




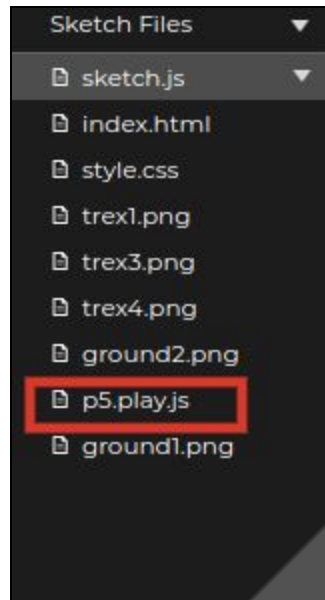
| Topic | TREX AND THE INFINITE GAME WORLD | |
|--|--|--|
| Class Description | Students design a dinosaur for the Trex game with jump and run animations. Students learn to create an infinite game world in the limited screen space available to the players. | |
| Class | PRO-C9 | |
| Class time | 45 mins | |
| Goal | <ul style="list-style-type: none"> • Build a dinosaur with jump and run animations. • Learn to scale the images in the game. • Learn to log messages / outputs from the program into the console for testing purposes. • Create an infinitely scrolling ground for the dinosaur to run on. | |
| Resources Required | <ul style="list-style-type: none"> • Teacher Resources <ul style="list-style-type: none"> ○ p5 login ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen • Student Resources <ul style="list-style-type: none"> ○ p5 login ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen | |
| Class structure | Warm Up - Slide show option Teacher-Led Activity Student-Led Activity Wrap Up - Slide show option | 15 Mins 8 Mins 30 Mins 5 Mins |
| WARM UP SESSION - 15mins | | |
| <div>  </div> <p>Teacher starts slideshow from slides 1 to 20 Refer to speaker notes and follow the instructions on each slide.</p> | | |
| Activity details | | Solution/Guidelines |

| <p>Hi, so good to see you again! How have you been? So this is going to be the first class of our infinite runner game. Are you excited to learn something new?</p> <p>Click on the slide show tab and present the slides.</p> <p>Run the presentation from slide 1 to slide 6.</p> <p>Following are the warm up session deliverables:</p> <ul style="list-style-type: none"> • Connecting students to the previous class. • Explaining Animation with real life Scenarios. | <p>ESR: Thanks, yes I am excited about it.</p> |
|---|---|
| <p style="text-align: center;">QnA Session</p> | |
| Question | Answer |
| <p>Which of the following lines of code can be used to add gravity?</p> <p>A. <code>trex.velocityY = trex.velocityY + 0.5;</code> B. <code>trex.velocityY = trex.velocityY - 0.5;</code> C. <code>trex.velocity = trex.velocityY + 0.5;</code> D. <code>trex.velocity = trex.velocityY - 0.5;</code></p> | <p>A</p> |
| <p>To move the ding to the right, which of the following instructions can be used?</p>  <p>A. <code>ding.x = ding.x + 2</code> B. <code>ding.y = ding.y + 2</code> C. <code>ding.x = ding.x - 2</code></p> | <p>A</p> |

| | | |
|---|--|--|
| D. $\text{ding.y} = \text{ding.y} - 2$ | | |
| Continue the warm up session | | |
| Activity details | | Solution/Guidelines |
| Run the presentation from slide 7 to slide 20 to set the problem statement. Following are the warm up session deliverables: <ul style="list-style-type: none"> • Introduce students to the Animation concept. • Steps to write and run the code. | | Narrate the story by using hand gestures and voice modulation methods to bring in more interest in students. |
| Teacher ends slideshow  | | |
| TEACHER-LED ACTIVITY - 8mins | | |
| Teacher Initiates Screen Share | | |
| <u>CHALLENGE</u> <ul style="list-style-type: none"> • Use of logging into the console for testing purpose • Trick the player to perceive an infinitely scrolling ground. | | |
| Step 2: Teacher-led Activity (10 min) | <p>Great! Let's open and look at the activity for the class(<u>Teacher Activity 2</u>)</p> <p>Check under the files and folders tab, I have uploaded different pictures of the T rex dinosaur.</p> <p>We will use them to create the animation.</p> <p>You will also observe that there is a new file called p5.play.js.</p> | - |

This is a game library which code.org was internally using to allow you to create games!

We have added the library in p5 now and we can create games similar to how we were creating in code.org.



```

< index.html • Saved: 1 minute ago
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.8.0/p5.js"></script>
5     <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.8.0/addons/p5.dom.min.js">
6     <script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.8.0/addons/p5.sound.min.js">
7     <script src="./p5.play.js"></script>
8     <link rel="stylesheet" type="text/css" href="style.css">
9     <meta charset="utf-8" />
10
11   </head>
12   <body>
13     <script src="sketch.js"></script>
14   </body>
15 </html>
16
  
```

| | | |
|--|--|---|
| | <p>Creating animation is similar to adding images.</p> <p>p5 has something called <code>loadAnimation</code> and <code>addAnimation</code> to add animations to the sprite.</p> <p>We use loadAnimation to load all the images for the animation and then use addAnimation to add the animation to the sprite.</p> | |
| | <p>Let's code and create a sprite somewhere near the ground. Let's give a name to our sprite and store it in a variable.</p> <p><i>Ask the student to give inputs while you are writing the code.</i></p> <p><i>Teacher runs the code to check the output.</i></p> | <p>ESR: <i>Student asks the teacher to use <code>createSprite()</code> to create a sprite and give it specific positions, width and height.</i></p> <p><i>The student uses <code>drawSprites()</code> to draw the sprite on the screen.</i></p> |

```

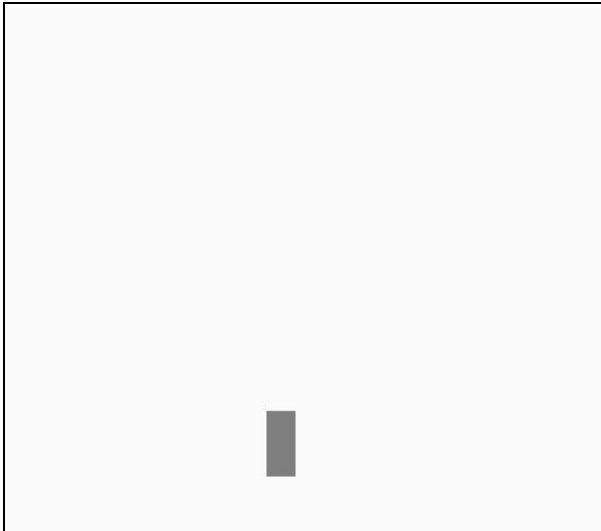
var trex

function setup(){
  createCanvas(600,200)

  //create a trex sprite
  trex = createSprite(50,160,20,50);
}

function draw(){
  drawSprites();
}
  
```

Output:



Let's load our animation for our Trex sprite in function preload and then set animation.

Note:

using **loadAnimation()** load the animations in the preload function. you can load multiple images in a single variable to make it animated.

addAnimation takes two arguments - a label which could be any string and the name of the animation which was loaded.

Student observes how to load and set animation for the Trex.

```

1
2 var trex ,trex_running;
3 function preload(){
4   trex_running = loadAnimation("trex1.png", "trex3.png", "trex4.png");
5 }
6
7
8 function setup(){
9   createCanvas(600,200)
10
11   //create a trex sprite
12   trex = createSprite(50,160,20,50);
13   trex.addAnimation("running", trex_running);
14 }
15
16 function draw(){
17   background("white")
18   drawSprites();
19 }
20

```

| | | |
|--|--|--|
| | <p>We have our running TRex. Now, let's make it jump when we press "space". The dinosaur should jump when we press space and then drop back to the ground. Any ideas on how to do it?</p> <p>Let's recollect some physics and gravity here. What happens to a ball when it bounces off the ground? How does its velocity change?</p> <p><i>Teacher demonstrates the physics behind a falling object with a physical example.</i></p> | <p>ESR: varied</p> <p>ESR: When the ball jumps off the ground, it has some velocity but due to gravity the velocity keeps reducing. It becomes 0 at the top but due to gravity it starts gaining velocity again when it starts falling towards the ground.</p> |
| | <p>Exactly! Let's give the ball some velocity when space is pressed. Do you know how to do that?</p> | <p>ESR: Using if condition</p> |

| | | |
|--|--|--|
| | <p>Which direction should we give the velocity?</p> <p>In our game is upwards velocity positive(+) or negative(-)?</p> <p>Let's give a velocity of -10 to the TRex in vertical direction when space is pressed. Remember to add background() to the game.</p> <p><i>Teacher writes code to give the dinosaur a velocity of -10 when the "space" key is pressed.</i></p> <p><i>Teacher runs the code to see the output.</i></p> | <p>ESR:</p> <p>Upwards?</p> <p>Negative</p> |
|--|--|--|

```

1  var trex, trex_running, edges;
2
3  function preload(){
4    trex_running = loadAnimation("trex1.png","trex3.png","trex4.png");
5  }
6
7  function setup(){
8    createCanvas(600,200);
9    trex = createSprite(50,160,20,50);
10   trex.addAnimation("running", trex_running);
11
12 }
13
14
15 function draw(){
16   background("white");
17
18   if(keyDown("space")){
19     trex.velocityY = -10;
20   }
21
22   drawSprites();
23 }
```


| | | |
|--|--|---|
| | What happened? | The T-Rex jumped off the screen. |
| | <p>This is because there was no gravity! Let's give some gravity. What does gravity do?</p> <p>What will be the effect of gravity on the velocity of the TRex which is going up?</p> <p>Let's add a line in our code which will do that.</p> <p>trex.velocityY = trex.velocityY + 0.5; Since velocity is -ve, the +0.5 will reduce the velocity everytime in the upward direction and bring it to 0. Then, it will make the trex move in the other direction.</p> <p>Teacher adds the line of code which will give the effect of gravity for the dinosaur.</p> <p>Teacher runs the code to see the output.</p> | <p>ESR: It pulls the object towards itself.</p> <p>Gravity will slow down the TRex and reduce its velocity to 0. It will then pull the trex down.</p> |

```

10   trex.addAnimation("running", trex_running);
11   edges = createEdgeSprites();
12 }
13
14
15 function draw(){
16   background("white");
17
18   if(keyDown("space")){
19     trex.velocityY = -10;
20   }
21
22   trex.velocityY = trex.velocityY + 0.5;
23   drawSprites();
24 }

```

| | | |
|--|--|---|
| | <p>Do you see the gravity?</p> <p>What is the problem now?</p> | <p>ESR:</p> <p>There is gravity but the Trex falls off the ground.</p> |
| | <p>Any ideas on how to solve this.</p> <p>Let's create the edges and make the trex collide with the bottom edge so that it does not fall off the ground.</p> | <p>ESR:</p> <p>We can write code to create the edges and make the trex collide with the bottom edge.</p> |

```

7 function setup(){
8   createCanvas(600,200);
9   trex = createSprite(50,160,20,50);
10  trex.addAnimation("running", trex_running);
11  edges = createEdgeSprites();
12 }
13
14
15 function draw(){
16   background("white");
17
18   if(keyDown("space")){
19     trex.velocityY = -10;
20   }
21
22   trex.velocityY = trex.velocityY + 0.5;
23   trex.collide(edges[3])
24   drawSprites();
25 }

```

| | | |
|--|--|---|
| | Looks like we have the jumping and running dinosaur now! And that was quick! | |
| | <p>"Our code doesn't have any comments. Can you quickly help me in writing comments for the code?"</p> | <p>The student helps the teacher in writing comments for different blocks of lines in the code.</p> |

```

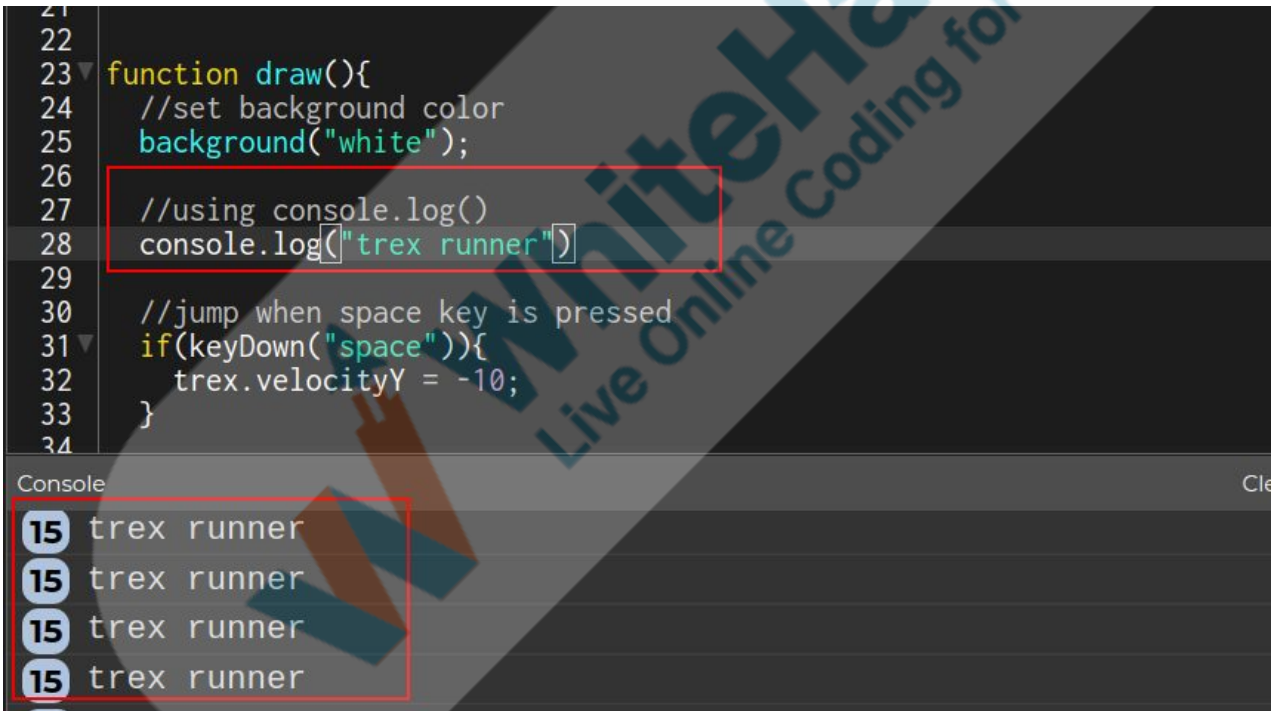
3 function preload(){
4   trex_running = loadAnimation("trex1.png","trex3.png","trex4.png");
5 }
6
7 function setup(){
8   createCanvas(600,200);
9
10  // creating trex
11  trex = createSprite(50,160,20,50);
12  trex.addAnimation("running", trex_running);
13  edges = createEdgeSprites();
14 }
15
16
17 function draw(){
18   //set background color
19   background("white");
20
21   //jump when space key is pressed
22   if(keyDown("space")){
23     trex.velocityY = -10;
24   }
25
26   trex.velocityY = trex.velocityY + 0.5;
27
28   //stop trex from falling down
29   trex.collide(edges[3])
30   drawSprites();
31 }

```

| | | |
|--|---|--|
| | <p>"Good work. Now anyone can read our code and understand easily. Don't you think our dinosaur is too huge compared to the original game?"</p> | ESR: Yes |
| | <p>"There should be something in our Sprite module to help us scale the dinosaur. Help me look for it.</p> <p>Let us look for the sprite object in p5.play documentation.</p> | <p>The student helps the teacher in looking for the <code>sprite.scale</code> in the sprite documentation.</p> |

| | |
|---|---|
| <div><div>Animation</div><div>Camera</div><div>Group</div><div>p5.play</div><div>Sprite</div><div>SpriteSheet</div></div> <div><h2>Sprite</h2><div>Module: p5.play Parent Module: p5.play</div><p>A Sprite is the main building block of p5.play: an element able to move, change position and visibility. A Sprite can have a collider that can overlap with other sprites and mouse interactions.</p><p>To create a Sprite, use <code>createSprite</code>.</p></div> | |
| <div><div><div>• Properties</div><div><ul style="list-style-type: none">◦ _rotation◦ animation◦ collider◦ debug◦ depth◦ friction◦ groups◦ height◦ immovable◦ life◦ mass◦ maxSpeed◦ mouseActive◦ mouseIsOver◦ mouseIsPressed◦ originalHeight◦ originalWidth◦ position◦ previousPosition◦ removed◦ restitution◦ rotateToDirection◦ rotation◦ rotationSpeed◦ scale◦ shapeColor◦ touching◦ velocity◦ visible◦ width</div></div></div> | |
| | <div><div>"Let us use the <code>sprite.scale</code> instruction to scale the dinosaur to half its size. <i>Teacher writes and runs the code.</i> Does the dinosaur look the same size as in the original game?"</div><div>ESR: Yes</div></div> |
| | <div><div>"How can we position the dinosaur to the extreme left?"</div><div>ESR: By setting the x position of the Trex sprite.</div></div> |

| | | |
|--|---|--|
| | <p>"Let us set the x position of the Trex sprite so that it is to the left. What could be the instruction?"</p> <p><i>Teacher writes and runs the code.</i></p> | <p>ESR: trex.x = 50;</p> |
|  <pre> 3 4 function preload(){ 5 trex_running = loadAnimation("trex1.png","trex3.png","trex4.png"); 6 groundImage = loadImage("ground2.png") 7 } 8 9 function setup(){ 10 createCanvas(600,200); 11 12 // creating trex 13 trex = createSprite(50,160,20,50); 14 trex.addAnimation("running", trex_running); 15 edges = createEdgeSprites(); 16 17 //adding scale and position to trex 18 trex.scale = 0.5; 19 trex.x = 50 20 } 21 22 23 function draw(){ 24 //set background color 25 background("white"); 26 27 //jump when space key is pressed 28 if(keyDown("space")){ 29 trex.velocityY = -10; 30 } 31 </pre> | | |
| | <p>"Awesome. Do you see our dinosaur doesn't jump correctly right now? Can you identify the problem?"</p> | <p>ESR: The dinosaur keeps moving up and doesn't fall if we keep the space key pressed.</p> |
| | <p>"Right. This is a problem that we will solve later. For now, I will tell you an important tool using which you can find</p> | |

| | | |
|--|---|--|
| | <p>out what's happening inside the program while the program is running. This will help us find out and correct errors in our program."</p> | |
| | <p>"Our editor has a console window where we can log any message while the program is running. We do this using console.log() instruction.</p> <p>Let us log the name of the game in the console window"</p> <p>Teacher writes and runs the code.</p> | <p><i>Student observes how to use console.log() instruction.</i></p> |
|  <pre> 21 22 23 function draw(){ 24 //set background color 25 background("white"); 26 27 //using console.log() 28 console.log("trex runner"); 29 30 //jump when space key is pressed 31 if(keyDown("space")){ 32 trex.velocityY = -10; 33 } 34 </pre> <p>Console</p> <pre> 15 trex runner 15 trex runner 15 trex runner 15 trex runner </pre> | | |
| | <p>Do you know just like movies, games are made up of many frames. Anything we write inside the function draw() runs for every frame. Anything we write outside the function draw() runs only once.</p> | |

| | | |
|--|---|--|
| | <p>What would happen if we run console.log() inside the draw() function?"</p> <p>"Let us write the console.log() instruction inside the draw() function. Also instead of logging the name of the game, let us log the y position of the trex sprite."</p> <p><i>Teacher writes the code and runs.</i></p> | <p>ESR: We would see the message from console.log() multiple times in the console window for every frame.</p> <p><i>Student observes.</i></p> |
|--|---|--|

```

23 function draw(){
24   //set background color
25   background("white");
26
27   //logging the y position of the trex
28   console.log(trex.y)
29
30   //jump when space key is pressed
31   if(keyDown("space")){
32     trex.velocityY = -10;
33   }
34
35


```


Console

```

160.5
161.5
163
165
167.5
170.5
174
178

```

| | | |
|---|--|--|
| | <p>Teacher runs the code and presses the space key to make the dinosaur jump.</p> <p>"What do you see in the console window? Why do the numbers change?"</p> | <p>ESR: There are numbers as output in the console window. These numbers correspond to the y position of the trex. When the trex jumps, the y position of the trex changes and the numbers in the console change.</p> |
| | <p>"Isn't it amazing to see how the y position of the trex is changing?"</p> <p>console.log() is a powerful tool to help us understand what our program is doing at each stage.</p> <p>We will use it repeatedly in our projects.</p> <p>For now, I want you to get started on moving the dinosaur along an infinite ground space. I will guide you to do that."</p> | <p>ESR: Yes.</p> |
| Teacher Stops Screen Share | | |
| <p>Teacher starts slideshow  :Slides 21-23 (Only 3 slides for this Activity)</p> | | |
| Run the presentation for slide 21-23 to set the student activity context. | | |
| Why don't you share your screen and create a moving ground? | <p><i>The student shares his/her screen, opens the Student Activity, duplicates the code and adds more code to it.</i></p> | |

| | | |
|---|---|---|
| <div>  </div> <p>Teacher ends slideshow</p> | | |
| STUDENT-LED ACTIVITY - 8mins | | |
| <ul style="list-style-type: none"> Ask Student to press ESC key to come back to panel Guide Student to start Screen Share Teacher gets into Fullscreen | | |
| <p>ACTIVITY</p> <ul style="list-style-type: none"> Create a ground sprite and make it move backwards. Reset the ground position when the dinosaur reaches the end of the ground. Use a symmetrical ground image for the ground animation. | | |
| <p>Step 3: Student-Led Activity (20 mins)</p> | <p>Let us create a rectangular sprite called ground. This is where the T rex dinosaur will run. The ground sprite should ideally cover the entire screen.</p> <p>What will be the height and width of such a sprite? What will be its x and y position?</p> <p>Guide the student to create a ground sprite.</p> | <p><i>Student opens the Student Activity 2 and duplicates the code.</i></p> <p>ESR: Height : 20 Width: 400 x: 200 (the sprite should be centred on the screen) y: 180</p> <p><i>Student writes the code to create a ground sprite with the guidance of the teacher.</i></p> |

```

9
10 function setup() {
11   createCanvas(600, 200);
12
13   //create a trex sprite
14   trex = createSprite(50,160,20,50);
15   trex.addAnimation("running", trex_running);
16
17   //adding scale and position to trex
18   trex.scale = 0.5;
19   trex.x = 50
20
21   //create ground sprite
22   ground = createSprite(200,180,400,20);
23
24 }

```

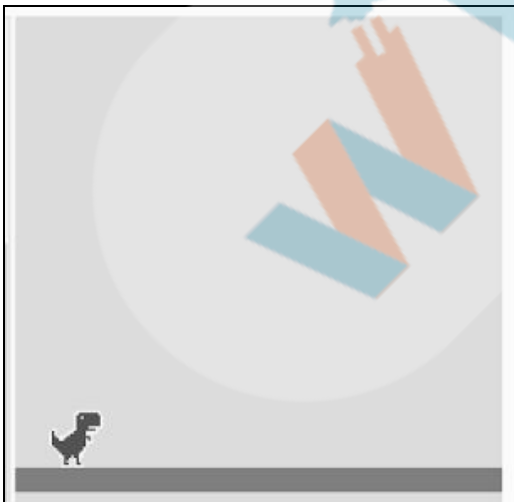
Right now the ground is on the dinosaur, don't you think the dinosaur should collide with the ground instead of the edges?

Let's fix this. Why don't we remove the edges and make the Trex collide with the ground sprite.

ESR: Yes

Student writes the code.

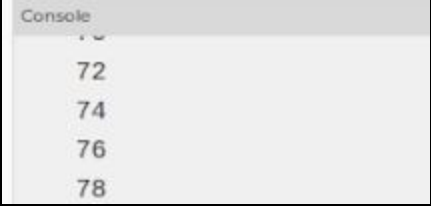
```
20
21 //create ground sprite
22 ground = createSprite(200,180,400,20);
23
24 }
25
26 function draw() {
27   background(220);
28
29
30   //jumping the trex on space key press
31   if(keyDown("space")) {
32     trex.velocityY = -10;
33   }
34
35   trex.velocityY = trex.velocityY + 0.8
36
37
38   //stop trex from falling down
39   trex.collide(ground);
40   drawSprites();
41 }
```

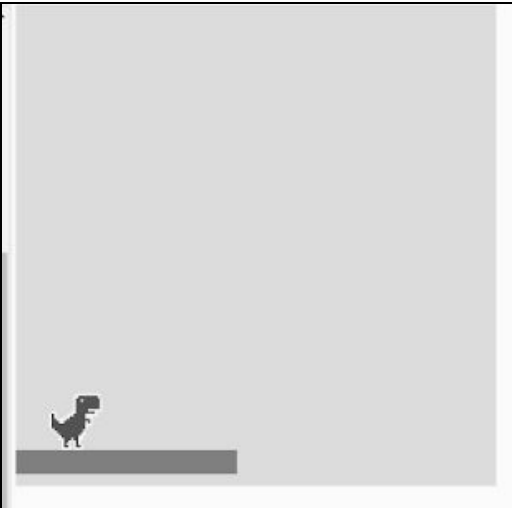


| | | |
|--|---|--|
| | <p>Ok, now let's move the dinosaur. There are two ways to make the player feel that the dinosaur is moving.</p> <p>One would be to give some forward velocity to the trex and the other would be to give some backward velocity to the ground.</p> <p>Let us try each of the ways. Let us first give some forward velocity to the trex and see what happens. Let's keep logging both the trex.x and trex.y in the console window.</p> | <p><i>Student writes code to give some x velocity to the trex.</i></p> <p><i>The student first logs the trex.x and then trex.y in the console window.</i></p> <p><i>Student runs the code and observes both the output and the console window.</i></p> |
|--|---|--|

```

20
27 function draw() {
28   background(220);
29
30   ground.velocityX = -2
31   console.log(ground.x)
32   |
33
34   //jumping the trex on space key press
35   if(keyDown("space")) {
36     trex.velocityY = -10;
37   }
38
39   trex.velocityY = trex.velocityY + 0.8
40
41
42   //stop trex from falling down
43   trex.collide(ground);
44   drawSprites();
45 }
  
```

| | | |
|---|--|--|
| | <p>"What is happening to the trex?"</p> <p>Guide the student to look at the log messages and infer</p> | <p>ESR:</p> <p>The Trex is going outside the screen and then falling.</p> <p>trex.x is increasing constantly.</p> <p>trex.y is increasing exponentially.</p> |
|  | | |
| | <p>"Let's try the other way now. Let us try to give some backward velocity to the ground and see what happens. Also let us log the ground.x now"</p> | <p>The student writes code to give some x velocity to the ground in the negative directions.</p> <p>Student logs the ground.x on the console window.</p> <p>Student runs the code and observes both the output and the console window.</p> |
| | | |

| | | |
|---|---|--|
|  | | |
| | What do you observe? | ESR: Trex reaches the edge of the ground and then falls off. |
| | "Can you think of a way where the trex never falls and the ground keeps scrolling infinitely?" | ESR: varied |
| | <p>"One simple way would be to reset the ground back to the center if $\text{ground.x} < 0$.</p> <p>Once the ground has crossed the screen to the left, we are bringing it back to its original position. This way the ground will always be there.</p> <p>Let's do that and see what happens. How will we do that?"</p> <p>Teacher guides the student to write code to reset the ground to the centre if $\text{ground.x} < 0$</p> | <p>ESR: Using conditional programming</p> <p><i>Student writes and runs the code to reset the ground if $\text{ground.x} < 0$</i></p> |

```

> sketch.js Saved: 25 sec
17
18 //adding scale and position to trex
19 trex.scale = 0.5;
20 trex.x = 50
21
22 //create ground sprite
23 ground = createSprite(200,380,400,20);
24
25 }
26
27 function draw() {
28   background(220);
29
30   ground.velocityX = -2
31   console.log(ground.x)
32
33   if (ground.x<0){
34     ground.x = 200;
35   }
36
37 //jumping the trex on space key press
38 if(keyDown("space")) {
39   trex.velocityY = -10;
40 }
41
42 trex.velocityY = trex.velocityY + 0.8
43
44
  
```

| | | |
|--|--|---|
| | <p>"What do you see now?"</p> <p>"What is the problem right now?"</p> | <p>ESR: The ground keeps resetting itself.</p> <p>"We can see the ground moving and resetting itself."</p> |
| | <p>"How can we solve this problem?"</p> | <p>ESR: varied</p> |
| | <p>Let us join two identical ground images. When half of the ground image goes off the screen, we will reset the ground.</p> <p>This will build an illusion that there is always ground to cover.</p> <p>The player will never feel that the ground has reset itself and will see an infinitely scrolling ground. Let us use the</p> | |

image "ground1" already uploaded in the files directory. It has two ground images - one is red and the other is green - so that you can see what is happening. Later we will use a ground image as in the game.

Where should the ground.x be for this image?

We want the ground to be symmetrically placed on the screen.

So let us keep $\text{ground.x} = \text{ground.width} / 2$;

When $\text{ground.x} < 0$, we will reset the ground back to its center.

ESR:

At half the width of the ground.

Student sets the ground's sprite animation to ground1 and makes changes in the code.

Student runs the code and sees the output.

```

> sketch.js Saved: just now
18 //adding scale and position to trex
19 trex.scale = 0.5;
20 trex.x = 50
21
22 //create ground sprite
23 ground = createSprite(200,380,400,20);
24 ground.addImage("ground",groundImage);
25 ground.x = ground.width / 2;
26 }
27
28 function draw() {
29   background(220);
30
31   ground.velocityX = -2;
32   console.log(ground.x)
33
34   if (ground.x < 0){
35     ground.x = ground.width/2;
36   }
37
38   //jumping the trex on space key press
39   if(keyDown("space")) {
40     trex.velocityY = -10;
41   }
42
43   trex.velocityY = trex.velocityY + 0.8
44
45
  
```

| | | |
|--|---|--|
| | <p>You can see the ground resetting itself. Let us use the actual ground image. There is a ground image uploaded on the animations tab. It contains two ground images joined to each other.</p> | <p><i>Student sets the new animation for the ground and runs the code.</i></p> |
|--|---|--|



```




> sketch.js Saved: just now
18 //adding scale and position to trex
19 trex.scale = 0.5;
20 trex.x = 50
21
22 //create ground sprite
23 ground = createSprite(200,380,400,20);
24 ground.addImage("ground",groundImage);
25 ground.x = ground.width / 2;
26 }
27
28 function draw() {
29   background(220);
30
31   ground.velocityX = -2
32   console.log(ground.x)
33
34   if (ground.x<0){
35     ground.x = ground.width/2;
36   }
37
  
```

Teacher Guides Student to Stop Screen Share

Quiz time - Click on in-class quiz

| Question | Answer |
|--|----------|
| <p>Which instruction is given to make the T rex direct to ground after the space key is pressed?</p> <p>A. Trex.VelocityY=-10 B. Trex.VelocityY=10 C. Trex.VelocityX=10 D. Trex.VelocityX=-10</p> | B |
| <p>What instruction is used to resize the sprite?</p> <p>A. resize B. reshape C. minimize</p> | D |

| | | |
|---|---|---|
| D. scale | | |
| Which instruction can be used to keep a log of what is happening in the program in the real time? A. control.log() B. console.logarithmic() C. console.log() D. set.log() | | C |
| End the quiz panel | | |
| WRAP UP SESSION - 5 Mins | | |
| <div>  Teacher starts slideshow Slide 24-33 </div> | | |
| Activity details | | Solution/Guidelines |
| Run the presentation from slide 24 to slide 33. Following are the warm up session deliverables: <ul style="list-style-type: none"> ● Explain the facts and trivias ● Next class challenge ● Project for the day ● Additional Activity | | Guide the student to develop the project and share with us. |
| <div>  Teacher ends slideshow </div> | | |
| <u>FEEDBACK</u> <ul style="list-style-type: none"> ● Appreciate the student for their efforts in the class. ● Ask the student to make notes for the reflection journal along with the code they wrote in today's class. | | |
| | You get Hats Off for your excellent work! | <i>Make sure you have given at least 2 Hats Off during the class for:</i> |

| | | |
|-------------------------|--|---|
| | <p>Awesome! and remember we have a couple of bugs (problems) in our game. In computer programs, if we don't get the output we expect, we term it as a bug!</p> <p>You might try to fix these bugs on your own. Let us meet in the next class again. Keep your game energy alive.</p> <p>Also, your next class project is going to be about creating your own version of an endless runner game!</p> <p>We will need you to call one of your friends in the class - it is going to be a fun and exciting class where you will brainstorm with your friend about the game and generate new ideas. You can call any of your classmates or anyone who could help you in brainstorming about your game!</p> | <div>Creatively Solved Activities  +10</div> <div>Great Question  +10</div> <div>Strong Concentration  +10</div> |
| | Also, keep writing your notes in the reflection journal. | - |
| Project Overview | Note: This is a tiered project with multiple tasks. All students must do the main task. The main task is very similar to the projects that are already live. | <i>Students engage with the teacher over the project.</i> |

| | | |
|---|---|---|
| | <p>Each tiered project has two or more additional tasks which are optional.</p> <p>TRAVELLING IN SHIP</p> <p>Goal of the Project: In Class 9 you have learned how to scale the images in the game and how to create an infinitely scrolling ground.</p> <p>In this project, you will have to practice and apply what you have learned in the class and Create a background of a sea with a ship in it.</p> <p>Story: Johana is planning to go on a solo trip. But due to covid situation she is not able to go out. Now she is planning to create a game in which she can travel by ship and see the different views.</p> <p>Can you help Meera design the game?</p> <p>I am very excited to see your project solution and I know you will do really well.</p> <p>Bye Bye!</p> | |
| <div> <div>Teacher Clicks</div> <div>✕ End Class</div> </div> | | |
| Additional Activities | <p><i>Challenge the student to create the running ghost animation using Student</i></p> <p><u>Additional Activity Link</u></p> <p><i>Images are uploaded</i></p> | <p><i>Student creates the running ghost animation similar to Teacher</i></p> |

| | | |
|--|--|---|
| | <p><i>Teacher can show the output to the student on what the ghost animation should look like.</i></p> | <p><u>Additional Activity Reference</u> provided</p> |
| | <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? • 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 | Trex Game | http://www.trex-game.skipser.com/ |
| Teacher Activity 2 | Trex Stage 0 | https://editor.p5js.org/whitehatjr/sketches/s42U2SBau |
| Teacher Activity 3 | Sprite documentation | https://molleindustria.github.io/p5.play/docs/classes/Sprite.html |
| Teacher Activity 3 | Reference code (Trex Stage 1) | https://editor.p5js.org/Abhijeet/sketches/bqCypXeR0 |

| | | |
|-----------------------------------|--------------------------------|---|
| | | |
| Student Activity 1 | Trex Game Link | http://www.trex-game.skipser.com/ |
| Student Activity 2 | Trex Stage 0.5 | https://editor.p5js.org/whitehatjr/sketches/MJwoAMYpY |
| Student Additional Activity | Running Ghost (Empty Activity) | https://editor.p5js.org/whitehatjr/sketches/CGeUr7i3h |
| Teacher Additional Activity | Reference | https://editor.p5js.org/whitehatjr/sketches/vBXI1xC5u |
| Teacher Reference visual aid link | Visual aid link | https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/PRO-C9-withcues.html |
| Teacher Reference In-class quiz | In-class quiz | https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/PRO-C9.docx.pdf |