Министерство Образования и Науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего профессионального образования

**“Санкт-Петербургский государственный электротехнический университет “ЛЭТИ” им. В.И. Ульянова (Ленина)”**

**(СПбГЭТУ)**

кафедра МОЭВМ

**Пояснительная записка**

**к курсовой работе**

**по дисциплине «Базы Данных»**

**тема: «Разработка и реализация базы данных»**

Вариант 16.

Выполнил: студент группы 1304

Давыдово А.В.

Проверила: Фомичева Т.Г.

г. Санкт-Петербург

2013г.

1. **Формулировка задания**

## Задание 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?

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.

1. **Проектирование БД методом «сущность-связь»**

**ER-диаграмма**

Address

Area

Улица, дом

Номер участка

Doctor

Sick

Номер паспорта

Card

Номер паспорта

Cabinet

Номер, День

Sick-list

Номер карточки

Номер больничного

Сущность «Doctor» содержит ключ «Номер паспорта»

Сущность «Area» содержит ключ «Номер участка»

Сущность «Address» содержит ключ «Улица, дом»

Сущность «Cabinet» содержит составной ключ из атрибутов «Номер», «День»

Сущность «Sick-list» содержит ключ «Номер больничного»

Сущность «Card» содержит ключ «Номер карточки»

Сущность «Sick» содержит ключ «Номер паспорта»

Отношение «Doctor» - «Cabinet» имеет связь 1:n. В кабинете в некоторое время одновременно может находиться только один врач. Врач может работать в нескольких кабинетах. Связь обязательна со всех сторон, потому что врач должен находиться хотя бы в одном кабинете, а пустых кабинетов не может существовать. По правилу генерации отношений получаем два отношения:

Doctor (Номер паспорта)

Cabinet (Номер, День, Номер паспорта)

Отношение «Doctor» - «Area» имеет связь 1:n. Обслуживать участок может только один врач, но в тоже время врач может обслуживать несколько участков. Связь обязательна со обеих сторон, потому что врач не может работать без хотя бы одного участка, а участок не может пустовать. По правилу генерации отношений получаем два отношения:

Doctor (Номер паспорта)

Area (Номер участка, Номер паспорта)

Отношение «Area» - «Address» имеет связь 1:n. Участок может содержать множество адресов, а адрес может входить только в один участок. Связь обязательна с обеих сторон, так как участок не может быть бесхозным, врач должен обслуживать хотя бы один участок. По правилу генерации отношений получаем два отношения:

Area (Номер участка)

Address (Улица, дом, Номер участка)

Отношение «Address» - «Sick» имеет связь 1:n. По одному адресу может проживать множество потенциальных пациентов, больной живет по одному адресу. Связь обязательна только со стороны больного, потому что больной обязан иметь место жительства, а по некоторому адресу могут не проживать больные. По правилу генерации отношений получаем два отношения:

Address (Улица, дом)

Sick (Номер паспорта, Улица, дом)

Отношение «Sick» - «Card» имеет связь 1:1. Карточка заводится только на одного больного, каждый больной имеет в регистратуре только одну карточку. Связь обязательна со всех сторон, так как больной обязательно должен иметь карточку в регистратуре, а карточка заводится на конкретного больного. По правилу генерации отношений получаем одно отношение:

Card (Номер карточки, номер паспорта)

Связь между сущностями «Doctor», «Sick-list», «Card» называется 3-сторонней. Врач может обслуживать любое количество больных, в результате он может выписывать больничные листы. Больничный лист может быть выписан в результате нескольких посещений больным врача. Связь необязательна со стороны больничного листа, так как больничный лист может быть и не выписан в результате посещения. По правилу генерации отношений получим четыре отношения:

Card (Номер карточки)

Sick-list (Номер больничного)

Doctor (Номер паспорта)

Records (Номер карточки, Номер паспорта врача, Номер больничного)

В итоге получим следующие отношения:

Card (Номер карточки, Номер паспорта, Улица,дом)

Sick-list (Номер больничного)

Doctor (Номер паспорта)

Records (Номер карточки, номер паспорта врача, номер больничного)

Address (Улица, дом, Номер участка)

Area (Номер участка, номер паспорта врача)

Cabinet (Номер кабинета, день приёма, номер паспорта врача)

Дополним отношения оставшимися атрибутами:

Card (Номер карточки, Номер паспорта, ФИО, Улица, дом, пол, Дата рождения, Номер полиса)

Sick-list (Номер больничного, Дата создания, Срок истечения действия)

Doctor (Номер паспорта, ФИО, Стаж, Категория, Дата рождения)

Records (Номер карточки, номер паспорта врача, номер больничного, Дата посещения, Жалобы, Диагноз, Назначение)

Address (Улица, дом, Номер участка)

Area (Номер участка, номер паспорта врача)

Cabinet (Номер кабинета, день приёма, номер паспорта врача)

В отношении Card (Номер карточки, Номер паспорта, ФИО, Улица, дом, пол, Дата рождения, Номер полиса) функциональные зависимости выглядят следующим образом:

Номер карточки - Номер паспорта, ФИО, Улица, дом, пол, Дата рождения, Номер полиса

Номер паспорта - Номер карточки, ФИО, Улица, дом, пол, Дата рождения, Номер полиса

Номер карточки - Номер паспорта, ФИО, Улица, дом, пол, Дата рождения, Номер полиса

ФИО, Улица, дом – Номер карточки, Номер паспорта, пол, Дата рождения, Номер полиса

Номер полиса – Номер карточки, номер паспорта, ФИО, Улица, дом, пол, Дата рождения

Других зависимостей нет, а это значит, что отношение находится в НФБК.

В отношении Sick-list (Номер больничного, Дата создания, Срок истечения действия) функциональные зависимости выглядят следующим образов

Номер больничного – Дата создания, Срок истечения действия

Других зависимостей нет, а это значит, что отношение находится в НФБК.

В отношении Doctor (Номер паспорта, ФИО, Стаж, Категория, Дата рождения) функциональные зависимости выглядят следующим образом:

Номер паспорта - ФИО, Стаж, Категория, Дата рождения

Других зависимостей нет, а это значит, что отношение находится в НФБК.

В отношении Address (Улица, дом, Номер участка) функциональные зависимости выглядят следующим образом:

Улица, дом - Номер участка

Других зависимостей нет, а это значит, что отношение находится в НФБК.

В отношении Area (Номер участка, номер паспорта врача) функциональные зависимости выглядят следующим образом:

Номер участка - номер паспорта врача

Других зависимостей нет, а это значит, что отношение находится в НФБК.

В отношении Cabinet (Номер кабинета, день приёма, номер паспорта врача) функциональные зависимости выглядят следующим образом:

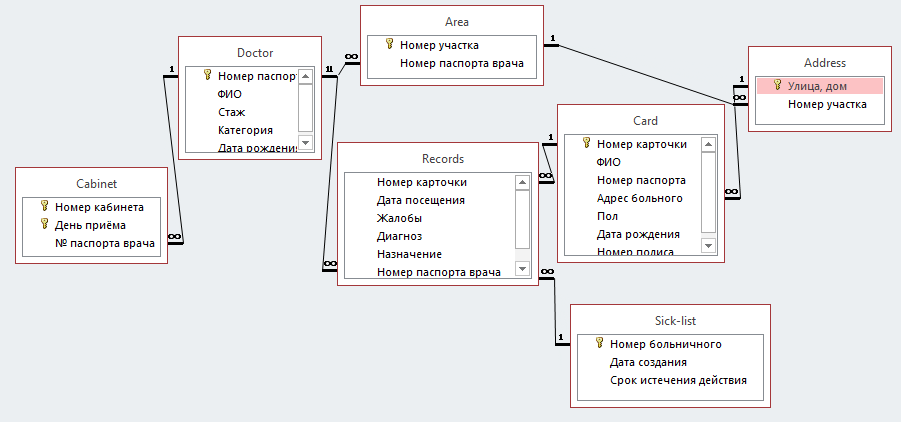
Номер кабинета, день приёма - номер паспорта врача

Других зависимостей нет, а это значит, что отношение находится в НФБК.

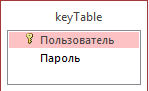
Этап проектирования можно считать завершенным.

1. **Структуры таблиц и постоянные связи между таблицами**

В результате этапа проектирования получим окончательную структуру таблиц и связей между ними:

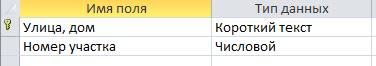


По заданию требуется разбиение пользователей на группы: главный врач, регистратура, врач. Поэтому в окончательную структуру данных необходимо добавить таблицу «keyTable» с именами и паролями этих пользователей. Также добавим пользователя-администратора с абсолютными правами на доступ.



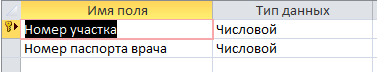
**Структура таблиц:**

1. Таблица «Address»



Правило проверки поля «Улица, дом» - «Like "\* , \*"»

1. Таблица «Area»

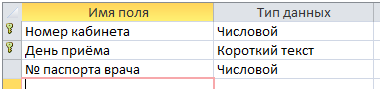


Размер поля «Номер участка» - Длинное целое

Размер поля «Номер паспорта врача» - Длинное целое

Число десятичных знаков поля «Номер паспорта врача» - 10

1. Таблица «Cabinet»

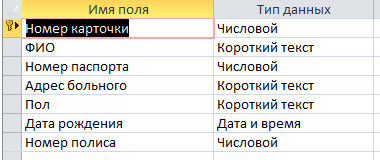


Размер поля «День приёма» - 2

Правило проверки поля «День приёма» - «="пн" Or ="вт" Or ="ср" Or ="чт" Or ="пт" Or ="сб" Or ="вс"»

Число десятичных знаков поля «№ паспорта врача» - 10

1. Таблица «Card»



Правило проверки поля «ФИО» - «Like "\* ?.?."»

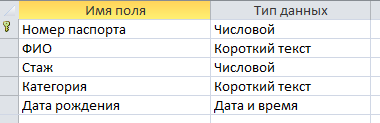
Число десятичных знаков поля «Номер паспорта» - 10

Правило проверки поля «Адрес больного» - «Like "\* , \*"»

Размер поля «Пол» - 1

Правило проверки поля «Пол» - «Like "\* , \*"»

1. Таблица «Doctor»

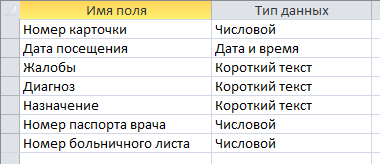


Размер поля «Номер паспорта» - 10

Правило проверки поля «ФИО» - «Like "\* ?.?."»

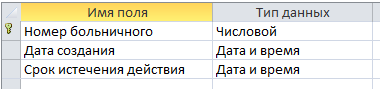
Правило проверки поля «Категория» - «="вторая" Or ="первая" Or ="высшая"»

1. Таблица «Records»

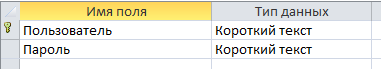


Размер поля «Номер паспорта врача» - 10

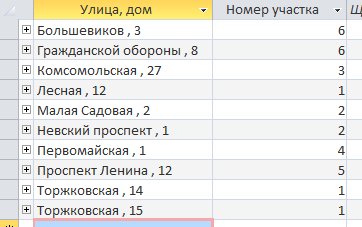
1. Таблица «Sick-list»



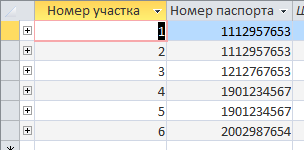
1. Таблица «keyTable»



1. **Содержимое таблиц, на котором проводилась отладка системы**
2. Содержимое таблицы «Address»



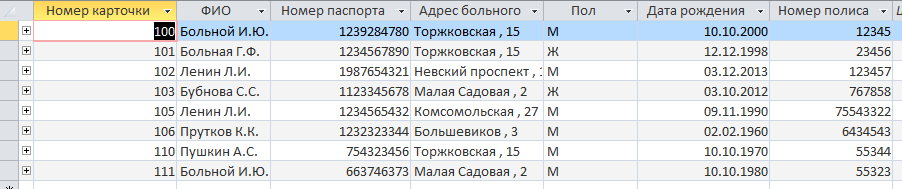
1. Содержимое таблицы «Area»



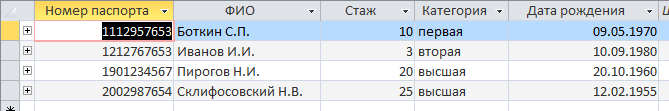
1. Содержимое таблицы «Cabinet»



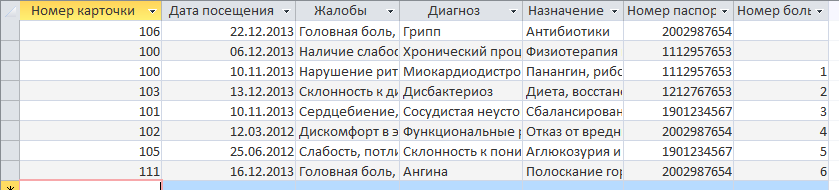
1. Содержимое таблицы «Card»



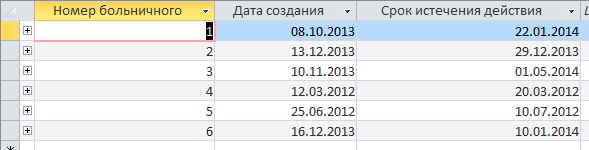
1. Содержимое таблицы «Doctor»



1. Содержимое таблицы «Records»



1. Содержимое таблицы «Sick-list»



1. **Схема иерархии интерфейса**

Вход

Форма «ВходВрач»

Форма «ВходГлавныйВрач»

Форма «New\_rec»

Форма «sick-list»

Форма «Регистратура»

Форма «Номер кабинета и дни приёма врача»

Форма «Врач, работающий в кабинете»

Форма «Больные на лечении у врача»

Форма «ВходAdmin»

Форма «Информация о больном»

Форма «Назначение при указанном заболевании»

Форма «New\_card»

Отчет «Расписание работы врачей»

Отчет «Справка»

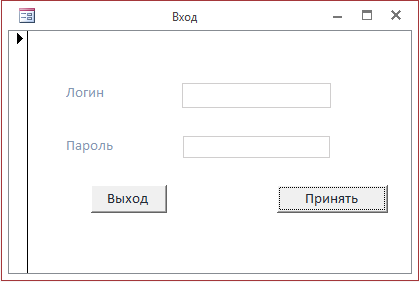
Форма «Информация о количестве пациентов за последний месяц»

Форма «Посещение больницы больным за последний месяц

Форма «ФИО лечащего врача больного»

Форма «Уволить врача»

1. **Описание экранных форм**
2. Форма «Вход»



На этой форме по клику на «Принять» происходить аутентификация пользователя по введенному логину и паролю.

По нажатию на «Выход» происходит выход из текущей базы данных.

Код, соответствующий кнопке «Принять»:

*Private Sub Кнопка0\_Click()*

*Dim login, password, N As String*

*login = Me.логин*

*password = Me.пароль*

*If IsNull(login) Then*

*MsgBox "Отсутствует логин"*

*Exit Sub*

*End If*

*If IsNull(password) Then*

*MsgBox "Отсутствует пароль"*

*Exit Sub*

*End If*

*If CurrentProject.Connection.Execute("SELECT Пользователь FROM keyTable WHERE ( Пользователь = '" & login & "' AND Пароль = '" & password & "')").RecordCount = 0 Then*

*MsgBox "Пользователь не найден"*

*Exit Sub*

*End If*

*N = CurrentProject.Connection.Execute("SELECT Пользователь FROM keyTable WHERE ( Пользователь = '" & login & "' AND Пароль = '" & password & "')").Fields(0)*

*If N = "Admin" Then*

*DoCmd.OpenForm "ВходAdmin"*

*End If*

*If N = "Врач" Then*

*DoCmd.OpenForm "ВходВрач"*

*End If*

*If N = "Главный врач" Then*

*DoCmd.OpenForm "ВходГлавныйВрач"*

*End If*

*If N = "Регистратура" Then*

*DoCmd.OpenForm "ВходРегистратура"*

*End If*

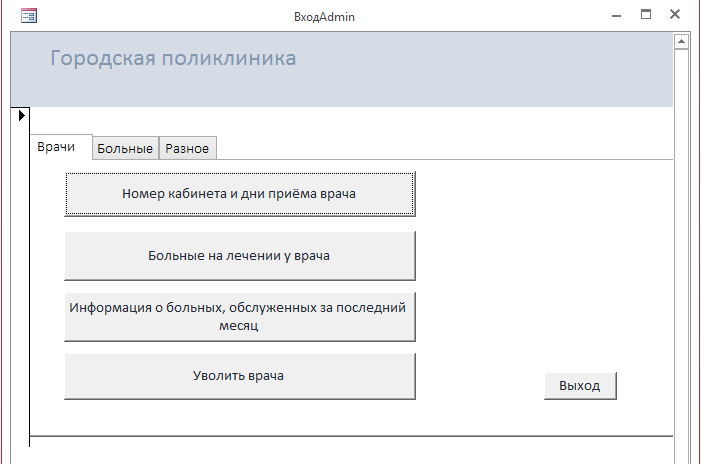
*DoCmd.Close acForm, Me.Form.Name*

*End Sub*

По кнопке Выход вызывается макрос «exit»

1. Форма «ВходAdmin»

На этой форме находятся 3 вкладки. В первой можно узнать информацию по врачам, на второй – по больным, на третьей – остальную информацию по заданию



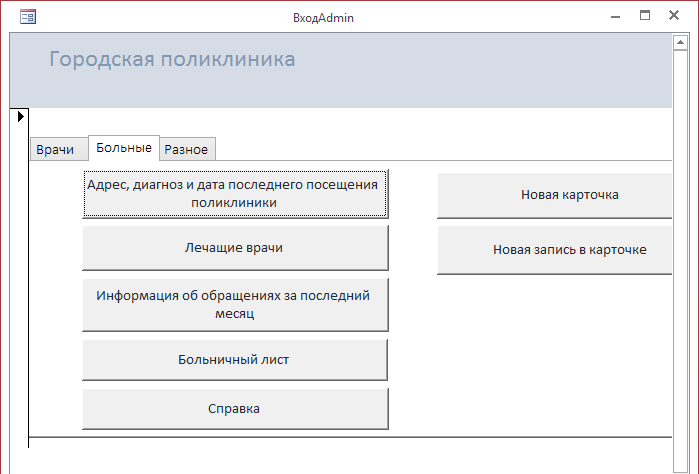
По кнопке «Выход» осуществляется закрытие БД.

Действие кнопка «Номер кабинета и дни приёма врача» осуществляет открытие формы «Номер кабинета и дни приёма врача»

По клику на кнопке «Больные на лечении у врача» запускается форма «Больные на лечении у врача»

По кнопке «Информация о больных, обслуженных за последний месяц» осуществляется открытие формы «Информация о количестве пациентов за последний месяц»

По кнопке «Уволить врача» открывается форма «Уволить врача»



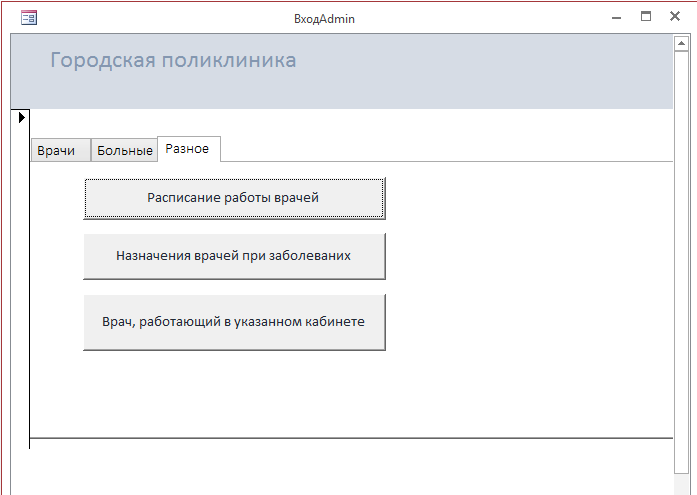
По кнопке «Новая карточка» открывается форма «New\_card», по кнопке «Новая запись в карточке» - форма «New\_rec», по кнопке «Больничный лист» - форма «sick-list».

По клику на кнопку «Справка» открывается отчет «Справка».

По нажатию на кнопку «Адрес, диагноз и дата последнего посещения поликлиники» осуществляется открытие формы «Информация о больном»

По кнопке «Лечащие врачи» открывается форма «ФИО лечащего врача»

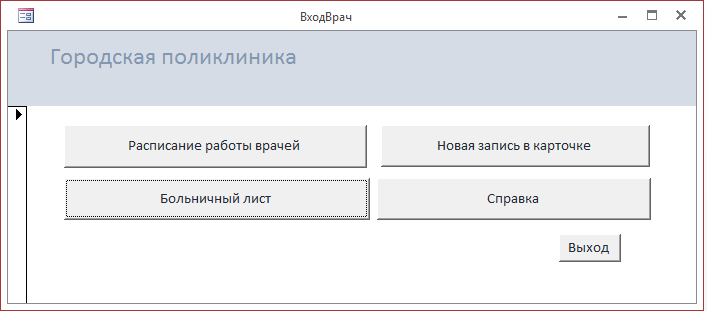
По кнопке «Информация об обращениях за последний месяц» открывается форма «Информация об обращениях за последний месяц»



По клику на кнопке «Расписание работы врачей» открывается отчет «Расписание работы врачей».

По клику на «Назначения врачей при заболеваниях» открывается форма «Назначение при указанном заболевании», на «Врач, работающий в указанном кабинете» - «Врач, работающий в кабинете».

1. Форма «ВходВрач»

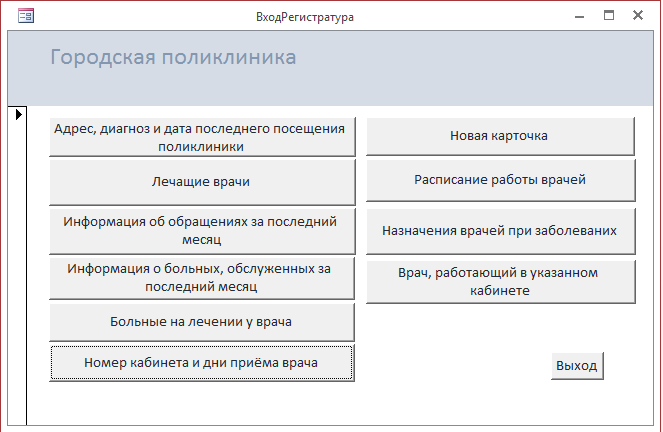


По клику на кнопке «Расписание работы врачей» открывается отчет «Расписание работы врачей».

По кнопке «Новая запись в карточке» - форма «New\_rec», по кнопке «Больничный лист» - форма «sick-list».

По клику на кнопку «Справка» открывается отчет «Справка».

1. Форма «ВходРегистратура»



По кнопке «Новая карточка» открывается форма «New\_card».

По нажатию на кнопку «Адрес, диагноз и дата последнего посещения поликлиники» осуществляется открытие формы «Информация о больном»

По кнопке «Лечащие врачи» открывается форма «ФИО лечащего врача»

По кнопке «Информация об обращениях за последний месяц» открывается форма «Информация об обращениях за последний месяц»

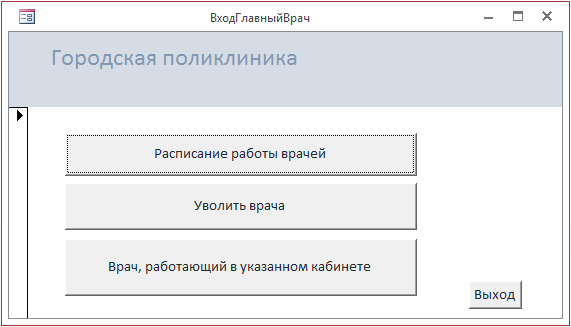
Действие кнопка «Номер кабинета и дни приёма врача» осуществляет открытие формы «Номер кабинета и дни приёма врача»

По клику на кнопке «Больные на лечении у врача» запускается форма «Больные на лечении у врача»

По клику на кнопке «Расписание работы врачей» открывается отчет «Расписание работы врачей».

По клику на «Назначения врачей при заболеваниях» открывается форма «Назначение при указанном заболевании», на «Врач, работающий в указанном кабинете» - «Врач, работающий в кабинете».

1. Форма «ВходГлавныйВрач»

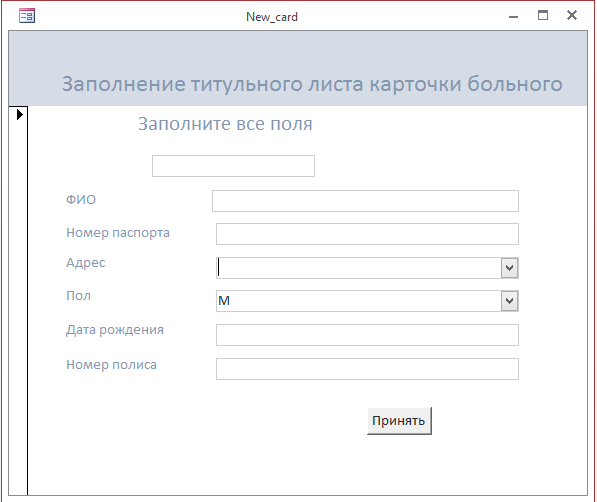
****

По клику на кнопке «Расписание работы врачей» открывается отчет «Расписание работы врачей».

По кнопке «Уволить врача» открывается форма «Уволить врача»

По клику на кнопке «Врач, работающий в указанном кабинете» - «Врач, работающий в кабинете».

1. Форма «New\_card»



Все поля свободные.

Поле со списком «Адрес» имеет следующий источник строк:

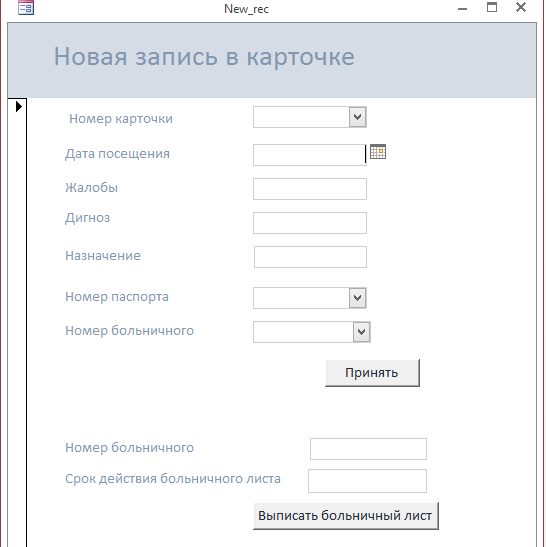
*SELECT Address.[Улица, дом], Address.[Номер участка] FROM Address;*

Поле со списком «Пол» имеет следующий источник строк:

*"М";"Ж"*

На форме необходимо ввести данные нового пациента и на него будет заведена карта по кнопке «Принять». Нужно заполнить поля ФИО, Номер паспорта, Адрес, Пол, Дата рождения, Номер полиса и в верхнее поле ввести номер карты больного. По клику на кнопке «Принять» выполняется запрос на добавление «new\_card».

1. Форма «New\_rec»



Все поля свободные.

Источник строк поля со списком «Номер карточки»:

*SELECT [Card].[Номер карточки], [Card].[ФИО], [Card].[Дата рождения] FROM Card;*

Источник строк поля со списком «Номер паспорта»:

*SELECT [Doctor].[Номер паспорта], [Doctor].[ФИО] FROM Doctor;*

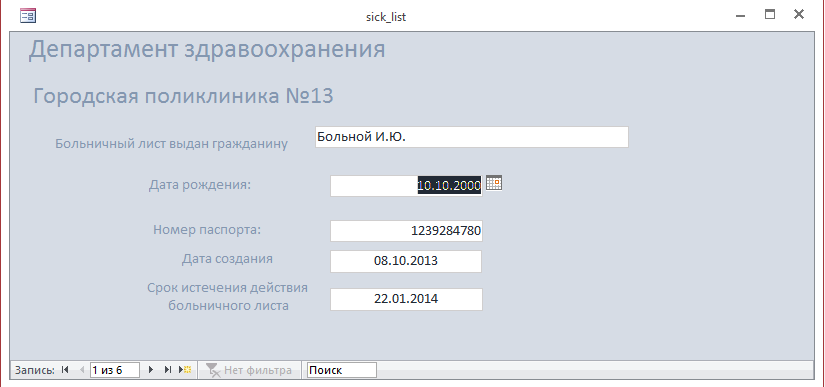
Источник строк поля со списком «Номер больничного»:

*SELECT [Sick-list].[Номер больничного], [Sick-list].[Дата создания], [Sick-list].[Срок истечения действия] FROM [Sick-list];*

Для того, чтобы создать новую запись о больном, врачу необходимо заполнить поля Номер карточки, Дата посещения, Жалобы, Диагноз, Назначение, Номер паспорта, необязательное поле Номер больничного и нажать кнопку «Принять». Также есть возможность создать новый больничный лист. Для этого нужно указать Номер больничного, Срок действия больничного листа и нажать кнопку «Выписать больничный лист». Номер только что выписанного б/л можно выбирать из списка «Номер больничного».

По клику на кнопке «Принять» происходит выполнение запроса «new\_rec», на кнопке «Выписать больничный лист» - запроса «insert\_list»

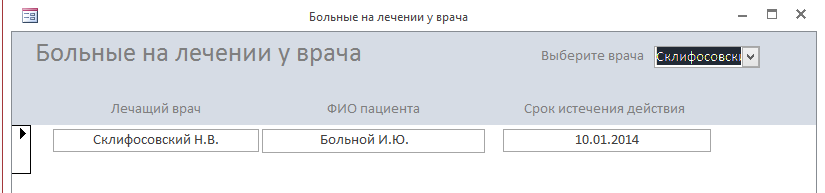
1. Форма «sick\_list»



Источником записей для формы является запрос «sick\_list»

Форма для предоставления больничных листов больным. Выводится информация по каждому больничному листу базы, а именно кому выдан, дата рождения больного, номер паспорта, дата создания б/л и срок истечения действия б/л. Возможен переход по записям для поиска нужного больничного листа.

1. Форма «Больные на лечении у врача»



На форме можно получить ФИО больных, у которых не истек срок действия больничного листа. По выбору ФИО из поля со списком происходит фильтрация записей по ФИО лечащего врача.

VBA код для события поля со списком «После обновления»:

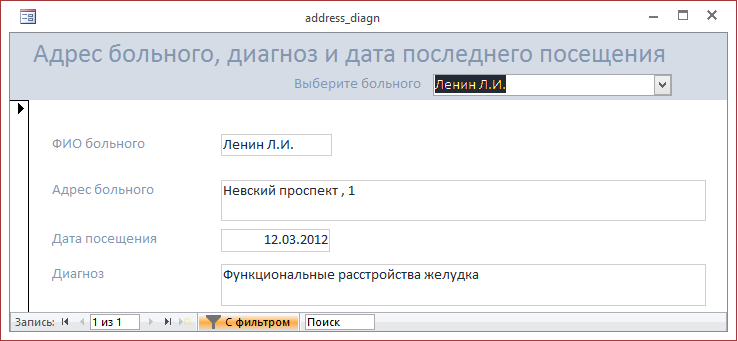
*Private Sub Комбинированная9\_AfterUpdate()*

*Me.Filter = "Doctor.ФИО = '" & Me.Комбинированная9 & "'"*

*Me.FilterOn = True*

*End Sub*

1. Форма «Информация о больном»



На форме можно получить информацию о каждом больном – его адрес, когда он в последний раз посещал больницу и какой диагноз ему поставлен.

Как и в предыдущей форме можно осуществлять переход по записям и поиск нужных больных.

По выбору ФИО из поля со списком происходит фильтрация записей по ФИО больного.

VBA код для события поля со списком «После обновления»:

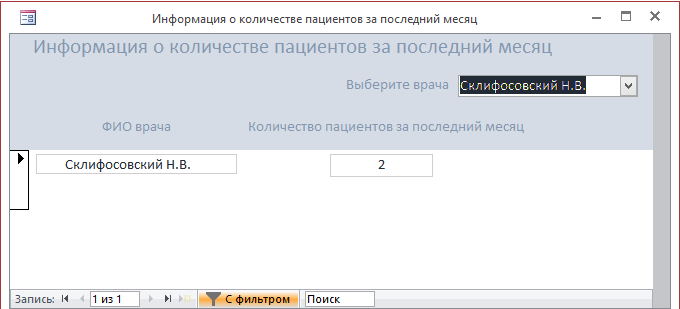
*Private Sub Комбинированная8\_AfterUpdate()*

*Me.Filter = "[First-ФИО] = '" & Me.Комбинированная8 & "'"*

*Me.FilterOn = True*

*End Sub*

1. Форма «Информация о количестве пациентов за последний месяц»



На форме выводится список врачей с указанием количества пациентов, обслуженных за последний месяц.

VBA код для события поля со списком «После обновления»:

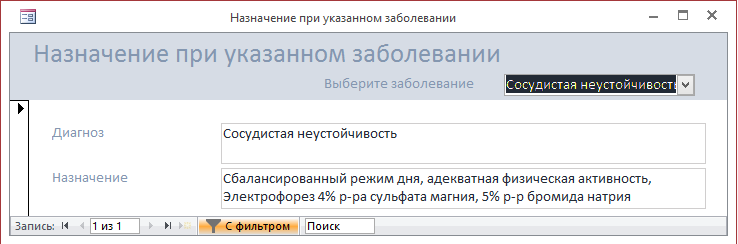
*Private Sub Комбинированная5\_AfterUpdate()*

*Me.Filter = "ФИО = '" & Me.Комбинированная5 & "'"*

*Me.FilterOn = True*

*End Sub*

1. Форма «Назначение при указанном заболевании»



В данной форме выводятся диагнозы, которые ставили врачи, и соответствующие им назначения. Возможен переход по записям и поиск.

VBA код ля события поля со списком «После обновления»:

*Private Sub Комбинированная5\_AfterUpdate()*

*Me.Filter = "Диагноз = '" & Me.Комбинированная5 & "'"*

*Me.FilterOn = True*

1. Форма «Номер кабинета и дни приёма врача»



В этой форме одна запись содержит номер кабинета, день приёма и ФИО врача, который работает в данное время в этом кабинете.

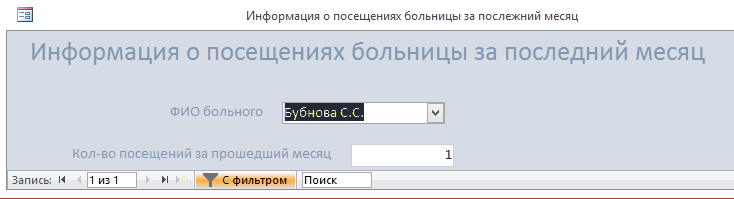
VBA код ля события поля со списком «После обновления»:

*Private Sub Комбинированная5\_AfterUpdate()*

*Me.Filter = "ФИО = '" & Me.Комбинированная5 & "'"*

*Me.FilterOn = True*

1. Форма «Посещение больницы больным за последний месяц»



В записях содержится информация только о тех больных, которые посещали поликлинику за последний месяц хотя бы один раз. Выводится количество посещений и ФИО больного.

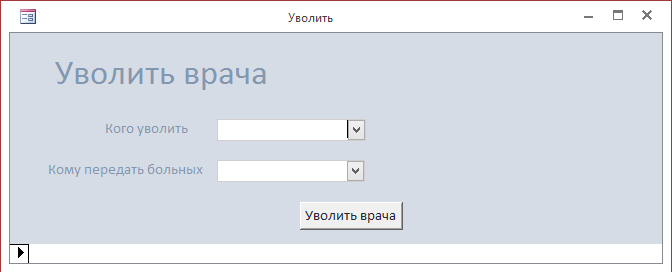
VBA код ля события поля со списком «После обновления»:

*Private Sub Комбинированная5\_AfterUpdate()*

*Me.Filter = "ФИО = '" & Me.Комбинированная5 & "'"*

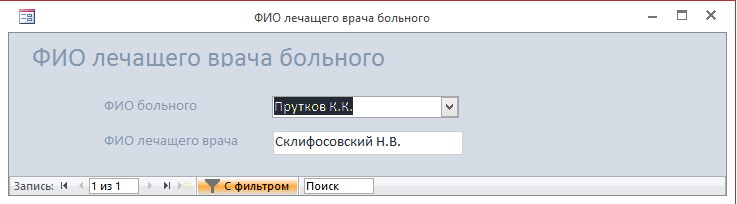
*Me.FilterOn = True*

1. Форма «Уволить»



В списке «Кого уволить» нужно выбрать номер паспорта врача, информация о котором будет удаляться их БД. В списке «Кому передать больных» нужно выбрать номер паспорта врача, который будет в дальнейшем обслуживать участки удаляемого врача и его пациентов. По кнопке «Уволить врача» запускается макрос «mac\_del\_doct». Уволить врача может только Главный врач или Администратор БД.

1. Форма «ФИО лечащего врача больного»



В данной форме выводятся ФИО больных и их лечащих врачей. Возможен переход по записям и поиск врачей и больных.

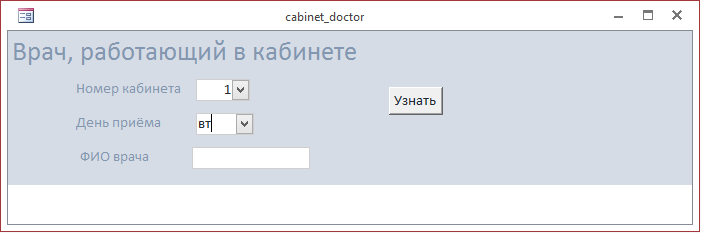
VBA код ля события поля со списком «После обновления»:

*Private Sub Комбинированная5\_AfterUpdate()*

*Me.Filter = "ФИО = '" & Me.Комбинированная5 & "'"*

*Me.FilterOn = True*

1. Форма «Врач, работающий в кабинете»



Необходимо указать номер кабинета, день приёма и нажать кнопку «Узнать».

В поле «ФИО врача» отобразится врач, работающий в указанном месте в указанное время.

Код, выполняемый по клику:

*Option Compare Database*

*Private Sub Кнопка8\_Click()*

*Dim N As String*

*If IsNull(Me.num) Or IsNull(Me.day) Then*

*MsgBox "Неправильные входные значения"*

*Exit Sub*

*End If*

*If CurrentProject.Connection.Execute("SELECT ФИО FROM Cabinet INNER JOIN Doctor ON Cabinet.[№ паспорта врача] = Doctor.[Номер паспорта] WHERE (( [День приёма] = '" & Me.day & "') AND ([Номер кабинета] = " & Me.num & "))").EOF = True Then*

*MsgBox "Кабинет пустует"*

*Me("doc").Value = ""*

*Exit Sub*

*End If*

*N = CurrentProject.Connection.Execute("SELECT ФИО FROM Cabinet INNER JOIN Doctor ON Cabinet.[№ паспорта врача] = Doctor.[Номер паспорта] WHERE (( [День приёма] = '" & Me.day & "') AND ([Номер кабинета] = " & Me.num & "))").Fields(0)*

*Me("doc").Value = N*

*End Sub*

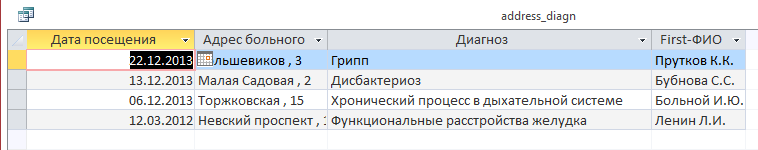
1. **Описание запросов**
2. Запрос «addres\_diag» - получение информации о адресе больного, дате последнего посещения и поставленном диагнозе.

*SELECT First(Records.[Дата посещения]) AS [First-Дата посещения], Card.[Адрес больного], First(Records.Диагноз) AS [First-Диагноз], First(Card.ФИО) AS [First-ФИО]*

*FROM Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]*

*GROUP BY Card.[Адрес больного]*

*ORDER BY First(Records.[Дата посещения]) DESC;*

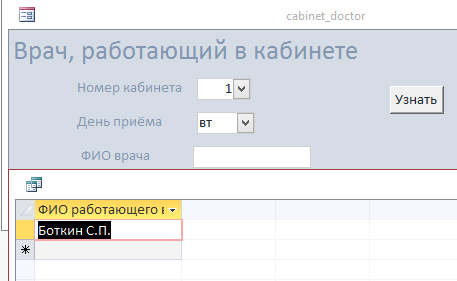
**

1. Запрос «cabinet\_doctor» - ФИО врача, который работает в указанном кабинете в указанное время.

*SELECT Doctor.ФИО*

*FROM Doctor INNER JOIN Cabinet ON Doctor.[Номер паспорта] = Cabinet.[№ паспорта врача]*

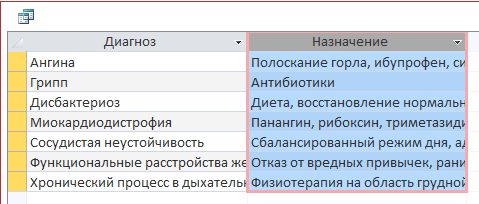
*WHERE (((Cabinet.[Номер кабинета])=[Forms]![Врач, работающий в кабинете]![num]) AND ((Cabinet.[День приёма])=[Forms]![Врач, работающий в кабинете]![day]));*



1. Запрос «diag\_nazn» - получение списка заболеваний и назначений по ним

*SELECT DISTINCT Records.Диагноз, Records.Назначение*

*FROM Records;*

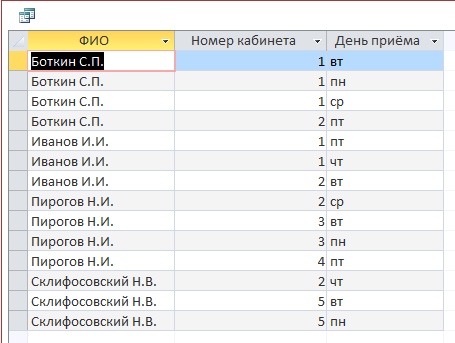


1. Запрос «doctor\_cabinet» - информация по всем дням и кабинетам с указанием работающего врача

*SELECT Doctor.ФИО, Cabinet.[Номер кабинета], Cabinet.[День приёма]*

*FROM Doctor INNER JOIN Cabinet ON Doctor.[Номер паспорта] = Cabinet.[№ паспорта врача]*

*GROUP BY Doctor.ФИО, Cabinet.[Номер кабинета], Cabinet.[День приёма];*

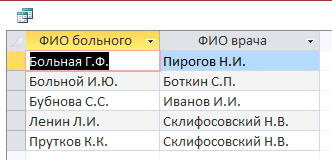


1. Запрос «doctor\_FIO» - список больных с указанием их лечащего врача

*SELECT Card.ФИО, First(Doctor.ФИО) AS [First-ФИО]*

*FROM Doctor INNER JOIN (Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]) ON Doctor.[Номер паспорта] = Records.[Номер паспорта врача]*

*GROUP BY Card.ФИО;*



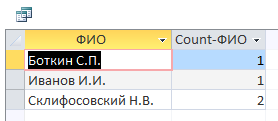
1. Запрос «m\_count\_doctor» - список врачей с указанием количества пациентов за последний месяц

*SELECT Doctor.ФИО, Count(Doctor.ФИО) AS [Count-ФИО]*

*FROM Doctor INNER JOIN Records ON Doctor.[Номер паспорта] = Records.[Номер паспорта врача]*

*WHERE (((Records.[Дата посещения])>=DateAdd("m",-1,Now())))*

*GROUP BY Doctor.ФИО;*



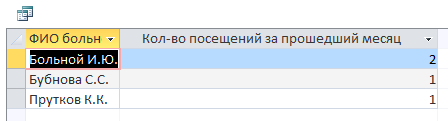
1. Запрос «m\_count\_rec» - список больных с указанием количества записей в карточку за последний месяц

*SELECT Card.ФИО, Count(Card.ФИО) AS [Count-ФИО]*

*FROM Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]*

*WHERE ((([Records]![Дата посещения])>=DateAdd("m",-1,Now())))*

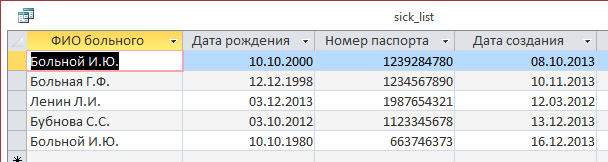
*GROUP BY Card.ФИО;*



1. Запрос «sick\_list» - список больничных листов, кому и когда они выписаны

*SELECT Card.ФИО, Card.[Дата рождения], Card.[Номер паспорта], [Sick-list].[Дата создания], [Sick-list].[Срок истечения действия]*

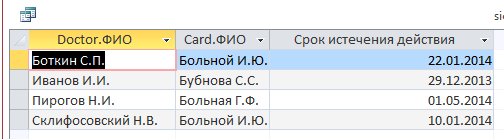
*FROM (Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]) INNER JOIN [Sick-list] ON Records.[Номер больничного листа] = [Sick-list].[Номер больничного];*



1. Запрос «sick\_list\_now» - список только тех больных с б/л, у которых срок больничного листа еще не истек.

*SELECT Card.ФИО, Card.[Дата рождения], Card.[Номер паспорта], [Sick-list].[Дата создания], [Sick-list].[Срок истечения действия]*

*FROM (Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]) INNER JOIN [Sick-list] ON Records.[Номер больничного листа] = [Sick-list].[Номер больничного];*



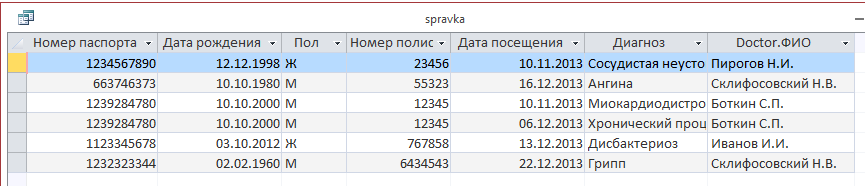
1. Запрос «spravka» - информация о больных и их посещениях поликлиники за последний месяц. Запрос необходим для выдачи справки.

*SELECT Card.ФИО, Card.[Номер паспорта], Card.[Дата рождения], Card.Пол, Card.[Номер полиса], Records.[Дата посещения], Records.Диагноз, Doctor.ФИО*

*FROM Doctor INNER JOIN (Card INNER JOIN Records ON Card.[Номер карточки] = Records.[Номер карточки]) ON Doctor.[Номер паспорта] = Records.[Номер паспорта врача]*

*GROUP BY Card.ФИО, Card.[Номер паспорта], Card.[Дата рождения], Card.Пол, Card.[Номер полиса], Records.[Дата посещения], Records.Диагноз, Doctor.ФИО*

*HAVING (((Year([Records]![Дата посещения]))=Year(Now())));*



1. Запрос «insert\_list» - запрос на добавление нового больничного листа

*INSERT INTO [Sick-list] ( [Номер больничного], [Дата создания], [Срок истечения действия] )*

*VALUES (Формы![New\_rec]![numb], Date(), Формы![New\_rec]![end]);*

1. Запрос «new\_card» - запрос на создание карточки для нового больного

*INSERT INTO Card ( [Номер карточки], ФИО, [Номер паспорта], [Адрес больного], Пол, [Дата рождения], [Номер полиса] )*

*VALUES ([Формы]![New\_card]![number], [Формы]![New\_card]![FIO], [Формы]![New\_card]![pas], [Формы]![New\_card]![adr], [Формы]![New\_card]![pol], [Формы]![New\_card]![data], [Формы]![New\_card]![polis]);*

1. Запрос «new\_rec» - запрос на добавление новой записи о посещении поликлиники

*INSERT INTO Records ( [Номер карточки], [Дата посещения], Жалобы, Диагноз, Назначение, [Номер паспорта врача], [Номер больничного листа] )*

*SELECT [Формы]![New\_rec]![number] AS Выражение1, [Формы]![New\_rec]![data] AS Выражение2, [Формы]![New\_rec]![gal] AS Выражение3, [Формы]![New\_rec]![diag] AS Выражение4, [Формы]![New\_rec]![nazn] AS Выражение7, [Формы]![New\_rec]![doct] AS Выражение5, [Формы]![New\_rec]![sick] AS Выражение6;*

1. Запрос «update\_area» - запрос на передачу участков другому врачу

*UPDATE Doctor INNER JOIN Area ON Doctor.[Номер паспорта] = Area.[Номер паспорта врача] SET Area.[Номер паспорта врача] = [Формы]![Уволить]![pasp\_new]*

*WHERE (((Doctor.[Номер паспорта])=[Формы]![Уволить]![pasp\_uvol]));*

1. Запрос «update\_bol» - запрос на передачу больных другому врачу

*UPDATE Records SET Records.[Номер паспорта врача] = [Формы]![Уволить]![pasp\_new]*

*WHERE (((Records.[Номер паспорта врача])=[Формы]![Уволить]![pasp\_uvol]));*

1. Запрос «del\_cab» - удаление записей из таблицы «Cabinet», которые соответствуют указанному врачу

*DELETE Cabinet.[№ паспорта врача]*

*FROM Cabinet*

*WHERE (((Cabinet.[№ паспорта врача])=[Формы]![Уволить]![pasp\_uvol]));*

1. Запрос «del\_doct» - непосредственное удаление записи о враче из таблицы «Doctor»

*DELETE Doctor.[Номер паспорта]*

*FROM Doctor*

*WHERE (((Doctor.[Номер паспорта])=[Формы]![Уволить]![pasp\_uvol]));*

1. **Описание макросов**
2. Макрос «AutoExec» выполняется при запуске приложения БД, открывает форму аутентификации.

*Function AutoExec()*

*On Error GoTo AutoExec\_Err*

*DoCmd.Echo True, "Запуск приложения"*

*DoCmd.Hourglass True*

*DoCmd.OpenForm "Вход", acNormal, "", "", , acNormal*

*AutoExec\_Exit:*

*Exit Function*

*AutoExec\_Err:*

*MsgBox Error$*

*Resume AutoExec\_Exit*

*End Function*

1. Макрос «exit» производит выход из приложения БД. Выполняется по клику на кнопках «Выход», которые расположены на формах «Вход», «ВходAdmin», «ВходГлавныйВрач», «ВходРегистратура», «ВходВрач»

*Function exit ()*

*On Error GoTo exit\_Err*

*DoCmd.RunCommand acCmdCloseDatabase*

*exit\_Exit:*

*Exit Function*

*exit\_Err:*

*MsgBox Error$*

*Resume exit\_Exit*

*End Function*

1. Макрос «mac\_del\_doct» запускает запросы на удаление и изменении информации о враче (они необходимы для удаления врача из базы). Вызывается по клику по кнопке из формы «Удалить».

*Function mac\_del\_doct()*

*On Error GoTo mac\_del\_doct\_Err*

*DoCmd.OpenQuery "update\_bol", acViewNormal, acEdit*

*DoCmd.OpenQuery "update\_area", acViewNormal, acEdit*

*DoCmd.OpenQuery "del\_cab", acViewNormal, acEdit*

*DoCmd.OpenQuery "del\_doct", acViewNormal, acEdit*

*mac\_del\_doct\_Exit:*

*Exit Function*

*mac\_del\_doct\_Err:*

*MsgBox Error$*

*Resume mac\_del\_doct\_Exit*

*End Function*

1. Макрос «mac\_sick» запускает запрос на добавление нового больничного листа в базу и обновляется форму «New\_rec» для того, чтобы можно было сразу выбрать новый б/л из списка.

*Function mac\_sick()*

*On Error GoTo mac\_sick\_Err*

*DoCmd.OpenQuery "insert\_list", acViewNormal, acEdit*

*DoCmd.RepaintObject acForm, "New\_rec"*

*mac\_sick\_Exit:*

*Exit Function*

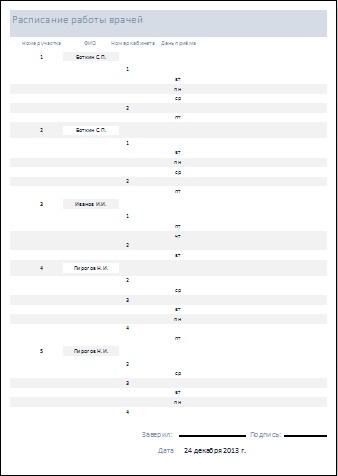
*mac\_sick\_Err:*

*MsgBox Error$*

*Resume mac\_sick\_Exit*

*End Function*

1. **Описание отчетов**
2. Отчет «Расписание работы врачей»

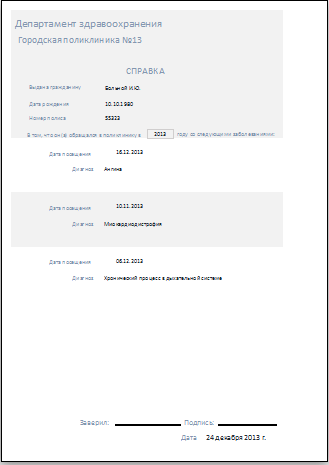


Выводит информацию о врачах, участках, на которых они работают, время и дни приёма. Расписание можно повесить в холе. К просмотру допускаются все группы пользователей данной БД.

Источник записей для отчета:

*SELECT [Address].[Улица, дом], [Area].[Номер участка], [Cabinet].[Номер кабинета], [Cabinet].[День приёма], [Doctor].[ФИО] FROM (([Doctor] INNER JOIN ([Area] INNER JOIN [Address] ON [Area].[Номер участка] =[Address].[Номер участка]) ON [Doctor].[Номер паспорта] =[Area].[Номер паспорта врача]) INNER JOIN [Cabinet] ON [Doctor].[Номер паспорта] =[Cabinet].[№ паспорта врача])*

1. Отчет «Справка»



Выводим информацию о посещениях поликлиники за последний год с указанием даты посещения и диагноза. Справку может выписать только Врач и Администратор.

Источник записей – запрос «spravka»

1. **Вывод**

В результате выполнения данной курсовой работы была создана и протестирована база данных, предназначенная для врачей и для работников регистратуры поликлиники.