Scratch en el cole





Reconocimiento-NoComercial-Compartirlgual 4.0 Internacional

http://www.aprendizdetecnologo.com antonio@aprendizdetecnologo.com

- Scratch es un lenguaje de programación basado en el uso de bloques (no hay que escribir código), desarrollado en el MIT
- Se aprende a programar en Scratch mediante el juego y la experimentación. Hagamos lo que hagamos, algo pasará.

- Podemos descargar una app para programar en Scratch sin conectarnos a Internet, aquí.
- Lo más normal (y rápido, en una clase Althia) será utilizar la versión online
- Podemos registrarnos para compartir nuestras creaciones, pero no es necesario. https://scratch.mit.edu

• Una vez en la web, podemos acceder directamente a la opción **Crear**.

https://scratch.mit.edu

Para los profes, es también muy interesante la opción **Explorar**, donde podemos acceder a miles de programas en Scratch

https://scratch.mit.edu

 Tanto si estamos registrados o no, todos nuestros programas pueden guardarse en nuestro ordenador

https://scratch.mit.edu



En Scratch, programamos objetos(sprites) dentro de un escenario, de modo que cambien sus propiedades de acuerdo a la sucesión de distintos eventos. (¿A que suena súper difícil? ¡Veréis como no es así!) En todos los proyectos empezaremos con el gatete símbolo de Scratch, que tendrá el nombre objeto1



Todos los objetos tienen unas propiedades comunes: nombre, coordenadas x/y, tamaño en %, ángulo de giro y modo de giro (normal, espejo o bloqueado)

Podemos añadir tantos objetos como queramos a nuestro programa. Al principio, de tipo prefabricado, pero luego querremos crear nuestros propios objetos.





Aparte, todos los objetos pueden tener uno o varios disfraces (apariencia) y, en su caso, **sonidos**. De nuevo, podemos crear los nuestros propios o utilizar los prefabricados de Scratch

Los disfraces pueden modificarse a nuestro gusto, utilizando un interface similar a nuestro querido Paint





Los sonidos son también perfectamente configurables. Uno de los mayores juegos en clase consiste en crear y "tunear" cada objeto al gusto de los niños.

Un caso particular de **objeto** o sprite es el Escenario en el que queremos que se desarrolle el programa. Tiene las mismas propiedades y comportamiento que el resto de objetos, pero sólo puede haber uno (aunque puede cambiar su fondo múltiples veces)









Realizamos nuestro código para que el programa reaccione a **EVENTOS**: tiene que pasar algo para que el ordenador haga otra cosa.

IMPORTANTE: cada bloque de código que utilicemos para programar hará referencia EXCLUSIVAMENTE al objeto (o escenario) que tengamos activo en ese momento. En ocasiones estaremos buscando un programa que creemos haber perdido, simplemente porque hemos cambiado de objeto de programación

Podemos empezar moviendo un sprite cada vez que se haga click en la bandera verde, al tiempo que se oiga uno de los sonidos del objeto:





Ese mismo evento (u otro distinto) podría, por ejemplo, producir un cambio en el fondo del escenario:

Mozilla Firefox



Mozilla Firefox



Este es el primer principio que es fácil perder de vista: cada elemento en pantalla es un objeto, y cualquier programa que realicemos con él es independiente del resto de los objetos. Un mismo evento puede disparar varios programas en varios objetos a la vez.

Vamos a tratar de mover un sprite con los cursores. Para ello, utilizaremos bloque de código de **Eventos, Movimiento, Apariencia** y **Control**



Es en estos casos cuando entenderemos para qué sirve el bloqueo de giro dentro de las propiedades generales del objeto:





Ideal para programadores perezosos (como yo): se pueden duplicar y reconfigurar bloques de código

!Ya podemos controlar nuestro sprite mediante los cursores!

Podemos programar cualquier objeto para que esté pendiente **de algo** mediante sensores, y que reaccione a cualquier cambio en ese aspecto.

Empecemos con algo sencillo: vamos a modificar el fondo de nuestro escenario para que incluya algo de color verde

Vamos a incluir este bloque dentro de cada uno de los cuatro programas del gato (izquierda, derecha, arriba y abajo)









¡Solucionado!. Ya tenemos un gatito controlado por cursores y con una extraña fobia cromática.

Una aplicación típica de este conjunto de conocimientos es la realización de laberintos:



Una **variable**, en programación, es un valor con un **nombre** cuyo **valor** puede **variar** (de ahí su valor)

Vamos a hacer un minijuego sobre tablas de multiplicar. Sólo necesitaremos el sprite original, **(objeto1**), y un escenario con cualquier fondo. Crearemos dos variables: **num1** y **num2**

En el apartado de variables, hay varias posibilidades: crear variables, modificar su valor, mostrarlas o esconderlas

Creemos num1 y num2





El bucle principal del programa se iniciará con la bandera verde y necesita encerrarse dentro de un bloque de control Por siempre

A cada inicio de **Por siempre**, desde los grupos de **Variable** y **Operadores**

asignaremos valores aleatorios a ambas variables.



Desde los grupos **Sensores, Variables** y **Operadores** (para unir cadenas de texto), haremos la pregunta al jugador .



En el grupo **Sensores**, el valor **Respuesta** recoge lo que escriba el niño o niña.



Con un bloque de control **Si..si no...** combinado con el operador *, comprobaremos si la respuesta es acertada.



Podemos mejorar el juego todo lo que queramos. Podemos poner un sistema de vidas, incluir movimientos, sonidos, etc...



En ocasiones, querremos que un **objeto** o **sprite** realice el programa que hemos construido para él, y a continuación otro **objeto** o incluso el mismo **escenario** realice otra tarea a continuación. Para ello, utilizaremos un tipo especial de evento: **EL MENSAJE**

	2
Objeto Perrete \rightarrow x 142 \ddagger y -91	Escenario
Mostrar O Ø Tamaño 100 Dirección -88	Fondos

Ilustremos este concepto: vamos a trabajar con dos **objetos** a los que daremos los nombres **Gatete** y **Perrete**

Para Gatete,

recuperaremos el programa que movía al objeto con los cursores:





Ahora, en cada uno de los cuatro bloques de movimiento, incorporaremos un sensor que compruebe si toca a **Perrete**, emitiendo un mensaje:

Los bloques de mensaje están dentro del grupo **Eventos.**

Podemos crear tantos mensajes como queramos:





Cada mensaje nuevo puede utilizarse como un evento que dispare cualquier programa en cualquier objeto. Vamos a iniciar una conversación entre Gatete y Perrete

Cuando **Perrete** recibe el mensaje, ladra (sonido) y envía el mensaje **"LadridoHecho"**



Comunicación entre sprites: mensajes El mensaje "LadridoHecho" a



El mensaje **"LadridoHecho"**, a su vez, será el evento que dispare un nuevo programa que muestre el susto de **Gatete**

Como podéis ver, el uso de **mensajes** dispara la interactividad entre elementos del programa hasta el infinito

Durante esta actividad, hemos hecho un breve repaso de las posibilidades de Scratch en el aula.

Es fundamental no perder de vista que la principal potencialidad de Scratch está en que el alumno aprende, sobre todo, a través de la **experimentación** y el **juego**. No hay que obsesionarse con cumplir un guión predeterminado.

No está mal para el poco tiempo que le hemos dedicado, ¿no?.

¡Hasta la vista!. ¡Sed felices!. ¡Siempre creciendo!. ¡Siempre aprendiendo!



