

Colección de actividades Aprender Conectados  
Nivel Primario

## Programación

# Salvemos la Tierra



## Actividad N° 8

# Autoridades

## **Presidente de la Nación**

Mauricio Macri

## **Jefe de Gabinete de Ministros**

Marcos Peña

## **Ministro de Educación, Cultura, Ciencia y Tecnología**

Alejandro Finocchiaro

## **Secretario de Gobierno de Cultura**

Pablo Avelluto

## **Secretario de Gobierno de Ciencia, Tecnología e Innovación Productiva**

Lino Barañao

## **Titular de la Unidad de Coordinación General del Ministerio de Educación, Cultura, Ciencia y Tecnología**

Manuel Vidal

## **Secretaria de Innovación y Calidad Educativa**

Mercedes Miguel

## **Subsecretario de Coordinación Administrativa**

Javier Mezzamico

## **Directora Nacional de Innovación Educativa**

María Florencia Ripani

ISBN en trámite

Este contenido fue producido por el Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación en el marco del Plan Aprender Conectados



## Introducción

El Plan Aprender Conectados es la primera iniciativa en la historia de la política educativa nacional que se propone implementar un programa integral de alfabetización digital, con una clara definición sobre los contenidos indispensables para toda la Argentina.

En el marco de esta política pública, el Consejo Federal de Educación aprobó, en 2018, los Núcleos de Aprendizajes Prioritarios (NAP) de Educación Digital, Programación y Robótica (EDPR) para toda la educación obligatoria, es decir, desde la sala de 4 años hasta el fin de la secundaria. Abarcan un campo de saberes interconectados y articulados, orientados a promover el desarrollo de competencias y capacidades necesarias para que los estudiantes puedan integrarse plenamente en la cultura digital, tanto en la socialización, en la continuidad de los estudios y el ejercicio de la ciudadanía, como en el mundo del trabajo.

La incorporación de Aprender Conectados en la Educación Primaria permite poner a disposición de los estudiantes y docentes, tecnología y contenidos digitales que generen nuevas oportunidades para reconocer y construir la realidad: abre una ventana al mundo, facilita la comunicación y la iniciación a la producción digital.

La alfabetización digital integra nociones sobre los lenguajes de las computadoras, sobre cuya base están contruidos los contenidos fundamentales de nuestra sociedad, particularmente aquellos ligados a los consumos culturales de niños y jóvenes. La programación y la robótica como el pensamiento computacional resultan relevantes para el aprendizaje: al comprender sus lenguajes y su lógica en la resolución de problemas, los alumnos se preparan para entender y cambiar el mundo.

En este marco, Aprender Conectados presenta actividades, proyectos y una amplia variedad de recursos educativos para orientar la alfabetización digital en la educación obligatoria en todo el país. La actividad que se presenta a continuación y el resto de los recursos del Plan son un punto de partida sobre el cual cada docente podrá construir propuestas y desafíos que inviten a los alumnos y alumnas a disfrutar y construir la aventura del aprender.

María Florencia Ripani  
Directora Nacional de Innovación Educativa

# Objetivos generales

## Núcleos de Aprendizajes Prioritarios

### Educación Digital, Programación y Robótica – Nivel Primario

Ofrecer situaciones de aprendizaje que promuevan en los estudiantes:

- La formulación de problemas simples y la construcción de estrategias para su resolución, incluyendo su descomposición en pequeñas partes, utilizando secuencias ordenadas de instrucciones, valiéndose de la creatividad y experimentando con el error como parte del proceso.
- La integración de recursos digitales variados en el desarrollo de actividades creativas, interactivas y multimedia, incluyendo el diseño de interfaces simples e incorporando conceptos básicos de programación.
- El diseño de narrativas que combinen diversos lenguajes y medios digitales y permitan construir conocimiento en un marco lúdico y creativo.
- El trabajo colaborativo y solidario mediado por TIC para la resolución de problemas, favoreciendo el intercambio de ideas, y la comunicación de forma clara y secuenciada de las estrategias de solución.
- El reconocimiento, la valoración y la aplicación de los derechos de propiedad intelectual — incluyendo el manejo general de distintos tipos de licencias— sobre las propias producciones digitales y las de otros.

## Objetivos específicos

Esta actividad permitirá introducir el entorno de programación visual Scratch y está orientada a desarrollar conocimientos iniciales vinculados con los siguientes objetivos de aprendizaje:

- Reconocer el entorno gráfico de Scratch.
- Reconocer los objetos, sus características y el escenario con sus disfraces.
- Identificar bloques básicos para la incorporación de sonidos.
- Conocer y usar apropiadamente las estructuras de repetición y los condicionales.
- Profundizar en el concepto de evento con la inclusión de comunicaciones entre objetos.
- Identificar situaciones que demandan la creación de variables y usarlas correctamente.
- Profundizar el concepto de sensores con el uso del cronómetro.
- Identificar situaciones donde es necesario clonar objetos.

## Materiales y recursos



Computadoras



Scratch 2.0

## Desafío

Max y los ciberamigos tienen una misión muy especial: volando en la nave espacial, tienen que interceptar una lluvia de meteoritos para impedir que lleguen a la Tierra y la destruyan.

¡Hay solo 10 segundos para interceptar la mayor cantidad de meteoritos posibles!

Misión de los programadores: ubicar en el espacio a la nave donde viajan Max y los ciberamigos y crear una lluvia de meteoritos. El juego consiste en mover la nave a través del puntero del *mouse* (o del *mousepad*), para impedir que los meteoritos lleguen a la Tierra, tratando de interceptar la mayor cantidad en 10 segundos.



## Momentos de la actividad

### < Inicio >

#### Disparador

Conversar con los alumnos y las alumnas a qué juegos estuvieron jugando Max y sus ciberamigos: a las adivinanzas, a volar, a Preguntamos, al laberinto. Ahora Max y sus ciberamigos van a jugar a atrapar meteoritos con su nave espacial para evitar que lleguen a la Tierra y la destruyan.

El o la docente invita a los alumnos y alumnas a crear un videojuego donde la nave espacial tiene que atrapar meteoritos que llegan del espacio utilizando el programa Scratch.

*“¿Se acuerdan a qué les gusta jugar a Max y a sus ciberamigos?  
¿Qué otros juegos podrían jugar? ¿En qué planeta vivimos? ¿Cómo  
podemos cuidar el planeta Tierra? ¿Qué es un meteorito? ¿Alguna vez  
llegó a la Tierra uno? ¿Dónde? ¿Cuándo?”*



## <Desarrollo >

El o la docente invita a los alumnos y alumnas a pensar cómo realizar el proyecto para resolver el desafío propuesto.

La primera tarea es hacer una lista de pasos que se necesitan para cumplir con el desafío. En toda la actividad se intentará que, además de los objetivos específicos, se ponga en juego la creatividad, la cooperación entre pares y el aprendizaje a partir de la exploración y el error.



SUGERENCIAS

- Los alumnos y alumnas pueden trabajar en grupos de a dos para ayudarse mutuamente en los primeros pasos de la programación.
- Se les puede proponer realizar una ficha con una lista de los pasos a seguir, asignándoles un orden.

### Lista de instrucciones o pasos

El docente hace una puesta en común de lo que cada grupo sugiere. Se debería concluir que se necesitan los siguientes pasos:

1. Seleccionar el escenario de una galaxia y los tres objetos: nave, meteorito y Megatomi.
2. Crear una lluvia de meteoritos que tendrán que ser interceptados por la nave.
3. Calcular el tiempo del juego.
4. Programar la nave para que detecte cuando pasaron 10 segundos, que es el tiempo que dura el juego y dar mensaje de final del juego.
5. Fin del juego.

## Manos a la obra

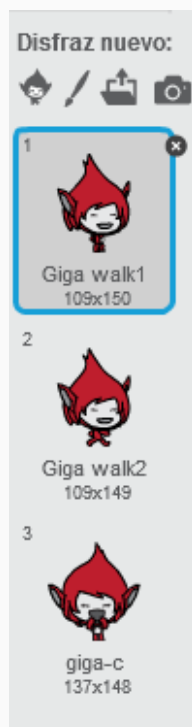
- 1 Seleccionar el escenario de una galaxia y los tres objetos: nave, Meteorito y Megatomi.

Para el fondo del escenario, descargar cualquier imagen de licencia libre de una galaxia en la computadora y subirla como fondo del escenario desde “cargar fondo desde archivo” o usar el fondo “stars” (estrellas) de la biblioteca de Scratch.

En objeto nuevo, seleccionar a Megatomi caminando, dibujar el objeto meteorito o seleccionar de la biblioteca el objeto “ball”(pelota) y dibujar una nave, o descargar cualquier imagen de una nave espacial con licencia libre o seleccionar “spaceship” (nave) de la biblioteca de Scratch. Asignarle a cada objeto sus nombres: “Megatomi”, “Meteorito”, “nave”.



A Megatomi dejarle dos disfraces de caminar y agregar un disfraz de Giga-c (que es el que está con una taza).

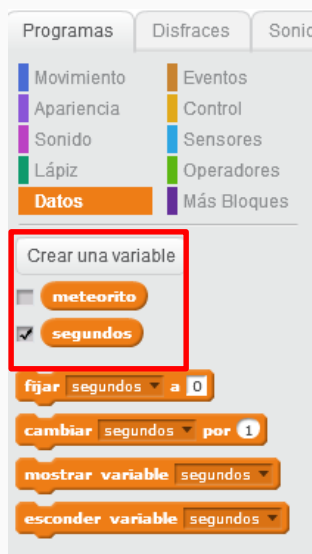


Recuerden guardar el juego. La primera vez les pedirá un título. Se sugiere “Salvemos a la Tierra”.



## 2 Crear una lluvia de meteoritos que tendrán que ser interceptados por la nave.

Para ello, tendrán que crear un clon del meteorito por cada un segundo, hacer que aparezca y que se mueva por el escenario de manera aleatoria, y que cuando toca la nave, se reproduzca un sonido, desaparezca y sume 1 a la cantidad de una variable, que cuente los meteoritos interceptados.



Comenzar por programar el objeto Meteorito, hay que crear dos variables: “**meteorito**” para contar la cantidad de meteoritos interceptados, y “**segundos**” será la que alojará el tiempo de duración del juego.

La **variable segundos** va a estar visible en el escenario y por eso tiene un tilde a su izquierda.

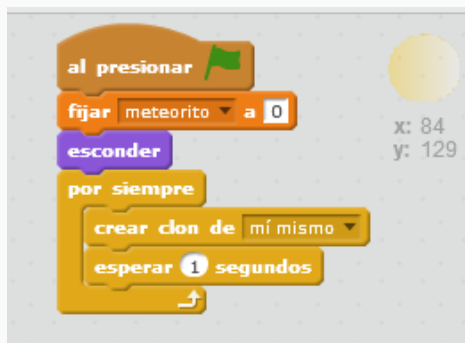
La **variable meteorito** no va a estar visible en el escenario, por eso está destilada.

Como en el juego habrá una lluvia de meteoritos, en vez de incorporar muchos objetos meteoritos, se puede **crear clones**.

¿Qué es un clon? Es una copia exacta de un objeto, de forma que: ejecuta el mismo programa, cuenta con los mismos componentes (disfraces, variables, sonidos) y cuando se crea, aparece en la misma posición en la que está el objeto.

Por lo tanto, el primer bloque debe ser para su ubicación (posicionamiento en el escenario).

El **objeto Meteorito** tendrá que ser programado para que, al comenzar con bandera verde, ponga en 0 la variable meteorito, se esconda, y “por siempre” cree un clon cada un segundo.



Como el **objeto Meteorito** crea clones, estos se programan como un programa más del objeto **Meteorito**, pero con el bloque de la categoría “Control”, **al comenzar como clon**.

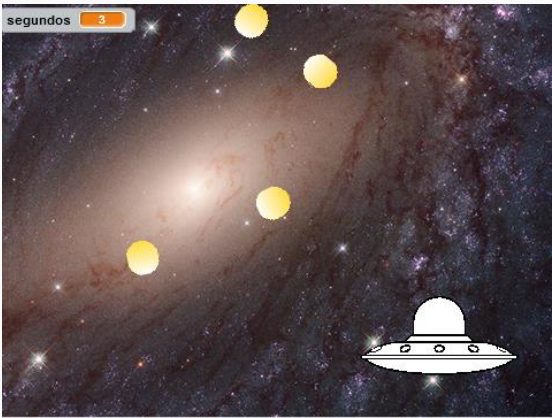
En este programa, se ubica el clon en una posición aleatoria de **x** (eje horizontal) que toma valores entre -180 y 180, y valores aleatorios para **y** (eje vertical) entre 0 y 180, o sea que siempre los clones van a aparecer de la mitad del escenario hacia arriba. Y también aparecen con una dirección (orientación) aleatoria entre -180° y 180°.

Luego el clon va a entrar en un **ciclo** (repetición) moviéndose 10 pasos por el escenario y rebotando si toca un borde **hasta que toque la nave**.

```
repetir hasta que ¿tocando nave ?
  mover 10 pasos
  rebotar si toca un borde
```

Y cuando toca la nave, el clon sale del ciclo de movimiento, reproduce un sonido, cambia la variable “meteorito” por “1”, es decir, suma un punto al valor que tiene la variable “meteorito”, y se borra el clon (desaparece ese meteorito clonado del escenario). La **programación completa del objeto Meteorito al comenzar como clon** es la siguiente:

```
al comenzar como clon
  ir a x: número al azar entre -180 y 180 y: número al azar entre 0 y 180
  apuntar en dirección número al azar entre -180 y 180
  mostrar
  repetir hasta que ¿tocando nave ?
    mover 10 pasos
    rebotar si toca un borde
  tocar sonido laser2
  cambiar meteorito por 1
  borrar este clon
```



Cuando el objeto Meteorito recibe el mensaje “fin”, finaliza el juego: se **detienen todos los programas del objeto Meteorito** (no los programas de los otros objetos).

```
al recibir fin
  detener otros programas en el objeto
```

x: 84  
y: 129

3 Calcular el tiempo del juego.

Incorporar el cronómetro y la variable “segundos” para mostrar el valor de los segundos sin decimales en la **programación del objeto nave**.

En este juego, se incorpora el factor tiempo: dura 10 segundos. Para ello se creó en el paso anterior la **variable “segundos”**, que va a estar visible en el escenario. Esta variable va a mostrar los segundos que indica el cronómetro, pero sin los decimales.

Cuando se inicia el juego, los segundos deben comenzar en 0 y también hay que **inicializar** el cronómetro de la categoría “Sensores” con el bloque

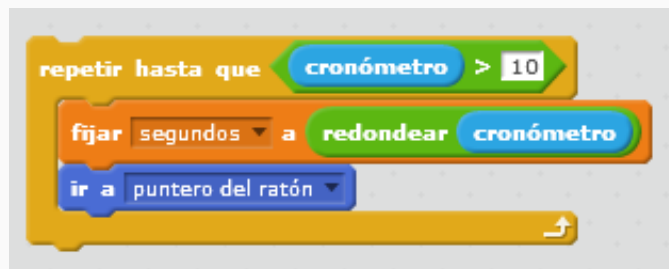
A blue Scratch block with the text "reiniciar cronómetro" in white.

4 Programar la nave para que detecte cuándo pasaron 10 segundos, que es el tiempo que dura el juego, y dar mensaje de final del juego.

El control del tiempo del juego se realiza a través de la programación **del objeto nave**, que en forma continua va a actualizar la **variable segundos** con el valor entero (sin decimales) del cronómetro hasta que el valor del cronómetro sea mayor a 10.

Necesitarán un ciclo repetir hasta que el cronómetro sea mayor a 10:

- Le asigna a la **variable “segundos”** el valor redondeado (sin decimales) del sensor cronómetro.
- Se mueve a la posición del puntero del *mouse*. El efecto de esto es que la nave sigue al *mouse*. En el juego funciona como que la nave se maneja con el *mouse* (o *mousepad*).



¿Qué sucede cuando el **cronómetro** es menor o igual a 10?

Finaliza el bucle “repetir”, vuelve a asignarle a la **variable “segundos”** el valor entero del cronómetro y **envía mensaje “fin”**.

Así queda la programación completa del **objeto nave**:



## 5 Fin del juego.

Cuando el juego termina y Megatomi recibe el mensaje “fin”, se reproduce una música. Megatomi camina por el escenario y comunica (dice) la cantidad de meteoritos que logró interceptar el jugador.

Al iniciar el juego, Megatomi está escondido, por lo tanto hay que programar al objeto con el siguiente bloque:



Cuando Megatomi recibe el **mensaje “fin”** suceden dos acciones:

a) Reproduce 3 veces un bucle musical, que en este caso es el archivo **triumph** de la biblioteca de Scratch.



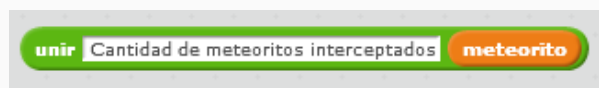
b) Se muestra en el escenario, camina, espera, cambia de disfraz y dice “¡Tiempo finalizado!” y “Cantidad de meteoritos interceptados:...”.

Para armar la frase que dice Megatomi es necesario unir el texto con el contenido de la **variable “meteorito”**.

Para eso, en categoría “Operadores”, seleccionar el bloque “unir”:



Así se pueden unir dos frases o, como en este caso, una frase con el contenido de la variable “meteorito”:



La programación del objeto “Megatomi” cuando recibe mensaje “fin” quedaría así:



Al finalizar, el juego se verá así:





Hemos construido nuestro proyecto en Scratch. Ahora llegó el momento de ejecutarlo, presionando la bandera verde del escenario.

Y para verlo en la pantalla completa, en el ícono cuadrado azul y luego la bandera verde.



## ¡Nuestro desafío está terminado!

Para cerrar la actividad, los alumnos y alumnas jugarán en grupos, compartiendo con sus compañeros sus proyectos.

Podrán intercambiar las computadoras para comparar sus códigos, probando si funciona bien la programación y ayudarse entre ellos para corregir o mejorar el juego. Pueden jugar con compañeros y compañeras de otros grados.

### Para seguir aprendiendo

Se ha creado un videojuego para interceptar meteoritos para que no lleguen al planeta Tierra con un control del tiempo.

Ahora los alumnos y las alumnas están en condiciones de seguir avanzando, modificando el tiempo de juego, la cantidad de meteoritos (clones) que aparecen, e incorporando otros objetos/obstáculos, como por ejemplo una nave enemiga, que al interceptarla, haga ganar más puntos.