



¿Qué es Scratch?

Scratch es un entorno de programación desarrollado por un grupo de investigadores del Instituto de Tecnología de Massachusetts (MIT) que aprovecha los avances en el diseño de interfaces para hacer que la programación sea atractiva y accesible para todo aquel que se enfrente por primera vez a programar.

Programando con Scratch se pueden crear historias interactivas, juegos, animaciones, música y producciones artísticas. La página web del MIT también permite compartir creaciones e ideas con otros jóvenes con intereses similares. Desde su lanzamiento al público en Mayo de 2007, la Web de Scratch se ha convertido en una gran comunidad en línea, donde se aprende, se intercambian ideas y se mezclan programas propios con los de otros. Cada día, los *scratchers* de todo el mundo, suben al sitio más de 1.500 proyectos nuevos con el código de acceso libre (cualquiera que esté registrado puede ver cómo se ha creado el proyecto, bajárselo, modificarlo, etc). La edad de la mayoría de visitantes del sitio web se ubica entre los 8 y los 16 años (con un pico en los 12 años) aunque un grupo considerable de adultos participa también.

Scratch está disponible en múltiples idiomas, entre ellos el castellano.

Web Oficial de Scratch: <http://scratch.mit.edu/>




Primer Scratch Day en Valladolid

“Hoy en día se dice que los niños son nativos digitales pero, ¿es eso cierto? Evidentemente que sí; sin embargo, debemos enseñarles a desenvolverse correctamente en este nuevo mundo digital. Está claro que manejan con facilidad cualquier dispositivo que les caiga entre manos pero tienen más dificultades a la hora de crear. Es como si les enseñáramos a leer pero no a escribir. Con Scratch, los niños aprenden a crear integrando y gestionando múltiples elementos multimedia, de tal forma que, a medida que ganan experiencia, se vuelven más receptivos y críticos con las tecnologías y los medios de comunicación que les rodean.”

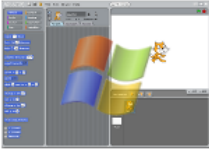
¿Cómo empezar con Scratch?

Lo primero para empezar a trabajar con Scratch es tener el programa instalado en el ordenador. Scratch es gratuito, libre y multiplataforma (Windows, Mac y Linux). Puede descargar el instalador desde la página oficial a través del siguiente enlace: http://info.scratch.mit.edu/es/Scratch_1.4_Download

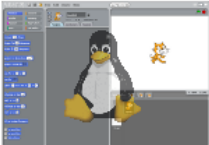
Scratch 1.4 Download



Scratch Installer For Mac OS X
Compatible with Mac OS X 10.4 or later
[MacScratch1.4.dmg](#)

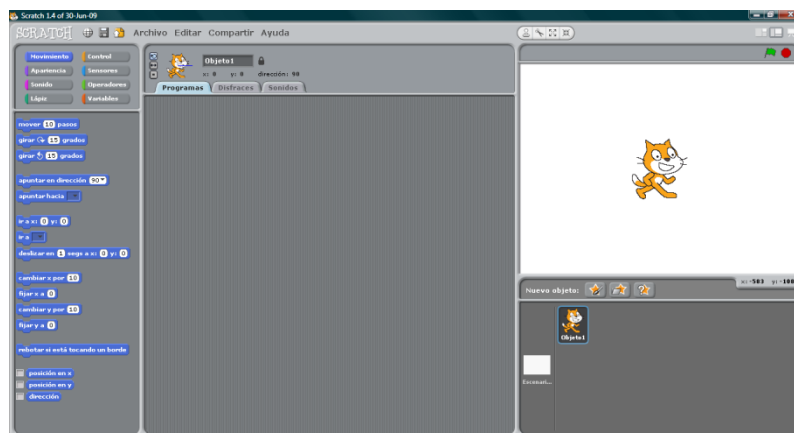


Scratch Installer for Windows
Compatible with Windows 2000, XP, Vista, and 7
[ScratchInstaller1.4.exe](#)
See below for additional Windows options



Scratch Installer for Ubuntu
Compatible with Ubuntu version 9.04 and later
[Scratch_1.4.0.1-0ubuntu5_i386.deb](#)
See the [Scratch on Linux](#) page for more information

Al ejecutarlo, aparecerá la pantalla siguiente:



La programación con Scratch es extremadamente sencilla. Basta con arrastrar los bloques de código de la columna de la izquierda hacia la del medio. En unos segundos se pueden crear sencillos programas sin necesidad de conocer ningún lenguaje ni tener conocimientos de programación.

En cualquier caso, existen guías para iniciarse con el programa.

- Guía de inicio: <http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/files/file/translated-docs/GuiaDeInicioScratch.pdf>
- Guía de referencia: http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/files/file/RG_14_es.pdf

Y también se pueden ver algunos vídeos introductorios muy interesantes a través del siguiente enlace:

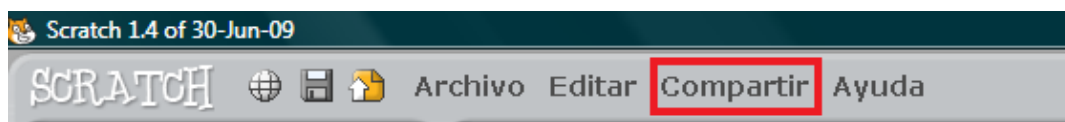
<http://info.scratch.mit.edu/es/Support/Videos>

¿Para qué Scratch?

Atendiendo a las competencias para el siglo XXI definidas por la OCDE y la UNESCO, se puede comprobar que, utilizando Scratch, los estudiantes pueden desarrollarlas de una forma completa, favoreciendo un desarrollo integral del alumno a la vez que se prepara para un futuro profesional incierto pero, sin duda, rodeado de tecnología. Repasemos algunas de estas competencias.

- **Pensamiento creativo:** Scratch favorece la curiosidad intelectual y el pensamiento creativo, habilidades ambas con importancia creciente en el actual mundo cambiante. Scratch implica a la gente joven en la búsqueda de soluciones innovadoras a problemas inesperados, no sólo aprendiendo cómo resolver problemas predefinidos sino enfrentándose a nuevos desafíos y creando nuevas soluciones.
- **Razonamiento abstracto:** Con Scratch, como resultado directo de las clases de programación, el alumno se enfrenta a caminos de razonamiento divergentes, deben anticipar errores y evaluar rápidamente escenarios mentales. Este don sirve hoy en día tanto para manejar una cuenta de correo como para sacar el coche de un aparcamiento.
- **Pensamiento computacional:** A medida que los alumnos aprenden a programar en Scratch, desarrollan el pensamiento crítico y computacional. Para construir los proyectos, los alumnos necesitan coordinar el tiempo y las interacciones entre los diferentes “sprites” (objetos programables). La habilidad de programar de forma interactiva proporciona a los estudiantes una experiencia directa con los conceptos fundamentales sobre sistemas a través de la percepción y con un feedback inmediato.
- **Resolución de problemas:** Scratch favorece la identificación de problemas y su solución en un contexto significativo, innovador y atractivo. Crear un proyecto en Scratch requiere pensar en una idea, preguntarse cómo dividir el problema en varios pasos y después implementarlos utilizando los bloques de código dispuestos para ello. Los estudiantes pueden cambiar dinámicamente el código y ver inmediatamente los resultados. A lo largo del proceso, los estudiantes experimentan y se entrenan en la resolución iterativa de problemas.
- **Aprendizaje autónomo:** Tener una idea y pensar en cómo programarla requiere persistencia y práctica. Cuando los jóvenes trabajan en los proyectos que surgen de sus ideas, encuentran un significado personal. Trabajar sobre sus ideas les proporciona una motivación interna para sobrellevar los desafíos y las frustraciones que encuentren en el proceso de diseño y resolución de problemas.
- **Trabajo colaborativo:** Debido a que los programas de Scratch son construidos con bloques gráficos, el código programado es más fácil de leer y de compartir que en otros lenguajes de programación. Los objetos visuales y el código modular propicia la colaboración, permitiendo a los estudiantes trabajar juntos en proyectos e intercambiar objetos y código. Además, en la página web de Scratch los estudiantes pueden subir sus proyectos y recibir valoraciones y críticas y en la que también se pueden apoyar y colaborar. El concepto de compartir es parte integral de la interfaz de usuario de Scratch la cual tiene, en la parte superior de la página, tanto un menú como un icono de “Compartir” claramente visibles. Explorando los miles de proyectos de la librería existente, los “Scratchers” tienen ideas para nuevos proyectos y aprenden nuevas técnicas de programación.

- **Comunicación:** Una comunicación efectiva requiere, hoy en día, algo más que saber leer y escribir textos. Scratch implica a los jóvenes en la elección, manipulación e integración de distintos elementos multimedia para expresar su creatividad y para poner en práctica su capacidad de persuasión. Cuando los alumnos crean un proyecto, tienen una audiencia en mente y necesitan pensar en cómo responderá otra gente ante sus proyectos. Como los proyectos de Scratch son muy fáciles de revisar y cambiar, los estudiantes pueden modificar sus proyectos basándose en el feedback recibido por otras personas.
- **Manejo de TIC:** Es habitual referirse a las personas jóvenes como “nativos digitales” debido a su aparente fluidez con las TIC. Pero, aunque interactúan con los medios digitales todo el tiempo, pocos son capaces de crear sus propios juegos, animaciones o simulaciones. Es como si pudieran “leer” pero no “escribir”. La competencia digital requiere no solamente tener habilidad para chatear, navegar o interactuar sino también la habilidad de diseñar, crear e inventar con los nuevos medios. Al trabajar en los proyectos de Scratch, los alumnos aprenden a crear, seleccionar y gestionar múltiples elementos multimedia, incluyendo texto, imágenes, animaciones y grabaciones de audio. A medida que los estudiantes ganan experiencia en multimedia, se vuelven más receptivos y críticos en los medios de comunicación que les rodean.
- **Life Long Learning (Aprendizaje a largo plazo):** Con el uso de Scratch, el alumno es quien dirige su aprendizaje. De esa forma, se desarrolla la competencia de aprender a aprender y los estudiantes, al conseguir sus objetivos se percatan de que realmente son capaces de aumentar sus conocimientos por sí mismos, sin depender de instrucción externa.



¿Por qué Scratch?

Para muchos autores, la programación de computadores vuelve tangibles las matemáticas y la computación mediante procesos de diseño en los que los estudiantes se comprometen en la solución de problemas de manera experimental y repetitiva.

Aprender a programar tiene que ver, en el fondo, con aprender a pensar lógicamente y aprender a abordar los problemas metódicamente.

La mayoría de la gente mira la programación de computadores como una actividad especializada y tediosa, a la que solo tienen acceso aquellos con entrenamiento técnico avanzado. Ciertamente, es difícil iniciarse en la programación con los lenguajes tradicionales como Java o C++.

Los bloques de construcción sobre los que se apoyan los programadores para construir soluciones, son relativamente simples: secuencia, iteración y condición. En Scratch, esos bloques ya están implementados y, para utilizarlos, sólo es necesario arrastrarlos al programa que se quiere construir. Además, los bloques de código se pueden cambiar de forma dinámica y el resultado se observa inmediatamente.

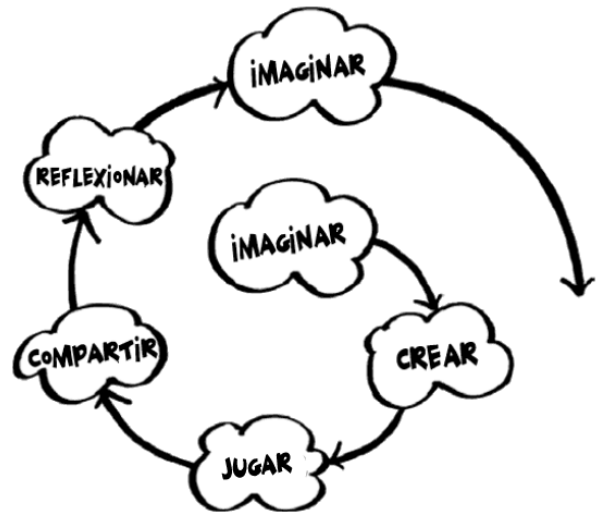
Scratch permite enfrentarse a la resolución de problemas rápidamente, sin necesidad de conocer la sintaxis, a diferencia de los lenguajes de programación tradicionales con los que la curva de aprendizaje es plana, ya que es necesario aprender a leerlo y escribirlo antes de ponerse a trabajar con ellos.

También hay quienes argumentan que en la actualidad prácticamente todo está programado. En este contexto, es indudable que herramientas como Scratch contribuyen efectivamente a preparar a los estudiantes desde la Primaria, para que puedan insertarse activamente al mundo altamente programado que los espera. Pero va más allá: Scratch promueve el desarrollo de algunas habilidades y capacidades intelectuales de orden superior que, en el nivel escolar, son responsabilidad de cualquier sistema educativo de calidad.

Por otro lado, Scratch engancha a los alumnos y permite que cometan errores y copien de lo que hacen otros, lo cual es motor del aprendizaje. Permite que el alumno sea autónomo en su aprendizaje y compartirlo con sus pares a través de los foros oficiales o con sus compañeros de clase y amigos. Además, se divierte aprendiendo lo que favorece la retención a largo plazo de los conceptos realmente importantes.

Foros Oficiales Scratch: <http://scratch.mit.edu/forums/>

Con Scratch, los alumnos desarrollan su creatividad y aprenden; y todo ello lo consiguen ¡programando!



¿Cuándo utilizar Scratch?

Puede que, en un primer momento, se vea la programación como algo exclusivamente reservado a las áreas de Informática y Tecnología. ¡Nada más lejos de la realidad!

Scratch es una herramienta muy potente para la enseñanza de cualquier otra materia ya que es capaz de presentar los conocimientos desde otro punto de vista. Es más, si el alumno desarrolla un proyecto en Scratch sobre un tema de cualquier asignatura, no sólo estará poniendo en marcha sus habilidades como programador, sino que estará directamente implicado con el tema a tratar. El aprendizaje es mucho más efectivo si “haces” y no sólo “ves” y “escuchas”.

De esta manera, Scratch es una herramienta transversal que favorece una enseñanza efectiva del resto de áreas de conocimiento. Indudablemente, las Matemáticas están muy relacionadas con la programación así que, con Scratch, se pueden tratar de manera directa otros aspectos como trigonometría, geometría, operaciones matemáticas y lógicas... Además, utilizando Scratch se puede facilitar la enseñanza de Idiomas, de Geografía y de Historia. Los alumnos pueden, por ejemplo, desarrollar proyectos Scratch representando algún acontecimiento histórico, trabajar con un programa hecho por el profesor de forma sencilla para aprender los ríos españoles, o repasar el vocabulario de inglés sobre “elementos de una cocina” desarrollando ellos mismos una aplicación interactiva.

Por otro lado, al ser una herramienta libre y gratuita, permite que los alumnos puedan seguir trabajando en su casa si así lo desean. El uso de Scratch no está reservado únicamente a las aulas, sino a cualquier lugar donde se disponga de un ordenador.

Scratch es una herramienta muy sencilla de manejar que posibilita resultados muy rápidos a pesar de que el nivel de programación inicial sea nulo. Por eso, es también muy adecuado para el ámbito familiar donde padres, madres e hijos pueden adentrarse juntos en el mundo de la programación. De esta forma, se contribuye a que las horas que sus hijos “gastan” con el ordenador, sean en su propio beneficio, desarrollando sus habilidades y aumentando sus conocimientos.

En definitiva, Scratch es una herramienta muy flexible y adaptable que puede ser utilizada en cualquier contexto y que puede llegar a ser muy eficaz para la enseñanza de múltiples materias a la vez que los alumnos desarrollan y explotan todo su potencial adquiriendo los beneficios que acarrea la programación.

¿Qué hay después del Scratch?

En esta sección se responde a una pregunta que siempre aparece cuando se introduce algún tipo de herramienta novedosa destinada a los niños y/o adolescentes: ¿Qué pasa después?

Habitualmente, los chicos y chicas que se interesan por Scratch, tras muchos proyectos creados, alcanzan un nivel de experto en el manejo de la herramienta y, aunque les sigue pareciendo atractivo lo que hacen, hay que darles otras alternativas para no correr el riesgo de que abandonen el mundo de la programación por falta de estímulo.

Por esa razón, Scratch nos brinda la posibilidad de conectar la herramienta con distintos tipos de robots, de tal forma que los *scratchers* podrán dar vida a objetos reales, lo cual despierta pasiones entre los más jóvenes.

Por otro lado, también existen otros lenguajes de programación que se adaptan perfectamente a los mini-programadores con algo de experiencia pero que aportan un mayor grado de desafío.



Enlaces de interés para educadores

Web Oficial de Scratch. <http://scratch.mit.edu/>

Conceptos de programación en Scratch.

<http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/files/file/ScratchProgrammingConcepts-v14.pdf>

Documentación oficial Scratch. <http://info.scratch.mit.edu/es/Languages>

Guía de inicio.

<http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/files/file/translated-docs/GuiaDeInicioScratch.pdf>

Guía de referencia. http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/files/file/RG_14_es.pdf

Preguntas más frecuentes. http://info.scratch.mit.edu/es/Support/Scratch_FAQ

Web Oficial para educadores Scratch. <http://info.scratch.mit.edu/Educators>

Comunidad de educadores de Scratch. <http://scratched.media.mit.edu/>

Webgrafía

[1] Habilidades de aprendizaje para el siglo XXI. Natalie Rusk, Mitchel Resnick, and John Maloney. Inglés. <http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/docs/Scratch-21stCenturySkills.pdf> (últ. acceso 22/04/2012)

[2] Scratch en la educación. Juan Carlos López García. Castellano. <http://edtk.co/Crwoh> (últ. acceso 22/04/2012)

[3] Scratch para los futuros científicos de la computación. David J. Malan. Castellano. <http://edtk.co/gFIKn> (últ. acceso 22/04/2012)

[4] Programación para todos. Mitchel Resnick, John Maloney y otros. Castellano. <http://edtk.co/eIRy8> (últ. acceso 22/04/2012)

[5] Programando con Scratch. MIT Media Lab. Castellano. <http://edtk.co/g2tRP> (últ. acceso 22/04/2012)

[6] Aprendiendo con Scratch. MIT Media Lab. Castellano. <http://edtk.co/31DWx> (últ. acceso 22/04/2012)

[7] Creando con Scratch. MIT Media Lab. Castellano. <http://edtk.co/yyLVj> (últ. acceso 22/04/2012)

[8] Aprender diseñando. Mitchel Resnick. Inglés.

<http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/docs/learning-by-designing.pdf> (últ. acceso 22/04/2012)

[9] Conferir a los niños la capacidad de crear y compartir medios de comunicación programables. Andrés Monroy-Hernández y Mitchel Resnick. Inglés.

http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/files/file/interactions_acm_2007_monroy-hernandez_resnick.pdf (últ. acceso 22/04/2012)