

Московский авиационный институт
Национальный исследовательский университет
Факультет «Компьютерные науки и прикладная математика»
Кафедра «Вычислительная математика и программирование»

Курсовая работа
по курсу «Базы данных»

Санаторий №1 (Меню-раскладка)

Студентка: Кузнецова Д.А.
Группа: М80-305Б-20
Преподаватель: Киндинова В.В.
Оценка:

Москва, 2022

1. Техническое задание

Требуется разработать прикладное программное обеспечение для Санатория. Данное приложение разработано для работы администратора с данными пользователя. Основная часть программы предоставляет возможность добавлять, удалять или изменять основные данные для посетителя санатория. О каждом посетителе имеются следующие основные данные: id мед книжки, имя, фамилия, возраст, пол, даты заезда и выезда. Также в программе предусмотрена возможность просмотра посетителей по заданным критериям (по диетам и по диагнозам) и просмотра всех продуктов по введенному диагнозу.

Для того чтобы добавить нового посетителя, необходимо выбрать нужную кнопку и в новом окне ввести данные посетителя. После сохранения нового посетителя его данные будут добавлены в базу данных. Для того чтобы удалить данные посетителя, после выбора нужной кнопки, в новом окне необходимо ввести id медицинской книжки этого посетителя. Изменение основных данных посетителей также происходит при помощи id его медицинской книжки.

Создаваемое программное обеспечение должно, обладая соответствующей функциональностью, обеспечивать информационную поддержку следующих процессов:

- Поиск основной информации о посетителях через форму;
- Добавление нового посетителя;
- Удаление пользователя;
- Изменение данных посетителей;
- Составление отчетов;
- Обеспечение перехода между формами;

Чтобы удовлетворять поставленным требованиям к функциональности, программа должна содержать следующие формы:

- Форма «Главное Меню» позволяет выбрать необходимое действие для администратора. Она обеспечивает переход между остальными формами.
- Форма «Добавление пользователя» содержит поля для ввода информации нового пользователя и кнопку для сохранения этих данных в базе данных.
- Форма «Удаление пользователя» позволяет удалить данные о пользователе. После ввода в текстовое поле Id мед книжки пользователя и нажатия на кнопку «Удалить», пользователь с таким номером мед книжки будет удален из базы данных.
- Форма «Изменение пользователя» содержит текстовое поле для ввода id мед книжки пользователя, данные которого необходимо изменить. После поиска такого пользователя в форме отображаются дополнительные поля для изменения и сохранения его новых данных.
- Форма «Просмотр посетителей по диагнозам» содержит поле для ввода id диагноза. После его написания и поиска в отведенном поле появятся все пользователи с данным диагнозом. Для полученной таблицы можно составить отчет.
- Форма «Просмотр посетителя по диетам» содержит поле для ввода id диеты. После поиска в отведенном поле появятся все пользователи, которые имеют такую диету. Также по полученной таблице можно составить отчет.
- Форма «Просмотр диагнозов» содержит текстовое поле для ввода id диагноза. После выбора диагноза будет выведена в отведенном поле таблица с наименованием продуктов и их калорий, подходящих для данного диагноза. Для полученных данных можно составить отчет.
- Форма «Найти посетителя» позволяет вывести информацию о посетителе по введенному в текстовое поле id его медицинской книжки.

Id_med_inf – integer (Номер Медицинской книжки)

	Id_med_inf	Id_diagnos	Allergy	Blood_type	Weight	Height
	457	8	отсутствует	3	45.5	165.4
	343	2	шерсть	3	55.2	163.0
	234	2	морепродукты	1	44.0	172.2
	453	5	отсутствует	2	88.2	188.0
	643	4	холод	4	68.2	176.2
	239	6	шерсть	4	87.0	162.2
	783	7	быстрые углеводы	2	69.2	169.1
	998	8	отсутствует	1	55.2	178.0
	876	5	отсутствует	3	61.1	167.2
	321	8	морепродукты	4	87.2	178.2
	567	7	пыль	1	67.2	166.0
	495	4	стиральный порошок	4	50.3	177.2
	227	4	цитрусовые	4	44.2	164.9
	811	6	пенициллин	1	78.3	179.2
	798	3	тараканы	3	61.0	178.0
	981	8	отсутствует	2	43.1	152.8
	902	3	солнечный свет	1	68.1	178.0
	223	7	отсутствует	1	58.2	152.0

Рисунок 2. Таблица Med_info

Типы данных:

Id_med_inf – integer (Номер медицинской книжки)

Id_diagnosis – integer (Номер диагноза для связи с таблицей Diagnosis)

Allergy – varchar (Аллергия)

Blood_type – integer (Группа крови)

Weight – float (Вес)

Height – float (Рост)

	Id_diagnos	Id_diet	Name
	8	5	язва желудка
	1	5	язва двенадцатиперстной кишки
	2	1	гастрит
	3	1	колиты
	4	2	мочекаменная болезнь
	5	2	подагра
	6	3	сахарный диабет
	7	4	сосудисто-сердечные заболевания

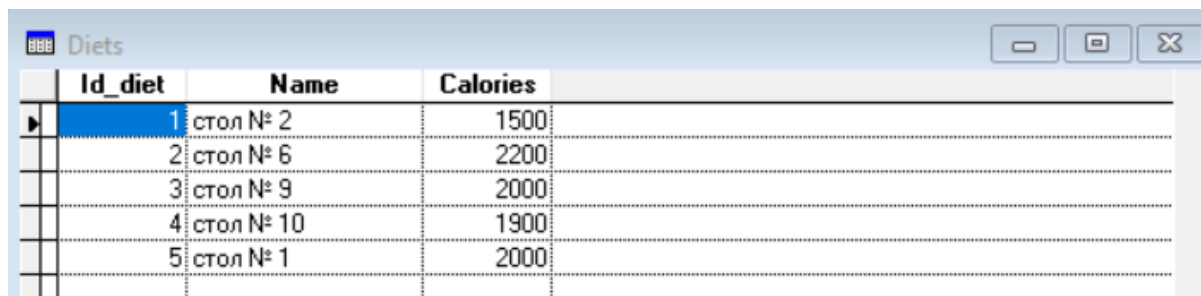
Рисунок 3. Таблица Diagnosis

Типы данных:

Id_diagnosis – integer (Номер диагноза)

Id_diet – integer (Номер диеты для связи с таблицей Diets)

Name – varchar (Название диагноза)



	Id_diet	Name	Calories
▶	1	стол № 2	1500
	2	стол № 6	2200
	3	стол № 9	2000
	4	стол № 10	1900
	5	стол № 1	2000

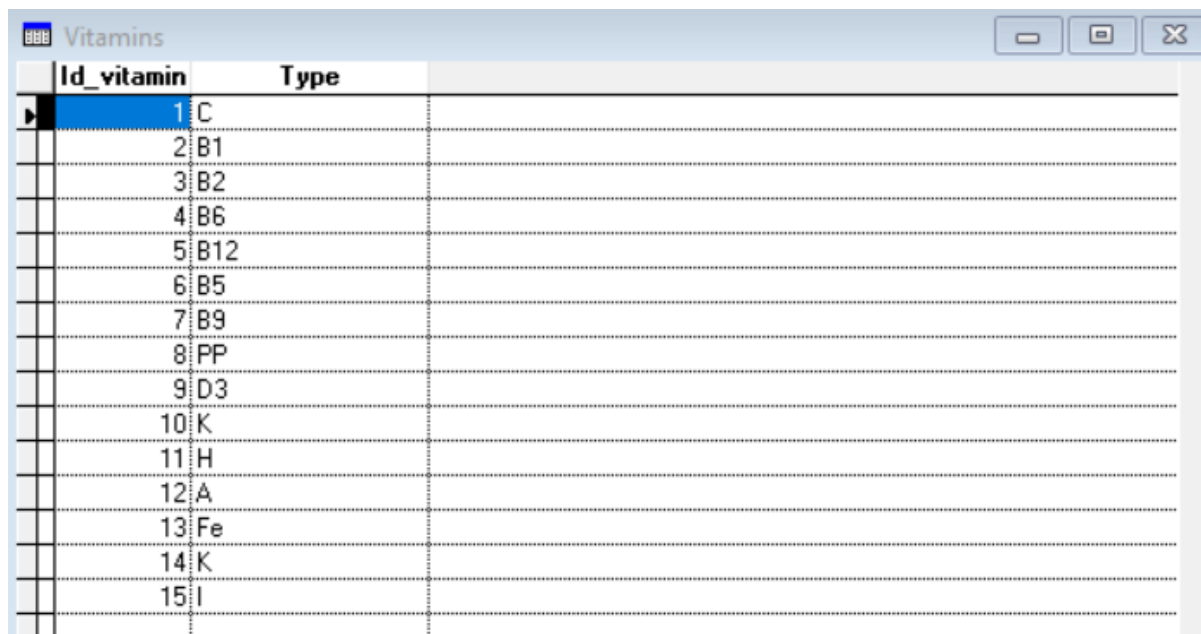
Рисунок 4. Таблица Diet

Типы данных:

Id_diet – integer (Номер диеты)

Name – varchar (Номер стола в столовой)

Calories – integer (Калорийность диеты для составления рациона питания)



	Id_vitamin	Type
▶	1	C
	2	B1
	3	B2
	4	B6
	5	B12
	6	B5
	7	B9
	8	PP
	9	D3
	10	K
	11	H
	12	A
	13	Fe
	14	K
	15	I

Рисунок 5. Таблица Vitamins

Типы данных:

Id_vitamin – integer (Номер витаминов)

Type – varchar (Витамин)

Food						
	Id_food	Name	Proteins	Carbohydr	Fats	Calories
	1	Молоко	3	5	3	64
	2	Кефир	4	5	2	50
	3	Яйцо	12	2	10	122
	4	Сок	1	13	1	60
	5	Картофель	2	17	1	77
	6	Лук	1	10	1	40
	7	Морковь	1	7	1	35
	8	Капуста	1	7	1	25
	9	Рис	7	30	2	300
	10	Макароны	4	25	6	337
	11	Гречка	3	50	10	300
	12	Курица	20	4	30	350
	13	Индейка	20	5	40	360
	14	Говядина	40	3	79	400
	15	Свинина	35	12	60	350
	16	Яблоко	1	13	1	60
	17	Банан	3	22	3	97
	18	Апельсин	2	9	0	38
	19	Мандарин	1	9	1	40
	20	Нектарин	2	5	1	23
	21	Дыня	1	8	1	35
	22	Лимон	1	9	0	27
	23	Хлеб б	8	50	2	100
	24	Хлеб ч	5	40	1	90
	25	Мёд	8	80	1	308
	26	Зефир	20	80	1	299
	27	Шоколад	7	60	35	577
	28	Мармелад	1	77	2	296
	29	Фундук	10	20	50	600
	30	Миндаль	13	24	47	579

Рисунок 6. Таблица Food

Типы данных:

Id_food – integer (Номер еды)

Name – varchar (Еда)

Proteins – integer (Количество белков на 100 г)

Carbohydr – integer (Количество углеводов на 100 г)

Fats – integer (Количество жиров на 100 г)

Calories – integer (Калорийность данного продукта)

Diets_vitamins	
Id_diet	Id_vitamin
5	2
5	3
5	4
5	5
5	6
5	7
1	1
1	12
2	1
2	5
2	10
2	11
2	12
2	13
2	14
2	15
3	5
3	13
4	1
4	2
4	3
4	12

Рисунок 7. Таблица Diet_vitamins

Таблица для связи диеты и витаминов, входящих в нее.

Типы данных:

Id_diet – integer (Номер диеты)

Id_vitamin – integer (Номер витамина)

Diets_food		
	Id_diet	Id_food
	5	1
	5	3
	5	4
	5	5
	5	6
	5	7
	5	9
	5	10
	5	11
	5	12
	5	13
	5	14
	5	17
	5	20
	5	25
	2	1
	2	2
	2	3
	2	4
	2	5
	2	6
	2	7
	2	9
	2	11
	2	12
	2	13
	2	14
	2	15
	2	16
	2	17
	2	18
	2	19
	2	20
	2	21
	2	22
	2	23
	2	24
	2	25
	2	26
	2	28

Рисунок 8. Таблица Diets_food (1)

Diets_food		
	Id_diet	Id_food
	2	26
	2	28
	3	1
	3	2
	3	3
	3	5
	3	6
	3	8
	3	11
	3	12
	3	13
	3	14
	3	15
	3	18
	3	19
	3	22
	3	24
	4	1
	4	2
	4	4
	4	5
	4	6
	4	7
	4	8
	4	11
	4	12
	4	13
	4	16
	4	17
	4	18
	4	19
	4	20
	4	21
	4	22
	4	23
	4	24
	1	1
	1	2
	1	3
	1	4

Рисунок 9. Таблица Diets_food(2)

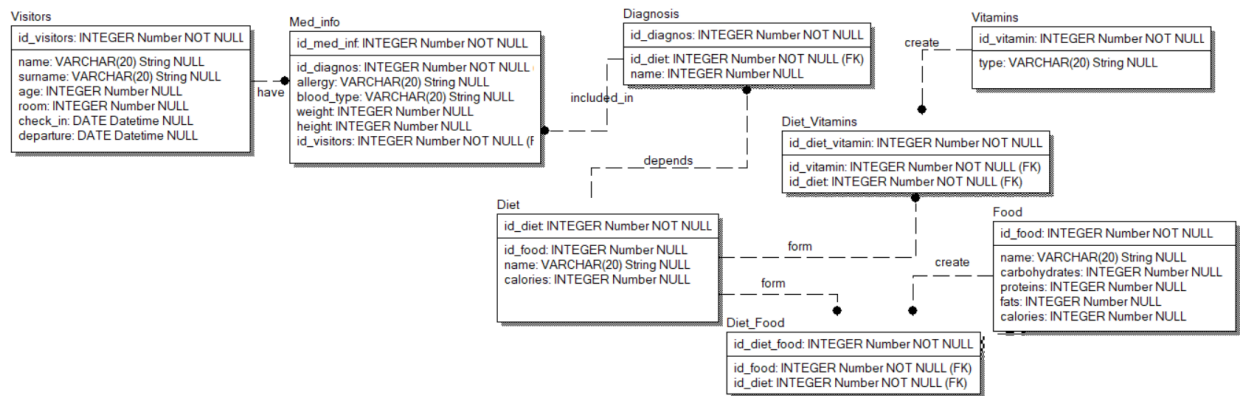


Рисунок 2 Физическая схема

4. Описание сгенерированной БД

```

CREATE TABLE Diagnosis
(
    id_diet          INTEGER NOT NULL,
    name            INTEGER NULL,
    id_diagnos      INTEGER NOT NULL
);
ALTER TABLE Diagnosis
ADD PRIMARY KEY (id_diagnos);
CREATE UNIQUE INDEX AK2 ON Diagnosis
(
    name
);
CREATE TABLE Diet
(
    id_diet          INTEGER NOT NULL,
    name            VARCHAR(20) NULL,
    id_food          INTEGER NULL,
    calories         INTEGER NULL
);
ALTER TABLE Diet
ADD PRIMARY KEY (id_diet);
CREATE UNIQUE INDEX AK1 ON Diet
(
    name
);
CREATE TABLE Diet_Food
(
    id_diet          INTEGER NOT NULL,
    id_food          INTEGER NOT NULL,
    id_diet_food     INTEGER NOT NULL
);
ALTER TABLE Diet_Food
ADD PRIMARY KEY (id_diet_food);
CREATE TABLE Diet_Vitamins
(
    id_diet          INTEGER NOT NULL,
    id_vitamin       INTEGER NOT NULL,
    id_diet_vitamin  INTEGER NOT NULL
);
ALTER TABLE Diet_Vitamins
ADD PRIMARY KEY (id_diet_vitamin);
CREATE TABLE Food
(
    id_food          INTEGER NOT NULL,
    name            VARCHAR(20) NULL,
    carbohydrates    INTEGER NULL,
    proteins         INTEGER NULL,
    fats            INTEGER NULL,

```

```

        calories                INTEGER NULL
    );
    ALTER TABLE Food
    ADD PRIMARY KEY (id_food);
    CREATE UNIQUE INDEX AK1 ON Food
    (
        name
    );
    CREATE TABLE Med_info
    (
        id_diagnos                INTEGER NOT NULL,
        allergy                    VARCHAR(20) NULL,
        blood_type                VARCHAR(20) NULL,
        weight                    INTEGER NULL,
        height                    INTEGER NULL,
        id_med_inf                INTEGER NOT NULL,
        id_visitors                INTEGER NOT NULL
    );
    ALTER TABLE Med_info
    ADD PRIMARY KEY (id_med_inf);
    CREATE TABLE Visitors
    (
        name                      VARCHAR(20) NULL,
        surname                   VARCHAR(20) NULL,
        age                      INTEGER NULL,
        room                     INTEGER NULL,
        check_in                 DATE NULL,
        departure                 DATE NULL,
        id_visitors                INTEGER NOT NULL
    );
    ALTER TABLE Visitors
    ADD PRIMARY KEY (id_visitors);
    CREATE INDEX IE1 ON Visitors
    (
        surname
    );
    CREATE TABLE Vitamins
    (
        id_vitamin                INTEGER NOT NULL,
        type                     VARCHAR(20) NULL
    );
    ALTER TABLE Vitamins
    ADD PRIMARY KEY (id_vitamin);
    CREATE UNIQUE INDEX AK1 ON Vitamins
    (
        type
    );
    ALTER TABLE Diagnosis
    ADD FOREIGN KEY depends (id_diet) REFERENCES Diet (id_diet);
    ALTER TABLE Diet_Food
    ADD FOREIGN KEY form (id_diet) REFERENCES Diet (id_diet);
    ALTER TABLE Diet_Food
    ADD FOREIGN KEY create (id_food) REFERENCES Food (id_food);
    ALTER TABLE Diet_Vitamins
    ADD FOREIGN KEY form (id_diet) REFERENCES Diet (id_diet);
    ALTER TABLE Diet_Vitamins
    ADD FOREIGN KEY create (id_vitamin) REFERENCES Vitamins (id_vitamin);
    ALTER TABLE Med_info
    ADD FOREIGN KEY included_in (id_diagnos) REFERENCES Diagnosis (id_diagnos);
    ALTER TABLE Med_info
    ADD FOREIGN KEY have (id_visitors) REFERENCES Visitors (id_visitors);

```

5. Интерфейс Базы Данных

Это программа, которая имеет несколько функций –

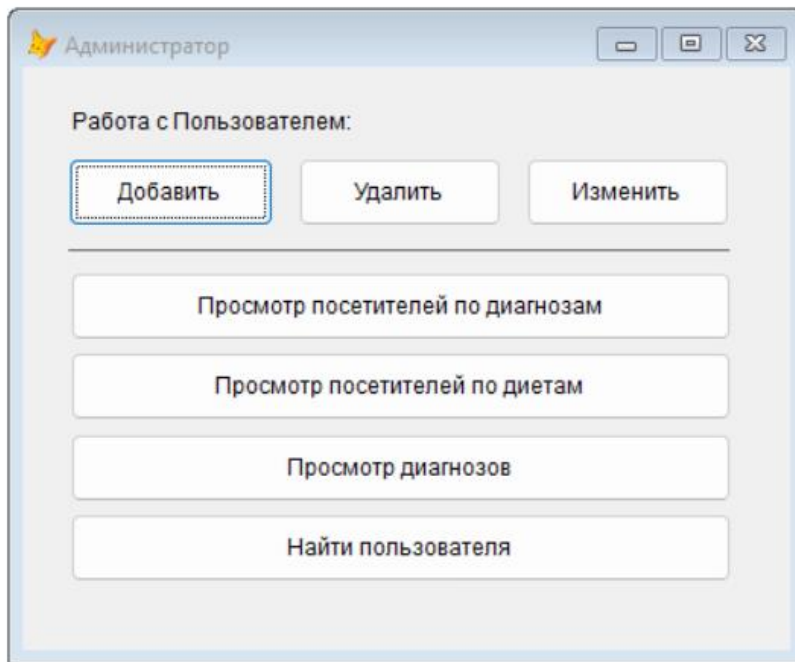
1) Работа с посетителем:

- Добавить нового посетителя

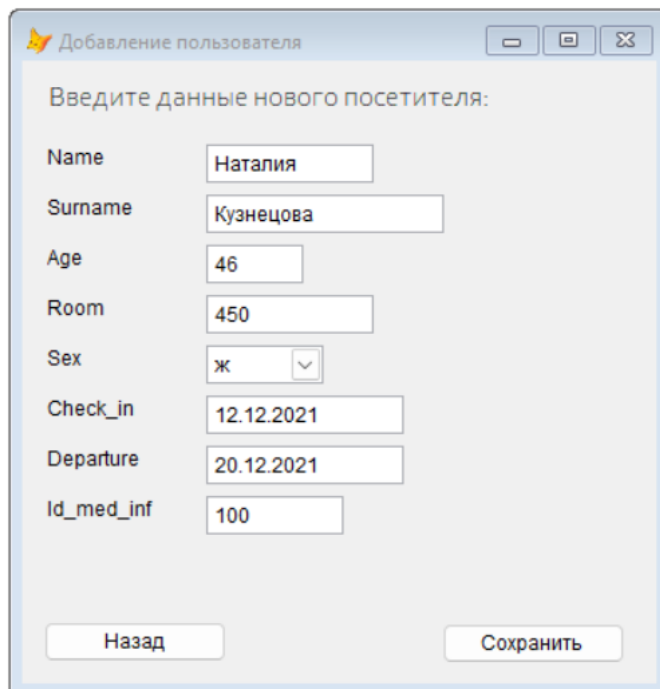
- Удалить посетителя
- Изменить информацию о посетителе
- 2) Просмотр посетителей по диагнозам
- 3) Просмотр посетителей по диетам
- 4) Просмотр диагнозов
- 5) Найти посетителя

Интерфейс взаимодействует с базой данных и отправляет в нее запросы.

Главное меню:



1. Добавление нового посетителя:



После нажатия кнопки «Сохранить», открывается окно:

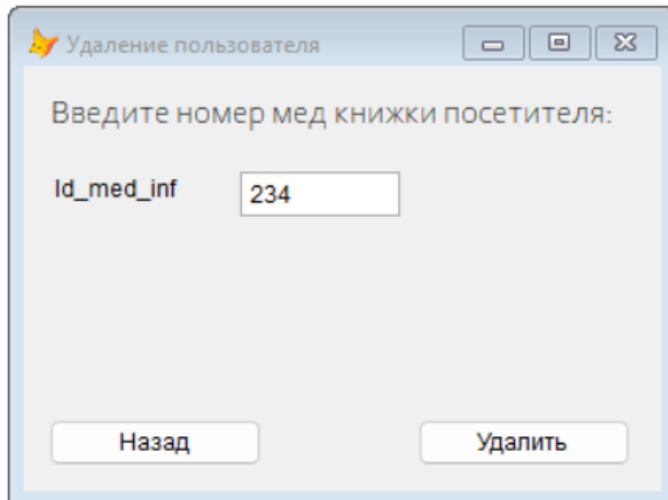
Добавление ✕

Посетитель успешно добавлен!

ОК

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.

2. Удаление посетителя:



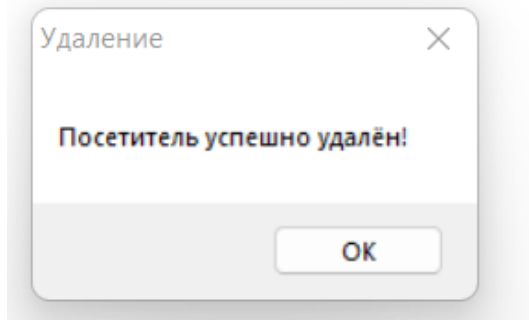
Удаление пользователя

Введите номер мед книжки посетителя:

Id_med_inf 234

Назад Удалить

После нажатия кнопки «Удалить», происходит удаление посетителя с данным номером медкнижки:



Удаление ✕

Посетитель успешно удалён!

ОК

3. Изменение пользователя:

Изменение пользователя

Введите номер мед книжки посетителя:

Id_med_inf 234 Найти

Назад Изменить

Если посетителя с данным номером медкнижки нет, то:

Внимание

Посетитель с таким id_med_inf не найден!

OK

Если ввести правильный номер медкнижки, то:

Изменение пользователя

Введите номер мед книжки посетителя:

Id_med_inf 234 Найти

Name София

Surname Милютинa

Age 20

Room 444

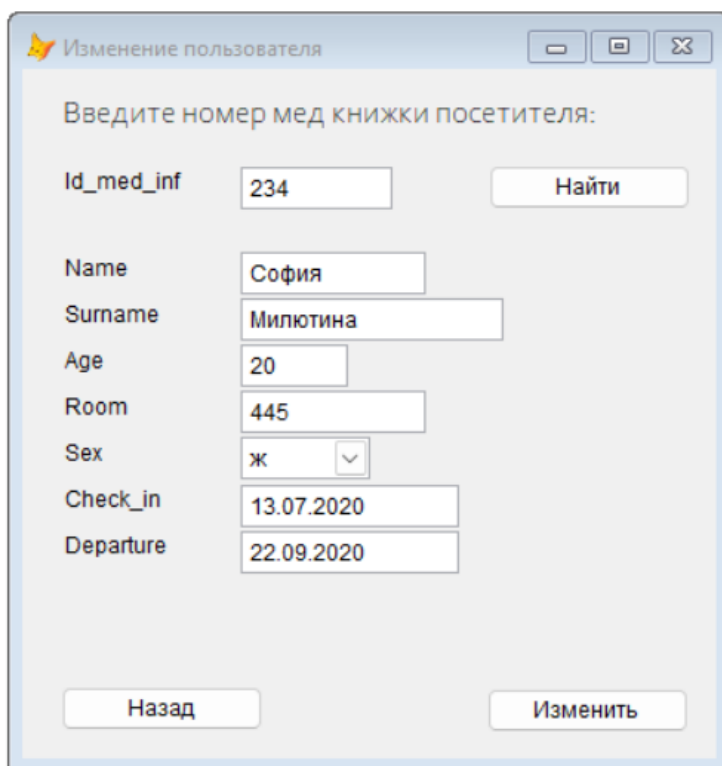
Sex ж

Check_in 13.07.2020

Departure 22.09.2020

Назад Изменить

Поменяем , например, номер комнаты (Переселение в комнату 445):



Изменение пользователя

Введите номер мед книжки посетителя:

Id_med_inf 234 Найти

Name София

Surname Милютина

Age 20

Room 445

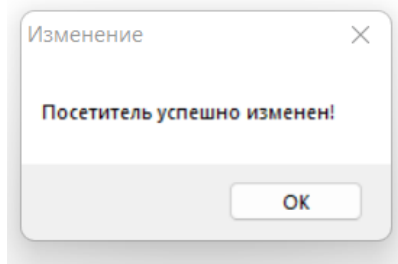
Sex ж

Check_in 13.07.2020

Departure 22.09.2020

Назад Изменить

После нажатия кнопки «Изменить»:



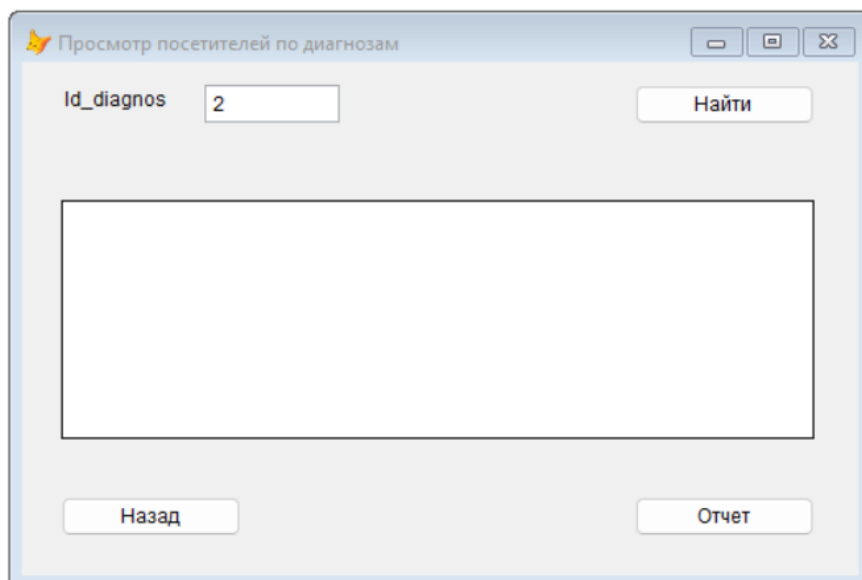
Изменение

Посетитель успешно изменен!

ОК

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.

4. Просмотр посетителей по диагнозам:



Просмотр посетителей по диагнозам

Id_diagnos 2 Найти

Назад Отчет

При нажатии кнопки «Найти»:

Просмотр посетителей по диагнозам

Id_diagnos

гастрит

	Name	Surname	Age	Sex	Room	Id_med_
	Наталья	Попова	19	ж	425	343
	София	Милютина	20	ж	445	234

При нажатии кнопки «Отчёт»:

СПИСОК ПОСЕТИТЕЛЕЙ

ДИАГНОЗ: гастрит

Name	Surname	Age	Sex	Room	Id_med_inf	Allergy	Blood_type
Наталья	Попова	19	ж	425	343	шерсть	3
София	Милютина	20	ж	444	234	морепродукты	1

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.

5. Просмотр посетителя по диетам:

Просмотр посетителей по диетам

Id_diet

При нажатии кнопки «Найти»:

Просмотр посетителей по диетам

Id_diet Найти

	Name	Surname	Age	Sex	Room	Id_med_
<input type="checkbox"/>	Грегори	Семенихин	40	м	545	239
<input type="checkbox"/>	Андрей	Голоков	31	м	502	811
<input type="checkbox"/>						
<input type="checkbox"/>						
<input type="checkbox"/>						

Назад Отчет

При нажатии кнопки «Отчёт»:

СПИСОК ПОСЕТИТЕЛЕЙ

Диета: стол № 9		Калории: 2000				
Name	Surname	Age	Sex	Room	Id_med_inf	Id_diagnos
Грегори	Семенихин	40	м	545	239	6
Андрей	Голоков	31	м	502	811	6

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.


6. Просмотр диагнозов:

Просмотр диагнозов

Id_diagnos Найти

Назад Отчет

При нажатии кнопки «Найти»:


Просмотр диагнозов

Id_diagnos

4

Найти

мочекаменная болезнь

	Id_diet	Name	Food	Calories
	2	стол № 6	Молоко	64
	2	стол № 6	Кефир	50
	2	стол № 6	Яйцо	122
	2	стол № 6	Сок	60
	2	стол № 6	Картофель	77
	2	стол № 6	Лук	40

Назад

Отчет

При нажатии кнопки «Отчёт»:

СПИСОК ПРОДУКТОВ

ДИАГНОЗ: мочекаменная болезнь

Name_diet	Food	Calories
стол № 6	Молоко	64
стол № 6	Кефир	50
стол № 6	Яйцо	122
стол № 6	Сок	60
стол № 6	Картофель	77
стол № 6	Лук	40
стол № 6	Морковь	35
стол № 6	Рис	300
стол № 6	Гречка	300
стол № 6	Курица	350
стол № 6	Индейка	360
стол № 6	Говядина	400
стол № 6	Свинина	350
стол № 6	Яблоко	60
стол № 6	Банан	97
стол № 6	Апельсин	38
стол № 6	Мандарин	40

стол № 6	Нектарин	23
стол № 6	Дыня	35
стол № 6	Лимон	27
стол № 6	Хлеб б	100
стол № 6	Хлеб ч	90
стол № 6	Мёд	308
стол № 6	Зефир	299
стол № 6	Мармелад	296

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.

7. Найти посетителя:

Найти пользователя

Id_med_inf 120 Найти

Name

Surname

Age

Room

Sex

Check_in

Departure

Weight

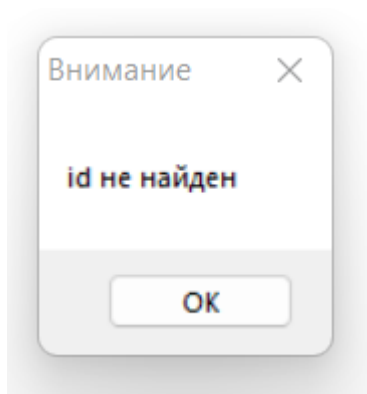
Height

Diagnos

Diet

Назад

Если посетителя с такой медкнижкой нет, то:



Если посетитель с такой медкнижкой есть, то:

Id_med_inf	343	Найти
Name	Наталья	
Surname	Попова	
Age	19	
Room	425	
Sex	ж	
Check_in	28.08.2020	
Departure	22.09.2020	
Weight	55.2	
Height	163.0	
Diagnos	гастрит	
Diet	стол № 2	
Назад		

При нажатии кнопки «Назад», мы возвращаемся в Главное меню.

6. Листинг

Форма «Главное меню»:

Листинг кнопки «Добавить»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\new_form.scx"  
RELEASE THISFORM
```

Листинг кнопки «Удалить»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\del_form.scx"  
RELEASE THISFORM
```

Листинг кнопки «Изменить»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\change_form.scx"  
RELEASE THISFORM
```

Листинг кнопки «Просмотр посетителей по диагнозам»:

```
O FORM "c:\users\дарина\desktop\бд\бд\diag_form.scx"
RELEASE THISFORM
```

Листинг кнопки «Просмотр посетителей по диетам»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\diet_form.scx"
RELEASE THISFORM
```

Листинг кнопки «Просмотр диагнозов»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\eat_vit_form.scx"
RELEASE THISFORM
```

Листинг кнопки «Найти пользователя»:

```
DO FORM "c:\users\дарина\desktop\бд\бд\find_form.scx"
RELEASE THISFORM
```

Форма «Добавить нового посетителя»

Листинг кнопки «Сохранить»:

```
IF EMPTY(thisform.Text1.Value) THEN
    MESSAGEBOX("Введите name", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text2.Value) THEN
    MESSAGEBOX("Введите surname", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text3.Value) THEN
    MESSAGEBOX("Введите age", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text4.Value) THEN
    MESSAGEBOX("Введите room", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text5.Value) THEN
    MESSAGEBOX("Введите check_in", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text6.Value) THEN
    MESSAGEBOX("Введите departure", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите id_med_inf", 0, "Внимание")
    RETURN
ENDIF

IF EMPTY(thisform.Combol.Value) THEN
    MESSAGEBOX("Введите дату рождения", 0, "Внимание")
    RETURN
ENDIF

SELECT * FROM visitors WHERE id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO
CURSOR vis

SELECT vis
LOCATE FOR id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF NOT FOUND() THEN
    INSERT INTO visitors VALUES;
    (thisform.Text1.Value,;
    thisform.Text2.Value,;
    CAST(thisform.Text3.Value AS Integer),;
    CAST(thisform.Text4.Value AS Integer),;
    thisform.Combol.Value,;
```

```

        thisform.Text5.Value,;
        thisform.Text6.Value,;
        CAST(thisform.Text7.Value AS Integer))

MESSAGEBOX("Посетитель успешно добавлен!", 0, "Добавление")

ELSE
    MESSAGEBOX("Посетитель уже существует с таким id_med_inf!", 0, "Внимание")
ENDIF

thisform.Text1.Value = ''
thisform.Text2.Value = ''
thisform.Text3.Value = ''
thisform.Text4.Value = ''
thisform.Text5.Value = ''
thisform.Text6.Value = ''
thisform.Text7.Value = ''
thisform.Combo1.Value = ''

THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "c:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Удалить посетителя»

Листинг кнопки «Найти»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name as name_dig, diets.name as name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
        thisform.Label19.Caption = vis_2.Departure
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''

```



```

        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
ELSE
    thisform.Text7.Value = ''
    thisform.Label13.Caption = ''
    thisform.Label14.Caption = ''
    thisform.Label15.Caption = ''
    thisform.Label16.Caption = ''
    thisform.Label17.Caption = ''
    thisform.Label18.Caption = ''
    thisform.Label19.Caption = ''
    thisform.Label20.Caption = ''
    thisform.Label21.Caption = ''
    thisform.Label22.Caption = ''
    thisform.Label23.Caption = ''
    MESSAGEBOX("id не найден", 0, "Внимание")
ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "c:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Изменить посетителя»

Листинг кнопки «Изменить»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name as name_dig, diets.name as name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
        thisform.Label19.Caption = vis_2.Departure
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
    
```

```

        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
    ELSE
        thisform.Text7.Value = ''
        thisform.Label13.Caption = ''
        thisform.Label14.Caption = ''
        thisform.Label15.Caption = ''
        thisform.Label16.Caption = ''
        thisform.Label17.Caption = ''
        thisform.Label18.Caption = ''
        thisform.Label19.Caption = ''
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("id не найден", 0, "Внимание")
    ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "c:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Просмотр посетителей по диагнозам»

Листинг кнопки «Найти»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name as name_dig, diets.name as name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
        thisform.Label19.Caption = vis_2.Departure
        thisform.Label20.Caption = ''
    ENDIF
ENDIF

```

```

        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
ELSE
    thisform.Text7.Value = ''
    thisform.Label13.Caption = ''
    thisform.Label14.Caption = ''
    thisform.Label15.Caption = ''
    thisform.Label16.Caption = ''
    thisform.Label17.Caption = ''
    thisform.Label18.Caption = ''
    thisform.Label19.Caption = ''
    thisform.Label20.Caption = ''
    thisform.Label21.Caption = ''
    thisform.Label22.Caption = ''
    thisform.Label23.Caption = ''
    MESSAGEBOX("id не найден", 0, "Внимание")
ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "с:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Просмотр посетителей по диетам»

Листинг кнопки «Найти»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name AS name_dig, diets.name AS name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
        thisform.Label19.Caption = vis_2.Departure
        thisform.Label20.Caption = ''
    
```

```

        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
    ELSE
        thisform.Text7.Value = ''
        thisform.Label13.Caption = ''
        thisform.Label14.Caption = ''
        thisform.Label15.Caption = ''
        thisform.Label16.Caption = ''
        thisform.Label17.Caption = ''
        thisform.Label18.Caption = ''
        thisform.Label19.Caption = ''
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("id не найден", 0, "Внимание")
    ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "c:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Просмотр диагнозов»

Листинг кнопки «Назад»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name as name_dig, diets.name as name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
        thisform.Label19.Caption = vis_2.Departure
    
```

```

        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
    ELSE
        thisform.Text7.Value = ''
        thisform.Label13.Caption = ''
        thisform.Label14.Caption = ''
        thisform.Label15.Caption = ''
        thisform.Label16.Caption = ''
        thisform.Label17.Caption = ''
        thisform.Label18.Caption = ''
        thisform.Label19.Caption = ''
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("id не найден", 0, "Внимание")
    ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "c:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Форма «Найти посетителя»

Листинг кнопки «Найти»:

```

IF EMPTY(thisform.Text7.Value) THEN
    MESSAGEBOX("Введите номер мед книжки", 0, "Внимание")
    RETURN
ENDIF

SELECT visitors.name, visitors.surname, visitors.age, visitors.room, visitors.sex,;
visitors.departure, visitors.check_in, med_info.id_med_inf, med_info.id_diagnos,;
med_info.weight, med_info.height, diagnosis.name as name_dig, diets.name as name_diet;
FROM visitors, med_info, diagnosis, diets WHERE visitors.Id_med_inf =
med_info.Id_med_inf;
AND med_info.id_diagnos = diagnosis.id_diagnos AND diagnosis.id_diet = diets.id_diet;
AND visitors.Id_med_inf = CAST(thisform.Text7.Value AS Integer) INTO CURSOR vis

SELECT * FROM visitors INTO CURSOR vis_2

SELECT vis
LOCATE FOR vis.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
IF FOUND() THEN
    thisform.Label13.Caption = vis.Name
    thisform.Label14.Caption = vis.Surname
    thisform.Label15.Caption = CAST(vis.Age AS Char(10))
    thisform.Label16.Caption = CAST(vis.Room AS Char(10))
    thisform.Label17.Caption = vis.Sex
    thisform.Label18.Caption = vis.Check_in
    thisform.Label19.Caption = vis.Departure
    thisform.Label20.Caption = CAST(vis.weight AS Char(10))
    thisform.Label21.Caption = CAST(vis.height AS Char(10))
    thisform.Label22.Caption = vis.name_dig
    thisform.Label23.Caption = vis.name_diet
ELSE
    SELECT vis_2
    LOCATE FOR vis_2.Id_med_inf = CAST(thisform.Text7.Value AS Integer)
    IF FOUND() THEN
        thisform.Label13.Caption = vis_2.Name
        thisform.Label14.Caption = vis_2.Surname
        thisform.Label15.Caption = CAST(vis_2.Age AS Char(10))
        thisform.Label16.Caption = CAST(vis_2.Room AS Char(10))
        thisform.Label17.Caption = vis_2.Sex
        thisform.Label18.Caption = vis_2.Check_in
    
```

```

        thisform.Label19.Caption = vis_2.Departure
        thisform.Label20.Caption = ''
        thisform.Label21.Caption = ''
        thisform.Label22.Caption = ''
        thisform.Label23.Caption = ''
        MESSAGEBOX("У пользователя с таким id нет мед книжки", 0, "Внимание")
ELSE
    thisform.Text7.Value = ''
    thisform.Label13.Caption = ''
    thisform.Label14.Caption = ''
    thisform.Label15.Caption = ''
    thisform.Label16.Caption = ''
    thisform.Label17.Caption = ''
    thisform.Label18.Caption = ''
    thisform.Label19.Caption = ''
    thisform.Label20.Caption = ''
    thisform.Label21.Caption = ''
    thisform.Label22.Caption = ''
    thisform.Label23.Caption = ''
    MESSAGEBOX("id не найден", 0, "Внимание")
ENDIF
ENDIF
THISFORM.REFRESH()

```

Листинг кнопки «Назад»:

```

DO FORM "с:\users\дарина\desktop\бд\бд\start_form.scx"
RELEASE THISFORM

```

Литература:

1. Людмила Омельченко «Самоучитель Visual Foxpro 8» (Дата написания: 2003 Объем: 674 стр.)
2. <http://www.vvet.narod.ru/foxpro/vfp03.html>
3. <https://softclipper.net/foxpro/cpravochnik-po-visual-foxpro/>
4. <http://codenet.ru/db/vfp/foxbasics.php>