Практична робота № 2

**Тема**: Програмування на QT.

**Мета**: Навчитись розробляти об'єктно-орієнтовані програми на QT під ОС Windows

Завдання:

**Завдання 1:**

Розробити програму, яка в консольному (!) режимі виводить "Hello, world!"

**Завдання 2:**

Розробити програму, яка в консольному (або візуальному) режимі запитує в користувача якісь числові дані (наприклад, номер дня тижня або номер місяця) та виводить на екран текстове повідомлення з його назвою.

**Завдання 3:**

Розробити програму для знаходження сідлових точок прямокутної матриці. (Сідловою точкою матриці будемо називати елемент, який одночасно є і найменшим в своєму стовпці і найбільшим в своєму рядку, та навпаки (одночасно найбільшим в своєму стовпці і найменшим в своєму рядку). Сідлових точок в матриці може бути декілька або навіть не бути жодної) Для відображення матриці на екрані використати таблицю. Введення даних - надати і можливіть користувачу вводити з клавіатрури, і наявність кнопки "Згенерувати випадкові значення". Також має бути кнопка "Очистити значення елементів". Користувач може змінювати розміри (кількість рядків та кількість стовпців) матриці - для цього використати або поля введення з кнопками підтвердження, або лічильники типу UpDown (поля введення з кнопками-"трикутниками", які роблять +1 чи -1), або "повзунки", або інший спосіб. Результат виконання пошуку (який відбувається після натиснення кнопки "Пошук") повинен відображатися і на екрані (або виведенням окремого вікна з текстовим повідомленням, або знайдені елементи (чи комірки з ними) змінюють колір/шрифт), і в файлі (ім'я файлу можна брати за умовчуванням і не запитувати в користувача; в файл потрібно спочатку вивести елементи матриці, по якій проводився пошук, а далі вже результати пошуку); якщо в матриці сідлових точок немає, то на екран (і в файл) виводиться відповідне повідомлення.

**Завдання 4:**

Описати клас для збереження інформації про прямокутник (довжина, ширина, координати ОДНІЄЇ вершини), сторони якого паралельні осям координат. Додатково до методів для збереження/зміни значень полів класу, клас також повинен містити метод обчислення площі, метод для зміни розмірів прямокутника, метод для переміщення прямокутника на площині на певну відстань по обом координатам, операцію визначення прямокутника, який є спільною частиною двох прямокутників, над якими виконується дана операція, та операцію визначення найменшого прямокутника, в межах якого вміщаються два прямокутника, над якими виконується дана операція.

Написати програму для демонстрації можливостей розробленого класу.

**Завдання 5:**

Скласти програму яка містить наступну інформацію про книги в бібліотеці:

* + Номер УДК
  + Прізвище і ініціали автора
  + Назву
  + Рік видавництва
  + Кількість екземплярів книги в бібліотеці

Програма повинна забезпечити:

* Початкове формування даних про всі книги в бібліотеці у вигляді списку
* При взятті кожної книги вводиться номер УДК, і програма зменшує значення кількості книг на одиницю або видає повідомлення про те що необхідної книги немає в бібліотеці або, що дана книга знаходиться на руках
* При поверненні кожної книги вводиться номер УДК і програма збільшує значення кількості книг на одиницю
* По запиту виводиться інформація про наявність книг у бібліотеці

Хід роботи:

**Завдання 1:**

Код програми:

main.cpp:

#include<QDebug>

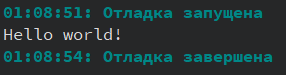
int **main**()

{

qDebug() << "Hello world!\n";

*return* 0;

}

Приклад роботи програми 1:

Завдання 2:

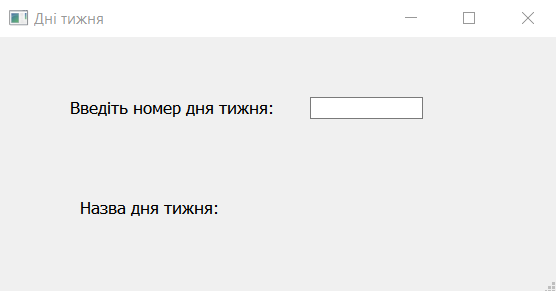


Рисунок 1 – Основне вікно

*<?xml* version="1.0" encoding="UTF-8"*?>*

*<ui* version="4.0"*>*

*<class>*MainWindow*</class>*

*<widget* class="QMainWindow" name="MainWindow"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*558*</width>*

*<height>*258*</height>*

*</rect>*

*</property>*

*<property* name="windowTitle"*>*

*<string>*Дні тижня*</string>*

*</property>*

*<widget* class="QWidget" name="centralwidget"*>*

*<widget* class="QLabel" name="label"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*70*</x>*

*<y>*50*</y>*

*<width>*211*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Введіть номер дня тижня:*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_2"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*80*</x>*

*<y>*150*</y>*

*<width>*211*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Назва дня тижня:*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_3"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*320*</x>*

*<y>*155*</y>*

*<width>*161*</width>*

*<height>*21*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<family>*Franklin Gothic Heavy*</family>*

*<pointsize>*10*</pointsize>*

*<italic>*true*</italic>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="lineEdit"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*310*</x>*

*<y>*60*</y>*

*<width>*113*</width>*

*<height>*22*</height>*

*</rect>*

*</property>*

*<property* name="maxLength"*>*

*<number>*1*</number>*

*</property>*

*</widget>*

*</widget>*

*<widget* class="QMenuBar" name="menubar"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*558*</width>*

*<height>*25*</height>*

*</rect>*

*</property>*

*</widget>*

*<widget* class="QStatusBar" name="statusbar"*/>*

*</widget>*

*<resources/>*

*<connections/>*

*</ui>*

Основне вікно програми:

mainwindow.h

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

QT\_BEGIN\_NAMESPACE

*namespace* **Ui** { *class* **MainWindow**; }

QT\_END\_NAMESPACE

*class* **MainWindow** : *public* QMainWindow

{

Q\_OBJECT

*public*:

**MainWindow**(QWidget \*parent = *nullptr*);

~***MainWindow***();

*private* slots:

void **on\_textEdit\_textChanged**();

void **on\_lineEdit\_textChanged**(*const* QString &arg1);

*private*:

Ui::MainWindow \*ui;

};

#endif *//* *MAINWINDOW\_H*

mainwindow.cpp

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include<QIntValidator>

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(*new* Ui::MainWindow)

{

ui->setupUi(*this*);

ui->lineEdit->setValidator(*new* QIntValidator(1, 7, *this*));

}

MainWindow::~***MainWindow***()

{

*delete* ui;

}

void MainWindow::**on\_lineEdit\_textChanged**(*const* QString &arg1)

{

int number = ui->lineEdit->text().toInt();

*switch* (number) {

*case* 1:

{

ui->label\_3->setText("Понеділок");

*break*;

}

*case* 2:

{

ui->label\_3->setText("Вівторок");

*break*;

}

*case* 3:

{

ui->label\_3->setText("Середа");

*break*;

}

*case* 4:

{

ui->label\_3->setText("Четвер");

*break*;

}

*case* 5:

{

ui->label\_3->setText("П'ятниця");

*break*;

}

*case* 6:

{

ui->label\_3->setText("Субота");

*break*;

}

*case* 7:

{

ui->label\_3->setText("Неділя");

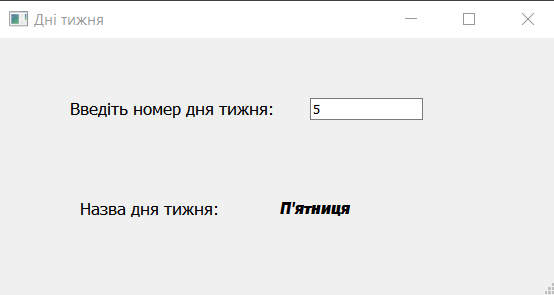
*break*;

}

*default*: ui->label\_3->setText("Немає такого дня тижня");

}

}

***Приклад роботи програми 2:***

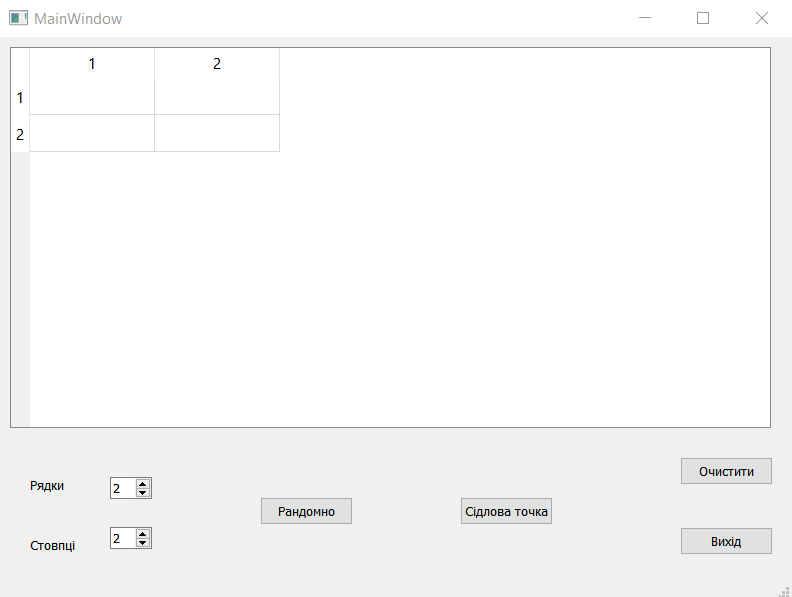
**Завдання 3:**

Рисунок 1 – Основне вікно

*<?xml* version="1.0" encoding="UTF-8"*?>*

*<ui* version="4.0"*>*

*<class>*MainWindow*</class>*

*<widget* class="QMainWindow" name="MainWindow"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*792*</width>*

*<height>*563*</height>*

*</rect>*

*</property>*

*<property* name="windowTitle"*>*

*<string>*MainWindow*</string>*

*</property>*

*<property* name="autoFillBackground"*>*

*<bool>*false*</bool>*

*</property>*

*<widget* class="QWidget" name="centralwidget"*>*

*<widget* class="QPushButton" name="pushButton"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*680*</x>*

*<y>*420*</y>*

*<width>*93*</width>*

*<height>*28*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Очистити*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*30*</x>*

*<y>*440*</y>*

*<width>*55*</width>*

*<height>*16*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Рядки*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_2"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*30*</x>*

*<y>*500*</y>*

*<width>*55*</width>*

*<height>*16*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Стовпці*</string>*

*</property>*

*</widget>*

*<widget* class="QSpinBox" name="spinBox"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*110*</x>*

*<y>*440*</y>*

*<width>*42*</width>*

*<height>*22*</height>*

*</rect>*

*</property>*

*<property* name="minimum"*>*

*<number>*2*</number>*

*</property>*

*<property* name="value"*>*

*<number>*2*</number>*

*</property>*

*</widget>*

*<widget* class="QSpinBox" name="spinBox\_2"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*110*</x>*

*<y>*490*</y>*

*<width>*42*</width>*

*<height>*22*</height>*

*</rect>*

*</property>*

*<property* name="minimum"*>*

*<number>*2*</number>*

*</property>*

*<property* name="value"*>*

*<number>*2*</number>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="pushButton\_2"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*260*</x>*

*<y>*460*</y>*

*<width>*93*</width>*

*<height>*28*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Рандомно*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="pushButton\_3"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*460*</x>*

*<y>*460*</y>*

*<width>*93*</width>*

*<height>*28*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Cідлова точка*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="pushButton\_4"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*680*</x>*

*<y>*490*</y>*

*<width>*93*</width>*

*<height>*28*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Вихід*</string>*

*</property>*

*</widget>*

*<widget* class="QTableWidget" name="tableWidget"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*10*</x>*

*<y>*10*</y>*

*<width>*761*</width>*

*<height>*381*</height>*

*</rect>*

*</property>*

*</widget>*

*</widget>*

*<widget* class="QMenuBar" name="menubar"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*792*</width>*

*<height>*25*</height>*

*</rect>*

*</property>*

*</widget>*

*<widget* class="QStatusBar" name="statusbar"*/>*

*</widget>*

*<resources/>*

*<connections/>*

*</ui>*

Основне вікно:

mainwindow.h:

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include<QStandardItemModel>

#include <QFile>

#include<QTextStream>

#include<climits>

#include<QMessageBox>

#include<QTableWidget>

#include "cells.h"

QT\_BEGIN\_NAMESPACE

*namespace* **Ui** { *class* **MainWindow**; }

QT\_END\_NAMESPACE

*class* **MainWindow** : *public* QMainWindow

{

Q\_OBJECT

*public*:

Cells cell;

*public*:

**MainWindow**(QWidget \*parent = *nullptr*);

~***MainWindow***();

*private* slots:

void **CreateEmptyItems**(int I, int J);

void **on\_spinBox\_valueChanged**(int arg1);

void **on\_spinBox\_2\_valueChanged**(int arg1);

void **on\_pushButton\_4\_clicked**();

void **on\_pushButton\_2\_clicked**();

void **on\_pushButton\_3\_clicked**();

void **on\_pushButton\_clicked**();

*private*:

Ui::MainWindow \*ui;

QStandardItemModel \*model;

};

#endif *//* *MAINWINDOW\_H*

mainwindow.cpp:

#include "mainwindow.h"

#include "ui\_mainwindow.h"

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(*new* Ui::MainWindow)

{

ui->setupUi(*this*);

ui->tableWidget->setRowCount(ui->spinBox->value());

ui->tableWidget->setColumnCount(ui->spinBox\_2->value());

CreateEmptyItems(0, 0);

}

MainWindow::~***MainWindow***()

{

*delete* ui;

}

void MainWindow::**CreateEmptyItems**(int I, int J)

{

*for*(int i = I; i < ui->tableWidget->rowCount(); i++)

*for*(int j = J; j < ui->tableWidget->columnCount(); j++)

ui->tableWidget->setItem(i, j, *new* QTableWidgetItem*(*""*)*);

}

void MainWindow::**on\_spinBox\_valueChanged**(int arg1)

{

ui->tableWidget->setRowCount(ui->spinBox->value());

CreateEmptyItems(cell.RowCount, 0);

cell.RowCount = ui->spinBox->value();

}

void MainWindow::**on\_spinBox\_2\_valueChanged**(int arg1)

{

ui->tableWidget->setColumnCount(ui->spinBox\_2->value());

CreateEmptyItems(0, cell.ColumnCount);

cell.ColumnCount = ui->spinBox\_2->value();

}

*//Random*

void MainWindow::**on\_pushButton\_2\_clicked**()

{

int rows, columns;

rows = ui->tableWidget->rowCount();

columns = ui->tableWidget->columnCount();

int Arr[rows][columns];

*for* (int i = 0; i < rows ; i++)

*for*(int j = 0; j < columns; j++)

{

Arr[i][j] = rand() % 9 + 1;

ui->tableWidget->setItem(i, j,

*new* QTableWidgetItem*(*QString*::number(Arr[i][j]))*);

}

}

*//Sidlo*

void MainWindow::**on\_pushButton\_3\_clicked**()

{

int rows, columns;

rows = ui->tableWidget->rowCount();

columns = ui->tableWidget->columnCount();

int Arr[rows][columns];

int minInRow[rows];

int maxInRow[rows];

int minInColumn[columns];

int maxInColumn[columns];

bool empty = *false*;

*for*(int i = 0; i < rows; i++)

*for*(int j = 0; j < columns; j++)

*if*(ui->tableWidget->item(i, j)->text().isEmpty())

empty = *true*;

*if*(!empty)

{

*//Init* *Array*

*for* (int i = 0; i < rows ; i++)

*for*(int j = 0; j < columns; j++)

{

Arr[i][j] = ui->tableWidget->item(i, j)->text().toInt();

ui->tableWidget->item(i, j)->setBackground(QColor::fromRgb(255, 255, 255));

}

*//Writting*

QFile file("D:\\PracticeQT\\Task3\\out.txt");

QTextStream out(*&file*);

*if*(file.*open*(QIODevice::*WriteOnly* | QIODevice::*Append* | QIODevice::*Text*))

{

*for* (int i = 0; i < rows ; i++)

{

*for*(int j = 0; j < columns; j++)

out << Arr[i][j]<<" ";

out<<endl;

}

}

*//Alorithm*

bool MinMax = *false*;

bool MaxMin = *false*;

out<<"Sidlovi tochky: ";

*for* (int i = 0; i < rows; i++)

{

minInRow[i] = INT\_MAX;

*for* (int j = 0; j < columns; j++)

{

*if* (Arr[i][j] < minInRow[i])

{

minInRow[i] = Arr[i][j];

}

}

}

*for* (int j = 0; j < columns; j++)

{

maxInColumn[j] = INT\_MIN;

*for* (int i = 0; i < rows; i++)

{

*if* (Arr[i][j] > maxInColumn[j])

{

maxInColumn[j] = Arr[i][j];

}

}

}

*for* (int i = 0; i < rows; i++)

{

*for* (int j = 0; j < columns; j++)

{

*if* (minInRow[i] == maxInColumn[j])

{

out<<Arr[i][j]<<"["<<i+1<<"]"<<"["<<j+1<<"]"<<" ";

ui->tableWidget->item(i, j)->setBackground(QColor::fromRgb(52, 235, 229));

MinMax = *true*;

}

}

}

*if*(!MinMax)

{

*for* (int i = 0; i < columns; i++)

{

minInColumn[i] = INT\_MAX;

*for* (int j = 0; j < rows; j++)

{

*if* (Arr[i][j] < minInColumn[i])

{

minInColumn[i] = Arr[i][j];

}

}

}

*for* (int j = 0; j < rows; j++)

{

maxInRow[j] = INT\_MIN;

*for* (int i = 0; i < columns; i++)

{

*if* (Arr[i][j] > maxInRow[j])

{

maxInRow[j] = Arr[i][j];

}

}

}

*for* (int i = 0; i < rows; i++)

{

*for* (int j = 0; j < columns; j++)

{

*if* (minInRow[i] == maxInColumn[j])

{

out<<Arr[i][j]<<"["<<i+1<<"]"<<"["<<j+1<<"]"<<" ";

ui->tableWidget->item(i, j)->setBackground(QColor::fromRgb(52, 235, 229));

MaxMin = *true*;

}

}

}

}

*if*(!MinMax && !MaxMin)

{

QMessageBox::information(*this*, "Info ", "Сідлових точок немає");

out<<"nemae";

}

out<<"\n\n";

file.*close*();

}

*else*

QMessageBox::information(*this*, "Info ", "Заповніть всі клітинки!");

}

*//Clear*

void MainWindow::**on\_pushButton\_clicked**()

{

ui->tableWidget->clear();

CreateEmptyItems(0, 0);

}

*//Exit*

void MainWindow::**on\_pushButton\_4\_clicked**()

{

close();

}

Клас для збереження інформації про попередній розмір таблиці:

cells.h

#ifndef CELLS\_H

#define CELLS\_H

*class* **Cells**

{

*public*:

int RowCount;

int ColumnCount;

*public*:

**Cells**();

~**Cells**();

};

#endif *//* *CELLS\_H*

cells.cpp

#include "cells.h"

Cells::**Cells**()

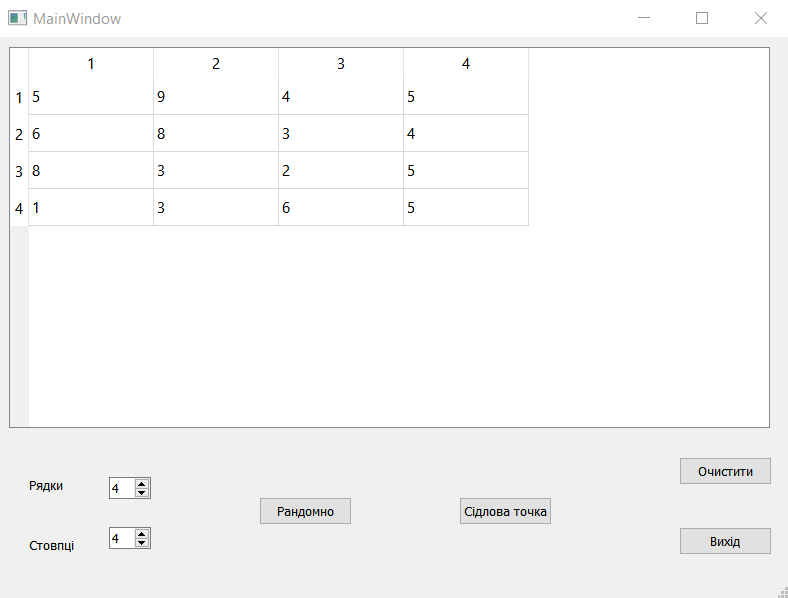
{

RowCount = 2;

ColumnCount = 2;

}

Cells::~**Cells**(){}

**Приклад роботи програми 3**:  
  
  
Рисунок 2 – Заповнення таблиці

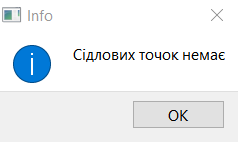


Рисунок 3 – Повідомлення якщо точок не знайдено

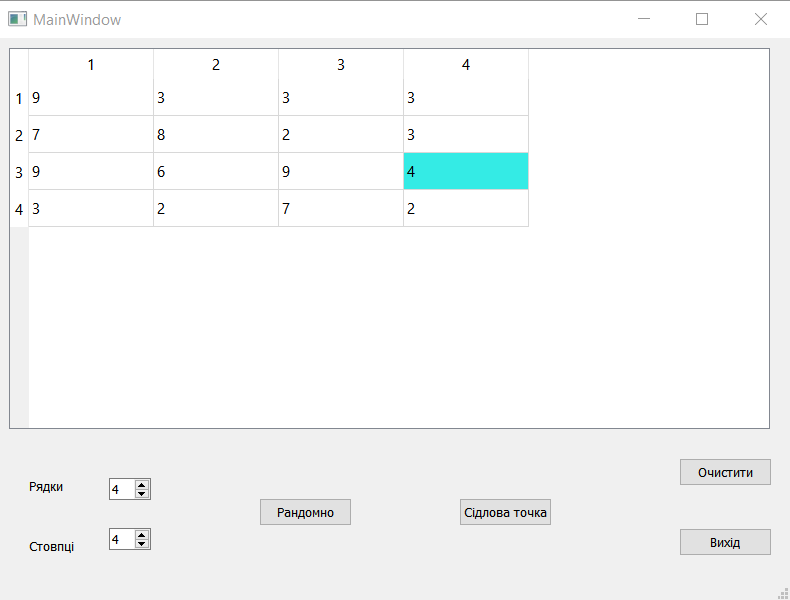


Рисунок 4 – Результат пошуку сідлових точок

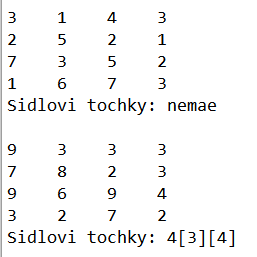
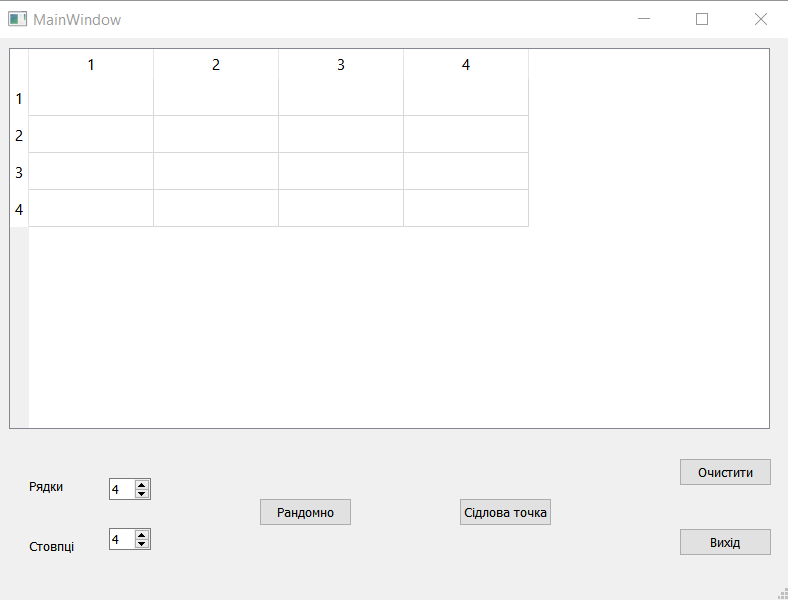


Рисунок 5 – Запис результатів у файл

  
Рисунок 6 – Результат роботи кнопки очищення

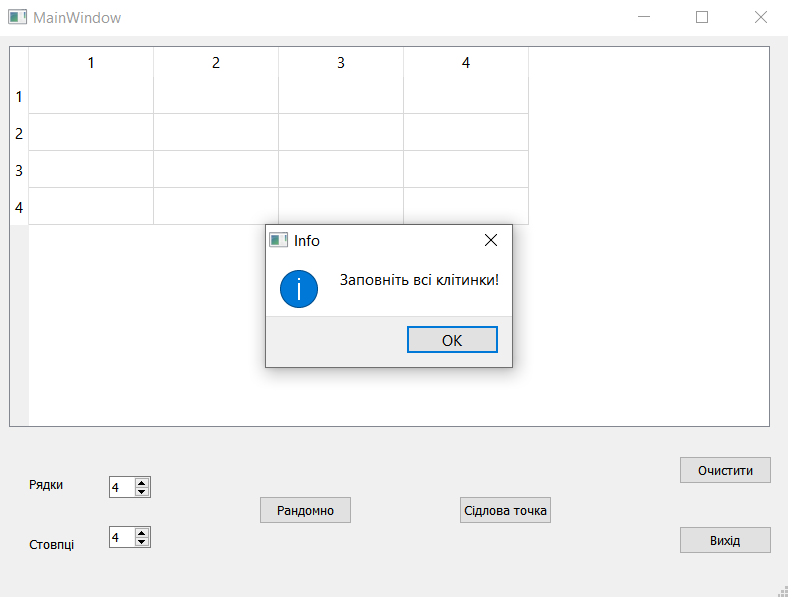


Рисунок 7 – При спробі знайти сідлові точки, коли таблиця не заповнена повністю, виводиться повідомлення

**Завдання 4**:

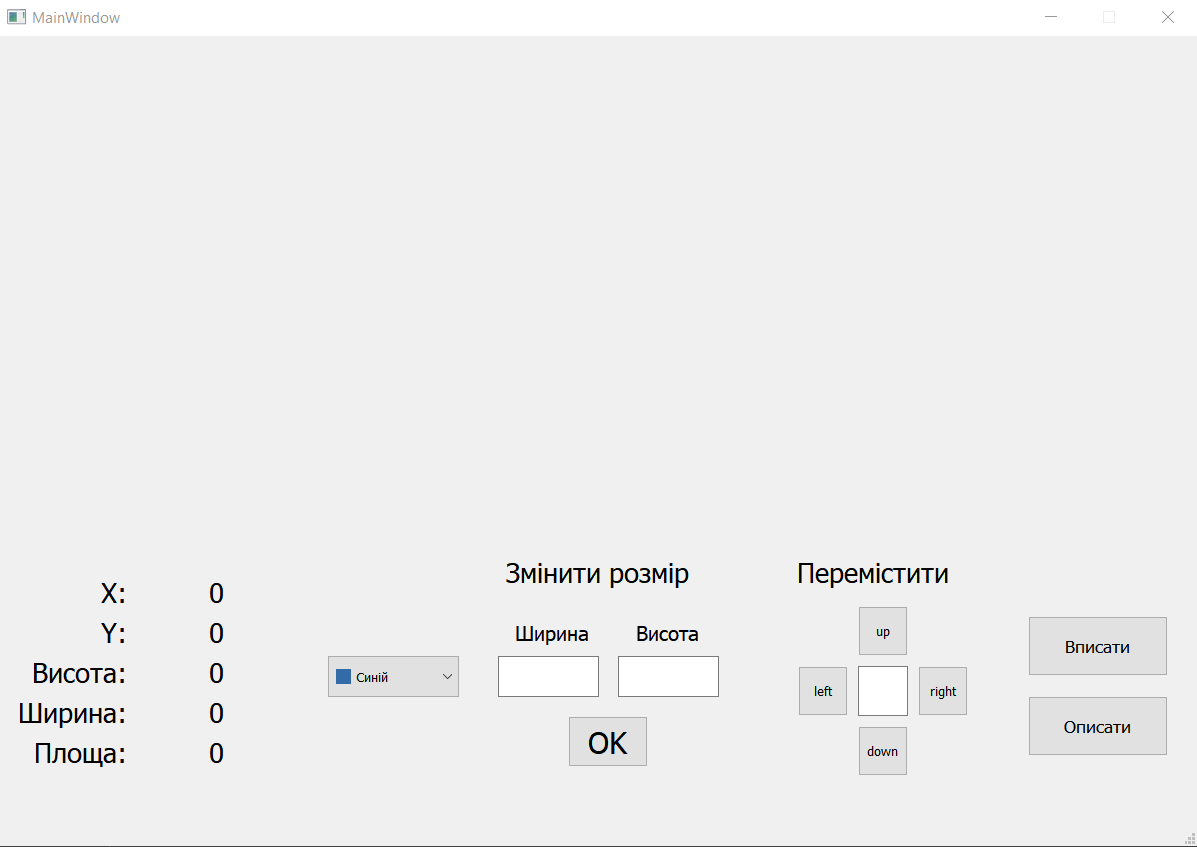


Рисунок 1 – Основне вікно

**Код програми:**Основне вікно:

*<?xml* version="1.0" encoding="UTF-8"*?>*

*<ui* version="4.0"*>*

*<class>*MainWindow*</class>*

*<widget* class="QMainWindow" name="MainWindow"*>*

*<property* name="windowModality"*>*

*<enum>*Qt::NonModal*</enum>*

*</property>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*1200*</width>*

*<height>*810*</height>*

*</rect>*

*</property>*

*<property* name="sizePolicy"*>*

*<sizepolicy* hsizetype="Fixed" vsizetype="Fixed"*>*

*<horstretch>*0*</horstretch>*

*<verstretch>*0*</verstretch>*

*</sizepolicy>*

*</property>*

*<property* name="minimumSize"*>*

*<size>*

*<width>*1200*</width>*

*<height>*810*</height>*

*</size>*

*</property>*

*<property* name="maximumSize"*>*

*<size>*

*<width>*1200*</width>*

*<height>*810*</height>*

*</size>*

*</property>*

*<property* name="windowTitle"*>*

*<string>*MainWindow*</string>*

*</property>*

*<widget* class="QWidget" name="centralwidget"*>*

*<widget* class="QWidget" name="widget" native="true"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*1200*</width>*

*<height>*500*</height>*

*</rect>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="widthLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*500*</x>*

*<y>*620*</y>*

*<width>*101*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="heightLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*620*</x>*

*<y>*620*</y>*

*<width>*101*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="okBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*570*</x>*

*<y>*680*</y>*

*<width>*80*</width>*

*<height>*51*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*18*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*OK*</string>*

*</property>*

*</widget>*

*<widget* class="QComboBox" name="comboBox"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*330*</x>*

*<y>*620*</y>*

*<width>*131*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="currentIndex"*>*

*<number>*-1*</number>*

*</property>*

*<property* name="maxVisibleItems"*>*

*<number>*4*</number>*

*</property>*

*<property* name="maxCount"*>*

*<number>*4*</number>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="twoBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*1030*</x>*

*<y>*660*</y>*

*<width>*140*</width>*

*<height>*60*</height>*

*</rect>*

*</property>*

*<property* name="sizePolicy"*>*

*<sizepolicy* hsizetype="Minimum" vsizetype="Fixed"*>*

*<horstretch>*0*</horstretch>*

*<verstretch>*0*</verstretch>*

*</sizepolicy>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Описати*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="oneBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*1030*</x>*

*<y>*580*</y>*

*<width>*140*</width>*

*<height>*60*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Вписати*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="upBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*860*</x>*

*<y>*570*</y>*

*<width>*50*</width>*

*<height>*50*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*up*</string>*

*</property>*

*<property* name="iconSize"*>*

*<size>*

*<width>*20*</width>*

*<height>*20*</height>*

*</size>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="downBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*860*</x>*

*<y>*690*</y>*

*<width>*50*</width>*

*<height>*50*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*down*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="leftBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*800*</x>*

*<y>*630*</y>*

*<width>*50*</width>*

*<height>*50*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*left*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="rightBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*920*</x>*

*<y>*630*</y>*

*<width>*50*</width>*

*<height>*50*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*right*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_6"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*500*</x>*

*<y>*520*</y>*

*<width>*191*</width>*

*<height>*33*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Змінити розмір*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_7"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*510*</x>*

*<y>*580*</y>*

*<width>*81*</width>*

*<height>*33*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Ширина*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_8"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*630*</x>*

*<y>*580*</y>*

*<width>*71*</width>*

*<height>*33*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Висота*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="distanceLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*860*</x>*

*<y>*630*</y>*

*<width>*50*</width>*

*<height>*50*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="maxLength"*>*

*<number>*3*</number>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_9"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*790*</x>*

*<y>*520*</y>*

*<width>*161*</width>*

*<height>*33*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Перемістити*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QWidget" name="layoutWidget"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*20*</x>*

*<y>*540*</y>*

*<width>*281*</width>*

*<height>*197*</height>*

*</rect>*

*</property>*

*<layout* class="QFormLayout" name="formLayout"*>*

*<item* row="0" column="0"*>*

*<layout* class="QVBoxLayout" name="verticalLayout\_2"*>*

*<item>*

*<widget* class="QLabel" name="label"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*X:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_2"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Y:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_4"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Висота:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_3"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Ширина:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_5"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Площа:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*</item>*

*</layout>*

*</item>*

*<item* row="0" column="1"*>*

*<layout* class="QVBoxLayout" name="verticalLayout\_3"*>*

*<item>*

*<widget* class="QLabel" name="xLb"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*0*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="yLb"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*0*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="widthLb"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*0*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="heightLb"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*0*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="squareLb"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*16*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*0*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*</layout>*

*</item>*

*</layout>*

*</widget>*

*</widget>*

*<widget* class="QMenuBar" name="menubar"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*1200*</width>*

*<height>*25*</height>*

*</rect>*

*</property>*

*</widget>*

*<widget* class="QStatusBar" name="statusbar"*/>*

*</widget>*

*<resources/>*

*<connections/>*

*</ui>*

Основне вікно:

mainwindow.h:

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <QPainter>

#include <QPushButton>

#include <QMessageBox>

#include "rectangle.h"

QT\_BEGIN\_NAMESPACE

*namespace* **Ui** { *class* **MainWindow**; }

QT\_END\_NAMESPACE

*class* **MainWindow** : *public* QMainWindow

{

Q\_OBJECT

*public*:

Rectangle rect[4];

*public*:

**MainWindow**(QWidget \*parent = *nullptr*);

~***MainWindow***();

void **AddItem**();

*protected*:

void ***paintEvent***(QPaintEvent \*event);

*private* slots:

void **allUpdate**();

void **on\_comboBox\_activated**();

void **on\_okBtn\_clicked**();

void **on\_oneBtn\_clicked**();

void **on\_twoBtn\_clicked**();

void **on\_rightBtn\_clicked**();

void **on\_leftBtn\_clicked**();

void **on\_upBtn\_clicked**();

void **on\_downBtn\_clicked**();

*private*:

Ui::MainWindow \*ui;

};

#endif *//* *MAINWINDOW\_H*

mainwindow.cpp:

#include "mainwindow.h"

#include "ui\_mainwindow.h"

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(*new* Ui::MainWindow)

{

ui->setupUi(*this*);

AddItem();

}

MainWindow::~***MainWindow***()

{

*delete* ui;

}

void MainWindow::**AddItem**()

{

QPixmap px(15,15);

px.fill(QColor(50, 107, 168));

QIcon icon(px);

ui->comboBox->addItem(icon,"Синій");

px.fill(QColor(237, 19, 44));

icon.addPixmap(px);

ui->comboBox->addItem(icon,"Червоний");

}

void MainWindow::***paintEvent***(QPaintEvent \*event)

{

Q\_UNUSED(event);

QPainter painter(*this*);

QBrush Brushes[4] = {QBrush(QColor(50, 107, 168, 45)),

QBrush(QColor(237, 19, 44, 45)),

QBrush(Qt::*BDiagPattern*),

QBrush(Qt::*FDiagPattern*)};

*for*(int i = 0; i < 4; i++)

{

painter.setBrush(Brushes[i]);

painter.drawRect(rect[i].x, rect[i].y, rect[i].width, rect[i].height);

}

}

void MainWindow::**allUpdate**()

{

ui->xLb->setText(QString::number(rect[ui->comboBox->currentIndex()].x));

ui->yLb->setText(QString::number(rect[ui->comboBox->currentIndex()].y));

ui->widthLb->setText(QString::number(rect[ui->comboBox->currentIndex()].width));

ui->heightLb->setText(QString::number(rect[ui->comboBox->currentIndex()].height));

ui->squareLb->setText(QString::number(rect[ui->comboBox->currentIndex()].Square()));

rect[2].SetData(0, 0, 0, 0);

rect[3].SetData(0, 0, 0, 0);

update();

}

void MainWindow::**on\_comboBox\_activated**()

{

allUpdate();

}

void MainWindow::**on\_okBtn\_clicked**()

{

*if*(ui->widthLE->text().isEmpty()

|| ui->heightLE->text().isEmpty()

|| ui->widthLE->text().toDouble() <= 0

|| ui->heightLE->text().toDouble() <= 0)

QMessageBox::information(*this*, "Підказка", "Заповніть всі поля значеннями більше нуля."

" Для виділення дробової частини використовуйте крапку");

*else*

{

rect[ui->comboBox->currentIndex()].Resize(

ui->widthLE->text().toDouble(),

ui->heightLE->text().toDouble());

ui->widthLE->setText(0);

ui->heightLE->setText(0);

allUpdate();

}

}

void MainWindow::**on\_oneBtn\_clicked**()

{

rect[3] = Rectangle::IntersectionRectangle(rect[0], rect[1]);

update();

}

void MainWindow::**on\_twoBtn\_clicked**()

{

rect[2] = Rectangle::MinRectangle(rect[0], rect[1]);

update();

}

void MainWindow::**on\_rightBtn\_clicked**()

{

rect[ui->comboBox->currentIndex()].Move('r', ui->distanceLb->text().toDouble());

allUpdate();

}

void MainWindow::**on\_leftBtn\_clicked**()

{

rect[ui->comboBox->currentIndex()].Move('l', ui->distanceLb->text().toDouble());

allUpdate();

}

void MainWindow::**on\_upBtn\_clicked**()

{

rect[ui->comboBox->currentIndex()].Move('u', ui->distanceLb->text().toDouble());

allUpdate();

}

void MainWindow::**on\_downBtn\_clicked**()

{

rect[ui->comboBox->currentIndex()].Move('d', ui->distanceLb->text().toDouble());

allUpdate();

}

Клас, що відбовідає за прямокутник:

rectangle.h

#ifndef RECTANGLE\_H

#define RECTANGLE\_H

#include <QString>

#include "math.h"

*class* **Rectangle**

{

*public*:

double x, y;

double width, height;

*public*:

**Rectangle**();

~**Rectangle**();

void **SetData**(double x, double y, double width, double height);

double **Square**();

void **Resize**(double width, double height);

void **Move**(char direction, double distance);

*static* bool **Intersection**(Rectangle rectA, Rectangle rectB);

*static* Rectangle **MinRectangle**(Rectangle rectA, Rectangle rectB);

*static* Rectangle **IntersectionRectangle**(Rectangle rectA, Rectangle rectB);

};

#endif *//* *RECTANGLE\_H*

rectangle.cpp

#include "rectangle.h"

Rectangle::**Rectangle**()

{

x = 0;

y = 0;

width = 0;

height = 0;

}

Rectangle::~**Rectangle**(){}

void Rectangle::**SetData**(double x, double y, double width, double height)

{

*this*->x = x;

*this*->y = y;

*this*->width = width;

*this*->height = height;

}

double Rectangle::**Square**()

{

*return* width\*height;

}

void Rectangle::**Resize**(double width, double height)

{

*this*->width = width;

*this*->height = height;

}

void Rectangle::**Move**(char direction, double distance)

{

*switch* (direction)

{

*case* 'u':{ *this*->y -= distance; *break*;}

*case* 'd':{ *this*->y += distance; *break*;}

*case* 'l':{ *this*->x -= distance; *break*;}

*case* 'r':{ *this*->x += distance; *break*;}

*default*: *break*;

}

}

bool Rectangle::**Intersection**(Rectangle rectA, Rectangle rectB)

{

*if*(((rectA.x + rectA.width) > rectB.x) && ((rectB.x + rectB.width)> rectA.x)

&& ((rectA.y + rectA.height) > rectB.y) && ((rectB.y + rectB.height)> rectA.y))

*return* *true*;

*return* *false*;

}

Rectangle Rectangle::**MinRectangle**(Rectangle rectA, Rectangle rectB)

{

Rectangle rect;

double xMin = 0, xMax = 0, yMin = 0,

yMax = 0, width = 0, height = 0,

xMaxWidth = 0, yMaxHeight = 0,

xEnd = 0, yEnd = 0;

*if*((rectA.width && rectB.width && rectA.height && rectB.height) !=0)

{

rectA.x < rectB.x ? xMin = rectA.x : xMin = rectB.x;

rectA.y < rectB.y ? yMin = rectA.y : yMin = rectB.y;

*if*(rectA.x + rectA.width < rectB.x + rectB.width)

{xMax = rectB.x; xMaxWidth = rectB.width;}

*else* {xMax = rectA.x; xMaxWidth = rectA.width;}

*if*(rectA.y + rectA.height < rectB.y + rectB.height)

{yMax = rectB.y; yMaxHeight = rectB.height;}

*else* {yMax = rectA.y; yMaxHeight = rectA.height;}

xEnd = xMax + xMaxWidth;

yEnd = yMax + yMaxHeight;

width = xEnd - xMin;

height = yEnd - yMin;

rect.SetData(xMin, yMin, width, height);

}

*return* rect;

}

Rectangle Rectangle::**IntersectionRectangle**(Rectangle rectA, Rectangle rectB)

{

Rectangle rect;

double xMin = 0, xMax = 0, yMin = 0,

yMax = 0, width = 0, height = 0,

xMinWidth = 0, yMinHeight = 0,

xEnd = 0, yEnd = 0;

*if*((rectA.width && rectB.width && rectA.height && rectB.height) !=0

&& Rectangle::Intersection(rectA, rectB) == *true*)

{

rectA.x < rectB.x ? xMax = rectB.x : xMax = rectA.x;

rectA.y < rectB.y ? yMax = rectB.y : yMax = rectA.y;

*if*(rectA.x + rectA.width < rectB.x + rectB.width)

{xMin = rectA.x; xMinWidth = rectA.width;}

*else* {xMin = rectB.x; xMinWidth = rectB.width;}

*if*(rectA.y + rectA.height < rectB.y + rectB.height)

{yMin = rectA.y; yMinHeight = rectA.height;}

*else* {yMin = rectB.y; yMinHeight = rectB.height;}

xEnd = xMin + xMinWidth;

yEnd = yMin + yMinHeight;

width = xEnd - xMax;

height = yEnd - yMax;

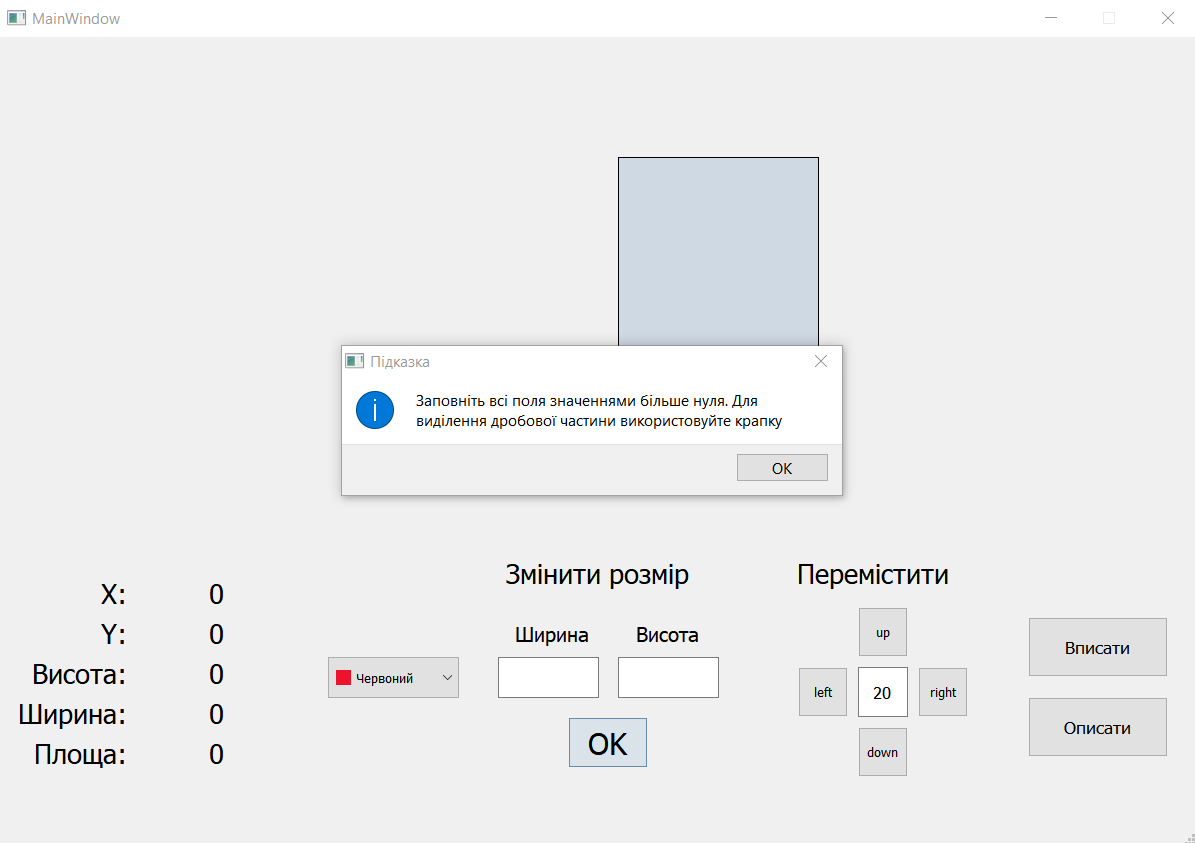
rect.SetData(xMax, yMax, width, height);

}

*return* rect;

}

**Приклад роботи програми 4**:

  
Рисунок 2 – Повідомлення при спробі змінити розміри прямокутника не заповнивши поля вводу значеннями

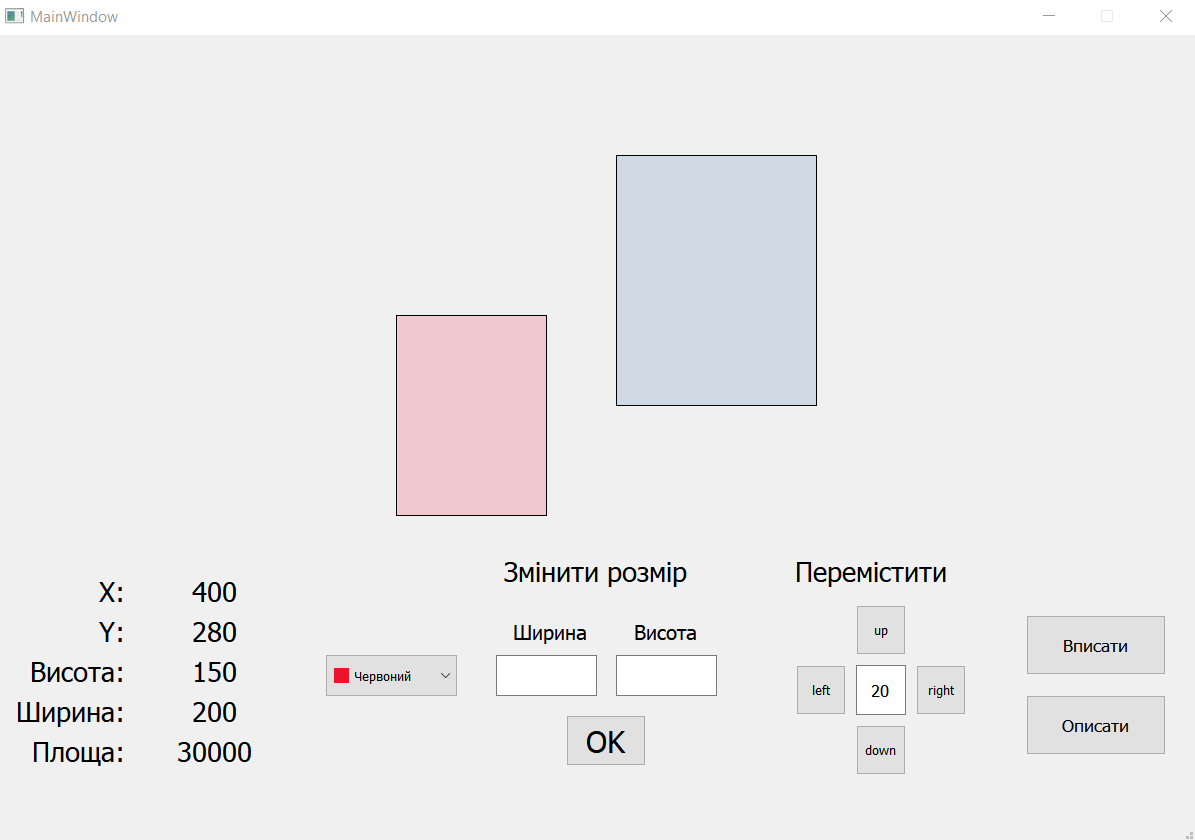


Рисунок 3 – Результат роботи функції переміщення

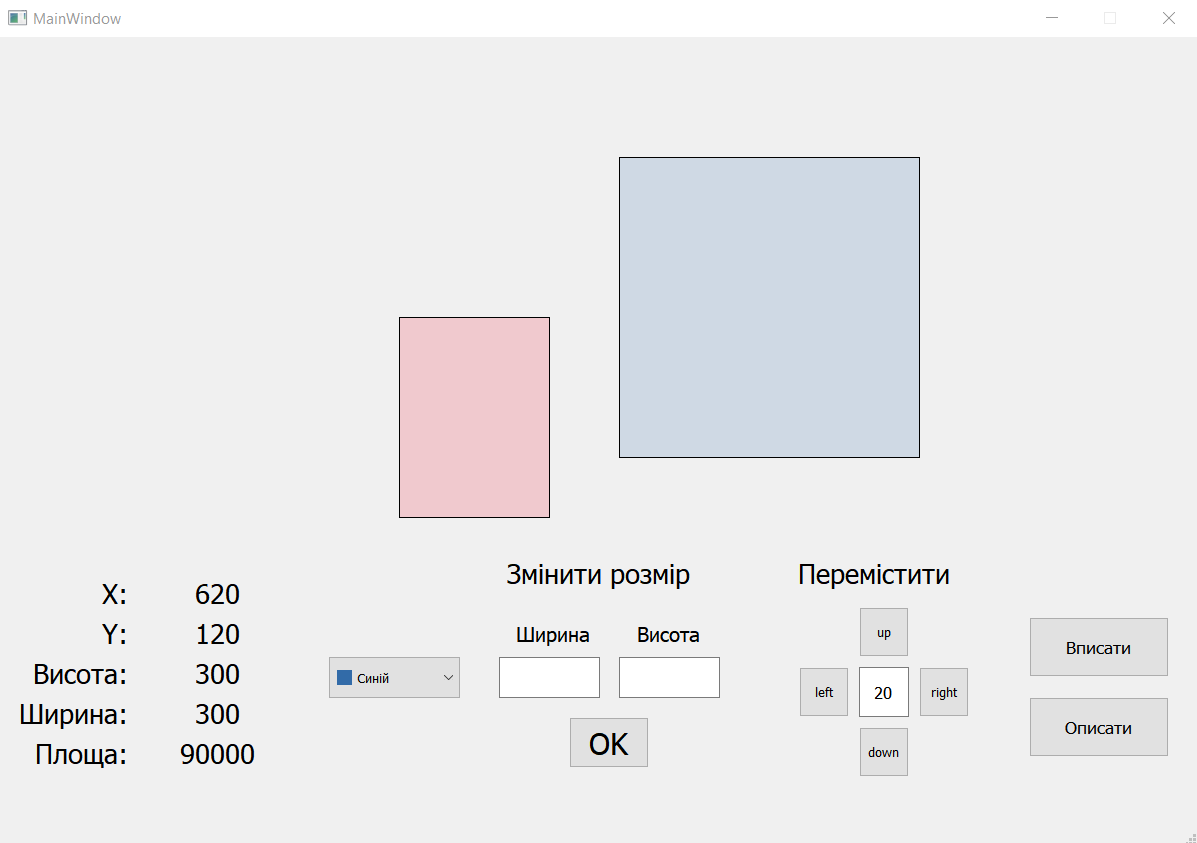


Рисунок 4 – Результат роботи функції зміни розміру прямокутника

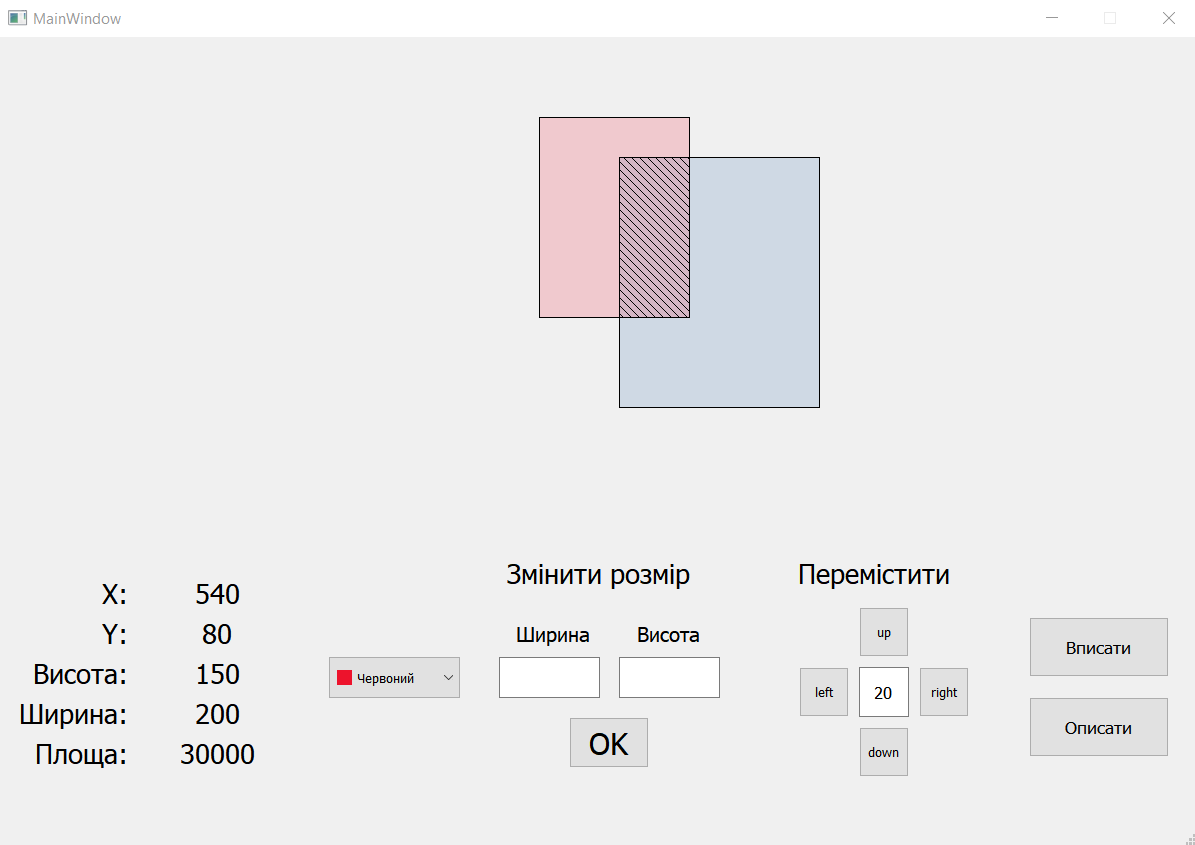


Рисунок 4 – Результат роботи функції знаходження вписаного прямокутника



Рисунок 5 – Результат роботи функції знаходження описаного прямокутника

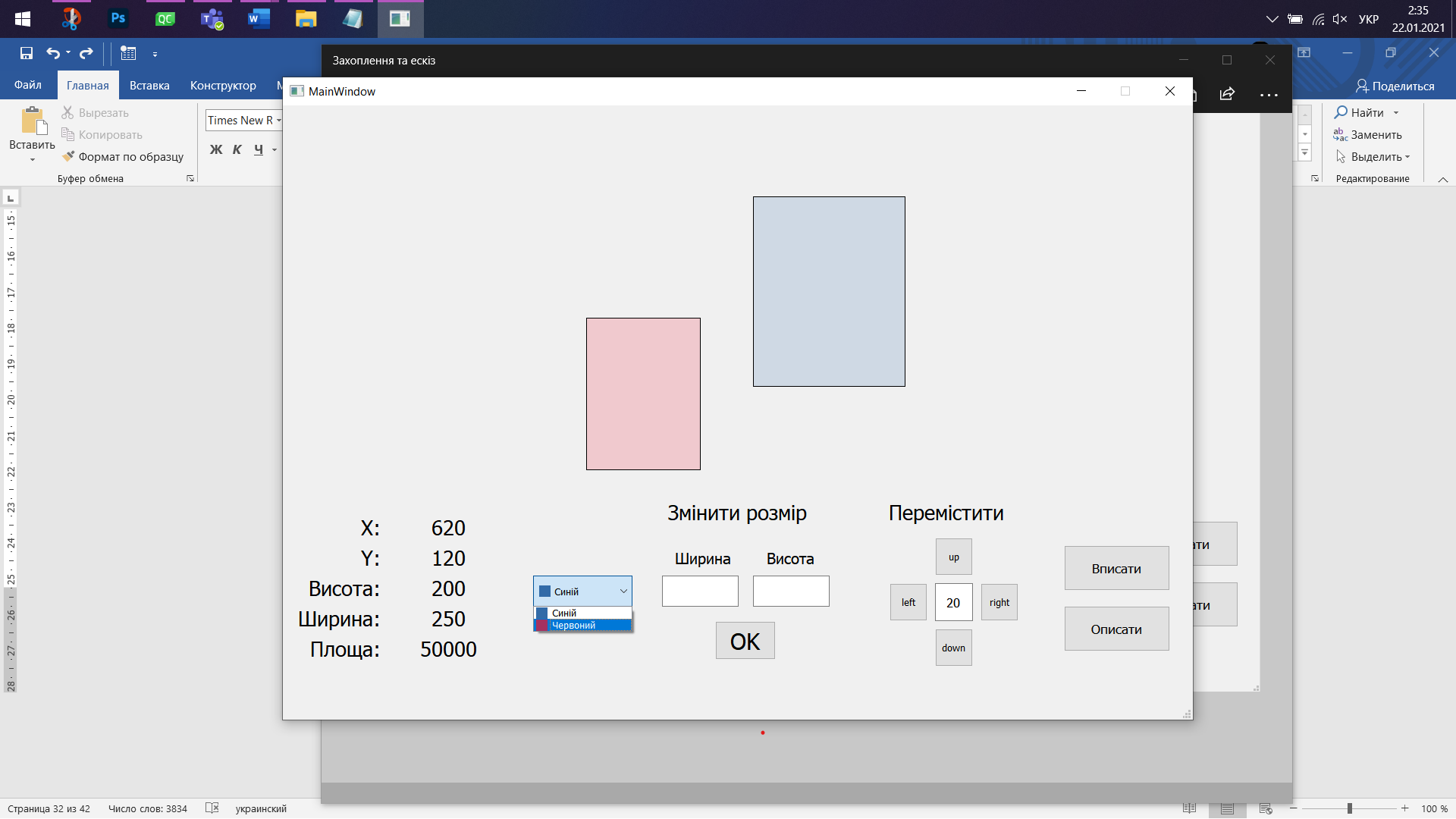
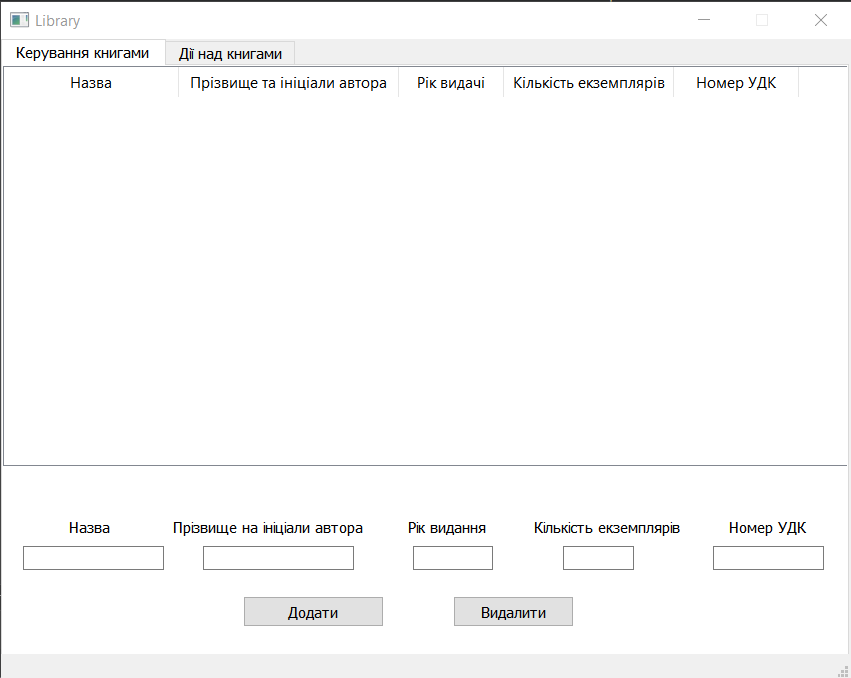


Рисунок 6 – Можливість зміни прямокутників для керування

**Завдання 5**:



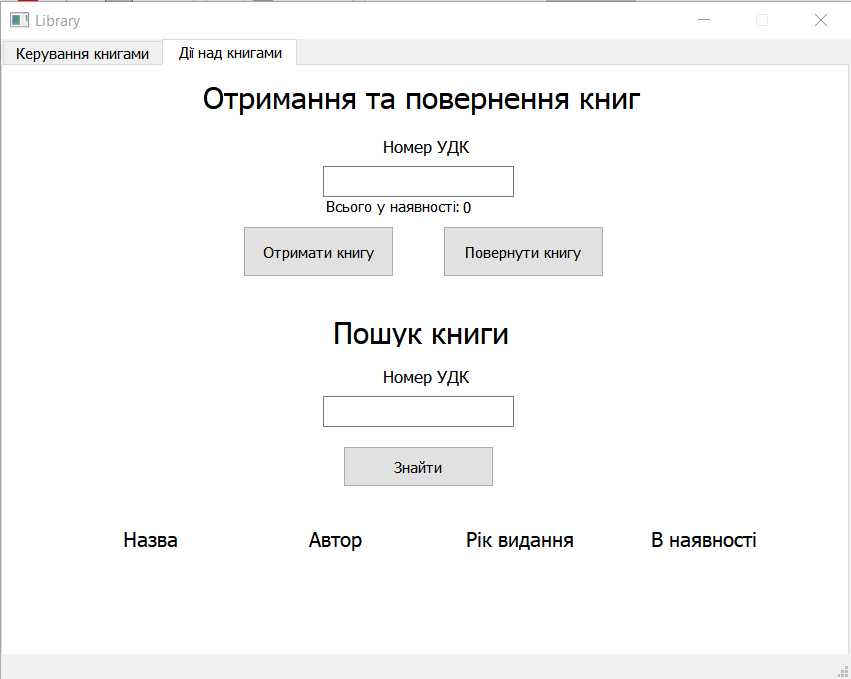


Рисунок 1-2 – Основне вікно

**Код програми:**

*<?xml* version="1.0" encoding="UTF-8"*?>*

*<ui* version="4.0"*>*

*<class>*Library*</class>*

*<widget* class="QMainWindow" name="Library"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*850*</width>*

*<height>*640*</height>*

*</rect>*

*</property>*

*<property* name="minimumSize"*>*

*<size>*

*<width>*850*</width>*

*<height>*640*</height>*

*</size>*

*</property>*

*<property* name="maximumSize"*>*

*<size>*

*<width>*850*</width>*

*<height>*640*</height>*

*</size>*

*</property>*

*<property* name="windowTitle"*>*

*<string>*Library*</string>*

*</property>*

*<widget* class="QWidget" name="centralwidget"*>*

*<widget* class="QTabWidget" name="tabWidget"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*850*</width>*

*<height>*650*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="tabPosition"*>*

*<enum>*QTabWidget::North*</enum>*

*</property>*

*<property* name="tabShape"*>*

*<enum>*QTabWidget::Rounded*</enum>*

*</property>*

*<property* name="currentIndex"*>*

*<number>*0*</number>*

*</property>*

*<property* name="elideMode"*>*

*<enum>*Qt::ElideNone*</enum>*

*</property>*

*<widget* class="QWidget" name="tab"*>*

*<attribute* name="title"*>*

*<string>*Керування книгами*</string>*

*</attribute>*

*<widget* class="QTableWidget" name="tableWidget"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*850*</width>*

*<height>*400*</height>*

*</rect>*

*</property>*

*<property* name="maximumSize"*>*

*<size>*

*<width>*850*</width>*

*<height>*400*</height>*

*</size>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="nameLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*20*</x>*

*<y>*480*</y>*

*<width>*141*</width>*

*<height>*24*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="yearLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*410*</x>*

*<y>*480*</y>*

*<width>*80*</width>*

*<height>*24*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="countLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*560*</x>*

*<y>*480*</y>*

*<width>*71*</width>*

*<height>*24*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="authorLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*200*</x>*

*<y>*480*</y>*

*<width>*151*</width>*

*<height>*24*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="udkLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*710*</x>*

*<y>*480*</y>*

*<width>*111*</width>*

*<height>*24*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="addBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*240*</x>*

*<y>*530*</y>*

*<width>*141*</width>*

*<height>*31*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Додати*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="delBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*450*</x>*

*<y>*530*</y>*

*<width>*121*</width>*

*<height>*31*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Видалити*</string>*

*</property>*

*</widget>*

*<widget* class="QWidget" name=""*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*10*</x>*

*<y>*440*</y>*

*<width>*831*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<layout* class="QHBoxLayout" name="horizontalLayout"*>*

*<item>*

*<widget* class="QLabel" name="label"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Назва*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_2"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Прізвище на ініціали автора*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_4"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Рік видання*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_5"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Кількість екземплярів*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="label\_3"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Номер УДК*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*</layout>*

*</widget>*

*</widget>*

*<widget* class="QWidget" name="tab\_2"*>*

*<attribute* name="title"*>*

*<string>*Дії над книгами*</string>*

*</attribute>*

*<widget* class="QLabel" name="label\_6"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*380*</x>*

*<y>*70*</y>*

*<width>*91*</width>*

*<height>*20*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Номер УДК*</string>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="udkNumLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*320*</x>*

*<y>*100*</y>*

*<width>*191*</width>*

*<height>*31*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="allLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*460*</x>*

*<y>*130*</y>*

*<width>*51*</width>*

*<height>*21*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignLeading|Qt::AlignLeft|Qt::AlignVCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="getBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*240*</x>*

*<y>*160*</y>*

*<width>*151*</width>*

*<height>*51*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Отримати книгу*</string>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="setBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*440*</x>*

*<y>*160*</y>*

*<width>*161*</width>*

*<height>*51*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Повернути книгу*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_7"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*330*</x>*

*<y>*250*</y>*

*<width>*191*</width>*

*<height>*31*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*18*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Пошук книги*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_8"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*380*</x>*

*<y>*300*</y>*

*<width>*91*</width>*

*<height>*20*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*10*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Номер УДК*</string>*

*</property>*

*</widget>*

*<widget* class="QLineEdit" name="udkSearchLE"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*320*</x>*

*<y>*330*</y>*

*<width>*191*</width>*

*<height>*31*</height>*

*</rect>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QPushButton" name="searchBtn"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*340*</x>*

*<y>*380*</y>*

*<width>*151*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string>*Знайти*</string>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="nameLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*50*</x>*

*<y>*510*</y>*

*<width>*191*</width>*

*<height>*20*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="authorLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*260*</x>*

*<y>*510*</y>*

*<width>*141*</width>*

*<height>*21*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="yearLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*460*</x>*

*<y>*510*</y>*

*<width>*101*</width>*

*<height>*20*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="numberLb"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*630*</x>*

*<y>*510*</y>*

*<width>*141*</width>*

*<height>*16*</height>*

*</rect>*

*</property>*

*<property* name="text"*>*

*<string/>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_9"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*320*</x>*

*<y>*130*</y>*

*<width>*141*</width>*

*<height>*20*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*9*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Всього у наявності:*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*<widget* class="QLabel" name="label\_10"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*200*</x>*

*<y>*10*</y>*

*<width>*451*</width>*

*<height>*41*</height>*

*</rect>*

*</property>*

*<property* name="font"*>*

*<font>*

*<pointsize>*18*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Отримання та повернення книг*</string>*

*</property>*

*</widget>*

*<widget* class="QWidget" name=""*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*59*</x>*

*<y>*460*</y>*

*<width>*731*</width>*

*<height>*26*</height>*

*</rect>*

*</property>*

*<layout* class="QHBoxLayout" name="horizontalLayout\_2"*>*

*<item>*

*<widget* class="QLabel" name="nameLb\_2"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Назва*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="nameLb\_3"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Автор*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="nameLb\_4"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*Рік видання*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*<item>*

*<widget* class="QLabel" name="nameLb\_5"*>*

*<property* name="font"*>*

*<font>*

*<pointsize>*12*</pointsize>*

*</font>*

*</property>*

*<property* name="text"*>*

*<string>*В наявності*</string>*

*</property>*

*<property* name="alignment"*>*

*<set>*Qt::AlignCenter*</set>*

*</property>*

*</widget>*

*</item>*

*</layout>*

*</widget>*

*</widget>*

*</widget>*

*</widget>*

*<widget* class="QMenuBar" name="menubar"*>*

*<property* name="geometry"*>*

*<rect>*

*<x>*0*</x>*

*<y>*0*</y>*

*<width>*850*</width>*

*<height>*25*</height>*

*</rect>*

*</property>*

*</widget>*

*<widget* class="QStatusBar" name="statusbar"*/>*

*</widget>*

*<resources/>*

*<connections/>*

*</ui>*

Основне вікно:

library.h:

#ifndef LIBRARY\_H

#define LIBRARY\_H

#include <QMainWindow>

#include <QList>

#include <QTableWidget>

#include <QMessageBox>

#include <QValidator>

#include "book.h"

QT\_BEGIN\_NAMESPACE

*namespace* **Ui** { *class* **Library**; }

QT\_END\_NAMESPACE

*const* int columnCount = 5;

*class* **Library** : *public* QMainWindow

{

Q\_OBJECT

*private*:

QList<Book> books;

*public*:

**Library**(QWidget \*parent = *nullptr*);

~***Library***();

void **TableStyle**();

bool **isIndentical**();

Book **CreateElement**();

void **UpdateTable**();

int **NumberOfBooks**();

void **ClearLE**();

bool **isEmptyLE**();

*private* slots:

void **on\_addBtn\_clicked**();

void **on\_delBtn\_clicked**();

void **on\_getBtn\_clicked**();

void **on\_searchBtn\_clicked**();

void **on\_setBtn\_clicked**();

*private*:

Ui::Library \*ui;

};

#endif *//* *LIBRARY\_H*

library.cpp:

#include "library.h"

#include "ui\_library.h"

#include "book.h"

Library::**Library**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(*new* Ui::Library)

{

ui->setupUi(*this*);

TableStyle();

ui->yearLE->setValidator(*new* QIntValidator(*this*));

ui->countLE->setValidator(*new* QIntValidator(*this*));

ui->udkLE->setValidator(*new* QIntValidator(*this*));

ui->udkNumLE->setValidator(*new* QIntValidator(*this*));

ui->udkSearchLE->setValidator(*new* QIntValidator(*this*));

ui->allLb->setText(QString::number(NumberOfBooks()));

}

Library::~***Library***()

{

*delete* ui;

}

void Library::**TableStyle**()

{

QString headerName[columnCount] = {"Назва", "Прізвище та ініціали автора", "Рік видачі", "Кількість екземплярів", "Номер УДК"};

int headerItemWidth[5] = {175, 220, 105, 170, 125};

ui->tableWidget->setEditTriggers(0);

ui->tableWidget->setColumnCount(columnCount);

*for*(int i = 0; i < columnCount; i++)

{

ui->tableWidget->setHorizontalHeaderItem(i, *new* QTableWidgetItem*(headerName[i])*);

ui->tableWidget->horizontalHeader()->resizeSection(i, headerItemWidth[i]);

}

}

void Library::**UpdateTable**()

{

ui->tableWidget->setRowCount(books.length());

*for*(int i = 0; i < books.length(); i++)

*for*(int j = 0; j < columnCount; j++)

{

Book book = books.at(i);

ui->tableWidget->setItem(i, j,

*new* QTableWidgetItem*(book.GetData().at(j))*);

}

ui->allLb->setText(QString::number(NumberOfBooks()));

}

void Library::**ClearLE**()

{

ui->udkLE->setText("");

ui->authorLE->setText("");

ui->nameLE->setText("");

ui->yearLE->setText("");

ui->countLE->setText("");

}

bool Library::**isEmptyLE**()

{

*if*(ui->udkLE->text().isEmpty()

|| ui->authorLE->text().isEmpty()

|| ui->nameLE->text().isEmpty()

|| ui->yearLE->text().isEmpty()

|| ui->countLE->text().isEmpty())

*return* *true*;

*return* *false*;;

}

bool Library::**isIndentical**()

{

*for*(int i = 0; i < books.length(); i++)

*if*(ui->tableWidget->item(i, 4)->text().toInt() == ui->udkLE->text().toInt())

*return* *true*;

*return* *false*;

}

Book Library::**CreateElement**()

{

Book book;

book.Initialization(

ui->udkLE->text().toInt(),

ui->authorLE->text(),

ui->nameLE->text(),

ui->yearLE->text().toInt(),

ui->countLE->text().toInt());

*return* book;

}

void Library::**on\_addBtn\_clicked**()

{

*if*(!isEmptyLE())

{

*if*(!isIndentical())

{

books.append(CreateElement());

ClearLE();

UpdateTable();

}

*else*

QMessageBox::information(*this*, "Інформація", "Запис з даним номером УДК вже існує");

}

*else*

QMessageBox::information(*this*, "Підказка", "Заповніть всі поля вводу");

}

void Library::**on\_delBtn\_clicked**()

{

*if*(!isEmptyLE())

{

*for*(int i = 0; i < books.length(); i++)

*if*(ui->tableWidget->item(i, 4)->text().toInt() == ui->udkLE->text().toInt())

{

books.removeAt(i);

ClearLE();

UpdateTable();

}

}

*else*

QMessageBox::information(*this*, "Підказка", "Заповніть всі поля вводу");

}

int Library::**NumberOfBooks**()

{

int numberOfBooks = 0;

*for*(Book book : books)

numberOfBooks += book.GetNumber();

*return* numberOfBooks;

}

void Library::**on\_getBtn\_clicked**()

{

*if*(ui->udkNumLE->text().isEmpty())

QMessageBox::information(*this*, "Інформація", "Введіть номер УДК");

*else*

{

bool identUDK = *false*;

bool availability = *false*;

*for*(Book &book : books)

{

*if*(book.GetUdk() == ui->udkNumLE->text().toInt())

{

identUDK = *true*;

*if*(book.GetNumber() > 0)

{

book.GetBook();

availability = *true*;

}

}

}

*if*(identUDK)

{

*if*(availability)

{

QMessageBox::information(*this*, "Інформація", "Книга успішно отримана");

ui->udkNumLE->setText("");

}

*else*

QMessageBox::information(*this*, "Інформація", "Всі екземпляри книги на руках");

}

*else*

QMessageBox::information(*this*, "Інформація", "Даної книги немає у бібліотеці");

ui->allLb->setText(QString::number(NumberOfBooks()));

}

}

void Library::**on\_searchBtn\_clicked**()

{

ui->nameLb->setText("");

ui->authorLb->setText("");

ui->yearLb->setText("");

ui->numberLb->setText("");

bool availability = *false*;

*if*(ui->udkSearchLE->text().isEmpty())

QMessageBox::information(*this*, "Інформація", "Введіть номер УДК");

*else*

{

*for*(Book book : books)

{

*if*(book.GetUdk() == ui->udkSearchLE->text().toInt())

{

ui->nameLb->setText(book.GetName());

ui->authorLb->setText(book.GetAuthor());

ui->yearLb->setText(QString::number(book.GetYear()));

ui->numberLb->setText((QString::number(book.GetNumber())));

availability = *true*;

}

}

*if*(!availability)

QMessageBox::information(*this*, "Інформація", "Даної книги немає у бібліотеці");

*else*

QMessageBox::information(*this*, "Інформація", "Книгу знайдено");

}

}

void Library::**on\_setBtn\_clicked**()

{

*if*(ui->udkNumLE->text().isEmpty())

QMessageBox::information(*this*, "Інформація", "Введіть номер УДК");

*else*

{

bool identUDK = *false*;

bool availability = *false*;

int index = 0;

*for*(Book &book : books)

{

*if*(book.GetUdk() == ui->udkNumLE->text().toInt())

{

identUDK = *true*;

*if*(book.GetNumber() < ui->tableWidget->item(index, 3)->text().toInt())

{

book.SetBook();

availability = *true*;

index ++;

}

}

}

*if*(identUDK)

{

*if*(availability)

{

QMessageBox::information(*this*, "Інформація", "Книга успішно повернена");

ui->udkNumLE->setText("");

}

*else*

QMessageBox::information(*this*, "Інформація", "Всі екземпляри книги знаходяться у бібліотеці");

}

*else*

QMessageBox::information(*this*, "Інформація", "Даної книги немає у бібліотеці");

ui->allLb->setText(QString::number(NumberOfBooks()));

}

}

Клас книги:

book.h

#ifndef BOOK\_H

#define BOOK\_H

#include <QString>

#include <QList>

*class* **Book**

{

*private*:

int udkNumber;

QString author;

QString name;

int year;

int number;

*public*:

**Book**();

~**Book**();

void **Initialization**(int udkNumber, QString author, QString name, int year, int number);

int **GetUdk**();

int **GetNumber**();

int **GetYear**();

QString **GetName**();

QString **GetAuthor**();

void **GetBook**();

void **SetBook**();

QList<QString> **GetData**();

};

#endif *//* *BOOK\_H*

book.cpp

#include "book.h"

Book::**Book**()

{

udkNumber = 0;

author = "";

name = "";

year = 0;

number = 0;

}

Book::~**Book**(){}

void Book::**Initialization**(int udkNumber, QString author, QString name, int year, int number)

{

*this*->udkNumber = udkNumber;

*this*->author = author;

*this*->name = name;

*this*->year = year;

*this*->number = number;

}

int Book::**GetUdk**() { *return* udkNumber; };

int Book::**GetNumber**() { *return* number; };

int Book::**GetYear**() { *return* year; };

QString Book::**GetName**() { *return* name; };

QString Book::**GetAuthor**() { *return* author; };

QList<QString> Book::**GetData**()

{

QList<QString> data;

data.append(name);

data.append(author);

data.append(QString::number(year));

data.append(QString::number(number));

data.append(QString::number(udkNumber));

*return* data;

}

void Book::**GetBook**()

{

number -=1;

}

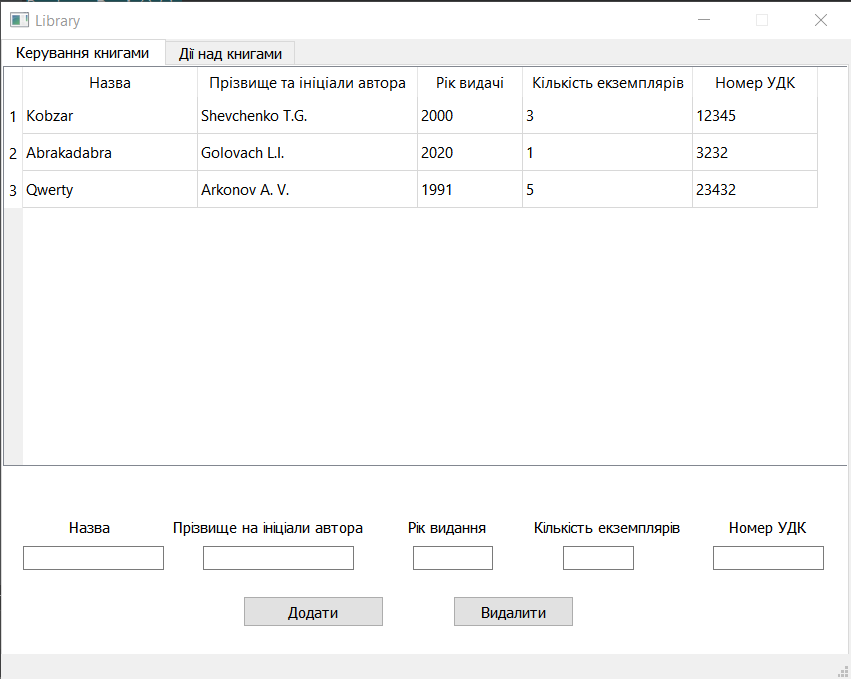
void Book::**SetBook**()

{

number += 1;

}

**Приклад роботи програми 5**:

  
Рисунок 3 – Приклад додавання запису

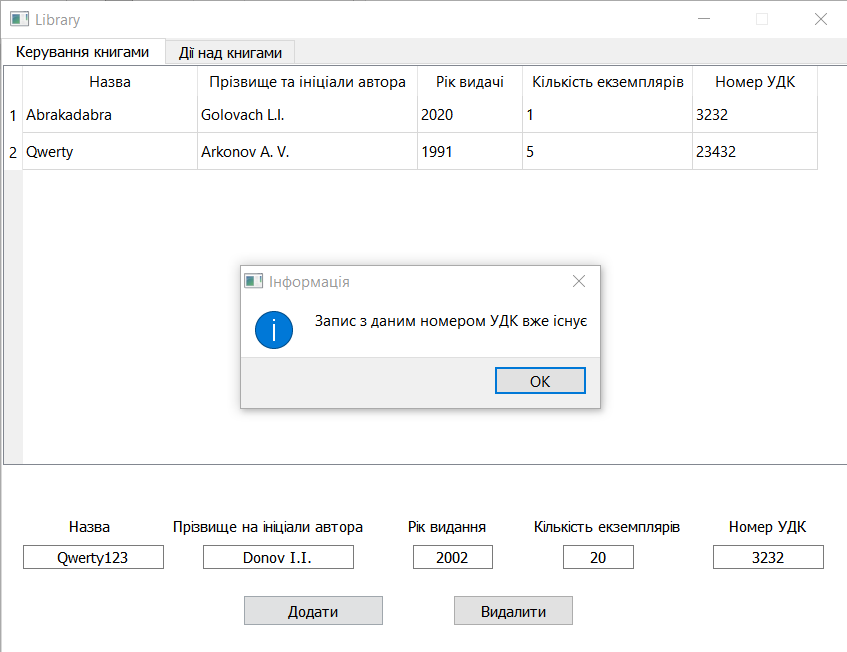


Рисунок 4 – Якщо запис з певним номером УДК був доданий раніше – виводиться повідомлення про неможливість додавання запису

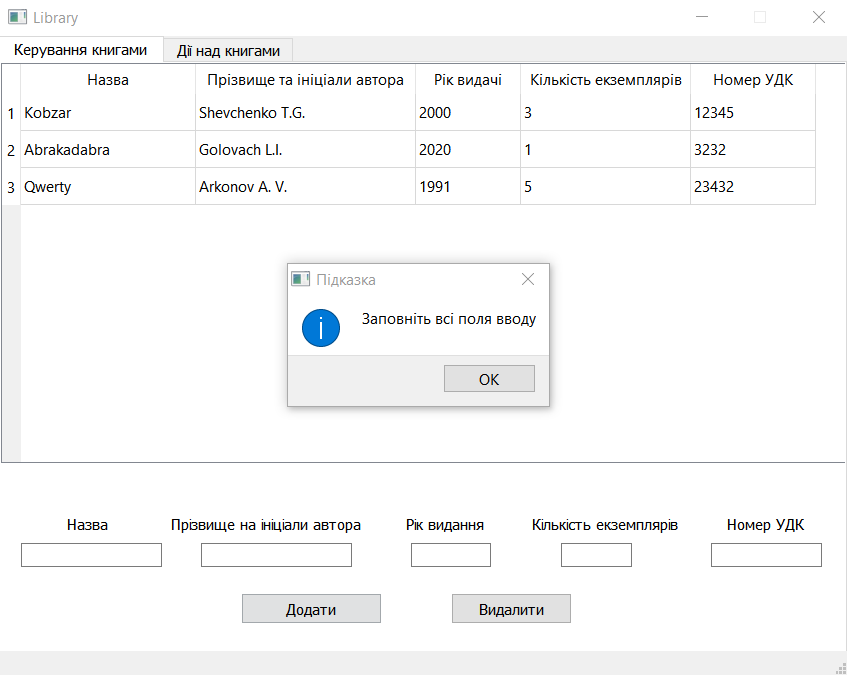
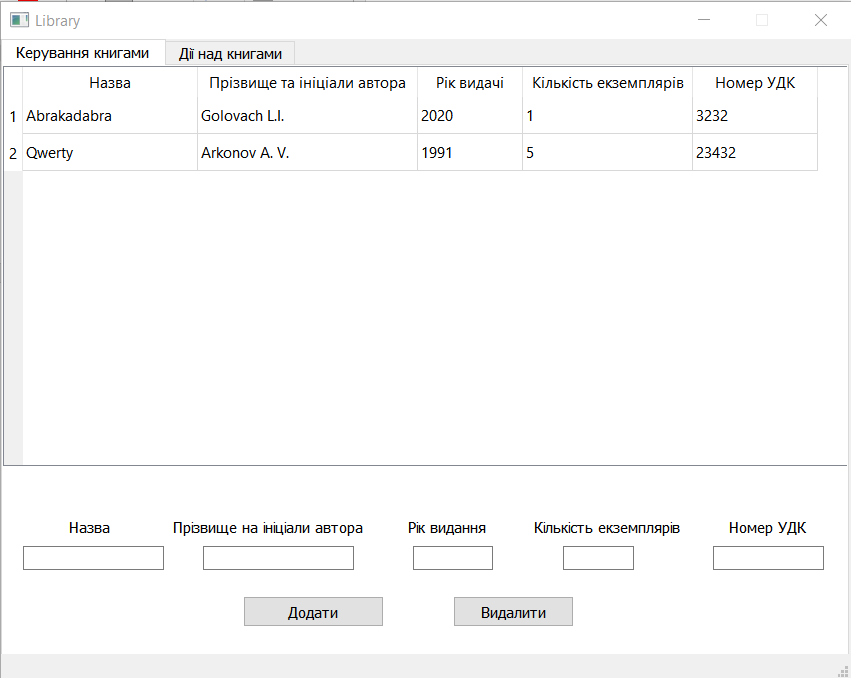
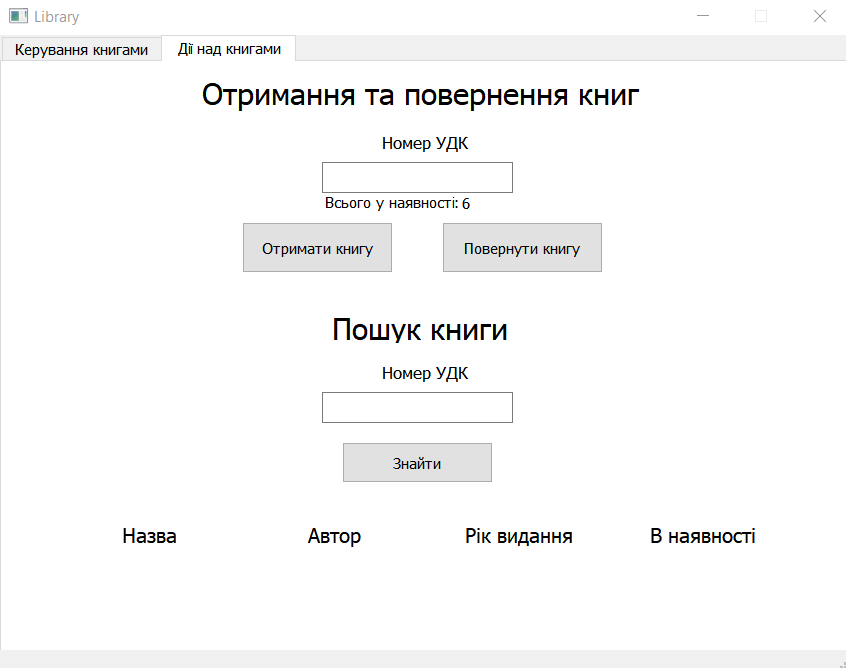
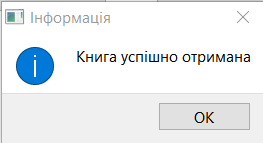


Рисунок 5 – Якщо хоч одне поле не заповнене – вивід повідомлення

  
Рисунок 6 – Приклад видалення запису

  
Рисунок 7 – Відображення кількості наявних книг

  
Рисунок 8 – Вивід повідомлення про успішну видачу книги

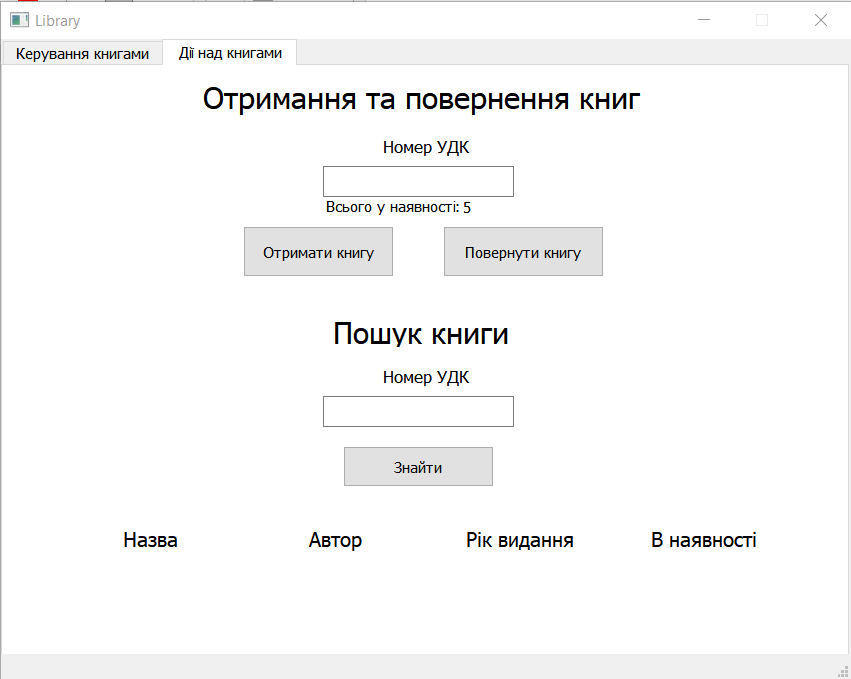


Рисунок 9 – Зменшення кількості наявних книг після видачі

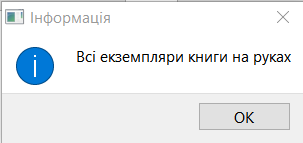


Рисунок 10 – Повідомлення при спробі отримати екземпляр книги якщо всі екземпляри знаходяться на руках

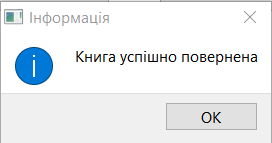


Рисунок 11 – Вивід повідомлення про успішне повернення книги

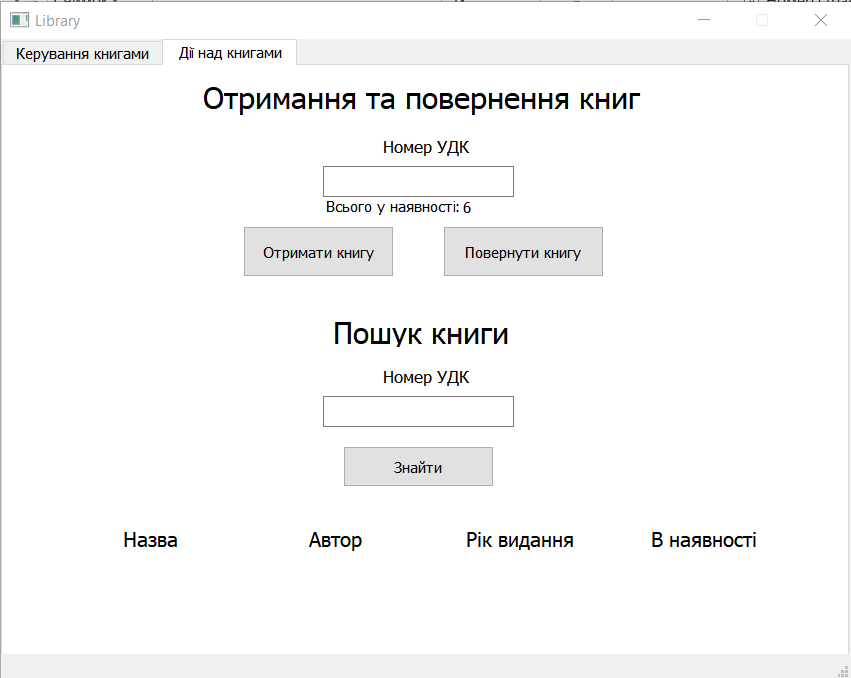


Рисунок 12 – Збільшення кількості наявних книг після повернення

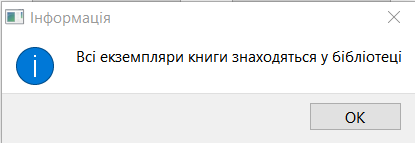


Рисунок 13 – Вивід повідомлення при спробі повернути екземпляр книги який незареєстрований у бібліотеці

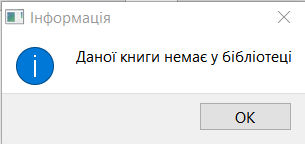


Рисунок 14 – Вивід повідомлення при спробі знайти, отримати або повернути книгу якщо вона незареєстрована в бібліотеці

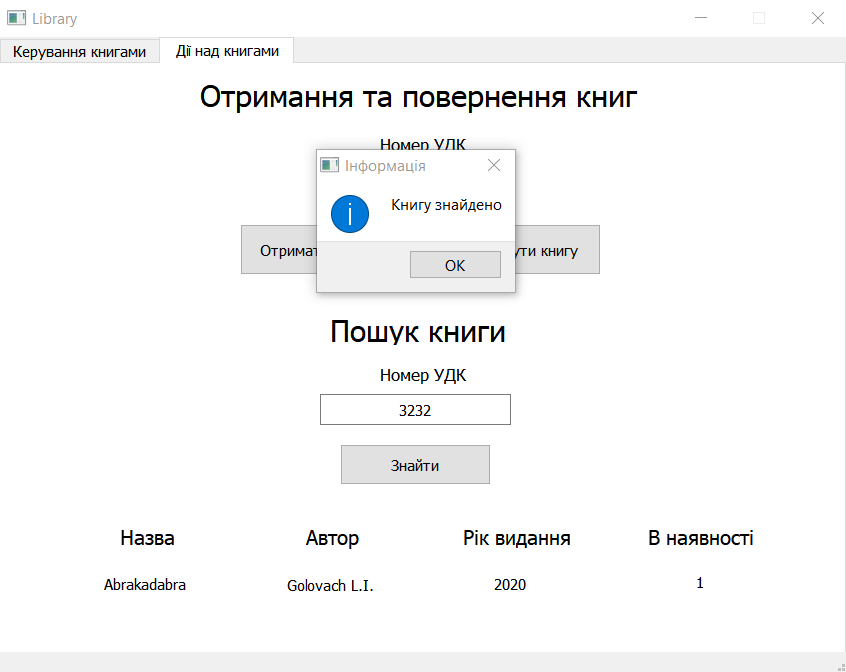


Рисунок 15 – Вивід повідомлення при успішному знахожденні книги та вивід інформації про цю книгу та її наявність

Висновок: за період проведення практичних занять я здобув навички розробки об'єктно-орієнтованих програм використовуючи QT Creator, а також покращив своє розуміння концепції об'єктно-орієнтованого програмування.