

Основы программирования

Состав класса

Индексаторы

Индексаторы

- Индексатор – это разновидность свойства.
- Применяется, как правило, в том случае, если в классе описано скрытое поле – массив.
- Индексатор позволяет обращаться к элементу массива по индексу, записываемому у самого объекта, без явного обращения к массиву внутри объекта. То есть индексатор позволяет интерпретировать объект, содержащий массив, как просто массив.

Синтаксис

```
[атрибуты]    [модификаторы]    тип_возврата  
this(список параметров)  
{  
    [атрибуты]    [модификаторы] get  
    {  
        код доступа для чтения элемента поля-массива;  
    }  
    [атрибуты]    [модификаторы] set  
    {  
        код доступа для записи элемента поля-массива;  
    }  
}
```

...

- Блоки `get` и `set` могут иметь разные модификаторы доступа, не расширяющие область видимости всего индексатора.
- Если для блоков модификаторы не указываются, то на блоки распространяется спецификатор доступа индексатора.
- Также как и в свойстве, в индексаторе может отсутствовать либо блок `set`, либо блок `get`, но не оба сразу.

- «тип_возврата» – это тип элементов массива.
- Список параметров – индексы элементов в массиве и(или) необходимые дополнительные параметры.
- Индексатор может быть перегруженным, то есть, если в классе описывается несколько индексаторов, то все они должны иметь разные списки параметров.

Применение индексаторов

В индексаторах, как и в свойствах, содержатся:

- проверки допустимости переданных значений индексов,
- дополнительная проверка данных,
- проверки прав доступа.

В общем случае в индексаторе могут содержаться любые операторы.

Пример

```
class MyArray
{
    int[] ar; //объявление приватного поле-массива

    public MyArray(int len) {
        ar = new int[len]; //создание массива
    }

    public int Length {
        get {
            return ar.Length;    }
        }
    }
}
```

Пример

```
public int this[int i]      {  
    get {  
        if (i >= 0 && i < ar.Length) return ar[i];  
        else return null;  
    }  
    set {  
        if (i >= 0 && i < ar.Length) ar[i] = value; }  
    }
```


Пример

...

```
public static void Main() {  
    MyArray my = new MyArray(10);  
    for (int i = 0; i < my.Length; i++)      {  
        my[i] = i * 5;          /*обращение к объекту  
                                как к массиву, а без  
                                свойства было бы так  
my.arr[i] */  
    }  
  
    for (int i = 0; i < my.Length; i++)      {  
        Console.Write(my[i]+" "); /*обращение к объекту  
                                как к массиву */  
    }  
}
```

Имитация поля-массива

...

```
class MyArray {
```

...

```
    public int this[int i]      {
```

```
        get {
```

```
            return i;  }
```

```
        /*индексатор возвращает в качестве элемента  
        массива переданный параметр*/
```

```
    }
```

...

```
}
```

...

Рекомендуемые источники

1. Индексаторы

<https://metanit.com/sharp/tutorial/4.10.php>