

Niñas PRO(grama doras)

COORPORACIÓN C1⁰⁰

TALLER DÍA DE NIÑAS EN TIC

CORPORACION C¹⁰⁰



TALLER DÍA DE NIÑAS EN TIC

INTRODUCCIÓN

El ejercicio siguiente tiene como objetivo aprender los conceptos básicos sobre programación.

Para crear programas necesitamos software especializado que interpreta nuestras instrucciones de manera que las pueda entender el computador. En este caso usaremos: <https://alpha.editor.p5js.org/>

ACTIVIDAD PASO A PASO

1. Coloquemos elementos en la pantalla:

Antes de comenzar, vamos a aprender algunos conceptos importantes.

·Variables: una **variable** es un espacio en la memoria del computador que nos sirve para guardar un valor. Para definir una variable en p5.js comenzamos con la palabra var y luego el nombre que tendrá la variable, por ejemplo:

```
var alto;  
var mensaje;
```

Para guardar un valor en la variable escribimos el nombre de la variable, el signo igual = y el valor que deseamos guardar:

```
alto = 400;  
mensaje = 'Hola';
```

Teniendo en cuenta que si el valor que queremos guardar es texto, va entre comillas simples.

·Funciones: Una función es un conjunto de instrucciones que realiza una tarea específica y puede o no devolver un resultado. La manera de definir una función en p5.js es con la palabra function seguida del nombre que daremos a la función, un par de paréntesis y un par de llaves que determinan el inicio y final de la función:

```
function saludar() {  
}
```

La función puede requerir algún valor de entrada para realizar su tarea, este valor que le daremos se denomina **parámetro** y se coloca su nombre entre los paréntesis:

```
function saludar(mensaje) {  
}
```

Existen varias funciones que la librería p5.js ya tiene incluidas. Las que usaremos en el paso 1 son las siguientes:

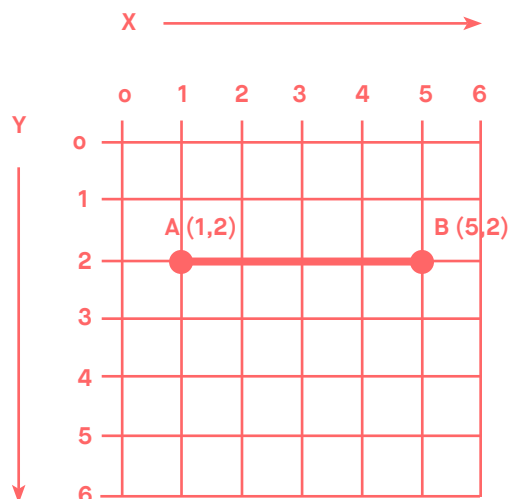
-createCanvas: crea el lienzo o superficie sobre el que vamos a ubicar los demás elementos gráficos que usaremos.

-createElement: crea un campo de texto donde podremos escribir.

-createElement: crea un elemento gráfico, por ejemplo un texto, imagen, video, etc.

-createButton: crea un botón.

-position: define la posición donde se ubicará el elemento gráfico, de acuerdo al sistema de coordenadas X, Y de p5.js:



Comentario: es una anotación que sirve de guía al programador, el computador no lo tomará en cuenta cuando ejecute el programa. Se crea utilizando doble slash al comienzo de la línea, por ejemplo:

```
//este texto es informativo, será ignorado por el computador ;-)
```

<h2>¿Cómo te llamas?</h2> <input type="text"/> <input type="submit" value="submit"/>	<pre>var input, button, mensajeB; function setup() { var x = 20; var y = 65; alto = 400; ancho = 710; // creamos el canvas createCanvas(ancho, alto); mensajeB = createElement('h2', '¿Cómo te llamas?'); mensajeB.position(x, 5); input = createInput(); input.position(x, y); button = createButton('submit'); button.position(x + input.width, y); }</pre>
--	---

3

2. Mostremos un saludo:

En este paso veremos que los botones tienen una función especial llamada `mousePressed` que se ejecuta cada vez que les hacemos clic. Para definir qué debe pasar cuando ocurre un clic, le pasamos a `mousePressed` el nombre de otra función que nosotros creamos, de la siguiente manera:

```
//cuando presionemos el botón se ejecutará la función "saludo"
button.mousePressed(saludo);
```

```
function saludo(){
}
```

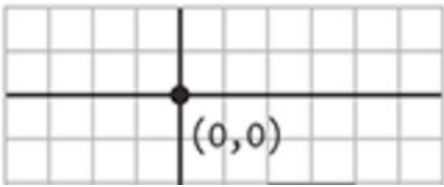
<h2>Hola Niñas Programadoras!</h2> <input type="text"/> <input type="submit" value="submit"/>	<pre>var input, button, mensajeB; function setup() { var x = 20; var y = 65; alto = 400; ancho = 710; // create canvas createCanvas(ancho, alto); mensajeB = createElement('h2', '¿Cómo te llamas?'); mensajeB.position(x, 5); input = createInput(); input.position(x, y); button = createButton('submit'); button.position(x + input.width, y); button.mousePressed(saludo); }</pre>
---	--

```
}  
  
function saludo(){  
  var nombre = input.value();  
  textAlign(CENTER);  
  textSize(50);  
  mensajeB.html('Hola '+ nombre + '!');  
  input.value('');  
}
```




3. Juguemos con texto, colores y posición:

Ahora vamos a dibujar elementos de texto en la pantalla. Para definir la posición y el color disponemos de las siguientes funciones:

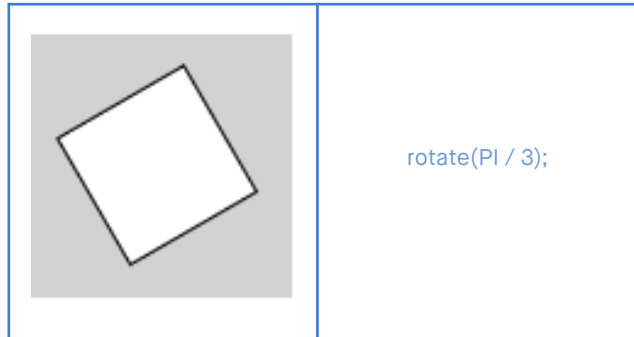
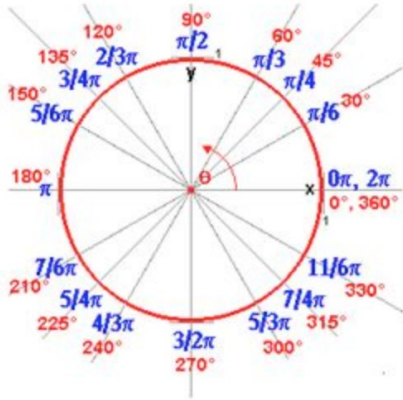
- push: permite empezar a cambiar los parámetros del dibujo (posición, tamaño, color, etc.).
- translate: traslada el punto (0,0) del sistema de coordenadas. Por ejemplo:

	<pre>translate (40,20);</pre>
--	-------------------------------

- TEXT: dibuja texto en la pantalla.
- POP: restablece la configuración (posición, color, tamaño, etc.) que estaba guardada hasta antes de llamar a la función push.
- FILL: determina el color del que se pintará los elementos en la pantalla. El color se escribe en términos de la escala RGB, que usa tres números del 0 al 255 para indicar el grado de intensidad de rojo, verde y azul que tiene el color. Por ejemplo:

	<pre>// intensidad de rojo: 0 // intensidad de verde: 0 // intensidad de azul: 255 fill (0, 0, 255);</pre>
	<pre>// intensidad de rojo: 0 // intensidad de verde: 255 // intensidad de azul: 100 fill (0, 255, 100);</pre>
	<pre>// intensidad de rojo: 255 // intensidad de verde: 250 // intensidad de azul: 0 fill (255, 250, 0);</pre>

-**Rotate**: rota la figura en el ángulo que se le pasa como parámetro. Por defecto, el ángulo se pasa en radianes:



EJEMPLO:

Ahora vamos a continuar con el ejercicio, dibujando texto en la pantalla y especificando cuál será su posición, color y ángulo de rotación mediante las funciones que acabamos de conocer:

<p>Hola Fran!</p> <input type="text" value="Fran"/> <input type="button" value="submit"/>	<pre>function posicionar(name){ push(); translate(300,100) text(name, 0 , 0); pop(); } function saludo(){ var nombre = input.value(); textAlign(CENTER); textSize(50); mensajeB.html('Hola '+nombre+'!'); input.value(''); posicionar(nombre); }</pre>
<p>Hola Fran!</p> <input type="text" value="Fran"/> <input type="button" value="submit"/>	<pre>//Se cambia el color del nombre function posicionar(nombre){ push(); fill(100,255, 255); translate(300, 100); text(nombre, 0, 0); pop(); }</pre>

<p>Hola Fran!</p> <input type="text" value=""/> <input type="button" value="submit"/>	<pre><i>//Se gira el nombre</i> function posicionar(nombre){ push(); fill(100,255, 255); translate(300, 100); rotate(PI/2); text(nombre, 0, 0); pop(); }</pre>
--	---

4. Cambiemos los colores y posición al azar:

Algunas veces necesitamos generar valores al azar, por ejemplo, al crear un juego donde simulamos el lanzamiento de dados. P5.js tiene una función que nos permite generar números aleatoriamente, denominada `random`. A esta función podemos enviarle 0, 1 o 2 parámetros. Si no le pasamos parámetros, nos dará números al azar entre 0 y 1. Si le pasamos 1 parámetro, nos dará números al azar entre 0 y el número que le pasamos y si usamos dos parámetros, nos dará números al azar entre esos dos parámetros.

Veamos algunos ejemplos:

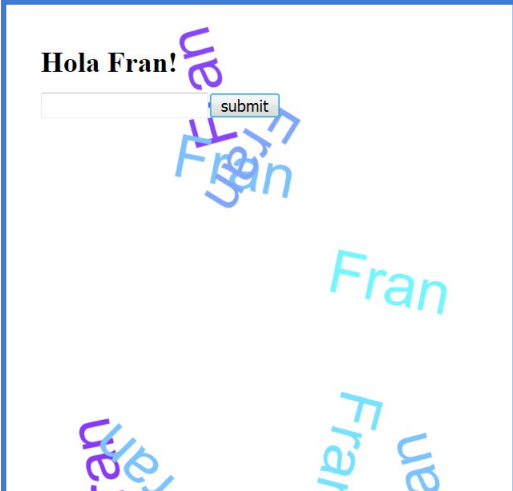
<pre>0.3452345567456 0.9275194576090</pre>	<pre>random();</pre>
<pre>5.8096726569002 2.3433671946518</pre>	<pre>random(6);</pre>
<pre>2009.9576437858 2006.3504792646</pre>	<pre>random(2000,2018);</pre>

Ahora vamos a utilizar la función `random` para que la posición, el ángulo de rotación y el color del texto sean definidos al azar:

<p>Hola Fran!</p> <input type="text" value=""/> <input type="button" value="submit"/>	<pre>function posicionar(nombre){ push(); fill(random(255),random(255), random(255)); translate(random(ancho),random(alto)); rotate(random(2*PI)); text(nombre, 0, 0); pop(); }</pre>
--	--

5. Llenemos de saludos la pantalla

Probemos dibujando el texto varias veces, cada una con su propia posición, rotación y color elegidos al azar:

	<pre>function saludo() { var nombre = input.value(); textAlign(CENTER); textSize(50); mensajeB.html('Hola '+nombre+'!'); input.value(''); posicionar(nombre); posicionar(nombre); posicionar(nombre); posicionar(nombre); posicionar(nombre); posicionar(nombre); posicionar(nombre); posicionar(nombre); }</pre>
---	---

¿Y si quisiera mostrarlo 100 veces, habrá una forma más fácil?

La respuesta es SI :-). Para esto aprenderemos el concepto de ciclos.

Existen muchos ejemplos en la vida diaria en la que repetimos acciones. Por ejemplo, al subir una escalera, repetimos una determinada cantidad de veces el subir un escalón hasta que se cumple la condición de "llegar al siguiente piso".

A esto le llamamos **ciclos** en computación y tienen la siguiente estructura:

Desde (valor inicial) hasta (valor final):
Haré algo

O

Mientras (cumple condición):
Haré algo

En p5.js para crear un ciclo de tipo for especificamos sus tres partes: **valor inicial**, **condición** y **actualización** (incremento o decremento):

```
for (int i = 0; i < 10; i = i + 1) {
}
```


Donde:

El valor inicial es 0.


La condición para ejecutar el ciclo es: mientras el valor sea menor que 10.

La actualización va incrementando el valor en 1 cada vez que se ejecuta el ciclo, de esta manera el ciclo se ejecuta en total 10 veces.

Probemos ahora dibujando el nombre en pantalla con un ciclo que se repetirá ocho veces:

	<pre>function saludo() { var nombre = input.value(); textAlign(CENTER); textSize(50); mensajeB.html('Hola '+nombre+'!'); input.value(''); for(var i = 0; i<8;i=i+1){ posicionar(nombre); } }</pre>
---	---

Luego vamos aumentando la cantidad de veces que se repite el ciclo, por ejemplo hasta 150:

	<pre>function saludo() { var nombre = input.value(); textAlign(CENTER); textSize(50); mensajeB.html('Hola '+nombre+'!'); input.value(''); for(var i = 0; i<150;i=i+1){ posicionar(nombre); } }</pre>
--	---

8

¿Quieres seguir aprendiendo?

Veamos cómo cambiar el color en base a la cantidad de letras que escribimos en el campo de texto. Para realizar esta tarea necesitamos aprender el concepto de condicionales.

A veces tenemos que tomar decisiones dependiendo de una condición determinada. Existen muchos ejemplos en la vida real en que esto ocurre. Por ejemplo, si es lunes vamos a clases, pero si es domingo nos quedamos en casa.

A esto le llamamos condicionales en computación y tienen la siguiente estructura:

<pre>Si (se cumple condición) entonces: Haré algo Si no: Haré otra cosa</pre>

En p5.js escribimos un condicional if de la siguiente forma:


```
if (condición){  
  
}  
else{  
  
}
```

Por ejemplo:

```
if (i = 0){  
  
}  
else{  
  
}
```

Donde la condición que se verifica es si el valor de la variable *i* es igual a cero. En caso de que sea cierto, se ejecutarán las instrucciones que están entre las llaves del if; caso contrario, se ejecutarán las instrucciones que están entre las llaves del else.

En nuestro ejercicio, aplicaremos el condicional if para verificar si el texto que escribimos tiene menos de 10 letras. En caso de que se cumpla esta condición, pintaremos el texto en tonos azules. Si la condición no se cumple, pintaremos el texto en tonos verdes:

	<pre>function posicionar(nombre){ push(); //verificamos la longitud del texto if(nombre.length<10){ fill(random(100),random(100),255); } else{ fill(random(100),255,random(100)); } translate(random(ancho), random(alto)); rotate(random(2*PI)); text(nombre, 0, 0); pop(); }</pre>
---	---

CONCLUSIONES

Los conceptos que aprendiste ahora te dan una idea básica de lo que se puede hacer mediante los programas informáticos y cómo escribirlos.

Si te interesa conocer más, te invitamos a inscribirte en el taller de programación:

<http://ninasprogramadoras.cl/>