



Programación condicional

¿Cuál es nuestro objetivo para este módulo?

En esta clase, aprendimos a crear un sprite de paleta y hacer rebotar la pelota al colisionar.

¿Qué logramos en clase el día de hoy?

Creamos una función personalizada para encontrar el mayor número entre dos números.

Creamos una función personalizada para verificar si el número es positivo o negativo.

Creamos un objeto de paleta del jugador en el juego, e hicimos que la pelota rebotara en los bordes del lienzo y en la paleta.

¿Qué conceptos/bloques de código cubrimos el día de hoy?

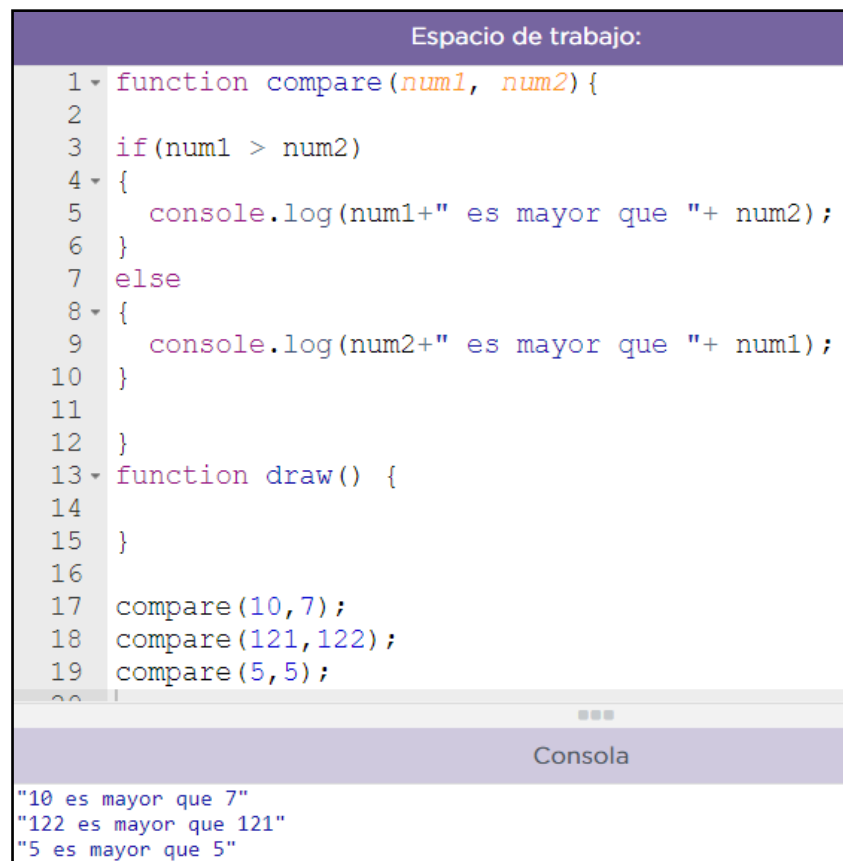
Usar el operador de comparación.

Programación condicional.

Acerca de **World.mouseX** y **World.mouseY**.

¿Cómo HICIMOS las actividades?

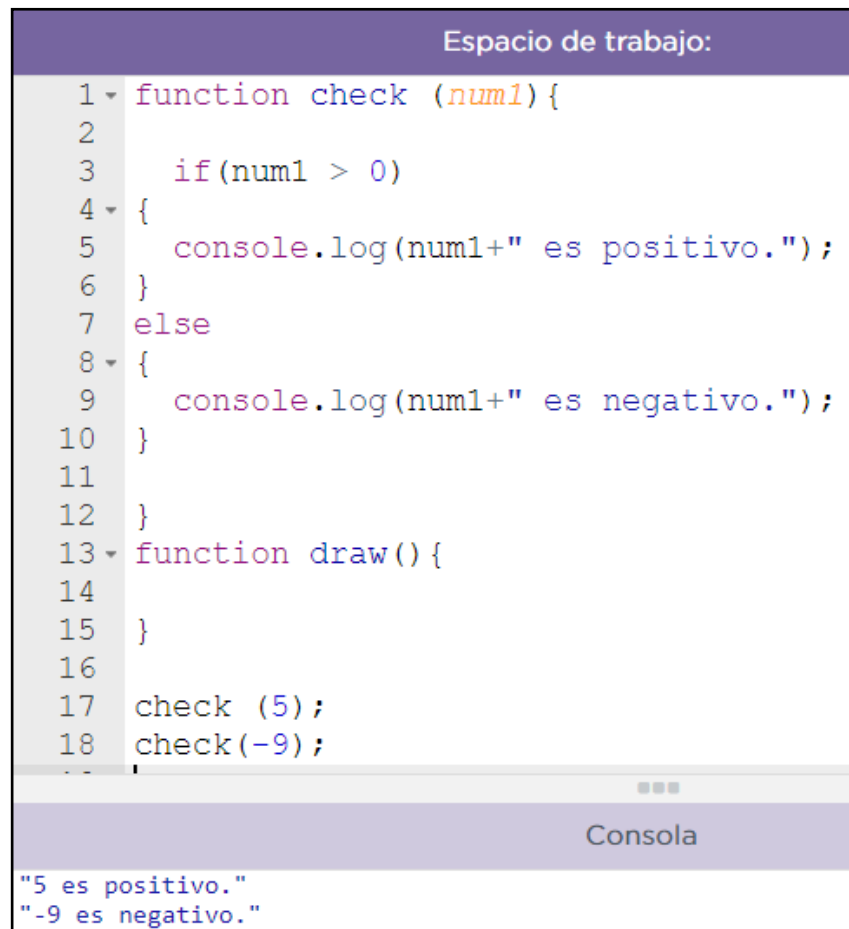
1. Escribir una función **compare()** (comparar) usando la “condición if” (condición si) para encontrar el número más grande entre dos números.



```

Espacio de trabajo:
1 function compare(num1, num2) {
2
3   if(num1 > num2)
4   {
5     console.log(num1+" es mayor que "+ num2);
6   }
7   else
8   {
9     console.log(num2+" es mayor que "+ num1);
10  }
11 }
12
13 function draw() {
14
15 }
16
17 compare(10,7);
18 compare(121,122);
19 compare(5,5);
20
Consola
"10 es mayor que 7"
"122 es mayor que 121"
"5 es mayor que 5"
```

2. Escribir una función para verificar **si** el número es **positivo** o **negativo**.



The image shows a code editor window titled "Espacio de trabajo:". It contains the following JavaScript code:

```
1 function check (num1) {  
2  
3     if (num1 > 0)  
4     {  
5         console.log(num1+" es positivo.");  
6     }  
7     else  
8     {  
9         console.log(num1+" es negativo.");  
10    }  
11  
12 }  
13 function draw() {  
14  
15 }  
16  
17 check (5);  
18 check (-9);
```

Below the code editor is a console window titled "Consola". It displays the output of the code execution:

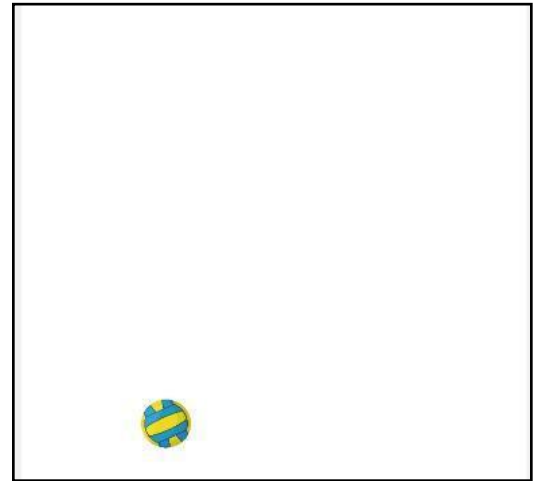
```
"5 es positivo."  
"-9 es negativo."
```

- Usar la función **mousePressed()** (pulsar el mouse) y establecer **velocityX** y **velocityY**, para el sprite de la pelota que hace que la pelota se mueva solo cuando se presiona el mouse.

CÓDIGO:

```

Espacio de trabajo:
1  var ball;
2
3  ball = createSprite(100,100,20,20);
4  ball.setAnimation("volleyball2_1");
5  ball.scale = 0.1;
6
7  createEdgeSprites();
8
9  function draw() {
10     background("white");
11     ball.bounceOff(edges);
12     drawSprites();
13 }
14
15
16 function mousePressed() {
17     ball.velocityX = 4;
18     ball.velocityY = 3;
19 }
```

OUTPUT:

- Asignar **World.mouseX** y **World.mouseY** a **ball.x** y **ball.y** respectivamente dentro de la función **mousePressed()** para usar la posición del mouse para actualizar la posición de la pelota cada vez que se hace clic en el mouse.

Espacio de trabajo:

```
1  var ball;
2
3  ball = createSprite(100,100,20,20);
4  ball.setAnimation("volleyball2_1");
5  ball.scale = 0.1;
6
7  createEdgeSprites();
8
9  function draw() {
10     background("white");
11     ball.bounceOff(edges);
12     drawSprites();
13 }
14
15
16 function mousePressed() {
17     ball.velocityX = 4;
18     ball.velocityY = 3;
19
20     ball.x = World.mouseX;
21     ball.y = World.mouseY;
22 }
```

5. Crear un sprite de paleta y escribir las condiciones para verificar que la paleta no se mueva más allá del lienzo.

CÓDIGO:

OUTPUT:



Espacio de trabajo:

```
1  var ball;
2
3  ball = createSprite(100,100,20,20);
4  ball.setAnimation("volleyball2_1");
5  ball.scale = 0.1;
6
7  ball.velocityX = 0;
8  ball.velocityY = 0;
9
10 var paddle = createSprite(200, 350, 120, 10);
11
12 createEdgeSprites();
13
14 function draw() {
15   background("white");
16
17   paddle.x = World.mouseX;
18
19   if(paddle.x < 60){
20     paddle.x = 60;
21   }
22
23   if(paddle.x > 340){
24     paddle.x = 340;
25   }
26
27   drawSprites();
28   ball.bounceOff(edges);
29 }
```

6. Agregar la función **bounceOff()** (rebotar en) de manera que la pelota rebote al tocar la paleta y los bordes.

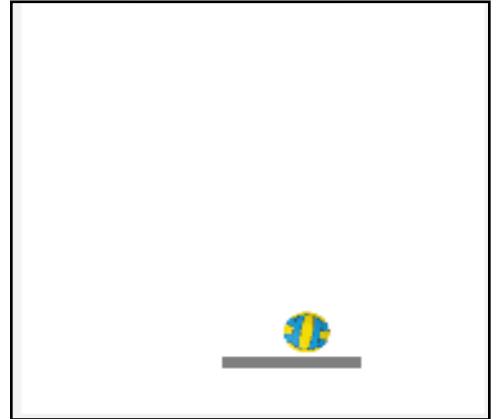
CÓDIGO:

```
background( white );

paddle.x = World.mouseX;

if(paddle.x < 60){
  paddle.x =60;
}

if(paddle.x > 340){
  paddle.x =340;
}
drawSprites();
ball.bounceOff(edges);
ball.bounceOff(paddle);
}
```

OUTPUT:

- Finalmente, agregar la función **bounceOff()** de modo que la pelota solo rebote al tocar los bordes superior, derecho e izquierdo del lienzo.

CÓDIGO:

```
22
23  if(paddle.x > 340){
24    paddle.x = 340;
25  }
26  drawSprites();
27  ball.bounceOff(topEdge);
28  ball.bounceOff(leftEdge);
29  ball.bounceOff(rightEdge);
30  ball.bounceOff(paddle);
31
32 }
33
```

OUTPUT:**¿Qué sigue?**

En la siguiente clase aprenderemos sobre los bucles y comenzaremos a crear un juego de escape.

Amplía tu conocimiento:

Guarda el siguiente enlace: será una referencia para World.mouseY

<https://studio.code.org/docs/gamelab/World.mouseY/>