

Лабораторная работа 6.

Лабораторная работа 6. Создание табличных форм

Цель: научиться создавать табличные формы

Перейдём теперь к созданию табличных форм для отображения данных. В данной главе также затрагиваются вопросы фильтрации и сортировки данных, а также реализуется поиск информации в таблице.

Рассмотрим создание табличной формы на примере формы, отображающей таблицу «Студенты». Добавьте в проект новую форму и на неё поместите следующие объекты:

- четыре надписи (Label),
- пять кнопок (Button),
- выпадающий список (ComboBox),
- текстовое поле ввода (TextBox),
- группирующую рамку (GroupBox),
- список (ListBox),
- два переключателя (RadioButton).

Расположите объекты как показано на рисунке 6.1.

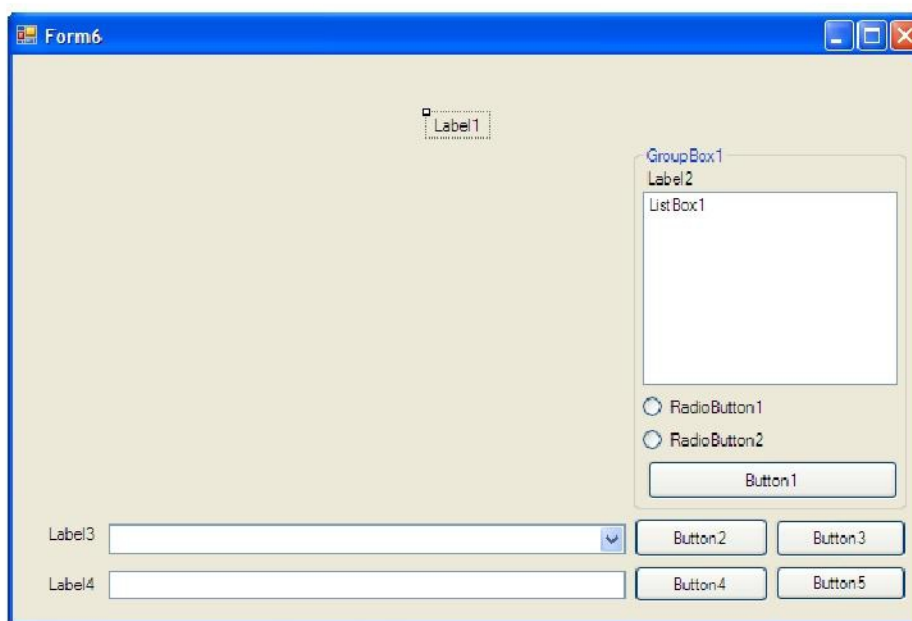


Рис.6.1

Замечание: Для создания объекта группирующая рамка используется кнопка ### на панели объектов (Toolbox), а для создания переключателя - кнопка ##.

Добавим на форму таблицу для отображения данных (DataGridView) из таблицы «Студенты». Для этого на панели «Источники данных» (Data Sources), нажмите кнопку , расположенную справа от таблицы «Студенты». В появившемся списке объектов для отображения всей таблицы выберите «DataGridView» (Рис.6.2).

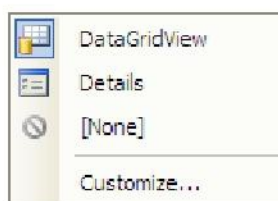


Рис.6.2

Перетащите таблицу «Студенты» из панели «Источники данных» на форму. Форма примет следующий вид (Рис.6.3):

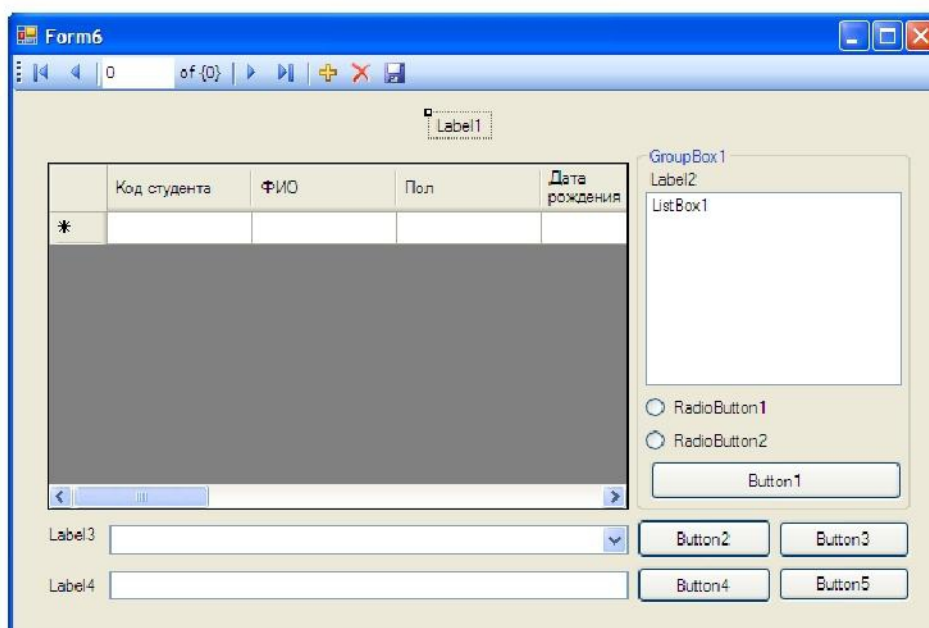


Рис.6.3

Обратите внимание на то, что на форме появилась таблица для отображения данных, подключённая к таблице «Студенты». Также появились объекты связи и панель навигации (Рис.6.4).



Рис.6.4

Теперь перейдём к настройке свойств объектов. Начнём с настройки свойств формы. Задайте свойства формы следующим образом:

- FormBorderStyle (Стиль границы формы): Fixed3D;
- MaximizeBox (Кнопка развёртывания формы во весь экран): False;
- MinimizeBox (Кнопка свёртывания формы на панель задач): False;
- Text (Текст надписи в заголовке формы): Таблица «Студенты» (Табличный вид).

Задайте свойства надписей (Label1, Label2, Label3 и Label4) как:

- AutoSize (Авторамер): False;
- Text (Текст надписи): «Таблица «Студенты» (Табличный вид)», «Поле для сортировки», «ФИО:» и «Критерий» (Соответственно для Label1, Label2, Label3 и Label4).

Для надписи Label1 задайте:

- Font (Шрифт): Microsoft Sans Serif, размер 14;
- ForeColor (Цвет текста): Тёмно синий;

- TextAlign (Выравнивание текста): MiddleCenter.

Задайте надписи на кнопках как: «Сортировать», «Фильтровать», «Показать все», «Найти» и «Заккрыть» (Соответственно для кнопок Button1, Button2, Button3, Button4 и Button5). Для того чтобы нельзя было произвести сортировку не выбрав поля изначально заблокируем кнопку «Сортировать» (Button1).

У группирующей рамки задайте заголовок (Свойство Text) равным «Сортировка». У переключателей (Объекты RadioButton1 и RadioButton2) задайте надписи как «Сортировка по возрастанию» и «Сортировка по убыванию», а у переключателя «Сортировка по возрастанию» (RadioButton1) задайте свойство Checked (Включён) равное True (Истина).

Заполните список (ListBox1) значениями, представленными на рисунке 6.5, а затем нажмите кнопку «Ok».

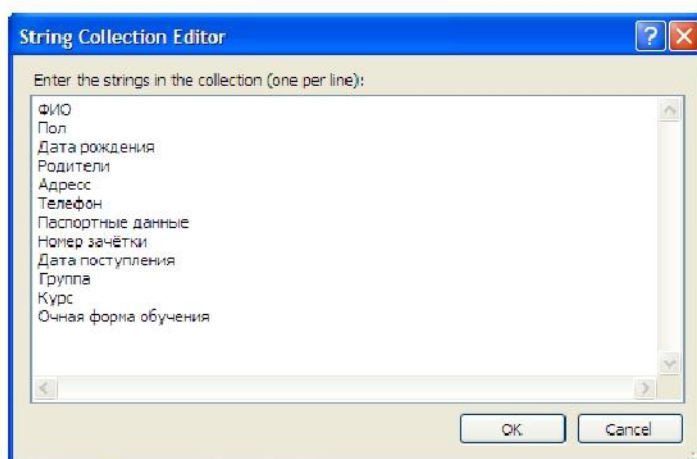



Рис.6.5

Настроим таблицу для отображения данных, удалив из неё поля с кодами. Выделите таблицу на форме и отобразите её меню действий, щёлкнув ЛКМ по кнопке ,  расположенной в верхнем правом углу таблицы. В меню действий выберите пункт «Edit columns...» (Рис.6.6).

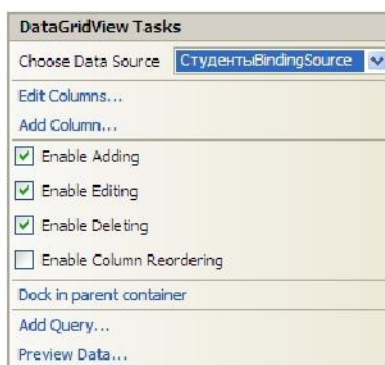


Рис.6.6

Появится окно настройки свойств полей таблицы «Edit Columns» (Рис.6.7).

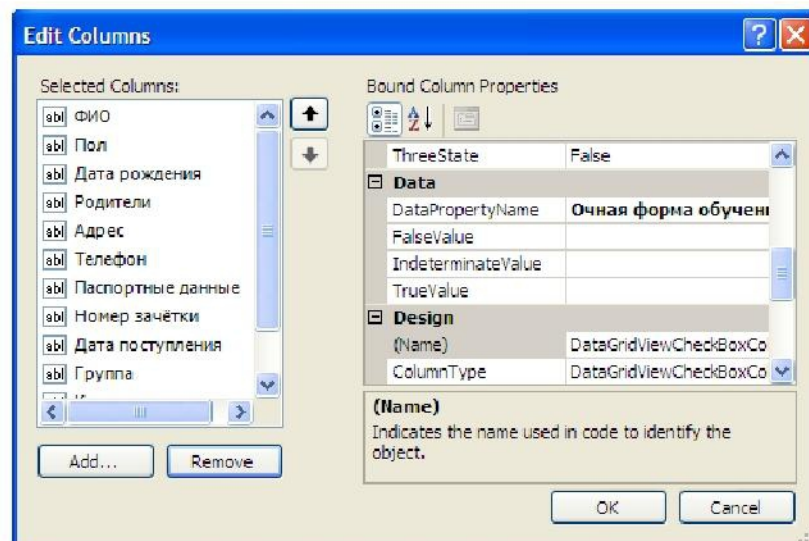


Рис.6.7

В окне «Edit Columns» из списка полей удалите поля «Код студента» и «Код специальности», выделив их и нажав кнопку «Remove» (Удалить). Список полей примет вид показанный на рисунке 6.7. Для закрытия окна редактирования полей, и сохранения изменений нажмите кнопку «Ok».

Настроим заполнение выпадающего списка именами студентов из таблицы студенты. Отобразите меню действий выпадающего списка. Включите опцию «Use Data Bound Items». Установите параметр «Data Source» равным «Other Data Sources\Project Data Sources\StudentsDataSet\Студенты», а параметр «Display Member» равным «ФИО». Остальные параметры оставьте без изменений (Рис.6.8).

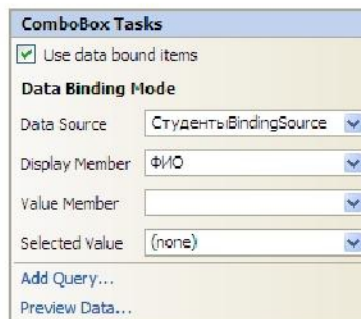


Рис.6.8

Закройте окно действий выпадающего списка. На панели невидимых объектов появится дополнительный объект связи «СтудентыBindingSource1», предназначенный для заполнения выпадающего списка (Рис 6.9).

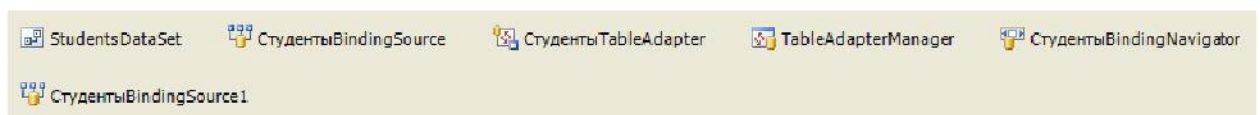


Рис.6.9

После настройки всех вышеперечисленных свойств объектов новая форма примет вид (Рис.6.10):

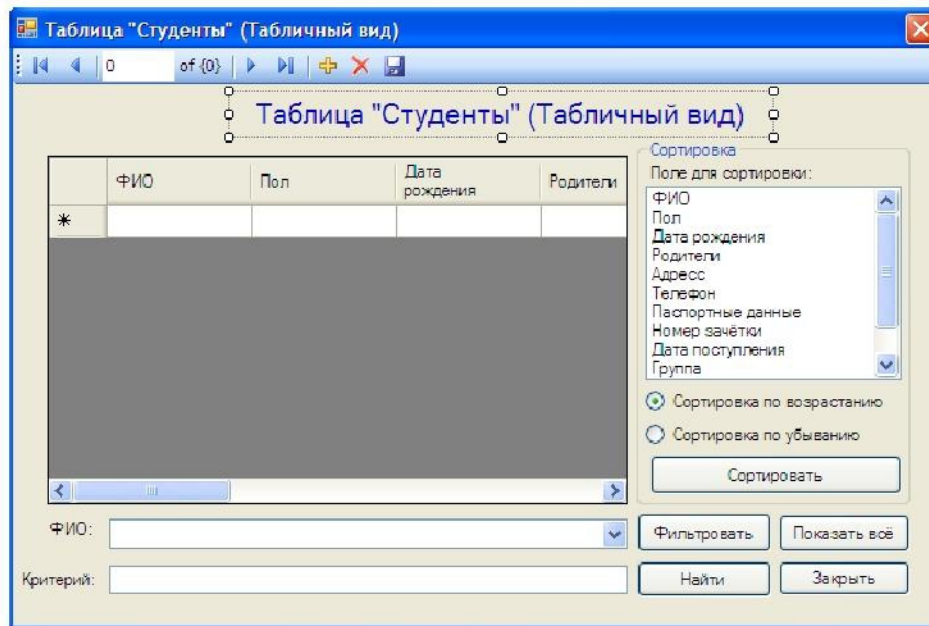


Рис.6.10

На этом мы заканчиваем настройку свойств объектов и переходим к написанию кода обработчиков событий объектов.

Работу с кодом начнём с написания кода для разблокирования кнопки «Сортировать», при выборе пункта списка (ListBox1). Для создания процедуры события дважды щёлкните ЛКМ по списку. Появится процедура обработки события, происходящего при выборе пункта списка (ListBox1_SelectedIndexChanged). В процедуре наберите команду разблокировки кнопки «Сортировать» (Button1): `button1.Enabled = true` (Рис.6.11).

```
private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    button1.Enabled = true;
}
```

Рис.6.11

Теперь перейдём к созданию кода сортирующего нашу таблицу в зависимости от выбранного поля и порядка сортировки при нажатии кнопки «Сортировать». Дважды щёлкните ЛКМ по кнопке «Сортировать». Появится процедура «Button1_Click», выполняемая при щелчке ЛКМ по кнопке. В процедуре наберите код, представленный на рисунке 6.12.

```

private void button1_Click(object sender, EventArgs e)
{
    System.Windows.Forms.DataGridViewColumn Col= dataGridViewTextBoxColumn2;
    switch (listBox1.SelectedIndex)
    {
        case 0: Col = dataGridViewTextBoxColumn2;
            break;
        case 1: ;
            break;
        case 2: Col = dataGridViewTextBoxColumn4;
            break;
        case 3: Col = dataGridViewTextBoxColumn5;
            break;
        case 4: Col = dataGridViewTextBoxColumn6;
            break;
        case 5: Col = dataGridViewTextBoxColumn7;
            break;
        case 6: Col = dataGridViewTextBoxColumn8;
            break;
        case 7: Col = dataGridViewTextBoxColumn9;
            break;
        case 8: Col = dataGridViewTextBoxColumn10;
            break;
        case 9: Col = dataGridViewTextBoxColumn11;
            break;
        case 10: Col = dataGridViewTextBoxColumn12;
            break;
    }
    if (radioButton1.Checked)
        СтудентыDataGridView.Sort(Col, System.ComponentModel.ListSortDirection.Ascending);
    else
        СтудентыDataGridView.Sort(Col, System.ComponentModel.ListSortDirection.Descending);
}

```

Рис.6.12

Рассмотрим код более подробно:

- Переменная Col типа `System.Windows.Forms.DataGridViewColumn` создается для хранения имени выбранного столбца таблицы;
- Затем следует блок `Select Case`, присваивающий в переменную Col имя выбранного столбца таблицы в зависимости от номера выбранного пункта списка (`ListBox1.SelectedIndex`). Если выбран первый пункт списка, то в переменную Col записывается столбец `DataGridViewTextBoxColumn2`, если второй, то - `DataGridViewTextBoxColumn3` и так далее. Хотелось бы отметить тот факт, что нумерация пунктов списка начинается с нуля, а нумерация столбцов с единицы. Первый столбец «ФИО» носит имя `DataGridViewTextBoxColumn2`, так как имя `DataGridViewTextBoxColumn1` имеет столбец заголовков строк;
- Блок `If` выполняет следующую операцию: если включён переключатель «Сортировка по возрастанию» (`RadioButton1`), то отсортировать таблицу по полю заданному в переменной Col по возрастанию (`СтудентыDataGridView.Sort (Col, System.ComponentModel.ListSortDirection.Ascending)`), иначе по убыванию (`СтудентыDataGridView.Sort (Col, System.ComponentModel.ListSortDirection.Descending)`).

Рассмотрим код обработчика события нажатия кнопки «Фильтровать» (`Button2`). Дважды щёлкните по кнопке «Фильтровать» и в процедуре обработки события «`Button2_Click`» наберите код: `СтудентыBindingSource.Filter = "ФИО='" + comboBox1.Text + "'"` (Рис.6.13).

```

private void button2_Click(object sender, EventArgs e)
{
    СтудентыBindingSource.Filter = "ФИО='" + comboBox1.Text + "'";
}

```

Рис.11.13

Замечание: У объекта `СтудентыBindingSource` имеется текстовое свойство `Filter` (Рис.6.13), которое определяет условие фильтрации. Условие фильтрации имеет синтаксис: "`<Имя поля><Оператор>'<Значение>'`". В нашем случае значение поля «ФИО» приравнивается к значению, выбранному в выпадающем списке (`ComboBox1.Text`) (Рис.6.13).

Теперь перейдём к кнопке «Показать всё», отменяющей фильтрацию записей. Дважды щёлкните по вышеперечисленной кнопке. Появится процедура `Button2_Click`. В появившейся процедуре наберите команду `СтудентыBindingSource.Filter = ""` (Рис.6.14).

```
private void button3_Click(object sender, EventArgs e)
{
    СтудентыBindingSource.Filter = "";
}
```

Рис.6.14

Заметим, что если присвоить свойству «`Filter`» значение пустой строки, то его действие будет отменено (Рис.6.14).

Далее рассмотрим реализацию поиска информации в таблице. Дважды щёлкните по кнопке «Найти». В появившейся процедуре обработки нажатия кнопки «`Button4_Click`» наберите следующий код (Рис.6.15).

```
private void button4_Click(object sender, EventArgs e)
{
    for (int i=0; i<СтудентыDataGridView.ColumnCount; i++)
        for (int j=0; j<СтудентыDataGridView.RowCount; j++)
        {
            СтудентыDataGridView[i, j].Style.BackColor=Color.White;
            СтудентыDataGridView[i, j].Style.ForeColor=Color.Black;
        }
    for (int i=0; i<СтудентыDataGridView.ColumnCount; i++)
        for (int j=0; j<СтудентыDataGridView.RowCount-1; j++)
            if (СтудентыDataGridView[i, j].Value.ToString().Contains(textBox1.Text))
            {
                СтудентыDataGridView[i, j].Style.BackColor=Color.AliceBlue;
                СтудентыDataGridView[i, j].Style.ForeColor=Blue;
            }
}
```

Рис.6.15

Рассмотрим более подробно код вышеприведённой процедуры. Данная процедура состоит из двух частей:

- Первый блок `For` перебирает все ячейки таблицы и устанавливает в них белый цвет фона и чёрный цвет текста. То есть, отменяет результаты предыдущего поиска;
- Второй блок `For` перебирает все ячейки таблицы и если они содержат текст, введённый в поле ввода (`TextBox1`), то устанавливает в них голубой цвет фона и синий цвет текста, чем выделяет искомые ячейки.

Наконец рассмотрим код для кнопки «Закрыть». Дважды щёлкните ЛКМ по этой кнопке и в появившейся процедуре «`Button5_Click`» наберите команду «`this.Close()`», закрывающую выше рассматриваемую форму (Рис.6.16).

```
private void button5_Click(object sender, EventArgs e)
{
    this.Close();
}
```

Рис.6.16

В заключение создадим кнопку на ленточной форме, отображающей таблицу «Студенты», для отображения соответствующей табличной формы. Откройте ленточную форму для таблицы «Студенты» (Form4) и поместите на неё новую кнопку, как это показано на рисунке 6.17.

Table "Студенты"

0 of {0}

Table "Студенты"

ФИО:

Пол:

Дата рождения: 23 ноября 2008 г.

Родители:

Адрес:

Телефон: +7 () - -

Паспортные данные: - - - -

Номер зачёта:

Дата поступления: 23 ноября 2008 г.

Группа:

Курс: 0

Код специальности:

Очная форма обучения: ☐

Первая Предыдущая Добавить

Последняя Следующая Удалить

Button8 Сохранить

Рис.6.17

Задайте надпись у новой кнопки (свойство Text), как «Таблица». Форма примет следующий вид (Рис.6.18):

Таблица "Студенты"

Ф.И.О.:

Пол:

Дата рождения: 23 ноября 2008 г.

Родители:

Адрес:

Телефон: +7 () - -

Паспортные данные: - -

Номер зачётки: - -

Дата поступления: 23 ноября 2008 г.

Группа:

Курс: 0

Код специальности:

Очная форма обучения: ☐

Первая Предидущая Добавить

Последняя Следущая Удалить

Таблица Сохранить

Рис.6.18

Подключим к кнопке «Таблица» созданную ранее табличную форму (Form6). Для этого дважды щёлкните ЛКМ по кнопке «Таблица» и в появившейся процедуре «Button8_Click» наберите команду согласно рис. 6.19.

```
private void button8_Click (object sender, EventArgs e)
{
    Form form6=new Form();
    form6.Show();
}
```

Рис.6.19

Теперь проверим работоспособность созданной табличной формы. Запустите проект и на главной кнопочной форме нажмите кнопку «Таблица «Студенты»». На появившейся ленточной форме, отображающей таблицу «Студенты» нажмите кнопку «Таблица». Появится новая табличная форма (Рис.6.20).

Таблица "Студенты" (Табличный вид)

1 of 9

Таблица "Студенты" (Табличный вид)

	ФИО	Пол	Дата рождения	Родители
▶	Иванов А.И.	Мужской	12.12.1983	Отец и
	Петрова И.И.	Женский	01.11.1982	Мать
	Мухин М.А.	Мужской	14.05.1982	Отец
	Сидорова В.К.	Женский	27.09.1981	Нет
	Кожеников А.А.	Мужской	12.04.1981	Мать
	Пальчикова Н.Е.	Женский	02.09.1983	Отец и
	Царегородцев Е...	Мужской	17.02.1980	Отец
	Баранова Г.В.	Женский	09.07.1980	Отец и
	Павлов П.Г.	Мужской	26.07.1979	Нет

Сортировка

Поле для сортировки:

- ФИО
- Пол
- Дата рождения
- Родители
- Адрес
- Телефон
- Паспортные данные
- Номер зачетки
- Дата поступления
- Группа

☒ Сортировка по возрастанию

☐ Сортировка по убыванию

Сортировать

ФИО: Иванов А.И.

Критерий:

Фильтровать Показывать все

Найти Заккрыть

Рис.6.20

Проверьте, как работает поиск, фильтрация и сортировка записей в таблице, нажимая на соответствующие кнопки. После проверки работы формы для возвращения в среду разработки просто закройте все формы.