

федеральное государственное автономное образовательное учреждение высшего
образования
«Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

Лабораторная работа №2
«Запросы на выборку и модификацию
данных, представления и индексы в
PostgreSQL» по дисциплине
«Проектирование и
реализация баз данных»

Выполнила:
студентка 2 курса ФИКТ
группы K3241
Барышева З.А.

Проверила:
Говорова М.В.

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, pgadmin 4.

Практическое задание:

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Индивидуальное задание. Вариант 5. БД «Издательство компьютерной литературы»

Описание предметной области: Издательство занимается выпуском литературы по различным областям информатики. Покупатели книг приобретают книги на базе издательства. Когда на базе заканчиваются книги, издается дополнительный тираж. В каждом заказе заказчик может заказать разную литературу.

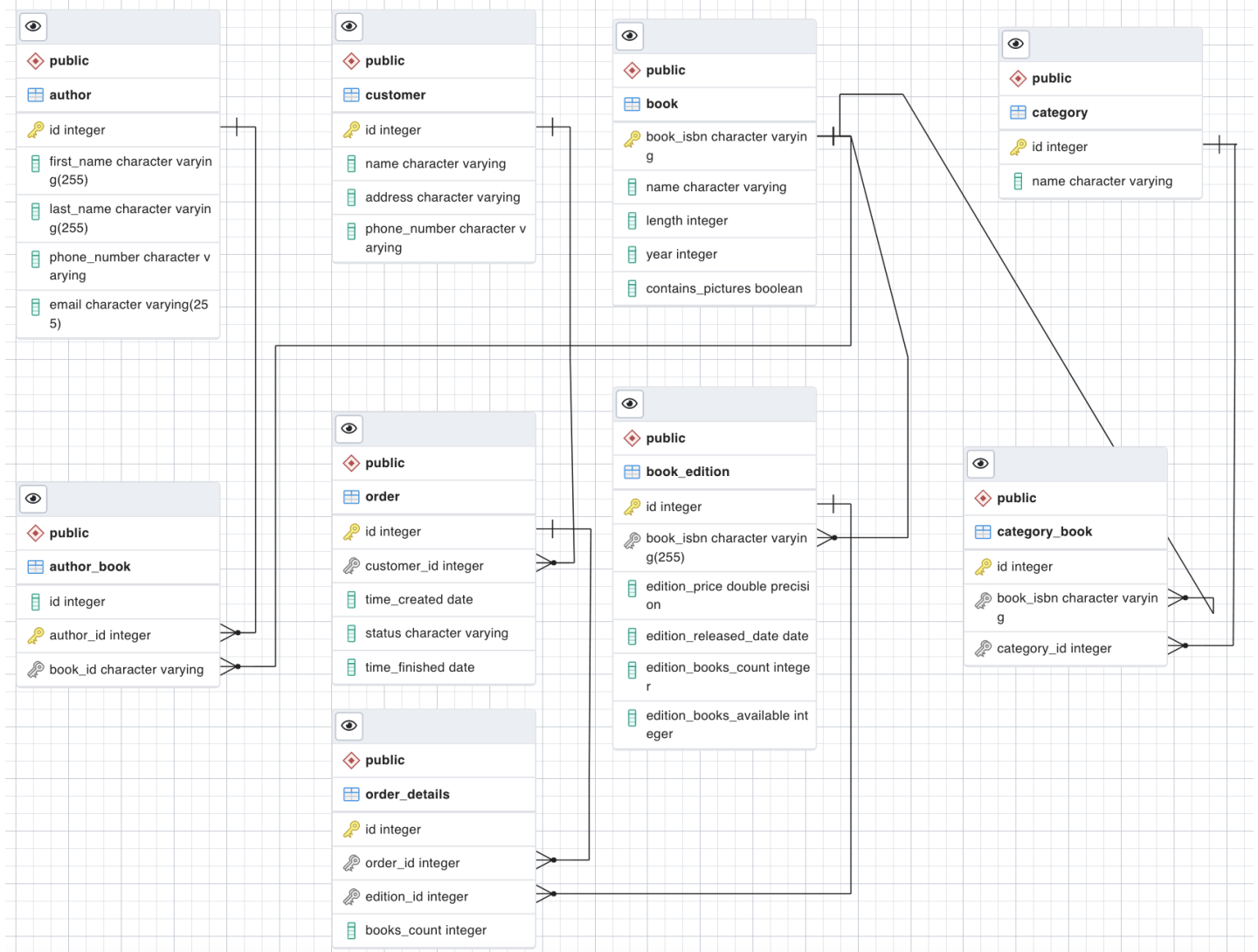
БД должна содержать следующий минимальный набор сведений: Фамилия автора. Имя автора. Отчество автора. Код автора. E-mail автора. Код ISBN. Название книги. Количество страниц. Наличие иллюстраций. Код категории книги. Категория книги. Количество страниц. Год начала издания. Розничная цена книги. Тираж. Дата тиража. Количество экземпляров на базе издательства. Код заказчика. Фамилия заказчика. Имя заказчика. Отчество заказчика. Адрес заказчика. Телефон заказчика. Код заказа. Дата заказа. Срок заказа. Количество экземпляров книги в заказе. Статус заказа.

1. БД «Издательство компьютерной литературы»

2. Сущности:

- Писатель (Код автора (PK), ФИО автора, Электронная почта автора)
- Книга (Код ISBN (PK), название книги, количество страниц, наличие иллюстраций, год начала издания)
- Категория (Код категории книги (PK), название категории)
- Издание (Номер издания (PK), дата тиража, Код ISBN, тираж, кол-во экземпляров на базе издательства, код заказа, розничная цена книги)
- Заказчик (Код заказчика (PK), ФИО заказчика, адрес заказчика, телефон заказчика)
- Заказ (Код заказа (PK), код заказчика, дата заказа, срок заказа, статус заказа)
- Состав заказа (Номер состава заказа (PK), номер заказа, номер издания, кол-во экземпляров книги в заказе)

3. ERD-диаграмма



4. Запросы:

- Список книг, изданных в текущем году и относящихся к категории “художественная литература”.

Query

Query History

```

1 SELECT * FROM book WHERE book_isbn IN (
2     SELECT category_book.book_isbn FROM category, category_book
3     WHERE category.id = category_book.category_id
4     AND category.name = 'Художественная литература'
5 )
  
```

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	book_isbn [PK] character varying	name character varying	length integer	year integer	contains_pictures boolean
1	1	1984	585	1949	true

- Список покупателей, заказавших книг на сумму, превышающую среднюю сумму заказа.

Query

Query History

```

1 SELECT DISTINCT c.name
2 FROM "order"
3     LEFT JOIN order_details od on "order".id = od.order_id
4     LEFT JOIN customer c on c.id = "order".customer_id
5     LEFT JOIN book_edition be on be.id = od.edition_id
6 WHERE od.books_count * be.edition_price >
7     (
8         SELECT avg(avgprice)
9         FROM (SELECT order_details.count * book_edition.edition_price as avgprice
10              FROM order_details
11                  LEFT JOIN "order" on order_details.order_id = "order".id
12                  LEFT JOIN book_edition on order_details.edition_id = book_edition.id
13              GROUP BY book_edition.edition_price)
14     )

```

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	name character varying
1	Artem
2	Alex

- Список книг, которые не заказывались в течение последних двух кварталов.

Query

Query History

```

1 SELECT *
2 FROM book
3 WHERE book.book_isbn NOT IN
4     (SELECT be.book_isbn
5      FROM "order" o
6          LEFT JOIN order_details od on o.id = od.order_id
7          LEFT JOIN book_edition be on be.id = od.edition_id
8      WHERE o.time_created > (CURRENT_DATE - INTERVAL '6 months'));

```

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	book_isbn [PK] character varying	name character varying	length integer	year integer	contains_pictures boolean
1	3	A tour of C++	890	2013	true
2	5	Преступление и наказан...	500	1866	false

- Список авторов, не написавших ни одной книги, относящейся к категории “художественная литература”.

Query

Query History

1

SELECT author.first_name, author.last_name

2

FROM author

3

WHERE id NOT IN (SELECT ab.author_id

4

FROM book

5

LEFT JOIN category_book cb on book.book_isbn = cb.book_isbn

6

LEFT JOIN category c on c.id = cb.category_id

7

LEFT JOIN author_book ab on book.book_isbn = ab.book_id

8

WHERE c.name = 'Художественная литература');

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	first_name character varying (255) 🔒	last_name character varying (255) 🔒
1	Владимир	Путин
2	Бьёрн	Страуструп
3	Линус	Торвальдс

- Список книг, в названиях которых содержится слово “Мысли” и которые присутствуют на базе в количестве, превышающем 50 экземпляров.

Query

Query History

1

2

3

4

5

6

SELECT

book.name,

book_edition.edition_books_available

FROM

book,

book_edition

WHERE

book.book_isbn = book_edition.book_isbn

AND

book.name LIKE '%Мысли%'

AND

book_edition.edition_books_available >= 50;

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	<div>name</div> <div>character varying</div> <div>🔒</div>	<div>edition_books_available</div> <div>integer</div> <div>🔒</div>
1	Мысли о России	100

- Список книг, не попавших ни в один из заказов в течение последнего года.

Query

Query History

1

SELECT name

2

FROM book

3

WHERE book_isbn NOT IN (SELECT distinct be.book_isbn

4

FROM order_details oc

5

LEFT JOIN book_edition be on be.id = oc.edition_id

6

LEFT JOIN "order" o on o.id = oc.order_id

7

WHERE o.time_created > CURRENT_DATE - INTERVAL '1 year')

8

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

name

character varying

🔒

1

A tour of C++

2

Преступление и наказание

- Представление, содержащее сведения о количестве заказанных экземпляров каждой книги, изданной в текущем году

Query Query History

```

1 CREATE OR REPLACE VIEW current_year_books AS
2 SELECT book.name, sum(od.books_count)
3 FROM book
4     LEFT JOIN book_edition be on book.book_isbn = be.book_isbn
5     LEFT JOIN order_details od on be.id = od.edition_id
6 WHERE book.year = date_part('year', CURRENT_DATE)
7 GROUP BY book.name, od.books_count;
8
9 SELECT * FROM current_year_books;
10

```

Data output Messages Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	name character varying	sum bigint
1	Ради удовольствия	2

- Представление количества заказов по покупателям за последний год

Query Query History

```

1 CREATE OR REPLACE VIEW top_customers AS
2 SELECT c.name, count(*) count_books
3 FROM customer c
4     LEFT JOIN "order" o on c.id = o.customer_id
5     LEFT JOIN order_details od on o.id = od.order_id
6 WHERE date_part('year', o.time_created) = date_part('year', CURRENT_DATE)
7 GROUP BY c.name
8 ORDER BY count_books DESC;
9
10 SELECT * FROM top_customers;

```

Data output Messages Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	name character varying	count_books bigint
1	Alex	6
2	Artem	3

5. DELETE, UPDATE, INSERT с подзапросами

Добавить новую книгу

```

INSERT INTO book(book_isbn, name, length, "year", contains_pictures)
VALUES ((SELECT max(book_isbn::int) + 1 FROM book), 'Три мушкетера', 440, 1923, false);

```

Удалить все книги Ф.М. Достоевского:

```
DELETE
FROM book
WHERE book_isbn IN (SELECT book_id
                    FROM author_book
                    JOIN author a on a.id = author_book.author_id
                    WHERE a.first_name = 'Федор' AND a.last_name = 'Достоевский')
```

(Работает при опции ON DELETE CASCADE у fk)

Отменить все заказы, для которых на складе отсутствуют книги

```
UPDATE "order"
SET status='отменен'
WHERE status = 'оплачен'
AND "order".id IN (SELECT "order".id
                  FROM "order"
                  JOIN order_details od on "order".id = od.order_id
                  LEFT JOIN book_edition be on be.id = od.edition_id
                  WHERE be.edition_books_available < 1)
```

6. Работа с индексами:

Время выполнения запроса без индексов:

Query Query History

```
1  SELECT book.name, book_edition.edition_books_available
2  FROM book,
3       book_edition
4  WHERE book.book_isbn = book_edition.book_isbn
5         AND book.name LIKE '%Мысли%'
6         AND book_edition.edition_books_available >= 50;
```

Data output Messages Explain × Notifications

Successfully run. Total query runtime: 160 msec.
1 rows affected.

Создадим индекс и запустим еще раз:

Query	Query History
1	CREATE INDEX book_name ON book(name);
2	
3	SELECT book.name, book_edition.edition_books_available
4	FROM book,
5	book_edition
6	WHERE book.book_isbn = book_edition.book_isbn
7	AND book.name LIKE '%Мысли%'
8	AND book_edition.edition_books_available >= 50;

Data output	Messages	Explain	×	Notifications
Successfully run. Total query runtime: 56 msec. 1 rows affected.				

Время исполнения 56 мс вместо 160 мс, заметное ускорение.

7. **Вывод:** В ходе выполнения работы с помощью SQL были получены, изменены, удалены и добавлены данные в таблицы, были созданы индексы. С индексами запросы работают намного быстрее