

# db2000 eXperiences

*semplice come un click!*



## le guide facili



# Jabaco

**l'aggiunta e l'utilizzo di archivi jar esterni**

# 140520

documento redatto in proprio nel mese di maggio 2014

copyright © 2014 **Massimo Mascalchi**

*Anche se all'utente, autorizzato all'utilizzo del prodotto descritto in questo manuale, gli è permesso di effettuare la stampa si fa presente che nessuna parte del presente documento può essere riprodotta e/o memorizzata in un sistema che ne permetta l'elaborazione, né trasmessa in qualsiasi forma e con qualsiasi mezzo elettronico e/o meccanico senza previo consenso scritto dell'autore tranne nel caso di brevi citazioni contenute in articoli di critica o recensioni.*

*I testi e le immagini e tutte le altre informazioni riportate in questo documento non potranno mai comportare specifiche responsabilità dell'autore anche in caso di eventuali errori, inesattezze o per un uso scorretto.*

*L'autore detiene i diritti per tutte le immagini e i testi che compongono questo documento, tali diritti sono riservati a norma di legge e a norma delle convenzioni internazionali. I nomi dei prodotti e dei marchi descritti e/o discussi in questo documento appartengono ai loro rispettivi proprietari e pertanto soggetti alle regolamentazioni sui diritti d'autore e del copyright di quei paesi dove gli stessi prodotti e marchi sono rappresentati e/o commercializzati.*



## PRESENTAZIONE

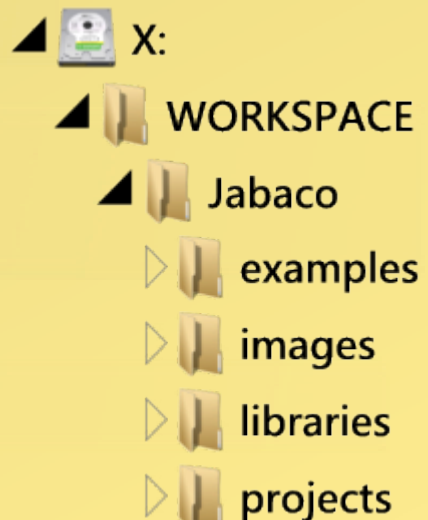
Nell'universo del software *Java* è un mondo sempre in evoluzione e, grazie anche alla sua caratteristica *cross-platform*, è in continua espansione; infatti non è possibile determinare quanti applicativi sono stati progettati e realizzati in *Java*, qualsiasi proiezione rischia di essere in difetto, tanto meno si riesce ad avere una stima ufficiale su quanti apparati è presente la *virtual machine (JVM)* che ne permette il funzionamento, si parla di miliardi di dispositivi senza contare che *Java* è anche un riferimento per ecosistemi come *Android*. Sono stati sviluppati software per le più svariate esigenze in ogni settore a soluzione di qualsiasi bisogno o pretesa ed era quindi inevitabile che si venissero a realizzare una miriade di archivi contenenti classi e metodi, insomma delle proprie e vere librerie. *Java* è stato ed è molto apprezzato soprattutto dal mondo *open source* dove si è visto aumentare il numero degli sviluppatori che di fatto sono i primi responsabili di questo progresso dove sono stati approntati diversi ambienti di sviluppo non solo da software house o aziende, che avevano e che hanno interessi specifici, ma anche da programmatori non professionisti (di solito studenti ma anche semplici persone che ne coltivano l'hobby) mossi da un unico interesse: la loro passione per la programmazione! In questo contesto **Jabaco** vede la luce nel 2008 da un'idea di *Manuel Siekmann* che lo dota di un *IDE* e di un *framework* che permettono una programmazione simile a quando avveniva in *VB6* (in alcuni casi è stato persino possibile importare direttamente e senza particolari problemi i progetti realizzati con il vecchio *IDE* dell'azienda di *Redmond*), questa condizione è stata colta da molti sviluppatori come una buona opportunità tanto è vero che sul forum dedicato opera una discreta comunità che mantiene e aggiorna il *framework*. **Jabaco** non può essere considerato ancora un prodotto maturo, forse non lo sarà mai considerando la continua evoluzione di *Java*, di sicuro rappresenta un buon compromesso tra semplicità di utilizzo e risultati ottenuti e dove l'attuale *framework* non arriva si può sopperire con l'aggiunta di archivi esterni che poi è anche l'argomento di questa guida. Ad oggi ho realizzato diverse applicazioni in **Jabaco**, alcune anche molto complesse, onestamente non ho incontrato grossi problemi e tutte (ripeto: tutte!) le volte che ho avuto la necessità di funzioni e/o componenti particolari con una semplice ricerca in *internet* ho trovato quanto mi serviva con tanto di documentazione e spesso anche con la possibilità di scelta tra più elementi trovati, inutile dire che poi tutto questo abbassa i tempi di *debug* e di realizzazione di qualsiasi progetto. Sì, **Jabaco** mi ha regalato e mi sta regalando diverse soddisfazioni che spero siano uguali anche per chi si appresta a seguire questa modesta guida.

Massimo Mascalchi  
(maXim)

### PRIMA DI INIZIARE

A prescindere dalla cartella dove si è provveduto ad installare **Jabaco** (di solito si tratta di *C:\Programmi\Jabaco*), gli esempi riportati in questa guida fanno riferimento ad un'area di lavoro predefinita alla quale si consiglia di adattarsi; pertanto si invita a creare la suddetta area secondo lo schema qui di fianco riportato.

X: rappresenta l'unità dove l'area di lavoro è stata allocata.



## L'AGGIUNTA E L'UTILIZZO DI ARCHIVI JAR ESTERNI

L'esempio proposto in questa guida si avvale dell'archivio esterno (*jar*) *SteelSeries* il quale non è altro che una libreria di componenti dove si trovano classi e metodi utili per realizzare e controllare indicatori sia lineari che radiali tipo orologi analogici, strumenti di bordo come quelli che si trovano nei cruscotti delle auto e degli aerei, display digitali e altro ancora. La libreria *SteelSeries* ha delle dipendenze con *trident*, un altro archivio esterno, che ha lo scopo di fornire altre classi e metodi utili per estendere alcune funzioni di animazione in quelle applicazioni *Java* che lo richiedono. Gli archivi *SteelSeries* e *trident* possono essere scaricati dai seguenti link:

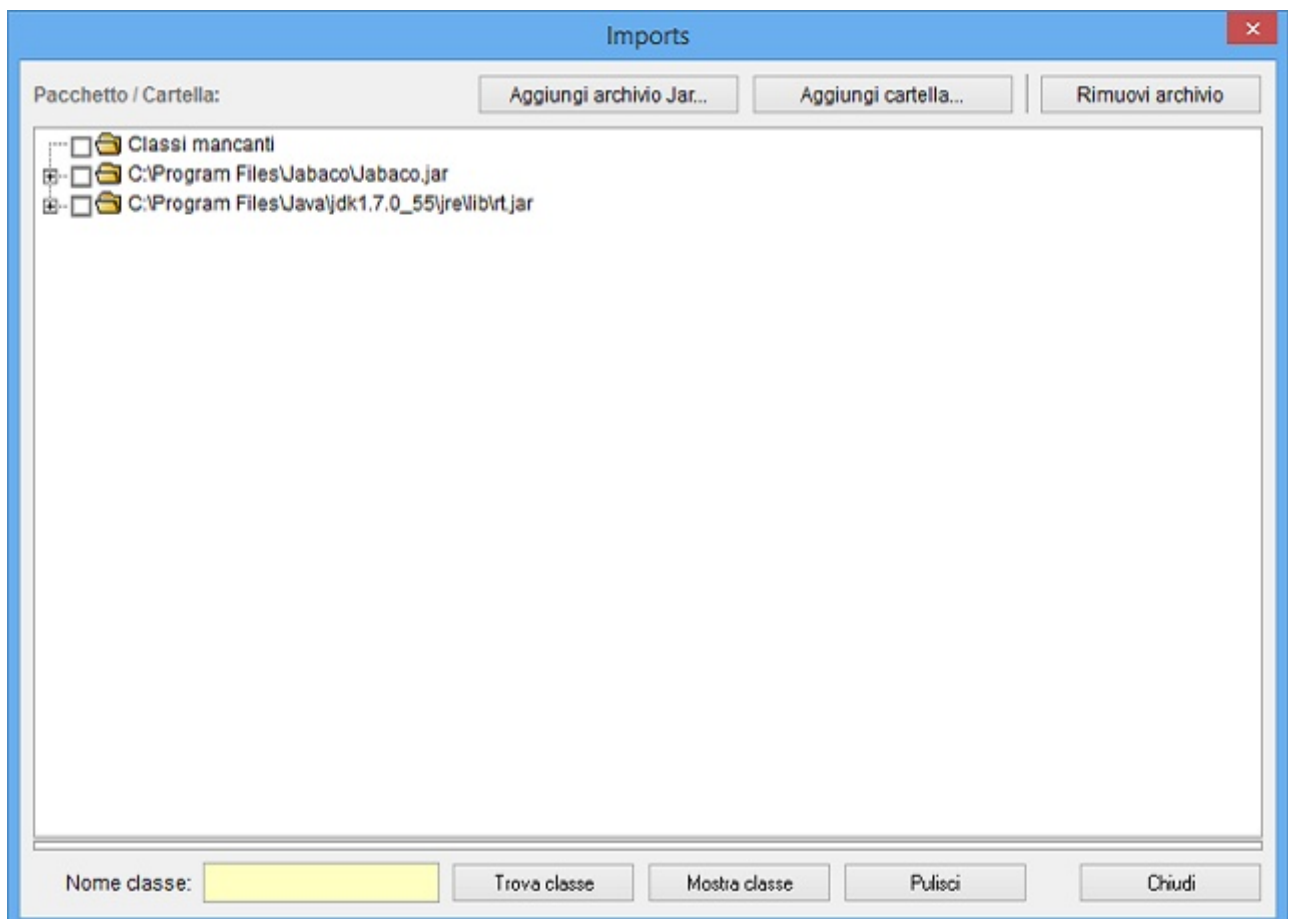
**SteelSeries:** <http://grepcode.com/snapshot/repo1.maven.org/maven2/eu.hansolo/SteelSeries/3.9.6>

**trident:** <http://grepcode.com/snapshot/repo1.maven.org/maven2/com.github.insubstantial/trident/6.3>

... per le specifiche delle varie classi e metodi e per i manuali delle due librerie si rimanda alle ricerche in *internet* dove sono presenti anche diversi esempi esaustivi in puro codice *Java*.

Una volta in possesso degli archivi sopra citati questi avranno dei nomi tipo *SteelSeries\_x.x.x.jar* e *trident\_x.x.x.jar*, i numeri presenti *x.x.x* identificano la versione. Si consiglia di rinominare i file in **SteelSeries.jar** e **trident.jar** e di copiarli nella cartella **libraries** presente nell'aria di lavoro (vedere la nota nella pagina precedente).

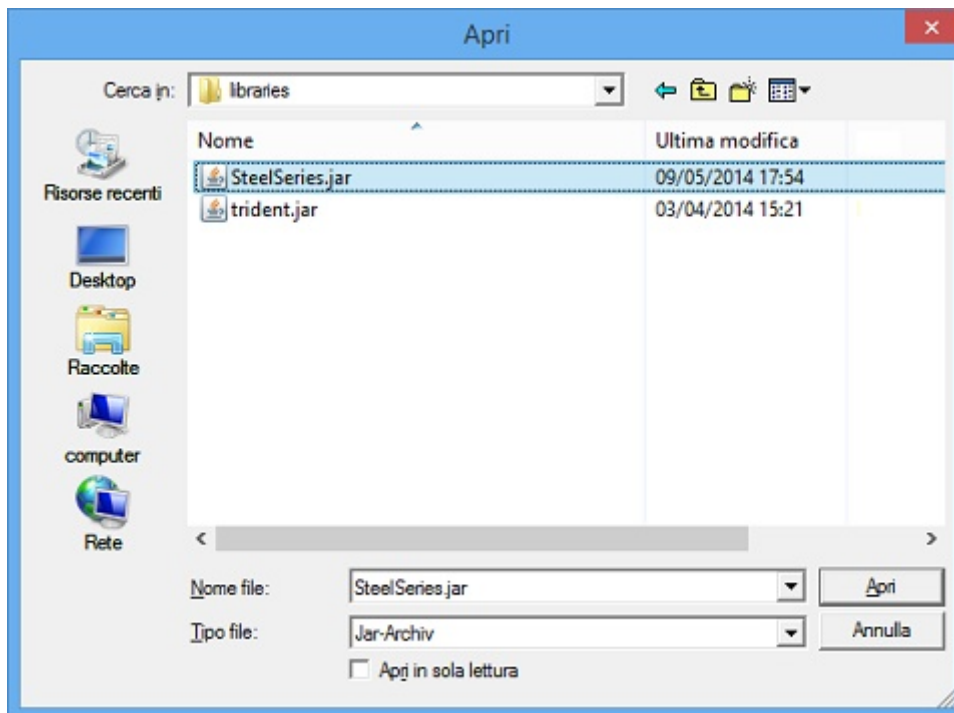
Avviare l'*IDE* di **Jabaco** e scegliere di creare una nuova applicazione **SDI**. Selezionare da menu le voci **Progetto, Riferimenti (Classpath)**... oppure premere il tasto funzione **F1**, si aprirà la finestra **Imports**:



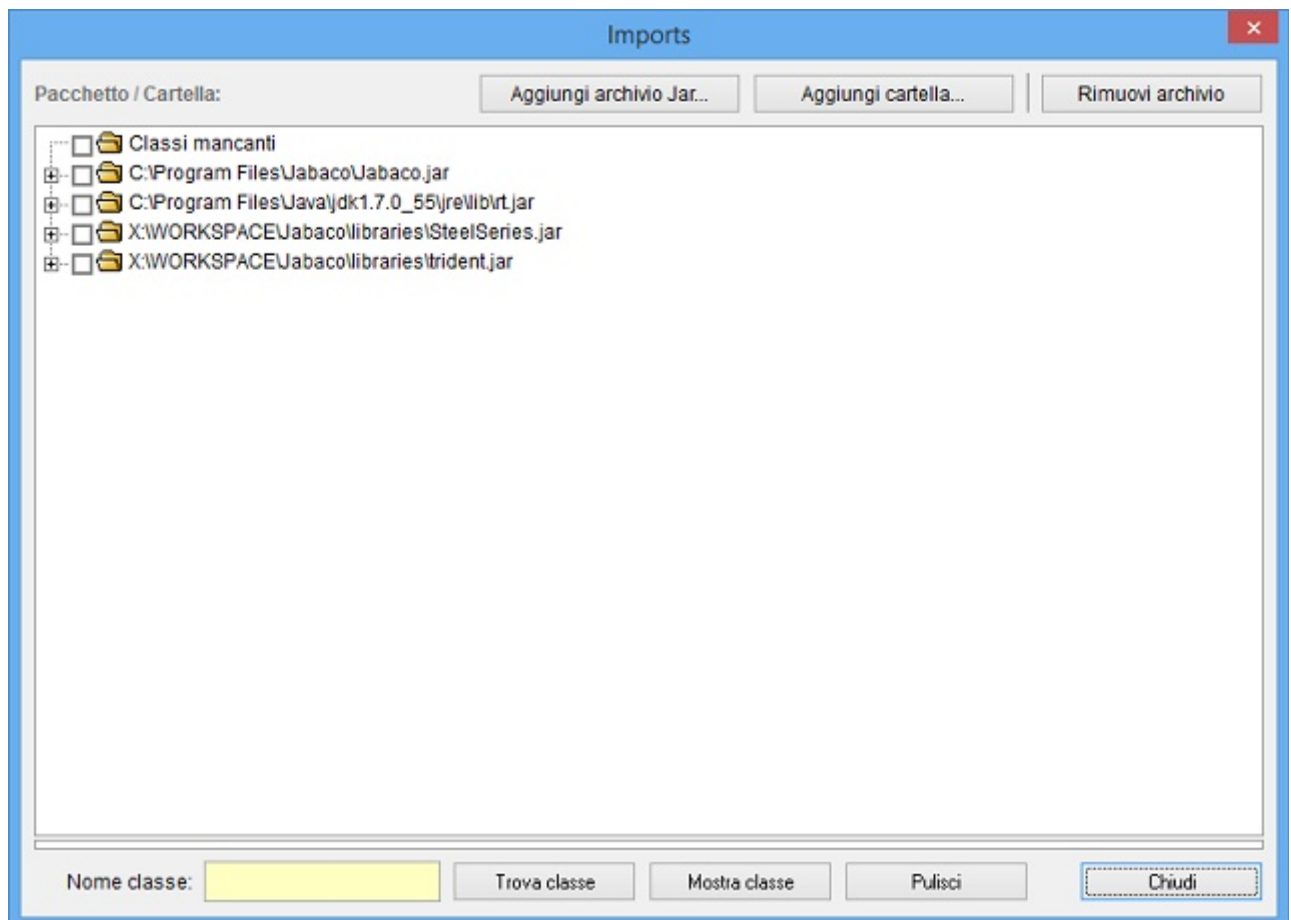
### AVVERTENZA!

*Al momento che questa guida è stata realizzata si è fatto uso del JDK 1.7 e di Jabaco Beta 1.5.2 con il framework 142.*

... agire sul pulsante **Aggiungi archivio Jar...**, si aprirà la successiva finestra di dialogo dove andremo a selezionare il file **SteelSeries.jar** precedentemente copiato nella cartella **libraries** e che confermeremo agendo sul pulsante **Apri**:

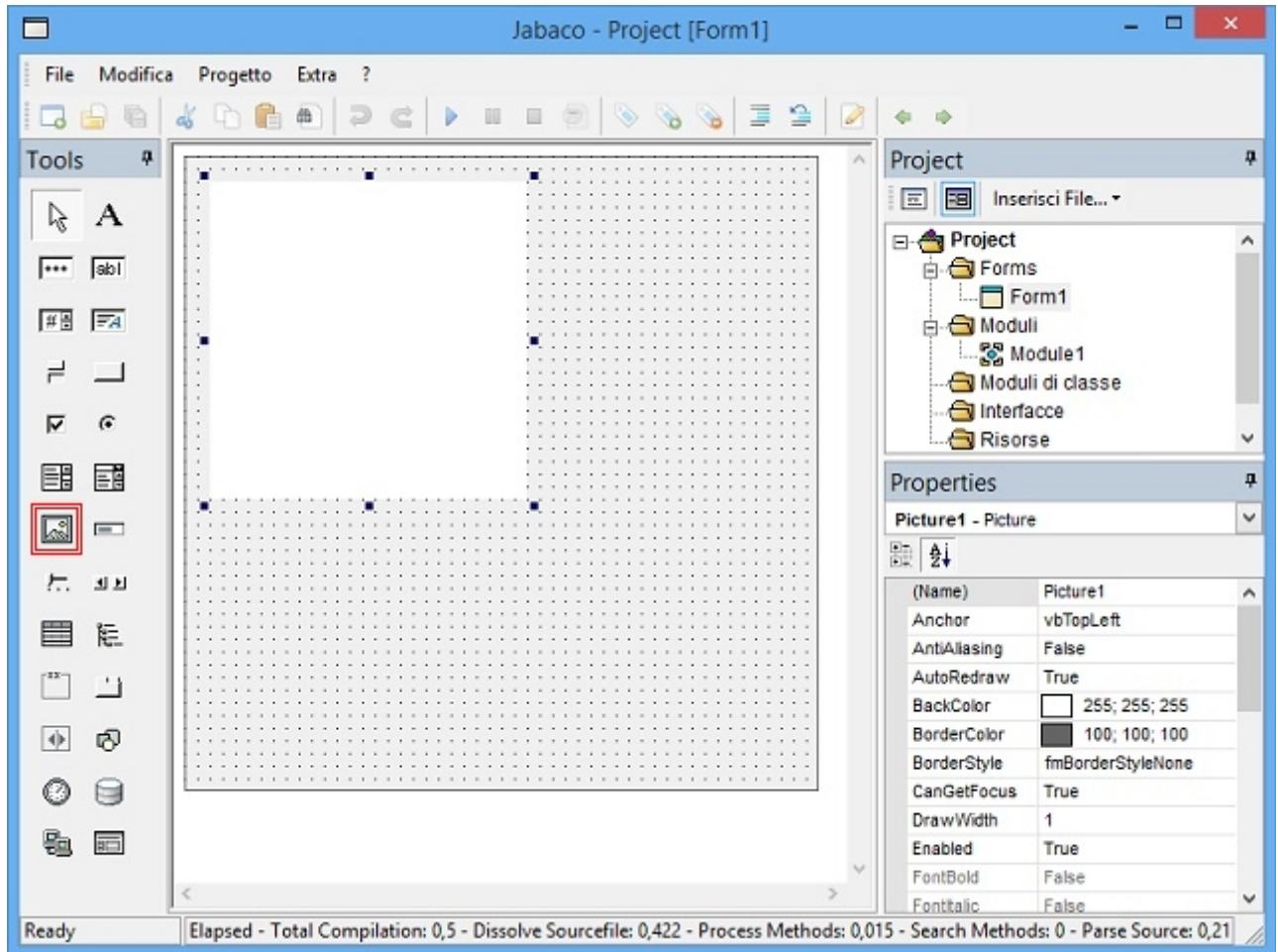


... ripetere l'operazione appena eseguita per importare anche l'archivio **trident.jar**...



... gli archivi esterni necessari agli esempi proposti in questa guida sono stati assegnati, agire sul pulsante **Chiudi** per ritornare all'IDE di **Jabaco**.

Aggiungere al *Form* un controllo *Picture* (*Height* = 200, *Width* = 200, *BorderStyle* = *fmBorderStyleNone*):



... **Picture1** sarà il contenitore del componente della libreria *SteelSeries* che verrà inizializzato direttamente da codice:

```

Option Explicit

Import org#pushingpixels#trident
Import eu#hansolo#steelseries#gauges
Import eu#hansolo#steelseries#tools

Private MyGauge As New Radial

Public Sub Form_Load()

    Picture1.BackColor = Me.BackColor

    Picture1.Parent.add(MyGauge)

    MyGauge.setSize(200, 200)
    MyGauge.setLocation(0, 0)
    MyGauge.init(1, 1)
    MyGauge.setTitle("tachimetro")
    MyGauge.setUnitString("Km")
    MyGauge.setDigitalFont(True)
    MyGauge.setTrackVisible(True)
    MyGauge.setLcdColor(LcdColor.STANDARD_LCD)
    MyGauge.setMaxValue(100)
    MyGauge.setLedBlinking(False)

End Sub

```

' \

' assegnazione delle librerie

' /

' creazione dell'oggetto

' uniforma il colore di sfondo

' assegna l'oggetto al contenitore

' assegnazione dei parametri

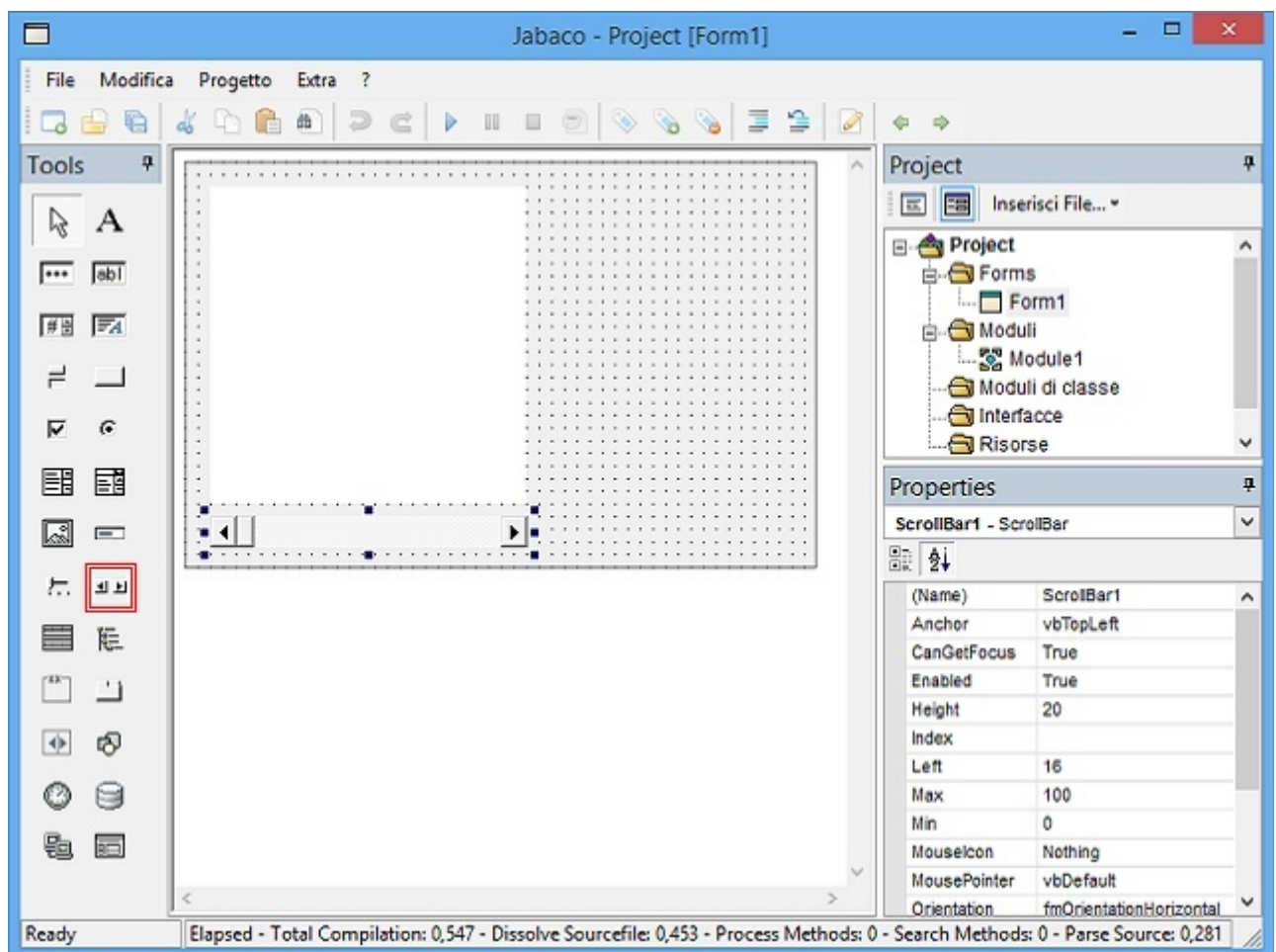
'

*per i valori da assegnare ai vari parametri si rimanda alle ricerche in internet dove sono elencate altre proprietà, classi e metodi*

... una volta avviata la procedura questo sarà il risultato:



... per impostare il valore da rappresentare sono presenti due proprietà: **setValue** e **setValueAnimated**, quest'ultima permette l'animazione dell'indicatore fino al raggiungimento del valore impostato. Per la simulazione del funzionamento dell'indicatore si propone di aggiungere al *Form*, sotto la *Picture*, il controllo *ScrollBar* (*Min* = 0, *Max* = 100):





... ecco il codice da implementare per la simulazione:

```
Public Sub ScrollBar1_Scroll()  
    MyGauge.SetValueAnimated(ScrollBar1.Value) ' assegna il valore all'indicatore  
    ' verifica se il valore rientra nel range: ' (limite in questa simulazione = 50Km)  
    If ScrollBar1.Value >= 50 Then  
        MyGauge.setLedBlinking(True) ' NO, attiva il lampeggio del led  
    Else  
        MyGauge.setLedBlinking(False) ' SI, disattiva il lampeggio del led  
    End If  
End Sub
```

... con la procedura in esecuzione, agendo sul controllo *ScrollBar*, si animerà l'indicatore:



Compilato il progetto si otterrà un file **.JAR** che contiene al suo interno tutte le classi utilizzate, sarà sufficiente distribuire solo questo file senza alcun altro allegato o libreria aggiunta salvo in casi particolari dove ci sono richieste per specifiche dipendenze.

## LINK UTILI

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

<http://www.jabaco.org/>

<http://www.jabaco.org/download.html>

<http://www.jabaco.org/board/>

[http://www.jabaco.org/wiki/Latest\\_JabacoFramework\\_Binary](http://www.jabaco.org/wiki/Latest_JabacoFramework_Binary)

<http://harmoniccode.blogspot.it/2010/08/java-swing-component-library.html>

<https://kenai.com/projects/steelseries>

<https://github.com/HanSolo/SteelSeries-Swing>

<http://www.db2000web.net/db2000%20eXperiences/Jabaco/examples/140520.zip>

*Tutti i link qui sopra riportati risultano, al 20 maggio 2014, attivi e regolarmente funzionanti con i browser più comuni.*

*Hai trovato utile questa guida?*

*Se si clicca su l'immagine sottostante e fai una libera donazione tramite **PayPal**, ci aiuterai così nella realizzazione di nuove guide e nel mantenimento dei loro aggiornamenti.*

**Donate via PayPal**

*il team di db2000 eXperiences ringrazia*

