

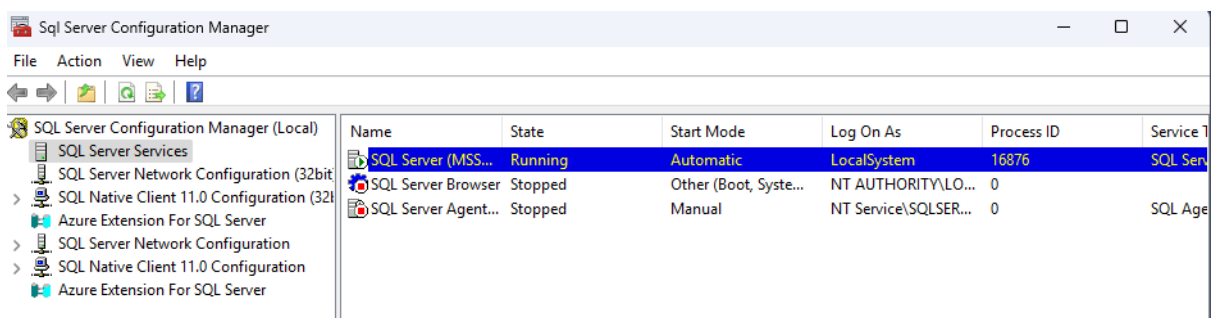
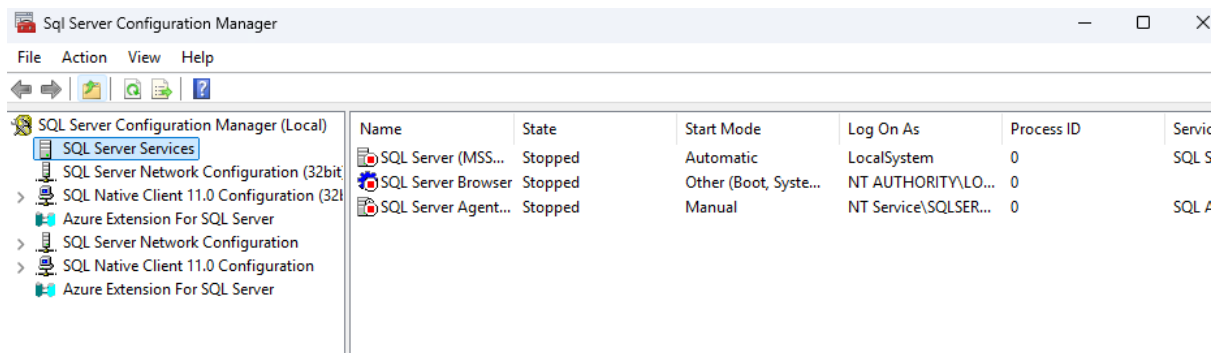
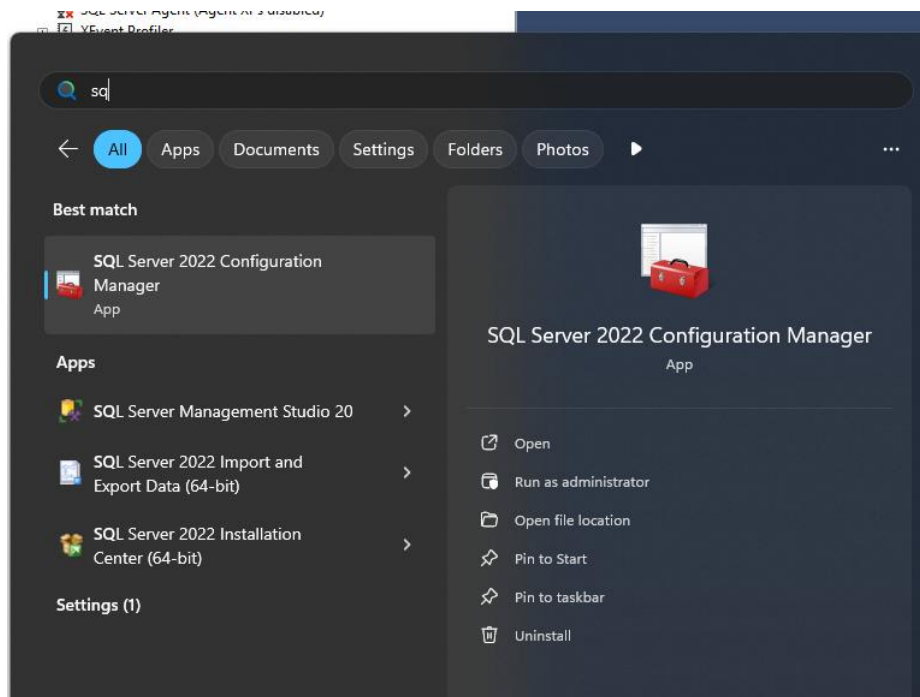
Оглавление

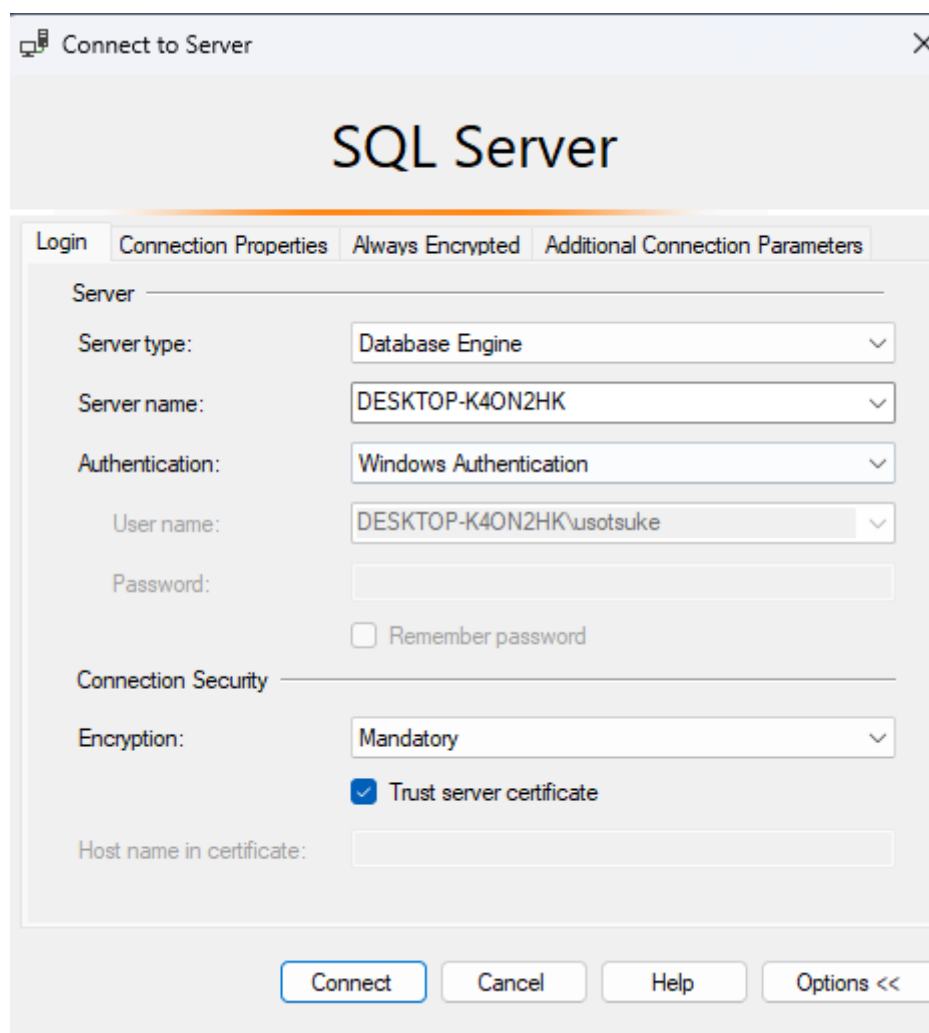
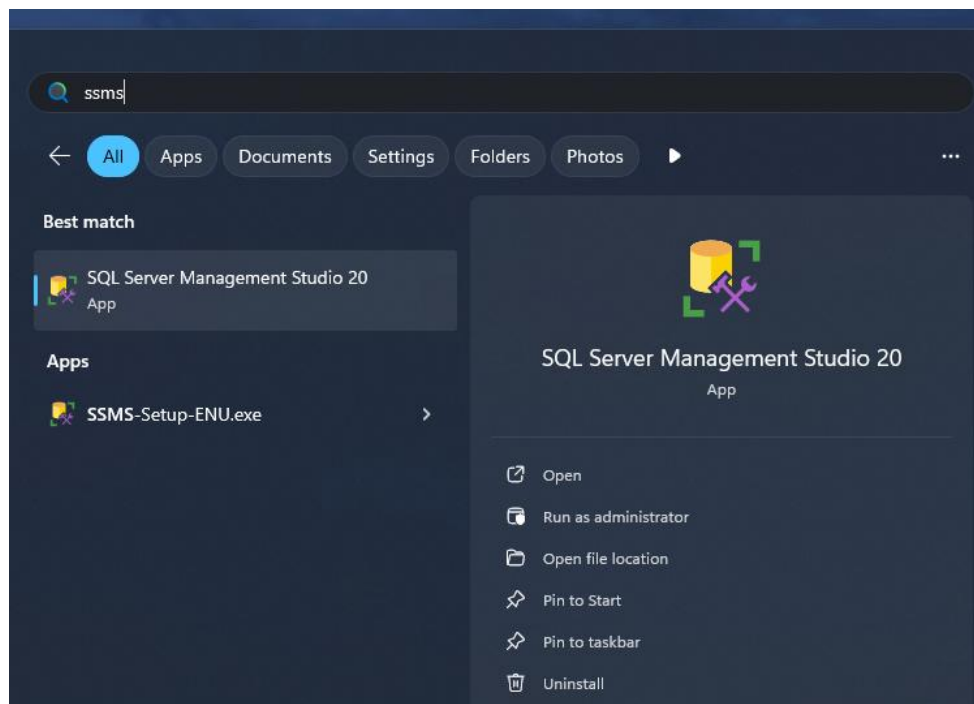
Настройка БД.....	2
Включаем сервер.....	2
Создаем нового пользователя	5
Создание БД	12
Код.....	13
Если нужно изменить БД.....	15
Создание проекта	16
Основные элементы демо-экзамена	21
DataGridView.....	21
Использование	22
ListBox.....	32
Важно.....	34
POST/GET requests into database.....	37
GET request	37
POST request.....	38
ПРОЕКТ	39
Начало.....	39
Создаем окно информации о номерах	40
Создаем окно резервирования номеров.....	41
Создаем DataGridView, где выбираем таблицу room	41
Создаем функционал создания резервирования	42
Кнопка резервирования	43
Создаем функционал удаления резервирования	44
Кнопка Удаления резервирования.....	45
Итог	45
Окно заселения	46
Функционал заселения.....	47
Создаем окно выселения.....	49
Создаем окно календаря уборки	50
Создание уборки.....	50

Создание функционала создания	52
Создание Уборки	52
Создаем функционал удаления	52
Создаем изменение статуса уборки	53
Создание функционала изменения статуса уборки	54
Создание окна Владельца	55
Функционал получения процента загрузки номерного фонда.....	56
Функционал получения ADR	56
Функционал получения RevPAR	57
Функционал закрытия Main Forms	57
Приложения	59
GitHub.....	59
Database Schema	59

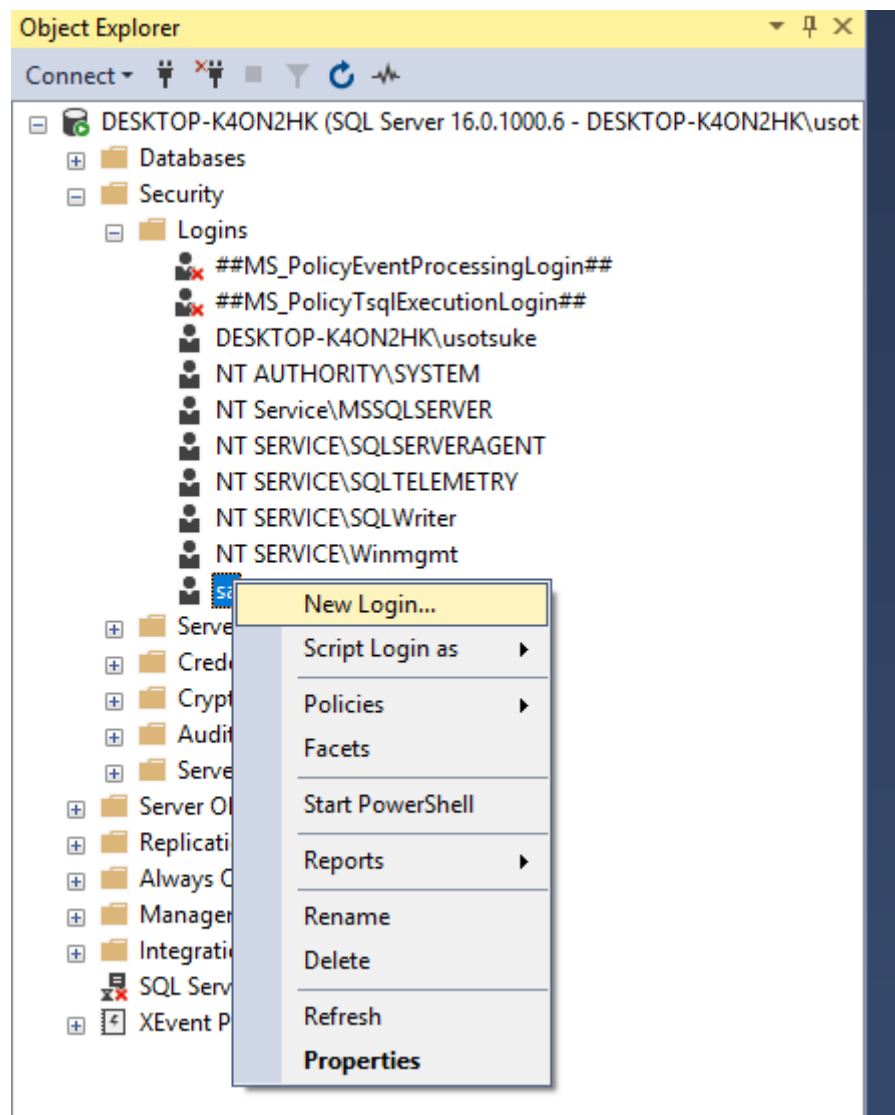
Настройка БД

Включаем сервер





Создаем нового пользователя



Login - New

Select a page

General
Server Roles
User Mapping
Securables
Status

Connection

Server:
DESKTOP-K4ON2HK
Connection:
DESKTOP-K4ON2HK\usotsuke
View connection properties

Progress

Ready

Script Help

Login name: Student Search...

☐ Windows authentication
☐ Microsoft Entra ID authentication
☒ SQL Server authentication

Password:
Confirm password:
☐ Specify old password
Old password:
☒ Enforce password policy
☒ Enforce password expiration
☐ User must change password at next login

☐ Mapped to certificate
☐ Mapped to asymmetric key
☐ Map to Credential Add

Mapped Credentials

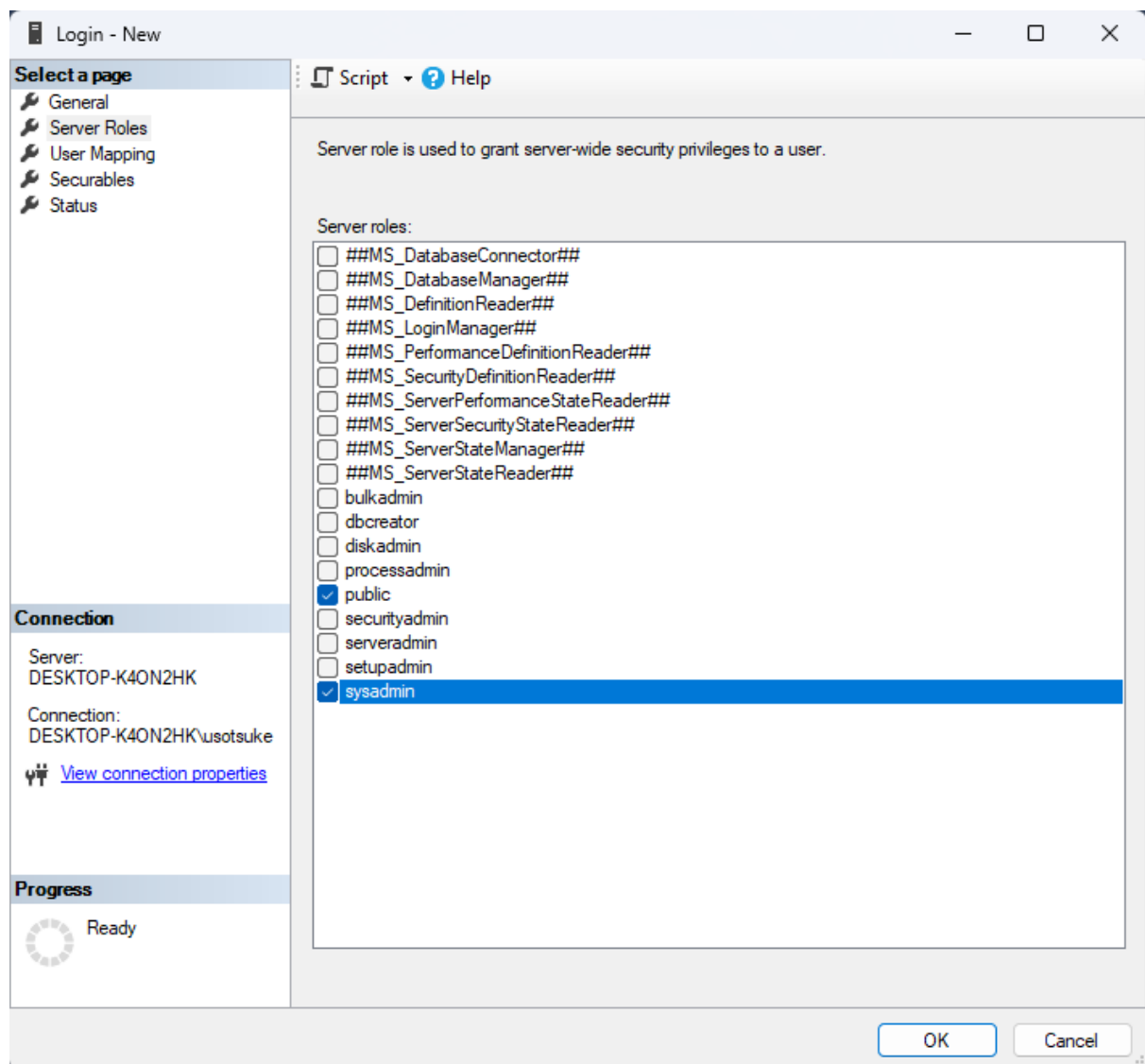
Credential	Pro

Remove

Default database: hotel
Default language: <default>

OK

Cancel



Login - New

Select a page

General

Server Roles

User Mapping

Securables

Status

Connection

Server:
DESKTOP-K4ON2HK

Connection:
DESKTOP-K4ON2HK\usotsuke

View connection properties

Progress

Ready

Script

Help

Settings

Permission to connect to database engine:

Grant

Deny

Login:

Enabled

Disabled

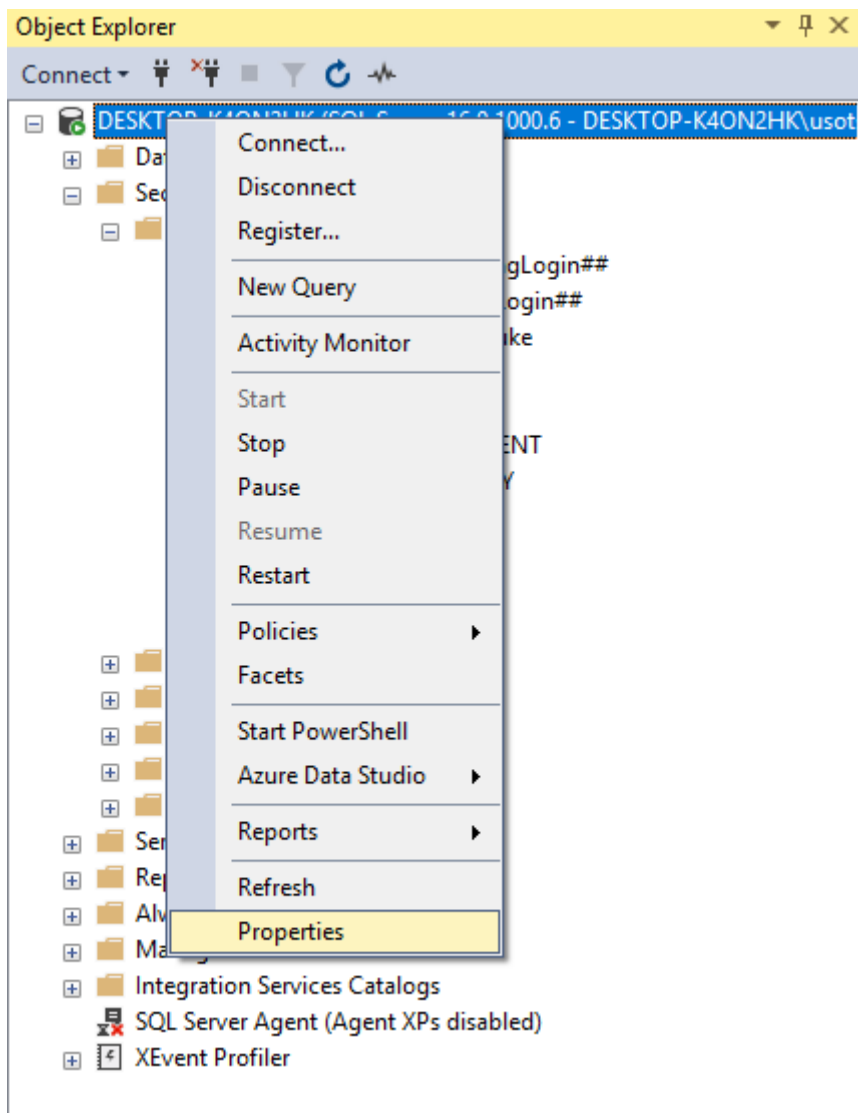
Status

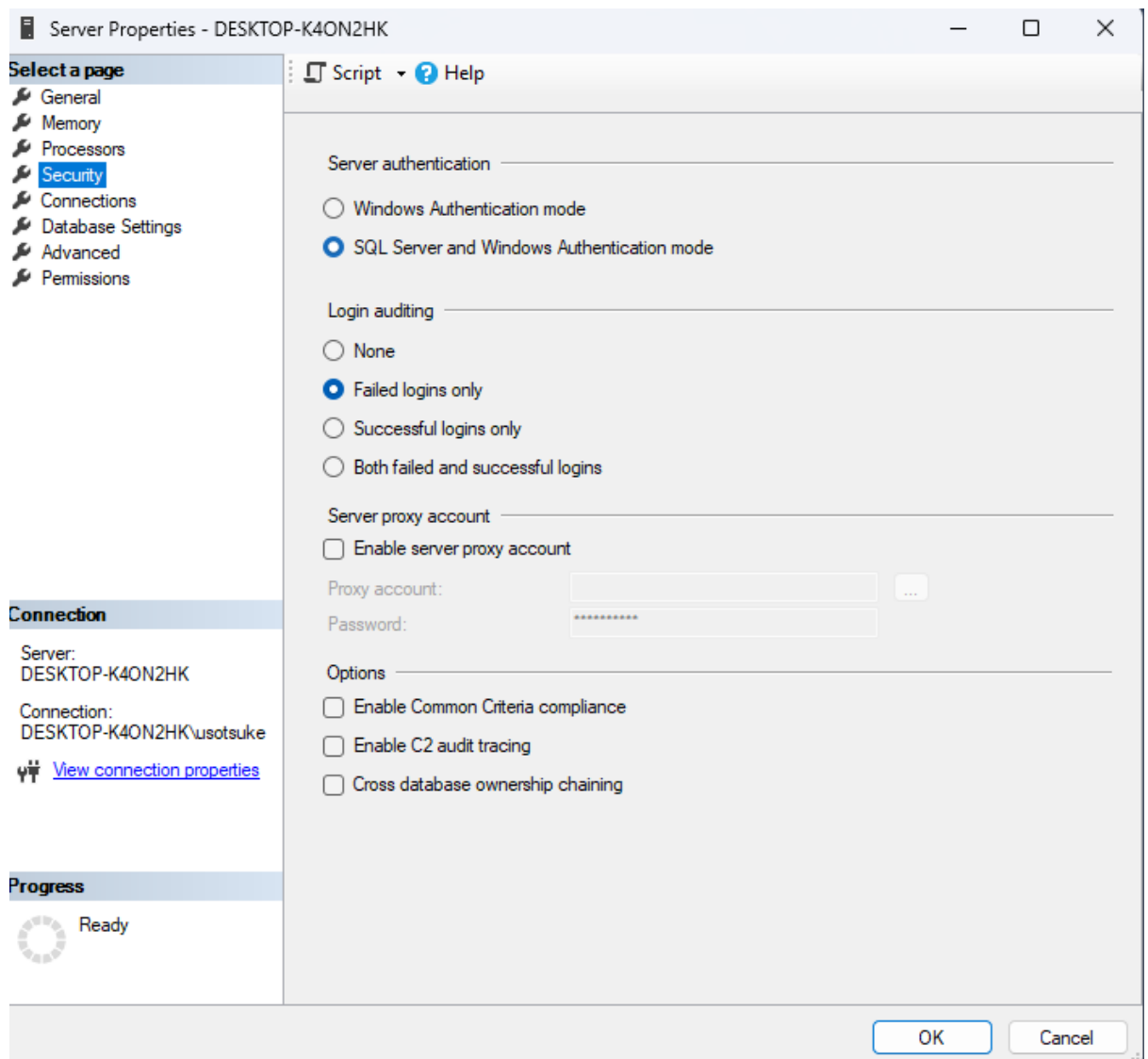
SQL Server authentication:

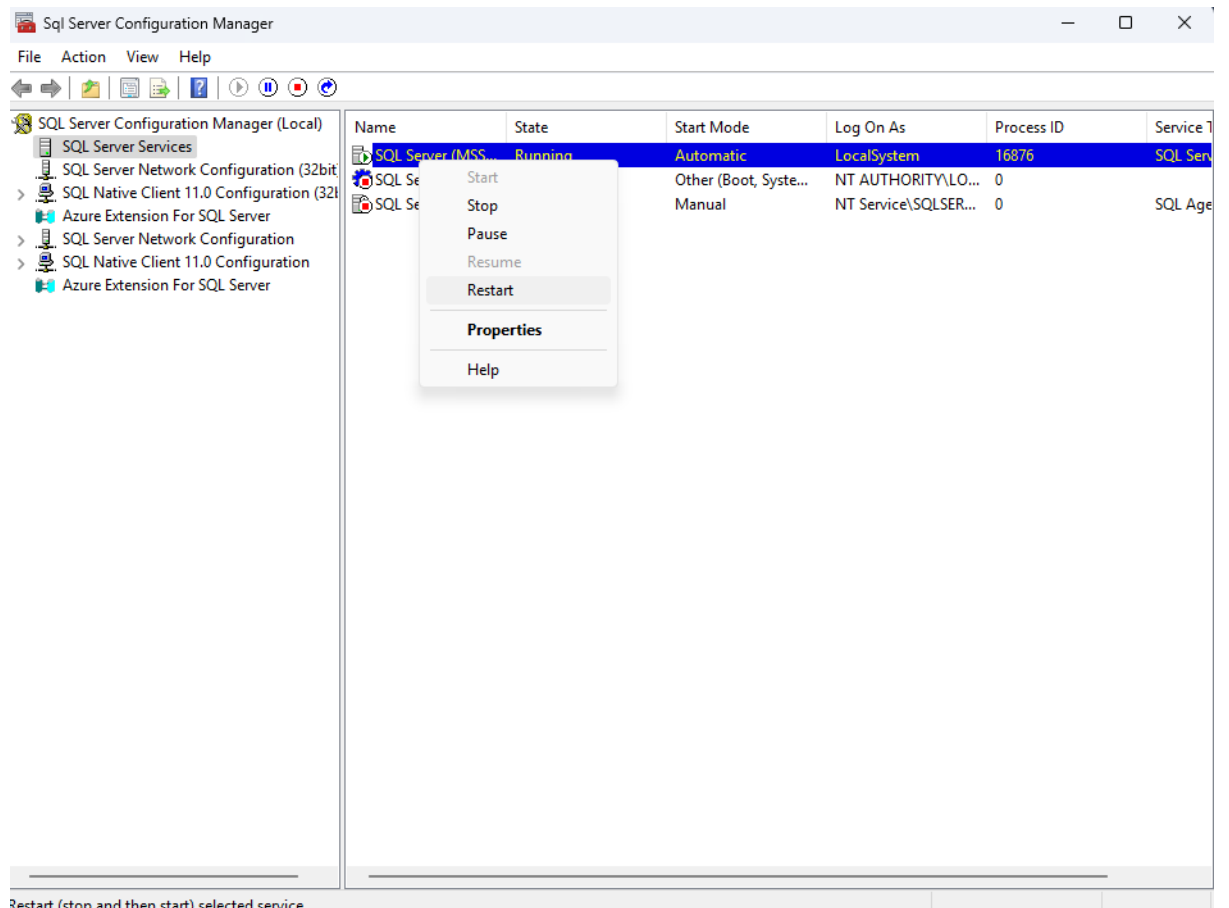
Login is locked out

OK

Cancel

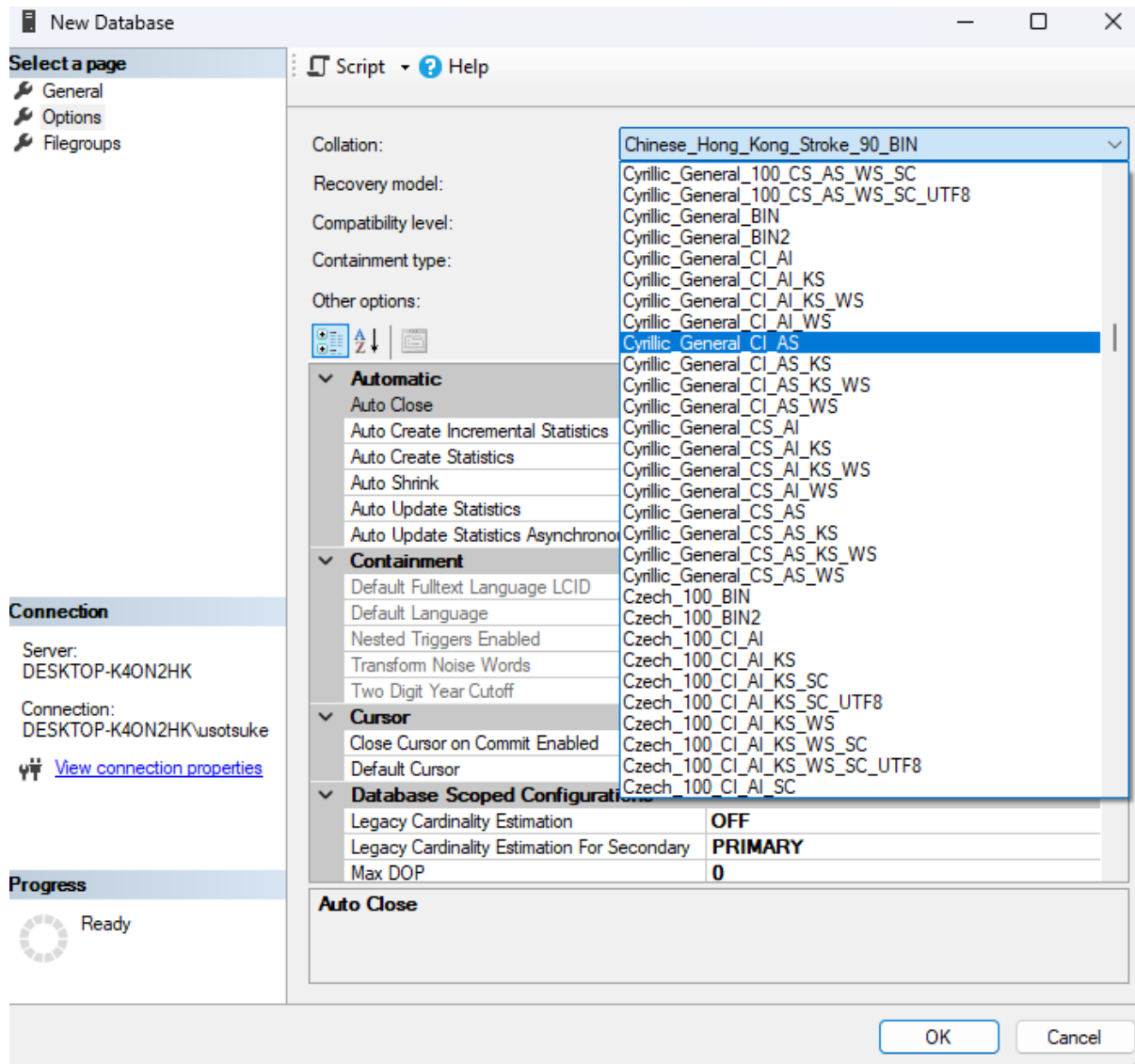






Создание БД

Создаем БД из SSMS, выбирая Collation!!!!



Код

```
USE Hotel;
SET DATEFORMAT ymd;
CREATE TABLE Users(
    Login        VARCHAR(50) PRIMARY KEY,
    Password     VARCHAR(50),
    Role         TINYINT
);
CREATE TABLE Room(
    Id           INT PRIMARY KEY,
    Type         VARCHAR(100),
    Floor        TINYINT,
    Status       VARCHAR(50) DEFAULT 'Чистый',
    Money        INT
);
CREATE TABLE RoomReservation(
    Id           INT PRIMARY KEY IDENTITY,
    RoomId       INT,
    Fdate        DATE,
    Ldate        DATE,
    ClientFname  VARCHAR(300),
    Guests       INT DEFAULT 1,
    FOREIGN KEY (RoomId) REFERENCES Room (Id),
);
CREATE TABLE Employee(
    Id           INT PRIMARY KEY,
    Fname        VARCHAR(300)
);
CREATE TABLE CleaningCalendar(
    Id           INT PRIMARY KEY IDENTITY,
    RoomId       INT,
    EmployeeId   INT,
    Day          DATE,
    Time         VARCHAR(10),
    Status       VARCHAR(50),
    FOREIGN KEY (RoomId) REFERENCES Room (Id),
    FOREIGN KEY (EmployeeId) REFERENCES Employee (Id),
);
Create Table Analytic(
    Id INT PRIMARY KEY IDENTITY,
    RoomId INT,
    Guests INT,
    Money INT,
    days INT,
    Day DATE,
```

```

FOREIGN KEY (RoomId) REFERENCES Room (Id),
)
INSERT INTO Users VALUES ('Administrator', '123123', 1), ('Owner', 'qwerty', 2);
INSERT INTO Room VALUES ('101', 'Одноместный стандарт', '1', 'Чистый', 5000);
INSERT INTO Room VALUES ('102', 'Одноместный стандарт', '1', 'Чистый', 5000);
INSERT INTO Room VALUES ('103', 'Одноместный эконом', '1', 'Чистый', 5000);
INSERT INTO Room VALUES ('104', 'Одноместный эконом', '1', 'Чистый', 5000);
INSERT INTO Room VALUES ('105', 'Одноместный стандарт', '1', 'Чистый', 5000);
INSERT INTO Room VALUES ('106', 'Стандарт двухместный с 2 отдельными
кроватями', '1', 'Чистый', 7000);
INSERT INTO Room VALUES ('107', 'Стандарт двухместный с 2 отдельными
кроватями', '1', 'Чистый', 7000);
INSERT INTO Room VALUES ('108', 'Эконом двухместный с 2 отдельными
кроватями', '1', 'Чистый', 7000);
INSERT INTO Room VALUES ('109', 'Эконом двухместный с 2 отдельными
кроватями', '1', 'Чистый', 7000);
INSERT INTO Room VALUES ('110', '3-местный бюджет', '1', 'Чистый', 7000);
INSERT INTO Room VALUES ('201', 'Бизнес с 1 или 2 кроватями', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('202', 'Бизнес с 1 или 2 кроватями', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('203', 'Бизнес с 1 или 2 кроватями', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('204', 'Двухкомнатный двухместный стандарт с 1 или 2
кроватями', '2', 'Назначен к уборке', 7000);
INSERT INTO Room VALUES ('205', 'Двухкомнатный двухместный стандарт с 1 или 2
кроватями', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('206', 'Двухкомнатный двухместный стандарт с 1 или 2
кроватями', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('207', 'Одноместный стандарт', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('208', 'Одноместный стандарт', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('209', 'Одноместный стандарт', '2', 'Чистый', 7000);
INSERT INTO Room VALUES ('301', 'Студия', '3', 'Чистый', 7000);
INSERT INTO Room VALUES ('302', 'Студия', '3', 'Чистый', 7000);
INSERT INTO Room VALUES ('303', 'Студия', '3', 'Чистый', 7000);
INSERT INTO Room VALUES ('304', 'Люкс с 2 двуспальными кроватями', '3',
'Чистый', 7000);
INSERT INTO Room VALUES ('305', 'Люкс с 2 двуспальными кроватями', '3',
'Чистый', 7000);
INSERT INTO Room VALUES ('306', 'Люкс с 2 двуспальными кроватями', '3',
'Чистый', 7000);
INSERT INTO Employee VALUES (1, 'Иванов Иван Иванович');
INSERT INTO Employee VALUES (2, 'Дмитриев Дмитрий Дмитриевич');
INSERT INTO Employee VALUES (3, 'Александрова Александра Александровна');
INSERT INTO Employee VALUES (4, 'Артемьев Артем Артемович');
INSERT INTO Employee VALUES (5, 'Евгеньев Евгений Евгеньевич');

```

```

INSERT INTO RoomReservation (RoomId, ClientFname, Fdate, Ldate, Guests) VALUES
(101, 'Шевченко Ольга Викторовна', '14.02.2025', '02.03.2025', 1),
(104, 'Семеняка Юрий Геннадьевич', '23.02.2025', '02.02.2025', 1),
(105, 'Савельев Олег Иванович', '01.03.2025', '07.03.2025', 1),
(107, 'Бунин Эдуард Михайлович', '27.02.2025', '22.04.2025', 1),
(107, 'Бахшиев Павел Иннокентьевич', '24.02.2025', '17.03.2025', 1 ),
(108, 'Тюренкова Наталья Сергеевна', '15.02.2025', '20.03.2025', 1 ),
(109, 'Любашева Галина Аркадьевна', '27.02.2025', '12.03.2025', 1 ),
(110, 'Александров Петр Константинович', '14.02.2025', '02.02.2025', 1 ),
(201, 'Мазалова Ольга Николаевна', '24.02.2025', '17.03.2025', 1 ),
(203, 'Лапшин Виктор Романович', '25.02.2025', '07.03.2025', 1 ),
(205, 'Гусев Семен Петрович', '01.03.2025', '04.03.2025', 1 ),
(206, 'Гладилина Вера Михайловна', '02.02.2025', '02.02.2025', 1 ),
(207, 'Масюк Динара Викторовна', '25.02.2025', '04.03.2025', 1 ),
(208, 'Лукин Илья Федорович', '25.02.2025', '04.03.2025', 1 ),
(304, 'Филь Марина Федоровна', '28.02.2025', '15.03.2025', 1 ),
(102, 'Мазалова Ирина Львовна', '28.02.2025', '15.03.2025', 1),
(209, 'Петров Станислав Игоревич', '27.02.2025', '28.02.2025', 1),
(306, 'Михайлов Игорь Вадимович', '11.02.2025', '28.02.2025', 1);

```

Если нужно изменить БД

```

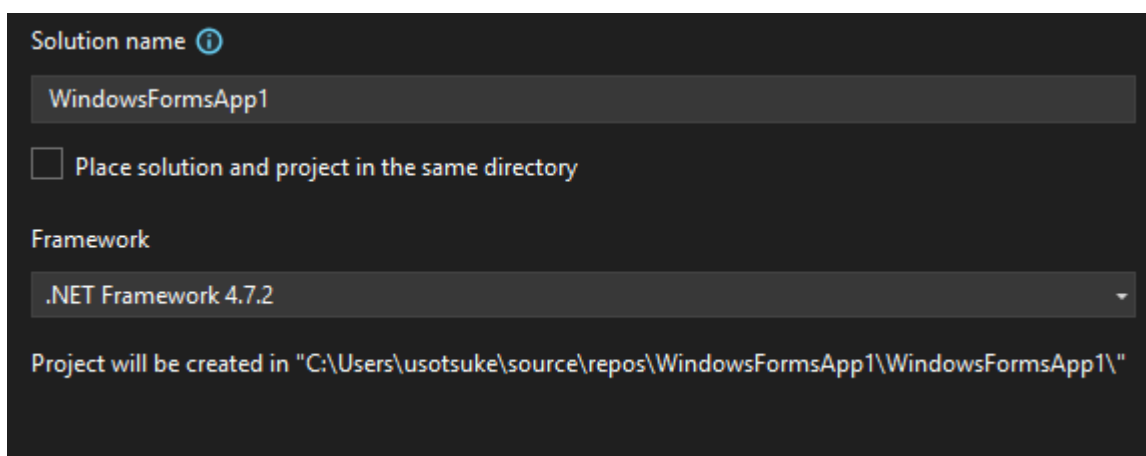
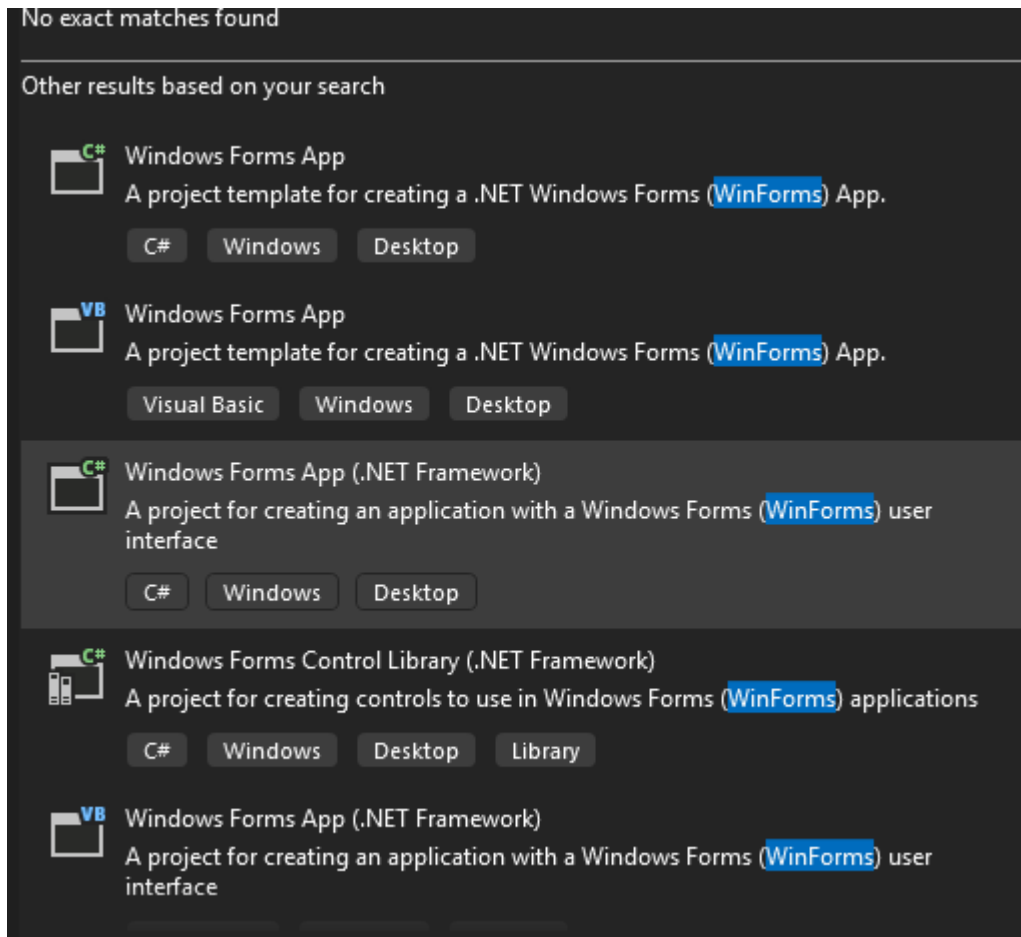
use master;
alter database Hotel set single_user with rollback immediate

drop database Hotel;

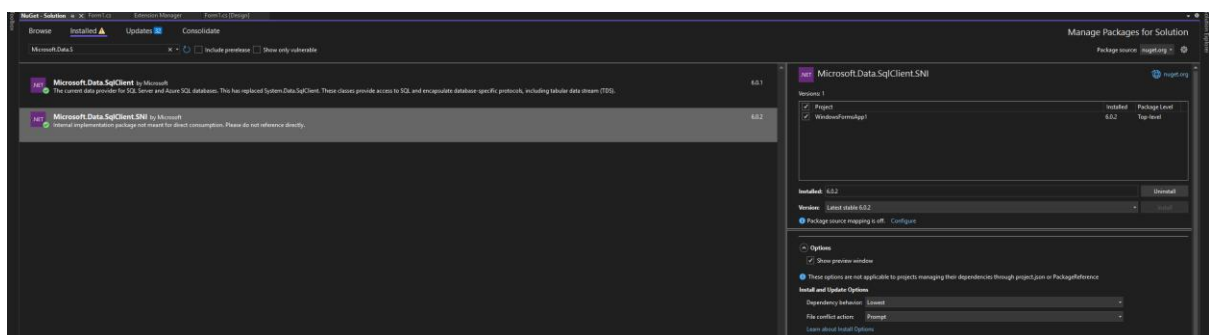
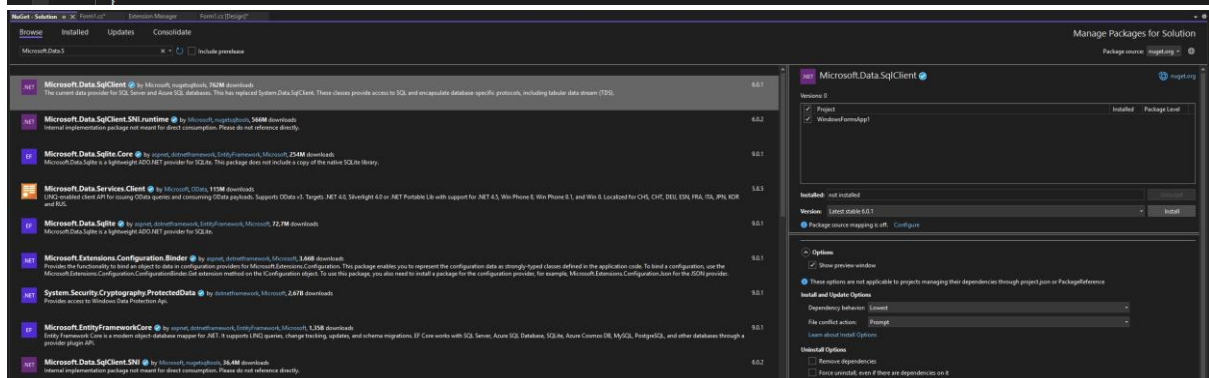
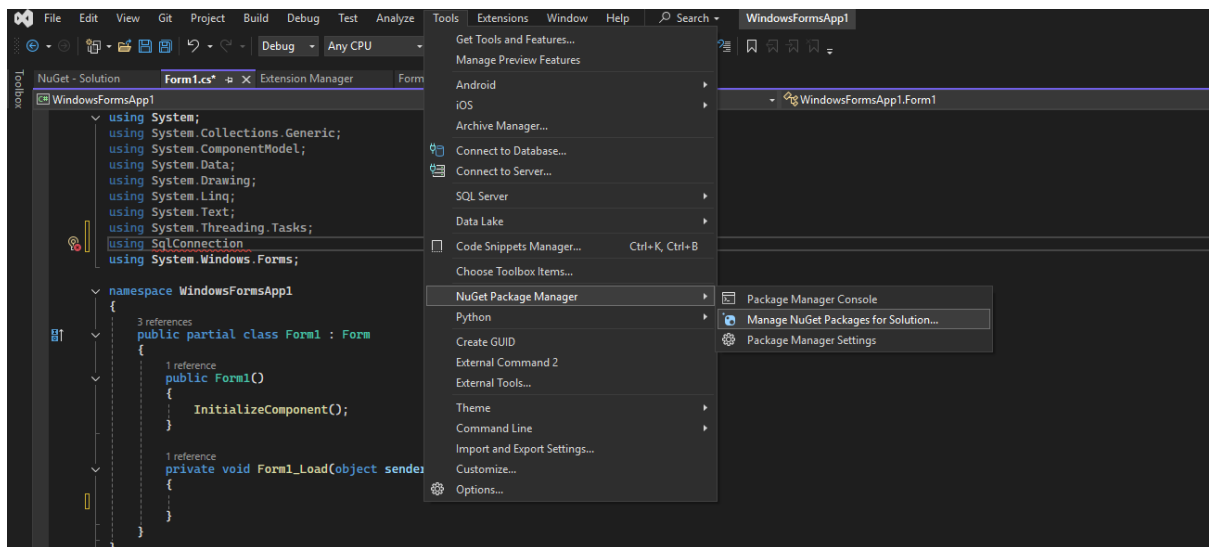
```

Далее опять создаем БД с COLLATE.

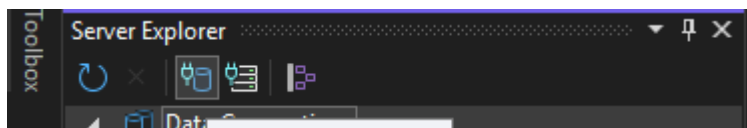
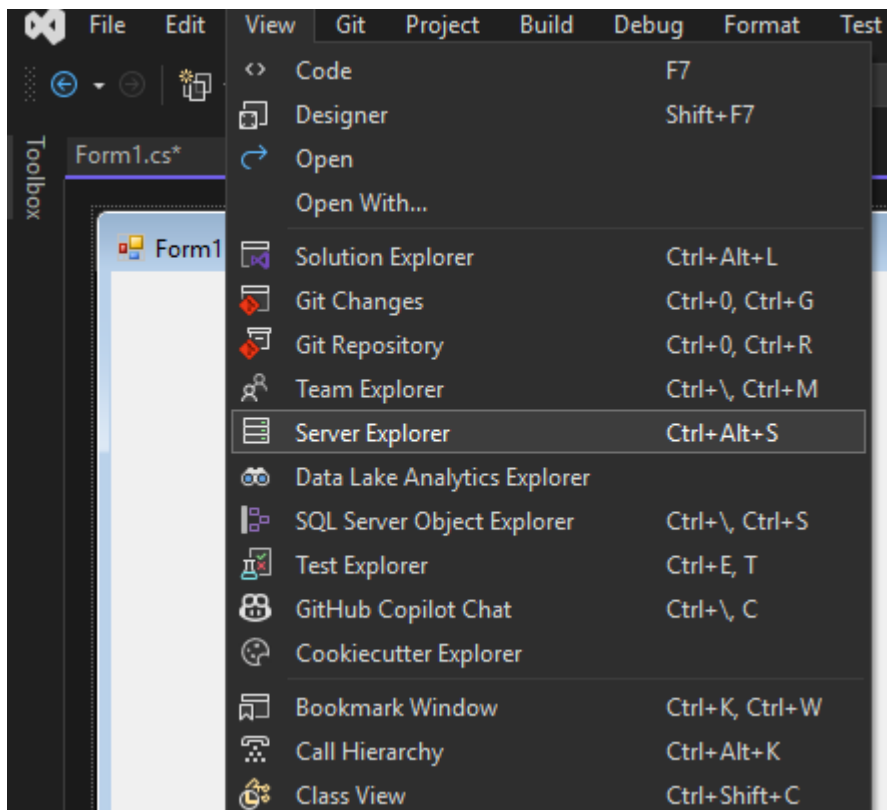
Создание проекта



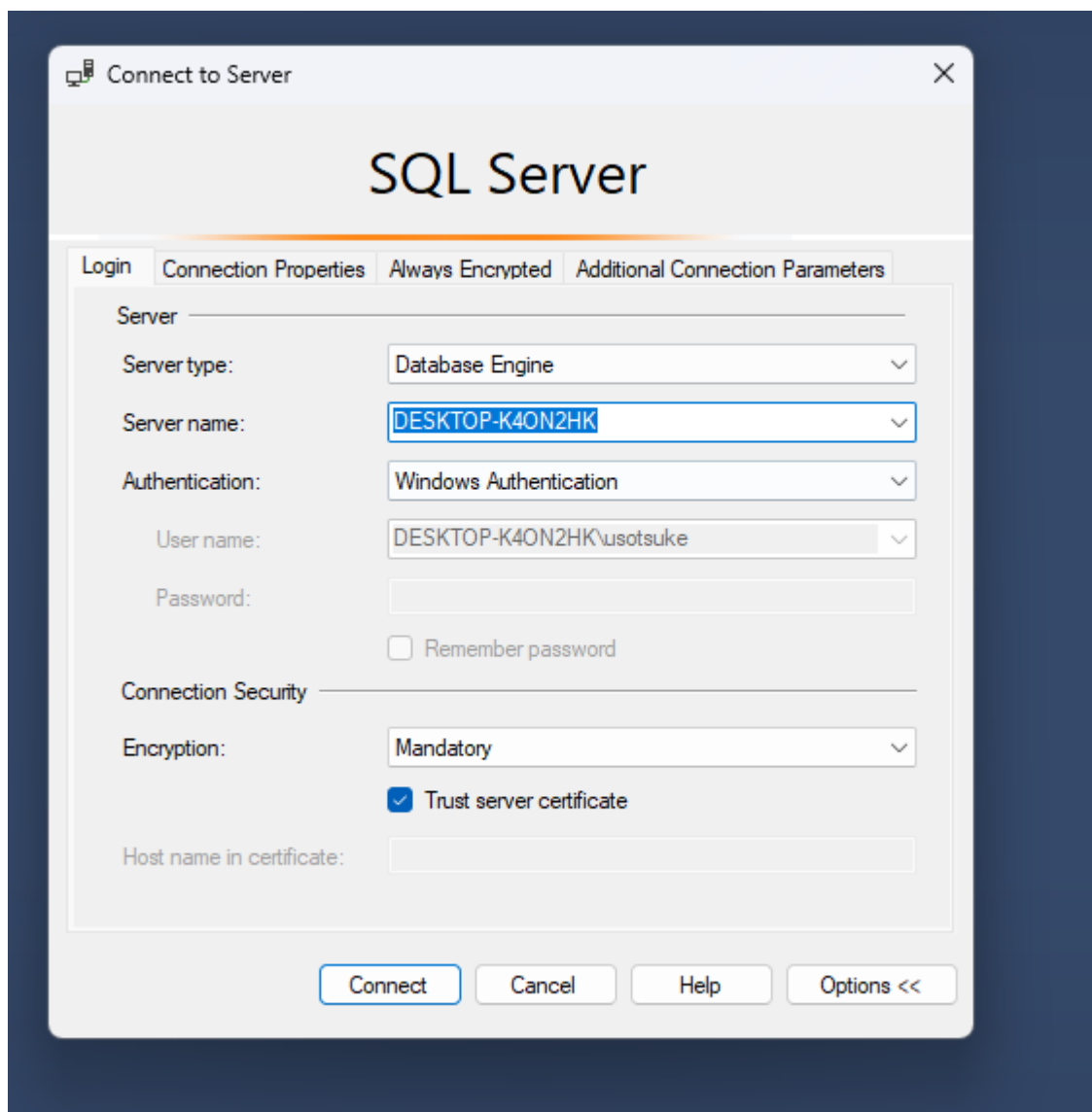
После создания проекта устанавливаем библиотеку SQL



Подключаем сервер к IDE



Копируем из SSMS Id сервера



Вставляем в меню IDE, авторизируемся, используя логин и пароль созданного юзера в БД и выбираем следующие параметры:

Add Connection ? X

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
Microsoft SQL Server (Microsoft SqlClient) Change...

Server name:
DESKTOP-K4ON2HK Refresh

Log on to the server

Authentication: SQL Server Authentication

User name: Student

Password: ••••••••

Encrypt: Mandatory (True)

☒ Trust Server Certificate

☐ Save my password

Connect to a database

☒ Select or enter a database name:
Hotel

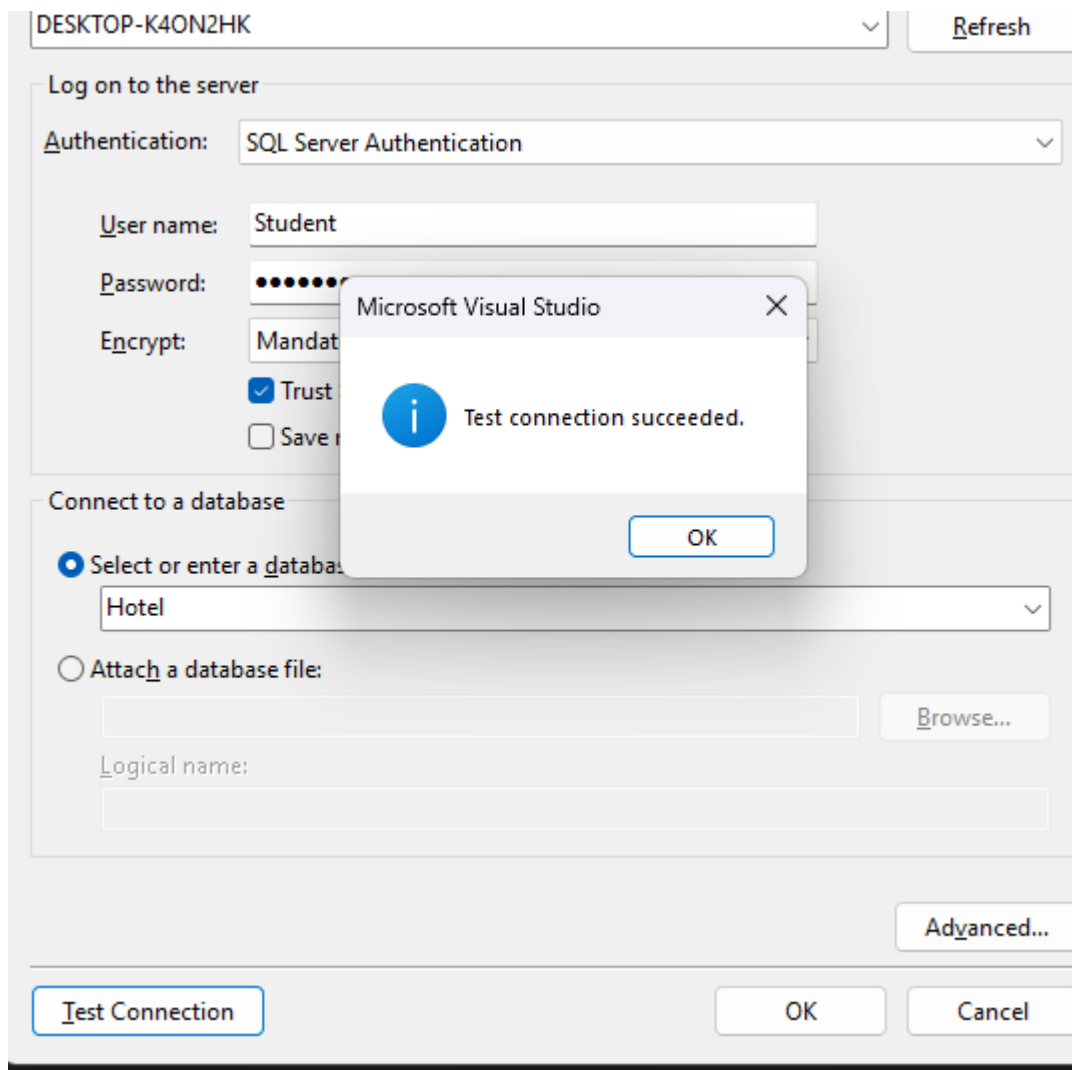
☐ Attach a database file:
Browse...

Logical name:

Advanced...

Test Connection OK Cancel

При нажатии Button “Test Connection” Должно появиться следующее окно:



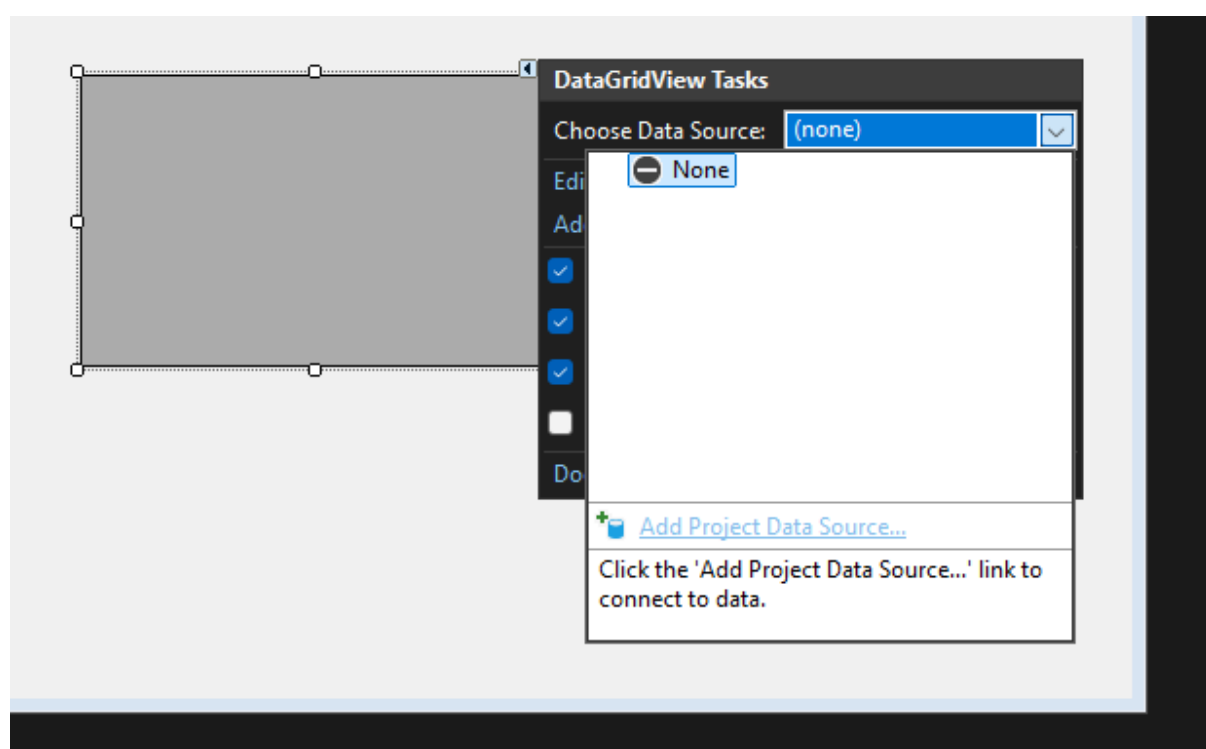
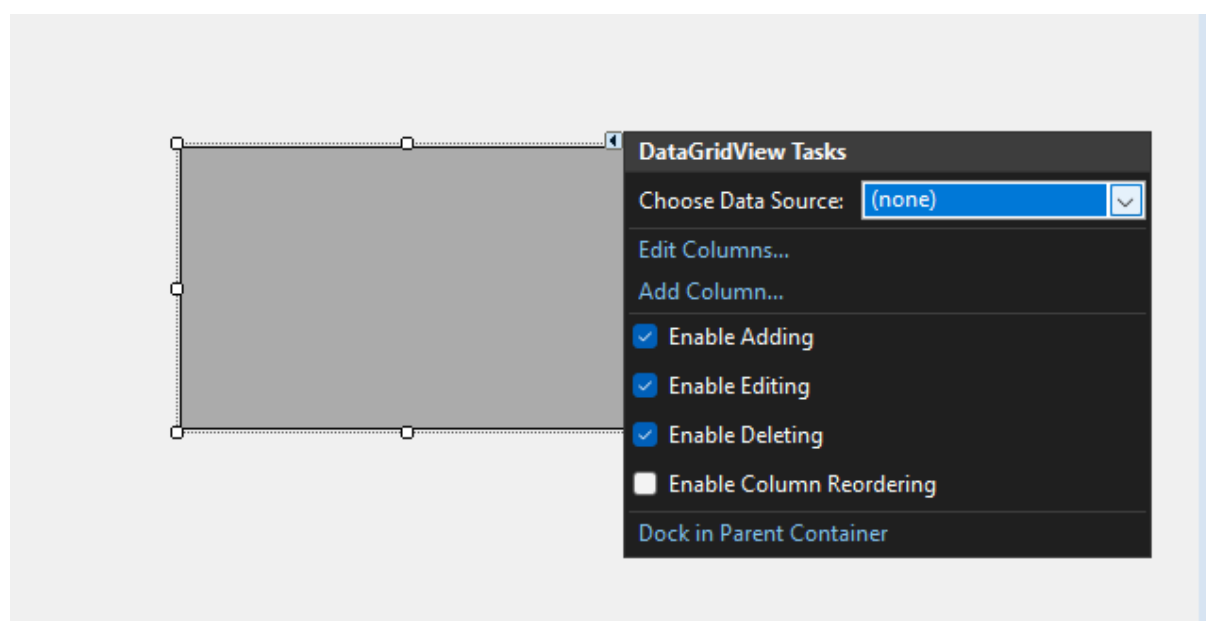
Это значит, что IDE Connection Granted и можно закрывать окно, нажав OK, если что-то пошло не так, попробуйте повторить шаги снова или посмотреть в интернете.

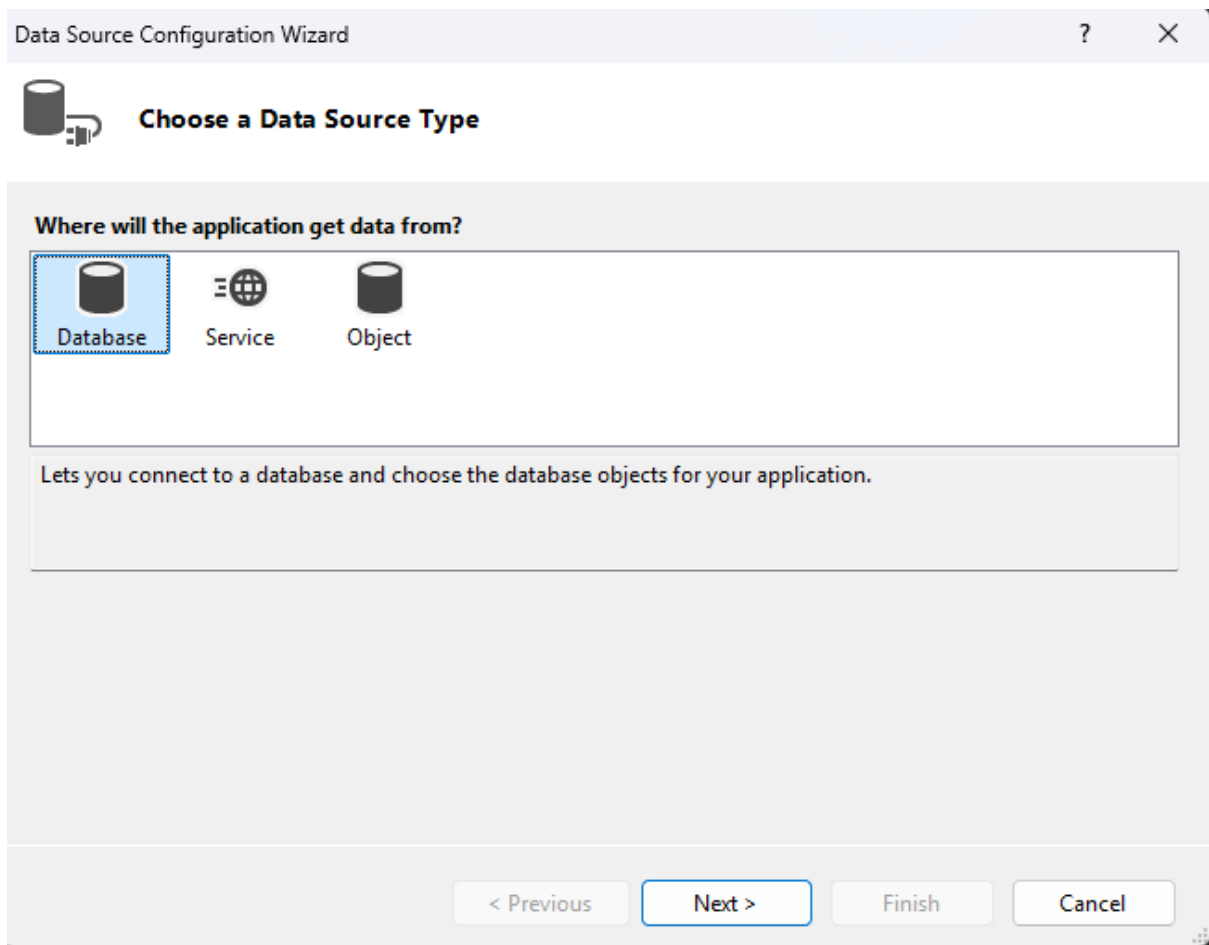
Основные элементы демо-экзамена

DataGridView

DataGridView – таблица, в которую мы выводим запрос из БД. Важно – в Community IDE Version мы не можем выводить информацию с помощью WHERE.

Использование





**Choose a Database Model**

What type of database model do you want to use?



Dataset

The database model you choose determines the types of data objects your application code uses. A dataset file will be added to your project.

< Previous

Next >

Finish

Cancel

Add Connection?×

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:

Microsoft SQL Server (Microsoft SqlClient)Change...

Server name:

▼

Refresh

Log on to the server

Authentication:

Windows Authentication▼

User name:

Password:

Encrypt:

Mandatory (True)▼

☐ Trust Server Certificate

☐ Save my password

Connect to a database

☒ Select or enter a database name:

▼

☐ Attach a database file:

Browse...

Logical name:

Advanced...

Test Connection

OK

Cancel

Add Connection?×

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:

Microsoft SQL Server (Microsoft SqlClient)Change...

Server name:

DESKTOP-K4ON2HKRefresh

Log on to the server

Authentication: SQL Server Authentication

User name: Student

Password: ●●●●●●●●

Encrypt: Mandatory (True)

☒ Trust Server Certificate

☒ Save my password

Connect to a database

☒ Select or enter a database name:

Hotel

☐ Attach a database file:

Browse...

Logical name:

Advanced...

Test Connection

OK

Cancel

**Choose Your Data Connection**

Which data connection should your application use to connect to the database?

desktop-k4on2hk.Hotel.dbo



New Connection...

This connection string appears to contain sensitive data (for example, a password), which is required to connect to the database. However, storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

- ☐ No, exclude sensitive data from the connection string. I will set this information in my application code.
- ☒ Yes, include sensitive data in the connection string.

☐ Show the connection string that you will save in the application

< Previous

Next >


Finish

Cancel

Data Source Configuration Wizard

?

×



Save the Connection String to the Application Configuration File

Storing connection strings in your application configuration file eases maintenance and deployment. To save the connection string in the application configuration file, enter a name in the box and then click Next.

Do you want to save the connection string to the application configuration file?

☒ Yes, save the connection as:

HotelConnectionString

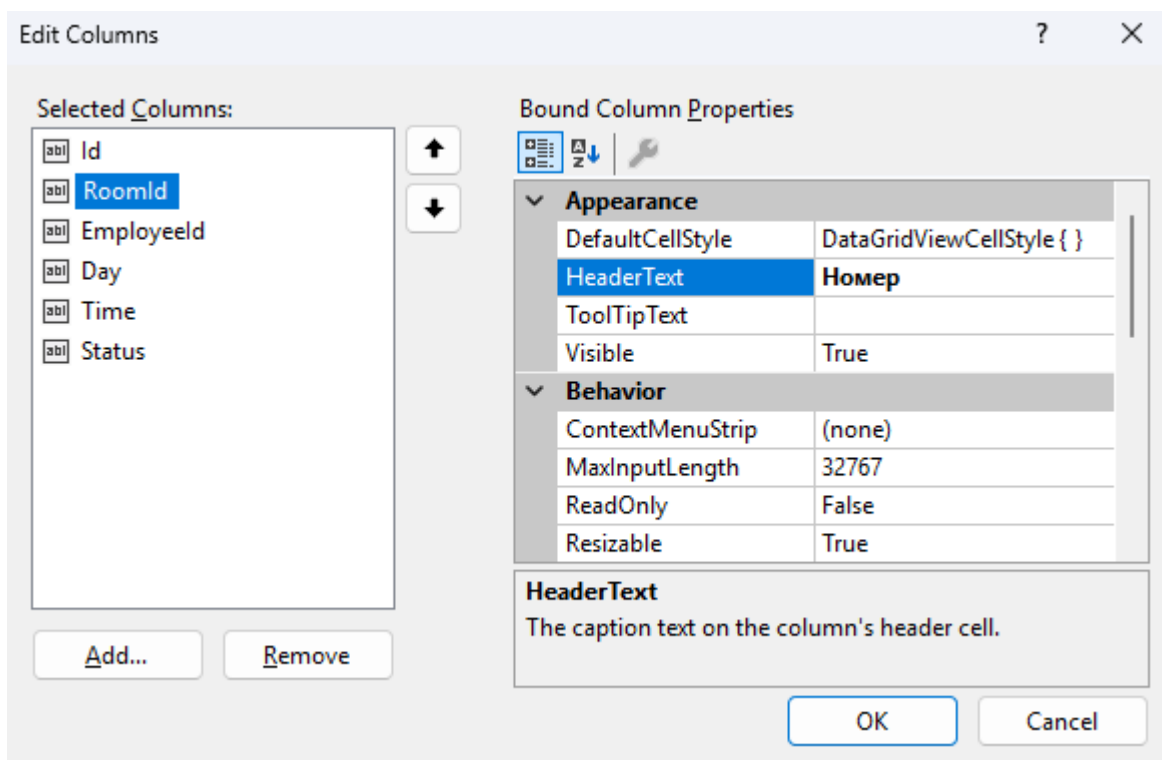
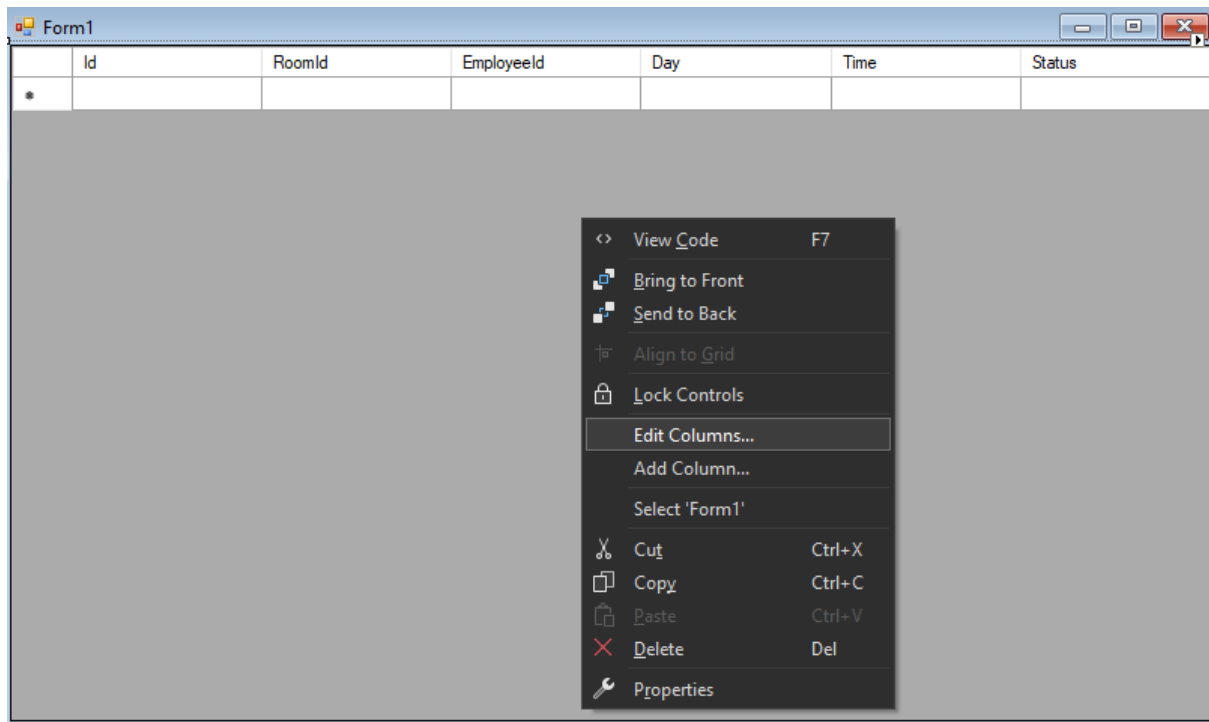
< Previous

Next >

Finish

Cancel

Layout	
Anchor	Top, Left
AutoSizeColumnsMode	Fill
AutoSizeRowsMode	None
Dock	Fill
Location	0; 0
Margin	3; 3; 3; 3
MaximumSize	0; 0
MinimumSize	0; 0
RowHeadersWidth	41



Edit Columns

Selected Columns:

- Id
- Номер
- Работник
- День
- Время
- Статус

Bound Column Properties

Appearance

DefaultCellStyle	DataGridViewCellStyle { }
HeaderText	Статус
ToolTipText	
Visible	True

Behavior

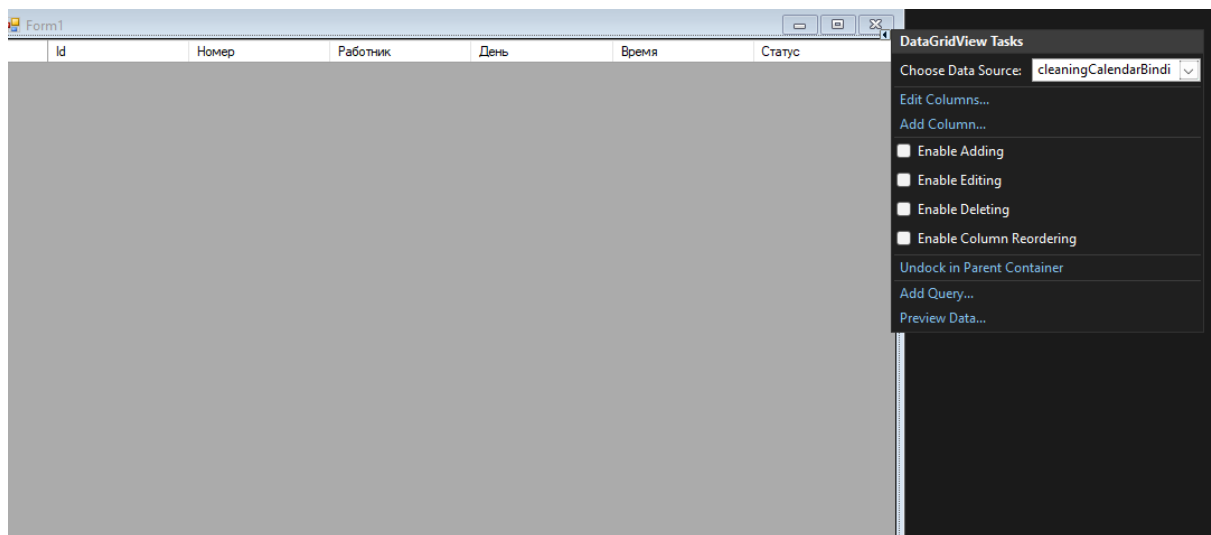
ContextMenuStrip	(none)
MaxInputLength	32767
ReadOnly	False
Resizable	True

HeaderText
The caption text on the column's header cell.

OK Cancel

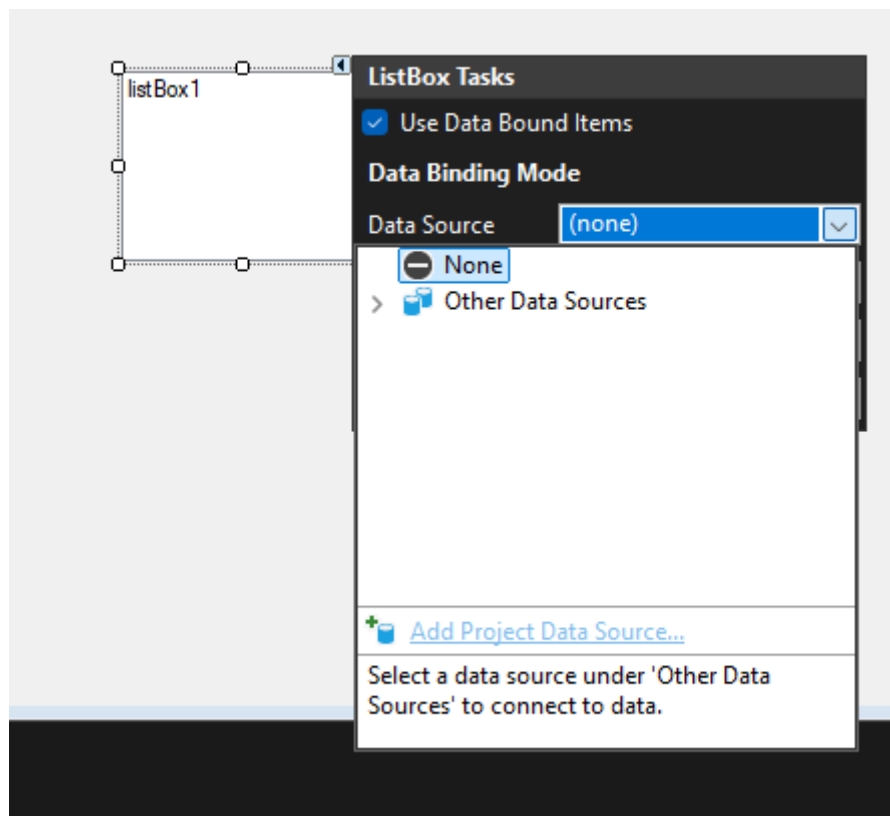
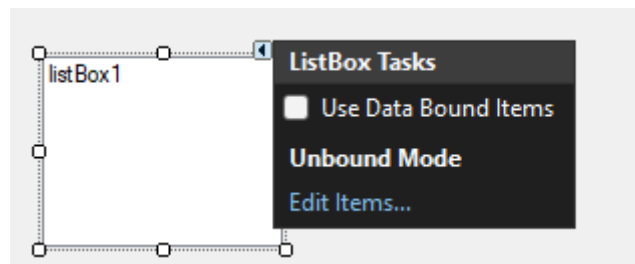
Form1

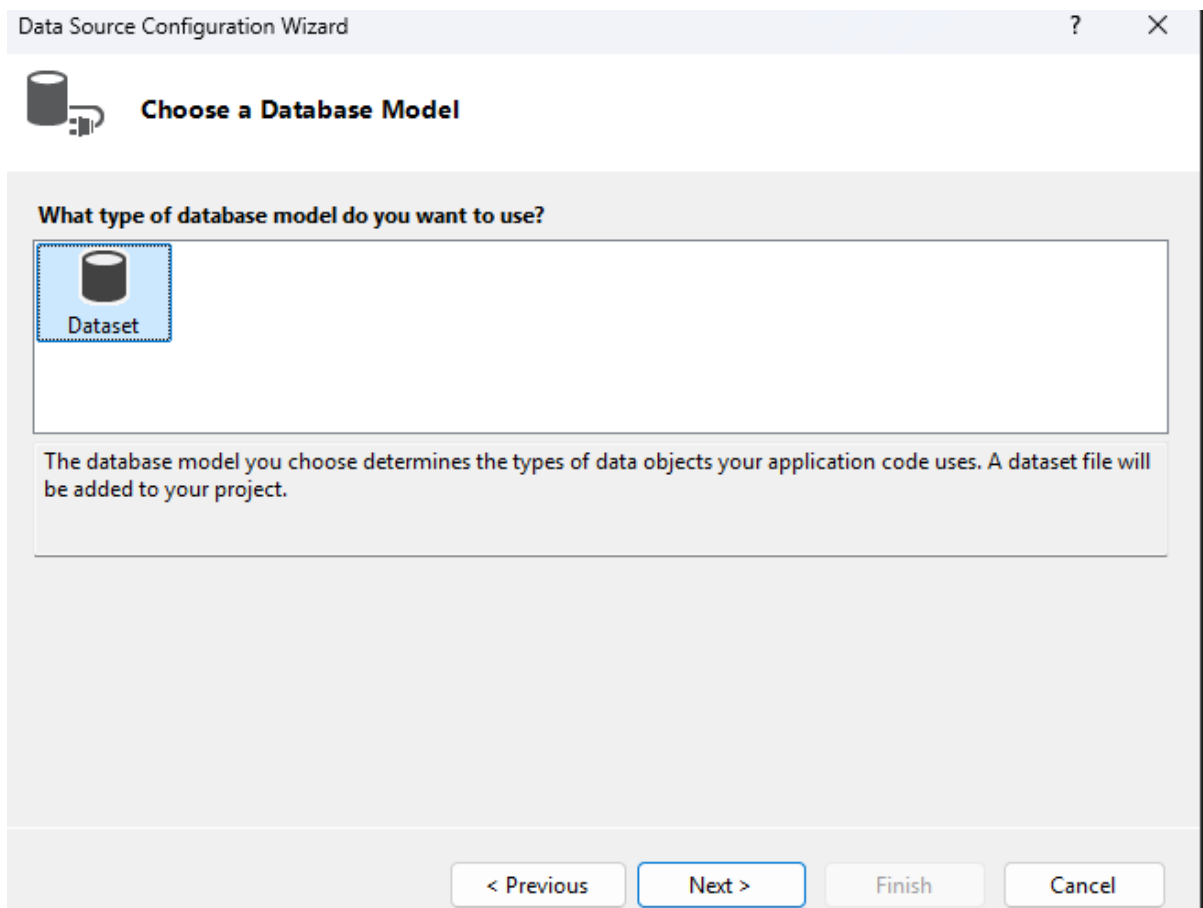
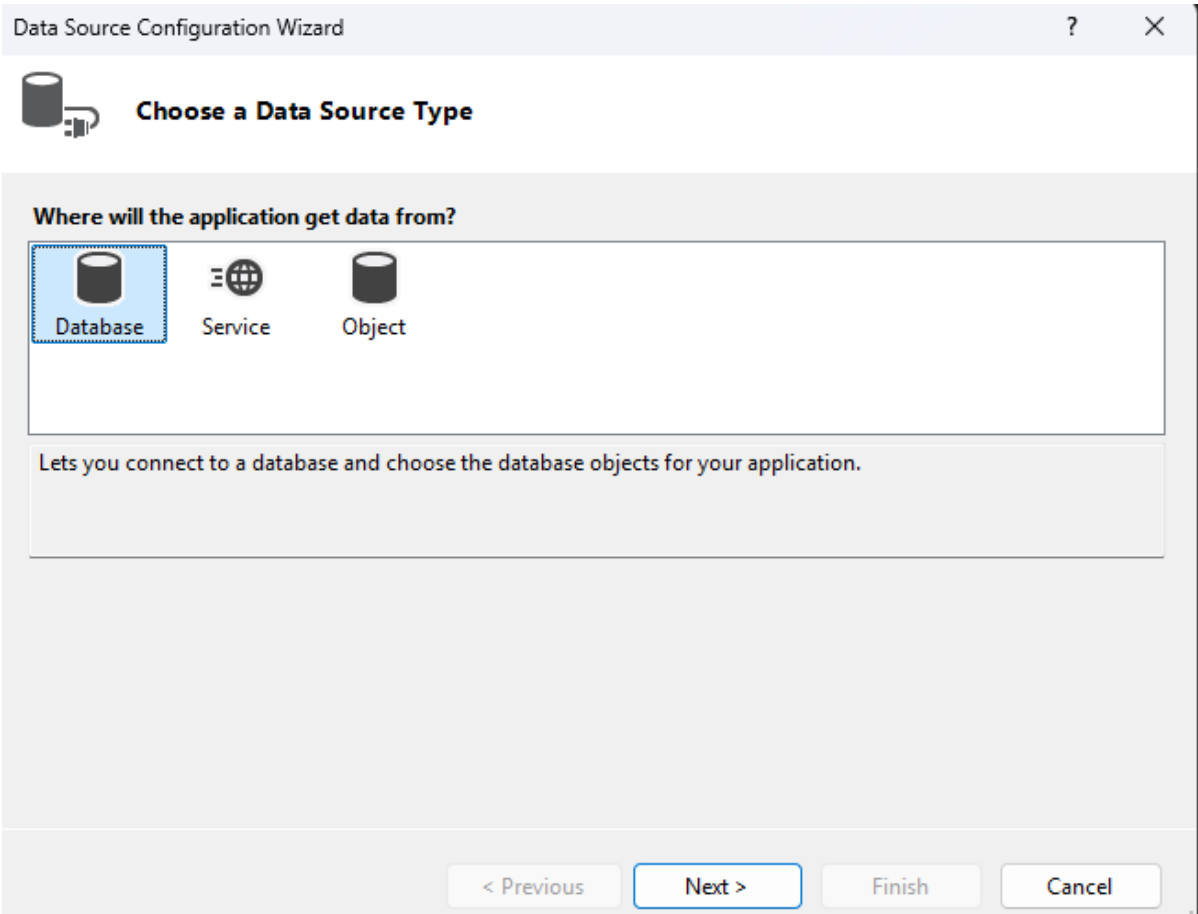
	Id	Номер	Работник	День	Время	Статус
*						



	Id	Номер	Работник	День	Время	Статус
▶	41	101	1	02.02.2025	12:12	Выполнено
	42	101	1	02.02.2025	12:12	Выполнено
	43	101	1	02.02.2025	12:12	Выполнено
	44	101	1	02.02.2025	12:12	Выполнено
	45	101	1	02.02.2025	12:12	Выполнено
	46	101	1	02.02.2025	12:12	Выполнено
	47	101	1	02.02.2025	12:12	Выполнено
	48	105	1	02.02.2025	12:12	Выполнено
	49	106	1	02.02.2025	12:12	Выполнено
	50	105	3	02.02.2025	12:12	Выполнено

ListBox





**Choose Your Data Connection**

Which data connection should your application use to connect to the database?

HotelConnectionString (Settings) ▼

New Connection...

This connection string appears to contain sensitive data (for example, a password), which is required to connect to the database. However, storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

- ☐ No, exclude sensitive data from the connection string. I will set this information in my application code.
- ☐ Yes, include sensitive data in the connection string.

☒ Show the connection string that you will save in the application

Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;Persist Security Info=True;User ID=Student;Password=qwe123QWE;TrustServerCertificate=True

< Previous

Next >

Finish

Cancel

**Choose Your Database Objects**

Which database objects do you want in your dataset?

Tables (partially selected)

- ☐ Analytic
- ☐ CleaningCalendar
- ☒ Employee
 - ☒ Id
 - ☒ Fname
- ☐ Room
- ☐ RoomReservation
- ☐ Users
- ☐ Views
- ☐ Stored Procedures
- ☐ Functions

DataSet name:

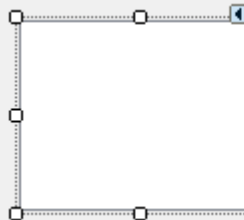
HotelDataSet1

< Previous

Next >

Finish

Cancel

**ListBox Tasks**☒ Use Data Bound Items**Data Binding Mode**

Data Source employeeBindingSour

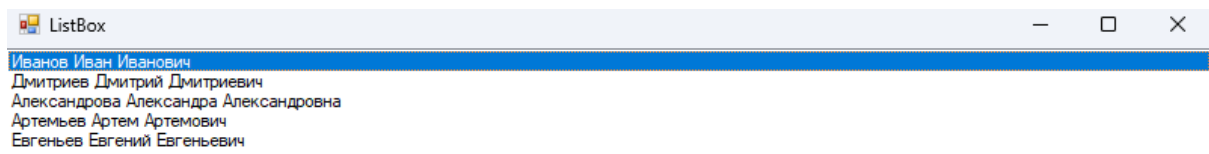
Display Member Fname

Value Member Id

Selected Value (none)

Add Query...

Preview Data...



POST/GET requests into database

GET request

```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=hotel;User Id=Student;Password=qwe123QWE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "SELECT Role FROM Users WHERE login= @login and password= @password;";
        query.Parameters.AddWithValue("@login", login);
        query.Parameters.AddWithValue("@password", password);

        using (var reader = query.ExecuteReader())
        {
            while(reader.Read())
            {
                role = reader.GetByte(0);
            }
        }
        connection.Close();
    }
}
```

`connection.Open();` – Открытие connection с БД

`SqlConnection` – библиотека, которую мы установили через NuGet

`SqlConnection(".....")` – строка подключения. [Откуда ее взять?](#)

`query.CommandText` – Query

`query.Parameters.AddWithValue("@login", login);` – Заглушки, куда вставляются переменные (общепринятые правила базовой безопасности)

```
using (var reader = query.ExecuteReader())
{
    while(reader.Read())
    {
        role = reader.GetByte(0);
    }
}
```

Чтение данных

`connection.Close();` – Закрытие connection с БД

POST request

```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QwE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "UPDATE Room SET Status = @status WHERE Id=@Id";
        query.Parameters.AddWithValue("@status", "Грязный");
        query.Parameters.AddWithValue("@Id", listBox1.SelectedValue);
        query.ExecuteNonQuery();
        connection.Close();
    }
}
```

Все тоже самое, кроме строки

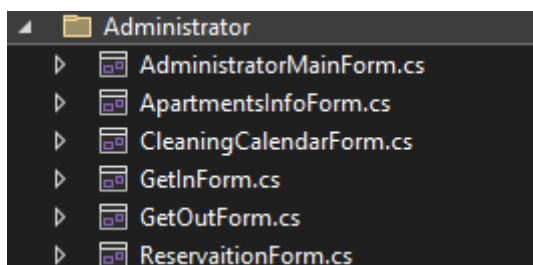
`query.ExecuteNonQuery();` – отправка запроса в БД

ПРОЕКТ

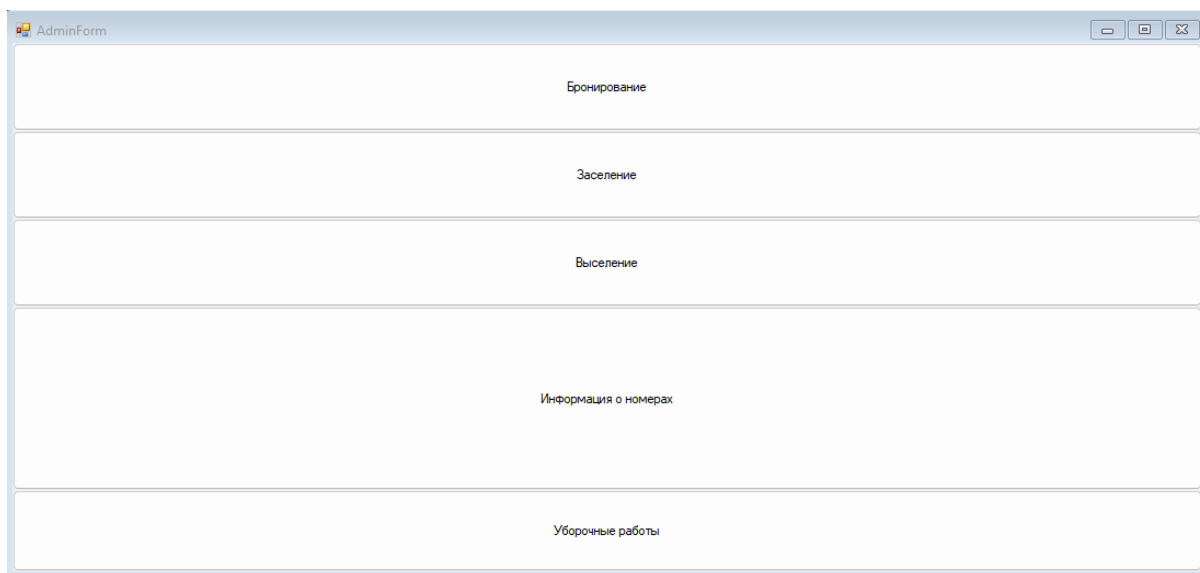
Начало

1. Создаем папку для администратора
2. Создаем Main menu для администратора
3. Создаем Формы для каждой функции

p.s. Просто окна, без функционала



4. Создаем Buttons для Main menu со ссылками на окна



```

private void buttonReservaition_Click(object sender, EventArgs e)
{
    try
    {
        ReservaitionForm _ReservaitionForm = new ReservaitionForm();
        _ReservaitionForm.Show();
    }
    catch {}
}

```

Создаем окно информации о номерах

Номер	Тип	Этаж	Статус
-------	-----	------	--------

[Ссылка](#)

Создаем окно резервирования номеров

ReservationForm

Дата заселения: 2 февраля 2025 г.

Номер:

ФИО клиента:

Количество гостей:

Зарезервировать

Дата выселения: 2 февраля 2025 г.

Номер резервирования:

Отменить резервирование

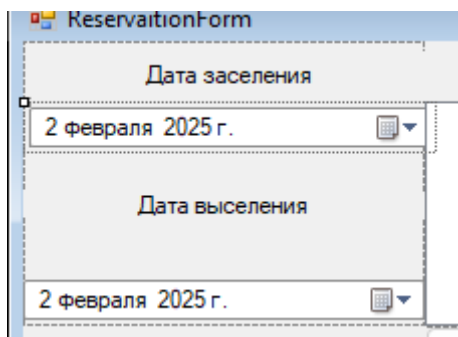
Id	Номер	Дата заселения	Дата выселения	ФИО клиента	Количество гостей
----	-------	----------------	----------------	-------------	-------------------

Создаем DataGridView, где выбираем таблицу room

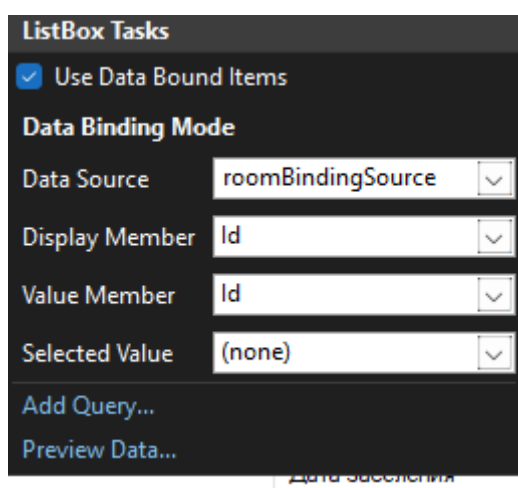
Layout	Top, Left
Anchor	Fill
AutoSizeColumnsMode	AllCells
AutoSizeRowsMode	Fill
Dock	0; 242
Location	3; 3; 3; 3
Margin	0; 0
MaximumSize	0; 0
MinimumSize	41
RowHeadersWidth	Both
ScrollBars	1503; 208
Size	(Collection)
Misc	
Columns	

Создаем функционал создания резервирования

Добавляем DateTimePicker на заселение и выселение



Добавляем [ListBox](#), в который добавляем Id-шники номеров.



- а. Добавляем TextBox количества гостей
- б. Добавляем кнопку резервирования

Кнопка резервирования

```
private void button1_Click(object sender, EventArgs e)
{
    //Проверяем на то, если дата заселения выше, чем дата выселения
    if (dateTimePickerStart.Value < dateTimePickerEnd.Value)
    {
        //Проверяем ФИО на пустоту
        if (textBoxFnameClient.Text.Trim().Length > 0)
        {
            //Проверяем количество гостей на Int32
            if (int32.TryParse(textBoxGuestCount.Text.Trim(), out int guests))
            {
                using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
                {
                    connection.Open();
                    using (var query = connection.CreateCommand())
                    {
                        //Добавляем в БД запись о резервировании
                        query.CommandText = "INSERT INTO RoomReservation (RoomId, Fdate, Ldate, ClientFname, Guests) VALUES (@roomId, @Fdate, @Ldate, @cFname, @gCount)";
                        query.Parameters.AddWithValue("@roomId", listBoxRoom.SelectedValue);
                        query.Parameters.AddWithValue("@Fdate", dateTimePickerStart.Value);
                        query.Parameters.AddWithValue("@Ldate", dateTimePickerEnd.Value);
                        query.Parameters.AddWithValue("@cFname", textBoxFnameClient.Text.Trim());
                        query.Parameters.AddWithValue("@gCount", textBoxGuestCount.Text.Trim());
                        query.ExecuteNonQuery();
                        connection.Close();
                    }
                }
            }
        }
    }
    //Создается самин фреймворк
}
```

Для того, чтобы при добавлении записи резервирования у нас сразу добавлялись данные в таблице приложения копируем созданные фреймворком строки кода, ответственные за обновление данных в таблицах и вставляем их в конец функции

```
public partial class ResvairaitonForm : Form
{
    1 reference
    public ResvairaitonForm()
    {
        InitializeComponent();
    }
    1 reference
    private void ResvairaitonForm_Load(object sender, EventArgs e)
    {
        // TODO: This line of code loads data into the 'hotelDataSet29.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter5.Fill(this.hotelDataSet29.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet28.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter4.Fill(this.hotelDataSet28.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet26.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter3.Fill(this.hotelDataSet26.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet25.RoomReservation' table. You can move, or remove it, as needed.
        // TODO: This line of code loads data into the 'hotelDataSet12.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter1.Fill(this.hotelDataSet12.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet11.Room' table. You can move, or remove it, as needed.
        this.roomTableAdapter.Fill(this.hotelDataSet11.Room);
        // TODO: This line of code loads data into the 'hotelDataSet10.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter.Fill(this.hotelDataSet10.RoomReservation);
    }
}
```

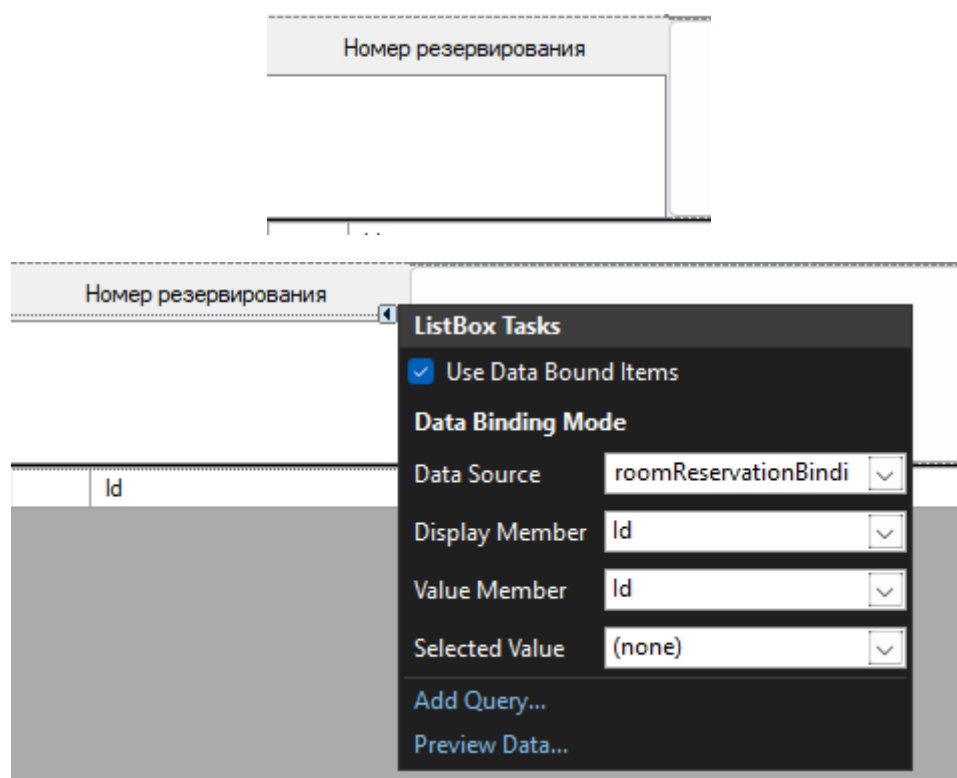
```
using (var query = connection.CreateCommand())
{
    //Добавляем в БД запись о резервировании
    query.CommandText = "INSERT INTO RoomReservation (RoomId, Fdate, Ldate, ClientFname, Guests) VALUES (@roomId, @Fdate, @Ldate, @cFname, @gCount)";
    query.Parameters.AddWithValue("@roomId", listBoxRoom.SelectedValue);
    query.Parameters.AddWithValue("@Fdate", dateTimePickerStart.Value);
    query.Parameters.AddWithValue("@Ldate", dateTimePickerEnd.Value);
    query.Parameters.AddWithValue("@cFname", textBoxFnameClient.Text.Trim());
    query.Parameters.AddWithValue("@gCount", textBoxGuestCount.Text.Trim());
    query.ExecuteNonQuery();
    connection.Close();
}
//Создается самин фреймворк
// TODO: This line of code loads data into the 'hotelDataSet29.RoomReservation' table. You can move, or remove it, as needed.
this.roomReservationTableAdapter5.Fill(this.hotelDataSet29.RoomReservation);
// TODO: This line of code loads data into the 'hotelDataSet28.RoomReservation' table. You can move, or remove it, as needed.
this.roomReservationTableAdapter4.Fill(this.hotelDataSet28.RoomReservation);
// TODO: This line of code loads data into the 'hotelDataSet26.RoomReservation' table. You can move, or remove it, as needed.
this.roomReservationTableAdapter3.Fill(this.hotelDataSet26.RoomReservation);
// TODO: This line of code loads data into the 'hotelDataSet25.RoomReservation' table. You can move, or remove it, as needed.
// TODO: This line of code loads data into the 'hotelDataSet12.RoomReservation' table. You can move, or remove it, as needed.
this.roomReservationTableAdapter1.Fill(this.hotelDataSet12.RoomReservation);
// TODO: This line of code loads data into the 'hotelDataSet11.Room' table. You can move, or remove it, as needed.
this.roomTableAdapter.Fill(this.hotelDataSet11.Room);
// TODO: This line of code loads data into the 'hotelDataSet10.RoomReservation' table. You can move, or remove it, as needed.
this.roomReservationTableAdapter.Fill(this.hotelDataSet10.RoomReservation);
}
```

Далее добавляем функционал else который высвечивает ответ о том, что пользователь ввел bad values

```
        this.roomReservationTableAdapter.Fill(this.hotelDataSet12.RoomReservation);  
        // TODO: This line of code loads data into the 'hotelDataSet11.Room' table. You can move, or remove it, as needed.  
        this.roomTableAdapter.Fill(this.hotelDataSet11.Room);  
        // TODO: This line of code loads data into the 'hotelDataSet10.RoomReservation' table. You can move, or remove it, as needed.  
        this.roomReservationTableAdapter.Fill(this.hotelDataSet10.RoomReservation);  
    }  
    else  
    {  
        MessageBox.Show("Неверное количество гостей");  
    }  
    else  
    {  
        MessageBox.Show("Неверное ФИО клиента");  
    }  
    else  
    {  
        MessageBox.Show("Неверное время");  
    }  
}
```

Создаем функционал удаления резервирования

Создаем [ListBox](#), который выводит Id резервирования



Кнопка Удаления резервирования

```
private void button2_Click(object sender, EventArgs e)
{
    try
    {
        using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4QNZHK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
        {
            connection.Open();
            using (var query = connection.CreateCommand())
            {
                query.CommandText = "DELETE RoomReservation WHERE Id = @id";
                query.Parameters.AddWithValue("@id", listBoxDeleteItem.SelectedValue);
                query.ExecuteNonQuery();
                connection.Close();
            }
        }

        // TODO: This line of code loads data into the 'hotelDataSet29.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter5.Fill(this.hotelDataSet29.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet28.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter4.Fill(this.hotelDataSet28.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet26.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter3.Fill(this.hotelDataSet26.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet25.RoomReservation' table. You can move, or remove it, as needed.
        // TODO: This line of code loads data into the 'hotelDataSet12.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter1.Fill(this.hotelDataSet12.RoomReservation);
        // TODO: This line of code loads data into the 'hotelDataSet11.Room' table. You can move, or remove it, as needed.
        this.roomTableAdapter.Fill(this.hotelDataSet11.Room);
        // TODO: This line of code loads data into the 'hotelDataSet10.RoomReservation' table. You can move, or remove it, as needed.
        this.roomReservationTableAdapter.Fill(this.hotelDataSet10.RoomReservation);
    }
    catch { }
}
```

Итог

ReservationForm

Дата заселения	Номер	ФИО клиента	Количество гостей
5 февраля 2025 г.	101	qwd	123
	102		
	103		
	104		
	105		
	106		
	107		
	108		

Дата выселения

6 февраля 2025 г.

Номер резервирования

25

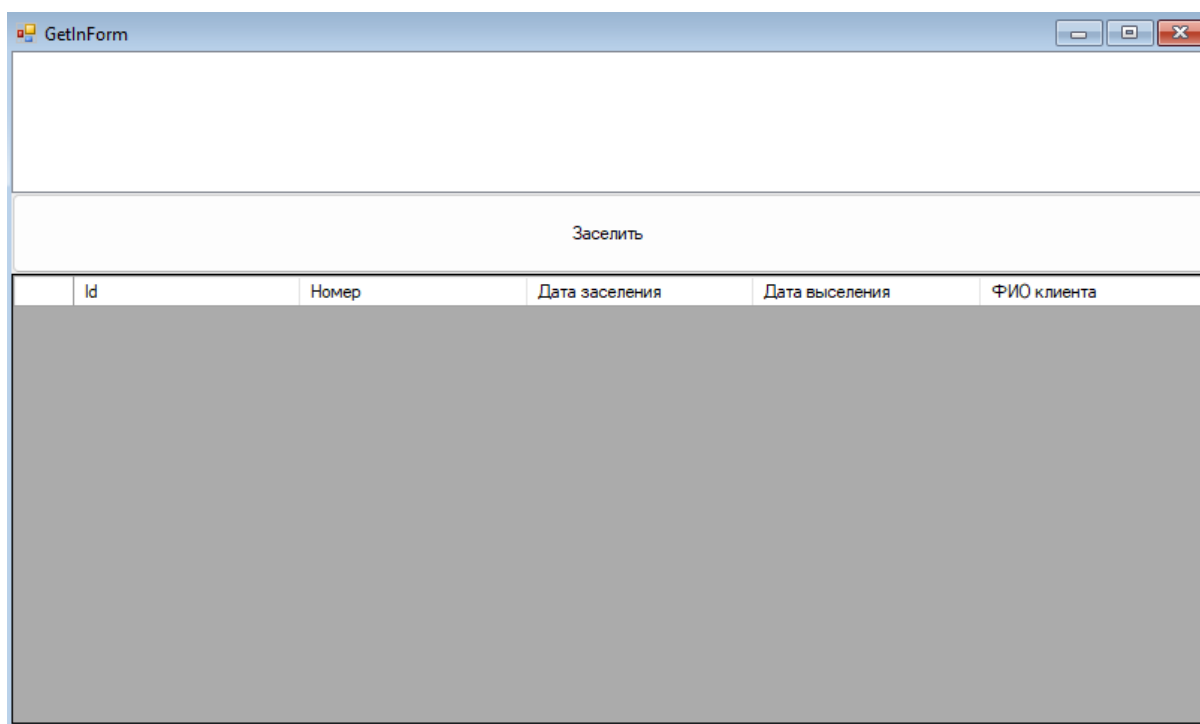
26

Зарезервировать

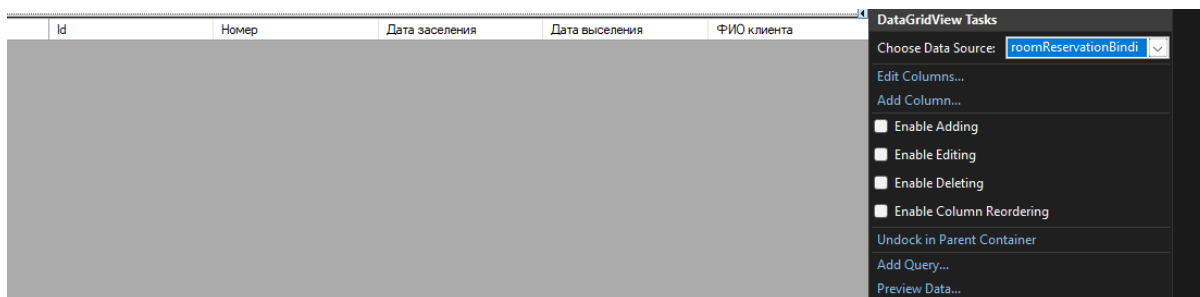
Отменить резервирование

Id	Номер	Дата заселения	Дата выселения	ФИО клиента	Количество гостей
125	306	11.02.2025	28.02.2025	Михайлов Игорь Вадимович	1
126	105	15.02.2025	16.02.2025	qwd	123

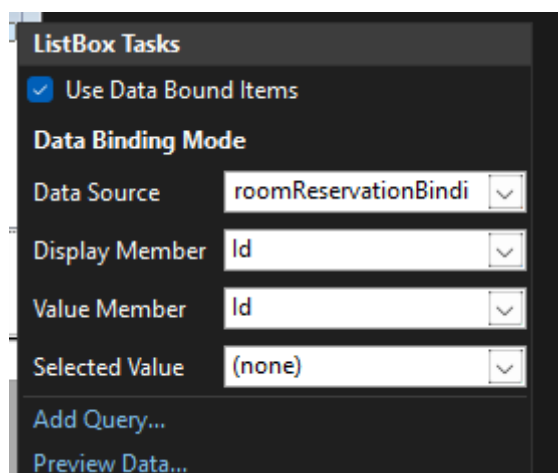
Окно заселения



Создаем DataGridView, где указываем RoomReservation



Создаем ListBox, где указываем Roomreservation Id.



Функционал заселения

Функционал заселения нужно разделить на 2 части:

- Добавлении записи о том, что люди заселились
- Добавление статистики

Сначала напишем функционал записи о том, что люди заселились

```
query.CommandText = "UPDATE Room SET Status = @status WHERE Id = @roomId";
query.Parameters.AddWithValue("@status", "Занят");
query.Parameters.AddWithValue("@roomId", roomId);
query.ExecuteNonQuery();
query.Parameters.Clear();
```

Теперь перейдем к статистике

```
Create Table Analytic(
    Id INT PRIMARY KEY IDENTITY,
    RoomId INT,
    Guests INT,
    Money INT,
    days INT,
    Day DATE,
    FOREIGN KEY (RoomId) REFERENCES Room (Id),
)
```

Это таблица статистики

```
int roomId = 0;
int guests = 0;
int money = 0;
DateTime Fdate = DateTime.Now;
DateTime Ldate = DateTime.Now;
query.CommandText = "SELECT RoomId, Guests, Fdate, Ldate FROM RoomReservation WHERE Id=@Id";
query.Parameters.AddWithValue("@Id", listBox1.SelectedValue);
using (var reader = query.ExecuteReader())
{
    while (reader.Read())
    {
        roomId = reader.GetInt32(0);
        guests = reader.GetInt32(1);
        Fdate = reader.GetDateTime(2);
        Ldate = reader.GetDateTime(3);
    }
}
query.Parameters.Clear();
```

Забираем Id номера, количество гостей, даты заселения и выселения

```

query.CommandText = "SELECT Money FROM Room WHERE Id=@roomId";
query.Parameters.AddWithValue("@roomId", roomId);
using (var reader = query.ExecuteReader())
{
    while (reader.Read())
    {
        money = reader.GetInt32(0);
    }
}
query.Parameters.Clear();

```

Забираем информацию о том, сколько денег за одну ночь

```

int days = Ldate.Subtract(Fdate).Days;
query.CommandText = "INSERT INTO Analytic (RoomId, Guests, Money, days, day) VALUES (@roomId, @guests, @money, @days, @day)";
query.Parameters.AddWithValue("@roomId", roomId);
query.Parameters.AddWithValue("@guests", guests);
query.Parameters.AddWithValue("@money", money*days);
query.Parameters.AddWithValue("@day", DateTime.Now);
query.Parameters.AddWithValue("@days", days);
query.ExecuteNonQuery();

```

Добавляем данные в статистику

** int days = Ldate.Subtract(Fdate).Days; - строка, спомощью которой мы получаем количество дней заселения

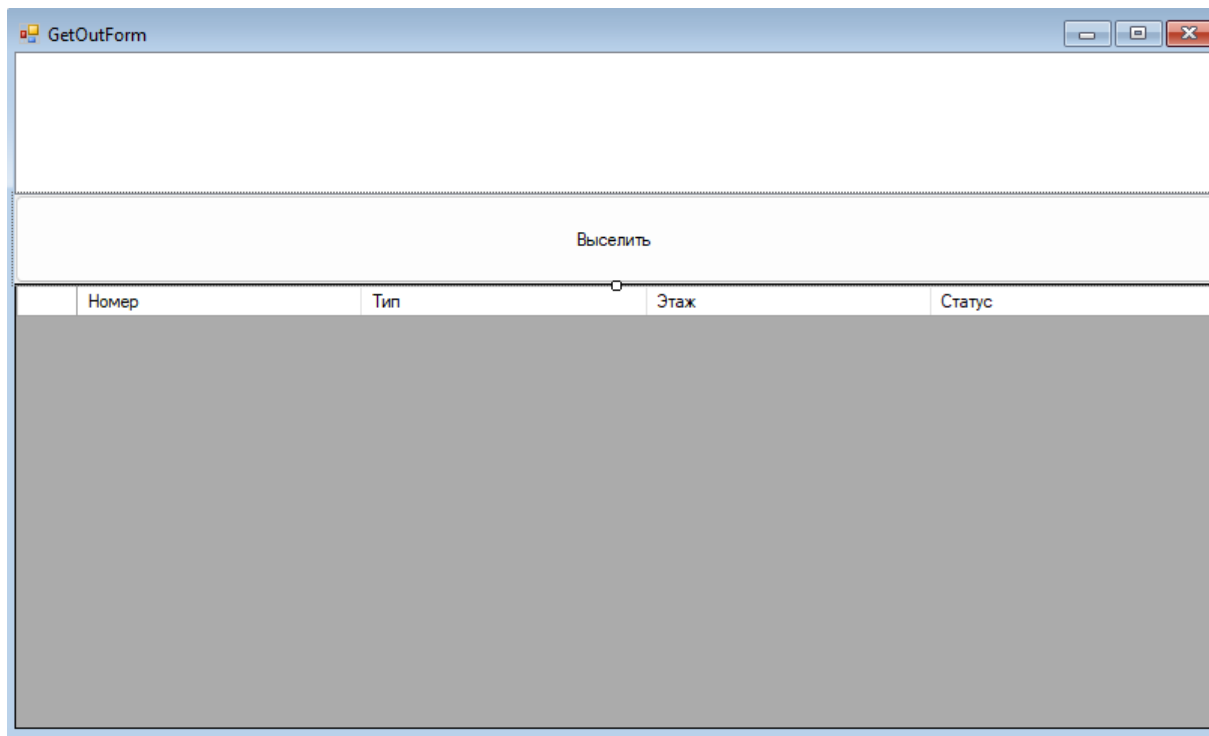
```

query.CommandText = "DELETE RoomReservation WHERE Id = @id";
query.Parameters.AddWithValue("@id", listBox1.SelectedValue);
query.ExecuteNonQuery();
query.Parameters.Clear();
connection.Close();

```

Удаляем запись о резервировании

Создаем окно выселения



Создаем DataGridView – берем данные из Room

ListBox – Берем данные из Room-id

Функционал выселения

```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "UPDATE Room SET Status = @status WHERE Id=@Id";
        query.Parameters.AddWithValue("@status", "Грязный");
        query.Parameters.AddWithValue("@Id", listBox1.SelectedValue);
        query.ExecuteNonQuery();
        connection.Close();
    }

    // TODO: This line of code loads data into the 'hotelDataSet23.Room' table. You can move, or remove it, as needed.
    this.roomTableAdapter2.Fill(this.hotelDataSet23.Room);
    // TODO: This line of code loads data into the 'hotelDataSet16.Room' table. You can move, or remove it, as needed.
    this.roomTableAdapter1.Fill(this.hotelDataSet16.Room);
    // TODO: This line of code loads data into the 'hotelDataSet15.Room' table. You can move, or remove it, as needed.
    this.roomTableAdapter.Fill(this.hotelDataSet15.Room);
}
```

Создаем окно календаря уборки

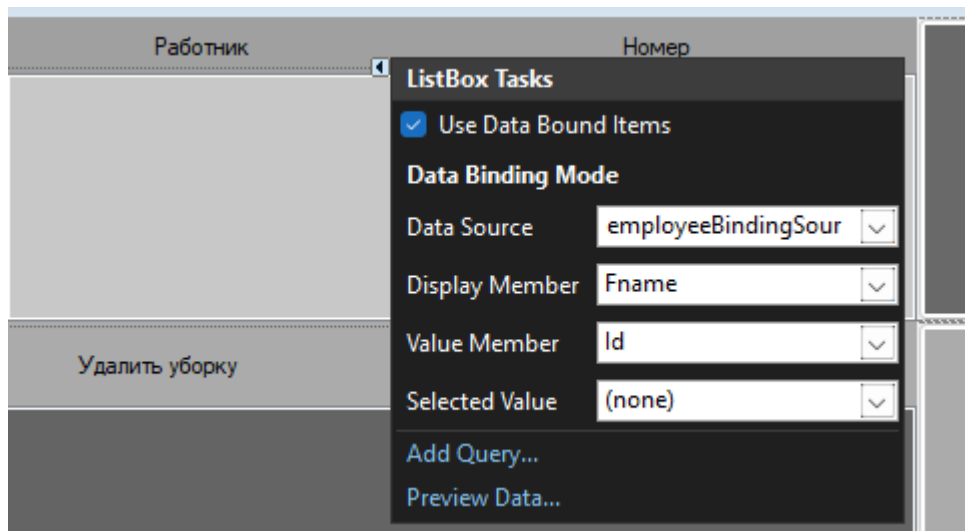
The screenshot shows a Windows application window titled "CleaningCalendarForm". The interface includes a date selection area at the top left with a calendar icon and the text "2 февраля 2025 г.". Below this are two dropdown menus labeled "Час" and "Минута". To the right of these are two input fields for "Работник" and "Номер". Further right is a large button labeled "Добавить уборку". Below the date and worker fields is a button labeled "Удалить уборку". At the bottom left is a button labeled "Изменить статус". To its right is a list box containing three items: "Ожидается", "В процессе", and "Выполнено". At the bottom right is a button labeled "Изменить статус уборки". The bottom of the window features a DataGrid with the following columns: "Id", "Номер", "Работник", "День", "Время", and "Статус".

DataGridView – берем из CleaningCalendar

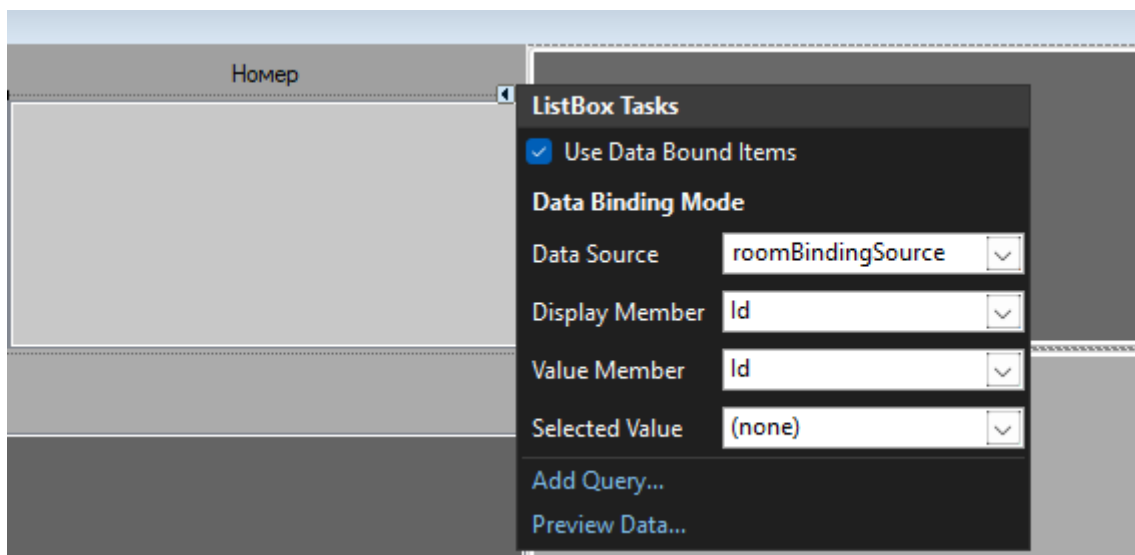
Создание уборки

This is a partial screenshot of the "CleaningCalendarForm" window, showing only the top portion of the interface. It includes the date picker set to "2 февраля 2025 г.", the "Час" and "Минута" dropdowns, the "Работник" and "Номер" input fields, and the "Добавить уборку" button.

1. Создаем DateTimePicker – нужен будет для выставления дня уборки
2. TextBox 2шт для выбора часов и минут.
3. ListBox Employees



4. ListBox RoomId



5. Создаем кнопку

Создание функционала создания

Проверка часов и минут на int

```
if (Int32.TryParse(textBoxHour.Text, out int hour) && hour > 0 && hour < 24)
{
    if (Int32.TryParse(textBoxMinute.Text, out int minute) && minute > 0 && minute < 60)
    {
```

Создаем запись об уборке

```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "INSERT INTO CleaningCalendar (RoomId, EmployeeId, Day, Time, Status) VALUES (@roomId, @employeeId, @day, @time, @status)";
        query.Parameters.AddWithValue("@day", dateTimePickerDate.Value);
        query.Parameters.AddWithValue("@time", $"{hour}:{minute}");
        query.Parameters.AddWithValue("@employeeId", listBoxEmployee.SelectedValue);
        query.Parameters.AddWithValue("@roomId", listBoxRoom.SelectedValue);
        query.Parameters.AddWithValue("@status", "Ожидается");
        query.ExecuteNonQuery();
        connection.Close();
    }
}
```

Важно: в заглушку Time добавляем две переменные: час и минута

Создание Уборки



Создаем ListBox – id CleaningCalendar

Создаем кнопку

Создаем функционал удаления

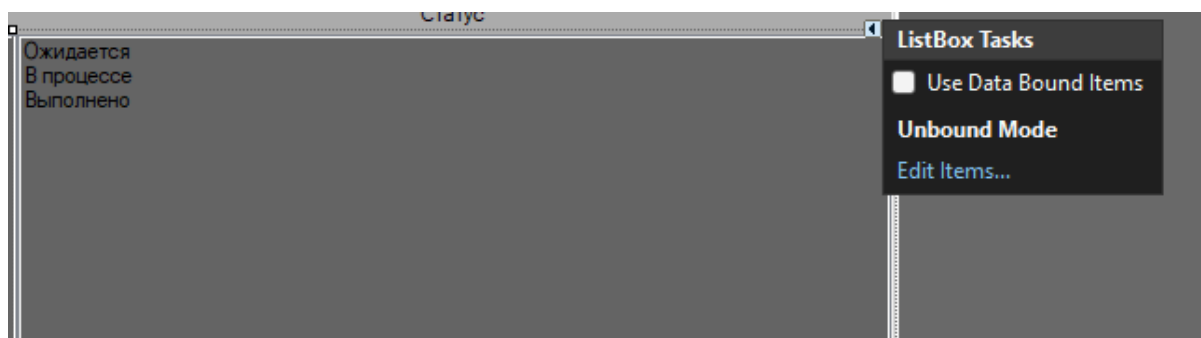
```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "DELETE CleaningCalendar WHERE Id = @id";
        query.Parameters.AddWithValue("@id", listBoxDeleteItem.SelectedValue);
        query.ExecuteNonQuery();
        connection.Close();
    }
}
```

Создаем изменение статуса уборки

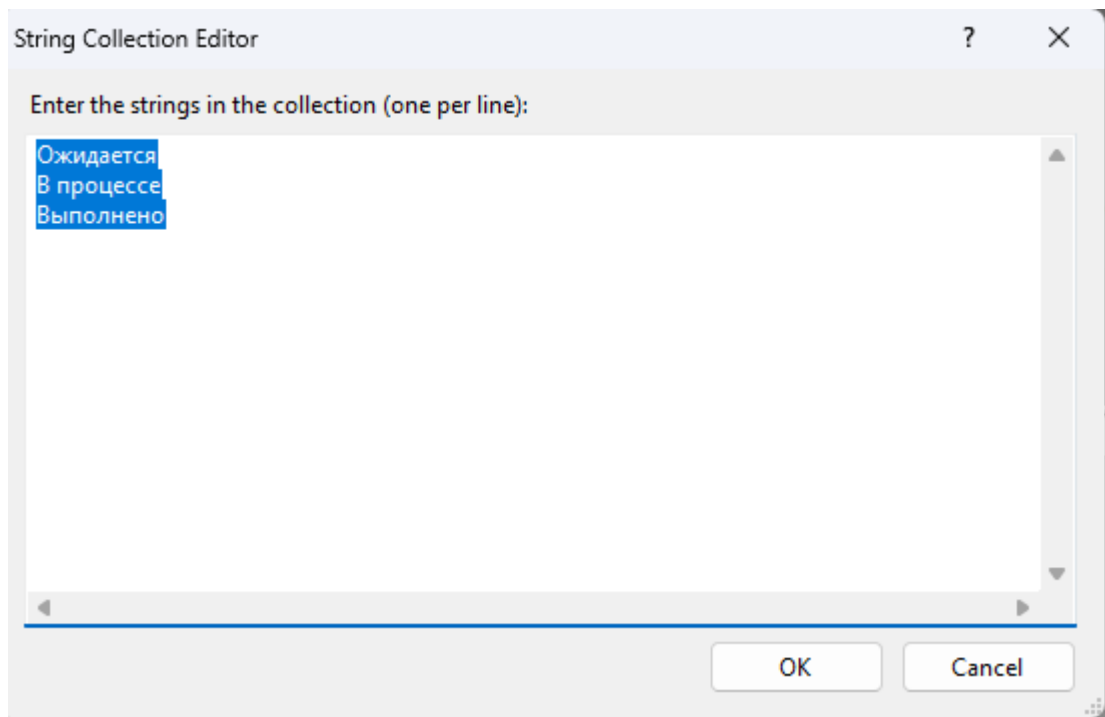


Создаем ListBox – id CleaningCalendar

Создаем listBox со статическими переменными



Edit items...



Создаем кнопку изменения

Создание функционала изменения статуса уборки

```
using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User Id=Student;Password=qwe123QWE"))
{
    connection.Open();
    using (var query = connection.CreateCommand())
    {
        query.CommandText = "UPDATE CleaningCalendar SET Status = @status WHERE Id = @id";
        query.Parameters.AddWithValue("@id", listBoxChangeItem.SelectedValue);
        query.Parameters.AddWithValue("@status", listBoxStatus.SelectedItem);
        query.ExecuteNonQuery();
        if (listBoxStatus.SelectedItem == "Выполнено")
        {
            int roomId = 0;
            query.CommandText = "SELECT RoomId FROM CleaningCalendar WHERE Id=@Idroom";
            query.Parameters.AddWithValue("@Idroom", listBoxChangeItem.SelectedValue);
            using (var reader = query.ExecuteReader())
            {
                while (reader.Read())
                {
                    roomId = reader.GetInt32(0);
                }
            }
            query.CommandText = "UPDATE Room SET Status = @status2 WHERE Id=@IDup";
            query.Parameters.AddWithValue("@status2", "Чистый");
            query.Parameters.AddWithValue("@IDup", roomId);
            query.ExecuteNonQuery();
        }
    }
    connection.Close();
}
```

Update status уборки

Если статус уборки Выполнено – Статус номера – “Чистый”

Создание окна Владельца

2 февраля 2025 г.

Получить

ADR (Average Daily Rate)

2 февраля 2025 г.

Получить

RevPAR (средний доход на комнату)

2 февраля 2025 г.

Получить

	Id	номер	Количество гостей	Выручка	Количество дней	День заселения
*						

DataGridView – Берем из таблицы Analytic

Функционал получения процента загрузки номерного фонда

```
private double getProcent(in string date)
{
    using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K40N2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
    {
        connection.Open();
        using (var query = connection.CreateCommand())
        {
            int analitcCount = 0;
            int roomCount = 0;
            query.CommandText = "SELECT SUM(days) FROM Analytic WHERE Day = @day";
            query.Parameters.AddWithValue("@day", date);
            using (var reader = query.ExecuteReader())
            {
                try
                {
                    while (reader.Read())
                    {
                        analitcCount = reader.GetInt32(0);
                    }
                }
                catch
                {
                    MessageBox.Show("На данный день отсутствует аналитика");
                }
            }
            query.CommandText = "SELECT COUNT(*) FROM Room";
            using (var reader = query.ExecuteReader())
            {
                while (reader.Read())
                {
                    roomCount = reader.GetInt32(0);
                }
            }
            connection.Close();
            return (double)analitcCount / roomCount;
        }
    }
}
```

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show($"{getProcent(dateTimePicker2.Value.ToString())}");
}
```

Функционал получения ADR

```
private double GetADR(in string date)
{
    using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K40N2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
    {
        connection.Open();
        using (var query = connection.CreateCommand())
        {
            int Money = 0;
            int Days = 0;
            query.CommandText = "SELECT SUM(Money), SUM(days) FROM Analytic WHERE Day = @day";
            query.Parameters.AddWithValue("@day", date);
            using (var reader = query.ExecuteReader())
            {
                try
                {
                    while (reader.Read())
                    {
                        Money = reader.GetInt32(0);
                        Days = reader.GetInt32(1);
                    }
                }
                catch
                {
                    MessageBox.Show("На данный день отсутствует аналитика");
                }
            }
            connection.Close();
            return (double)Money * 40 / 100 / Days;
        }
    }
}
```



```

1 reference
private void buttonADR1_Click(object sender, EventArgs e)
{
    MessageBox.Show($"{GetADR(dateTimePicker1.Value.ToString())}");
}

```

Функционал получения RevPAR

```

private double getRevPAR(in string date)
{
    using (SqlConnection connection = new SqlConnection("Data Source=DESKTOP-K4ON2HK;Initial Catalog=Hotel;User id=Student;Password=qwe123QWE"))
    {
        connection.Open();
        using (var query = connection.CreateCommand())
        {
            int days = 0;
            query.CommandText = "SELECT SUM(days) FROM Analytic WHERE Day = @day";
            query.Parameters.AddWithValue("@day", date);
            using (var reader = query.ExecuteReader())
            {
                try
                {
                    while (reader.Read())
                    {
                        days = reader.GetInt32(0);
                    }
                }
                catch
                {
                    MessageBox.Show("На данный день отсутствует аналитика");
                }
            }
            return (double)GetADR(date) / days;
        }
    }
}

```

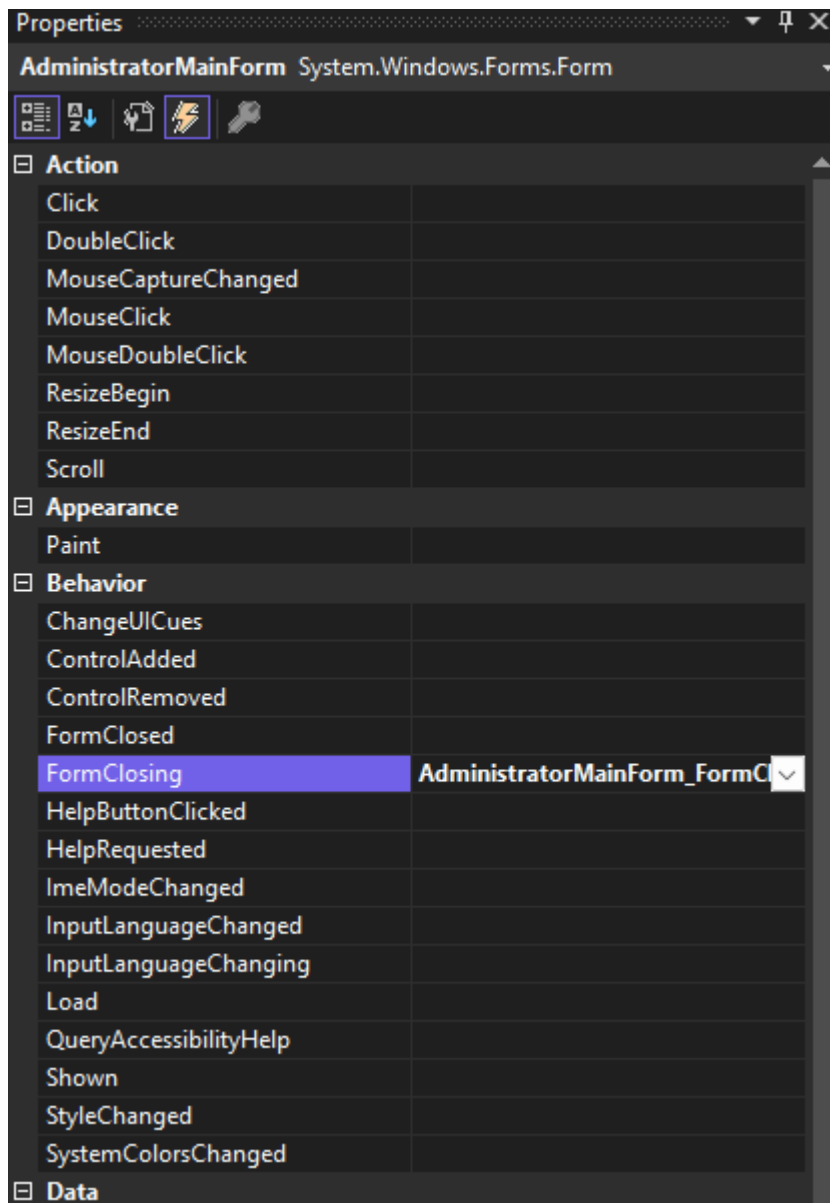
```

1 reference
private void buttonRevPAR_Click(object sender, EventArgs e)
{
    MessageBox.Show($"{getRevPAR(dateTimePicker3.Value.ToString())}");
}

```

Функционал закрытия Main Forms

1. Кликаем на форму
2. В Properties выбираем "Грозу"



- 3.
4. Focus=>Deactivate

```

private void AdministratorMainForm_FormClosing(object sender, FormClosingEventArgs e)
{
    Application.Exit();
}

```

- 5.

Приложения

GitHub

<https://github.com/Untarget/demoexam>

Database Schema

