

Создание простейшей анимации при помощи компонента Timer в C# и платформе .NET.

Для создания этой программы нам потребуются два компонента: Timer и TrackBar.

Timer является компонентом для запуска действий, повторяющихся через определенный промежуток времени. Хотя он не является визуальным элементом, но его также можно перетащить с Панели Инструментов на форму:

Наиболее важные свойства и методы таймера:

- Свойство **Enabled**: при значении true указывает, что таймер будет запускаться вместе с запуском формы
- Свойство **Interval**: указывает интервал в миллисекундах, через который будет срабатывать обработчик события Tick, которое есть у таймера
- Метод **Start()**: запускает таймер
- Метод **Stop()**: останавливает таймер

TrackBar представляет собой элемент, который с помощью перемещения ползунка позволяет вводить числовые значения.

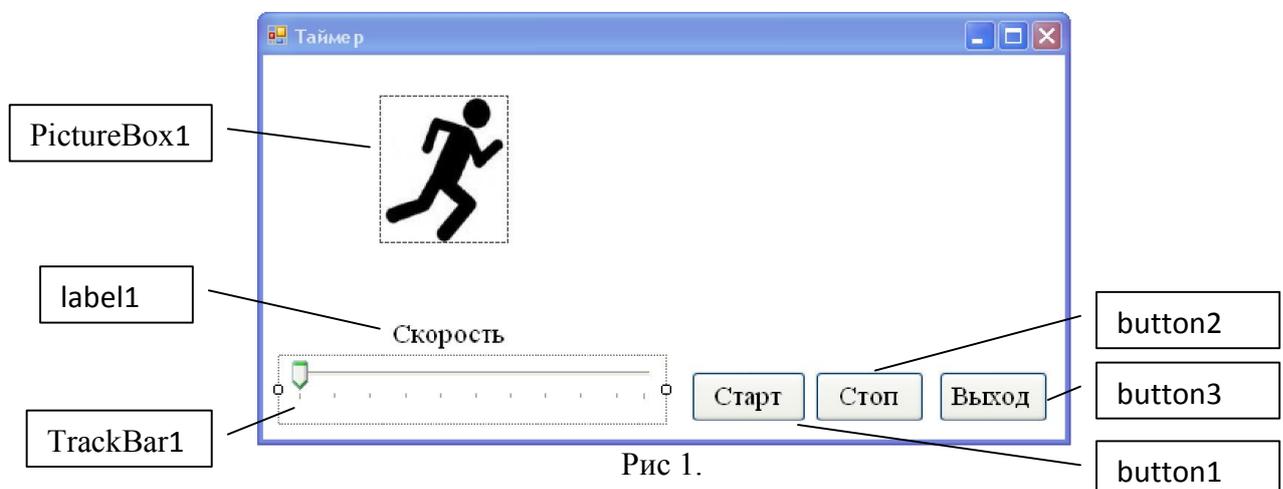
Некоторые важные свойства TrackBar:

- **Orientation**: задает ориентацию ползунка - расположение по горизонтали или по вертикали
- **TickStyle**: задает расположение делений на ползунке
- **TickFrequency**: задает частоту делений на ползунке
- **Minimum**: минимальное возможное значение на ползунке (по умолчанию 0)
- **Maximum**: максимальное возможное значение на ползунке (по умолчанию 10)
- **Value**: текущее значение ползунка. Должно находиться между Minimum и Maximum

Свойство TickStyle может принимать ряд значений:

- None: деления отсутствуют
- Both: деления расположены по обеим сторонам ползунка
- BottomRight: у вертикального ползунка деления находятся справа, а у горизонтального - снизу
- TopLeft: у вертикального ползунка деления находятся слева, а у горизонтального - сверху (применяется по умолчанию)

К наиболее важным событиям элемента следует отнести событие **Scroll**, которое позволяет обработать перемещение ползунка от одного деления к другому. Что может быть полезно, если нам надо, например, устанавливать соответствующую громкость звука в зависимости от значения ползунка, либо какие ни будь другие настройки.





Разместим на форме объекты как показано на рисунке 1, фон формы (BackColor) сделаем белого цвета под фон картинки. Для создания анимации нам потребуются две картинки, человечка бегущего влево и вправо. Их можно сделать из одного рисунка, отразив в графическом редакторе картинку слева на право и сохранив как новый рисунок. Первую картинку загрузим непосредственно в PictureBox. Так как движение у нас начнется слева на право. Что бы разные по размеру картинки, отображались точно по размеру окна, свойство **SizeMode** для PictureBox изменим на **StretchImage**: изображение растягивается или сжимается таким образом, чтобы вписаться по всей ширине и высоте элемента PictureBox.

Разместим на форме не визуальный компонент Timer. Он будет отображаться внизу формы в отдельном окне. Его свойство Interval установим равное 10.



Щелкнув по нему двойным щелчком мыши, получим событие, связанное с таймером. В разделе глобальных переменных опишем целочисленную переменную d, отвечающую за скорость и направление движения.

```
public partial class Form1 : Form
{
    int d = 1;
    public Form1()
```

Само событие должно выглядеть следующим образом:

```
private void timer1_Tick(object sender, EventArgs e)
{
    //изменяем расстояние до края формы
    pictureBox1.Left = pictureBox1.Left + d;
    //если достигнут левый или правый край формы
    if (pictureBox1.Left >= 640 || pictureBox1.Left <= 0)
    {
        //в зависимости от того куда двигались,
        //меняем рисунок
        if (d > 0) pictureBox1.Image = Image.FromFile("Ris2.jpg");
        if (d < 0) pictureBox1.Image = Image.FromFile("Ris1.jpg");
        //изменяем направление движения
        d = -d;
    }
}
```

Рисунки загружаются с диска, по этому они должны располагаться там, где находится исполняемый файл, или к ним должен быть прописан путь.

Для кнопки «Старт» создадим событие запускающее таймер.

```
private void button3_Click(object sender, EventArgs e)
{
    timer1.Enabled = true;
}
```

Для кнопки «Стоп» создадим событие останавливающее таймер.

```
private void button2_Click(object sender, EventArgs e)
{
    timer1.Enabled = false;
}
```

Элемент TrackBar будет отвечать за скорость движения. Настроим его свойства в инспекторе объектов. Maximum установим 50, Minimum равный 1, а TickFrequency (задает частоту делений на ползунке) равным 5.

Двойным щелчком мыши по TrackBar создадим событие, происходящее при изменении положения ползунка.

```
private void trackBar1_Scroll(object sender, EventArgs e)
{
    if (d>=0) d=trackBar1.Value;
    if (d<0) d=-trackBar1.Value;
}
```

Осталась кнопка «Выход»:

```
private void button1_Click(object sender, EventArgs e)
{
    this.Close();
}
```