

Лабораторная работа №9

Использование стандартных интерфейсов

1 Цель работы

- 1.1 Изучить процесс реализации стандартных интерфейсов на языке C#.

2 Литература

- 2.1 <https://metanit.com/sharp/tutorial/> – гл.7.

3 Подготовка к работе

- 3.1 Повторить теоретический материал (см. п.2).
3.2 Изучить описание лабораторной работы.

4 Основное оборудование

- 4.1 Персональный компьютер.

5 Задание

Разработать класс по варианту из ЛР№5. В классе должны быть открытые автосвойства (вместо закрытых полей) и метод для вывода значений или ToString.

5.1 Реализация стандартного интерфейса IComparable (без обобщений)

Реализовать в классе интерфейс IComparable для того, чтобы объекты сравнивались по последнему числовому полю.

Для выполнения сравнения в методе `public int CompareTo(object obj)` привести объект `obj` к типу разработанного класса. Далее можно использовать условный оператор для сравнения значения полей объектов или вызывать метод `CompareTo` у поля текущего объекта.

Для проверки создать массив объектов класса и отсортировать этот массив, вызвав `Array.Sort(массив)`. Если все сделано корректно, массив будет отсортирован по возрастанию значений поля. На экран должен быть выведен исходный и отсортированный массив.

5.2 Реализация стандартного интерфейса IEquatable<T> (с обобщениями)

Реализовать в классе интерфейс IEquatable для того, чтобы объекты сравнивались по всем полям (истина должна возвращаться при равенстве всех значений полей).

Для выполнения сравнения в методе `public bool Equals(тип obj)` использовать условный оператор для сравнения значения полей объектов.

Для проверки создать два объекта класса и сравнить эти объекты. На экран вывести результат сравнения.

5.3 Реализация стандартного интерфейса ICloneable

Реализовать в классе интерфейс ICloneable для того, чтобы выполнялось копирование объекта (создавать его на основе значений исходного).

Для проверки создать два объекта класса и клонировать один из объектов. Проверить, как изменяются значения клонированного объекта и неклонированного объекта (объекта, которому присвоено значение другого объекта).

5.4 Реализация стандартного интерфейса IComparer<T> (с обобщениями)

Реализовать в классе интерфейс IComparer в классе DescendingComparer для того, чтобы сортировать по убыванию последнего числового значения.

Реализовать интерфейс IComparer в классе StringComparer для того, чтобы сортировать по возрастанию первого строкового значения.

Для проверки создать массив объектов класса и отсортировать этот массив, вызвав Array.Sort(массив, new ИмяComparer()). Если все сделано корректно, массив будет отсортирован по выбранному компаратору. На экран должен быть выведен исходный и отсортированный массив.

6 Порядок выполнения работы

6.1 Запустить MS Visual Studio и создать консольное приложение C# (Console Application).

6.2 Выполнить все задания из п.5 в одном решении LabWork9, каждое — в своем проекте (классы и интерфейсы должны быть описаны в отдельных файлах .cs и протестированы в основной программе).

При разработке считать, что пользователь ввел данные требуемого типа, остальные возможные ошибки обрабатывать.

При выполнении заданий использовать минимально возможное количество команд и переменных и выполнять форматирование и рефакторинг кода.

6.3 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Для чего используется интерфейс IComparable?

8.2 Для чего используется интерфейс IEquatable?

8.3 В чем отличие между обобщенным и необобщенным интерфейсами?