

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ

«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ И  
ИНФОРМАТИКИ»

**Е.И. Ситняковская**

# **Основы Визуального Программирования на языке C#**

Методические указания по выполнению лабораторных работ

Часть 2

Новосибирск

2017

## Часть 2. Работа с базами данных в C#.

Цель работы: Создание небольшой реляционной базы данных и выгрузка ее в форму, так чтобы все связи изначальной базы данных были сохранены. Также необходимо создать инструменты управления выгруженными данными и создать поиск студента по заданной комбинации или по его точной фамилии.

Базы данных

	Код	Факультет	Курс	Количество групп
▶	1	МТС	1	4
	2	ИВТ	2	10
	3	АЭС	3	6
*				

... || ◀ | 1 для 3 ▶ || + ✕

	Код	Название группы	Фамилия старосты	Количество
▶	11	P-31	Петров	17
	12	P-32	Иванов	20
	13	P-33	Сидоров	18
*				

◀ || ... | 1 для 3 ▶ || + ✕

	Код	ФИО	Адрес	Телефон
	111	Филиппов Е. В.	Некрасова, 134	321-31-12
	112	Мазунин В.П.	Кирова, 117	343-32-54
	113	Федоринин А.В.	Выборная, 133	345-32-23
▶*				

◀ || ... | 4 для 4 ▶ || + ✕

Поиск по символам      Точный поиск      Сортировать

## Выполнение работы.

### Шаг 1. Создание базы данных.

В данной лабораторной работе будет рассмотрен способ создания БД через Microsoft Access. Все примеры и скриншоты будут взяты именно с него. Версия Access – 2013, Visual Studio – 2013.

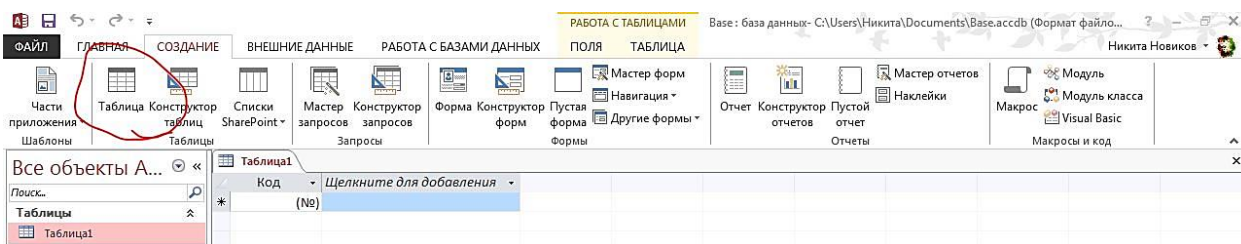
Задание: База данных должна содержать в себе три таблицы: факультеты, группы, студенты и иметь следующий тип связей:



Рисунок 1 Связи

### Выполнение:

Для начала создадим новый пустой проект в Access. По заданию требуется создать 3 таблицы. Одна таблица уже задана по умолчанию. Добавляем еще 2: заходим в пункт «СОЗДАНИЕ» и выбираем 2 раза «Таблица».



Далее обработаем каждую из них. Нужно переименовать их и добавить требуемые поля. Для этого кликаем правой кнопкой мыши по таблице и выбираем «Конструктор». Предлагается ввести имя таблицы. Первую называем «Факультет», вторую «Группа», третью – «Студенты». Далее открывается конструктор полей. Здесь нужно добавить соответствующие поля для каждой из трех таблиц (см. рисунок 1). Тип данных можно оставить по умолчанию (Короткий текст) кроме нескольких полей: «Код» (для всех трех таблиц), «Факультет ID» (для таблицы Группа) и «ID\_GR» (для таблица Студенты) – там ставим тип «Числовой». Нужно это для того, чтобы в дальнейшем создать связи между этими полями. Так как для того, чтобы

создать связи без потери данных (см ниже), необходимо, чтобы поля, между которыми будет связь, имели **один** тип данных.

Группа		Студенты	Факультет
Имя поля		Тип данных	
Код	Числовой	Факультет	Короткий текст
Курс	Короткий текст	Количество групп	Короткий текст

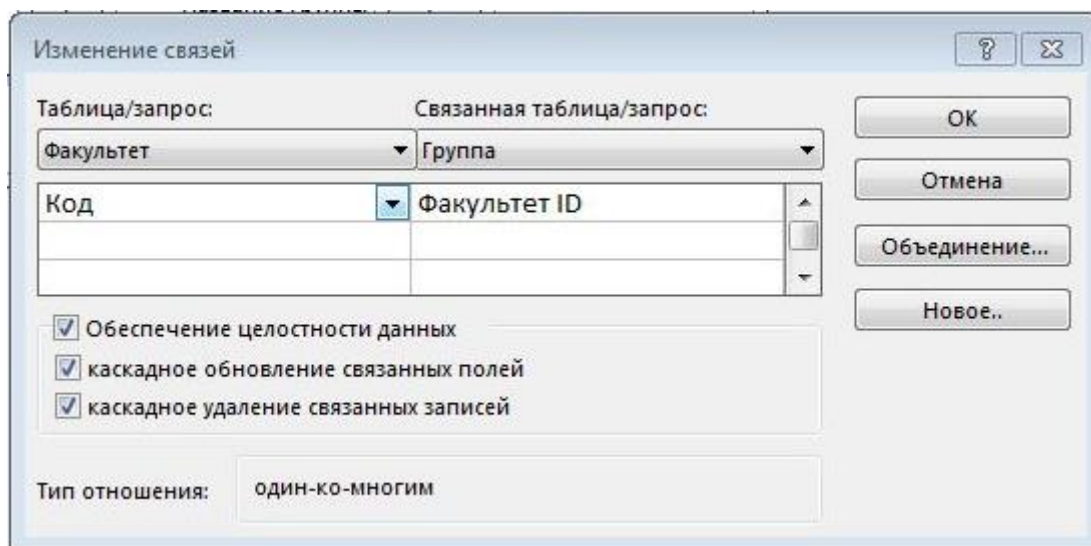
Группа		Студенты	Факультет
Имя поля		Тип данных	
Код	Числовой	ФИО	Короткий текст
Адрес	Короткий текст	Телефон	Короткий текст
ID_GR	Числовой		

Группа		Студенты	Факультет
Имя поля		Тип данных	
Код	Числовой	Название группы	Короткий текст
Фамилия старосты	Короткий текст	Количество	Короткий текст
Факультет ID	Числовой		

После создания полей закрываем все 3 конструктора, сохраняя все результаты и переходим к созданию связей.



После нажатия на «Схема данных» откроется пустое поле и появится окно, которое запросит какие таблицы нужно поместить в это поле для создание связей. Добавляем все три таблицы и закрываем окно. Далее хватаем поле «Код» у таблицы «Факультет» и тащим его на поле «Факультет ID» таблицы «Группа». Открывается менеджер связей:



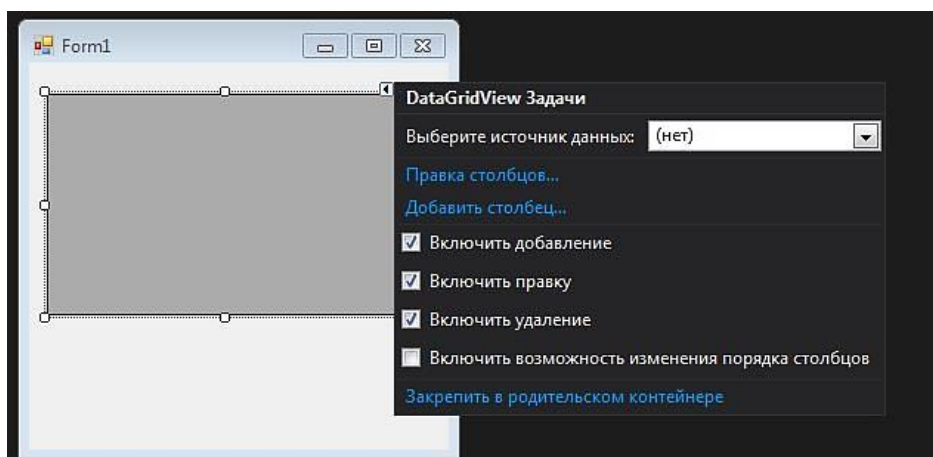
Выставляем все галочки и нажимаем «ОК». После чего проделываем такую же операцию с таблицей «Группа» и «Студенты». Результат показан на рисунке 1. После достигнутого результата сохраняем базу и выходим из Access.

## Шаг 2. Выгрузка базы в форму.

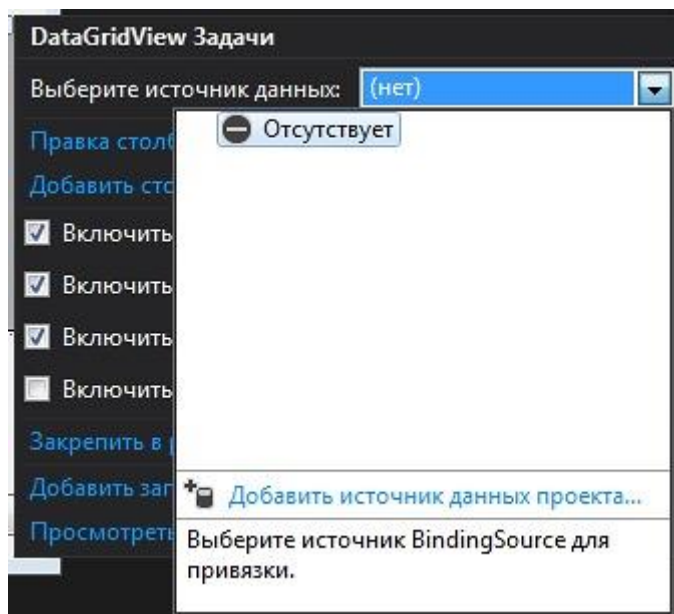
Создаем новый проект VC# и помещаем на форму элемент DataGridView.

Примечание: в данной лабораторной работе описывается выгрузка базы данных в форму с помощью автоматической привязки базы. Однако этот способ имеет множество недостатков, например, очень сложно отвязать базу обратно, также сложно обновить привязанную базу в случае внесения Вами в нее каких-либо изменений. Для того чтобы избежать всех этих недостатков, базу можно привязывать вручную используя заполнение таблиц с помощью драйвера OLEDB. Подробную информацию об этом можно прочитать в MSDN.

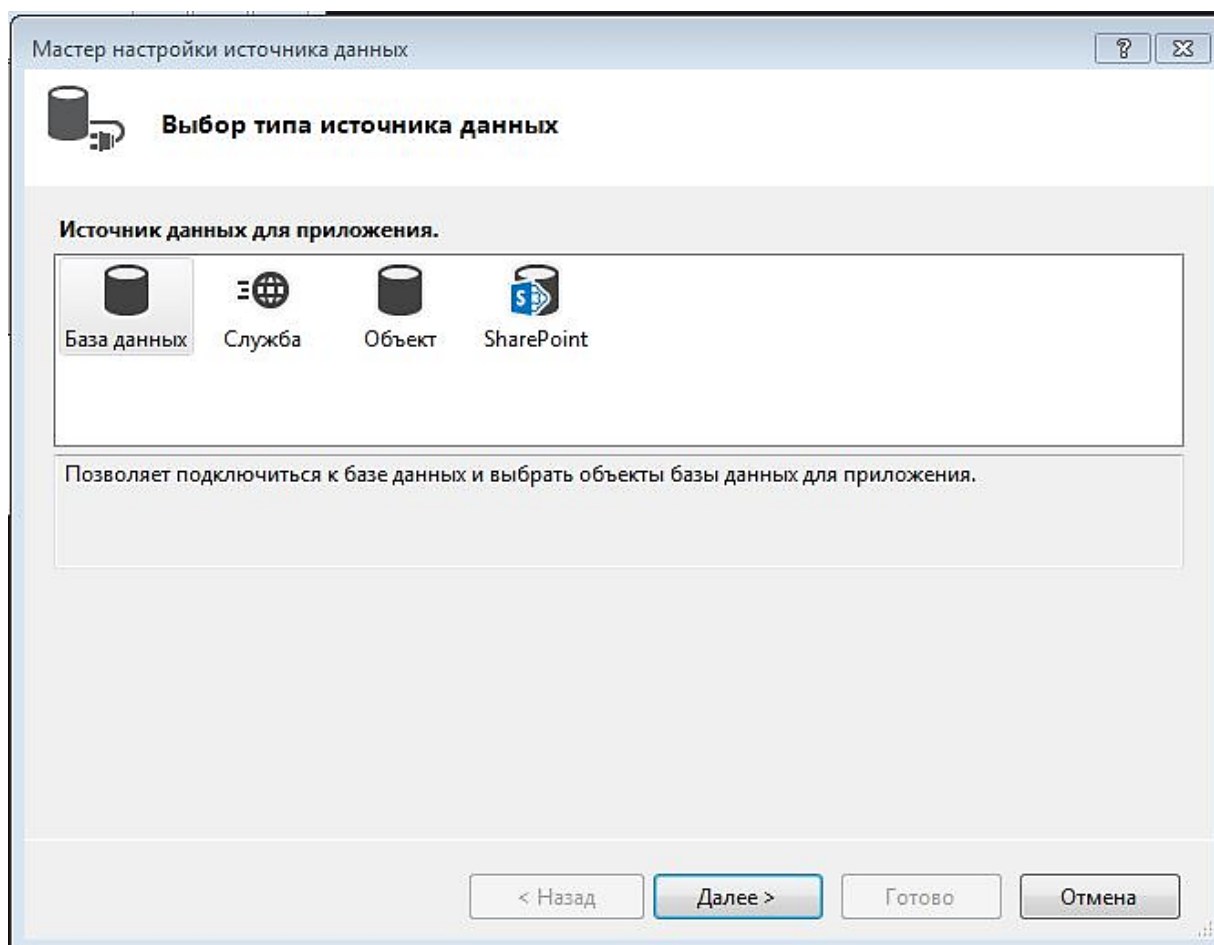
После добавления DataGridView нужно привязать нашу базу данных.



Т.к. источника данных пока еще нет для привязки БД, его нужно добавить. Открываем выпадающее меню пункта «Выберите источник данных» и щелкаем «Добавить источник данных проекта...»

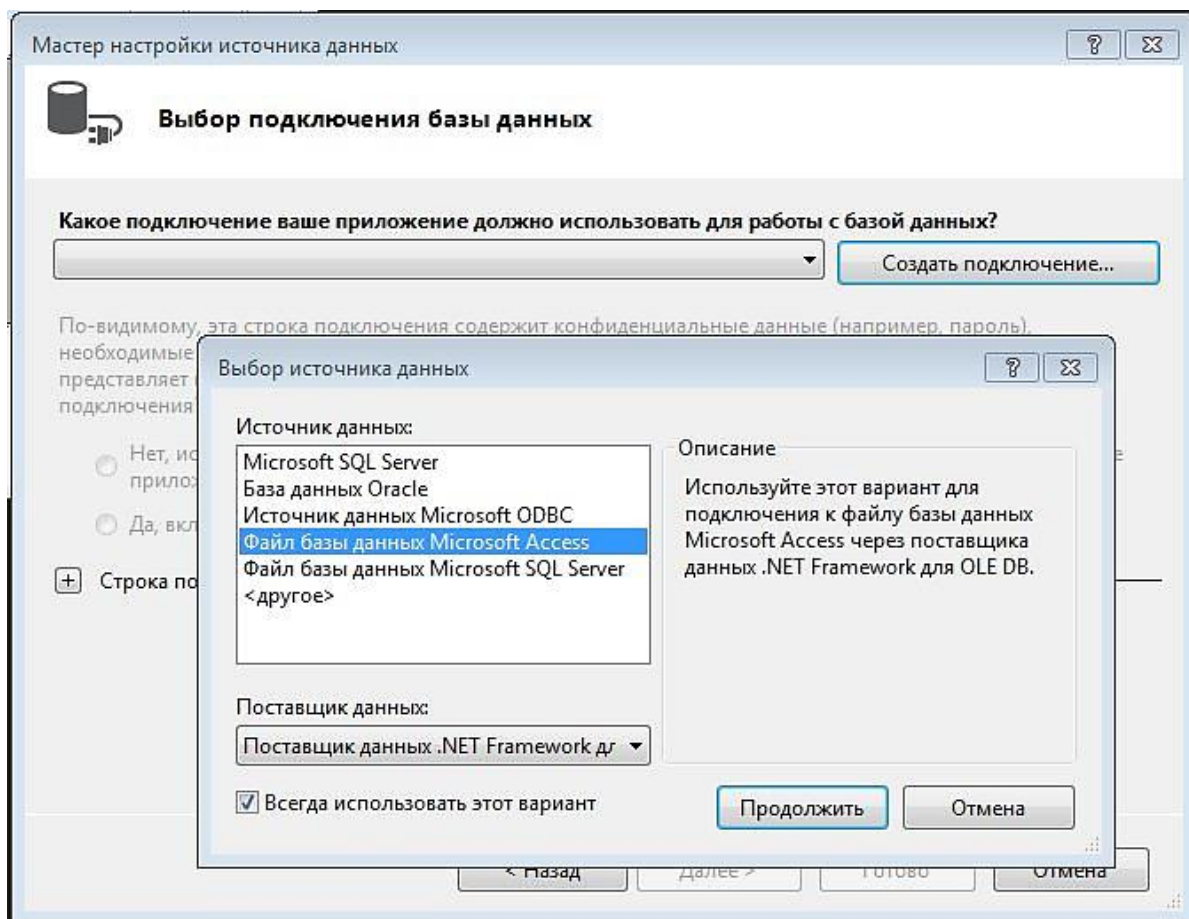


Нажимаем и получаем следующее окно:



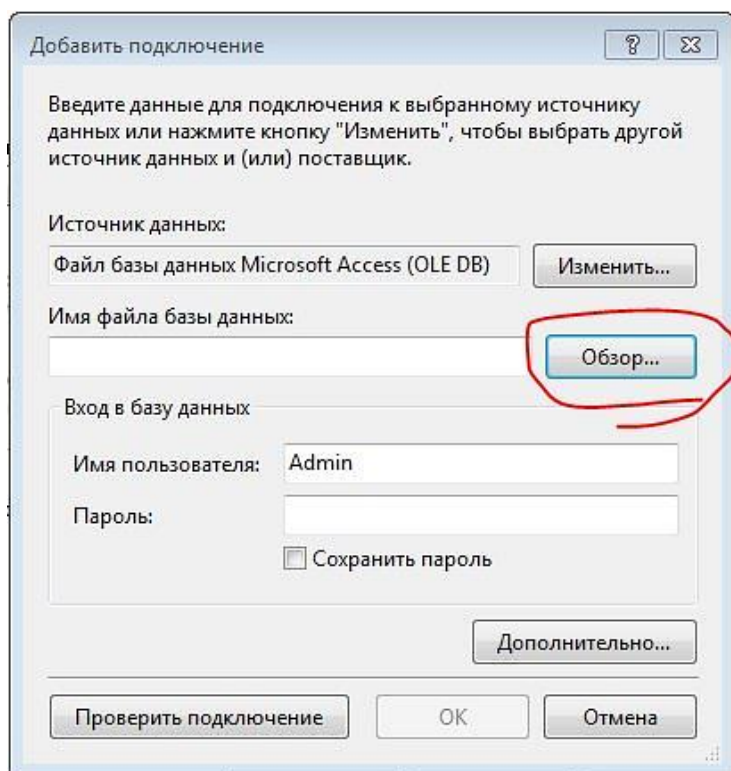
Если нас интересуют базы данных, их и выбираем.

Щелкаем «Далее» пару раз пока не наткнемся на вопрос «Какое подключение ваше приложение должно использовать для работы с базой данных?», где мы смело нажимаем на кнопку «Создать подключение...» и в набросившемся на нас окне с выбором источника данных выбираем «Файл базы данных Microsoft Access». Щелкаем «Продолжить».

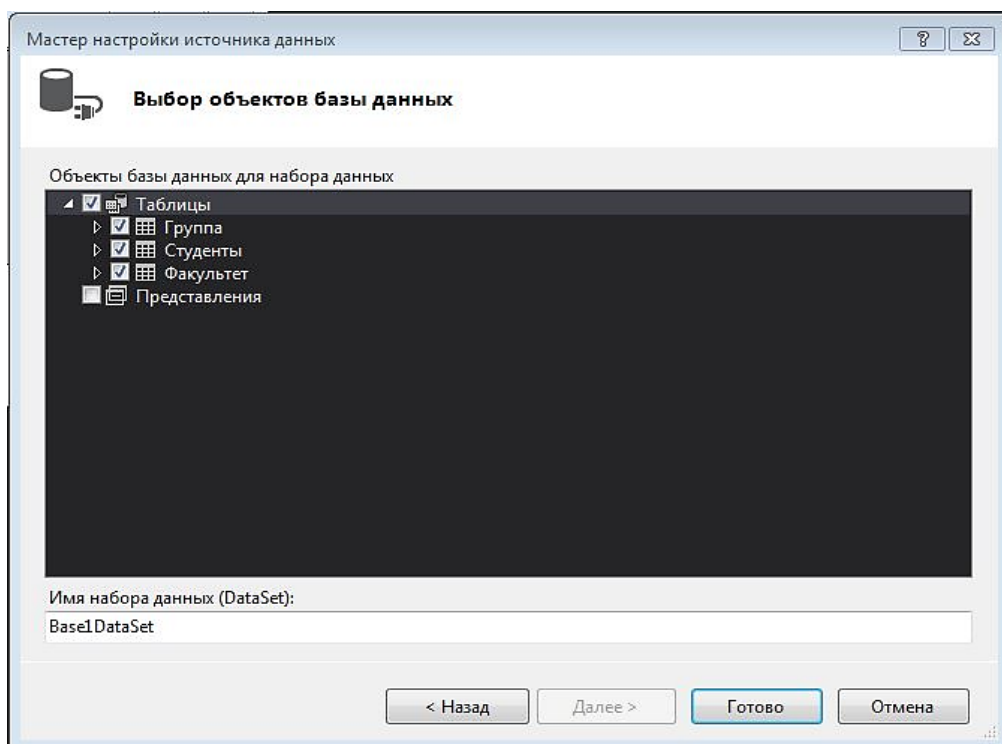




Теперь нужно указать путь на саму базу данных, которую мы выполнили в первом шаге. За это ответственное действие отвечает следующая кнопка:



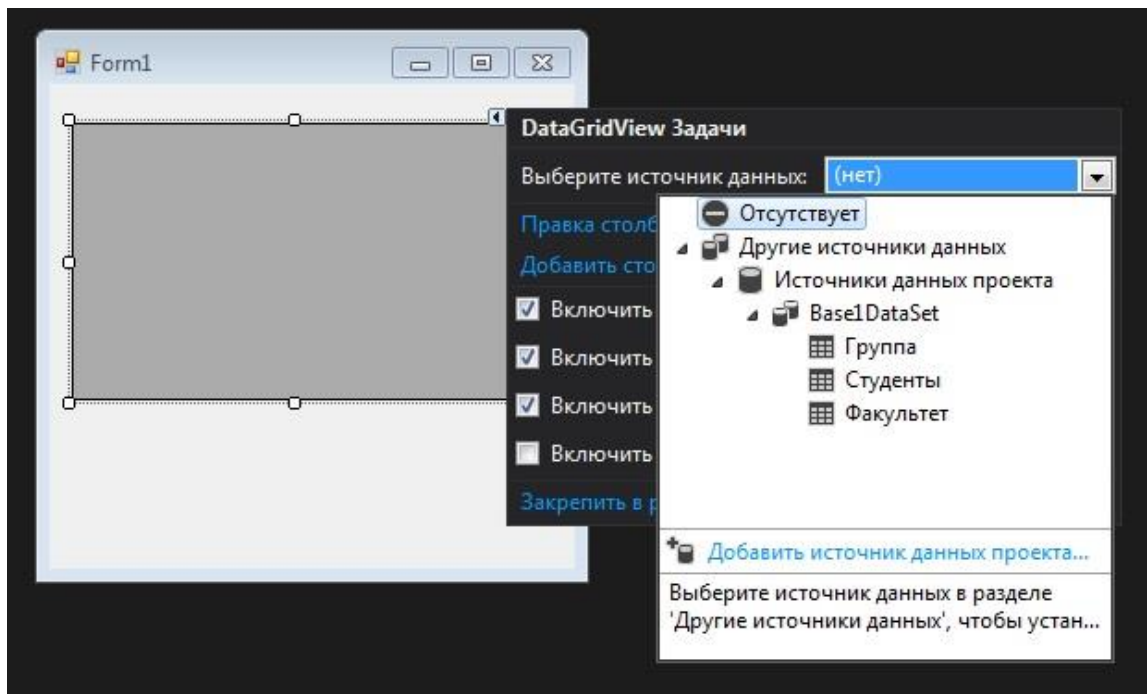
После выбора базы нажимаем «Ок» и возвращаемся в форму мастера настройки источника данных. Там нажимаем «Далее», честно соглашаемся с внезапно появившимся окном о том, что *«выбранное подключение использует локальный файл данных»* и наконец то нам дают доступ к базе:





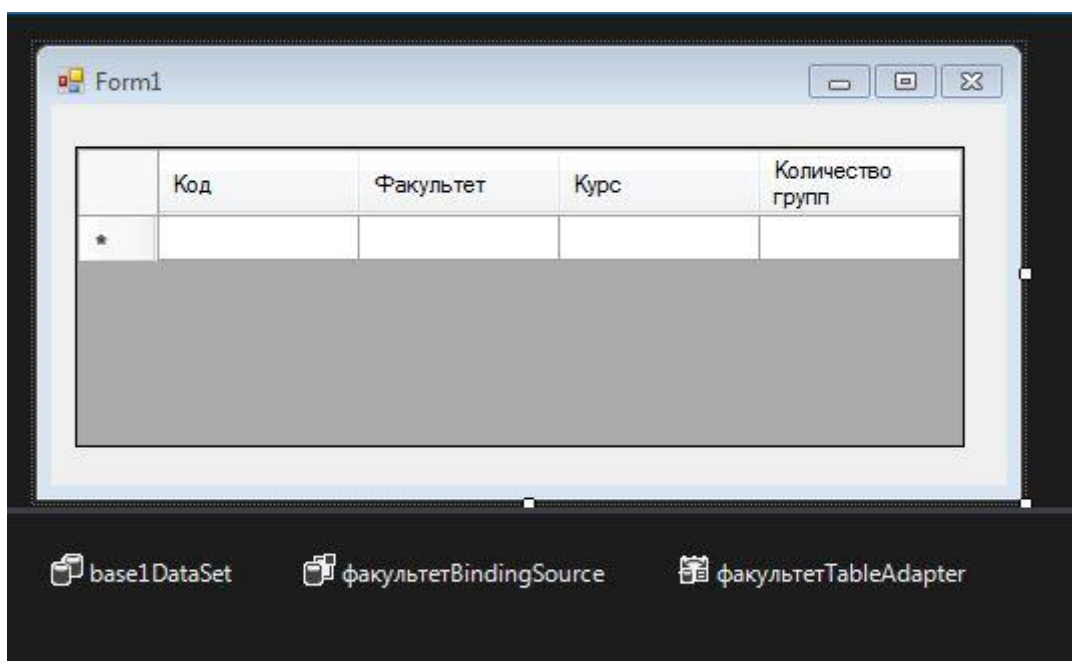
Нас спрашивают какие таблицы из БД мы бы хотели видеть в своей программе. Нас интересуют соответственно все три. Отметив все поля нажимаем «Готово» и возвращаемся в конструктор формы.

Теперь нужно снова нажать на выпадающее окно «Выберите источник данных» и выбрать появившуюся базу:

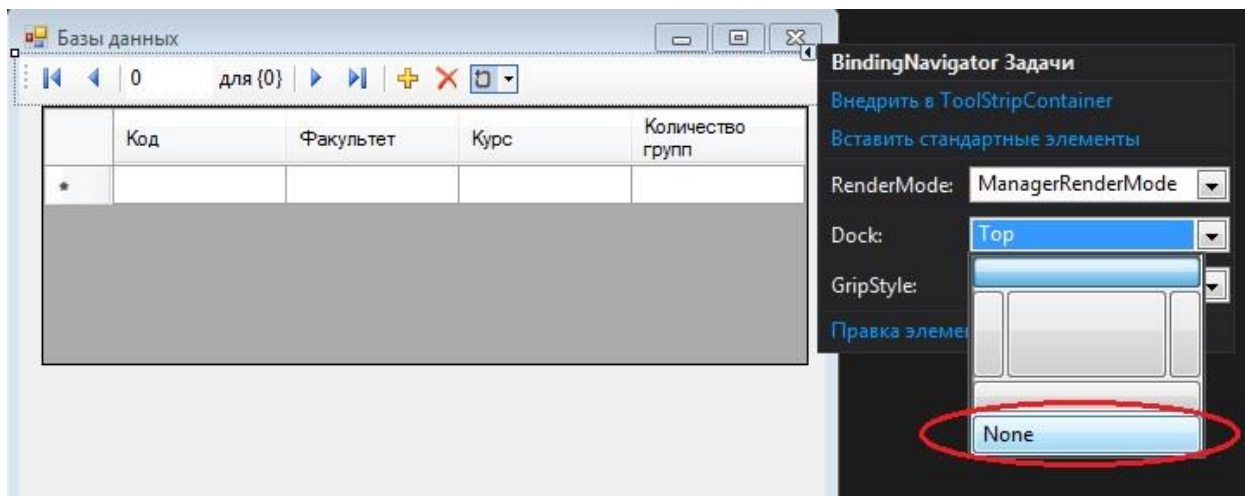


Для первого элемента DataGridView выбираем **Факультет**.

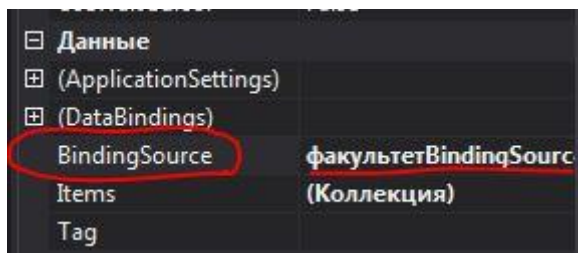
Как мы видим, в первом DataGridView появились заголовки столбцов таблицы факультетов, а на нижнюю панель добавились соответствующие этому DataGridView BindingSource и TableAdapter.



Теперь добавим немного навигации. Помещаем на форму элемент управления **BindingNavigator**. При его перетаскивании на форму он появится вверху, поэтому нужно изменить у него свойство «Dock» на None:

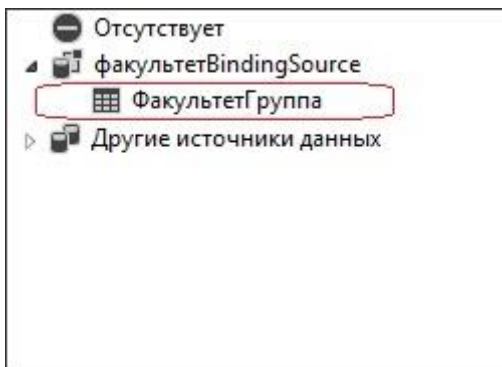


Теперь нужно привязать BindingNavigator к DataGridView. Для этого открываем свойства BindingNavigator-а и меняем пункт BindingSource:

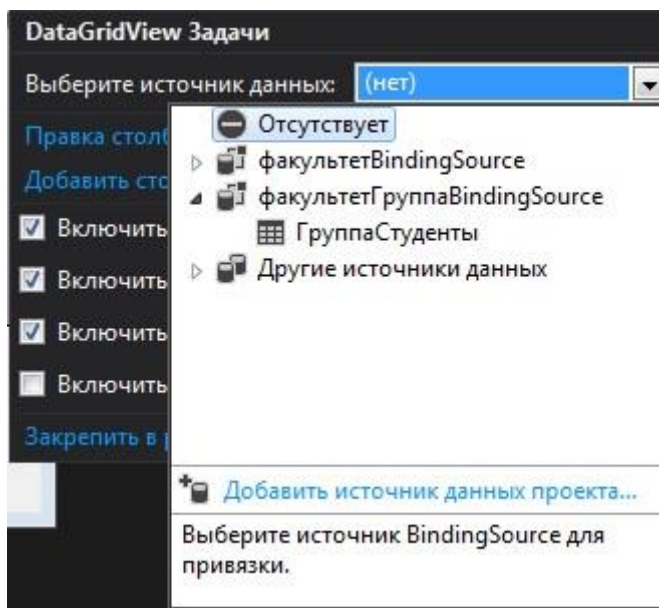


Данный элемент позволяет редактировать отображение в DataGridView, к которому он привязан. НО! Элемент ничего не изменяет в самой базе данных, он работает именно с DGV.

Теперь, аналогично добавляем еще 2 элемента DataGridView и 2 элемента BindingNavigator, делая их по образу и подобию первых DataGridView и BindingNavigator, но с одним отличием для DataGridView. Источник данных для второго и третьего DataGridView будем выбирать иначе. В таблице выбора источника появился уже созданный нами при привязке таблицы Факультеты BindingSource, причем видно, что у него существуют подпункты, открываем эти подпункты и видим там подтаблицу отношения ФакультетГруппа, выбираем ее и получаем во второй DataGridView таблицу, которая отображает результаты для выбранного в первой таблице факультета.



Аналогично для третьего DataGridView выбираем уже созданный в результате предыдущего действия BindingSource и в нем подтаблицу отношения ГруппаСтуденты.



Таким образом, мы получили все нужные нам таблицы с сохраненными между ними связями.

### Шаг 3. Реализация поиска и сортировки.

Добавляем на форму два textBox и одну кнопку. Кнопка будет служить для сортировки, первый textBox для поиска по символам, а второй textBox для поиска целиком.

Прописываем код для сортировки:

```
private void button1_Click(object sender, EventArgs e)
{
    if (dataGridView3.SortOrder == SortOrder.Ascending)
        dataGridView3.Sort(dataGridView3.Columns[1], ListSortDirection.Descending);
    else dataGridView3.Sort(dataGridView3.Columns[1],
        ListSortDirection.Ascending);
}
```

Код для поиска по символам:

```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    for (int i = 0; i < dataGridView3.RowCount - 1; i++)
    {
        string str = dataGridView3.Rows[i].Cells[1].Value.ToString();
        if (str.Contains(textBox1.Text) == true) dataGridView3.Rows[i].Selected
= true;

        else dataGridView3.Rows[i].Selected = false;
        if (textBox1.Text == "") dataGridView3.Rows[i].Selected = false;
    }
}
```

Код для поиска целиком:

```
private void textBox2_TextChanged(object sender, EventArgs e)
{
    for (int i = 0; i < dataGridView3.RowCount - 1; i++)
    {
        string str = dataGridView3.Rows[i].Cells[1].Value.ToString();
        if (str == textBox2.Text) dataGridView3.Rows[i].Selected = true;
        else dataGridView3.Rows[i].Selected = false;
    }
}
```

## Результаты.

Исходная база:

	Код	Факультет	Курс	Количество групп
▶	1	МТС	1	4
	2	ИВТ	2	10
	3	АЭС	3	6
*				

	Код	Название группы	Фамилия старосты	Количество
▶	11	Р-31	Петров	17
	12	Р-32	Иванов	20
	13	Р-33	Сидоров	18
*				

	Код	ФИО	Адрес	Телефон
	111	Филиппов Е. В.	Некрасова, 134	321-31-12
	112	Мазунин В.П.	Кирова, 117	343-32-54
	113	Федоринин А.В.	Выборная, 133	345-32-23
▶*				

Поиск по символам:  Точный поиск:  Сортировать

## Сортировка (первое нажатие на кнопку)

Базы данных

Код	Факультет	Курс	Количество групп
1	МТС	1	4
2	ИВТ	2	10
3	АЭС	3	6

1 для 3

Код	Название группы	Фамилия старосты	Количество
11	P-31	Петров	17
12	P-32	Иванов	20
13	P-33	Сидоров	18

1 для 3

Код	ФИО	Адрес	Телефон
112	Мазунин	Кирова, 117	343-32-54
114	Петренко	Кирова, 118	323-32-23
113	Фидринов	Выборная, 133	345-32-23
111	Филиппов	Некрасова, 134	321-31-12

1 для 4

Поиск по символам      Точный поиск

Сортировать

## Сортировка (второе нажатие на кнопку)

Базы данных

Код	Факультет	Курс	Количество групп
1	МТС	1	4
2	ИВТ	2	10
3	АЭС	3	6

1 для 3

Код	Название группы	Фамилия старосты	Количество
11	P-31	Петров	17
12	P-32	Иванов	20
13	P-33	Сидоров	18

1 для 3

Код	ФИО	Адрес	Телефон
111	Филиппов	Некрасова, 134	321-31-12
113	Фидринов	Выборная, 133	345-32-23
114	Петренко	Кирова, 118	323-32-23
112	Мазунин	Кирова, 117	343-32-54

1 для 4

Поиск по символам      Точный поиск

Сортировать

## Поиск по символам:

Базы данных

Код	Факультет	Курс	Количество групп
1	МТС	1	4
2	ИВТ	2	10
3	АЭС	3	6

1 для 3

Код	Название группы	Фамилия старосты	Количество
11	P-31	Петров	17
12	P-32	Иванов	20
13	P-33	Сидоров	18

1 для 3

Код	ФИО	Адрес	Телефон
111	Филиппов	Некрасова, 134	321-31-12
112	Мазунин	Кирова, 117	343-32-54
113	Фидринов	Выборная, 133	345-32-23
114	Петренко	Кирова, 118	323-32-23

3 для 4

Поиск по символам      Точный поиск

Фи      Сортировать

## Поиск целиком:

Базы данных

Код	Факультет	Курс	Количество групп
1	МТС	1	4
2	ИВТ	2	10
3	АЭС	3	6

1 для 3

Код	Название группы	Фамилия старосты	Количество
11	P-31	Петров	17
12	P-32	Иванов	20
13	P-33	Сидоров	18

1 для 3

Код	ФИО	Адрес	Телефон
111	Филиппов	Некрасова, 134	321-31-12
112	Мазунин	Кирова, 117	343-32-54
113	Фидринов	Выборная, 133	345-32-23
114	Петренко	Кирова, 118	323-32-23

1 для 4

Поиск по символам      Точный поиск

Филиппов      Сортировать