### Интерфейсы(interface)

*Интерфейс* – это контракт или договор, который гарантирует, что класс реализует все заявленные элементы интерфейса.

Интерфейс содержит сигнатуры без реализации. Сейчас в С# можно указывать интерфейсы в реализации по умолчанию.

Общая форма:

*модификаторы interface IИмяИнтерфейса*

*{ИмяМетода (параметры);*

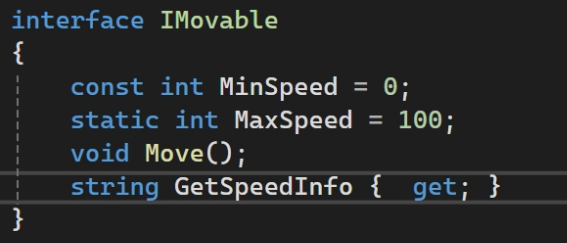
*ИмяСвойства (get; set;)*

*this[индексы] (get; set;)}*

Отличие интерфейса от абстрактного класса:

* Интерфейс не может содержать данные и реализацию,
* В интерфейсе все public.
* В С# c версии 8 в интерфейсах можно:
* Объявлять статические поля и константы,
* Указывать модификаторы доступа у элементов интерфейса,
* Создавать реализацию по умолчанию,
* Создавать реализацию у приватных(private) и статических(static) методов и свойств.

Пример:



Особенности:

* Реализация интерфейса совпадает с наследованием абстрактного класса и переопределением методов,
* Тип данных может реализовывать множество интерфейсов,
* При переопределении слово override писать не надо.

модификаторы class ИмяКласса : IИмяИнтерфейса1, IИмяИнтерфейса2, …

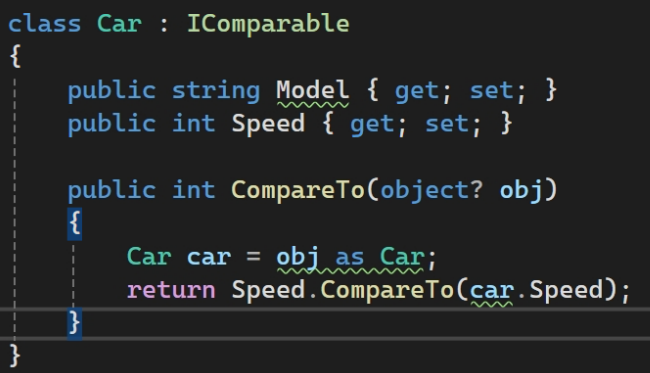
{…}

В С# есть множество стандартных интерфейсов:

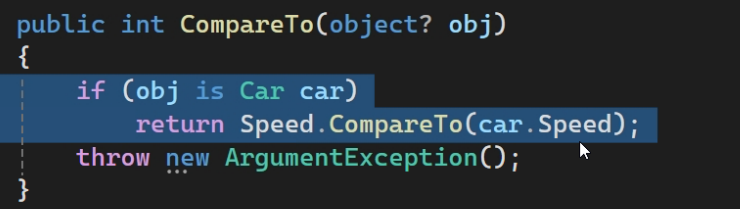
1. IComparable – используется при сравнении и сортировке,
2. IEquatable - отвечает за сравнение объектов,
3. IComparer – отвечает за сравнение,
4. IEnumarable – обеспечивает перебор элементов коллекции,
5. INotifyPropertyChanged – позволяет информировать об изменение значения свойств

Пример реализации IComparable:

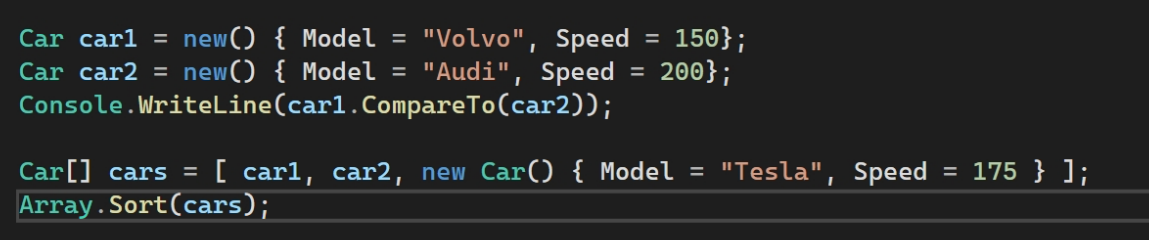
Содержит в себе метод **CompareTo**, который возвращает для простых типов данных или строк 1 если объект больше (>), 0 если объяты равны (==), -1 если объект меньше (<).



Или



Пример:

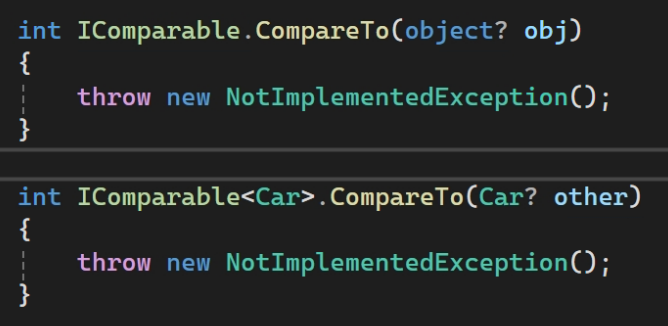


Создание массива его сортировка по полю Speed.

#### Явная реализация интерфейсов

Если класс реализует интерфейсы с совпадающей сигнатурой методов, то перед названием методов пишется название интерфейса.

Пример явной реализации интерфейса:



При реализации можно указать модификаторы (**new**, **abstract**, **virtual**, **override**)

#### Наследование интерфейсов

У интерфейса можно указать 1 и более родительских интерфейса

Общая форма:

interface IИмяИнтерфейса : IИмяРодителя1, IИмяРодителя2, …

{

// без элементов родительских интерфейсов

}