



PRONTUARIO DE PROCESSING

NOTAS: Este documento lo puedes descargar, copiar, imprimir y distribuir de forma libre.

1.-"Arduino Castilla Education Project" es iniciativa del Equipo Arduino, con David Cuartielles a la cabeza, el Centro Regional de Formación del Profesorado de Castilla-La Mancha y Fundación Telefónica.

2.-Este documento ha sido elaborado por profesores del Departamento de Tecnología del I.E.S. "Sta. Mª de Alarcos" de Ciudad Real, España, especialmente dirigido a profesores y alumnos de Enseñanza Secundaria.

3.-Puedes acceder a la página del proyecto pulsando sobre el logotipo de la cabecera y a la referencia completa, desde el la aplicación Help -> Reference.

Plantilla de Processing

// Estos símbolos permiten escribir comentarios de una línea

// Declaración de variables globales (para todo el programa)

```
void setup() {
```

```
    // Este bloque sólo se ejecuta al cargar el programa
```

```
}
```

```
void draw() {
```

```
    // Este bloque se ejecuta repetidamente
```

```
    // Declaración de variables locales (de la función)
```

```
}
```

Tabla de funciones

Función	Descripción	Ejemplo
GRAFICOS	Coordenadas cartesianas: (X,Y) <ul style="list-style-type: none"> • X: horizontal • Y: vertical • (0,0) -> esquina Sup. Izqda. 	// Coordenada absoluta (10,10) // indica la posición de ratón (mouseX,mouseY)
size(tamañoX,tamañoY);	Tamaño de la pantalla: <ul style="list-style-type: none"> • tamañoX: horizontal • tamañoY: vertical 	// Pantalla de 300x300 size(300,300);
stroke(R,G,B);	Color para dibujar en RGB: <ul style="list-style-type: none"> • R: rojo (Red) • G: verde (Green) • B: azul (Blue) Los números de 0-255, siendo: <ul style="list-style-type: none"> • 0 -> Nada de intensidad • 255 -> Intensidad máxima 	// Dibujo en rojo stroke(255,0,0); // Dibujo en amarillo stroke(255,255,0);
line(Xi,Yi,Xf,Yi);	Dibujo de línea de (Xi,Yi) a (Xf,Yf): <ul style="list-style-type: none"> • Xi: valor horizontal inicial • Yi: valor vertical inicial • Xf: valor horizontal final • Yf: valor vertical final 	// horizontal (10,10)-> (30,10) line(10,10,30,10);



TecnoAlarcos Arduino

Función	Descripción	Ejemplo
fill(R,G,B);	Relleno de color RGB: <ul style="list-style-type: none"> ● R: rojo (Red) ● G: verde (Green) ● B: azul (Blue) Los números de 0-255, siendo: <ul style="list-style-type: none"> ● 0 -> Nada de intensidad ● 255 -> Intensidad máxima 	<pre>// Relleno verde fill(0,255,0); // Relleno morado fill(255,0,255);</pre>
ellipse(centroX,centroY,tamañoX,tamañoY);	Dibuja círculos y elipses : <ul style="list-style-type: none"> ● centroX: horizontal del centro ● centroY: vertical del centro ● tamañoX: semieje horizontal ● tamañoY: semieje vertical si tamañoX=tamañoY -> círculo	<pre>// Dibujo círculo radio=10 en // (20,20) ellipse(20,20,10,10);</pre>
arc(cX,cY,tX,tY,angi,angf);	Dibuja arcos circulares, elípticos y círculos: <ul style="list-style-type: none"> ● cX->X del centro ● cY->Y del centro ● tX ->anchura ● tY ->altura ● angi->ángulo inicial (en radianes) ● angf->ángulo final (en radianes) 	<pre>//Dibujo círculo como anterior arc(20,20,10,10,0,2*PI);</pre>
rect(Xi,Yi,Xf,Yf); rect(Xi,Yi,Xf,Yf,radio); rect(Xi,Yi,Xf,Yf,Rai,Rad,Rbi,Rbd);	Dibuja rectángulos : <ul style="list-style-type: none"> ● Xi->horizontal inicial ● Yi->vertical inicial ● Xf->horizontal final ● Yf->vertical final ● radio-> radio de los bordes ● Rai->radio esquina arriba izqda ● Rai->radio esquina arriba dcha ● Rai->radio esquina bajo izqda ● Rai->radio esquina bajo dcha 	<pre>// cuadrado de 10 de lado en // (20,20) rect(20,20,30,30); //Lo mismo de bordes //redondeados radio 3 rect(20,20,30,30,3);</pre>
triangle(x1,y1,x2,y2,x3,y3);	Dibuja triángulos : <ul style="list-style-type: none"> ● Coordenadas x,y de cada vértice 1, 2 y 3 	<pre>// triángulo tipo escuadra de lado // 20 en (30,10) triangle(30,10,30,30,50,30);</pre>
<pre>// Hay que utilizar las 3 PTimage nvar; nvar=loadImage("im.jpg"); image(X,Y,nvar);</pre>	Carga de imágenes : Para que se pueda cargar una imagen, esta debe estar en la carpeta "Data" del programa. <ul style="list-style-type: none"> ● nvar -> nombre de la variable ● im.jpg -> nombre del fichero ● X,Y -> coordenada sup izda de la foto en la pantalla 	<pre>// Carga una imagen PImage imagen; void setup(){ imagen = loadImage("logoard.jpg"); } void draw(){ image(imagen,0,0); }</pre>
text(nvar,X,Y);	Escribe texto : <ul style="list-style-type: none"> ● nvar: nombre de la variable con el texto a visualizar. Hay que declararla como String ● X,Y: coordenadas iniciales de la cadena de caracteres Si quiero convertir un número a caracteres debo emplear: <ul style="list-style-type: none"> ● String.valueOf(numero) 	<pre>// Escribo "Hola" en (10,10) String palabra = "Hola"; text(palabra, 10,10); //Escribo el día del mes int día = day(); String tdia = String.valueOf(día); text(tdia,10,10);</pre>
background(gris) background(gris,transpar) background(R,G,B,transpar)	Para animar se pinta un fondo de pantalla : <ul style="list-style-type: none"> ● Gris -> 0 (negro)-255 (blanco) ● transpar: <ul style="list-style-type: none"> ○ 0 -> transparente ○ 255 -> opaco ● R: rojo (Red) 	<pre>// Fondo blanco background(255); // Fondo gris con trazo de ratón background(200,0.0001);</pre>



TecnoAarcos Arduino

	<ul style="list-style-type: none"> ● G: verde (Green) ● B: azul (Blue) <p>Los números de 0-255, siendo:</p> <ul style="list-style-type: none"> ● 0 -> Nada de intensidad ● 255 -> Intensidad máxima 	
Función	Descripción	Ejemplo
GESTIÓN DEL TIEMPO	Con estas funciones podemos saber el año, mes, día, hora, minuto, segundo... Así como duración, retrasos...	
<pre>int nvar=year(); int nvar=month(); int nvar=day(); int nvar=hour(); int nvar=minute(); int nvar=second(); int nvar=millis();</pre>	<p>Carga en la variable nvar el año</p> <p>Carga en la variable nvar el mes</p> <p>Carga en la variable nvar el día</p> <p>Carga en la variable nvar la hora</p> <p>Carga en la variable nvar el minuto</p> <p>Carga en la variable nvar el segundo</p> <p>Carga en la variable nvar el milisegundo</p>	*ver text()
DECISIONES Y CONDICIONES	En determinadas condiciones hay que hacer o dejar de hacer el ciclo normal de repeticiones	
<pre>if(condicion) { // si se cumple } else { // si no se cumple }</pre>	<p>No es imprescindible “else”</p> <ul style="list-style-type: none"> ● Si se cumple la condición, entonces se hace lo que hay entre las llaves de después de if ● si no se cumple, lo que se ponga después de else <p>Los operadores para la if:</p> <ul style="list-style-type: none"> ● == igual, != distinto ● > mayor, < menor ● >= mayor o igual, <= menor o igual ● && y, o 	<pre>// Si i=3 => escribe “tonto” if(i==3) { text(“tonto”, 10,10); } // Si j>4 => escribe “apto” // si no => escribe “suspenso” if (j>4) { text(“apto”,10,10); } else { text(“suspenso”,10,10); }</pre>
<pre>for(cond_ini; cond_d; modif) { // repeticiones si cumple }</pre>	<p>Desde la condición inicial “cond_ini”, hasta condición “cond_d”, se realiza lo que hay entre las llaves modificando la variable según “modif”.</p> <p>Operadores válidos: =,>,<,>=,<=</p>	<pre>// Escribe durante 10 segundos for (i=1; i<=10;i=i+1) { delay(1000); //espero 1s background(255); text(String.valueOf(i),10,10); }</pre>
<pre>while(condición) { // si se cumple }</pre>	<p>Mientras se cumpla la condición, se repite lo que haya entre las llaves</p>	<pre>// Escribe durante 10 segundos int i=0; for (i<11) { delay(1000); background(255); text(String.valueOf(i),10,10); i=i+1; }</pre>
<pre>switch(variable) { case valor1: // Hacer si cierto break; case valor2: // Hacer si cierto break; ... case valorn: // Hacer si cierto break; default: // Hacer si no cierto ninguna</pre>	<p>Para la “variable” se hacen las sentencias según el valor, si no coincide ninguna, se realiza lo que hay en default</p>	<pre>switch(letra) { case 'A': text(“es A”,10,10); break; case 'B': text(“es B”,10,10); break; case 'C': text(“es C”,10,10); break; default: text(“?”,10,10); break;</pre>



TecnoAlarcos Arduino

break; }		}
ENTRADAS	Teclado y ratón	
keyPressed() keyPressed key keyCode	Detección de teclas presionadas: <ul style="list-style-type: none"> • keyPressed(): función tipo void • keyPressed: detección de tecla • key: indica la tecla • keyCode: si son flechas, Ctr... 	// Si pulsas tecla rellena de blanco if (keyPressed == true) { fill(0); } else { fill(255); } rect(25, 25, 50, 50);
mousePressed() mouseMoved() mouseDragged() mouseButton	Se pueden usar como funciones tipo void <ul style="list-style-type: none"> • mousePressed->activa cuando se pulse un botón • mouseMoved->detecta el movimiento del ratón • mouseDragged->detecta el arrastre con el ratón • mouseButton->indica qué botón es el pulsado 	// cambia el valor del rectángulo // al pulsar el botón izdo void draw() { if (mousePressed && (mouseButton == LEFT)) { fill(0); } else if (mousePressed && (mouseButton == RIGHT)) { fill(255); } else { fill(126); } rect(25, 25, 50, 50); }

Tabla de colores RGB

RO	JOS	VER	DES	AZU	LES	GRI	SES
COLOR	R,G,B	COLOR	R,G,B	COLOR	R,G,B	COLOR	R,G,B
Red	255,0,0	Verde lima	0,255,0	Azul	0,0,255	Blanco	255,255,255
Rojo indio	205,92,92	Verde-amarill	173,255,47	Azul agua	255,255,0	Blanco nieve	255,250,250
Coral claro	240,128,128	Verde pálido	152,251,152	Light Cyan	224,255,255	Gris claro	230,230,230
Salmon	250,128,114	Verde agua	46,139,87	Aguamarina	127,255,212	Gris Plata	192,192,192
Crimson	220,20,60	Verd-Bosque	34,139,34	Turquesa	64,224,208	Gris	128,128,128
Ladrillo	178,34,34	Aceituna	128,128,0	Azul metal	70,130,180	Gris oscuro	47,79,79
Rojo oscuro	139,0,0	Verde claro	144,238,144	Azul marino	0,0,128	Negro	0,0,0
RO	SAS	AMA	RILLOS	PURP	URAS	NARA	NJAS
COLOR	R,G,B	COLOR	R,G,B	COLOR	R,G,B	COLOR	R,G,B
Rosa	255,192,203	Amarillo	255,255,0	Violeta	238,130,238	Coral	255,127,80
Rosa claro	255,182,193	Oro	255,215,0	Magenta	255,0,255	Naranja	255,165,0
Rosa intenso	255,20,147	Amarillo claro	255,255,224	Púrpura	128,0,128	Naranja osc	255,140,0