

INFORME DE ACTIVIDAD DESARROLLADA

Nombre	Oliver Bladimir Moriset Rojas	Modalidad	Semanal
Asignatura	Base de Datos II	Turno	Mañana
Docente	Lic. Faviola Gabriela Soliz Tapia	Modulo	1/II/2020
Nro. Actividad	4	Fecha	22/09/2020



DESCRIPCIÓN DE LA ACTIVIDAD

Título de la actividad:

Khan Academy

Url Repositorio en GitHub:

<https://oli9810.github.io/TecnologiaWeb1/>

Descripción breve de la actividad realizada:

- Resolvi las actividades propuestas en khan Academy

Imágenes (captura de pantalla) como evidencia de la actividad realizada:

Desafío: dientes de conejo

Cambia los ojos

Los ojos de Bucky se dibujan como elipses, y los dos últimos números representan su ancho y alto. Crea una variable para almacenar el tamaño del ojo y úsala para cambiar el tamaño de ambos ojos. Una vez que hayas hecho eso, diviértete arrastrando el deslizador para hacer ojos ENORMES y ojos pequeñísimos.

Pista ¿Qué es esto?

```

var  = ;
ellipse(170, 150, , ); // left eye
ellipse(230, 150, , ); // right eye

```

```

1 | var eyes= 15;
2 | var dientes=20;
3 | ellipse(150, 70, 60, 120); // left ear
4 | ellipse(240, 70, 60, 120); // right ear
5 |
6 | ellipse(200, 170, 150, 150); // face
7 |
8 | fill(0, 0, 0);
9 | ellipse(170, 150, eyes, eyes); // left eye
10 | ellipse(230, 150, eyes, eyes); // right eye
11 |
12 | line(150, 200, 250, 200); // mouth
13 |
14 | noFill();
15 | rect(185, 200, 15, dientes); // left tooth
16 | rect(200, 200, 15, dientes); // right tooth
17 |
18 |

```

¡Felicitades!
¡Ganaste 1500 puntos!

Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Funciones

Parámetros de funciones

Desafío: topos en los agujeros

Valores de regreso de una función

Desafío: calculadora

Variables locales y globales

Funciones especiales de ProcessingJS

Repaso: funciones

¡Pruébala! Llama la función `drawMole` para hacer que aparezca un topo.

```

1 var drawMole = function(moleX, moleY) {
2
3
4   noStroke();
5   fill(125, 93, 43);
6   ellipse(moleX, moleY, 60, 60); // face
7   fill(255, 237, 209);
8   ellipse(moleX, moleY+10, 33, 28);
9   fill(0, 0, 0);
10  ellipse(moleX-10, moleY-15, 10, 10); // eyes
11  ellipse(moleX+10, moleY-15, 10, 10);
12  ellipse(moleX, moleY-5, 10, 10); // nose
13  ellipse(moleX, moleY+10, 20, 5); // mouth
14 };
15
16 background(52, 168, 83); // green grass
17 fill(0, 0, 0);
18 ellipse(200, 200, 100, 30); // holes!
19 ellipse(70, 119, 100, 30);
20 ellipse(300, 60, 100, 30);
21 ellipse(297, 350, 100, 30);
22
23 drawMole(200,190);
24 drawMole(70,100);

```

Deshacer **Volver a empezar**

Paso 1/3 **Derivado**

¡Felicidades!
¡Ganaste 1500 puntos!

Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Funciones

Valores de regreso de una función

Desafío: calculadora

Variables locales y globales

Funciones especiales de ProcessingJS

Repaso: funciones

Proyecto: pecera

Siguiente lección
Declaraciones lógicas y co...

- multiply regrese el producto de los dos argumentos;

- divide regrese el cociente entre los dos argumentos;

```

1 var add = function(num1, num2) {
2   return num1 + num2;
3 };
4 var subtract = function(num1, num2) {
5   return num1 - num2;
6 };
7 var multiply = function(num1, num2) {
8   return num1 * num2;
9 };
10 var divide = function(num1, num2) {
11   return num1 / num2;
12 };
13
14 fill(255, 0, 0);
15 text("15 + 3 is " + add(15, 3), 10, 20);
16 text("15 - 3 is " + subtract(15, 3), 10, 50);
17 text("15 * 3 is " + multiply(15, 3), 10, 80);
18 text("15 / 3 is " + divide(15, 3), 10, 110);
19
20 text("8 + 4 is " + add(8, 4), 10, 170);
21 text("8 - 4 is " + subtract(8, 4), 10, 200);
22 text("8 * 4 is " + multiply(8, 4), 10, 230);
23 text("8 / 4 is " + divide(8, 4), 10, 260);

```

Deshacer **Volver a empezar**

Paso 1/1 **Derivado**

¡Felicidades!
¡Ganaste 2100 puntos!

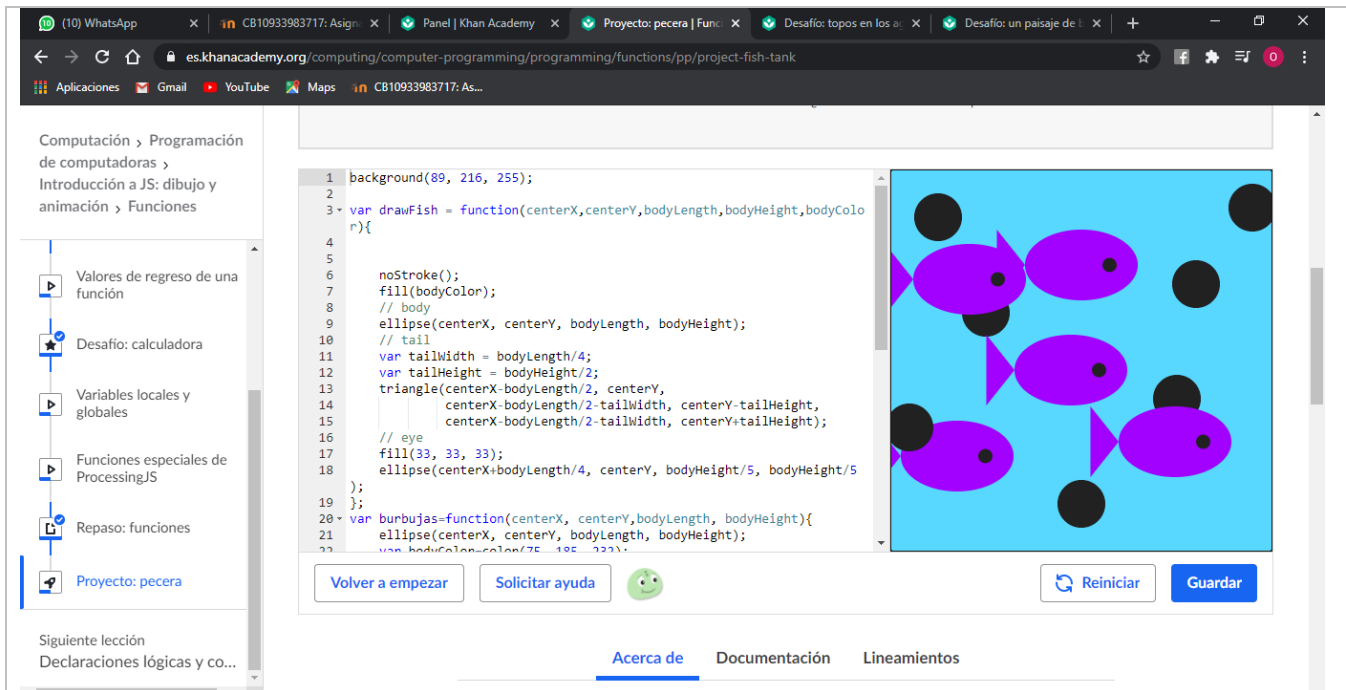
Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Funciones

Proyecto: pecera

```

1 background(89, 216, 255);
2
3 var drawFish = function(centerX, centerY, bodyLength, bodyHeight, bodyColor) {
4
5
6   noStroke();
7   fill(bodyColor);
8   // body
9   ellipse(centerX, centerY, bodyLength, bodyHeight);
10  // tail
11  var tailWidth = bodyLength/4;
12  var tailHeight = bodyHeight/2;
13  triangle(centerX-bodyLength/2, centerY,
14           centerX-bodyLength/2-tailWidth, centerY-tailHeight,
15           centerX-bodyLength/2-tailWidth, centerY+tailHeight);
16  // eye
17  fill(33, 33, 33);
18  ellipse(centerX+bodyLength/4, centerY, bodyHeight/5, bodyHeight/5);
19 };
20
21 var burbujas=function(centerX, centerY, bodyLength, bodyHeight){
22   ellipse(centerX, centerY, bodyLength, bodyHeight);
23   var bodyColor= color(75, 185, 232);

```



Volver a empezar Solicitar ayuda Reiniciar Guardar

Acerca de Documentación Lineamientos

Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Declaraciones lógicas y condicionales (if)

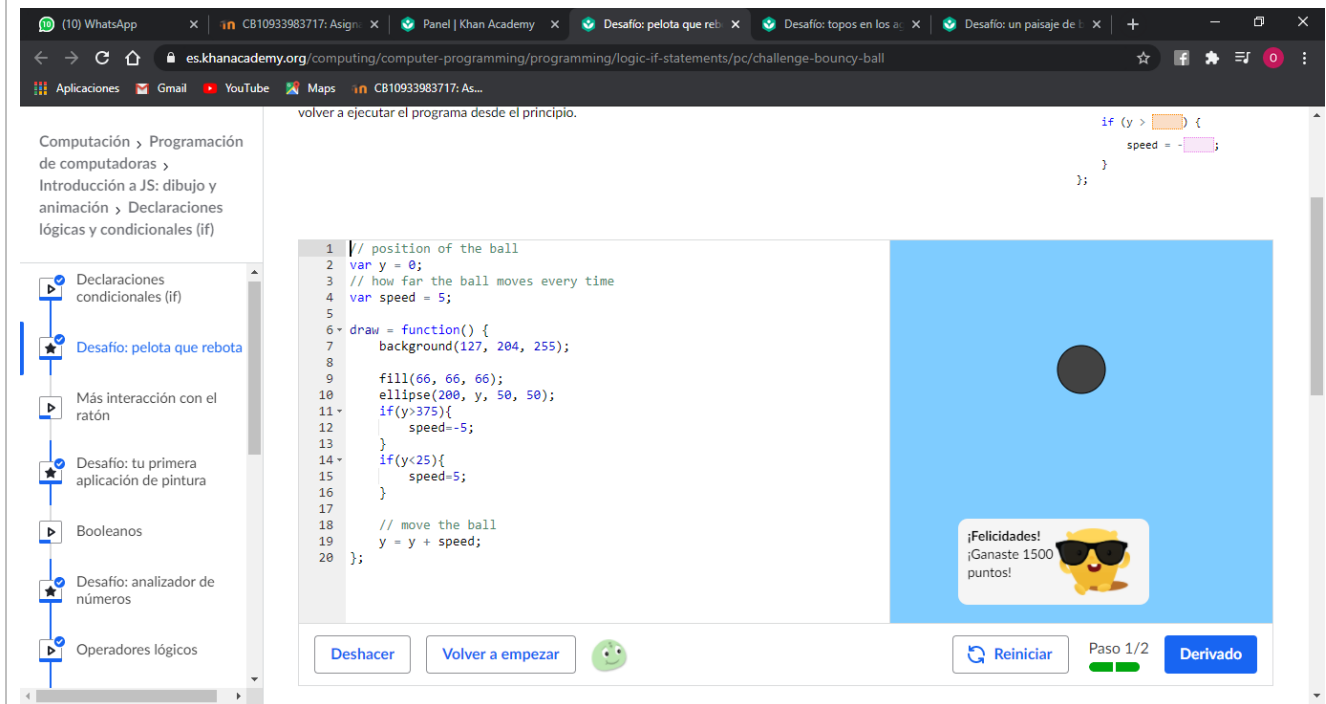
Desafío: pelota que rebota

volver a ejecutar el programa desde el principio.

```

1 // position of the ball
2 var y = 0;
3 // how far the ball moves every time
4 var speed = 5;
5
6 draw = function() {
7   background(127, 204, 255);
8
9   fill(66, 66, 66);
10  ellipse(200, y, 50, 50);
11  if(y>375){
12    speed=-5;
13  }
14  if(y<25){
15    speed=5;
16  }
17
18  // move the ball
19  y = y + speed;
20 };

```



Deshacer Volver a empezar Reiniciar Derivado

Paso 1/2

Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Declaraciones lógicas y condicionales (if)

If/Else - parte 1

Desafío: tarjetas didácticas llamativas

If/Else - parte 2

Repaso: declaraciones lógicas y condicionales (if)

Números aleatorios

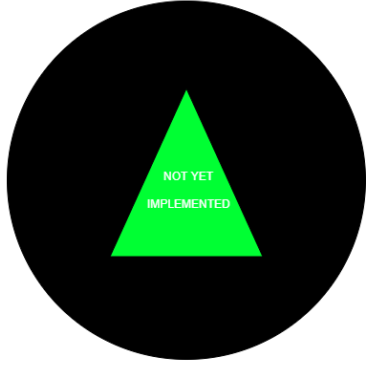
Proyecto: Bola 8 mágica

Siguiente lección
Depurar programas

In the meantime, keep learning! [Prueba una evaluación tú mismo](#)

¿Solicitaste una evaluación por accidente? [Cancela tu solicitud de evaluación](#)

```
1 fill(0, 0, 0);
2 ellipse(200, 200, 375, 375);
3
4
5
6 var answer = floor(random(1, 5));
7 if (answer===1) {
8   fill(60, 0, 255);
9   triangle(200, 104, 280, 280, 120, 280);
10  fill(255, 255, 255);
11  text("NOT YET", 176, 200);
12  text("IMPLEMENTED", 159, 229);
13
14
15
16 }
17
18 if (answer===2) {
19   fill(0, 204, 255);
20   triangle(200, 104, 280, 280, 120, 280);
21   fill(255, 255, 255);
22   text("NOT YET", 176, 200);
23   text("IMPLEMENTED", 159, 229);
24 }
```



Volver a empezar

Solicitar ayuda

Reiniciar

Guardar

Computación > Programación de computadoras > Introducción a JS: dibujo y animación > Hacer bucles

Desafío: una regla de bucles

Más bucles while: Hopper con globos

Desafío: un paisaje de bucles

¡Bucles for! Una nueva clase de bucle

Desafío: papel rayado


Bucles for anidados

Repaso: hacer bucles

Proyecto: construye una

Consejo: ten cuidado con la condición que usas para tu bucle while y asegúrate de incrementar tu variable de manera correcta, de otro modo tu bucle while se ejecutará por siempre.

```
1 fill(17, 0, 255);
2
3 // a handy dandy ruler across the top
4 var x = 0;
5 while(x<400){
6   text(x, x, 10);
7   x+=50;
8 }
9 var y=0;
10 while(y<400){
11   text(y, 10, y);
12   y+=50;
13 }
14
15
16
17
```



Desahacer

Volver a empezar

Paso 1/2

Derivado