**ToString()使用方法彙總(C#)**C  
貨幣  
2.5.ToString(“C”)  
￥2.50  
D  
十進位制數  
25.ToString(“D5”)  
00025  
E  
科學型  
25000.ToString(“E”)  
2.500000E 005  
F  
固定點  
25.ToString(“F2”)  
25.00  
G  
常規  
2.5.ToString(“G”)  
2.5

數字  
2500000.ToString(“N”)  
2,500,000.00  
X  
十六進位制  
255.ToString(“X”)  
FF

formatCode 是可選的格式化程式碼字串。（詳細內容請搜尋”格式化字串”檢視）  
必須用”{“和”}”將格式與其他字元分開。如果恰好在格式中也要使用大括號，可以用連續的兩個大括號表示一個大括號，即： “{{“或者”}}”。  
**常用格式舉例：**  
（1） int i=12345;  
this.textBox1.Text=i.ToString();  
//結果 12345（this指當前物件，或叫當前類的例項）  
this.textBox2.Text=i.ToString(“d8”);  
//結果 00012345  
（2） int i=123;  
double j=123.45;  
string s1=string.Format(“the value is {0,7:d}”,i);  
string s2=string.Format(“the value is {0,7:f3}”,j);  
this.textBox1.Text=s1 ;  
//結果 the value is 123  
this.textBox2.Text=s2;  
//結果 the value is 123.450  
（3）double i=12345.6789;  
this.textBox1.Text=i.ToString(“f2”); //結果 12345.68  
this.textBox2.Text=i.ToString(“f6”);  
//結果 12345.678900  
（4）double i=12345.6789;  
this.textBox1.Text=i.ToString(“n”); //結果 12,345.68  
this.textBox2.Text=i.ToString(“n4”); //結果 12,345.6789  
（5）double i=0.126;  
string s=string.Format(“the value is {0:p}”,i);  
this.textBox1.Text=i.ToString(“p”); //結果 12.6%  
this.textBox2.Text=s; //結果 the value is 12.6%  
（6） DateTime dt =new DateTime(2003,5,25);  
this.textBox1.Text=dt.ToString(“yy.M.d”);  
//結果 03.5.25  
this.textBox2.Text=dt.ToString(“yyyy年M月”);  
//結果 2003年5月  
Convert.ToDateTime(“2005/12/22 22:22:22”).ToString(“yyyy/MM/dd HH:mm:ss”)  
“2005/12/22 22:22:22”  
（7） int i=123;  
double j=123.45;  
string s=string.Format(“i:{0,-7},j:{1,7}”,i,j);  
//-7表示左對齊，佔7位  
this.textBox1.Text=s ;  
//結果i:123 ,j: 123.45  
DateTime.ToString()用法詳解  
我們經常會遇到對時間進行轉換,達到不同的顯示效果，預設格式為:2006-6-6 14:33:34  
如果要換成成200606,06-2006,2006-6-6或更多的格式該怎麼辦呢？  
這裡將要用到:DateTime.ToString的方法(String, IFormatProvider)  
**示例：**  
using System;  
using System.Globalization;  
String format=”D”;  
DateTime date=DataTime.Now;  
Response.Write(date.ToString(format, DateTimeFormatInfo.InvariantInfo));  
**結果輸出**  
Thursday, June 16, 2006

**在這裡列出了引數format格式詳細用法**=======================  
格式字元 關聯屬性/說明  
d ShortDatePattern  
D LongDatePattern  
f 完整日期和時間（長日期和短時間）  
F FullDateTimePattern（長日期和長時間）  
g 常規（短日期和短時間）  
G 常規（短日期和長時間）  
m、M MonthDayPattern  
r、R RFC1123Pattern  
s 使用當地時間的 SortableDateTimePattern（基於 ISO 8601）  
t ShortTimePattern  
T LongTimePattern  
u UniversalSortableDateTimePattern 用於顯示通用時間的格式  
U 使用通用時間的完整日期和時間（長日期和長時間）  
y、Y YearMonthPattern  
**下表列出了可被合併以構造自定義模式的模式**  
========================================  
這些模式是區分大小寫的；例如，識別”MM”，但不識別”mm”。如果自定義模式包含空白字元或用單引號括起來的字元，則輸出字串頁也將包含這些字元。未定義為格式模式的一部分或未定義為格式字元的字元按其原義複製。  
格式模式 說明 ：  
d 月中的某一天。一位數的日期沒有前導零。  
dd 月中的某一天。一位數的日期有一個前導零。  
ddd 週中某天的縮寫名稱，在 AbbreviatedDayNames 中定義。  
dddd 週中某天的完整名稱，在 DayNames 中定義。  
M 月份數字。一位數的月份沒有前導零。  
MM 月份數字。一位數的月份有一個前導零。  
MMM 月份的縮寫名稱，在 AbbreviatedMonthNames 中定義。  
MMMM 月份的完整名稱，在 MonthNames 中定義。  
y 不包含紀元的年份。如果不包含紀元的年份小於 10，則顯示不具有前導零的年份。  
yy 不包含紀元的年份。如果不包含紀元的年份小於 10，則顯示具有前導零的年份。  
yyyy 包括紀元的四位數的年份。  
gg 時期或紀元。如果要設定格式的日期不具有關聯的時期或紀元字串，則忽略該模式。  
h 12 小時制的小時。一位數的小時數沒有前導零。  
hh 12 小時制的小時。一位數的小時數有前導零。  
H 24 小時制的小時。一位數的小時數沒有前導零。  
HH 24 小時制的小時。一位數的小時數有前導零。  
m 分鐘。一位數的分鐘數沒有前導零。  
mm 分鐘。一位數的分鐘數有一個前導零。  
s 秒。一位數的秒數沒有前導零。  
ss 秒。一位數的秒數有一個前導零。  
f 秒的小數精度為一位。其餘數字被截斷。  
ff 秒的小數精度為兩位。其餘數字被截斷。  
fff 秒的小數精度為三位。其餘數字被截斷。  
ffff 秒的小數精度為四位。其餘數字被截斷。  
fffff 秒的小數精度為五位。其餘數字被截斷。  
ffffff 秒的小數精度為六位。其餘數字被截斷。  
fffffff 秒的小數精度為七位。其餘數字被截斷。  
t 在 AMDesignator 或 PMDesignator 中定義的 AM/PM 指示項的第一個字元（如果存在）。  
tt 在 AMDesignator 或 PMDesignator 中定義的 AM/PM 指示項（如果存在）。  
z 時區偏移量（” “或”-“後面僅跟小時）。一位數的小時數沒有前導零。例如，太平洋標準時間是”-8″。  
zz 時區偏移量（” “或”-“後面僅跟小時）。一位數的小時數有前導零。例如，太平洋標準時間是”-08″。  
zzz 完整時區偏移量（” “或”-“後面跟有小時和分鐘）。一位數的小時數和分鐘數有前導零。例如，太平洋標準時間是”-08:00″。  
: 在 TimeSeparator 中定義的預設時間分隔符。  
/ 在 DateSeparator 中定義的預設日期分隔符。  
% c 其中 c 是格式模式（如果單獨使用）。如果格式模式與原義字元或其他格式模式合併，則可以省略”%”字元。  
\ c 其中 c 是任意字元。照原義顯示字元。若要顯示反斜槓字元，請使用”\\”。  
只有上面第二個表中列出的格式模式才能用於建立自定義模式；在第一個表中列出的標準格式字元不能用於建立自定義模式。自定義模式的長度至少為兩個字元；例如，  
DateTime.ToString( “d”) 返回 DateTime 值；”d”是標準短日期模式。  
DateTime.ToString( “%d”) 返回月中的某天；”%d”是自定義模式。  
DateTime.ToString( “d “) 返回後面跟有一個空白字元的月中的某天；”d”是自定義模式。  
比較方便的是,上面的引數可以隨意組合,並且不會出錯,多試試,肯定會找到你要的時間格式  
如要得到2005年06月 這樣格式的時間  
可以這樣寫:  
date.ToString(“yyyy年MM月”, DateTimeFormatInfo.InvariantInfo)  
如此類推.  
**下面列出一些Asp.net中具體的日期格式化用法：**  
============================================  
1.繫結時格式化日期方法:

2.資料控制元件如DataGrid/DataList等的件格式化日期方法:  
e.Item.Cell[0].Text = Convert.ToDateTime(e.Item.Cell[0].Text).ToShortDateString();  
3.用String類轉換日期顯示格式:  
String.Format( “yyyy-MM-dd “,yourDateTime);  
4.用Convert方法轉換日期顯示格式:  
Convert.ToDateTime(“2005-8-23”).ToString  
(“yyMMdd”,System.Globalization.DateTimeFormatInfo.InvariantInfo); //支援繁體資料庫  
5.直接用ToString方法轉換日期顯示格式:  
DateTime.Now.ToString(“yyyyMMddhhmmss”);  
DateTime.Now.ToString(“yyyy/MM/dd hh:mm:ss”)  
6.只顯示年月  
DataBinder.Eval(Container.DataItem,”starttime”,”{0:yyyy-M}”)  
7.顯示時間所有部分，包括：年月日時分秒  
DataFormatString='{0:yyyy-MM-dd HH24:mm:ss}’>  
用DateTime.ToString(string format)輸出不同格式的日期

DateTime.ToString()函式有四個過載。一般用得多的就是不帶引數的那個了。殊不知，DateTime.ToString(string format)功能更強大，能輸出不同格式的日期。以下把一些情況羅列出來，供大家參考。有些在MSDN上有的就沒有列出來了。  
1. y代表年份，注意是小寫的y，大寫的Ｙ並不代表年份。  
2. M表示月份。  
3. d表示日期，注意D並不代表什麼。  
4. h或H表示小時，h用的是12小時制，H用的是24小時制。  
5. m表示分鐘。  
6. s表示秒。注意S並不代表什麼。  
格式  
輸出  
示例  
年  
y  
7  
string yy = DateTime.Now.ToString(“y-MM”)  
yy=”7-05″  
yy  
07  
string yy = DateTime.Now.ToString(“yy-MM”)  
yy=”07-05″  
yyy或更多的y  
1984  
string yy = DateTime.Now.ToString(“yyyy”);  
yy=”2007″  
月  
M  
5.  
string mon = DateTime.Parse(“1984-05-09”)ToString(“yyyy-M”)  
mon = “1984-5”  
MM  
05.  
string mon = DateTime.Parse(“1984-05-09”)ToString(“MＭ”)  
mon = “05”  
MMM  
如果是中文版的作業系統，則會輸出：五月.  
如果是英文作業系統，則輸入月份前三個字母的簡寫：May  
string mon = DateTime.Parse(“2006-07-01”).ToString(“MMM”)  
英文版作業系統：Jul  
中文版作業系統：七月  
MMMM或更多的Ｍ  
如果是中文版的作業系統，則會輸出：五月.  
如果是英文作業系統，則輸入月份的全寫  
string mon = DateTime.Parse(“2006-07-01”).ToString(“MMM”)  
英文版作業系統：July  
中文版作業系統：七月  
日期或星期  
d  
9  
string dd= DateTime.Parse(“1984-05-09”)ToString(“d”)  
dd= “9”  
dd  
09  
string dd= DateTime.Parse(“1984-05-09”)ToString(“dd”)  
dd= “09”  
ddd  
如果是中文版的作業系統，則會輸出星期，如星期三。.  
如果是英文作業系統，則輸出星期的簡寫：如  
Wed  
string dd = DateTime.Parse(“2006-07-01”).ToString(“ddd”)  
英文版作業系統：Wed  
中文版作業系統：星期三  
dddd或更多的d  
如果是中文版的作業系統，則會輸出星期，如星期三。.  
如果是英文作業系統，則輸出星期：如  
Wednesday  
string dd = DateTime.Parse(“2006-07-01”).ToString(“dddd”)  
英文版作業系統：Wednesday  
中文版作業系統：星期三  
小時  
h  
小時範圍：1-12  
string hh = DateTime.Now.ToString(“h”);  
hh = 8  
hh或更多的h  
小時範圍：1-12  
string hh = DateTime.Now.ToString(“hh”);  
hh = 08  
H  
小時範圍：0-23  
string hh = DateTime.Now.ToString(“yyyy-H”);  
hh = 2006-8  
HH或更多的H  
小時範圍：0-23  
string hh = DateTime.Now.ToString(“yyyy-HH”);  
hh = 2006-08  
string hh = DateTime.Pare(“2006-7-4 18:00:00”).ToString(“yyyy-HH”);  
hh = 2006-18  
分鐘  
m  
6  
string mm = DateTime.Now.ToString(“yyyy-MM-dd-m”);  
mm = “2006-07-01-6”;  
mm或更多的m  
06  
string mm = DateTime.Now.ToString(“yyyy-MM-dd-mm”);  
mm = “2006-07-01-06”;  
秒

6  
string mm = DateTime.Now.ToString(“yyyy-MM-dd-s”);  
mm = “2006-07-01-6”;  
ss或更多的s  
06  
string mm = DateTime.Now.ToString(“yyyy-MM-dd-ss”);  
mm = “2006-07-01-06”;