '9', '3', '8', '4', '20-02-2019 20:54:34'),
( '10', '3', '9', '4', '21-02-2019 11:23:55'),
( '11', '3', '1', '3', '22-02-2019 15:45:01'),
( '12', '4', '2', '4', '23-02-2019 09:23:58'),
( '13', '5', '1', '4', '18-02-2019 16:54:43'),
( '14', '5', '2', '5', '21-02-2019 19:13:55'),
( '15', '5', '3', '4', '22-02-2019 21:57:25');
```

<Filter criteria>					
	id	reader_id	book_id	rating	date
1	1	1	1	4	2019-02-22 12:54:55.000000
2	2	1	2	3	2019-02-23 17:42:50.000000
3	3	1	4	5	2019-02-12 13:30:55.000000
4	4	1	8	4	2019-02-23 18:51:31.000000
5	5	2	1	5	2019-02-11 14:54:45.000000
6	6	2	3	3	2019-02-05 05:58:35.000000
7	7	3	5	4	2019-02-21 10:11:22.000000
8	8	3	7	5	2019-02-22 12:23:55.000000
9	9	3	8	4	2019-02-20 20:54:34.000000
10	10	3	9	4	2019-02-21 11:23:55.000000
11	11	3	1	3	2019-02-22 15:45:01.000000
12	12	4	2	4	2019-02-23 09:23:58.000000
13	13	5	1	4	2019-02-18 16:54:43.000000
14	14	5	2	5	2019-02-21 19:13:55.000000
15	15	5	3	4	2019-02-22 21:57:25.000000

Результаты выполнения запросов (запросы:

https://github.com/lemcha/SQL/edit/master/Final_project.sql) :

1. Непрочитанные книги - Посчитать число книг без оценок



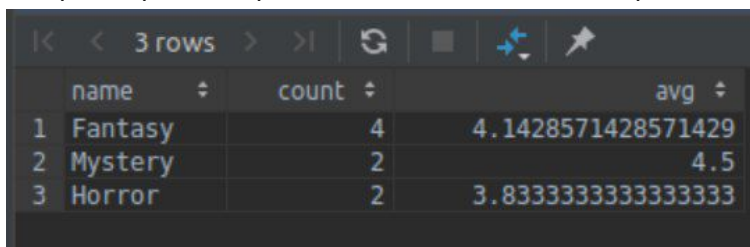
Output count(*) bigint x

count
1

2. Рейтинг книг - Вывести список всех книг и их средний рейтинг

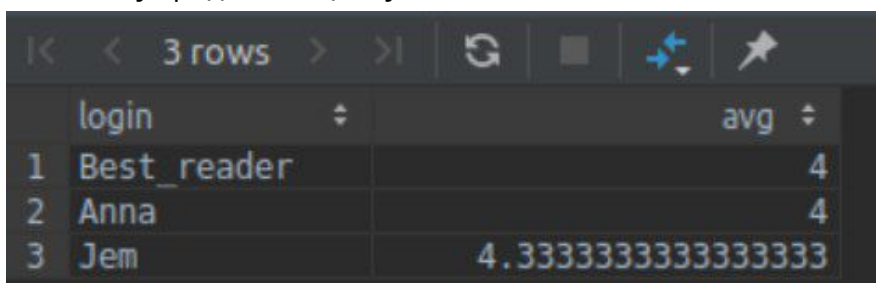
	name	avg
1	A Clash of Kings	4
2	The Green Mile	4
3	A Game of Thrones	5
4	The Shinning	4
5	Angels&Demons	5
6	A Storm Of Swords	4
7	The Lost Symbol	<null>
8	Misery	3.5
9	The Da Vinci Code	4

3. Статистика по жанрам - Вывести список названий жанров, количество книг этого жанра и средний рейтинг книг каждого жанра



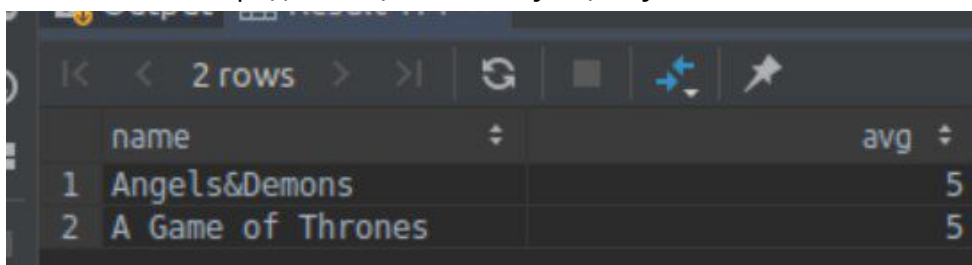
	name	count	avg
1	Fantasy	4	4.1428571428571429
2	Mystery	2	4.5
3	Horror	2	3.8333333333333333

4. Лояльные читатели - Вывести 3 пользователей, у которых средняя оценка выше 3.5, и самую среднюю оценку



	login	avg
1	Best_reader	4
2	Anna	4
3	Jem	4.3333333333333333

5. Самые высокооцениваемые книги - Вывести список названий книг с максимальной средней оценкой и эту оценку



	name	avg
1	Angels&Demons	5
2	A Game of Thrones	5

6. Самые активные читатели и их средняя оценка - Вывести список пользователей с максимальным количеством оценок и среднюю оценку

	login	cnt	avg
1	Best_reader	5	4

7. Вывести названия книг, у которых минимум 3 оценки и средний рейтинг выше 3.5

	name	avg	cnt
1	The Green Mile	4	3
2	The Shinning	4	4

8. Статистика по авторам - Вывести список авторов, количество их книг и средний рейтинг книг этого автора

	name	count	avg
1	G. R. R. Martin	3	4.25
2	S. E. King	3	3.8888888888888889
3	D. G. Brown	3	4.5

9. Авторы, работающие в нескольких жанрах - Вывести список авторов, у которых есть книги хотя бы в 2 жанрах и количество их книг

	name	cnt
1	S. E. King	3

10. Вывести список названий книг с указанием автора, которые изданы после 1995 и до 2000

	name	name	year
1	S. E. King	The Green Mile	1996
2	G. R. R. Martin	A Clash of Kings	1998
3	G. R. R. Martin	A Game of Thrones	1996