# Лабораторная работа №2Механизм привязки данных.Разработка компонентов Интерфейса пользователя

**Цель работы**: Изучение базовых понятий механизма привязки данных к элементам управления пользовательского интерфейса в среде разработки *Visual Studio*, закрепление навыков компонентно-ориентированного подхода к разработке программ.

## Задание

Разработать программу, которая состоит из нескольких модулей (сборок, *assembly*):

* основного приложения с графическим интерфейсом (*Windows Forms Application*);
* модуля (*Class Library*) с описанием структуры данных предметной области на основе обобщенных коллекций (*generic collections*);
* модуля (*Class Library*) с компонентами (*User Control*) отображения данных предметной области.

Модули и основное приложение должны быть реализованы в виде отдельных проектов в составе одного рабочего пространства (*solution*). Для организации связи между объектами предметной области и объектами интерфейса пользователя использовать механизм связывания (*binding*), например, с помощью классов *BindingSource*, *BindingNavigator* и т.д. Основные требования к работе представлены в таблице 2.1.

Основная программа должна содержать три компонента, которые на данном этапе отображают таблицы предметной области без учета связей между ними. Реализация внутренних связей и контроль за непротиворечивостью данных будут рассмотрены в следующей лабораторной работе.

## Порядок выполнения работы

1. Создать пустое рабочее пространство (*Blank Solution*) без проектов.
2. В созданное рабочее пространство с помощью контекстного меню добавить три новых проекта (сначала *Windows Forms Application*, затем *Class Library* и *Windows Forms Control Library*).
3. Добавить в проектах необходимые ссылки (*Reference*) на используемые сборки. В некоторых случая может потребоваться указание зависимостей между проектами для правильного порядка компиляции.
4. Добавить в проект *Class Library* описание структур данных предметной области (классы, соответствующие одной записи/сущности в таблице)
5. С помощью одной из обобщенных коллекций создать классы, каждый из которых описывает набор/таблицу сущностей предметной области (реализованных в виде классов в соответствии с п.4).
6. В рабочем пространстве выбрать проект *Windows Forms Control Library* и добавить источник(и) данных (*Data/Add New Data Source…*), которые ссылаются на коллекции из модуля со структурами данных предметной области.
7. В проект *Windows Forms Control Library* добавить три компонента *UserControl* каждый из которых должен отображать одну таблицу из предметной области.
8. Используя технологию *Drag&Drop*, перетащить класс-коллекцию из окна источников данных на рабочее поле *UserControl*'a. В результате автоматически будут добавлены элементы отображения и управления данными (*DataGridView*, *BindingSource*, *BindingNavigator* и пр.). Повторить действия для каждой таблицы и компонента отображения данных.
9. Разместить созданные компоненты пользовательского интерфейса на основной форме приложения и проверить/продемонстрировать работоспособность основных функций работы с набором данных (добавление, удаление, редактирование без учета связей между таблицами).

|  |  |  |  |
| --- | --- | --- | --- |
| № | Требования к работе | Баллы | Оценка |
|  | Реализация задания в виде группы из трёх проектов в одном рабочем пространстве | 1 |  |
|  | Использование механизма привязки данных (*data binding*) для создания компонентов отображения данных | 2 |  |
|  | Применение выпадающего списка в *DataGridView* с привязкой к данным из связанных таблиц предметной области | 2 |  |
|  | Компиляция всех модулей и приложения в одном каталоге, который задается разработчиком | 1 |  |
|  | Реализация сортировки данных, отображаемых в *DataGridView* | 2 |  |

## Мой вариант задания

