Задание 6
Лим В. БПИ225

Задача 1:

Заполним данными базу:

```
INSERT INTO "Publisher" ("PubName", "PubAddress") VALUES
('Publisher1', 'Moscow, Red Square 1'),
('Publisher2', 'St. Petersburg, Nevsky Prospect 10'),
('Publisher3', 'New York, 5th Avenue 500'),
('Publisher4', 'Berlin, Alexanderplatz 20'),
('Publisher5', 'London, Oxford Street 15'),
('Publisher6', 'Paris, Champs-Élysées 45'),
('Publisher7', 'Tokyo, Shibuya 8'),
('Publisher8', 'Sydney, George Street 35'),
('Publisher9', 'Rome, Via Nazionale 10'),
('Publisher10', 'Madrid, Gran Vía 25'),
('Publisher11', 'Moscow, Tverskaya 5'),
('Publisher12', 'St. Petersburg, Neva Embankment 100'),
('Publisher13', 'New York, Park Avenue 450'),
('Publisher14', 'Berlin, Unter den Linden 50'),
('Publisher15', 'London, Piccadilly Circus 5'),
('Publisher16', 'Paris, Rue de Rivoli 30'),
('Publisher17', 'Tokyo, Ginza 2'),
('Publisher18', 'Sydney, Pitt Street 20'),
('Publisher19', 'Rome, Piazza di Spagna 1'),
('Publisher20', 'Madrid, Paseo de la Castellana 120');


INSERT INTO "Reader" ("ID", "LastName", "FirstName",
"Address", "BirthDate") VALUES
(1, 'Ivanov', 'Ivan', 'Moscow, Tverskaya 3', '1985-03-15'),
(2, 'Petrov', 'Petr', 'Moscow, Arbat 15', '1990-06-20'),
(3, 'Sidorov', 'Sidr', 'Moscow, Pushkin Square 7',
'1980-01-25'),
(4, 'Vasiliev', 'Vladimir', 'St. Petersburg, Nevsky Prospect
8', '1992-11-11'),
(5, 'Dmitriev', 'Dmitry', 'St. Petersburg, Rubinstein Street
18', '1982-05-05'),
(6, 'Kuznetsov', 'Nikolai', 'New York, Brooklyn 100',
'1989-12-01'),
(7, 'Morozov', 'Alexey', 'Berlin, Friedrichstrasse 10',
'1987-08-16'),
```

```
(8, 'Smirnov', 'Mikhail', 'London, Regent Street 55',
'1995-07-21'),
(9, 'Stepanova', 'Anastasia', 'Paris, Boulevard Saint-Germain
12', '1984-09-10'),
(10, 'Karpov', 'Andrei', 'Tokyo, Akihabara 7', '1988-03-03'),
(11, 'Gorshkov', 'Evgeny', 'Moscow, Sadovaya 20',
'1991-11-11'),
(12, 'Romanov', 'Maksim', 'St. Petersburg, Vosstaniya Street
21', '1993-02-22'),
(13, 'Novikov', 'Sergey', 'New York, Manhattan 200',
'1983-07-30'),
(14, 'Petrova', 'Irina', 'Berlin, Kreuzberg 5', '1981-04-25'),
(15, 'Tikhonova', 'Elena', 'London, Baker Street 22',
'1990-10-09'),
(16, 'Sokolov', 'Dmitry', 'Paris, Rue Saint-Honoré 45',
'1986-01-11'),
(17, 'Alekseev', 'Vladimir', 'Tokyo, Shinjuku 3',
'1989-02-05'),
(18, 'Zaharova', 'Svetlana', 'Sydney, Darling Harbour 12',
'1983-12-01'),
(19, 'Mikhailov', 'Vladislav', 'Rome, Via del Corso 18',
'1985-06-15'),
(20, 'Fedorova', 'Olga', 'Madrid, Plaza Mayor 7',
'1991-01-12');

INSERT INTO "Book" ("ISBN", "Title", "Author", "PagesNum",
"PubYear", "PubName") VALUES
('978-1-234-56789-0', 'Book Title 1', 'Author A', 300, 2010,
'Publisher1'),
('978-1-234-56789-1', 'Book Title 2', 'Author B', 450, 2015,
'Publisher2'),
('978-1-234-56789-2', 'Book Title 3', 'Author C', 210, 2018,
'Publisher3'),
('978-1-234-56789-3', 'Book Title 4', 'Author A', 320, 2020,
'Publisher4'),
('978-1-234-56789-4', 'Book Title 5', 'Author D', 500, 2019,
'Publisher5'),
('978-1-234-56789-5', 'Book Title 6', 'Author E', 280, 2012,
'Publisher6'),
('978-1-234-56789-6', 'Book Title 7', 'Author F', 360, 2017,
'Publisher7'),
('978-1-234-56789-7', 'Book Title 8', 'Author G', 220, 2013,
'Publisher8'),
```

```
('978-1-234-56789-8', 'Book Title 9', 'Author H', 480, 2021,
'Publisher9'),
('978-1-234-56789-9', 'Book Title 10', 'Author I', 390, 2010,
'Publisher10'),
('978-1-234-56789-10', 'Book Title 11', 'Author J', 150, 2008,
'Publisher11'),
('978-1-234-56789-11', 'Book Title 12', 'Author A', 310, 2016,
'Publisher12'),
('978-1-234-56789-12', 'Book Title 13', 'Author K', 270, 2014,
'Publisher13'),
('978-1-234-56789-13', 'Book Title 14', 'Author L', 430, 2011,
'Publisher14'),
('978-1-234-56789-14', 'Book Title 15', 'Author M', 350, 2020,
'Publisher15'),
('978-1-234-56789-15', 'Book Title 16', 'Author N', 460, 2019,
'Publisher16'),
('978-1-234-56789-16', 'Book Title 17', 'Author O', 280, 2018,
'Publisher17'),
('978-1-234-56789-17', 'Book Title 18', 'Author P', 500, 2017,
'Publisher18'),
('978-1-234-56789-18', 'Book Title 19', 'Author Q', 300, 2015,
'Publisher19'),
('978-1-234-56789-19', 'Book Title 20', 'Author R', 320, 2021,
'Publisher20');

INSERT INTO "Category" ("CategoryName", "ParentCat") VALUES
('Fiction', NULL),
('Science Fiction', 'Fiction'),
('Fantasy', 'Fiction'),
('Biography', NULL),
('History', NULL),
('Science', NULL),
('Romance', NULL),
('Mystery', NULL),
('Adventure', 'Fiction'),
('Thriller', 'Mystery');

INSERT INTO "Copy" ("ISBN", "CopyNumber", "ShelfPosition")
VALUES
('978-1-234-56789-0', 1, 'A1'),
('978-1-234-56789-1', 1, 'B2'),
('978-1-234-56789-2', 1, 'C3'),
('978-1-234-56789-3', 1, 'D4'),
('978-1-234-56789-4', 1, 'E5'),
```

```
('978-1-234-56789-5', 1, 'F6'),
('978-1-234-56789-6', 1, 'G7'),
('978-1-234-56789-7', 1, 'H8'),
('978-1-234-56789-8', 1, 'I9'),
('978-1-234-56789-9', 1, 'J10'),
('978-1-234-56789-10', 1, 'K11'),
('978-1-234-56789-11', 1, 'L12'),
('978-1-234-56789-12', 1, 'M13'),
('978-1-234-56789-13', 1, 'N14'),
('978-1-234-56789-14', 1, 'O15'),
('978-1-234-56789-15', 1, 'P16'),
('978-1-234-56789-16', 1, 'Q17'),
('978-1-234-56789-17', 1, 'R18'),
('978-1-234-56789-18', 1, 'S19'),
('978-1-234-56789-19', 1, 'T20');

INSERT INTO "Borrowing" ("ReaderNr", "ISBN", "CopyNumber",
"ReturnDate") VALUES
(1, '978-1-234-56789-0', 1, '2024-12-10'),
(2, '978-1-234-56789-1', 1, '2024-12-11'),
(3, '978-1-234-56789-2', 1, '2024-12-12'),
(4, '978-1-234-56789-3', 1, '2024-12-13'),
(5, '978-1-234-56789-4', 1, '2024-12-14'),
(6, '978-1-234-56789-5', 1, '2024-12-15'),
(7, '978-1-234-56789-6', 1, '2024-12-16'),
(8, '978-1-234-56789-7', 1, '2024-12-17'),
(9, '978-1-234-56789-8', 1, '2024-12-18'),
(10, '978-1-234-56789-9', 1, '2024-12-19'),
(11, '978-1-234-56789-10', 1, '2024-12-20'),
(12, '978-1-234-56789-11', 1, '2024-12-21'),
(13, '978-1-234-56789-12', 1, '2024-12-22'),
(14, '978-1-234-56789-13', 1, '2024-12-23'),
(15, '978-1-234-56789-14', 1, '2024-12-24'),
(16, '978-1-234-56789-15', 1, '2024-12-25'),
(17, '978-1-234-56789-16', 1, '2024-12-26'),
(18, '978-1-234-56789-17', 1, '2024-12-27'),
(19, '978-1-234-56789-18', 1, '2024-12-28'),
(20, '978-1-234-56789-19', 1, '2024-12-29');

INSERT INTO "BookCat" ("ISBN", "CategoryName") VALUES
('978-1-234-56789-0', 'Fiction'),
('978-1-234-56789-1', 'Science Fiction'),
('978-1-234-56789-2', 'Fantasy'),
('978-1-234-56789-3', 'Biography'),
```

```
('978-1-234-56789-4', 'History'),
('978-1-234-56789-5', 'Science'),
('978-1-234-56789-6', 'Romance'),
('978-1-234-56789-7', 'Mystery'),
('978-1-234-56789-8', 'Adventure'),
('978-1-234-56789-9', 'Thriller'),
('978-1-234-56789-10', 'Fiction'),
('978-1-234-56789-11', 'Science Fiction'),
('978-1-234-56789-12', 'Fantasy'),
('978-1-234-56789-13', 'Biography'),
('978-1-234-56789-14', 'History'),
('978-1-234-56789-15', 'Science'),
('978-1-234-56789-16', 'Romance'),
('978-1-234-56789-17', 'Mystery'),
('978-1-234-56789-18', 'Adventure'),
('978-1-234-56789-19', 'Thriller');
```

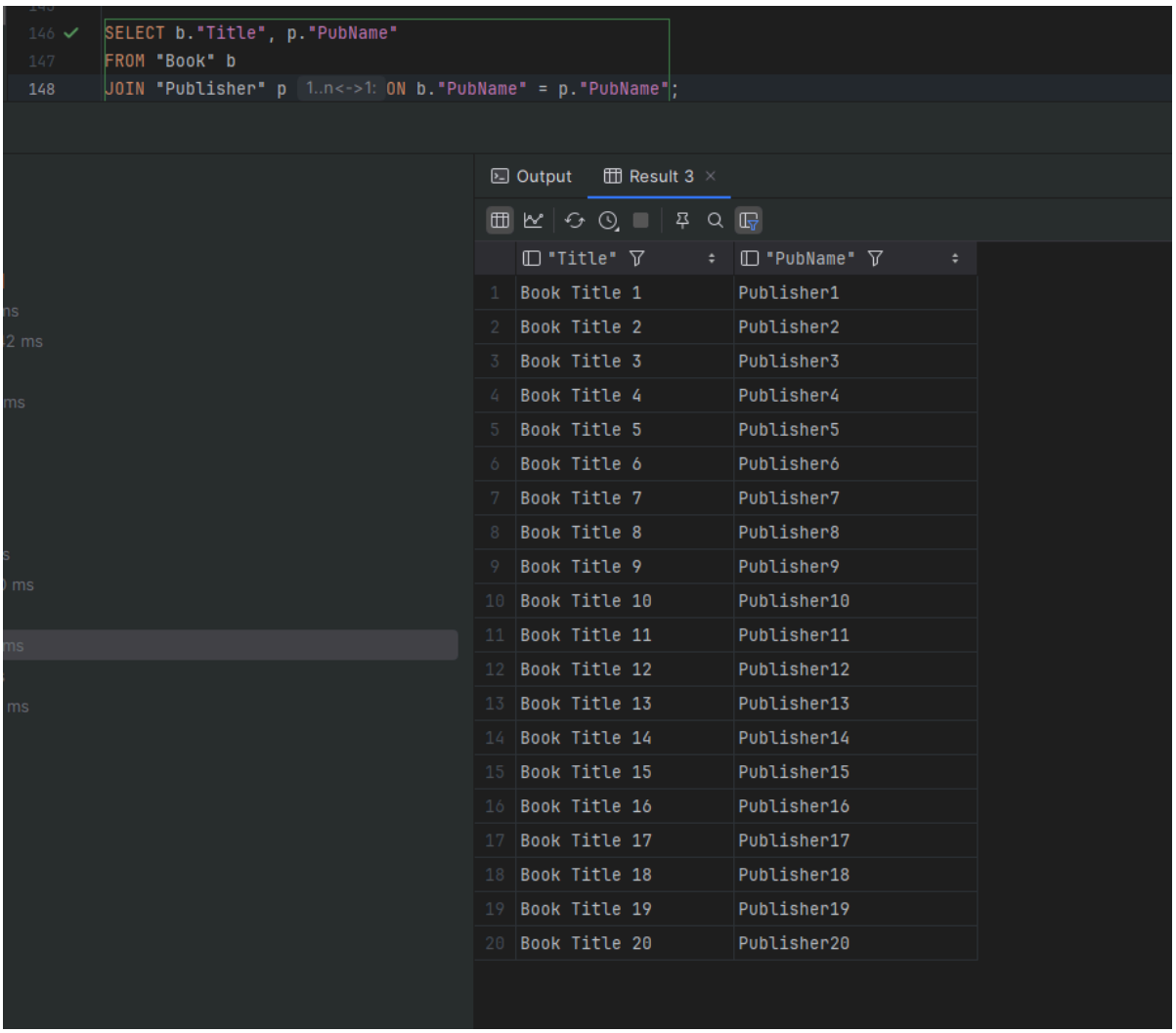
1. Показать все названия книг вместе с именами издателей.

У нас 20 книг и у каждой один издатель.

Запрос:

```
SELECT b."Title", p."PubName"
FROM "Book" b
JOIN "Publisher" p ON b."PubName" = p."PubName";
```

Результаты:



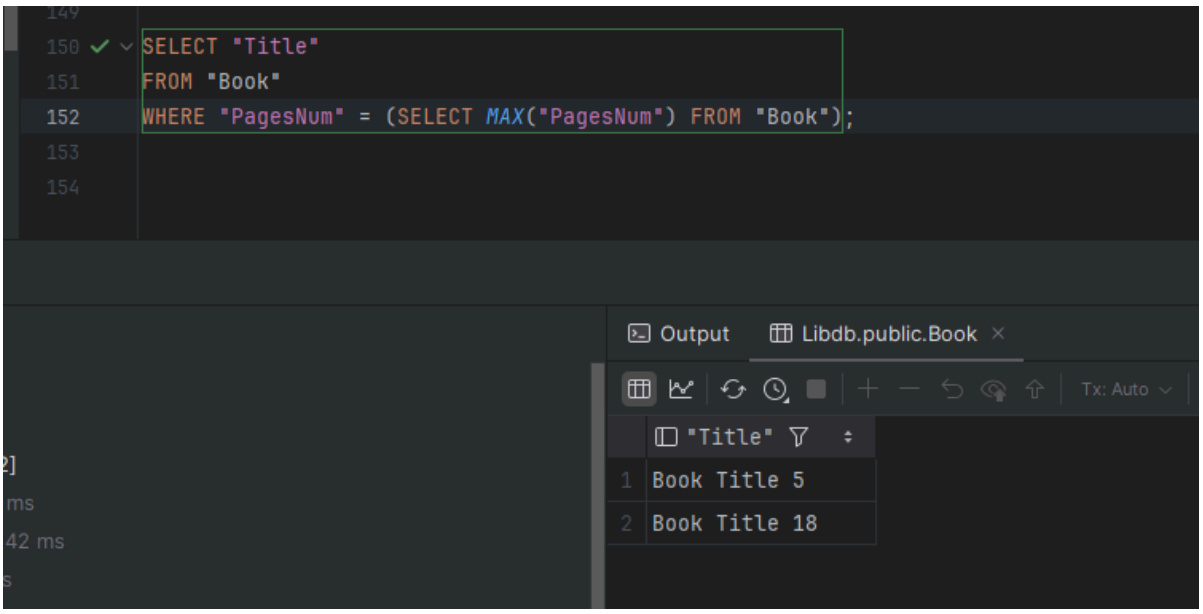
2. В какой книге наибольшее количество страниц?

У нас две книги по 500 страниц.

Запрос:

```
SELECT "Title"
FROM "Book"
WHERE "PagesNum" = (SELECT MAX("PagesNum") FROM "Book");
```

Результат:



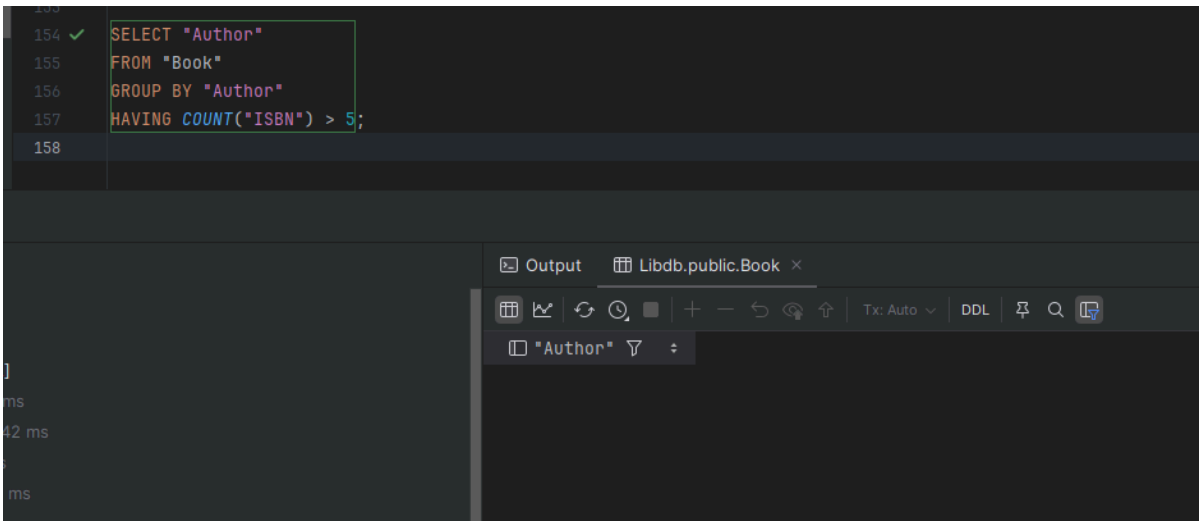
3. Какие авторы написали более 5 книг?

У нас нет тех, кто написал больше 5 книг.

Запрос:

```
SELECT "Author"
FROM "Book"
GROUP BY "Author"
HAVING COUNT("ISBN") > 5;
```

Результат:



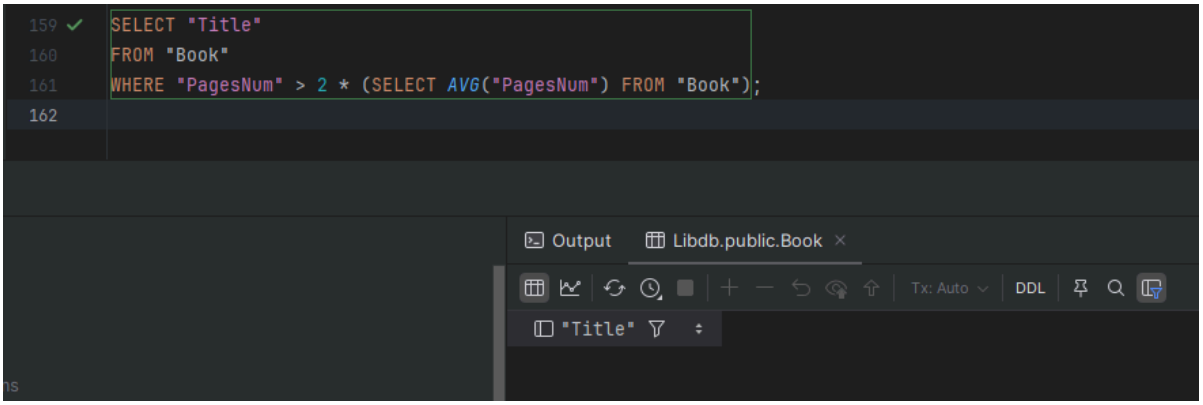
4. В каких книгах более чем в два раза больше страниц, чем среднее количество страниц для всех книг?

У нас нет книг, в которых количество страниц больше в два раза, чем среднее, потому что среднее больше 250, а максимальное количество у нас 500.

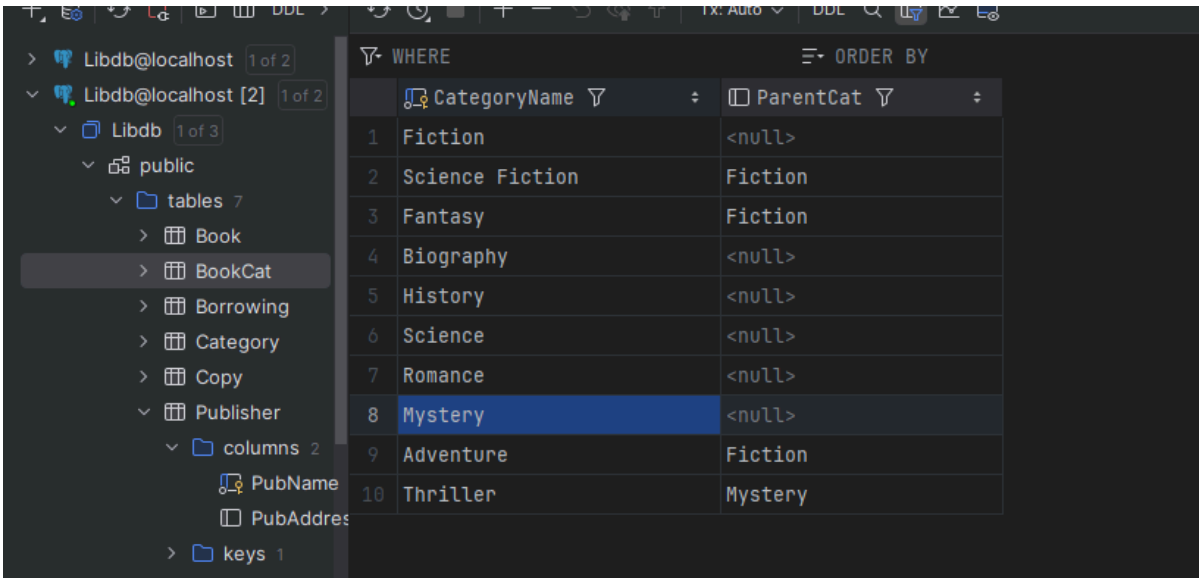
Запрос:

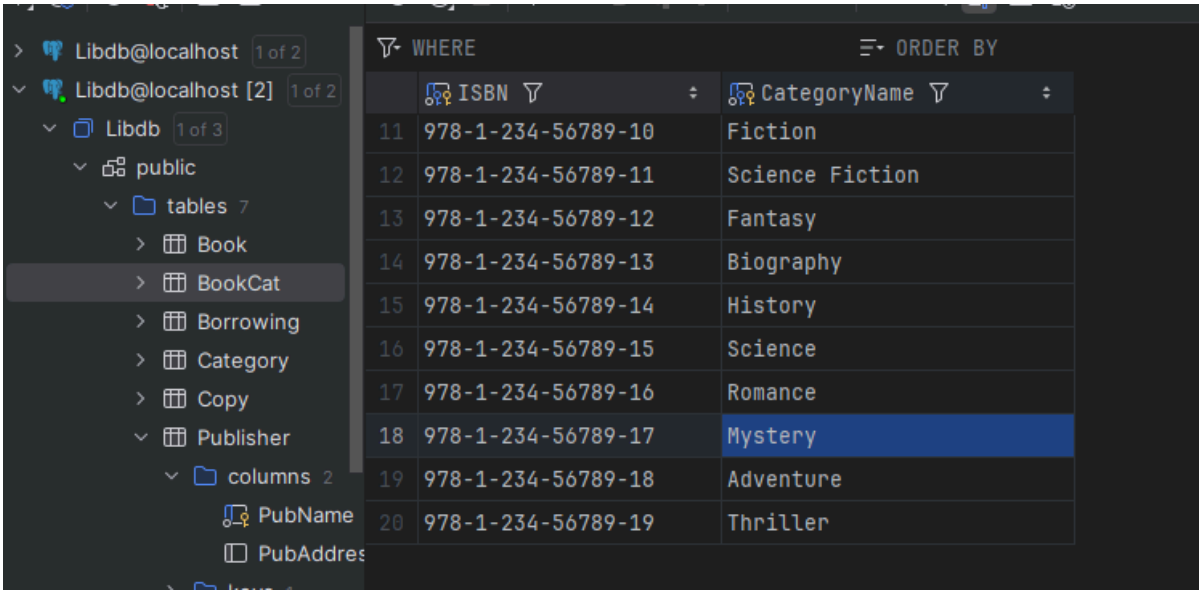
```
SELECT "Title"
FROM "Book"
WHERE "PagesNum" > 2 * (SELECT AVG("PagesNum") FROM "Book");
```

Результат:



5. Какие категории содержат подкатегории?
У нас такие категории, остальное - подкатегории наших:

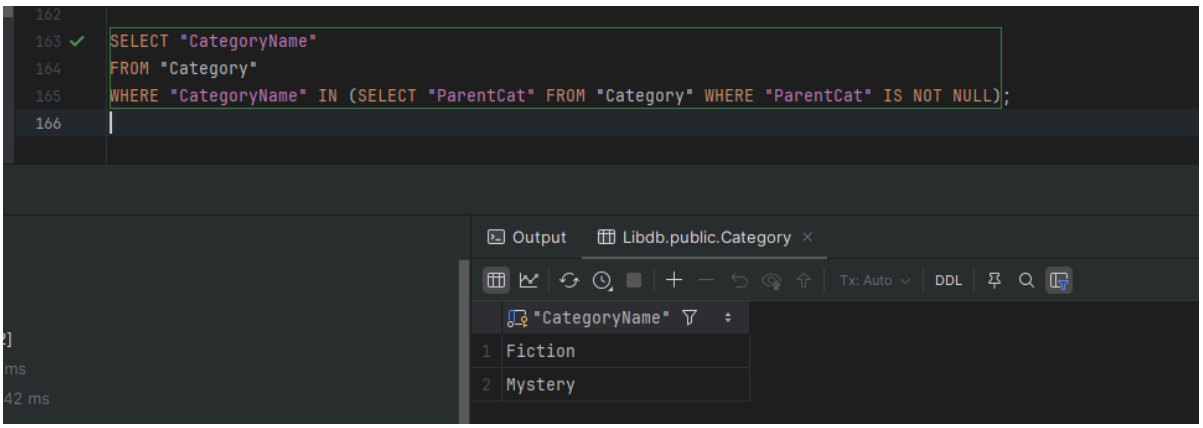




Запрос:

```
SELECT "CategoryName"
FROM "Category"
WHERE "CategoryName" IN (SELECT "ParentCat" FROM "Category" WHERE
"ParentCat" IS NOT NULL);
```

Результат:



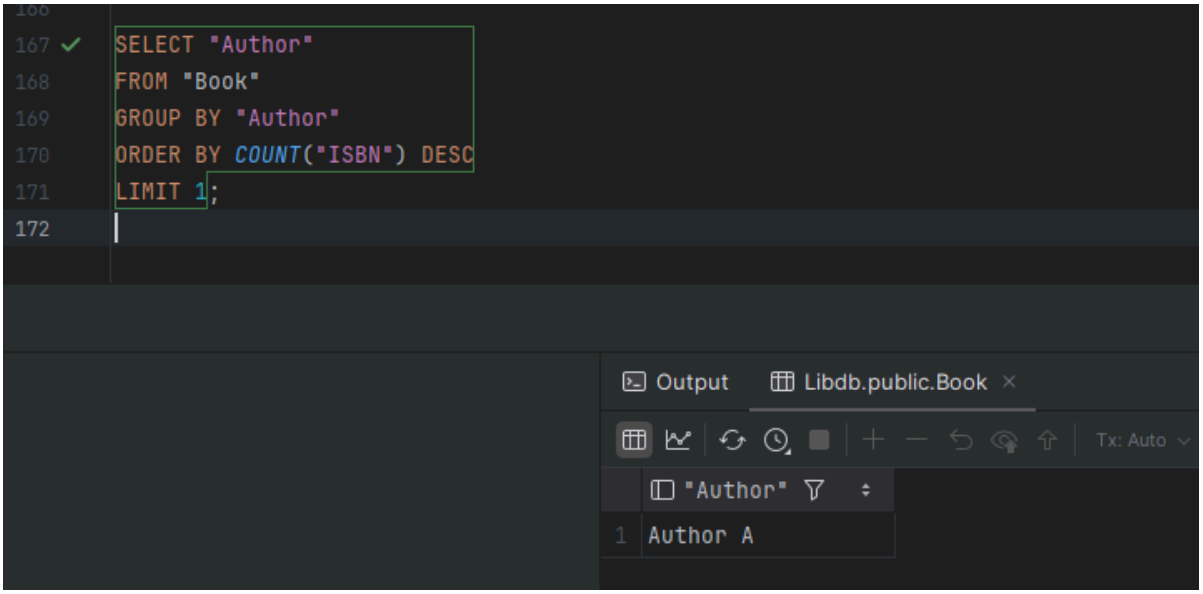
6. У какого автора написано максимальное количество книг?

Упорядочим по количеству и выберем первого.

Запрос:

```
SELECT "Author"
FROM "Book"
GROUP BY "Author"
ORDER BY COUNT("ISBN") DESC
LIMIT 1;
```

Результат:



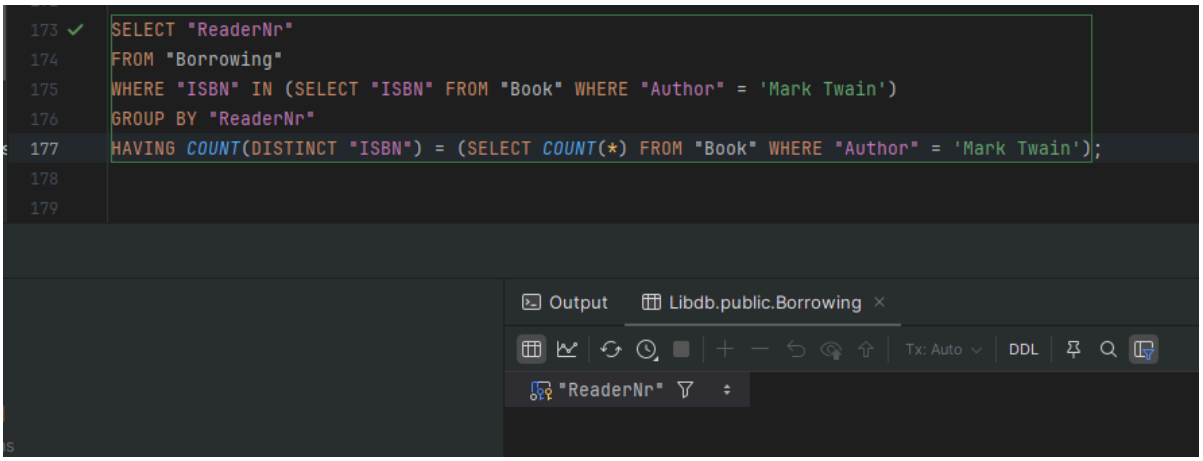
7. Какие читатели забронировали все книги (не копии), написанные "Марком Твеном"?

Нет у нас таких, к сожалению…

Запрос:

```
SELECT "ReaderNr"
FROM "Borrowing"
WHERE "ISBN" IN (SELECT "ISBN" FROM "Book" WHERE "Author" = 'Mark Twain')
GROUP BY "ReaderNr"
HAVING COUNT(DISTINCT "ISBN") = (SELECT COUNT(*) FROM "Book" WHERE
"Author" = 'Mark Twain');
```

Результат:



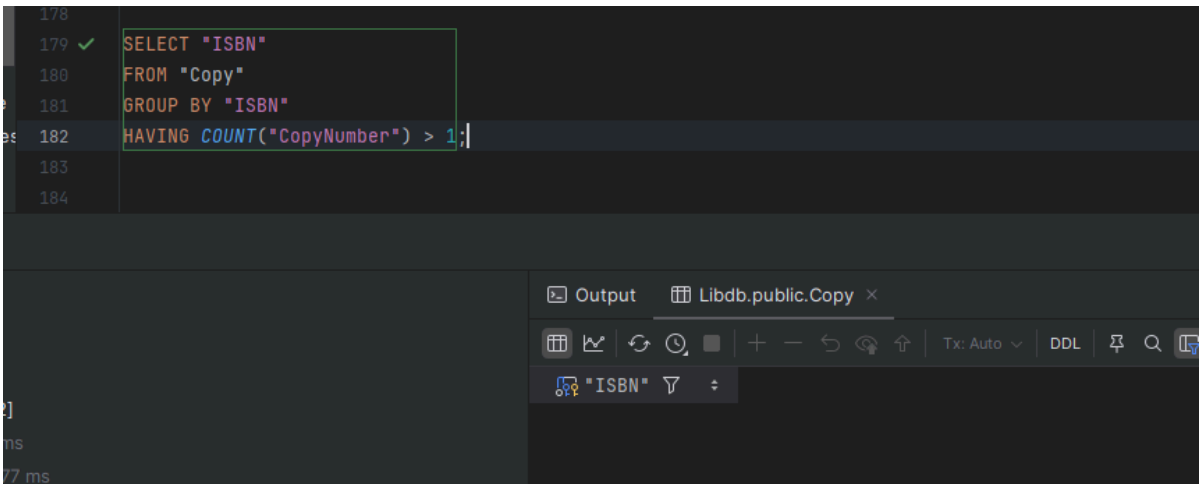
8. Какие книги имеют более одной копии?

Все у нас по одной копии имеют.

Запрос:

```
SELECT "ISBN"
FROM "Copy"
GROUP BY "ISBN"
HAVING COUNT("CopyNumber") > 1;
```

Результат:



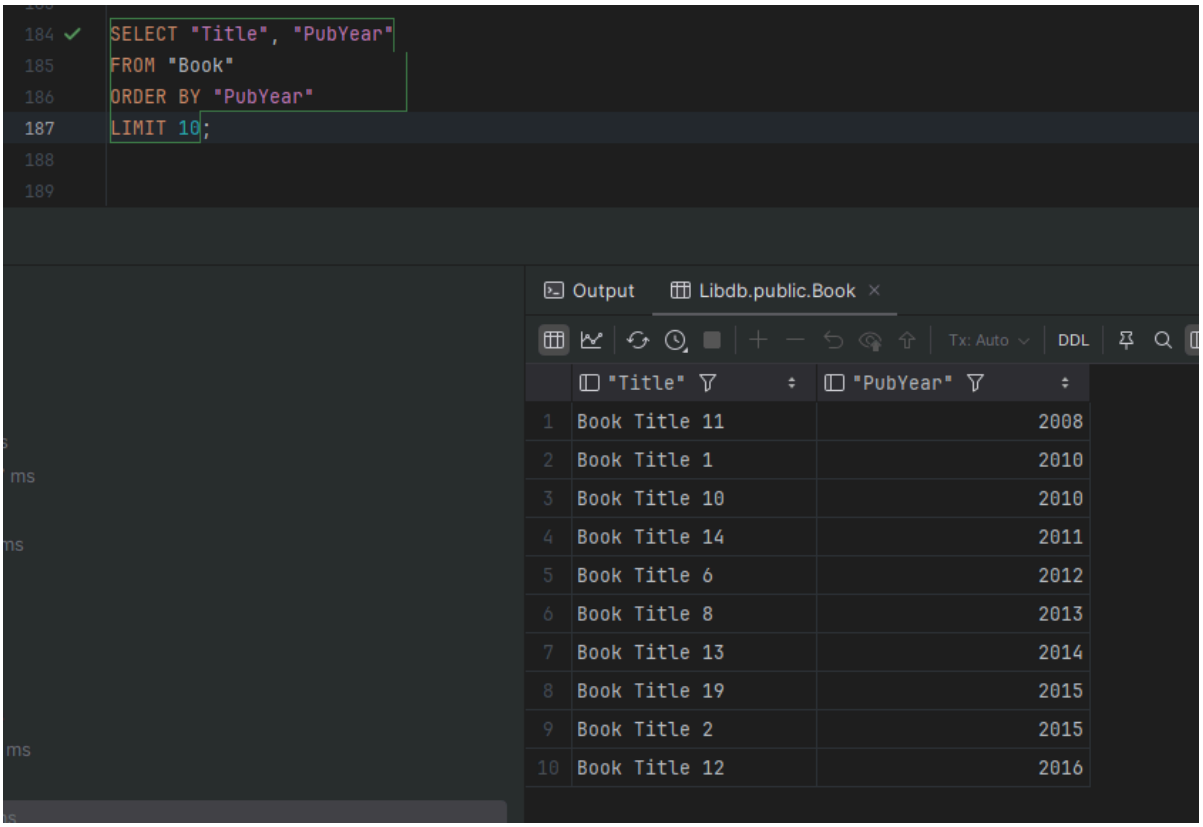
9. ТОП 10 самых старых книг.

Упорядочиваем книги по году публикации и берем первые 10.

Запрос:

```
SELECT "Title", "PubYear"
FROM "Book"
ORDER BY "PubYear"
LIMIT 10;
```

Результат:



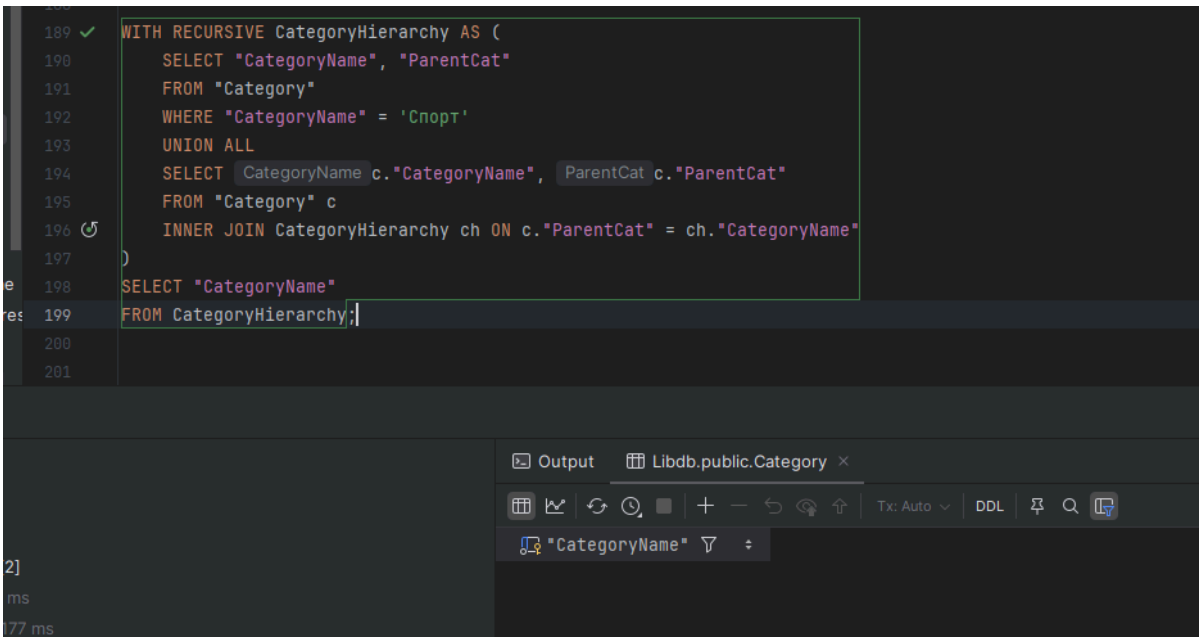
10. Перечислите все категории в категории “Спорт” (с любым уровнем вложености).

У нас

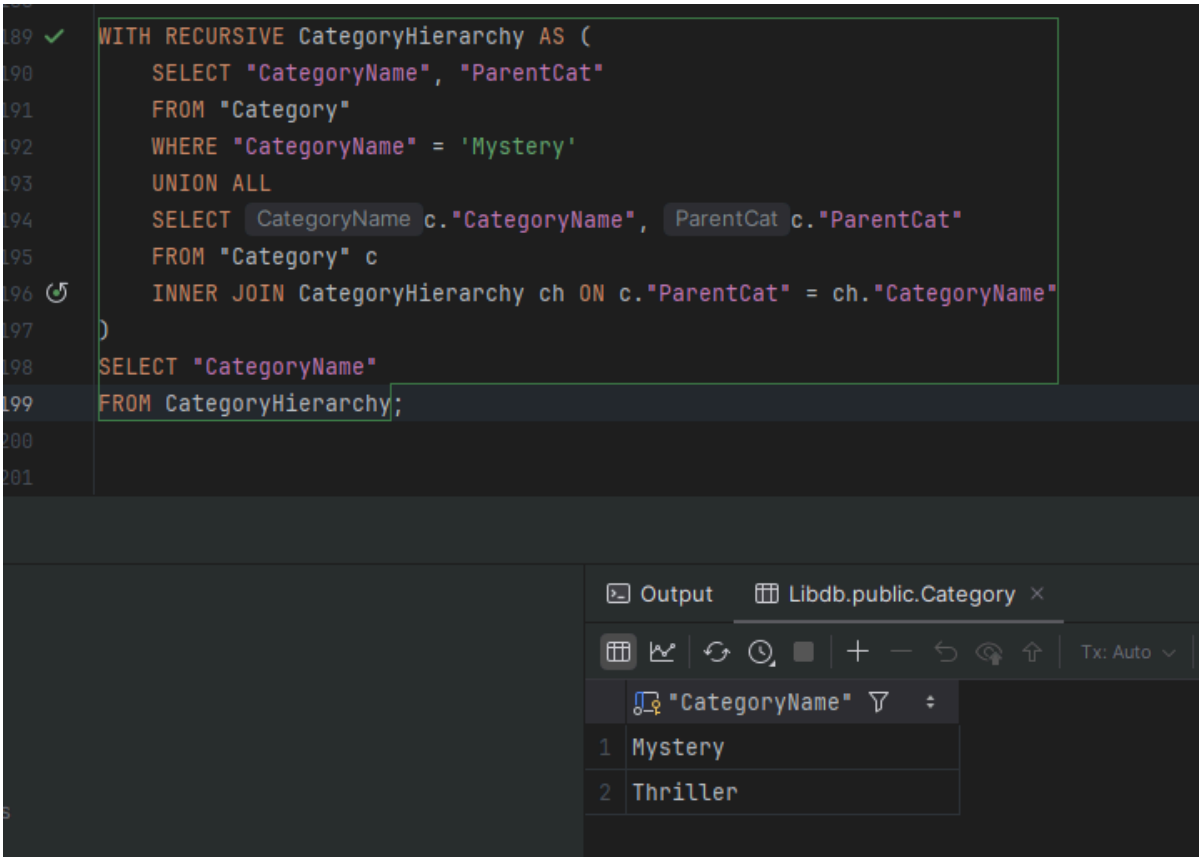
Запрос:

```
WITH RECURSIVE CategoryHierarchy AS (
    SELECT "CategoryName", "ParentCat"
    FROM "Category"
    WHERE "CategoryName" = 'Спорт'
    UNION ALL
    SELECT c."CategoryName", c."ParentCat"
    FROM "Category" c
    INNER JOIN CategoryHierarchy ch ON c."ParentCat" = ch."CategoryName"
)
SELECT "CategoryName"
FROM CategoryHierarchy;
```

Результат:



У нас категории “Спорт” нет, но можем для Mystery запустить:



Задача 2:

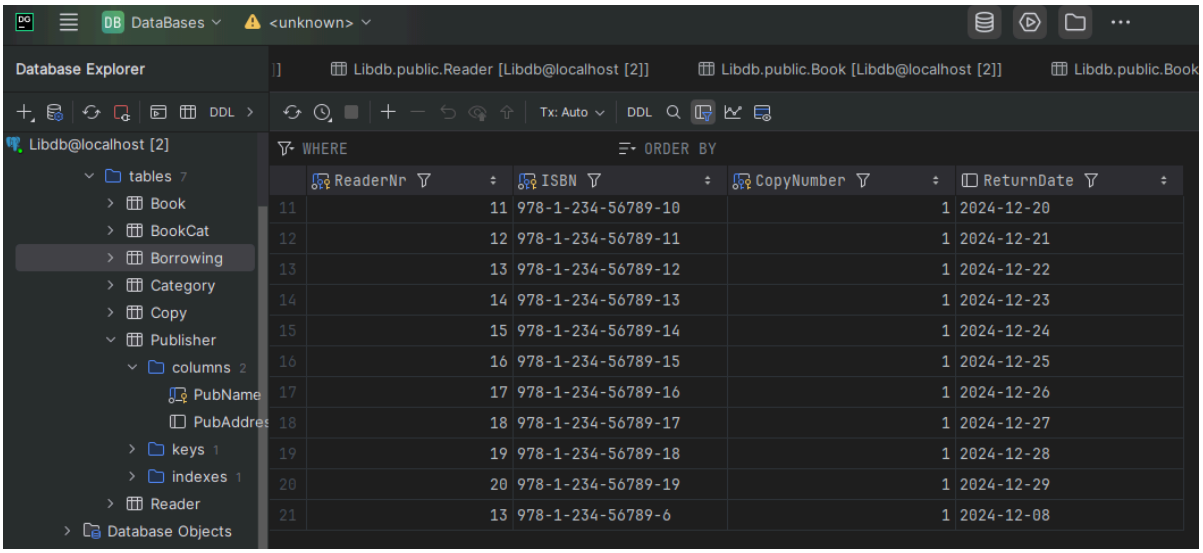
1. Добавьте запись о бронировании читателем ‘Василеем Петровым’ книги с ISBN 123456 и номером копии 4.

Сделаем то же самое, только для какого-нибудь человека, который есть в бд и книгу и копию, которая тоже есть в бд.

Запрос:

```
WITH reader AS (
   SELECT "ID"
   FROM "Reader"
   WHERE "LastName" = 'Novikov' AND "FirstName" = 'Sergey'
)

INSERT INTO "Borrowing" ("ReaderNr", "ISBN", "CopyNumber", "ReturnDate")
SELECT "ID", '978-1-234-56789-6', 1, '2024-12-08'  FROM reader;
```

Результат:



У нас внизу добавилась запись. У Novikov Sergey id=13:

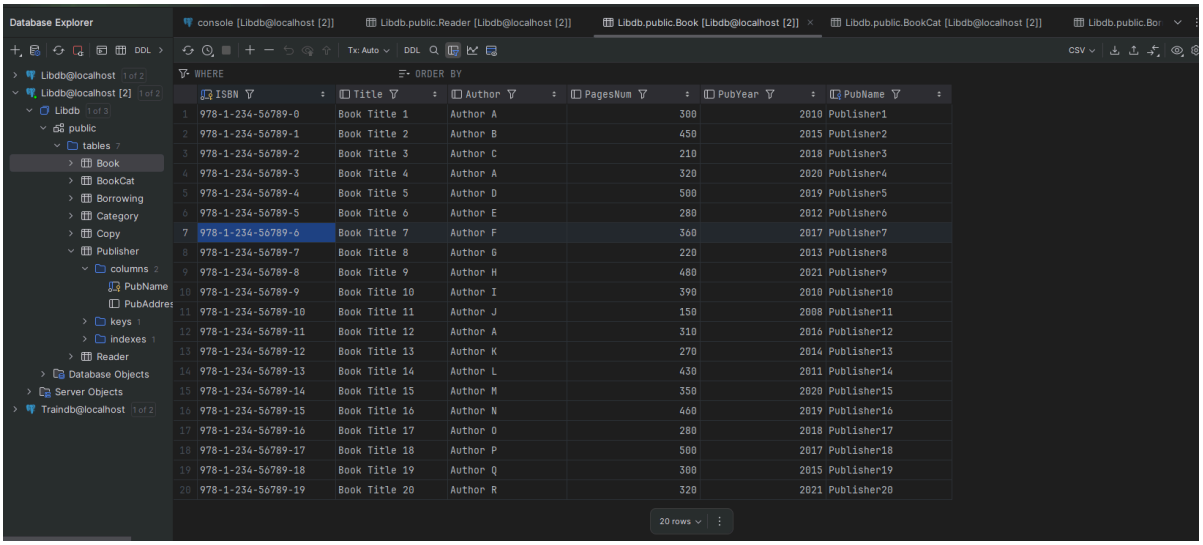
2. Удалить все книги, год публикации которых превышает 2000 год.

Думаю лучше взять 2010, чтобы не удалять все.

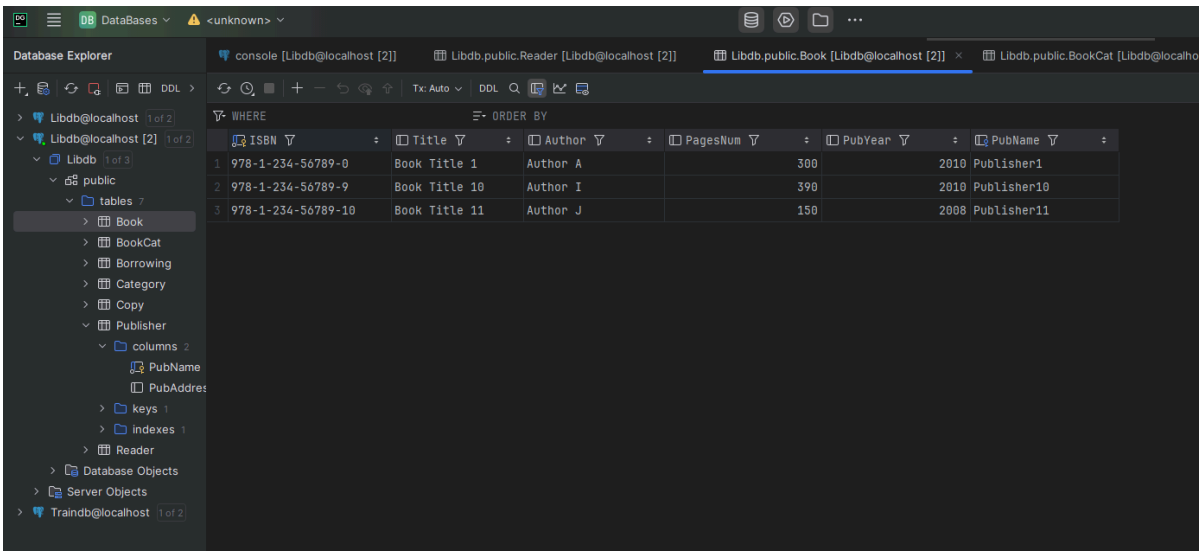
Запрос:

```
DELETE FROM "Borrowing" WHERE "ISBN" IN (SELECT "ISBN" FROM "Book" WHERE "PubYear" > 2010);

 DELETE FROM "Copy" WHERE "ISBN" IN (SELECT "ISBN" FROM "Book" WHERE "PubYear" > 2010);

DELETE FROM "BookCat"
WHERE "ISBN" IN (SELECT "ISBN" FROM "Book" WHERE "PubYear" > 2010);

DELETE FROM "Book"
WHERE "PubYear" > 2010;
```

До удаления:



После удаления:

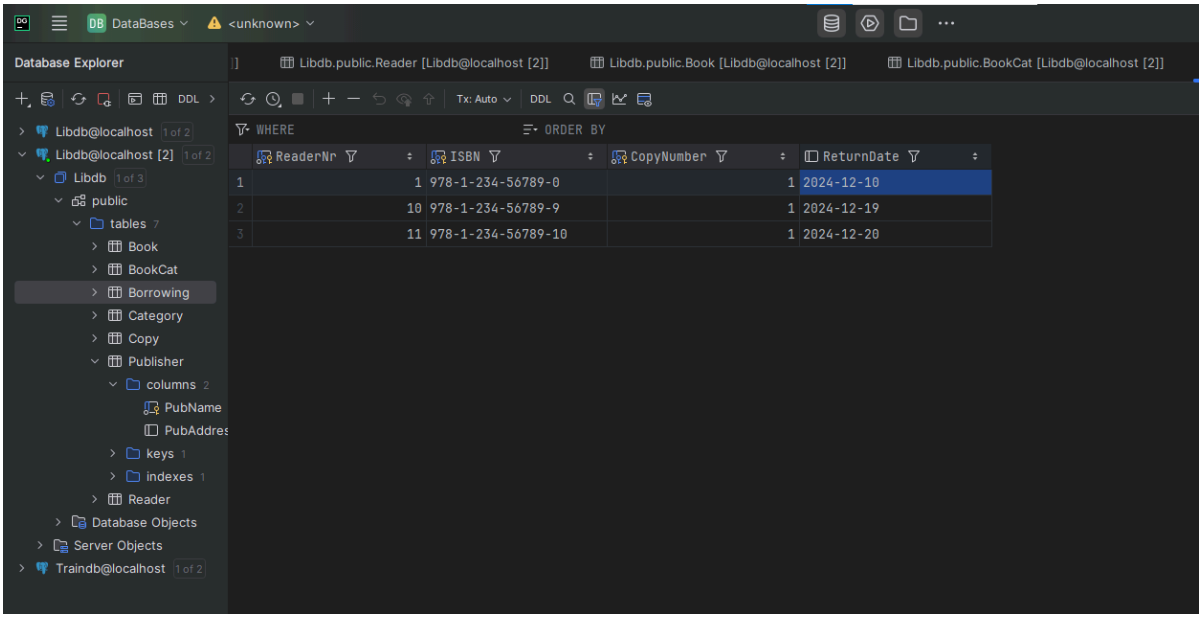


3. Измените дату возврата для всех книг категории "Базы данных", начиная с 01.01.2016, чтобы они были в заимствовании на 30 дней дольше.

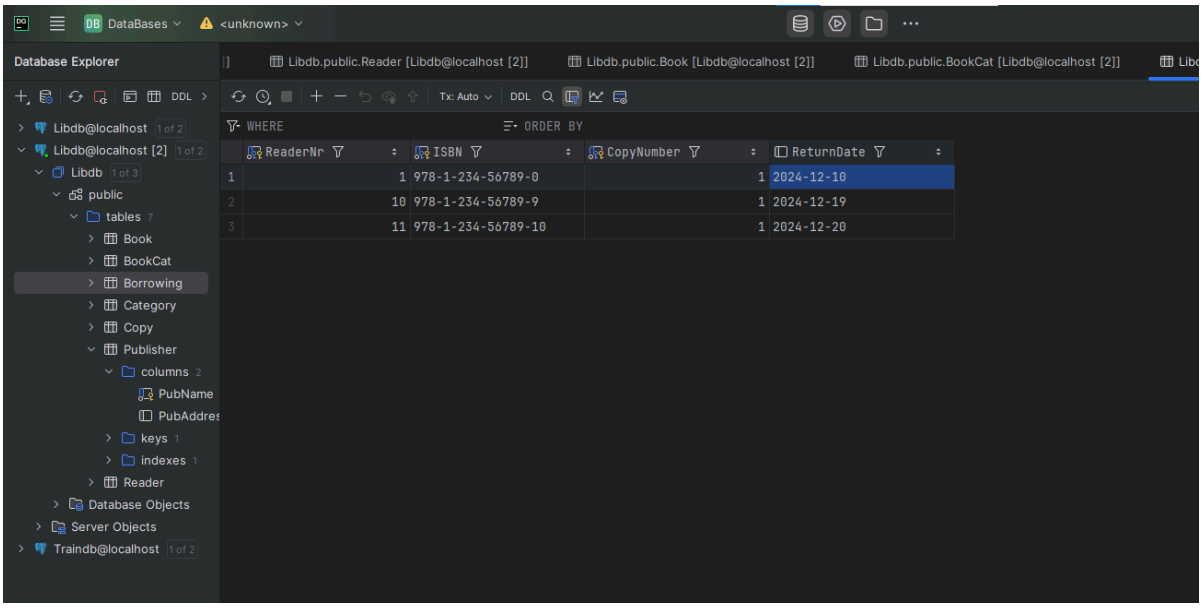
Запрос:

```
WITH BooksWithCategory AS (
    SELECT b."ISBN"
    FROM "Book" b
    JOIN "BookCat" bc ON b."ISBN" = bc."ISBN"
    WHERE bc."CategoryName" = 'Fiction'
)
UPDATE "Borrowing" b
SET "ReturnDate" = "ReturnDate" + INTERVAL '30 days'
WHERE b."ISBN" IN (SELECT "ISBN" FROM BooksWithCategory)
  AND b."ReturnDate" >= '2016-01-01';
```

До продления:

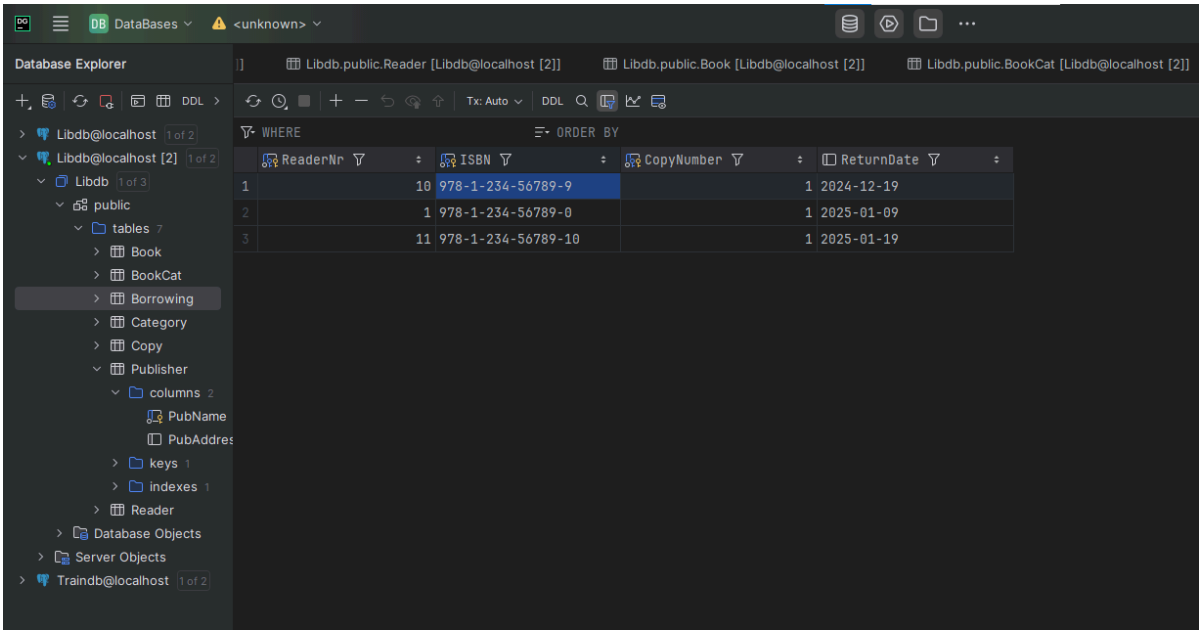


После продления:



Категории “Базы данных” нет, предлагаю запустить для “Fiction”:

После продления:



Задача 3

1.

```
SELECT s.Name, s.MatrNr FROM Student s
WHERE NOT EXISTS (
SELECT * FROM Check c WHERE c.MatrNr = s.MatrNr AND c.Note >= 4.0 ) ;
```

Объяснение:

Запрос возвращает имена и номера зачеток (MatNr)всех студентов, которые не сдали ни одной лекции на оценку 4.0 и выше. Это студенты, которые либо не сдали экзамен вообще, либо получили оценки ниже 4.0 по всем предметам.

2.

```
( SELECT p.ProfNr, p.Name, sum(lec.Credit)
FROM Professor p, Lecture lec
WHERE p.ProfNr = lec.ProfNr
GROUP BY p.ProfNr, p.Name)
UNION
( SELECT p.ProfNr, p.Name, 0
FROM Professor p
WHERE NOT EXISTS (
SELECT * FROM Lecture lec WHERE lec.ProfNr = p.ProfNr ));
```

Объяснение:

Этот запрос возвращает список профессоров с количеством кредитов, которые они преподавали:

- **Профессоры, которые ведут лекции, получают общую сумму кредитов.**
- **Профессора, которые не ведут лекций, получают 0 в поле с суммой кредитов.**

3.

```
SELECT s.Name, p.Note
FROM Student s, Lecture lec, Check c
WHERE s.MatrNr = c.MatrNr AND lec.LectNr = c.LectNr AND c.Note >= 4
AND c.Note >= ALL (
SELECT c1.Note FROM Check c1 WHERE c1.MatrNr = c.MatrNr )
```

Объяснение:

Запрос возвращает имена студентов и их оценки по конкретным экзаменам, где их оценка была наивысшей среди всех их оценок, и при этом эта оценка была 4 или выше.