SQL. Анализ базы данных сервиса для чтения книг по подписке

Вводные данные

Компания решила запустить новый продукт - приложение для тех кто любит читать продук. Для этого Компания купила крупный сервис для чтения книг по подписке.

Нам доступна база данных, в которой содержится информация о книгах, издательствах, авторах, а также пользовательские обзоры книг.

Предполагается, что эти данные помогут сформулировать ценностное предложение для нового продукта - приложения для тех кто любит читать.

Цель исследования:

Получение конкретных данных для формулировки ценностного предложения для нового продукта.

Задачи:

Анализ иформации из базы данных сервиса для чтения книг по подписке и получение ответов на вопросы:

- посчитать сколько книг вышло после 1 января 2000 года;
- посчитать какое количество обзоров и какая средняя оценка для каждой книги;
- определить издательство, которое выпустило наибольшее число книг толще 50 страниц (это исключит из анализа брошюры);
- определить автора с самой высокой средней оценкой книг — учитывать только книги с 50 и более оценками;
- Посчитать среднее количество обзоров от пользователей, которые поставили больше 48 оценок.

План проекта

1. Импорт библиотек. Настройка параметров доступа к БД
2. Исследование таблицы — вывести первые строки, посчитать количество строк в каждой таблице;
3. Решение поставленных задач:
    - посчитать сколько книг вышло после 1 января 2000 года;
    - посчитать какое количество обзоров и какая средняя оценка для каждой книги ;
    - определить издательство, которое выпустило наибольшее число книг толще 50 страниц (это исключит из анализа брошюры);
    - определить автора с самой высокой средней оценкой книг — учитывать только книги с 50 и более оценками;
    - Посчитать среднее количество обзоров от пользователей, которые поставили больше 48 оценок.

      Решение каждой задачи оформлять по алгоритму:
        - SQL-запрос
        - вывод результата запроса
        - описание выводов по результату запроса
4. Итоги анализа

Описание данных

Таблицы: `books`, `authors`, `publishers`, `ratings`, `reviews`

Таблица `books` Содержит данные о книгах:

- `book_id` — идентификатор книги;
- `author_id` — идентификатор автора;
- `title` — название книги;
- `num_pages` — количество страниц;
- `publication_date` — дата публикации книги;
- `publisher_id` — идентификатор издателя.

Таблица `authors` Содержит данные об авторах:

- `author_id` — идентификатор автора;
- `author` — имя автора.

Таблица `publishers` Содержит данные об издательствах:

- `publisher_id` — идентификатор издательства;
- `publisher` — название издательства; **Таблица `ratings`** Содержит данные о пользовательских оценках книг:
- `rating_id` — идентификатор оценки;
- `book_id` — идентификатор книги;
- `username` — имя пользователя, оставившего оценку;
- `rating` — оценка книги. **Таблица `reviews`** Содержит данные о пользовательских обзорах на книги:
- `review_id` — идентификатор обзора;
- `book_id` — идентификатор книги;
- `username` — имя пользователя, написавшего обзор;
- `text` — текст обзора.

Материалы [Картинка ER-диаграммы]

Импорт библиотек. Настройка параметров доступа к БД

In [1]:

```
#имортируем бибилиотеки
import pandas as pd
import sqlalchemy as sa
#для запуска локально возможно потребуется
!pip install psycopg2
import os
from sqlalchemy import text, create_engine
#снимаем ограничение на ширину столбцов
pd.set_option('display.max_colwidth', None)
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: psycopg2 in c:\users\79181\appdata\roaming\python\python39\site-packages (2.9.9)
```

In [2]:

```
#установка библиотеки python-dotenv
!pip install python-dotenv
#поиск файла .env и загрузка из него переменных среды
from dotenv import load_dotenv
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: python-dotenv in c:\users\79181\appdata\roaming\python\python39\site-packages (1.0.1)
```

In [3]:

```
#грузим пароли запуская поиск файла .env локально
load_dotenv()
```

Out[3]:

```
True
```

In [4]:

```
#устанавливаем параметры
# получаем параметры из хранимого в специальном скрытом файле .env - не храним парол
явном виде
```

```
db_config = {
    'user': 'praktikum_student',#имя пользователя
    'pwd': os.getenv('pwd'), #пароль  подгружается из локального файла
    'host': 'rc1b-wcoijxj3yxfsf3fs.mdb.yandexcloud.net',
    'port': 6432, #порт подключения
    'db': 'data-analyst-final-project-db' #название базы данных
}

connection_string = 'postgresql://{user}:{pwd}@{host}:{port}/{db}'.format(**db_confi
```

In [6]:

```
# сохраняем коннектор
engine = create_engine(connection_string, connect_args={'sslmode':'require'})
```

Исследование таблиц

Выведем первые строки, посчитать количество строк в каждой из представленной таблиц:

- books,
- authors,
- publishers,
- ratings,
- reviews

Функция для запросов get_sql_data

Так как предстоит много запросов, то для удобства выполнения SQL-запроса, напишем функцию с использованием Pandas.

In [7]:

```
# функцию для SQL зароса
def get_sql_data(query:str, engine:sa.engine.base.Engine=engine) -> pd.DataFrame:
    '''Открываем соединение, получаем данные из sql, закрываем соединение'''
    with engine.connect() as con:
        return pd.read_sql(sql=sa.text(query), con = con)
```

Исследование таблицы books

Выведем первые строки, затем посчитаем количество строк в каждой таблице.

In [8]:

```
#формируем запрос
query = '''

SELECT *
 FROM books
LIMIT 5;

'''
# вызываем функцию
get_sql_data(query)
```

Out[8]:

	book_id	author_id	title	num_pages	publication_date	publisher_id
0	1	546	'Salem's Lot	594	2005-11-01	93
1	2	465	1 000 Places to See Before You Die	992	2003-05-22	336

2	3	407	13 Little Blue Envelopes (Little Blue Envelope #1)	322	2010-12-21	135
3	4	82	1491: New Revelations of the Americas Before Columbus	541	2006-10-10	309
4	5	125	1776	386	2006-07-04	268

In [9]:

```
# формируем запрос
query = '''

SELECT COUNT(*)
 FROM books;

'''
# вызываем функцию
print(f"Количество строк в таблице:\n{get_sql_data(query)}")
```

Количество строк в таблице:

	count
0	1000

Исследование таблицы `authors`

Выведем первые строки, затем посчитаем количество строк в каждой таблице;

In [10]:

```
# формируем запрос
query = '''

SELECT *
 FROM authors
LIMIT 5;

'''
# вызываем функцию
get_sql_data(query)
```

Out[10]:

	author_id	author
0	1	A.S. Byatt
1	2	Aesop/Laura Harris/Laura Gibbs
2	3	Agatha Christie
3	4	Alan Brennert
4	5	Alan Moore/David Lloyd

In [11]:

```
# формируем запрос
query = '''

SELECT COUNT(*)
 FROM authors;

'''
```

```
# вызываем функцию
print(f"Количество строк в таблице:\n{get_sql_data(query)}")
```

Количество строк в таблице:
   count
0 636

Исследование таблицы `publishers`

Выведем первые строки, затем посчитаем количество строк в каждой таблице;

In [12]:

```
# формируем запрос
query = '''

SELECT *
 FROM publishers
LIMIT 5;

'''
# вызываем функцию
get_sql_data(query)
```

Out[12]:

	publisher_id	publisher
0	1	Ace
1	2	Ace Book
2	3	Ace Books
3	4	Ace Hardcover
4	5	Addison Wesley Publishing Company

In [13]:

```
# формируем запрос
query = '''

SELECT COUNT(*)
 FROM publishers;

'''
# вызываем функцию
print(f"Количество строк в таблице:\n{get_sql_data(query)}")
```

Количество строк в таблице:
   count
0 340

Исследование таблицы `ratings`

Выведем первые строки, затем посчитаем количество строк в каждой таблице;

In [14]:

```
# формируем запрос
query = '''

SELECT *
 FROM ratings
LIMIT 5;

'''
# вызываем функцию
get_sql_data(query)
```

Out[14]:

	rating_id	book_id	username	rating
0	1	1	ryanfranco	4
1	2	1	grantpatricia	2
2	3	1	brandtandrea	5
3	4	2	lorichen	3
4	5	2	mariokeller	2

In [15]:

```
# формируем запрос
query = '''

SELECT COUNT(*)
 FROM ratings;

'''
# вызываем функцию
print(f"Количество строк в таблице:\n{get_sql_data(query)}")
```

Количество строк в таблице:
   count
0 6456

Исследование таблицы `reviews`

Выведем первые строки, затем посчитаем количество строк в каждой таблице;

In [16]:

```
# формируем запрос
query = '''

SELECT *
 FROM reviews
LIMIT 5;

'''
# вызываем функцию
get_sql_data(query)
```

Out[16]:

	review_id	book_id	username	text
0	1	1	brandtandrea	Mention society tell send professor analysis. Over provide race technology continue these.
1	2	1	ryanfranco	Foot glass pretty audience hit themselves. Among admit investment argue security.
2	3	2	lorichen	Listen treat keep worry. Miss husband tax but person sport treatment industry. Kitchen decision deep the. Social party body the.
3	4	3	johnsonamanda	Finally month interesting blue could nature cultural bit. Prepare beat finish grow that smile teach. Dream me play near.
4	5	3	scotttamara	Nation purpose heavy give wait song will. List dinner another whole positive radio fast. Music staff many green.

In [17]:

```
# формируем запрос
query = '''

SELECT COUNT(*)
 FROM reviews;

'''
# вызываем функцию
print(f"Количество строк в таблице:\n{get_sql_data(query)}")
```

Количество строк в таблице:
   count
0   2793

Итоги исследования таблиц

Ознакомились с данными, которые содержатся в таблицах. Для выполнения запросов использовали написанную функцию `get_sql_data`.

По каждой из таблиц вывели по пять строк и подсчитали сколько строк содержит таблица. Результаты исследования:

- 1000 строк содержит таблица `books`,
- 636 строк содержит таблица `authors`,
- 340 строк содержит таблица `publishers`,
- 6456 строк содержит таблица `ratings`,
- 2793 строк содержит таблица `reviews`

Получение данных

Сформируем запросы и получим ответы по следующим вопросам:

- посчитать сколько книг вышло после 1 января 2000 года;
- посчитать какое количество обзоров и какая средняя оценка для каждой книги ;
- определить издательство, которое выпустило наибольшее число книг толще 50 страниц (это исключит из анализа брошюры);
- определить автора с самой высокой средней оценкой книг — учитывать только книги с 50 и более оценками;
- посчитать среднее количество обзоров от пользователей, которые поставили больше 48 оценок.

Сколько книг вышло после 1 января 2000 года

In [18]:

```
# формируем запрос
query = '''

SELECT COUNT(books.book_id)
 FROM books
WHERE books.publication_date > '2000-01-01';

'''
# вызываем функцию
get_sql_data(query)
```

Out[18]:

	count
0	819

Выводы по результату запроса

После 1 января 2000 года вышло 819 книг.

Можно уточнить у заказчика, включаем мы 1 января или нет.

Посчитать какое количество обзоров и какая средняя оценка для каждой книги

Посчитаем количество обзоров для каждой книги и выведем результаты в столбце `reviews_count`.

Посчитаем среднюю оценку для каждой книги и выведем результаты в столбце `avg_rating` в порядке убывания.

In [19]:

```
# формируем запрос
query = '''

SELECT b.book_id,
      b.title,
      COUNT(DISTINCT rw.review_id) AS reviews_count,
      ROUND(AVG(r.rating),3) AS avg_rating
 FROM books AS b
LEFT JOIN reviews AS rw ON b.book_id = rw.book_id
LEFT JOIN ratings AS r ON b.book_id = r.book_id
GROUP BY b.book_id, b.title
ORDER BY avg_rating DESC;

'''
# вызываем функцию
get_sql_data(query)
```

Out[19]:

	book_id	title	reviews_count	avg_rating
0	86	Arrows of the Queen (Heralds of Valdemar #1)	2	5.00
1	901	The Walking Dead Book One (The Walking Dead #1-12)	2	5.00
2	390	Light in August	2	5.00

3	972	Wherever You Go There You Are: Mindfulness Meditation in Everyday Life	2	5.00
4	136	Captivating: Unveiling the Mystery of a Woman's Soul	2	5.00
...	...	...	...	...
995	915	The World Is Flat: A Brief History of the Twenty-first Century	3	2.25
996	316	His Excellency: George Washington	2	2.00
997	202	Drowning Ruth	3	2.00
998	371	Junky	2	2.00
999	303	Harvesting the Heart	2	1.50

1000 rows × 4 columns

Выводы по результату запроса

Посчитали количество обзоров для каждой книги и вывели результаты в столбце `reviews_count`. Средняя оценка для книг выведена в столбце avg_rating. Минимальная оценка 1.5, максимальная 5.0.
Обзоров больше всего на Сумерки.

Определить издательство, которое выпустило наибольшее число книг, толще 50 страниц

Исключение книг тоньше 50 страниц позволяет не учитывать брошюры.

In [20]:

```
# формируем запрос
query = '''

WITH t AS
   (SELECT books.publisher_id,
           books.book_id
      FROM books
     WHERE books.num_pages > 50)

SELECT t.publisher_id,
      p.publisher
 FROM t
INNER JOIN publishers AS p ON t.publisher_id = p.publisher_id
GROUP BY t.publisher_id, p.publisher
ORDER BY COUNT(t.book_id) DESC
LIMIT 1;
'''
# вызываем функцию
get_sql_data(query)
```

Out[20]:

	publisher_id	publisher
0	212	Penguin Books

Выводы по результату запроса

`Penguin Books` издательство, которое выпустило наибольшее число книг толще 50 страниц - 212 штук.

Определить автора с самой высокой средней оценкой книг только книги с 50 и более оценками

In [21]:

```
# формируем запрос
query = '''

WITH t AS
   (SELECT book_id
      FROM ratings
    GROUP BY book_id
    HAVING COUNT(rating_id) >= 50)

SELECT a.author_id,
      a.author,
      AVG(r.rating) AS avg_rating
 FROM books AS b
INNER JOIN t ON b.book_id = t.book_id
INNER JOIN authors AS a ON b.author_id = a.author_id
INNER JOIN ratings AS r ON t.book_id = r.book_id
GROUP BY a.author_id, a.author
ORDER BY AVG(r.rating) DESC
LIMIT 1;

'''
# вызываем функцию
get_sql_data(query)
```

Out[21]:

	author_id	author	avg_rating
0	236	J.K. Rowling/Mary GrandPré	4.287097

Выводы по результату запроса

Автор с самой высокой средней оценкой книг, имеющих 50 и более оценок - это `J.K. Rowling/Mary GrandPré`.

Посчитать среднее количество обзоров от пользователей, которые поставили больше 48 оценок

Посчитаем среднее количество обзоров от пользователей, который поставили болеее 48 оценок и выведем результат как `avg_rw_count`.

In [22]:

```
# формируем запрос
query = '''

SELECT COUNT(DISTINCT rw.text)/COUNT(DISTINCT rw.username) AS avg_rw_count
 FROM ratings AS r
INNER JOIN reviews AS rw ON r.username = rw.username
WHERE r.username in (SELECT ratings.username
                       FROM ratings
                   GROUP BY ratings.username
                   HAVING COUNT(ratings.rating_id) > 48);
```

```
'''
# вызываем функцию
get_sql_data(query)
```

Out[22]:

	avg_rw_count
0	24

Выводы по результату запроса
Среднее количество обзоров от пользователей, которые поставили более 48 оценок равно 24.

Итоги

Исследовали таблицы базы данных, в которой содержится информация о книгах, издательствах, авторах, а также пользовательские обзоры книг. Для выполнения запросов использовали написанную функцию `get_sql_data`.

По каждой из таблиц вывели по пять строк и подсчитали сколько строк содержит таблица. Результаты исследования:

- 1000 строк содержит таблица `books`,
- 636 строк содержит таблица `authors`,
- 340 строк содержит таблица `publishers`,
- 6456 строк содержит таблица `ratings`,
- 2793 строк содержит таблица `reviews`

С помощью SQL запросов получена следующая информация:

- `819` книг вышло после 1 января 2000 года;
- `1.5 -5.0` среднии оценки для книг;
- `Penguin Books` - издательство, которое выпустило наибольшее число книг толще 50 страниц - 212 штук;
- `J.K. Rowling/Mary GrandPré` - автор с самой высокой средней оценкой книг, имеющих 50 и более оценок;
- `24` - число среднего количества обзоров от пользователей, которые поставили болеее 48 оценок.