**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы данных»**

Тема: Реализация базы данных в СУБД PostgreSQL

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Смирнов Д.Ю. |
| Преподаватель |  | Заславский М.М. |

Санкт-Петербург

2023

**Цель работы**

Развернуть локально PostgreSQL, написать запросы для создания и заполнения таблиц, написать запросы к БД, отвечающие на вопросы задания.

**Задание**

Вариант 19

Пусть требуется создать программную систему, предназначенную для работников технического архива предприятия. Технический архив содержит стеллажи, полки и ячейки, в которых хранится документация. Ячейка архива может быть пустой или хранить все экземпляры одного документа. Каждый экземпляр документации имеет инвентарный номер и название. В базе данных должна храниться следующая информация о каждом документе архива: номер стеллажа, номер полки, номер ячейки, где хранится документ, название документа и название темы, к которой он относится, его инвентарный номер, количество экземпляров документа, содержащихся в ячейке, дата поступления документа в архив. Документ может быть востребован абонентом архива. Абонент характеризуется фамилией, именем, отчеством, номером и телефоном отдела, где он работает. Работники архива, выдавая документ, должны зафиксировать, когда и кому он был выдан. Архив может пополняться документами, как новыми, так и копиями уже имеющихся в архиве. Экземпляр документа может быть утрачен. Возможна закупка новых стеллажей и списание старых. Документ может поменять место хранения и инвентарный номер. Возможно и изменение сведений об абонентах. Абонент может поменять фамилию, перейти в другой отдел, уволится с предприятия. Возможно изменение номеров телефонов отделов. Работнику архива могут потребоваться следующие сведения:

* Название наиболее востребованного документа?
* Общее количество документов на заданную тему?
* Тема документа по заданному названию?
* Название документа, который имеется в архиве в максимальном количестве экземпляров?
* Фамилия, имя и отчество абонента, который брал указанный документ последним?
* Есть ли в архиве пустые стеллажи, полки, ячейки, и в каком количестве?
* Список документов, не востребованных в течение более, чем 5 лет?

**Выполнение работы**

Создана база данных под названием «Архив», рисунок 1.

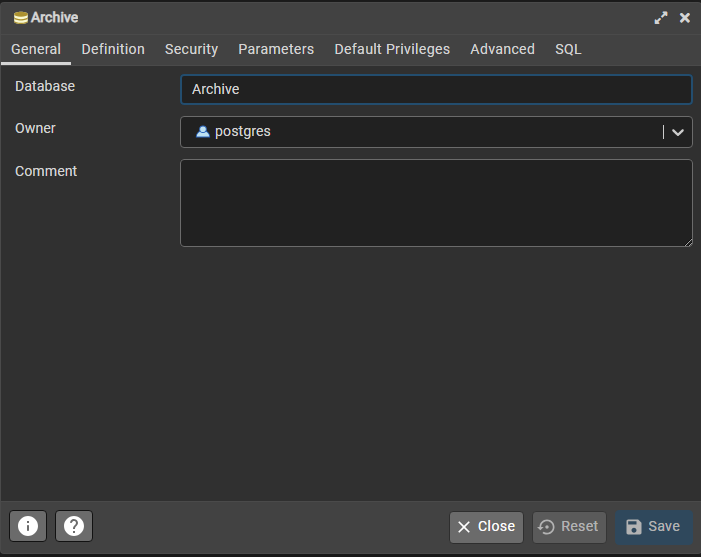


Рисунок 1 – Создание БД "Архив"

Созданы таблицы «*shelf*», «*rack*», «*cell*», «*instance*», «*department*», «*abonent*», «*issues*», «*issues\_archive*» в соответствии со структурой БД, а также тип перечисление «*instance\_status*». Запросы на создание представлены на рисунках 2-3.

Исходный код всех запросов представлен в приложении Б.

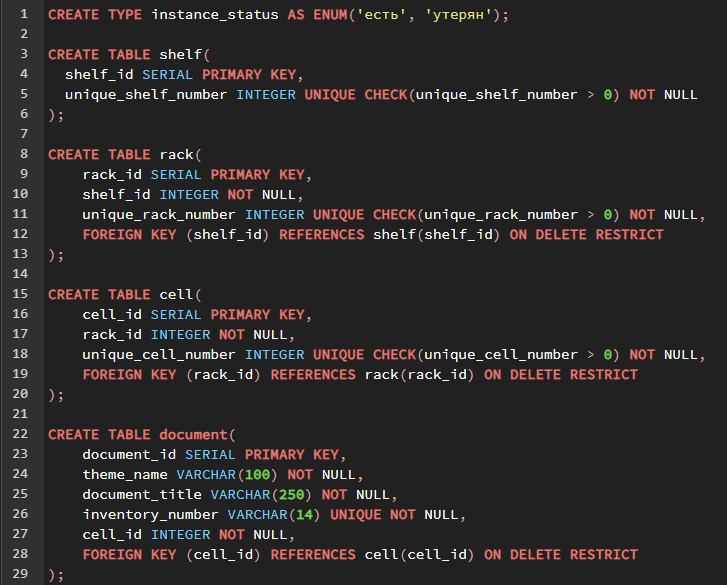


Рисунок – запросы на создание, часть 1

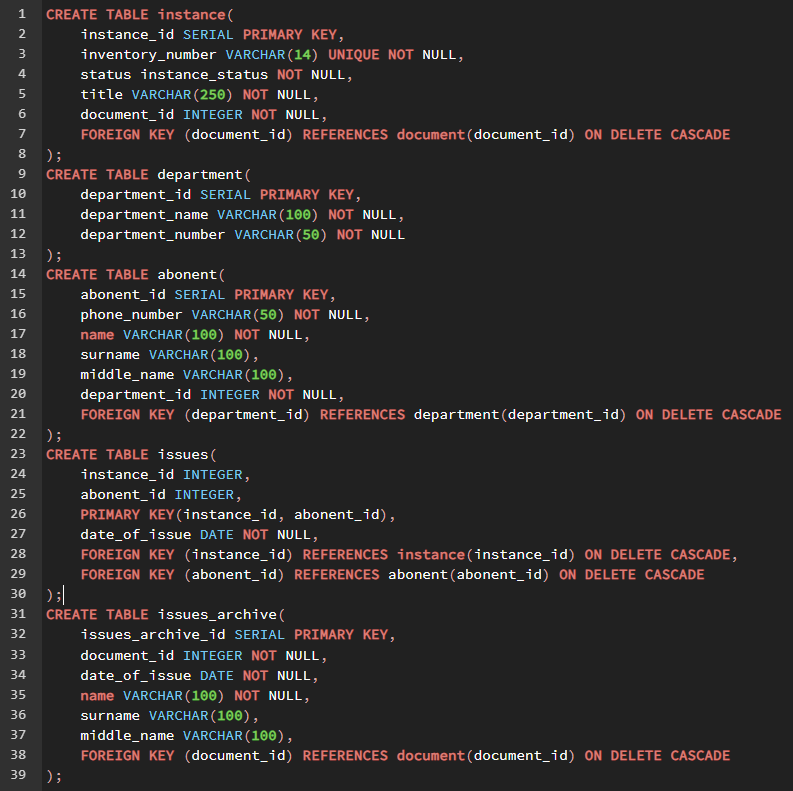


Рисунок – запросы на создание, часть 2

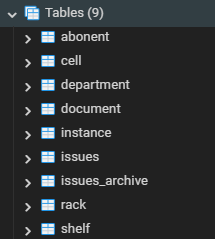


Рисунок 4 – Полученные таблицы

Запросы на заполнение таблиц представлены на рисунках 5-6.

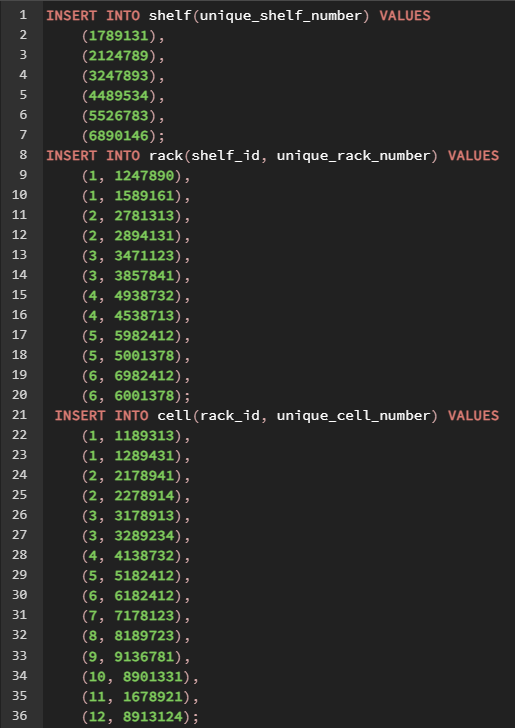


Рисунок – заполнение таблиц, часть 1

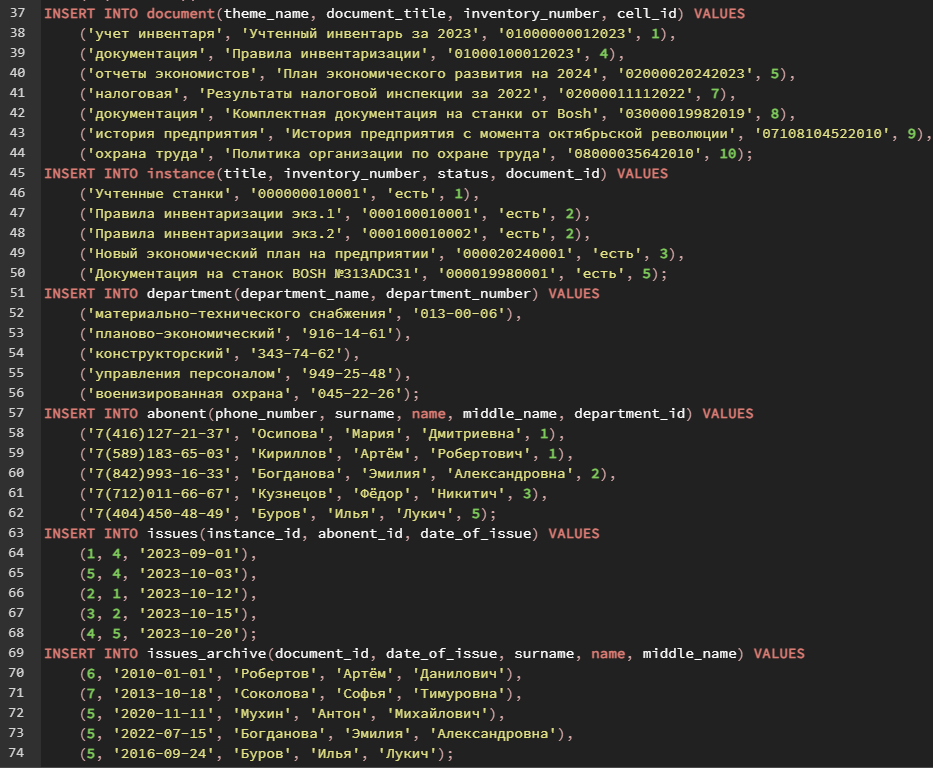


Рисунок – заполнение таблиц, часть 2

Написаны запросы для работника архива:

1. Название наиболее востребованного документа, представлен на рисунке 7.

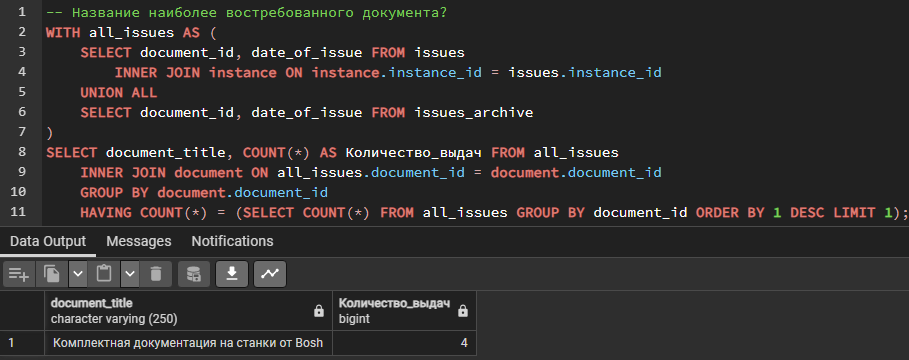


Рисунок – наиболее востребованный документ

1. Количество документов на заданную тему, представлен на рисунке 8.

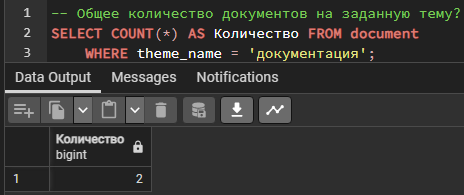


Рисунок – количество документов на заданную тему

1. Тема документа по указанному названию, представлен на рисунке 9.

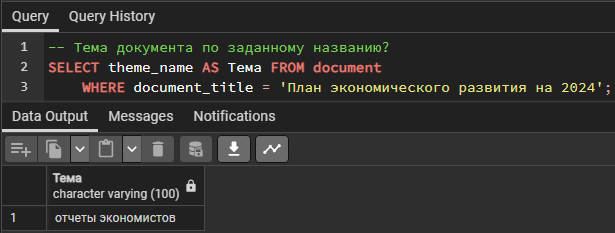


Рисунок - тема документа по названию

1. Название документа, который имеется в архиве в максимальном количестве экземпляров, представлен на рисунке 10.

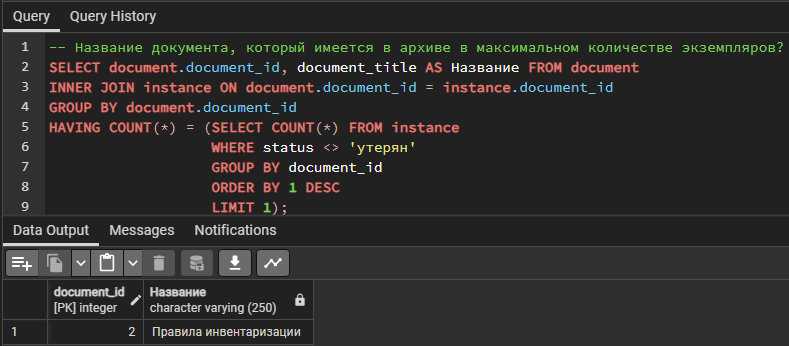


Рисунок – название документа, у которого максимальное количество экземпляров

1. ФИО абонента, который брал указанный документ последним, представлен на рисунке 11.

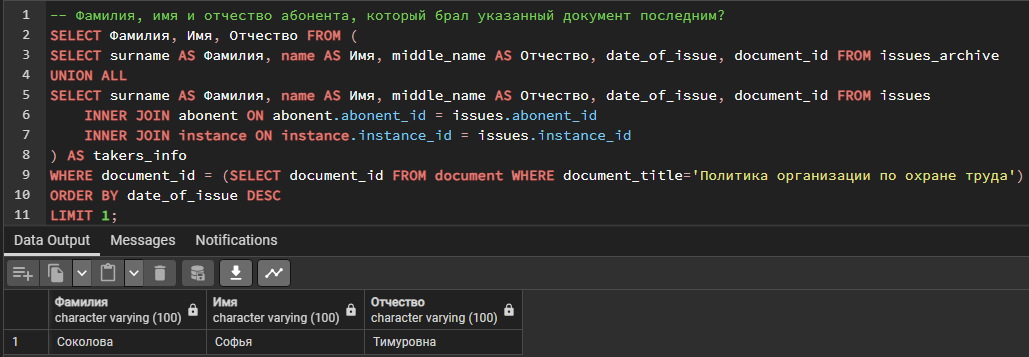


Рисунок – ФИО абонента, бравшего указанный документ последним

1. Есть ли в архиве пустые ячейки, полки, стеллажи и в каком количестве, представлен на рисунке 12.

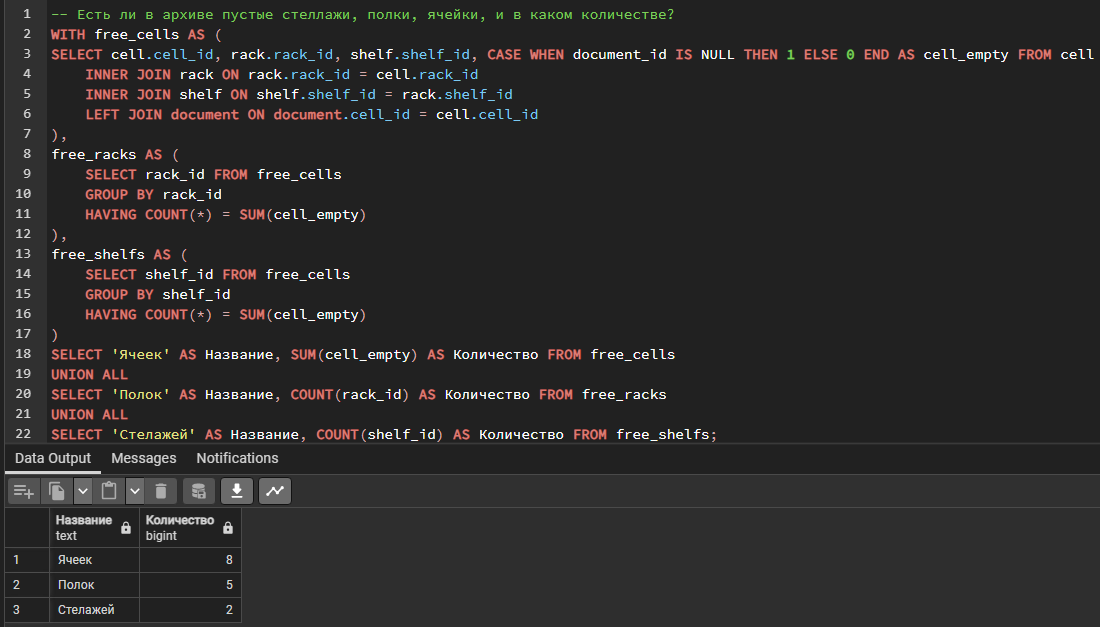


Рисунок – запрос на пустые ячейки, полки, стеллажи

1. Список документов, не востребованных более чем 5 лет, представлен на рисунке 13.

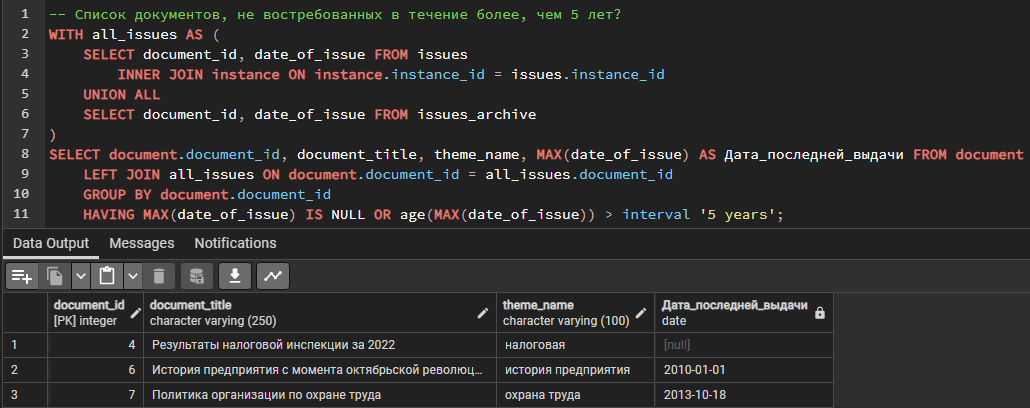


Рисунок – документы, не востребованные более чем 5 лет

**Вывод.**

В ходе выполнения работы был развернут локально PostgreSQL Server, написаны запросы для создания и заполнения таблиц в соответствии со структурой БД, получен опыт написания различных запросов к БД.

**ПРИЛОЖЕНИЕ А**

**ССЫЛКИ**

Pull Request: <https://github.com/moevm/sql-2023-1303/pull/33>

DB Fiddle: <https://www.db-fiddle.com/f/4aeZHHq6yKf4YRBujHyDjB/73>

**ПРИЛОЖЕНИЕ Б**

**ИСХОДНЫЙ КОД**

Файл lab2.sql:

CREATE TYPE instance\_status AS ENUM('есть', 'утерян');

CREATE TABLE shelf(

shelf\_id SERIAL PRIMARY KEY,

unique\_shelf\_number INTEGER UNIQUE CHECK(unique\_shelf\_number > 0) NOT NULL

);

CREATE TABLE rack(

rack\_id SERIAL PRIMARY KEY,

shelf\_id INTEGER NOT NULL,

unique\_rack\_number INTEGER UNIQUE CHECK(unique\_rack\_number > 0) NOT NULL,

FOREIGN KEY (shelf\_id) REFERENCES shelf(shelf\_id) ON DELETE RESTRICT

);

CREATE TABLE cell(

cell\_id SERIAL PRIMARY KEY,

rack\_id INTEGER NOT NULL,

unique\_cell\_number INTEGER UNIQUE CHECK(unique\_cell\_number > 0) NOT NULL,

FOREIGN KEY (rack\_id) REFERENCES rack(rack\_id) ON DELETE RESTRICT

);

CREATE TABLE document(

document\_id SERIAL PRIMARY KEY,

theme\_name VARCHAR(100) NOT NULL,

document\_title VARCHAR(250) NOT NULL,

inventory\_number VARCHAR(14) UNIQUE NOT NULL,

cell\_id INTEGER NOT NULL,

FOREIGN KEY (cell\_id) REFERENCES cell(cell\_id) ON DELETE RESTRICT

);

CREATE TABLE instance(

instance\_id SERIAL PRIMARY KEY,

inventory\_number VARCHAR(14) UNIQUE NOT NULL,

status instance\_status NOT NULL,

title VARCHAR(250) NOT NULL,

document\_id INTEGER NOT NULL,

FOREIGN KEY (document\_id) REFERENCES document(document\_id) ON DELETE CASCADE

);

CREATE TABLE department(

department\_id SERIAL PRIMARY KEY,

department\_name VARCHAR(100) NOT NULL,

department\_number VARCHAR(50) NOT NULL

);

CREATE TABLE abonent(

abonent\_id SERIAL PRIMARY KEY,

phone\_number VARCHAR(50) NOT NULL,

name VARCHAR(100) NOT NULL,

surname VARCHAR(100),

middle\_name VARCHAR(100),

department\_id INTEGER NOT NULL,

FOREIGN KEY (department\_id) REFERENCES department(department\_id) ON DELETE CASCADE

);

CREATE TABLE issues(

instance\_id INTEGER,

abonent\_id INTEGER,

PRIMARY KEY(instance\_id, abonent\_id),

date\_of\_issue DATE NOT NULL,

FOREIGN KEY (instance\_id) REFERENCES instance(instance\_id) ON DELETE CASCADE,

FOREIGN KEY (abonent\_id) REFERENCES abonent(abonent\_id) ON DELETE CASCADE

);

CREATE TABLE issues\_archive(

issues\_archive\_id SERIAL PRIMARY KEY,

document\_id INTEGER NOT NULL,

date\_of\_issue DATE NOT NULL,

name VARCHAR(100) NOT NULL,

surname VARCHAR(100),

middle\_name VARCHAR(100),

FOREIGN KEY (document\_id) REFERENCES document(document\_id) ON DELETE CASCADE

);

INSERT INTO shelf(unique\_shelf\_number) VALUES

(1789131),

(2124789),

(3247893),

(4489534),

(5526783),

(6890146);

INSERT INTO rack(shelf\_id, unique\_rack\_number) VALUES

(1, 1247890),

(1, 1589161),

(2, 2781313),

(2, 2894131),

(3, 3471123),

(3, 3857841),

(4, 4938732),

(4, 4538713),

(5, 5982412),

(5, 5001378),

(6, 6982412),

(6, 6001378);

INSERT INTO cell(rack\_id, unique\_cell\_number) VALUES

(1, 1189313),

(1, 1289431),

(2, 2178941),

(2, 2278914),

(3, 3178913),

(3, 3289234),

(4, 4138732),

(5, 5182412),

(6, 6182412),

(7, 7178123),

(8, 8189723),

(9, 9136781),

(10, 8901331),

(11, 1678921),

(12, 8913124);

INSERT INTO document(theme\_name, document\_title, inventory\_number, cell\_id) VALUES

('учет инвентаря', 'Учтенный инвентарь за 2023', '01000000012023', 1),

('документация', 'Правила инвентаризации', '01000100012023', 4),

('отчеты экономистов', 'План экономического развития на 2024', '02000020242023', 5),

('налоговая', 'Результаты налоговой инспекции за 2022', '02000011112022', 7),

('документация', 'Комплектная документация на станки от Bosh', '03000019982019', 8),

('история предприятия', 'История предприятия с момента октябрьской революции', '07108104522010', 9),

('охрана труда', 'Политика организации по охране труда', '08000035642010', 10);

INSERT INTO instance(title, inventory\_number, status, document\_id) VALUES

('Учтенные станки', '000000010001', 'есть', 1),

('Правила инвентаризации экз.1', '000100010001', 'есть', 2),

('Правила инвентаризации экз.2', '000100010002', 'есть', 2),

('Новый экономический план на предприятии', '000020240001', 'есть', 3),

('Документация на станок BOSH №313ADC31', '000019980001', 'есть', 5);

INSERT INTO department(department\_name, department\_number) VALUES

('материально-технического снабжения', '013-00-06'),

('планово-экономический', '916-14-61'),

('конструкторский', '343-74-62'),

('управления персоналом', '949-25-48'),

('военизированная охрана', '045-22-26');

INSERT INTO abonent(phone\_number, surname, name, middle\_name, department\_id) VALUES

('7(416)127-21-37', 'Осипова', 'Мария', 'Дмитриевна', 1),

('7(589)183-65-03', 'Кириллов', 'Артём', 'Робертович', 1),

('7(842)993-16-33', 'Богданова', 'Эмилия', 'Александровна', 2),

('7(712)011-66-67', 'Кузнецов', 'Фёдор', 'Никитич', 3),

('7(404)450-48-49', 'Буров', 'Илья', 'Лукич', 5);

INSERT INTO issues(instance\_id, abonent\_id, date\_of\_issue) VALUES

(1, 4, '2023-09-01'),

(5, 4, '2023-10-03'),

(2, 1, '2023-10-12'),

(3, 2, '2023-10-15'),

(4, 5, '2023-10-20');

INSERT INTO issues\_archive(document\_id, date\_of\_issue, surname, name, middle\_name) VALUES

(6, '2010-01-01', 'Робертов', 'Артём', 'Данилович'),

(7, '2013-10-18', 'Соколова', 'Софья', 'Тимуровна'),

(5, '2020-11-11', 'Мухин', 'Антон', 'Михайлович'),

(5, '2022-07-15', 'Богданова', 'Эмилия', 'Александровна'),

(5, '2016-09-24', 'Буров', 'Илья', 'Лукич');

-- Название наиболее востребованного документа?

WITH all\_issues AS (

SELECT document\_id, date\_of\_issue FROM issues

INNER JOIN instance ON instance.instance\_id = issues.instance\_id

UNION ALL

SELECT document\_id, date\_of\_issue FROM issues\_archive

)

SELECT document\_title, COUNT(\*) AS Количество\_выдач FROM all\_issues

INNER JOIN document ON all\_issues.document\_id = document.document\_id

GROUP BY document.document\_id

HAVING COUNT(\*) = (SELECT COUNT(\*) FROM all\_issues GROUP BY document\_id ORDER BY 1 DESC LIMIT 1);

-- Общее количество документов на заданную тему?

SELECT COUNT(\*) AS Количество FROM document

WHERE theme\_name = 'документация';

-- Тема документа по заданному названию?

SELECT theme\_name AS Тема FROM document

WHERE document\_title = 'План экономического развития на 2024';

-- Название документа, который имеется в архиве в максимальном количестве экземпляров?

SELECT document.document\_id, document\_title AS Название FROM document

INNER JOIN instance ON document.document\_id = instance.document\_id

GROUP BY document.document\_id

HAVING COUNT(\*) = (SELECT COUNT(\*) FROM instance

WHERE status <> 'утерян'

GROUP BY document\_id

ORDER BY 1 DESC

LIMIT 1);

-- Фамилия, имя и отчество абонента, который брал указанный документ последним?

SELECT Фамилия, Имя, Отчество FROM (

SELECT surname AS Фамилия, name AS Имя, middle\_name AS Отчество, date\_of\_issue, document\_id FROM issues\_archive

UNION ALL

SELECT surname AS Фамилия, name AS Имя, middle\_name AS Отчество, date\_of\_issue, document\_id FROM issues

INNER JOIN abonent ON abonent.abonent\_id = issues.abonent\_id

INNER JOIN instance ON instance.instance\_id = issues.instance\_id

) AS takers\_info

WHERE document\_id = (SELECT document\_id FROM document WHERE document\_title='Политика организации по охране труда')

ORDER BY date\_of\_issue DESC

LIMIT 1;

-- Есть ли в архиве пустые стеллажи, полки, ячейки, и в каком количестве?

WITH free\_cells AS (

SELECT cell.cell\_id, rack.rack\_id, shelf.shelf\_id, CASE WHEN document\_id IS NULL THEN 1 ELSE 0 END AS cell\_empty FROM cell

INNER JOIN rack ON rack.rack\_id = cell.rack\_id

INNER JOIN shelf ON shelf.shelf\_id = rack.shelf\_id

LEFT JOIN document ON document.cell\_id = cell.cell\_id

),

free\_racks AS (

SELECT rack\_id FROM free\_cells

GROUP BY rack\_id

HAVING COUNT(\*) = SUM(cell\_empty)

),

free\_shelfs AS (

SELECT shelf\_id FROM free\_cells

GROUP BY shelf\_id

HAVING COUNT(\*) = SUM(cell\_empty)

)

SELECT 'Ячеек' AS Название, SUM(cell\_empty) AS Количество FROM free\_cells

UNION ALL

SELECT 'Полок' AS Название, COUNT(rack\_id) AS Количество FROM free\_racks

UNION ALL

SELECT 'Стелажей' AS Название, COUNT(shelf\_id) AS Количество FROM free\_shelfs;

-- Список документов, не востребованных в течение более, чем 5 лет?

WITH all\_issues AS (

SELECT document\_id, date\_of\_issue FROM issues

INNER JOIN instance ON instance.instance\_id = issues.instance\_id

UNION ALL

SELECT document\_id, date\_of\_issue FROM issues\_archive

)

SELECT document.document\_id, document\_title, theme\_name, MAX(date\_of\_issue) AS Дата\_последней\_выдачи FROM document

LEFT JOIN all\_issues ON document.document\_id = all\_issues.document\_id

GROUP BY document.document\_id

HAVING MAX(date\_of\_issue) IS NULL OR age(MAX(date\_of\_issue)) > interval '5 years';