

# Manual Básico de Scratch



Como apuntábamos en el artículo de introducción a la programación para los más jóvenes, SCRATCH es una de las herramientas que nos van a permitir comenzar a utilizar un lenguaje de programación de una manera simple pero eficiente.

Se utiliza la metáfora de “piezas encajables” para animar objetos que se encuentran en la pantalla, con un uso muy sencillo e intuitivo.

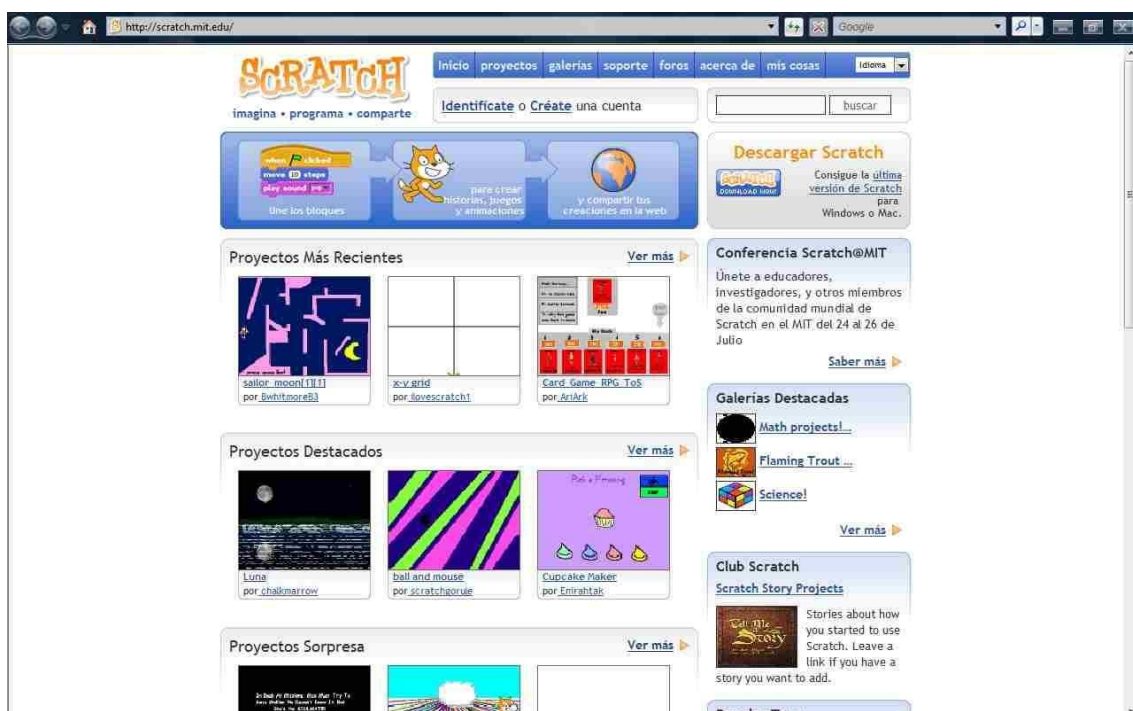
Scratch se utiliza desde un “entorno de desarrollo” que muestra de un sólo golpe de vista todos los elementos necesarios: escenario, objetos y elementos del lenguaje.

Podemos tener tantos escenarios y objetos como deseemos, utilizando aquellos que ya están disponibles con la instalación estándar de la herramienta, o bien creando los nuestros. Este es un factor motivacional más a la hora de trabajar con el alumnado desde edades tempranas.

Los elementos disponibles no son únicamente dibujos, sino también sonidos. Podemos utilizar los que vienen por defecto, añadir sonidos nuevos desde la web del proyecto, o incorporar nuestras propias grabaciones, bien a través de la grabadora incorporada en el entorno, bien a través de cualquier otra herramienta externa.

Prácticamente todo se lleva a cabo arrastrando y soltando elementos con el ratón, y modificando con el teclado únicamente valores numéricos, textos...  
**DESCARGA E INSTALACIÓN DEL PROGRAMA**

Directamente, desde la web del proyecto: <http://scratch.mit.edu/>



En la zona “Descargar Scratch” podemos encontrar un instalador para Windows y Mac. No hay aún paquetes específicos para Linux (indican que están trabajando en ello), pero existen unas instrucciones (en inglés) sobre cómo hacerlo funcionar bajo distintas distribuciones de este sistema en los foros del proyecto:

<http://scratch.mit.edu/forums/viewtopic.php?id=21>

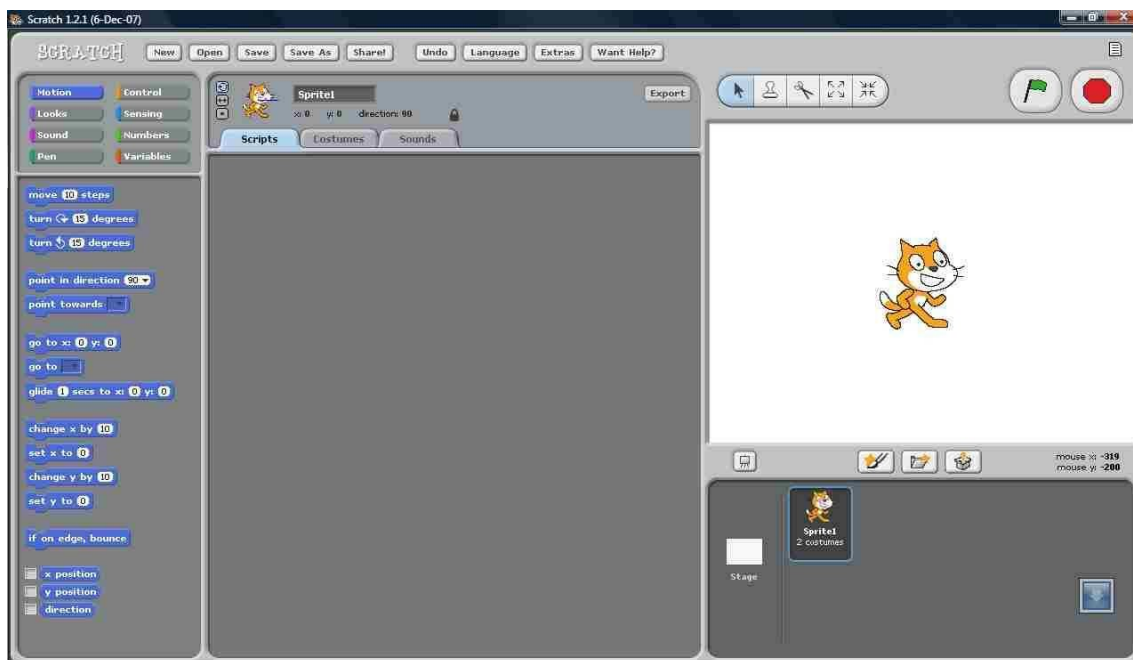
Para los menos familiarizados con el inglés, también existe un tutorial sobre cómo instalarlo en Linux, paso a paso (¡en español!), que se menciona en dicho foro:

<http://eccp.blogspot.com/2007/01/cmo-instalar-scratch-en-linux.html>

Una vez instalado Scratch en nuestro sistema, estamos preparados para utilizarlo por primera vez.

**PRIMERA EJECUCIÓN DE SCRATCH. CAMBIAR EL LENGUAJE POR DEFECTO.**

En la primera ejecución del programa, el entorno se encuentra en inglés:



Para cambiar el idioma, simplemente debemos pulsar sobre el botón "Language", seleccionar "Español"...

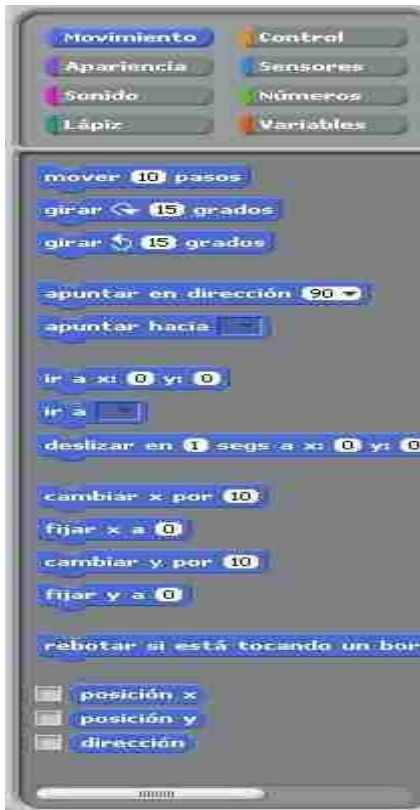


Y ya tenemos nuestro entorno en castellano.  
ELEMENTOS DEL ENTORNO

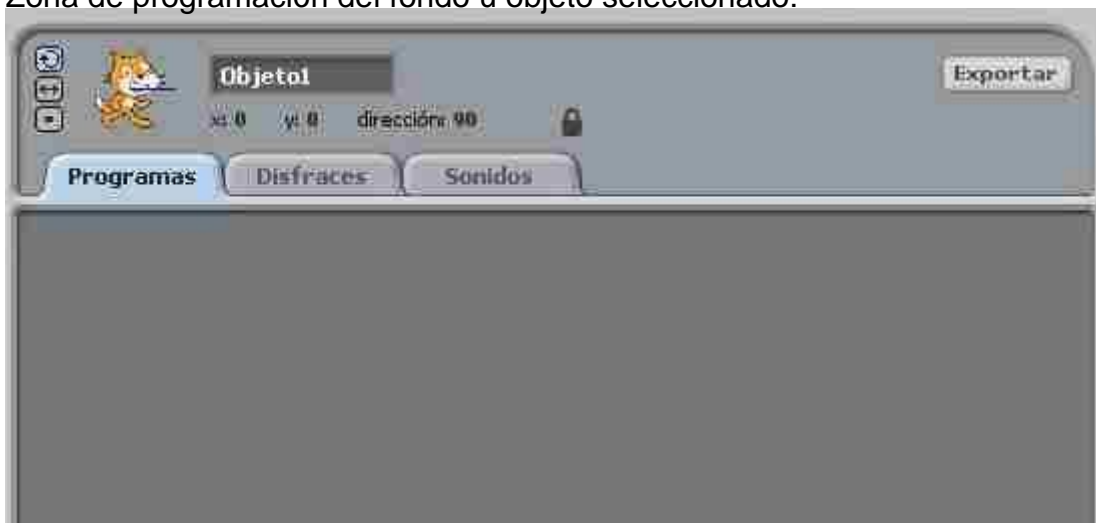
Menú:



Bloques o elementos del lenguaje (cambia en función de la categoría seleccionada):



Zona de programación del fondo u objeto seleccionado:





Las “pestañas”, que nos permiten modificar las características del objeto seleccionado.

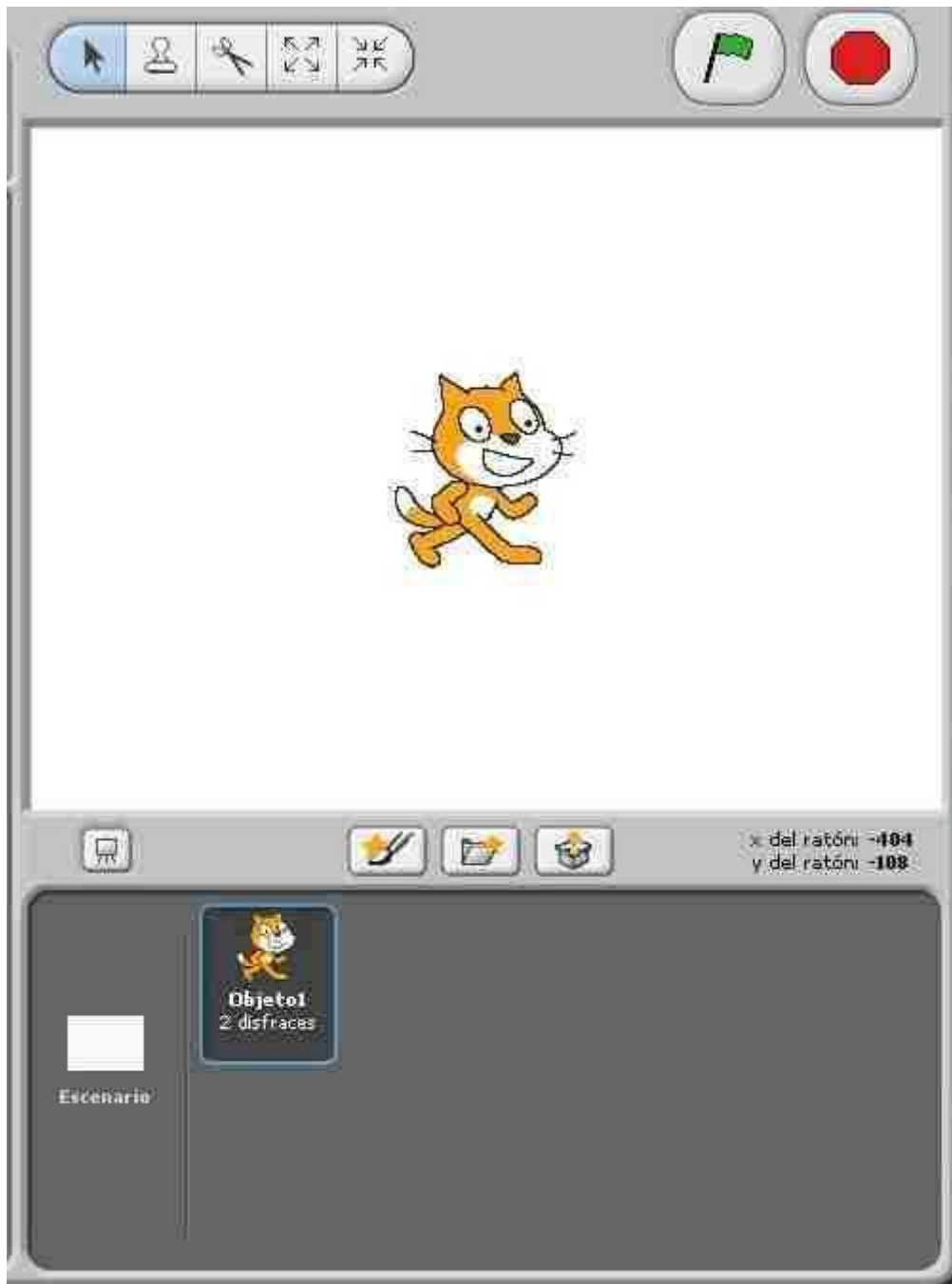
Si es un “sprite” (una figura que podemos animar), tenemos las opciones de “Programas” (scripts), “Disfraces” y “Sonidos”:



Si es el escenario, tenemos “Programas”, “Fondos” y “Sonidos”:



Paleta de herramientas, botones de ejecución, escenario y lista de objetos (sprites):



## SELECCIONAR OBJETO, ARRASTRAR Y SOLTAR

Como dijimos en la introducción, se utilizan “piezas” que se encajan entre sí para crear los programas (programas que normalmente son animaciones, pero no únicamente). Cuando abrimos el entorno, encontramos por defecto el “gato” que representa a Scratch en la “pizarra” de ejecución (escenario).



Tenemos por tanto un escenario, al que podemos añadir tantos fondos como deseemos (editándolos desde el entorno o cargándolos desde fichero), y los

objetos que deseemos añadir (también se pueden crear en el escenario o bien cargarlos desde archivo).

Para programar cada uno de los elementos, basta con seleccionarlo con el ratón desde la lista de objetos:



Inmediatamente, el entorno se prepara para recibir las órdenes asociadas al elemento seleccionado:



El entorno se adecua al nuevo motivo, para que podamos establecer sus propiedades:

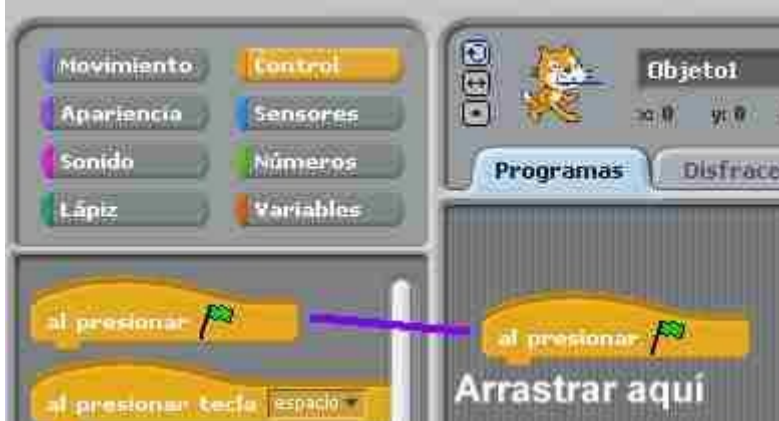




Para añadir órdenes al objeto seleccionado, basta con seleccionarlo y arrastrar los bloques que lleven a cabo las acciones que deseemos “ordenar” al mismo. Conviene comenzar el bloque por un elemento de control de “Al presionar...”, que se puede encontrar el primero al seleccionar los bloques de “control”:



Arrastraremos el elemento a la zona de elaboración de “scripts”:



Así podremos ir “encajando” piezas, hasta formar nuestro programa, e incluso añadiendo elementos de repetición, de bifurcación, etc. Veamos un programa sencillo ya elaborado, seleccionando “piezas” de los diferentes tipos (Movimiento, Control, Apariencia, Sonido...):



Este pequeño programa hará que el “Objeto1”, al presionar la “bandera verde” de ejecución, repita 20 veces la siguiente secuencia: moverse cinco pasos, cambiar disfraz y tocar el sonido “miau”.

Si por el contrario, quisiéramos que el sonido “miau” sólo sonara al final de la repetición de la secuencia, el encaje de bloques debería ser algo así:



## “HOLA MUNDO” CON SCRATCH

Como dijimos más arriba, cuando iniciamos un nuevo proyecto aparece siempre por defecto un escenario en blanco con nuestro ya conocido gato “Scratch” en medio del mismo.

Vamos a crear una secuencia diferente, representando nuestro primer programa con Scratch.

La secuencia sería la siguiente:

Crear el escenario y los objetos

Modificar el fondo blanco del escenario por otro de los que se incluyen en la instalación estándar del programa

Eliminar el objeto del gato y añadir un nuevo objeto desde la biblioteca de elementos incorporados

Programar el elemento para que al iniciar la ejecución se posicione en la parte inferior izquierda de la pantalla y se desplace hasta el lateral derecho, caminando y “diciendo algo” al final...

Arracaremos Scratch o pulsaremos el botón “Nuevo” si ya estaba abierto y hemos realizado alguna modificación en el entorno. Si Scratch nos pregunta si queremos guardar el proyecto actual, contestaremos “No”.



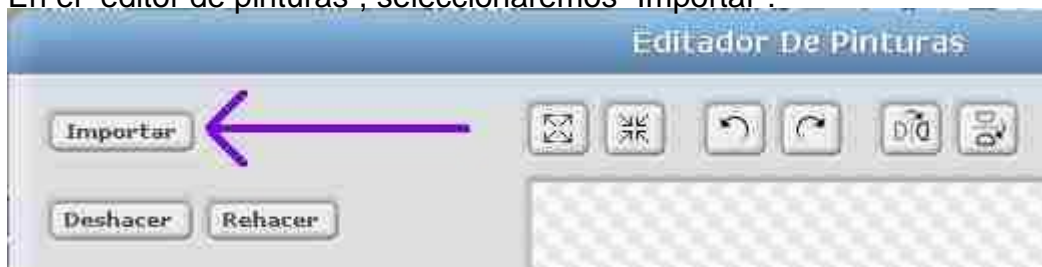
Elegiremos la herramienta borrar y pulsamos sobre el objeto que vamos a eliminar, en este caso, el gato:



Ya tenemos nuestro escenario en blanco. Ahora, elegiremos la opción de "Pintar objeto nuevo":



En el "editor de pinturas", seleccionaremos "Importar":



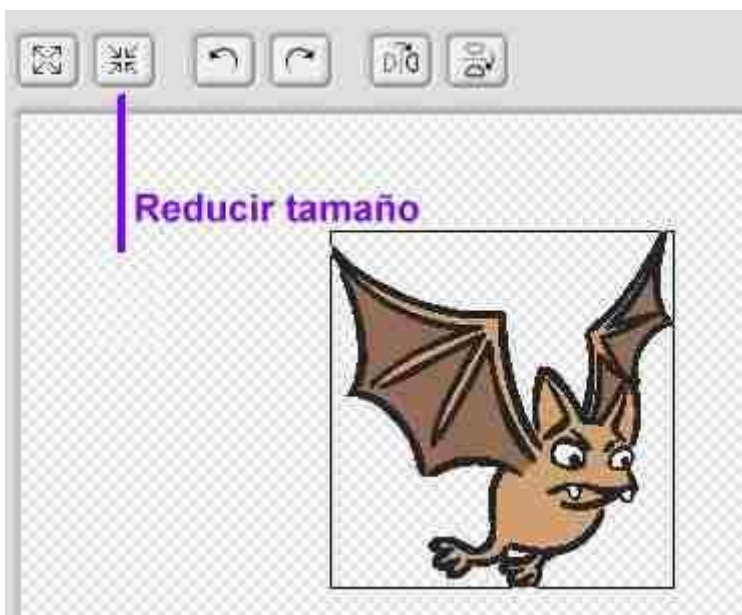
Abriremos la carpeta "Animals" (animales):



Seleccionamos a nuestro primer protagonista: “bat1-a”:



Una vez en el editor de pinturas, le modificaremos el tamaño para hacerlo un poco más pequeño. Después, pulsaremos “Aceptar”.



Ya tenemos nuestro objeto en el escenario. Ahora, le vamos a añadir un segundo “disfraz”, para poder dar la sensación de que está “volando”.

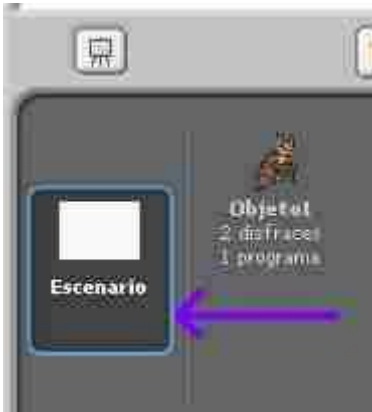
Seleccionamos la pestaña “Disfraces”, para poder “importar” un disfraz nuevo, en este caso, “bat1-b”. Lo reduciremos también, si procede:



Ya tenemos un objeto (llamado "Objeto1") con dos disfraces, y situado en el centro de la pantalla:



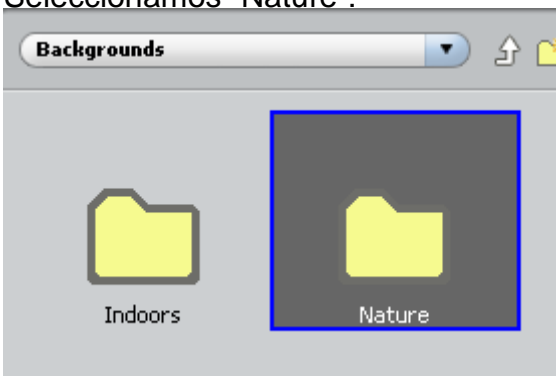
Vamos a cargar también un nuevo escenario para darle un poco más de entidad a nuestro "objeto". Para ello, seleccionamos "Escenario":



Seleccionamos "Fondos -> Importar":



Seleccionamos "Nature":



Y elegimos el fondo que nos guste más. En nuestro caso, vamos a escoger "Stars":



Ahora, vamos a programar el desplazamiento de nuestro murciélago, “montando” el puzzle con las siguientes piezas. Recordemos que basta con arrastrar y soltar cada pieza, eligiéndola de las secciones que corresponda. Hacemos “clic” en la pestaña “Programas”, teniendo a nuestro “Objeto 1” seleccionado:



Y ahora empezamos a arrastrar, en el orden indicado, hasta que obtengamos el programa que se puede ver en la imagen:

1) Control



2) Movimiento



Nótense los valores de x e y.

3) Control



Nótese el valor “20”

4) Movimiento



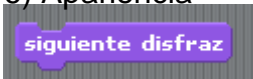
Nótese el valor “5”

## 5) Control



Nótese el valor “0.25”

## 6) Apariencia



El programa resultante sería algo así:



Si ahora pulsamos sobre la bandera verde de la zona superior derecha de la pantalla, podremos ver a nuestro murciélago realizar algo “similar” a un vuelo...



Podemos guardar nuestro proyecto con el botón “Guardar” del Menú. Ya tenemos nuestro primer “script” con Scratch construido. Ahora, podemos “jugar” con él y modificar valores para ver cómo queda el vuelo, por ejemplo, con un tiempo de espera inferior o superior, colocando a “Objeto1” en otro lugar de la pantalla, etc.

Obviamente, podemos añadir otros elementos y “programarlos” para que realicen por la pantalla el movimiento que nos parezca oportuno, que se escuchen determinados sonidos, etc.



Un factor motivacional importante es que el alumnado puede elaborar sus propias creaciones (dibujos, imágenes, fotografías, sonidos, etc.) e incorporarlos a Scratch y a su proyecto en particular a través de la opción “Importar” tanto de la sección “Objetos” como “Escenario”.

Podemos complicar nuestro proyecto y añadirle otros elementos, de manera que con cada aleteo se escuche un “pop” y el personaje diga o piense “algo” al finalizar su “viaje”...:



En este caso hemos utilizado elementos de Control, Movimiento, Apariencia y Sonido. Pero podemos añadir contadores, variables, y un largo etcétera. Por ejemplo, podemos crear una variable que se incremente y que provoque la ejecución de un evento determinado...:

Así, el programa:



Al presionar la bandera verde, el objeto se desplaza a la posición (-140,-30).

Se fija el valor de la variable “Tocar” (la variable se crea en la pestaña “variables”) a 0 en cada ejecución. Se repite la ejecución de movimiento 20 veces. En cada pasada, se comprueba si el resto de “Tocar” dividido entre 5 es

igual a 0, y en este caso toca el sonido "Pop" (es decir, va a tocar "pop" en los valores 5, 10, 15, 20...).

Para que "Tocar" incremente su valor en cada pasada, añadimos la sentencia "cambiar Tocar por 1". Cuando nuestro objeto ha llegado a la repetición 20, piensa "¡Llegué!" durante dos segundos.

En pantalla se puede muestra el valor de la variable "Tocar"...



... la cual se ha generado en la opción correspondiente...:



Las variables pueden ser "locales" al objeto o globales para el programa. Si son globales, nos servirán para que los diferentes objetos interactúen entre sí y con el escenario.

Esto es útil, por ejemplo, para cambiar los escenarios dadas las condiciones que nosotros establezcamos.

Como podemos ver con este simple ejemplo, de una manera sencilla e intuitiva estamos manejando conceptos avanzados de programación, en un entorno de ejecución controlado y fácil: variables, bucles, eventos, propiedades y métodos de objetos, etc.

PROPUESTAS DE TRABAJO POR NIVELES

Una de las cosas más interesantes de Scratch es la capacidad de utilizar materiales externos a la propia herramienta, elaborados por el alumnado, de manera que lo que ellos preparan (sus propias fotografías, sus dibujos e imágenes, su música...) puede ser incorporado al programa a través de las opciones de importación.

Otra de las características que convierten a Scratch en una herramienta colaborativa interesante es la posibilidad de compartir las creaciones que desarrollamos con la herramienta en la misma Web de la aplicación, a través de la opción "Compartir", disponible desde el propio entorno de desarrollo. Para esto hace falta haber creado previamente un perfil (una cuenta de usuario) en la web oficial de Scratch. El propio programa se encarga de comprimir todo el material utilizado y enviarlo a la web para ponerlo a disposición del resto de usuarios de Scratch repartidos por el mundo.

Sin necesidad de enviar los materiales a la web oficial de Scratch, es evidente que también entre nuestro alumnado podemos fomentar la compartición de recursos, de manera que todos puedan utilizar los materiales elaborados por todos, y así crear trabajos derivados de los mismos.

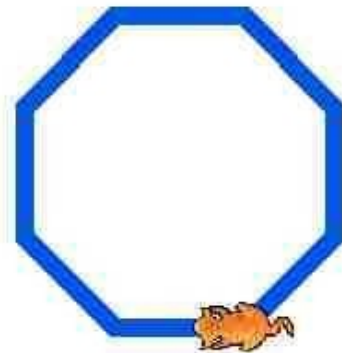
Las sugerencias siguientes no pretenden agotar las posibilidades disponibles, sino sugerir únicamente algunas actividades que pongan en marcha al grupo. Seguro que una vez que comencemos a utilizar la herramienta, se nos ocurren muchísimas posibilidades para explorar con nuestro alumnado desde diversas áreas del currículo...

## **EDUCACIÓN PRIMARIA**

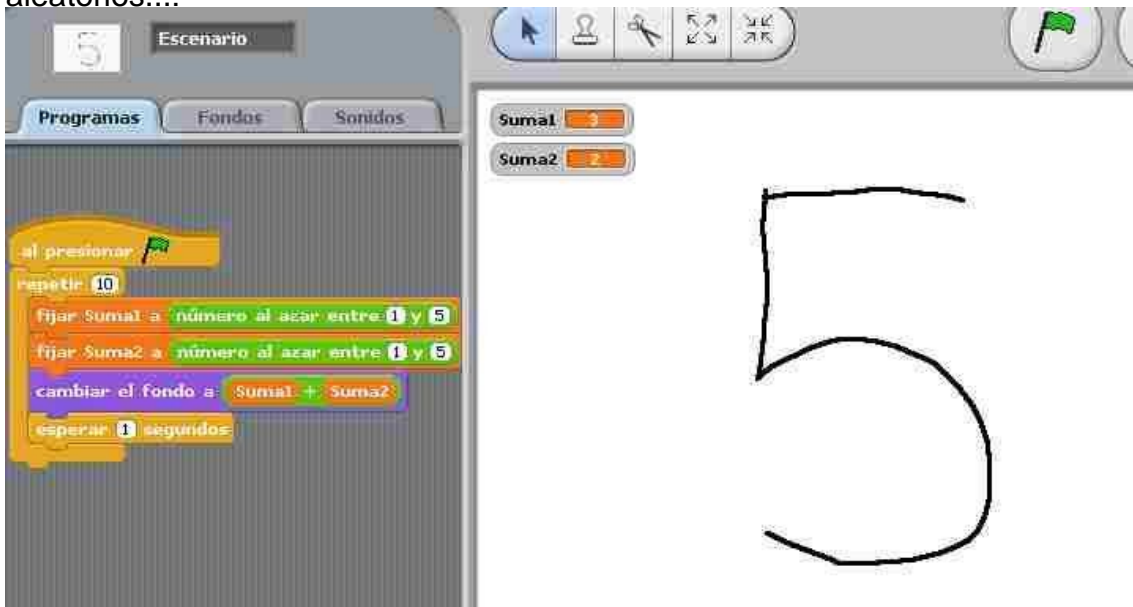
En un artículo anterior dijimos que Scratch se puede comenzar a utilizar con alumnado a partir de los 8 años de edad (aunque para comenzar a utilizar algunas de las funcionalidades deberíamos esperar, seguramente, hasta los 10 años de edad). Estamos hablando de alumnas y alumnos de 3º de Primaria. Dependiendo de la madurez del grupo, podremos llegar a utilizar más o menos elementos del lenguaje, y a construir "scripts" más o menos complejos.

Podemos comenzar a utilizar la herramienta cuando deseemos, incluso con alumnado muy joven, haciendo uso únicamente de proyectos ya realizados, los cuales pueden visualizar, o con los que pueden interactuar, para ir conociendo la herramienta, familiarizarse con el entorno y los personajes, etc.

A partir de segundo Ciclo, la propuesta sería comenzar a utilizar la herramienta como una forma de contar historias. Podemos ubicar personajes y crear diálogos entre ellos, hacer que se desplacen por la pantalla, etc. También podemos "jugar" con Scratch a la manera de la consabida tortuga de Logo, haciendo que nuestros personajes dibujen por la pantalla imágenes o diseños a través de la construcción de nuestro script:



Ni que decir tiene que Scratch, como cualquier herramienta de programación, es una magnífica calculadora. Aquí hemos hecho una pequeña “trampa” para que se nos muestre como fondo el resultado de la suma de dos números aleatorios...:



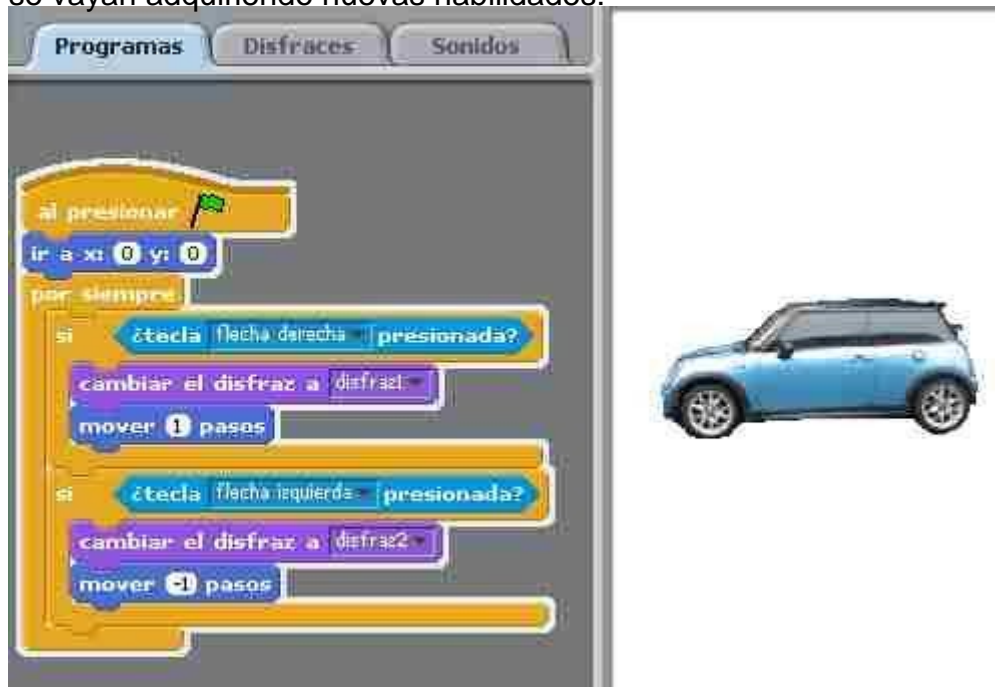
Se pueden utilizar, obviamente, los “objetos” y “disfraces” ya disponibles que contienen letras y dígitos.

Para finalizar, en cualquiera de los cursos de Primaria podemos comenzar a trabajar con el análisis de pequeños programas ya realizados para mejorarlos, cambiar los elementos de una aplicación para personalizarlos, etc.

## EDUCACIÓN SECUNDARIA OBLIGATORIA

Comenzamos a poder trabajar más profundamente conceptos abstractos.

En primer lugar, y sin perjuicio de que esto se podría comenzar en algún curso anterior, podemos comenzar a asignar eventos a nuestros personajes, a utilizar el teclado y el ratón para manejar los objetos, y sobre todo a asignar acciones que ocurren cuando varios objetos interaccionan entre sí. Así podremos conseguir efectos interesantes, que se irán complicando sucesivamente según se vayan adquiriendo nuevas habilidades:



Un elemento interesante que se puede trabajar en la ESO (probablemente antes también, según las características del grupo) son los “mensajes” que unos objetos pueden enviar a otros. Cuando se produce una condición determinada, un objeto “envía un mensaje” a otros objetos, de manera que estos lleven a cabo una acción determinada.

En las siguientes imágenes establecemos que un objeto que se desplaza por la pantalla envíe un mensaje al resto de objetos cuando se da una determinada condición. En el momento de producirse dicho mensaje, el escenario se modifica para dar paso al fondo siguiente que hayamos generado en nuestra aplicación:



## **A PARTIR DE SECUNDARIA OBLIGATORIA...**

En todas las etapas educativas donde se vaya a trabajar cualquier herramienta que sirva para comenzar a programar un PC, hay que intentar plantear las actividades evitando que nuestro alumnado se “lance” a programar inmediatamente con el teclado delante del PC sin hacer ningún tipo de planificación previa.

Entendemos que antes de comenzar a diseñar cualquier pequeña historia, script o aplicación, se debe comenzar haciendo un diseño básico de lo que se pretende hacer, una planificación del trabajo, una distribución del mismo, etc.

De alguna manera, este es un hábito de trabajo que se debe adquirir en general con cualquier actividad que se lleve a cabo en el aula, pero es especialmente relevante hacerlo explícito en cualquier actividad relacionada con la programación.

Si en todas las etapas educativas esto sería importante, a partir de los últimos cursos de la ESO y obviamente en cualquier etapa posterior, podemos establecer un sistema de trabajo más similar al que se plantea en equipos y entornos de desarrollo reales, sin dejar por ello de mantener los factores motivacionales que hemos estado defendiendo en todas las etapas anteriores.

También a partir de 3º de la ESO (y específicamente en las etapas posteriores) se podrán elaborar diagramas de flujo, plantillas con el pseudocódigo de la aplicación / historia que vamos a desarrollar, etc.

En definitiva, en todas las etapas (teniendo en cuenta que a mayor edad, mayor nivel de exigencia al respecto), hay que establecer un sistema de trabajo en el cual...:

Se planteen en común los objetivos de la actividad que se va a realizar.

Se establezca claramente el resultado final que esperamos obtener.

Se favorezca la creación de grupos heterogéneos en los que podamos tener alumnado diverso para favorecer también creaciones diversas.

Los grupos se planteen, diseñen, planifiquen y valoren qué actividad van a realizar y cuáles son los requisitos para llevarla a cabo.

Se establezcan fechas y rutinas de “entrega” de trabajos parciales, si procede

Se establezcan las responsabilidades de cada miembro dentro del grupo, los momentos de trabajo individual, de trabajo en subgrupos y de trabajo en común.

Se lleve a cabo sesiones de planificación y se manejen “documentos” (dependiendo de la edad, éstos serán simples diseños o plantillas y cuadros más complejos...) con los requisitos, con las fechas de entrega, con los elementos diseñados, con las fórmulas de integración de los mismos, con los diagramas de flujo, los “story-boards”, o lo que corresponda.

Se prueben los resultados obtenidos y se valoren por parte de todo el grupo, así como por agentes externos al mismo.

Por último, se “libere” el resultado final del proyecto que se ha llevado a cabo, y se modifique si procede o si se desea en función de las sugerencias de todos aquellos que han podido acceder al mismo.



## DESCRIPCIÓN DE BLOQUES

Los bloques de Scratch están organizados dentro de ocho categorías de códigos de color: movimiento, apariencia, sonido, lápiz, control, sensores, números y variables.

MOVIMIENTO	
	Mueve el Objeto hacia adelante o hacia atrás.
	Rota el Objeto en el sentido de las manecillas del reloj.
	Rota el Objeto en el sentido contrario a las manecillas del reloj.
	Apunta el Objeto en la dirección especificada (0=arriba; 90=derecha; 180=abajo; -90=izquierda).
	Apunta el Objeto hacia el puntero del ratón o hacia otro Objeto.
	Mueve el Objeto hacia una posición específica de X, Y en el escenario.
	Mueve el Objeto a la ubicación del puntero del ratón o de otro Objeto.
	Mueve el Objeto suavemente a una posición determinada en un lapso de tiempo específico.
	Cambia la posición X del Objeto en una cantidad determinada (incrementa).
	Fija la posición X del Objeto a un valor específico.
	Modifica la posición Y del Objeto en una cantidad determinada (incrementa).
	Fija la posición Y del Objeto a un valor específico.
	Gira el Objeto en sentido contrario, cuando este toca un borde del escenario.
	Informa la posición X del Objeto. (Rango entre -240 a 240)
	Informa la posición Y del Objeto (Rango entre -180 a 180)
	Informa la dirección del Objeto (0=arriba; 90=derecha; -90=izquierda; 180=abajo)








APARIENCIA	
	Modifica la apariencia del Objeto cambiando de disfraz.
	Cambia el disfraz del Objeto por el siguiente disfraz en la lista de disfraces (cuando llega al final del listado de estos, vuelve a comenzar con el primer disfraz).
	Informa el número correspondiente al presente disfraz del Objeto
	Modifica la apariencia del escenario cambiando a un fondo diferente.
	Modifica la apariencia del escenario pasando al siguiente fondo disponible en el listado de estos.
	Reporta el número del fondo actual del escenario.
	Despliega una nube de diálogo del Objeto durante un lapso de tiempo determinado
	Despliega una nube de diálogo del Objeto (se puede eliminar esta burbuja de diálogo ejecutando este bloque sin texto alguno).
	Despliega una nube de pensamiento del Objeto durante un determinado lapso de tiempo.
	Despliega una nube de pensamiento del Objeto.
	Modifica un efecto visual del Objeto en una cantidad especificada (use el menú desplegable para seleccionar el efecto).
	Establece un efecto visual a un número dado (la mayoría de efectos visuales se ubica en un rango de 0 a 100).
	Limpia o borra todos los efectos gráficos de un Objeto
	Modifica el tamaño del Objeto en una cantidad especificada (incrementa).
	Ajusta el tamaño del Objeto en un porcentaje (%) específico respecto a su tamaño original.
	Informa el tamaño del Objeto como porcentaje (%) de su tamaño original.
	Hace aparecer un Objeto en el escenario.
	Hace desaparecer un Objeto del escenario (cuando el Objeto está escondido, otros Objetos no lo pueden detectar con el bloque "¿tocando?").
	Ubica el Objeto al frente de todos los demás Objetos (capa superior).
	Mueve el Objeto hacia atrás, un número determinado de capas, de manera que pueda ocultarse detrás de otros Objetos.

## SONIDO

	Comienza la reproducción del sonido seleccionado del menú
--	---



	desplegable, e inmediatamente pasa al siguiente bloque aunque el sonido se esté ejecutando aún.
tocar sonido <b>miau</b> y esperar	Reproduce un sonido y espera hasta que el sonido termine, antes de continuar con el bloque siguiente.
detener todos los sonidos	Detiene todos los sonidos.
tocar tambor <b>48</b> durante <b>0.2</b> pulsos	Reproduce un determinado número de sonido de tambor, seleccionado del menú desplegable, durante un número específico de pulsos.
tocar nota <b>60</b> durante <b>0.5</b> pulsos 	Reproduce una nota musical (número altos para tonos altos) durante un número específico de pulsos.
silencio por <b>0.2</b> pulsos	Descansa, no toca nada, durante un número específico de pulsos.
fijar instrumento a <b>1</b>	Establece el tipo de instrumento que usa el Objeto para los bloques de tocar notas (cada Objeto tiene su propio instrumento).
cambiar volumen por <b>-10</b>	Modifica el volumen del sonido del Objeto en un valor especificado (incrementa o decrementa el volumen).
fijar volumen a <b>100</b>	Fija el volumen del sonido del Objeto a un valor específico.
 <b>volumen</b>	Informa el volumen del sonido del Objeto.
cambiar tempo por <b>20</b>	Modifica el <i>tempo</i> del Objeto en una cantidad específica (incremental).
fijar tempo a <b>60</b> ppm	Fija el tempo del Objeto a un valor especificado de pulsos por minuto.
 <b>tempo</b>	Informa el tempo del Objeto en pulsos por minuto.



LÁPIZ	
	Borra todas las marcas de lápiz y de sellos (estampados) del Escenario.
	Baja el lápiz del Objeto, de manera que este pinte a medida que se mueve.
	Levanta el lápiz del Objeto, de manera que no pinte cuando se mueva.
	Establece el color del lápiz, basado en la selección hecha en la paleta de color.
	Modifica el color del lápiz en una cantidad específica.
	Establece el color del lápiz a un valor determinado. (color-lápiz=0 en el borde rojo del arco iris; color-lápiz=100 en el borde azul del arco iris)
	Modifica la intensidad del lápiz en una cantidad especificada (incrementa).
	Establece un valor específico para la intensidad del lápiz (sombra-lápiz=0 es muy oscura; sombra-lápiz=100 es muy clara).
	Cambia el grosor del lápiz (incrementa en una cantidad específica).
	Establece el grosor del lápiz.
	Estampa o copia la imagen del Objeto en el Escenario.



CONTROL	
	Ejecuta el programa que tiene debajo al hacer clic en la bandera verde.
	Ejecuta el programa que tiene debajo al presionar una tecla específica.
	Ejecuta el programa que tiene debajo al hacer clic en un Objeto.
	Espera un número determinado de segundos y continúa luego con el bloque siguiente.
	Ejecuta continuamente los bloques en su interior.
	Ejecuta, un número específico de veces, los bloques en su interior.
	Envía un mensaje a todos los Objetos, activándolos para que hagan algo y espera a que todos terminen antes de continuar con el siguiente bloque.
	Envía un mensaje a todos los Objetos y luego continúa con el bloque siguiente sin esperar a que se realicen las acciones de los Objetos activados.
	Ejecuta el programa que tiene debajo cuando recibe un mensaje específico "enviar a todos" (Broadcast).
	Comprueba continuamente si una condición es verdadera; cada que es verdadera, ejecuta los bloques en su interior.
	Si la condición es verdadera, ejecuta los bloques en su interior.
	Si la condición es verdadera, ejecuta los bloques dentro de la porción si; si no, ejecuta los bloques que están dentro de la porción si no.
	Espera hasta que la condición sea verdadera, para ejecutar los bloques siguientes.
	Comprueba si la condición es falsa; si lo es, ejecuta los bloques en su interior y vuelve a chequear la condición. Si la condición es verdadera, pasa a los bloques siguientes.
	Detiene el programa del Objeto que contiene esta instrucción.
	Detiene todos los programas de todos los Objetos.



SENSORES	
	Informa la posición "X" del puntero del ratón.
	Informa la posición "Y" del puntero del ratón.
	Informa verdadero, si el botón del ratón está presionado.
	Informa verdadero, si una tecla específica está presionada.
	Informa verdadero, si el Objeto está tocando un Objeto específico, un borde o el puntero del ratón (seleccionados del menú desplegable).
	Informa verdadero, si el Objeto está tocando un color específico. (Haga clic en la paleta de color y luego utilice el gotero para seleccionar el color).
	Reporta verdadero si el primer color (dentro del Objeto), está tocando un segundo color (tanto en el fondo como en otro Objeto). Haga clic en la paleta de color y luego utilice el gotero para seleccionar el color.
	Informa la distancia desde un Objeto específico o desde el puntero del ratón.
	Fija el cronómetro en 0.
	Reporta el valor del cronometro en segundos (el cronómetro siempre está contando).
	Informa una propiedad o variable de otro Objeto.
	Reporta el volumen de los sonidos captados por el micrófono del computador (entre 1 y 100).
	Reporta verdadero, si el volumen del sonido captado por el micrófono del computador es mayor de 30 (en escala de 1 a 100).
	Informa el valor de un sensor específico. Para usar este bloque se necesita tener un "PicoBoard" conectado a su computador. Para aprender más consulte: <a href="http://www.playfulinvention.com/picoboard.html">http://www.playfulinvention.com/picoboard.html</a>
	Informa verdadero, si un sensor específico está presionado. Para usar este bloque se necesita tener un "PicoBoard" conectado a su computador. Para aprender más consulte: <a href="http://www.picocricet.com/picoboard.html">http://www.picocricet.com/picoboard.html</a>



NÚMEROS	
	Suma dos números.
	Resta dos números
	Multiplica dos números.
	Divide dos números.
	Selecciona un número entero al azar dentro de un rango especificado.
	Informa verdadero, si el primer valor es menor que el segundo.
	Reporta verdadero, si dos valores son iguales.
	Informa verdadero, si el primer valor es mayor que el segundo.
	Informa verdadero, si ambas condiciones son verdaderas.
	Informa verdadero, si una de las dos condiciones es verdadera.
	Reporta verdadero, si la condición es falsa; reporta falso si la condición es verdadera.
	Reporta el resultado de una función seleccionada (abs, raíz cuadrada, sin, cos, tan, asin, acos, atan, ln, log, e^, 10^) aplicada a un número específico.
	Informa el resto (módulo) de la división del primer número por el segundo número.
	Informa el entero más cercano a un número.

VARIABLES	
	Permite crear y nombrar una nueva variable. Cuando usted genera una variable, aparecen los bloques correspondientes a ella. Se puede escoger si la variable es para todos los Objetos (global) o solo para un Objeto (local)
	Borra todos los bloques asociados con una variable
	Informa el valor de la variable
	Modifica la variable en una cantidad determinada (Si se tiene más de una variable, utilice el menú desplegable para seleccionar el nombre de la variable)
	Fija la variable a un valor específico.
	Muestra el monitor de la variable en el escenario
	Esconde el monitor de la variable para que no aparezca en el escenario
	Permite crear y nombrar una nueva lista. Cuando se genera una lista, aparecen los bloques para esa lista. Se puede escoger si la lista es para todos los Objetos (global) o solo para un Objeto (local)





	Borra los bloques asociados a una lista.
	Reporta todos los elementos que tiene la lista.
	Adiciona el elemento especificado al final de la lista (el elemento puede ser un número o una cadena de letras u otros caracteres).
	Borra uno o todos los elementos de una lista. Se puede escoger del menú desplegable o usar un número para indicar qué elemento borrar. Si escoge "último" borrará el último elemento de la lista. Si escoge "todos" borra todo lo que contiene la lista. Borrar, decrementa la longitud de la lista.
	Inserta un elemento en un lugar específico de la lista. Se puede escoger del menú desplegable o usar un número para indicar dónde insertar el elemento dentro de la lista. Si escoge "último" adiciona el elemento al final de la lista. Si se escoge "cualquiera" lo inserta aleatoriamente en la lista. La longitud de la lista se incrementa en 1.
	Reemplaza un elemento de la lista con un valor específico. Se puede escoger del menú desplegable o usar un número para especificar el elemento que va a reemplazar. Si escoge "último", reemplaza el último elemento de la lista. Si escoge "cualquiera" reemplaza aleatoriamente un elemento de la lista. La longitud de la lista no se modifica.
	Reporta el elemento en una ubicación específica dentro de la lista. Se puede insertar este bloque dentro de otros bloques como
	Reporta cuántos elementos hay en la lista.

# SCRATCH

Scratch es un nuevo entorno de programación visual y multimedia basado en Squeak destinado a la realización y difusión de secuencias animadas con o sin sonido y al aprendizaje de programación.

Desarrollado por Media Laboratorios de MIT y por el grupo KIDS de UCLA, Universidad de California, Los Ángeles.

Ofrece una interface intuitiva y muy fácil de comprender, donde podremos manipular imágenes, fotos, sonido, música, etc. y sobre todo una forma de programación visual con todos estos elementos. Podremos crear nuestros personajes, cambiar su apariencia y hacerlos interactuar con los otros. Por ultimo podremos publicar nuestras creaciones para que todo el mundo pueda verlas.

Con Scratch recuperamos el modelo constructivista de Logo o de los E-Toys de Squeak.

Scratch está disponible sobre Windows, Mac OS X y Linux y se difunde bajo licencia libre MIT.

Cuando se trabaja con Scratch se comprenden fácilmente conceptos matemáticos e informáticos que están muy bien integrados en el programa, como son:

- Los procesos interactivos (bucles),
- Los criterios condicionales (si, entonces, si-no),
- Las coordenadas en un plano,
- Las variables, etc

Además estos conceptos se aprenden dentro de un contexto significativo y motivador. No es lo mismo comprender el significado de las variables en un contexto de aprendizaje de la programación tradicional, que cuando se utilizan para el control de la visualización de una animación o en juego que uno mismo está construyendo.

Los alumnos trabajan sus propios proyectos, aprenden mucho sobre el proceso de su concepción. Partiendo de una idea tiene que crear el prototipo funcional, es decir, un modelo, y aportar y experimentar las soluciones que considere. Cuando no funcionan tendrá que volver hacia atrás y corregir. Se crea una espiral continua: partiendo de una idea se crea un proyecto, que da lugar a nuevas ideas, que generan nuevos proyectos, y de este modo indefinidamente.

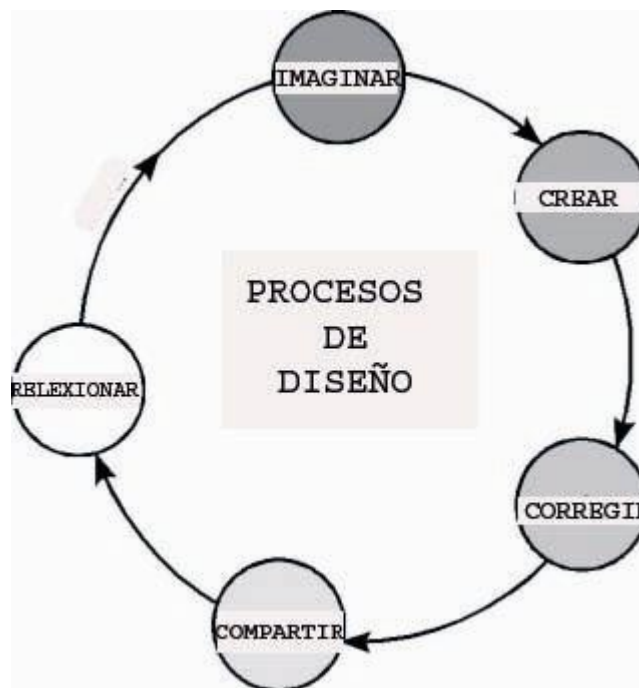
Estos procesos de concepción de proyectos y su construcción, desarrollan las competencias necesarias para llegar a tener un pensamiento creativo, una



comunicación clara, un análisis sistemático, capacidad de colaboración, la reflexión interactiva, etc.

De la misma forma que es necesario aprender no solamente a leer sino también a escribir, lo mismo se debe conseguir con la informática, nuestros alumnos no solamente deben aprender a utilizar e interactuar con el ordenador, también debe aprender a crear con esta herramienta.

Lógicamente no todos los alumnos van a ser programadores profesionales, pero aprender a programar permite a los alumnos experimentar de forma creativa y ayuda al desarrollo del pensamiento lógico a la vez que comprenden el funcionamiento de las nuevas tecnologías que encontrarán en su vida cotidiana.



### ¿Dónde podemos encontrar Scratch?

De la página web de Scratch: <http://scratch.mit.edu/>

Zona de descarga de Scratch:

<http://llk.media.mit.edu/projects/scratch/download/>

Cuando ejecutas el programa se muestra una pantalla en la que tienes las siguientes partes:

**ESCENARIO:** La zona más grande, blanca, donde está la mascota de Scratch, único actor en ese momento. Ha se desarrollara la acción.

**BOTONES NUEVOS SPRITES:** Justo debajo del escenario hay tres botones que nos van a permitir buscar o incluso crear nuevos actores para nuestra acción.

**MODO PRESENTACIÓN:** Si pinchamos en el botón que queda a la izquierda de los anteriores podemos ampliar el escenario a pantalla completa.

**LISTA DE SPRITES:** En la zona que queda debajo de la anterior aparecerán las miniaturas de los sprites que vayan a actuar. Hacer clic para seleccionar y editar un sprite.

**BARRA DE HERRAMIENTAS:** Aparece encima del escenario y nos sirven para mover los objetos, copiar, cortar, aumentar su tamaño o disminuir su tamaño.

**BANDERA VERDE:** Sirve para ejecutar los guiones o programas que hayamos creado.

**BOTON ROJO:** Sirve para detener la acción.

**NOTAS DEL PROYECTO:** Para añadir comentarios al proyecto.

**PALETA DE BLOQUES:** A la izquierda. Es nuestra caja de instrucciones para que nuestros actores hagan cosas. Son los bloques para crear los scripts o programas. Tenemos 8 cajas diferentes de piezas, para distintas acciones, cada una de un color. Tendremos que ir familiarizándonos con ellas: Movimiento, apariencia, sonido, lápiz, control, sensores, números y variables.

**AREA DE SCRIPTS:** Entre las dos zonas anteriores queda otra muy importante, es área de scripts. Aquí iremos arrastrando los bloques de nuestras cajas de piezas para ir construyendo los guiones o programas.

**PESTAÑAS:** Justo encima del área de scripts tenemos tres pestañas que nos permiten añadir guiones, disfraces o sonidos al guión que construimos.

**ESTILO DE ROTACIÓN:** Nos permite determinar la rotación de los sprites.

**INFORMACIÓN SPRITE SELECCIONADO:** En el cuadro aparecerá el nombre del sprite y una miniatura del mismo.

A partir de este momento debemos recordar los dos elementos fundamentales de nuestro lenguaje de programación: **sprite y guión**.

**ESTILO DE ROTACIÓN**  
Controla si los disfraces rotan con el sprite

**PESTAÑAS**  
Hacer clic para editar guiones, disfraces o sonidos

**INFORMACIÓN SPRITE SELECCIONADO**

**BARRA DE HERRAMIENTAS**

**BANDERA VERDE**  
Una forma de ejecutar guiones.

**NOTAS DEL PROYECTO**

**ESCENARIO**  
Desde cobran vida las creaciones de Scratch.

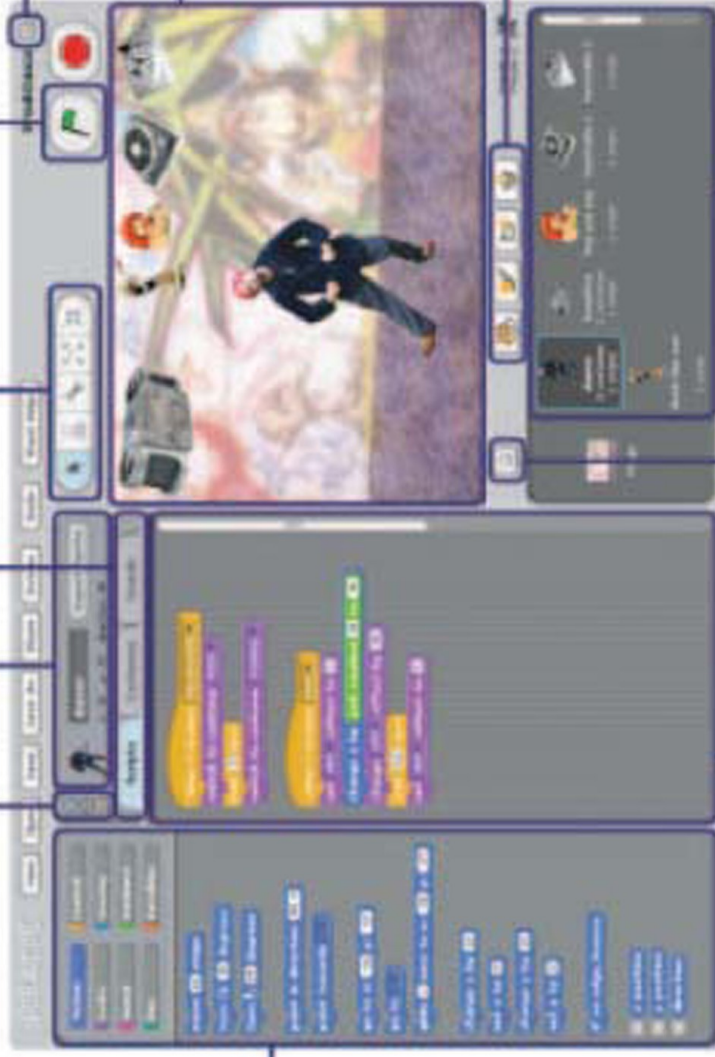
**PALETA DE BLOQUES**  
Bloques para programar sprites

**AREA DE SCRIPTS**  
Arrastrar bloques a ella, apilándolos juntos en guiones.

**BOTONES NUEVOS SPRITES**  
Crear un nuevo objeto para el proyecto.

**MODO PRESENTACIÓN**

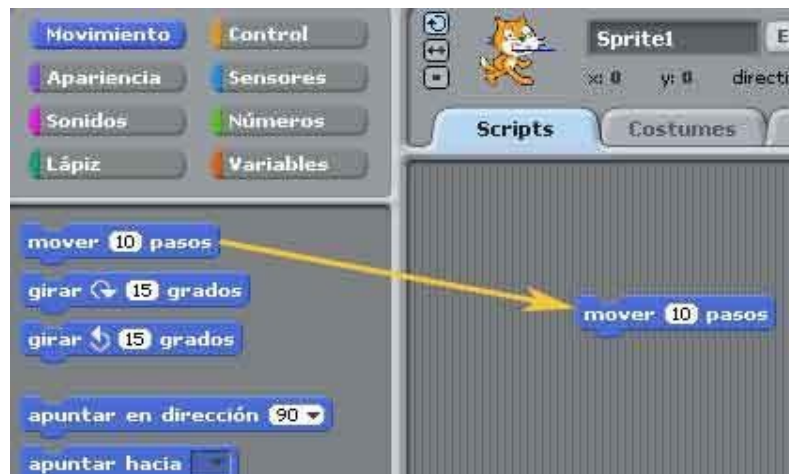
**LISTA DE SPRITES**  
Miniaturas de todos los sprites. Hacer clic para seleccionar y editar un sprite



## 1. JUGANDO CON NUESTRO GATO (CAT1)

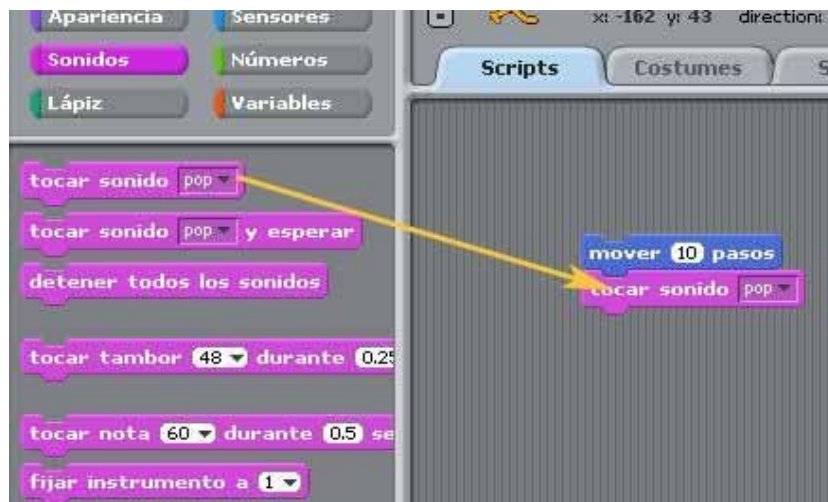
Para empezar, una vez instalado el programa, jugamos un poco con el software para comprender su funcionamiento, bastante simple e intuitivo.

Con nuestro gato como protagonista en el escenario vamos a la zona de trabajo y de la caja azul que contiene las **piezas de movimiento** deslizamos el ladrillo **moverse 10 pasos** a la zona de scripts



Hacemos doble clic sobre el ladrillo y mirar el gato.....¡se mueve!

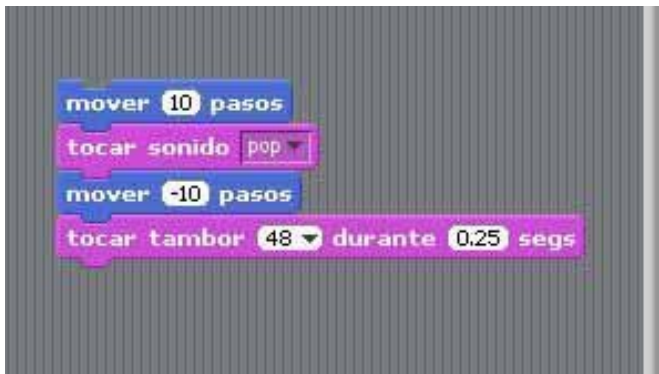
Ahora vamos a probar las piezas de la **caja de sonidos**. Pulsar en sonido y seleccionar el ladrillo **tocar sonido**, arrastrar sobre el ladrillo "moverse 10 pasos", para que queden ensamblados.



Hacemos doble clic sobre los ladrillos ensamblados y escuchamos....¡Suenan!

Si queremos cambiar sonido solo hay que pinchar en el nombre del sonido y vemos un menú donde nos ofrece otros sonidos.

Elegimos ahora la opción movimiento y arrastramos sobre el ladrillo de percusión otro ladrillo **mover a**. Modificamos el valor 10 por -10, además volvemos a seleccionar sonido y añadimos otro ladrillo **tocar percusión** y le encajamos al anterior.



Hacemos doble clic sobre los ladrillos ensamblados..... **el gato avanzara y retrocederá mientras se escucha el sonido.**

Seleccionamos ahora la caja con las piezas de **Controles** y deslizamos el ladrillo **repetir indefinidamente**



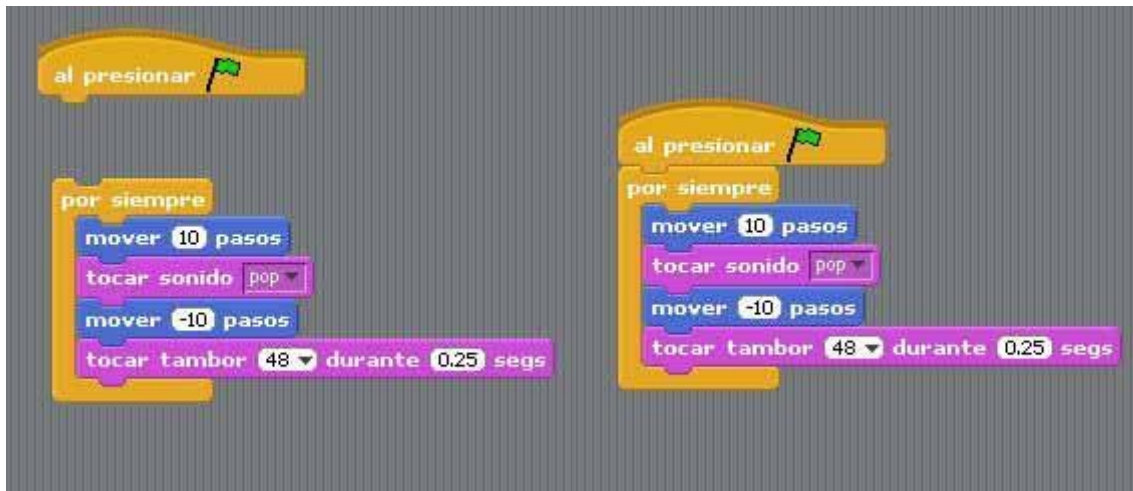
Arrastramos el bloque de ladrillos al interior de la estructura de control “repetir indefinidamente”

Hacemos doble clic en el nuevo bloque. Para terminar pulsamos el botón parada



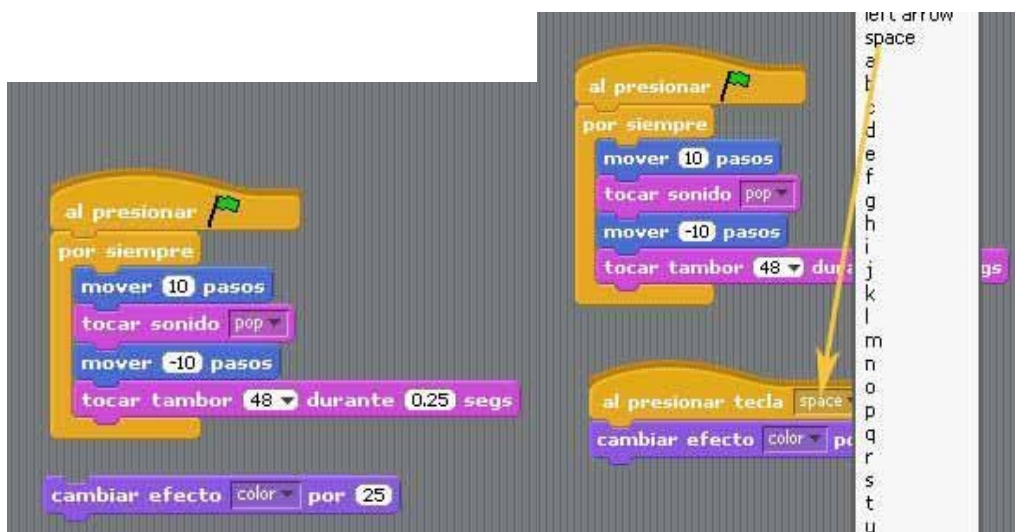


Añadimos otro control: arrastramos el ladrillo que tiene la bandera verde **al presionar** y la añadimos al bloque.



Cuando pulsemos en la bandera verde comenzara a ejecutarse las acciones. Vamos a probar ahora alguna cosa diferente:

Pulsamos la opción “apariencia” y deslizamos el ladrillo **cambiar el efecto**:



Seleccionamos ahora el ladrillo **cuando pulses espacio** y se lo añadimos al ladrillo **cambiar efecto**.

Pulsamos ahora sobre la bandera verde y vemos que sucede con nuestro gato.....

Hemos dado nuestros primeros pasos jugando con Sprite1. Hemos visto algunos de los ladrillos que tenemos en la paleta de bloques:

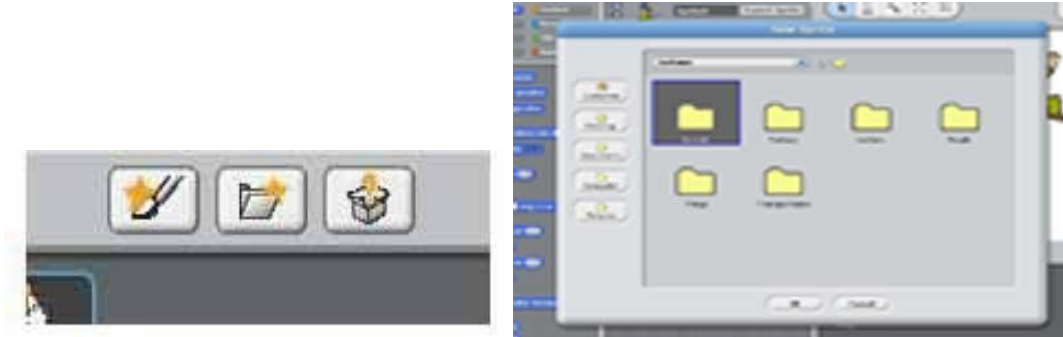


En las paletas de bloques encontramos los ladrillos necesarios para construir nuestro programa o guión, en la zona Scripts, para conseguir que nuestro objeto actúe de la forma deseada.

Ahora habrá que realizar el primer ejercicio.

## 2. AÑADIENDO OBJETOS:

Partiendo del trabajo anterior vamos a añadir un objeto en nuestra escena. Si observamos en el entorno del escenario encontramos tres botones:



El primero nos permite crear nuestro propio objeto, el segundo crear un nuevo objeto a partir de un fichero y el tercero introducir un objeto al azar del banco de objetos. Vamos a probar añadiendo un objeto de la carpeta de ficheros.

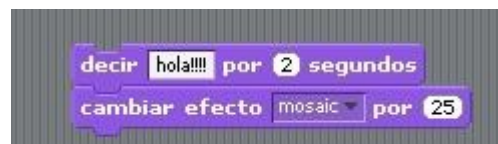
Para ello pinchamos en el icono central y se abrirá la pantalla con las carpetas y si pulsamos en la carpeta en Fantasy encontramos los objetos incluidos en la misma y seleccionaremos al Dragon2.

Seleccionamos el dragon2, además queremos hacerle hablar, para lo cual seleccionamos Apariencia y el ladrillo “decir hola” durante 2 segundos. Si pulsamos sobre el ladrillo comprobamos lo que sucede.



Volvemos a la sección Apariencia y cliqueamos sobre “Cambiar el efecto”.

De los efectos que tenemos disponibles y que vemos si pulsamos en la flecha que aparece al lado de color, seleccionamos mosaic:

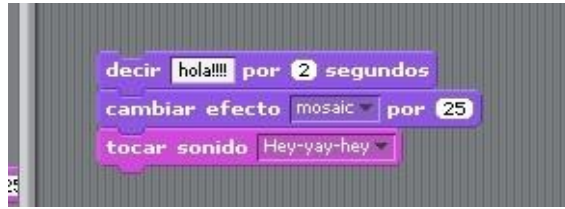


Ejecutamos y vemos que sucede. Probamos el botón:

Volvemos a nuestra zona de scripts y añadimos el ladrillo “tocar sonido pop” cambiando pop por el que acabamos de seleccionar:

Hacemos doble clic en el bloque y comprobamos que sucede con nuestro dragón.



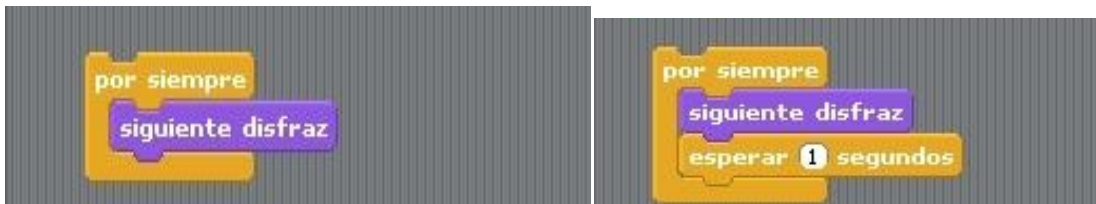


Hacemos doble clic en el bloque y comprobamos que sucede con nuestro dragón. Vamos ahora a añadir un poco de animación. Para ello vamos a seleccionar la pestaña Costumes:



Pinchamos en la pestaña Costumes y después en Import, aparecerán una serie de carpetas en las que tenemos objetos disponibles. Seleccionamos la carpeta People. De las imágenes que vemos seleccionamos la imagen Jodi2.

Volvemos al scripts y a la sección Control y arrastramos el ladrillo: "repetir indefinidamente". Ahora de Apariencia elegimos el ladrillo "Siguiente disfraz" y se arrastra dentro del control "repetir indefinidamente".



Como podemos comprobar el bucle indefinido que supone que se repita el cambiar de objeto de forma indefinida hace que sea demasiado rápido, por lo que volvemos a los controles y seleccionamos "Esperar 1 segundo". Añadimos este ladrillo al bloque.

Y ahora sí que podemos ver la animación. Para detener pulsamos el botón rojo

## 4. Animación sencilla de un objeto

Vamos a realizar una sencilla animación de un objeto al que vamos a modificar su aspecto inicial.



Para ellos abrimos las carpetas donde guardamos nuestros objetos

Y volvemos a seleccionar nuestro dragón anterior. Para deshacernos de los demás objetos que tengamos en el escenario nos ponemos sobre ellos, pulsamos en el botón derecho y elegimos delete (borrar).

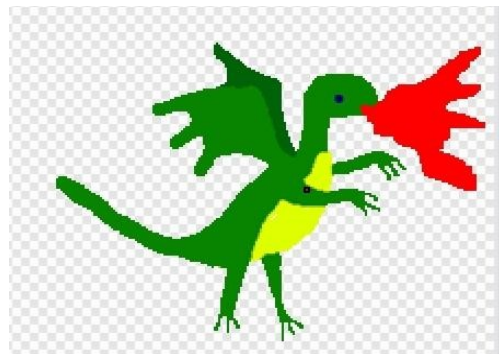
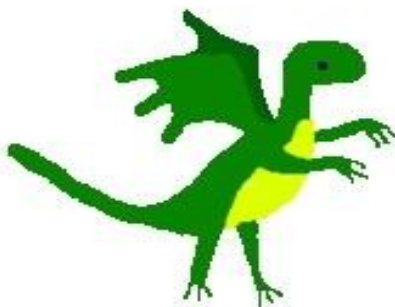
Abrimos la pestaña Costumes y hacemos una copia de nuestro objeto. .



Seleccionamos el nuevo dragon3 que hemos creado y pulsamos en EDIT.

Modificamos un poco nuestro dragón para hacerle más temible de acuerdo con nuestras posibilidades de edición gráfica, con la práctica se consiguen buenos resultados. En caso de que no nos guste pulsamos Undo. Cuando estemos satisfechos pulsamos ok.

Tenemos ahora dos dragones:



Con un simple clic podremos animar nuestro objeto.

En la sección Control elegimos el ladrillo “Al presionar la bandera verde” y la arrastramos a la zona de scripts. En la misma sección seleccionamos el control de “Repetir indefinidamente” y arrastramos al scripts. Seleccionamos ahora dos ladrillos de “esperar un segundo”. **No los encajes enseguida.**

En la sección Apariencia seleccionamos y arrastramos dos ladrillos “cambiar el disfraz a..” sin apresurarnos en el ensamblado de las piezas.



Cuando pulsamos en la bandera se ejecuta y cuando pulsamos en el botón rojo se para.



## 5. Juego sencillo:

Vamos a realizar un juego muy sencillo: un gato que corre detrás del ratón.

Lo primero que tenemos que hacer para crear un nuevo proyecto es hacer desaparecer los objetos que tengamos en el escenario.

Comenzamos el **primer proyecto** que consiste en un juego:

En la carpeta donde tenemos los animales seleccionamos nuestro gato para esta actividad: cat4, casi un tigre.



Lo primero que hacemos es ir a la sección de Contro y elegimos dos ladrillos:



Que no ensamblamos todavía.



En la sección Movimiento elegimos y arrastramos dos ladrillos:

En el primero le decimos que apunte hacia el ratón y en el segundo cambiamos a mover 3 pasos.



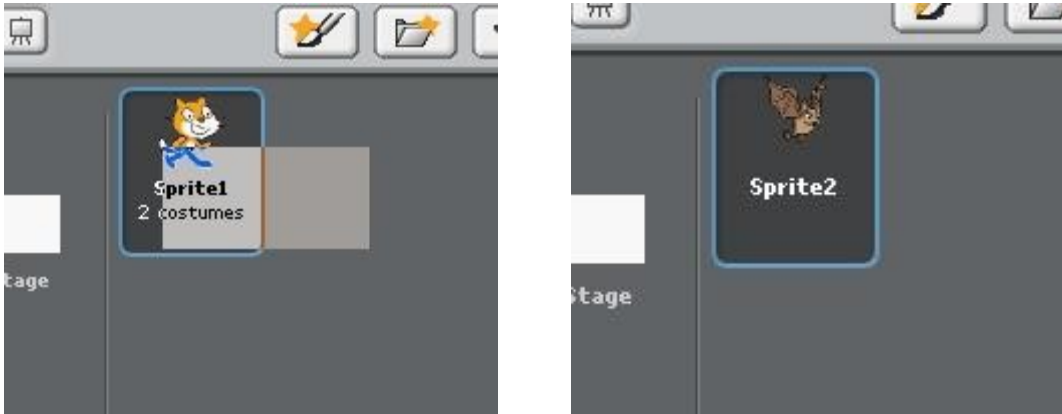
El ensamblado quedara como se ve en la figura.

Solo falta pulsar en la bandera verde y comprobar si nuestro fiero gato llega a cazar al ratón.

## 6. Imagen Interactiva:

Ahora vamos a explorar una serie de características que nos permiten modificar de forma interactiva una imagen.

Crear un nuevo proyecto y quitar el gato del escenario y sustituirlo por:



Arrastrar a la zona de script el ladrillo que indica que presionemos la bandera verde, de la zona de control seleccionar la iteración Repetir indefinidamente.

Ahora en apariencia seleccionar fijar efecto y colocar en la iteración y cambiar la opción color a whirl. Después, del cajón de sensores seleccionar el ladrillo sensor x y colocar este ladrillo sobre el cero que aparece en el ladrillo de fijar efecto.

Obtenemos el siguiente script:



Añadirle otro ladrillo fijar efecto y cambiar color por mosaic y el cero por el sensor y.

Pulsar la bandera verde para que se ejecute.

## 6. Animación y desplazamiento:

Hasta este momento algunas de las características más entretenidas del Scartach, ahora vamos a ver cómo podemos modificar de forma interactiva una imagen. Para ello crearemos un nuevo proyecto en el que nuestro protagonista será un murciélago: bat2-a que encontramos en la carpeta de animales.



Pulsamos sobre el botón Import y recuperamos la figura del otro murciélago bat2-b.



Después pulsamos en el primer murciélago. Deslizamos el objeto sobre la escena en altura y hacia la izquierda.



Con el botón



Hacemos disminuir el tamaño del murciélago. Los dos murciélagos deben tener el mismo tamaño.

Ahora construiremos el scripts que nos permita hacer volar a nuestra figura.

Al pulsar sobre la bandera veremos a nuestro murcielago volar hacia abajo y luego subir cuando llega al borde.

La dirección del vuelo la definimos con las señales que aparecen a la izquierda de la miniatura de la figura, justo encima de la palabra Scripts.



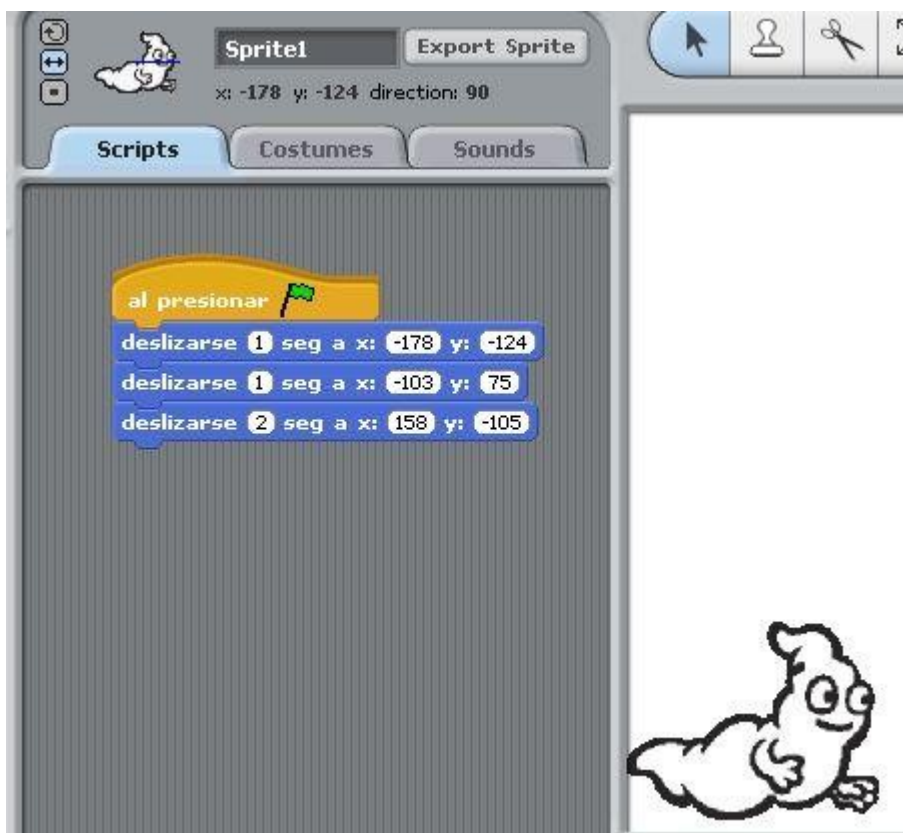
## 8. Desplazamiento de los objetos:

Realizaremos ahora un pequeño proyecto donde los objetos se desplazaran fácilmente. Para este proyecto elegimos como protagonista a ghost, que será nuestro fantasma.

Buscamos la figura en la carpeta y la llevamos al escenario. Hacemos disminuir un poco el tamaño. Luego le colocamos de forma que en la línea donde se indica la posición aparezca: x: -178 y: -124 dirección 90

Con esto hemos fijado su desplazamiento.

Escribimos el scripts:

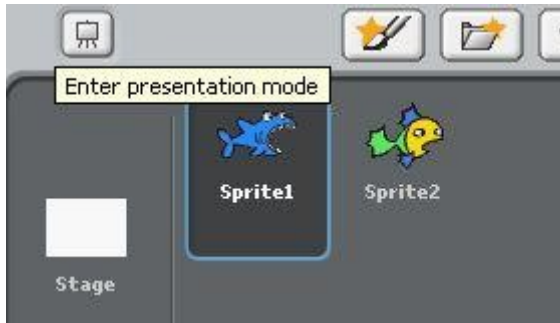


Solo queda ejecutar pulsando la bandera para ver su desplazamiento.



## 9.- Utilización de variables

Cuando programamos uno de los elementos fundamentales son las variables. Una variable es un dato que cambia de valor en la ejecución del programa. Como podemos ver en Scratch también disponemos de la posibilidad de trabajar con variables. Para ver su utilización vamos a realizar un proyecto en el cual shark1-b perseguirá fish3 intentando tragárselo. Seleccionamos los peces en la carpeta de Animals:



Los dos peces son del mismo tamaño y Sprite1 debe ser mas grande que Sprite 2 y para ello vamos a utilizar las variables.

Lo primero que tenemos que hacer es dar nombre a la variable. Para ello pulsamos sobre la opción variable:



Vemos que aparecen dos opciones:

Nueva variable, que nos permite crear las variables y Borrar variable para eliminarla.

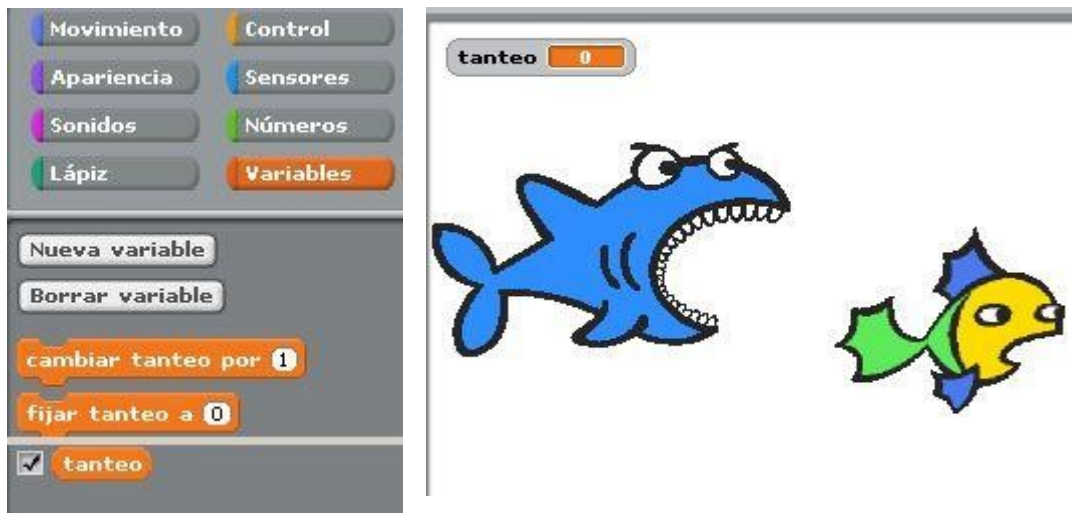
Pulsamos en Nueva variable y aparece una nueva ventana en la cual podremos dar nombre a nuestra variable.

En esta ocasión vamos a denominarla **tanteo**

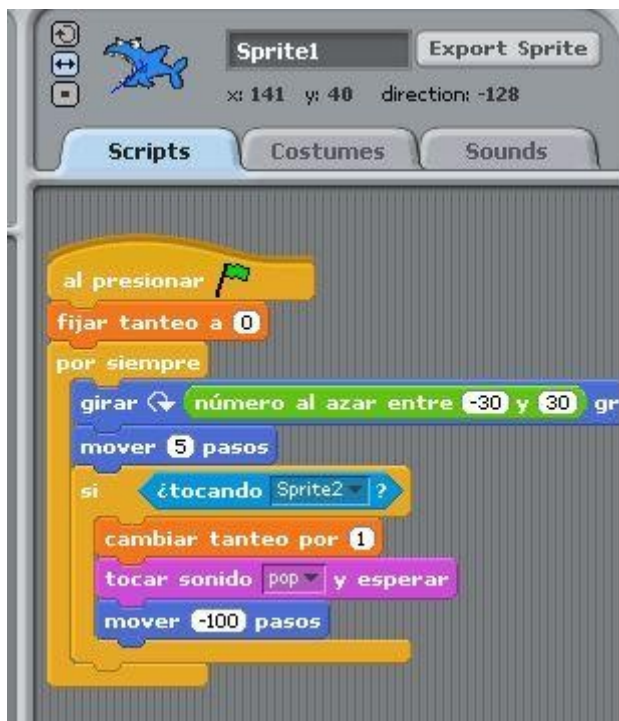


Vemos ahora que en columna anterior aparece dos nuevas cajas que nos van a permitir cambiar los valores de las variables. Las variables pueden recibir el nombre que nosotros queramos, podíamos haberlas denominado x, peces, etc.... Si activamos la opción tanteo en el escenario se vera la variable con su valor.





Ahora vamos a crear el Script:



De esta forma sprite1 se va moviendo por el escenario y cuando toca a sprite2 la variable tanteo aumenta en uno.

Al finalizar la ejecución tanteo contendrá el numero de veces que ha tocado al pez pequeño.

Solo nos queda ponerle un decarado al escenario. Para ello nos colocamos sobre:



Luego elegimos la pestaña background e importamos una imagen que nos va a servir de fondo marino para los peces.

Elegimos underwater





Conseguiremos una imagen de la siguiente forma:



## 10.- Un balón sorprendente:

Ahora vamos a crear un botón **cliqueable** a partir de un objeto insertado en nuestra escena.

Iniciamos un nuevo proyecto para el cual vamos a elegir la carpeta Things de Costumes.

Elegimos la pelota de colores y la denominamos pelota:



A continuación creamos el scrip:



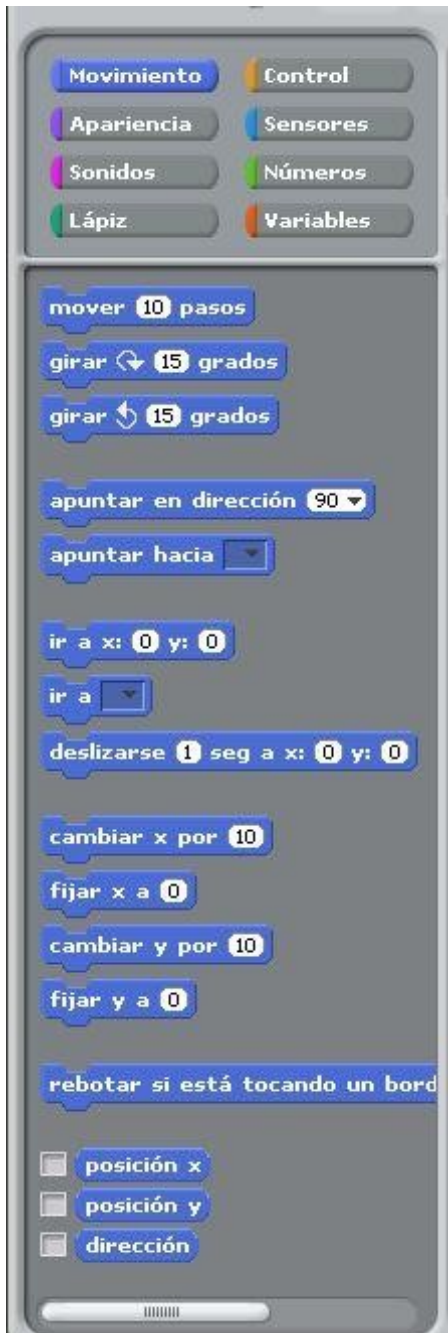
Mediante el ladrillo



Conseguimos que cuando pulsamos la tecla espaciadora, además de cambiar de colores, se oye una nota elegida al azar.

De esta forma conseguimos que nuestro programa funcione de forma interactiva.

# MOVIMIENTO



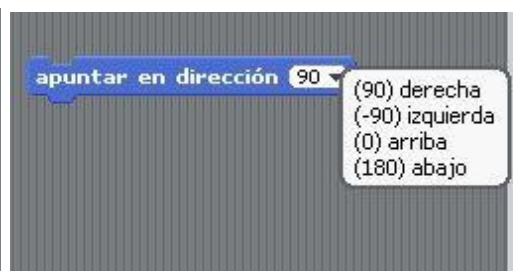
Para desplazar el objeto por el escenario. Si el numero es negativo se desplazará en dirección contraria.



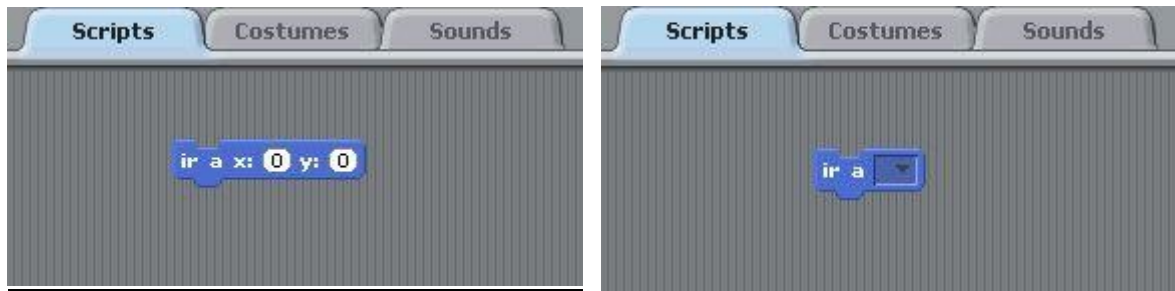
Rotar los grados indicados en sentido del reloj



Rotar en sentido contrario al reloj



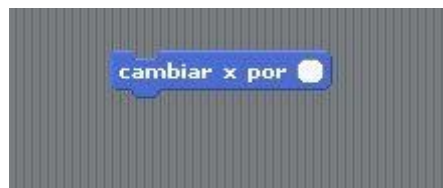
Permiten indicar la dirección del desplazamiento, en grados o por el nombre del objeto al que debe dirigirse.



En el primer caso diremos las coordenadas que determinan el punto hacia el que se mueven, en el segundo el nombre del objeto.



Se le indica el tiempo durante el cual se desplazara hacia el punto que le indican las coordenadas.



Para fijar el valor de la coordenada x



Para fijar el valor de la coordenada y



Permite cambiar la dirección de desplazamiento del objeto al tocar un borde del escenario



Al activar las casillas correspondiente obtenemos las coodenadas y la dirección del objeto que tenemos en el escenario.

Sprite2's posición x **-135.0**  
Sprite2's posición y **13.0**  
Sprite2's dirección **180.0**

