**ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ОЛЕСЯ ГОНЧАРА**

**Факультет прикладної математики**

**Кафедра математичного забезпечення ЕОМ**

**Звіт**

з практичної роботи № 1

з курсу «Бази даних»

на тему: «Розробка бази даних в СУБД MS SQL Server з графічним багатовіконним користувацьким інтерфейсом»

Виконав:

студент 3-го курсу

групи ПЗ-21у-1

Войцехов М.О.

м. Дніпро

2022 р.

1. Постановка задачі.

Розробити базу даних, засобами СУБД MS SQL Server, що забезпечує введення, перегляд, редагування й виконання складних запитів користувача для декількох зв'язаних таблиць.

Опис предметної області

Ви є керівником служби планування Приватної клініки. Вашим завданням є відстеження показників роботи клініки. У клініці працюють лікарі різних спеціальностей, що мають різну кваліфікацію. Щодня в клініку звертаються хворі (пацієнти). Усі вони проходять обов'язкову реєстрацію, при якій у базу даних заносяться стандартні анкетні дані (прізвище, ім'я, по батькові, рік народження). Кожний пацієнт може звертатися в клініку кілька раз, потребуючи на різну медичну допомогу. Усі звернення пацієнтів фіксуються, при цьому встановлюється діагноз, визначається вартість лікування, запам'ятовується дата звернення.

Таблиці

Лікарі (Код лікаря, Прізвище, Ім'я, По батькові, Спеціальність, Категорія). Пацієнти (Код пацієнта, Прізвище, Ім'я, По батькові, Рік народження).Звернення (Код звернення, Код лікаря, Код пацієнта, Дата, Діагноз, Вартість лікування).

2. Опис ходу розв'язання.

2.1 Побудова таблиць, структура таблиць

Створено базу даних натиснувши правою кнопкою миші по папці “Бази даних” у MS SQL Server Management Studio, введено назву БД “Hospital”, натиснувши правою кнопкою миші по папці “Таблиці” створено наступні таблиці БД, показано на рисунках 2.1-2.3.

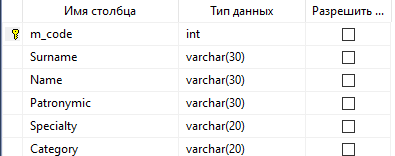


Рисунок 2.1 – Створення таблиці Medic

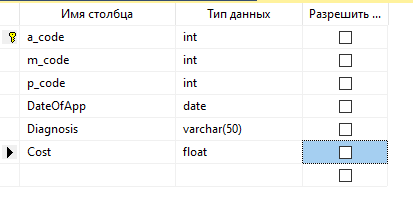


Рисунок 2.2 – Створення таблиці Appeal

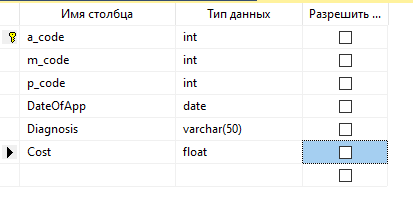


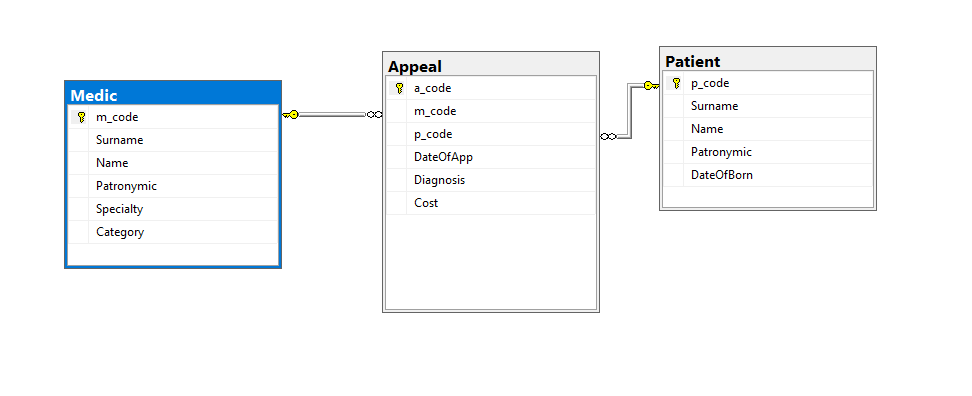
Рисунок 2.3 – Створення таблиці Patient

2.2. Встановлення зв'язків

В таблиці Medicє полеm\_code(int) яке є **primary key autoincrement**, аналогічно у таблиці Patient є полеp\_code(int) яке також **primary key autoincrement.**

У талиціAppealполя m\_code (int), та p\_code(int) є **foreign key.**

2.3. Схема зв'язків



2.4. Проектування та реалізація запитів

**Запит 1**

Виведення усіх записів з таблиці Лікарі(Medic). Однотабличний запит.

Програмний код 2.1 – Запит 1 в MS SQL Server

SELECT [m\_code]

,[Surname]

,[Name]

,[Patronymic]

,[Specialty]

,[Category]

FROM [Hospital].[dbo].[Medic]

Програмний код 2.2 – Запит 1 на SQL мові

SELECT \*FROM Medic

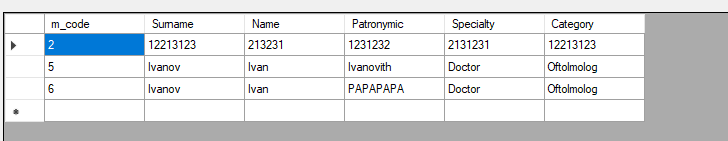


Рисунок 2.4 – Виконання запиту 1

**Запит 2**

Виведення записів лікарів, які відповідають відповідній спеціальності. Однотабличний запит.

Програмний код 2.3 – Запит 2 в MS SQL Server

SELECT [m\_code]

,[Surname]

,[Name]

,[Patronymic]

,[Specialty]

,[Category]

FROM [Hospital].[dbo].[Medic]

where [Specialty] = 'Doctor'

Програмний код 2.4 – Запит 2 на SQL мові

SELECT \*FROM Medic where Specialty = 'Doctor'

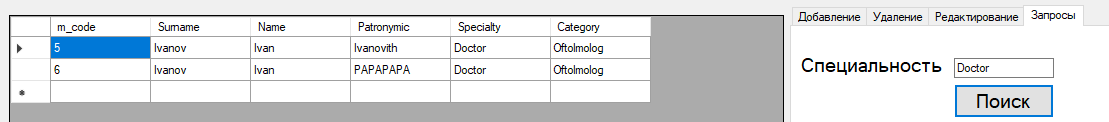


Рисунок 2.5 – Виконання запиту 2

**Запит 3**

Виведення усіх записів з таблиці Паціенти. Однотабличний запит.

Програмний код 2.5 – Запит 3 в MS SQL Server

SELECT [p\_code]

,[Surname]

,[Name]

,[Patronymic]

,[DateOfBorn]

FROM [Hospital].[dbo].[Patient]

Програмний код 2.6 – Запит 3 на SQL мові

SELECT \*FROM Patient

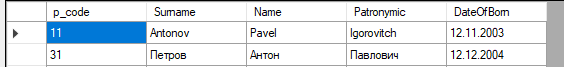


Рисунок 2.6 – Виконання запиту 3

**Запит 4**

Виведення пацієнтів, які народились в певний проміжок років. Однотабличний запит.

Програмний код 2.7 – Запит 4 в MS SQL Server

SELECT [p\_code]

,[Surname]

,[Name]

,[Patronymic]

,[DateOfBorn]

FROM [Hospital].[dbo].[Patient]

where YEAR([DateOfBorn]) BETWEEN 2000 AND 2004

Програмний код 2.8 – Запит 4 на SQL мові

SELECT \*FROM Patient where YEAR(DateOfBorn) BETWEEN 2000 AND 2004"

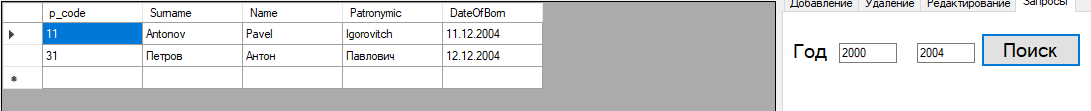


Рисунок 2.7 – Виконання запиту 4

**Запит 5**

Виведення пацієнтів, за певним Прізвищем. Однотабличний запит.

Програмний код 2.9 – Запит 5 в MS SQL Server

SELECT [p\_code]

,[Surname]

,[Name]

,[Patronymic]

,[DateOfBorn]

FROM [Hospital].[dbo].[Patient]

Where [Surname] = 'Antonov'

Програмний код 2.10 – Запит 5 на SQL мові

SELECT \*FROM Patient where Surname = 'Antonov'

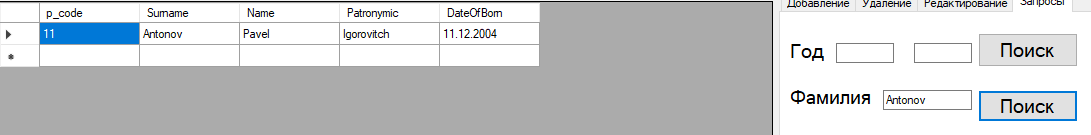


Рисунок 2.8 – Виконання запиту 5

**Запит 6**

Визначити кількість паціентів в кожному місяці 2022 року. Тип запиту однотабличний з агрегатною функцією.

Програмний код 2.11 – Запит 6 в MS SQL Server

SELECT count(\*)

, MONTH([DateOfApp])

as 'месяц'

from [Hospital].[dbo].[Appeal]

where year([DateOfApp]) = 2022

group by month([DateOfApp])

Програмний код 2.12 – Запит 6 на SQL мові

SELECT count(\*) as 'Количество', MONTH(DateOfApp) as 'месяц' from Appeal where year(DateOfApp) = 2022 group by month(DateOfApp)

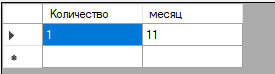


Рисунок 2.9 – Виконання запиту 6

**Запит 7**

Визначити кількість працівників по кожній посаді. Запит з групуванням.

Програмний код 2.13 – Запит 7 в MS SQL Server

SELECT count(\*) as 'Количество'

, [Specialty] as 'Специальность'

from [Hospital].[dbo].[Medic]

group by [Specialty]

Програмний код 2.14 – Запит 7 на SQL мові

SELECT count(\*) as 'Количество', Specialty as 'Специальность' from Medic group by Specialty

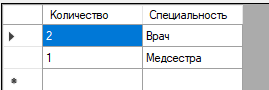


Рисунок 2.10 – Виконання запиту 7

**Запит 8**

Виведення кількості записів в таблиці Лікарі. Запит з агрегатною функцією.

Програмний код 2.15 – Запит 8 в MS SQL Server

SELECT COUNT(m\_code)

FROM [Hospital].[dbo].[Medic]

Програмний код 2.16 – Запит 8 на SQL мові

SELECT COUNT(m\_code) FROM Medic

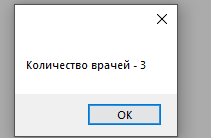


Рисунок 2.11 – Виконання запиту 8

**Запит 9**

Вивести інформацію по пацієнтам, яким було поставлено діагноз «Перелом».

Тип запиту багатотабличний.

Програмний код 2.17 – Запит 9 в MS SQL Server

SELECT [Patient].[p\_code], [Surname], [Name], [Patronymic], [DateOfBorn] FROM [Hospital].[dbo].[Patient] join [Hospital].[dbo].[Appeal] on ([Appeal].[p\_code] = [Patient].[p\_code]) and ([Appeal].[Diagnosis] = ‘Перелом’)

Програмний код 2.18 – Запит 9 на SQL мові

SELECT Patient.p\_code, Surname, Name, Patronymic, DateOfBorn FROM Patient join Appeal on (Appeal.p\_code = Patient.p\_code) and (Appeal.Diagnosis = ‘Перелом’)



Рисунок 2.12 – Виконання запиту 9

**Запит 10**

Вивести інформацію по пацієнтам, яких лікував лікар з прізвищем «Павлов».

Тип запиту багатотабличний.

Програмний код 2.19 – Запит 10 в MS SQL Server

SELECT \*FROM [Hospital].[dbo].[Patient], [Hospital].[dbo].[Appeal], [Hospital].[dbo].[ Medic] where ([Patient].[p\_code] = [Appeal].[p\_code]) and ([Medic].[m\_code] = [Appeal].[m\_code]) and ([Medic].[Surname] = 'Павлов')

Програмний код 2.20 – Запит 10 на SQL мові

SELECT \*FROM Patient, Appeal, Medic where (Patient.p\_code = Appeal.p\_code) and (Medic.m\_code = Appeal.m\_code) and (Medic.Surname = 'Павлов')



Рисунок 2.13 – Виконання запиту 10

**Запит 11**

Вивести суму заробітку та дані про лікаря с прізвищем «Павлов».

Тип запиту багатотабличний.

Програмний код 2.21 – Запит 11 в MS SQL Server

SELECT sum([Appeal].[Cost]), [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic] from [Hospital].[dbo].[Appeal], [Hospital].[dbo].[Medic] where [Medic].[m\_code] = [Appeal].[m\_code] and [Medic].[m\_code] = (select [Medic].[m\_code] from Medic where [Medic].[Surname] = 'Павлов') group by [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic]

Програмний код 2.22 – Запит 11 на SQL мові

SELECT sum(Appeal.Cost), Medic.Surname, Medic.Name, Medic.Patronymic from Appeal, Medic where Medic.m\_code = Appeal.m\_code and Medic.m\_code = (select Medic.m\_code from Medic where Medic.Surname = 'Павлов') group by Medic.Surname, Medic.Name, Medic.Patronymic

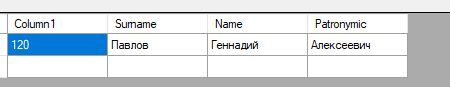


Рисунок 2.14 – Виконання запиту 11

**Запит 12**

Вивести кількість паціентів у кожного лікаря.

Тип запиту багатотабличний.

Програмний код 2.23 – Запит 12 в MS SQL Server

SELECT [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic], count([Patient].[p\_code]) as 'Всего больных' from [Medic], [Appeal], [Patient] where [Medic].[m\_code] = [Appeal].[m\_code] and [Patient].[p\_code] = [Appeal].[p\_code] GROUP BY [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic]

Програмний код 2.24 – Запит 12 на SQL мові

SELECT Medic.Surname, Medic.Name, Medic.Patronymic, count(Patient.p\_code) as 'Всего больных' from Medic, Appeal, Patient where Medic.m\_code = Appeal.m\_code and Patient.p\_code = Appeal.p\_code GROUP BY Medic.Surname, Medic.Name, Medic.Patronymic

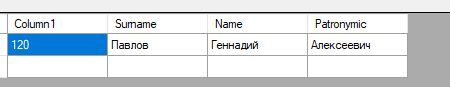


Рисунок 2.15 – Виконання запиту 12

**Запит 13**

Вивести кількість лікарів у котрих побував пацієнт.

Тип запиту багатотабличний.

Програмний код 2.25 – Запит 13 в MS SQL Server

SELECT [Patient].[Surname], [Patient].[Name], [Patient].[Patronymic], count([Medic].[m\_code]) as 'Всего врачей' from [Medic], [Appeal], [Patient] where [Patient].[p\_code] = [Appeal].[p\_code] and [Medic].[m\_code] = [Appeal].[m\_code] GROUP BY [Patient].[Surname], [Patient].[Name], [Patient].[Patronymic]

Програмний код 2.26 – Запит 13 на SQL мові

SELECT Patient.Surname, Patient.Name, Patient.Patronymic, count(Medic.m\_code) as 'Всего врачей' from Medic, Appeal, Patient where Patient.p\_code = Appeal.p\_code and Medic.m\_code = Appeal.m\_code GROUP BY Patient.Surname, Patient.Name, Patient.Patronymic



Рисунок 2.15 – Виконання запиту 13

**Запит 14**

Вивести пацієнтів які потратили загальну суму на лікування більшу за 6000.

Тип запиту багатотабличний.

Програмний код 2.27 – Запит 14 в MS SQL Server

SELECT [Patient].[Surname], [Patient].[Name], [Patient].[Patronymic], sum([Appeal].[Cost]) as 'Потраченно' from [Appeal], [Patient] where [Patient].[p\_code] = [Appeal].[p\_code] group by [Patient].[Surname], [Patient].[Name], [Patient].[Patronymic] having sum([Appeal].[Cost]) > 6000

Програмний код 2.28 – Запит 14 на SQL мові

SELECT Patient.Surname, Patient.Name, Patient.Patronymic, sum(Appeal.Cost) as 'Потраченно' from Appeal, Patient where Patient.p\_code = Appeal.p\_code group by Patient.Surname, Patient.Name, Patient.Patronymic having sum(Appeal.Cost) > 6000

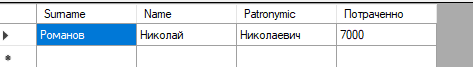


Рисунок 2.16 – Виконання запиту 14

**Запит 15**

Вивести лікарів, які заробили більше ніж 2000.

Тип запиту багатотабличний.

Програмний код 2.29 – Запит 15 в MS SQL Server

SELECT [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic], sum([Appeal].[Cost]) as 'Заработано' from [Appeal], [Medic] where [Medic].[m\_code] = [Appeal].[m\_code] group by [Medic].[Surname], [Medic].[Name], [Medic].[Patronymic] having sum([Appeal].[Cost]) > 2000"

Програмний код 2.30 – Запит 15 на SQL мові

SELECT Medic.Surname, Medic.Name, Medic.Patronymic, sum(Appeal.Cost) as 'Заработано' from Appeal, Medic where Medic.m\_code = Appeal.m\_code group by Medic.Surname, Medic.Name, Medic.Patronymic having sum(Appeal.Cost) > 2000"

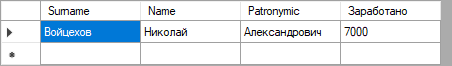


Рисунок 2.17 – Виконання запиту 15

2.5. Звіти

За допомоги бібліотеки iTextSharp зробленозбереження даних запросів до pdf файлу. В даному випадку зберігаються дані певного пацієнта.

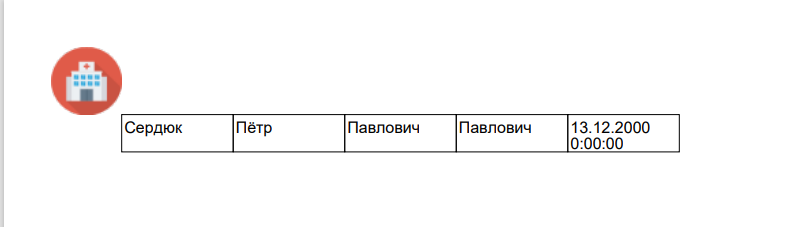


Рисунок 2.18 – Виконання звіту

1. Опис інтерфейсу (керівництво користувача).

При старті програми відкривається 3 вікна, в яких є можливість працювати з таблицями. На кожному вікні знаходиться таблиця та з правої сторони вкладки, переключаючи які можливо робити різні дії.

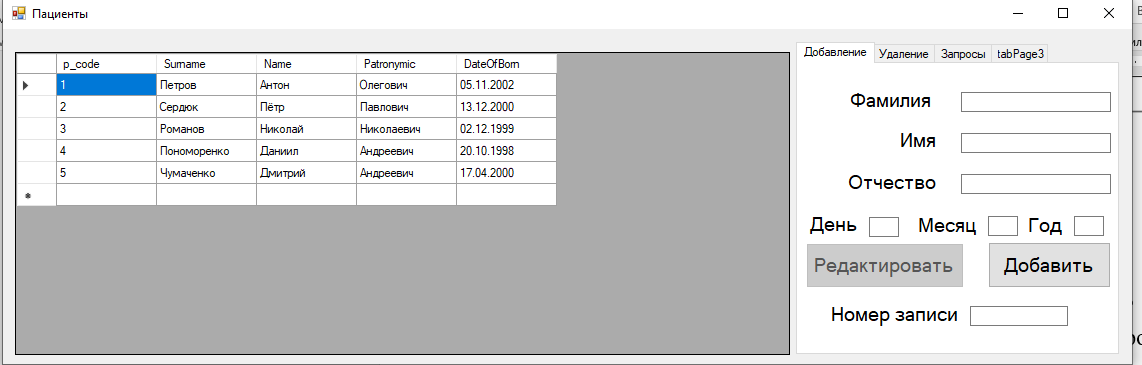


Рисунок 2.19 – Вікно пацієнтів

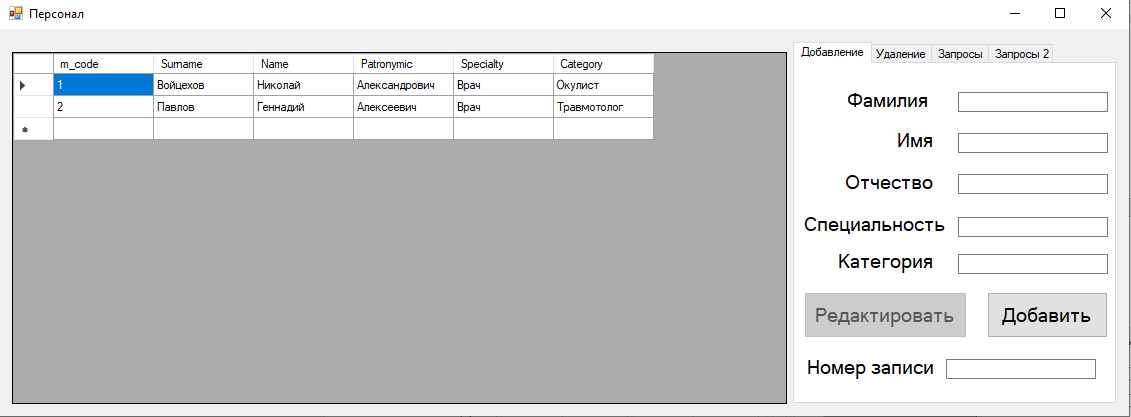


Рисунок 2.20 – Вікно працівників

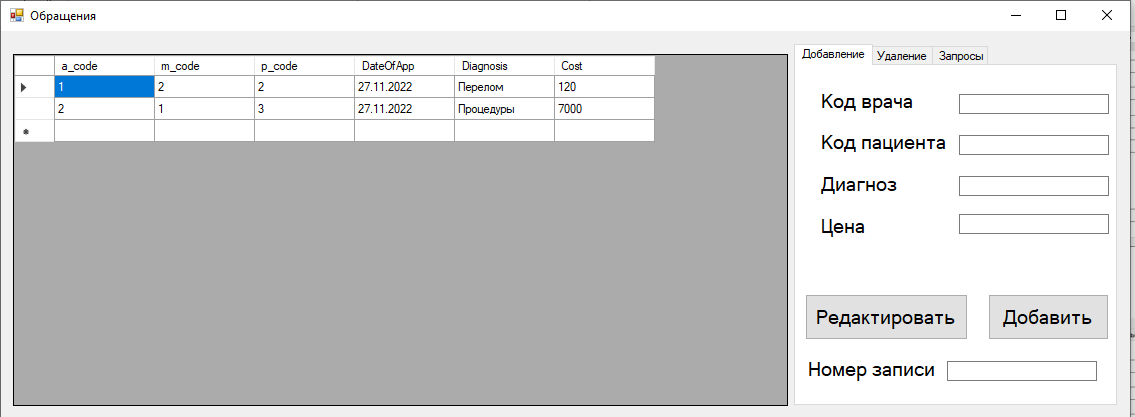


Рисунок 2.21 – Вікно звертань

4. Тестування з коментарями



Рисунок 2.22 – Кількість пацієнтів

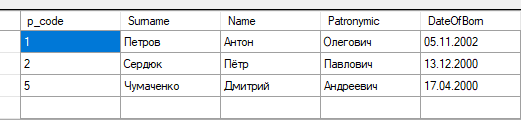


Рисунок 2.23 – Пошук пацієнтів з діапазоном року народження 2000-2002



Рисунок 2.24 – Пошук пацієнтів за прізвищем «Петров»

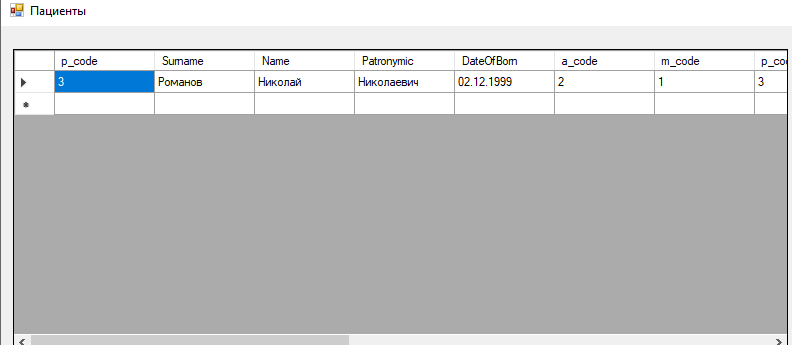


Рисунок 2.25 – Пошук пацієнтів яких лікував лікар з прізвищем «Войцехов»



Рисунок 2.26 – Пошук лікарів з спеціальністю «Врач»

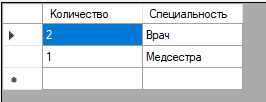


Рисунок 2.27 – Кількість лікарів за кожною спеціальністю

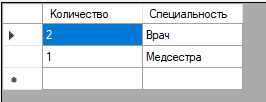


Рисунок 2.27 – Кількість лікарів за кожною спеціальністю

5. Висновки ( за кожним пунктом завдання)

Було здобуто навички з створення баз даних, та роботою з MS SQL Management Studio. Реалізацією багатотабличних запросів та застосування, відображення їх у візуальному програмному додатку WSForms.