# Теоретическая часть

* Особенности использования контекстного меню в **TreeView** – по материалам домашнего задания
* Понятие о покадровой анимации, простейшая реализация покадровой анимации
* Понятие о стартовом экране, перемещение окна без заголовка, управление видимостью элементов управления
* Понятие о **DataGridView**, основные свойства **DataGridView**
* Заполнение **DataGrodView** данными
* Заполнение **DataGridView** данными из коллекции без использования свойства **DataSource**
* Контекстное меню в **DataGridView**, обработка выбранных строк **DataGridView**

# Практическая часть

**Задача 1.** Разработайте приложение Windows Forms с использованием изученных элементов интерфейса (строковое меню, контекстное меню, панель инструментов, лист-бокс, комбо-бокс, строка состояния, строки ввода, чек-боксы и радиокнопки, вкладки, **DataGridView**).

Разработайте класс, описывающий электроприбор: название; мощность; цена; состояние прибора: включен/выключен; название комнаты, в которой размещен прибор (кухня, прихожая, …).

Разработайте класс, описывающий электрооборудование квартиры: адрес квартиры, условное название квартиры (Васин дом, Орлиная слободка, …) коллекция электроприборов. Отдельно хранить названия комнат не надо.

Приложение должно использовать вкладки. Сведения о квартире и список электроприборов выводить в первой вкладке главной формы.

Сведения о квартире разместить в виде дерева в **TreeView** в левой части вкладки, в правой части – вывод этих же сведений в **DataGridView** в табличном формате. Выводить картинку состояния в таблицу не надо, только текст «включен» или «выключен».

При построении дерева используйте следующие соответствия: корень дерева – условное название квартиры, узлы – названия комнат, листья – электроприборы в комнате. Изображение узла дерева – условное обозначение электроприбора.

Приложение должно иметь следующий функционал (при помощи пунктов меню, контекстного меню, кнопок панели инструментов):

* Сворачивать приложение в трей, восстанавливать форму приложения из контекстного меню иконки в трее. В этом же контекстом меню разместите команды для завершения приложения, вызова формы со сведениями о приложении и разработчике
* При запуске приложения проверять наличие в папке исполняемого файла папки **App\_Data** и файла **appliances.json** с данными о квартире. Если папки и/или файл нет, то создать папку, заполнить начальными данными объект, описывающий электрооборудование квартиры, сериализовать данные в формате JSON
* Выводить коллекцию электроприборов в **DataGridView**, состояние прибора отображать текстом «включен», «выключен»
* Переформирование коллекции электроприборов и сведений о квартире с сериализацией
* Сохранение данных в выбранном файле – сериализация в формате JSON. Файл для сохранения выбирать стандартным диалогом
* Загрузка данных из выбранного стандартным диалогом файла, перетаскиванием файла на **DataGridView** главной вкладки, на **TreeView** главной вкладки, десериализация в формате **JSON**.
* Добавление электроприбора в коллекцию (ввод данных в отдельной форме), сериализация данных
* Удаление электроприбора из коллекции, сериализация данных
* Редактирование выбранного электроприбора в отдельной форме, сериализация данных
* Редактирование названия квартиры, комнаты, электроприбора непосредственно в узлах дерева – редактирование меток узлов дерева
* Редактирование данных квартиры – адрес и название, сериализация данных
* Упорядочивание копии коллекции электроприборов, вывод упорядоченной копии коллекции в отдельной вкладке. При выборе команды переходить на вкладку
  + По названию
  + По состоянию
  + По мощности
  + По убыванию цены
* Включение/выключение выбранного электроприбора
* Включение/выключение всех электроприборов квартиры
* Включение и выключение всех электроприборов комнаты
* Выборка и вывод в отдельной вкладке главной формы коллекции электроприборов, с заданным названием. При выборе команды переходить на вкладку
* Выборка и вывод в отдельной вкладке главной формы коллекции электроприборов заданного состояния. При выборе команды переходить на вкладку

В меню и контекстных меню (**DataGridView**, **MainForm, NotifyIcon**) включите команды выхода, вывода окна со сведениями о программе и ее разработчике.

**Задача 2.**  Разработайте приложение для обработки коллекции животных, описываемых следующим классом **Animal**:

* индекс иконки животного в коллекции **ImageList**, иконки есть в папке **Animals-mini**,выводить картинку в **DataGridView** не надо
* вид животного (те, для которых есть картинка в **Animals-mini**: кот, лев, слон, рыба, …)
* кличка животного
* вес животного (в кг)
* возраст животного (в годах)
* цвет (масть) животного,
* фамилия и инициалы владельца (Иванов И.И., …).

Создать коллекцию животных (не менее 20 животных), отобразить в главной вкладке окна приложения, в элементах **TreeView**, **DataGridView**. В отдельных вкладках также в **DataGridView** выводить:

* выборку коллекции животных с максимальным весом
* выборку коллекции животных, фамилия владельцев которых содержит заданную подстроку (подстроку вводить в этой же вкладке)
* отсортированную по возрасту копию коллекции животных
* отсортированную по виду копии коллекции животных
* отсортированную по фамилиям владельцев копию коллекции животных

В **TreeView** корнем дерева является условное название коллекции, каждое животное – узел дерева, выводить условное обозначение животного его вид и кличку, листьями узла должны быть вес животного, полное имя владельца.

Реализовать добавление животных в коллекцию, редактирование данных животного в коллекции и удаление животных из коллекции. Создание животного выполняйте в отдельной форме, удаление – из главной вкладке, при помощи контекстного меню, клавиши Del, кнопки на панели инструментов.

Используйте комбо-бокс для выбора вида животного. Используйте и другие изученные элементы интерфейса для построения интерфейса пользователя.

Сохраняйте коллекцию в файле типа **JSON**, в папке **App\_Data** у исполняемого файла. Загружайте коллекцию из файла **JSON**. Файл выбирайте при помощи стандартного диалога, а также при помощи перетаскивания на главную вкладку из Проводника. При первом запуске приложения или при отсутствии файла коллекции – создавайте коллекцию генератором.

Выводите текущее количество животных в коллекции и имя файла данных в строку состояния.

В меню и контекстных меню (**DataGridViewView**, **MainForm, NotifyIcon**) включите команды выхода, вывода окна со сведениями о программе и ее разработчике.

# Дополнительно

Запись занятия можно скачать [**по этой ссылке**](https://cloud.mail.ru/public/vLGQ/W5QGfLRB2), материалы занятия в этом же архиве.