

## Шпаргалка: Компонент Timer в Windows Forms

Основные сведения:

Timer — компонент, позволяющий выполнять действия через определенные интервалы времени

Находится в пространстве имен System.Windows.Forms.

Является невидимым компонентом (не отображается в окне приложения).

Отображается в области компонентов под формой.

Основные свойства:

- Enabled — включает (true) или выключает (false) таймер
- Interval — интервал в миллисекундах между срабатываниями таймера
- Tag — используется для хранения дополнительной информации

Основные методы:

- Start() — запускает таймер (устанавливает Enabled = true)
- Stop() — останавливает таймер (устанавливает Enabled = false)

Основные события:

- Tick — происходит по истечении интервала времени, заданного в свойстве Interval

Пример 1: Создание простых часов.

```
// Объявление и настройка в конструкторе формы
public Form1()
{
    InitializeComponent();

    // Создаем таймер
    timer1 = new Timer();
    timer1.Interval = 1000; // интервал 1 секунда
    timer1.Tick += Timer1_Tick;
    timer1.Start(); // запускаем таймер
}
```

```
// Обработчик события Tick
private void Timer1_Tick(object sender, EventArgs e)
{
    // Обновляем текст метки текущим временем
    labelClock.Text = DateTime.Now.ToString("HH:mm:ss");
}
```

Пример 2: Таймер обратного отсчета

```
// Оставшееся количество секунд для обратного отсчета
private int secondsLeft = 60;

public Form1()
{
    // Инициализация компонентов формы
    InitializeComponent();

    // Установка интервала таймера в 1000 миллисекунд (1 секунда)
    timer1.Interval = 1000;

    // Запуск таймера (устанавливает свойство Enabled в true)
    timer1.Start();
}
```

```
// Обработчик события Tick, вызывается каждый раз по истечении интервала таймера
private void timer1_Tick(object sender, EventArgs e)
{
    // Уменьшаем оставшееся количество секунд на 1
    secondsLeft--;

    // Обновляем текст метки, отображая оставшееся время
    label1.Text = secondsLeft.ToString();

    // Проверяем, закончилось ли время обратного отсчета
    if (secondsLeft <= 0)
    {
        // Останавливаем таймер, чтобы прекратить дальнейшее выполнение обратного отсчета
        timer1.Stop();

        // Отображаем сообщение пользователю о завершении отсчета
        MessageBox.Show("Время вышло!");
    }
}
```

Пример 3: Анимация с помощью таймера

```
private int positionX = 0;

private void AnimationTimer_Tick(object sender, EventArgs e)
{
    // Перемещаем объект по оси X
    positionX += 5;
    pictureBox1.Left = positionX;

    // Если достигли края формы, возвращаем в начало
    if (positionX > this.Width)
    {
        positionX = -pictureBox1.Width;
    }
}
```

Пример 4: Утилизация таймера при нажатии кнопки "стоп"

```
private void buttonStop_Click(object sender, EventArgs e)
{
    // Проверяем, что таймер существует и не был ранее утилизирован
    if (timer1 != null)
    {
        // Останавливаем таймер
        timer1.Stop();

        // Отписываемся от события Tick, чтобы избежать утечек памяти
        timer1.Tick -= timer1_Tick;

        // Освобождаем ресурсы таймера
        timer1.Dispose();

        // Обнуляем ссылку на таймер
        timer1 = null;

        // Обновляем интерфейс, например, меняем текст метки
        label1.Text = "Таймер остановлен";

        // Опционально: можно изменить состояние кнопок
        buttonStop.Enabled = false;
        buttonStart.Enabled = true; // Если есть кнопка для повторного запуска
    }
}
```