



Tutorial Processing 3.0

Programación de Arte Digital.

COLECCIÓN DE APLICACIONES GRATUITAS PARA CONTEXTOS EDUCATIVOS

Plan Integral de Educación Digital

Gerencia Operativa Incorporación de Tecnologías (InTec)

buenosaires.gob.ar/educacion   educaciónBA



Buenos Aires Ciudad

Ministerio de Educación del Gobierno de la Ciudad de Buenos Aires

01-02-2026



EN TODO ESTÁS VOS

Prólogo

Este tutorial se enmarca dentro de los lineamientos del Plan Integral de Educación Digital (PIED) del Ministerio de Educación del Gobierno de la Ciudad Autónoma de Buenos Aires que busca integrar los procesos de enseñanza y de aprendizaje de las instituciones educativas a la cultura digital.

Uno de los objetivos del PIED es “fomentar el conocimiento y la apropiación crítica de las Tecnologías de la Información y de la Comunicación (TIC) en la comunidad educativa y en la sociedad en general”.

Cada una de las aplicaciones que forman parte de este banco de recursos son herramientas que, utilizándolas de forma creativa, permiten aprender y jugar en entornos digitales. El juego es una poderosa fuente de motivación para los alumnos y favorece la construcción del saber. Todas las aplicaciones son de uso libre y pueden descargarse gratuitamente de Internet e instalarse en cualquier computadora. De esta manera, se promueve la igualdad de oportunidades y posibilidades para que todos puedan acceder a herramientas que desarrolleen la creatividad.

En cada uno de los tutoriales se presentan “consideraciones pedagógicas” que funcionan como disparadores pero que no deben limitar a los usuarios a explorar y desarrollar sus propios usos educativos.

La aplicación de este tutorial no constituye por sí misma una propuesta pedagógica. Su funcionalidad cobra sentido cuando se integra a una actividad. Cada docente o persona que quiera utilizar estos recursos podrá construir su propio recorrido.

Índice

¿Qué es?	4
Requerimientos técnicos.....	4
Consideraciones pedagógicas	4
Nociones básicas.....	5
▪ Descarga del programa.....	5
▪ Acceder al programa	6
▪ Interfaz de Processing	7
▪ Ubicación de coordenadas	9
Paso a Paso	10
▪ Comenzar nuevo proyecto.....	10
▪ Colocar comentarios.....	10
▪ Instrucciones para dibujar.....	11
▪ Ejemplo en modo básico	11
▪ Funciones	13
▪ Guardar un proyecto	16
▪ Abrir un proyecto	17
▪ Importar librería.....	17
▪ Agregar una imagen como fondo.....	19
Enlaces de interés	23

¿Qué es?

Processing es un lenguaje de programación y entorno de desarrollo integrado de código abierto basado en Java, que sirve como medio para la enseñanza y producción de proyectos multimedia e interactivos de diseño digital. Es desarrollado por artistas y diseñadores como una herramienta alternativa al **software** propietario. Puede ser utilizado tanto para aplicaciones locales como para aplicaciones en la web (*Applets*).

Requerimientos técnicos

- Puede utilizarse bajo los sistemas operativos Windows y GNU/Linux.
- Se descarga de manera gratuita.
- No requiere acceso a internet una vez instalado el programa en el equipo.
- URL para descargar el programa: <https://processing.org/>

Consideraciones pedagógicas

Nivel: primario (segundo ciclo) y medio.

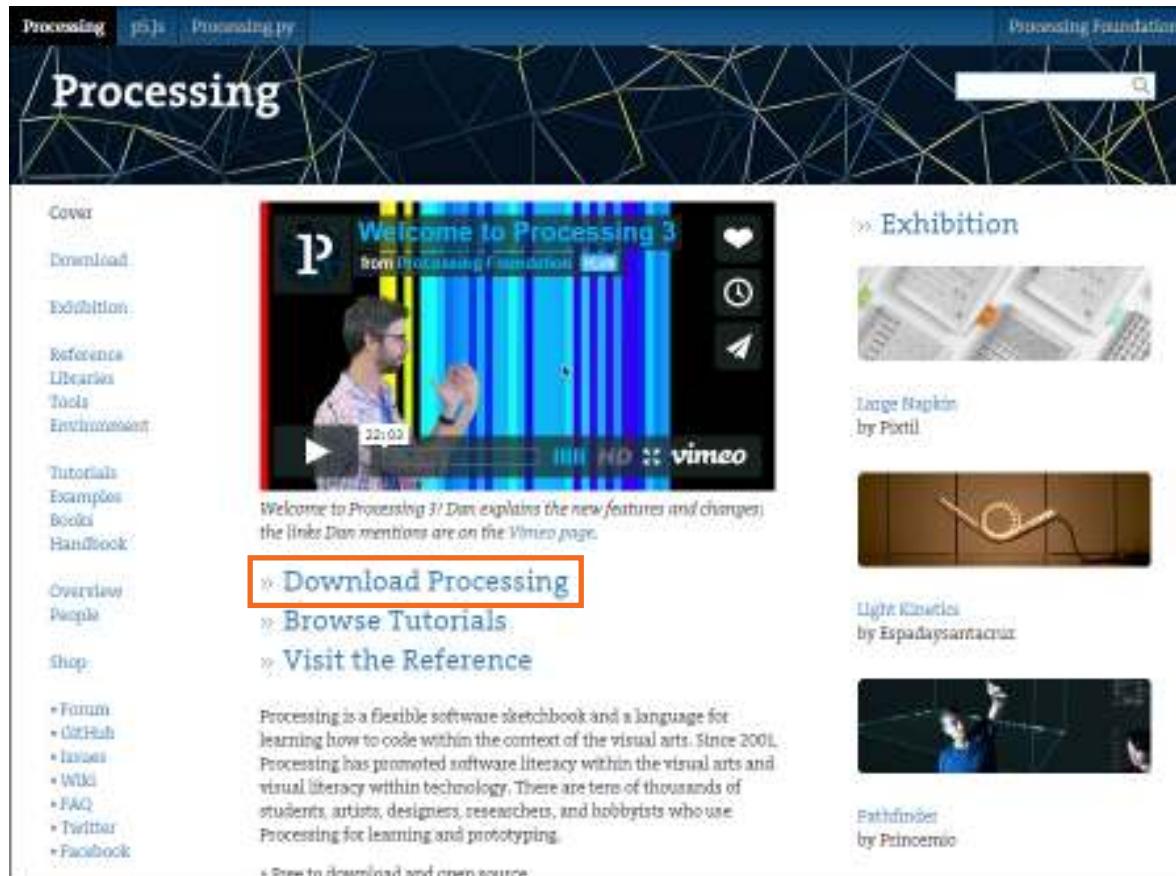
Áreas sugeridas: todas

- Desde el área de Plástica se pueden analizar las propiedades de los objetos virtuales 3D, experimentar con perspectivas, formas y colores.
- Construir un objeto básico mediante un lenguaje de programación.
- Interactuar con un entorno de programación para recrear producciones mediante la modificación de variables.

Nociones básicas

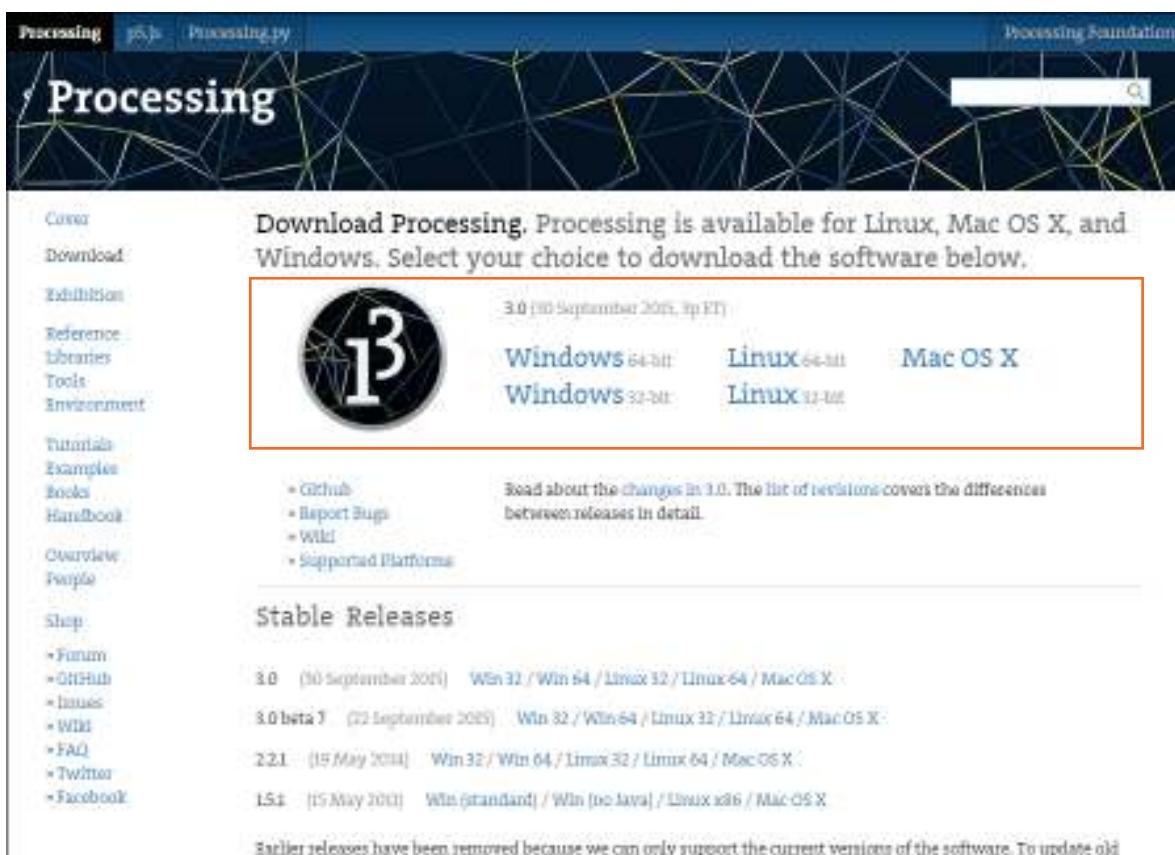
► Descarga del programa

La descarga del programa se puede realizar desde la página <https://processing.org/>. Elegir la opción **Download Processing**.



The screenshot shows the official Processing Foundation website. The main navigation bar at the top includes 'Processing', 'p5.js', 'Processing.py', and 'Processing Foundation'. Below the navigation, there's a large 'Processing' logo and a search bar. On the left, a sidebar lists various links: 'Cover', 'Download', 'Exhibition', 'Reference', 'Libraries', 'Tools', 'Environment', 'Tutorials', 'Examples', 'Books', 'Handbook', 'Overview', 'People', 'Shop', and social media links for 'Forum', 'GitHub', 'Issues', 'Wiki', 'FAQ', 'Twitter', and 'Facebook'. The central content area features a video player showing a video titled 'Welcome to Processing 3' from the Processing Foundation. Below the video, there are three links: 'Download Processing' (which is highlighted with a red box), 'Browse Tutorials', and 'Visit the Reference'. To the right of the video, there's a section titled 'Exhibition' with two examples: 'Large Napkin' by Pixtil and 'Light Kinetics' by Espadaysantacruz, each with a small thumbnail image. At the bottom of the main content area, there's a link 'Page to download and open source'.

La versión utilizada para la elaboración de este tutorial es **3.0**



Processing | pde | Processing.py | Processing Foundation | Search

Processing

Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below.

3.0 (30 September 2015, 3p ET)

Windows 64-bit | **Windows 32-bit** | **Linux 64-bit** | **Linux 32-bit** | **Mac OS X**

3.0 (30 September 2015, 3p ET)

Read about the changes in 3.0. The list of revisions covers the differences between releases in detail.

• GitHub | • Report Bugs | • Wiki | • Supported Platforms

Stable Releases

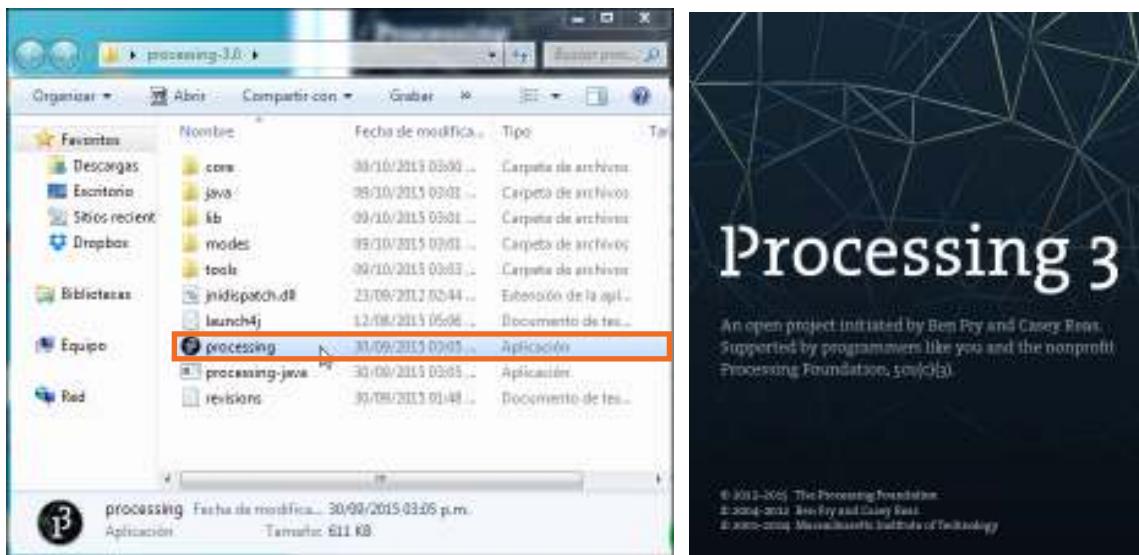
3.0 (30 September 2015) | Win 32 / Win 64 / Linux 32 / Linux 64 / Mac OS X | 3.0 beta 3 (22 September 2015) | Win 32 / Win 64 / Linux 32 / Linux 64 / Mac OS X | 3.2.1 (15 May 2015) | Win 32 / Win 64 / Linux 32 / Linux 64 / Mac OS X | 1.5.1 (15 May 2013) | Win (standard) / Win (no Java) / Linux x86 / Mac OS X

Earlier releases have been removed because we can only support the current versions of the software. To update old

Una vez descargado el archivo será necesario descomprimirlo para comenzar a utilizarlo.

► Acceder al programa

Una vez descomprimido el programa, abrir el archivo **processing**.



processing-3.0

Organizar | Abrir | Compartir con | Grabar |

Favoritos	Nombre	Fecha de modificación	Tipo
Descargas	core	30/10/2015 03:00...	Carpetas de archivos
Escritorio	java	30/10/2015 03:01...	Carpetas de archivos
Síos recient	lib	09/10/2015 03:01...	Carpetas de archivos
Dropbox	modes	09/10/2015 03:01...	Carpetas de archivos
Biblioteca	tools	09/10/2015 03:01...	Carpetas de archivos
	jnidispatch.dll	23/09/2015 02:44...	Extensión de la apl.
	launch4j	12/08/2015 05:06...	Documento de texto
Equipo	processing	30/09/2015 03:01...	Aplicación
Red	processing-java	30/09/2015 03:01...	Aplicación
	revisions	30/08/2015 01:41...	Documento de texto

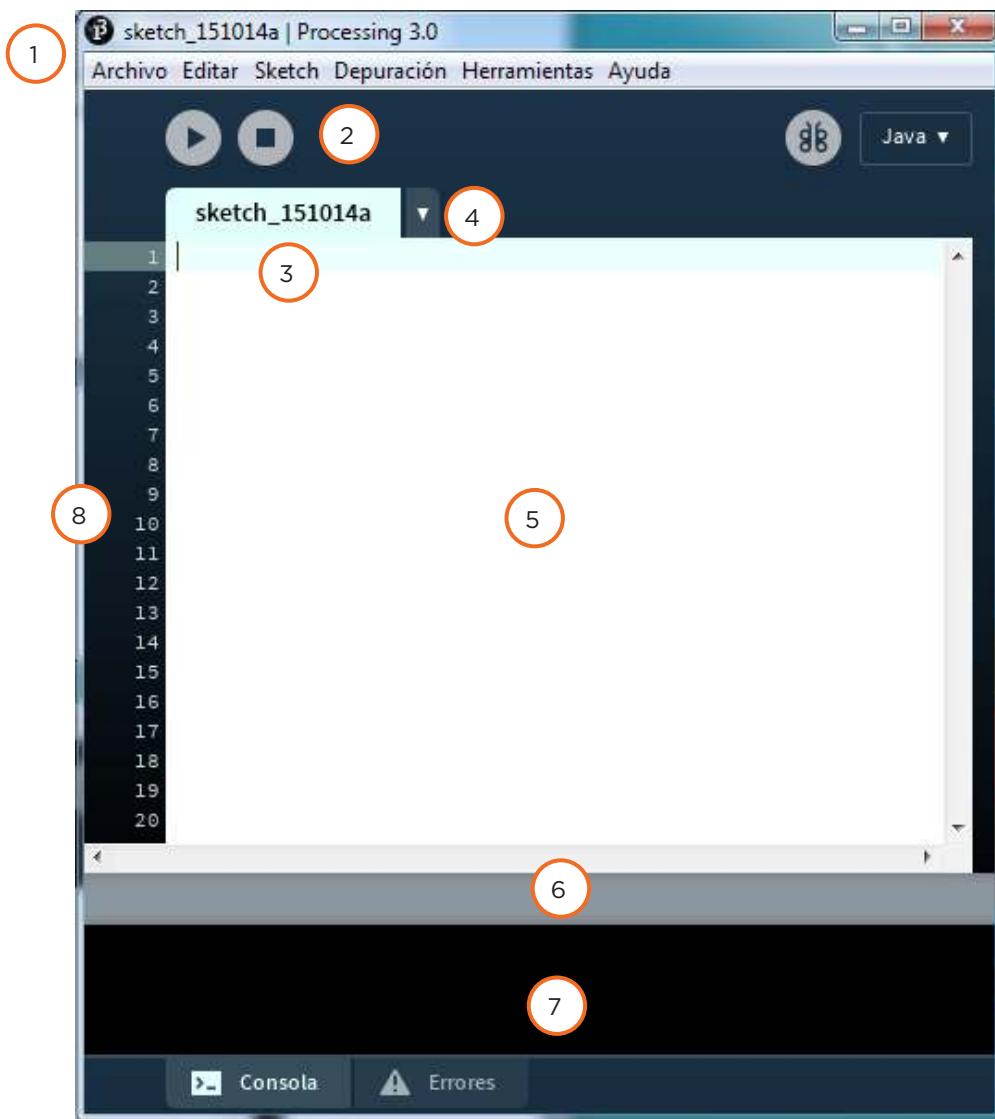
processing | Fecha de modificación: 30/09/2015 03:05 p.m. | Tamaño: 611 KB

Processing 3

An open project initiated by Ben Fry and Casey Reas. Supported by programmers like you and the nonprofit Processing Foundation, 501(c)(3).

© 2002-2015 The Processing Foundation. © 2004-2012 Ben Fry and Casey Reas. © 2001-2004 Massachusetts Institute of Technology

► Interfaz de Processing



1. Barra de menú

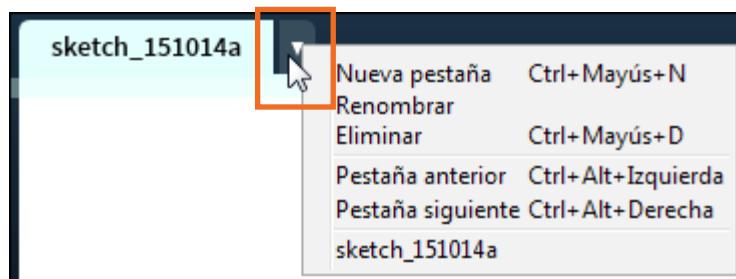
2. **Barra de herramientas.** Cuenta con los botones de acciones más comunes.



- a. **Ejecutar (Run).** Corre el *sketch*, compila el código y abre una nueva ventana.
- b. **Detener (Stop).** Termina el *sketch* que se ejecuta.

3. **Solapas (Tabs)**. Permiten manejar los archivos

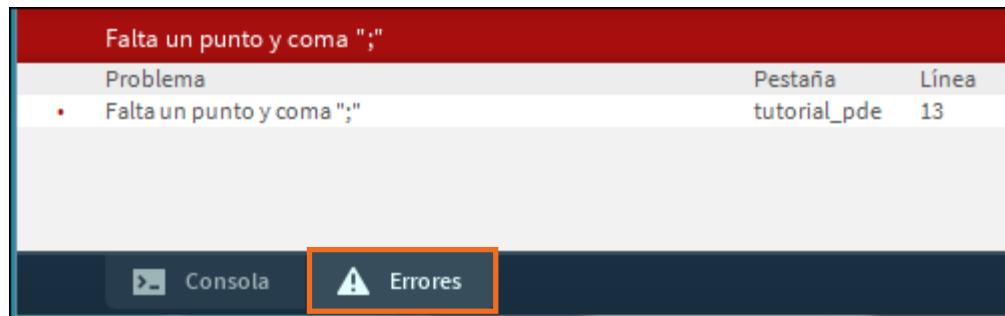
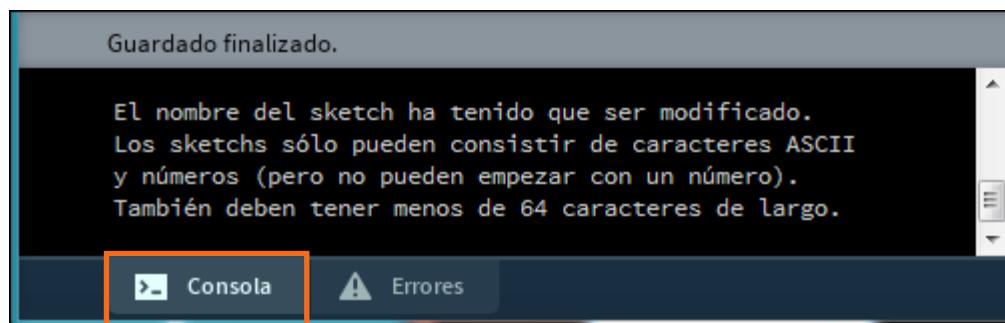
4. Gestión de pestañas o solapas



5. **Editor de texto**. Área de escritura de código.

6. **Área de mensajes**. Muestra mensajes mientras se guarda y exporta.

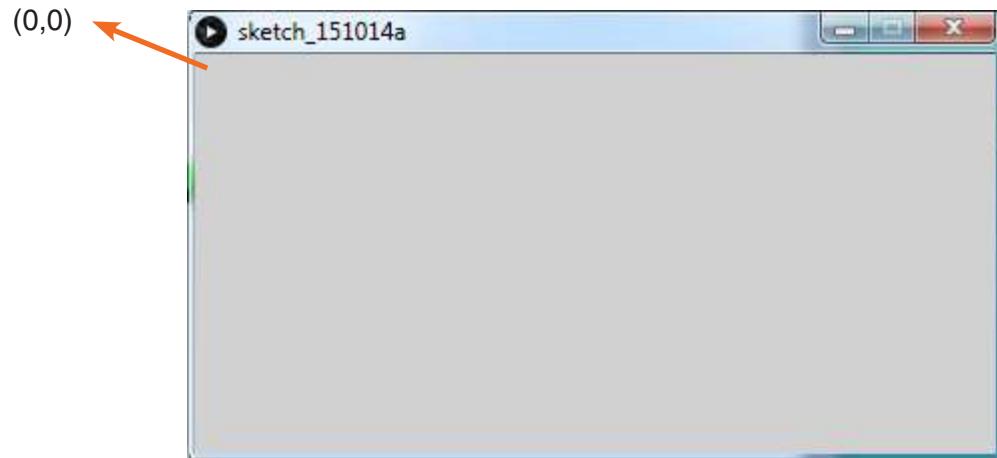
7. **Consola y Errores**. Muestra mensajes de acciones o bien de error.



8. **Número de línea**

► Ubicación de coordenadas

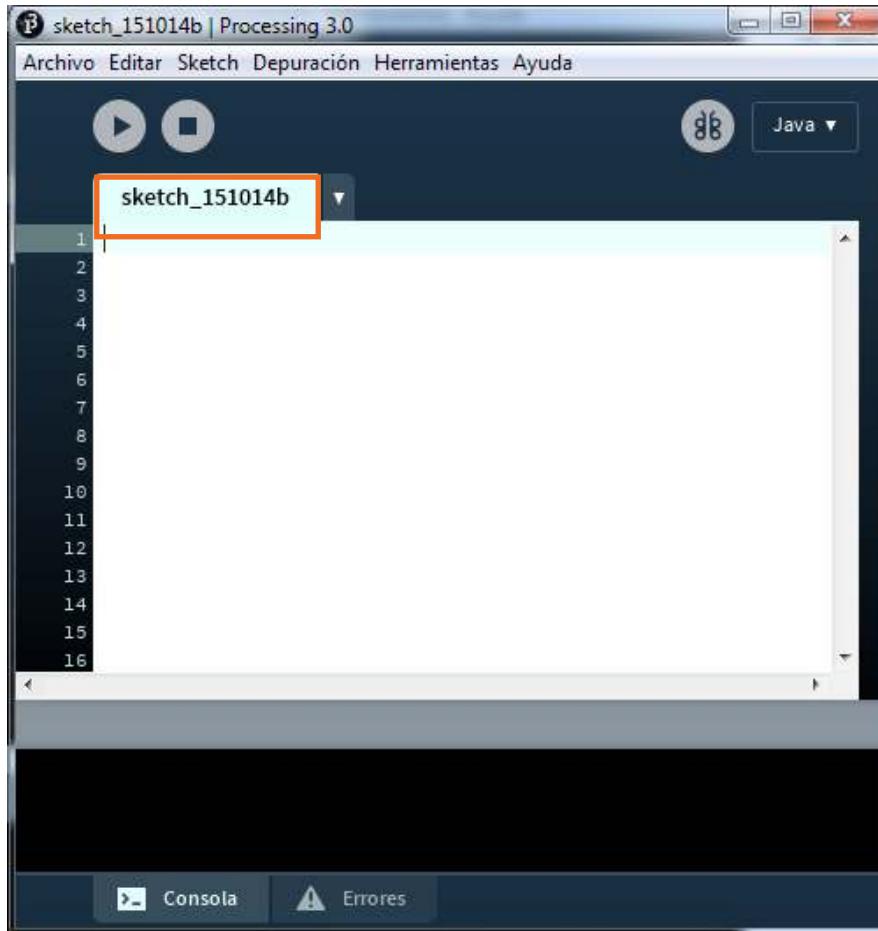
Un elemento a tener en cuenta al momento de ubicar objetos en pantalla es que las coordenadas 0,0 se encuentran en el extremo superior izquierdo de la ventana.



Paso a Paso

► Comenzar nuevo proyecto

Para comenzar un nuevo proyecto, elegir del menú **Archivo** la opción **Nuevo** o presionar la combinación de teclas **CTRL + N**. Se verá a continuación la siguiente ventana:



► Colocar comentarios

El programa permite colocar dos tipos de comentarios:

```
// Este comentario va hasta el final de línea
/* Este comentario es
multilínea */
```

► Instrucciones para dibujar

- **Establecer color**

(r,g,b) = rojo (*red*), verde (*green*) y azul (*blue*). El valor de color varía desde 0 a 255, mientras más se acerca al 255 más se acerca al color.

- **Color de fondo de la aplicación**

background(r,g,b);
background(color);

- **Color de línea y borde**

stroke(r,g,b);
stroke(color);

- **Punto en pantalla con el color de stroke**

point(x,y);

- **Línea entre dos puntos de coordenadas (x1,y1) y (x2,y2)**

line(x1,y1,x2,y2);

- **Rectángulo**

rect(x1,y1,ancho,alto);

- **Color de relleno**

fill(r,g,b);
fill(color);

- **Alto y ancho de pantalla**

size(ancho,alto);

► Ejemplo en modo básico

Se dibujará un rectángulo dentro de una pantalla de 400x200, con fondo blanco. El rectángulo tendrá borde color verde y relleno rojo.

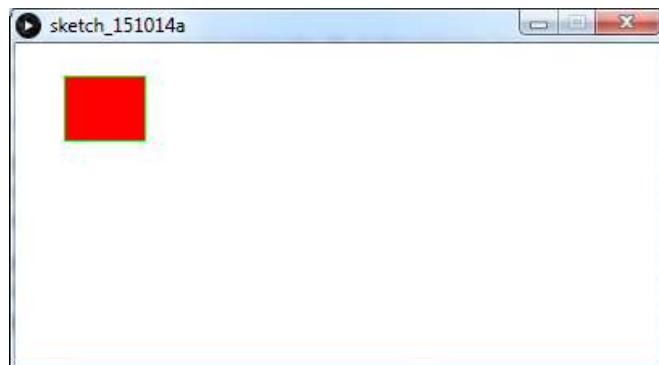
size(400,200); //tamaño de pantalla ancho 400x200 de alto.

background(255,255,255); //color de fondo de pantalla blanco.

```
stroke(0,255,0); //borde de rectángulo verde.  
fill(255, 0, 0); //relleno de rectángulo rojo.  
rect(30, 20, 50, 40); //rectángulo ubicado en 30 del eje X y 20 del Y, con un  
ancho de 50 y alto de 40.
```



Luego de ingresar los comandos, presionar el botón **Ejecutar** para compilar el programa y mostrar el resultado en una nueva ventana:



► Funciones

Una **función** puede definirse como un conjunto de instrucciones que permiten procesar las variables para obtener un resultado.

Este modo llamado continuo se basa en dos construcciones: *setup* y *draw*.

- **la función *setup*:** se ejecuta una única vez cuando se inicializa el programa. Se utiliza para definir las propiedades iniciales del ambiente como el color de fondo, cargar imágenes, inicializar variables. Sólo puede existir una función ***setup*** por *sketch*.
- **la función *draw*:** se ejecuta continuamente y es utilizada para dibujar elementos en pantalla. El número de veces que la función ***draw*** es ejecutada puede ser controlado por la función ***delay*** o suministrando el número de marcos por segundo con la función ***framerate***. También es posible ejecutarla una sola vez incluyendo la llamada a la función ***noLoop*** en ***setup***.

Ejemplo en modo continuo

```
void setup () { //abre la función setup
    size(400, 300); //tamaño de pantalla ancho 400x300 de alto
    noStroke(); //sin borde de rectángulo.
    background(0); //color de fondo de pantalla negro, pone 0 es igual a Ω(0,0,0).
    fill(255, 0, 0); //relleno de la esfera rojo.
    noLoop(); //quiere decir que la esfera se repita indefinidamente
} //cierra la función setup
void draw(){ //abre la función draw.
    circles(int(random(400)),int(random(300)),int(random(50))); //le pido que dibuje círculos en ubicación y radio aleatorias (random).
} //cierra la función draw.
void circles(int x, int y, int r)//crea la función círculo y sus variables de ubicación
(x,y)y radio(r)
{//abre función.
ellipse(x,y,r,r); //crea la elipse de ubicación x,y y radios r,r.
} //cierra función.
```

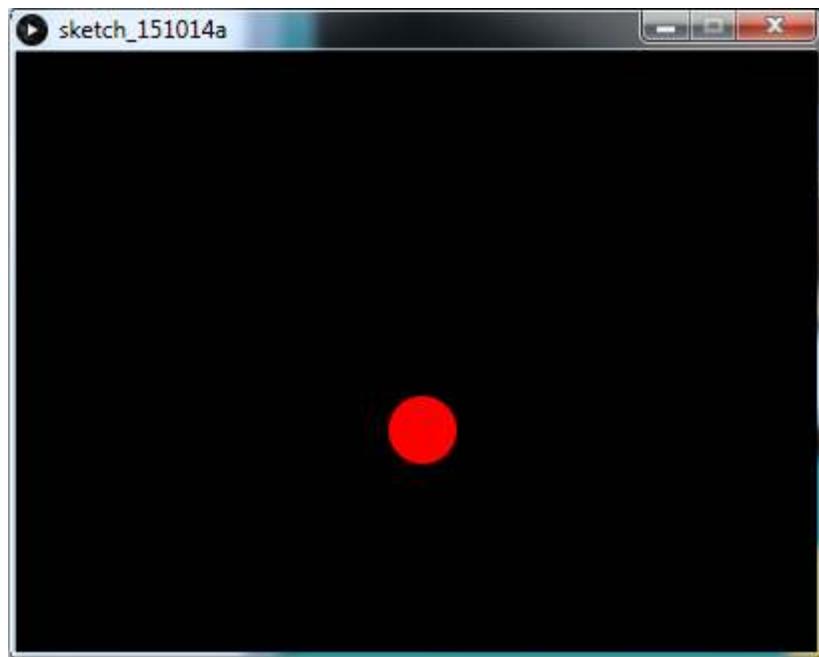


The screenshot shows the Processing 3.0 IDE interface. The title bar reads "sketch_151014a | Processing 3.0". The menu bar includes "Archivo", "Editar", "Sketch", "Depuración", "Herramientas", and "Ayuda". The toolbar has a play button and a stop button. The code editor window contains the following code:

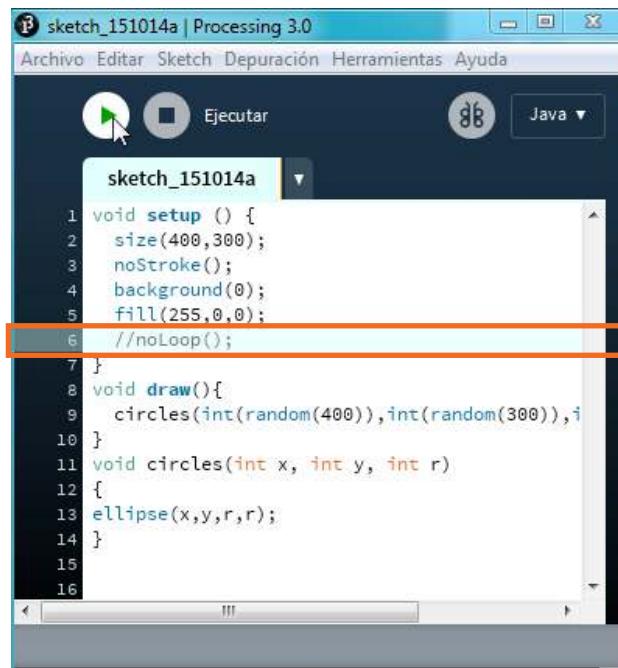
```
1 void setup () {
2     size(400,300);
3     noStroke();
4     background(0);
5     fill(255,0,0);
6     noLoop();
7 }
8 void draw(){
9     circles(int(random(400)),int(random(300)),i
10 }
11 void circles(int x, int y, int r)
12 {
13     ellipse(x,y,r,r);
14 }
15
16
```

The preview window below the code editor is currently empty. At the bottom of the interface, there are tabs for "Consola" and "Errores".

Al ejecutar el procedimiento se obtiene:

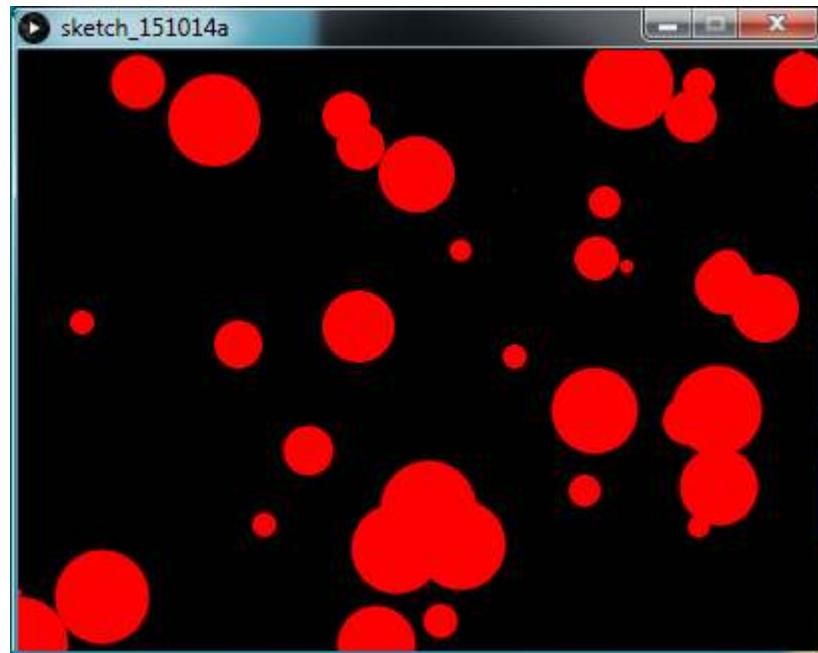


Para que la función se repita indefinidamente, desactivar la orden *noLoop*.



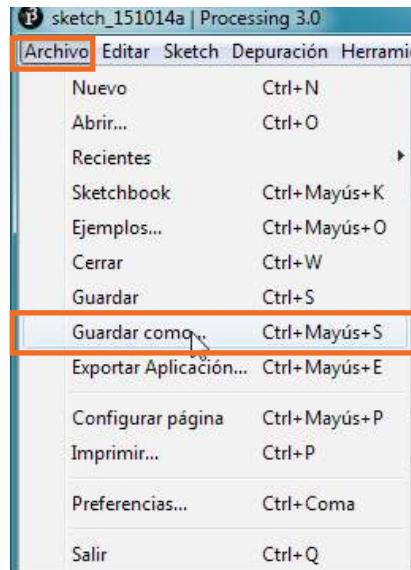
```
1 void setup () {
2   size(400,300);
3   noStroke();
4   background(0);
5   fill(255,0,0);
6   //noLoop();
7 }
8 void draw(){
9   circles(int(random(400)),int(random(300)),10);
10 }
11 void circles(int x, int y, int r)
12 {
13   ellipse(x,y,r,r);
14 }
15
16
```

Así se obtendrá:

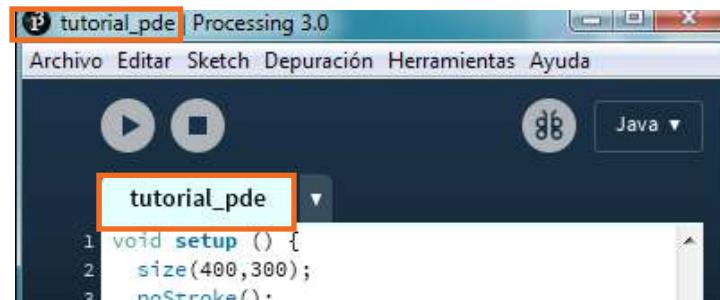


► Guardar un proyecto

Una vez finalizada la actividad, el proyecto debe ser guardado. Elegir del menú **Archivo** la opción **Guardar como...** Indicar a continuación el nombre del proyecto.

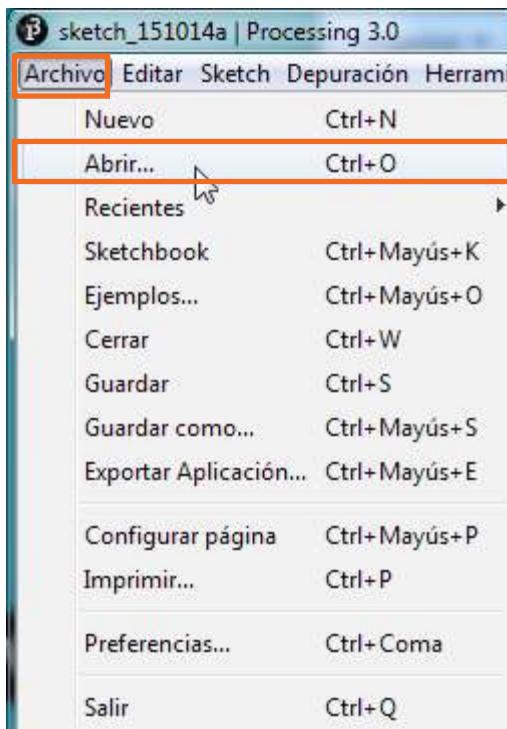


Puede observarse que al guardar el programa, el nombre de la solapa se modifica.



► Abrir un proyecto

Para continuar un proyecto guardado con anterioridad, se debe elegir del menú **Archivo** la opción **Abrir...** o bien presionar la combinación de teclas **CTRL+O**. Seleccionar la carpeta donde se encuentra el trabajo guardado.



► Importar librería

Una **librería** es un conjunto de subprogramas que complementan las posibilidades del programa.

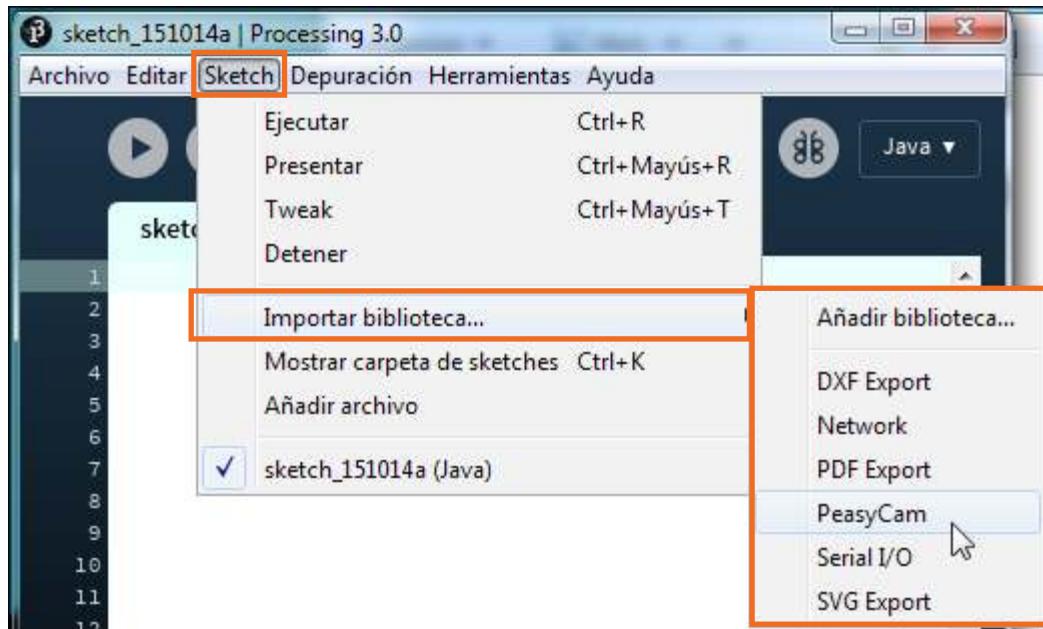
Al descargar el programa *Processing*, el mismo viene con algunas funcionalidades.

Pero al agregarle librerías realizadas por diversos desarrolladores, las posibilidades del programa aumentan y permiten mayores prestaciones.

Descargar la librería elegida de la página <https://processing.org/reference/libraries/>

Copiar la biblioteca descargada dentro de la carpeta **C:\Archivos de Programa\processing-3.0\modes\java\libraries**.

Desde el proyecto de *Processing* que se encuentra abierto importarla eligiendo del menú **Sketch** la opción **Importar biblioteca...** Seleccionar a continuación la librería.



Una vez importada, se verán al inicio las cuatro líneas de código indicando las librerías cargadas.

```

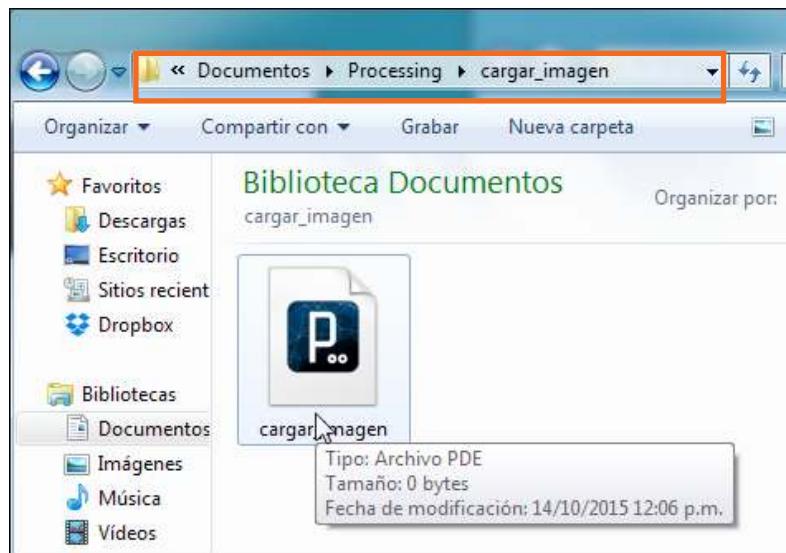
1 import peasy.*;
2 import peasy.org.apache.commons.math.*;
3 import peasy.org.apache.commons.math.geometry.*;
4 import peasy.text.*;

5 void setup(){
6     size(400,300);
7     noStroke();
8     background(0);
9     fill(255,0,0);
10    noLoop();
11 }
12 void draw(){
13     circles(int(random(400)),int(random(300)),int(random(50))
14 }
15 void circles(int x,int y,int r)
16 {
17     ellipse(x,y,r,r);
18 }
19
20

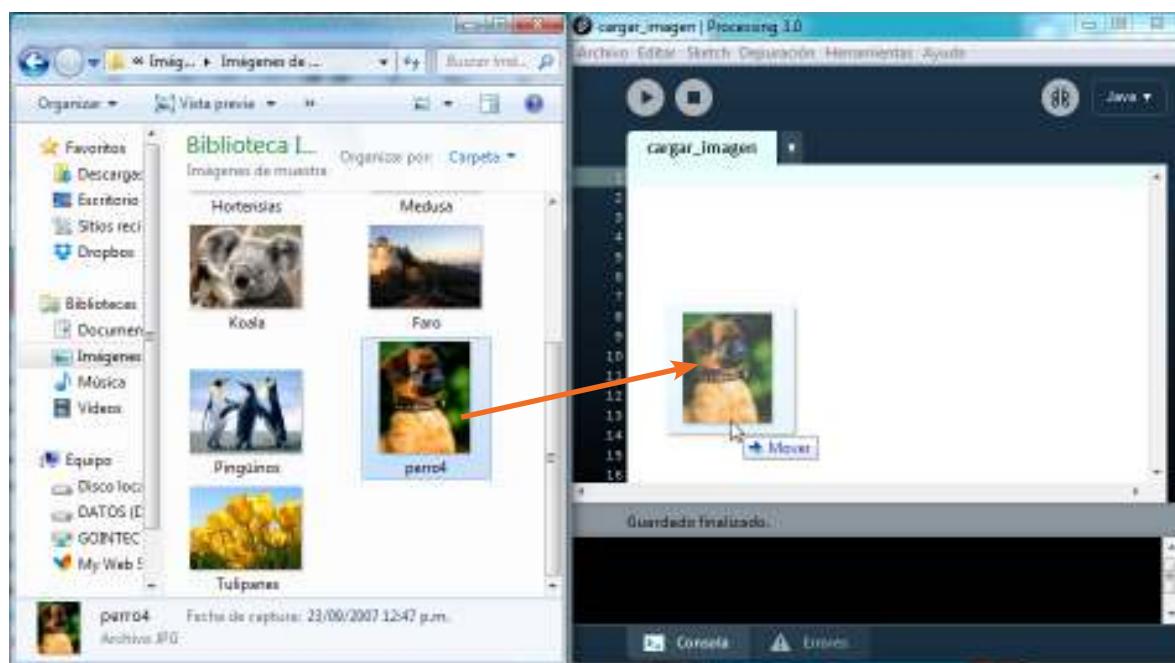
```

► Agregar una imagen como fondo

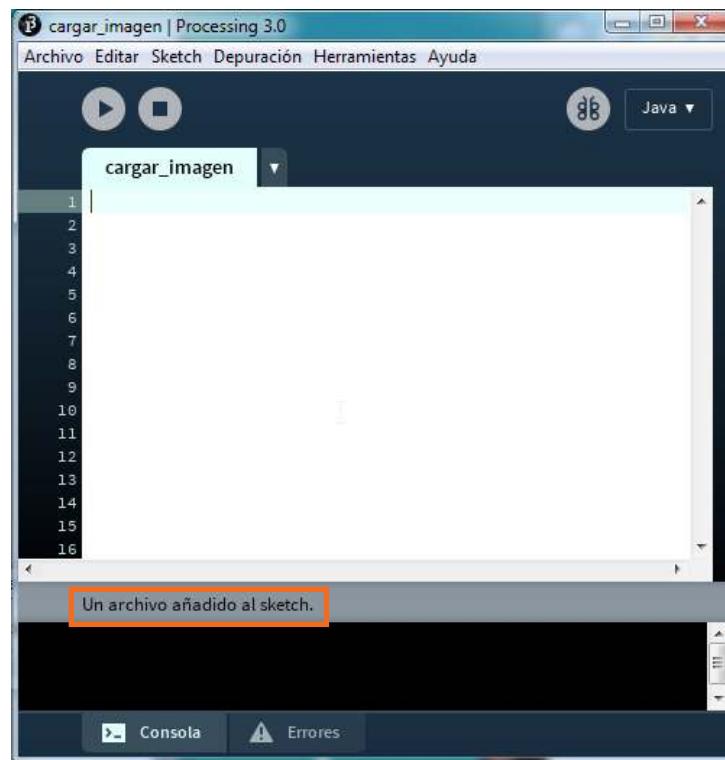
En primer lugar abrir la carpeta donde se encuentra guardado el proyecto **Documentos\Processing\Nombre del proyecto**. Dentro de la carpeta solo se encuentra el archivo **.pde**.



Ubicar en otra ventana la imagen que se desea agregar y arrastrarla al proyecto abierto.



Puede observarse en la parte inferior de la ventana el mensaje **Un archivo añadido al sketch.**

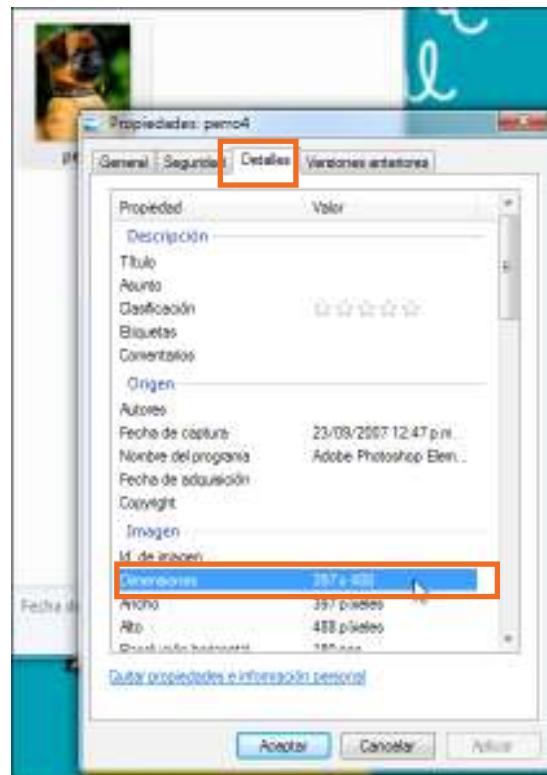


En la carpeta del proyecto se incorporó la carpeta **data** dentro de la cual se encuentra la copia de la imagen.



La imagen se agregará ahora al proyecto. En el siguiente ejemplo se mostrará como imagen de fondo.

Se debe tener en cuenta dentro de las propiedades de la imagen su tamaño.



Este tamaño debe ser igual o menor al tamaño de la ventana que se indica en la orden `size`.

A continuación se indica el código para llamar la imagen:

```
PIImage foto; //Se declara la variable del tipo PIImage y de nombre foto.
void setup(){
size(397,488); // tamaño de la pantalla que debe ser mayor o igual al tamaño de la
imagen.
foto=loadImage ("perro4.jpg"); //Carga la imagen. Se escribe entre comillas el
nombre completo de la imagen.
```

```
background (255,0,0); //Cambia el color de fondo. Solo se verá el color si la imagen
es de menor tamaño
```

```
}
```

```
void draw(){
  image(foto,0,0); //Dibuja la cara en la posición 0,0 que se encuentra en el extremo
  superior izquierdo.
}
```



Se obtiene como resultado:



Enlaces de interés

Sitio oficial: <https://processing.org/>

Librerías recomendadas: <https://processing.org/reference/libraries/>

Contacto:
ayuda.pedagogico.digital@bue.edu.ar



Esta obra se encuentra bajo una Licencia Attribution-NonCommercial-ShareAlike 2.5 Argentina de Creative Commons. Para más información visite <http://creativecommons.org/licenses/by-nc-sa/2.5/ar/>

