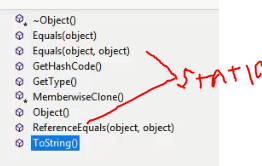
Object

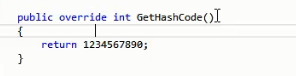


~Object – деструктор, вызывается гарбачка лектором неявно, вызвать его пользователь не может

GetType() – возвращает ссылку на экземпляр класса Type, вся информация о типе

Membervise Clone() – клонирование объекта

GetHashCode() – хэш код объекта

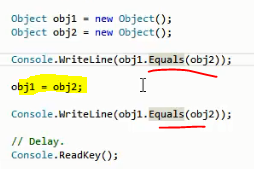


ToString() – выводит полное квалифицированное имя типа



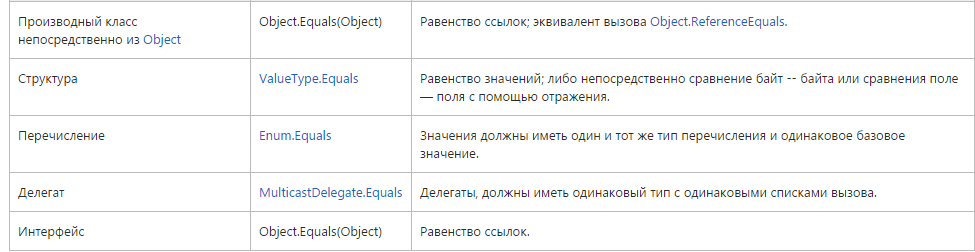
Equals(object) - obj1.Equals(obj2)

|  |  |  |
| --- | --- | --- |
|  | Object.Equals(Object) | Равенство ссылок; эквивалент  вызова [Object.ReferenceEquals](http://msdn.microsoft.com/ru-ru/library/system.object.referenceequals(v=vs.110).aspx). |

итог 

true, сравнивает значения

false, сравнивает поля



При переопределении виртуального метода Equals() бызового класса Object мы должны переопределить метод GetHashCode()

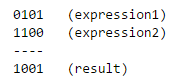


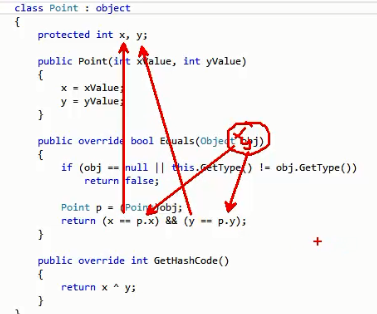
В данном примере в методе Equals() мы возвращаем false

* Если obj равен null, либо он другого типа чем объект на котором вызвали метод для проверки

После мы приводим obj к типу объекта на котором вызван метод и сравниваем поля

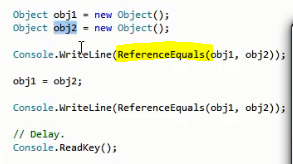
В методе GetHashCode() делаем исключающее или ^

пример работы ИЛИ



ReferenceEquals() – сравнивает ссылки объектов

Вызывается без объекта Object() !!!!!!!!!!!!!!!!!!!!!!!!!!!!!



Статический метод [**Equals**](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object/Equals(Object,Object):Boolean)

public static [bool](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Boolean) [**Equals**](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object/Equals(Object,Object):Boolean)([object](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object) objA, [object](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object) objB)

если объекты по ссылкам (сравнение по ссылкам ==) равны, то true

если оба объекта не равны null, тогда можем вызвать на объекте

[object](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object) objA метод objA.[Equals](http://www.aisto.com/roeder/dotnet/Default.aspx?Target=code://mscorlib:4.0.0.0:b77a5c561934e089/System.Object/Equals(Object):Boolean)(objB); (если метод переопределён вернет равенство в случае совпадение сравниваемых полей) если не переопределен то сравниваются ссылки



Клонирование (наследование клонирует глубоко, а ассоциации поверхностно(называется частичное или неполное клонирование))

(еще называется прототипирование) КЛОНИРУЕТСЯ ТОЛЬКО ГРАФ НАСЛЕДОВАНИЕ

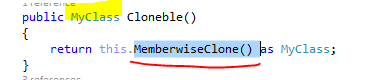
Создал новый объект в памяти (клонировал)

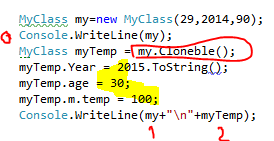
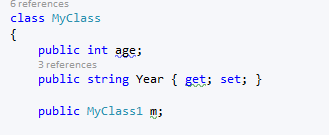
Пример – ГЛУБОКОЕ КЛОНИРОВАНИЕ (создается новый объект в памяти)



Метод MemberwiseClone() protected, в связи с этим его необходимо либо переопределять в классе либо создавать свой метод и внутри него вызвать MemberwiseClone

Который возвращает экземпляр типа данного класса

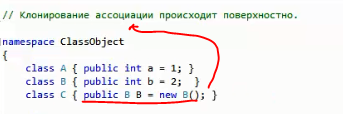


При клонировании типы значений копируются (в данном примере поле и свойство), а ссылочные копируют адрес на объект, в связи с этим оба объекта будут ссылаться на одни и те же поля 

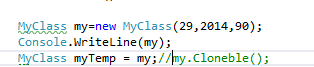
Результат 

Клонирование ассоциаций происходит поверхностно (вложенных ссылочных типов)

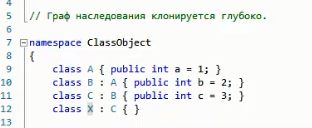
копируют адрес на объект

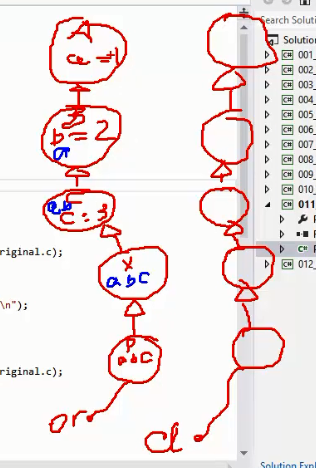


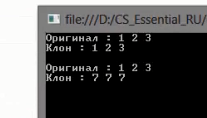
Пример - ПОВЕРХНОСТНОЕ КЛОНИРОВАНИЕ

Ссылаемся на один и тот же объект (копируется ссылка на объект)

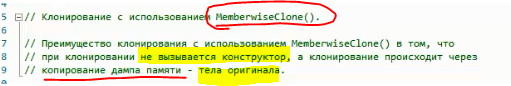
При клонировании создается цепочка новых объектов в памяти наследуемые друг от друга





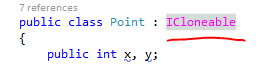


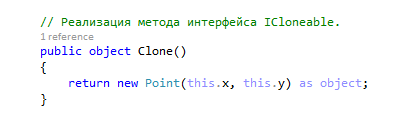
MemberwiseClone() - Копирует байты памяти из одной области памяти в другую, не вызывая конструктор! А при клонировании, когда создается новый экземпляр класса, может потребоваться гораздо больше времени



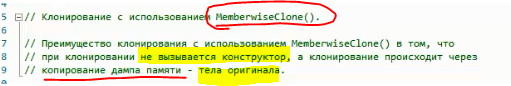
Интерфейс ICloneable

Класс реализовавший данный интерфейс должен реализовать один метод Clone()









Пример ГЛУБОКОГО КЛОНИРОВАНИЯ

При помощи метода MemberwiseClone() клонируем объект, а после на свойстве создаем новый экземпляр типа IdInfo

