**Тема: «Технологии создания графического пользовательского интерфейса»**

**Практическая часть**

Используем предыдущие задание, для выполнения текущего практического задания.

Каждая задача (вариант) должна реализовывать три основных аспекта работы с привязкой данных в WPF:

1. **DataBinding** (одно- и двусторонняя привязка) – привязка элементов UI к данным в модели.
2. **Работа с коллекциями в WPF (ObservableCollection)** – реализация динамического обновления интерфейса при изменении данных.
3. **DataTemplates и ControlTemplates** – настройка внешнего вида элементов списка и кастомизация UI-элементов.

**1. АРМ работника склада**

**Функционал:** управление списком товаров, отображение информации о складе.

**1. DataBinding (Привязка данных)**

* Привязать TextBox к свойствам Название товара, Количество и Цена через TwoWay привязку.
* Привязать ComboBox для выбора категории товара (OneWay привязка).

**2. Работа с коллекциями (ObservableCollection)**

* Хранить список товаров в ObservableCollection<Product>, обновляя UI при добавлении или удалении.
* Реализовать фильтрацию списка товаров по категории.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения товаров в ListView: Название, Количество, Цена.
* ControlTemplate для стилизации кнопки "Добавить товар" (например, с иконкой +).

**2. CRM для небольшого бизнеса**

**Функционал:** управление клиентской базой.

**1. DataBinding (Привязка данных)**

* Привязка TextBox к имени, номеру телефона и email клиента (TwoWay привязка).
* ComboBox для выбора типа клиента (OneWay привязка).

**2. Работа с коллекциями (ObservableCollection)**

* Использовать ObservableCollection<Client> для хранения списка клиентов.
* Реализовать возможность удаления и добавления клиентов с обновлением UI.

**3. DataTemplates и ControlTemplates**

* DataTemplate для списка клиентов с ФИО, телефоном и статусом (VIP, обычный клиент).
* ControlTemplate для кнопки "Сохранить изменения" с кастомным дизайном.

**3. Электронный дневник студента**

**Функционал:** ведение оценок студентов.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для изменения оценок (TextBox внутри DataGrid).
* OneWay привязка для отображения средней оценки.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<StudentGrade> для хранения оценок студентов.
* Автоматическое обновление среднего балла при изменении данных.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения имени студента, предмета и оценок в ListView.
* ControlTemplate для кнопки "Обновить оценки" (например, иконка с галочкой).

**4. Управление задачами (Task Manager)**

**Функционал:** список задач, установка статусов.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для изменения названия и дедлайна задачи (TextBox).
* OneWay привязка для отображения статуса задачи (Label).

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<TaskItem> для хранения списка задач.
* Фильтрация по статусу (Выполнено, В работе, Ожидание).

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения задачи с названием, статусом и сроком.
* ControlTemplate для кнопки "Изменить статус" (например, смена цвета при клике).

**5. Система бронирования отеля**

**Функционал:** бронирование номеров.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для изменения данных клиента (ФИО, контакты).
* OneWay привязка для отображения информации о номере.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Room> для списка номеров.
* Автоматическое обновление статуса номеров при бронировании.

**3. DataTemplates и ControlTemplates**

* DataTemplate для карточки номера: номер комнаты, цена, статус.
* ControlTemplate для кнопки "Забронировать".

**6. Управление заказами в интернет-магазине**

**Функционал:** список заказов, статус доставки.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для изменения количества товара в заказе.
* OneWay привязка для отображения статуса заказа.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Order> для хранения заказов.
* Автоматическое обновление статуса доставки.

**3. DataTemplates и ControlTemplates**

* DataTemplate для списка заказов (Товар, Количество, Статус).
* ControlTemplate для кнопки "Изменить статус" (например, выпадающий список).

**7. Автоматизация учета сотрудников**

**Функционал:** управление базой сотрудников.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для редактирования данных сотрудника.
* OneWay привязка для отображения должности.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Employee> для списка сотрудников.
* Фильтрация по отделу.

**3. DataTemplates и ControlTemplates**

* DataTemplate для карточки сотрудника: ФИО, Должность.
* ControlTemplate для кнопки "Добавить сотрудника".

**8. Приложение для учета финансов**

**Функционал:** список доходов и расходов.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для редактирования суммы операции.
* OneWay привязка для отображения текущего баланса.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Transaction> для хранения финансовых операций.
* Фильтрация по дате.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения списка транзакций (Дата, Категория, Сумма).
* ControlTemplate для кнопки "Добавить запись".

**9. Бронирование билетов в кинотеатр**

**Функционал:** выбор фильма и сеанса.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для выбора количества билетов.
* OneWay привязка для отображения названия фильма.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<MovieSession> для списка сеансов.
* Фильтрация по времени.

**3. DataTemplates и ControlTemplates**

* DataTemplate для карточки фильма.
* ControlTemplate для кнопки "Купить билет".

**10. Система контроля версий документов**

**Функционал:** отслеживание изменений файлов.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для редактирования названия документа.
* OneWay привязка для отображения версии файла.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Document> для списка документов.
* Автоматическое обновление при загрузке нового файла.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения документа (Имя, Дата изменения).
* ControlTemplate для кнопки "Загрузить новую версию".

**11. Программа для учета медицинских записей**

**Функционал:** управление картами пациентов, просмотр истории приемов.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для изменения данных пациента (ФИО, возраст, диагноз).
* OneWay привязка для отображения списка прошлых приемов.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<MedicalRecord> для хранения записей пациентов.
* Фильтрация по дате последнего посещения.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения списка пациентов с краткими сведениями (ФИО, возраст, последний прием).
* ControlTemplate для кнопки "Добавить запись" (например, стилизованная кнопка с медицинским крестом).

**12. Управление учебным процессом (электронный журнал преподавателя)**

**Функционал:** ведение списка студентов, оценок и посещаемости.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для ввода оценок и статуса посещаемости (CheckBox).
* OneWay привязка для отображения средней успеваемости студента.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<StudentGrade> для хранения оценок и посещаемости.
* Автоматический пересчет среднего балла при изменении данных.

**3. DataTemplates и ControlTemplates**

* DataTemplate для списка студентов с их оценками и посещаемостью.
* ControlTemplate для кнопки "Экспортировать в Excel" (можно стилизовать под стандартную иконку экспорта).

**13. Приложение для аренды автомобилей**

**Функционал:** управление списком доступных автомобилей, бронирование.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для выбора автомобиля и даты бронирования.
* OneWay привязка для отображения стоимости аренды.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Car> для хранения доступных автомобилей.
* Фильтрация списка по классу автомобиля и наличию свободных мест.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения списка автомобилей (марка, модель, цена в сутки).
* ControlTemplate для кнопки "Забронировать" (например, стилизованная кнопка с иконкой автомобиля).

**14. Приложение для организации мероприятий**

**Функционал:** создание и управление мероприятиями, регистрация участников.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для редактирования названия и даты мероприятия.
* OneWay привязка для отображения количества зарегистрированных участников.

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Event> для хранения списка мероприятий.
* Автоматическое обновление количества участников при регистрации.

**3. DataTemplates и ControlTemplates**

* DataTemplate для списка мероприятий (Название, Дата, Количество участников).
* ControlTemplate для кнопки "Зарегистрироваться" (например, стилизованная кнопка с иконкой человека).

**15. Электронная библиотека**

**Функционал:** управление списком книг, поиск по каталогу.

**1. DataBinding (Привязка данных)**

* TwoWay привязка для ввода названия книги и автора.
* OneWay привязка для отображения статуса книги (в наличии / занята).

**2. Работа с коллекциями (ObservableCollection)**

* ObservableCollection<Book> для хранения списка книг.
* Фильтрация по автору и жанру.

**3. DataTemplates и ControlTemplates**

* DataTemplate для отображения списка книг (Название, Автор, Статус).
* ControlTemplate для кнопки "Взять книгу" (например, стилизованная кнопка с иконкой книги).