
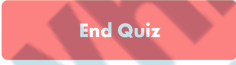


	<p>I have an exciting quiz question for you! Are you ready to answer this question?</p>  <p>Teacher click on the button on the bottom right corner of your screen to start the In-Class Quiz.</p> <p>A quiz will be visible to both you and the student.</p> <p>Encourage the student to answer the quiz question.</p> <p>The student may choose the wrong option, help the student to think correctly about the question and then answer again.</p> <p>After the student selects the correct option, the  button will start appearing on your screen.</p> <p>Click the End quiz to close the quiz pop-up and continue the class.</p>	<p>ESR: Yes</p>
	<p>In today's class, we will be writing code to make the pigs vanish when they are attacked by the bird.</p> <p>In the last class, we looked at mouse trigger events - like mouseDragged, mouseReleased. In today's class, we will also look at the keyBoard and how the computer recognizes which keys are pressed.</p> <p>Let's get started.</p>	<p>-</p>

Teacher Initiates Screen Share		
<p align="center">CHALLENGE</p> <ul style="list-style-type: none"> Remove the pig object from the world when the pig is hit. 		
<p>Step 2: Teacher-led Activity (15 min)</p>	<p><i>Teacher helps the student clone the code from the Student Activity 1.</i></p> <p><i>Teacher opens the link and starts the live collaboration.</i></p>	<p><i>Student clones the project from Student Activity 1 and opens it in VS Code.</i></p> <p><i>Enables the live share and shares the link with the teacher.</i></p>
<p>\$ git clone https://github.com/whitehatjr/AngryBirdsStage5</p>		
	<p>Our game is working fine. But when the angry bird attacks the enemy pigs, they remain on the screen. What should happen to them, when they are attacked?</p> <p><i>Teacher launches the web server and shows the game to the student.</i></p>	<p>ESR:</p> <p>The enemy pigs should vanish from the screen.</p>
	<p>How do we know that a bird attack has hit the pig?</p> <p>Not really, the log is already touching the pig right now. Also a touch doesn't imply it has been sufficiently hit.</p> <p>When a body is hit, some of its properties will change - like its speed, momentum etc.</p> <p>For now let's try to output the pig's speed when it is hit by the bird and see how/if it changes.</p>	<p>ESR:</p> <p>When the logs, boxes or the bird touches the pig?</p>

We are dealing with the pig objects.
Which file should we write our code in?

Yes! We can also write our code in
script.js. But to preserve readability,
we should write in pig.js

How can we display the pig's speed?

display() function is called in every
frame since it is called inside draw().
Let's write the console.log() command
here.

*Teacher writes code to print the pig
object's speed on the console before
displaying it.*

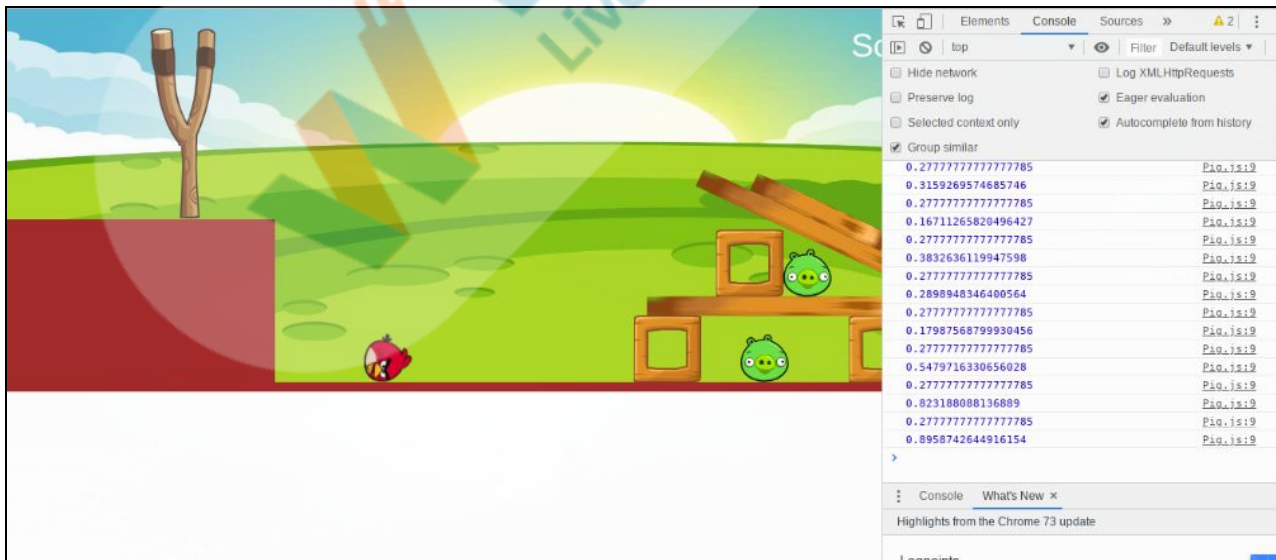
*Teacher runs the code and hits the
pig with the bird.*

What do you see?

ESR:
Pig.js?

By using console.log() and
displaying the pig's speed
inside it.

ESR:
The pig's speed increases
when it is hit.



```

1  class Pig extends BaseClass {
2      constructor(x, y){
3          super(x,y,50,50);
4          this.image = loadImage("sprites/enemy.png");
5          this.Visiblity = 255;
6      }
7
8      display(){
9          console.log(this.body.speed);
10         super.display();
11     }
12
13     |
14
15 };

```

Let's use this change in speed of the pig object to decide when to vanish the pig.

We will have to choose a threshold. We want the pig to be displayed only when the pig's speed is below the threshold. If the pig's speed is above the threshold, we don't display the pig.

Teacher writes the code to display the pigs only when their speed is below the threshold of 3.

The student observes and learns.

```

AngryBirdsStage5 ▶ JS Pig.js ▶ Pig ▶ display
1  class Pig extends BaseClass {
2      constructor(x, y){
3          super(x,y,50,50);
4          this.image = loadImage("sprites/enemy.png");
5          this.visibility = 255;
6      }
7
8      display(){
9          console.log(this.body.speed);
10         if(this.body.speed < 3){
11             super.display();
12         }
13         else{
14             //do nothing
15         }
16     }
17 }
18
19
20
21 };
```

What should we do when the speed crosses the threshold?

Just like we added the pig object to the world, we need to remove the pig object from the world when it is hit and crosses a certain speed threshold.

Teacher writes code to remove the object from the world. Teacher runs the code and shows the output.

ESR:

We need to remove the pig.

The student observes and learns.

```

1  class Pig extends BaseClass {
2      constructor(x, y){
3          super(x,y,50,50);
4          this.image = loadImage("sprites/enemy.png");
5          this.visibility = 255;
6      }
7
8      display(){
9          console.log(this.body.speed);
10         if(this.body.speed < 3){
11             super.display();
12         }
13         else{
14             world.remove(world, this.body);
15         }
16     }
17 }
18
19
20
21 };
```

What do you see happen when the bird attacks the pig successfully?

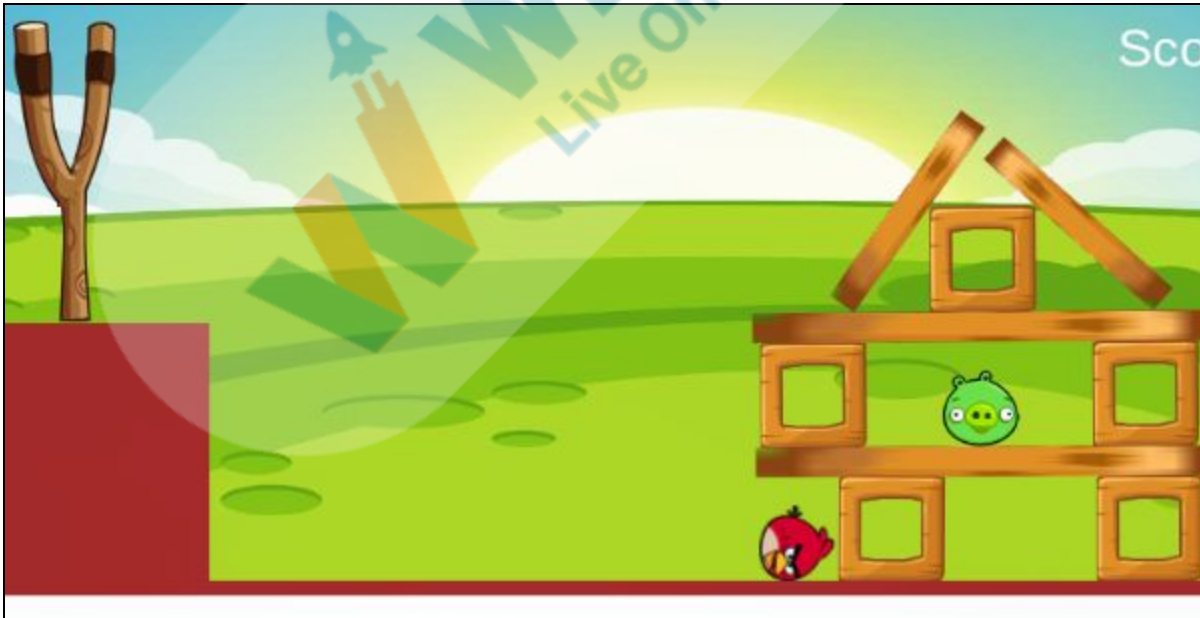
Isn't this vanishing abrupt? Wouldn't we want the pig to vanish more gradually?


ESR:

The pig vanishes from the screen.

ESR:

Yes!



	<p>Any ideas on how to go about doing that?</p> <p>One way to do it is using tint() function in p5.</p> <p><i>Teacher opens tint() documentation. (Teacher Activity 2).</i></p>	<p>ESR: varied</p>
		
	<p>By changing the number in the tint's argument, we can create a vanishing effect...Tint adds a transparent layer to the image.</p> <p>Why don't you try doing this? I will guide you.</p>	<p>ESR: Sure.</p>
<p>Teacher Stops Screen Share</p>		
	<p>Now it's your turn. Please share your screen with me.</p>	

- Ask the Student to press ESC key to come back to panel
- Guide Student to start Screen Share
- Teacher gets into Fullscreen

ACTIVITY

- Add a vanishing effect to the pig.
- Use keyboard space pressed event to attach the bird back to the sling for multiple attacks.

Step 3: Student-Led Activity (15 min)

Guide the student to create a property inside the Pig Constructor called `this.Visibility`.

We can tell the computer about how much visible the pig object should be by assigning a number to `this.visibility`.

Colors / transparency is often measured between 0 to 255.

In Programming we use binary digits. Also a unit of memory is 1 byte (8 bits).

0 -> 1 fully OFF byte (00000000)
255 -> 1 fully ON byte (11111111)

So, what would `this.Visibility = 255` imply?

Student writes the code.

ESR:

It should imply that the pig is fully visible.

```

AngryBirdsStage5 ▶ JS Pig.js ▶ Pig ▶ display
1  class Pig extends BaseClass {
2      constructor(x, y){
3          super(x,y,50,50);
4          this.image = loadImage("sprites/enemy.png");
5          this.Visiblity = 255;
6      }
7
8      display(){
9          console.log(this.body.speed);
10         if(this.body.speed < 3){
11             super.display();
12         }
13         else{
14             World.remove(world, this.body);
15         }
16     }
17 }
18
19
20
21 };
```

We want to give an appearance of the object slowly being removed. After removing the object from the world, let's replace the object with the image of the object at the same position. We will make the image slowly disappear .

Teacher opens the link and explores the tint() from Teacher Activity 2

Also let's give a tint to the image. The image will be fully visible if the tint() function's 2nd argument is this.Visiblity.

The student writes code to replace the pig object with the image after it has been attacked.

Student opens link form Student Activity 2 and follows along.

<pre> 1 class Pig extends BaseClass { 2 constructor(x, y){ 3 super(x,y,50, any 4 this.image = loadImage("sprites/enemy.png"); 5 this.Visiblity = 255; 6 } 7 8 display(){ 9 console.log(this.body.speed); 10 if(this.body.speed < 3){ 11 super.display(); 12 } 13 else{ 14 World.remove(world, this.body); 15 tint(255,this.Visiblity); 16 image(this.image, this.body.position.x, this.body.position.y, 50, 50); 17 } 18 } 19 } 20 21 22 23 };</pre>	<p>display() function is called in every frame. Let's reduce the visibility by 5 each time and see what happens.</p>	<p><i>The student reduces the visibility of the pig class by 5 each time.</i></p> <p><i>Student runs the code and sees the screen getting stuck after the bird successfully attacks one of the pigs.</i></p>
---	--	--

```

1  class Pig extends BaseClass {
2      constructor(x, y){
3          super(x,y,50,50);
4          this.image = loadImage("sprites/enemy.png");
5          this.Visibility = 255;
6      }
7
8      display(){
9          console.log(this.body.speed);
10         if(this.body.speed < 3){
11             super.display();
12         }
13         else{
14             World.remove(world, this.body);
15             //...
16             this.Visibility = this.Visibility - 5;
17             tint(255,this.Visibility);
18             image(this.image, this.body.position.x, this.body.position.y, 50, 50);
19             //...
20         }
21     }
22 }
23
24
25
26 };
```



The screen is getting stuck because the tint visibility setting is being used by all the objects once activated.

Do you remember how to add a setting and then preserve your previous setting?

ESR:

Yes push() and pop().

Student writes the code and runs it to get back the settings for other draw() events.

	<p>Just like we had events for mouse triggers like mousePressed, mouseDragged etc,. The respective functions get called when a mouse event is triggered.</p> <p>Similarly we have a keyPressed event. When a key is pressed, the function gets called.</p> <p>Each key is identified by a 'keyCode' - Numerical value which we call ASCII value. Computer understands which key is pressed by associating this numerical value with the key on the keyboard. You can look at the value of any key by looking at the ASCII chart. (refer Teacher Activity 3)</p> <p>Let's use the key "SPACE" to attach the bird back to the slingshot. What is the value of SPACE?</p>	<p><i>The student looks into the ASCII table to find the value of space as 32.</i> <i>Student Activity 3</i></p>
--	--	--

Code	Character	Code	Character	Code	Character	Code	Character	Code	Character
0	NUL	26	SUB	52	4	78	N	104	h
1	SOH	27	Escape	53	5	79	O	105	i
2	STX	28	FS	54	6	80	P	106	j
3	ETX	29	GS	55	7	81	Q	107	k
4	EOT	30	RS	56	8	82	R	108	l
5	ENQ	31	US	57	9	83	S	109	m
6	ACK	32	(Space)	58	:	84	T	110	n
7	BEL	33	!	59	;	85	U	111	o
8	Backspace	34	"	60	<	86	V	112	p
9	HTab	35	#	61	=	87	W	113	q
10	Line Feed	36	\$	62	>	88	X	114	r
11	VTAB	37	%	63	?	89	Y	115	s
12	Form Feed	38	&	64	@	90	Z	116	t
13	CR	39	'	65	A	91	[117	u
14	SO	40	(66	B	92	\	118	v
15	SI	41)	67	C	93]	119	w
16	DLE	42	*	68	D	94	^	120	x
17	DC1	43	+	69	E	95	_	121	y
18	DC2	44	,	70	F	96	`	122	z
19	DC3	45	-	71	G	97	a	123	{
20	DC4	46	.	72	H	98	b	124	
21	NAK	47	/	73	I	99	c	125	}
22	SYN	48	0	74	J	100	d	126	~
23	ETB	49	1	75	K	101	e	127	DEL
24	CAN	50	2	76	L	102	f		
25	EM	51	3	77	M	103	g		

Guide the student to write the code which is triggered when a Space key is pressed.

Student writes the code.


```

AngryBirdsStage5 ▸ JS sketch.js ▸ keyPressed
66   log3.display();
67
68   box5.display();
69   log4.display();
70   log5.display();
71
72   bird.display();
73   platform.display();
74   //log6.display();
75   slingshot.display();
76 }
77
78 function mouseDragged(){
79   Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
80 }
81
82
83 function mouseReleased(){
84   slingshot.fly();
85 }
86
87 function keyPressed(){
88   if(keyCode === 32){
89     //write code here
90   }
91 }

```

We need to write a function inside slingshot.js to attach the bird back to the sling.
(refer Teacher Activity 4)
We need to call this function in our code which is triggered by the press of a SPACE button.

Student writes the code and runs it.

(Student can refer Student Activity 4 for pressed function)

```
AngryBirdsStage5 ▶ JS sketch.js ▶ keyPressed
66   log3.display();
67
68   box5.display();
69   log4.display();
70   log5.display();
71
72   bird.display();
73   platform.display();
74   //log6.display();
75   slingshot.display();
76 }
77
78 function mouseDragged(){
79   Matter.Body.setPosition(bird.body, {x: mouseX , y: mouseY});
80 }
81
82
83 function mouseReleased(){
84   slingshot.fly();
85 }
86
87 function keyPressed(){
88   if(keyCode === 32){
89     slingshot.attach(bird.body);
90   }
91 }
```

```

JS Bird.js JS Slingshot.js x JS sketch.js JS Pig.js
AngryBirdsStage5 JS Slingshot.js SlingShot
5 pointB: pointB,
6 stiffness: 0.04,
7 length: 10
8 }
9 this.sling1 = loadImage('sprites/sling1.png');
10 this.sling2 = loadImage('sprites/sling2.png');
11 this.sling3 = loadImage('sprites/sling3.png');
12 this.pointB = pointB
13 this.sling = Constraint.create(options);
14 World.add(world, this.sling);
15 }
16 attach(body){
17   this.sling.bodyA = body;
18 }
19
20 fly(){
21   this.sling.bodyA = null;
22 }
23
24 display(){
25   image(this.sling1,200,20);
26   image(this.sling2,170,20);
27   if(this.sling.bodyA){
28     var pointA = this.sling.bodyA.position;
29     var pointB = this.pointB;
30     push();
31
32     stroke(48,22,8);
33     if(pointA.x < 220) {
34       strokeWeight(7);
35       line(pointA.x - 20, pointA.y, pointB.x -10, pointB.y);
36       line(pointA.x - 20, pointA.y, pointB.x + 30, pointB.y - 3);
37       image(this.sling3,pointA.x -30, pointA.y -10,15,30);
38     }
39   }

```

Wow! Now we have as many shots at the pigs as we want. -

Teacher Guides Student to Stop Screen Share




FEEDBACK

- Encourage the student to make reflection notes in markdown format.
- Complement the student for her/his effort in the class.
- Review the content of the lesson.

Step 4:
Wrap-Up
(5 mins)

Great work! How are you feeling?

ESR:
varied

	Can we quickly capture our learning from today?	ESR: We learned about: <ul style="list-style-type: none"> - ASCII values - Keyboard events - Removing a body from the world - Adding transparency to images
	<p>You get a hats off.</p> <p>Awesome! That's a lot of learning!</p> <p>You can spend some time experimenting with these before our next class.</p>	<p><i>Make sure you have given at least 2 Hats Off during the class for:</i></p> <div>    </div>
Project Overview	TOWER SIEGE – 2 Goal of the Project: <p>Today you have learnt to use the vanishing effect for a pig and use keyboard events to attach the bird back to the sling in the Angry Birds game.</p> <p>In this project, you will have to practice and apply what you have learnt in the class and continue adding more functionality to the Tower Siege game.</p>	<p><i>Student engages engages with the teacher over the project.</i></p>

	<p>Story:</p> <p>In the game design competition in your school, you are asked to make a game related to knocking down objects.</p> <p>Create a Tower Siege Game where your friends can throw a rock at a group of stacked objects and crash them and they disappear.</p> <p>I am very excited to see your project solution and I know you will do really well.</p> <p>Bye Bye!</p>	
<p style="text-align: center;">Teacher Clicks</p> <p style="text-align: center;">✕ End Class</p>		
Additional Activities	<p><i>Encourage the student to write reflection notes in their reflection journal using markdown.</i></p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> • What happened today? <ul style="list-style-type: none"> - Describe what happened - Code I wrote • How did I feel after the class? • What have I learned about programming and developing games? • What aspects of the class helped me? What did I find difficult? 	<p><i>Student uses the markdown editor to write her/his reflection as a reflection journal.</i></p>

Activity	Activity Name	Links
Teacher Activity 1	Angry Birds Stage 5	https://github.com/whitehatjr/AngryBirdsStage5
Teacher Activity 2	tint() function documentation	https://p5js.org/reference/#/p5/tint
Teacher Activity 3	ASCII value chart	https://www.chegg.com/homework-help/look-ascii-chart-appendix-determine-codes-letter-first-name-a-chapter-1-problem-3e-solution-9780133985078-exc
Teacher Activity 4	keyPressed() function documentation	https://p5js.org/reference/#/p5/keyPressed
Teacher Activity 5	Reference Link	https://github.com/whitehatjr/AngryBirdsStage6
Student Activity 1	Angry Birds Stage 5	https://github.com/whitehatjr/AngryBirdsStage5
Student Activity 2	tint() function documentation	https://p5js.org/reference/#/p5/tint
Student Activity 3	ASCII value chart	https://www.chegg.com/homework-help/look-ascii-chart-appendix-determine-codes-letter-first-name-a-chapter-1-problem-3e-solution-9780133985078-exc
Student Activity 4	keyPressed() function documentation	https://p5js.org/reference/#/p5/keyPressed