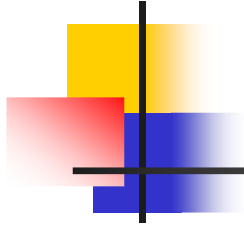


# Calcolatrice

The image shows a screenshot of a simple calculator application window titled "Calcolatrice". The window has a blue title bar with standard minimize, maximize, and close buttons. The main area is light beige and contains the following elements:

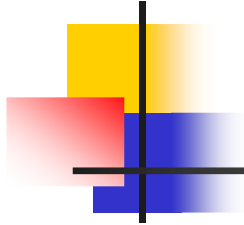
- 1° operando:** A text label followed by an empty input field.
- 2° operando:** A text label followed by an empty input field.
- Operazione:** A text label followed by an empty dropdown menu.
- Risultato:** A text label followed by an empty input field.
- Buttons:** A vertical column of four buttons for addition (+), subtraction (-), multiplication (x), and division (/). Below these are two buttons: "=" and "Cancella". A separate "Esci" button is located at the bottom right of the window.



# Calcolatrice

---

- Realizzare un'applicazione visuale che permetta di simulare il funzionamento di una semplice calcolatrice.
- Prerequisiti
  - saper utilizzare Form, Textbox, Button, messagebox;
  - saper utilizzare le stringhe e convertire un dato di tipo stringa in un dato di tipo numerico.



# Calcolatrice

---

## Specifiche di progetto:

- i valori inseriti sono di tipo reale;
- alla pressione dei tasti degli operatori(+,-,x,/) e del tasto "=" verificare la presenza degli operandi e segnalarne il mancato inserimento con una "MessageBox";
- la divisione per zero deve essere inibita e segnalata con una "MessageBox";
- realizzare il metodo **Elabora** che ricevuti in ingresso i due operandi e l'operatore ne calcoli il risultato. Il valore di ritorno del metodo indicherà se l'operazione è andata a buon fine ( 0=risultato corretto, -1 risultato errato: divisione per 0);
- strutturare il codice in sottoprogrammi;
- tempo di realizzazione cinque unità orarie.

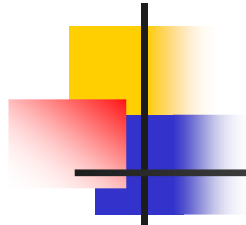


# Approfondimenti

---

- Button (esempio sender)
- TextBox
- MessageBox (esempio message box)

# ATTACCARE UN GESTORE DI EVENTO AD UN EVENTO



```
this.btnAdd.MouseClick += new  
System.Windows.Forms.MouseEventHandler(this.button1_MouseClick);
```

`btnAdd.MouseClick` rappresenta l'evento da gestire

```
+= new System.Windows.Forms.MouseEventHandler(this.button1_MouseClick);
```

`+= new System.Windows.Forms.MouseEventHandler(this.button1_MouseClick);`  
rappresenta il tipo di delega attaccata all'evento

`this.button1_MouseClick` è il nome del metodo che gestisce l'evento



# ATTACCARE UN GESTORE DI EVENTO AD UN EVENTO

---

```
private void button1_MouseClick(object sender, MouseEventArgs e)
{
    if (sender == btnAdd)
        operatore='+';
    else if (sender == btnSub)
        operatore='-';
    else if (sender == btnMultiply)
        operatore='x';
    else if (sender == btnDiv)
        operatore = '/';
}
```

Il primo parametro di tale metodo è di tipo Object e fa riferimento all'istanza che genera l'evento, mentre il secondo parametro è derivato dal tipo MouseEventArgs e contiene i dati dell'evento.

# Visualizzare messaggi: classe "MessageBox"

```
MessageBox.Show("Divisione per zero", "Errore", MessageBoxButtons.OK,  
    MessageBoxIcon.Error);
```

Divisione per zero Messaggio d'errore

Errore messaggio da visualizzare sulla barra del titolo

MessageBoxButtons.OK bottone da visualizzare

MessageBoxIcon.Error icona da visualizzare



COSTANTE	ICONA
Asterisk	
Error	
Exclamation	
Hand	
Information	
None	Nessuna
Question	
Stop	
Warning	

# Visualizzare messaggi: classe "MessageBox"

COSTANTE	ICONA
Asterisk	
Error	
Exclamation	
Hand	
Information	
None	Nessuna
Question	
Stop	
Warning	

Tabella 12-10 Elenco dei possibili valori del parametro buttons.

COSTANTE	DESCRIZIONE
AbortRetryIgnore	Sono visualizzati i bottoni <b>Termina, Riprova, Ignora</b> .
OK	E' visualizzato il bottone <b>OK</b> .
OKCancel	Sono visualizzati i bottoni <b>OK, Annulla</b> .
RetryCancel	Sono visualizzati i bottoni <b>Riprova, Annulla</b> .
YesNo	Sono visualizzati i bottoni <b>Si, No</b> .
YesNoCancel	Sono visualizzati i bottoni <b>Si, No, Annulla</b> .