

# Converts a Date to a string using relative time.

```
//Microsoft Public License (Ms-PL)
//http://visualizer.codeplex.com/license
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Redwerb.BizArk.Core.DateExt
{
    ///
    /// Provides extension methods for dates.
    ///
    public static class DateExt
    {
        ///
        /// Converts a Date to a string using relative time.
        ///
        /// ///
        public static string ToRelativeTimeString(this DateTime value)
        {
            const int SECOND = 1;
            const int MINUTE = 60 * SECOND;
            const int HOUR = 60 * MINUTE;
            const int DAY = 24 * HOUR;
            const int MONTH = 30 * DAY;

            var ts = DateTime.Now.Subtract(value);
            double seconds = ts.TotalSeconds;

            // Less than one minute
```

```
if (seconds < 1 * MINUTE) return ts.Seconds == 1 ? "one second ago" : ts.Seconds + " seconds ago"; if (seconds < 60 * MINUTE) return ts.Minutes + " minutes ago"; if (seconds < 120 * MINUTE) return "an hour ago"; if (seconds < 24 * HOUR) return ts.Hours + " hours ago"; if (seconds < 48 * HOUR) return "yesterday"; if (seconds < 30 * DAY) return ts.Days + " days ago"; if (seconds < 12 * MONTH) { int months = System.Convert.ToInt32(Math.Floor((double)ts.Days / 30)); return months <= 1 ? "one month ago" : months + " months ago"; } int years = System.Convert.ToInt32(Math.Floor((double)ts.Days / 365)); return years <= 1 ? "one year ago" : years + " years ago"; } } } [/csharp]
```