



Processing è un linguaggio di programmazione open source dedicato all'elaborazione di immagini, animazioni e suoni.

I suoi inventori sono Ben Fry (Broad Institute) e Casey Reas (UCLA)

[www.processing.org](http://www.processing.org)

SDK

<http://dev.processing.org/>

HD

<http://hardware.processing.org/>

Mobile

<http://mobile.processing.org/>



Processing è disponibile per:

Windows

Linux

Mac OSX

ver: 0124 Beta

Principali Caratteristiche:

Color (hue,saturation,brightness)

Image (trasparency, hystogram)

3D Image (declare,transform,light)

Transform (translate,scale,rotate)

Motion (linear,moving on curve)

Input (mouse,keyboard)

Simulate (particle,smoke,fluid)

Video

Net (http,tcp/udp)



### Particle Show:

SuperNova (Robert Hodgkin)

[video\SuperNova.mov](#)

Grass (Barbarian Group)

[video\Grass.mov](#)

### Image Processing:

Shadow Monster (Philip Worthington)

[video\shadow Monsters3.mov](#)

Manual Input Session (Levin/Lieberman)

Messa di Voce (Levin/Lieberman)

[video\MessaDiVoce.mov](#)

### Multimonitor:

Mpe (Shiffman)

[video\mpe.mov](#)

[video\run.mov](#)

Kaleido (Suzung Kim)

[video\kaleido.mov](#)

### Hardware:

Piixel Roller (rAndom International)

[video\Pixel Roller image.mov](#)

SuddenMotion (Shiffman student)

[video\fontana.mov](#)



examples



lib



libraries



reference



ICE\_JNIRegistry.dll



jikes



processing



revisions  
Documento di testo  
118 KB

Libraries:

LibrariesName/library



data



Keyboard  
File PDE  
1 KB



Osc  
File PDE  
1 KB



Danza\_4x4\_aif\_cerchi\_optimized  
File PDE  
7 KB



MusicQuads  
File PDE  
3 KB



utilsDraw  
File PDE  
1 KB

Nome Progetto:

Danza\_4x4\_cerchi\_optimized

Estensione file:

.pde

data:

Tutti i contenuti multimediali  
e i font



```

Processing - 0124 Beta
File Edit Sketch Tools Help
Danza_4x4_aif_OK Keyboard Osc Utilities
import krister.Ess.*;

//Suono
AudioChannel myChannel0,myChannel1,myChannel2,myChannel3,my
FFT myFFT;

int bufS0,bufS1,bufS2,bufS3,bufS4,bufS5,bufS6,bufS7,bufS8,b
int bufD0,bufD1,bufD2,bufD3,bufD4,bufD5,bufD6,bufD7,bufD8,b

int musicSet=0;
boolean flagMusic=false;

Osc o;
Utilities u;
//deve essere lungo N°di punti mocap
int[] suonoScelto;
//deve essere lungo in numero uguale ai punti
int [] ultimoScelto={16,16,16,16,16};

int cellW,cellH;

```

[Abridged \(A-Z\)](#) \ [Extended \(A-Z\)](#) \ [Changes](#)

Language (API). The Processing Language has been designed to facilitate the creation of sophisticated visual and conceptual structures.

#### Structure

[.](#) ([dot](#))  
[setup\(\)](#)  
[void](#)  
[class](#)  
[size\(\)](#)  
[/\\*\\* \\*/](#) ([multiline comment](#))  
[false](#)  
[noLoop\(\)](#)  
[this](#)  
[true](#)  
[new](#)  
[/\(comment\)](#)  
[.\(comma\)](#)  
[delay\(\)](#)  
[super](#)  
[exit\(\)](#)  
[{ }](#) ([curly braces](#))  
[/\\*\\* \\*/](#) ([doc comment](#))  
[draw\(\)](#)  
[loop\(\)](#)  
[implements](#)  
[redraw\(\)](#)

#### Shape

*2D Primitives*  
[triangle\(\)](#)  
[line\(\)](#)  
[arc\(\)](#)  
[point\(\)](#)  
[quad\(\)](#)  
[ellipse\(\)](#)  
[rect\(\)](#)

*Curves*  
[bezierTangent\(\)](#)  
[bezierDetail\(\)](#)  
[curveTightness\(\)](#)  
[bezierPoint\(\)](#)  
[curveDetail\(\)](#)  
[curvePoint\(\)](#)  
[curve\(\)](#)  
[bezier\(\)](#)

*3D Primitives*

#### Color

*Setting*  
[background\(\)](#)  
[colorMode\(\)](#)  
[stroke\(\)](#)  
[noFill\(\)](#)  
[noStroke\(\)](#)  
[fill\(\)](#)

*Creating & Reading*  
[red\(\)](#)  
[brightness\(\)](#)  
[blue\(\)](#)  
[saturation\(\)](#)  
[green\(\)](#)  
[hue\(\)](#)  
[alpha\(\)](#)  
[color\(\)](#)  
[blendColor\(\)](#)  
[lerpColor\(\)](#)



```
//importo le librerie necessarie
import processing.candy.*;

//Eseguo le istruzioni di SetUP
//Istanzio le classi e le var di cui ho bisogno
//Genero la finestra di output
setup
{
  //.....
}

//E' composta da un ciclo infinito che
//aggiorna la visualizzazione a schermo
//e esegue i metodi che elaborano l'immagine
draw
{
  //....
}

//Gestione della tastiera (Nb anche su file esterno)
void keyReleased()
{
  //.....
}
```



**File/Sketchbook/Example/Structure/**

Coordinates:

Background();

Disegnare una croce dato un array di punti;

**File/Sketchbook/Example/Form/**

ShapePrimitives:

Stroke(); Fill();

Disegnare forme colorate

**File/Sketchbook/Example/Drawing/**

Pattern:

**File/Sketchbook/Example/Image/**Displaying:

```
loadImage(); //( .gif, .jpg, .tga, .png)
```

```
image(img, x, y, width, height);
```

Alpha Mask:

Usare gli altri metodi di PImage

Trasparency:

```
tint();
```

Controllare i parametri di tint da tastiera

**File/Sketchbook/Example/Transform/**Translate:

Giocare coi parametri di translate e size  
Provare background, fill e stroke

Scale:

```
rectMode();
```

```
pushMatrix(); popMatrix();
```