Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования «Уральский федеральный университет   
имени первого Президента России Б. Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ

**Отчёт по Лабораторной работе №5**

по дисциплине «Средства управления информационными ресурсами АС»

**Тема:** Использование справочников в формах диалога с БД

Научный руководитель Парфенов Ю. П.

Курс, группа РИЗ-300016у

Студент Кулаков М. И.

Екатеринбург, 2022

# СОДЕРЖАНИЕ

1 Постановка задачи 3

2 Образы экранных форм в конструкторе и исполняемой программе с пояснениями элементов управления 3

3 Добавленный программный код с пояснениями 4

4 Подробное описание впервые использованных свойств и методов 7

# 1 Постановка задачи

1. На пятой вкладке «Лабораторная работа № 5» постройте диалог, демонстрирующий работу с главной таблицей БД задания, в которой значение справочного поля выбирается из списка, содержащего значения из справочной таблицы.
2. Проверьте возможность управления данными в полученной форме.
3. Проверьте, как изменение данных справочной таблицы отражается на выборе справочных значений в ***DataGridView***.
4. Изучите работу со справочными значениями в отдельных полях. Добавьте в форму элемент ***ComboBox*** и настройте его свойства для выбора значений из справочной таблицы (или запроса):
   * в разделе **DataBinding** установите в свойстве **SelectedValue** связь с полем кода источника данных (***BindingSource***) основной таблицы;
   * в свойстве **DataSource** задайте источник (***BindingSource***) справочника данных;
   * в свойстве **DisplayMember** выберите поле для отображения;
   * в свойстве **ValueMember** выберите поле кода из справочника.

# 2 Образы экранных форм в конструкторе и исполняемой программе с пояснениями элементов управления

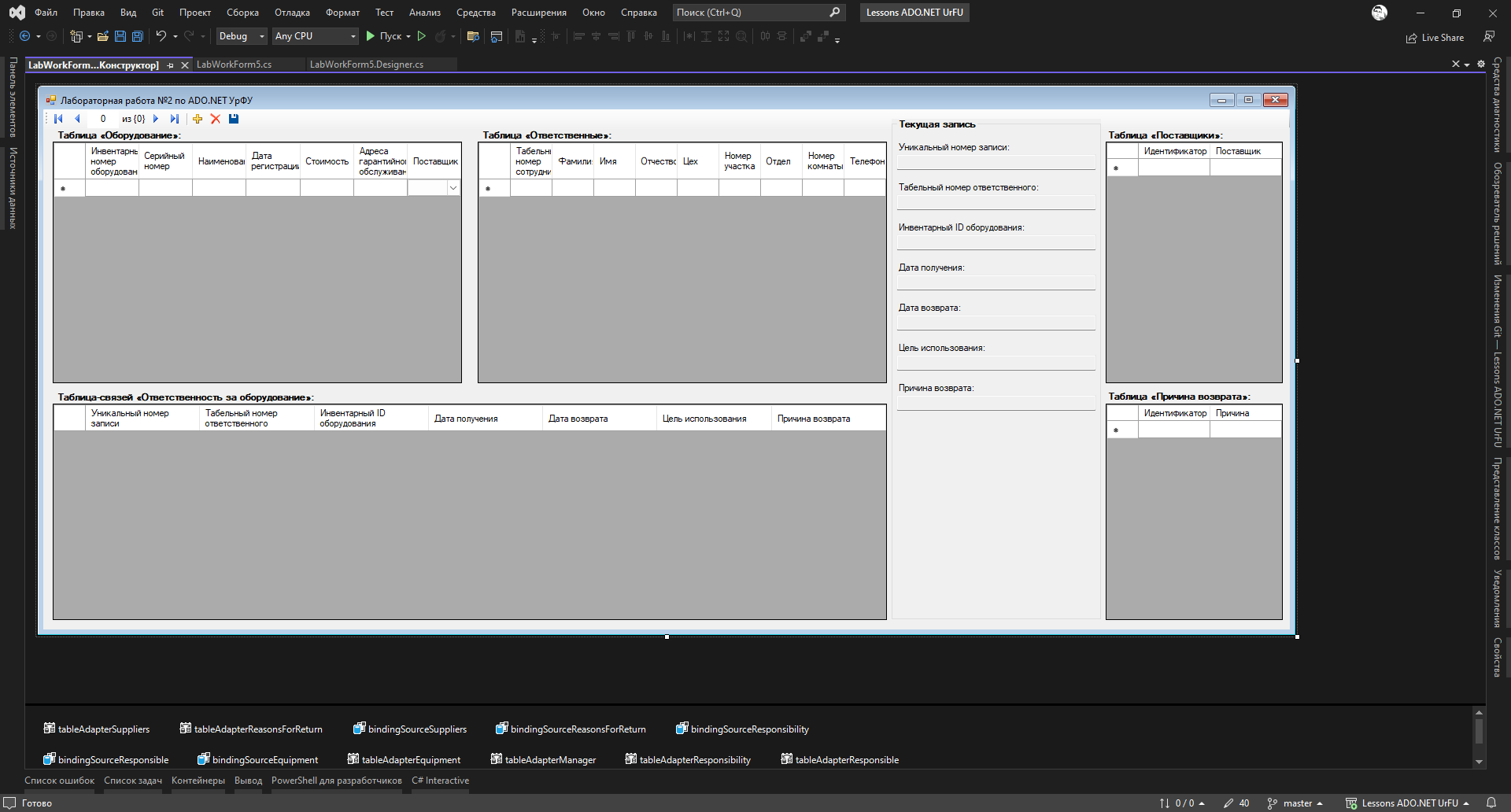


Рисунок №1 – Экранная форма в конструкторе

Изображение выглядит как стол

Автоматически созданное описание

Рисунок №2 – Исполняемая программа

Лабораторная работа № 5 была сделана на основе лабораторной работы №2. По сравнению с ней было добавлено 2 табличных представления ***DataGridView*** (для таблиц справочников), 2 текстовые метки ***Label***, а также были изменены свойства у двух столбцов таблиц «Ответственные» и «Ответственность за оборудование».

Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание

Рисунок №3 – Окно мастера настройки источников данных

Все необходимые таблицы были добавлены в **dataSetLessonsUrfu** ещё при выполнении лабораторной работы №1.

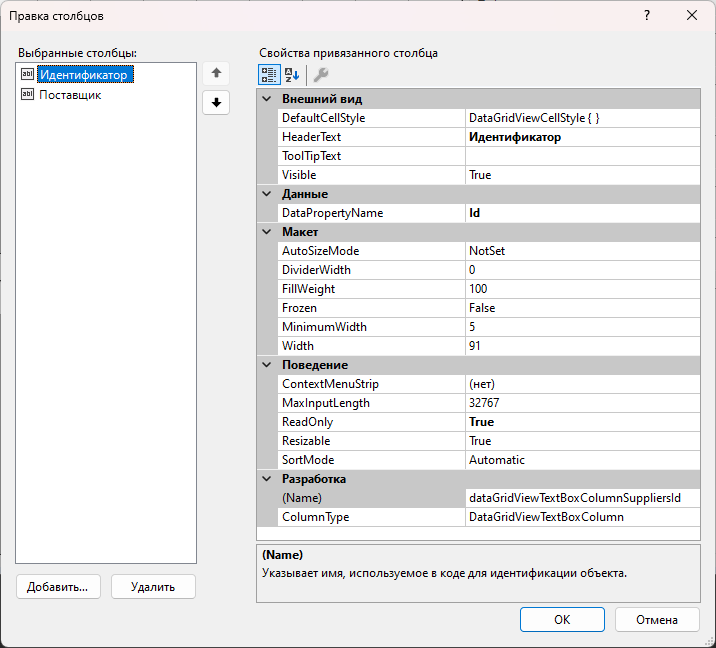


Рисунок №4 – Настройка столбцов таблицы-справочника   
«Поставщики оборудования»

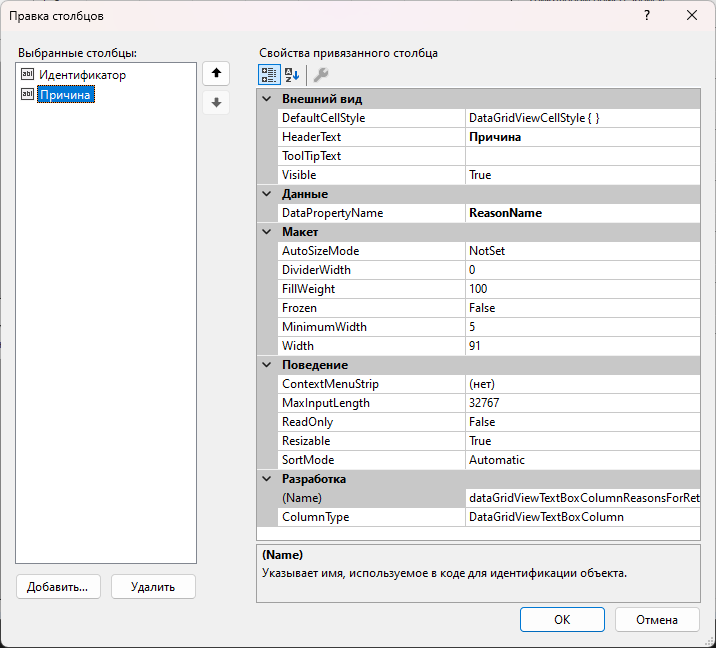


Рисунок №5 – Настройка столбцов таблицы-справочника   
«Причина возврата»

Согласно заданию, для таблиц «Поставщики оборудования» и «Причина возврата» были переименованы названия столбцов на русский язык, с помощью свойства **HeaderText**.

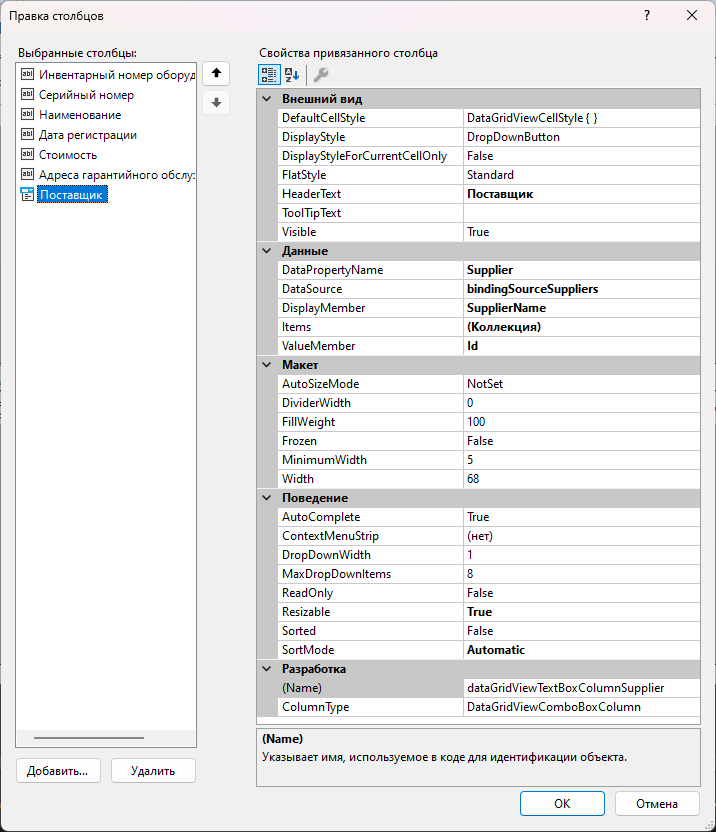


Рисунок №6 – Настройка подстановки справочных значений в столбец «Поставщик»

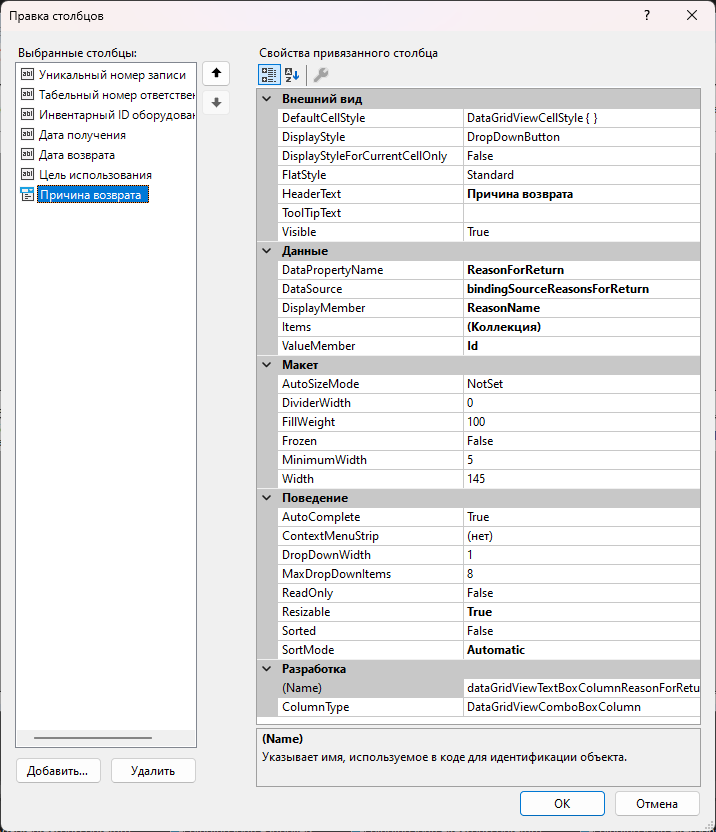


Рисунок №7 – Настройка подстановки справочных значений в столбец «Причина возврата»

Для столбцов, связанных с таблицами-справочниками, были настроены подстановки справочных значений. Тем самым, вместо идентификатора поставщика или причины возврата оборудования, в таблице отображается название. У данных столбцов был изменён ряд свойств:

* **ColumnType = DataGridViewComboBoxColumn**: переводит столбец в режим выбора значения из списка.
* **DataSource = bindingSource${tableName}**: в качестве источника данных для списка используется источник данных таблицы-справочника. **${tableName}** – название таблицы-справочника.
* **DisplayMember = ${columnName}**: в качестве отображаемого значения используется текстовое справочное значение из таблицы-справочника.
* **ValueMember = Id**: для сопоставления значений используется числовой идентификатор из исходной таблицы, связанный внешним ключом идентификатором из таблицы-справочника.

# 3 Добавленный программный код с пояснениями

Были изменены методы, отвечающие за обработку событий.

В метод **LoadForm(object, EventArgs)** были добавлены команды для загрузки данных из таблиц-справочников.

private void LoadMainForm(object sender, EventArgs e)

{

tableAdapterSuppliers.Fill(dataSetLessonsUrfu.Suppliers);

tableAdapterReasonsForReturn.Fill(dataSetLessonsUrfu.ReasonsForReturn);

tableAdapterResponsibility.Fill(dataSetLessonsUrfu.Responsibility);

tableAdapterResponsible.Fill(dataSetLessonsUrfu.Responsible);

tableAdapterEquipment.Fill(dataSetLessonsUrfu.Equipment);

}

В метод **SaveItem(object, EventArgs)** были добавлены команды для завершения редактирования данных в таблицах. Формирование запроса и его отправка на сервер базы данных выполняется в рамках методов, сгенерированных мастером настройки источников данных.

/// <summary>

/// Сохранение элементов в таблицы

/// </summary>

private void SaveItem(object sender, EventArgs e)

{

Validate();

bindingSourceEquipment.EndEdit();

bindingSourceResponsible.EndEdit();

bindingSourceResponsibility.EndEdit();

bindingSourceSuppliers.EndEdit();

bindingSourceReasonsForReturn.EndEdit();

tableAdapterManager.UpdateAll(dataSetLessonsUrfu);

}

Для всех табличных преставлений используется один обработчик события **SelectionChanged**. Как и в прошлых лабораторных работах он выполняет смену источника данных для панели навигации, переключение связанного источника данных для **bindingSourceResponsibility** (если фокус находится в таблицах «Ответственные» и «Оборудование»), а также отображения значений текущей строки таблицы-связки «Ответственность за оборудование» в текстовых полях.

Так как таблица-связка «Ответственность за оборудование» имеет внешний ключ, связанный с таблицей-справочником «Причина возврата», обработчик не выполняет полный перенос значений из исходной таблицы в текстовые поля. Для последнего столбца выполняется поиск текстового значения методом перебора.

/// <summary>

/// Обработчик события выполняющий смену источника данных для панели навигации, при выборе элемента,

/// а также изменение полей текущей записи в GroupBox

/// </summary>

private void SelectionChanged(object sender, EventArgs e)

{

var source = (sender as DataGridView)!.DataSource as BindingSource;

var textBoxes = new[]

{

textBoxCurrentId,

textBoxCurrentResponsible,

textBoxCurrentEquipment,

textBoxCurrentDateOfReceiving,

textBoxCurrentReturnDate,

textBoxCurrentPurposeOfUse,

textBoxCurrentReasonForReturn

};

// Изменяем привязку

if (source == bindingSourceEquipment)

{

bindingSourceResponsibility.DataSource = source;

bindingSourceResponsibility.DataMember = "FK\_Responsibility\_Equipment";

}

else if (source == bindingSourceResponsible)

{

bindingSourceResponsibility.DataSource = source;

bindingSourceResponsibility.DataMember = "FK\_Responsibility\_Responsible";

}

// Изменяем значения полей

if (dataGridViewResponsibility.CurrentRow != null)

{

for (var i = 0; i < textBoxes.Length - 1; i++)

{

textBoxes[i].Text = dataGridViewResponsibility.CurrentRow.Cells[i].Value?.ToString();

}

// Поиск названия причины возврата

var reasonForReturnId = (dataGridViewResponsibility.CurrentRow.Cells[textBoxes.Length - 1].Value ?? "").ToString();

foreach (DataGridViewRow row in dataGridViewReasonsForReturn.Rows)

{

if (row.Cells[0].Value?.ToString() == reasonForReturnId)

{

textBoxCurrentReasonForReturn.Text = row.Cells[1].Value?.ToString();

break;

}

textBoxCurrentReasonForReturn.Text = "";

}

}

else

{

foreach (var textBox in textBoxes)

{

textBox.Text = "";

}

}

tablesBindingNavigator.BindingSource = source;

}

# 4 Подробное описание впервые использованных свойств и методов

Свойства класса ***DataGridViewComboBoxColumn***:

* **ColumnType**: отображает тип столбца, используемый табличным представлением ***DataGridView***. Отображается в конструкторе формы, как свойство, но представляет собой класс объекта, представляющего собой столбец.
* **DataSource**: возвращает или принимает источник данных, используемый для отображения значений в столбце. Принимает и возвращает объект типа ***object***.
* **DisplayMember**: возвращает или принимает строку, указывающую на столбец, используемый для получения доступных значений для отображения в столбце. Принимает и возвращает объект типа ***string***.
* ***ValueMember***: возвращает или принимает строку, указывающую на столбец, используемый для получения значений, которые будут сопоставляться со значениями **DisplayMember**. Принимает и возвращает объект типа ***string***.