1,System.Form.Timer

2.System.Threading.Timer

3,System.Timers.Timer.

System.Timers名称空间中的Timer类的构造函数只需要一个时间间隔。经过该时间间隔后应调

用的方法用Elapsed事件指定。这个事件需要一个ElapsedEventHandler类型的委托,这个委托需要的对象和ElapsedEventArgs参数与TimerAction方法相同。AutoSet 属性指定计时器是否重复触发。

如果把这个属性设置为false,事件就只触发一次。调用start()方法允许计时器触发事件。除了调用

Start()方法之外,还可以把Enabled属性设置为true。在后台,start()方法什么也不做。Stop()方法把Enabled属性设置为false,以停止计时器。

Timer 组件是基于服务器的计时器，它使您能够指定在应用程序中引发 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件的周期性间隔。 然后可通过处理这个事件来提供常规处理。 例如，假设您有一台关键性服务器，必须每周 7 天、每天 24 小时都保持运行。 可以创建一个使用 Timer 的服务，以定期检查服务器并确保系统开启并在运行。 如果系统不响应，则该服务可以尝试重新启动服务器或通知管理员。

基于服务器的 Timer 是为在多线程环境中用于辅助线程而设计的。 服务器计时器可以在线程间移动来处理引发的 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件，这样就可以比 Windows 计时器更精确地按时引发事件。

基于 [Interval](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.interval(v=vs.110).aspx) 属性的值，Timer 组件引发 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件。 您可以处理这个事件来执行必要的处理。 例如，假设您有一个联机销售应用程序，它持续向一个数据库投递销售订单。 编译要传送的指令的服务成批处理订单，而不是单独地处理每份订单。 可以使用 Timer 每 30 分钟启动一次批处理。

|  |
| --- |
| **说明** |
| 当 [AutoReset](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.autoreset(v=vs.110).aspx) 设置为 **false** 时，Timer 只在第一个 [Interval](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.interval(v=vs.110).aspx) 过后引发一次 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件。 若要保持以 [Interval](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.interval(v=vs.110).aspx) 时间间隔引发 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件，请将 [AutoReset](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.autoreset(v=vs.110).aspx) 设置为 **true**。 |

在 .NET Framework 2.0 版及更早版本中，Timer 组件捕获 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件的事件处理程序所引发的所有异常并禁止显示这些异常。 在将来的 .NET Framework 版本中，此行为可能会有更改。

如果 [SynchronizingObject](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.synchronizingobject(v=vs.110).aspx) 属性为 **null**，则在 [ThreadPool](http://msdn.microsoft.com/zh-cn/library/system.threading.threadpool(v=vs.110).aspx) 线程上引发 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件。 如果 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件的处理时间比 [Interval](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.interval(v=vs.110).aspx) 长，在另一个 [ThreadPool](http://msdn.microsoft.com/zh-cn/library/system.threading.threadpool(v=vs.110).aspx) 线程上将会再次引发此事件。 在这种情况下，事件处理程序应当是可重入的。

|  |
| --- |
| **说明** |
| 在一个线程调用 [Stop](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.stop(v=vs.110).aspx) 方法或将 [Enabled](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.enabled(v=vs.110).aspx) 属性设置为 **false** 的同时，可在另一个线程上运行事件处理方法。 这可能导致在计时器停止之后引发 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件。 [Stop](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.stop(v=vs.110).aspx) 方法的代码示例演示了一种避免此争用条件的方法。 |

即使 [SynchronizingObject](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.synchronizingobject(v=vs.110).aspx) 不是 **null**，[Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 在 [Dispose](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.dispose(v=vs.110).aspx) 或 [Stop](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.stop(v=vs.110).aspx) 方法调用后或 [Enabled](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.enabled(v=vs.110).aspx) 属性设置为 **false** 后，事件可以发生，这是因为引出 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件的信号总是排队等待执行线程池线程。 解决此争用条件的一种方法是：设置一个标志，通知 [Elapsed](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.elapsed(v=vs.110).aspx) 事件的事件处理程序忽略随后的事件。

如果您使用 Timer 与用户界面元素如窗体或控件，没有放置计时器的该用户界面元素，指定包含 Timer 窗体或控件到 [SynchronizingObject](http://msdn.microsoft.com/zh-cn/library/system.timers.timer.synchronizingobject(v=vs.110).aspx) 属性，以使该事件封送到用户界面线程。

Timer 在运行时是不可见的。