

Практическая работа №10

Редактирование данных в ASP.NET

Цель – изучение основных приемов обработки данных в веб-приложениях; формирование и отрисовка запросов с модификацией данных; декларативный и программный методы редактирования данных с помощью серверных элементов управления (DataList, GridView, FormView); разработка приложения, обеспечивающего вывод, редактирование и выборку данных.

Информационные материалы:

Курс лекций «Компьютерные сети и Интернет-технологии».

Эспозито Д. Microsoft ASP.NET 2.0 Базовый курс. Мастер-класс /Пер.с англ. – М.: Изд-во «Русская редакция»; СПб.: Питер, 2007. – 668 с.

Эспозито Д. Microsoft ASP.NET 2.0 Углубленное изучение /Пер.с англ. – М.: Изд-во «Русская редакция»; СПб.: Питер, 2007. – 592 с.

Общие сведения о серверных элементах управления ASP.NET с привязкой к данным, [Электронный ресурс]. – Режим доступа:

[http://msdn.microsoft.com/ru-ru/library/ms228214\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/ms228214(v=vs.100).aspx)

Общие сведения о серверном веб-элементе GridView, [Электронный ресурс]. – Режим доступа:

[http://msdn.microsoft.com/ru-ru/library/2s019wc0\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/2s019wc0(v=vs.100).aspx)

Общие сведения о серверном веб-элементе FormView, [Электронный ресурс]. – Режим доступа:

[http://msdn.microsoft.com/ru-ru/library/ms227992\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/ms227992(v=vs.100).aspx)

Программное обеспечение:

Microsoft Visual Studio 2005, SQL Server Express, Internet Explorer.

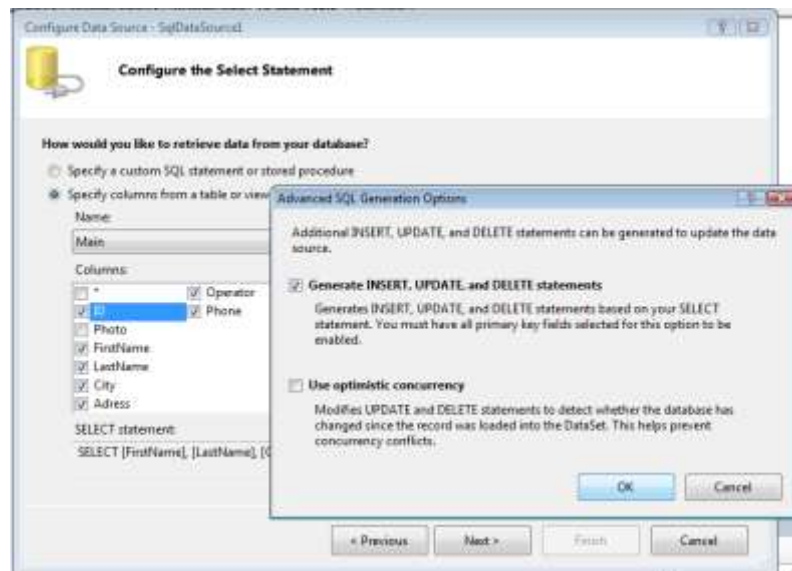
Практикум

Корректировка данных с помощью декларативных функций элементов управления GridView (DataGrid) и FormView

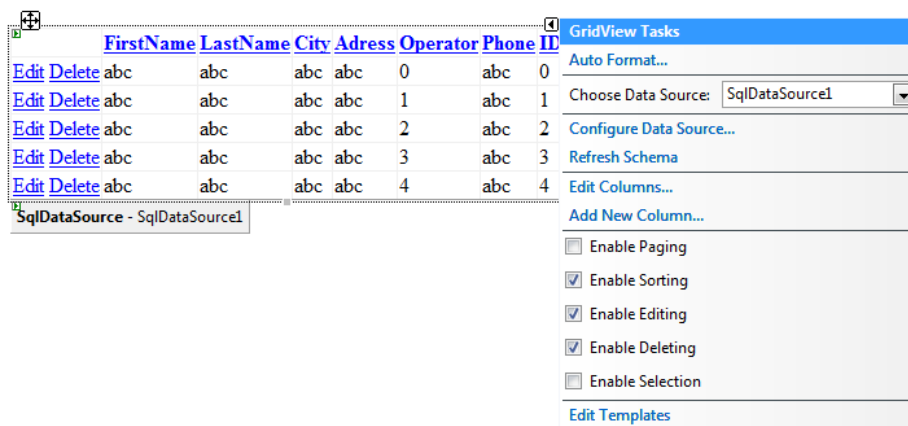
1. Откройте свой проект, активизируйте соединение с базой данных Students и создайте в проекте новую aspx-страницу (Default3.aspx).

2. Разместите серверный элемент управления данными GridView, установите связь с таблицей Main, выбрав указанные на рисунке поля.

При этом добавьте возможность вставки, обновления и удаления данных (кнопка Advanced...)

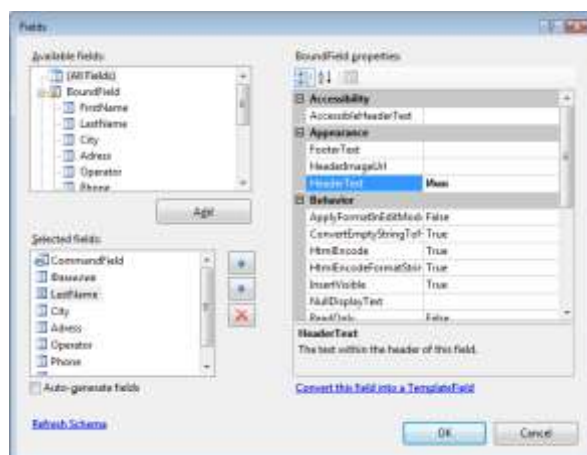


3. В шаблоне элемента GridView задайте декларативные функции сортировки, редактирования и удаления элементов таблицы.



4. Используя команду EditColumns, измените наименования столбцов таблицы (Фамилия, Имя, Город, Улица, Оператор и Телефон).

Для поля ID установите для свойства Visible значение false, а поле Operator ID используется только для чтения



5. Изучите раздел Практического руководства "Динамическая установка ширины столбцов элемента GridView" и реализуйте обработчик событий RowDataBound для базы данных Students.

Режим доступа к электронному ресурсу:

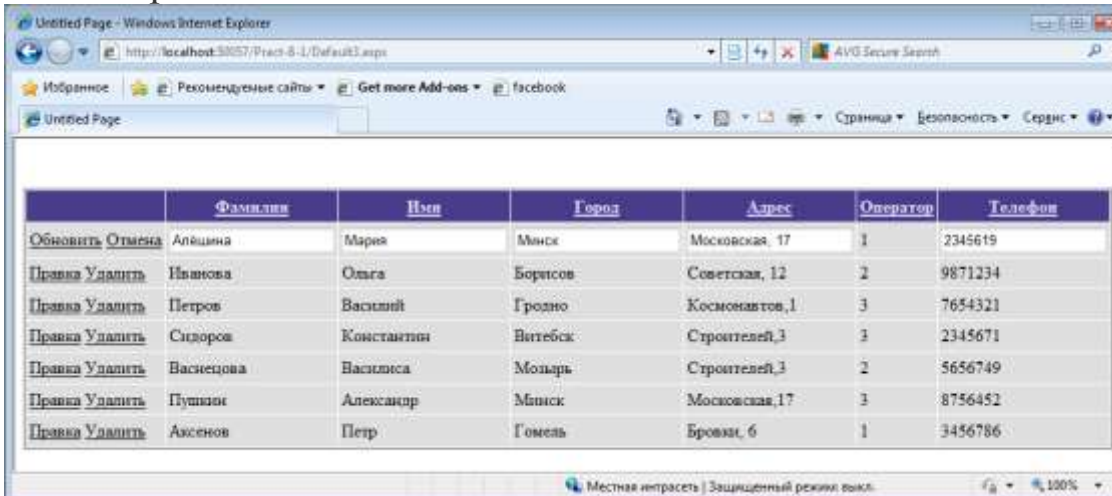
[http://msdn.microsoft.com/ru-ru/library/ms178296\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/ms178296(v=vs.100).aspx)

6. Выполните компиляцию (Ctrl+F5)

Основной режим:

	Фамилия	Имя	Город	Адрес	Оператор	Телефон
Правка Удалить	Аксенов	Петр	Гомель	Бровки, 6	1	3456786
Правка Удалить	Алёшина	Мария	Минск	Московская, 17	1	2345619
Правка Удалить	Васнецова	Василиса	Мозырь	Строителей,3	2	5656749
Правка Удалить	Иванова	Ольга	Борисов	Советская, 12	2	9871234
Правка Удалить	Петров	Василий	Гродно	Космонавтов,1	3	7654321
Правка Удалить	Пушкин	Александр	Минск	Московская,17	3	8756452
Правка Удалить	Сидоров	Константин	Витебск	Строителей,3	3	2345671

Режим правки:



Обновить Отмена	Алёшина	Мария	Минск	Московская, 17	1	2345619
Правка Удалить	Иванова	Ольга	Борисов	Советская, 12	2	9871234
Правка Удалить	Петров	Василий	Гродно	Космонавтов,1	3	7654321
Правка Удалить	Сидоров	Константин	Витебск	Строителей,3	3	2345671
Правка Удалить	Васнецова	Василиса	Мозырь	Строителей,3	2	5656749
Правка Удалить	Пушкин	Александр	Минск	Московская,17	3	8756452
Правка Удалить	Аксенов	Петр	Гомель	Бровки, 6	1	3456786

Протестируйте работоспособность элемента:

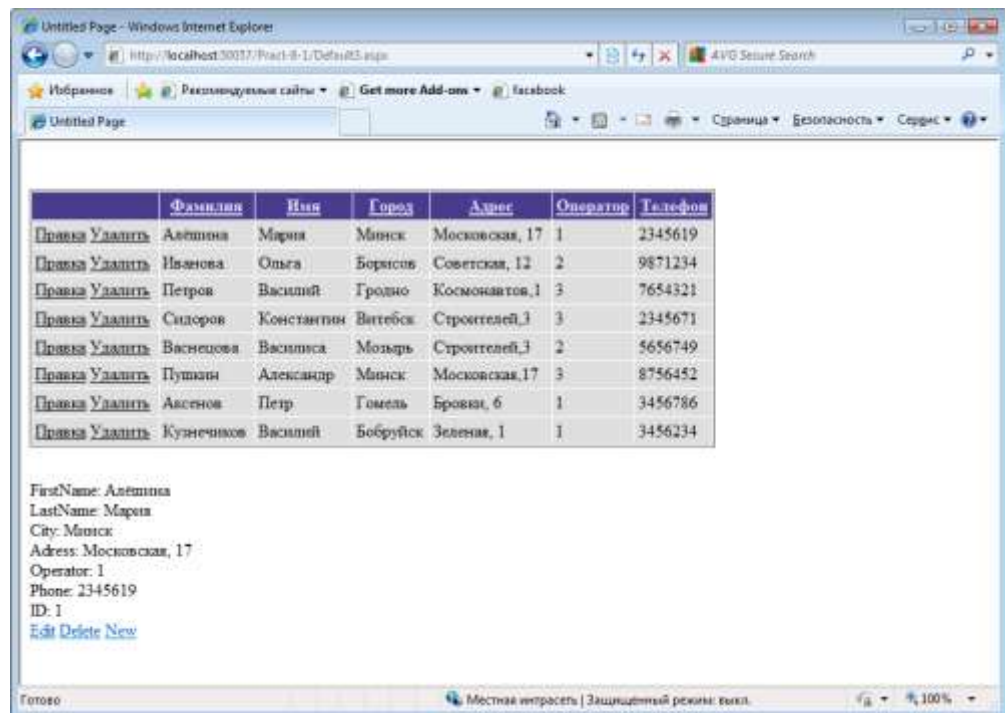
- сортировка по фамилии (в алфавитном порядке);
- сортировка по номеру оператора (по возрастанию);
- редактирование элементов (например, измените адрес или телефон абонента).

В отчете представляются скриншоты, демонстрирующие динамику выполнения этих операций.

Какие компоненты и классы ASP.NET позволяют выполнять функции корректировки значений в базах данных в декларативной форме?

7. Изучите самостоятельно принцип действия элемента FormView, разработайте собственный дизайн шаблона ItemTemplate и добавьте 3 новых абонента, используя декларативный метод New. Реализуйте добавление новых записей в базу данных после выполнения события «Добавить новую запись».

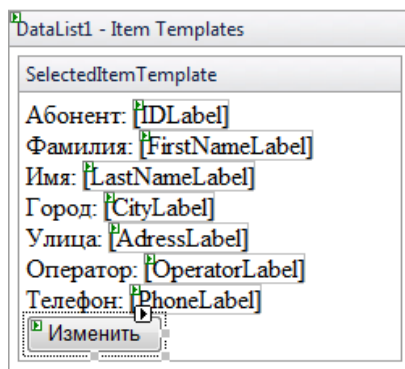
На рисунке элемент GridView предоставляет сведения с учетом добавленных данных.



Каким образом Вы установили возможность перехода к другой записи в элементе FormView?

Редактирование записей с помощью программного кода

Для решения воспользуемся разработкой элемента DataList, в котором уже реализовано свойство Select. В таблице Main добавьте возможность нулевых значений для элементов.



1. В источнике данных DataList устанавливаем возможность их редактирования.

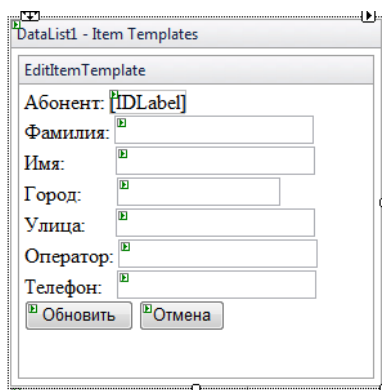
2. В шаблоне SelectedItemTemplate добавляем кнопку Button-«Изменить», для которой устанавливаем для свойства CommandName значение edit.

3. В окне Properties элемента DataList, выбираем закладку Events  и двойным щелчком на значении EditCommand создаем событие редактирования.

Программный код использует свойство Item выбранного элемента.

```
protected void DataList1_EditCommand1(object source, DataListCommandEventArgs e)
{
    DataList1.EditItemIndex = e.Item.ItemIndex;
    DataList1.DataBind();
}
```

4. Создаем содержимое шаблона EditItemTemplate, которое будет отрисовываться в результате выполнения EditCommand.



Как видно на рисунке, в шаблон добавлены элементы:

TextBox, в которых привязка к конкретным полям таблицы реализуется декларативным методом Eval (используем SmartTag);

Button-«Обновить» и Button-«Отмена», которым аналогично кнопке «Изменить» устанавливаются свойства CommandName соответственно update и cancel.

5. С помощью закладки Events создаем CancelCommand, в котором значение Index выбранного пользователем элемента заменяется на -1 и осуществляется отрисовка источника данных.

```
protected void DataList1_CancelCommand1(object source, DataListCommandEventArgs e)
{
    DataList1.EditItemIndex = -1;
    DataList1.DataBind();
}
```

6. Обработку события update рассмотрим на примере поля City:

```
protected void DataList1_UpdateCommand1(object source, DataListCommandEventArgs e)
{
    //Получение ID выбранного элемента
    String ID = DataList1.DataKeys[e.Item.ItemIndex].ToString();

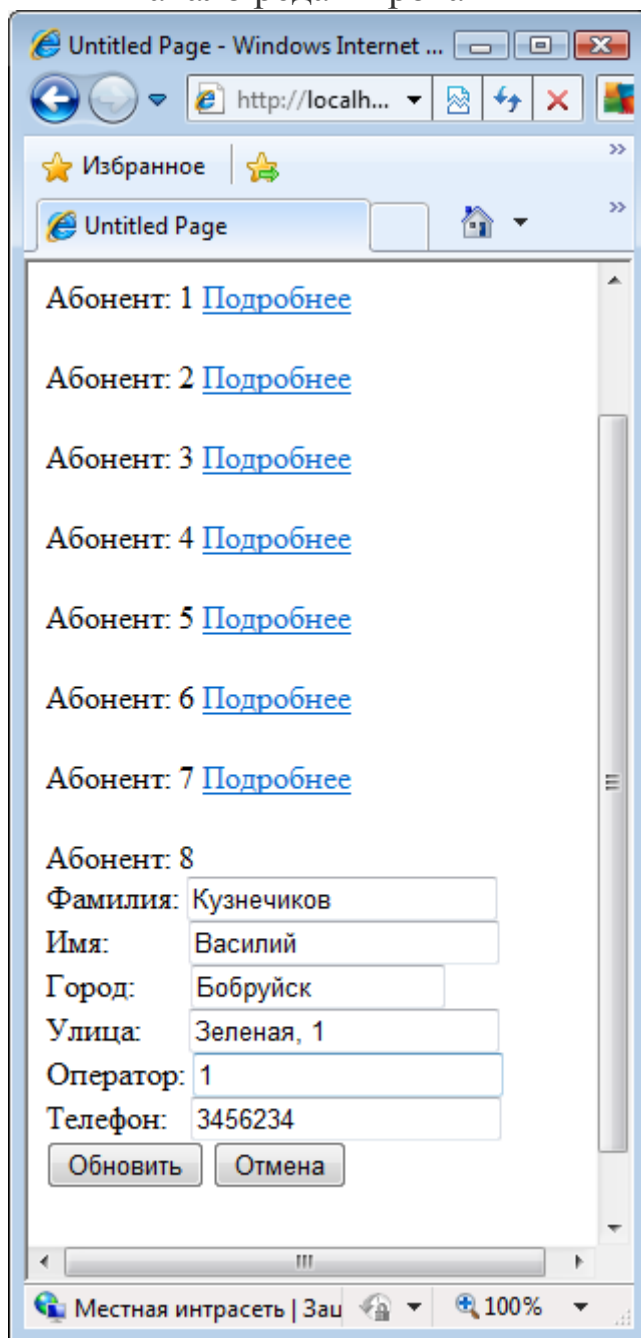
    //Получение измененного текста выбранной записи
    String City = ((TextBox)e.Item.FindControl("TextBox1")).Text;

    //Передача значений источнику данных
    SqlDataSource1.UpdateParameters["ID"].DefaultValue = ID;
    SqlDataSource1.UpdateParameters["City"].DefaultValue = City;

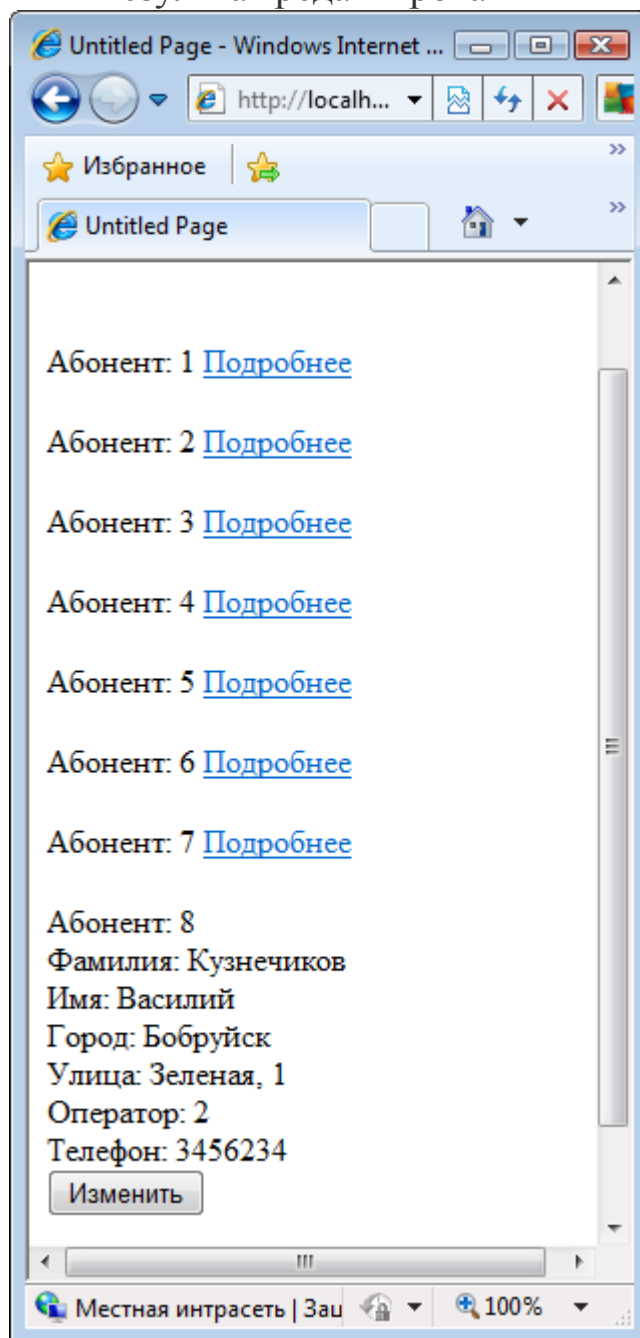
    //Отрисовка измененных значений
    DataList1.EditItemIndex = -1;
    DataList1.DataBind();
}
```


7. В результате пользователь может, нажав кнопку «Подробно», редактировать значения базы данных

Начало редактирования



Результат редактирования



8. Самостоятельно добавьте кнопку Delete и обработчик события DeleteCommand.