
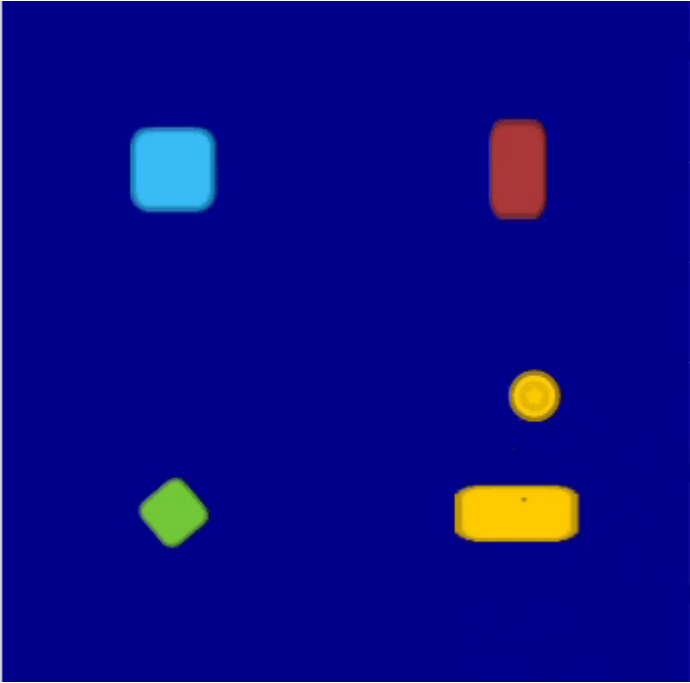
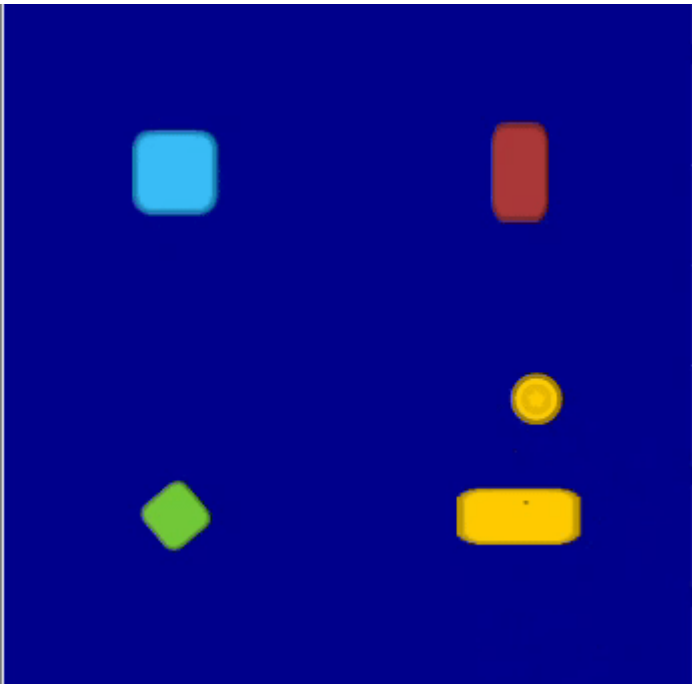


| Tema  | Estados del juego   |   |
|---|---|---|
| Descripción de la clase   | El alumno aprende a almacenar el estado (modo) de un juego en una variable. También asigna un comportamiento diferente a los objetos en el juego, dependiendo del estado del juego.   |   |
| Clase   | PRO-C6  |   |
| Duración de la clase  | 45 minutos  |   |
| Objetivo  | <ul style="list-style-type: none"> <li>Almacenar el estado (modo) de un juego en una variable.</li> <li>Mostrar información diferente en la pantalla según el estado del juego.</li> <li>Utilizar programación condicional y operadores lógicos para asignar diferentes comportamientos a los objetos en el juego, dependiendo del estado (modo) del juego.</li> </ul>  |   |
| Recursos requeridos   | <ul style="list-style-type: none"> <li>Recursos para maestras <ul style="list-style-type: none"> <li>Iniciar sesión en Code.org.</li> <li>Laptop con conectividad a internet.</li> <li>Auriculares con micrófono.</li> <li>Cuaderno y bolígrafo.</li> </ul> </li> <li>Recursos para alumnos <ul style="list-style-type: none"> <li>Iniciar sesión en Code.org.</li> <li>Laptop con conectividad a internet.</li> <li>Auriculares con micrófono.</li> <li>Cuaderno y bolígrafo.</li> </ul> </li> </ul> |   |
| Estructura de la clase  | <b>Diapositivas de Rompiendo el hielo.</b><br><b>Actividad dirigida por la maestra 1.</b><br><b>Actividad dirigida por el alumno 1.</b><br><b>Conclusión</b>  | <b>10 minutos</b><br><b>15 minutos</b><br><b>15 minutos</b><br><b>5 minutos</b> |
| <b>Rompiendo el hielo - 10 minutos</b>  |   |   |
| <div>  </div> <p>La maestra inicia la presentación de diapositivas desde la diapositiva 1 a la 11</p> <p>Consulta las notas en tu pantalla y sigue las instrucciones de cada diapositiva.</p> |   |   |
| <b>Detalles de la actividad</b>   |   | <b>Solución/Lineamientos</b>  |

| <p>¿Cómo has estado? ¿Estás emocionado de aprender algo nuevo?</p> <p><b>Ejecuta la presentación desde la diapositiva 1 a la diapositiva 4.</b></p> <p><b>Los siguientes son los entregables de la sesión de Rompiendo el hielo:</b></p> <ul style="list-style-type: none"> <li>• Conecta al alumno con la clase anterior.</li> <li>• Sesión del quiz de inicio.</li> </ul> | <p><b>REA:</b> Respuesta variada.</p> <p>Haz clic en la pestaña de presentación de diapositivas y presenta las diapositivas.</p> |
|---|--|
| <p><b>Sesión de PyR</b></p>   |  |
| <p><b>Pregunta</b></p>  | <p><b>Respuesta</b></p>  |
| <p>Selecciona el bloque de código que verifica si la moneda rebota en el sprite blue_power.</p>  <p>A. <pre>if(coin.bounceOff(blue_power)) {</pre></p> <p>B. <pre>if(coin.bounceOff(blue)) {</pre></p>  | <p><b>A</b></p>  |

|   |  |
|---|--|
| <div>C.<div><pre>if(coin.bounceOff(power)) {<br/>}<br/>if(coin.bounce(blue_power)) {<br/>}</pre></div></div> <div>D.<div><pre>if(coin.bounce(blue_power)) {<br/>}</pre></div></div>   |  |
| <p>Selecciona la línea de código que incrementa el ancho del sprite yellow_power en 5 unidades.</p> <div></div> <div>A. yellow_power.width -=5;<br/>B. yellow_powder.width = 5;<br/><b>C. yellow_powder.width +=5;</b><br/>D. yellow_powder.width ++ 5;</div> | <div>C</div>   |
| Continúa la sesión de Rompiendo el hielo  |  |
| Detalles de la actividad  | Solución/Lineamientos  |
| <p>Ejecuta la presentación desde la diapositiva 5 a la diapositiva 13 para establecer el planteamiento del problema.</p> <p>Los siguientes son los entregables de la sesión para romper el hielo:</p> <ul style="list-style-type: none"><li>• Uso de gameStates (<i>estados del juego</i>).</li></ul>   | <p>Narra las diapositivas mediante el uso de gestos con las manos y métodos de modulación de voz, para atraer más el interés de los alumnos.</p> |



La maestra finaliza la presentación de diapositivas

### ACTIVIDAD DIRIGIDA POR LA MAESTRA 1-15 minutos

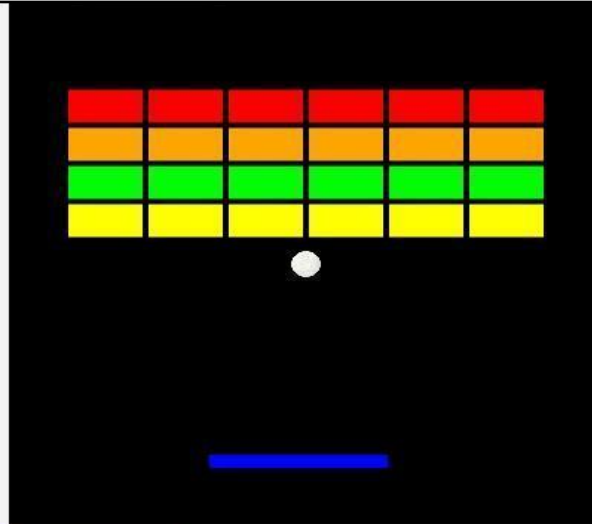
La maestra comienza a compartir su pantalla

#### DESAFÍO

- Almacena el estado (modo) del juego en una variable.
- Asigne comportamiento a los objetos del juego en función del estado (modo) del juego, mediante programación condicional.

#### Actividad dirigida por la maestra (15 minutos)

| Acción de la maestra  | Acción del alumno |
|---|-------------------|
| <p>En la última clase agregamos sonido, puntuación y aprendimos cómo mostrar texto en la pantalla del lienzo.</p> <p>Hoy agregaremos algunos textos más al juego, que pueden ayudar a los jugadores a comprender cómo iniciar o reiniciar el juego. También aprenderemos un concepto muy importante de <b>Game State</b> (mode) o <i>Estado del Juego (modo)</i>.</p> |                   |
| <p>Vamos a mostrar las instrucciones o los controles del juego como texto en la pantalla del lienzo, para que los jugadores comprendan cómo iniciar el juego.</p> <p><i>La maestra abre el <a href="#">Enlace de la actividad de la maestra 1</a> (Juego de Escape 1.5)</i></p>   |                   |



¿Observaste que tenemos que hacer clic en el botón del mouse para iniciar el juego?

**REA:** "¡Sí!"

¿Crees que deberíamos poner esta información en algún lugar de la pantalla para que el jugador sepa qué hacer?

**REA:** Si.

¿Cómo hacemos eso? ¿Qué instrucción usamos?

**REA:** instrucción text()?

```
strokeWeight(size)
rgb(r, g, b)
rect(x, y, w, h)
ellipse(x, y, w, h)
text(str, x, y)
textAlign(horiz, vert)
textFont(font)
```

Hagamos eso. Agreguemos una instrucción **text()**, cerca del centro que diga **"Haz clic para sacar la pelota"**.

*El alumno observa la pantalla y aprende el código.*

Recuerda que para la instrucción text() necesitamos dar la cadena que se mostrará, la posición x y la posición y.

*La maestra hace clic en la opción 'Show grid' (Mostrar la cuadrícula) para decidir la posición x e y del texto en el lienzo. (120, 250) puede ser uno de los valores. Anima al niño a que decida lo mismo.*

**Código:**

```
function draw() {
  background("black");
  textSize(20);
  text("Clic para sacar la pelota.", 120,250);

  text("Puntuación: "+score,40,25);

  text("Vidas: "+lives, 40, 45);
}
```

**Output:**



Entonces, ¿cómo decidimos cuándo darle la instrucción a la computadora para que saque la pelota y que la retire cuando comience el juego?

Podemos hacerlo asignando Estados del Juego al juego.

Como en la vida real, la materia cambia de sólido a líquido y a gaseoso. De la misma manera, en el juego

**REA:** variado

hay un cambio de estado (modo) como "Start, Playing, End" (*Inicio, Jugando y Fin*).

Por ejemplo, el agua almacenada en el refrigerador cambia a estado sólido (hielo); si se hierve, pasa a estado gaseoso (vapor).

Definiremos los estados del juego para nuestro juego, para averiguar en qué estado se encuentra y así poder mostrar los mensajes relevantes según el estado actual.

Si miras el juego terminado, te darás cuenta de que nuestro juego tiene 3 estados (modos) diferentes.

*El alumno escucha, observa y aprende.*

*La maestra abre el Juego de Escape terminado. (Juego de Escape completo) desde el enlace de la actividad y muestra el juego mientras explica los estados (modos).*

***Nota: la maestra también puede dar un ejemplo de cualquier otro juego del mundo real para explicar los estados (modos) del juego. (Tenis, Ping Pong)***

*Cada juego tiene 3 modos en común:*

*1) The initial Mode (el modo inicial), donde los objetos del juego están en reposo.*

*2) The Play Mode (el modo Jugando), donde se juega.*

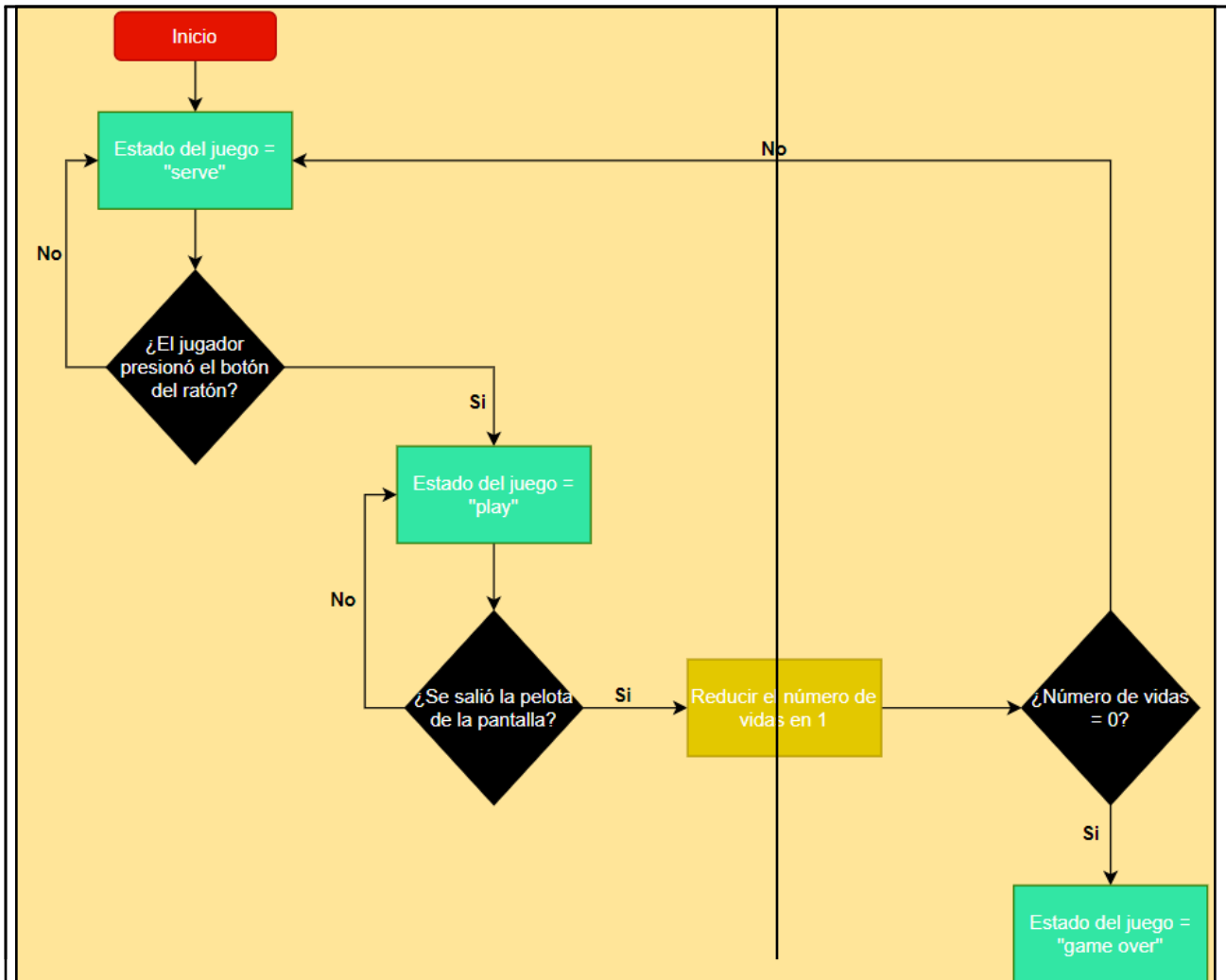
*3) The GameOver Mode (El modo Fin del Juego), donde el juego se detiene.*

***Estos estados o modos siempre siguen el mismo orden que se indicó anteriormente.***

- El primer estado (modo) es cuando la pelota está en el centro y el usuario necesita hacer "clic" para sacar la pelota. Podemos llamar a este estado en el juego **"SERVE" STATE (Serve Mode)**, que significa *ESTADO DE SERVICIO (Modo de servicio)*.
- El segundo estado (modo) inicia cuando

|   |  |
|---|--|
| <p>comienza el juego y la pelota está en movimiento. Podemos llamar a este estado (modo) en el juego <b>"PLAY" STATE (Play Mode)</b>, que significa <i>ESTADO JUGANDO (Modo de juego)</i>.</p> <ul style="list-style-type: none"> <li>El tercer estado (modo) ocurre cuando la pelota sale de la pantalla. Luego, el juego termina y el jugador debe presionar "Reset" (<i>Restablecer</i>) para reiniciar el juego. Podemos llamar a este <b>"GAME OVER" STATE (Game Over Mode)</b>, que significa <i>fin del juego (modo fin del juego)</i>.</li> </ul>   |  |
| <p>Los objetos del juego se comportan de manera diferente en distintas etapas. Incluso la información que se muestra es diferente en distintas etapas.</p> <p>Podemos almacenar la información sobre el estado (modo) de un juego y usar la programación condicional para indicarle a la computadora que se comporte de manera distinta para diferentes estados (modos).</p> <p><i>La maestra explica rápidamente la imagen del diagrama de flujo al niño, para comprender la transición entre los estados del juego o toma un ejemplo de un juego de tenis, para explicar los 3 estados del juego.</i></p> |  |





¿Cómo crees que podemos almacenar información sobre el estado (modo) del juego?

¿Qué hace la variable?

¡Sí! Vamos a crear una variable llamada **gamestate (mode)** y darle un starting state (mode) de **"serve"**.

Siempre que estemos almacenando un texto dentro de una variable, lo ponemos dentro de " ".

*La maestra escribe el código para crear una variable llamada **gamestate**.*

**REA:** ¿Usando variables?

**REA:**

Ocupa algo de espacio en la memoria de la computadora y almacena el valor que hemos asignado allí.

```

1 var ball;
2 var score = 0;
3 var gamestate = "serve";
4 ball = createSprite(200,200,10,10);
5 ball.setAnimation("golfball_1");
6 ball.scale = 0.05;
7 var paddle = createSprite(200, 350, 120, 10);
8 paddle.shapeColor = color(0,0,255);
9
10 createEdgeSprites();
11 var colors = [color(255,0,0),color(255,165,0), color(0,255,0),color(0,0,255)];
12 var BRICK_W = 50;
13 var BRICK_H = 25;
14 var BRICK_MARGIN = 4;
15
16 var offsetY = 80;

```

Ahora tenemos una variable llamada **gamestate** que tiene información sobre el estado (modo) del juego.

¿Cómo le decimos a la computadora que muestre "Haz clic para servir la pelota" sólo cuando el juego está en el 'serve state'?

**REA:**

Usamos programación condicional. Si el estado del juego (modo) es "serve", entonces muestra el texto.

¡Exactamente! Y lo hacemos así:

*La maestra escribe el código para mostrar texto únicamente cuando el estado del juego esté en "serve".*

*El alumno observa la pantalla y aprende el código.*

```

function draw() {
  background("black");
  textSize(20);
  text("Puntuación: "+score,40,25);
  text("Vidas: "+lives, 50, 50);
  if(gamestate == "serve")
  {
    text("Clic para sacar la pelota.", 120, 250);
  }
}

```

También queremos que el estado del juego (modo),

|  |   |
|--|---|
| <p>cambie después de que el usuario haga clic en un botón del mouse.</p> <p>¿Cuál queremos que sea el nuevo estado (modo) del juego?</p> <p>Cambiamos la variable gamestate a "play" después de pulsar la tecla de espacio. ¡Recuerda que los valores dentro de las variables pueden cambiar! Por eso se llaman "variables".</p> <p><i>La maestra escribe y ejecuta el código.</i></p> | <p><b>REA:</b><br/>¿Estado PLAY?</p> <p><i>El alumno observa y aprende.</i></p> |
| <pre> 22 } 23 } 24 25 function mousePressed() 26 { 27     ball.velocityY = -7; 28     ball.velocityX = 7; 29 30     if(gamestate == "serve") 31     { 32         gamestate = "play"; 33     } 34 } 35 36 } 37 </pre>   |   |
| <p>¿El texto de la pantalla desaparece cuando pulsamos el mouse y el juego está en modo play?</p> <p>Esto se debe a que gamestate (mode) cambia al hacer clic con el mouse, y le hemos pedido a la computadora que muestre un texto solo cuando el estado del juego (modo) está en "serve the state (mode)".</p>   | <p><b>REA:</b><br/>sí</p>   |
| <p>Sin embargo, todavía hay otro problema.</p> <p>Presionar el mouse en cualquier momento cambia la velocidad y la dirección de la pelota. Queremos darle velocidad y dirección a la pelota solo cuando el usuario presione la barra espaciadora en el "serve state (mode)".</p>   | <p><b>REA:</b><br/>Podemos poner instrucciones</p>                              |

|   |  |
|---|--|
| <p>¿Cómo hacemos eso?</p> <p>Podemos instruir a la computadora de la siguiente manera: si el juego está en el estado SERVE (mode), entonces solo sirve la pelota.</p> <p>Agreguemos estas instrucciones solo dentro de la "condición if".</p> <p><i>La maestra escribe y ejecuta el código.</i></p> <p><i>La maestra pulsa el "botón del mouse" mientras el juego está en modo play, para mostrar que la pelota no está cambiando de dirección a mitad de camino.</i></p> | <p>de velocidad dentro de la "condición if", para verificar si gamestate es igual a serve.</p> |
| <div data-bbox="289 802 1279 1331"> <pre> }  function mousePressed() {   if(gamestate == "serve")   {     gamestate = "play";     ball.velocityY = -7;     ball.velocityX= 7;   } } </pre> </div>   |  |
| <p>Genial, hemos realizado con éxito una transición del estado "Serve" a "Play".</p> <p>Te toca. ¿Puedes programar para el tercer modo, es decir, Game Over?</p>  |  |
| <p><b>La maestra deja de compartir la pantalla</b></p>  |  |
| <p><b>ACTIVIDAD DIRIGIDA POR EL ALUMNO 1 - 15 minutos</b></p>   |  |
| <p>Comparte tu pantalla conmigo.</p>  |  |
| <ul style="list-style-type: none"> <li>• Pídele al alumno que presione la tecla ESC para volver al panel.</li> <li>• Guía al alumno para que comience a compartir la pantalla.</li> <li>• La maestra cambia a pantalla completa.</li> </ul>   |  |

### DESAFÍO

- El alumno muestra diferente información de texto en la pantalla, según el estado del juego.
- El alumno construye la característica de "vidas de los jugadores" en el juego.
- El alumno asigna diferentes comportamientos a diferentes objetos en el juego, dependiendo del estado del juego.



La maestra inicia la presentación de diapositivas desde la diapositiva 13 a la 16

Consulta las notas en tu pantalla y sigue las instrucciones de cada diapositiva.

| Acción de la maestra   | Acción del alumno   |
|--|---|
| <p>Vamos a programar el último estado del juego, es decir, el modo Game Over.</p> <p>¿Cuántas vidas/oportunidades totales debe tener un jugador antes de terminar el juego?</p> <p>¿Cuándo crees que debería comenzar el estado Game Over?</p> <p>Estupendo. Ahora, ¿puedes decirme cuándo un jugador perderá una vida?</p> <p>Muy bien. Por último, ¿puedes indicarme una condición para comprobar si la pelota ha salido de la pantalla?</p> <p>La maestra guía al alumno para que abra la Actividad del alumno 1 y escriba el código para detectar el toque entre la pelota y el borde inferior del lienzo.</p> | <p><b>REA:</b> Variada.</p> <p><b>REA:</b> Cuando se acaben todas las vidas del jugador.</p> <p><b>REA:</b> Cuando la pelota salga de la pantalla.</p> <p><b>REA:</b> Cuando la pelota toca el borde inferior.</p> <p>El Alumno abre el <a href="#">Enlace de la actividad del alumno 1</a></p> |

```

if(!bricks[0])
{
    //console.log(";Ganaste!");
    ball.velocityX = 0;
    ball.velocityY = 0;
    text(";Ganaste!",150,200);
}
if(ball.isTouching(bottomEdge)) {
    lifeover();
}

```

*Guía al alumno para que cree una función **lifeover()** (vidas terminadas) y llama a la función cuando la pelota toque el borde inferior.*

*Además, dentro de la función **lifeover()**:*

*i) Reduce el número de vidas en 1*

*ii) if lives >= 1, cambia la variable **gamestate** de nuevo a "serve" state (mode)*

*iii) else (es decir, lives = 0), cambia el estado a "Game Over".*

*Cuida que el alumno no cometa errores tipográficos.*

*El alumno cambia el **estado del juego** de regreso a "serve" state (mode) cuando la pelota cruza la pantalla.*

*El alumno ejecuta el código para ver si el output es el deseado.*

```

3
4 function lifeover()
5 {
6     lives = lives-1;
7
8     if(lives >= 1)
9     {
10         gamestate = "serve";
11     }
12     else
13     {
14         gamestate = "end";
15     }
16 }
17

```

¡Gran trabajo! El juego parece bastante completo ahora con los estados Serve, Play y Game Over.

Hagamos nuestro código más legible creando una función **gameplay()** (como se juega) y moviendo nuestro código 'play' (como el movimiento de la paleta y la pelota rebotando) dentro de esa función.

**\*\* Nota: asegúrate de que la instrucción `drawSprites()` permanezca dentro de `draw()` solamente.**

```
function gameplay(){
  paddle.x = World.mouseX;

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

  if(paddle.x > 340)
  {
    paddle.x = 340;
  }

  //rotación = rotation + 5;
  ball.bounceOff(topEdge);
  ball.bounceOff(leftEdge);
  ball.bounceOff(rightEdge);
  //ball.bounceOff(paddle);
  ball.bounceOff(bricks, brickHit);
  if(ball.bounceOff(paddle))
  {
    playSound("sound://category_tap/puzzle_game_organic_wood_block_tone_tap_1.mp3");
  }
  if(!bricks[0])
  {
    //console.log("¡Ganaste!");
    ball.velocityX = 0;
    ball.velocityY = 0;
    text("¡Ganaste!", 150, 200);
  }
  if(ball.isTouching(bottomEdge)) {
    lifeover();
  }
}
```

Aunque hemos detectado y cambiado los estados con éxito, todavía necesitamos mostrar algunas instrucciones de texto más para que el jugador sepa si es un estado serve o un estado game over.

**La maestra ayuda al niño a completar las instrucciones de texto que se mostrarán para los diferentes estados del juego en la pantalla.**

```
function draw() {
  background("black");

  textSize(20);
  text("Puntuación: "+score,40,25);
  text("Vidas: "+lives, 40, 45);

  if(gamestate == "serve"){
    text("Clic para sacar la pelota.", 120,250);
    ball.velocityX =0;
    ball.velocityY =0;
    ball.x = 200;
    ball.y = 200;
  }
  else if(gamestate == "over") {
    text(";Fin del Juego!", 150, 250);
    ball.remove();
  }
  else {
    gameplay();
  }

  drawSprites();
}
```

Increíble. El juego parece completo ahora.

En la próxima clase te diré cómo aumentar la complejidad del juego y cómo aplicar inteligencia artificial.

### CONCLUSIÓN - 5 MINUTOS

La maestra inicia la presentación de diapositivas desde la diapositiva 17 a la 25.



### Sesión de PyR: haz clic en el quiz de la clase

| Pregunta  | Respuesta |
|---|-----------|
| ¿Cuáles son los 3 modos comunes en un juego?<br>A. The Serve mode ( <i>El modo servir</i> ) | <b>B</b>  |



|   |                                    |
|---|------------------------------------|
| <p>The Initial mode (<i>El modo inicial</i>)<br/>The End mode (<i>El modo fin</i>)</p> <p><b>B. The Initial Mode (<i>El modo inicial</i>)<br/>The Play Mode (<i>El modo de juego</i>)<br/>The GameOver (<i>El fin del juego</i>)</b></p> <p>C. The Play mode (<i>El modo de juego</i>)<br/>The GameOver mode (<i>El modo fin del juego</i>)<br/>The Begin mode (<i>El modo de inicio</i>)</p> <p>D. The Initial Mode (<i>El modo inicial</i>)<br/>The End (<i>El fin</i>)<br/>The GameOver (<i>El fin del juego</i>)</p>  |                                    |
| <p>¿Cuál de los siguientes es un tipo de enunciado condicional?</p> <p><b>A. Declaración if</b></p> <p>B. Bucle for</p> <p>C. var</p> <p>D. función</p>   | <b>A</b>                           |
| <p>¿Qué hace el siguiente código en el Juego de Escape?</p> <div data-bbox="190 919 889 1138" data-label="Text"> <pre> } if(ball.isTouching(bottomEdge)) {     lifeover(); } </pre> </div> <p>A. Detecta el toque entre la pelota y el borde inferior del lienzo y llama a una función predefinida llamada <b>lifeover()</b>.</p> <p>B. Llama a una función <b>lifeover()</b> cuando la pelota toca cualquiera de los topEdge (<i>bordes superiores</i>).</p> <p>C. Llama a una función <b>lifeover()</b> cuando la pelota toca alguno de los bordes.</p> <p><b>D. Detecta el contacto entre la pelota y el borde inferior del lienzo y llama a una función personalizada lifeover().</b></p> | <b>D</b>                           |
|   |                                    |
| <p><b>Descripción del proyecto</b></p>  | <p><i>El alumno interactúa</i></p> |

*con la maestra durante el proyecto.*

## EL JUEGO DE PELOTA

### Objetivo del proyecto:

Hoy aprendiste el uso de la programación condicional, las funciones y la colocación de estados de juego en el proyecto.

En este proyecto tendrás que practicar y aplicar lo que has aprendido hasta ahora y crear un juego de pelota para un jugador, para una empresa que hace algunos juegos innovadores para niños.

### Historia:

Una empresa, Crafty Child, se centra en la creación de juegos innovadores para niños. Esta vez están tratando de hacer una nueva versión de un juego muy popular, que seguramente jugaste en equipo.

Están tratando de hacer un juego de pelota para un solo jugador, con el cual un niño puede practicar el juego incluso si no puede salir o no tiene un equipo con quien jugar.

¿Puedes ayudar a Crafty Child a crear este nuevo e interesante juego?

Estoy muy emocionada de ver la solución de tu proyecto y sé que lo harás muy bien.

¡Adiós!

**La maestra finaliza la presentación de diapositivas**



**ACTIVIDAD ADICIONAL (solo para alumnos rápidos)**  
**(Para crear la función de reanudación de pausa en el juego: 7-10 minutos)**

¿Alguna vez te ha pasado que estás jugando y de repente tus padres te llaman para un trabajo urgente?

**REA:** Sí.

¿No sería genial si tuviéramos una función de pausa y reanudación en nuestro juego?

**REA:** Sí.

¿Cómo pausamos generalmente nuestro juego o película?

**REA:** Pulsando la tecla "espacio" o pulsando la tecla

“pausa”.

Entendamos rápidamente cómo podemos detectar si se pulsa una tecla. Por cada pulsación de tecla, se activan 3 eventos en JavaScript:

- i) **keyWentDown(key)** o *la tecla va hacia abajo(tecla)*: **se dispara** una vez tan pronto como se presiona la tecla.
- ii) **keyDown(key)** o *la tecla está abajo(tecla)*: se **sigue disparando** hasta que la tecla está abajo.
- iii) **keyWentUp(key)** o *la tecla va hacia arriba(tecla)*: **se dispara** tan pronto como se suelta la tecla.

*La maestra debe mostrar estas tres instrucciones al niño en la pestaña World (Mundo) en la Caja de herramientas.*

¿Puedes decirme qué evento debemos verificar para detectar si la tecla de espacio está presionada?

**REA:** keyWentDown('SPACE');  
- *la Tecla va hacia abajo (espacio)*

¡Estupendo! Ahora conoces los eventos de pulsación de teclas, déjame mostrarte el diagrama de flujo para crear estados de pausa y reanudar el juego en nuestro juego.

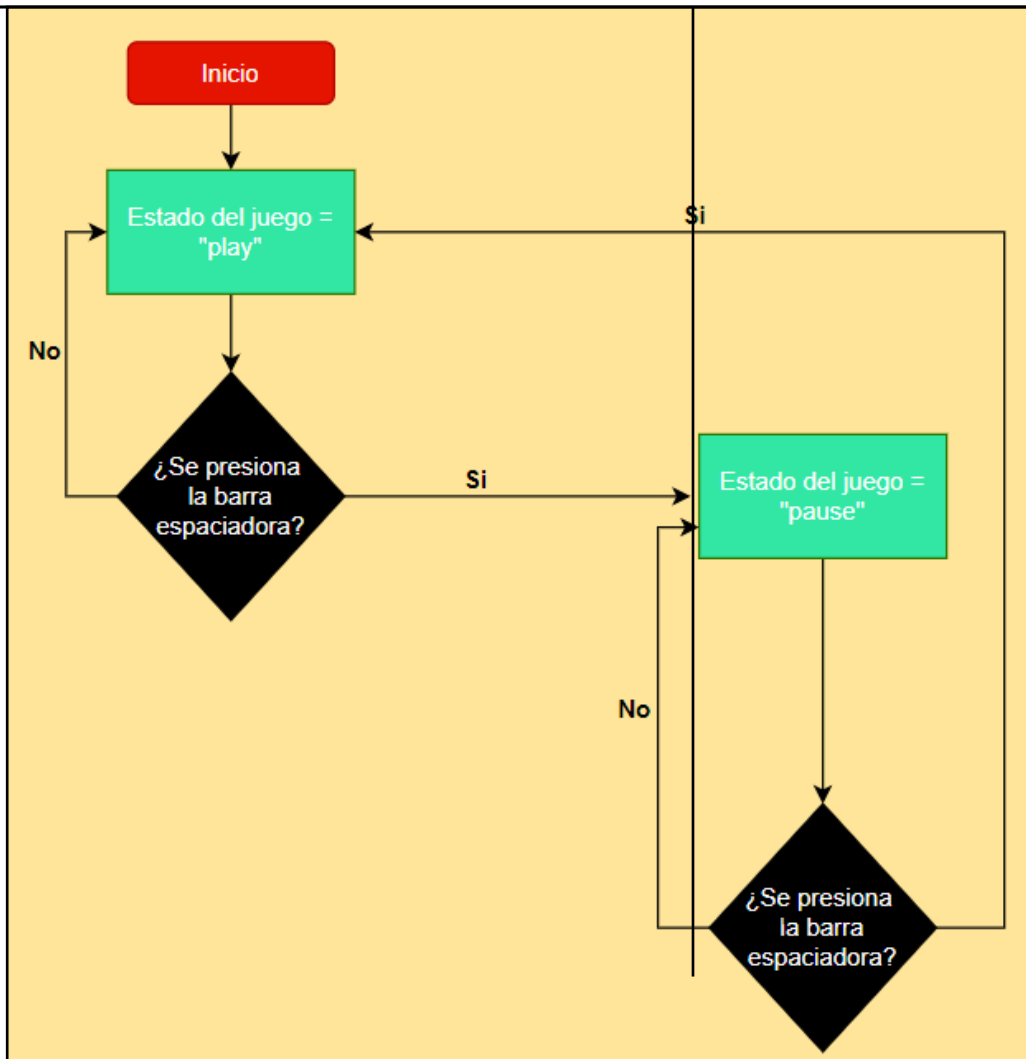


Diagrama de flujo

*La maestra guía al alumno con el Paso 1.*

*Paso 1: detecta una pulsación de tecla en la barra espaciadora y cambia el estado del juego entre play - jugar y pause - pausa.*

```

    }
    if (keyWentDown("space"))
    {
        if(gamestate == "pause")
        {
            gamestate = "play";
            ball.velocityY = -7;
            ball.velocityX= 7;
        }
        else if(gamestate == "play")
        {
            gamestate = "pause";
            ball.velocityX = 0;
            ball.velocityY = 0;
        }
    }
    drawSprites();
}

```

*¿Notaste que cuando reanudamos el juego después de la pausa, el “sprite de la pelota” va en la misma dirección? ¿Parece un poco extraño?*

*¿No crees que el “sprite de la pelota” debería reanudarse en la misma dirección y con la misma velocidad que tenía antes de la pausa?*

*La maestra le explicará al niño cómo almacenar las últimas velocidades en la dirección x e y en otra variable global. (Las variables que se pueden usar en cualquier parte del código)  
Y también asigna las mismas velocidades al sprite de la pelota cuando el juego se reanude desde la pausa.*

```

55     }
56
57     if (keyWentDown("space"))
58     {
59         if(gamestate == "pause")
60         {
61             gamestate = "play";
62             ball.velocityY = lastvy;
63             ball.velocityX= lastvx;
64         }
65         else if(gamestate == "play")
66         {
67             gamestate = "pause";
68             lastvx = ball.velocityX;
69             lastvy = ball.velocityY;
70             ball.velocityX = 0;
71             ball.velocityY = 0;
72         }
73     }
74 }
75 drawSprites();
76 }
77

```

Vamos a mostrar algunas instrucciones de texto dentro de draw() para que el jugador sepa si es un pause state (estado pausa).

```

else if(gamestate == "end")
{
    text("¡Fin del Juego! , 150, 250);
    ball.remove();
}
else if(gamestate == "pause")
{
    text("Presiona la barra espaciadora para reiniciar." , 110, 250);
}
else
{
    gameplay();
}

```

| Actividad   | Nombre de la actividad           | Enlaces   |
|---|----------------------------------|---|
| Enlace de la actividad de la maestra 1                        | Juego de Escape 1.5              | <a href="https://studio.code.org/projects/gamelab/WRxEHfbi6Xs9Rhwaq0i64KkM0MVfyOoRDmPYEkn1TE">https://studio.code.org/projects/gamelab/WRxEHfbi6Xs9Rhwaq0i64KkM0MVfyOoRDmPYEkn1TE</a>           |
| Enlace de la actividad de la maestra 1 (código de referencia) | Estado del Juego                 | <a href="https://studio.code.org/projects/gamelab/WRxEHfbi6Xs9Rhwaq0i6_hAjcBZsewqvx_HwZaA0A0">https://studio.code.org/projects/gamelab/WRxEHfbi6Xs9Rhwaq0i6_hAjcBZsewqvx_HwZaA0A0</a>           |
| Enlace de la actividad del alumno 1                           | Estado del Juego                 | <a href="https://studio.code.org/projects/gamelab/-pXwW9J1hMupJ61eZQHY2RQNMirC4InAYz2uZkYYIBU">https://studio.code.org/projects/gamelab/-pXwW9J1hMupJ61eZQHY2RQNMirC4InAYz2uZkYYIBU</a>         |
| Código de referencia de la maestra (Juego de Escape 1.6)      | Juego de Escape 1.6              | <a href="https://studio.code.org/projects/gamelab/iyi5VICb2iS9bog3rM2ilQ8CL7i1ilyXWfJz-9OZqYc">https://studio.code.org/projects/gamelab/iyi5VICb2iS9bog3rM2ilQ8CL7i1ilyXWfJz-9OZqYc</a>         |
| Solución del proyecto   | EL JUEGO DE PELOTA               | <a href="https://studio.code.org/projects/gamelab/D5NI7zWuDout0rx4kCGm1kNIOP2KYMRApBNfBtBBYw8">https://studio.code.org/projects/gamelab/D5NI7zWuDout0rx4kCGm1kNIOP2KYMRApBNfBtBBYw8</a>         |
| Enlace de ayuda visual de referencia de la maestra            | Enlace de ayuda visual con notas | <a href="https://s3-whjr-curriculum-uploads.whjr.online/0873e9d9-507c-414a-9942-beaa5c866e41.html">https://s3-whjr-curriculum-uploads.whjr.online/0873e9d9-507c-414a-9942-beaa5c866e41.html</a> |
| Enlace de apoyo visual de referencia de la maestra            | Enlace de ayuda visual sin notas | <a href="https://s3-whjr-curriculum-uploads.whjr.online/40c088c3-a8d1-4dd9-8219-36b7ec48f604.html">https://s3-whjr-curriculum-uploads.whjr.online/40c088c3-a8d1-4dd9-8219-36b7ec48f604.html</a> |
| Referencia del Quiz en clase de la maestra                    | Quiz en clase                    | <a href="https://s3-whjr-curriculum-uploads.whjr.online/2992472c-7cc8-4734-8460-da5aa8a77375.pdf">https://s3-whjr-curriculum-uploads.whjr.online/2992472c-7cc8-4734-8460-da5aa8a77375.pdf</a>   |
| Enlace de la actividad adicional (Código de                   | Pausar el juego                  | <a href="https://studio.code.org/projects/gamelab/YMML_Kj4pxA_k4TGwXg_Za68beaoWQ1QYEbbyuv2-7M">https://studio.code.org/projects/gamelab/YMML_Kj4pxA_k4TGwXg_Za68beaoWQ1QYEbbyuv2-7M</a>         |

|             |  |  |
|-------------|--|--|
| referencia) |  |  |
|-------------|--|--|