زنگ سی شارپ – قسمت پنجم

نوشته‌ی مسعود درویشیان

لینک مستقیم این مطلب در وب‌تارگت

در قسمت قبل در مورد Floating-Point Data Types و چگونگی دریافت مقدار از کاربر و چند نکته‌ی دیگر، توضیحاتی داده شد. در این قسمت به توضیح دیتا تایپ‌های Char و Bool خواهیم پرداخت. کاراکتر و بولین دو بحث پایه‌ای و مهم در برنامه‌نویسی هستند که کابردهای اساسی و مهمی دارند.

Characters

در سی‌شارپ از دیتاتایپ char برای نگه‌داری هر کاراکتری می‌توان استفاده کرد. در متغیری از جنس char شما می‌توانید کاراکترهای تکی (Single Character) را نگه‌داری کنید. سی‌شارپ بر اساس استاندارد یونی‌کد از ۱۶ بیت برای نمایش کاراکترها استفاده می‌کند، در صورتی که اکثر زبان‌های دیگر از ۸ بیت استفاده می‌کنند، مثل C++ و این می‌تواند مشکل‌ساز شود زیرا بعضی از حروف الفبایی را (مثل زبان چینی) نمی‌توان در ۸ بیت نمایش داد. این ویژگی سی‌شارپ باعث می‌شود بتوانید مقادیر جالبی را با فرمت یونی‌کد تولید کنید. برای مثال حروف الفبای خارجی که برای نمایش آن‌ها باید از کاراکتر استفاده کرد مثل حروف یونانی، عبری، چینی و آلمانی همچنین علامت‌های خاص از جمله علامت‌های پول‌های خارجی، علامت‌های ریاضی، اشکال هندسی و غیره که در صفحه‌کلید استاندارد موجود نیستند اما با استاندارد یونی‌کد قابل دسترسی هستند.

مقادیر کاراکتر را باید بین علامت single quotation (سینگل کوتیشن ‘ ’) قرار داد، به‌دلیل این‌که کامپیوتر برای نگه‌داری مقادیر character و integer تفاوت قائل می‌شود.

برای مثال هر دو عبارت زیر درست هستند:

```
char aCharValue = '9';
int aNumValue = 9;
```

در صورتی که هر دو عبارت زیر نادرست هستند:

```
char aCharValue = 9;
int aNumValue = '9';
// error, won't work
```

یک عدد می‌تواند یک کاراکتر باشد، که در این‌صورت باید بین **single quotation** قرار بگیرد و نوع آن‌هم **char** باشد. همچنین یک حرف الفبایی نمی‌تواند در متغیری از نوع عددی قرار بگیرد.

چند نمونه:

```
char myInitial = 'J';
char percentSign = '%';
char numThatIsAChar = '9';
```

کاراکترها را می‌توان در خروجی نمایش داد:

```
static void Main(string[] args)
{
    char ch;
    ch = 'X';
    Console.WriteLine("This is a ch: " + ch);
}
```

شما می‌توانید هر کاراکتری از جمله کاراکترهایی مثل **backspace** و **tab** را ذخیره کنید. برای ذخیره‌ی این نوع کاراکترها **(Escape Sequence)** از دو علامت استفاده می‌شود که همیشه اولین آن‌ها علامت **backslash** است. این جفت علامت، یک کاراکتر را نمایش می‌دهند. برای مثال کد زیر، کاراکتر **backspace** و کاراکتر **tab** را به ترتیب در متغیرهای **aBackSpaceChar** و **aTabChar** ذخیره می‌کند:

```
char aBackspaceChar = '\b';
char aTabChar = '\t';
```

در این مثال، **escape sequence** یک مقدار را برای هر کاراکتر (یک **backspace** و یک **tab** به‌جای حروف **b** و **t**) نشان می‌دهد.

جدول زیر برخی از **escape sequence** های رایج در سی‌شارپ را نشان می‌دهد.

Escape Sequence	Character Name
\'	Single quotation mark
\"	Double quotation mark
\\	Backslash
\0	Null
\a	Alert
\b	Backspace
\f	Form feed
\n	Newline
\r	Carriage return
\t	Horizontal tab
\v	Vertical tab

Common escape sequences

متغیری از جنس **char** تنها یک کاراکتر را در خود نگه می‌دارد. برای این‌که رشته‌ای از کاراکترها را داشته باشید (مثل نام شخصی خودتان) باید از **string** استفاده کنید. در طول انجام مثال‌ها با نحوه‌ی استفاده از **char** و این‌که در چه مواردی استفاده دارد بیشتر آشنا خواهید شد.

The bool Type

دیتا تایپ **bool** یک منطق بولی (بولین - **boolean**) است که بر پایه‌ی این دو مقدار است: **false** و **true**. یک متغیر از جنس **int** می‌تواند میلیون‌ها مقدار مختلف در زمان‌های مختلف را در خود ذخیره کند ولی متغیری از جنس **bool** تنها می‌تواند یکی از این دو مقدار را در خود نگه دارد: **true** یا **false**. برای اعلام یک متغیر بولین باید از کلمه‌کلیدی **bool** استفاده کنید.

مثال زیر دو مقدار را در دو متغیر از جنس **bool** ذخیره می‌کند:

```
bool isItMonday = false;
bool areYouTired = true;
```

شما همچنین می‌توانید بر اساس نتیجه‌ی مقایسه‌ی دو مقدار، مقدار **true** یا **false** را در یک متغیر بولین قرار دهید. عملگرهای مقایسه‌ای دو آیتم را با هم مقایسه می‌کنند، یک عبارت که شامل عملگر مقایسه‌ای است، یک مقدار بولین دارد.

جدول زیر ۶ عملگر مقایسه‌ای که سی‌شارپ پشتیبانی می‌کند را نشان می‌دهد:

Operator	Description	true Example	false Example
<	کوچکتر	3 < 8	8 < 3
>	بزرگتر	4 > 2	2 > 4
==	مساوی	7 == 7	3 == 9
<=	کوچکتر مساوی	5 <=5	8 <= 6
>=	بزرگتر مساوی	7 >= 3	1 >= 2
!=	مخالف (نامساوی)	5 != 6	3 != 3

Comparison operators

هنگامی که شما از عملگرهای مقایسه‌ای که دارای دو علامت هستند (=>، =<، !=، ==) استفاده می‌کنید، نمی‌توانید بین دو علامت از **space** استفاده کنید و توجه داشته باشید که علامت = همیشه در سمت راست است.

به مثال زیر توجه کنید:

```
static void Main(string[] args)
{
    bool isSixBigger = 6 > 5; // Value stored would be true
    bool isSevenSmallerOrEqual = 7 <= 4; // Value stored would be false

    Console.WriteLine(isSixBigger);
    Console.WriteLine(isSevenSmallerOrEqual);
}
```

در این مثال دو عدد با هم مقایسه می‌شوند و حاصل آن‌ها که یا **true** است یا **false**، در متغیر بولین قرار داده می‌شود. هرچند این روش مقداردهی به متغیرهای بولین تا حدودی بلا استفاده است ولی برای مثال و یادگیری در این‌جا مناسب است. استفاده از مقادیر بولین زمانی معنای بهتری پیدا می‌کند که شما از متغیرها به‌جای اعداد ثابت در مقایسه‌ها استفاده کنید.

به مثال زیر توجه کنید:

```
static void Main(string[] args)
{
    int hoursWorked = 45;
    int annualIncome = 80000;

    bool doesEmployeeReceiveOvertime = hoursWorked > 40;
    bool isEmployeeInHighTaxBracket = annualIncome > 100000;

    Console.WriteLine(doesEmployeeReceiveOvertime);
    Console.WriteLine(isEmployeeInHighTaxBracket);
}
```

در بولین اول این برنامه متغیر **hoursWorked** با مقدار ثابت ۴۰ مقایسه می‌شود، اگر مقدار **hoursWorked** کم‌تر از ۴۰ باشد پس مقدار متغیر **doesEmployeeReceiveOvertime** برابر با **false** می‌شود، در غیر این‌صورت برابر با **true** می‌شود.

در بولین بعدی، متغیر **annualIncome** باید مقداری بیشتر از ۱۰۰۰۰۰ را داشته باشد تا متغیر **isEmployeeInHighTaxBracket** برابر با **true** شود.

نکته‌ی مهم: توجه داشته باشید که علامت انتساب یک "تک‌مساوی" است و علامت مقایسه‌ی دو مقدار "دومساوی" است.

به مثال زیر توجه کنید:

```
using System;
using System.Collections.Generic;
```

```
using System.Linq;
using System.Text;

namespace test
{
    class Program
    {
        static void Main(string[] args)
        {
            bool b;
            b = false;
            Console.WriteLine("b is " + b);
            b = true;
            Console.WriteLine("b is " + b);

            Console.WriteLine("10 > 9 is " + (10 > 9));
            Console.WriteLine("10 == 9 is " + (10 == 9));
        }
    }
}
```

خروجی:

```
B is False
B is True
10 > 9 is True
10 == 9 is False
```

همین‌طور که می‌بینید هنگامی که مقدار بولین در خروجی توسط Console.WriteLine() نمایش داده می‌شود، مقدار آن یا به‌صورت True نمایش داده می‌شود یا False. همین‌طور (10 > 9) و (10 == 9) شامل مقدار بولی هستند.

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.