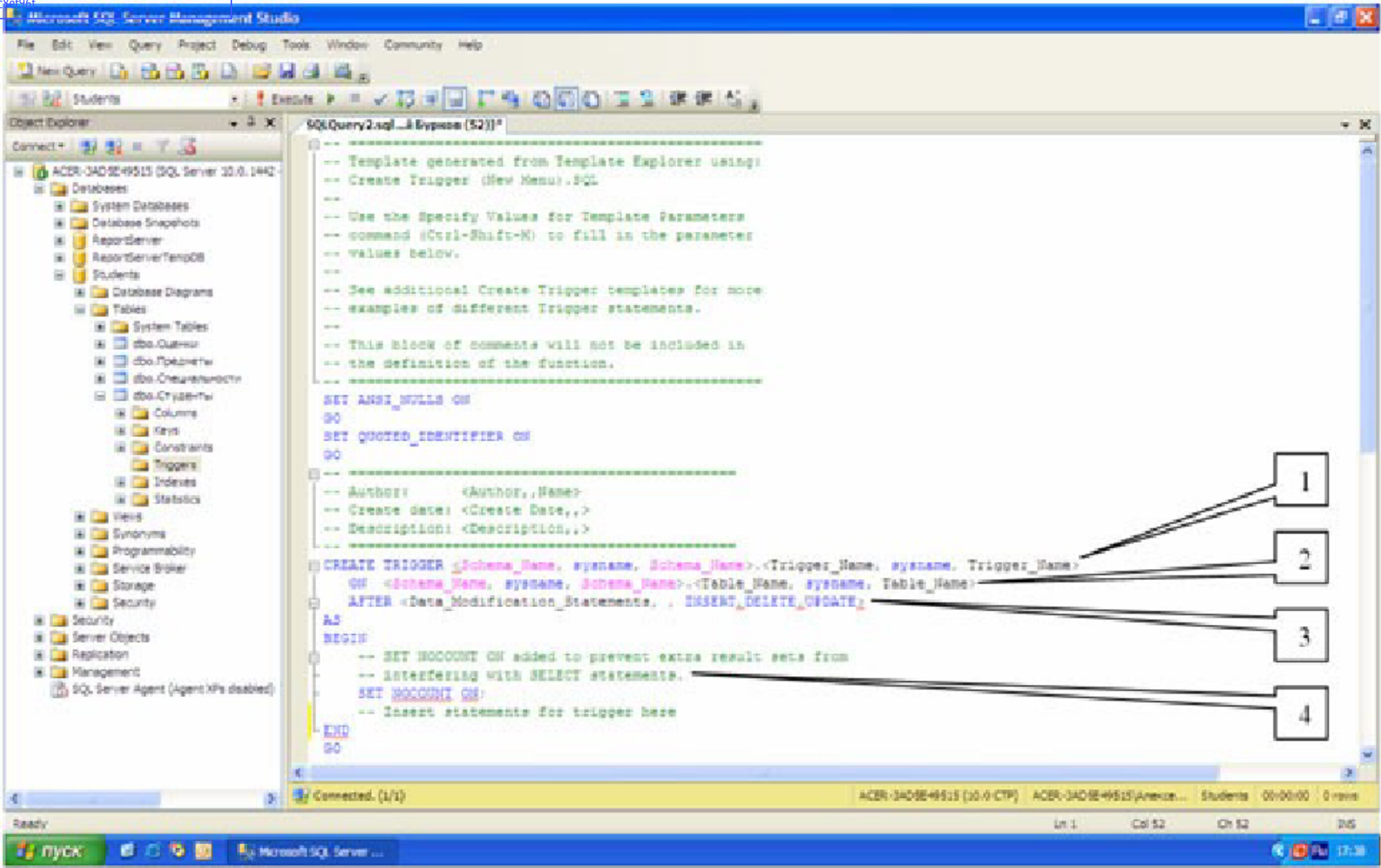




увеличить изображение

Рис. 14.10.

Рассмотрим структуру триггеров:

1. Область определения имени функции (**Trigger_Name**);
2. Область, показывающая для какой таблицы создается триггер (**Table_Name**);
3. Область, показывающая когда выполнять триггер (**INSERT** - при создании записи в таблице, **DELETE** - при удалении и **UPDATE** - при изменении) и как его выполнять (**AFTER** - после выполнения операции, **INSTEAD OF** - вместо выполнения операции);
4. Тело триггера, содержит команды языка программирования запросов T-SQL.

В окне нового триггера наберите код как показано на [рис. 14.11](#).

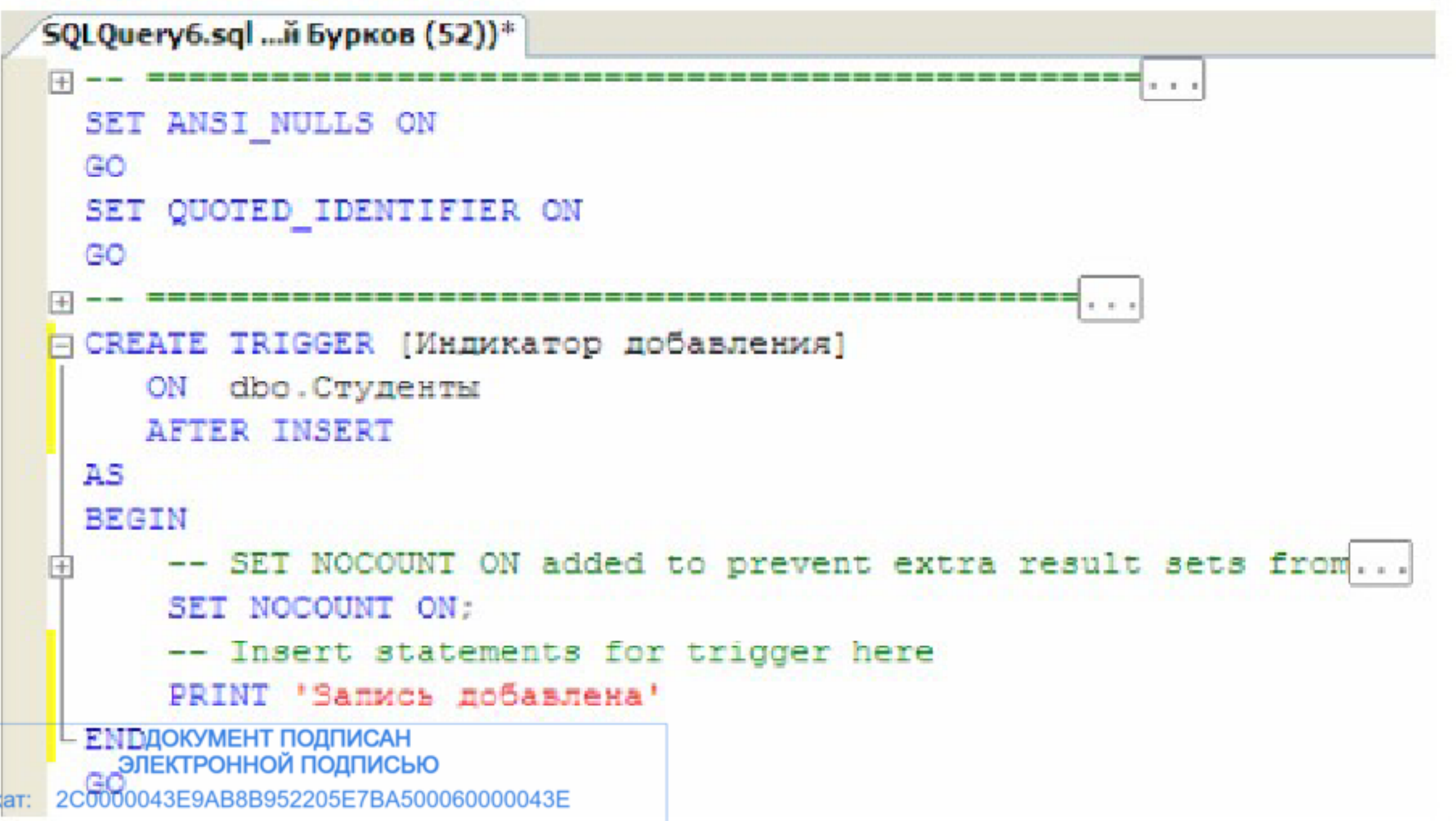
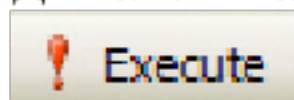


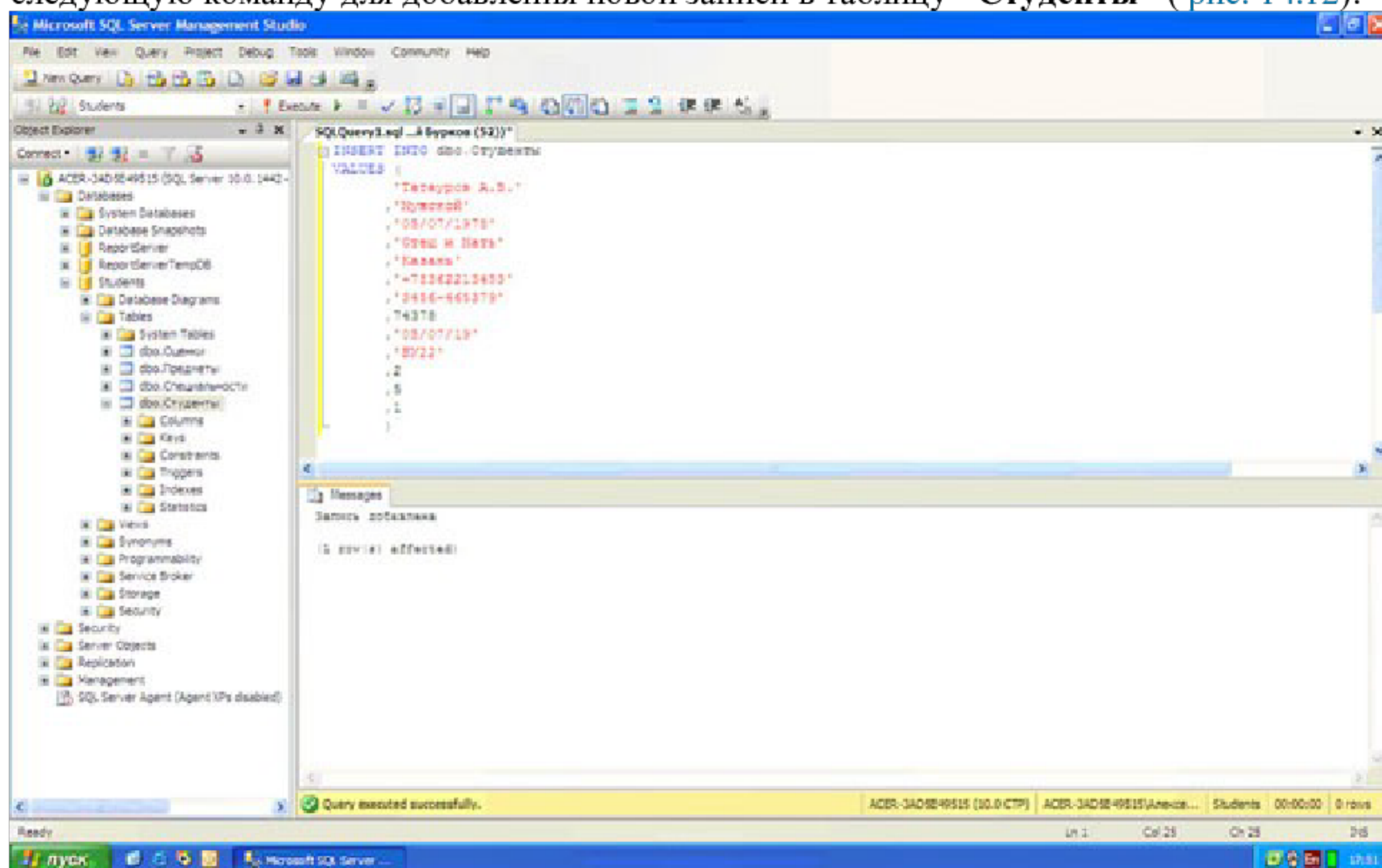
Рис. 14.11.

Из [рис. 14.11](#) видно, что создаваемый триггер "**Индикатор добавления**" выполняется после добавления записи (**AFTER INSERT**) в таблицу "Студенты" (**ON dbo.Студенты**). После добавления записи триггер выведет на экран сообщение "Запись добавлена" (**PRINT 'Запись добавлена'**). Выполните набранный код, нажав кнопку



на панели инструментов. В нижней части окна с кодом появится сообщение "**Command(s) completed successfully.**".

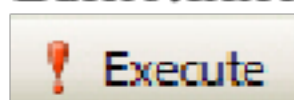
Проверим, как работает новый триггер. Создайте новый пустой запрос и в нем наберите следующую команду для добавления новой записи в таблицу "Студенты" ([рис. 14.12](#)):



[увеличить изображение](#)

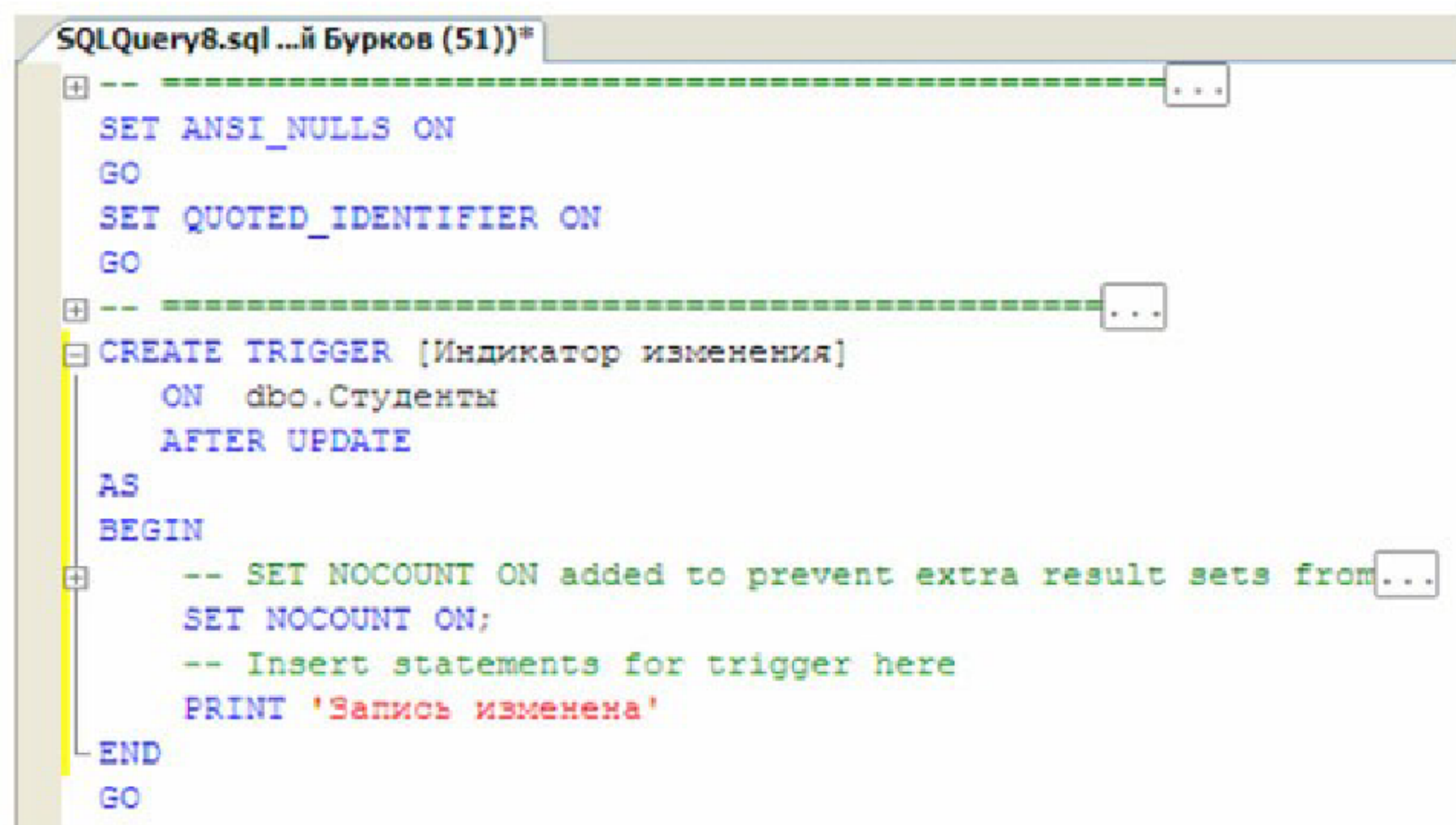
Рис. 14.12.

Выполните набранную команду, нажав кнопку



на панели инструментов. В таблицу будет добавлена новая запись, и триггер выведет сообщение "**Запись добавлена**" ([рис. 14.12](#)).

Теперь создадим триггер отображающий сообщение "**Запись изменена**". Создайте новый триггер, как в предыдущем случае. В окне нового триггера наберите следующий код ([рис. 14.13](#)):

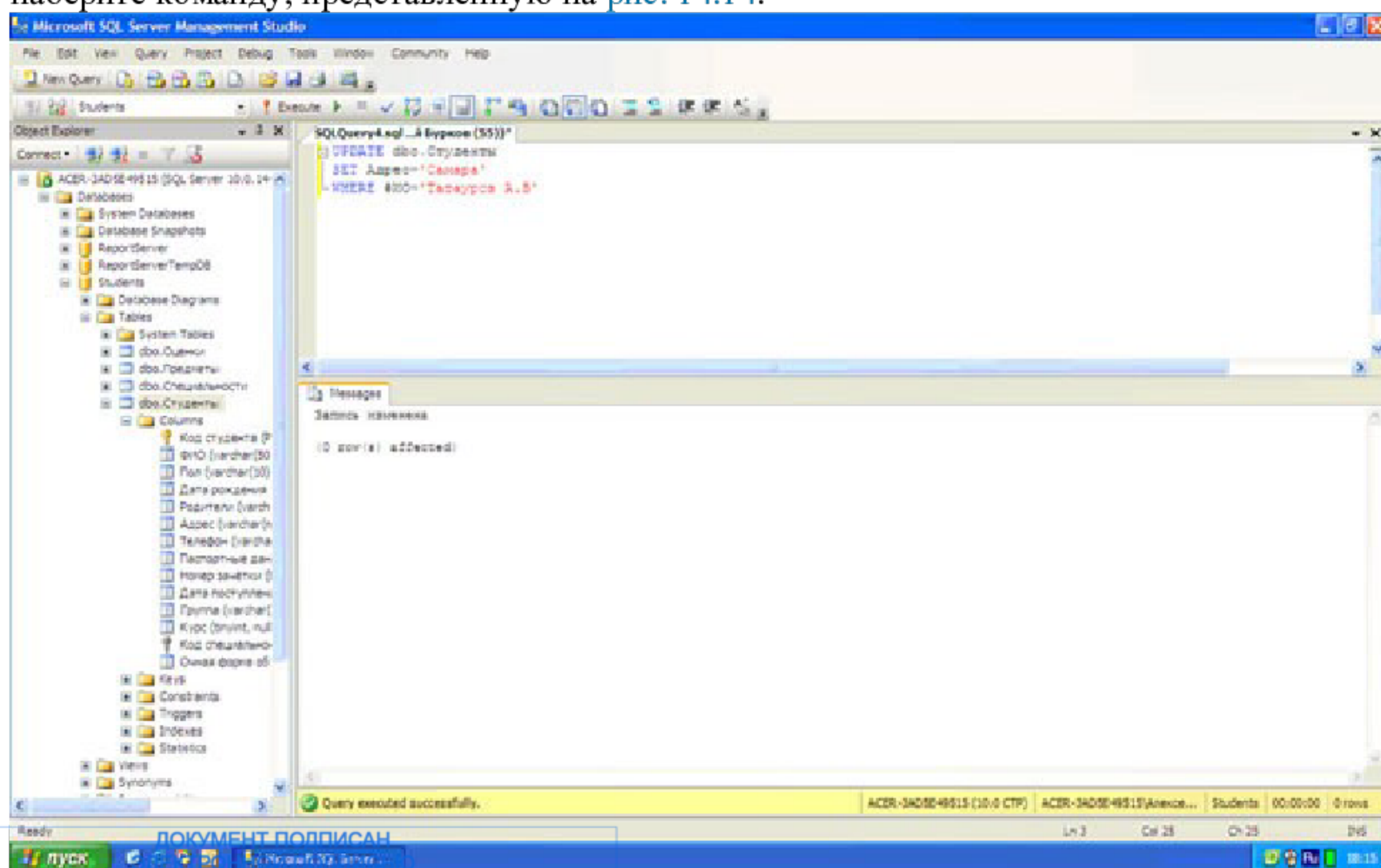


увеличить изображение

Рис. 14.13.

Из [рис. 14.13](#) видно, что новый триггер **"Индикатор изменения"** выполняется после изменения записи (**AFTER UPDATE**) в таблице **"Студенты"** (**ON dbo.Студенты**). После изменения записи триггер выведет на экран сообщение **"Запись изменена"** (**PRINT 'Запись изменена'**). Выполните набранный код. В нижней части окна с кодом появится сообщение **"Command(s) completed successfully."**.

Проверим работоспособность созданного триггера. Создайте новый запрос и в нем наберите команду, представленную на [рис. 14.14](#).

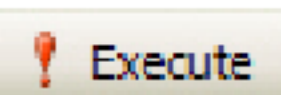


увеличить изображение

Рис. 14.14.

Выполните набранную команду, нажав кнопку

Действителен: с 19.08.2022 по 19.08.2023



на панели инструментов. В таблицу будет добавлена новая запись, и триггер выведет сообщение "Запись изменена" (рис. 14.14).

Для полноты картины создадим триггер, выводящий сообщение при удалении записи из таблицы "Студенты". Создайте новый триггер и в нем наберите код, показанный на рис. 14.15.

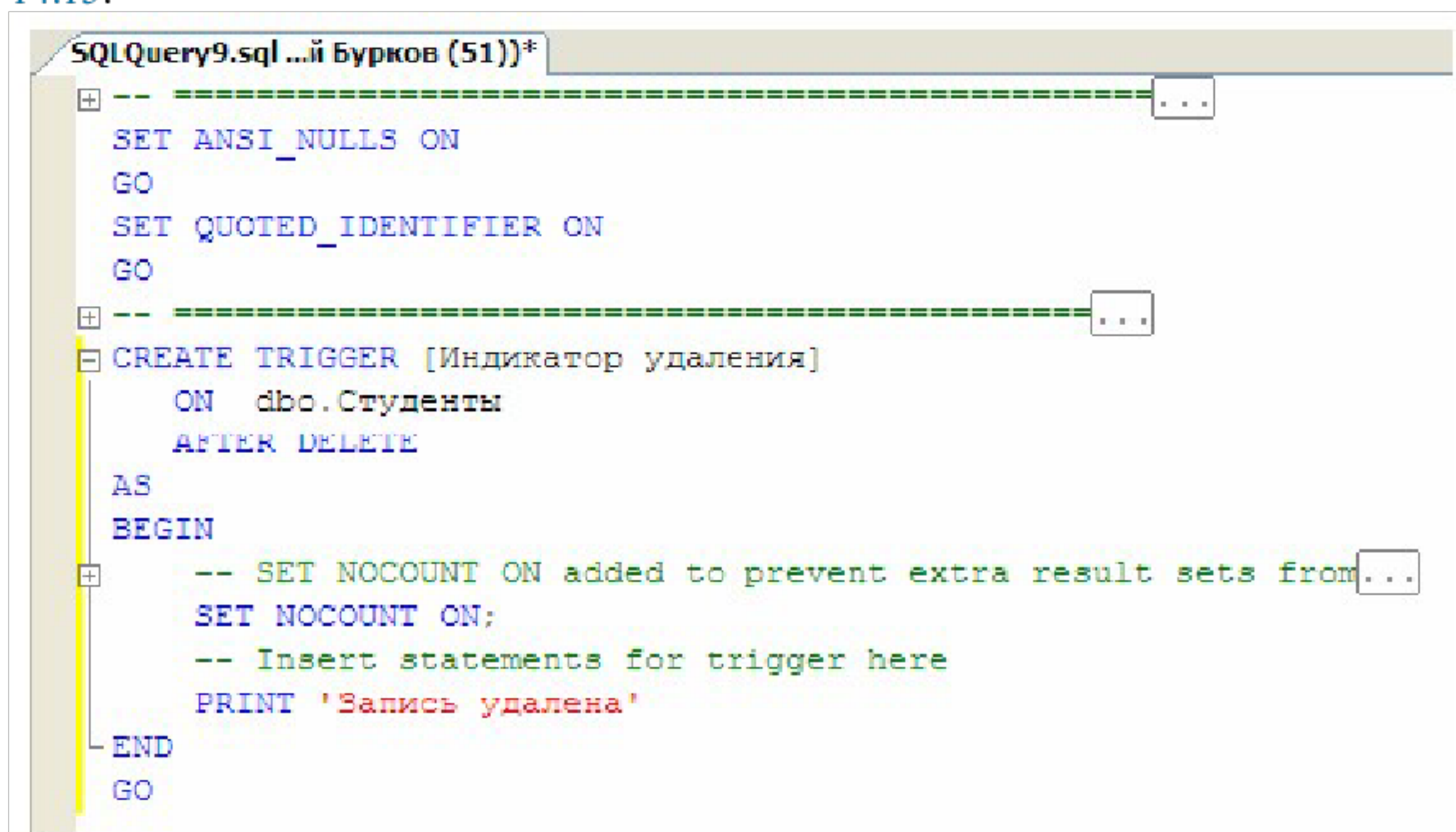
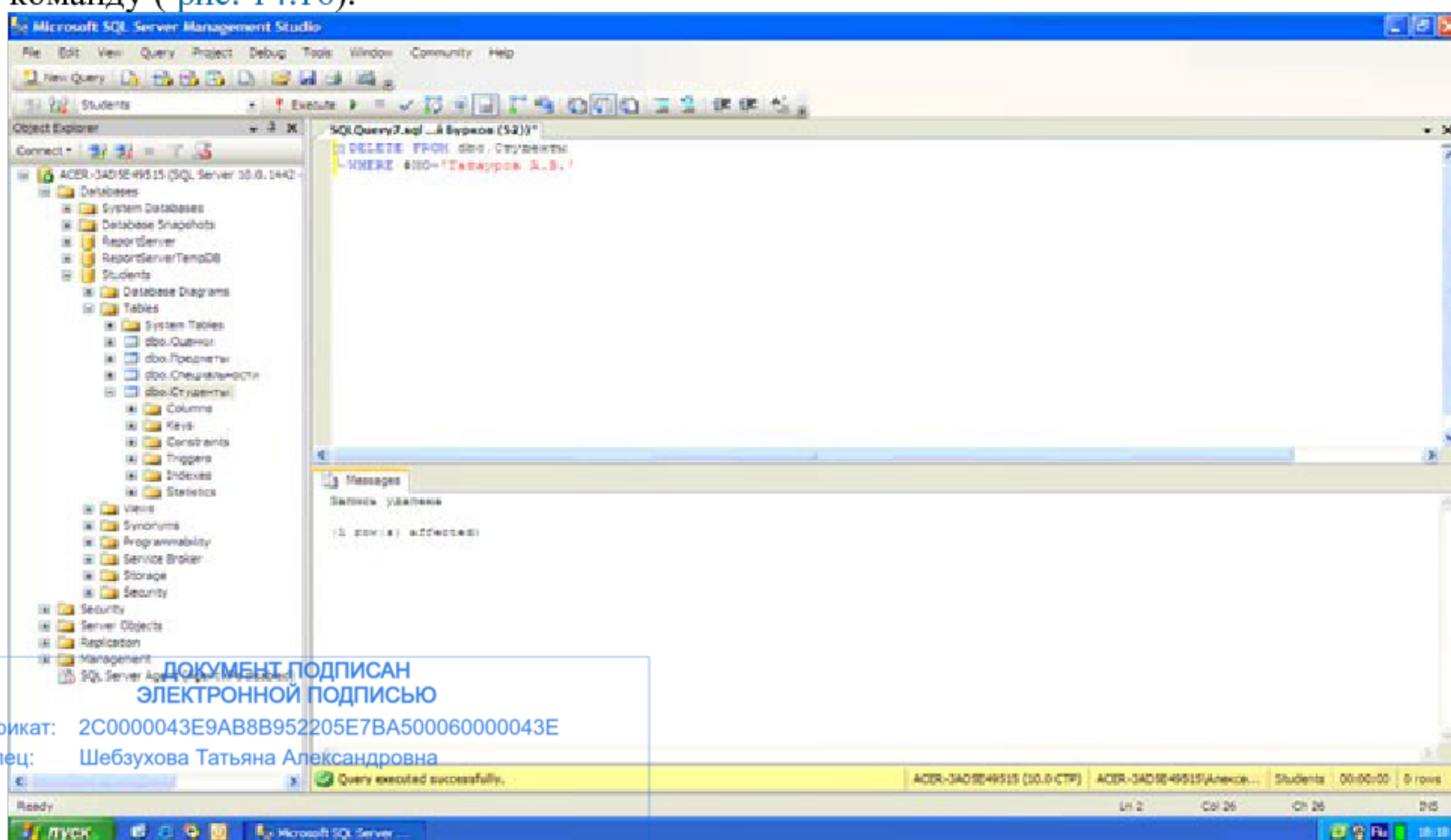


Рис. 14.15.

Создаваемый триггер "Индикатор удаления" выполняется после удаления записи (**AFTER DELETE**) из таблицы студенты (**ON dbo.Студенты**). После удаления записи триггер выводит сообщение "Запись удалена" (**PRINT 'Запись удалена'**).

Выполните код, представленный рис. 14.15. В нижней части окна с кодом появится сообщение "Command(s) completed successfully."

Проверим работу триггера "Индикатор удаления" удалив созданную ранее запись из таблицы "Студенты". Для этого создайте новый запрос и в нем наберите следующую команду (рис. 14.16):



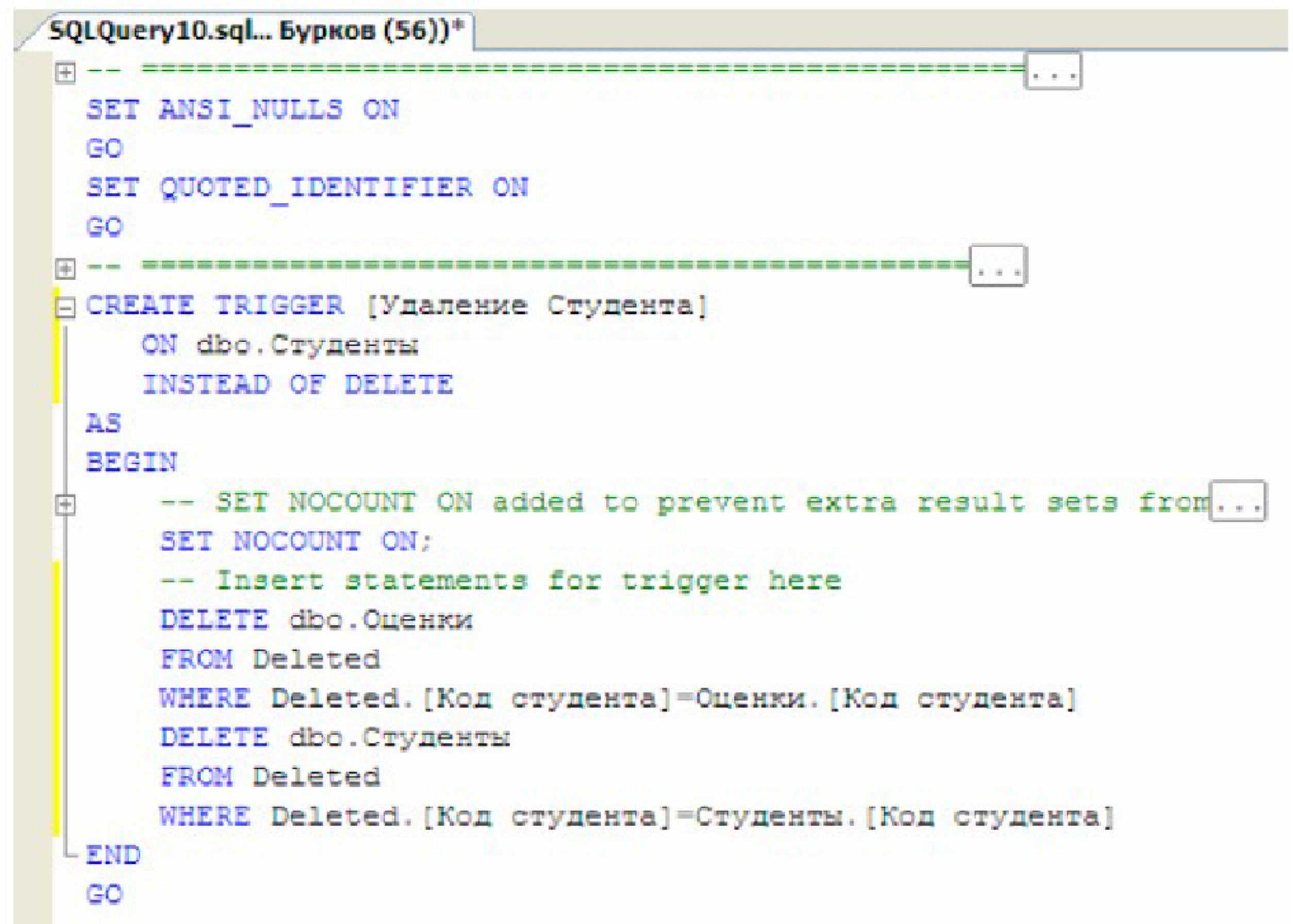
[увеличить изображение](#)

Рис. 14.16.

Выполните вышеприведенную команду. После удаления записи триггер "**Индикатор удаления**" отобразит сообщение "**Запись удалена**" ([рис. 14.16](#)).

В заключение рассмотрим пример применения триггеров для обеспечения целостности данных. Создадим триггер "**Удаление студента**", который при удалении записи из таблицы **Студенты** сначала удаляет все связанные с ней записи из таблицы "**Оценки**", а затем удаляет саму запись из таблицы "**Студенты**", тем самым обеспечивается целостность данных.

Создайте новый триггер и в нем наберите следующий код ([рис. 14.17](#)):



```
SQLQuery10.sql... Бурков (56))*)
-- =====
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
CREATE TRIGGER [Удаление Студента]
ON dbo.Студенты
INSTEAD OF DELETE
AS
BEGIN
-- SET NOCOUNT ON added to prevent extra result sets from
SET NOCOUNT ON;
-- Insert statements for trigger here
DELETE dbo.Оценки
FROM Deleted
WHERE Deleted.[Код студента]=Оценки.[Код студента]
DELETE dbo.Студенты
FROM Deleted
WHERE Deleted.[Код студента]=Студенты.[Код студента]
END
GO
```

Рис. 14.17.

Создаваемый триггер "**Удаление студента**" выполняется вместо удаления записи (**INSTEAD OF DELETE**) из таблицы "**Студенты**" (**ON dbo.Студенты**).

Замечание: При срабатывании триггера вместо удаления записи создается временная константа **Deleted**, содержащая имя таблицы из которой должно было быть произведено удаление.

После срабатывания триггера из таблицы "**Оценки**" удаляется запись, у которой значение поля "**Код студента**" равно значению такого же поля у удаляемой записи из таблицы "**Студенты**". Эту операцию выполняют следующие команды:

```
DELETE dbo.Оценки FROM Deleted
WHERE Deleted.[Код студента] = Оценки.[Код студента]
```

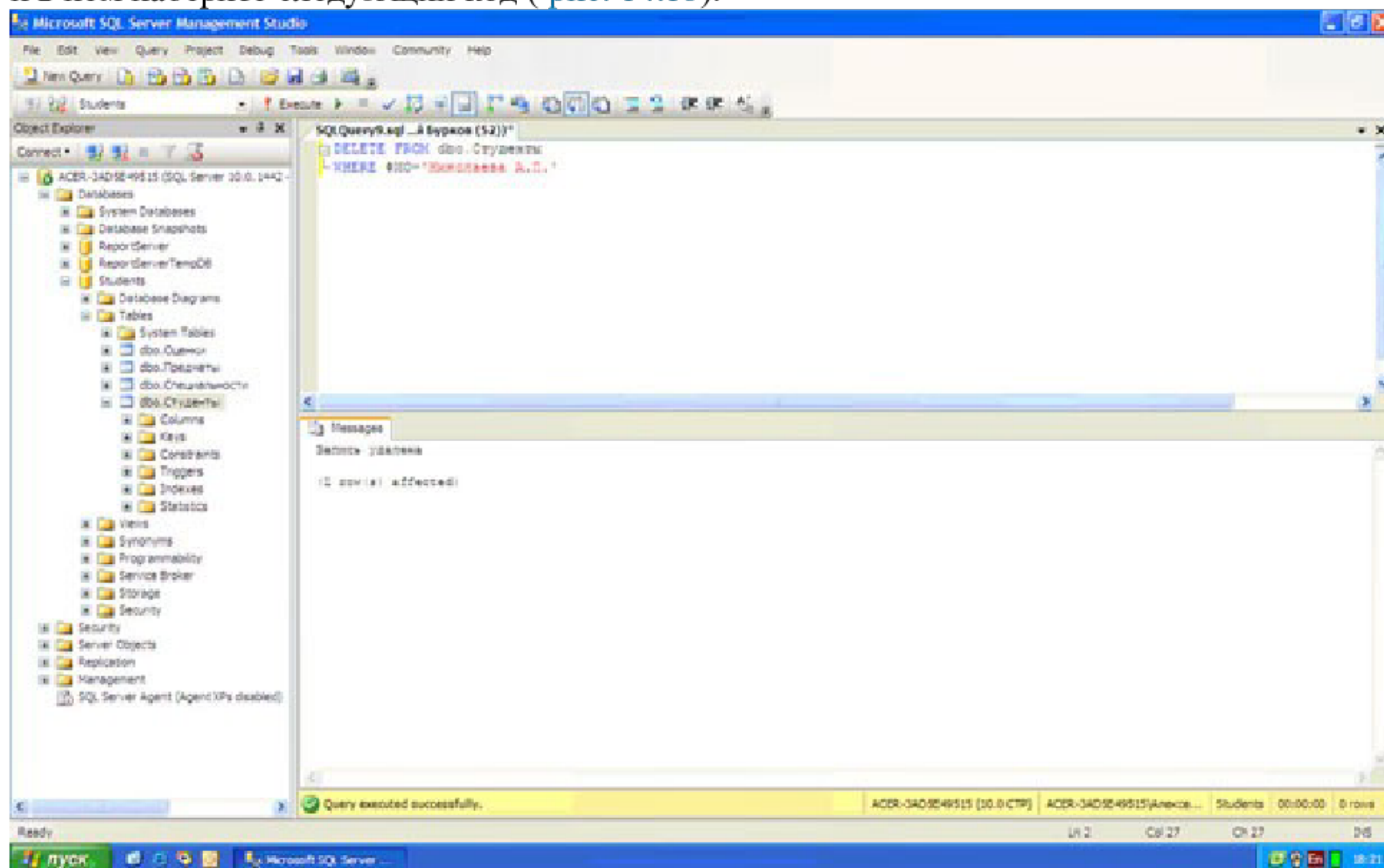
Затем удаляется запись из таблицы "**Студенты**", которую удаляли до срабатывания триггера. Удаление выполняется следующими командами:

```
DELETE dbo.Студенты
FROM Deleted
WHERE Deleted.[Код студента] = Студенты.[Код студента]
```

Выполните код, представленный на [рис. 14.17](#). В нижней части окна с кодом появиться сообщение "**Command(s) completed successfully.**".

Документ подписан
ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат: 200000043E7A000432206E7BA06000000043E
Владелец: Ильясов Тимур Азаматович
Действителен: с 19.08.2022 по 19.08.2023

Проверим, как работает триггер **"Удаление студента"**. Для этого создайте новый запрос и в нем наберите следующий код (рис. 14.18):



[увеличить изображение](#)

Рис. 14.18.

При срабатывании триггера сначала из таблицы **"Оценки"** удалятся все связанные с удаляемой записью записи, а затем удаляется сама удаляемая запись из таблицы **"Студенты"**, при этом сохраняется целостность данных.

Замечание: Хотелось бы заметить, что без использования триггера **"Удаление студента"** нам бы не удалось удалить запись из таблицы **"Студенты"**. Команда удаления была бы заблокирована диаграммой **"Диаграмма БД Студенты"** во избежание нарушения целостности данных.

На этом мы завершаем работу с диаграммами и триггерами. После выполнения всех вышеописанных действий обозреватель объектов будет иметь следующий вид (рис. 14.19):

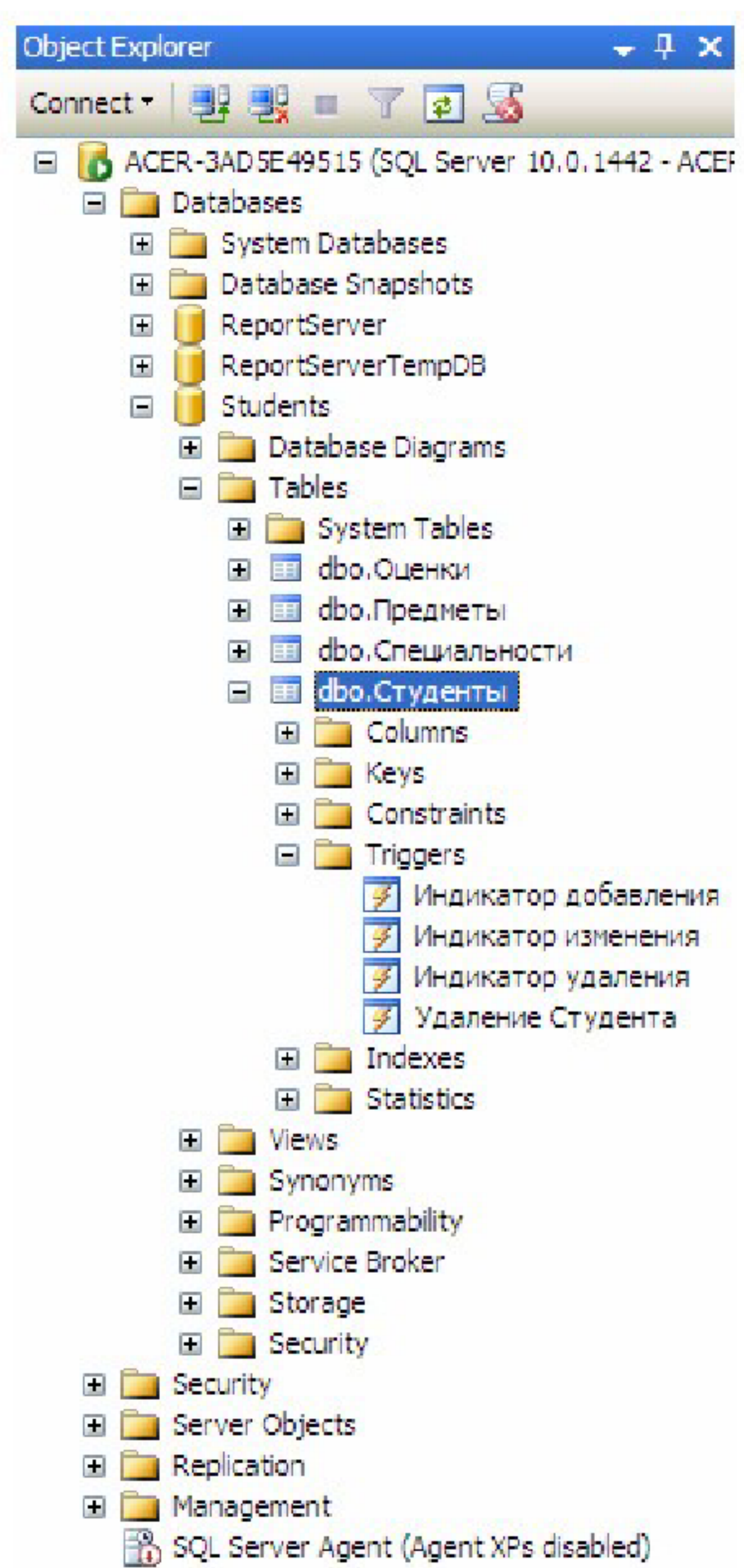


Рис. 14.19.

Контрольные вопросы

1. Структура триггеров.
2. Для чего используется диаграмма.
3. Связи между таблицами.

Лабораторная работа № 10

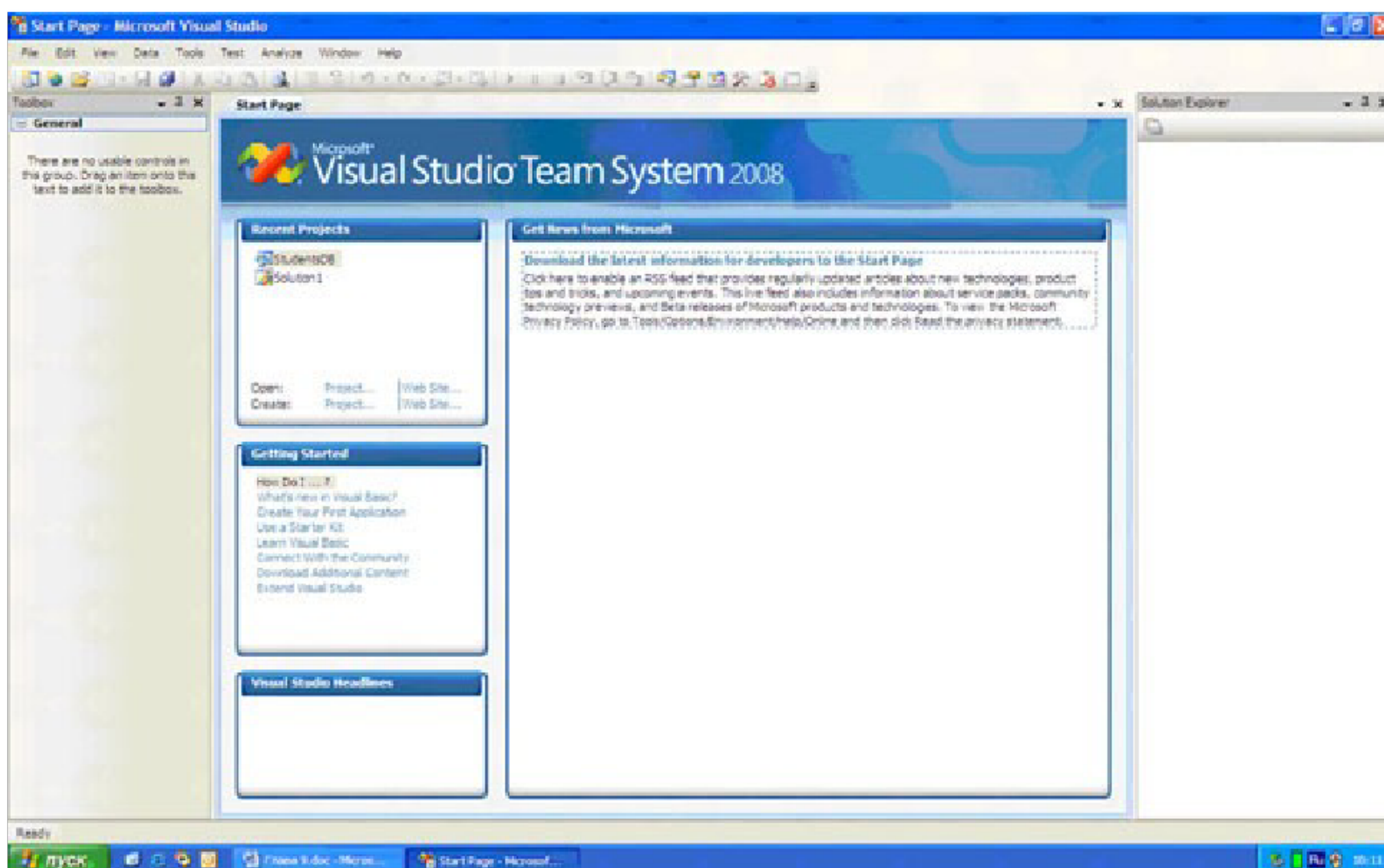
«Создание простых ленточных форм для работы с базами данных в Visual Studio 2012.»

Цель работы:

Научиться создавать пользовательский интерфейс (главная кнопочная форма, простые ленточные формы для работы с данными).

Перейдем теперь к созданию пользовательского интерфейса. Его создание начнем с создания главной кнопочной формы. Запустите "Microsoft Visual Studio 2008" и откройте созданный ранее проект "StudentsDB", щелкнув по его значку в области "Recent Projects" стартовой страницы "Start Page" (рис. 18.1).

Документ подписан
электронно
Сертификат: 2C0000043E9AB8B952205E7BA500060000043E
Владелец: Шелухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023



увеличить изображение

Рис. 18.1.

После появления стандартного окна среды разработки в рабочей области на форму поместите надпись (**Label**) и четыре кнопки (**Button**) как показано на [рис. 18.2](#).

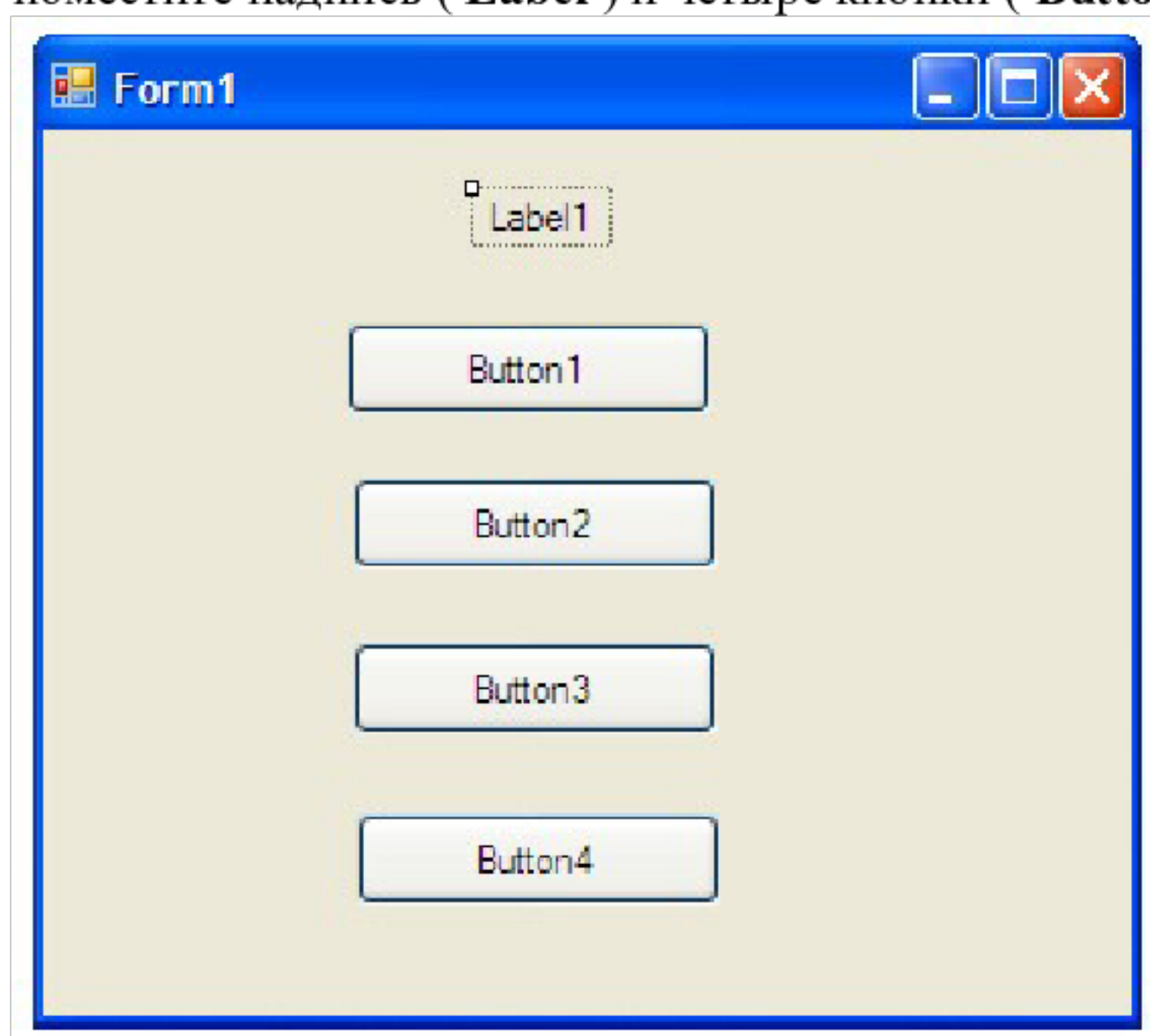


Рис. 18.2.

Замечание: Для создания надписи на панели объектов необходимо нажать кнопку

 Label

а затем нарисовать прямоугольник мышью на форме, удерживая **ЛКМ**. Кнопки создаются таким же образом, только на панели объектов нажмите кнопку

 Button

Действителен: с 19.08.2022 по 19.08.2023

После создания объектов перейдем к настройке их свойств. Начнем с настройки свойств формы. Выделите форму, щелкнув **ЛКМ** в пустом месте формы. На панели свойств задайте свойства формы как представлено ниже:

- **FormBorderStyle** (Стиль границы формы): Fixed3D;
- **MaximizeBox** (Кнопка разворачивания формы во весь экран): False;
- **MinimizeBox** (Кнопка свертывания формы на панель задач): False;
- **Text** (Текст надписи в заголовке формы): База данных "Студент".

На форме выделите надпись, щелкнув по ней **ЛКМ** и на панели свойств, задайте свойства надписи следующим образом:

- **AutoSize** (Авторамер): False;
- **Font** (Шрифт): Microsoft Sans Serif, размер 14;
- **ForeColor** (Цвет текста): Темно синий;
- **Text** (Текст надписи): База данных "Студент";
- **TextAlign** (Выравнивание текста): MiddleCenter.

У кнопок задайте надписи (свойство **"Text"**) как показано на [рис. 18.3](#).

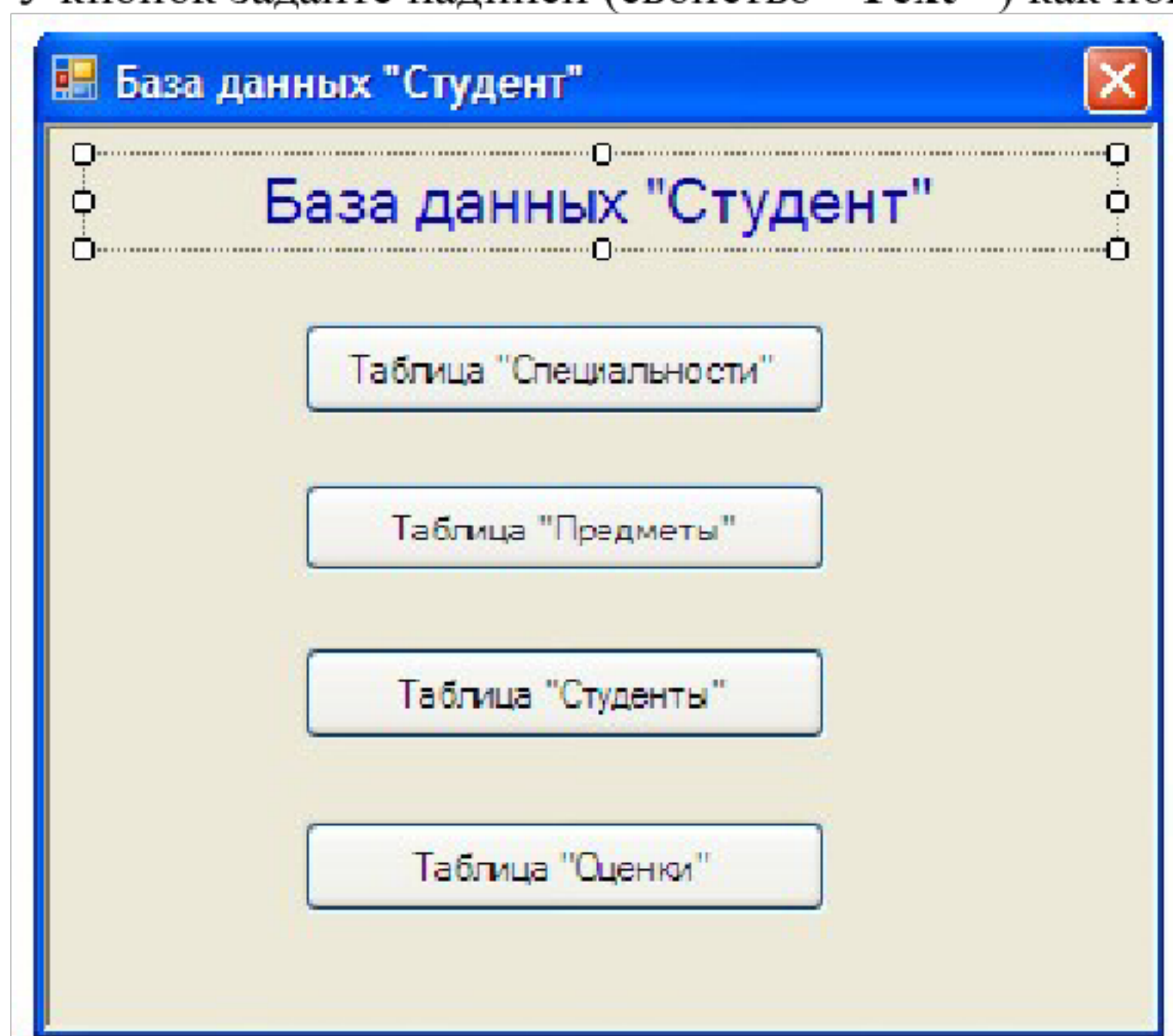
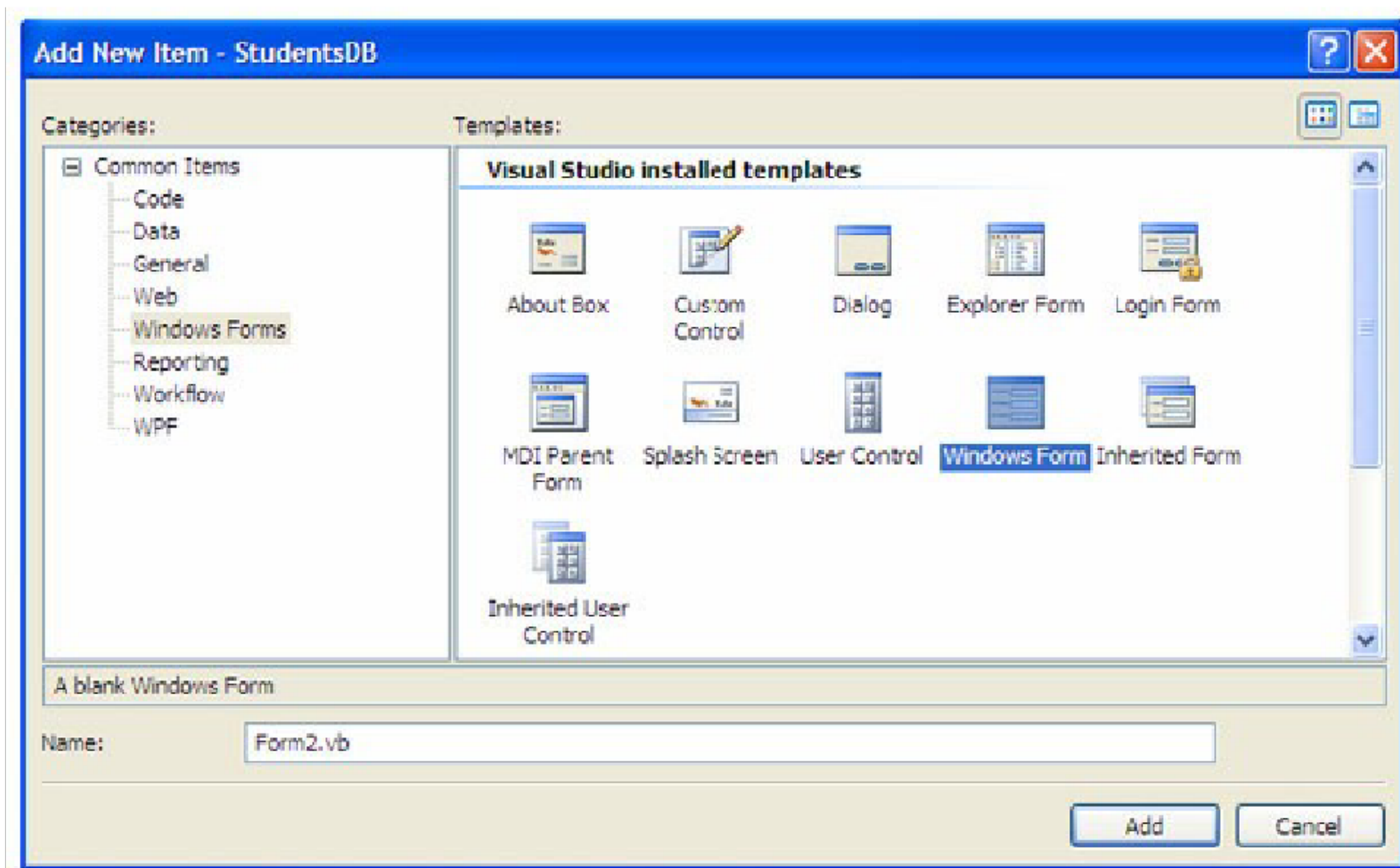


Рис. 18.3.

После настройки свойств вышеперечисленных объектов форма примет вид представленный на [рис. 18.3](#).

Теперь перейдем к созданию простых ленточных форм для работы с данными. Для начала создадим ленточную форму, отображающую таблицу **"Специальности"**. Добавим в проект новую пустую форму. Для этого в оконном меню выберите пункт **"Project/Add Windows Form"**. Появится окно **"Add New Item - StudentsDB"** (Добавить новый компонент) ([рис. 18.4](#)).



[увеличить изображение](#)

Рис. 18.4.

В данном окне в разделе **"Categories:"** (Категории) выберите **"Windows Forms"** (Формы Windows), затем в разделе **"Templates:"** (Шаблоны) выберите **"Windows Form"** (Форма Windows) и нажмите кнопку **"Add"** (Добавить). Новая пустая форма появится в рабочей области среды разработки.

В верхней части новой формы создайте надпись (**Label**), как это показано на [рис. 18.5](#).



Рис. 18.5.

Перейдем к настройке свойств формы и надписи. Выделите форму, щелкнув **ЛКМ** в пустом месте формы. На панели свойств задайте свойства формы следующим образом:

- **FormBorderStyle** (Стиль границы формы): Fixed3D;

Действителен: с 19.08.2022 по 19.08.2023

- **MaximizeBox** (Кнопка разворачивания формы во весь экран): False;
- **MinimizeBox** (Кнопка сворачивания формы на панель задач): False;
- **Text** (Текст надписи в заголовке формы): Таблица "Специальности".

На форме выделите надпись, щелкнув по ней **ЛКМ** и на панели свойств, задайте свойства надписи как показано ниже:

- **AutoSize** (Автора размер): False;
- **Font** (Шрифт): Microsoft Sans Serif, размер 14;
- **ForeColor** (Цвет текста): Темно синий;
- **Text** (Текст надписи): Таблица "Специальности";
- **TextAlign** (Выравнивание текста): MiddleCenter.

После настройки всех вышеперечисленных свойств форма будет выглядеть следующим образом (рис. 18.6):

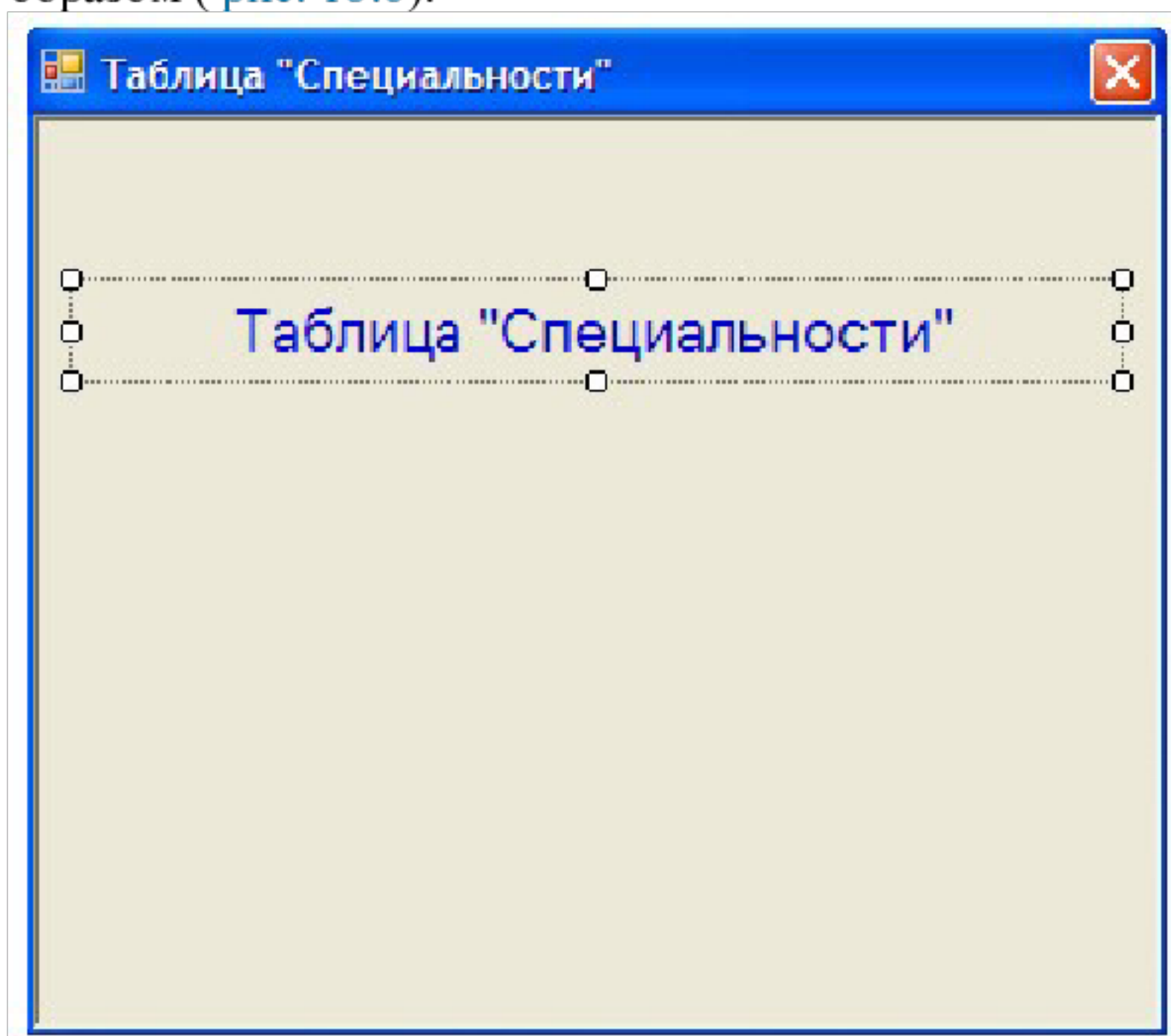


Рис. 18.6.

Теперь поместим на форму поля таблицы "Специальности". Сначала откройте панель "Источники данных" (Data Sources), щелкнув по ее вкладке в правой части окна среды разработки (смотри рис. 18.6). На панели "Источники данных" отобразите поля таблицы "Специальности", щелкнув по значку "+", расположенному слева от имени таблицы (рис. 18.7).

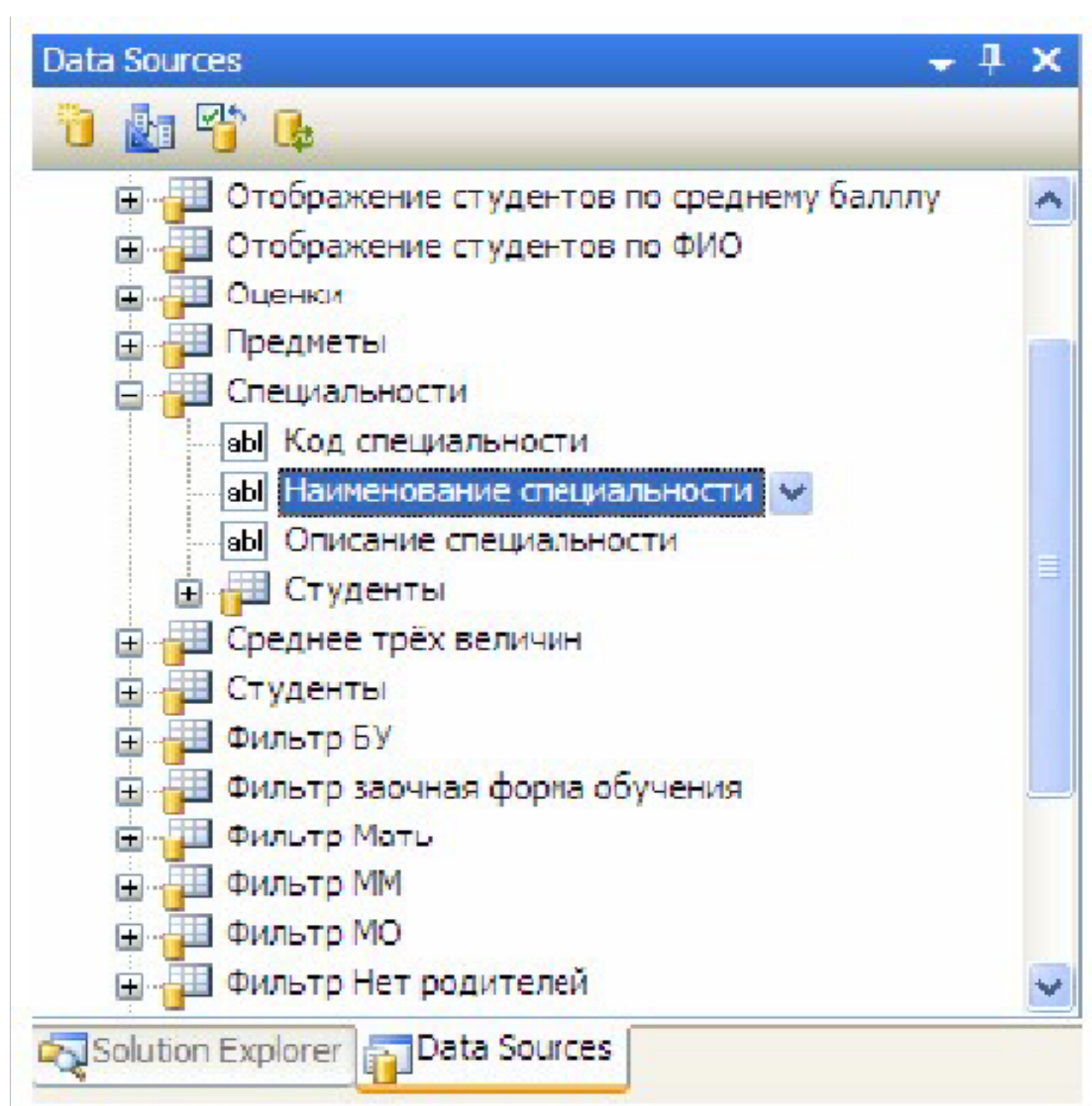


Рис. 18.7.

Панель "Источники данных" примет вид, представленный на [рис. 18.7](#).

Замечание: Под полями таблицы специальности в виде подтаблицы располагается таблица "Студенты" ([рис. 18.7](#)). Подтаблица показывает, что таблица "Студенты" является вторичной по отношению к таблице специальности.

Замечание: При выделении, какого либо поля таблицы, оно будет отображаться в виде выпадающего списка ([рис. 18.7](#)), позволяющего выбирать объект, отображающий содержимое выделенного поля ([рис. 18.8](#)).

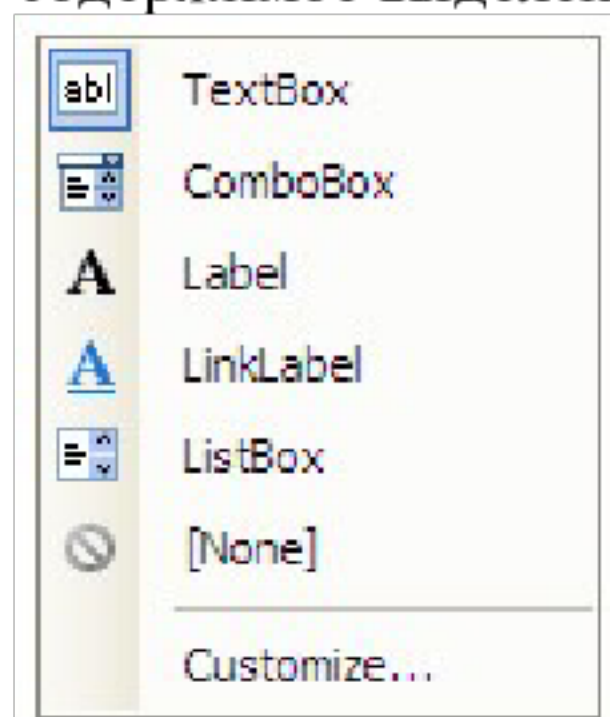
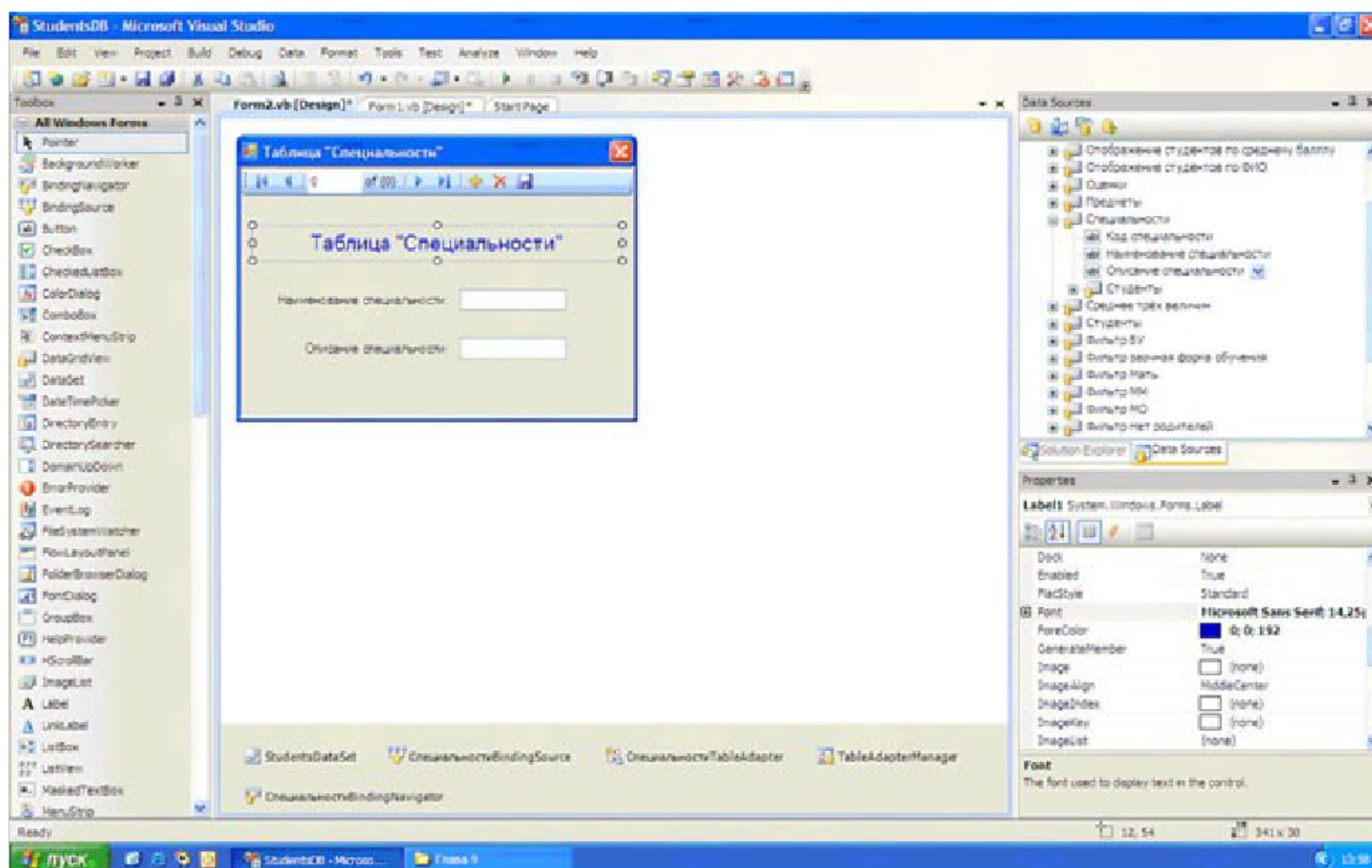


Рис. 18.8.

Для того чтобы поместить на новую форму поля таблицы их необходимо перетащить из панели "Источники данных" на форму. Из таблицы "Специальности" перетащите мышью на форму поля "Наименование специальности" и "Описание специальности". Форма примет вид, представленный на [рис. 18.9](#)



увеличить изображение

Рис. 18.9.

Замечание: Мы не помещаем поле "**Код специальности**" на нашу форму, так как данное поле является первичным полем связи и заполняется автоматически. Конечный пользователь не должен видеть такие поля.

Замечание: Обратите внимание, что после перетаскивания полей с панели "**Источники данных**" на форму в верхней части формы появилась навигационная панель, а в нижней части рабочей области среды разработки появились пять невидимых объектов. Эти объекты предназначены для связи нашей формы с таблицей "**Специальности**", расположенной на сервере. Рассмотрим функции этих объектов:

- **StudentDataSet** (Набор данных Student) - обеспечивает подключение формы к конкретной БД на сервере (в нашем случае это БД Students);
- **СпециальностиBindingSource** (Источник связи для таблицы "**Специальности**") - обеспечивает подключение к конкретной таблице (в нашем случае к таблице специальности), а также позволяет управлять таблицей;
- **СпециальностиTableAdapter** (Адаптер таблиц для таблицы "**Специальности**") - обеспечивает передачу данных с формы в таблицу и наоборот.
- **TableAdapterManager** (Менеджер адаптера таблиц) - управляет работой объекта **СпециальностиTableAdapter** ;
- **СпециальностиBindingNavigator** (Панель управления таблицей "**Специальности**") - голубая панель с кнопками управления таблицей, расположенная в верхней части формы

Теперь нам необходимо проверить работоспособность новой формы. Для отображения формы "**Специальности**" ее необходимо подключить к главной кнопочной форме, а затем запустить проект и открыть форму "**Специальности**" при помощи кнопки на главной кнопочной форме.

Отобразите главную кнопочную форму в рабочей области среды разработки, щелкнув по вкладке **Form1.vb [Design]** в верхней части рабочей области. Для подключения новой формы "**Специальности**" к главной кнопочной форме дважды щелкните ЛКМ по кнопке "**Таблица "Специальности"**", расположенной на главной кнопочной форме (рис. 18.3).

В появившемся окне кода формы в процедуре "**Button1_Click**" наберите команду

ДОКУМЕНТ ПОДПИСАН
Электронный документ
Сертификат: 2C0010613E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023

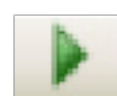
"Form2.Show()", предназначенную для открытия формы "Таблица "Специальности"" (Form2), как это показано на [рис. 18.10](#).

```
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Form2.Show()
    End Sub
End Class
```

[увеличить изображение](#)

Рис. 18.10.

Теперь запустим проект, нажав на панели инструментов кнопку



На экране появится главная кнопочная форма. Для открытия формы, отображающей таблицу "Специальности" на главной кнопочной форме нажмите кнопку "Таблица "Специальности"". Появится форма с соответствующей таблицей ([рис. 18.11](#)).

Рис. 18.11.

Проверьте работу панели навигации, расположенной в верхней части формы, понажимав на ней различные кнопки. Вернитесь в среду разработки, просто закрыв форму с таблицей "Специальности" и главную кнопочную форму.

Теперь создадим форму для просмотра таблицы предметы. Добавьте в проект новую форму. На форму добавьте надпись. Настройте свойства формы и надписи, как это было сделано для формы таблицы "Специальности". Затем из таблицы "Предметы" на новую форму поместите поля "Наименование предмета" и "Описание предмета". После выполнения всех вышеописанных действий форма для таблицы предметы будет выглядеть следующим образом ([рис. 18.12](#)):

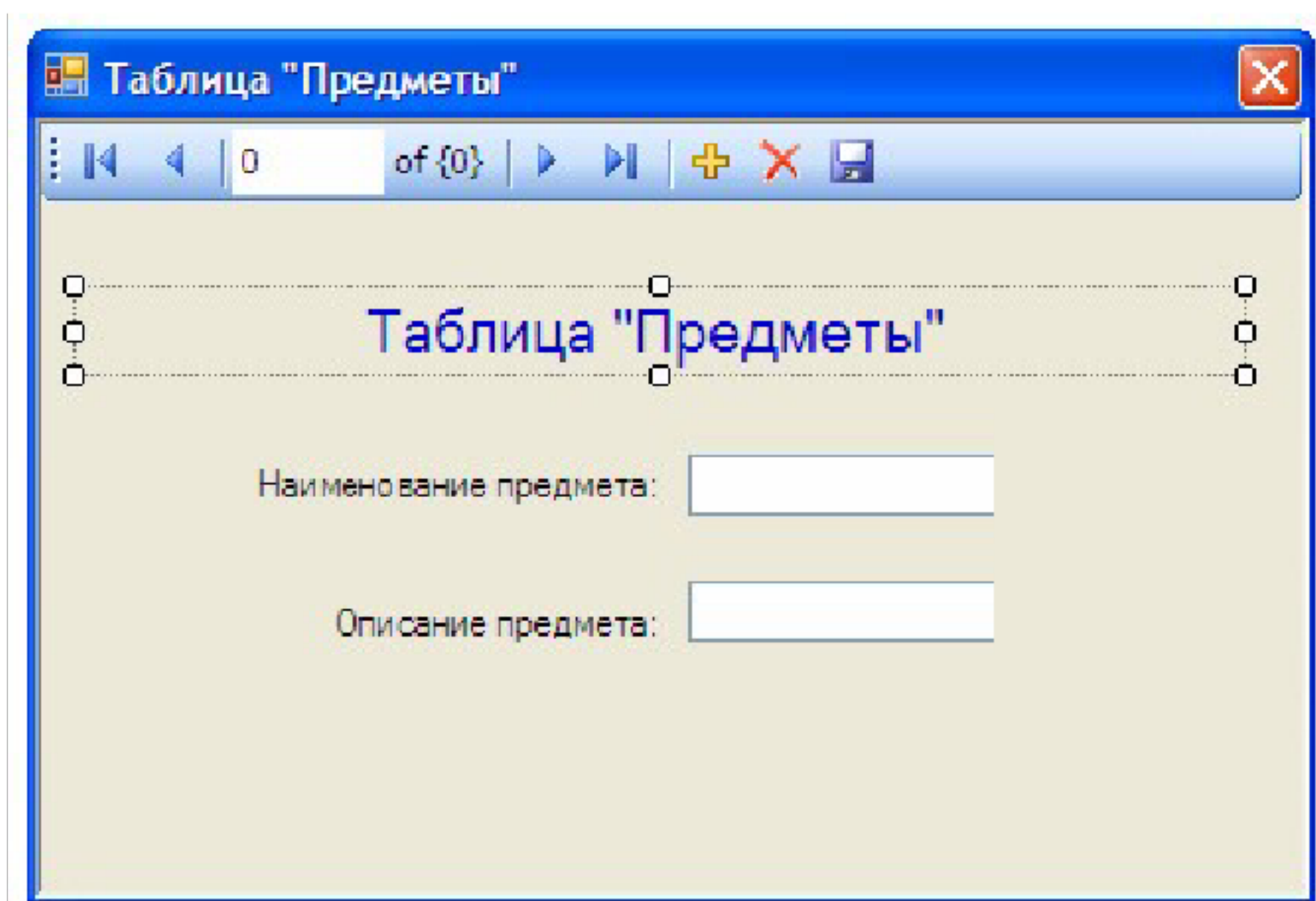


Рис. 18.12.

На главной кнопочной форме дважды щелкните ЛКМ по кнопке **"Таблица "Предметы"** и в появившемся окне кода в процедуре **"Button2_Click"** наберите **"Form3.Show()"** (рис. 18.13).

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    Form3.Show()
End Sub
End Class
```

увеличить изображение

Рис. 18.13.

Проверим работу новой формы, отображающей таблицу **"Предметы"**. Запустите проект и на главной кнопочной форме нажмите кнопку **"Таблица "Предметы"**. Отобразится таблица предметов имеющая следующий вид (рис. 18.14):

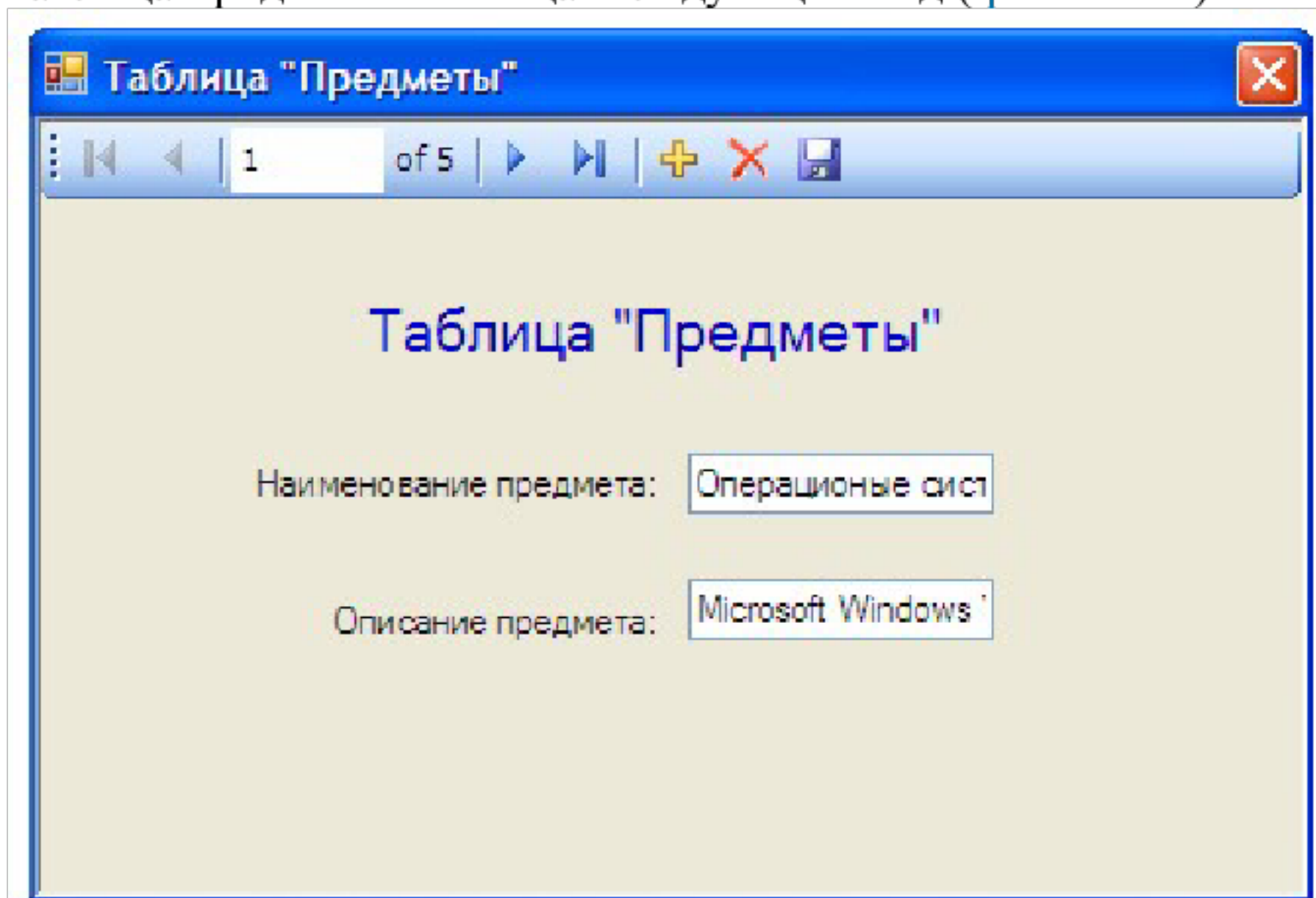


Рис. 18.14.

Проверьте работу панели навигации, нажатием на кнопки на данной панели в верхней части формы. Для возвращения в среду разработки закройте форму таблицы **"Предметы"** и главную кнопочную форму.

ДОКУМЕНТ ПОДПИСАН
Электронно-цифровой подписью
Сертификат: 2C8B00043E9AB93B652205573A4500060000043E
Владелец: Шебухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023

Теперь создадим простую ленточную форму для отображения таблицы "Студенты". Для начала отобразите поля таблицы "Студенты" на панели "Источники данных", щелкнув ЛКМ по знаку "+", расположенному слева от названия таблицы. Отобразятся все поля таблицы "Студенты" (рис. 18.15).

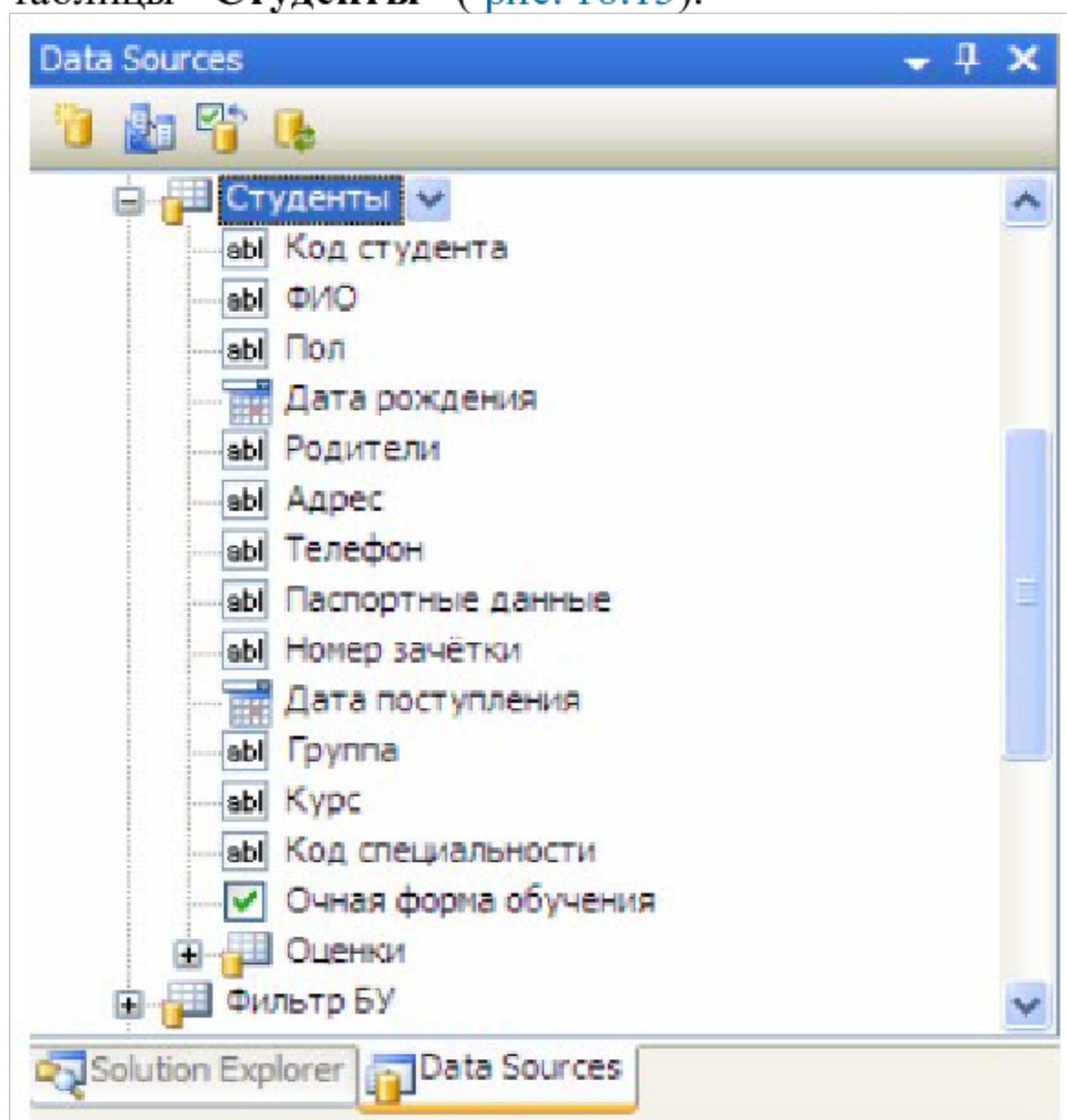


Рис. 18.15.

Замечание: Обратите внимание на тот факт, что поля "Дата рождения" и "Дата поступления" отображаются объектом "Выбор даты" (DataPicker), так как данные поля содержат значения дат. Поле "Очная форма обучения" является логическим, следовательно, для его отображения используется объект "Переключатель" (CheckBox). Остальные поля отображаются при помощи текстовых полей ввода (TextBox) (рис. 18.15).

Создайте новую форму и поместите в ее верхнюю часть надпись. Задайте заголовок формы как "Таблица "Студенты"". В верхнюю часть формы поместите надпись. В качестве текста надписи задайте тот же самый текст, что был задан в качестве заголовка формы. Настройте свойства формы и надписи, аналогично формам созданным ранее. На форму с панели "Источники данных" переместите все поля кроме поля "Код студента". Так как данное поле является первичным полем связи. Новая форма примет вид (рис. 18.16):

Рис. 18.16.

Обратите внимание на объекты, отображающие поля **"Дата рождения"**, **"Дата поступления"** и **"Очная форма обучения"**.

Подключим форму, отображающую таблицу **"Студенты"** к главной кнопочной форме. Отобразите главную кнопочную форму и на ней дважды щелкните ЛКМ по кнопке **"Таблица "Студенты"**". В появившемся окне кода, в процедуре **"Button3_Click"** наберите следующую команду для открытия формы таблицы **"Студенты"** - **"Form4.Show"** (рис. 18.17).

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
    Form4.Show()
End Sub
```

[увеличить изображение](#)

Рис. 18.17.

Теперь запустим проект. На экране появится главная кнопочная форма. Для открытия формы, отображающей таблицу **"Студенты"** на главной кнопочной форме нажмите кнопку **"Таблица "Студенты"**". Появится форма с соответствующей таблицей (рис. 18.18).

The screenshot shows a web browser window with the title "Таблица 'Студенты'". The browser's address bar shows "1 of 9" and navigation icons. The form itself has a title "Таблица 'Студенты'" and contains the following fields:

- ФИО:
- Пол:
- Дата рождения: (dropdown arrow)
- Родители:
- Адрес:
- Телефон:
- Паспортные данные:
- Номер зачетки:
- Дата поступления: (dropdown arrow)
- Группа:
- Курс:
- Код специальности:
- Очная форма обучения: ☒

Рис. 18.18.

Проверьте работу формы нажатием кнопок на панели навигации, расположенной в верхней части формы. Закройте форму, отображающую таблицу "Студенты" и главную кнопочную форму.

Аналогичным образом создайте форму для отображения таблицы "Оценки". Добавьте на новую форму надпись, добавьте на форму все поля из таблицы "Оценки" и настройте их свойства, как описано выше. В итоге, форма для отображения таблицы "Оценки" будет выглядеть следующим образом (рис. 18.19):

Рис. 18.19.

Подключите вновь созданную форму таблицы **"Оценки"** к главной кнопочной форме. Для этого отобразите главную кнопочную форму и на ней дважды щелкните ЛКМ по кнопке **"Таблица "Оценки"**". В появившемся окне с кодом, в процедуре **"Button4_Click"** наберите команду **"Form5.Show"** ([рис. 18.20](#)).

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
    Form5.Show()
End Sub
```

[увеличить изображение](#)

Рис. 18.20.

Проверьте работу формы таблицы **"Оценки"**, запустив проект, и на главной кнопочной форме нажмите кнопку **"Таблица "Оценки"**". Появится вновь созданная форма ([рис. 18.21](#)).

Цель работы:

Научиться создавать сложные ленточные формы для работы с данными.

Модернизируем форму для таблицы "Студенты". Сначала программно продублируем кнопки панели навигации, расположенной в верхней части формы. Откройте проект "StudentsDB" и отобразите форму таблицы студенты (**Form4**). В нижней части формы расположите семь кнопок, как это показано на [рис. 20.1](#).

Таблица "Студенты"

0 of {0}

ФИО:

Пол:

Дата рождения: 15 ноября 2008 г.

Родители:

Адрес:

Телефон:

Паспортные данные:

Номер зачётки:

Дата поступления: 15 ноября 2008 г.

Группа:

Курс:

Код специальности:

Очная форма обучения: ☐

Button1 Button2 Button3

Button4 Button5 Button6

Button7

Рис. 20.1.

В качестве надписей на созданных кнопках (Свойство "Caption") задайте как: "Первая", "Предыдущая", "Добавить", "Последняя", "Следующая", "Удалить" и "Сохранить" ([рис. 20.2](#)).

Рис. 20.2.

Дважды щелкните ЛКМ по кнопке "Первая" и в появившемся окне кода формы "Form4" в процедуре "Button1_Click" наберите команду для перехода к первой записи "СтудентыBindingSource.MoveFirst()" (рис. 20.3).

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    СтудентыBindingSource.MoveFirst()
End Sub
```

увеличить изображение

Рис. 20.3.

Дважды щелкните ЛКМ по кнопке "Предыдущая" и в появившемся окне кода формы "Form4" в процедуре "Button2_Click" наберите команду для перехода к предыдущей записи "СтудентыBindingSource.MovePrevious()" (рис. 20.4).

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    СтудентыBindingSource.MovePrevious()
End Sub
```

увеличить изображение

Рис. 20.4.

Сертификат: 220000043E9AB8B952205E7BA500060000043E
Владелец: Шебзухова Татьяна Александровна

Действителен: с 19.08.2022 по 19.08.2023

Дважды щелкните ЛКМ по кнопке "Добавить" и в появившемся окне кода формы "Form4" в процедуре "Button3_Click" наберите команду для добавления новой записи "СтудентыBindingSource.AddNew()" (рис. 20.5).

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
    СтудентыBindingSource.AddNew()
End Sub
```

увеличить изображение

Рис. 20.5.

Дважды щелкните ЛКМ по кнопке "Последняя" и в появившемся окне кода формы "Form4" в процедуре "Button4_Click" наберите команду для перехода к последней записи "СтудентыBindingSource.MoveLast()" (рис. 20.6).

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
    СтудентыBindingSource.MoveLast()
End Sub
```

увеличить изображение

Рис. 20.6.

Дважды щелкните ЛКМ по кнопке "Следующая" и в появившемся окне кода формы "Form4" в процедуре "Button5_Click" наберите команду для перехода к следующей записи "СтудентыBindingSource.MoveNext()" (рис. 20.7).

```
Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click
    СтудентыBindingSource.MoveNext()
End Sub
```

увеличить изображение

Рис. 20.7.

Дважды щелкните ЛКМ по кнопке "Удалить" и в появившемся окне кода формы "Form4" в процедуре "Button6_Click" наберите команду для удаления текущей записи "СтудентыBindingSource.RemoveCurrent()" (рис. 20.8).

```
Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button6.Click
    СтудентыBindingSource.RemoveCurrent()
End Sub
```

увеличить изображение

Рис. 20.8.

Дважды щелкните ЛКМ по кнопке "Сохранить" и в появившемся окне кода формы "Form4" в процедуре "Button7_Click" наберите команду для сохранения изменений, отображенную на рис. 20.9.

```
Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button7.Click
    Me.Validate()
    Me.СтудентыBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.StudentsDataSet)
End Sub
```

увеличить изображение

Рис. 20.9.

Рассмотрим последнюю процедуру более подробно. Она содержит следующие команды:

- Me.Validate() - проверяет введенные в поля данные на соответствие типам данных полей;
- Me.СтудентыBindingSource.EndEdit() - закрывает подключение с сервером;
- Me.TableAdapterManager.UpdateAll(Me.StudentsDataSet) - обновляет данные на сервере.

Для проверки работы созданных кнопок запустите проект откройте форму "Таблица Студенты" и нажмите каждую из кнопок.

Теперь изменим объекты, отображающие поля для более удобного ввода информации.

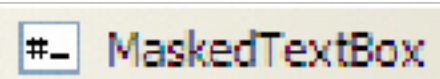
Для начала удалите текстовые поля ввода (TextBox), отображающие следующие поля таблицы "Студенты": "Пол", "Родители", "Телефон", "Паспортные данные",

Документ подписан
Электронной подписью
Сертификат
Владелец
Шебзухова Татьяна Александровна
Действителен: с 19.08.2022 по 19.08.2023

"Номер зачетки", "Курс" и "Код специальности". После удаления, перечисленных полей форма, отображающая таблицу "Студенты" примет следующий вид ([рис. 20.10](#)):

Рис. 20.10.

Для отображения полей "Телефон", "Паспортные данные" и "Номер зачетки" будем использовать текстовые поля ввода по маске (**MaskedTextBox**). Объект текстовое поле ввода по маске отсутствует в выпадающем списке объектов для отображения полей в окне "Источники данных", поэтому будем создавать данные объекты при помощи панели объектов (**Toolbox**), а затем подключать их к соответствующим полям вручную. Для создания текстовых полей ввода по маске на панели объектов используется кнопка



Создайте текстовые поля ввода по маске справа от надписей "Телефон", "Паспортные данные" и "Номер зачетки", как это показано на [рис. 20.11](#).

Рис. 20.11.

Теперь у созданных объектов настроим маски ввода. Начнем с объекта, отображающего номер зачетки. На форме выделите соответствующее полю **"Номер зачетки"** текстовое поле ввода по маске. Для задания маски в меню действий с объектом выберите пункт **"Set Mask..."** (Установить маску...) (рис. 20.12).

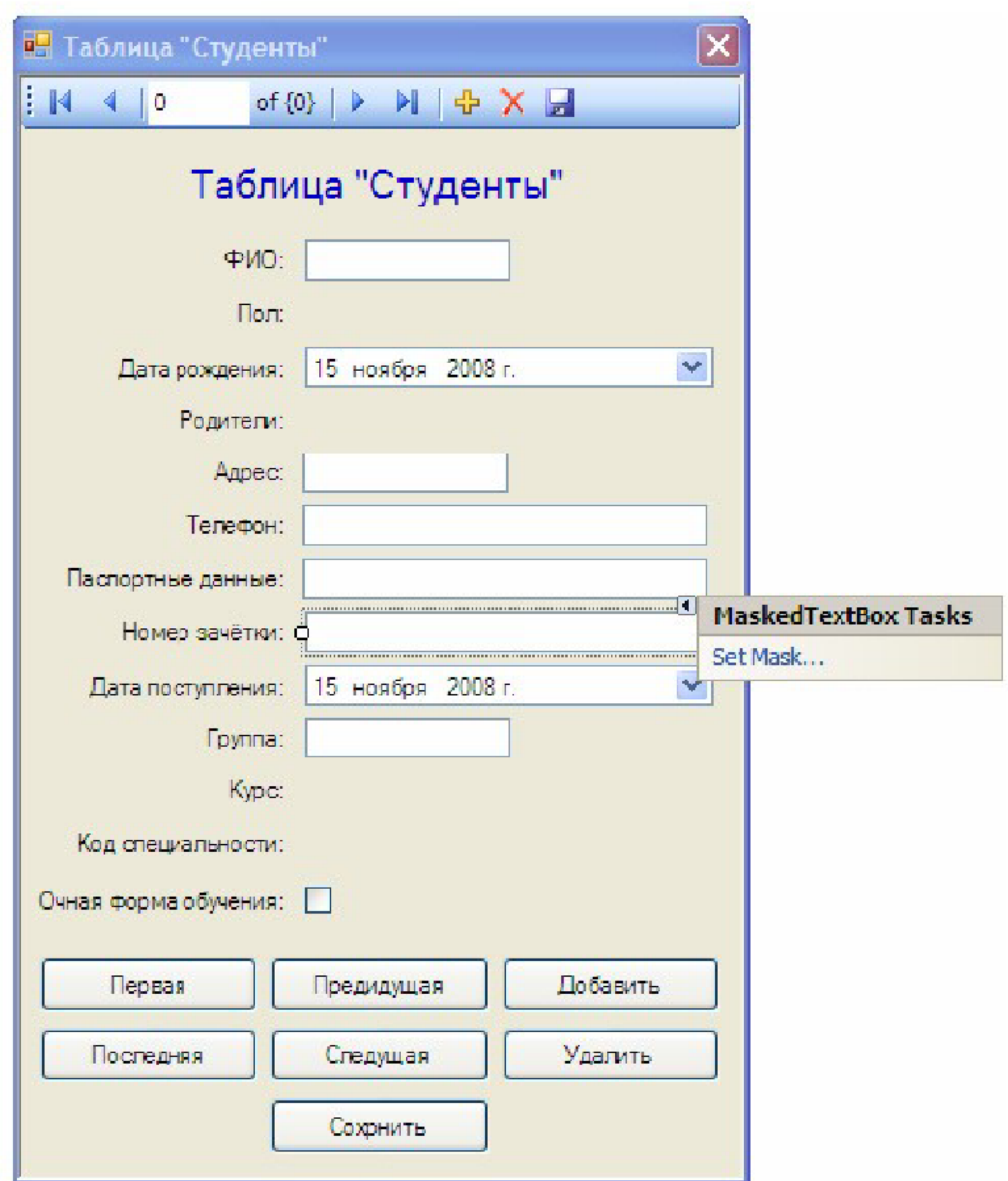


Рис. 20.12.

Замечание: Для отображения меню действий в верхнем правом углу объекта необходимо нажать кнопку



(рис. 20.12).

После выбора пункта **"Set Mask..."** на экране появится окно задания маски **"Input Mask"** (Введите маску) (рис. 20.13).

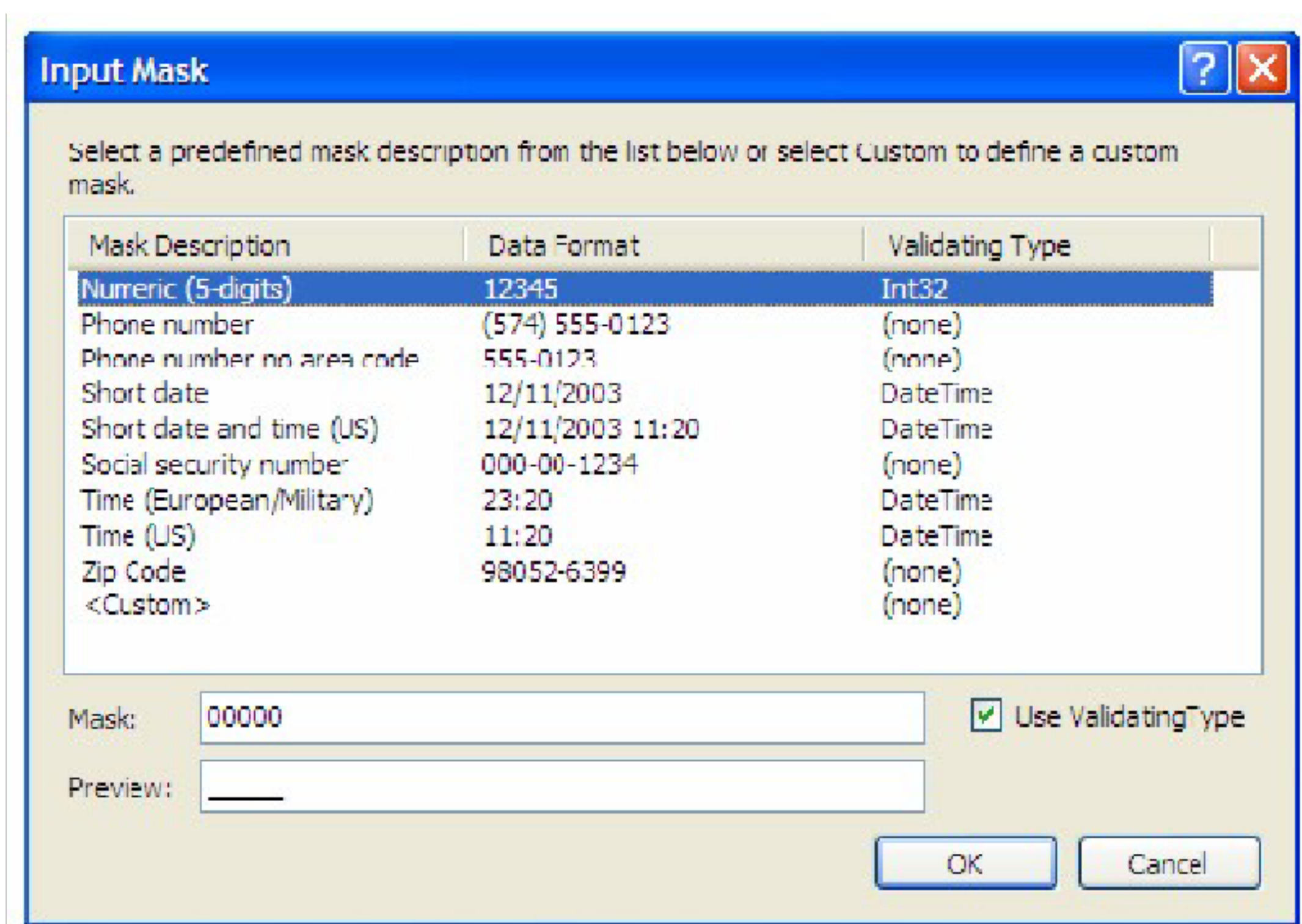


Рис. 20.13.

В окне **"Input Mask"** выберите маску **"Numeric (5-digits)"** (Числовое (5-цифр)) и нажмите кнопку **"Ok"** (рис. 20.13).

Для текстового поля ввода по маске для поля **"Паспортные данные"** задайте маску как показано на рис. 20.14.

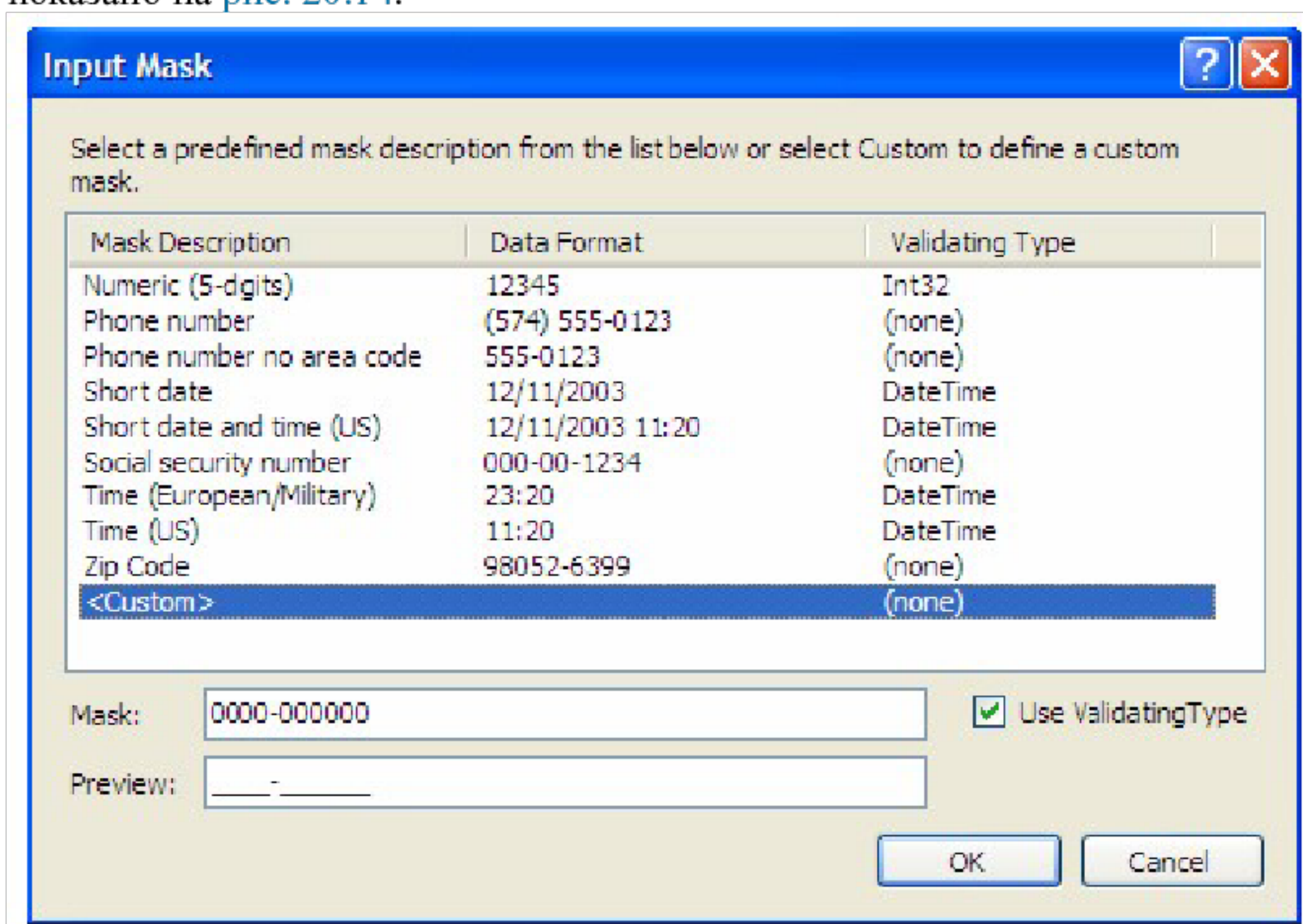


Рис. 20.14.

Замечание: Обратите внимание, что паспортные данные отображаются как четыре числа, три, шесть чисел. То есть в поле **"Mask"** (Маска) нужно задать **"0000-000000"**. Знак **"0"** обозначает цифру. В поле **"Preview"** (Предварительный просмотр) отображается вид текстового поля ввода по маске на форме.