

# APRENDIZAJE processing

Free unaffiliated eBook created from **Stack Overflow contributors.** 

#processing

## Tabla de contenido

Acerca de
Capítulo 1: Comenzando con el procesamiento
Observaciones
Versiones
Examples
Instalación y configuración
Hola Mundo4
Capítulo 2: Colores en Procesamiento
Introducción7
Sintaxis7
Parámetros7
Observaciones
Examples7
Notación de color
Capítulo 3: Dibujo de formas básicas
Introducción9
Sintaxis9
Parámetros9
Observaciones
Examples10
Dibujando una linea
Dibujando un rectángulo10
Dibujando un elipse11
Dibujando un triangulo
Dibujando un triangulo12
Capítulo 4: Empezando con Processing para Android13
Observaciones
Examples
Instalación13
Capítulo 5: Formas y funciones básicas utilizando P3D

Sintaxis17
Parámetros17
Examples17
Traducción 3D
Rotación 3D18
Dibujando un cuboide
Capítulo 6: Usando el procesador P3D de Processing
Observaciones
Examples
Empezando
Capítulo 7: Uso de procesamiento con editores de códigos alternativos como Sublime y Atom 23
Examples
Usando el procesamiento con texto sublime
Usando Processing con el editor Atom
Usando Procesando con Eclipse24
Creditos



You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: processing

It is an unofficial and free processing ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official processing.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

# Capítulo 1: Comenzando con el procesamiento

### Observaciones

El procesamiento es un lenguaje y entorno de programación de código abierto para personas que desean crear imágenes, animaciones e interacciones.

El procesamiento se refiere al lenguaje creado sobre Java y al IDE mínimo con el que se envía. Es gratuito y de código abierto, se ejecuta en Linux, Mac OS X y Windows, y puede dar salida a pantallas, impresión, paquetes 3D e impresión CNC.

El lenguaje simplifica una gran cantidad de conceptos complejos y facilita la entrada de diseñadores, artistas y no programadores al mundo de la programación.

A lo largo de los años se usó para producir una serie de proyectos que van desde la visualización de datos hasta la computación física , juegos , 3D , sonido , rendimiento en vivo y más .

Debido a su vibrante comunidad, Processing no solo cuenta con una contribución de más de 100 bibliotecas, sino que también está presente en las principales plataformas móviles, como Android y iOS.

Hay comunidades en línea para compartir contenido de procesamiento, como OpenProcessing .

Algunos sitios web incluso permiten que los usuarios aprendan y utilicen el procesamiento directamente en el navegador, como el SketchPatch controlado por Flash y el HasCanvas, Sketchpad y p5.js (puro JS) controlados por JavaScript.

También hay puertos de procesamiento a los siguientes idiomas:

- JavaScript utilizando ProcessingJS o p5js
- ActionScript
- Python (ver NodeBox, Field, pyProcessing o el nuevo modo oficial de Python)
- Scala
- Clojure
- Rubí

El modo Android permite ejecutar bocetos de procesamiento como aplicaciones de Android con pocos o ningún cambio en el código al automatizar tareas desde la configuración del proyecto hasta la exportación de archivos *.apk*. Los bocetos de procesamiento de Android también tienen acceso a los dispositivos y sensores Android subyacentes.

Los usuarios avanzados no están limitados al IDE de procesamiento; Pueden configurar proyectos de Procesamiento en Eclipse; use proclipsing o, alternativamente, use Sublime Text para crear y ejecutar bocetos a través del paquete de procesamiento sublime.

## Versiones

Versión	Fecha de lanzamiento
1.5.1	2011-05-15
2.2.1	2014-05-19
3.1.2	2016-07-29
3.2.1	2016-08-19

## Examples

Instalación y configuración

La forma más fácil de usar el procesamiento es descargar el editor de procesamiento desde la página de descarga de procesamiento .

Eso viene como un archivo zip. Descomprima ese archivo en cualquier lugar, y tendrá un directorio que contiene un ejecutable (en Windows, ese es el archivo processing.exe).

Ejecutar ese ejecutable abre el editor de Procesamiento:

B sketch_160803a   Processing 3.1.1						_		×	<
<u>File E</u> dit <u>S</u> ketch <u>D</u> ebug <u>T</u> ools <u>H</u> elp									
sketch_160803a						<b>8</b> B	Jav	a ▼	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19									× ×
<	_	 _	_	 	_	_	_	>	4
Console 🛕 Error									

El editor de Procesamiento (también llamado el Entorno de Desarrollo de Procesamiento o PDE) contiene muchas herramientas que hacen mucho trabajo por usted. Le permite escribir código de procesamiento, que se convierte automáticamente a Java y luego compila y ejecuta para usted.

El PDE contiene muchas funciones, pero por ahora solo escriba su código de procesamiento dentro de la sección blanca del editor y luego presione el botón de reproducción para ejecutar su código. Vea la sección de Hello World a continuación para ver un código de ejemplo.

También puede escribir código de procesamiento utilizando otros editores de códigos básicos como Atom o Sublime Text, o con un IDE más avanzado como eclipse.

#### Hola Mundo

La forma más fácil de escribir código de procesamiento es simplemente llamar a una serie de funciones. Presione el botón Ejecutar en el editor de Procesamiento, y Procesando ejecutará su código. Aquí hay un ejemplo:

```
size(200, 200);
```

background(0, 0, 255); fill(0, 255, 0); ellipse(100, 100, 100, 100);

Este código crea una ventana de 200x200, dibuja un fondo azul, cambia el color de relleno a verde y luego dibuja un círculo en el centro de la pantalla.



Sin embargo, la mayoría de los bocetos de procesamiento utilizarán las funciones de setup() y draw() predefinidas.

- La función setup() es llamada automáticamente por Processing, una vez al principio del boceto. Esta función se utiliza para realizar la configuración inicial, como el size y la carga de recursos, como los archivos de imagen y sonido.
- La función draw() se llama automáticamente procesando 60 veces por segundo. Esta función se utiliza para dibujar y obtener la entrada del usuario.

```
void setup() {
   size(200, 200);
}
void draw(){
   background(0);
   ellipse(mouseX, mouseY, 25, 25);
}
```

Este código crea una ventana de 200x200 y luego dibuja un círculo en la posición actual del mouse.



Lea Comenzando con el procesamiento en línea: https://riptutorial.com/es/processing/topic/3538/comenzando-con-el-procesamiento

# Capítulo 2: Colores en Procesamiento

## Introducción

Este artículo le mostrará los distintos formatos de colores en Procesamiento y las formas en que se pueden utilizar.

## Sintaxis

- color (r, g, b);
- color (r, g, b, alfa);
- color gris);
- color (gris, alfa);
- color (h, s, l); // El modo debe ser HSB. Puedes cambiar esto usando colorMode.

## Parámetros

Parámetros	Detalles
r	Es el rojo del color cuando el modo es RGB .
sol	Es el verde del color cuando el modo es RGB .
segundo	Es el azul del color cuando el modo es RGB .
alfa	Es la opacidad del color.
h	El tono del color cuando el modo es HSB.
S	La saturación del color cuando el modo es HSB .
I	El brillo / luminosidad del color cuando el modo es ${}_{\rm HSB}$ .
gris	El valor entre negro (siendo 0) y blanco (siendo 255).

## Observaciones

Aunque no se ha mencionado en la documentación oficial de Procesamiento, hay un modo CMYK que puede utilizar.

## Examples

Notación de color

Hay varias formas de usar colores en el procesamiento, ya que el procesamiento es muy flexible con los formatos de color.

#### **RGB y RGBA**

Esta es la notación estándar RGB (A) y el modo de color predeterminado. Los primeros tres valores de color (rojo, verde, azul) van de o a 255. Por ejemplo, el siguiente ejemplo es el color rojo, ya que el rojo tiene un máximo de 255 mientras que los demás colores están en o. El blanco está en (255, 255, 255) y el negro está en (0, 0, 0). El cuarto parámetro opcional indica el valor alfa, es decir, la transparencia. Como en otros componentes, el rango de valores es nuevamente [0-255]; o siendo completamente transparente y 255 completamente sólido.

color(255, 0, 0) // This is red color(0, 255, 0, 255) // This is opaque green, and is the same as color(0, 255, 0) color(255, 255, 0, 10) // This is almost transparent yellow

#### HSB

La notación HSB es similar a la notación RGB, excepto por el hecho de que el rojo, el verde y el azul se reemplazan con matiz, saturación y brillo, respectivamente. Puedes cambiar a HSB usando colorMode(HSB).

color(0, 0, 255) //This is white

Al igual que con RGB, HSB también tiene el valor alfa como cuarto parámetro.

#### Valores grises

Si se especifica un parámetro para una función de color, se interpretará como la cantidad entre negro y blanco. El blanco se representa como 255 y el negro como 0. Es el mismo que el color (param1, param1, param1) en el modo RGB. Si se especifican dos parámetros, entonces el primer parámetro se interpretará como arriba y el segundo será el valor alfa.

Lea Colores en Procesamiento en línea: https://riptutorial.com/es/processing/topic/8283/coloresen-procesamiento

# Capítulo 3: Dibujo de formas básicas

## Introducción

En Procesamiento, dibujar formas es clave para el programa. De lo contrario, nada aparecería en la pantalla. Esta sección te dirá cómo se dibujan las formas básicas.

## Sintaxis

- línea (float x1, float y1, float x2, float y2)
- línea (flotador x1, flotador y1, flotador z1, flotador x2, flotador y2, flotador z2)
- elipse (flotar x, flotar y, flotar w, flotar h)
- rect (flotar x, flotar y, flotar w, flotar h)
- triángulo (flotador x1, flotador y1, flotador x2, flotador y2, flotador x3, flotador y3)

## Parámetros

Parámetro	Detalles
x1	coordenada x del primer punto
y1	coordenada y del primer punto
z1	coordenada z del primer punto
x2	coordenada x del segundo punto
y2	coordenada y del segundo punto
z2	coordenada z del segundo punto
x3	coordenada x del tercer punto
уЗ	coordenada y del tercer punto
Х	coordenada x
у	coordenada y
w	anchura
h	altura

## Observaciones

Usted puede encontrar una referencia en la fundación de Processing.

Página de inicio de procesamiento

## Examples

Dibujando una linea

El procesamiento proporciona un método llamado line() para dibujar una línea en la pantalla. Este código dibuja una línea blanca de 10 píxeles sobre fondo negro.

```
void setup() {
    size(500, 500);
    background(0);
    stroke(255);
    strokeWeight(10);
}
void draw() {
    line(0, 0, 500, 500);
}
```

La firma de la line() de método line() es esta.

line(x1, y1, x2, y2);

x1 y y1 es una coordenada del punto de inicio. x2 y y2 es una coordenada del punto final.

El método del stroke() se usa para especificar el color de la línea que dibujará.

El método strokeWeight() se usa para especificar el grosor de la línea que dibujará. (en píxeles)

Dibujando un rectángulo

El procesamiento proporciona el método rect() para dibujar un rectángulo. Este código dibuja un rectángulo blanco de 50 x 50 sobre fondo negro.

```
void setup() {
    size(500, 500);
    background(0);
    fill(255);
    noStroke();
}
void draw() {
    rect(225, 225, 50, 50);
}
```

La firma del método rect () es esta.

rect(x, y, w, h);

 ${\rm x}$  e  ${\rm y}$  es la coordenada del rectángulo.  ${\rm w}$  y  ${\rm h}$  es el ancho y la altura del rectángulo.

El método fill() se usa para especificar el color de relleno del rectángulo y otras formas, como elipse, triángulo, polígono.

El método nostroke() se usa para especificar que no hay trazos alrededor del rectángulo. Este método también afecta a otras formas como elipse, triángulo, polígono.

Dibujando un elipse

El procesamiento proporciona ellipse método para dibujar elipse. Este código dibuja un círculo blanco que tiene un radio de 25 píxeles.

```
void setup() {
    size(500, 500);
    background(0);
    fill(255);
    noStroke();
}
void draw() {
    ellipse(225, 225, 50, 50);
}
```

La firma del método ellipse() es esta.

```
ellipse(x, y, w, h);
```

 $_{\rm x}$  e  $_{\rm y}$  es la coordenada de la elipse.  $_{\rm w}$  y  $_{\rm h}$  es el ancho y la altura de la elipse.

#### Dibujando un triangulo

El procesamiento proporciona el triangle del método para dibujar un triángulo. El siguiente código dibuja un triángulo casi equilátero de 25 píxeles entre cada punto de definición.

```
void setup() {
    size(500, 500);
    background(0);
}
void draw() {
    triangle(0, 0, 25, 0, 12, 12);
}
```

La firma del triangle es así:

triangle(x1, y1, x2, y2, x3, y3);

Cada punto x corresponde al eje x del punto, y y al eje y. Los tres puntos se unirán para formar un triángulo.

#### Dibujando un triangulo

El procesamiento proporciona el método triangle() para dibujar un triángulo. Este código dibuja un triángulo blanco sobre fondo negro.

```
void setup() {
    size(500, 500);
    background(0);
    fill(255);
    noStroke();
}
void draw() {
    triangle(250, 225, 225, 275, 275, 275);
}
```

Lea Dibujo de formas básicas en línea: https://riptutorial.com/es/processing/topic/7277/dibujo-deformas-basicas

# Capítulo 4: Empezando con Processing para Android

## Observaciones

Puede encontrar más información sobre Processing para Android en http://android.processing.org/

## Examples

#### Instalación

Antes de comenzar con Processing para Android, asegúrese de tener instalado Processing.

- 1. Procesamiento abierto
- 2. Haga clic en Agregar modo ... ubicado en la parte superior derecha de la ventana:



3. Busque "android" en el Administrador de contribución y elija la opción que tiene el autor "The Processing Foundation" y haga clic en Instalar

Libraries       Modes       Tools       Examples         android		Contribution Manager		
android         Status       Name         Android Mode   Create projects with Processing for Andro       P The Processing Foundation         P       Android Mode 3.0.1         The Processing Foundation       3.0.1         Create projects with Processing for Android devices       4 Install         X       Removie	Librarie	es <mark>Modes</mark> Tools Examples		
Status       Name       Author         Image: Android Mode   Create projects with Processing for Andro       Image: The Processing Foundation         Image: Android Mode 3.0.1       Image: Android Mode 3.0.1         The Processing Foundation       3.0.1         Create projects with Processing for Android devices       Image: Android Mode 3.0.1         Image: Create projects with Processing for Android devices       Image: Android Mode 3.0.1         Image: Create projects with Processing for Android devices       Image: Android Mode 3.0.1         Image: Create projects with Processing for Android devices       Image: Android Mode 3.0.1         Image: Create Projects with Processing for Android devices       Image: Android Mode 3.0.1		android		
Android Mode 3.0.1       Install         The Processing Foundation       3.0.1         Create projects with Processing for Android devices       Update         Image: Note that the processing for Android devices       Image: Note that the processing for Android devices	Status	Name Android Mode   Create projects with Processing for Andro	Author P The Processi	ng Founda
Android Mode 3.0.1       Install         The Processing Foundation       3.0.1         Create projects with Processing for Android devices       Cupdat         K Remove       K Remove				
Create projects with Processing for Android devices	P	Android Mode 3.0.1 The Processing Foundation		↓ Install
× Remov		Create projects with Processing for Android devices		3.0.1 i
× Remov				© Update
				× Remov

- 4. Una vez que la instalación haya finalizado, cambie el modo a Android
- 5. Esta ventana de abajo debería aparecer. Si no tiene el SDK de Android, haga clic en "Descargar SDK automáticamente" para instalarlo. Si ya ha instalado el SDK de Android, haga clic en "Localizar la ruta del SDK manualmente".

	Is the Android SDK installed?
	The Android SDK does not appear to be installed,
	because the ANDROID_SDK variable is not set.
	In it is installed, click "Locate SDK path" to select the
	Processing download the SDK automatically.
	If you want to download the SDK manually, you can get
	the command line tools from here. Make sure to install
	the SDK platform for API 15 (Android 4.0.3) or higher.
Loca	te SDK path manually Download SDK automatically
_	

5,5. Para mí, localicé mi ruta del SDK en /Users/my-username/Library/Android/sdk 6. Has terminado

Lea Empezando con Processing para Android en línea: https://riptutorial.com/es/processing/topic/7586/empezando-con-processing-para-android

# Capítulo 5: Formas y funciones básicas utilizando P3D.

### Sintaxis

- traducir (flotar x, flotar y, flotar z)
- rotateX (ángulo de flotación)
- rotarY (ángulo de flotación)
- rotateZ (ángulo de flotación)
- caja (tamaño del flotador)
- caja (flotador w, flotador h, flotador d)

## Parámetros

Parámetros	Detalles
ángulo	el angulo esta en radianes
tamaño	La dimensión de la caja que se utilizará para todas sus dimensiones.
w	La dimensión de la caja en el x-axis
h	La dimensión de la caja en el y-axis
re	La dimensión de la caja en el z-axis

## Examples

Traducción 3D

Aquí es cómo traducir objetos en P3D:

```
size(200, 200, P3D); //Starting P3D renderer
fill(255, 0, 0, 150); //transparent red
rect(10, 10, 100, 100); //first rectangle
fill(0, 0, 255, 150); //transparent blue
translate(50, 50, 50); //translate x, y and z by 50 pixels
rect(0, 0, 100, 100); //second rectangle (same dimensions as the first one)
```

4		

Rojo: primer rectángulo Azul: segundo rectángulo

Como puede verse en el boceto anterior, el segundo rectángulo solo parece ser más grande que el primero, cuando en realidad está "más cerca" de la pantalla como resultado de traducir el rectángulo 50 píxeles a lo largo del z-axis (y de Por supuesto, el rectángulo se ha traducido a lo largo de los ejes  $x e_y$ ).

#### **Rotación 3D**

Hay tres funciones para la rotación 3D: rotateX(angle), rotateY(angle) y rotateZ(angle) para rotar en sus respectivos ejes donde el angle está en radianes.

size(200, 200, P3D); //Starting P3D renderer fill(255, 0, 0, 150); //transparent red translate(width/2, height/2);//translate to centre, ie (100, 100) rectMode(CENTER);//This makes the rectangle centre in (100, 100) rect(0, 0, 100, 100); //first rectangle fill(0, 0, 255, 150); //transparent blue rotateX(PI/4); //rotate in the x-axis by PI/4 radians (45 degrees) rect(0, 0, 100, 100); //second rectangle (same dimensions as the first one)



rotateY(radians(45)); //rotate in the y-axis by passing the radians conversion of 45 degrees



rotateZ(3\*PI/4); //rotate in the z-axis by 3\*PI/4 radians (270 degrees)



Nota: las transformaciones (como las traducciones y rotaciones) se agregan a la transformación anterior.

#### Dibujando un cuboide

Para dibujar un cuboide, tienes que usar la función box () dando sus dimensiones como sus parámetros.

```
size(200, 200, P3D); //Starting the P3D renderer
translate(width/2, height/2); //Translating to the centre of the sketch
rotateY(PI/4); //rotate so that...
rotateX(PI/6); //... it will be easy to see the box
noFill(); //disabling the box's fill, so that we will be able to see its edges
box(100, 50, 75); //the box function requires its dimensions as its parameters
```

$\bigwedge$

Tenga en cuenta que la función box() no acepta su posición como parámetros

También hay una forma de llamar a la función box() con un solo parámetro. En este caso, será un cubo.

stroke(0, 100, 255); //change the edges' colour fill(0, 0, 255); //fill the `box` in a blue colour box(100); //draw a cube



Lea Formas y funciones básicas utilizando P3D. en línea: https://riptutorial.com/es/processing/topic/7591/formas-y-funciones-basicas-utilizando-p3d-

# Capítulo 6: Usando el procesador P3D de Processing

## Observaciones

Una referencia para usar P3D se encuentra en https://processing.org/tutorials/p3d/ en un tutorial de Daniel Shiffman.

## Examples

#### Empezando

Para poder usar el procesador P3D, debe especificarlo en la función size()

```
size(200, 200, P3D);
```

Ahora, hay tres ejes:  ${\tt x}$  ,  ${\tt y}$  y  ${\tt z}$  .



Ahora puedes proceder a dibujar en 3D.

Lea Usando el procesador P3D de Processing en línea: https://riptutorial.com/es/processing/topic/7589/usando-el-procesador-p3d-de-processing

## Capítulo 7: Uso de procesamiento con editores de códigos alternativos como Sublime y Atom

## Examples

Usando el procesamiento con texto sublime

Para agregar un idioma a Sublime Text, usa Package Control . Puede hacerlo pegando el código de Python correcto (disponible en el sitio de Control de Paquetes vinculado anteriormente), o descargando el archivo del paquete .sublime (también disponible para descargar en el sitio). Una vez que configure el control de paquetes, reinicie el texto sublime.

Para instalar el paquete de procesamiento desde el control de paquetes, siga los siguientes pasos:

- Ejecutar herramientas | Instale processing-java para instalar la herramienta processing-java. Esta herramienta es necesaria para crear un boceto de procesamiento en la línea de comandos y editores como Sublime Text y Atom.
- Abra la paleta de comandos (Ctrl + Shift + P o Cmd + Shift + P)
- Buscar 'Control de paquetes: instalar paquete'
- Busque 'Procesando' e instale el paquete
- Reiniciar texto sublime

Después de seguir estos pasos, debería poder seleccionar Procesando como idioma. Esto facilitará la codificación en el procesamiento en texto sublime.

#### Usando Processing con el editor Atom

Hay varios paquetes que pueden ejecutar bocetos de procesamiento en el editor Atom. Estas instrucciones utilizan el paquete Script . También hay paquetes disponibles para el resaltado de sintaxis y autocompletar , que se requieren para que Script identifique los tipos de archivos de Procesamiento.

Instale el complemento Script. apm install script desde la línea de comando o busque el paquete 'script' de *rgbkrk* en la pestaña Install en la configuración de Atom ( command + , acceso directo command + , 0 ctrl + , ).

Una vez que se instale Script, deberá instalar processing-java. Esta herramienta viene con el software de procesamiento principal y es necesaria para crear bocetos de procesamiento en la línea de comandos y en los editores:

• MacOS: Ejecutar Tools > Install "processing-java".

• Linux: agregue el directorio de Procesamiento a su PATH entorno PATH (reemplace /path/to/processing con la ruta donde está instalado el Proceso):

```
sudo ln -s /path/to/processing/processing-java /usr/local/bin/
```

- Windows: agregue el directorio de procesamiento a su PATH entorno PATH :
  - Abra la Configuración avanzada del sistema ejecutando sysdm.cpl o buscando en el Panel de control.
  - Haga clic en el botón Variable de entorno en la pestaña Avanzado.
  - Edite la variable PATH para incluir el directorio de procesamiento en las variables de usuario (solo para su cuenta) o en las variables del sistema (para todos los usuarios).

Ahora, puede ejecutar Procesando bocetos ejecutando Packages > Script > Run Script . El acceso directo predeterminado es command + shift + b 0 ctrl + shift + b , pero para disminuir aún más el dolor de la transición, puede enlazar el acceso directo Procesando IDE para ejecutar el boceto. Para hacer eso:

- Abra el archivo de mapa de teclas Atom ejecutando File > Keymap
- Pegue las siguientes líneas al final del archivo (siéntase libre de cambiar el enlace a lo que desee).

```
'atom-text-editor':
    'ctrl-r': 'script:run'
```

#### **Usando Procesando con Eclipse**

Para usar Processing en Eclipse, comience creando un nuevo proyecto Java. Luego, seleccione File > Import y luego elija General > File System para ubicar el archivo core.jar. Se puede encontrar en PATH\_TO\_PROCESSING/core/library/ para Windows o /Applications/Processing 3.app/Contents/Java/core/library/ para Mac. Una vez que se haya completado, haga clic derecho en core.jar y agréguelo a la ruta de compilación.

La placa de calderas para usar Processing en Eclipse es la siguiente:

```
import processing.core.PApplet;
public class UsingProcessing extends PApplet {
    // The argument passed to main must match the class name
    public static void main(String[] args) {
        PApplet.main("UsingProcessing");
    }
    // method used only for setting the size of the window
    public void settings() {
    }
    // identical use to setup in Processing IDE except for size()
```

```
public void setup(){
    }
    // identical use to draw in Prcessing IDE
    public void draw(){
    }
}
```

El método de settings() se utiliza para establecer el tamaño de la ventana. Por ejemplo, para crear una ventana de 400x400, escriba lo siguiente:

```
public void settings(){
    size(400,400);
}
```

Todo lo demás como se describe en la documentación de Hello World en términos del uso de setup() y draw() aplica aquí.

Como ejemplo final, aquí está el código del ejemplo de Dibujar una línea donde se escribiría en Eclipse:

```
import processing.core.PApplet;
public class UsingProcessing extends PApplet {
    // The argument passed to main must match the class name
    public static void main(String[] args) {
       PApplet.main("UsingProcessing");
    }
    // method for setting the size of the window
    public void settings() {
       size(500, 500);
    }
    // identical use to setup in Processing IDE except for size()
    public void setup() {
       background(0);
       stroke(255);
       strokeWeight(10);
    }
    // identical use to draw in Prcessing IDE
    public void draw() {
       line(0, 0, 500, 500);
    }
}
```

Lea Uso de procesamiento con editores de códigos alternativos como Sublime y Atom en línea: https://riptutorial.com/es/processing/topic/5594/uso-de-procesamiento-con-editores-de-codigosalternativos-como-sublime-y-atom

## Creditos

S. No	Capítulos	Contributors
1	Comenzando con el procesamiento	aspiring_sarge, Community, Cows quack, Kevin Workman, MasterBob, Michael Andersen, user2314737
2	Colores en Procesamiento	Cows quack, MasterBob, Mert Toka
3	Dibujo de formas básicas	Cows quack, MasterBob, sohnryang
4	Empezando con Processing para Android	Cows quack
5	Formas y funciones básicas utilizando P3D.	Cows quack
6	Usando el procesador P3D de Processing	Cows quack
7	Uso de procesamiento con editores de códigos alternativos como Sublime y Atom	bakahoe, CodeMacabre, Kevin Workman, Mert Toka, Peter, sohnryang