زنگ سی‌شارپ – قسمت سوم

نوشته‌ی مسعود درویشیان

لینک مستقیم این مطلب در وب‌تارگت

در قسمت قبل با یک برنامه‌ی ساده شروع کردیم که تنها در خروجی یک پیغام را چاپ می‌کرد، در این قسمت قصد داریم با متغیرها (variable)، value type و عملگرها (operator) بیشتر آشنا شویم.

شاید هیچ سازه‌ای به‌اندازه‌ی متغیرها برای یک زبان برنامه‌نویسی اهمیت نداشته باشد، متغیر در واقع مکانی در حافظه است که می‌توان یک مقدار را به آن اختصاص داد. به‌این دلیل به آن متغیر می‌گویند که مقدار آن می‌تواند در طول اجرای برنامه تغییر کند. به‌عبارت دیگر محتوای متغیرها قابل تغییر هستند و همیشه ثابت نمی‌مانند.

visual studio را اجرا کنید و یک پروژه جدید بسازید اگر این کار را نمی‌توانید انجام دهید حتماً قسمت قبلی را مطالعه بفرمایید.

به برنامه زیر دقت کنید، این برنامه دو متغیر را که اسم آن‌ها x و y است می‌سازد:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Example
{
    class Program
    {
        static void Main(string[] args)
        {
            int x;
            int y;

            x = 100; // this assign 100 to x

            Console.WriteLine("X Contains " + x);

            y = x / 2;

            Console.WriteLine("y contains x / 2: ");
            Console.WriteLine(y);
        }
    }
}
```

وقتی برنامه را اجرا کردید (توسط Ctrl + F5) خروجی زیر برای شما نمایش داده می‌شود:

```
x contains 100
y contains x / 2: 50
```

این برنامه شامل چندین مفهوم جدید است که در ادامه به شرح آن‌ها می‌پردازیم:

```
int x;
```

اعلام یک متغیر که نام آن x و نوع آن عدد صحیح (integer) است. در سی‌شارپ ، همه‌ی متغیرها پیش از این‌که مورد استفاده قرار بگیرند ابتدا باید اعلام شوند. علاوه بر این، نوع مقداری را که متغیر در خود نگه می‌دارد باید مشخص شود. در این مورد x می‌تواند مقادیر integer (عدد صحیح) را در خود نگه دارد. در سی‌شارپ، برای اعلام یک متغیر از نوع عدد صحیح (integer) قبل از نام آن از کلمه‌کلیدی int استفاده می‌کنند. پس int x یک متغیر را اعلام می‌کند که اسم آن x و نوع آن integer (عدد صحیح) است.

```
int y;
```

توجه داشته باشید که در این مورد هم یک متغیر از نوع عدد صحیح ساخته می‌شود با این تفاوت که اسم آن فرق می‌کند. به صورت کلی شما برای اعلام یک متغیر از این الگو می‌توانید پیروی کنید:

```
type var-name;
```

که type نوع متغیر و var-name نام متغیر را مشخص می‌کند. علاوه بر int سی‌شارپ از نوع‌های داده‌ای (data type) دیگر هم پشتیبانی می‌کند که کامل به‌شرح آن‌ها خواهیم پرداخت.

```
x = 100;
```

این خط کد مقدار ۱۰۰ را به x اختصاص می‌دهد. در سی‌شارپ، عملگر انتساب یک تک‌مساوی است و بدین صورت عمل می‌کند که یک کپی از مقدار سمت راست خودش را در متغیر سمت چپش قرار می‌دهد. همان‌طور که می‌بینید در این‌جا علامت انتساب باعث شده است که یک کپی از مقدار سمت راست (عدد ۱۰۰) در متغیر سمت چپ (متغیر x) قرار بگیرد.

خط کد بعدی مقدار x را در خروجی نمایش می‌دهد که قبل از آن رشته‌ی " x contains" قرار دارد:

```
Console.WriteLine("x contains " + x);
```

در این خط کد علامت + موجب می‌شود که مقدار x بعد از string (رشته) نمایش داده شود. این روش می‌تواند یک حالت کلی به‌خود بگیرد. با علامت + شما می‌توانید آیتم‌های دیگری را که می‌خواهید در یک WriteLine() داشته باشید، به هم وصل کنید.

خط کد بعدی مقدار x را که بر ۲ تقسیم شده است به y اختصاص می‌دهد:

```
y = x / 2;
```

این خط کد مقدار x را بر ۲ تقسیم و سپس نتیجه‌ی آن را در y ذخیره می‌کند. بنابراین بعد از اجرای این خط، y شامل مقدار ۵۰ می‌شود و مقدار x بدون تغییر باقی می‌ماند. سی‌شارپ مانند تمام زبان‌های برنامه‌نویسی طیف گسترده‌ای از عملگرهای محاسباتی را پشتیبانی را می‌کند. از جمله این عملگرها:

+	Addition
-	Subtraction
*	Multiplication
/	Division

دو خط بعدی برنامه:

```
Console.Write("y contains x / 2: ");
Console.WriteLine(y);
```

دو چیز در این‌جا جدید است: اول Console.Write() است که رشته‌ی "y contains x / 2: " را در خروجی نمایش می‌دهد. تفاوت این دستور با دستور Console.WriteLine() این است که وقتی خروجی بعدی تولید می‌شود از همان خط شروع می‌شود، نه از خط بعدی. دوم این‌که Console.WriteLine() متغیر y را به‌عنوان ورودی گرفته است و مقدار آن را در خروجی نمایش می‌دهد. پس توجه داشته باشید، هنگامی که می‌خواهید مقدار یک متغیر را در خروجی نمایش دهید کافی است اسم آن را در Console.WriteLine() یا Console.Write() وارد کنید. همان‌طور که مشاهده می‌کنید مقدار y در در جلوی رشته‌ی y contains x / 2: نمایش داده می‌شود زیرا برای چاپ این رشته از Console.Write() استفاده کردیم ولی اگر از Console.WriteLine() استفاده می‌کردیم مقدار y در خط بعدی نمایش داده می‌شد.

یک نکته‌ی جالب دیگر در مورد اعلام متغیرها این است که شما می‌توانید دو یا بیشتر از دو متغیر را در یک تعریف متغیر اعلام کنید. فقط کافی است اسم متغیرها را با کاما از هم جدا کنید، برای مثال متغیرهای y و x را از این روش هم می‌توان اعلام کرد:

```
int x, y;
```

همچنین نیازی نیست که ابتدا متغیر را تعریف کنید و بعد به آن مقدار بدهید، می‌توانید این کار را هم‌زمان با تعریف متغیر انجام دهید. بدین صورت:

```
int x = 100;
```

تا اینجا ما فقط از اعداد صحیح برای محاسبه استفاده کردیم، مثل عدد ۱۸ یا ۲۲ و غیره. برای استفاده از اعداد اعشاری مثل ۲۲/۵ یا ۱۵/۶ باید از نوع عددی **float**، **double** یا **decimal** استفاده کنیم. نوع عددی **float** برای مقادیر ممیز شناور با دقت کمتر و نوع عددی **double** برای مقادیر ممیز شناور با دقت بیشتر است به‌طوری که دقت **double** دو برابر **float** است. **Decimal** یک **data type** ۱۲۸ بیتی است که در مقایسه با **float** و **double** از دقت بیشتر و بازه‌ی کمتری برخوردار است و این خاصیت آن باعث می‌شود که برای محاسبات پولی و مالی بسیار مناسب باشد.

نوع **float** ۳۲ بیتی است و بازه‌ی اعداد آن را می‌توانید در جدول زیر مشاهده کنید:

Type	Approximate range	Precision
float	$\pm 1.5 \times 10^{-45}$ to $\pm 3.4 \times 10^{38}$	7 digits

نوع **double** ۶۴ بیتی است و محدوده‌ی آن در جدول زیر مشخص است:

Type	Approximate range	Precision
double	$\pm 5.0 \times 10^{-324}$ to $\pm 1.7 \times 10^{308}$	15-16 digits

بازه‌ی تقریبی و دقت **decimal** را در جدول زیر می‌توانید مشاهده کنید:

Type	Approximate Range	Precision
decimal	$\pm 1.0 \times 10^{-28}$ to $\pm 7.9 \times 10^{28}$	28-29 significant digits

سی‌شارپ شامل دو دسته‌بندی برای **data type** (نوع اطلاعات) است:

- **Value types**
- **Reference types**

تفاوت بین این دو، مقداری است که آن متغیر دارد. برای **value type**، متغیر یک مقدار واقعی را در خود نگه می‌دارد مثل: ۲۲ یا ۳۴۵/۶ اما برای **Reference type**، متغیر یک ارجاع به **Value** را در خود نگه می‌دارد که این اکثراً در کلاس مورد استفاده قرار می‌گیرد. در مورد **Reference type** بعداً بیشتر صحبت خواهیم کرد، فعلاً به **value type** می‌پردازیم.

در جدول زیر لیست کاملی از تمام **Value type** ها را می‌بینید:

Type	Meaning
bool	Represents true/false values
byte	8-bit unsigned integer
char	Character
decimal	Numeric type for financial calculations
double	Double-precision floating point
float	Single-precision floating point
int	Integer
long	Long integer
sbyte	8-bit signed integer
short	Short integer
uint	An unsigned integer
ulong	An unsigned long integer
ushort	An unsigned short integer

سی‌شارپ همچنین ۹ **integer type** را تعریف می‌کند که **char**، **byte**، **sbyte**، **short**، **ushort**، **int**، **uint**، **long** و **ulong** هستند. نوع **char** اصولاً به نمایندگی از کاراکتر استفاده می‌شود که در مورد آن بعداً صحبت خواهیم کرد:

Type	Width in Bits	Range
byte	8	0 to 255
sbyte	8	-128 to 127
short	16	-32,768 to 32,767
ushort	16	0 to 65,535
int	32	-2,147,483,648 to 2,147,483,647
uint	32	0 to 4,294,967,295
long	64	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
ulong	64	0 to 18,446,744,073,709,551,615

در قسمت بعدی با چندین مثال به تشریح کامل‌تر مباحثی که در این‌جا به صورت تئوری بیان شد می‌پردازیم. هم‌چنین فراموش نکنید که حتماً سوالات و مشکلات خود را در هر قسمت بیان کنید تا موضوع برای شما و دیگر خوانندگان عزیز قابل فهم‌تر شود.

ضمناً سعی کنید که حتماً اصطلاحات انگلیسی استفاده شده در این مقالات را یاد بگیرید چراکه برنامه‌نویسی اصطلاحات زیادی دارد و اکثراً واژه‌ی معادل فارسی ندارند یا در صورت داشتن، معادل فارسی آن‌ها آن‌چنان که باید مناسب نیست و ما ترجیح می‌دهیم که از واژه و اصطلاحات اصلی آن‌ها استفاده کنیم.

کلیه حقوق مادی و معنوی برای وب‌سایت وب‌تارگت محفوظ است.
استفاده از این مطلب در سایر وب‌سایت‌ها و نشریات چاپی تنها با ذکر و درج لینک منبع مجاز است.