

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ
по лабораторной работе №11
«Создание приложения»
по дисциплине «Базы данных»

Студент гр. 8383

Преподаватель

Киреев К.А.

Фомичева Т.Г.

Санкт-Петербург

2020

Содержание.

Содержание.....	2
Цель работы.....	4
Задание.....	4
Выполнение работы.....	6
1. Этап проектирования БД методом: «сущность - связь» (ER-методом).....	6
2. Генерация предварительных отношений и проверка их на соответствие НФБК. 7	
3. Проектирование базы данных в MS Access	14
4. Разработка запросов	17
5. Разработка интерфейса пользователя	28
5.1. Основная информация.....	28
5.2. Форма авторизации	30
5.3. Форма «АвторизацияРегистратура».....	31
5.4. Форма «АвторизацияГлавныйВрач»	33
5.5. Форма «АвторизацияВрач»	35
5.6. Форма «1 Информация о больном».....	35
5.7. Форма «2 ФИО лечащего врача больного».....	36
5.8. Форма «3 Дни приёма врача».....	37
5.9. Форма «Расписание подчиненная форма1».....	38
5.10. Форма «4 Больные на лечении».....	38
5.11. Форма «5 Назначения при заболевании».....	39
5.12. Форма «6 Врач в кабинете».....	40
5.13. Форма «7 Посещения за последний месяц»	42
5.14. Форма «8 Количество пациентов за прошедший месяц».....	43
5.15. Форма «Добавить_кабинет».....	44
5.16. Форма «Добавить_врача».....	44

5.17.	Форма «Новое_посещение»	45
5.18.	Форма «Подчиненная_адрес_врач»	46
5.19.	Форма «Новая_карточка»	47
5.20.	Форма «Уволить_врача»	48
5.21.	Форма «Выдать_больничный_лист»	49
5.22.	Отчет «Расписание_работы»	49
5.23.	Отчет «Справка»	51
	Выводы.....	52

Цель работы.

Создание небольшой автоматизированной системы, объединяющей таблицы, запросы, экранные формы и отчеты, и предназначенной для решения задачи из некоторой предметной области.

Задание.

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 врачах, 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?

Необходимо предусмотреть возможность выдачи пациенту больничного листа и справки установленного образца, в которой указано, когда, и по какому поводу пациент обращался в поликлинику в текущем году. Расписание работы врачей поликлиники, которое можно повесить в холле, также должно формироваться автоматически.

Выполнение работы.

1. Этап проектирования БД методом: «сущность - связь» (ER-методом).

Была спроектирована ER-модель, представленная на рис. 1.1.

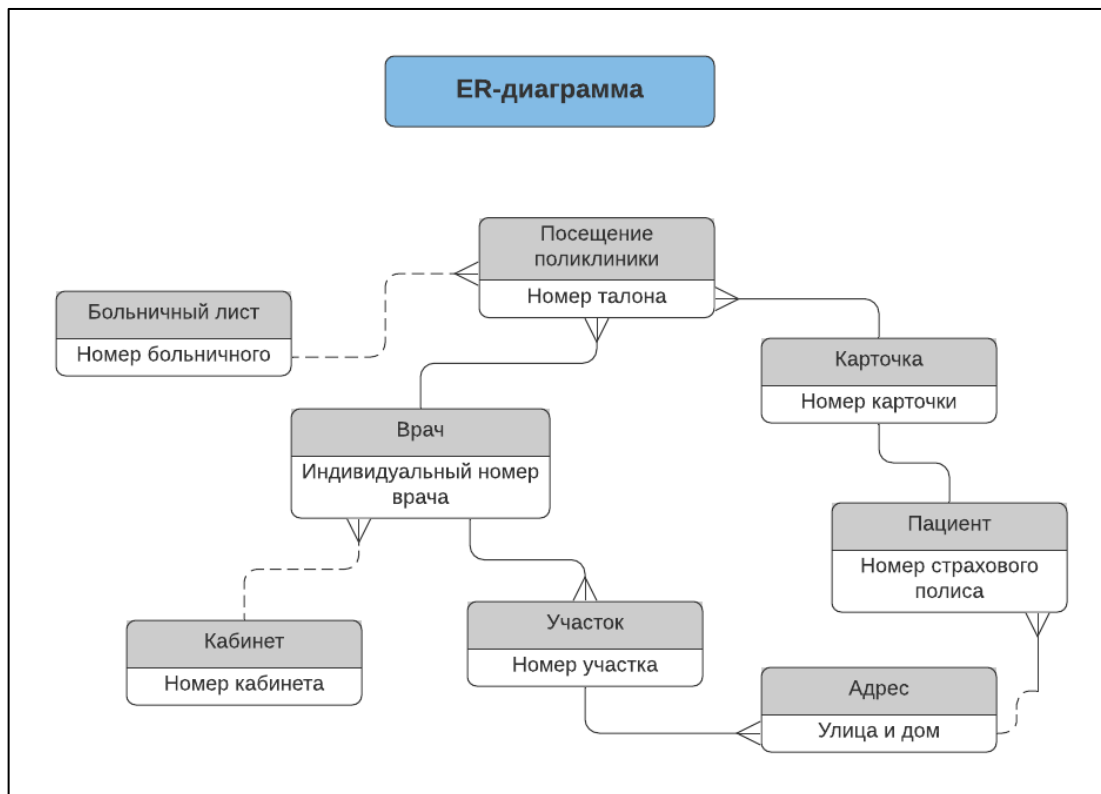


Рисунок 1.1 – ER-модель

ER-модель содержит следующие сущности:

1. Сущность «Посещение поликлиники» содержит основную информацию о посещении пациентом поликлиники. Ключевое поле – Номер талона.
2. Сущность «Больничный лист» содержит данные о больничном листе пациента. Ключевое поле – Номер больничного.
3. Сущность «Карточка» содержит информацию о пациентах, прикрепленных к поликлинике, требуемую для лечения. Ключевое поле – Номер карточки

4. Сущность «Пациент» содержит личную информацию о пациентах, прикрепленных к поликлинике. Ключевое поле – Номер страхового полиса.
5. Сущность «Врач» содержит информацию о врачах, работающих в поликлинике. Ключевое поле – Индивидуальный номер врача.
6. Сущность «Кабинет» содержит информацию о кабинетах поликлиники. Ключевое поле – Номер кабинета.
7. Сущность «Участок» содержит информацию об участках, относящихся к поликлинике. Ключевое поле – Номер участка.
8. Сущность «Адрес» содержит информацию о прикреплении пациента к участку поликлиники. Ключевое поле – Улица и дом.

2. Генерация предварительных отношений и проверка их на соответствие НФБК.

Рассмотрим правила генерации отношений:



Связь между сущностью «Карточка» и сущностью «Пациент» со стороны сущности «Карточка» можно обозначить «карточка принадлежит пациенту», а со стороны сущности «Пациент»: «пациент владеет карточкой». Карточка заводится только на одного пациента, и каждый пациент имеет в регистратуре только одну карточку. Связь обязательна с обеих сторон, так как больной обязательно должен иметь карточку в регистратуре, а карточка заводится на конкретного больного. Следовательно, связь между отношениями должна быть 1:1.

По правилу 1 генерируется одно отношение:

Карточка (Номер карточки, Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст, Номер страхового полиса, Дата заполнения карточки, Номер паспорта)



Связь между сущностью «Больничный лист» и сущностью «Посещение поликлиники» со стороны сущности «Больничный лист» можно обозначить «БЛ может быть выписан в результате посещения», а со стороны сущности «Посещение поликлиники»: «посещение поликлиники может изменить больничный лист». Больничный лист может быть выписан в результате нескольких посещений больным врача, а посещение поликлиники не всегда связано с выдачей, продлением или закрытием больничного листа. Следовательно, связь между отношениями должна быть 1:n.

По правилу 5 генерируется три отношения:

Больничный лист (Номер больничного, Срок истечения)

Изменение больничного листа (Номер больничного, Номер талона, Действие)

Посещение поликлиники (Номер талона, Дата посещения, Жалобы, ИИ врача, Номер карточки, Диагноз, Назначение)



Связь между сущностью «Посещение поликлиники» и сущностью «Карточка» со стороны сущности «Посещение поликлиники» можно обозначить «посещение заносится в карточку», а со стороны сущности «Карточка»: «карточка заполняется посещениями». Посещение записывается в одну конкретную карточку, а карточка может заполняться несколькими посещениями. Следовательно, связь между отношениями должна быть 1:n.

По правилу 4 генерируется два отношения, ключ сущности «Карточка» был добавлен как атрибут в отношение «Посещение поликлиники»:

Посещение поликлиники (Номер талона, Дата посещения, Жалобы, ИИ врача, Номер карточки, Диагноз, Назначение)

Карточка (Номер карточки, Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст, Номер страхового полиса, Дата заполнения карточки, Номер паспорта)



Связь между сущностью «Врач» и сущностью «Посещение поликлиники» со стороны сущности «Врач» можно обозначить «врач осматривает пациента во время посещения», а со стороны сущности «Посещение поликлиники»: «посещение пациентом поликлиники обслуживается врачом». Осмотр во время посещения проводит только один врач, а врач может осматривать разных пациентов в разные посещения. Следовательно, связь между отношениями должна быть 1:n.

По правилу 4 генерируется два отношения, ключ сущности «Врач» был добавлен как атрибут в отношение «Посещение поликлиники»:

Посещение поликлиники (Номер талона, Дата посещения, Жалобы, ИИ врача, Номер карточки, Диагноз, Назначение)

Врач (Индивидуальный номер врача, Фамилия, Имя, Отчество, Категория, Стаж работы, Дата рождения)



Связь между сущностью «Врач» и сущностью «Кабинет» со стороны сущности «Врач» можно обозначить «врач принимает в кабинете», а со стороны сущности «Кабинет»: «кабинет может быть занят врачом». Врач принимает в одном и том же кабинете, но в другую смену или в другие дни в этом кабинете может работать другой, приписанный к этому кабинету врач. Следовательно, связь между отношениями должна быть 1:n.

По правилу 5 генерируется три отношения:

Врач (Индивидуальный номер врача, Фамилия, Имя, Отчество, Категория, Стаж работы, Дата рождения)

Расписание (ИН кабинета, ИН врача, Дни, Смена)

Кабинет (ИН кабинета, Корпус, Этаж)

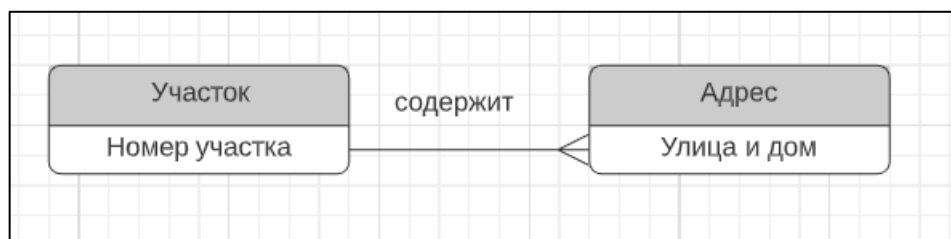


Связь между сущностью «Врач» и сущностью «Участок» со стороны сущности «Врач» можно обозначить «врач обслуживает участок», а со стороны сущности «Участок»: «участок обслуживается врачом». На участке один врач, но допускается обслуживание одним врачом нескольких участков. Следовательно, связь между отношениями должна быть 1:n.

По правилу 4 генерируется два отношения, ключ сущности «Врач» был добавлен как атрибут в отношение «Участок»:

Участок (Номер участка, ИН врача)

Врач (Индивидуальный номер врача, Фамилия, Имя, Отчество, Категория, Стаж работы, Дата рождения)

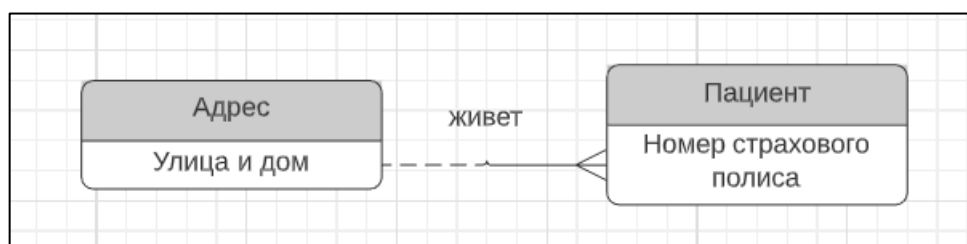


Связь между сущностью «Адрес» и сущностью «Участок» со стороны сущности «Адрес» можно обозначить «адрес входит в участок», а со стороны сущности «Участок»: «участок содержит адреса». Участок может содержать множество адресов, а адрес может входить только в один участок. Следовательно, связь между отношениями должна быть 1:n.

По правилу 4 генерируется два отношения, ключ сущности «Участок» был добавлен как атрибут в отношение «Адрес»:

Адрес (Улица и дом, Номер участка)

Участок (Номер участка, ИН врача)



Связь между сущностью «Адрес» и сущностью «Пациент» со стороны сущности «Адрес» можно обозначить «по адресу проживает пациент», а со стороны сущности «Пациент»: «пациент имеет адрес проживания». По одному адресу живет много пациентов, но каждый из них имеет только один адрес. Связь

обязательна только со стороны пациента: пациент обязан иметь место жительства, но пациенты не обязаны проживать по каждому адресу. Следовательно, связь между отношениями должна быть 1:n.

По правилу 4 генерируется два отношения, ключ сущности «Адрес» был добавлен как атрибут в отношение «Карточка». Стоит отметить, что в предыдущих шагах сущности «Пациент» и «Карточка» были объединены в одно отношение «Карточка»:

Карточка (Номер карточки, Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст, Номер страхового полиса, Дата заполнения карточки, Номер паспорта)

Адрес (Улица и дом, Номер участка)

В результате были получены следующие предварительные отношения:

Посещение поликлиники (Номер талона, Дата посещения, Жалобы, ИИН врача, Номер карточки, Диагноз, Назначение)

Врач (Индивидуальный номер врача, Фамилия, Имя, Отчество, Категория, Стаж работы, Дата рождения)

Карточка (Номер карточки, Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст, Номер страхового полиса, Дата заполнения карточки, Номер паспорта)

Адрес (Улица и дом, Номер участка)

Участок (Номер участка, ИИН врача)

Расписание (ИИН кабинета, ИИН врача, Дни, Смена)

Кабинет (ИИН кабинета, Корпус, Этаж)

Больничный лист (Номер больничного, Срок истечения)

Изменение больничного листа (Номер больничного, Номер талона, Действие)

Определим функциональную зависимость м/у атрибутами каждого отношения и проверим соответствие ее НФБК.

1. В отношении «Врач» функциональные зависимости выглядят следующим образом:

Индивидуальный номер врача → Фамилия, Имя, Отчество, Категория, Стаж работы, Дата рождения

Детерминант является ключом отношения, других зависимостей нет, поэтому отношение находится в НФБК.

2. В отношении «Кабинет» функциональные зависимости выглядят следующим образом:

ИН кабинета → Корпус, Этаж

Детерминант является ключом отношения, других зависимостей нет, поэтому отношение находится в НФБК.

3. В отношении «Расписание» функциональных зависимостей нет, как и ключей, так что отношение находится в НФБК

4. В отношении «Участок» функциональные зависимости выглядят следующим образом:

Номер участка → ИН врача

Детерминант является ключом, других зависимостей нет, поэтому отношение «Участок» находится в НФБК.

5. В отношении «Адрес» функциональные зависимости выглядят следующим образом:

Улица и дом → Номер участка

Детерминант является ключом, других зависимостей нет, поэтому отношение «Адрес» находится в НФБК.

6. В отношении «Карточка» функциональные зависимости выглядят следующим образом:

Номер карточки → Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст, Номер страхового полиса, Дата заполнения карточки, Номер паспорта

Номер паспорта → Фамилия пациента, Имя пациента, Отчество пациента, Улица и дом, Номер квартиры, Пол, Возраст

Номер страхового полиса → Фамилия пациента, Имя пациента, Отчество пациента, Пол, Возраст, Номер паспорта

Все детерминанты являются возможными ключами, других зависимостей нет, поэтому отношение «Карточка» находится в НФБК.

7. В отношении «Посещение поликлиники» функциональные зависимости выглядят следующим образом:

Номер талона → Дата посещения, Жалобы, ИН врача, Номер карточки, Диагноз, Назначение

Детерминант является ключом, других зависимостей нет, поэтому отношение «Посещение поликлиники» находится в НФБК.

8. В отношении «Больничный лист» функциональные зависимости выглядят следующим образом:

Номер больничного → Срок истечения

Детерминант является ключом, других зависимостей нет, поэтому отношение «Больничный лист» находится в НФБК.

9. В отношении «Изменение больничного листа» функциональных зависимостей нет, как и ключей, так что отношение находится в НФБК

Все отношения находятся в НФБК.

3. Проектирование базы данных в MS Access

В MS Access была спроектирована база данных, основанная на вышеперечисленных отношениях. Схема данных представлена на рис. 1.2.

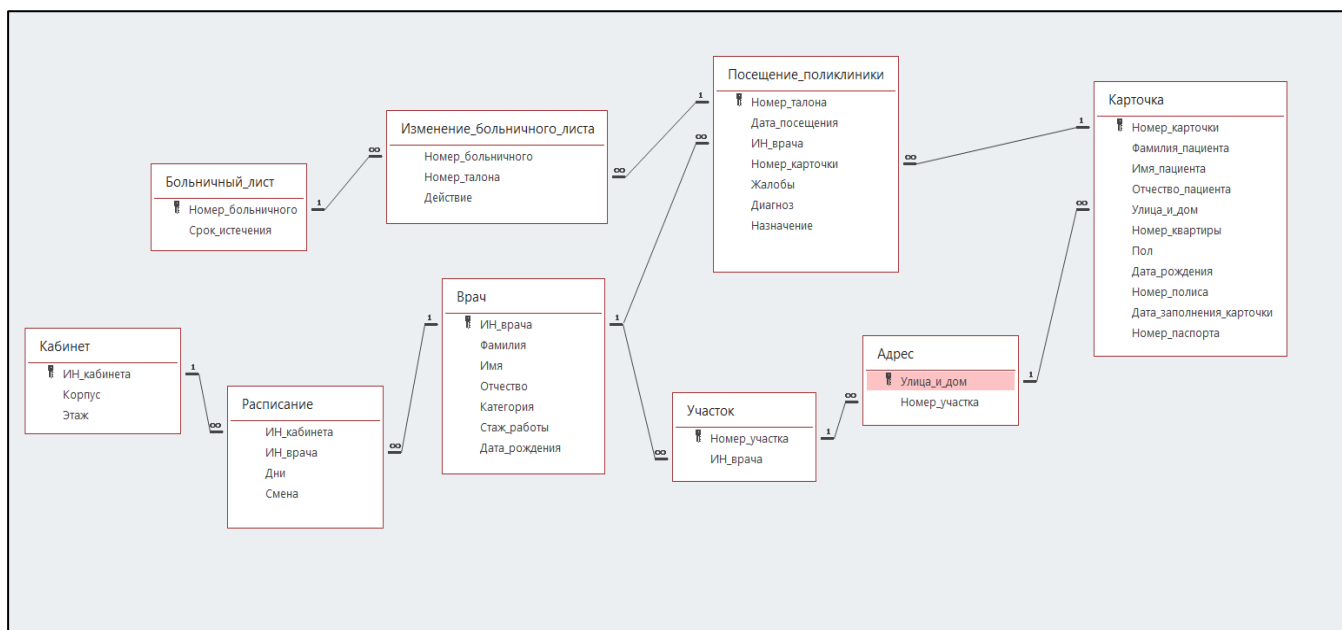


Рисунок 1.2 – Схема данных в MS Access

Далее было выполнено заполнение таблиц данными.

Таблица «Кабинет» представлена на рис. 2.1.

	ИН_кабинета	Корпус	Этаж
+	1321	1	3
+	2221	2	2
+	2222	2	2
+	3201	3	2
+	4132	4	1
+	4133	4	1
+	5132	5	1

Рисунок 2.1 – Таблица «Кабинет»

Таблица «Врач» представлена на рис. 2.2.

	ИН_врача	Фамилия	Имя	Отчество	Категория	Стаж_работы	Дата_рождения
+	100101	Иванов	Иван	Иванович	высшая	20	11.04.1957
+	100102	Степанов	Илья	Владимирович	первая	12	04.08.1977
+	100103	Сергеев	Сергей	Максимович	вторая	11	05.01.1985
+	100104	Клюев	Николай	Иванович	вторая	11	12.12.1968
+	100105	Зайцев	Дмитрий	Николаевич	первая	9	11.11.1983
+	100106	Гоголев	Михаил	Петрович	высшая	25	10.04.1960

Рисунок 2.2 – Таблица «Врач»

Таблица «Участок» представлена на рис. 2.3.

	Номер_участка ▾	ИН_врача ▾
+	1	100101
+	2	100103
+	3	100102
+	4	100104
+	5	100105
+	6	100106

Рисунок 2.3 – Таблица «Участок»

Таблица «Адрес» представлена на рис. 2.4.

	Улица_и_дом ▾	Номер_участка ▾
+	Жукова, 1	2
+	Звездная, 5	2
+	Ленина, 8	1
+	Менделеева, 3	3
+	Невский проспект, 8	3

Рисунок 2.4 – Таблица «Адрес»

Таблица «Больничный лист» представлена на рис. 2.5.

	Номер_больничного ▾	Срок_истечения ▾
+	101	31.12.2020
+	102	30.01.2021
+	103	30.12.2020

Рисунок 2.5 – Таблица «Больничный лист»

Таблица «Изменение больничного листа» представлена на рис. 2.6.

	Номер_талс ▾	Номер_больничного ▾	Действие ▾
	2001	101	Открыть
	2002	102	Открыть
	2003	103	Открыть

Рисунок 2.6 – Таблица «Изменение больничного листа»

Таблица «Расписание» представлена на рис. 2.7.

ИН_кабинета ▾	ИН_врача ▾	Дни ▾	Смена ▾
1321	100103	пятница	утро
1321	100103	среда	утро
2221	100101	вторник	утро
2222	100101	четверг	вечер
4132	100103	понедельник	утро
4133	100102	понедельник	вечер

Рисунок 2.7 – Таблица «Расписание»

Таблица «Карточка» представлена на рис. 2.8.

Номер_карточки ▾	Фамилия_пациента ▾	Имя_пациента ▾	Отчество_пациента ▾	Улица_и_дом ▾	Ном ▾	Пол ▾	Дата_рожд ▾	Номер_полиса ▾	Дата_заполн ▾	Номер_пасп ▾
51	Киреев	Константин	Александрович	Ленина, 8	18	М	09.07.2000	900101	13.11.2020	2590 111111
52	Муковский	Даниил	Владимирович	Звездная, 5	4	М	12.12.1900	900102	15.10.2020	2552 135325
53	Сосновский	Дмитрий	Николаевич	Жукова, 1	77	М	01.02.2009	900103	03.12.2020	2601 983432
54	Осовик	Евгений	Евгеньевич	Невский проспе	12	М	13.12.1997	900104	27.12.2020	2981 793432
55	Ефремов	Иван	Никитич	Менделеева, 3	18	М	01.04.2000	900111	27.12.2020	2201 983432

Рисунок 2.8 – Таблица «Карточка»

Таблица «Посещение поликлиники» представлена на рис. 2.9.

Номер_талона ▾	Дата_посещения ▾	ИН_врача ▾	Номер_карточки ▾	Жалобы ▾	Диагноз ▾	Назначение ▾
2001	01.12.2020	100101	51	плохое самочувствие	простуда	ринза
2002	13.12.2020	100103	52	тяжелое дыхание	рак легких	отдых
2003	18.12.2020	100103	52	боли в животе	рак кишечника	отдых
2004	19.12.2020	100103	53	болит голова	простуда	чай
2005	27.12.2020	100102	54	болит голова	рак головы	арбидол

Рисунок 2.9 – Таблица «Посещение поликлиники»

Для полей были настроены их свойства, например, поле номер паспорта в таблице «Карточка» имеет условие на значение «Like "#####"», что обеспечивает возможность корректного ввода номера паспорта.

4. Разработка запросов

Для наполнения интерфейса пользователя и обеспечения его действий был разработан ряд запросов.

- Запрос «1 Адрес, дата и диагноз» необходим для получения информации об адресе больного, дате его последнего посещения поликлиники и поставленном диагнозе. SQL-код запроса представлен ниже:

```
SELECT Карточка.Фамилия_пациента, Карточка.Улица_и_дом, Карточка.Номер_квартиры,
Max(Посещение_поликлиники.Дата_посещения) AS [Max-Дата_посещения], Last(Посеще-
ние_поликлиники.Диагноз) AS [Last-Диагноз]
FROM Карточка INNER JOIN Посещение_поликлиники ON Карточка.Номер_карточки = По-
сещение_поликлиники.Номер_карточки
GROUP BY Карточка.Фамилия_пациента, Карточка.Улица_и_дом, Карточка.Номер_квар-
тиры, Карточка.Номер_полиса
ORDER BY Карточка.Фамилия_пациента, Max(Посещение_поликлиники.Дата_посещения)
DESC;
```

Результатом запроса является таблица, в которой содержатся необходимые сведения о пациентах. Таблица представлена на рис. 3.

Фамилия_пациента ▾	Улица_и_дом ▾	Ном ▾	Max-Дата_посещения ▾	Last-Диагно ▾
Киреев	Ленина, 8	18	01.12.2020	простуда
Муковский	Звездная, 5	4	18.12.2020	рак кишечника
Осовик	Невский проспе	12	27.12.2020	рак головы
Сосновский	Жукова, 1	77	19.12.2020	простуда

Рисунок 3 – Результат запроса «1 Адрес, дата и диагноз»

- Запрос «2 ФИО лечащего врача» необходим для получения списка пациен-
тов с указанием их лечащих врачей. SQL-код запроса представлен ниже:

```
SELECT Карточка.Фамилия_пациента, [Врач].[Фамилия] & " " & Left([Врач].[Имя],1)
& "." & Left([Врач].[Отчество],1) & "." AS Выражение1
FROM Карточка INNER JOIN (Врач INNER JOIN Посещение_поликлиники ON Врач.ИН_врача
= Посещение_поликлиники.ИН_врача) ON Карточка.Номер_карточки = Посещение_поли-
клиники.Номер_карточки
GROUP BY Карточка.Фамилия_пациента, [Врач].[Фамилия] & " " & Left([Врач].[Имя],1)
& "." & Left([Врач].[Отчество],1) & "." , Карточка.Номер_полиса;
```

Результатом является таблица с двумя столбцами, где указаны фамилии па-
циентов и ФИО лечащих врачей. Таблица представлена на рис. 4.

Фамилия_пациента ▾	Выражение: ▾
Киреев	Иванов И.И.
Муковский	Сергеев С.М.
Осовик	Степанов И.В.
Сосновский	Сергеев С.М.

Рисунок 4 – Результат запроса «2 ФИО лечащего врача»

- Запрос «3 Расписание врача» необходим для получения информации о расписании каждого врача, с указанием кабинета, дней и смен. SQL-код запроса представлен ниже:

```
SELECT Врач.Фамилия, Кабинет.ИН_кабинета, Расписание.Дни, Расписание.Смена, Ка-
бинет.Корпус, Кабинет.Этаж
FROM Кабинет INNER JOIN (Врач INNER JOIN Расписание ON Врач.ИН_врача = Расписа-
ние.ИН_врача) ON Кабинет.ИН_кабинета = Расписание.ИН_кабинета
ORDER BY Врач.Фамилия, Кабинет.ИН_кабинета, Расписание.Дни;
```

Результат выполнения запроса представлен на рис. 5.

Фамилия ▾	ИН_кабинета ▾	Дни ▾	Смена ▾	Корпус ▾	Этаж ▾
Иванов	2221	вторник	утро	2	2
Иванов	2222	четверг	вечер	2	2
Сергеев	1321	пятница	утро	1	3
Сергеев	1321	среда	утро	1	3
Сергеев	4132	понедельник	утро	4	1
Степанов	4133	понедельник	вечер	4	1

Рисунок 5 – Результат запроса «3 Расписание врача»

- Запрос «4 Текущие больные» необходим для определения пациентов, у которых ещё не истек срок больничного листа. SQL-код запроса представлен ниже:

```
SELECT Больничный_лист.Срок_истечения, [Больные у врача].Фамилия_пациента,
[Больные у врача].Имя_пациента, [Больные у врача].Отчество_пациента, [Больные у
врача].Фамилия
FROM Больничный_лист INNER JOIN ([Больные у врача] INNER JOIN Изменение_больнич-
ного_листа ON [Больные у врача].Номер_талона = Изменение_больничного_листа.Но-
мер_талона) ON Больничный_лист.Номер_больничного = Изменение_больничного_ли-
ста.Номер_больничного
```

GROUP BY Больничный_лист.Срок_истечения, [Больные у врача].Фамилия_пациента,
[Больные у врача].Имя_пациента, [Больные у врача].Отчество_пациента, [Больные у
врача].Фамилия
HAVING (((Больничный_лист.Срок_истечения)>Date()));

Результат выполнения запроса представлен на рис. 6.

Срок_истечения	Фамилия_пациента	Имя_пациента	Отчество_пациента	Фамилия
30.12.2020	Муковский	Даниил	Владимирович	Сергеев
31.12.2020	Киреев	Константин	Александрович	Иванов
30.01.2021	Муковский	Даниил	Владимирович	Сергеев

Рисунок 6 – Результат запроса «4 Текущие больные»

- Запрос «5 Назначения при заболевании» необходим для получения списка заболеваний и назначений по ним, с указанием фамилии врача, назначившего лечение. SQL-код представлен ниже:

SELECT Посещение_поликлиники.Назначение, Врач.Фамилия, Посещение_поликлиники.Диагноз

FROM Врач INNER JOIN Посещение_поликлиники ON Врач.ИН_врача = Посещение_поликлиники.ИН_врача;

Результат выполнения запроса представлен на рис. 7.

Диагноз	Назначение	Фамилия
простуда	ринза	Иванов
рак головы	арбидол	Степанов
рак легких	отдых	Сергеев
рак кишечника	отдых	Сергеев
простуда	чай	Сергеев

Рисунок 7 – Результат запроса «5 Назначения при заболевании»

- Запрос «6_1 Врач в кабинете» необходим для получения номера и ФИО врача, который работает в указанном кабинете в указанное время. SQL-код запроса представлен ниже:

SELECT Врач.ИН_врача, Врач.Фамилия, Врач.Имя, Врач.Отчество, Кабинет.ИН_кабинета, Расписание.Дни, Расписание.Смена

FROM Кабинет INNER JOIN (Врач INNER JOIN Расписание ON Врач.ИН_врача = Расписание.ИН_врача) ON Кабинет.ИН_кабинета = Расписание.ИН_кабинета;

Результат выполнения запроса представлен на рис. 8.

ИН_врача ▾	Фамилия ▾	Имя ▾	Отчество ▾	ИН_кабинета ▾	Дни ▾	Смена ▾
100101	Иванов	Иван	Иванович	2221	вторник	утро
100101	Иванов	Иван	Иванович	2222	четверг	вечер
100103	Сергеев	Сергей	Максимович	1321	пятница	утро
100103	Сергеев	Сергей	Максимович	1321	среда	утро
100103	Сергеев	Сергей	Максимович	4132	понедельник	утро
100102	Степанов	Илья	Владимирович	4133	понедельник	вечер

Рисунок 8 – Результат запроса «6_1 Врач в кабинете»

- Запрос «7 Посещения за последний месяц» необходим для получения списка пациентов с указанием количества посещений за последний месяц.

SQL-код запроса представлен ниже:

```
SELECT Карточка.Фамилия_пациента, Карточка.Имя_пациента, Карточка.Отчество_пациента, Count(Mid([Дата_посещения],4,2)) AS Выражение1
FROM Карточка INNER JOIN Посещение_поликлиники ON Карточка.Номер_карточки = Посещение_поликлиники.Номер_карточки
GROUP BY Карточка.Фамилия_пациента, Карточка.Имя_пациента, Карточка.Отчество_пациента, Mid([Дата_посещения],4,2), Карточка.Номер_карточки
HAVING (((Mid([Дата_посещения],4,2))=Month(Date())));
```

Результат выполнения запроса представлен на рис. 9.

Фамилия_пациента ▾	Имя_пациента ▾	Отчество_пациента ▾	Выражение1 ▾
Киреев	Константин	Александрович	1
Муковский	Даниил	Владимирович	2
Осовик	Евгений	Евгеньевич	1
Сосновский	Дмитрий	Николаевич	1

Рисунок 9 – Результат запроса «7 Посещения за последний месяц»

- Запрос «8 Количество пациентов за прошедший месяц» необходим для получения списка врачей с указанием количества пациентов за последний месяц. SQL-код запроса представлен ниже:

```
SELECT Врач.ИН_врача, Врач.Фамилия, Врач.Имя, Врач.Отчество, Count(Mid([Дата_посещения],4,2)) AS Выражение1
FROM Врач INNER JOIN Посещение_поликлиники ON Врач.ИН_врача = Посещение_поликлиники.ИН_врача
```

```
GROUP BY Врач.ИН_врача, Врач.Фамилия, Врач.Имя, Врач.Отчество, Mid([Дата_посещения],4,2)
HAVING (((Mid([Дата_посещения],4,2))=Month(Date()))))
ORDER BY Врач.ИН_врача;
```

Результат выполнения запроса представлен на рис. 10.

ИН_врача	Фамилия	Имя	Отчество	Выражение:
100101	Иванов	Иван	Иванович	1
100102	Степанов	Илья	Владимирович	1
100103	Сергеев	Сергей	Максимович	3

Рисунок 10 – Результат запроса «8 Количество пациентов за прошедший месяц»

- Запрос «Больные у врача» необходим для получения списка пациентов поликлиники с указанием их врачей и посещений. SQL-код запроса представлен ниже:

```
SELECT Карточка.Номер_карточки, Карточка.Фамилия_пациента, Карточка.Имя_пациента,
Карточка.Отчество_пациента, Посещение_поликлиники.Номер_талона, Врач.Фамилия
FROM Карточка INNER JOIN (Врач INNER JOIN Посещение_поликлиники ON Врач.ИН_врача
= Посещение_поликлиники.ИН_врача) ON Карточка.Номер_карточки = Посещение_поли-
клиники.Номер_карточки
```

```
GROUP BY Карточка.Номер_карточки, Карточка.Фамилия_пациента, Карточка.Имя_паци-
ента, Карточка.Отчество_пациента, Посещение_поликлиники.Номер_талона, Врач.Фами-
лия;
```

Результат выполнения запроса представлен на рис. 11.

Номер_карточки	Фамилия_пациента	Имя_пациента	Отчество_пациента	Номер_талона	Фамилия
51	Киреев	Константин	Александрович	2001	Иванов
52	Муковский	Даниил	Владимирович	2002	Сергеев
52	Муковский	Даниил	Владимирович	2003	Сергеев
53	Сосновский	Дмитрий	Николаевич	2004	Сергеев
54	Осовик	Евгений	Евгеньевич	2005	Степанов

Рисунок 11 – Результат запроса «Больные у врача»

- Запрос «Больные у врача» необходим для получения списка пациентов поликлиники с указанием их врачей и посещений. SQL-код запроса представлен ниже:

```
SELECT Карточка.Номер_карточки, Карточка.Фамилия_пациента, Карточка.Имя_пациента,
Карточка.Отчество_пациента, Посещение_поликлиники.Номер_талона, Врач.Фамилия
```

FROM Карточка INNER JOIN (Врач INNER JOIN Посещение_поликлиники ON Врач.ИН_врача = Посещение_поликлиники.ИН_врача) ON Карточка.Номер_карточки = Посещение_поликлиники.Номер_карточки

GROUP BY Карточка.Номер_карточки, Карточка.Фамилия_пациента, Карточка.Имя_пациента, Карточка.Отчество_пациента, Посещение_поликлиники.Номер_талона, Врач.Фамилия;

Результат выполнения запроса представлен на рис. 11.

Номер_карточки	Фамилия_пациента	Имя_пациента	Отчество_пациента	Номер_талона	Фамилия
51	Киреев	Константин	Александрович	2001	Иванов
52	Муковский	Даниил	Владимирович	2002	Сергеев
52	Муковский	Даниил	Владимирович	2003	Сергеев
53	Сосновский	Дмитрий	Николаевич	2004	Сергеев
54	Осовик	Евгений	Евгеньевич	2005	Степанов

Рисунок 11 – Результат запроса «Больные у врача»

- Запрос «Врач в данный момент» необходим для получения врача, который работает в указанном кабинете в текущий момент времени. SQL-код запроса представлен ниже:

SELECT Врач.ИН_врача, Врач.Фамилия, Врач.Имя, Врач.Отчество, Расписание.Дни, Расписание.Смена, Кабинет.ИН_кабинета

FROM Кабинет INNER JOIN (Врач INNER JOIN Расписание ON Врач.ИН_врача = Расписание.ИН_врача) ON Кабинет.ИН_кабинета = Расписание.ИН_кабинета

WHERE (((Расписание.Дни)=WeekdayName(Weekday(Date()),2))) AND ((Расписание.Смена)=IIf(Left(Time(),2)<12,"утро","вечер")) AND ((Кабинет.ИН_кабинета)=[Номер кабинета:]);

Результат выполнения запроса представлен на рис. 12.

ИН_врача	Фамилия	Имя	Отчество	Дни	Смена	ИН_кабинета
100103	Сергеев	Сергей	Максимович	понедельник	утро	4132

Рисунок 12 – Результат запроса «Врач в данный момент»

- Запрос «Выборка_справка» необходим для выдачи справки. Запрос предоставляет основную информацию о пациенте, даты посещений поликлиники и диагнозы. SQL-код запроса представлен ниже:

SELECT Карточка.Фамилия_пациента, Карточка.Имя_пациента, Карточка.Отчество_пациента, Карточка.Дата_рождения, Карточка.Номер_полиса, Посещение_поликлиники.Дата_посещения, Посещение_поликлиники.Диагноз

FROM Карточка INNER JOIN Посещение_поликлиники ON Карточка.Номер_карточки = Посещение_поликлиники.Номер_карточки
WHERE (((Карточка.Номер_полиса)=[Формы]![АвторизацияВрач]![Полис]));

Результат выполнения запроса представлен на рис. 13.

Фамилия_пациента	Имя_пациента	Отчество_пациента	Дата_рождения	Номер_полиса	Дата_посещения	Диагноз
Муковский	Даниил	Владимирович	12.12.1900	900102	13.12.2020	рак легких
Муковский	Даниил	Владимирович	12.12.1900	900102	18.12.2020	рак кишечника

Рисунок 13 – Результат запроса «Выборка_справка»

- Запрос «Расписание работы» необходим для создания расписания работы врачей. Запрос предоставляет информацию об участке, кабинете и смене врачей. SQL-код запроса представлен ниже:

```
SELECT Участок.Номер_участка, Врач.Фамилия, Врач.Имя, Врач.Отчество, Расписание.ИН_кабинета, Расписание.Дни, Расписание.Смена
FROM (Кабинет INNER JOIN (Врач INNER JOIN Расписание ON Врач.ИН_врача = Расписание.ИН_врача) ON Кабинет.ИН_кабинета = Расписание.ИН_кабинета) INNER JOIN Участок
ON Врач.ИН_врача = Участок.ИН_врача
ORDER BY Участок.Номер_участка, Врач.Фамилия, Расписание.ИН_кабинета, Расписание.Дни, Расписание.Смена DESC;
```

Результат выполнения запроса представлен на рис. 14.

Номер_участка	Фамилия	Имя	Отчество	ИН_кабинета	Дни	Смена
1	Иванов	Иван	Иванович	2221	вторник	утро
1	Иванов	Иван	Иванович	2222	четверг	вечер
2	Сергеев	Сергей	Максимович	1321	пятница	утро
2	Сергеев	Сергей	Максимович	1321	среда	утро
2	Сергеев	Сергей	Максимович	4132	понедельник	утро
3	Степанов	Илья	Владимирович	4133	понедельник	вечер

Рисунок 14 – Результат запроса «Расписание работы»

- Запрос «Новая_карточка» - запрос на добавление новой карточки пациента.

SQL-код запроса представлен ниже:

```
INSERT INTO Карточка ( Номер_карточки, Фамилия_пациента, Имя_пациента,
Отчество_пациента, Улица_и_дом, Номер_квартиры, Пол, Дата_рождения, Номер_полиса,
Номер_паспорта, Дата_заполнения_карточки )
```

```
SELECT      Max([Карточка]![Номер_карточки])+1      AS      Выражение1,
[Формы]![Новая_карточка]![Фамилия] AS Выражение2, [Формы]![Новая_карточка]![Имя]
AS      Выражение3,      [Формы]![Новая_карточка]![Отчество]      AS      Выражение4,
```



```

[Формы]![Новая_карточка]![УлицаДом] AS Выражение5,
[Формы]![Новая_карточка]![Квартира] AS Выражение6, [Формы]![Новая_карточка]![Пол]
AS Выражение7, [Формы]![Новая_карточка]![ДатаРождения] AS Выражение8,
[Формы]![Новая_карточка]![Полис] AS Выражение9,
[Формы]![Новая_карточка]![Паспорт] AS Выражение10,
[Формы]![Новая_карточка]![ДатаЗаполнения] AS Выражение11
FROM Карточка;

```

- Запрос «Новая_посещение» - запрос на добавление новой записи о посещении поликлиники. SQL-код запроса представлен ниже:

```

INSERT INTO Посещение_поликлиники ( Номер_талона, Дата_посещения, ИИ_врача,
Номер_карточки, Жалобы, Диагноз, Назначение )

SELECT      Max([Посещение_поликлиники]![Номер_талона])+1 AS      Выражение1,
[Формы]![Новое_посещение]![Дата] AS Выражение2, [Формы]![Новое_посещение]![Врач]
AS      Выражение3,      [Формы]![Новое_посещение]![Карточка]      AS      Выражение4,
[Формы]![Новое_посещение]![Жалобы] AS      Выражение5,
[Формы]![Новое_посещение]![Диагноз] AS      Выражение6,
[Формы]![Новое_посещение]![Назначение] AS      Выражение7
FROM Посещение_поликлиники;

```

- Запрос «Добавить_больничный_лист» - запрос на добавление нового больничного листа. SQL-код запроса представлен ниже:

```

INSERT INTO Больничный_лист ( Номер_больничного, Срок_истечения )

SELECT      Max([Больничный_лист]![Номер_больничного])+1 AS      Выражение1,
[Формы]![Выдать_больничный_лист]![Срок] AS      Выражение2
FROM Больничный_лист;

```

- Запрос «Добавить_действие_с_больничным» - запрос на добавление нового больничного листа. SQL-код запроса представлен ниже:

```

INSERT INTO Изменение_больничного_листа ( Номер_талона, Действие,
Номер_больничного )

SELECT      [Формы]![Выдать_больничный_лист]![Комбинированная0] AS      Выражение1,
"Открыть" AS      Выражение2,
Max([Изменение_больничного_листа]![Номер_больничного])+1 AS      Выражение3

```

FROM Изменение_больничного_листа;

- Запрос «Новый_адрес» - запрос на добавление нового адреса. SQL-код запроса представлен ниже:

```
INSERT INTO Адрес ( Улица_и_дом, Номер_участка )
```

```
SELECT      [Формы]![Новая_карточка]![УлицаДом]      AS      Выражение1,  
[Формы]![Новая_карточка]![Участок] AS Выражение2;
```

- Запрос «Новый_врач» - запрос на добавление нового адреса. SQL-код запроса представлен ниже:

```
INSERT INTO Врач ( ИН_врача, Фамилия, Имя, Отчество, Категория, Стаж_работы,  
Дата_рождения )
```

```
SELECT Max([Врач]![ИН_врача])+1 AS Выражение1, [Формы]![Добавить_врача]![Фамилия]  
AS      Выражение2,      [Формы]![Добавить_врача]![Имя]      AS      Выражение3,  
[Формы]![Добавить_врача]![Отчество]      AS      Выражение4,  
[Формы]![Добавить_врача]![Категория]      AS      Выражение5,  
[Формы]![Добавить_врача]![Стаж]      AS      Выражение6,  
[Формы]![Добавить_врача]![ДатаРождения] AS Выражение7
```

```
FROM Врач;
```

- Запрос «Новый_кабинет» - запрос на добавление нового адреса. SQL-код запроса представлен ниже:

```
INSERT INTO Кабинет ( ИН_кабинета, Корпус, Этаж )
```

```
SELECT      [Формы]![Добавить_кабинет]![Номер]      AS      Выражение1,  
[Формы]![Добавить_кабинет]![Корпус]      AS      Выражение2,  
[Формы]![Добавить_кабинет]![Этаж] AS Выражение3;
```

- Запрос «Обновить_посещение» - запрос на передачу больных другому врачу. SQL-код запроса представлен ниже:

```
UPDATE      Посещение_поликлиники      SET      Посещение_поликлиники.ИН_врача      =  
[Формы]![Уволить_врача]![Кому]
```

```
WHERE      (((Посещение_поликлиники.ИН_врача)=[Формы]![Уволить_врача]![Кого]));
```

- Запрос «Обновить_участок» - запрос на передачу участков другому врачу. SQL-код запроса представлен ниже:

```
UPDATE Врач INNER JOIN Участок ON Врач.ИН_врача = Участок.ИН_врача SET
Участок.ИН_врача = [Формы]![Уволить_врача]![Кому]
WHERE (((Врач.ИН_врача)=[Формы]![Уволить_врача]![Кого]));
```

- Запрос «Удалить_врача» - удаление записи о враче из таблицы «Врач».

SQL-код запроса представлен ниже:

```
DELETE Врач.ИН_врача
FROM Врач
WHERE (((Врач.ИН_врача)=[Формы]![Уволить_врача]![Кого]));
```

- Запрос «Удалить_кабинет» - удаление записи о расписании врача из таблицы «Расписание». SQL-код запроса представлен ниже:

```
DELETE Расписание.ИН_врача
FROM Расписание
WHERE (((Расписание.ИН_врача)=[Формы]![Уволить_врача]![Кого]));
```

5. Разработка интерфейса пользователя

5.1. Основная информация

Схематичное представление интерфейса пользователя представлено на рис.

15. Стрелочками обозначается возможность перехода между формами.

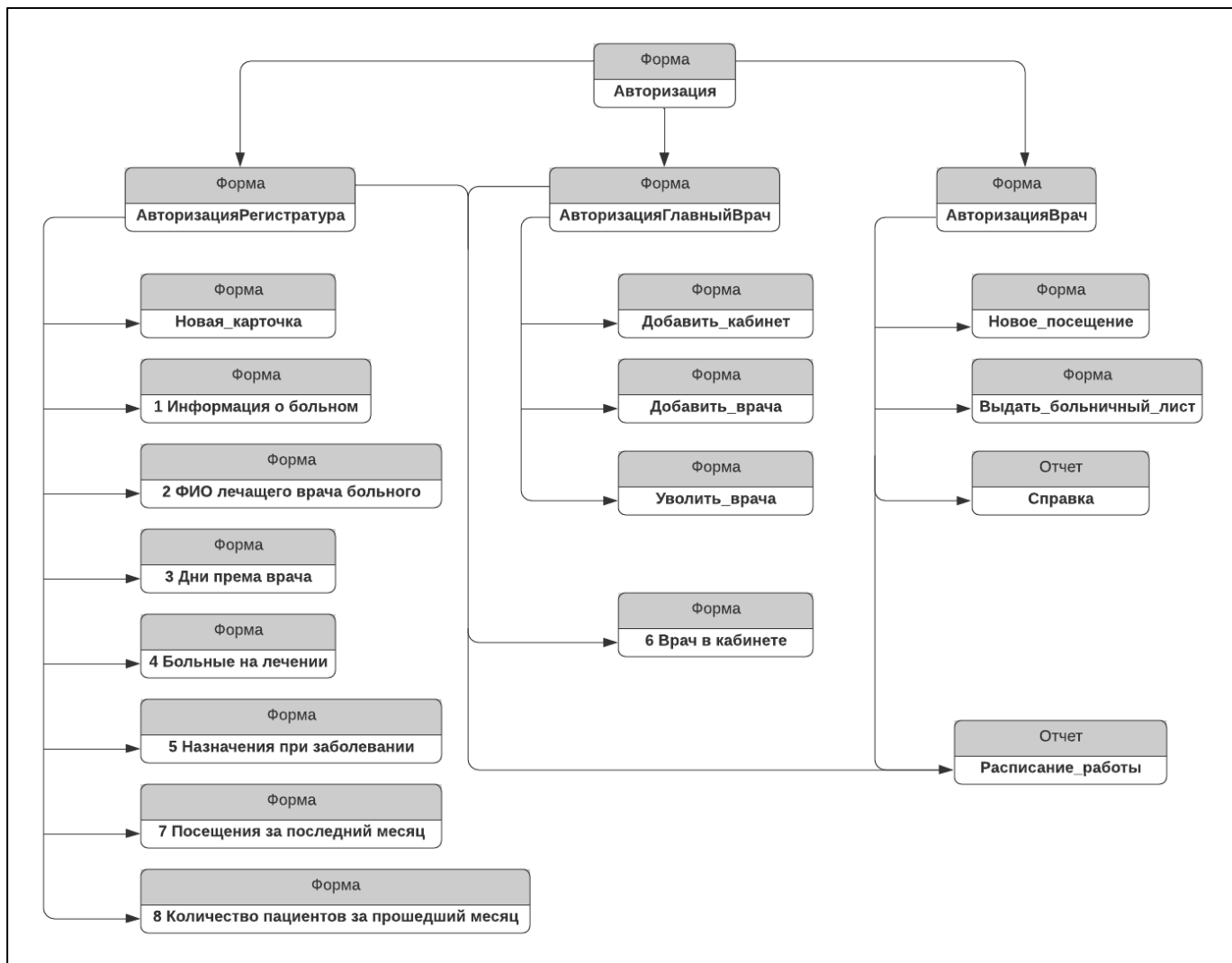


Рисунок 15 – Схема интерфейса

Основные задачи разработки приложения и формы, реализующие их, представлены ниже.

1. Уволить или добавить нового врача;

- а. Форма «Добавить_врача» позволяет добавлять врача (при добавлении можно указать основные данные)

- b. Форма «Удалить_врача» позволяет удалить врача поликлиники и передать его пациентов другому врачу
- 2. Добавить новый кабинет;
 - a. Форма «Добавить_кабинет» позволяет добавлять кабинет (при добавлении можно указать характеристики)
- 3. Зарегистрировать нового пациента и заполнить его карточку – Форма «Новая_карточка»;
- 4. Внести запись о новом посещении в карточку пациента – Форма «Новое_посещение»;
- 5. Выписать больничный лист при посещении – Форма «Выдать_больничный_лист»;
- 6. Узнать требуемую работникам поликлиники информацию о пациенте;
 - a. Форма «1 Информация о больном» позволяет узнать об адресе, последнем посещении и диагнозе пациента
 - b. Форма «2 ФИО лечащего врача больного» позволяет узнать ФИО лечащего врача пациента
 - c. Форма «7 Посещения за последний месяц» позволяет узнать количество посещений поликлиники пациентом за последний месяц
- 7. Узнать требуемую работникам поликлиники информацию о враче;
 - a. Форма «3 Дни приема врача» позволяет узнать о расписании приема врача
 - b. Форма «4 Больные на лечении» позволяет узнать пациентов, находящихся в данный момент на лечении у врача
 - c. Форма «6 Врач в кабинете» позволяет узнать какой врач находится в указанном кабинете в указанное время
 - d. Форма «8 Количество пациентов за прошедший месяц» позволяет узнать количество пациентов, осмотренных врачом

8. Узнать назначения врачей для конкретного заболевания – Форма «5 Назначения при заболевании»
9. Выдать справку, в которой указано, когда, и по какому поводу пациент обращался в поликлинику в текущем году;
 - а. Отчет «Справка»
10. Создать расписание, которое можно повесить в холле поликлиники;

5.2. Форма авторизации

При запуске приложения открывается форма авторизации. На форме при вводе логина и пароля и нажатия на кнопку «Войти» происходит аутентификация пользователя, который может авторизоваться как работник регистратуры, участковый врач или главный врач. При нажатии кнопки «Заккрыть» происходит выход из текущей базы данных. Вид формы авторизации представлен на рис. 16.1. Макрос кнопки «Войти» представлен на рис. 16.2.

Рисунок 16.1 – Форма авторизации

При соответствующей авторизации осуществляется переход на другие формы:

- АвторизацияРегистратура – логин: “reg”, пароль: “reg”
- АвторизацияВрач – логин: “doc”, пароль: “doc”
- АвторизацияГлавныйВрач – логин: “admin”, пароль: “admin”

<div> <div>Если</div> <div>[Логин]="admin" And [Пароль]="admin"</div> <div>то</div> </div> <div> <div>ОткрытьФорму</div> <div>(АвторизацияГлавныйВрач; Форма; ; ; Обычное)</div> </div> <div> <div>ЗакрытьОкно</div> <div>(Форма; Авторизация; Подсказка)</div> </div>
<div> <div>Иначе если</div> <div>[Логин]="doc" And [Пароль]="doc"</div> <div>то</div> </div> <div> <div>ОткрытьФорму</div> <div>(АвторизацияВрач; Форма; ; ; Обычное)</div> </div> <div> <div>ЗакрытьОкно</div> <div>(Форма; Авторизация; Подсказка)</div> </div>
<div> <div>Иначе если</div> <div>[Логин]="reg" And [Пароль]="reg"</div> <div>то</div> </div> <div> <div>ОткрытьФорму</div> <div>(АвторизацияРегистратура; Форма; ; ; Обычное)</div> </div> <div> <div>ЗакрытьОкно</div> <div>(Форма; Авторизация; Подсказка)</div> </div>
<div>Иначе</div> <div>Окно сообщения</div> <div>(Неверный логин или пароль; Нет; Отсутствует; Ошибка)</div>
Конец блока "Если"

Рисунок 16.2 – Макрос на нажатие кнопки «Войти»

5.3. Форма «АвторизацияРегистратура»

Интерфейс формы для работников регистратуры представлен на рис. 17 и 18. В форме имеется множество кнопок для перехода на другие формы или отчеты.

Регистратура

Врачи

Пациенты

Расписание работы врачей

Дни приема врача

Больные на лечении у врача

Назначения при заболевании

Врач в кабинете

Пациенты врача за последний месяц

Рисунок 17 – Интерфейс формы «АвторизацияРегистратура», вкладка «Врачи»

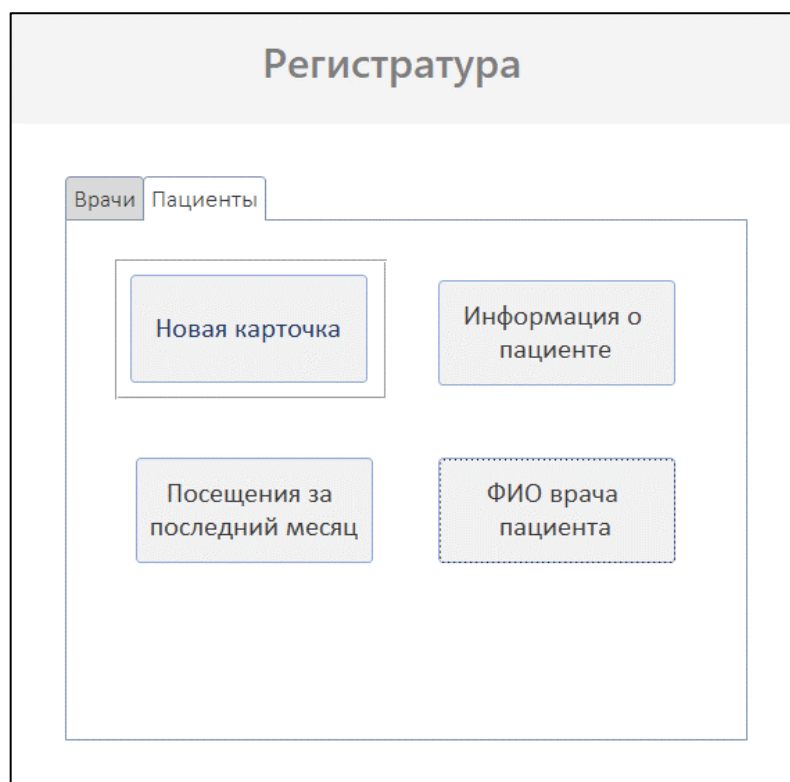


Рисунок 18 – Интерфейс формы «АвторизацияРегистратура», вкладка «Пациенты»

Вкладка «Врачи»:

- При нажатии на кнопку «Расписание работы врачей» открывается отчёт «Расписание_работы»
- При нажатии на кнопку «Дни приема врача» открывается форма «3 Дни приема врача»
- При нажатии на кнопку «Больные на лечении у врача» открывается форма «4 Больные на лечении»
- При нажатии на кнопку «Назначения при заболевании» открывается форма «5 Назначения при заболевании»
- При нажатии на кнопку «Врач в кабинете» открывается форма «6 Врач в кабинете»
- При нажатии на кнопку «Пациенты врача за последний месяц» открывается форма «8 Количество пациентов за прошедший месяц»

Вкладка «Пациенты»:

- При нажатии на кнопку «Новая карточка» открывается форма «Новая_карточка»
- При нажатии на кнопку «Информация о пациенте» открывается форма «1 Информация о больном»
- При нажатии на кнопку «Посещения за последний месяц» открывается форма «7 Посещения за последний месяц»
- При нажатии на кнопку «ФИО врача пациента» открывается форма «2 ФИО лечащего врача больного»

5.4. Форма «АвторизацияГлавныйВрач»

Интерфейс формы для главного врача поликлиники представлен на рис. 19 и 20. В форме имеются кнопки для перехода на другие формы или отчеты.

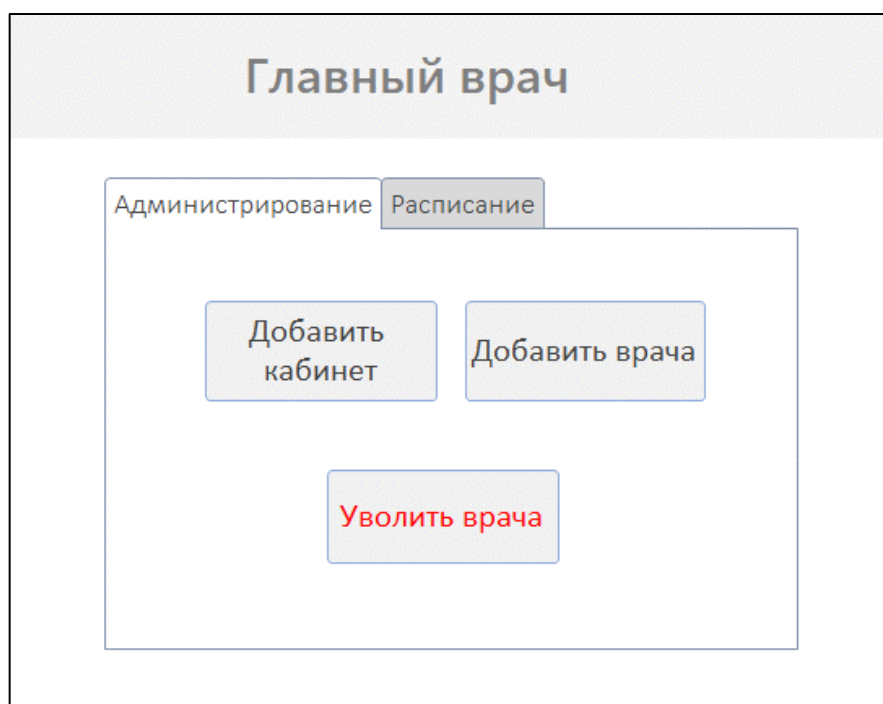


Рисунок 19 – Интерфейс формы «АвторизацияГлавныйВрач», вкладка «Администрирование»

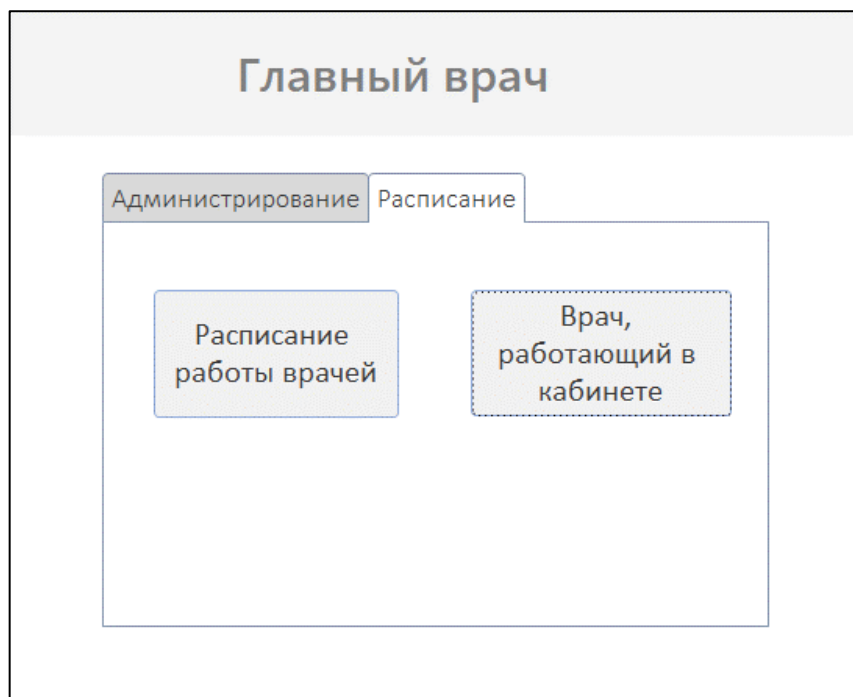


Рисунок 20 – Интерфейс формы «АвторизацияГлавныйВрач», вкладка «Расписание»

Вкладка «Администрирование»:

- При нажатии на кнопку «Добавить кабинет» открывается форма «Добавить_кабинет»
- При нажатии на кнопку «Добавить врача» открывается форма «Добавить_врача»
- При нажатии на кнопку «Уволить врача» открывается форма «Уволить_врача»

Вкладка «Расписание»:

- При нажатии на кнопку «Расписание» открывается отчет «Расписание_работы»
- При нажатии на кнопку «Врач, работающий в кабинете» открывается форма «6 Врач в кабинете»

5.5. Форма «АвторизацияВрач»

Интерфейс формы для участкового врача поликлиники представлен на рис.

21. В форме имеются кнопки для перехода на другие формы или отчеты.

Участковый врач

Расписание работы врачей

Новое посещение

Выдать справку

Номер полиса

Выдать больничный лист

Рисунок 21 – Интерфейс формы «АвторизацияВрач»

- При нажатии на кнопку «Расписание работы врачей» открывается отчет «Расписание_работы»
- При нажатии на кнопку «Новое посещение» открывается форма «Новое_посещение»
- При нажатии на кнопку «Выдать больничный лист» открывается форма «Выдать_больничный_лист»
- Для того, чтобы выдать справку о посещениях поликлиники пациентом за последний год нужно ввести его полис в поле «Номер полиса» и затем при нажатии кнопки «Выдать справку» открывается отчет «Справка» с нужными данными для указанного пациента

5.6. Форма «1 Информация о больном»

Форма «1 Информация о больном» показывает краткую информацию о пациенте поликлиники. На форме можно получить информацию о каждом больном – его адрес, когда он в последний раз посещал поликлинику и какой диагноз ему

поставлен. По выбору фамилии пациента из поля со списком происходит вывод нужных данных. SQL-код запроса для формы был представлен ранее в одноименном запросе.

Источник строк поля со списком «Выбор пациента»:

```
SELECT [1 Адрес, дата и диагноз].[Фамилия_пациента] FROM [1 Адрес, дата и диагноз];
```

Вид формы представлен на рис. 22.

Адрес больного, диагноз и дата последнего посещения

Выбор пациента: Сосновский

Фамилия пациента: Сосновский

Улица и дом: Жукова, 1

Номер квартиры: 77

Дата посещения: 19.12.2020

Диагноз: простуда

Рисунок 22 – Форма «1 Информация о больном»

5.7. Форма «2 ФИО лечащего врача больного»

Форма «2 ФИО лечащего врача больного» представляет собой два поля, в первом указывается фамилия пациента из списка больных поликлиники и во втором соответственно выводятся фамилия и инициалы лечащего врача пациента. SQL-код запроса для формы был представлен ранее в одноименном запросе.

Источник строк поля со списком «Фамилия пациента»:

```
SELECT [2 ФИО лечащего врача].[Фамилия_пациента] FROM [2 ФИО лечащего врача];
```

Вид формы представлен на рис. 23.

**Фамилия и инициалы
лечащего врача больного**

Фамилия пациента

Фамилия и инициалы врача

Рисунок 23 – Форма «2 ФИО лечащего врача больного»

5.8. Форма «3 Дни приёма врача»

В этой форме выводится информация о номере кабинета, днях приёма, смене врача, который работает в данное время в этом кабинете, а также дополнительная информация о самом кабинете, такая как корпус, в котором он находится и этаж. По выбору фамилии врача из поля со списком происходит вывод нужных данных в подчиненной форме «Расписание подчиненная форма1» (поле связи – «ИН врача»). SQL-код запроса для формы был представлен ранее в одноименном запросе. Вид формы представлен на рис. 24.

Номер кабинета и расписание приема врача

Выбор врача

Расписание

Номер кабинета	Дни	Смена	Корпус	Этаж
1321	пятница	утро	1	3
1321	среда	утро	1	3
4132	понедельник	утро	4	1

Рисунок 24 – Форма «3 Дни приёма врача»

Источник строк поля со списком «Выбор врача»:

```
SELECT [Врач].[ИН_врача], [Врач].[Фамилия] FROM Врач;
```

5.9. Форма «Расписание подчиненная форма1»

Данная форма является подчинённой формой для формы «3 Дни приёма врача». В этой форме выводится информация о номере кабинета, днях приёма, смене врача, который работает в данное время в этом кабинете, а также корпус, в котором находится кабинет и этаж. Вид формы представлен на рис. 25.

Номер кабинета	Дни	Смена	Корпус	Этаж
1321	пятница	утро	1	3
1321	среда	утро	1	3
2221	вторник	утро	2	2
2222	четверг	вечер	2	2
4132	понедельник	утро	4	1
4133	понедельник	вечер	4	1

Рисунок 25 – Форма «Расписание подчиненная форма1»

5.10. Форма «4 Больные на лечении»

На форме можно получить ФИО больных, у которых еще не истек срок действия больничного листа. Вид формы представлен на рис. 26.1. По выбору ФИО из поля со списком происходит фильтрация записей по ФИО лечащего врача. Вид формы после выбора врача из поля со списком представлен на рис. 26.2. SQL-код запроса для формы был представлен ранее в одноименном запросе. Источник строк поля со списком «Выбор врача»:

```
SELECT DISTINCT Врач.Фамилия FROM Врач;
```

Больные на лечении у врача

Выбор врача

Фамилия	Имя	Отчество	Срок истечения больничного	Фамилия врача
Сосновский	Дмитрий	Николаевич	30.12.2020	Сергеев
Киреев	Константин	Александрович	31.12.2020	Иванов
Муковский	Даниил	Владимирович	30.01.2021	Сергеев

Рисунок 26.1 – Форма «4 Больные на лечении»

Больные на лечении у врача

Выбор врача

Фамилия	Имя	Отчество	Срок истечения больничного	Фамилия врача
Сосновский	Дмитрий	Николаевич	30.12.2020	Сергеев
Муковский	Даниил	Владимирович	30.01.2021	Сергеев

Рисунок 26.2 – Форма «4 Больные на лечении» после выбора врача

5.11. Форма «5 Назначения при заболевании»

В данной форме выводятся диагнозы, которые ставили врачи, и соответствующие им назначения, а также фамилия врача, который поставил этот диагноз. Вид формы представлен на рис. 27.1. По выбору диагноза из поля со списком происходит фильтрация записей по диагнозу. Вид формы после выбора диагноза из поля со списком представлен на рис. 27.2. SQL-код запроса для формы был представлен ранее в одноименном запросе.

Источник строк поля со списком «Выбор диагноза»:

```
SELECT DISTINCT Посещение_поликлиники.Диагноз FROM Посещение_поликлиники;
```

Назначения при заболевании		
Выбор диагноза	<input type="text" value=""/>	
Диагноз	Назначения	Фамилия врача
простуда	ринза	Иванов
рак головы	арбидол	Степанов
рак легких	отдых	Сергеев
рак кишечника	отдых	Сергеев
простуда	чай	Сергеев

Рисунок 27.1 – Форма «5 Назначения при заболевании»

Назначения при заболевании		
Выбор диагноза	<input type="text" value="простуда"/>	
Диагноз	Назначения	Фамилия врача
простуда	ринза	Иванов
простуда	чай	Сергеев

Рисунок 27.2 – Форма «5 Назначения при заболевании» после выбора диагноза

5.12. Форма «6 Врач в кабинете»

В данной форме отображаются фамилия, имя, отчество и ИН врача для заданного кабинета и времени. Необходимо выбрать из полей со списком номер кабинета, день приёма и смену и нажать кнопку со знаком фильтрации. После этого происходит фильтрация записей по нужным полям. Вид формы представлен на рис. 28.1. Вид формы после выбора нужных атрибутов из полей со списком представлен на рис. 28.2. Также на форме имеется дополнительная кнопка для

сброса всех фильтров, которая возвращает записи в исходное состояние. SQL-код запроса для формы был представлен ранее в одноименном запросе.

Источник строк поля со списком «Поиск1»:

```
SELECT DISTINCT Кабинет.ИН_кабинета FROM Кабинет ORDER BY Кабинет.ИН_кабинета;
```

Источник строк поля со списком «Поиск2»:

```
SELECT DISTINCT Расписание.Дни FROM Расписание ORDER BY Расписание.Дни;
```

Источник строк поля со списком «Поиск3»:



```
SELECT DISTINCT Расписание.Смена FROM Расписание ORDER BY Расписание.Смена;
```

Врач в кабинете

Кабинет

День

Смена

ИН врача	Фамилия	Имя	Отчество
100101	Иванов	Иван	Иванович
100101	Иванов	Иван	Иванович
100103	Сергеев	Сергей	Максимович
100103	Сергеев	Сергей	Максимович
100103	Сергеев	Сергей	Максимович
100102	Степанов	Илья	Владимирович

Рисунок 28.1 – Форма «6 Врач в кабинете»

ИН врача	Фамилия	Имя	Отчество
100103	Сергеев	Сергей	Максимович

Рисунок 28.2 – Форма «6 Врач в кабинете» после фильтрации

Внедренный макрос, который активируется при нажатии кнопки фильтрации представлен на рис. 28.3.

ПрименитьФильтр Имя фильтра Условие отбора = [ИН_кабинета]=[Формы]![6 Врач в кабинете]![Поиск1] And [Дни]=[Формы]![6 Врач в кабинете]![Поиск2] And [Смена]=[Формы]![6 Врач в кабинете]![Поиск3]

Рисунок 28.3 – Макрос на кнопке фильтрации

5.13. Форма «7 Посещения за последний месяц»

В данной форме выводится количество посещений поликлиники за последний месяц и ФИО больного. Выводится информация только о тех больных, которые посещали поликлинику за последний месяц хотя бы один раз. По выбору фамилии пациента из поля со списком происходит вывод нужных данных. SQL-код запроса для формы был представлен ранее в одноименном запросе. Вид формы представлен на рис. 29.

Источник строк поля со списком «Выбор пациента»:

```
SELECT [7 Посещения за последний месяц].[Фамилия_пациента] FROM [7 Посещения за последний месяц];
```

Рисунок 29 – Форма «7 Посещения за последний месяц»

5.14. Форма «8 Количество пациентов за прошедший месяц»

На форме выводится ФИО врача с указанием количества пациентов, обслуженных им за последний месяц. По выбору фамилии врача из поля со списком происходит вывод нужных данных. SQL-код запроса для формы был представлен ранее в одноименном запросе. Вид формы представлен на рис. 30.

Источник строк поля со списком «Выбор врача»:

```
SELECT [8 Количество пациентов за прошедший месяц].[Фамилия] FROM [8 Количество
пациентов за прошедший месяц];
```

Рисунок 30 – Форма «8 Количество пациентов за прошедший месяц»

5.15. Форма «Добавить_кабинет»

С помощью данной формы главный врач может добавить новый кабинет в базу поликлиники. Сначала заполняются поля номера, корпуса и этажа кабинета и далее по кнопке «Добавить запись» выполняется запрос «Новый кабинет». SQL-код запроса для формы был представлен ранее. Вид формы представлен на рис. 31.

Новый кабинет

Номер

Корпус

Этаж

Добавить запись

Рисунок 31 – Форма «Добавить_кабинет»

5.16. Форма «Добавить_врача»

С помощью данной формы главный врач может добавить нового врача в базу поликлиники. Сначала заполняются поля фамилии, имя и отчества врача, затем выбирается категория врача из списка: высшая, первая или вторая, также заполняются поля стаж и дата рождения. Далее по кнопке «Добавить» выполняется запрос «Новый врач». SQL-код запроса для формы был представлен ранее. Вид формы представлен на рис. 32.

Источник строк поля со списком «Категория»:

"высшая"; "первая"; "вторая"

Рисунок 32 – Форма «Добавить_врача»

5.17. Форма «Новое посещение»

Для того, чтобы создать новую запись о посещении пациента в карточке, врачу необходимо заполнить поля Номер карточки, Жалобы, Диагноз, Назначение, Номер врача и нажать кнопку «Добавить». Дата заполнения устанавливается сегодняшним днем. Номер врача заполняется вручную, так как есть возможность того, что пациент посетил поликлинику, когда его участковый врач не работает. Если пациент пришел в правильное время, то с помощью подчиненной формы «Подчиненная_адрес_врач» (поле связи – «Номер карточки») можно узнать номер лечащего врача пациента и вписать его, при приеме пациента другим врачом он выбирается из списка. По клику на кнопку «Добавить» происходит выполнение запроса «Новое посещение».

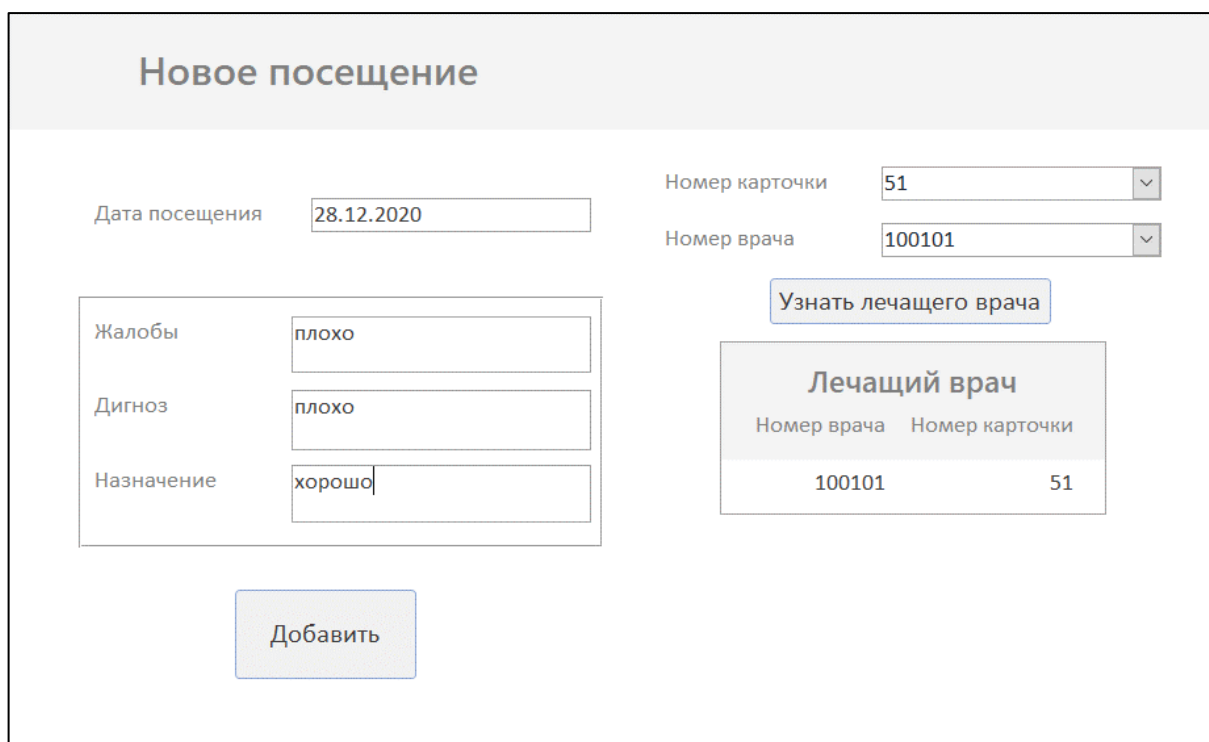
Источник строк поля со списком «Номер карточки»:

```
SELECT DISTINCT Карточка.Номер_карточки FROM Карточка;
```

Источник строк поля со списком «Номер врача»:

```
SELECT DISTINCT Врач.ИН_врача FROM Врач ORDER BY Врач.ИН_врача;
```

SQL-код запроса для формы был представлен ранее. Вид формы представлен на рис. 33.

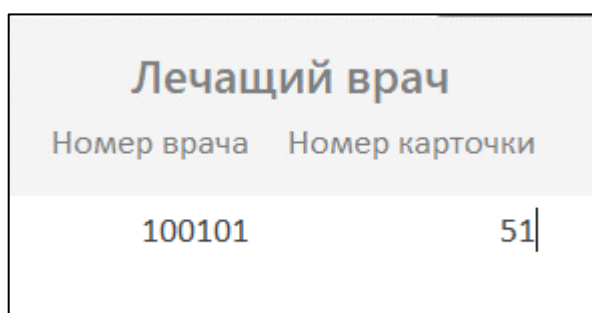


Новое посещение	
Дата посещения	28.12.2020
Номер карточки	51
Номер врача	100101
Узнать лечащего врача	
Лечащий врач	
Номер врача	Номер карточки
100101	51
Жалобы	плохо
Дигноз	плохо
Назначение	хорошо
Добавить	

Рисунок 33 – Форма «Новое_посещение»

5.18. Форма «Подчиненная_адрес_врач»

Данная форма является подчиненной для формы «Новая_посещение». На данной форме отображается номер участкового врача и номер карточки пациента. Вид формы представлен на рис. 34.



Лечащий врач	
Номер врача	Номер карточки
100101	51

Рисунок 34 – Форма «Подчиненная_адрес_врач»

5.19. Форма «Новая_карточка»

На форме необходимо ввести данные нового пациента и на него будет заведена карта по кнопке «Добавить». Нужно заполнить поля ФИО, Номер паспорта, Номер полиса, Пол, Дата рождения, Улица и дом, Квартира, Участок. Дата заполнения ставится сегодняшняя. По клику на кнопку «Принять» выполняется запрос на добавление «Новая_карточка». SQL-код запроса для формы был представлен ранее. Вид формы представлен на рис. 35.

Поле со списком «Номер участка» имеет следующий источник строк:

```
SELECT DISTINCT Участок.Номер_участка FROM Участок ORDER BY Участок.Номер_участка;
```

Поле со списком «Пол» имеет следующий источник строк:

"М"; "Ж"

Новая карточка

Фамилия	<input type="text"/>	Улица и дом	<input type="text"/>
Имя	<input type="text"/>	Квартира	<input type="text"/>
Отчество	<input type="text"/>	Номер участка	<input type="text"/>
Пол	<input type="text" value="М"/>	Номер полиса	<input type="text"/>
Дата рождения	<input type="text"/>	Номер паспорта	<input type="text"/>
		Дата заполнения	<input type="text" value="28.12.2020"/>

Добавить

Рисунок 35 – Форма «Новая_карточка»

5.20. Форма «Уволить_врача»

В списке «Кого уволить» нужно выбрать ИН врача, информация о котором будет удаляться из БД. В списке «Кому передать» нужно выбрать ИН врача, который будет в дальнейшем обслуживать участки удаляемого врача и его пациентов. По кнопке запускается макрос «Увольнение», который в свою очередь запускает по очереди четыре запроса: «Обновить_посещение», «Обновить_участок», «Удалить_врача», «Удалить_кабинет». Уволить врача может только Главный врач. SQL-код запросов для формы был представлен ранее. Вид формы представлен на рис. 36. Макрос «Увольнение» представлен на рис. 37.

Поля со списком «Кого уволить» и «Кому передать» имеет следующий источник строк:

```
SELECT DISTINCT Врач.ИН_врача FROM Врач ORDER BY Врач.ИН_врача;
```

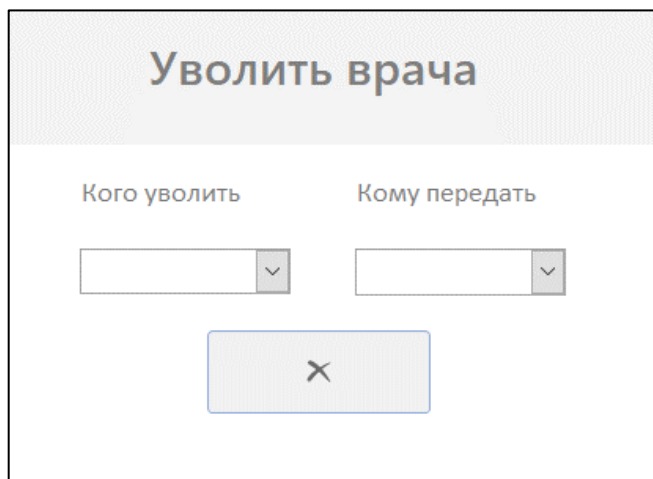


Рисунок 36 – Форма «Уволить_врача»

ЗаккрытьОкно	(Запрос; Обновить_посещение; Подсказка)
ОткрытьЗапрос	(Обновить_посещение; Таблица; Изменение)
ЗаккрытьОкно	(Запрос; Обновить_участок; Подсказка)
ОткрытьЗапрос	(Обновить_участок; Таблица; Изменение)
ЗаккрытьОкно	(Запрос; Удалить_кабинет; Подсказка)
ОткрытьЗапрос	(Удалить_кабинет; Таблица; Изменение)
ЗаккрытьОкно	(Запрос; Удалить_врача; Подсказка)
ОткрытьЗапрос	(Удалить_врача; Таблица; Изменение)
ОткрытьФорму	(Уволить_врача; Форма; ; ; Обычное)

Рисунок 37 – Макрос «Увольнение»

5.21. Форма «Выдать_больничный_лист»

С помощью данной формы врач может выдать больничный лист пациенту. Сначала заполняются поля номера талона и срока больничного и далее по кнопке «Выписать» выполняются запросы «Добавить_действие_с_больничным» и «Добавить_больничный_лист». SQL-код запросов для формы был представлен ранее. Срок больничного первоначально устанавливается в значение даты посещения плюс тридцать дней, но его можно изменить. Вид формы представлен на рис. 38.

Источник строк поля со списком «Номер талона»:

```
SELECT DISTINCT Посещение_поликлиники.Номер_талона FROM Посещение_поликлиники  
WHERE (((Посещение_поликлиники.Дата_посещения)=Date())) ORDER BY  
Посещение_поликлиники.Номер_талона;
```

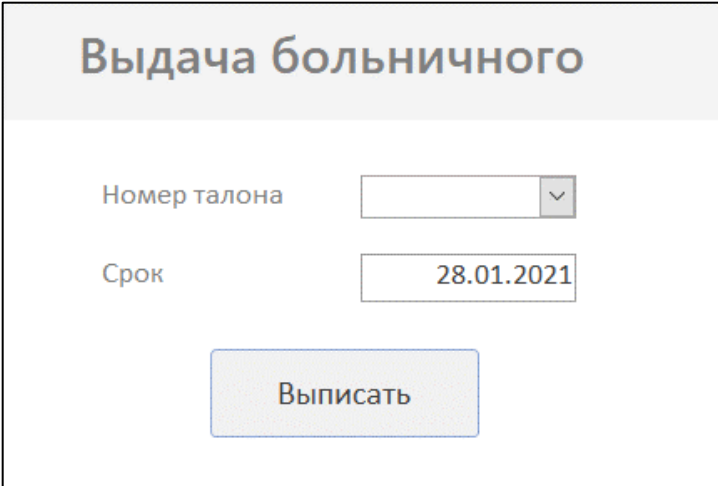


Рисунок 38 – Форма «Выдать_больничный_лист»

5.22. Отчет «Расписание_работы»

Выводит информацию о врачах, участках, на которых они работают, кабинет, дни приёма и смены. Расписание можно повесить в холле.

SQL-код запроса данных для отчета представлен ниже:

```
SELECT Участок.Номер_участка, Врач.Фамилия, Врач.Имя, Врач.Отчество, расписа-  
ние.ИН_кабинета, Расписание.Дни, Расписание.Смена
```

```
FROM Кабинет INNER JOIN ((Врач INNER JOIN Расписание ON Врач.ИН_врача = Расписа-
ние.ИН_врача) INNER JOIN Участок ON Врач.ИН_врача = Участок.ИН_врача) ON Каби-
нет.ИН_кабинета = Расписание.ИН_кабинета
ORDER BY Участок.Номер_участка, Врач.Фамилия, Расписание.ИН_кабинета, Расписа-
ние.Дни, Расписание.Смена DESC;
```

Данный отчет имеет следующую структуру:

- Номер участка, Фамилия, Имя, Отчество
 - Номер кабинета
 - Дни приема
 - Смена

Вид отчета представлен на рис. 39.

Расписание работы врачей						
Номер участка	Фамилия	Имя	Отчество	Номер кабинета	Дни приема	Смена
1	Иванов	Иван	Иванович			
				2221	вторник	утро
				2222	четверг	вечер
2	Сергеев	Сергей	Максимович			
				1321	пятница	утро
				1321	среда	утро
				4132	понедельник	утро
3	Степанов	Илья	Владимирович			
				4133	понедельник	вечер
				Дата	28 декабря 2020 г.	

Рисунок 39 – Отчет о расписании работы врачей

5.23. Отчет «Справка»

Выводит информацию о посещениях поликлиники за последний год с указанием даты посещения и диагноза в виде справки установленного образца.

SQL-код запроса данных для справки представлен ниже:

```
SELECT Карточка.Фамилия_пациента, Карточка.Имя_пациента, Карточка.Отчество_паци-
ента, Карточка.Дата_рождения, Карточка.Номер_полиса, Посещение_поликли-
ники.Дата_посещения, Посещение_поликлиники.Диагноз
FROM Карточка INNER JOIN Посещение_поликлиники ON Карточка.Номер_карточки = По-
сещение_поликлиники.Номер_карточки
WHERE (((Карточка.Номер_полиса)=[Формы]![АвторизацияВрач]![Полис]));
```

Вид справки представлен на рис. 40.

Справка №__			
Ф.И.О.	Муковский Д.В.		
Дата рождения	12.12.1900	Номер полиса	900102
Выдана	_____		
в том, что он(а) обращался(ась) в поликлинику в 2020 году			
в указанную дату по указанному поводу			
Дата посещения	13.12.2020		
Диагноз	рак легких		

Дата посещения	18.12.2020		
Диагноз	рак кишечника		

Дата	28 декабря 2020 г.		Врач _____

Рисунок 40 – Справка установленного образца

Выводы.

В результате выполнения лабораторной работы были получены навыки проектирования баз данных, построения ER диаграмм, реляционной модели, реализации пользовательского интерфейса в MS Access в соответствии с задачами, которые должны быть реализуемы пользователем. В результате была разработана база данных и программа к ней, которые позволяют администрировать работу поликлиники, просматривать в различных форматах данные о врачах и пациентах, изменять их, формировать отчет и справки.