# МИНОБРНАУКИ РОССИИ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

# «ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

**Кафедра МОЭВМ**

# ОТЧЕТ

**по лабораторной работе №2 по дисциплине «Базы данных»**

# Тема: Реализация базы данных в СУБД PostgreSQL

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1303 |  | Новак П.И. |
| 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 в задания.

# Текст задания

Вариант 16

Пусть требуется создать программную систему, предназначенную для врачей и работников регистратуры поликлиники. Такая система должна хранить сведения об участках, которые относятся к поликлинике, о расписании работы участковых врачей, информацию о врачах, а также карточки пациентов. Карточка имеет номер, в нее заносятся сведения о каждом посещении поликлиники пациентом: дата посещения, жалобы, предварительный диагноз, назначения, выписан или нет больничный лист, и, если выписан, то на какой срок, имя врача. В карточке на первой странице указаны также фамилия, имя, отчество пациента, его домашний адрес, пол и возраст, номер страхового полиса, дата заполнения карточки. В расписании работы врачей указывается, на каком участке работает врач, дни и часы приема, номер кабинета. Врач может обслуживать более одного участка. В случае увольнения врача его участок(участки)передается другим врачам. Данные о враче, которые хранятся в БД, - это фамилия, имя отчество, категория, стаж работы, дата рождения. В карточку больного при каждом его посещении поликлиники врачом заносится очередная запись. Работники регистратуры регистрируют пациента, заполняя первую страницу его карточки. Уволить врача имеет право только заведующий поликлиникой. Он удаляет из базы сведения о враче и передает его больных другому врачу. Работникам поликлиники могут потребоваться следующие сведения:

* Адрес данного больного, дата последнего посещения поликлиники и диагноз?
* Фамилия и инициалы лечащего врача данного больного?
* Номер кабинета, дни и часы приема данного врача?
* Больные, находящиеся в данный момент на лечении у данного врача(не истек срок больничного листа);
* Назначения врачей при указанном заболевании?
* Кто работает в данный момент в указанном кабинете?
* Сколько раз за прошедший месяц обращался в поликлинику указанный больной?
* Какое количество больных обслужил за прошедший месяц каждый из врачей поликлиники?

# Выполнение работы

Были созданы и заполнены таблицы в DataGrip:

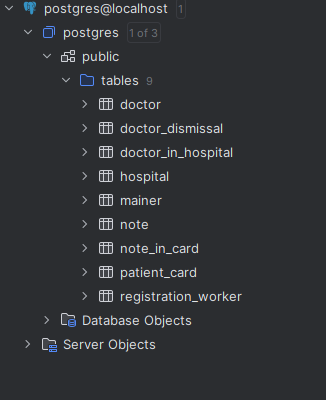
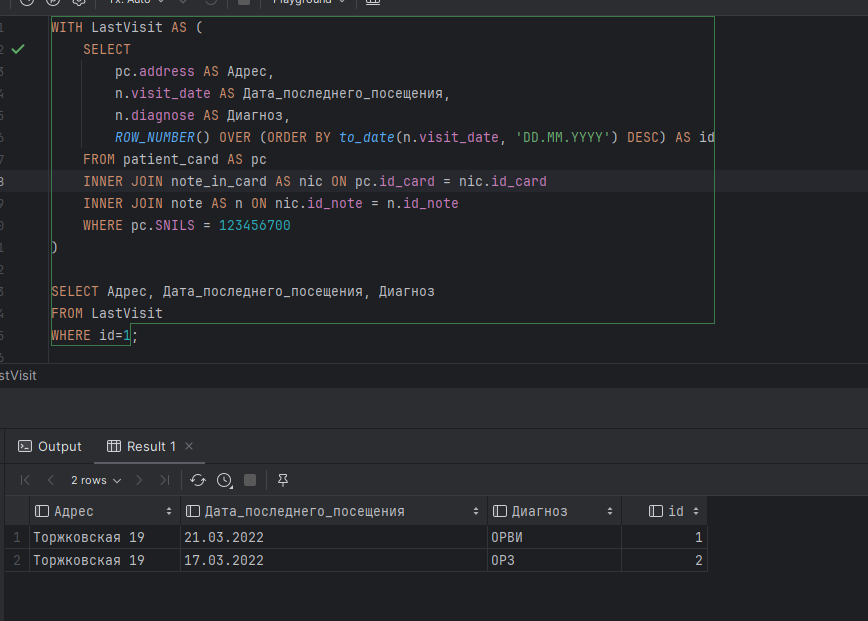
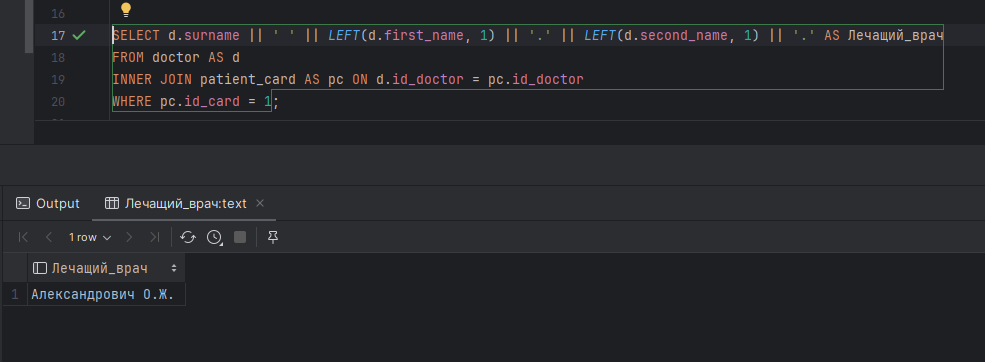


Рисунок 1 – Структура БД. Запросы к БД на получение данных:

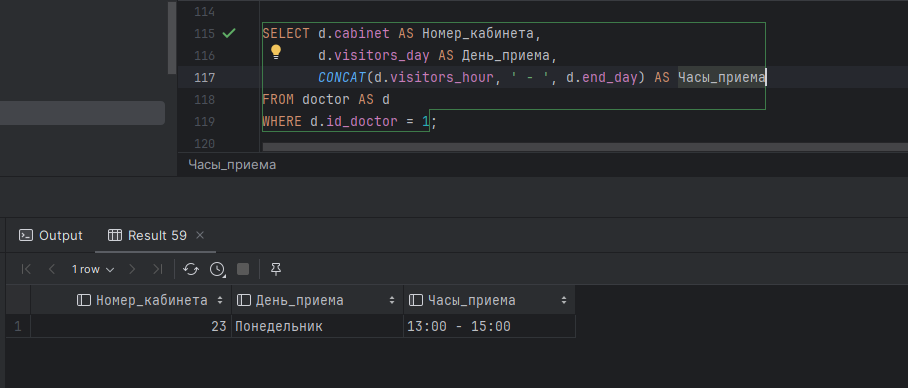
1. Получить адрес данного больного, дата последнего посещения поликлиники и диагноз.



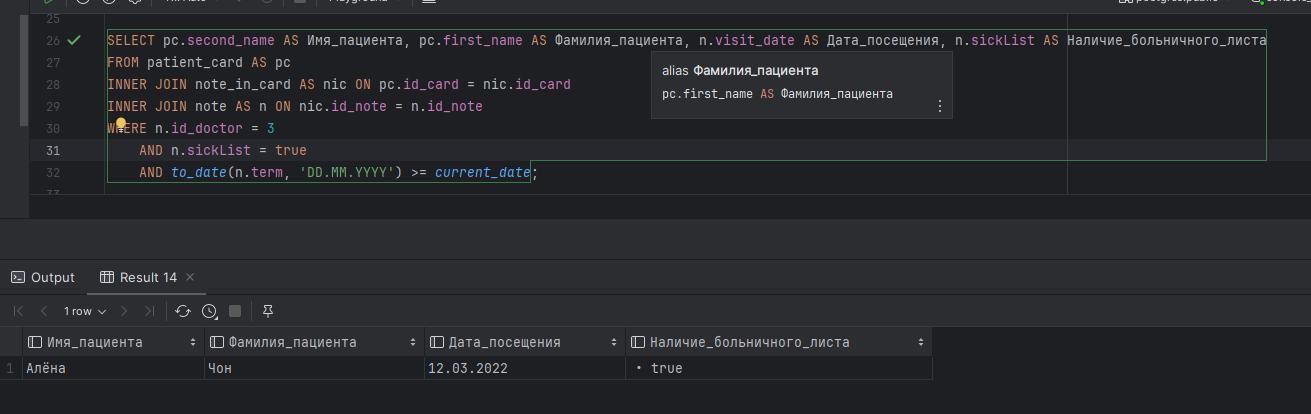
1. Получить фамилию и инициалы лечащего врача данного больного.



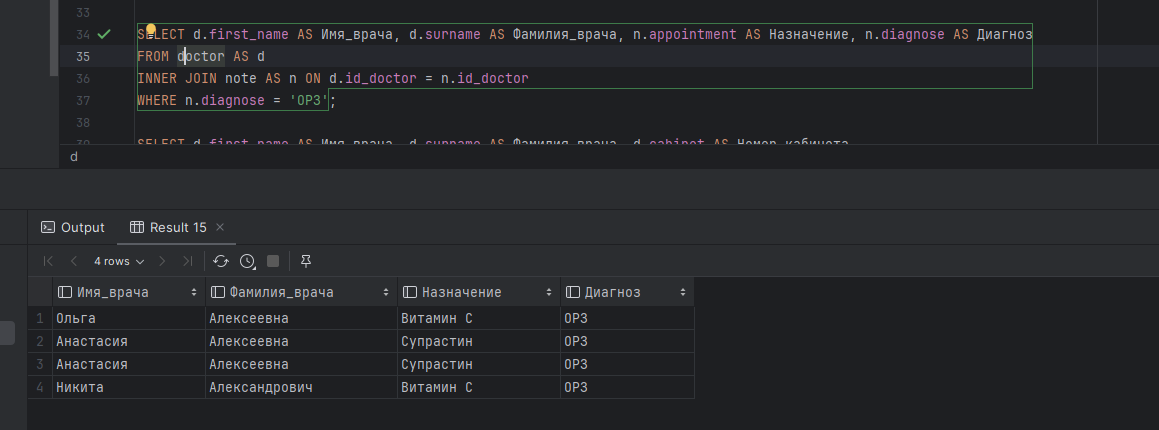
1. Получить номер кабинета, дни и часы приема данного врача.



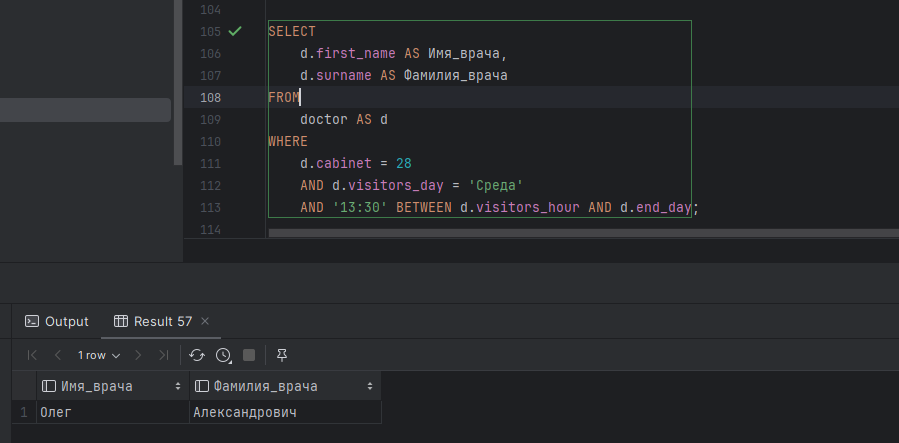
1. Узнать о больных, находящихся в данный момент на лечении у данного врача(не истек срок больничного листа).



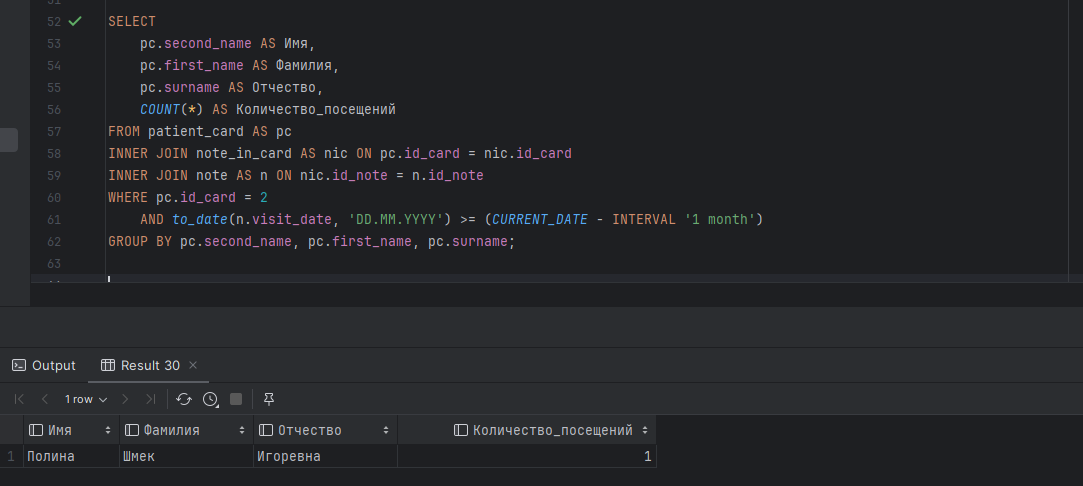
1. Получить информацию о назначениях врачей при указанном заболевании.



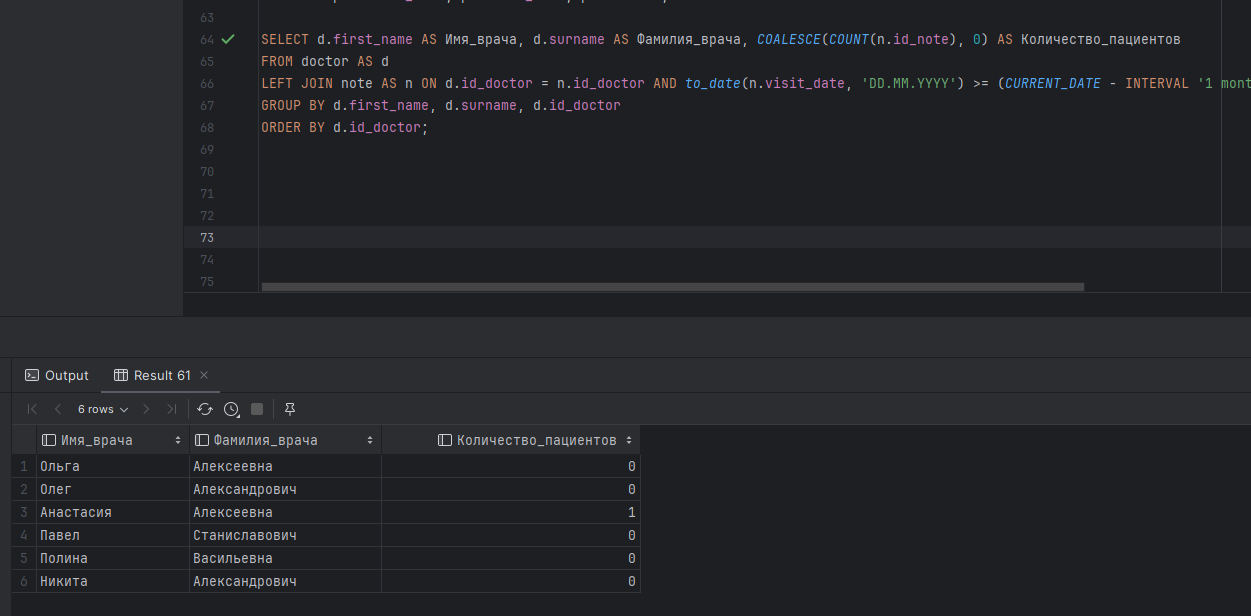
1. Получить информацию о том, кто работает в данный момент в указанном кабинете.



1. Получить информацию о том сколько раз за прошедший месяц обращался в поликлинику указанный больной?



1. Получить информацию о том какое количество больных обслужил за прошедший месяц каждый из врачей поликлиники.



# ПРИЛОЖЕНИЕ А

Pull request: https://github.com/moevm/sql-2023-1303/pull/31

DB-Fiddle: https://www.db-fiddle.com/f/eiTJDXFwsV2aavgCSxzZ8i/0

# ПРИЛОЖЕНИЕ Б

Файл lab2.sql:

CREATE TABLE doctor

(

id\_doctor INTEGER PRIMARY KEY,

first\_name TEXT not null,

second\_name TEXT not null,

surname TEXT not null,

category TEXT not null,

experience INTEGER,

date\_of\_birth DATE,

visitors\_day TEXT not null,

visitors\_hour TEXT not null,

cabinet INTEGER,

end\_day TEXT not null

);

INSERT INTO doctor

(id\_doctor, first\_name, second\_name, surname, category, experience, date\_of\_birth, visitors\_day, visitors\_hour, cabinet, end\_day)

VALUES (1, 'Ольга', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Понедельник', '13:00', 23, '15:00'),

(2, 'Алина', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Вторник', '14:00', 24, '16:00'),

(3, 'Олекса', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Понедельник', '13:00', 25, '15:00'),

(4, 'Алевтина', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Среда', '13:00', 26, '19:00'),

(5, 'Анастасия', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Пятница', '15:00', 27, '20:00'),

(6, 'Елена', 'Филевская', 'Алексеевна', 'Акушер', 2, '2000-03-12', 'Понедельник', '13:00', 28, '19:00');

CREATE TABLE hospital

(

hospital\_number INTEGER PRIMARY KEY

);

INSERT INTO hospital

(hospital\_number)

VALUES (1),

(2),

(3),

(4),

(5),

(6);

CREATE TABLE mainer

(

id\_mainer INTEGER PRIMARY KEY

);

INSERT INTO mainer

(id\_mainer)

VALUES (1),

(2),

(3),

(4),

(5),

(6);

CREATE TABLE doctor\_in\_hospital

(

hospital\_number INTEGER,

id\_doctor INTEGER,

PRIMARY KEY (hospital\_number, id\_doctor),

CONSTRAINT composite\_foreign\_key

FOREIGN KEY (hospital\_number) REFERENCES hospital (hospital\_number),

FOREIGN KEY (id\_doctor) REFERENCES doctor (id\_doctor)

);

INSERT INTO doctor\_in\_hospital

(hospital\_number, id\_doctor)

VALUES (1, 2),

(2, 3),

(3, 5),

(4, 4),

(5, 6),

(6, 1);

CREATE TABLE doctor\_dismissal

(

id\_mainer INTEGER,

id\_doctor INTEGER,

PRIMARY KEY (id\_mainer, id\_doctor),

CONSTRAINT composite\_foreign\_key1

FOREIGN KEY (id\_mainer) REFERENCES mainer (id\_mainer),

FOREIGN KEY (id\_doctor) REFERENCES doctor (id\_doctor)

);

INSERT INTO doctor\_dismissal

(id\_mainer, id\_doctor)

VALUES (1, 3),

(2, 4),

(4, 1),

(3, 6),

(6, 2);

CREATE TABLE patient\_card

(

id\_card INTEGER PRIMARY KEY,

SNILS INTEGER,

first\_name TEXT not null,

second\_name TEXT not null,

surname TEXT not null,

address TEXT not null,

sex TEXT not null,

age INTEGER,

id\_mainer SMALLINT not null REFERENCES mainer (id\_mainer) ON DELETE CASCADE,

id\_doctor SMALLINT not null REFERENCES doctor (id\_doctor) ON DELETE CASCADE

);

INSERT INTO patient\_card

(id\_card, SNILS, first\_name, second\_name, surname, address, sex, age, id\_mainer, id\_doctor)

VALUES (1, 123454789, 'Авьгинья','Александра', 'Александровна', 'Торжковская 15', 'Женский', 28, 1, 2),

(2, 123356789, 'Шмек','Полина', 'Игоревна', 'Торжковская 16', 'Женский', 35, 2, 4),

(3, 123459789, 'Чон','Алёна', 'Алексеевна', 'Торжковская 17', 'Женский', 48, 3, 1),

(4, 123456779, 'Хачмеле','Олег', 'Романович', 'Торжковская 18', 'Мужской', 18, 4, 5),

(5, 123456700, 'Рожовская','Роман', 'Олегович', 'Торжковская 19', 'Мужской', 19, 5, 6),

(6, 123406789, 'Дубен','Егор', 'Никитич', 'Торжковская 20', 'Мужской', 25, 6, 3);

CREATE TABLE note

(

id\_note INTEGER PRIMARY KEY,

visit\_date TEXT not null,

complaint TEXT not null,

diagnose TEXT not null,

appointment TEXT not null,

sick\_list BOOLEAN,

term TEXT not null,

completion\_date TEXT not null,

id\_doctor SMALLINT not null REFERENCES doctor (id\_doctor) ON DELETE CASCADE

);

INSERT INTO note

(id\_note, visit\_date, complaint, diagnose, appointment, sick\_list, term, completion\_date, id\_doctor)

VALUES (1, '12.03.2022', 'Плохое самочувствие', 'ОРЗ', 'Супрастин', true, '12.03.2024', '12.03.2022', 3),

(2, '15.03.2022', 'Отсуствие запаха', 'ОРВИ', 'Флюрография', true, '26.12.2024', '15.03.2022', 5),

(3, '21.03.2022', 'Плохое самочувствие', 'ОРВИ', 'Иммунал', true, '30.03.2024', '14.03.2022', 4),

(4, '13.03.2022', 'Плохое самочувствие', 'ОРЗ', 'Витамин С', true, '20.12.2024', '13.03.2022', 1),

(5, '16.03.2022', 'Онемение ноги', 'Анемия', 'Пройти обследование', true, '30.03.2024', '16.03.2022', 2),

(6, '17.03.2022', 'Плохое самочувствие', 'ОРЗ', 'Витамин С', true, '25.03.2024', '17.03.2022', 6);

CREATE TABLE note\_in\_card

(

id\_card INTEGER not null REFERENCES patient\_card (id\_card),

id\_note INTEGER PRIMARY KEY

);

INSERT INTO note\_in\_card

(id\_card, id\_note)

VALUES (1, 2),

(5, 3),

(3, 1),

(2, 5),

(5, 6),

(6, 4);

CREATE TABLE registration\_worker

(

id\_reg\_worker INTEGER PRIMARY KEY,

id\_doctor SMALLINT not null REFERENCES doctor (id\_doctor) ON DELETE CASCADE

);

INSERT INTO registration\_worker

(id\_reg\_worker, id\_doctor)

VALUES (1, 2),

(2, 3),

(3, 1),

(4, 5),

(5, 6),

(6, 4);

-- 1 вопрос

WITH LastVisit AS (

SELECT

pc.address AS Адрес,

n.visit\_date AS Дата\_последнего\_посещения,

n.diagnose AS Диагноз

FROM patient\_card AS pc

INNER JOIN note\_in\_card AS nic ON pc.id\_card = nic.id\_card

INNER JOIN note AS n ON nic.id\_note = n.id\_note

WHERE pc.SNILS = 123454789

ORDER BY to\_date(n.visit\_date, 'DD.MM.YYYY') DESC

)

SELECT Адрес, Дата\_последнего\_посещения, Диагноз

FROM LastVisit

LIMIT 1;

-- 2 вопрос

SELECT d.surname || ' ' || LEFT(d.first\_name, 1) || '.' || LEFT(d.second\_name, 1) || '.' AS Лечащий\_врач

FROM doctor AS d

INNER JOIN patient\_card AS pc ON d.id\_doctor = pc.id\_doctor

WHERE pc.id\_card = 1;

-- 3 вопрос

SELECT d.cabinet AS Номер\_кабинета,

d.visitors\_day AS День\_приема,

CONCAT(d.visitors\_hour, ' - ', d.end\_day) AS Часы\_приема

FROM doctor AS d

WHERE d.id\_doctor = 5;

-- 4 вопрос

SELECT pc.second\_name AS Имя\_пациента, pc.first\_name AS Фамилия\_пациента, n.visit\_date AS Дата\_посещения, n.sick\_list AS Наличие\_больничного\_листа

FROM patient\_card AS pc

INNER JOIN note\_in\_card AS nic ON pc.id\_card = nic.id\_card

INNER JOIN note AS n ON nic.id\_note = n.id\_note

WHERE n.id\_doctor = 3

AND n.sick\_list = true

AND to\_date(n.term, 'DD.MM.YYYY') >= current\_date;

-- 5 вопрос

SELECT d.first\_name AS Имя\_врача, d.surname AS Фамилия\_врача, n.appointment AS Назначение, n.diagnose AS Диагноз

FROM doctor AS d

INNER JOIN note AS n ON d.id\_doctor = n.id\_doctor

WHERE n.diagnose = 'ОРЗ';

-- 6 вопрос

SELECT

d.first\_name AS Имя\_врача,

d.surname AS Фамилия\_врача

FROM

doctor AS d

WHERE

d.cabinet = 28

AND d.visitors\_day = 'Среда'

AND '13:30' BETWEEN d.visitors\_hour AND d.end\_day;

-- 7 вопрос

SELECT

pc.second\_name AS Имя,

pc.first\_name AS Фамилия,

pc.surname AS Отчество,

COUNT(\*) AS Количество\_посещений

FROM patient\_card AS pc

INNER JOIN note\_in\_card AS nic ON pc.id\_card = nic.id\_card

INNER JOIN note AS n ON nic.id\_note = n.id\_note

WHERE pc.id\_card = 3

AND to\_date(n.visit\_date, 'DD.MM.YYYY') >= (CURRENT\_DATE - INTERVAL '1 month')

GROUP BY pc.second\_name, pc.first\_name, pc.surname;

-- 8 вопрос

SELECT d.first\_name AS Имя\_врача, d.surname AS Фамилия\_врача, COALESCE(COUNT(n.id\_note), 0) AS Количество\_пациентов

FROM doctor AS d

LEFT JOIN note AS n ON d.id\_doctor = n.id\_doctor AND to\_date(n.visit\_date, 'DD.MM.YYYY') >= (CURRENT\_DATE - INTERVAL '1 month')

GROUP BY d.first\_name, d.surname, d.id\_doctor

ORDER BY d.id\_doctor;