



موضوع مقاله : ایندکس ها در C#

تهیه کننده : لیلا موثقی امینی

بعضی اوقات یک کلاس دارای اطلاعاتی است که یک برنامه از آنها به عنوان لیستی از اعضا استفاده می کند. اینگونه کلاسها می توانند خصوصیات ویژه ای به نام خصوصیات `indexer` یا شافص دار تعریف کنند که امکان دسترسی به عناصر لیست را همانند `[]` آرایه فراهم نمایند. طبق قوانین آرایه ها در `C#`، بایستی شماره شافص یک مقدار صمیم باشد. مزیت خصوصیات شافص دار در این است که برنامه نویس می تواند هم شافص صمیم تعریف کند و هم شافص غیر صمیم. برای مثال برنامه نویس می تواند به کد سرویس گیرنده اجازه دهد تا با استفاده از رشته ای به عنوان شافص که نشان دهنده اسامی آیتم داده یا شافص هستند، مبادرت به دستکاری داده ها نماید. به هنگام کار با عناصر آرایه رایج در `C#`، عملگر شافص مبادرت به برگشت دادن همان نوع داده (نوع آرایه) می کند. فصیصه های شافص دار بسیار انعطاف پذیر بوده و می توانند هر نوع داده ای را برگشت دهند، حتی اگر نوع یکی از داده ها با داده های دیگر تفاوت داشته باشد.

اگر چه نمونه استفاده از ساب اسکریپت خصوصیات شافص دار همانند یک آرایه عادی است، اما شافص ها به عنوان خصوصیات کلاس تعریف شده اند بر خلاف خصوصیات عادی، که برنامه نویس می تواند یک نام دلخواه برای فصیصه انتخاب کند، شافص ها بایستی توسط کلمه کلیدی `this` تعریف شوند.

در حالت کلی شافص ها به فرم زیر هستند:

```

accessModifier returnType this [IndexType1 name1,IndexType2 name2,...]
{
    get
    {
        //use name1,name2,...here to get data
    }
    set
    {
        //use name1,name2,...here to set data
    }
}

```

پارامترهای IndexType در درون براکت ها ( [ ] ) مشخص می شوند ، از طریق توابع کمک کننده get و set در دسترس قرار می گیرند. این توابع نمونه استفاده از شایخص یا شایفص ها در بازیابی یا تغییر داده ، عضو مقتضی را مشخص می کنند . همانند خصوصیات ، شایفص گذار get بایع مقداری از نوع returnType را برگشت دهد و تابع set می تواند به طور ضمنی از پارامتر value برای اشاره به مقداری که باید به عنصر تفصیص داده شود استفاده کند توجه کنید که اعلان indexer به صورت استاتیک یک فطای نموی به دنبال فواهد داشت

استفاده از indexer در برنامه :

برنامه زیر حاوی دو کلاس است ، کلاس Box نشان دهنده یک جعبه با طول ، عرض و ارتفاع ، و

کلاس BoxTest برای نمایش قابلیت indexer در کلاس Box .

```

public class Box
{
    private string[] names = { "length", "width", "height" };
    private double[] dimensions = new double[3];
    public Box(double length,double width,double height)
    {
        dimensions[0]= length;
        dimensions[1] = width;
        dimensions[2] = height;
    }
    public double this[int index]
    {
        get
        {
            if ((index<0)|| (index>=dimensions.Length))
                return -1;
            else
                return dimensions[index];
        }
        set
        {
            if (index>=0&&index<dimensions.Length)
                dimensions[index]=value;
        }
    }
    public double this[string name]
    {
        get
        {
            int i = 0;
            while ((i < names.Length) && (name.ToLower()
!= names[i]))
                i++;
            return (i == names.Length) ? -1 :
dimensions[i];
        }
        set
        {
            int i = 0;
            while ((i < names.Length) && (name.ToLower()
!= names[i]))
                i++;
            if (i != names.Length)
                dimensions[i] = value;
        }
    }
}

```

```

public class BoxTest
{
    public static void Main(string[] args)
    {
        Box box = new Box(30, 30, 30);
        Console.WriteLine("Created a box with the
dimensions:");
        Console.WriteLine("box [0]={0}", box[0]);
        Console.WriteLine("box [1]={0}", box[1]);
        Console.WriteLine("box [2]={0}", box[2]);
        Console.WriteLine("\nSetting box[0] to 10...\n");
        box[0] = 10;
        Console.WriteLine("Setting box [\"width\"] to
20...\n");
        box["width"] = 20;
        Console.WriteLine("Now the box has the
dimensions:");
        Console.WriteLine("box[\"length\"]={0}",
box["length"]);
        Console.WriteLine("box[\"width\"]={0}",
box["width"]);
        Console.WriteLine("box[\"height\"]={0}",
box["height"]);
    }
}

```

اعضای داده private از کلاس Box ، یک آرایه به نام names از نوع string است که

ماوی اسامی ("width" و "length" و "height") برای ابعاد یک جعبه و آرایه

dimensions از نوع double است ، که ماوی سائز هر بعد است . هر عنصر در آرایه

names متناظر با یک عنصر در آرایه dimensions است .

Box دو indexer تعریف کرده است که هر کدام یک مقدار double برگشت می دهند که

نشان دهنده سائز ابعاد مشخص شده توسط پارامتر indexer است . همانند متدها ،

indexer ها هم می توانند سربارگذاری شوند . اولین indexer از یک شامص int برای پر

کردن یک عنصر در آرایه dimensions استفاده کرده است. دومین indexer از یک شفافص string که نشان دهنده نام بعد برای دستکاری کردن یک عنصر در آرایه dimensions استفاده کرده است. اگر تابع get هر indexer با شفافص نامعتبری مواجه شود عدد 1- را برگشت خواهد داد. هر تابع set متعلق به indexer مقداری به عنصر متناظر در آرایه dimensions تفصیص می دهد در صورتی که شفافص مشخص شده معتبر باشد، معمولاً در صورتی که indexer یک شفافص نامعتبر دریافت می کند با یک استثنا مواجه می شوید

دقت کنید که indexer دریافت کننده یک آرگومان رشته ای از یک عبارت while برای جستجوی رشته مطابق در آرایه names استفاده کرده است. اگر مطابقتی را پیدا کند، indexer عنصر متناظر در آرایه dimensions را دستکاری می کند.

کلاس BoxTest مبادرت به دستکاری اعضا داده private از کلاس Box از طریق indexer آن می کند. متغیر مملی box اعلان و با نمونه ای از کلاس جدید Box مقداردهی شده است. از سازنده کلاس Box برای مقداردهی box با ابعاد 30 و 30 و 30 استفاده کرده ایم. فطوط شروع شده با Console هایی که بعد از مقداردهی box در برنامه لفاظ شده، در واقع از indexer اعلان شده با پارامتر int استفاده کرده است تا سه بعد جعبه را بدست آورده و آنها را با WriteLine به نمایش درآورد. رابطه box[0] به طور ضمنی تابع get را برای بدست آوردن مقدار متغیر نمونه private از dimensions[0] فراخوانی کرده

است . به همین ترتیب تفصیص صورت گرفته `box[0]` که مقداردهی شده است به طور  
ضمنی تابع `set` را فراخوانی کرده است . تابع `set` به طور ضمنی مبادرت به تنظیم مقدار  
پارامتر آن با 10 کرده ، سپس `dimensions[0]` را با `value(10)` تنظیم می کند . فطوح  
پایانی برنامه عملکرد مشابهی را با استفاده از سربرگذاری `indexer` با یک پارامتر `string` در  
دستکاری خصوصیات `Box` دارند .

به طور خلاصه کلاس `Box` نشان دهنده یک جعبه با طول ، عرض و ارتفاع ارائه شده `indexer`  
است و در کلاس `BoxTest` ، `indexer` دسترسی به اعضای شی را فراهم می آورد