



## OGGETTI: proprietà, metodi, eventi

Ogni oggetto, o controllo, presente nella casella degli strumenti è dotato di un insieme di **proprietà, metodi ed eventi**.

Con il termine “**proprietà**” indichiamo le caratteristiche dell’oggetto equiparabili, nel linguaggio naturale, agli aggettivi ovvero alle qualità. Tutti noi conosciamo l’oggetto Radio. Le sue qualità sono costituite dal modo di funzionare (analogica o digitale), oppure dal suo colore (bianco o verde), oppure dal suo peso. Proseguendo nell’analogia possiamo anche capire come alcune proprietà siano modificabili mentre altre non lo sono: posso cambiare il colore della radio, ma non posso modificare l’insieme delle frequenze che riceve. Sugli oggetti possiamo poi eseguire determinate azioni che si chiamano “**metodi**”. Sempre in riferimento all’oggetto Radio possiamo eseguire un’azione costituita dall’accendersi, lo spegnersi, il sintonizzarsi, l’aumento o la diminuzione del volume.

### Proprietà

Le proprietà rappresentano gli attributi che definiscono l'aspetto e varie funzionalità di ogni controllo; ad esempio la proprietà Name indica il nome con cui quel controllo è identificato all'interno del codice; le proprietà Height e Width indicano l'altezza e la larghezza del controllo, ecc. Molte proprietà sono comuni a diversi oggetti (ad es. qualunque controllo dispone della proprietà Name), altre invece sono specifiche di un controllo particolare (ad es., la proprietà Intervallo timer è disponibile solo per il form). Solitamente le proprietà possono essere lette e anche impostate, ovvero è possibile sia leggere il valore della proprietà, sia assegnare ad essa un nuovo valore: in tal caso si dice che la proprietà è di lettura e scrittura; tuttavia esistono anche proprietà di sola lettura, alle quali non è possibile assegnare un nuovo valore, e di sola scrittura.

Le proprietà sono le qualità dell’oggetto, che lo definiscono e lo distinguono dagli altri oggetti simili che fanno parte del programma. Sono proprietà di un oggetto:

- il nome;
- la sua posizione sullo schermo;
- la sua altezza e la sua grandezza;
- il colore dello sfondo;
- l’immagine che eventualmente appare sullo sfondo o all’interno dell’oggetto;
- le caratteristiche delle scritte che eventualmente appaiono sull’oggetto (colore del testo, tipo di caratteri, grandezza dei caratteri, stile dei caratteri);
- il fatto che l’oggetto sia visibile o no, sia attivato o no, sia spostabile sullo schermo oppure no, sia riducibile ad icona, ecc.

Le proprietà degli oggetti vengono definite in fase di programmazione dal programmatore, utilizzando la Finestra delle Proprietà, ma possono essere modificate anche durante la fase di esecuzione, se accadono determinati eventi, scrivendo apposite istruzioni del codice. Per esempio, un form può avere il colore rosso assegnato come colore di fondo all’inizio di un programma, ma può cambiare il colore di fondo più volte nel corso dell’esecuzione, può diventare invisibile, può diventare inattivo, può diventare più grande o più piccolo, può cambiare stile e grandezza delle scritte...

### Metodi

un metodo è un'azione che l'oggetto può eseguire: ad es., l'oggetto form dispone del metodo Repaint. Il metodo Repaint completa qualsiasi aggiornamento di schermo in sospenso relativo a una determinata maschera. Come per le proprietà, alcuni metodi sono comuni a diversi controlli, altri invece sono specifici di un controllo particolare.

I metodi sono le istruzioni, impartite ad un oggetto, relative alle azioni che l'oggetto deve intraprendere in risposta ad un evento. Ogni tipo di oggetti ha un elenco di metodi diverso e specifico. Un form, ad esempio, può compiere queste azioni:

- Unload (il form si cancella dalla memoria temporanea del computer per lasciare il posto ad un altro form o per terminare il programma)
- Hide (il form si nasconde durante l'esecuzione del programma)
- Show (il form si mostra in un momento determinato, durante l'esecuzione del programma)
- Line (il form si appresta a tracciare delle linee al suo interno)
- Refresh (il form aggiorna il suo contenuto grafico tenendo conto delle ultime variazioni sopravvenute)
- Resize (il form cambia dimensioni).

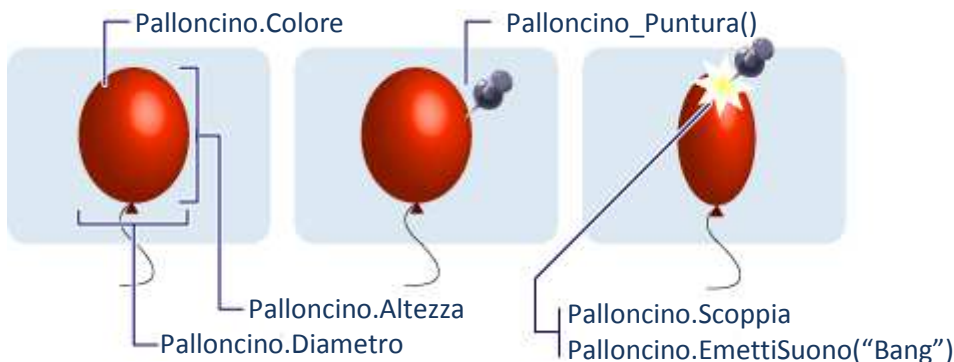
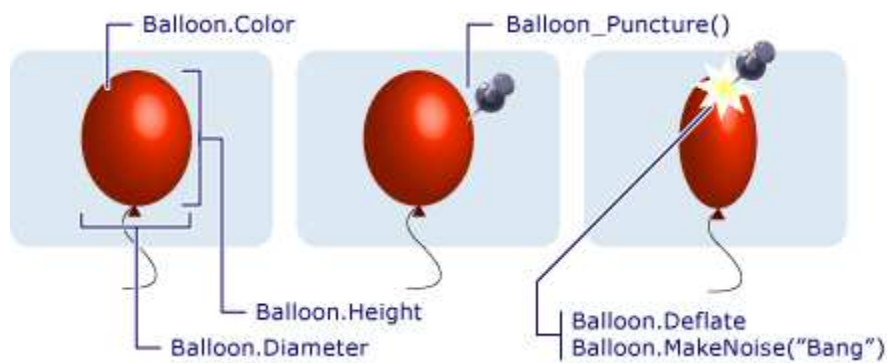
## Eventi

Gli eventi sono, come dice il nome, "situazioni" generate dal controllo quando si verificano certe condizioni; solitamente, ma non necessariamente, gli eventi si producono in conseguenza di un'azione dell'utente: per es., quando l'utente fa clic su un pulsante, il controllo che identifica quel pulsante genera un evento click; un esempio di evento non generato dall'utente è l'evento Timer che viene generato dopo un certo numero di millisecondi specificato dalla proprietà "Intervallo Timer" del form.

Gli eventi sono conseguenze di azioni, generalmente compiute dall'utente del programma, che l'oggetto riconosce perché hanno determinati effetti su di lui e lo attivano per compiere determinate azioni. Un form, ad esempio, riconosce questi eventi

- Click (l'utente ha fatto un clic con il tasto sinistro del mouse);
- DoubleClick (l'utente ha fatto un doppio clic con il mouse);
- MouseMove (l'utente ha fatto passare il mouse sopra il form senza premere alcun pulsante);
- DragOver (l'utente ha trascinato il mouse sopra il form (con il tasto sinistro del mouse premuto)
- KeyDown (l'utente ha premuto un tasto sulla tastiera).

In risposta ad ognuno di questi eventi, il form esegue le azioni per le quali il programmatore lo ha istruito (ad esempio: se avverti il mouse passare sopra di te scrivi la domanda "Posso esserti di aiuto?").



Un pallone dispone di proprietà (Colore, Altezza e Diametro), risponde a eventi (puntura) e può eseguire metodi (scoppia e EmettiSuono).

### Proprietà

Se fosse possibile programmare un palloncino, il codice Visual Basic sarebbe simile a quello riportato di seguito, in cui sono impostate le proprietà di un palloncino.

```
Palloncino.Colore = Red
Palloncino.Diametro = 10
Palloncino.gonfio = True
```

Si noti l'ordine degli elementi del codice: l'oggetto (Palloncino), seguito dalla proprietà (Colore) e dall'assegnazione del valore (= Red). È possibile cambiare il colore del palloncino specificando un valore differente.

### Metodi

I metodi di un palloncino vengono chiamati come indicato di seguito.

```
Palloncino.Gonfia
Palloncino.Scoppia
Palloncino.Salire(5)
```

L'ordine è simile a quello di una proprietà: l'oggetto (sostantivo) è seguito dal metodo (verbo). Nel terzo metodo è presente un elemento aggiuntivo, chiamato argomento, che specifica l'altezza raggiunta dal palloncino. Alcuni metodi dispongono di uno o più argomenti che consentono di descrivere ulteriormente l'azione da eseguire.

### Eventi

Il palloncino può rispondere a un evento come indicato di seguito.

```
Sub Balloon_Puncture()
    Palloncino.EmettiSuono("Bang")
    Palloncino.Scoppia
    Palloncino.Gonfio = False
```

End Sub

In questo caso, il codice descrive il comportamento del palloncino quando si verifica un evento Puncture. Quando si verifica questo evento, chiamare il metodo EmettiSuono con un argomento "Bang", (il tipo di

rumore da emettere) quindi chiamare il metodo Scoppia. Poiché il palloncino non è più gonfio, la proprietà Gonfio viene impostata su False.

Nella realtà non è possibile programmare un palloncino, ma è possibile programmare un form o un controllo Visual Basic. Questo compito spetta al programmatore, che deve decidere quali proprietà cambiare, quali metodi richiamare o a quali eventi rispondere per ottenere l'aspetto e il comportamento desiderati.