

Вариант 1

На основе одной из готовых обобщенных (шаблонных) объектных коллекций .NET создать класс «Организация», включающий сотрудников. Классы сотрудников должны образовывать иерархию с базовым классом. Сотрудники бывают двух типов: с фиксированной оплатой и почасовики. Описать в базовом классе абстрактный метод для расчета среднемесячной заработной платы. Для почасовиков формула для расчета такова: «среднемесячная заработная плата = $20.8 * 8 * \text{почасовую ставку}$ », для сотрудников с фиксированной оплатой «среднемесячная заработная плата = фиксированной месячной оплате», также к зарплате прибавляется фиксированная премия, размер которой зависит от должности сотрудника. В виде меню программы реализовать нижеприведенный функционал.

1. Упорядочить всю последовательность сотрудников по убыванию среднемесячного заработка. При совпадении зарплат – упорядочивать данные по алфавиту по ФИО. Вывести идентификатор работника, ФИО, день рождения и среднемесячный заработок для всех элементов списка.
2. Вывести первые 5 имен работников из полученного в пункте 1 списка.
3. Вывести последние 3 идентификатора работников из полученного в пункте 1 списка.
4. В реальном времени (в процессе заполнения списка сотрудников) рассчитывать и поддерживать в актуальном состоянии среднемесячную заработную плату по организации, сохранить значение как поле класса «Организация».
5. Организовать запись и чтение всех данных в/из файла. Реализовать поддержку 2-х форматов файлов: XML и JSON. Использовать либо стандартные средства (форматтеры и сериализаторы), либо собственные.
6. Организовать обработку некорректного формата входного файла.

Вариант 2

На основе одной из готовых обобщенных (шаблонных) объектных коллекций .NET создать класс «Авиакомпания», включающий список самолетов. Классы самолетов должны образовывать иерархию с базовым классом. Самолеты бывают двух типов: пассажирские и грузовые. Описать в базовом классе абстрактный метод для расчета взлетного веса самолета. Для пассажирских авиалайнеров взлетный вес – это количество мест на средний вес человека (62 кг.), для грузовиков взлетный вес задается явно исходя из фактического веса груза, также взлетный вес включает вес самого самолета, он определяется исходя из типа авиалайнера. В виде меню программы реализовать нижеприведенный функционал.

1. Упорядочить всю последовательность авиалайнеров по возрастанию взлетного веса. При совпадении взлетного веса – упорядочивать данные по номеру рейса. Вывести номер рейса, тип самолета, список ФИО экипажа и взлетный вес для всех элементов списка.
2. Вывести первые 6 бортов из полученного в пункте 1 списка.
3. Вывести последние 2 номера рейса борта из полученного в пункте 1 списка.
4. В реальном времени (в процессе заполнения списка авиалайнеров) рассчитывать и поддерживать в актуальном состоянии средний взлетный вес самолета по авиакомпании, сохранить значение как поле класса «Авиакомпания».
5. Организовать запись и чтение всех данных в/из файла. Реализовать поддержку 2-х форматов файлов: XML и JSON. Использовать либо стандартные средства (форматтеры и сериализаторы), либо собственные.
6. Организовать обработку некорректного формата входного файла.

Вариант 3

На основе одной из готовых обобщенных (шаблонных) объектных коллекций .NET создать класс «Графический редактор», включающий список нарисованных фигур: круг, квадрат, эллипс, прямоугольник. Классы фигур должны образовывать иерархию с базовым классом (фигура). Описать в базовом классе абстрактный метод для расчета площади фигуры. Площади фигур рассчитываются по стандартным геометрическим формулам, также в площади учитывается толщина рамки, заданная индивидуально для каждой фигуры. В виде меню программы реализовать нижеприведенный функционал.

1. Упорядочить всю последовательность графических фигур по возрастанию площади. При совпадении площади – упорядочивать данные по толщине рамки. Вывести тип, толщину рамки и площадь для всех элементов списка.
2. Вывести первые 3 типа фигуры из полученного в пункте 1 списка.
3. Вывести последние 2 фигуры из полученного в пункте 1 списка на консоль в виде символов «*» без учета толщины рамки, начало координат – центр консоли, если фигура выходит за пределы консоли, вывести предупреждение.
4. В реальном времени (в процессе заполнения списка фигур) рассчитывать и поддерживать в актуальном состоянии среднюю площадь фигуры, сохранить значение как поле класса «Графический редактор».
5. Организовать запись и чтение всех данных в/из файла. Реализовать поддержку 2-х форматов файлов: XML и JSON. Использовать либо стандартные средства (форматтеры и сериализаторы), либо собственные.
6. Организовать обработку некорректного формата входного файла.

Вариант 4

На основе одной из готовых обобщенных (шаблонных) объектных коллекций .NET создать класс «Управляющая компания», включающий список объектов недвижимости (строений). Классы строений должны образовывать иерархию с базовым классом. Объекты недвижимости бывают двух типов: жилые и нежилые. Описать в базовом классе абстрактный метод для расчета приближенного среднего количества жильцов/работников строения. Для жилых построек среднее количество жильцов – это количество квартир на количество комнат в квартире (тип квартиры) на 1.3, для нежилых строений среднее количество сотрудников пропорционально площади с коэффициентом 0.2. В виде меню программы реализовать нижеприведенный функционал.

1. Упорядочить всю последовательность объектов недвижимости по убыванию среднего количества жильцов/работников. При совпадении значения – упорядочивать данные по типу строения (жилые, нежилые), затем по алфавиту по адресу строения. Вывести тип строения, адрес строения, среднее количество жильцов/работников для всех элементов списка.
2. Вывести первые 3 объекта из полученного в пункте 1 списка.
3. Вывести последние 4 адреса объекта из полученного в пункте 1 списка.
4. В реальном времени (в процессе заполнения списка строений) рассчитывать и поддерживать в актуальном состоянии среднее количество жильцов/работников объекта недвижимости по компании в целом, сохранить значение как поле класса «Управляющая компания».
5. Организовать запись и чтение всех данных в/из файла. Реализовать поддержку 2-х форматов файлов: XML и JSON. Использовать либо стандартные средства (форматтеры и сериализаторы), либо собственные.
6. Организовать обработку некорректного формата входного файла.