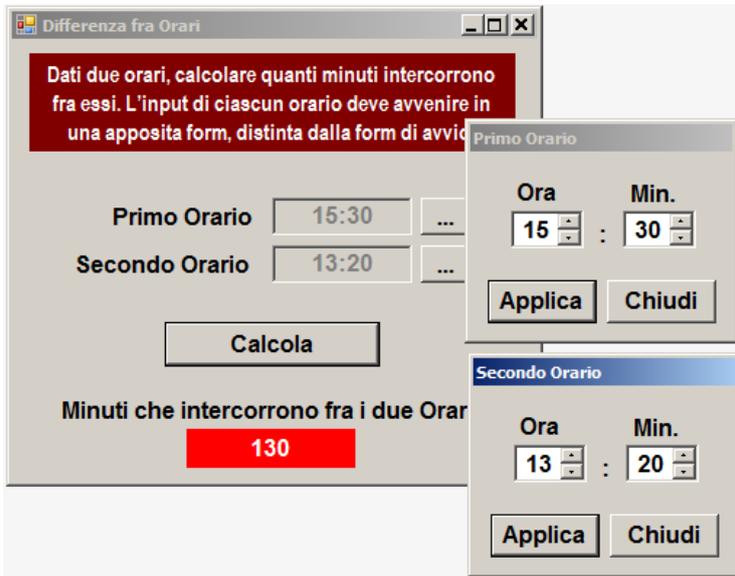


Esempio Pilota:
MultiFormDifferenzaFraDueOrari

Definizione delle Form



Codice della Form "frmAvvio"

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

```
namespace MultiFormDifferenzaFraDueOrari
```

```
{
    public partial class frmAvvio : Form
    {
        public frmAvvio()
        {
            InitializeComponent();
        }
    }
}
```

```
// Dichiaro e creo le due form per l'input dei due orari ...
// Nota: Usiamo la stessa classe frmInputOrario e passiamo "titoli" diversi!!
```

```
frmInputOrario frmPrimoOrario =
    new frmInputOrario("Primo Orario");

frmInputOrario frmSecondoOrario =
    new frmInputOrario("Secondo Orario");
```

```
private void frmAvvio_Load(object sender, EventArgs e)
{
    // Associa gli eventi HaModificato delle form ai rispettivi "gestori" ...
    frmPrimoOrario.HaModificato += frmPrimoOrario_HaModificato;
    frmSecondoOrario.HaModificato += frmSecondoOrario_HaModificato;
}
```

```
private void plsInputOrario1_Click(object sender, EventArgs e)
{
    // apri la form (apertura NON MODALE) per l'input del primo orario
    frmPrimoOrario.Show();
}
```

```
private void plsInputOrario2_Click(object sender, EventArgs e)
{
    // apri la form (apertura NON MODALE) per l'input del secondo ...
    frmSecondoOrario.Show();
}
```

```
private void plsCalcola_Click(object sender, EventArgs e)
{
    // Recupera i due oggetti Orario dalle due form ...
    Orario Ora1 = frmPrimoOrario.MioOrario;
    Orario Ora2 = frmSecondoOrario.MioOrario;

    // Calcola i minuti che intercorrono fra i due orari ...
    lblRisultato.Text = Ora1.MinutiDa ( Ora2 ).ToString();

    /* NOTA: Potevamo anche scrivere direttamente ...

    lblRisultato.Text = frmPrimoOrario.MioOrario.MinutiDa
        ( frmSecondoOrario.MioOrario ).ToString();
    */
}
```

```
private void frmPrimoOrario_HaModificato
    (object sender, EventArgs e)
{
    // Aggiorna la visualizzazione del primo orario ...
    txtOrario1.Text = frmPrimoOrario.MioOrario.ToString();
}
```

```
private void frmSecondoOrario_HaModificato
    (object sender, EventArgs e)
{
    // Aggiorna la visualizzazione del secondo orario ...
    txtOrario2.Text = frmSecondoOrario.MioOrario.ToString();
}
```

Codice della Form "frmInputOrario"

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

```
namespace MultiFormDifferenzaFraDueOrari
```

```
{
    public partial class frmInputOrario : Form
    {
```

```
// ATTRIBUTI: MioOrario contiene l'Orario digitato nella form ...
    public Orario MioOrario = new Orario();
```

```
// COSTRUTTORE: Io abbiamo "personalizzato"
// aggiungendo il parametro Titolo ...
```

```
public frmInputOrario (string Titolo)
{
    InitializeComponent();

    // imposta la proprietà Text della form, ossia il suo titolo ...
    Text = Titolo;
}
```

// EVENTI: L'evento HaModificato scatta al click su Applica o Chiudi ...

public event EventHandler HaModificato;

private void plsApplica_Click(**object** sender, EventArgs e)

```
{
    // Aggiorna l'oggetto MioOrario con i dati digitati ...
    MioOrario.Ora = (int)nudOra.Value;
    MioOrario.Minuti = (int)nudMinuti.Value;
```

```
    // Ha premuto Applica ... fai scattare l'evento HaModificato ...
    if (HaModificato != null)
        HaModificato (this, new EventArgs ( ));
}
```

private void plsChiudi_Click(**object** sender, EventArgs e)

```
{
    // Aggiorna l'oggetto MioOrario con i dati digitati ...
    MioOrario.Ora = (int)nudOra.Value;
    MioOrario.Minuti = (int)nudMinuti.Value;
```

// Anche se ha premuto Chiudi ... fai scattare l'evento HaModificato

```
...
if (HaModificato != null)
    HaModificato (this, new EventArgs ( ));
```

// Nascondi la form (non viene "chiusa" ma solo nascosta!) ...

Hide ();

```
}
```

Codice della classe "Orario"

using System;

namespace MultiFormDifferenzaFraDueOrari

```
{
    public class Orario
    {
```

// ATTRIBUTI INTERNI: Ora e Minuti
 // **Non introduciamo proprietà perchè, per semplicità,**
 // **non effettuiamo un controllo sulla validità dei dati.**

```
public int Ora = 0;
public int Minuti = 0;
```

// COSTRUTTORE: **Non introduciamo nemmeno il costruttore**
 // **vuoto perché inizializziamo direttamente Ora e Minuti !**

// METODI ...

// Il metodo MinutiDallaMezzanotte() restituisce i minuti che
 // intercorrono fra mezzanotte e l'orario stesso ...

```
public int MinutiDallaMezzanotte()
{ return Ora * 60 + Minuti; }
```

// Il metodo MinutiDa(AltroOrario) restituisce i minuti che intercorrono fra
 // l'orario stesso e un altro orario passato come parametro... Basta fare la
 // differenza dei minuti che intercorrono fra gli orari e la mezzanotte!

public int MinutiDa(Orario AltroOrario)

```
{
    return MinutiDallaMezzanotte ( ) -
        AltroOrario.MinutiDallaMezzanotte ( );
}
```

// Ridefiniamo il metodo ToString per comporre l'orario in un'unica stringa
 // garantendo sempre due cifre per le ore e due cifre per i minuti ...

public override string ToString()

```
{
    return Ora.ToString().PadLeft(2, '0') + ":" +
        Minuti.ToString().PadLeft(2, '0');
}
```

Struttura del Progetto

