Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный** **исследовательский политехнический университет»**

Факультет: Прикладной математики и механики

Кафедра: Вычислительной математики, механики и биомеханики

Направление: 09.03.02 «Информационные системы и технологии»

Профиль бакалавриата: «Информационные системы и технологии»

**ПРАКТИЧЕСКАЯ РАБОТА**

по дисциплине

**«КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ»**

Тема: **«Программная реализация ИС»**

Выполнил:

студент гр.\_\_\_\_ИСТ-19-1б \_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_Сученинова Е. В.\_\_\_\_\_\_\_\_\_

(Ф.И.О.)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(подпись)*

Принял:

ст. преподаватель, Банников Р.Ю.

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(должность, ФИО руководителя)*

*\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(оценка) (подпись)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

**Пермь 2023**

Содержание

[Постановка задачи 3](#_Toc129549180)

[Реализация 4](#_Toc129549181)

[Средства реализация программного кода 4](#_Toc129549182)

[Работа основного функционала 4](#_Toc129549183)

[Вывод 10](#_Toc129549184)

[Приложение 11](#_Toc129549185)

# Постановка задачи

Система предназначена для обеспечения работы непрерывного общения между посетителями и библиотекой, что даст пользователям ряд возможностей:

* для определения наличия или отсутствия данной книги в библиотеке;
* для оформления заказа книги для получения в библиотеке;
* для ознакомления с книгами, находящимися в фонде библиотеки.

В рамках практического задания нужно запрограммировать обозначенный в ТЗ функционал с учетом созданной БД, спроектированного интерфейса и разработанной структуры классов.

# Реализация

## Средства реализация программного кода

Программный продукт реализован под средством языкового программирования С# в среде Visual Studio. Для работы с БД была подключена библиотека Npgsql.dll и для работы с документами формата doc – Microsoft.Office.Interop.Word.dll.

## Работа основного функционала

1. Авторизация

Функционал вкладки «Авторизация» отображен ниже (см. Рисунок 1).

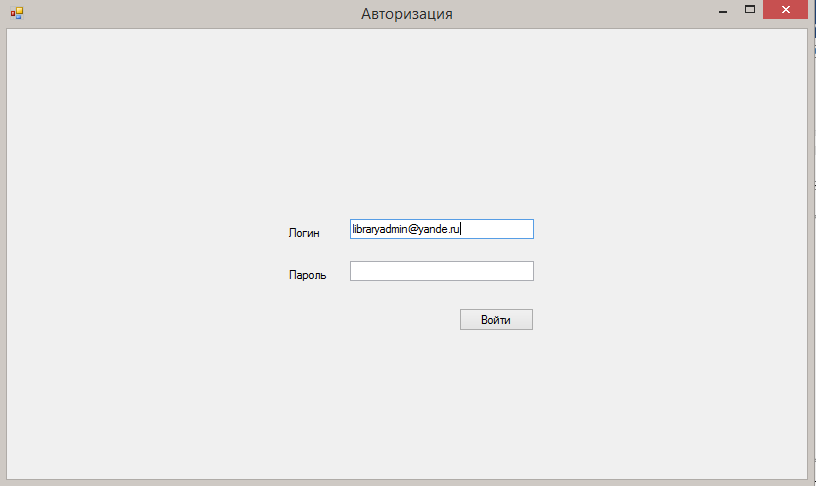


Рисунок 1 – Функционал вкладки «Авторизация».

При нажатие кнопки «Войти», идет обработка текста введенного в поля «Логин» и «Пароль». Обработка подразумевает, поиск введенного логина и проверку на корректный ввод пароля.

1. Функционал ИС под разными пользователями.

На рисунках 2 и 3 отображены функционал вкладок доступные для пользователя с ролью читатель. Читателю доступны 3 вкладки: авторизация, просмотр изданий и читательский билет.

На вкладке «Просмотр изданий» (рис. 2) пользователю доступна возможность выбора произведения для заказа и просмотр информации о нем, для этого необходимо кликнуть по необходимому произведению в таблице.

В выпадающем поле пользователь может выбрать конкретное издание для добавления заказа в корзину.

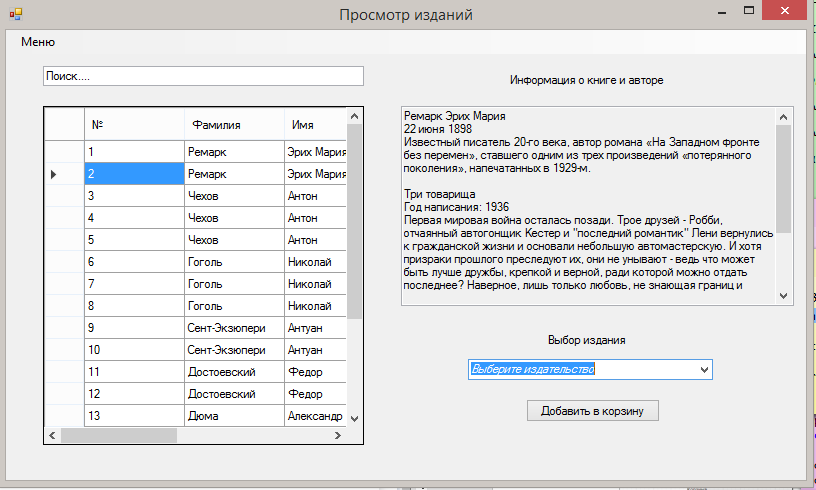


Рисунок 2 – Функционал вкладки «Выбор произведения».

На вкладке «Читательский билет» (рис. 3) реализованы функции: просмотр существующих и предыдущих заказов, просмотр книг, добавленных в корзину. Также возможно удаление выбранной книги из корзины.

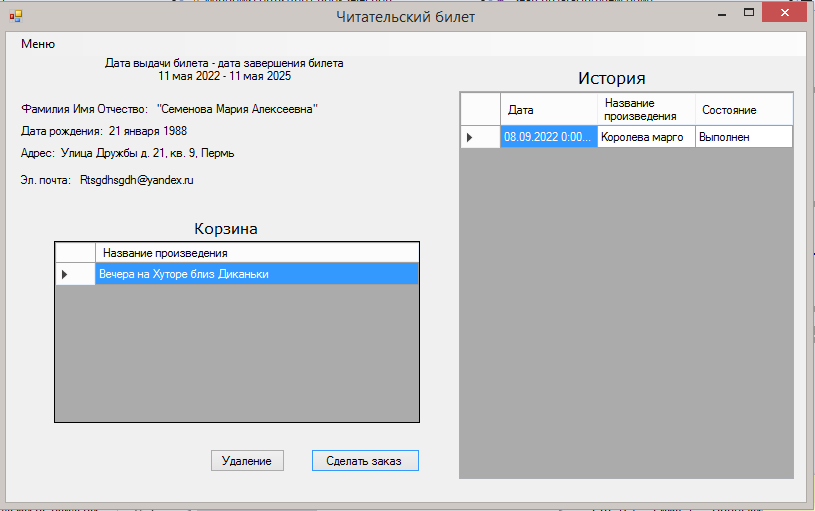


Рисунок 3 – Функционал вкладки «Читательский билет».

На рисунках 4, 5, 6 и 7 отображены функционал вкладок доступные для пользователя с ролью администратор. Администратору доступны 4 вкладки: авторизация, пользователи, управление заказами книг, выбор изданий.

Вкладка «Пользователи» включает в себя таблицу с полями: ФИО, адрес, электронная почта, телефон, дата выдачи билета, дата завершения действия билета, состояние билета на текущий момент (действует/окончен). Внизу расположен блок «режим редактирования», где можно

* вносить изменения в данные существующих пользователей
* добавлять новых пользователей
* удаления данных пользователей из системы.

Также в данной вкладке находятся кнопки по формированию отчетов по действиям пользователей и по кол-ву пользователей, которые реализованы через запрос к БД.

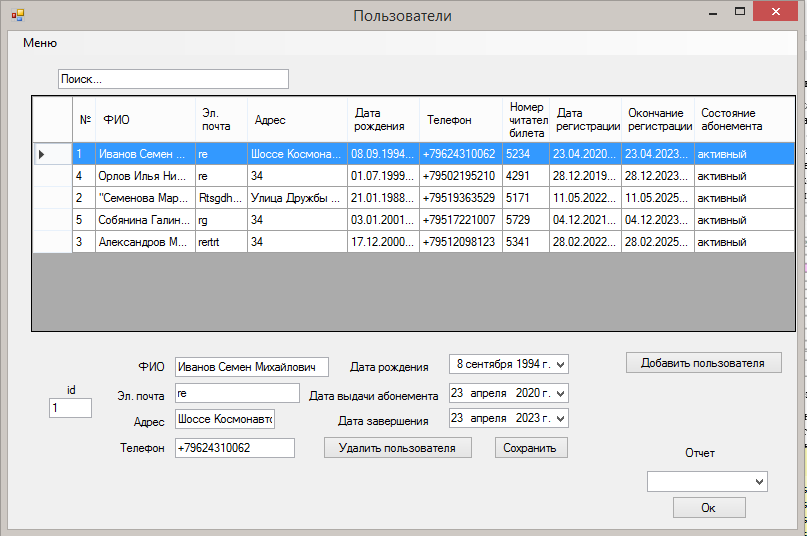


Рисунок 4 – Функционал вкладки «Пользователи».

Вкладка «Управление заказами книг» (рис 5) включает в себя таблицу с полями: ФИО, номер читательского билета, дата выдачи и возврата книги, статус готовности заказа. На данной странице реализованы следующие функции

* Функция поиска по таблице
* Функция изменения статуса заказа и даты возврата

Также в данной вкладке находятся кнопки по формированию отчетов по кол-ву книг у пользователей и о выполненных и отмененных заказах.

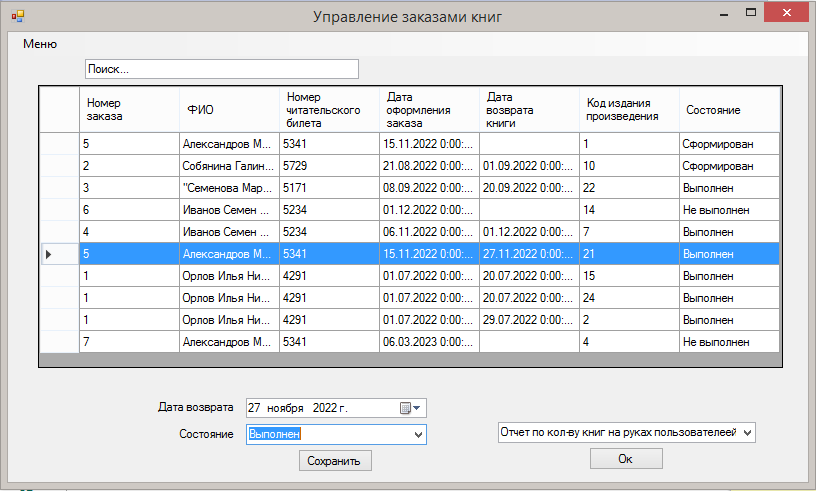


Рисунок 5 – Функционал вкладки «Управление заказами книг».

Функционал вкладки «Выбор изданий» (рис. 6) включает в себя просмотр информации о выбранном произведении и ознакомление с издательствами, доступными в библиотеке. Также просмотр отчета о кол-ве изданий в библиотеке.

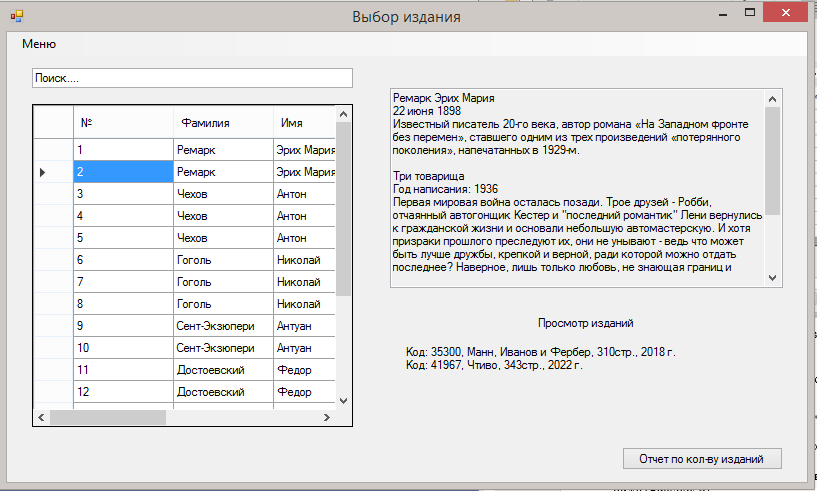


Рисунок 6 – Функционал вкладки «Выбор изданий»

1. Редактирование информации, составляющей БД

Функционал направленный на редактирование, добавление и удаление информации В БД присутствует на каждой вкладке системы. Например, на рисунке 4 реализованы сразу три взаимодействия с БД: добавление нового читателя в систему, редактирование информации о существующем пользователе, удаление пользователя из системы. Все эти функции реализованы через запросы.

1. Отчет

Функционал вкладки «Отчет» отображен ниже (Рисунок 6). На данной вкладки реализован просмотр запроса к БД, с возможностью написать имя, запрашиваемого отчет.

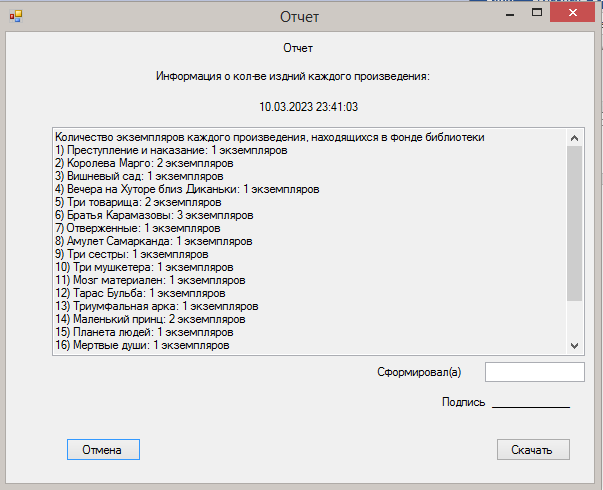


Рисунок 7 – Вкладка «Отчеты»

На рисунке 8 показан вид отчета, выводимого из системы в формат doc.

1. Реакция ИС на ошибочный ввод данных

При ошибочном вводе данных, под полями ввода будет всплывать сообщение. Если ошибка связана с вводом информации в некорректном формате, то текст ошибки: «Некорректный формат введенных данных». Если ошибка связана с несоответствием введенной информации, например, неправильно введен логин или пароль, то текст ошибки: «Неверный логин или пароль» (рис. 9)

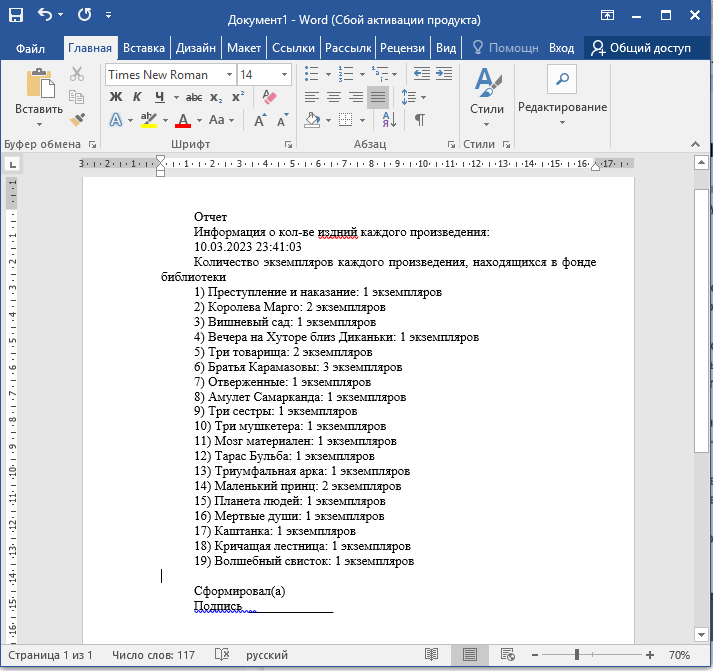


Рисунок 8

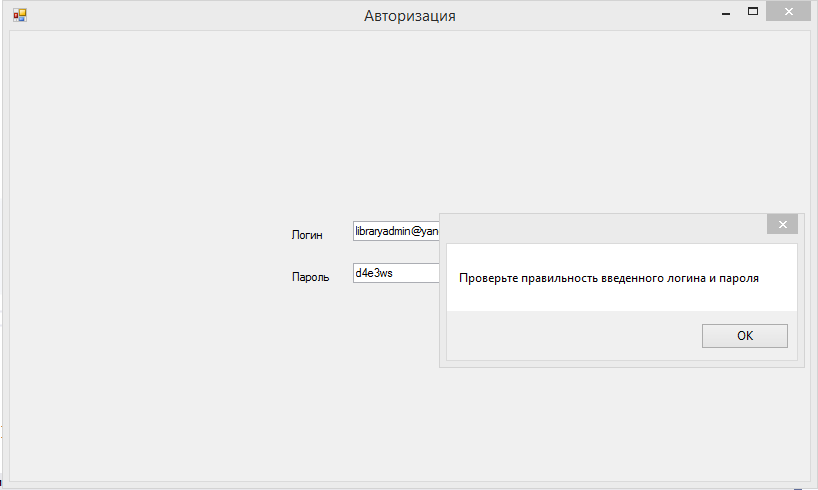


Рисунок 9 – Окно ошибки на введенный неверный пароль

# Вывод

В рамках практического задания была исполнена программная реализация информационной системы, согласно обозначенному в ТЗ функционалу с учетом созданной БД, спроектированного интерфейса и разработанной структуры классов. После программной реализации, идет этап тестирования.

# Приложение

public partial class Authorization : Form

{

DataBase db = new DataBase();

public Authorization()

{

InitializeComponent();}

String role = "";

int Number\_ticket = 0;

String login = "";

String password = "";

private void OpenDBuser()

{

string text\_request\_author = $"select r.number\_ticket, r.emaill, ru.password\_user, ru.role\_user from reader r JOIN role\_user ru ON r.id = ru.id\_user";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

while (read.Read())

{

if (login.Equals(read.GetString(1)))

{

String Pas = "";

if (!read.IsDBNull(2))

Pas = read.GetString(2);

if (password.Equals(Pas))

{

role = read.GetString(3);

Number\_ticket = read.GetInt32(0);

MessageBox.Show("СОВПАЛО");}

else

MessageBox.Show("Проверьте правильность введенного логина и пароля");}}

read.Close();}

private void button1\_Click(object sender, EventArgs e)

{

login=textBox1.Text;

password = textBox2.Text;

OpenDBuser();

if (role == "администратор")

{

OrderManagment om = new OrderManagment();

om.Show();

this.Visible = false; }

if (role == "читатель")

{

BookSelection bs = new BookSelection();

bs.Show();

bs.Transfer\_subscription\_number(Number\_ticket);

this.Visible = false; }}}

public partial class Book\_selection\_admin : Form

{

DataBase db = new DataBase();

public Book\_selection\_admin()

{

InitializeComponent();

Open\_table\_composition(); }

public void Open\_table\_composition()

{

{

string text\_request = $"select ca.id, au.surname, au.name, au.patronymic, co.name, co.year from author au JOIN composition\_author ca ON ca.author\_id = au.id JOIN composition co ON co.id = ca.composition\_id";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

dataGridView1.Columns.Add("id", "№");

dataGridView1.Columns.Add("surname", "Фамилия");

dataGridView1.Columns.Add("name", "Имя");

dataGridView1.Columns.Add("patronymic", "Отчество");

dataGridView1.Columns.Add("name", "Название произведения");

dataGridView1.Columns.Add("year", "Год написания");

while (read.Read())

{

if (!read.IsDBNull(3))

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetString(4), read.GetInt32(5));

else

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), null, read.GetString(4), read.GetInt32(5));}

read.Close(); }

private void Inform()

{

int index = dataGridView1.CurrentCell.RowIndex;

var id\_us = dataGridView1.Rows[index].Cells[0].Value.ToString();

var surname = dataGridView1.Rows[index].Cells[1].Value.ToString();

var name = dataGridView1.Rows[index].Cells[2].Value.ToString();

var patronymic = dataGridView1.Rows[index].Cells[3].Value.ToString();

var name\_composition = dataGridView1.Rows[index].Cells[4].Value.ToString();

var year = dataGridView1.Rows[index].Cells[5].Value.ToString();

InformAuthor(surname, name, patronymic); // Чтение информации о авторе

textBox2.Text += Environment.NewLine;

InformComposition(name\_composition, year); // Чтение информации о произведении

InformPublisher(name\_composition); // Чтение информации о издание }

private void InformAuthor(String surname, String name, String patronymic)

{

string text\_request\_author = $"select au.surname, au.name, au.patronymic, au.birthdate, au.biography from author au where au.surname='{surname}' and au.name='{name}' and au.patronymic='{patronymic}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

if (!read.IsDBNull(2))

textBox2.Text = read.GetString(0) + " " + read.GetString(1) + " " + read.GetString(2) + Environment.NewLine;

else

textBox2.Text = read.GetString(0) + " " + read.GetString(1) + " " + Environment.NewLine;

textBox2.Text += read.GetDateTime(3).ToString("dd MMMM yyyy") + Environment.NewLine;

textBox2.Text += read.GetString(4) + Environment.NewLine;

read.Close(); }

private void InformComposition(String nameComposition, String year)

{

string text\_request\_author = $"select DISTINCT pu.property from publish pu Where pu.name='{nameComposition}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

textBox2.Text += nameComposition + Environment.NewLine;

textBox2.Text += "Год написания: " + year + Environment.NewLine;

if (!read.IsDBNull(0))

textBox2.Text += read.GetString(0) + Environment.NewLine;

else

textBox2.Text += null;

read.Close(); }

int id\_publishCurrent = 0;

private void InformPublisher(String nameComposition)

{

string text\_request\_author = $"select pub.name, pu.size, pu.year, pub.adress, pub.mail, pub.city\_name, pu.id, re.date\_realease, re.date\_return, pu.code from publish pu LEFT JOIN release re ON pu.id = re.publish\_code JOIN publisher pub ON pu.publisher\_name = pub.id where pu.name = '{nameComposition}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

{

if ((read.IsDBNull(7) && read.IsDBNull(8)) || (!read.IsDBNull(7) && !read.IsDBNull(8)))

{

id\_publishCurrent = read.GetInt32(6);

label1.Text+="Код: " + read.GetInt32(9) + ", " + read.GetString(0) + ", " + read.GetInt32(1) + "стр., " + read.GetInt32(2) + " г." + Environment.NewLine; } }

read.Close(); }

private void Search(DataGridView dgw)

{

dgw.Rows.Clear();

String queryString = $"select ca.id, au.surname, au.name, au.patronymic, co.name, co.year from author au JOIN composition\_author ca ON ca.author\_id = au.id JOIN composition co ON co.id = ca.composition\_id where concat(ca.id, au.surname, au.name, au.patronymic, co.name, co.year) like '%" + textBox1.Text + "%'";

NpgsqlCommand com = new NpgsqlCommand(queryString, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

dgw.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetDateTime(4), read.GetString(5), read.GetInt32(6), read.GetDateTime(7), read.GetDateTime(8), read.GetString(9));

read.Close(); }

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

label1.Text="";

Inform(); }

private void button1\_Click(object sender, EventArgs e)

{

Report bs = new Report(1);

bs.Show();

bs.Number\_report(1);

this.Visible = false; }

private void выборПроизведенияToolStripMenuItem\_Click(object sender, EventArgs e)

{

User bs = new User();

bs.Show();

this.Visible = false; }

private void читательскийБилетToolStripMenuItem\_Click(object sender, EventArgs e)

{

OrderManagment bs = new OrderManagment();

bs.Show();

this.Visible = false; }

private void выходToolStripMenuItem\_Click(object sender, EventArgs e)

{

Authorization bs = new Authorization();

bs.Show();

this.Visible = false; }

private void textBox1\_TextChanged(object sender, EventArgs e)

{

Search(dataGridView1); }}

public partial class BookSelection : Form

{

DataBase db = new DataBase();

public BookSelection()

{

InitializeComponent();

Open\_table\_composition(); }

public void Open\_table\_composition()

{

string text\_request = $"select ca.id, au.surname, au.name, au.patronymic, co.name, co.year from author au JOIN composition\_author ca ON ca.author\_id = au.id JOIN composition co ON co.id = ca.composition\_id";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

dataGridView1.Columns.Add("id", "№");

dataGridView1.Columns.Add("surname", "Фамилия");

dataGridView1.Columns.Add("name", "Имя");

dataGridView1.Columns.Add("patronymic", "Отчество");

dataGridView1.Columns.Add("name", "Название произведения");

dataGridView1.Columns.Add("year", "Год написания");

while (read.Read())

{

if (!read.IsDBNull(3))

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetString(4), read.GetInt32(5));

else

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), null, read.GetString(4), read.GetInt32(5)); }

read.Close(); }

private void Inform()

{

int index = dataGridView1.CurrentCell.RowIndex;

var id\_us = dataGridView1.Rows[index].Cells[0].Value.ToString();

var surname = dataGridView1.Rows[index].Cells[1].Value.ToString();

var name = dataGridView1.Rows[index].Cells[2].Value.ToString();

var patronymic = dataGridView1.Rows[index].Cells[3].Value.ToString();

var name\_composition = dataGridView1.Rows[index].Cells[4].Value.ToString();

var year = dataGridView1.Rows[index].Cells[5].Value.ToString();

// Чтение информации о авторе

InformAuthor(surname, name, patronymic);

textBox2.Text += Environment.NewLine;

// Чтение информации о произведении

InformComposition(name\_composition, year);

// Чтение информации о издание

InformPublisher(name\_composition); }

private void InformAuthor(String surname, String name, String patronymic)

{

string text\_request\_author = $"select au.surname, au.name, au.patronymic, au.birthdate, au.biography from author au where au.surname='{surname}' and au.name='{name}' and au.patronymic='{patronymic}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

if (!read.IsDBNull(2))

textBox2.Text = read.GetString(0) + " " + read.GetString(1) + " " + read.GetString(2) + Environment.NewLine;

else

textBox2.Text = read.GetString(0) + " " + read.GetString(1) + " " + Environment.NewLine;

textBox2.Text += read.GetDateTime(3).ToString("dd MMMM yyyy") + Environment.NewLine;

textBox2.Text += read.GetString(4) + Environment.NewLine;

read.Close();

db.closeConnection(); }

private void InformComposition(String nameComposition, String year)

{

string text\_request\_author = $"select DISTINCT pu.property from publish pu Where pu.name='{nameComposition}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

textBox2.Text += nameComposition + Environment.NewLine;

textBox2.Text += "Год написания: "+year + Environment.NewLine;

if (!read.IsDBNull(0))

textBox2.Text += read.GetString(0) + Environment.NewLine;

else

textBox2.Text += null;

read.Close();

db.closeConnection(); }

int id\_publishCurrent = 0;

int Reader\_ticketds;

private void InformPublisher(String nameComposition)

{

string text\_request\_author = $"select pub.name, pu.size, pu.year, pub.adress, pub.mail, pub.city\_name, pu.id, re.date\_realease, re.date\_return, pu.code from publish pu LEFT JOIN release re ON pu.id = re.publish\_code JOIN publisher pub ON pu.publisher\_name = pub.id where pu.name = '{nameComposition}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

{

if ((read.IsDBNull(7) && read.IsDBNull(8)) || (!read.IsDBNull(7) && !read.IsDBNull(8)))

{

id\_publishCurrent = read.GetInt32(6);

comboBox1.Items.Add(read.GetString(0) + ", " + read.GetInt32(1) + "стр., " + read.GetInt32(2) + " г."); }}

read.Close();

db.closeConnection(); }

private void Add\_book()

{

var addQuery = $"insert into cart (cart\_number\_tickrt, cart\_id\_publish) values ('{Reader\_ticketds}', '{id\_publishCurrent}')";

var command2 = new NpgsqlCommand(addQuery, db.getConnection());

db.openConnection();

command2.ExecuteNonQuery();

MessageBox.Show("Книга добавлена в корзину!"); }

private void Search(DataGridView dgw)

{

dgw.Rows.Clear();

String queryString = $"select ca.id, au.surname, au.name, au.patronymic, co.name, co.year from author au JOIN composition\_author ca ON ca.author\_id = au.id JOIN composition co ON co.id = ca.composition\_id where concat(ca.id, au.surname, au.name, au.patronymic, co.name, co.year) like '%" + textBox1.Text + "%'";

NpgsqlCommand com = new NpgsqlCommand(queryString, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

dgw.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetDateTime(4), read.GetString(5), read.GetInt32(6), read.GetDateTime(7), read.GetDateTime(8), read.GetString(9));

read.Close(); }

public void Transfer\_subscription\_number(int Number)

{

Reader\_ticketds = Number; }

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

comboBox1.Text="Выберите издательство";

comboBox1.Items.Clear();

Inform();

if (comboBox1.Items.Count == 0)

button1.Enabled = false;

else

button1.Enabled = true; }

public void Prov()

{

string text\_request\_author = $"select case when ca.cart\_id\_publish ='{id\_publishCurrent}' then 'да' when ca.cart\_id\_publish <>'{id\_publishCurrent}' then 'нет' end FROM cart ca";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

if (read.GetString(0).Equals("да"))

{

button1.Enabled = false;

MessageBox.Show("Данное издание добавлено добавлено в корзину!"); }

else

{

db.closeConnection();

Add\_book(); }}

private void button1\_Click(object sender, EventArgs e)

{ Prov(); }

private void читательскийБилетToolStripMenuItem\_Click(object sender, EventArgs e)

{

readers\_ticket bs = new readers\_ticket(Reader\_ticketds);

bs.Show();

bs.Transfer\_subscription\_number(Reader\_ticketds);

this.Visible = false; }

private void textBox1\_TextChanged(object sender, EventArgs e)

{

Search(dataGridView1);}}

class DataBase

{

NpgsqlConnection con = new NpgsqlConnection("Server=localhost; Port=5432; Database=Book\_catalog; User Id=postgres; Password=hhim973v;");

public void openConnection()

{

if (con.State == System.Data.ConnectionState.Closed)

con.Open();}

public void closeConnection()

{

if (con.State == System.Data.ConnectionState.Open)

con.Close();}

public NpgsqlConnection getConnection()

{

return con;}}

public partial class OrderManagment : Form

{

DataBase db = new DataBase();

String role;

public OrderManagment()

{

InitializeComponent();

Open\_table(); }

public void Open\_table()

{

db.openConnection();

string text\_request = $"Select rel.code\_order, re.name, re.number\_ticket, rel.date\_realease, rel.date\_return, rel.publish\_code, st.state FROM reader re JOIN release rel ON re.id = rel.reader\_number\_ticket JOIN state\_release st ON rel.id = st.id\_release WHERE rel.id = st.id\_release";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

dataGridView1.Columns.Add("code\_order", "Номер заказа");

dataGridView1.Columns.Add("name", "ФИО");

dataGridView1.Columns.Add("number\_ticket", "Номер читательского билета");

dataGridView1.Columns.Add("date\_realease", "Дата оформления заказа");

dataGridView1.Columns.Add("date\_return", "Дата возврата книги");

dataGridView1.Columns.Add("publish\_code", "Код издания произведения");

dataGridView1.Columns.Add("state", "Состояние");

while (read.Read())

{

if(!read.IsDBNull(4))

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetInt32(2), read.GetDateTime(3), read.GetDateTime(4), read.GetInt32(5), read.GetString(6));

else

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetInt32(2), read.GetDateTime(3), null, read.GetInt32(5), read.GetString(6)); }

read.Close(); }

private void Search(DataGridView dgw)

{

dgw.Rows.Clear();

string queryString = $"";

queryString = $"SELECT rel.code\_order, re.name, re.number\_ticket, rel.date\_realease, rel.date\_return, rel.publish\_code, st.state FROM reader re JOIN release rel ON re.id = rel.reader\_number\_ticket JOIN state\_release st ON rel.id = st.id\_release Where concat(rel.code\_order, re.name, re.number\_ticket, rel.date\_realease, rel.date\_return, rel.publish\_code, st.state) like '%" + textBox1.Text + "%'";

NpgsqlCommand com = new NpgsqlCommand(queryString, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

{

if(!read.IsDBNull(4))

dgw.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetInt32(2), read.GetDateTime(3), read.GetDateTime(4), read.GetInt32(5), read.GetString(6));

else

dgw.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetInt32(2), read.GetDateTime(3), null, read.GetInt32(5), read.GetString(6)); }

read.Close();}

public void Change\_state()

{

DataGridViewRow row = dataGridView1.Rows[selectedRow];

String text\_state = row.Cells[6].Value.ToString();

String text\_publish\_code = row.Cells[5].Value.ToString();

int chislo\_publish\_code= Convert.ToInt32(text\_publish\_code);

string text\_request2= $"UPDATE state\_release st SET state = '{comboBox1.Text}' From release rel WHERE st.id\_release = rel.id and rel.publish\_code = '{chislo\_publish\_code}'";

dataGridView1.Rows[selectedRow].Cells[6].Value = comboBox1.Text;

var command = new NpgsqlCommand(text\_request2, db.getConnection());

command.ExecuteNonQuery();

MessageBox.Show("Запись успешно изменена!"); }

int selectedRow;

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

selectedRow = e.RowIndex;

DataGridViewRow row = dataGridView1.Rows[selectedRow];

comboBox1.Text = row.Cells[6].Value.ToString();

if (row.Cells[4].Value != null)

dateTimePicker1.Text = row.Cells[4].Value.ToString();

else

dateTimePicker1.Text = Convert.ToString(DateTime.Today);}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

Search(dataGridView1);}

private void button1\_Click(object sender, EventArgs e)

{

Change\_state(); }

private void выборПроизведенияToolStripMenuItem\_Click(object sender, EventArgs e)

{

User bs = new User();

bs.Show();

this.Visible = false; }

private void выходToolStripMenuItem\_Click(object sender, EventArgs e)

{

Authorization bs = new Authorization();

bs.Show();

this.Visible = false; }

private void просмотрИзданийToolStripMenuItem\_Click(object sender, EventArgs e)

{

Book\_selection\_admin bs = new Book\_selection\_admin();

bs.Show();

this.Visible = false; }

private void button2\_Click(object sender, EventArgs e)

{

Report bs;

if (comboBox2.SelectedIndex==0)

bs = new Report(2);

else

bs = new Report(3);

bs.Show();

this.Visible = false; } }

public partial class readers\_ticket : Form

{

DataBase db = new DataBase();

int Reader\_ticketds;

String role;

public readers\_ticket(int Reader\_ticketds)

{

InitializeComponent();

string text\_request = $"select re.date\_realease, pu.name, sr.state from release re JOIN publish pu ON re.publish\_code = pu.id JOIN reader r ON r.id = re.reader\_number\_ticket JOIN state\_release sr ON re.id = sr.id\_release Where r.number\_ticket = '{Reader\_ticketds}'";

Open\_table(dataGridView1, text\_request);

Inform\_readers(Reader\_ticketds);

text\_request = $"select pu.name, pu.id from publish pu JOIN cart ca ON ca.cart\_id\_publish = pu.id Where ca.cart\_number\_tickrt = '{Reader\_ticketds}'";

Open\_table(dataGridView2, text\_request); }

public void Transfer\_subscription\_number(int Number)

{

Reader\_ticketds = Number; }

private void Open\_table(DataGridView dgw, String text\_request)

{

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

if (dataGridView1 == dgw)

{

dataGridView1.Columns.Add("date\_realease", "Дата");

dataGridView1.Columns.Add("name", "Название произведения");

dataGridView1.Columns.Add("state", "Состояние");

while (read.Read())

dataGridView1.Rows.Add(read.GetDateTime(0), read.GetString(1), read.GetString(2));

read.Close();}

if (dataGridView2 == dgw)

{

dataGridView2.Columns.Add("name", "Название произведения");

while (read.Read())

{

dataGridView2.Rows.Add(read.GetString(0));

publish\_code\_list.Add(read.GetInt32(1));}

read.Close();} }

public void Inform\_readers(int number\_ticket)

{

string text\_request = $"select name, date, adress, telephone, emaill, date\_release, validity\_period from reader Where number\_ticket = '{number\_ticket}'";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

label4.Text += Environment.NewLine + read.GetDateTime(5).ToString("dd MMMM yyyy") + " - " + read.GetDateTime(6).ToString("dd MMMM yyyy");

label1.Text += read.GetString(0);

label2.Text += read.GetString(2);

label3.Text += read.GetString(4);

label6.Text += read.GetDateTime(1).ToString("dd MMMM yyyy");

read.Close();}

List<int> publish\_code\_list = new List<int>();

public void Making\_order()

{

//вычисление основных полей таблиц

// id

int id = 0;

string text\_request = $"SELECT MAX(id) FROM release";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

id = read.GetInt32(0);

read.Close();

db.closeConnection();

//вычисление номера заказа

//code\_order

int code\_oder = 0;

text\_request = $"SELECT MAX(code\_order) FROM release";

NpgsqlCommand com2 = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read2 = com2.ExecuteReader();

read2.Read();

code\_oder = read2.GetInt32(0);

read2.Close();

db.closeConnection();

//вычисление номера пользователя

//id\_user

int id\_user = 0;

text\_request = $"select id from reader Where number\_ticket='{Reader\_ticketds}'";

NpgsqlCommand com3 = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read3 = com3.ExecuteReader();

read3.Read();

id\_user = read3.GetInt32(0);

read3.Close();

db.closeConnection();

//Хз state\_release

//state\_release\_id

int state\_release\_id = 0;

text\_request = $"SELECT MAX(id) FROM state\_release";

NpgsqlCommand com4 = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read4 = com4.ExecuteReader();

read4.Read();

state\_release\_id = read4.GetInt32(0);

read4.Close();

Add\_oder(id, code\_oder, id\_user, state\_release\_id);

MessageBox.Show("Книга заказана"); }

public void Add\_oder(int id\_old, int code\_order, int id\_user, int state\_release\_id)

{

code\_order += 1;

int list\_pos = 0;

foreach (DataGridViewRow row in dataGridView2.Rows)

{

id\_old += 1;

int idd = publish\_code\_list[list\_pos];

var addQuery = $"insert into release (id, date\_realease, date\_return, reader\_number\_ticket, publish\_code, status, code\_order) values ('{id\_old}', '{DateTime.Today}', null, '{id\_user}', '{idd}', '{"не выполнено"}', '{code\_order}')";

var command = new NpgsqlCommand(addQuery, db.getConnection());

command.ExecuteNonQuery();

state\_release\_id += 1;

var addQuery2 = $"insert into state\_release (id, id\_release, state, save) values ('{state\_release\_id}', '{id\_old}', '{"Оформлен"}', '{DateTime.Now.AddDays(7)}')";

var command2 = new NpgsqlCommand(addQuery2, db.getConnection());

command2.ExecuteNonQuery();

list\_pos += 1;} }

int selectedRow;

public void Deleate(String delateQuery)

{

var command = new NpgsqlCommand(delateQuery, db.getConnection());

command.ExecuteNonQuery(); }

private void Update(DataGridView dgw, String text\_request)

{

dgw.Rows.Clear();

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

if (dataGridView1==dgw)

{

while (read.Read())

dataGridView1.Rows.Add(read.GetDateTime(0), read.GetString(1), read.GetString(2));

read.Close(); }

if (dataGridView2 == dgw)

{

while (read.Read())

{

dataGridView2.Rows.Add(read.GetString(0));

publish\_code\_list.Add(read.GetInt32(1)); }

read.Close(); } }

private void button1\_Click(object sender, EventArgs e)

{

Making\_order();

var delateQuery = $"delete from cart where cart\_number\_tickrt='{Reader\_ticketds}'";

Deleate(delateQuery);

string text\_request = $"select pu.name, pu.id from publish pu JOIN cart ca ON ca.cart\_id\_publish = pu.id Where ca.cart\_number\_tickrt = '{Reader\_ticketds}'";

Update(dataGridView2, text\_request);

text\_request = $"select re.date\_realease, pu.name, sr.state from release re JOIN publish pu ON re.publish\_code = pu.id JOIN reader r ON r.id = re.reader\_number\_ticket JOIN state\_release sr ON re.id = sr.id\_release Where r.number\_ticket = '{Reader\_ticketds}'";

Update(dataGridView1, text\_request);}

private void button2\_Click(object sender, EventArgs e)

{

var delateQuery = $"delete from cart where cart\_id\_publish='{publish\_code\_list[selectedRow]}'";

Deleate(delateQuery);

string text\_request = $"select pu.name, pu.id from publish pu JOIN cart ca ON ca.cart\_id\_publish = pu.id Where ca.cart\_number\_tickrt = '{Reader\_ticketds}'";

Update(dataGridView2, text\_request); }

private void dataGridView2\_CellClick(object sender, DataGridViewCellEventArgs e)

{

selectedRow = e.RowIndex; }

private void выборПроизведенияToolStripMenuItem\_Click(object sender, EventArgs e)

{

BookSelection bs = new BookSelection();

bs.Show();

bs.Transfer\_subscription\_number(Reader\_ticketds);

this.Visible = false;} }

public partial class Report : Form

{

int id\_report;

DataBase db = new DataBase();

public Report(int id)

{

InitializeComponent();

id\_report = id;

Text\_report(); }

public void Text\_report()

{

if (id\_report == 1)

Text\_report\_id\_1();

if (id\_report == 2)

Text\_report\_id\_2();

if (id\_report == 3)

Text\_report\_id\_3();

if (id\_report == 4)

Text\_report\_id\_4();

if (id\_report == 5)

Text\_report\_id\_5();}

private void Text\_report\_id\_1()

{

label2.Text = "Информация о кол-ве издний каждого произведения:";

label3.Text = DateTime.Now.ToString();

textBox2.Text = "Количество экземпляров каждого произведения, находящихся в фонде библиотеки"+Environment.NewLine;

string text\_request\_author = $"Select DISTINCT co.name, count(\*) over(partition by co.name) as func\_count FROM composition co JOIN composition\_publish cp ON cp.composition\_id = co.id JOIN publish pu ON pu.id = cp.publish\_id";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

int i = 1;

while (read.Read())

{

textBox2.Text += i + ") " + read.GetString(0)+": "+read.GetInt32(1)+" экземпляров" + Environment.NewLine;

i += 1; }

read.Close();

db.openConnection();}

private void Text\_report\_id\_2()

{

label2.Text = "Количество книг на руках пользователей библиотекой";

label3.Text = DateTime.Now.ToString();

textBox2.Text = "Количество книг на руках пользователей библиотеки: ";

string text\_request\_author = $"SELECT COUNT(\*) FROM release re JOIN state\_release sr ON re.id = sr.id\_release WHERE date\_return IS NULL AND sr.state = 'Выполнен'";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

read.Read();

textBox2.Text += read.GetInt32(0);

read.Close();

db.openConnection(); }

private void Text\_report\_id\_3()

{

label2.Text = "Статистика по статусам заказов пользователей";

label3.Text = DateTime.Now.ToString();

textBox2.Text = "Статистика заказов по статусам: " + Environment.NewLine;

string text\_request\_author = $"Select DISTINCT sr.state, count(\*) over(partition by sr.state) as func\_count FROM state\_release sr";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

{

textBox2.Text += read.GetString(0) + ": " + read.GetInt32(1) + Environment.NewLine;}

read.Close();

db.openConnection();}

private void Text\_report\_id\_4()

{

label2.Text = "Статистика по количеству пользователей ";

label3.Text = DateTime.Now.ToString();

textBox2.Text = "На текущий момент пользователи: " + Environment.NewLine;

string text\_request\_author = $"select re.name, ru.role\_user FROM reader re JOIN role\_user ru ON re.id=ru.id\_user";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

int i = 1;

while (read.Read())

{

textBox2.Text += i + ") " + read.GetString(0) + " (" + read.GetString(1) + ")" + Environment.NewLine;

i += 1;}

textBox2.Text += "Итого пользователей: " + i;

read.Close();

db.openConnection();}

private void Text\_report\_id\_5()

{

label2.Text = "Статистика по действиям пользователей в системе";

label3.Text = DateTime.Now.ToString();

textBox2.Text = "Действия пользователей : " + Environment.NewLine;

string text\_request\_author = $"select r.number\_ticket, r.name, re.date\_realease, pu.name, sr.state from release re JOIN publish pu ON re.publish\_code = pu.id JOIN reader r ON r.id = re.reader\_number\_ticket JOIN state\_release sr ON re.id = sr.id\_release";

NpgsqlCommand com = new NpgsqlCommand(text\_request\_author, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

int i = 1;

while (read.Read())

{

textBox2.Text += i + ") " + read.GetInt32(0)+" - " + read.GetString(1) +" ( "+ read.GetDateTime(2)+ " - "+read.GetString(3)+") "+ read.GetString(4)+ Environment.NewLine;

i += 1;}

read.Close();

db.openConnection();}

private void button2\_Click(object sender, EventArgs e)

{

exportWord.Application wordapp = new exportWord.Application();

wordapp.Visible = true;

exportWord.Document worddoc;

object wordobj = System.Reflection.Missing.Value;

worddoc = wordapp.Documents.Add(ref wordobj);

wordapp.Selection.TypeText(label1.Text+Environment.NewLine);

wordapp.Selection.TypeText(label2.Text + Environment.NewLine);

wordapp.Selection.TypeText(label3.Text + Environment.NewLine);

wordapp.Selection.TypeText(textBox2.Text + Environment.NewLine);

wordapp.Selection.TypeText("" + Environment.NewLine);

wordapp.Selection.TypeText("" + Environment.NewLine);

wordapp.Selection.TypeText(label5.Text+textBox1.Text + Environment.NewLine);

wordapp.Selection.TypeText(label6.Text + Environment.NewLine);

wordapp = null; }

public void Number\_report(int id)

{

id\_report = id; }

private void button1\_Click(object sender, EventArgs e)

{

Book\_selection\_admin bs = new Book\_selection\_admin();

bs.Show();

this.Visible = false;}}

public partial class User : Form

{

DataBase db = new DataBase();

public User()

{

InitializeComponent();

Open\_table();}

public void Open\_table()

{

string text\_request = $"select id, name, emaill, adress, date, telephone, number\_ticket, date\_release, validity\_period, case when validity\_period-date\_release > 0 then 'активный' when validity\_period-date\_release < 0 then 'не активный' end FROM reader Where role='читатель'";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

dataGridView1.Columns.Add("id", "№");

dataGridView1.Columns.Add("name", "ФИО");

dataGridView1.Columns.Add("emaill", "Эл. почта");

dataGridView1.Columns.Add("adress", "Адрес");

dataGridView1.Columns.Add("date", "Дата рождения");

dataGridView1.Columns.Add("telephone", "Телефон");

dataGridView1.Columns.Add("number\_ticket", "Номер читательского билета");

dataGridView1.Columns.Add("date\_release", "Дата регистрации");

dataGridView1.Columns.Add("validity\_period", "Окончание регистрации");

dataGridView1.Columns.Add("case", "Состояние абонемента");

while (read.Read())

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetDateTime(4), read.GetString(5), read.GetInt32(6), read.GetDateTime(7), read.GetDateTime(8), read.GetString(9));

read.Close();}

private void Update\_table(DataGridView dgw)

{

dgw.Rows.Clear();

string text\_request = $"select id, name, emaill, adress, date, telephone, number\_ticket, date\_release, validity\_period, case when validity\_period-date\_release > 0 then 'активный' when validity\_period-date\_release < 0 then 'не активный' end FROM reader Where role='читатель'";

NpgsqlCommand com = new NpgsqlCommand(text\_request, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetDateTime(4), read.GetString(5), read.GetInt32(6), read.GetDateTime(7), read.GetDateTime(8), read.GetString(9));

read.Close(); }

private void Search(DataGridView dgw)

{

dgw.Rows.Clear();

String queryString = $"select id, name, emaill, adress, date, telephone, number\_ticket, date\_release, validity\_period, case when validity\_period-date\_release > 0 then 'активный' when validity\_period-date\_release < 0 then 'не активный' end FROM reader where role = 'читатель' and concat(id, name, emaill, adress, date, telephone, number\_ticket, date\_release, validity\_period, case when validity\_period-date\_release > 0 then 'активный' when validity\_period-date\_release < 0 then 'не активный' end) like '%" + textBox1.Text + "%'";

NpgsqlCommand com = new NpgsqlCommand(queryString, db.getConnection());

db.openConnection();

NpgsqlDataReader read = com.ExecuteReader();

while (read.Read())

dataGridView1.Rows.Add(read.GetInt32(0), read.GetString(1), read.GetString(2), read.GetString(3), read.GetDateTime(4), read.GetString(5), read.GetInt32(6), read.GetDateTime(7), read.GetDateTime(8), read.GetString(9));

read.Close(); }

private void Delate()

{

int index = dataGridView1.CurrentCell.RowIndex;

db.openConnection();

var id = Convert.ToInt32(dataGridView1.Rows[index].Cells[0].Value);

var delateQuery = $"delete from reader where id='{id}'";

var command = new NpgsqlCommand(delateQuery, db.getConnection());

command.ExecuteNonQuery();

MessageBox.Show("Запись успешно удалена!");

dataGridView1.Rows[index].Visible = false; }

private void Update\_user()

{

var selectedRowIndex = dataGridView1.CurrentCell.RowIndex;

var id\_us = textBox6.Text;

var FIO = textBox2.Text;

var emaill = textBox3.Text;

var adress = textBox4.Text;

var phone = textBox5.Text;

var date = dateTimePicker1.Text;

var date\_issue = dateTimePicker2.Text;

var date\_completion = dateTimePicker3.Text;

if (dataGridView1.Rows[selectedRowIndex].Cells[0].Value.ToString() != string.Empty)

{

dataGridView1.Rows[selectedRowIndex].SetValues(id\_us, FIO, emaill, adress, phone, date, date\_issue, date\_completion);

int index = dataGridView1.CurrentCell.RowIndex;

id\_us = dataGridView1.Rows[index].Cells[0].Value.ToString();

FIO = dataGridView1.Rows[index].Cells[1].Value.ToString();

emaill = dataGridView1.Rows[index].Cells[2].Value.ToString();

adress = dataGridView1.Rows[index].Cells[3].Value.ToString();

phone = dataGridView1.Rows[index].Cells[4].Value.ToString();

date = dataGridView1.Rows[index].Cells[5].Value.ToString();

date\_issue = dataGridView1.Rows[index].Cells[6].Value.ToString();

date\_completion= dataGridView1.Rows[index].Cells[7].Value.ToString();

String changeQuery = $"update reader set name='{FIO}', emaill='{emaill}', date='{Convert.ToDateTime(date)}', adress='{adress}', telephone='{phone}', date\_release='{Convert.ToDateTime(date\_issue)}', validity\_period='{Convert.ToDateTime(date\_completion)}' where id='{id\_us}'";

var command = new NpgsqlCommand(changeQuery, db.getConnection());

command.ExecuteNonQuery();

MessageBox.Show("Запись успешно изменена!"); }}

public void Add\_user()

{

var id\_us = textBox6.Text;

var FIO = textBox2.Text;

var emaill = textBox3.Text;

var adress = textBox4.Text;

var phone = textBox5.Text;

var date = dateTimePicker1.Text;

var date\_issue = dateTimePicker2.Text;

var date\_completion = dateTimePicker3.Text;

//генерация абонемента

Random rnd = new Random();

var ticket = 0;

while (true)

{

int n = 0;

ticket = rnd.Next(1000, 10000);

foreach (DataGridViewRow row in dataGridView1.Rows)

{

if (row.Cells[0].Value != null) // Need to check for null if new row is exposed

{

int cellId = Convert.ToInt32(row.Cells[6].Value);

if (cellId != ticket)

n++;}}

if (n == dataGridView1.Rows.Count)

break; }

int idd = Convert.ToInt32(id\_us) + 1;

var addQuery = $"insert into reader (id, name, emaill, adress, date, telephone, number\_ticket, date\_release, validity\_period, role ) values ('{idd}', '{FIO}', '{emaill}', '{adress}', '{Convert.ToDateTime(date)}', '{phone}', '{ticket}', '{Convert.ToDateTime(date\_issue)}', '{Convert.ToDateTime(date\_completion)}','читатель')";

var command = new NpgsqlCommand(addQuery, db.getConnection());

command.ExecuteNonQuery();

MessageBox.Show("Запись успешно создана!"); }

private void Clear()

{

int currentMaximumId = -1;

foreach (DataGridViewRow row in dataGridView1.Rows)

{

if (row.Cells[0].Value != null) // Need to check for null if new row is exposed

{

int cellId = Convert.ToInt32(row.Cells[0].Value);

if (cellId > currentMaximumId)

currentMaximumId = cellId;} }

textBox6.Text = Convert.ToString(currentMaximumId + 1); // идентификатор

textBox2.Clear(); //имя

textBox3.Clear();//эл. почта

textBox4.Clear(); // адрес

textBox5.Clear(); //телефон

dateTimePicker1.ResetText(); //дата

dateTimePicker2.ResetText(); //дата регистрации

dateTimePicker3.ResetText(); //окончание регистрации }

int selectedRow;

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

button1.Text = "Сохранить";

button2.Visible = true;

selectedRow = e.RowIndex;

if (e.RowIndex >= 0)

{

DataGridViewRow row = dataGridView1.Rows[selectedRow];

textBox6.Text = row.Cells[0].Value.ToString(); // идентификатор

textBox2.Text=row.Cells[1].Value.ToString(); //имя

textBox3.Text=row.Cells[2].Value.ToString(); //эл. почта

textBox4.Text= row.Cells[3].Value.ToString(); // адрес

textBox5.Text= row.Cells[5].Value.ToString(); //телефон

dateTimePicker1.Text=row.Cells[4].Value.ToString(); //дата

dateTimePicker2.Text=row.Cells[7].Value.ToString(); //дата регистрации

dateTimePicker3.Text=row.Cells[8].Value.ToString(); //окончание регистрации} }

private void textBox1\_TextChanged(object sender, EventArgs e)

{

Search(dataGridView1); }

private void button1\_Click\_1(object sender, EventArgs e)

{

if (button1.Text == "Сохранить")

Update\_user();

else

{

Add\_user();

Update\_table(dataGridView1);}}

private void button3\_Click(object sender, EventArgs e)

{

label9.Visible = true;

Clear();

button2.Visible = false;

button1.Text = "Ок"; }

private void button2\_Click(object sender, EventArgs e)

{

Delate();

Update\_table(dataGridView1);

Clear();}

private void читательскийБилетToolStripMenuItem\_Click(object sender, EventArgs e)

{

OrderManagment bs = new OrderManagment();

bs.Show();

this.Visible = false;}

private void выходToolStripMenuItem\_Click(object sender, EventArgs e)

{

Authorization bs = new Authorization();

bs.Show();

this.Visible = false;}

private void просмотрИзданийToolStripMenuItem\_Click(object sender, EventArgs e)

{

Book\_selection\_admin bs = new Book\_selection\_admin();

bs.Show();

this.Visible = false; }

private void button4\_Click(object sender, EventArgs e)

{

//Открытие отчета

Report bs;

if (comboBox2.SelectedIndex == 0)

bs = new Report(5);

else

bs = new Report(4);

bs.Show();

this.Visible = false;

}

}