

АВТОНОМНАЯ МОДЕЛЬ ДОСТУПА К РЕЛЯЦИОННЫМ ДАННЫМ

Автономные типы *ADO.NET* позволяют эмулировать реляционные данные с помощью модели объектов, находящихся в памяти. Кроме простого моделирования табличных данных, состоящих из строк и столбцов, типы из пространства *System.Data* позволяют воспроизводить отношения между таблицами.

Синтаксис адресации полей DataSet и элементов DataGridView

При построении интерфейса могут возникать различные задачи, связанные с записью данных в конкретное поле объекта *DataSet* или конкретную ячейку *DataGridView*. Как адресоваться к этим элементам? Рассмотрим простые (пока без конкретной задачи) случаи передачи данных из текстовых окон (как пример) в конкретные поля *DataSet* и *DataGridView* как на рисунке 1.

Обратите внимание, что данные в *DataTable* представлены как «непрямоугольный» массив:

```
//Вставить конкретное значение в указанную ячейку DataSet
private void button4_Click(object sender, EventArgs e)
{
    ds.Tables[0].Rows[1]["telefon"] = textBox3.Text;
    dataGridView1.DataSource = ds.Tables[0];
}
```

Одним из привлекательных аспектов объекта *DataRow* является поддержка версий, которая позволяет получить различные значения заданного столбца определенной строки. Версии данных в *DataRow* собраны в системном перечислении ***DataRowVersion***.

```
//Вернуть первоначальное значение ячейки в DataSet
private void button5_Click(object sender, EventArgs e)
{
    ds.Tables[0].Rows[1]["telefon"] = ds.Tables[0].Rows[1]["telefon",
        DataRowVersion.Original];
    dataGridView1.DataSource = ds.Tables[0];
}
```

При необходимости обращения к конкретному полю *DataGridView* как на рисунке 2 можно использовать свойства текущей ячейки ***CurrentCellAddress.X*** и ***CurrentCellAddress.Y***:

```
//Вставить конкретное значение в указанную ячейку DataSet
private void button4_Click(object sender, EventArgs e)
{
    //ds.Tables[0].Rows[1]["telefon"] = textBox3.Text;
    //dataGridView1.DataSource = ds.Tables[0];

    //Адрес активной ячейки в DataSet
    dataGridView1[dataGridView1.CurrentCellAddress.X,
        dataGridView1.CurrentCellAddress.Y].Value = textBox3.Text;
}
```

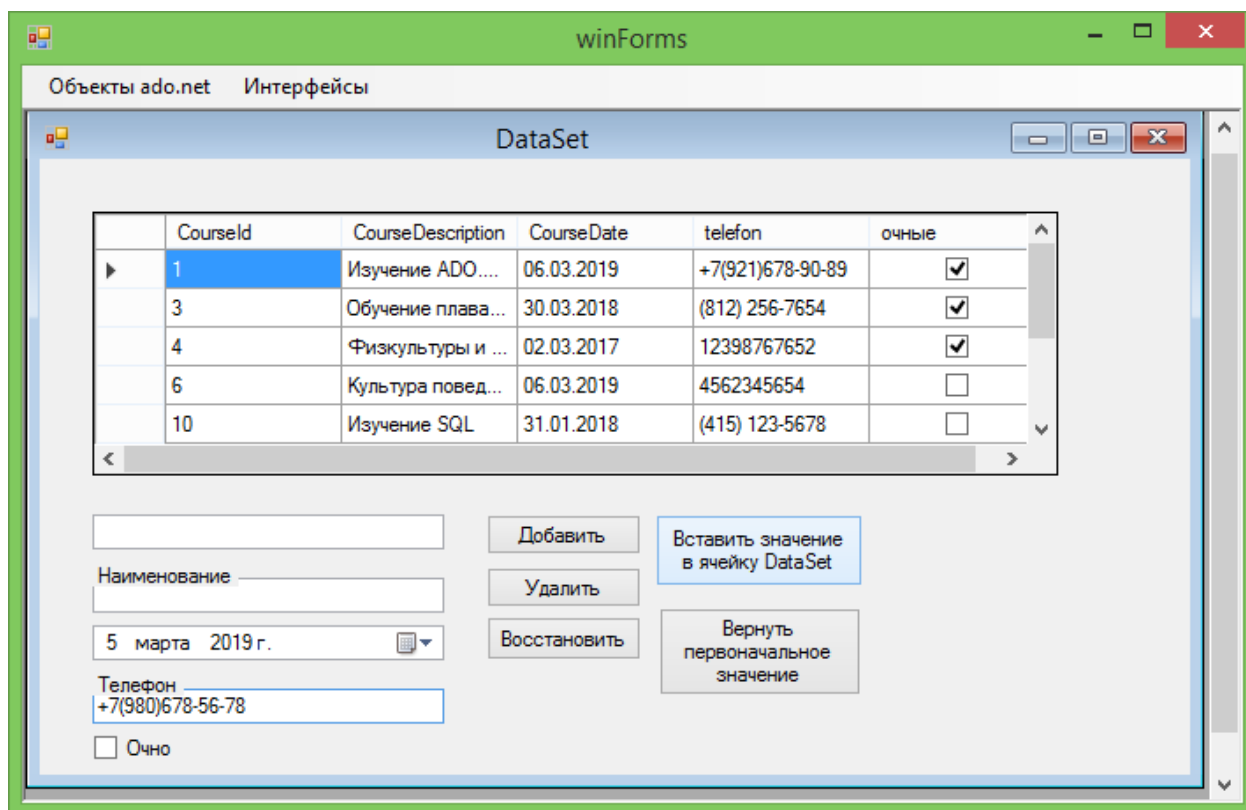


Рисунок 1 – Изменение значения конкретной ячейки в DataSet

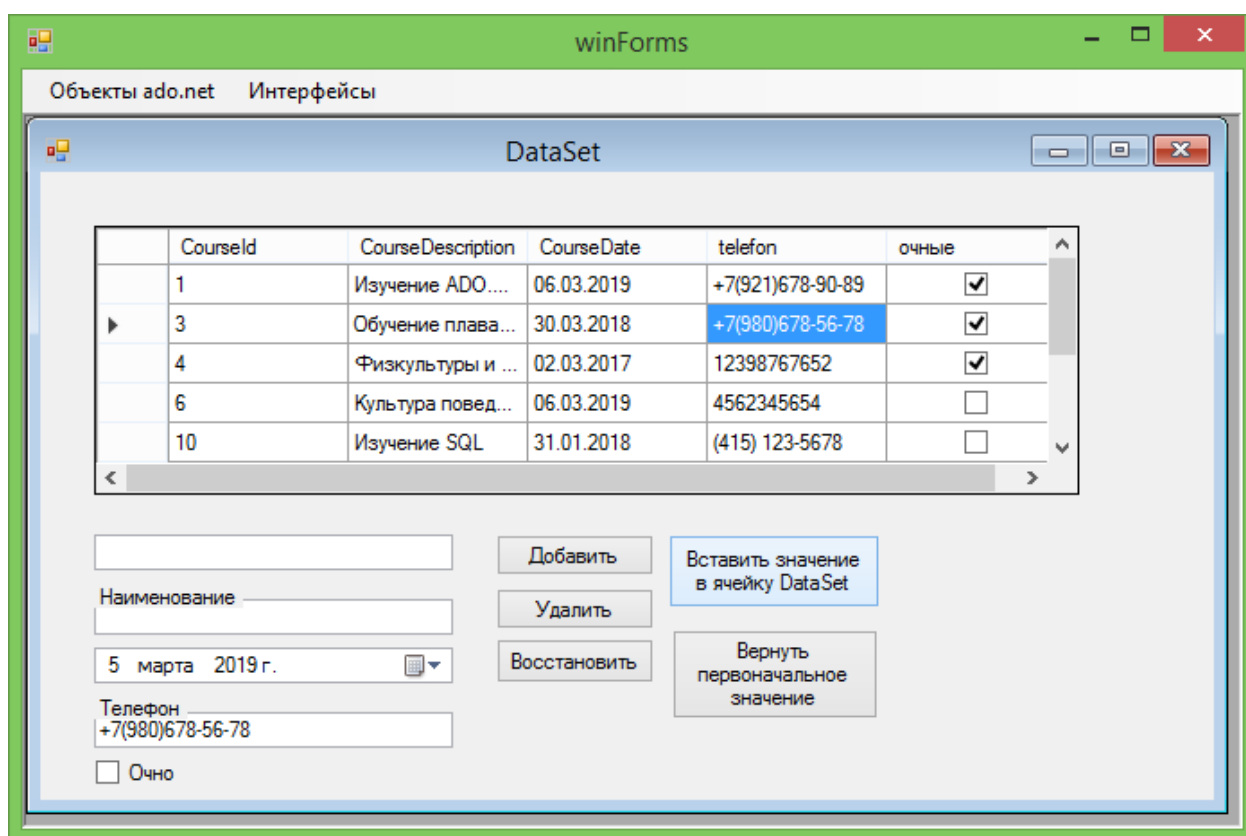


Рисунок 2 – Изменение значения активной ячейки DataSet

Извлечение данных из форм, являющихся справочниками

Для извлечения связанных записей из формы-справочника можно предложить следующую технологию. На рисунке 2 представлена форма «Руководители отделов», через которую пользователь вносит данные. Форма позволяет заполнять таблицу Dep.

	НомерОтдела	НазваниеОтдела	ТабНомерРуководителя
▶	1	Деканат	1
	2	Кафедра матем...	11
	3	Кафедра истории	29
	4	Кафедра инфор...	29
	5	Аспирантура	32
*			

Рисунок 2 - Интерфейс для заполнения таблицы Dep

Для выбора данных о сотрудниках разработана форма-справочник Сотрудники (рисунок 3).

ТабНомер	ФИО
1	Петров А.А.
11	Арбузова Г.Д.
29	Волнушкин Е.О.
32	Широков Р.Д.
37	Корнеев В.В.

Выбрать

Рисунок 3 - Форма-справочник для таблицы Emp

```

//Описание свойства формы для хранения выбранного пользователем значения из DataGridView
//Свойство только для чтения (read-only): есть блок get, но нет блока set
private string sEmp = "";
public string SelectedEmp
{
    get { return this.sEmp; }
}
private void Form5_Load(object sender, EventArgs e)
{
    cnn = new SqlConnection(ConfigurationManager.ConnectionStrings
        ["Employees"].ConnectionString);
    da = new SqlDataAdapter("select EmpId ТабНомер, FIO ФИО from Emp", cnn);
    ds.Clear();
    da.Fill(ds, "Emp");
    dataGridView1.DataSource = ds.Tables["Emp"];
    //Сбросить предыдущий выбор значения свойства
    this.sEmp = "";
    //Назначение кнопке свойства DialogResult позволяет при нажатии кнопки
    //закрывать форму-справочник
    Выбрать.DialogResult = System.Windows.Forms.DialogResult.OK;
}

private void Выбрать_Click(object sender, EventArgs e)
{
    //Запись выбранного пользователем значения из DataGridView в свойство
    SelectedEmp
    this.sEmp = ds.Tables["Emp"].Rows[this.BindingContext[ds.Tables
        [0]].Position]["ТабНомер"].ToString();
}

```

В форме Руководители отделов при щелчке по ячейке столбца Руководитель должна открываться форма Сотрудники:

```

//При щелчке по соответствующей ячейке экземпляр формы-справочника активизируется в модальном режиме
private void dataGridView1_CellClick(object sender, DataGridViewCellEventArgs e)
{
    if (dataGridView1.CurrentCell.ColumnIndex == dataGridView1.Columns["DepartmentHeadId"].DisplayIndex)
    {
        f.ShowDialog();
        if (f.DialogResult == DialogResult.OK)
        {
            dataGridView1[dataGridView1.CurrentCellAddress.X, dataGridView1.CurrentCellAddress.Y].Value = f.SelectedEmp;
        }
    }
}

```

Задание для самостоятельной работы в windows-форме

- 1) Реализуйте изменение данных в конкретной ячейке *DataSet* и затем возвращение к первоначальной версии данных.
- 2) Реализуйте следующие действия: пользователь вводит в элемент формы данные, двойной щелчок по конкретной ячейке *DataGridView* записывает данные в указанное поле.
- 3) Реализуйте интерфейс пользователя для работы с двумя таблицами, между которыми имеется отношение M:1, по технологии, рассмотренной в практикуме.