



Tema	Trucos y Consejos de Depuración	
Descripción de la Clase	Los alumnos aprenden consejos y trucos para minimizar errores y bugs en el código. Los alumnos, también aprenden cómo depurar los errores a través de las diferentes técnicas de depuración.	
Clase	C33	
Duración de la Clase	45 minutos	
Objetivo	<ul style="list-style-type: none"> • Aprender a minimizar errores y bugs en el código. • Aprender diferentes técnicas para depurar códigos. • Depurar la trayectoria del Angry Bird cuando se dispara varias veces. 	
Recursos Requeridos	<ul style="list-style-type: none"> • Recursos para Profesores <ul style="list-style-type: none"> ○ Laptop con conectividad a internet ○ Auriculares con micrófono ○ Cuaderno y bolígrafo • Recursos para Alumnos <ul style="list-style-type: none"> ○ Laptop con conectividad a internet ○ Auriculares con micrófono ○ Cuaderno y bolígrafo 	
Estructura de la Clase	Rompiendo el Hielo Actividad dirigida por el Profesor Actividad dirigida por el Alumno Conclusión	5 minutos 15 minutos 15 minutos 5 minutos
<div> <div></div> <div></div> </div>		
<p style="text-align: center;">CONTEXTO</p> <ul style="list-style-type: none"> • Introducir la depuración como parte normal de un ciclo de programación. 		
Pasos de Clase	Acción del Profesor	Acción del Alumno
Paso 1: Rompiendo el Hielo (5 minutos)	Hemos hecho mucha programación en las últimas 32 clases. Hemos	

	<p>escrito cientos de líneas de código.</p> <p>¿Han funcionado siempre como queríamos?</p>	<p>REA: No</p>
	<p>¿Qué hacemos cuando los programas no funcionan como queremos?</p>	<p>REA: Comprobamos si hay errores/bugs en el código, y los arreglamos.</p>
	<p>Sí, encontramos los bugs/errores y los arreglamos.</p> <p>Sabías que... ¿Cualquier programador de computadoras, en su vida dedicará más tiempo a depurar que a escribir un nuevo código?</p> <p>La depuración es una parte normal de la programación. Esto se debe a que le estamos dando instrucciones a una computadora, y estamos tratando de pensar como una computadora. Sin embargo, somos humanos y no pensamos naturalmente como las computadoras.</p> <p>Los buenos programadores aceptan que siempre habrá errores en su código, y DISFRUTAN del proceso de depuración y corrección de errores en su código.</p> <p>La depuración es tanto un arte como una ciencia de esa manera. Es como un trabajo de detective, en el que intentas encontrar qué línea de código o patrones de pensamiento en</p>	<p><i>El alumno escucha.</i></p>

	tu mente te llevan a escribir algo que está causando un error.	
	<p>¡Tengo una emocionante pregunta de prueba para ti! ¿Estás listo para responder a esta pregunta?</p> <p>Por favor haga clic en el botón</p>  <p>que está en la esquina inferior derecha de su pantalla para iniciar el Cuestionario en Clase.</p> <p>El cuestionario será visible para usted y el alumno.</p> <p>Anime al alumno a responder la pregunta del cuestionario.</p> <p>Puede que el alumno elija la opción incorrecta. Ayúdele a pensar bien la pregunta y pídale que responda de nuevo.</p> <p>Cuando elija la opción correcta el</p>  <p>botón aparecerá en su pantalla.</p> <p>Haga clic en Finalizar Cuestionario para cerrar la ventana y continúe con la clase.</p>	REA: Sí
	<p>Hoy, aprenderemos los consejos y trucos de cómo depurar de forma eficaz.</p> <p>Al final de la clase, el objetivo es sentirse cómodo con los errores, y no verlo como algo INCORRECTO.</p>	

	<p>Es una parte normal de la programación y deberíamos sentirnos cómodos con ella, incluso disfrutarla.</p> <p>¿Estás listo para comenzar?</p>	<p>REA: ¡Sí!</p>
El Profesor Inicia Compartir Pantalla		
<p><u>DESAFÍO</u></p> <ul style="list-style-type: none"> • Presenta las diferentes técnicas de depuración. • Aplica las técnicas de depuración a un problema en particular. 		
<p>Paso 2: Actividad dirigida por el Profesor (15 minutos)</p>	<p>Bien, describiré una serie de técnicas que los programadores usan para resolver un error/bug en su código.</p> <p>¡Entonces, juntos resolveremos un error en nuestro juego de Angry Birds! ¿Acaso no es genial?</p>	<p>REA: ¡Sí!</p>
	<p>¿Qué es lo primero que debes hacer cuando se te cuenta de que el código no está haciendo lo que tú quieres que haga?</p> <p>¡Lo primero que debes hacer es NO ENTRAR EN PÁNICO! Los buenos programadores no se asustan. Se dan cuenta de que la depuración es una parte normal de la programación. Los errores son normales. Todos los programas tendrán errores.</p> <p>Estarás en una posición mucho mejor para encontrar y resolver estos errores, si está tranquilo y sereno.</p>	<p>REA: variado</p>

	<p>¡Lo segundo que debe hacer es, verificar si hay errores en el registro de la consola!</p> <p>Si hay algún mensaje de error en la consola, a menudo te dicen qué sucedió y por qué ocurrió este error, o te indican en esa dirección.</p> <p>A menudo, puedes corregir los errores en tu código leyendo estos mensajes de error.</p> <p>Puedes abrir la consola en tu navegador presionando Ctrl + Shift + J</p> <p>Los errores más comunes son:</p> <ol style="list-style-type: none"> Errores tipográficos: Accidentalmente escribiste mal un nombre de variable o función, que la computadora no comprende. Uso Incorrecto de la función: Usaste una función de una manera que no estaba destinada a usarse. Usar variables fuera de tu alcance: Si estás utilizando variables fuera de tu alcance, la computadora no sabría el valor de estas variables. 	<p><i>El alumno escucha y hace preguntas.</i></p>
	<p>A menudo, no hay mensajes de error en la consola y, sin embargo, tu</p>	<p><i>El alumno escucha y hace preguntas.</i></p>

	<p>programa no se comporta de la forma esperada.</p> <p>O el mensaje de error no te dice todo lo que debe solucionarse.</p> <p>Existen algunas técnicas que te ayudarán a identificar qué está causando el error, o qué sección de tu código está provocando este error.</p> <ol style="list-style-type: none"> 1. Comentar secciones de tu código: Puedes comentar ciertas secciones de tu código para simplificar tu código, y verificar si tu código todavía arroja errores. De esta manera, puedes limitarte a la parte del código que está causando este error. 2. Impresión de valores de variables en la consola: Puedes imprimir los valores de las variables críticas en la consola, para ver visualmente cómo están cambiando en el código. Si no cambian como esperabas, está sucediendo algo inesperado. Puedes manipular los valores de las variables para identificar lo que está sucediendo. 3. Imprime mensajes en tu código: Puedes imprimir mensajes en tu código para 	
--	--	--

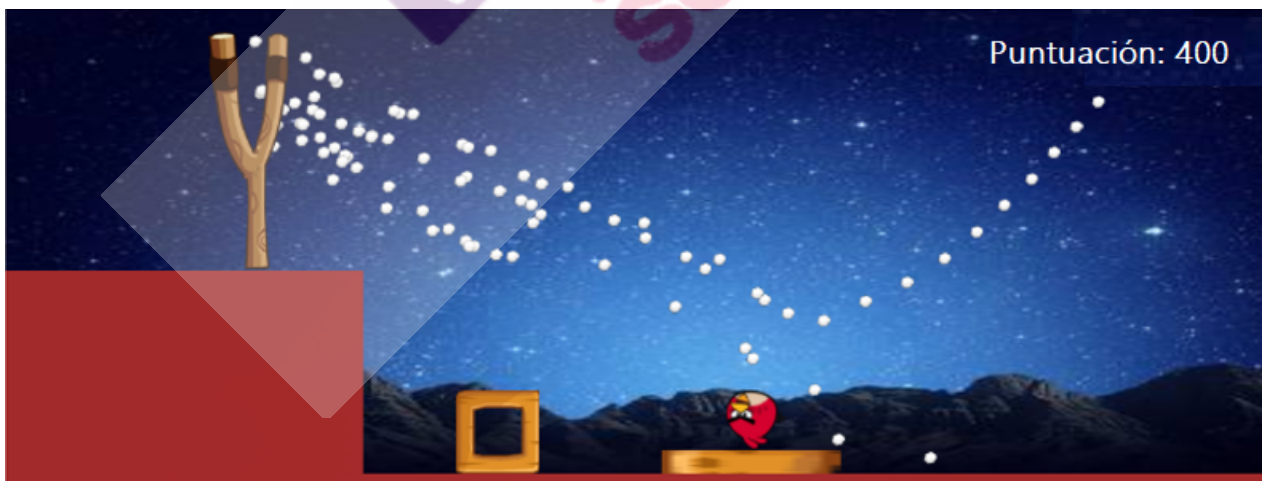
	<p>comprender visualmente cómo se ejecuta el código. Por ejemplo: puedes imprimir mensajes en la consola para comprender si mientras ejecutas el código, se está ejecutando un 'bloque if' o el 'bloque else'.</p> <p>También, puedes combinar los 3 pasos.</p> <p>Antes de ejecutar el código, uno siempre debe tener en mente lo que espera, y ver la diferencia entre lo que realmente está sucediendo.</p>	
	<p>Una última cosa, antes de comenzar nuestra misión de depuración:</p> <p>Escribir un código modular limpio, formatear tu código correctamente, dar un nombre adecuado a tus funciones, objetos y variables, agregar comentarios adecuados; todo esto toma solo un poco de tiempo, pero ahorra mucho tiempo cuando estás depurando tu código.</p> <p>Hace que tu código sea más fácil de leer/comprender y, por lo tanto, más fácil de identificar los errores.</p>	<i>El alumno escucha y hace preguntas.</i>
	<i>El profesor ayuda al alumno a clonar el código de la Actividad del Alumno 1.</i>	<i>El alumno clona el proyecto de Actividad del Alumno 1 y lo abre en VS Code.</i>

	<p><i>El profesor abre el enlace e inicia la colaboración en vivo.</i></p>	<p><i>Habilita la función de compartir en vivo, y comparte el enlace con el profesor.</i></p>
	<p>En la última clase, no permitimos que los jugadores dispararan múltiples veces a los cerdos. Digamos que queremos eso. Cambiemos el código para permitirle al jugador disparar a los cerdos tantas veces como quiera.</p> <p>El profesor comenta y descomenta partes del texto, que permitirían al jugador disparar múltiples disparos al objetivo.</p>	<p><i>El alumno observa.</i></p>


```

home > rajeev > Lessons > AngryBirdsStage8 > JS sketch.js > keyPressed
72
73   box5.display();
74   log4.display();
75   log5.display();
76
77   bird.display();
78   platform.display();
79   //log6.display();
80   slingshot.display();
81 }
82
83 function mouseDragged(){
84   //if (gameState!=="launched"){
85     Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
86   //}
87 }
88
89
90 function mouseReleased(){
91   slingshot.fly();
92   gameState = "launched";
93 }
94
95 function keyPressed(){
96   if(keyCode === 32){
97     slingshot.attach(bird.body);
98   }
99 }
100
101 async function getBackgroundImg(){
102   var response = await fetch("http://worldtimeapi.org/api/timezone/Asia/Kolkata");
103   var responseJSON = await response.json();
104
105   var datetime = responseJSON.datetime;
106   var hour = datetime.slice(11,13);

```




El profesor ejecuta el código y dispara varias veces hacia los cerdos.

REA:

- Hay múltiples trayectorias del ave en la pantalla.
Cuando reiniciamos el ave

	¿Cuál es el error/comportamiento no deseado que tenemos en el código?	de nuevo a la resortera. Quisiéramos que se borrara la trayectoria anterior. - El ave se balancea rápidamente cuando se reinicia. Esto no es deseado. También, provoca marcas de trayectoria adicionales.
	<p>Bien, sabemos que hay errores en nuestro código.</p> <p>¿Por qué no intentas resolverlos usando los consejos y técnicas de los que acabamos de hablar?</p>	<i>El alumno se prepara para la misión de depuración.</i>
El Profesor Detiene Compartir Pantalla		
	Ahora es tu turno. Comparte tu pantalla conmigo.	
<ul style="list-style-type: none"> • Pídale al Alumno que presione la tecla ESC para volver al panel • Guíe al Alumno para que comience a Compartir Pantalla • El Profesor entra en Pantalla Completa 		
<p style="text-align: center;"><u>ACTIVIDAD</u></p> <ul style="list-style-type: none"> • Depura la trayectoria del ave. 		
Paso 3: Actividad dirigida por el Alumno (15 minutos)	¿Qué es lo primero que debes hacer para comenzar a depurar?	REA: ¡Mantén la calma y no entres en pánico!

	<p>Bien. ¿Esperas algún error en la consola para este error?</p> <p>¿Por qué?</p>	<p>REA: No</p> <p>REA: El código no hace nada inesperado. Solo el resultado es indeseable para nosotros.</p>
	<p>¿Puedes identificar la parte del código que se encarga de trazar la trayectoria?</p>	<p>REA: ¡Sí!</p> <p><i>El alumno señala la trayectoria y la función de visualización en la consola donde se dibuja la imagen de la trayectoria.</i></p>
 <pre> 1 class Bird extends BaseClass { 2 constructor(x,y){ 3 super(x,y,50,50); 4 this.image = loadImage("sprites/bird.png"); 5 this.smokeImage = loadImage("sprites/smoke.png"); 6 this.trajectory = []; 7 } 8 9 display() { 10 //this.body.position.x = mouseX; 11 //this.body.position.y = mouseY; 12 13 super.display(); 14 15 if(this.body.velocity.x > 10 && this.body.position.x > 200){ 16 var position = [this.body.position.x, this.body.position.y]; 17 this.trajectory.push(position); 18 } 19 20 for(var i=0; i<this.trajectory.length; i++){ 21 image(this.smokeImage, this.trajectory[i][0], this.trajectory[i][1]); 22 } 23 } 24 } 25 </pre>		
	<p>Entonces, ¿qué harías si quisieras eliminar la trayectoria cuando se presiona la tecla de la barra espaciadora?</p>	<p><i>El alumno intenta algunas cosas.</i></p>

	<p><i>Permita que el alumno experimente con su código y depure este comportamiento.</i></p>	
	<p>Todas las posiciones de las aves se almacenan en la propiedad array (matriz) de Bird (Ave), llamada trajectory (trayectoria). Simplemente podemos vaciar esta matriz cada vez que se presione la tecla de la barra espaciadora.</p> <p><i>Guíe al alumno a vaciar la trayectoria del ave cuando se presione la tecla de la barra espaciadora.</i></p> <p>Simplemente puede vaciar una matriz sin asignar ningún valor entre corchetes vacíos.</p>	<p><i>El alumno escribe el código para vaciar la trayectoria del ave.</i></p> <p><i>El alumno ejecuta el código para ver el resultado.</i></p>

```

77   bird.display();
78   platform.display();
79   //log6.display();
80   slingshot.display();
81 }
82
83 function mouseDragged(){
84   //if (gameState!=="launched"){
85     Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
86   //}
87 }
88
89
90 function mouseReleased(){
91   slingshot.fly();
92   gameState = "launched";
93 }
94
95 function keyPressed(){
96   if(keyCode === 32){
97     bird.trajectory = [];
98     slingshot.attach(bird.body);
99   }
100 }
101
102 async function getBackgroundImg(){
103   var response = await fetch("http://worldtimeapi.org/api/timezone/Asia/Kolkata");
104   var responseJSON = await response.json();
105
106   var datetime = responseJSON.datetime;
107   var hour = datetime.slice(11,13);
108
109   if(hour>=0600 && hour<=1900){
110     bg = "sprites/bg1.png";

```

	<p>Entonces, la vieja trayectoria parece desaparecer cuando el ave se reinicia.</p> <p>¿Cuál es el otro error ahora?</p>	<p>REA:</p> <p>El ave se balancea ampliamente cuando se sujeta a la resorte.</p>
	<p>¿Por qué pasa eso?</p> <p>¿Qué podemos hacer para que no se balancee tanto?</p>	<p>REA:</p> <p>Porque el ave se engancha a la resorte desde su última posición.</p> <p>REA:</p> <p>Podemos restablecer la posición del ave cuando se</p>

		presiona la barra espaciadora.
	<p>¡Impresionante! Vamos a hacer eso.</p> <p><i>Guíe al alumno para que utilice <code>Matter.Body.setPosition</code> (primero por sí solo) para colocar al ave.</i></p> <p><i>Permita que el alumno cometa errores, y luego use los consejos y técnicas de depuración cuando encuentre los errores.</i></p> <p>Referencia de la Actividad 2 del Profesor</p>	<p><i>El alumno restablece la posición del ave cuando se presiona la barra espaciadora, en el lugar cerca de la resorte.</i></p> <p><i>Referencia de la Actividad del Alumno 2.</i></p> <p><i>El alumno ejecuta el código para verificar si hay errores, y luego lo depura.</i></p>

```

75   log5.display():
76
77   bird.display():
78   platform.display():
79   //log6.display():
80   slingshot.display():
81 }
82
83 function mouseDragged(){
84   //if (gameState!=="launched"){
85     Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
86   //}
87 }
88
89
90 function mouseReleased(){
91   slingshot.fly();
92   gameState = "launched";
93 }
94
95 function keyPressed(){
96   if(keyCode === 32){
97     bird.trajectory = [];
98     Matter.Body.setPosition(bird.body, {x:200, y:50});
99     slingshot.attach(bird.body);
100   }
101 }
102
103 async function getBackgroundImg(){
104   var response = await fetch("http://worldtimeapi.org/api/timezone/Asia/Kolkata");
105   var responseJSON = await response.json();
106
107   var datetime = responseJSON.datetime;
108   var hour = datetime.slice(11,13);




```

	Esto parece haber solucionado el error.	-
--	---	---

El Profesor Guía al Alumno para Dejar de Compartir Pantalla

- RETROALIMENTACIÓN**
- Anime al alumno a tomar notas de reflexión en markdown.
 - Felicite al alumno por sus esfuerzos durante la clase.
 - Repasen el contenido aprendido.

Paso 4: Conclusión (5 minutos)	<p>Recuerda, cada código tiene errores.</p> <p>Incluso ahora, todavía tenemos un error. Cuando presionas la barra espaciadora mientras el ave aún se está moviendo, observa lo que sucede.</p>	<i>El alumno ve que el ave se reinicia mientras el ave todavía se está moviendo.</i>
---	--	--

	<p>¿Cómo arreglarías este error? ¡Puedes intentar corregir este error como tarea, y usar las técnicas que aprendimos mientras lo hacías!</p>	<p><i>El alumno se compromete a solucionar este error.</i></p>
	<p>¡Felicidades por tu excelente trabajo!</p> <p>En la próxima clase, comenzaremos con un nuevo proyecto: un juego de carreras de autos multijugador, que tú y tus amigos pueden jugar en diferentes sistemas.</p> <p>Estoy muy emocionado de comenzar el proyecto contigo. ¡Espero que tú también lo estés!</p>	<p><i>Asegúrese de dar al menos 2 felicitaciones al estudiante por:</i></p> <div>  <p>Resolver Creativamente las Actividades</p> </div> <div>  <p>Muy Buena Pregunta</p> </div> <div>  <p>¡Te Concentraste!</p> </div>
<p><u>Nombre del Proyecto: Juego del Motor Físico</u></p>	<p>Objetivo del Proyecto:</p> <p>En la clase 33, aprendiste el concepto de depuración.</p> <p>En este proyecto, tendrás que practicar y aplicar los conceptos aprendidos en la clase, y tendrás que usar un motor físico para crear tu propio juego.</p> <p>Historia:</p> <p>Si puedes recordar, la mayoría de tus juegos favoritos y adictivos eran muy realistas al jugar, o simplemente, el mundo del juego era muy similar a tu propio mundo.</p>	<p><i>Los alumnos interactúa con el profesor en el proyecto.</i></p>

	<p>¡Hoy vas a desarrollar un juego similar para deleitar a tus jugadores! ¡¡Todo lo mejor!!</p> <p>Estoy muy emocionado de ver la solución de tu proyecto, y sé que lo harás realmente bien.</p> <p>¡Adiós!</p>	
<div> <div>El Profesor hace Clic en</div> <div>✕ Finalizar Clase</div> </div>		
Actividades Adicionales	<p><i>Guíe al alumno para que permita reinicios, solo cuando el ave haya dejado de moverse; agregando una condición adicional en la declaración if.</i></p>	<p>REA: ¡¡Sí!!</p> <p><i>El alumno primero usa <code>bird.body.speed === 0</code> como condición adicional.</i></p> <p>El código no funciona.</p> <p><i>El profesor anima al alumno a imprimir la velocidad del ave en la consola.</i></p> <p><i>El alumno luego usa la condición <code>bird.body.speed < 1</code> en el código.</i></p>

```
js sketch.js | draw
68 box4.display();
69 pig3.display();
70 pig3.score();
71 log3.display();
72
73 box5.display();
74 log4.display();
75 log5.display();
76
77 bird.display();
78 platform.display();
79 //log6.display();
80 slingshot.display();
81 console.log(bird.body.speed);
82 }
83
84 function mouseDragged(){
85   //if (gameState!=="launched"){
86     Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
87   //}
88 }
89
90
91 function mouseReleased(){
92   slingshot.fly();
93   gameState = "launched";
94 }
95
96 function keyPressed(){
97   if (keyCode === 32 && bird.body.speed < 1){
98     bird.trajectory = [];
99     Matter.Body.setPosition(bird.body, {x:200, y:50});
100    slingshot.attach(bird.body);
```

Actividad	Nombre de la Actividad	Enlaces
Actividad del Profesor 1	Angry Birds Etapa 8	https://github.com/whitehatjr/AngryBirdsStage8 En Español: https://github.com/alejandraluna1/AngryBirdsEtapa8.git
Actividad del Profesor 2	Enlace de Referencia Matter.Body	http://brm.io/matter-js/docs/classes/Body.html
Actividad del Profesor 3	Link de Referencia	https://github.com/whitehatjr/AngryBirdsStage9 En Español: https://github.com/alejandraluna1/AngryBirdsEtapa9.git
Actividad del Alumno 1	Angry Birds Etapa 8.5	https://github.com/whitehatjr/AngryBirdsStage8.5 En Español: https://github.com/alejandraluna1/AngryBirdsEtapa8.5.git
Actividad del Alumno 2	Enlace de Referencia Matter.Body	http://brm.io/matter-js/docs/classes/Body.html
Solución del Proyecto	Juego del Motor Físico	Al tratarse de un proyecto abierto, no existe solución específica