

کلیدها یا کاراکترهای کنترلی که در [ویکی پدیای فارسی](#) به نویسه‌های کنترلی ترجمه شده اند تنها یک خط تعریف دارند:

یک کاراکتر کنترلی، یک نقطه کدی است که به وسیله علائم نوشتاری قابل نمایش نباشد. مانند **Backspace**

تعریف بالا به ما می‌گوید که در یک متن نوشتاری، به غیر از کد حروفی که مشاهده می‌کنیم، کدهای دیگری هم هستن که قابل نمایش نیستند ولی بین متون وجود دارند. شاید شما تعدادی از آنها را بشناسید مثل کدهای 10 و 13 برای خط بعد و اول سطر که به [line feed](#) و [carriage return](#) معروف هستند. در این نوشتار قصد داریم با تعدادی از آنها آشنا شویم.

قبل از آغاز این نوشتار به شما توصیه می‌کنم یک نگاه اجمالی هم که شده بر نوشتار «[داستانی از unicode](#)» داشته باشید تا اطلاعات تکمیلی‌تری از این نوشتار به دست آورید. مبحث کلیدهای کنترلی از زمانی آغاز شد که کدهای اسکی ایجاد شدند و به دو دسته‌ی c0 و c1 تقسیم شدند. خود کدهای اسکی هم بر اساس کدهای تلگراف ایجاد شدند و بسیاری از کلیدهای کنترلی هم از آنجا به استاندارد اسکی پیوستند و برای ارتباط و کنترل دستگاه‌هایی چون چاپگرها و تهیه اطلاعات متا در مورد اطلاعاتی که قرار بود در نوار مغناطیسی ذخیره شوند به کار رفتند. به عنوان نمونه کد 10 به عنوان [line feed](#) در چاپگر، یک خط کاغذ را به سمت داخل می‌کشید و کد 13 هم باعث می‌شد چاپگر به ابتدای سطر بازگردد. البته بیشتر این کاراکترها در پردازش متون به خصوص امروزه استفاده نمی‌شوند و فقط یک سری از آنها رایج هستند؛ مثل دو موردی که در بالا و در همین خط به آنها اشاره شد. دسته‌ی c0 از کد 0 آغاز شده و تا کد 31 ادامه می‌یابد. دو کد بعدی که کدهای Space و DEL هستند در هیچ گروهی قرار نمی‌گیرند. گروه c1 از کدهای 128 آغاز شده و تا 159 ادامه می‌یابند که [جدول این گروه‌ها و کلیدها کنترلی](#) را می‌توانید مشاهده کنید. برای مثال اولین کلید کنترلی که کد آن 0 است به نام نال است که در قدیم هم برای بستن رشته‌ها در زبان سی از آن استفاده می‌کردیم. هر چند به مرور زمان هم تعدادی از همین کلیدهای کنترلی کاربرد خود را از دست دادند و برای آنها شکلک‌هایی چون خنده، قلب، نت موسیقی و ... را قرار دادند ولی گاهی اوقات برنامه نویسی‌ها هنوز در برنامه‌های خود از کد آنها برای کارهایی چون انجام عملیات بیتی استفاده می‌کنند.

استفاده‌های C0

کلیدهای کنترلی این دسته بیشتر برای منظم ساختن متن‌های ساده و همچنین ایجاد ارتباط در پروتکل ارتباطی و دستگاه‌های مختلف به کار می‌رفت؛ ارسال فرمان‌هایی چون آغاز و توقف کار یا انجام عملی خاص توسط هر یک از این کلیدها صورت می‌گرفت. دستگاه‌هایی چون کارت پانچ‌ها، ماشین تایپ و موارد مشابه، از این نوع هستند. با اینکه عمر این دستگاه به سر آمد ولی کلیدهای کنترلی جان سالم به در بردند.

استفاده‌های C1

این دسته در اواخر سال 1970 اضافه شدند و بیشتر برای ارتباط با چاپگر و صفحه‌ی نمایش به کار می‌رفتند؛ مثل پیمایش‌های افقی و عمودی، تعریف ناحیه‌ای برای پر کردن فرم و [Line-Break](#) و کلیدهای انتقالی (شیفت) برای پشتیبانی از کلیدهای کنترلی و قابل چاپ بیشتر. 2 تا از کلیدها هم برای استفاده‌ی خصوصی برنامه نویس کنار گذاشته شدند و 4 تا هم رزرو شده برای استفاده‌ی آینده، تا بعدا استانداردسازی شوند.

کلیدهای کنترلی در سی شارپ

بسیاری از ما از علامت \ در کدهایمان برای قرار دادن کلیدهای کنترلی استفاده می‌کنیم مثل \n که ترکیب دو کد CR و LF است.

برای شناسایی یک کلید کنترلی در سی شارپ از متد ایستای `Char.IsControl` استفاده می‌نماییم. کد زیر در مجموعه‌ی MSDN برای نشان دادن قابلیت این متد نوشته شده است که در طی یک حلقه رنجی از کد پوینت‌ها را بررسی کرده و نتیجه را به صورت شش ستونی در کنسول نمایش می‌دهد. یا [کد مشابه دیگر](#) که بر اساس دسیمال نمایش می‌دهد.

```
using System;

public class ControlChars
{
    public static void Main()
    {
        int charsWritten = 0;

        for (int ctr = 0x00; ctr <= 0xFFFF; ctr++)
        {
            char ch = Convert.ToChar(ctr);
            if (char.IsControl(ch))
            {
                Console.Write(@"\U{0:X4}    ", ctr);
                charsWritten++;
                if (charsWritten % 6 == 0)
                    Console.WriteLine();
            }
        }
    }
}

// The example displays the following output to the console:
//      \U0000    \U0001    \U0002    \U0003    \U0004    \U0005
//      \U0006    \U0007    \U0008    \U0009    \U000A    \U000B
//      \U000C    \U000D    \U000E    \U000F    \U0010    \U0011
//      \U0012    \U0013    \U0014    \U0015    \U0016    \U0017
//      \U0018    \U0019    \U001A    \U001B    \U001C    \U001D
//      \U001E    \U001F    \U007F    \U0080    \U0081    \U0082
//      \U0083    \U0084    \U0085    \U0086    \U0087    \U0088
//      \U0089    \U008A    \U008B    \U008C    \U008D    \U008E
//      \U008F    \U0090    \U0091    \U0092    \U0093    \U0094
//      \U0095    \U0096    \U0097    \U0098    \U0099    \U009A
//      \U009B    \U009C    \U009D    \U009E    \U009F
```

آیا هنوز برنامه نویسی‌ها از کلیدهای کنترلی استفاده می‌کنند؟

این سوال بستگی به برنامه‌ای دارد که شما می‌نویسید. باید گفت هنوز بسیاری از آن‌ها در بسیاری از برنامه‌ها استفاده می‌شوند. مانند بعضی از درایورها برای ارسال اطلاعات به سمت یک قطعه یا دستگاه یا حتی از شما می‌خواهند برنامه‌ای بنویسید که با دستگاه‌های قدیمی ارتباط برقرار کند. برنامه‌هایی که نیاز به کار با رشته‌ها دارند و ...

لیست زیر مشخص می‌کند که کدامیک از کلیدهای کنترلی تا چه اندازه امروزه توسط برنامه نویسان استفاده می‌شوند.

استفاده روزمره‌ای از آن در همه‌ی برنامه‌ها وجود دارد و نیاز به معرفی ندارد.	Null
این کلیدها که 10 عدد هستند شامل SOH , ACK , DLE , ENQ هستند. کاربردشان در انتقال اطلاعات بود ولی امروزه استفاده از آن‌ها به شدت کم شده است و انتقال داده‌ها با سوکت TCP/IP و HTTP و FTP و دیگر پروتکل‌ها به سرانجام رسید و گاهی برای بعضی کاربردهای ویژه استفاده می‌شوند.	Transmission Control
این مورد واقعا کاربردش را از دست داده است. وظیفه قبلی‌اش ارسال یه هشدار یا یک زنگ خطر به کاربر بود. مثلا برای اینکه ماشین تایپ به کاربر هشدار بدهد به آخر خط رسیده است، یک کد BELL به سمت آن ارسال می‌کرد.	BEL
کدهای این دسته عبارتند از BS , CR , FF , HT , HTJ , HTS , IND , LF , NEL , PLD , PLU , RI , VT , VTS هستند که احتمالا مهمترین کدهایی هستند که امروزه از آن‌ها استفاده می‌شود. کاربردشان در فرمت بندی یا قالب بندی متون نوشته شده یا همان کلیدهای قابل چاپ می‌باشد. CR و LF که همیشه معرف حضور ما هستند	Format Effectors

استفاده روزمره‌ای از آن در همه‌ی برنامه‌ها وجود دارد و نیاز به معرفی ندارد.	Null
و بودنشان در سیستم یک امر حیاتی است. HT که همان tab است. BS که همان Backspace است. FF و VT هم که امروزه به ندرت استفاده می‌شوند.	
هنوز برای ارتباط با دستگاه‌های مختلف مثل کار با پورتهای استفاده می‌شوند. کلیدهای معروف آن DC1 و DC3 هستند که به XON و XOFF هم شناخته می‌شوند. یکی از کاربردهای آن.	Device Control
یک نماد جایگزین که استفاده‌ی خود را از دست داده است. موقعیکه نمادی نامعتبر بود یا خطایی رخ می‌داد، این نماد جایگزین آن می‌شد. امروزه بیشتر از علامت ؟ در متون استفاده می‌شود. در یک صفحه کلید استاندارد این کد توسط فشرده شدن Ctrl+Z ارسال می‌شود.	SUB
کاربردی امروزه ندارد. CAN برای کنترل خطا به کار می‌رفت و EM در نوارهای مغناطیسی.	CAN , EM
شامل 4 کلید RS , GS , FS و US می‌شود که برای جداسازی داده‌ها از یکدیگر به کار می‌روند؛ ولی به‌خاطر جایگزینی آن‌ها با اسنادی مثل XML یا دیتابیس‌ها، استفاده از آن‌ها تا حدودی به پایان رسیده است.	Information Separators
همان کلید space است که نیاز به معرفی ندارد و کارش گویای همه چیز هست.	SP
همان کلید Delete است.	DELL
این کلید همان کاراکتر است که در کدهای HTML استفاده می‌شود.	NBSP
علامت - یا Hyphen است که به شدت استفاده از آن کم شده است.	SHY