



--NOTA: le linee di testo che iniziano come questa con "--" rappresentano i commenti al codice.
-- i commenti sono posti DOPO la riga di codice cui si riferiscono

tell application "InDesign CS"

-- i comandi sono inviati a InDesign

activate

-- porta l'applicazione in primo piano

set myDialog to make dialog

-- "make dialog" crea una finestra di dialogo in InDesign

-- il nome della finestra è inserito nella variabile "myDialog"

tell myDialog

-- le istruzioni seguenti, sono inviate alla finestra il cui nome è quello contenuto nella variabile "myDialog"

set name to "Crea campioni di colore"

-- Indico il nome della finestra

set Col0 to make dialog column -- creo una colonna, vedi poi

tell Col0

set Cornice1 to make border panel

-- anche per estetica creo una cornice attorno agli elementi d'interfaccia che poi creo

tell Cornice1

set Col1 to make dialog column

-- Gli elementi della interfaccia utente sono ordinati in colonne

-- la prima colonna avrà il suo identificativo dentro la variabile "Col1"

tell Col1

-- cosa faccio fare alla prima colonna della finestra?

make static text with properties {static label:"Campioni da generare:"}

make static text with properties {static label:"Tipo Colore:"}

make static text with properties {static label:"Metodo Colore:"}

make static text with properties {static label:"Limite inchiostri:"}

make static text with properties {static label:"Limite nero:"}

-- ho creato 5 blocchetti di testo che saranno mostrati nella Colonna più a sinistra.

-- alla loro destra appariranno menu e altre parti in cui indicare i dati

end tell -- qui finisco di dare istruzioni alla Colonna N° 1

set Col2 to make dialog column

-- Creo una seconda colonna nella finestra

tell Col2

set NumColori to make integer editbox with properties {small nudge:1, large nudge:5, minimum value:1, maximum

```

    value:100, edit value:10}
-- creo una variabile "NumColori" in cui metto il riferimento a un
  riquadro in cui indicare il numero di campioni da generare
-- il riquadro (editbox) consente di indicare solo numeri interi
  (integer)
-- I suoi parametri sono:
-- small nudge: X (il valore aumenta di X ogni volta che si
  preme "comando-freccia in su" )
-- large nudge: X (il valore aumenta di X ogni volta che si preme"
  comando-maiusc-freccia in su" )
-- minimum value: X (I valori indicati dovranno almeno pari a X)
-- maximum value: X (I valori indicati dovranno non più grand di
  X)
-- edit value: X (X è il valore proposto)

set TipoColore to make dropdown with properties {string list:
  {"Composito", "Tinta Piatta"}, selected index:0, min width:
  130}
-- creo una variabile "TipoColore" in cui metto il riferimento a un
  menu (dropdown) da cui scegliere il tipo di colore
-- il menu (dropdown) consente di scegliere solo valori
  predefiniti
-- i valori predefiniti sono quelli indicati come parametro della "
  string list"
-- la voce proposta (selected index) è la prima voce del menu
-- il conteggio delle voci di menu inizia sempre da 0 (zero)
-- il menu sarà largo almeno 130 pixel (min width) così da
  mostrare correttamente le sue voci

set MetodoColore to make dropdown with properties {string
  list:{"CMYK", "RGB"}, selected index:0, min width:130}
-- creo una variabile "MetodoColore" in cui metto il riferimento a
  un menu (dropdown) da cui scegliere il metodo con cui
  creare i colori

set LimiteInchiostri to make percent editbox with properties
  {small nudge:1, large nudge:5, minimum value:100,
  maximum value:400, edit value:270}
-- in questo caso, il riquadro di testo mostra i valori con a fianco
  il simbolo "percentuale"
-- Infatti, è un riquadro di tipo "percent editbox"

set LimiteNero to make percent editbox with properties {small
  nudge:1, large nudge:5, minimum value:0, maximum
  value:100, edit value:50}
-- Idem, come il precedente. In questo caso il massimo valore è
  il 100%

end tell -- qui finisco di dare istruzioni alla Colonna N° 2
end tell -- qui finisco di dare istruzioni alle colonne dentro al rettangolo

set Cornice2 to make border panel
-- creo una seconda cornice in cui inserire delle note
tell Cornice2
  make static text with properties {static label:"I limiti inchiostri valgono

```

solo per colori CMYK.} }

end tell

end tell

end tell

show myDialog -- mostra la finestra che InDesign si è creato

-- senza il comando "Show" la finestra non apparirebbe!

if the result then -- È un modo conciso per verificare che l'utente abbia premuto OK

-- se si preme OK viene messo nella variabile "result" il valore "true" che significa "vero"

-- le istruzioni che seguono sono pertanto eseguite solo se l'utente preme "OK"

set NumColori to the edit value of NumColori

-- con questo comando uso la variabile "NumColori" per indicare il valore (edit value) del

-- riquadro di testo il cui nome era prima contenuto nella medesima variabile.

-- NumColori conterrà il numero di campioni da generare

-- "Riciclare" una variabile anziché crearne una nuova, significa risparmiare qualche bit di memoria

set TipoColore to selected index of TipoColore

-- La variabile "TipoColore" conterrà non il nome della voce di menu scelta

-- circa il tipo di colore (colore composito o tinta piatta) bensì il numero

-- della voce scelta. Sarà "0" se si è scelto "Composito"

-- questa tecnica consente una maggiore flessibilità in caso si decida di tradurre lo script

set MetodoColore to selected index of MetodoColore

-- idem come il menu precedente

-- in questo caso sarà relativo alla scelta tra RGB o CMYK

set LimiteInchiostri to the edit value of LimiteInchiostri

-- "LimiteInchiostri" contiene il valore massimo come somma delle singole percentuali di inchiostro

set LimiteNero to the edit value of LimiteNero

-- Massima percentuale del nero

destroy myDialog

-- Dopo avere creato la finestra, averla mostrata e averne valutato i valori indicati dall'utente,

-- tolgo dalla memoria di InDesign la finestra. Risparmio memoria.

if MetodoColore = 0 then -- se si è scelto CMYK come tipo colore

repeat NumColori times -- ripeti tante volte quanti i colori da creare

-- se scrivessimo in Italiano, sarebbe proprio "ripeti xx volte"

my CreaCampioniCMYK(TipoColore, LimiteInchiostri, LimiteNero) --
crea colori in CMYK

-- La funzione "CreaCampioniCMYK" fa arte dello script;

-- non appartiene a InDesign o a AppleScript

-- e pertanto viene identificata da "My" per indicare che è una parte del mio script

-- tra le parentesi appaiono i parametri che passo alla funzione

```

end repeat
else -- con riferimento alla voce "if MetodoColore = 0 then"
  -- significa che se NON ho scelto come metodo colore la prima voce del
  menu
  -- per forza ho scelto la seconda (ossia RGB) e mi comporto di
  conseguenza
  repeat NumColori times -- ripeti tante volte quanti i colori da creare
    my CreaCampioniRGB(TipoColore) -- crea colori in RGB
    -- vedi funzione "CreaCampioniCMYK"
    -- in questo caso il limite degli inchiostri non sarà preso in
    considerazione
  end repeat
end if -- ho finito di valutare il metodo colore scelto
end if -- ho finito di valutare se l'utente aveva premuto OK nella finestra
end tell

on CreaCampioniCMYK(TipoColore, LimiteInchiostri, LimiteNero)
  -- Questa mia funzione riceve i tre parametri indicati tra parentesi.
  -- Per comprensibilità, mantengo i nomi delle variabili originali

  set TotaleInchiostri to 999
  -- siccome la prossima funzione si ferma solo se ha raccolto dati validi ossia se
  -- la somma della percentuale degli inchiostri è inferiore al massimo stabilito
  -- metto nella variabile un valore molto alto altrimenti la funzione non partirebbe
  -- trovando un valore pari 0 quindi inferiore al limite

  repeat until TotaleInchiostri ≤ LimiteInchiostri
    -- Come detto prima, il valore totale degli inchiostri deve essere inferiore o pari
    al limite stabilito.
    -- per ora è ancora pari a 999 come impostato quindi sicuramente, almeno una
    volta, queste
    -- operazioni sono svolte.

    set C to (round (random number (100.0)))
    -- Nella variabile "C" che rappresenta la percentuale del Ciano, mette un
    numero
    -- che pari a un numero casuale (random number) non superiore a 100
    -- il quale viene arrotondato (round) al numero intero più vicino.

    set M to (round (random number (100.0)))
    set Y to (round (random number (100.0)))
    set K to (round (random number LimiteNero))
    -- Nel caso del nero, il limite è pari al valore inserito dall'utente e non al 100%
    come per gli altri colori

    set TotaleInchiostri to C + M + Y + K
    -- La somma delle percentuali di inchiostro viene messa nella variabile "
    TotaleInchiostri"
    -- Se detta somma è inferiore al limite, si prosegue oltre
    -- altrimenti, come accade con i cicli ripetitivi, la serie di operazioni viene
    ripetuta
    -- fintanto che la condizione di verifica non viene raggiunta.
    -- Pertanto, se la somma degli inchiostri sarà superiore al limite, si generano
    altri numeri casuali e così via.
  
```

end repeat

if TipoColore = 0 **then**

-- se l'utente vuole un colore di tipo "process" ossia da generare con le tinte primarie

-- allora arrivo in questa funzione

tell application "InDesign CS" *-- richiamo InDesign*

tell the active document *-- il colore in questo caso viene creato per il documento attivo*

try *-- con try, "provo" a creare un colore. Se trovasse un colore che già esiste evito di avere messaggi di errore. È una scappatoia.*

make color with properties {model:process, space:CMYK, color value:{C, M, Y, K}, name:(((("C=" & C as string) & " M=" & M as string) & " Y=" & Y as string) & " K=" & K as string)}

-- make color fa sì che venga creato un colore secondo quanto indicato, ossia:

-- model:process È un colore di tipo Process ossia realizzato con le tinte primarie

-- space:CMYK Il colore viene indicato come percentuali di C, M, Y e K.

-- name:(((("C=" & C as string) eccetera Il nome sarà composto da "C="

-- seguito dalla percentuale di Ciano seguito da " M=" e dalla percentuale di magenta e così via

end try

end tell

end tell

else

tell application "InDesign CS" *-- richiamo InDesign*

tell the active document *-- il colore in questo caso viene creato per il documento attivo*

try *-- con try, "provo" a creare un colore. Se trovasse un colore che già esiste evito di avere messaggi di errore*

set MyCol to make color with properties {model:spot, space:CMYK, color value:{C, M, Y, K}, name:(((("C=" & C as string) & " M=" & M as string) & " Y=" & Y as string) & " K=" & K as string)}

-- in questo caso, il colore è un colore Spot ossia una tinta piatta

end try

end tell

end tell

end if

end CreaCampioniCMYK

on CreaCampioniRGB(TipoColore)

-- questa funzione è molto simile alla precedente.

-- I commenti sono inutili dato che si utilizzano i medesimi comandi anche se con parametri differenti

set R to (round (random number (255)))

set G to (round (random number (255)))

set B to (round (random number (255)))

if TipoColore = 0 **then**

```
tell application "InDesign CS"
  tell the active document
    try
      make color with properties {model:process, space:RGB, color
        value:{R, G, B}, name:((( "R=" & R as string) & " G=" & G as
        string) & " B=" & B as string)}
      -- in questo caso, il metodo colore è RGB
    end try
  end tell
end tell
else
  tell application "InDesign CS"
    tell the active document
      try
        make color with properties {model:spot, space:RGB, color
          value:{R, G, B}, name:((( "R=" & R as string) & " G=" & G as
          string) & " B=" & B as string)}
      end try
    end tell
  end tell
end if
end CreaCampioniRGB
```