Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образование учреждение высшего образования

«Северо-Восточной федеральной университет им. М. К. Аммосова»

Колледж инфраструктурных технологий

Кафедра эксплуатации и обслуживания информационных систем

**Отчет лабораторных работ**

Специальность: 09.02.07 Информационные системы и программирование

Квалификация выпускника: Программист

Выполнили: студенты группы ИСИП-19-3

Семенов Д. М.

Шамаев А. Г.

Проверила: Бускарова М. Ф.

Якутск – 2021 г.

**Сессия 1**

# **Восстановление базы данных из скрипта**

Для восстановления таблиц в созданную базы данных воспользуйтесь предоставленным скриптом (ms.sql или my.sql в зависимости от выбранной СУБД). В процессе разработки приложения Вы можете изменять базу данных на свое усмотрение.

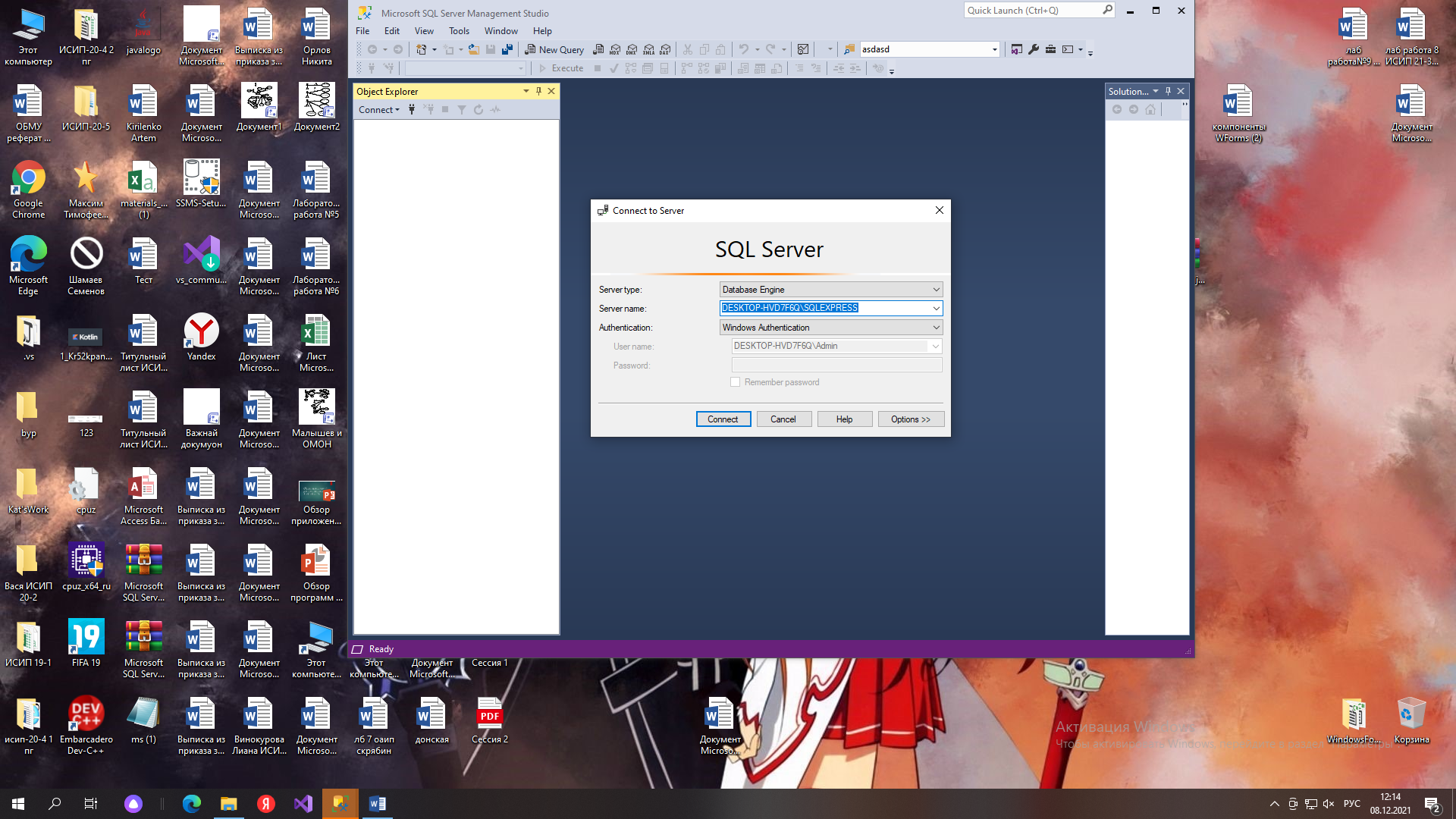
# **Импорт данных**

Заказчик системы предоставил файлы с данными (с пометкой import в ресурсах) для переноса в новую систему. Подготовьте данные файлов для импорта и загрузите в разработанную базу данных.

**Ход работы:**

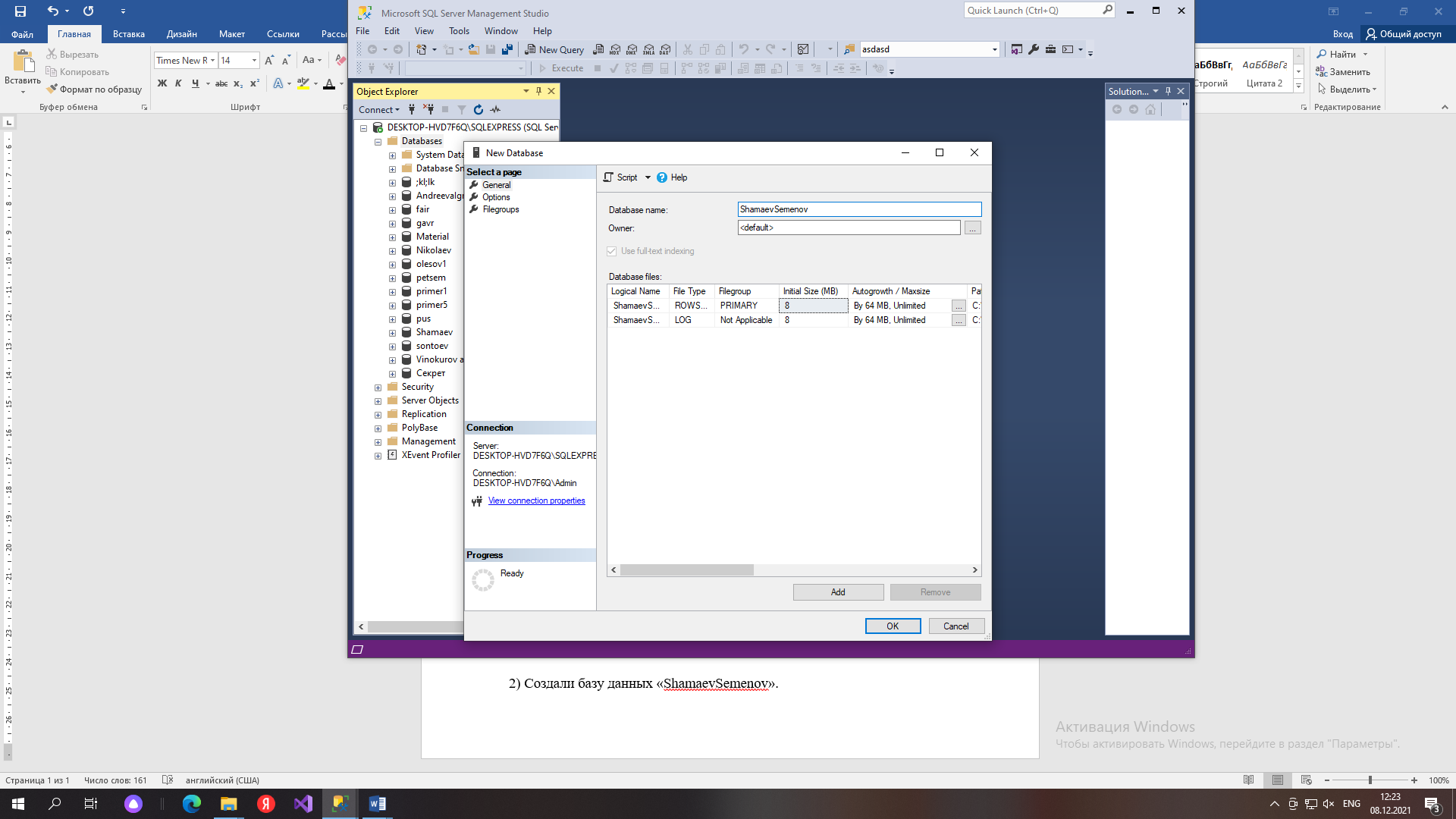
1) Запустили Microsoft SQL Server Management.

Рис. 1 Соединение с сервером



2) Создали базу данных «ShamaevSemenov».

Рис. 2 Создание б/д



3) Восстановили базу данных из скрипта.

Рис. 3.1 Восстановление таблиц

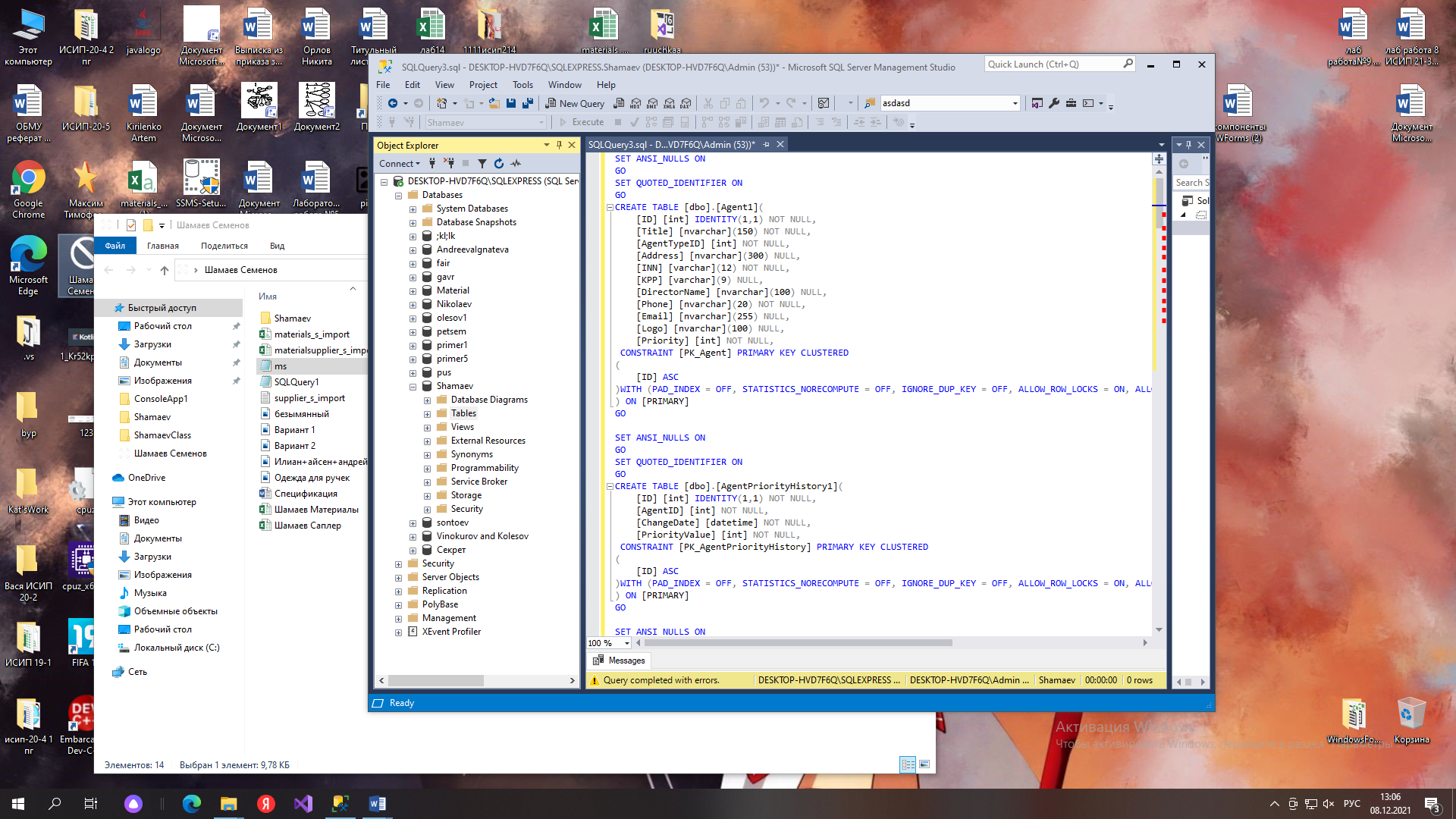
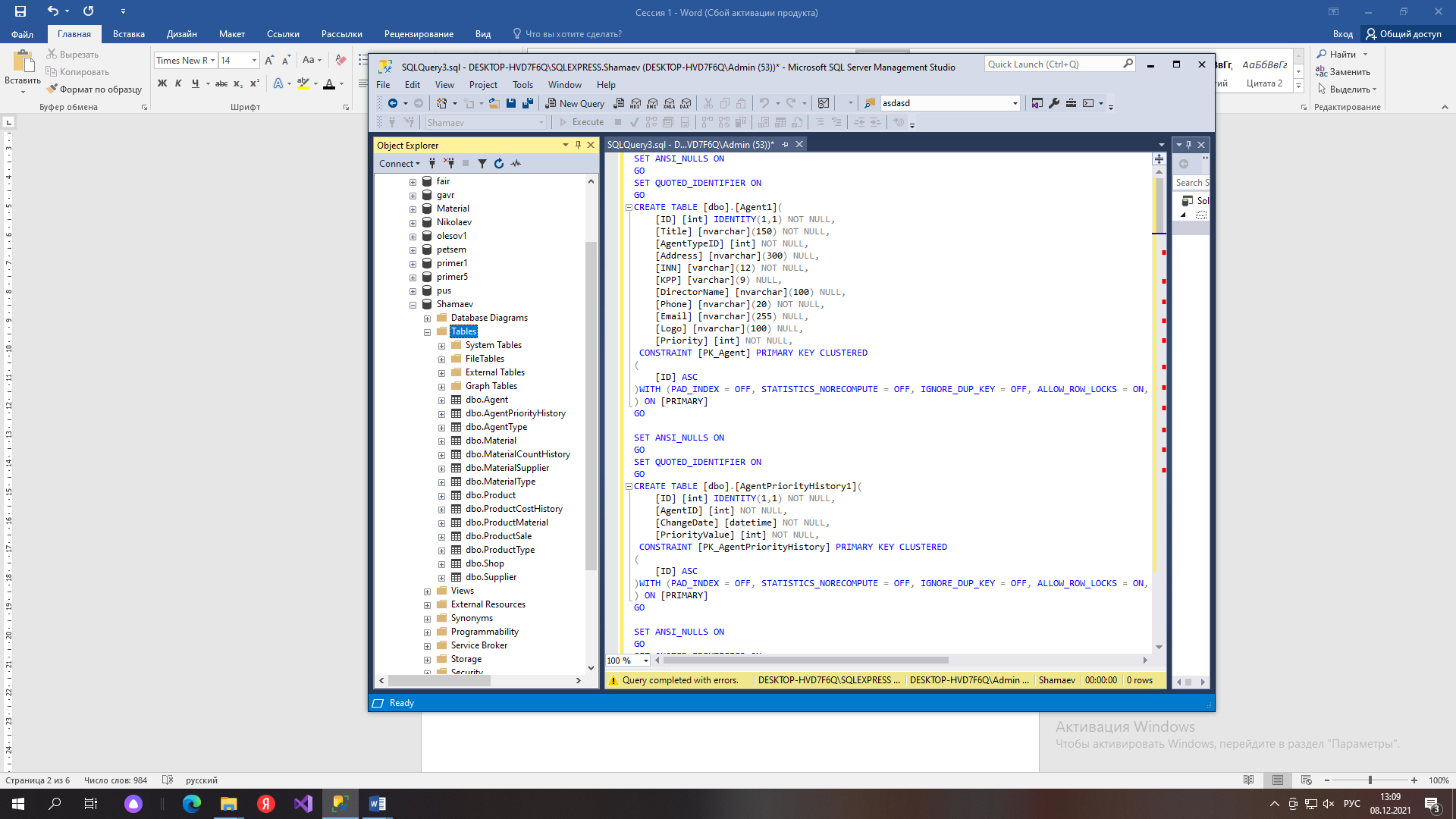


Рис. 3.2 Результат



4) Добавили с помощью импорта данные в таблицы.

Рис. 4.1 Импорт данных

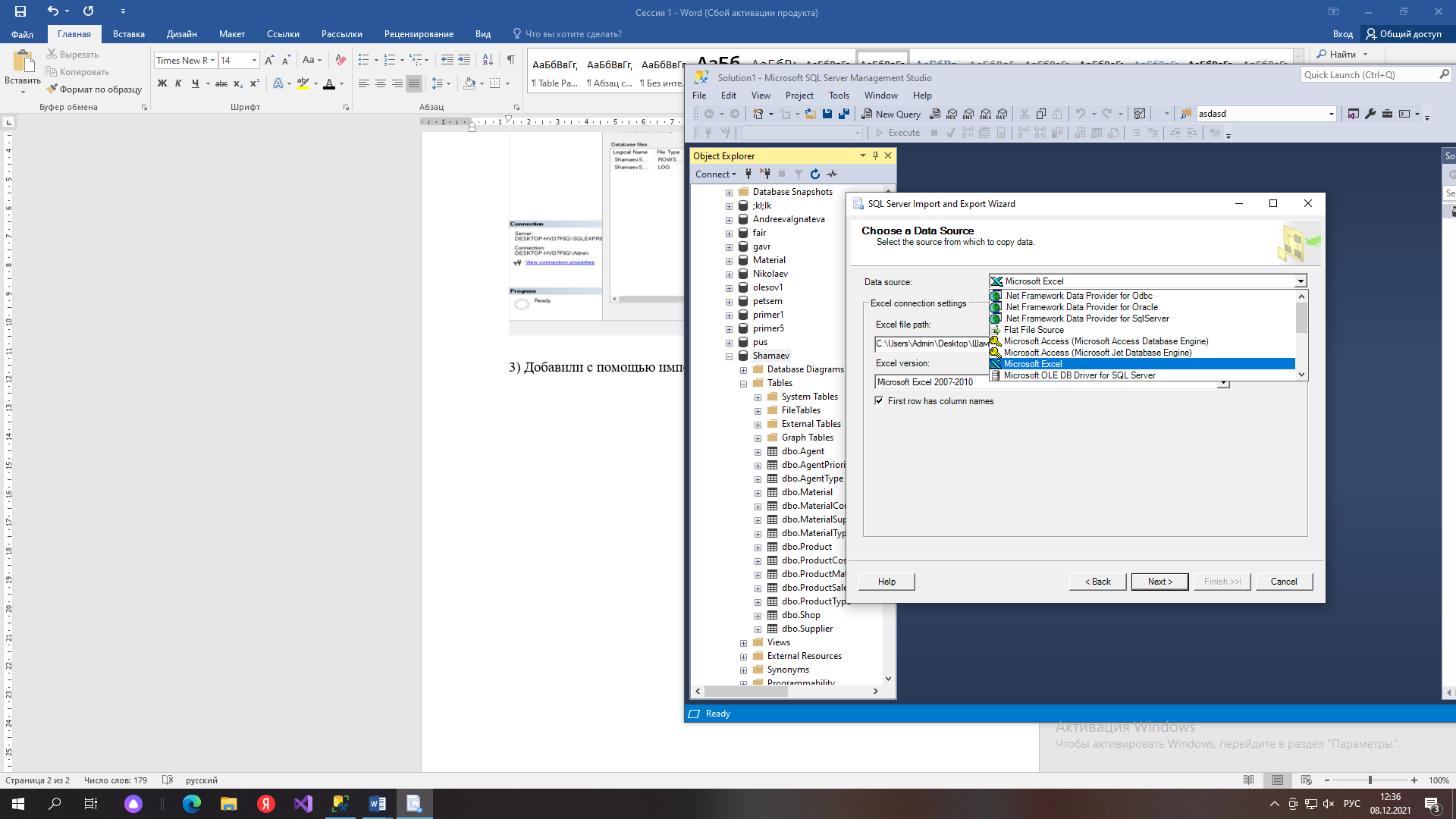
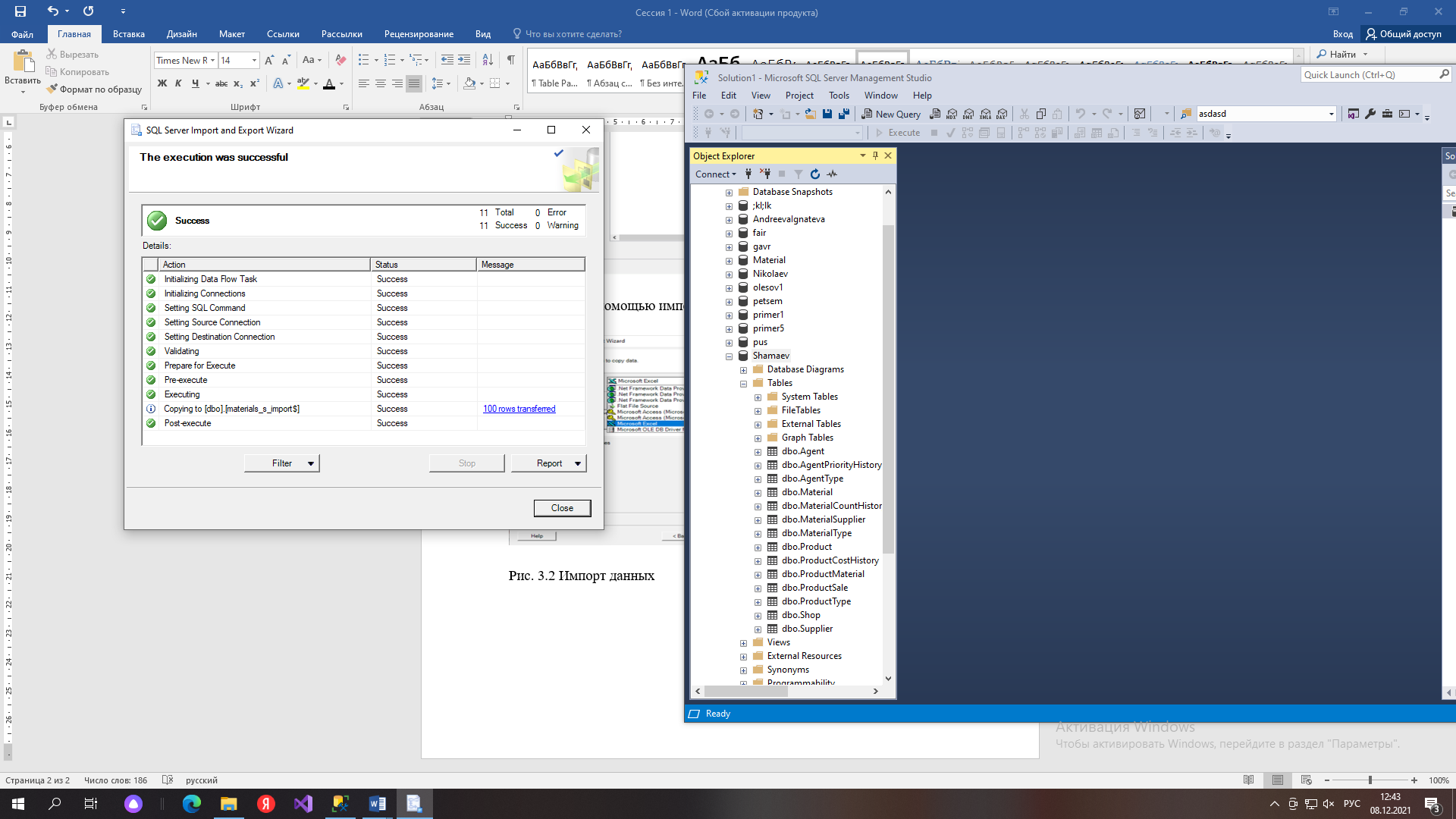
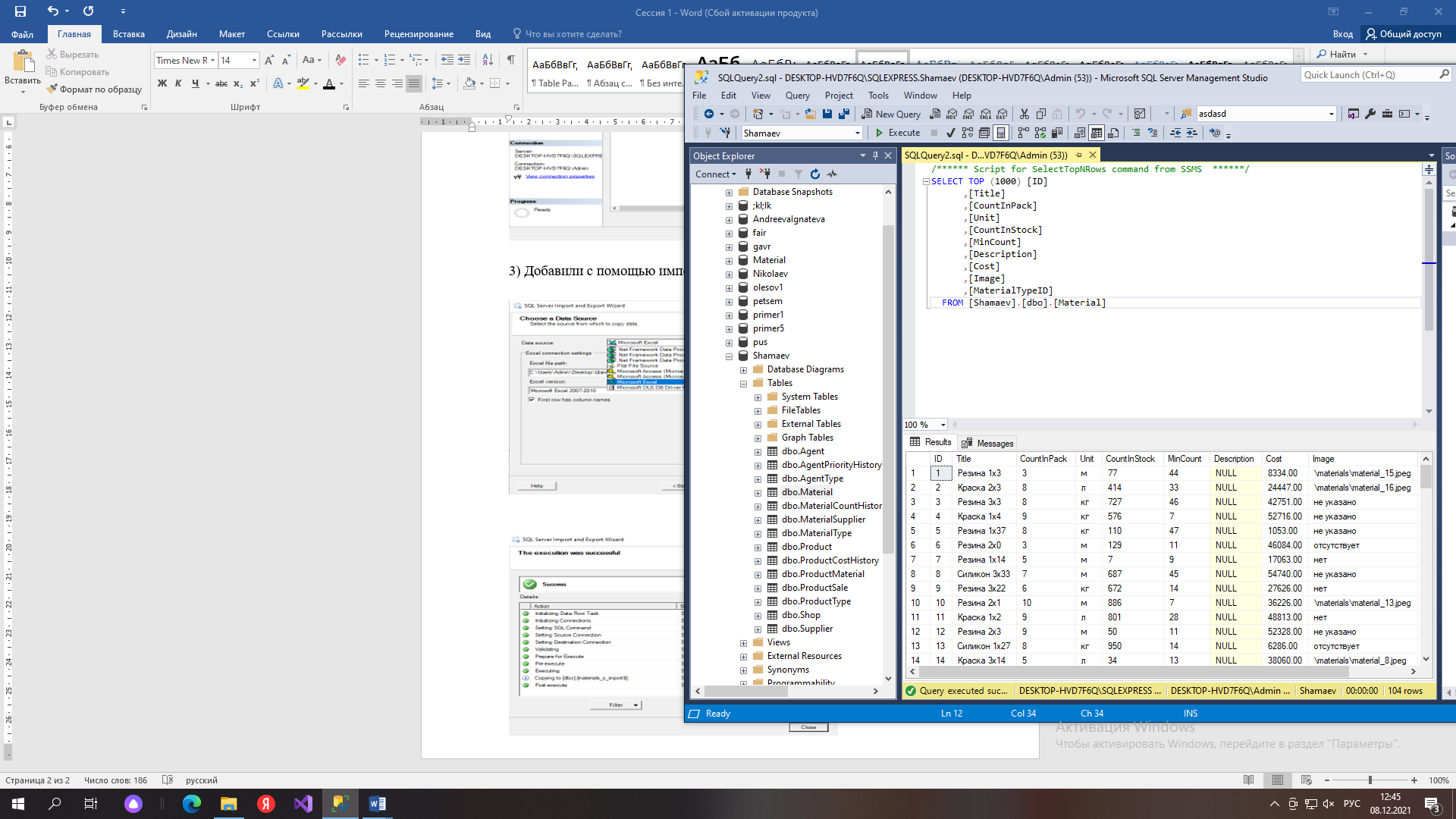


Рис. 4.2 Импорт данных



5) Импорт данных успешно завершен.

Рис. 5 Проверка данных



**Сессия 2**

**Разработка desktop-приложений**

* **Список материалов**

Очень важно постоянно получать актуальную информацию об остатках материалов на складах, поэтому необходимо реализовать вывод материалов, которые хранятся в базе данных, согласно предоставленному макету. При отсутствии изображения необходимо вывести картинку-заглушку из ресурсов (picture.png).

Вывод должен осуществляться постранично (по 15 записей на страницу). Для удобства навигации по страницам необходимо вывести список их номеров (как на макете) с возможностью перехода к выбранной странице, а также предусмотреть переходы к предыдущей и следующей страницам.

Пользователь должен иметь возможность отсортировать материалы (по возрастанию и убыванию) по следующим параметрам: наименование, остаток на складе и стоимость. Выбор сортировки должен быть реализован с помощью выпадающего списка.

Кроме этого, пользователь должен иметь возможность отфильтровать данные по типу материала. Все типы из базы данных должны быть выведены в выпадающий список для фильтрации. Первым элементом в выпадающем списке должен быть “Все типы”, при выборе которого настройки фильтра сбрасываются.

Пользователь должен иметь возможность искать материалы, используя поисковую строку. Поиск должен осуществляться по наименованию и описанию материала.

Поиск, сортировка и фильтрация должны происходить в реальном времени, без необходимости нажатия кнопки “найти”/”отфильтровать” и т.п. Фильтрация и поиск должны применяться совместно. Параметры сортировки, выбранные ранее пользователем, должны сохраняться и во время фильтрации с поиском.

В верхней части окна необходимо показывать количество выведенных данных и общее количество записей в базе. Например, 155 из 237. В случае если данные в таблицу выводятся после фильтрации или поиска, количество выведенных данных необходимо обновить исходя из размера выборки.

Необходимо подсвечивать строки с данными о конкретном материале в зависимости от минимального допустимого количества и наличия на складе. В случае если количество материала меньше, чем указанное минимальное количество, в качестве фона необходимо применить цвет #f19292. В случае если текущее количество составляет 300% от минимального количества, то необходимо использовать цвет с кодом #ffba01.

В списке материалов необходимо добавить возможность изменения минимального количества сразу для нескольких выбранных материалов. Для этой цели реализуйте возможность выделения сразу нескольких элементов в списке материалов, после чего должна появиться кнопка “Изменить минимальное количество на ...”. При нажатии на кнопку необходимо отобразить модальное окно с возможностью ввода числового значения, на которое и будет изменено значение минимального количества всех выбранных материалов. По умолчанию в поле должно быть введено максимальное значение этого параметра среди всех материалов. После нажатия кнопки “Изменить” минимальное количество выделенных материалов должна быть изменено в базе данных, а также обновлено в интерфейсе.

* **Добавление/редактирование материалов**

Необходимо реализовать возможность добавления и редактирования исходных материалов (сырья) для всей продукции компании.

Необходимо добавить возможность редактирования данных существующего материала, а также добавление нового материала в новом окне - форме для добавления/редактирования материала. Переходы на данное окно должны быть реализованы из главной формы списка: для редактирования - при нажатии на конкретный элемент, для добавления - при нажатии кнопки “Добавить материал”.

На форме должны быть предусмотрены следующие поля: наименование, тип материала (выпадающий список), количество на складе, единица измерения, количество в упаковке, минимальное количество, стоимость за единицу, изображение и подробное описание (с возможностью многострочного ввода). Стоимость материала может включать сотые части, а также не может быть отрицательной. Минимальное количество также не может принимать отрицательные значения.

Также необходимо реализовать вывод списка возможных поставщиков. В список можно добавлять новые позиции и удалять существующие. При добавлении поставщики должны выбираться из выпадающего списка с возможностью поиска по наименованию.

При открытии формы для редактирования все поля выбранного объекта должны быть подгружены в соответствующие поля из базы данных, а таблица заполнена актуальными значениями.

Для получения информации об объеме затрат на закупки необходимого количества материалов следует предусмотреть вывод информации о стоимости минимально необходимой партии. Если количество на складе меньше минимального количества материалов, необходимо получить разницу между этими значениями и рассчитать минимальный объем закупки, кратный количеству материалов в упаковке (так как материалы закупаются исключительно целыми упаковками). Сумму закупки следует рассчитать, ориентируясь на минимальный объем и цену за единицу материала.

Пользователь может добавить/заменить изображение у материала.

Для того чтобы администратор случайно не изменял несколько материалов, предусмотрите невозможность открытия более одного окна редактирования.

В окне редактирования материала должна присутствовать кнопка “Удалить”, которая удаляет материал из базы данных. При этом должны соблюдаться следующие условия. Если у материала есть информация о возможных поставщиках или история изменения количества, то эта информация должна быть удалена вместе с материалом. Но если он используется при производстве какой-либо продукции, то удаление материала из базы данных должно быть запрещено. После удаления материала система должна сразу вернуть пользователя обратно к списку материалов.

После редактирования/добавления/удаления материалов данные в окне списка материалов должны быть обновлены.

**Разработка тестовых сценариев (Test-cases)**

Для выполнения процедуры тестирования удаления материалов Вам нужно описать пять сценариев. Удаление может быть выполнимо, а может быть отклонено согласно требованиям предметной области. Необходимо, чтобы варианты тестирования демонстрировали различные исходы работы алгоритма. Для описания тестовых сценариев в ресурсах предоставлен шаблон testing-template.docx.

**Создание руководства пользователя**

Вам необходимо разработать руководство пользователя для вашего настольного приложения, которое описывает последовательность действий для выполнения всех функций вашей системы.

При подготовке документации старайтесь использовать живые примеры и скриншоты вашей системы для более наглядного пояснения шагов работы с различным функционалом.

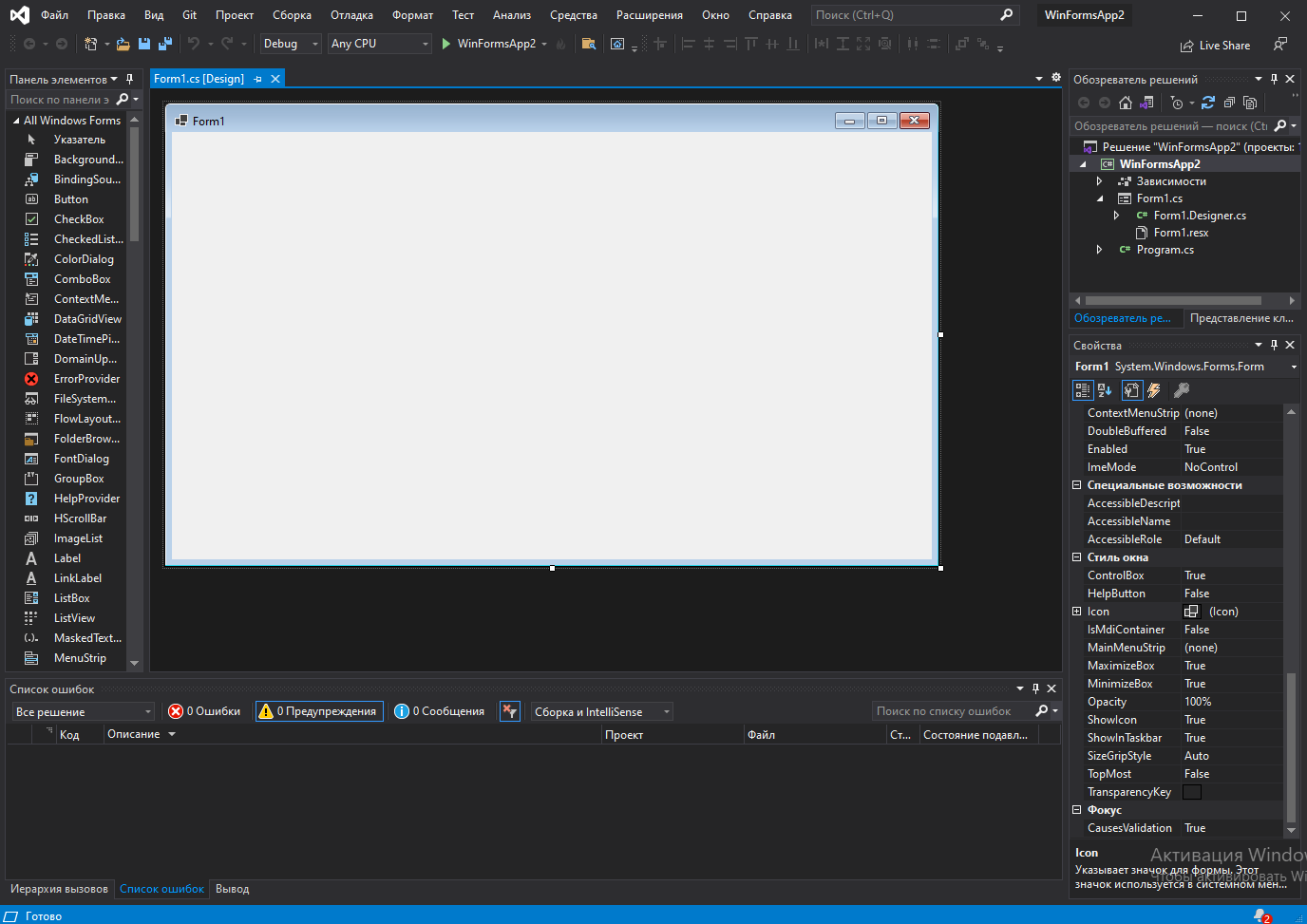
Обратите внимание на оформление документа: оформите титульный лист, используйте автоматическую нумерацию страниц, разделите руководство на подразделы и сформируйте оглавление, используйте ссылки на рисунки, нумерованные и маркированные списки для описания шагов и т.д.

Сохраните итоговый документ с руководством пользователя в формате Word, используя в качестве названия следующий шаблон: Руководство пользователя\_XX, где XX - номер вашего рабочего места.

**Ход работы:**

1) Запустили Visual Studio и создали новый проект “Создание проекта” -> “Приложение Windows Forms”.

Рис. 6 Создание формы



2) Добавили список материалов с помощью DataGridView и соединили с базой данных. Информацию взяли из таблицы “Material”.

Рис. 7.1 Добавление таблицы “Material”

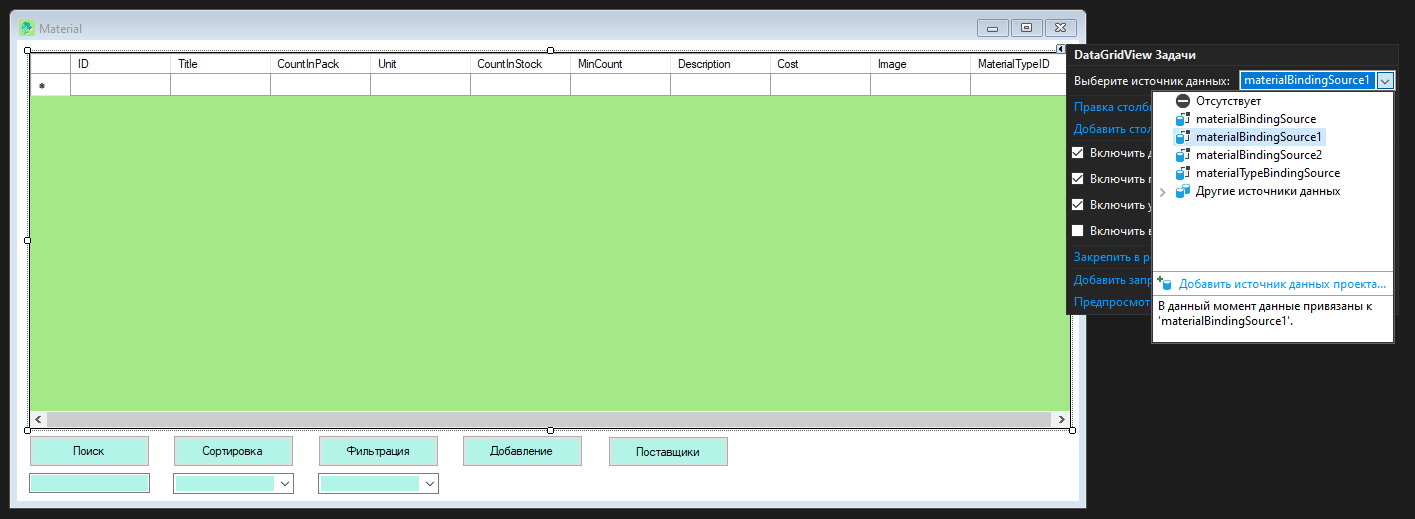
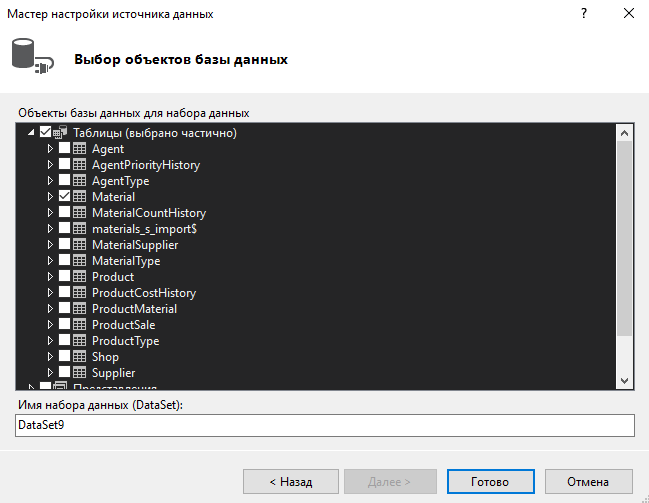


Рис. 7.2. Добавление таблицы “Material”



3) Добавили в форму “Material” кнопку “Поиск” с помощью button и textBox.

Код кнопки button “Поиск”:

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

*{*

*label1.Text = "";*

*{*

*int count1 = 0;*

*for (int i = 0; i < dataGridView1.RowCount; i++)*

*{*

*dataGridView1.Rows[i].Selected = false;*

*for (int j = 0; j < dataGridView1.ColumnCount; j++)*

*if (dataGridView1.Rows[i].Cells[j].Value != null)*

*if (dataGridView1.Rows[i].Cells[j].Value.ToString().Contains(textBox1.Text))*

*{*

*dataGridView1.Rows[i].Selected = true;*

*count1++;*

*break;*

*}*

4) Добавили в форму “Material” кнопку “Сортировка” с помощью button и comboBox.

Код кнопки button “Сортировка”:

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

*{*

*DataSet ds = new DataSet();*

*SqlConnection con = new SqlConnection(@"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev");*

*SqlDataAdapter da = new SqlDataAdapter();*

*da.SelectCommand = new SqlCommand("select \* from Material Order By Cost asc", con);*

*con.Open();*

*da.Fill(ds, "Material");*

*dataGridView1.DataSource = ds.Tables[0];*

*da.Dispose();*

*con.Dispose();*

*ds.Dispose();*

*}*

5) Добавили в форму “Material” кнопку “Фильтрация” с помощью button и comboBox.

Код кнопки button “Фильтрация”:

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

*{*

*for (int i = 0; i < dataGridView1.Rows.Count - 1; i++)*

*{*

*dataGridView1.CurrentCell = null;*

*dataGridView1.Rows[i].Visible = false;*

*for (int j = 0; j < dataGridView1.Columns.Count; j++)*

*{*

*if (dataGridView1.Rows[i].Cells[j].Value.ToString().Contains(comboBox2.Text))*

*{*

*dataGridView1.Rows[i].Visible = true;*

*break;*

*}*

*}*

*}*

*}*

6) Добавили в форму “Material” кнопку “Добавление” с помощью button.

Код:

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

*{*

*this.Hide();*

*Form2 f2 = new Form2();*

*f2.ShowDialog();*

*}*

7) Создали новую форму для кнопки “Добавление”. Добавили несколько “label”, “textBox” и “comboBox”. В “textBox” и “comboBox” поменяли цвет на “ 179; 244; 233”.

Рис. 8 Форма добавления “Material”



Для этой формы добавили кнопки “Добавить” и “Назад”.

Код кнопки “Добавить”:

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

*{*

*string Title = textBox1.Text.Trim();*

*int CountInPack = Convert.ToInt32 (textBox2.Text.Trim());*

*string Unit = textBox3.Text.Trim();*

*float CountInStock = Convert.ToSingle (textBox4.Text.Trim());*

*float MinCount = Convert.ToSingle (textBox5.Text.Trim());*

*string Description = textBox6.Text.Trim();*

*decimal Cost = Convert.ToDecimal (textBox7.Text.Trim());*

*string Image = textBox8.Text.Trim();*

*object MaterialTypeID = comboBox1.Text.Trim();*

*SqlConnection con = new SqlConnection(@"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev");*

*con.Open();*

*string query = "SELECT \* FROM MaterialType WHERE Title = '" + comboBox1.Text + "'";*

*SqlCommand cmd = new SqlCommand(query, con);*

*SqlDataReader reader = cmd.ExecuteReader();*

*if (reader.HasRows)*

*{*

*while (reader.Read())*

*{*

*object ID = reader.GetValue(0);*

*MaterialTypeID = ID;*

*}*

*reader.Close();*

*try*

*{*

*string insertquery = "INSERT INTO Material (Title, CountInPack, Unit, CountInStock, MinCount, Description, Cost, Image, MaterialTypeID) VALUES ('" + Title + "','" + CountInPack + "','" + Unit + "','" + CountInStock + "','" + MinCount + "','" + Description + "','" + Cost + "','" + Image + "','" + MaterialTypeID + "')";*

*SqlCommand cmd2 = new SqlCommand(insertquery, con);*

*cmd2.ExecuteNonQuery();*

*MessageBox.Show("Запись добавлена!");*

*}*

*catch (SqlException ex)*

*{*

*MessageBox.Show(ex.Message);*

*}*

*Form2.ActiveForm.Close();*

*}*

*con.Close();*

*}*

Код кнопки “Назад”:

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

*{*

*this.Hide();*

*Form1 f1 = new Form1();*

*f1.ShowDialog();*

*}*

8) Добавили в форму “Material” кнопку “Поставщики” с помощью button.

Код:

*private void button5\_Click(object sender, EventArgs e)*

*{*

*this.Hide();*

*Form3 f3 = new Form3();*

*f3.ShowDialog();*

*}*

9) Добавили в форму “Material” подсчет записей.

Код:

*private void label1\_Click(object sender, EventArgs e)*

*{*

*string constr = @"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev";*

*SqlConnection con = new SqlConnection(constr);*

*con.Open();*

*string query = "select \* from Material";*

*SqlCommand cmd = new SqlCommand(query, con);*

*SqlDataReader dr = cmd.ExecuteReader();*

*int count = 0;*

*while (dr.Read())*

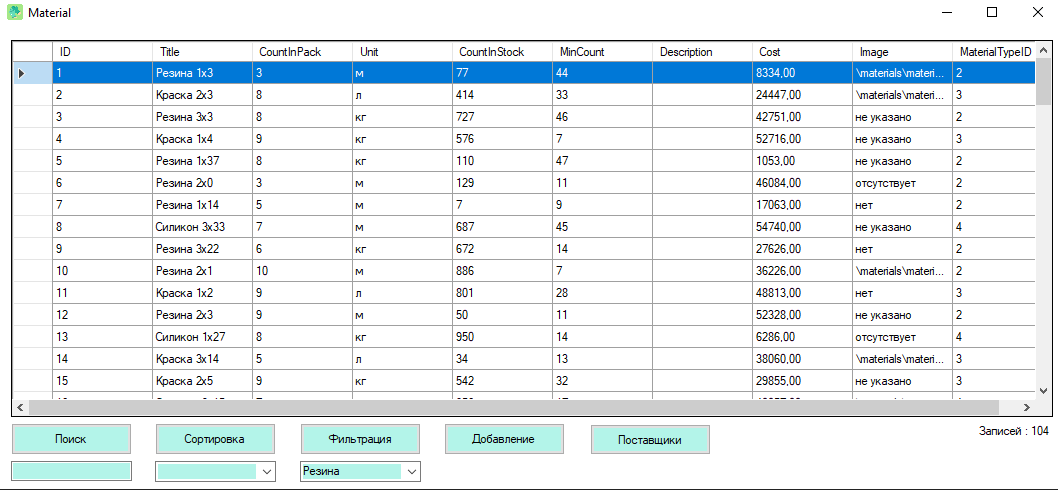
*{ count++; }*

*label1.Text = "Записей : " + count + " ";*

*}*

10) Цвет в кнопках сделали “179; 244; 233”, а в таблице “165; 232; 135”.

Рис. 9 Результат



11) Создали новую форму для кнопки “Поставщики”. Добавили список материалов с помощью DataGridView и соединили с базой данных. Информацию взяли из таблицы “Supplier”.

Рис. 10.1 Добавление таблицы “Supplier”

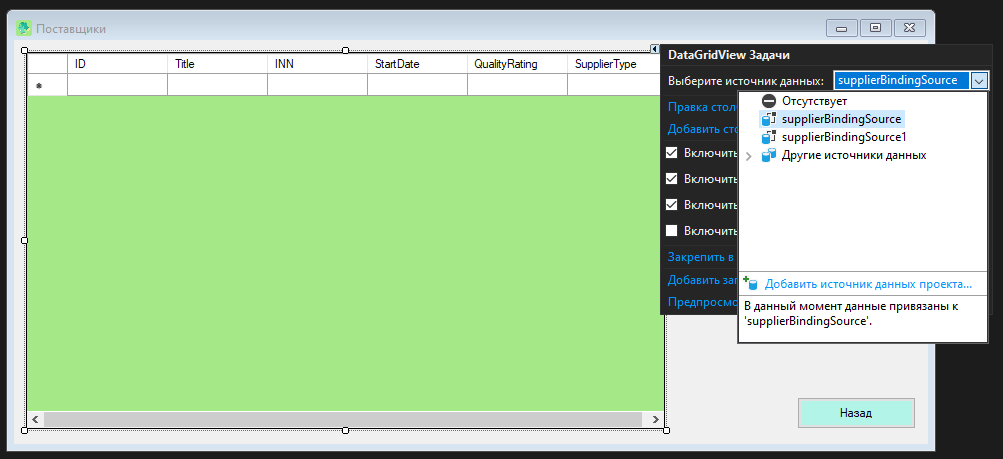
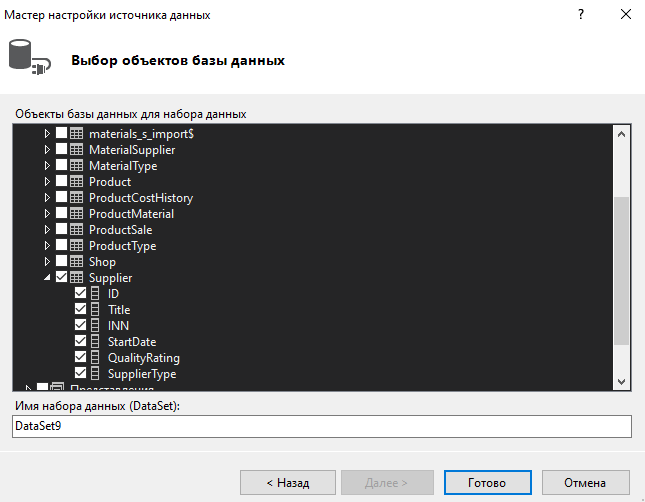


Рис. 10.2. Добавление таблицы “Supplier”



12) Добавили в форму “Supplier” кнопку “Поиск” с помощью button и textBox.

Код кнопки button “Поиск”:

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

*{*

*label1.Text = "";*

*{*

*int count1 = 0;*

*for (int i = 0; i < dataGridView1.RowCount; i++)*

*{*

*dataGridView1.Rows[i].Selected = false;*

*for (int j = 0; j < dataGridView1.ColumnCount; j++)*

*if (dataGridView1.Rows[i].Cells[j].Value != null)*

*if (dataGridView1.Rows[i].Cells[j].Value.ToString().Contains(textBox1.Text))*

*{*

*dataGridView1.Rows[i].Selected = true;*

*count1++;*

*break;*

*}*

*}*

*string constr = @"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev";*

*SqlConnection con = new SqlConnection(constr);*

*con.Open();*

*string query = "select \* from Supplier";*

*SqlCommand cmd = new SqlCommand(query, con);*

*SqlDataReader dr = cmd.ExecuteReader();*

*label1.Text = "Записей : " + count1;*

*}*

*}*

13) Добавили в форму “Supplier” кнопку “Сортировка” с помощью button и comboBox.

Код кнопки button “Сортировка”:

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

*{*

*DataSet ds = new DataSet();*

*SqlConnection con = new SqlConnection(@"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev");*

*SqlDataAdapter da = new SqlDataAdapter();*

*da.SelectCommand = new SqlCommand("select \* from Supplier Order By QualityRating asc", con);*

*con.Open();*

*da.Fill(ds, "Supplier");*

*dataGridView1.DataSource = ds.Tables[0];*

*da.Dispose();*

*con.Dispose();*

*ds.Dispose();*

*}*

14) Добавили в форму “Supplier” кнопку “Фильтрация” с помощью button и comboBox.

Код кнопки button “Фильтрация”:

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

*{*

*for (int i = 0; i < dataGridView1.Rows.Count - 1; i++)*

*{*

*dataGridView1.CurrentCell = null;*

*dataGridView1.Rows[i].Visible = false;*

*for (int j = 0; j < dataGridView1.Columns.Count; j++)*

*{*

*if (dataGridView1.Rows[i].Cells[j].Value.ToString().Contains(comboBox2.Text))*

*{*

*dataGridView1.Rows[i].Visible = true;*

*break;*

*}*

*}*

*}*

*}*

15) Добавили в форму “Supplier” кнопку “Добавление” с помощью button.

Код:

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

*{*

*this.Hide();*

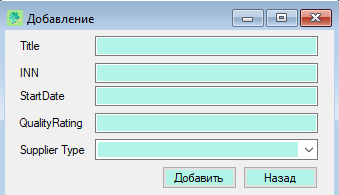
*Form4 f4 = new Form4();*

*f4.ShowDialog();*

*}*

16) Создали новую форму для кнопки “Добавление”. Добавили несколько “label”, “textBox” и “comboBox”. В “textBox” и “comboBox” поменяли цвет на “ 179; 244; 233”.

Рис. 11 Форма добавления “Supplier”



Для этой формы добавили кнопки “Добавить” и “Назад”.

Код кнопки “Добавить”:

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

*{*

*string Title = textBox1.Text.Trim();*

*int INN = Convert.ToInt32(textBox2.Text.Trim());*

*string StartDate = textBox3.Text.Trim();*

*int QualityRating = Convert.ToInt32(textBox4.Text.Trim());*

*object SupplierType = comboBox1.Text.Trim();*

*SqlConnection con = new SqlConnection(@"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev");*

*con.Open();*

*{*

*try*

*{*

*string insertquery = "INSERT INTO Supplier (Title, INN, StartDate, QualityRating, SupplierType) VALUES ('" + Title + "','" + INN + "','" + StartDate + "','" + QualityRating + "','" + SupplierType + "')";*

*SqlCommand cmd2 = new SqlCommand(insertquery, con);*

*cmd2.ExecuteNonQuery();*

*MessageBox.Show("Запись добавлена!");*

*}*

*catch (SqlException ex)*

*{*

*MessageBox.Show(ex.Message);*

*}*

*Form4.ActiveForm.Close();*

*}*

*con.Close();*

*}*

Код кнопки “Назад”:

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

*{*

*this.Hide();*

*Form3 f3 = new Form3();*

*f3.ShowDialog();*

*}*

17) Добавили в форму “Supplier” подсчет записей.

Код:

*label1.Text = "";*

*{*

*int count1 = 0;*

*for (int i = 0; i < dataGridView1.RowCount; i++)*

*{*

*dataGridView1.Rows[i].Selected = false;*

*for (int j = 0; j < dataGridView1.ColumnCount; j++)*

*if (dataGridView1.Rows[i].Cells[j].Value != null)*

*if (dataGridView1.Rows[i].Cells[j].Value.ToString().Contains(textBox1.Text))*

*{*

*dataGridView1.Rows[i].Selected = true;*

*count1++;*

*break;*

*}*

*}*

*string constr = @"Data Source =SQLNCLI11;Data Source=DESKTOP-HVD7F6Q\SQLEXPRESS;Integrated Security=SSPI;Initial Catalog=Shamaev";*

*SqlConnection con = new SqlConnection(constr);*

*con.Open();*

*string query = "select \* from Supplier";*

*SqlCommand cmd = new SqlCommand(query, con);*

*SqlDataReader dr = cmd.ExecuteReader();*

*label1.Text = "Записей : " + count1;*

*}*

18) Цвет в кнопках сделали “179; 244; 233”, а в таблице “165; 232; 135”.

Рис. 12 Результат

