

АДМИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт – Петербургский национальный исследовательский  
университет информационных технологий, механики и оптики»

Факультет инфокоммуникационных технологий

Лабораторная работа №4

“Реализация SQL-запросов”

Выполнила студентка группы К3242

Тойвонен Полина Денисовна

Санкт- Петербург, 2020

**Цель работы:** Требуется написать определенного количества запросов на определенное количество баллов, зависящее от оценки, на которую претендует студент.

**Оборудование:** УМК (факультет СПО Университета ИТМО)

**Программные средства:** PostgreSQL, PGAdmin

**Практическое задание:**

Создать программную систему, предназначенную для администратора лечебной клиники.

Прием пациентов ведут несколько врачей различных специализаций. На каждого пациента клиники заводится медицинская карта, в которой отражается вся информация по личным данным больного и истории его заболеваний (диагнозы). При очередном посещении врача в карте отражается дата и время приема, диагноз, текущее состояние больного, рекомендации по лечению. Так как прием ведется только на коммерческой основе, после очередного посещения пациент должен оплатить медицинские услуги (каждый прием оплачивается отдельно). Расчет стоимости посещения определяется врачом согласно прейскуранту по клинике.

Для ведения внутренней отчетности необходима следующая информация о врачах: фамилия, имя, отчество, специальность, образование, пол, дата рождения и дата начала и окончания работы в клинике, данные по трудовому договору. Для каждого врача составляется график работы с указанием рабочих и выходных дней.

Прием пациентов врачи могут вести в разных кабинетах. Каждый кабинет имеет определенный режим работы, ответственного и внутренний телефон.

Перечень возможных запросов к базе данных:

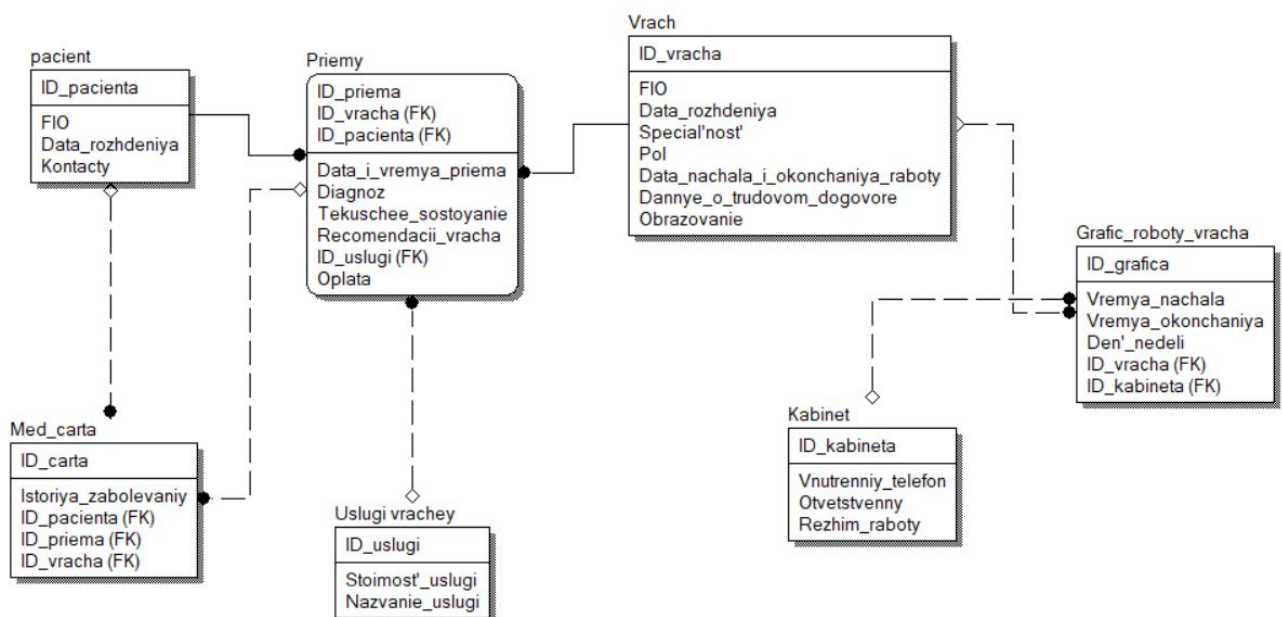
1. Вывести по алфавиту список всех пациентов заданного врача с датами и стоимостью приемов.

2. Вывести телефоны всех пациентов, которые посещали отоларингологов и год рождения которых больше, чем 1987.
3. Вывести список врачей, в графике которых среди рабочих дней имеется заданный.
4. Количество приемов пациентов по датам.
5. Вычислить суммарную стоимость лечения пациентов по дням и по врачам.
6. Список пациентов, уже оплативших лечение.

Перечень возможных отчетов:

1. Отчет о работе врачей в заданный промежуток времени с указанием списка принятых пациентов, их диагноза и стоимости услуг с вычислением суммарного дохода по каждому врачу.

**Скрин модели БД:**



**Список запросов:**

1. Выбор значений, заданных атрибутов из более, чем двух таблиц, с сортировкой;
2. Использование условий WHERE, состоящих из более, чем одного условия;
3. Использование функций для работы с датами;
4. Использование строковых функций;
5. Запрос с использованием подзапросов;
6. Вычисление групповой (агрегатной) функции;
7. Вычисление групповой (агрегатной) функции с условием HAVING;
8. Использование предикатов EXISTS, ALL, SOME и ANY;
9. Использование запросов с операциями реляционной алгебры (объединение, пересечение и т.д.);
10. Использование объединений запросов (inner join и т.д.).

#### Выполнение запросов:

1. Вывод id и ФИО всех врачей, которые учились в Медико-социальном институте, по возрастанию id.

```
SELECT id_doctor, fio FROM public."doctors" WHERE "doctors".education = 'Saint-Petersburg Medico-Social Institute' ORDER BY id_doctor;
```

	id_doctor [PK] integer	fio text
1	3	Sergeeva Anna Vladimirovna
2	4	Gonchar Dmitriy Valentinovich

2. Вывод информации об оплаченных приемах, где id услуги 3

```
SELECT * FROM public."appointments" WHERE payment = 'yes' and id_service = 3;
```

Результат	План выполнения		Сообщения		Notifications				
	id_appointment [PK] integer	id_doctor integer	id_pacient integer	id_service integer	date_and_time text	current_state text	diagnosis text	doctors_recommendation text	payment text
1	2	2	4	3	26.10.2019 10.00	heart pain, dizziness	rheumatic hear...	diclofenac (dicloran, voltaren, dicl...	yes

### 3. Вывод возраста всех врачей.

```
SELECT id_doctor, fio, age(current_date, "doctors"."date_of_birth") FROM
public."doctors";
```

	id_doctor [PK] integer	fio text	age interval
1	1	Chernik...	41 years 5 mons 19 days
2	2	Semenov...	55 years 6 days
3	3	Sergeev...	37 years 8 mons 26 days
4	4	Goncharov...	53 years 4 mons 4 days
5	5	Potashnikov...	40 years 5 mons 15 days

### 4. Переворачивание названий услуг.

```
SELECT reverse(the_name_of_the_service) as name_service FROM
public."medical_services";
```

	name_service text
1	noitatlusnoc srotcod
2	sdileye eht fo senarbmern suocum dna niks eht fo snoisel ngineb fo )noitaziropav( lavomer resaL
3	)traeh eht fo dnuosartlu ,yhpargoidracohcE( yhpargoidracohcE
4	)muidem ,llams( scinilc tneitaptuo ni gnisserD
5	niev a morf gnilpmas doolB

### 5. Вывод выборочной информации о приеме у кардиолога.

```
SELECT id_doctor, date_and_time, diagnosis, id_service FROM
public."appointments" WHERE id_doctor = (SELECT id_doctor FROM
public."doctors" WHERE (specialization = (SELECT specialization FROM
public."doctors" WHERE specialization = 'cardiologist')));
```

	id_doctor integer	date_and_time text	diagnosis text	id_service integer
1	2	26.10.2019 10.00	rheumatic hear...	3

## 6. Средний возраст врачей.

```
SELECT avg(age(current_date, "doctors"."date_of_birth")) FROM
public."doctors";
```

	avg interval
1	45 years 6 mons 38 days

## 7. Количество неоплаченных приёмов.

```
SELECT COUNT(*) FROM public."appointments" WHERE (payment = 'no');
```

	count bigint
1	1

## 8. Количество услуг, которые стоят больше 500.

```
SELECT COUNT(*) FROM public."medical_services" GROUP BY
service_cost HAVING service_cost >'500';
```

	count	
	bigint	
1		1
2		1
3		1

**9. Вывод информации о пациентах, которые были на приеме 4 августа в час дня.**

```
SELECT * FROM public."med_card" WHERE id_pacient = ANY ( SELECT
id_pacient FROM public.appointments WHERE date_and_time = '04.08.2019
13.00');
```

	id_card [PK] integer	id_pacient integer	id_appointment integer	id_doctor integer	the_history_of_diseases text
1	5	5	5	4	1. visit an ophthalmologist. myo...

**10. Вывод пациентов, которые имеют карту, но не были на приеме.**



```
SELECT id_pacient FROM public."med_card" EXCEPT SELECT id_pacient
FROM public."appointments";
```

	id_pacient integer

(Таких пациентов нет)

**11. Вывод ФИО пациента, у которого раньше всех был прием.**

```
SELECT "patients".fio as patient FROM public."patients" INNER JOIN
public."appointments" ON "patients".id_pacient = "appointments".id_pacient
ORDER BY "appointments".date_and_time limit 1;
```

	<p><b>pacient</b></p> <p>text</p> 
1	Miroshnichenko Vladimir Nikolaevich

**Вывод:** в ходе выполнения лабораторной работы были реализованы некоторые запросы к базе данных, содержащие в себе различные функции, условия и операции; получены навыки работы с SQL запросами.