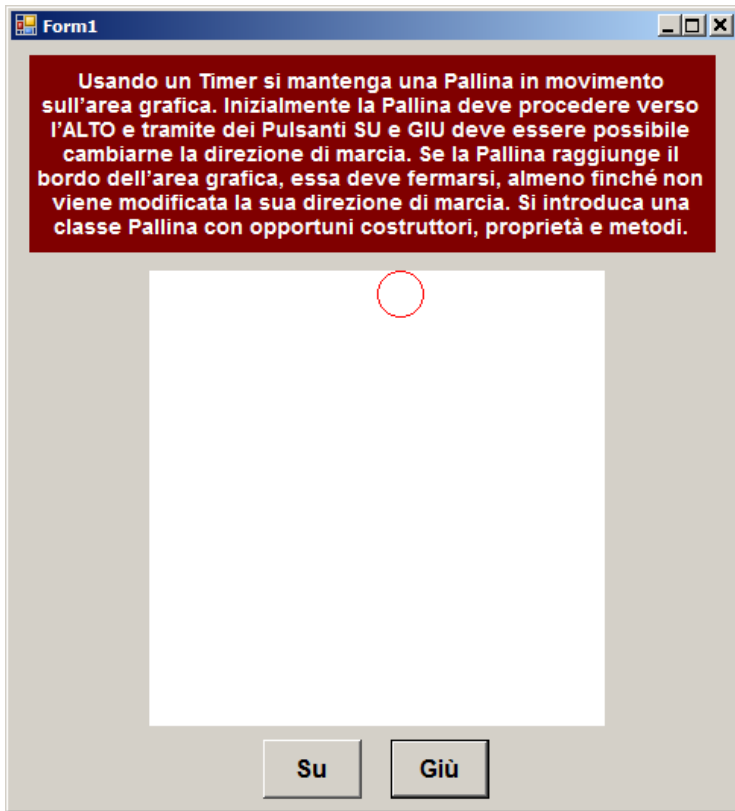


Esempio Pilota:
PallinaCheSiMuove

Definizione della Form



Panel Pn1

Button plsSu, Button plsGiù

Codice della Form

```
using System;
using System.Windows.Forms;

namespace PallinaCheSiMuove
{
    public partial class frmPallinaCheSiMuove : Form
    {
        public frmPallinaCheSiMuove()
        {
            InitializeComponent();
        }

        // Dichiarare un'istanza (oggetto) di classe Pallina ...
        Pallina miaPallina;

        private void frmPallinaCheSiMuove_Load
            (object sender, EventArgs e)
        {
            // Crea l'istanza (oggetto) miaPallina ...
            miaPallina = new Pallina (150, 150, 30);

            // Avvia il Timer per il movimento della pallina ...
            Tmr.Start ();
        }
    }
}
```

```
private void Tmr_Tick(object sender, EventArgs e)
{
    miaPallina.Muovi ( Pnl );

    if ( miaPallina.E_Fuori ( Pnl ) )
        miaPallina.RiposizionaSulBordo ( Pnl );
}
```

```
private void plsSu_Click(object sender, EventArgs e)
{ miaPallina.Direzione = "SU"; }
```

```
private void plsGiù_Click(object sender, EventArgs e)
{ miaPallina.Direzione = "GIU"; }
}
```

Codice della Classe Pallina

```
using System.Drawing;
using System.Windows.Forms;
```

```
namespace PallinaCheSiMuove
{
    class Pallina
    {
```

// --- ATTRIBUTI ...

// _X ed _Y indicano la posizione della pallina ...

```
private int _X;
private int _Y;
```

// _Diam e' il diametro della pallina ...

```
private int _Diam;
```

// _Dir è la direzione di movimento ("SU" o "GIU") ...

```
private string _Dir;
```

// --- COSTRUTTORI ...

```
public Pallina (int nuovaX, int nuovaY, int nuovoDiametro)
{
    _X = nuovaX;
    _Y = nuovaY;
    _Diam = nuovoDiametro;

    _Dir = "SU";
}
```

// --- PROPRIETA' ...

```
public int X
{
    get { return _X; }
    set { _X = value; }
}
```

```
public int Y
{
    get { return _Y; }
    set { _Y = value; }
}
```

```

public int Diametro
{
    get { return _Diam; }
    set { _Diam = value; }
}

public string Direzione
{
    get { return _Dir; }
    set
    {
        if ((value == "SU") || (value == "GIU"))
            _Dir = value;
        else
            MessageBox.Show("Direzione non riconosciuta.");
    }
}

```

```
// --- METODI ...
```

```

public void Muovi (Panel P)
{
    // Cancella la pallina ...
    Disegna (P, Pens.White);

    // Aggiorna la sua posizione ...
    switch ( _Dir )
    {
        case "SU": _Y = _Y - 2; break;
        case "GIU": _Y = _Y + 2; break;
    }

    // Ridisegna la pallina ...
    Disegna (P, Pens.Red);
}

```

```

public void Disegna (Panel P, Pen Penna)
{
    // Per poter disegnare, creo l'oggetto di classe Graphics ...
    Graphics G = P.CreateGraphics();

    // Disegna la pallina ...
    G.DrawArc(Penna, _X, _Y, _Diam, _Diam, 0, 360);
}

```

```

public bool E_Fuori (Panel P)
{
    return ( (_Y < 0) ||
            (_Y > P.Height - _Diam) );
}

```

```

public void RiposizionaSulBordo(Panel P)
{
    Disegna (P, Pens.White); // Cancella la pallina ...

    // Se è oltre il bordo superiore ...
    if (_Y < 0)
    {
        // ... riposiziona la pallina sul bordo superiore ...
        _Y = 0;
    }
}

```

```

else
{
    // Se è oltre il bordo inferiore ...
    if (_Y > P.Height - _Diam )

        // ... riposiziona la pallina sul bordo inferiore ...
        _Y = P.Height - _Diam;
}

Disegna (P, Pens.Red); // Ridisegna la pallina ...
}
}
}

```

Diagramma UML della Classe Pallina

