CURSO 2023-2024



RETO 16: SEGUNDO VIDEOJUEGO CON SCRATCH: SALTAR

COMPUTACIÓN Y ROBÓTICA 1ºESO

COLEGIO MARISTA LA INMACULADA CALLE SÓCRATES, 8 18002 - GRANADA



Reto	16	



Índice

0. Ubicación en la programación	2
1. Descripción del producto final a entregar	3
2. Aprendemos practicando	4
2.1 Videojuego paso a paso	4
3. Por si quieres seguir ampliando en casa	15



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1ºESO



0. Ubicación en la programación

Título: Reto 16. Segundo videojuego con Scratch: Saltar

Evaluación: Tercera

Temporalidad: 1 sesión (2 horas consecutivas)

Índice de contenidos: ..

Breve resumen de la situación: ...



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1ºESO



1. Descripción del producto final a entregar

En este tema

Los retos se califican según los siguientes criterios:

• Están completos





2. Aprendemos practicando

Vamos a simular el famoso "juego del dinosaurio" cuando el navegador Chrome trabaja sin conexión a internet:

https://dinorunner.com/es

2.1 Videojuego paso a paso

Creamos el suelo. Un objeto formado por un rectángulo negro sobre el que correrá nuestro personaje.



IMPORTANTE. En la zona de edición, ajusta el centro del rectángulo con el centro del objeto. Así la programación del objeto estará siempre vinculada al centro del rectángulo.







Ajustamos el tamaño del personaje principal a 40. Elegimos nuestro gato, por tener dos disfraces que dan sensación de estar el gato en movimiento. Situamos al gato sobre el suelo.



En estas imágenes, he situado al gato en las posiciones **x=-200**, **y=-135**. Estos valores los puede modificar cada creador, según la posición de su suelo. En estos apuntes asumiremos siempre las posiciones (x,y) indicadas.

De esta forma, en la zona de programación, al pulsar bandera verde para comenzar el juego, siempre colocaremos al personaje en esa posición de partida.



Además, en un bloque "por siempre", alternaremos sus dos disfraces cada 0.25 segundos para imitar la sensación de estar corriendo.







Pasemos al salto. Cuando pulsemos la flecha hacia arriba, el persona debe saltar. Y debe hacerlo de una manera realista: al principio del salto, avanza muy distancia. Y al final del salto, recorre menos distancia debido a la gravedad que frena el impulso del salto, hasta detener al personaje.

¿Cómo conseguirlo? En primer lugar creamos el evento "al presionar tecla flecha arriba", junto a otro bloque "por siempre".



Con un bloque condicional "si... si no" vamos a detectar si el personaje está en la posición inicial sobre el suelo. El bloque "posición en y" nos informa de la coordenada vertical del objeto.

🛫 Cóc	digo	Disfraces	() Sonidos	N 🖲	
Movimiento	dar a x el va	alor -200			
Apariencia	sumar a y	10			
Sonido	dar a y el va	alor -135		al hacer clic en 💌 al presionar tecla 🛛 flecha amba 💌	
Eventos				ir a x: -200 y: -135 por siempre	
Control	si toca un b	orde, rebotar		por siempre si posición en y = -135 entonces	
Sensores	fijar estilo d	e rotación a	izquierda-derecha 👻	siguiente disfraz	
Operadores		án an v		esperar 0.25 segundos	
Variables		ón en y			
Mis bloques	direct	ión		Objeto (perso	naje \leftrightarrow x -200 ‡ y -135 Escenario
	Apariencia	a		(O) Mostrar O	Ø Tamaño 40 Dirección 90
2	decir (Hol	al durante	2 segundos		Fondos
				Machila personaje	suelo

Creamos una variable "salto" para almacenar el valor del impulso vertical del personaje.

Iniciamos el valor de "salto" a 0.

Si el personaje se encuentra en la posición vertical y=-135, también forzamos que "salto" sea igual a 0.



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1ºESO



🗶 Có	digo 🥜 Disfraces 🌒 Sonidos		N	
Movimiento	Variables		salto 0	
	Crear una variable			
Apariencia	mi variable		· · · · ·	
Sonido	salto	al hacer clic en 🏴 🔹 🔹 al presionar tecla 🛛 flecha arriba 🔻	· · · · · · ·	
Eventos	dar a mi variable 🔹 el valor 🕕	dar a salto 👻 el valor 0 por siempre		
Control	sumar a mi variable 👻 🚺	ir a x: -200 y: -135 si posición en y = -135	entonces	
Sensores	mostrar variable mi variable 🝷	por siempre dar a salto → el valor 0		
Operadores	esconder variable mi variable 🔹	esperar 0.25 segundos		
Variables	Crear una lista			
	Mis bloques			Escenario
Mis bloques	Crear un bloque		Objeto	↔ x _200 ↓ y _135
			Mostrar 🧿 💋	Tamaño 40 Dirección 90
*		· · · · · · · · · · · · · · · · · · ·		Fondos
		Mochila	personaje suelo	

Si hemos detectado que el persona está sobre el suelo, en la posición vertical de partida, y pulsamos la flecha "hacia arriba", debemos impulsarlo con una velocidad inicial. Por ejemplo, salto = 15.

🛫 Có	digo 🥒 Disfraces 🌒 Sonidos			
Movimiento	Variables		saito 0	
	Crear una variable			
Apariencia	mi variable			
Sonido	salto	dar a salto ▼ el valor 0 por siempre		
Eventos	dar a mi variable 🔹 el valor 🕕	ir a x: -200 y: -135 si posición en y = -135 entonces	2	
Control	sumar a mi variable 👻 1	por siempre	personaje	Escenario
	mostrar variable mi variable 💌		x 200 x 135	
Sensores	occonder veriable mi variable =	si (tecla flecha arriba ▼ presionada?) entonces	,	
Operadores		esperar 0.25 segundos dar a salto ▼ el valor 15 de transferir da dar a salto ▼ el valor 15	× —	Fondos 1
Variables	Crear una lista		personaje suelo	
	Mis bloques	si no se companya de la si no se companya de la seconda		
mis bioques	Crear un bloque			
<u>e</u> *				
		· · · · · <u>· · · · · · · · · · · · · · </u>		
		Mochila		

Si el personaje no está sobre el suelo, significa que está por el aire saltando.

Para simular el efecto de la gravedad, debemos restar alguna cantidad fija al impulso inicial marcado por la variable salto. Por ejemplo, restar 1 a la variables salto en el bloque "si no".

Cuando mayor sea el valor inicial de salto, más alto llegará el personaje. Y cuando más cantidad restemos en el bloque "si no"; antes regresará al suelo.



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales







Nuestro personaje no se mueve porque solo hemos modificado el valor de la variable salto. Necesitamos que esta variable afecte a la coordenada vertical del personaje. Usamos el bloque "sumar a y". Y con esto, nuestro personaje salta de una forma bastante realista.

🖛 Có	digo 🕜 Disfraces 🌒 Sonidos			
Movimiento	Variables		saito 🔜	
Apariencia	Crear una variable	al hacer clic en 📕 🔹 al presionar tecla 🛛 flocha arriba 🔸		
Sonido	salto	dar a salto 👻 el valor 0 por siempre	\$	
Eventos	dar a mi variable 🔹 el valor 🕕	por siempre		
Control	sumar a 🛛 mi variable 👻 🚺	dar a sato • el valor 0 siguiente disfraz si tecla flecha arriba • presionada? entonces	personaje	Escenario
Sensores	mostrar variable mi variable 🔹	esperar 0.25 segundos dar a satto 🕶 el valor 15	x -200 y -60	
Operadores	esconder variable mi variable 🔹		X	Fondos 1
Variables	Crear una lista Mis bloques	sumar a salto 🔹 💽 💽 👘 👘 👘 👘 👘 👘	personaje suelo	
Mis bloques	Crear un bloque			
~				
		Mochila		

Nuestro código para saltar tiene un error sutil. Puede ocurrir, por el tiempo de procesado del ordenador al ejecutar los bloques de programación, que nuestro gato no vuelva a la posición de partida y=-135 tras terminar su salto. Acabando un poco por debajo de la línea del suelo.

Para solventar este error, con un nuevo condicional, detectamos si la posición vertical es inferior al valor y=-135 y situamos al personaje en la posición deseada con un bloque "dar a y el valor".



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1°ESO



🖛 Cód	ligo 🖌 Disfraces 🌒 Sonidos			
Movimiento	sumar a y 10	dar a salto v el valor 0 por siempre	esto O	
Apariencia	dar a y el valor -135	por siempre		
Sonido	si toca un borde, rebotar	siguiente disfraz si ¿tecla (Recha arriba • presionada?) entonces		
Eventos	fijar estilo de rotación a izquierda-derecha 🗸	esperar 0.25 segundos dar a salto - el valor 15	2	
Control			personaje	Escenario
Sensores	posición en y	sumar a salto 🗸 🧃 🧰 a subsection and a subsection of the subsec	x -200 y -135	
Operadores	dirección	sumar a y salto	* -	Fondos 1
Variables	Apariencia	si posición en y < 135 entonces	personaje	
Mis bloques	decir [Hola] durante 2 segundos	dar a y el valor 135		
±.	decir (Hola!			
	pensar Umm durante 2 segundos		(L)	

Pasemos a programar los obstáculos a saltar.

Creamos un nuevo objeto llamado "obstaculo". Recuerda que, por norma general, no ponemos tildes a los nombre de objetos ni de variables.

Este objeto va a tener tres disfraces: tres rectángulos de distinta altura, que haremos aparecer de manera aleatoria durante el juego. Recuerda hacer coincidir el centro de cada rectángulo con el centro del objeto.

Por ejemplo, el rectángulo verde es más bajo que el azul. Y el rectángulo azul es más bajo que el rojo.

🚝 Código	J Disfraces	() Sonidos		
1	Disfraz	tisfraz3 ► → Agrupar Desagrupar Adelante Atrás Al frente Al fondo	salto 0	
disfraz1 28 x 52	Rellenar	Borde D O Copiar Pegar Eliminar		
disfraz2 28 x 69				
3		♦		
disfraz 3 28 x 84	4	т		
	/	0		
			Objeto obstaculo ↔ x 36 ‡ y 28	3 Escenario
			Mostrar O Ø Tamaño 100 Dirección 90	
		$\bigcirc \qquad \bigcirc \qquad = \qquad \bigcirc$		Fondos

Al pulsar bandera verde, debemos esconder el objeto "obstaculo".

Y posteriormente, de manera aleatoria, iremos mostrando uno de los tres rectángulos cada vez que creemos un clon del objeto.



🚝 Có	digo 🚽 Dis	sfraces	() Sonic	los									
Movimiento	siguiente fondo											salto 0	
Apariencia	cambiar tamaño	por 10											
Sonido	fijar tamaño al	100 %											
Eventos						al hac	er cli					.	
Control	sumar al efecto		25			escor	nder						
Sensores	quitar efector or												
Operadores	quitar electos gr												
Variables	mostrar											6-4	
Mis bloques	esconder										\odot	Objeto obstaculo \leftrightarrow x 36 ‡ y	28 Escenario
	ir a capa delar	itera 👻										Mostrar 🧿 🧭 Tamaño 100 Dirección	90
=	ir 1 cap	as hacia de	elante 👻									👰 — 📫	Fondos

En un bloque "por siempre", cada 2 segundos, creamos un objeto "obstaculo". Podemos modificar este tiempo al gusto del creador.

Creamos una variable "crearObstaculo" para saber si se muestra el rectángulo o no con ayuda de un bloque "número aleatorio".

Y creamos una variable "tipoObstaculo" para saber si mostramos el rectángulo verde, azul o rojo, con ayuda de otro bloque "número aleatorio".

🛫 Có	digo	J Disfra	aces	() Sor	nidos																			N			2	
Movimiento	Opera	adores																						saito 0 crearObataculo 0				
	Ο	+							n 🏲																			
Apariencia	0						dar a	crear(Obstac		e	valor	0															
Sonido	0											valor											.					
Eventos	Q	\sim					dar a	tipoOl	bstacul	lo 🔻	el va	alor	0											\$		_		,
Control	núme	ero aleatorio e	ntre 🚺	y 10			esconde	эг																obstaculo		Es	cenario	
		> 50					por sien																	x 36 v	28			
Sensores	7	< 50						ar 🖸	2 se	aundo	s															L		
Operadores	7	= 50											(C						- 🎉 -	-	F	ondos 1	
Variables							dar a		arObst	aculo		el valo	or n	numer	ro alea	atoric	entre	1	у	10				personaje	suelo			
Mis bloques		У							و														\odot	I I Y				
		•																					Q	obstaculo				
+	no																						=					
<u> </u>	unir	manzana	plátano											1		1										(F.	

Por ejemplo: si "crearObstaculo" es mayor que 3, creamos el obstáculo. Esta probabilidad la podemos modificar al gusto del creador del programa.

Si hemos creado el obstáculo, deberemos elegir con un nuevo número aleatorio el tipo de obstáculo (verde sería el disfraz1, azul el disfraz2, y rojo sería el disfraz3).



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1ºESO



🛫 Có	digo 🖌 Disfraces	() Sonidos			
	Variables		dar a crearObstaculo - el valor 0	ento O	
	Crear una variable		dar a tipoObstaculo 🕶 el valor 🕕		
Apariencia	CrearObstaculo		esconder		
Sonido	mi variable				
Eventos	Salto		por siempre	\$	_
	tipoObstaculo		esperar 2 segundos estas e		Escenario
Control	dar a 🛛 crearObstaculo 👻 e	el valor 0	dar a crearObstaculo 🝷 el valor número aleatorio entre 🚺 y 10	obstaculo	
Sensores	sumar a crearObstaculo 👻	1		x 36 y 28	
Operadores	mostrar variable crearObsta	aculo 🔫	si crearObstaculo > 3 Pentonces		Fondos 1
Variables	esconder variable crearObs	staculo 💌	dar a tipoObstaculo 🔻 el valor número aleatorio entre 1 y 3	personaje suelo	
	Croar una lista		si no		
MIS Dioques	Mis bloques			obstaculo	
	Crear un bloque				
=					

Con nuevos bloques condicionales, seleccionamos un disfraz del objeto "obstáculo". Y al final creamos un clon de ese objeto.

No olvidar esconder el objeto, al final de todo el bloque de programación, para evitar que quede visible en caso de que la probabilidad del número aleatorio indique que el objeto no puede mostrarse en pantalla.

😂 Código 🛃 Disfraces 📣 Sonidos	
cambiar fondo a fondo 1 🔹	eresrObetsculo
Movimiento dar a crearObstaculo • el valor número aleatorio entre 1 y 10	. tpoObstacuio 2
Apariencia si crearObstaculo > 3 entonces	
dar a tipoObstaculo - el valor número aleatorio entre 1 y 3	
Eventos fijar tamaño al 100 %	
Control	obstaculo
sumar al efecto color 25 si tipoObstaculo = 2 entonces	× 36 × 28
Sensores dar al efecto color • el valor 0	× 30 y 20
Operadores Sino	Fondos
quitar efectos gráficos	personaje suelo
Variables	
Mis bloques mostrar crear don de mi mismo •	
esconder sino	obstaculo

¿Qué hace el clon de objeto "obstaculo"?

Aparecer por la derecha, a la altura del suelo, y avanzar hacia el personaje.

La posición (x,y) del obstáculo le elegimos libremente, para situar a los rectángulos sobre el suelo.

En estos apuntes hemos elegido x=240, y=-139.

No olvidemos el bloque "mostrar" si hemos iniciado el clon.



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales

Computación y Robótica 1ºESO





Un nuevo bloque por siempre va a controlar el movimiento de los rectángulos.

Con el bloque "sumar a x" podemos ajustar la velocidad a la que se desplazan los rectángulos.

😴 Código 💕 Disfraces 👍 Sonidos		
Movimiento ra posición alestoria -	salto 0 crearObstaculo 2	
Apariencia Rax 36 y. 28 al comenzar como clon	tipoObstaculo 2	
Sonido destizar en 1 segs a posición aleatonia •	1	
Eventos desizar en 1 segs a x 36 y. 29		
control apuntar en dirección 🗐 número alisatorio entre 1 y 10 por siempre		
Sensores apuntar hada puntero del ratón - Itonces sumar a x -5		
Operadores sumar ax 10 el valor número aleatorio entre 1 y 3		
Variables dar a x et valor 36 1 entionces de la construcción de		
Mis bloques	Objeto obstaculo ↔ x 36 ‡ y 28	Escenario
	Mostrar 🧿 🧭 Tamaño 100 Dirección 90	
	/ 👔 — 📑	Fondos

Ya hemos conseguido que los rectángulos, aleatoriamente cada 2 segundos y aleatoriamente entre sus tres disfraces, aparezcan por la derecha y se desplacen por la izquierda.

Ahora debemos conseguir, en primer lugar, si el obstáculo toca al personaje.

Para ello, usamos un condicional "si... si no". Y empleamos el bloque de sensores "¿tocando personaje?".



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales

Computación y Robótica 1ºESO



🖛 Cđ	digo 🥒 Disfraces 🜒 Sonidos	N 🕒			×
Movimiento	Sensores				
Anarianaia	¿tocando puntero del ratón ▼ ?	al comenzar como cion tra y 240 y 120			
Apanencia	¿tocando el color ?	mostrar			
Sonido	distancia a puntero del ratón - aleatorio e	ntre 1 y 10 r z por siempre			
Eventos	preguntar ¿Cómo te llamas? y esperar	sumar a x -5			
Sensores	Obstar respuesta valor número aleatorio	entre 1 y 3 si ¿tocando personaje • ? entonces			
Operadores	poObstaculo = 1 entonces ¿tecta espacio - presionada?	n an			
Variables	Ist ¿ratón presionado?				
Mis bloques	posición x del ratón tipoObstanuto 2 entonces posición y del ratón	Objeto obstaculo	↔ x 36 ‡ y 28	Escenar	rio
	r disfraz a disfraz2 🔹	Mostrar 🔘 Ø	Tamaño 100 Dirección 90		
=	fijar modo de arrastre a arrastrable 🔹			Fondos	s

Si un clon del objeto "obstáculo" toca al personaje, debe parar su movimiento y deben detenerse todos los programas.

Con el bloque "deslizar" podemos fijar de inmediato (0 segundos) la posición del rectángulo que impacta con el personaje. Y con "detener todos" pausamos todos los programas.

🚝 Có	digo	J Disfraces	Sonidos						R.
Movimiento	si no				al comenzar como clon	salto 0 crearObstaculo 8			
Apariencia	espera	ar hasta que			irax: 240 y: -129	tipoObstaculo 2			
Sonido	repetir	r hasta que	, 		mostrar				
Eventos	E	ervalor número			por siempre				
Control	detend	er todos 🕶 mere		1 y 3	si ¿tocando personaje • ? entonces				
Sensores		ento			deslizar en 0 segs a x: posición en x y: posición en y				
Operadores	Crear of	az1 👻			detener todos 👻				
Mis bloques	s elimina	ar este clon	tonces			Objeto obstaculo \leftrightarrow x -137 ‡ y -32	5	Escenari	io
	a dis Senso	sfraz2 🔻				Mostrar 💿 💋 Tamaño 100 Dirección 90			
1	étoc a dis	ando puntero del ra	atón 🔹 ?					Fondos	

Si el obstáculo no toca al personaje, significa que el gato ha saltado correctamente y sumamos un punto en el juego. Para ello, creamos una nueva variable "puntos", que podemos iniciar a 0 en el código del objeto "personaje".

Para añadir 1 a la puntuación, podemos elegir un valor de la posición x del obstáculo que esté a la izquierda de la posición horizontal del personaje. En las siguientes imágenes hemos asumido este valor como x=-230.



Segundo videojuego con Scratch: Saltar Saberes aplicados a contextos reales Computación y Robótica 1ºESO



🖛 Có	digo 🦪 Disfraces	() Sonidos				
Movimiento	Variables				al comenzar como clon	
	Crear una variable				ir a x: 240 y: -129	
Apariencia	crearObstaculo				mostrar	
Sonido			patorio entre	10		
Eventos						
Control	tipoObstaculo				obstaculo	Escenario
Sensores	a tipoObstaculo • dar a crearObstaculo •	el valor número a	leatorio entre 1 y	3	deslizar en 0 seos a x posición en x y posición en y	
	sumar a crearObstaculo	onc	es i i i i		delener todos	Fondos
Operadores						1
Variables	mostrar variable crearOt	ostaculo 🝷	h		si posición en x = -230 entonces e de d	
Mis bloques	esconder variable crear	Obstaculo -	nces		sumar a puntos 🔹 🚹 📩 su	
	ami Crear una lista disi	iraz2 •				
<u></u>	Mis bloques					
	Crear un bloque					

Finalmente, fíjate que los obstáculos quedan almacenados por detrás del personaje.

Si llegan al final, deben desaparecer. Por lo que podemos detectar su posición para ocultarlos.

🚝 Cóc	ligo 🥒 Disfraces 🌒 Sonidos			
Movimiento	cambiar tamaño por 10 delor número al pator 3 entonces	to entre 1 y 10 por siempre	asito O crearObetacuio O tipoObstacuio O puntos O	
Apariencia	tipoObstaculo マ el valor número aleator	vrio entre 1 y 3 si ¿tocando personaje • ? entonces		
Sonido	sumar al efecto color 25 entor es	deslizar en 0 segs a x: posición en x) y: posición en y		
Eventos	dar al efecto color - el valor 0		\$	
Control	quitar efectos gráficos		obstaculo	Escenario
Sensores	tipoObstaculo = 2 entonces		x -170 y -129	
	mostrar disfraz a disfraz2 ▼			Fondos
	esconder	si posición en x = -240 entonces	personaje suelo	1
Variables	ir a capa delantera 🔹	esconder		
Mis bloques	ir 1 capas hacia delante 🔹		obstaculo	
	número 🔻 de disfraz			
-	número 💌 de fondo			





3. Por si quieres seguir ampliando en casa

Podemos mejorar nuestro videojuego de salto con las siguientes opciones (de menor a mayor dificultad):

- 1. Introduce una música que acompañe al salto del personaje.
- 2. Introduce una melodía de "game over", cuando el personaje impacte con un obstáculo.
- 3. Muestra un fondo de "game over " cuando el juego finalice.
- 4. Crea otro tipo de obstáculos, jugando ahora con la anchura en vez de con la altura.

5. Introduce objetos voladores que, si son cazados por el personaje en su salto, generan 5 puntos extras en la puntuación.

Es posible que, para los pasos 4 y 5, debas ajustar los parámetros de salto y gravedad del código de programación.

jÁnimo!