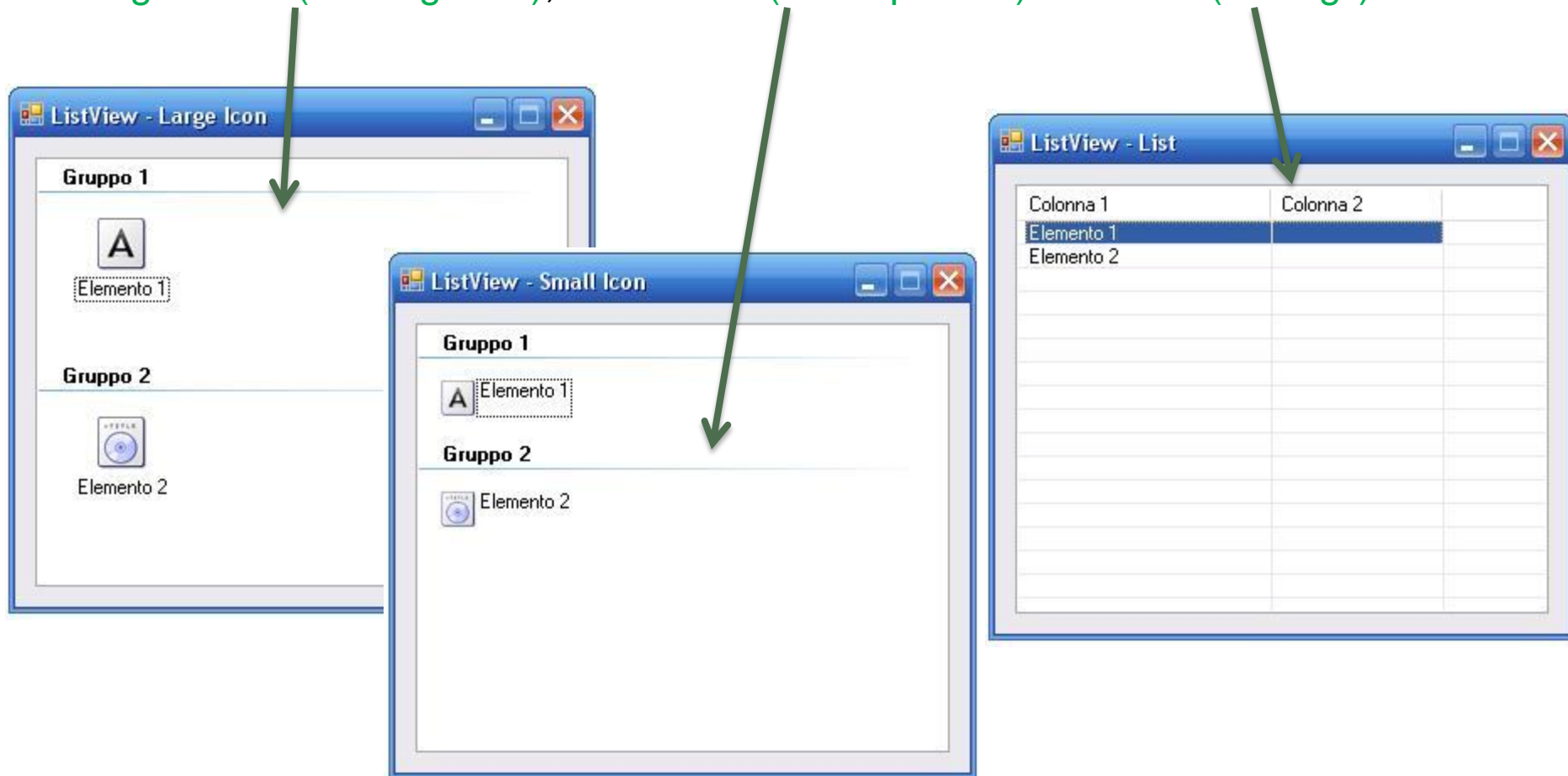
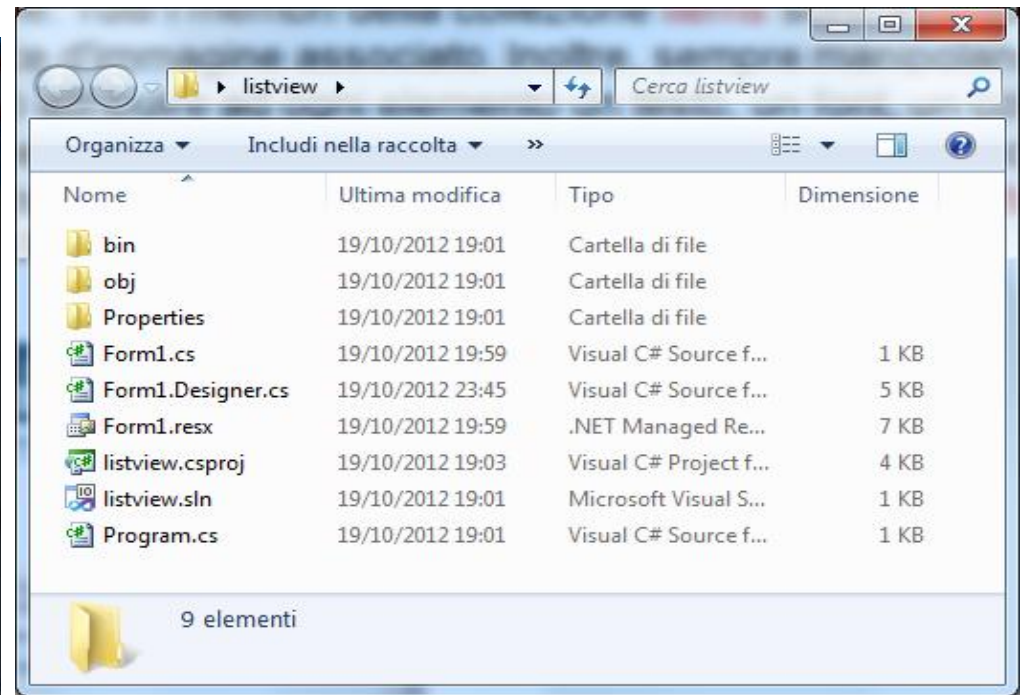

ListView

Prof. Francesco Accarino
IIS Altiero Spinelli Sesto San Giovanni

La ListView è un controllo complesso e di grande impatto visivo. È lo stesso tipo di lista usato dall'explorer di windows per visualizzare files e cartelle. Le sue proprietà permettono di personalizzarne la visualizzazione in cinque stili diversi: i più importanti di questi sono **Large Icone (Icone grandi)**, **Small Icon (Icone piccole)** e **Details (Dettagli)**.



Ogni elemento della **ListView** è contraddistinto da un oggetto **ListViewItem**, esso, non è una semplice stringa (o un tipo base facilmente rappresentabile) ma un oggetto a sè stante, del quale si possono personalizzare tutte le caratteristiche visive e stilistiche. Tutti i membri della collezione **Items** sono in grado di impostare l'indice d'immagine associato. Inoltre, sempre manipolando le proprietà, si può attribuire ad ogni elemento un testo, un font, un colore diverso a seconda delle necessità. Nella visualizzazione a dettagli si possono impostare tutti i valori corrispettivi ad ogni colonna mediante la proprietà **SubItems**, la quale contiene una collezione di oggetti **ListViewSubItem**



Alcune delle Proprietà Più Importanti

- ❑ **CheckBoxes** : indica se la listview debba visualizzare delle CheckBox vicino ad elemento
- ❑ **Columns** : collezione delle colonne disponibili. Ogni colonna è contraddistinta da un testo (Text), un indice d'immagine (ImageIndex) e un indice di visualizzazione (DisplayIndex) che specifica la sua posizione ordinale nella visualizzazione. Le colonne sono visibili sono con View = Details
- ❑ **FullRowSelect** : indica se evidenziare tutta la riga o solo il primo elemento, quando View = Details
- ❑ **GridLines** : indica su visualizzare le righe della griglia, quando View = Details
- ❑ **Groups** : collezione dei gruppi disponibili
- ❑ **HeaderStyle** : specifica se le intestazioni delle colonne possano essere cliccate o meno
- ❑ **Items** : collezione degli elementi della listview
- ❑ **LargeImageList** : ImageList per View = Large Icon / Tile / List
- ❑ **MultiSelect** : indica se si possano selezionare più elementi contemporaneamente
- ❑ **ShowGroups** : determina se visualizzare i gruppi
- ❑ **SmallIconList** : ImageList per View = Small Icon / Details
- ❑ **Sorting** : il tipo di ordinamento, se alfabetico ascendente o discendente.

Esempio:

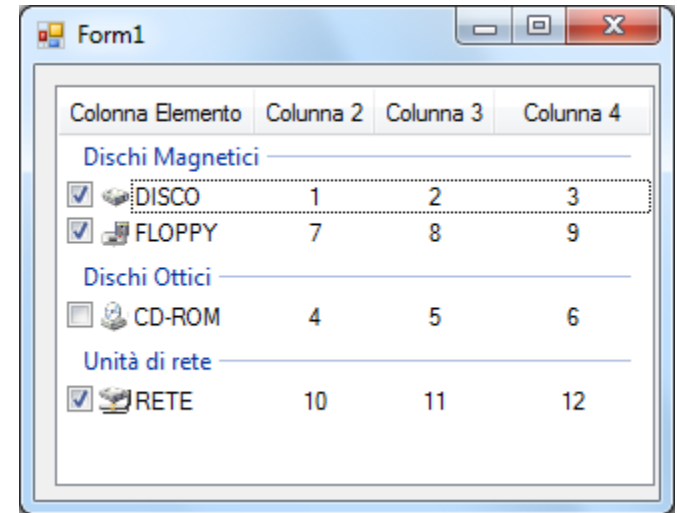
```
ListView lvListView1 = new ListView();
lvListView1.Bounds = new Rectangle(new Point(10,10), new Size(300,200));

// Imposto la view in modalità dettagli
lvListView1.View = View.Details;
// Abilito l'utente a modificare il testo
lvListView1.LabelEdit = true;
// Abilito l'utente a modificare le colonne
lvListView1.AllowColumnReorder = true;
// Visualizzo i CheckBox
lvListView1.CheckBoxes = true;
// Seleziono Item e SubItem quando viene effettuata una selezione
lvListView1.FullRowSelect = true;
// Visualizzo La griglia
lvListView1.GridLines = true;

//creazione di tre gruppi di visualizzazione
ListViewGroup lvgGruppo1 = new ListViewGroup("Dischi Magnetici");
ListViewGroup lvgGruppo2 = new ListViewGroup("Dischi Ottici");
ListViewGroup lvgGruppo3 = new ListViewGroup("Unità di rete");

//Aggiunta dei tre gruppi alla ListView
lvListView1.Groups.Add(lvgGruppo1);
lvListView1.Groups.Add(lvgGruppo2);
lvListView1.Groups.Add(lvgGruppo3);

// Creazione di quattro elementi e tre gruppi di sottoelementi per ogni elemento
// il costruttore riceve il testo e l'indice dell'immagine ad esso assegnati
ListViewItem lviItem1 = new ListViewItem("DISCO", 0, lvgGruppo1);
// Imposto il Check Box a spuntato per questo elemento
lviItem1.Checked = true;
lviItem1.SubItems.Add("1");
lviItem1.SubItems.Add("2");
lviItem1.SubItems.Add("3");
// Non imposto il Check Box a spuntato per questo elemento
ListViewItem lviItem2 = new ListViewItem("CD-ROM",1,lvgGruppo2);
lviItem2.SubItems.Add("4");
lviItem2.SubItems.Add("5");
lviItem2.SubItems.Add("6");
```



Esempio Continua:

```
ListViewItem lviItem3 = new ListViewItem("FLOPPY",2,lvGruppo1);
// Imposto il Check Box a spuntato per questo elemento
lviItem3.Checked = true;
lviItem3.SubItems.Add("7");
lviItem3.SubItems.Add("8");
lviItem3.SubItems.Add("9");
ListViewItem lviItem4 = new ListViewItem("RETE", 3,lvGruppo3);
// Imposto il Check Box a spuntato per questo elemento
lviItem4.Checked = true;
lviItem4.SubItems.Add("10");
lviItem4.SubItems.Add("11");
lviItem4.SubItems.Add("12");

// Creazione delle colonne per gli Items e i SubItems
// Larghezza -2 significa autoresize
lvListView1.Columns.Add("Colonna Elemento", -2, HorizontalAlignment.Center);
lvListView1.Columns.Add("Colonna 2", -2, HorizontalAlignment.Center);
lvListView1.Columns.Add("Colonna 3", -2, HorizontalAlignment.Center);
lvListView1.Columns.Add("Colonna 4", -2, HorizontalAlignment.Center);

//Aggiungo gli Items alla ListView.
lvListView1.Items.AddRange(new ListViewItem[]{lviItem1,lviItem2,lviItem3,lviItem4});

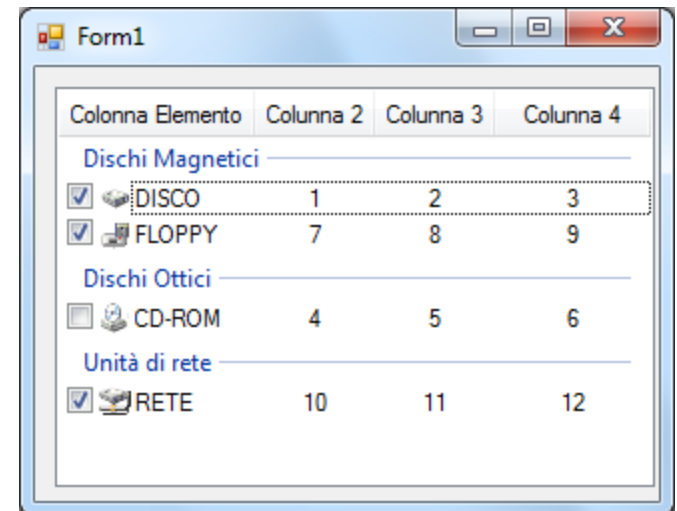
// Creo due ImageList objects.
ImageList imageListSmall = new ImageList();
ImageList imageListLarge = new ImageList();

// Inizializzo gli oggetti ImageList con i bitmaps.
imageListSmall.Images.Add(Bitmap.FromFile("disco.bmp"));
imageListSmall.Images.Add(Bitmap.FromFile("cd.bmp"));
imageListSmall.Images.Add(Bitmap.FromFile("floppy.bmp"));
imageListSmall.Images.Add(Bitmap.FromFile("rete.bmp"));

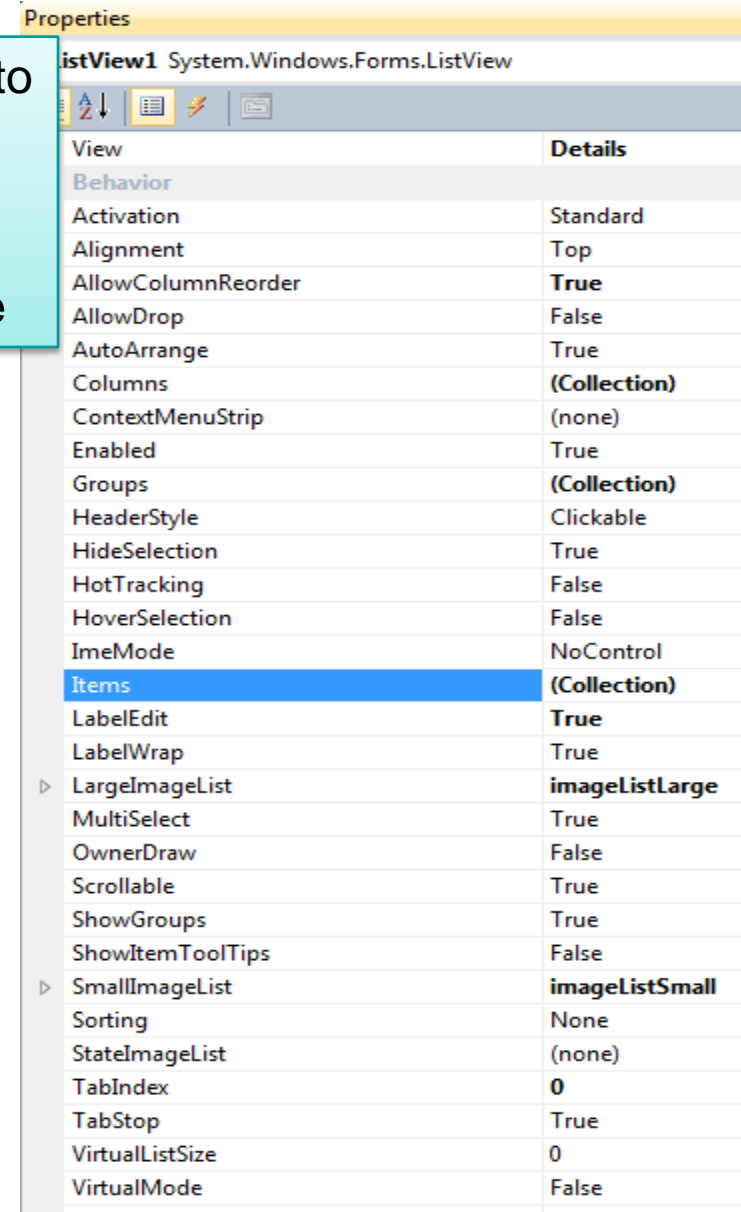
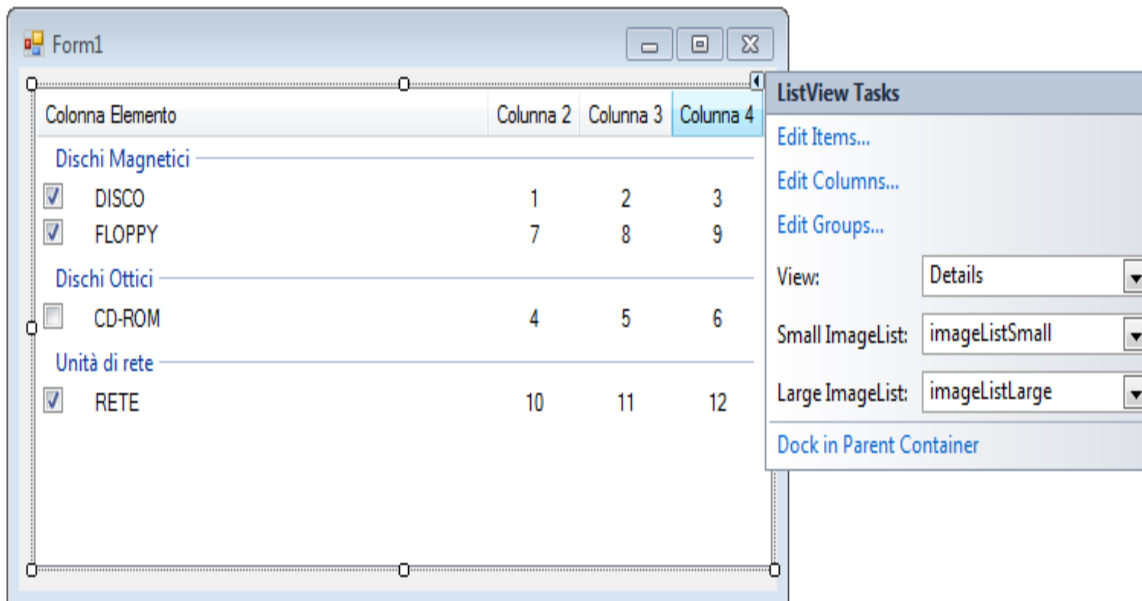
imageListLarge.Images.Add(Bitmap.FromFile("discol.bmp"));
imageListLarge.Images.Add(Bitmap.FromFile("cdl.bmp"));
imageListLarge.Images.Add(Bitmap.FromFile("floppy1.bmp"));
imageListLarge.Images.Add(Bitmap.FromFile("retel.bmp"));

//Assegno gli oggetti ImageList alla ListView
lvListView1.LargeImageList = imageListLarge;
lvListView1.SmallImageList = imageListSmall;

// Aggiungo la ListView ai Controls della form.
this.Controls.Add(lvListView1);
```

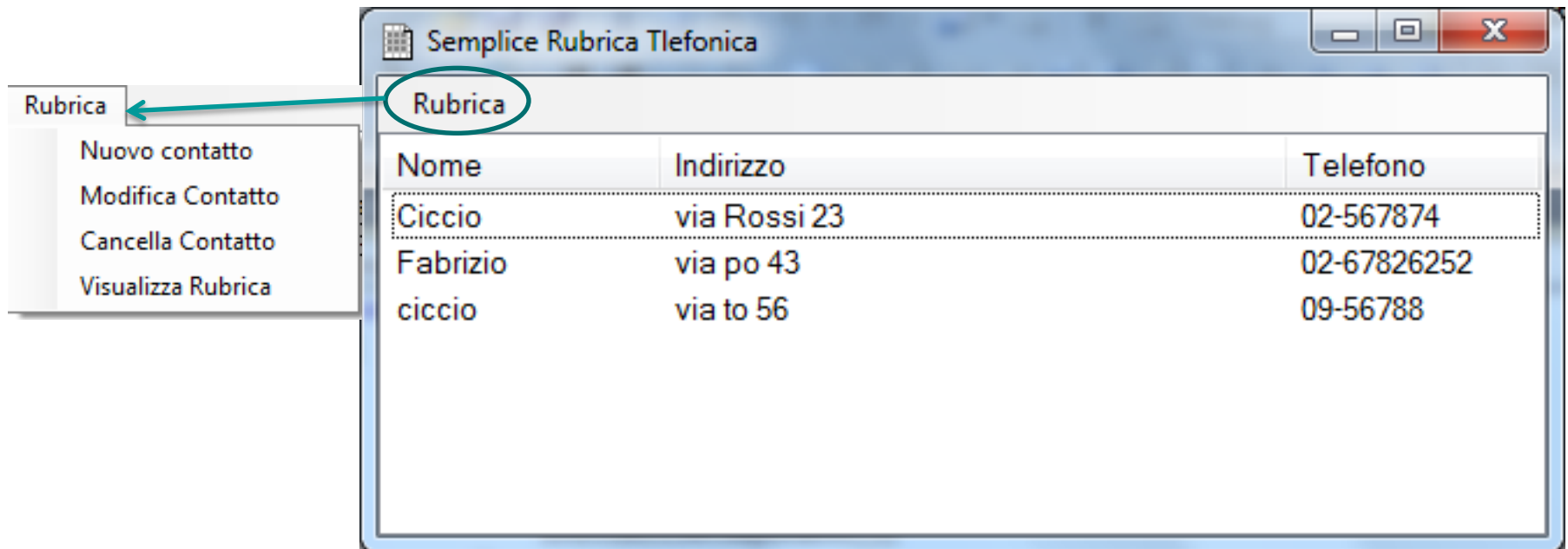


Ovviamente tutte le operazioni fatte nel codice visto precedentemente possono essere fatte nel Framework , ma queste vanno bene solo se statiche. Quasi sempre però la visualizzazione è dinamica ed è quindi fondamentale farle da codice



Esercitazione: Rubrica Telefonica

Vogliamo Realizzare Una semplice rubrica Telefonica come quella Mostrata nella figura



Utilizzeremo quindi una ListView con tre colonne in visualizzazione dettagli. Ogni Item avrà due SubItem

Facciamo Alcune Considerazioni su come andremo a strutturare i record del nostro archivio. Ovviamente come struttura logica i record sono strutturati in tre campi: **Nome** **Indirizzo** **Telefono**



Ovviamente dovremmo stabilire una lunghezza per il record e una lunghezza per ogni campo. Ma volendo utilizzare un file di testo come contenitore dell'archivio possiamo sfruttare le caratteristiche di questi particolari file ed utilizzare dei record a lunghezza Variabile

The screenshot shows a hex editor window titled "[E:\DISCOD\appunti\Tosciba\DOCUMENTI E PRESENTAZIONI CLASS\QUINTA\INFORMATICA\Esercitazioni\rubrica\bin\Debug\rubrica.dat] - Frhed". The main area displays a hex dump of the file's contents. The data is organized into rows, with the first four rows (00, 13, 26, 39) corresponding to the first record. The hex values are:

00: 43 69 63 63 69 6f 2c 76 69 61 20 52 6f 73 73 69 20 32 33

13: 2c 30 32 2d 35 36 37 38 37 34 0d 0a 46 61 62 72 69 7a 69

26: 6f 2c 76 69 61 20 70 6f 20 34 33 2c 30 32 2d 36 37 38 32

39: 36 32 35 32 0d 0a 63 69 63 63 69 6f 2c 76 69 61 20 74 6f

4c: 20 35 36 2c 30 39 2d 35 36 37 38 38 0d 0a

To the right of the hex dump, the corresponding ASCII text is displayed:

Ciccio,via Rossi 23

,02-567874..Fabrizi

o,via po 43,02-6782

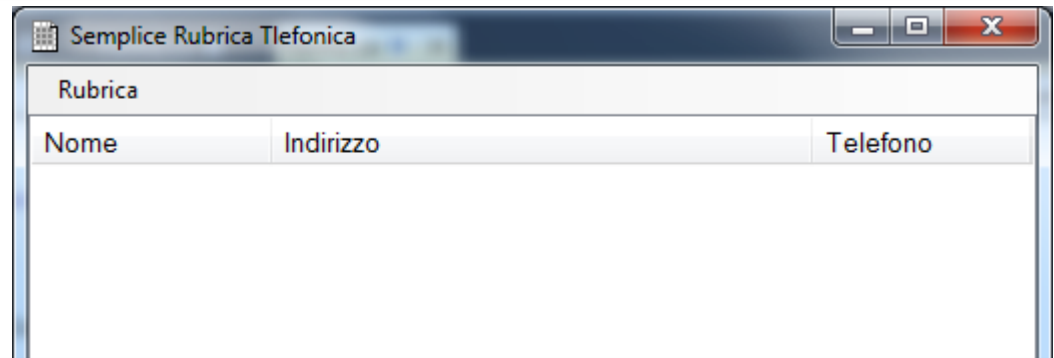
6252..ciccio,via to

56,09-56788..

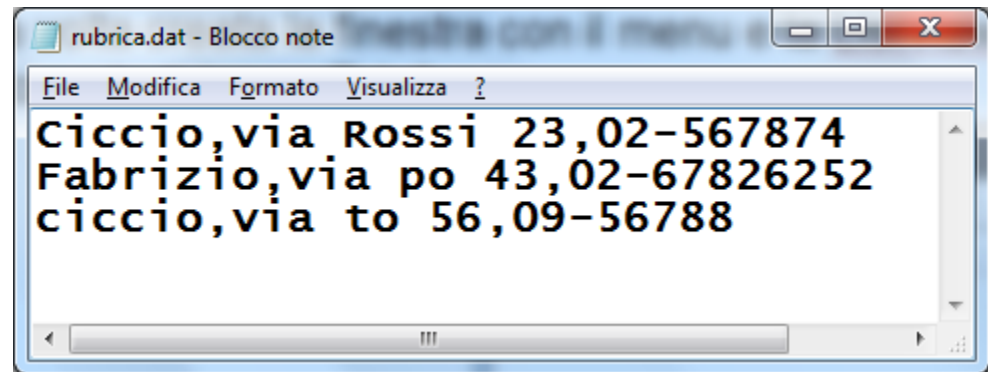
In the hex dump, several regions are highlighted with colored boxes: a red box highlights the hex values 6f 2c 76 69 61 20 (offset 26); a green box highlights 6f 20 34 33 2c (offset 26); a blue box highlights 32 35 32 0d 0a (offset 39). The ASCII text is aligned with these hex values.

The status bar at the bottom shows "Offset 90=0x5a", "END", "ANSI / OVR / L", and "Size: 90".

Quindi una volta creata la finestra con il menu e la ListView contenente le tre colonne Nome Indirizzo e Telefono



Apriamo BloccoNote, e scriviamo tre record come mostrato in figura. Salviamo il file chiamandolo rubrica.dat



Utilizziamo la virgola come separatore dei campi avremo così record a lunghezza variabile con campi a lunghezza variabile

Voce di menu Visualizza

Implementiamo ora la visualizzazione di questo file nella nostra applicazione.

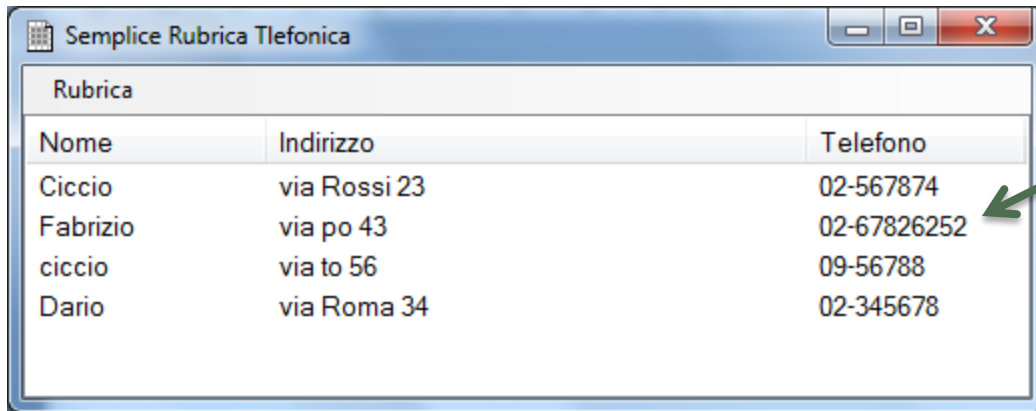
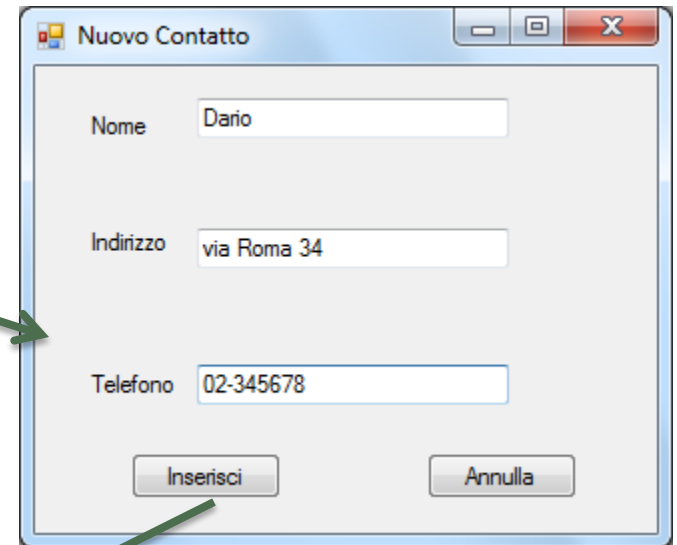
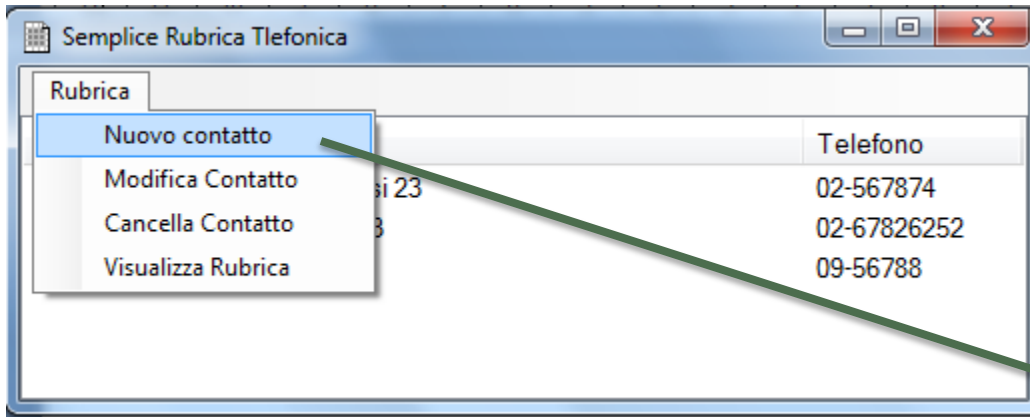
Qui di fianco è riportato il codice della funzione che risponde all'evento click sulla voce Visualizza

```
public void visualizzaRubricaToolStripMenuItem_Click(object sender, EventArgs e)
{
    ListViewItem lwiRiga;
    string contattoLetto;
    string[] campi;
    try{
        StreamReader sr = new StreamReader("rubrica.dat");
        if (sr.EndOfStream)
        {
            MessageBox.Show("La rubrica è vuota");
            sr.Close();
        }
        else
        {
            lwContatti.Clear();
            lwContatti.Columns.Add("Nome");
            lwContatti.Columns.Add("Indirizzo");
            lwContatti.Columns.Add("Telefono");
            lwContatti.Columns[0].Width = 120;
            lwContatti.Columns[1].Width = 270;
            lwContatti.Columns[2].Width = 109;
            lwContatti.View = View.Details;

            while (!sr.EndOfStream)
            {
                contattoLetto = sr.ReadLine();
                campi = contattoLetto.Split(',');
                lwiRiga = new ListViewItem(campi[0]);
                lwContatti.Items.Add(lwiRiga);
                lwiRiga.SubItems.Add(campi[1]);
                lwiRiga.SubItems.Add(campi[2]);
            }
            sr.Close();
        }
    }
    catch(Exception){
        MessageBox.Show("il file rubrica.dat non esiste");
    }
}
```

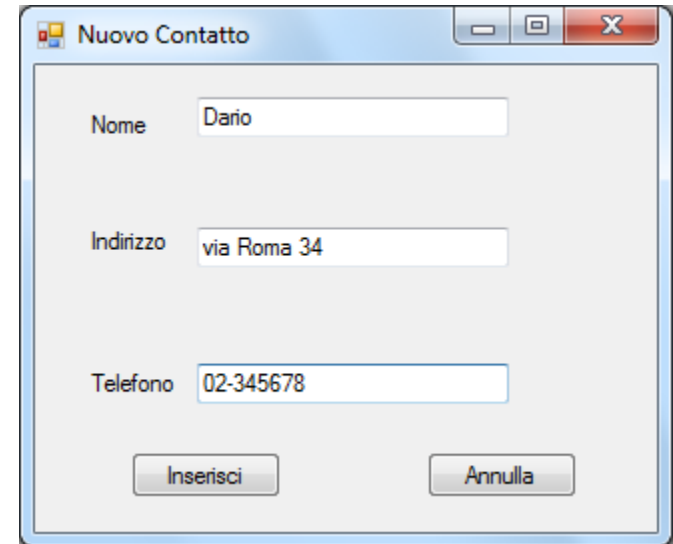
Le altre voci del menù dovranno realizzare le seguenti funzionalità:

Voce Nuovo Contatto



Azioni da svolgere:

1. Costruisco un secondo Form in Modalità grafica
2. Alla voce di menu NuovoContatto dovrà corrispondere una funzione che semplicemente fa un'istanza di questo form e lo visualizza
3. Al bottone annulla corrisponde una funzione che semplicemente chiude il form
4. Mentre al bottone inserisci sarà assegnata una funzione che:
 - A. Controlla che le caselle di testo siano tutte compilate altrimenti segnala l'errore
 - B. Costruisce una stringa con nome indirizzo e telefono inseriti separati da virgola
 - C. Apre uno StreamWriter sul file in append
 - D. Scrive la stringa sullo stream



The image shows a graphical user interface window titled "Nuovo Contatto". It features three text input fields with the following content: "Nome" (Dario), "Indirizzo" (via Roma 34), and "Telefono" (02-345678). Below the fields are two buttons: "Inserisci" and "Annulla". The window has a standard Windows-style title bar with minimize, maximize, and close buttons.

Voce Modifica Contatto

Nessun Contatto
Selezionato

Semplice Rubrica Telefonica

| Rubrica | | |
|----------|--------------|-------------|
| Nome | Indirizzo | Telefono |
| Ciccio | via Rossi 23 | 02-567874 |
| Fabrizio | via po 43 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |

devi selezionare un contatto per modificarlo

OK

Semplice Rubrica Telefonica

| Rubrica | | |
|-----------------|------------------|--------------------|
| Nome | Indirizzo | Telefono |
| Ciccio | via Rossi 23 | 02-567874 |
| Fabrizio | via po 43 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |

Semplice Rubrica Telefonica

| Rubrica | | |
|-----------------|------------------|--------------------|
| Nome | Indirizzo | Telefono |
| Ciccio | via Rossi 23 | 02-567874 |
| Fabrizio | via po 42 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |

Modifica Contatto

Nome

Indirizzo

telefono

ok annulla

Voce Modifica Modifica

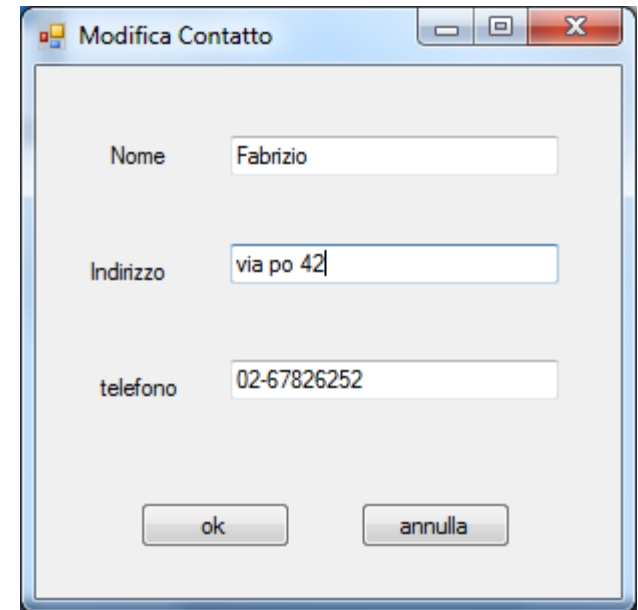
```
private void modificaContattoToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (lwContatti.SelectedIndices.Count>0)
    {
        Form3 modifica = new Form3(this.lwContatti);

        modifica.Show();
    }
    else MessageBox.Show("devi selezionare un contatto per modificarlo");
}
```

```
public partial class Form3 : Form
{
    ListView list;
    public Form3(ListView lista) {
        list = lista;
        InitializeComponent();

        this.nome.Text = lista.SelectedItems[0].Text;
        this.indirizzo.Text = lista.SelectedItems[0].SubItems[1].Text;
        this.telefono.Text = lista.SelectedItems[0].SubItems[2].Text;
    }

    public Form3()
    {
        InitializeComponent();
    }
}
```



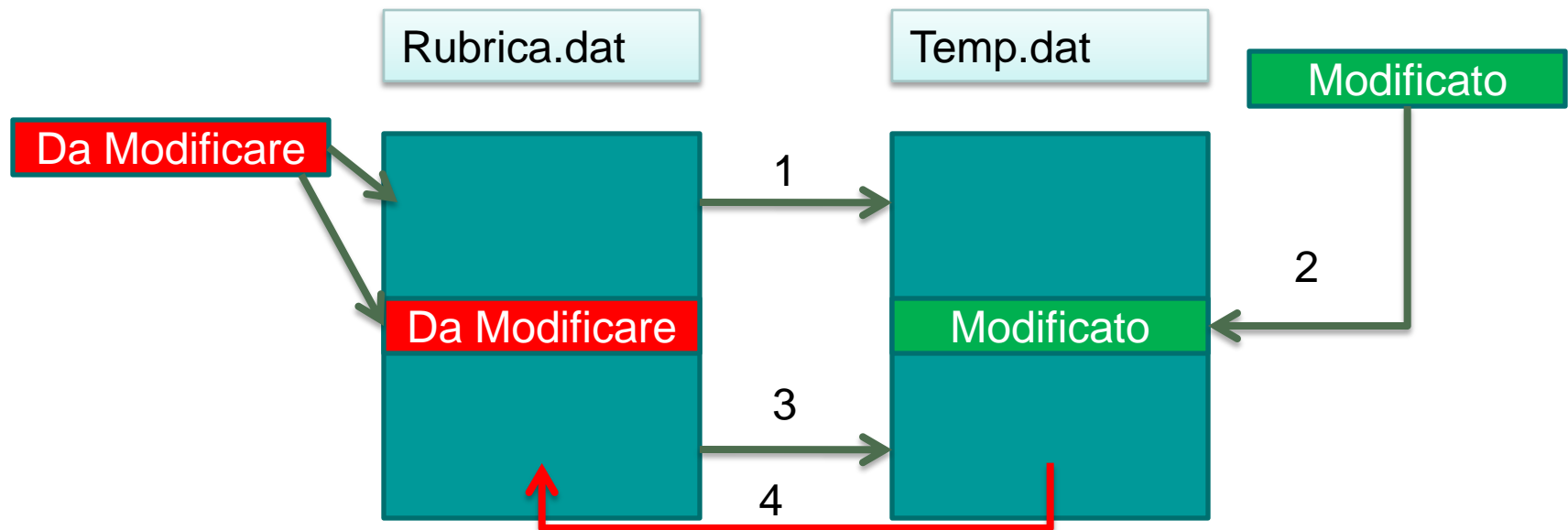
The screenshot shows a Windows-style dialog box titled "Modifica Contatto". It contains three text input fields with the following labels and values:

- Nome: Fabrizio
- Indirizzo: via po 42
- telefono: 02-67826252

At the bottom of the dialog, there are two buttons: "ok" and "annulla".

Quando si clicca su OK nella funzione corrispondente all'evento dobbiamo svolgere:

1. Creare una stringa contenente il record originale da modificare
2. Creare una stringa contenente il record modificato
3. Aprire il file rubrica Con uno StreamReader in lettura
4. Aprire un file Temp con uno StreamWriter
5. Scorrere Lo stream di lettura ricopiando i record diversi da quello da modificare e sostituirlo con quello modificato (1,2,3)
6. Chiudere i due Stream e riaprirli al contrario per ricopiare il file Temp su Rubrica (4)
7. aggiorna gli elementi della listview con la maodifica effettuata

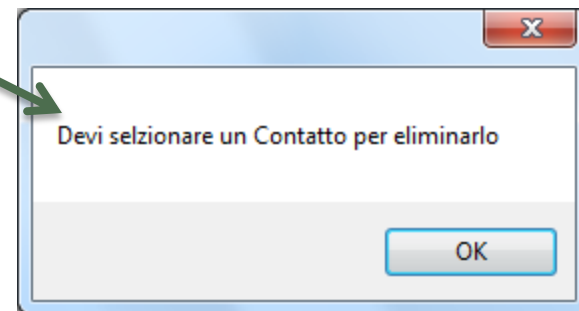


Voce Cancella Contatto

Semplice Rubrica Telefonica

| Rubrica | | |
|----------|--------------|-------------|
| Nome | Indirizzo | Telefono |
| Ciccio | via Rossi 23 | 02-567874 |
| Fabrizio | via po 43 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |

Nessun Contatto
Selezionato

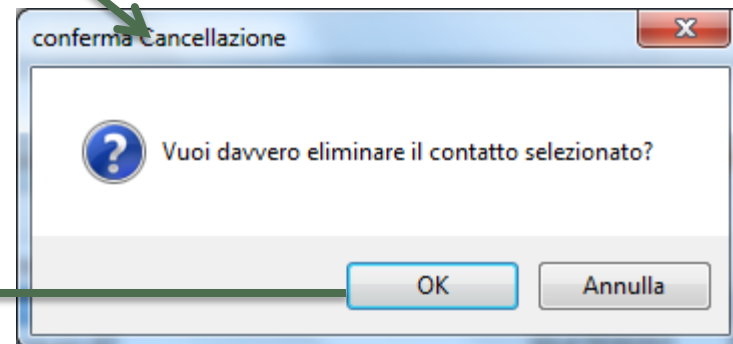


Semplice Rubrica Telefonica

| Rubrica | | |
|----------|--------------|-------------|
| Nome | Indirizzo | Telefono |
| Ciccio | via Rossi 23 | 02-567874 |
| Fabrizio | via po 42 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |

Semplice Rubrica Telefonica

| Rubrica | | |
|----------|-----------|-------------|
| Nome | Indirizzo | Telefono |
| Fabrizio | via po 42 | 02-67826252 |
| ciccio | via to 56 | 09-56788 |



La Funzione Cancella Contatto è simile alla modifica la lascio da sviluppare a voi vi ricordo solo schematicamente quello che accade

