

PRÁCTICA 1. LENGUAJES DE PROGRAMACIÓN. EL ENTORNO SCRATCH.

TEORÍA.

Un **lenguaje de programación** es un conjunto de “palabras” que sirve para crear programas o software (por ejemplo, juegos). Con un lenguaje de programación generamos **código fuente**, que es el conjunto de instrucciones ordenadas que vamos a ejecutar. Dicho código fuente se convierte en un **fichero ejecutable** mediante un **compilador o intérprete**, que es el que convierte el código fuente en un programa que el ordenador puede comprender.

En la imagen, un ejemplo del lenguaje “C”.



Scratch es un lenguaje de programación muy fácil de utilizar puesto que utiliza un entorno gráfico muy sencillo de manejar. En Scratch, el entorno de programación hará de intérprete de los programas que hagamos y nos devolverá la ejecución de los mismos.

PRÁCTICA.

1. Vamos a trabajar sobre la web de Scratch. Para ello abre un navegador de Internet y teclea la siguiente url: <https://scratch.mit.edu/>



2. Lo primero que podemos ver es un menú superior con las siguientes opciones:



2.1. La opción **Crear** nos permite directamente entrar a programar con Scratch.

2.2. La opción **Explorar** nos permite ver los proyectos de otras personas. Podremos ver el código fuente y también jugar con ellos.

2.3. Para poder almacenar nuestros proyectos en la nube tenemos que registrarnos en la página. Vamos a hacerlo. Para ello haz clic sobre la opción **Únete a Scratch**. **(NECESITAS TENER UNA CUENTA DE CORREO ELECTRÓNICO. SI NO LA TIENES AÚN, CONSULTA A TU PROFESOR)**.



Únete a Scratch

Es fácil (y gratis) registrar una cuenta Scratch.

Elige un nombre de usuario en Scratch

Elija una contraseña

Confirmar contraseña



1 2 3 4 

2.3.1. Elige un nombre de usuario y una contraseña, y apúntala en un lugar seguro. **NO compartas tu contraseña con nadie**. Si el usuario ya existe, debes elegir otro. La contraseña hay que escribirla dos veces. Pulsa **Siguiente**.



Únete a Scratch

Tus respuestas a estas preguntas se guardarán confidencialmente.

Por qué pedimos estos datos ?

Mes y año de nacimiento

Género Masculino Femenino

País



1 2 3 4 

- 2.3.2. Rellena los siguientes datos (lee primero por qué se te piden dichos datos). Decide con tu compañero si vais a meter datos verdaderos o falsos, y **escribe justificadamente lo que habéis decidido en tu cuaderno**. Pulsa **Siguiente**.



Únete a Scratch

Ingresa su dirección de correo electrónico y le enviaremos un mensaje de confirmación a su cuenta.

Correo electrónico

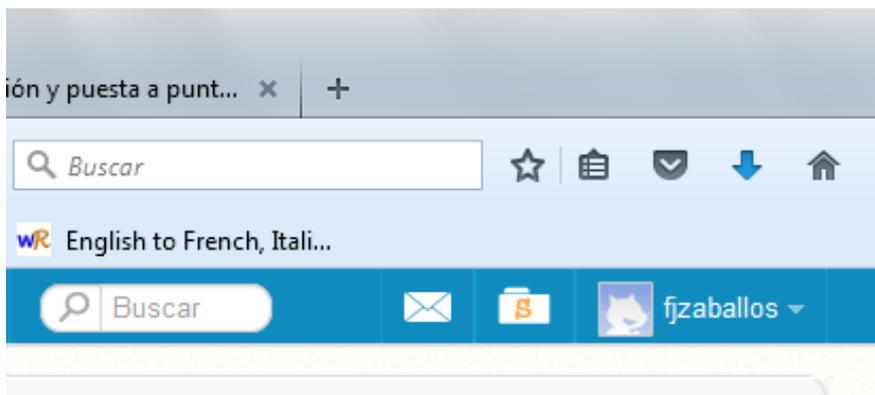
Confirmar su correo electrónico



1 2 3 4

- 2.3.3. Mete tu cuenta de correo electrónico dos veces. Pulsa **Siguiente**.

- 2.3.4. Ya estás registrado. Pulsa en **OK, Vamos**.



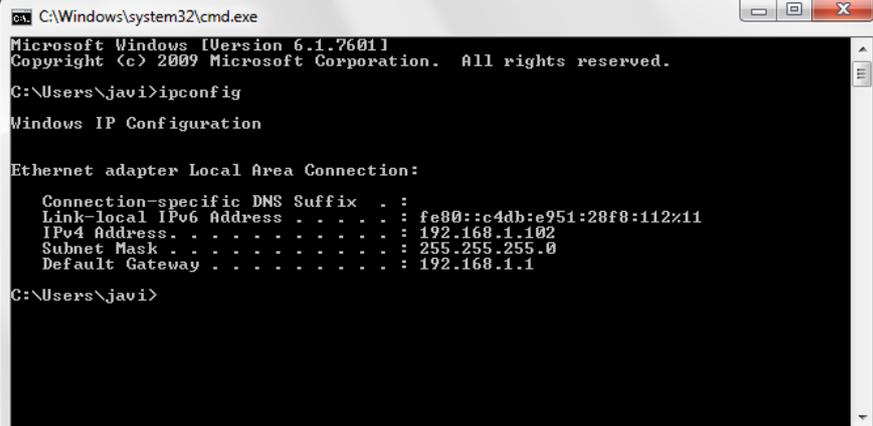
3. Si te fijas en la parte derecha del menú, aparece tu usuario. Cuando salgas de Scratch, tendrás que pulsar el botón **Ingresar** y meter tu usuario y password para volver a meterte.
4. Te llegará un email a tu cuenta de correo electrónica de Scratch para que confirmes que la dirección de correo electrónico es válida. Pincha en el enlace de dicho correo para confirmar tu dirección de correo.
5. Avisa a tu profesor de que has finalizado la práctica.
6. Si te ha sobrado tiempo, pulsa la opción **Explorar** e investiga sobre proyectos ya hechos con Scratch.

PRÁCTICA 2. LA INTERFAZ DE SCRATCH. DISFRACES. MI PRIMER PROGRAMA EN SCRATCH.

TEORÍA.

Una **interfaz** de usuario (del inglés “**interface**”) es el medio por el cual el usuario se comunica con una máquina. Hay dos tipos de interfaces de usuario:

- Interfaz de línea de comandos (CLI – Command Line Interface): el usuario se comunica con la máquina mediante una línea de comandos donde introduce las instrucciones al ordenador. Por ejemplo, MS-DOS.



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\javi>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::e4db:e951:28f8:112x11
    IPv4 Address. . . . . : 192.168.1.102
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

C:\Users\javi>
  
```

- Interfaz gráfica de usuario (GUI – Graphical User Interface): el usuario interactúa con la máquina mediante un entorno visual, como por ejemplo, Windows.

Scratch ofrece una interfaz gráfica muy fácil de manejar. Vamos a ver algunas de sus funciones en el apartado de la práctica.

Un **programa** es una secuencia de instrucciones que le damos al ordenador para realizar una tarea. Sería algo parecido a una receta de cocina, en la que le vamos diciendo al ordenador los “pasos” o instrucciones para ejecutar dicha tarea.

RECETA: Croquetas de Quinoa y Zanahoria

Ingredientes

- 1 taza de quinoa
- 2 zanahorias rallada fina
- ½ cebolla
- 2 huevos
- alcaparras
- sal, pimienta
- sésamo tostado
- aceite de oliva
- Perejil

Preparación:

1. Lavar la quinoa y cocer en 1 ½ taza de agua hirviendo
2. Cortar la cebolla. Rehogar
3. Rallar la zanahoria fina
4. Cortar las hojas de perejil
5. Mezclar los ingredientes en un bol. Salpimentar
6. Añadir 2 huevos batidos
7. Calentar el aceite. Formar las croquetas con las manos y freír
8. Poner sobre un plato con papel absorbente

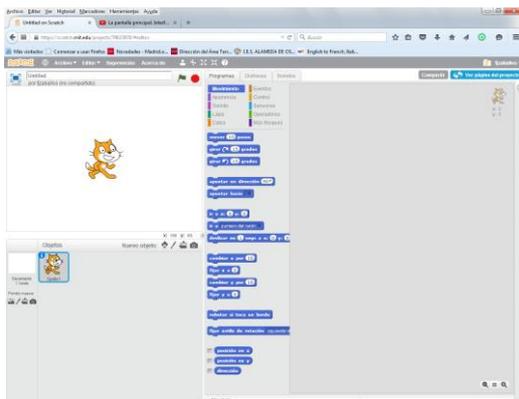


PRÁCTICA.

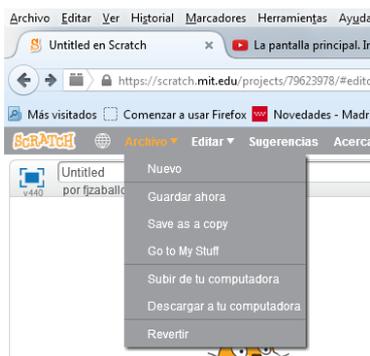
1. Vamos a trabajar sobre la web de Scratch. Para ello abre un navegador de Internet y teclea la siguiente url: <https://scratch.mit.edu/>.
2. Mete tu login y password.
3. Pulsa el botón “Crear”.



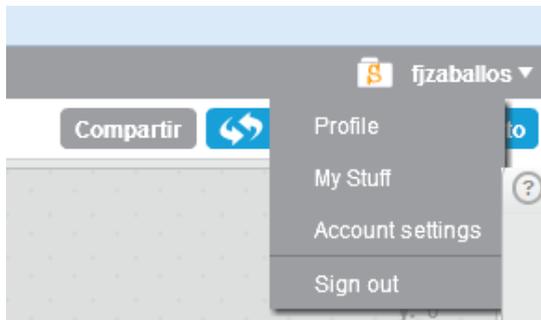
4. Va a aparecer la siguiente ventana. Es la interfaz para crear un proyecto.



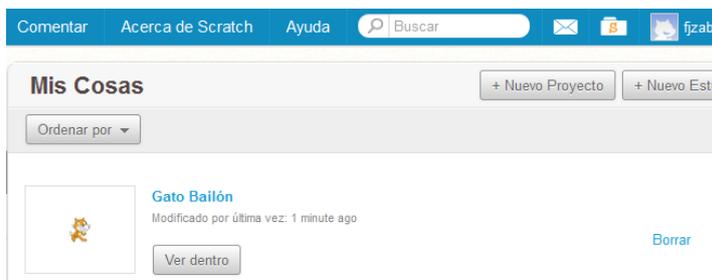
5. Veamos por partes dicha interfaz. Arriba a la izquierda encontramos el menú. Una de las entradas es “Archivo”. Las opciones de este menú nos ofrecen crear un nuevo proyecto (opción “Nuevo”), guardar lo que estamos haciendo (opción “Guardar ahora”), guardar lo que estamos haciendo como copia (opción “Save as a copy”), subir un proyecto que he guardado previamente en mi PC (opción “Subir de tu computadora”) o descargar un proyecto a mi PC (opción “Descargar a tu computadora”).



- 5.1. Vamos a ponerle al proyecto el nombre “Gato Bailón” (en el rectángulo debajo del menú “Archivo”, donde pone “Untitled”, sustitúyelo por dicho nombre.
- 5.2. Pulsa **Archivo** → **Guardar ahora** para guardar el fichero.
- 5.3. Comprueba que el proyecto se ha guardado correctamente. Para ello, pincha en tu nombre de usuario (arriba a la derecha), y elige la opción “My Stuff” (Mis Cosas).



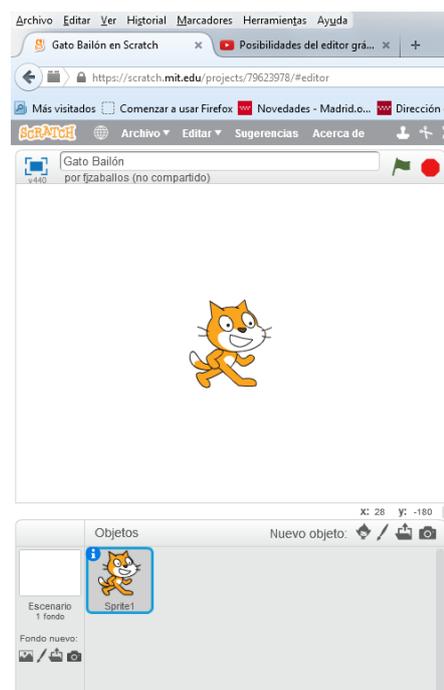
- 5.4. Aparecerá una ventana en la que tendrás un solo proyecto llamado “Gato Bailón”.



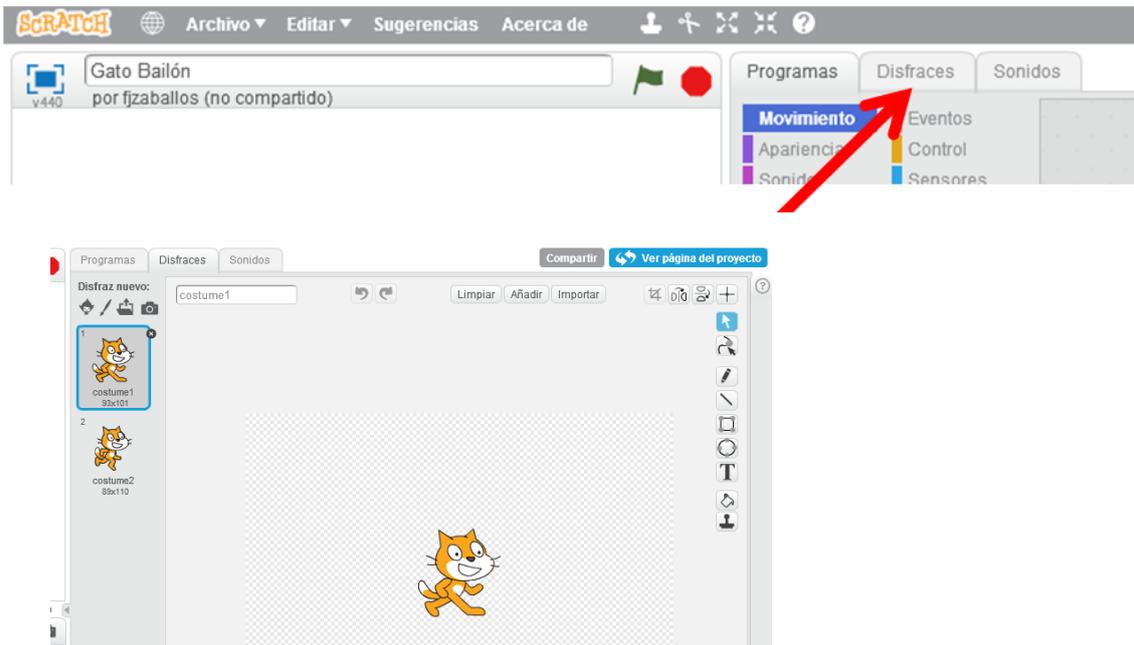
- 5.5. Pulsa “Ver dentro” para volver al proyecto.

6. Debajo del nombre que acabamos de poner vemos nuestro gato, que es el objeto actualmente seleccionado, en fondo blanco, y debajo, en fondo gris, aparecen todos los objetos que va a tener nuestro programa o juego.

- 6.1. Vamos a cambiarle el nombre a nuestro objeto para que se llame “Gatito”. Para ello, haz clic con el botón izquierdo sobre la “i” del objeto “Sprite1”, sustituye el nombre por el de “Gatito”, y pulsa la flecha azul a la izquierda del dibujo del gato.



7. Ahora vamos a ver los disfraces de “Gatito”. Para ello en la ventana principal elige la pestaña “Disfraces” (arriba, en el centro). Si pulsas alternativamente en los disfraces 1 y 2 (“costume1” y “costume2”) verás que alternando dichos disfraces, parece que el gato tiene movimiento.



8. Vamos por fin a escribir nuestro primer programa. Haz clic en la pestaña “Programas” (al lado de “Disfraces”).



- 8.1. Dentro de esta pestaña aparecerán distintas instrucciones relacionadas por diez categorías. Elige la categoría “Eventos”, y arrastra la instrucción “Al presionar *bandera verde*” a la ventana de la derecha.



- 8.2. Pincha ahora en la categoría “Movimiento”, y arrastra la instrucción “Mover 10 pasos” a la derecha, justo debajo de la instrucción anterior.



- 8.3. Pincha en la categoría “Apariencia”, y arrastra la instrucción “Cambiar disfraz a costume2” debajo de la anterior.



- 8.4. Bien, le vas cogiendo el tranquillo. Haz lo necesario para escribir el siguiente programa:



- 8.5. Ejecuta el programa (pulsando la bandera verde encima del gato). Notarás que el gato se mueve, pero no se ve el cambio de disfraz.

- 8.6. Tendrás que hacer la siguiente modificación en el programa para que se vea cómo se mueve tu gato.



PRÁCTICA 3. DISFRACES. OBJETOS.

TEORÍA.

Podemos hablar de varios modos de programar o hacer programas:

- Programación estructurada: es aquella en la que el programa tiene una secuencia de instrucciones ordenada (como la receta de cocina que vimos en la práctica anterior).
- Programación modular: en este tipo de programación, dividimos el programa en *subprogramas* (también llamados **funciones** y/o **procedimientos**) para resolver el problema.
- Programación orientada a objetos: en este tipo de programación, definimos **clases** con ciertos **atributos** y **comportamiento**. Los objetos serían la representación concreta de una clase. Scratch es un lenguaje que emplea objetos, a los que definimos comportamientos mediante los programas. Por ejemplo, en la práctica anterior definimos el comportamiento de nuestro objeto "Gatito" mediante un programa que lo hacía desplazarse a izquierda y a derecha. En esta práctica vamos a practicar con más de un objeto.

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto con los siguientes requisitos:
 - 2.1. El proyecto ha de tener al menos dos objetos (si no sabes añadir objetos, investiga cómo hacerlo). Puedes usar el editor gráfico para crear objetos.
 - 2.2. Dichos objetos deben moverse al pulsar la bandera verde.
 - 2.3. Debe haber al menos un escenario de fondo (si no sabes modificar el fondo, investiga cómo hacerlo).
 - 2.4. Cuanto más logrado esté el programa, mejor puntuado será.
 - 2.5. Ejemplos: podemos hacer que varios bailarines dancen en una pista de baile, que varios futbolistas se muevan en torno a un balón, etc. Elige un tema que te guste para la práctica.
3. Guarda el proyecto y muéstraselo a tu profesor.

PRÁCTICA 4. BUCLES.**TEORÍA.**

Los **bucles** son bloques de control que nos permiten repetir parte del programa. En esta práctica vamos a ver dos:

- El bucle repetir: repite el código un número de veces que programamos.
- El bucle infinito: es un bucle que repite el programa hasta que lo paremos manualmente.

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Abre el proyecto “Gato Bailón” que tienes en “Mis Cosas”.
3. Guarda una copia del proyecto (**Archivo** → **Save as a Copy**).



4. Ve a “Mis Cosas” y pulsa el botón “Ver dentro” del programa “Gato Bailón Copy”.
5. Renombra el proyecto como “Gato Bailón Bucle_Repetir”.
6. Vamos a crear un bucle que repita el código de avance y cambio de disfraz cinco veces. Para ello, en la pestaña “Programas”, selecciona la categoría “Control”. Ahora arrastra el icono que marca “Repetir 10” a la ventana donde está el programa que teníamos.



- 6.1. Mete el código que queremos repetir dentro del bucle, y borra el resto (con el botón de las tijeras, arriba en el centro). Cambia el 10 por 5.



6.2. Ahora vamos a duplicar este código. Pulsa con el botón derecho sobre “repetir 5” y duplica el código poniéndolo debajo. Cambia el “-90” por “90” para dejar al gato orientado a la derecha al final del programa.



6.3. Si ejecutas el programa, verás que hemos hecho lo mismo que con el anterior pero con menos líneas de código. Cambia el “5” del “repetir” por otros valores, por ejemplo “20”, para ver el efecto del bucle.

6.4. Guarda el proyecto (**Archivo** → **Guardar ahora**).

7. Vamos a modificar el programa para incluirle un bucle infinito, es decir, un bucle para que el programa no acabe nunca.

7.1. Guarda una copia del proyecto “Gato Bailón Bucle_Repetir”.

7.2. Ve a “Mis Cosas” y pulsa el botón “Ver dentro” del programa “Gato Bailón Bucle_repetir Copy”.

7.3. Renombra el proyecto como “Gato Bailón Bucle_infinito”.

7.4. En la pestaña “Programas”, selecciona la categoría “Control”. Ahora arrastra el icono que marca “por siempre” a la ventana donde está el programa que teníamos, justo debajo de la instrucción “Al presionar bandera verde”. Así, metemos el código dentro de este bucle. Si has hecho todo correctamente, tendrás el siguiente programa:



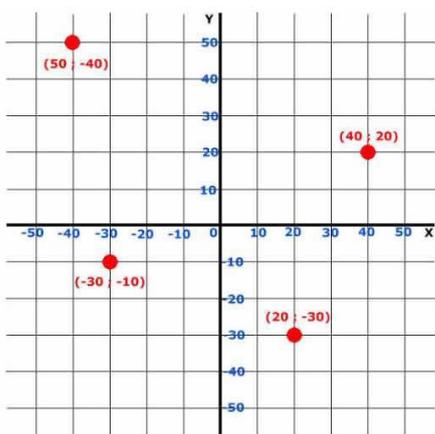
7.5. Si ejecutas el programa, verás que el gato ¡¡se ha vuelto loco!! Ahora no para de moverse. Lógicamente, ya que hemos programado un bucle infinito. Guarda el proyecto.

8. Muestra los dos proyectos a tu profesor.

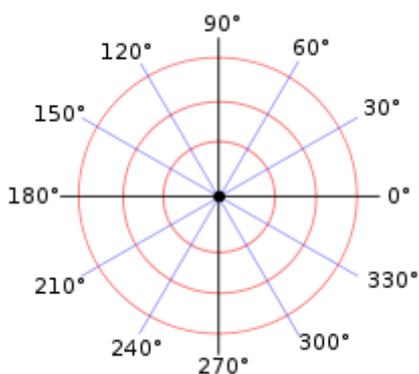
PRÁCTICA 5. MOVIMIENTOS EN COORDENADAS POLARES Y CARTESIANAS.**TEORÍA.**

La pantalla de Scratch es una pantalla de 360 x 480 píxeles. La coordenada (0,0) en Scratch está justo en el centro de la pantalla. Cada personaje en Scratch puede moverse de dos formas:

- Por coordenadas cartesianas: según los ejes x e y.



- Por coordenadas polares: según la orientación del personaje y su movimiento.

**PRÁCTICA.**

1. Entra en la web de Scratch con tu login y password.

1ª PARTE DE LA PRÁCTICA

2. Vamos a comprender cómo se utilizan las **coordenadas cartesianas**. Crea un nuevo proyecto y llámalo "nave cartesiana".
3. Borra al gato (botón derecho sobre el objeto gato → Borrar).



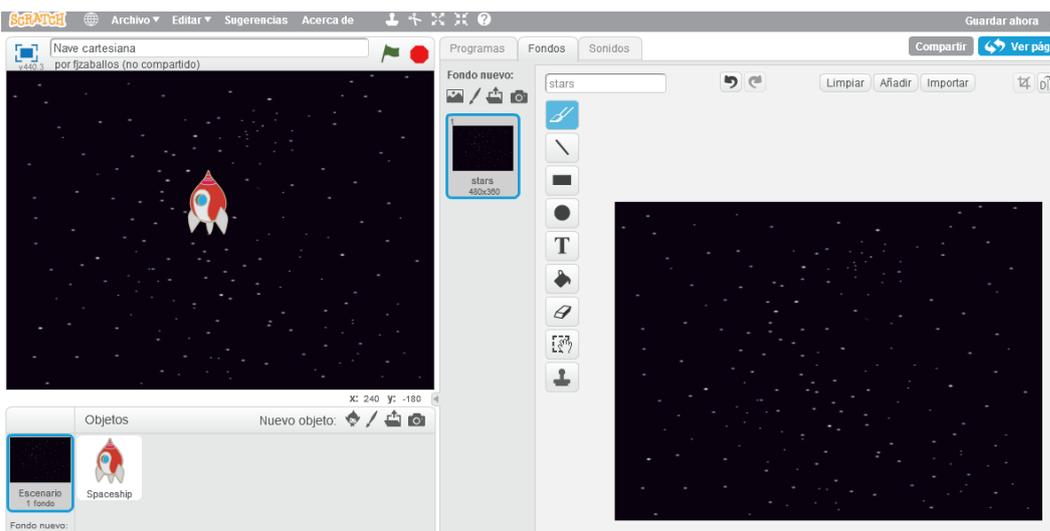
4. Inserta un nuevo objeto de la biblioteca. Elige el tema “espacio” (a la izquierda) y la nave espacial (“spaceship”) a la derecha. Dale doble clic para elegirla.



5. El objeto es muy grande. Lo vamos a reducir con el icono “encoger” (shrink). Selecciona “encoger” y haz clic en la nave varias veces para reducir su tamaño.



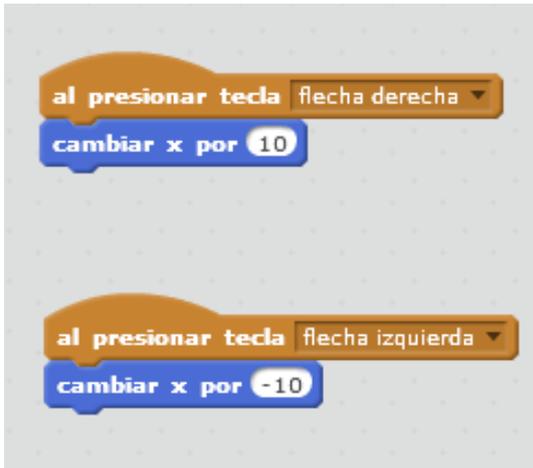
6. Vamos a elegir un escenario apropiado. Pulsa en “Fondo nuevo de la biblioteca”, elige el fondo “Stars”. Ahora tienes dos escenarios. Borra el fondo blanco con botón derecho sobre él.



7. Vamos a programar la nave para que se mueva a izquierda, derecha, arriba y abajo con los cursores. Pincha la nave, y la pestaña “Programas”.
8. Para que la nave se mueva hacia la derecha, escribimos el siguiente programa:



9. Para que la nave se mueva hacia la izquierda, escribimos el siguiente programa:



10. Haz lo mismo para las flechas arriba y abajo con la coordenada “y”.

11. Ya tienes una nave que se mueve de forma cartesiana. Prueba los distintos movimientos de los cursores.

12. Guarda el proyecto.

2ª PARTE DE LA PRÁCTICA

13. Vamos a comprender ahora cómo se utilizan las **coordenadas polares**. Duplica el proyecto “nave cartesiana” y renómbralo como “nave polar”.

14. Vamos a reprogramar las acciones tomadas al pulsar las flechas. Programa que la nave gire 15° a la derecha cuando pulsemos flecha derecha, y 15° a la izquierda cuando pulsemos flecha izquierda.

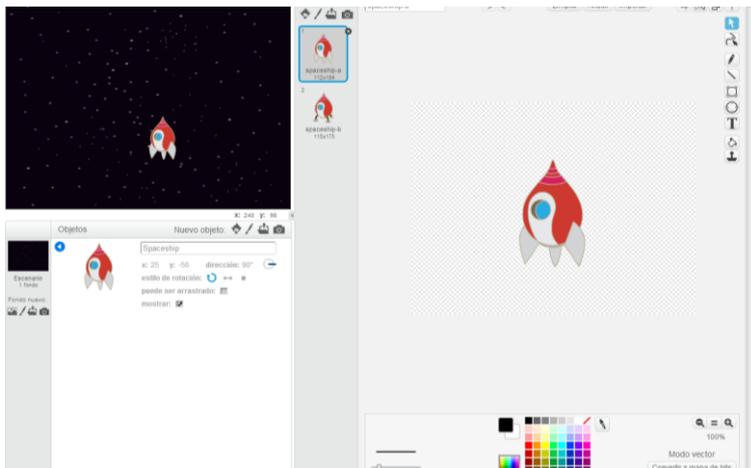
15. Programa también que cuando pulsemos flecha arriba, la nave se mueva 10 pasos, y cuando pulsemos flecha abajo, la nave retroceda 10 pasos (se mueva -10 pasos).

16. Debes tener ahora un programa así:

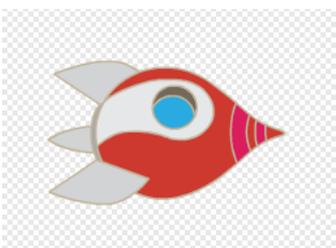


17. Prueba los distintos movimientos ahora. Verás que la nave no avanza bien. Si pulsas la tecla “i” azul del objeto “nave espacial”, verás que aunque la nave mira hacia arriba, su orientación es a 0°, es decir, hacia la derecha.

18. Vamos a rotar el disfraz de la nave. Pulsa la nave, y luego la pestaña “disfraces”.



19. Vamos a girar la nave con el editor gráfico. Pincha en el objeto nave. Después pulsa en el círculo que hay en la parte de arriba, y gira la nave 90° a la derecha, para que su orientación en el programa coincida con su orientación en el dibujo.



20. Prueba ahora los movimientos y verás que has reparado el problema.

21. Guarda el proyecto.

22. Muestra los dos proyectos a tu profesor.

PRÁCTICA 6. Interacción entre objetos. Condicionales

TEORÍA.

Los objetos de Scratch pueden interactuar entre ellos. Podemos programar que al tocar uno a otro, cada uno haga una determinada acción. También podemos hacer que un objeto envíe un mensaje a otros objetos, y que un objeto haga algo al recibir un mensaje.

Para la práctica nos vamos a basar en la función “si”. Nos la podemos encontrar en dos formatos en la pestaña “Programas”, categoría “control”.



La función “si” implica una condición, “si esto se cumple, entonces haz ...”. El programa evalúa si la condición es verdadera o no. Si lo es, ejecuta lo que pongamos dentro de la estructura condicional. En el caso de la segunda estructura, el programa hará dos acciones distintas en función de si la condición es verdadera o falsa.

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto, llámale “Ratón que te pilla el gato”.
3. Haz al gato más pequeño.
4. Agrega un nuevo objeto “ratón” (Mouse1) y hazlo más pequeño.
5. Vamos a generar un movimiento aleatorio en el ratón. Para ello, escribe el siguiente programa para el ratón. El bloque verde “número al azar entre 0 y 360” está en la categoría “Operadores”. Pruébalo.



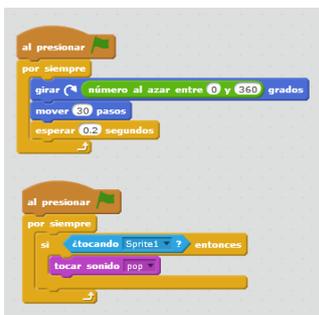
6. Habrás visto que el ratón va muy deprisa. Vamos a ponerle un retardo para que el ratón vaya un poquito más despacio. Agrega la instrucción “esperar 0.2 segundos”. Este tiempo va a agregar mayor o menor dificultad al juego, pues implica un movimiento más rápido o más lento del ratón. Prueba a ponerle más o menos tiempo y deja el tiempo que te parezca más adecuado.



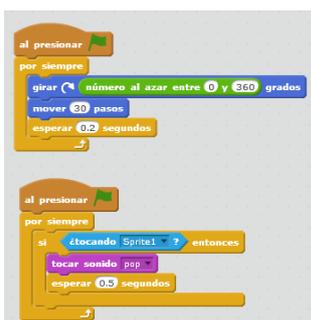
7. Al gato le vamos a poner un movimiento que se oriente hacia el puntero del ratón (del periférico, no del objeto). Escribe el siguiente programa para el gato. Puedes hacer que el gato se mueva más rápido o más lento cambiando el “3”.



8. Vamos a hacer ahora que cuando el gato toque al ratón, éste emita un sonido. Para ello, incluye la siguiente programación al ratón. Pruéballo.



9. Habrás notado que suenan muchos “pop”. Vamos a ponerle un tiempo de espera detrás de la instrucción “tocar sonido” para que no suene todo el rato. Incluye la instrucción “esperar 0.5 segundos”. Pruéballo.



10. Muestra el proyecto a tu profesor.

TAREA DE AMPLIACIÓN.

11. Baja un sonido de ratón de Internet. Impórtalo con el editor de sonido (pestaña al lado de los disfraces) y sustituye el sonido “pop” por el de un ratón. Puedes editar el sonido una vez subido con el editor de sonido que incluye Scratch.
12. Busca un escenario apropiado para el juego y cámbialo.

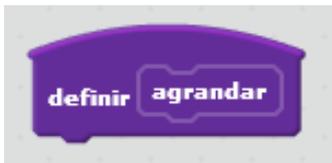
PRÁCTICA 7. Modularidad.

TEORÍA.

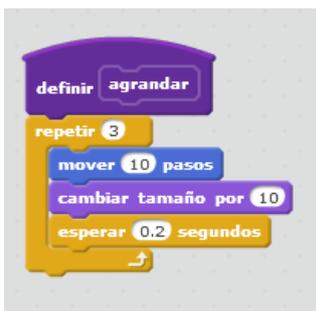
Una de las funciones importantes en todos los lenguajes de programación es la **modularidad**. La modularidad consiste en construir “instrucciones” nuevas que contengan otras instrucciones en su interior. Estas nuevas instrucciones en algunos lenguajes de programación se llaman **funciones** o **procedimientos**, en Scratch se les llama bloques. Estos bloques, funciones o procedimientos pueden utilizar unos **parámetros** de entrada, es decir, unos datos de entrada que se utilizan dentro de las instrucciones del bloque.

PRÁCTICA.

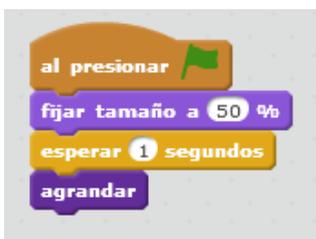
1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto, llámale “Agrandar al gato”.
3. Pincha en la pestaña “programas”, categoría “Más bloques”.
4. Con el botón “Crear un bloque” vamos a crear este conjunto de instrucciones para agrandar al gato. Pulsa el botón.
5. Ponle un nombre al bloque, por ejemplo, “agrandar”.



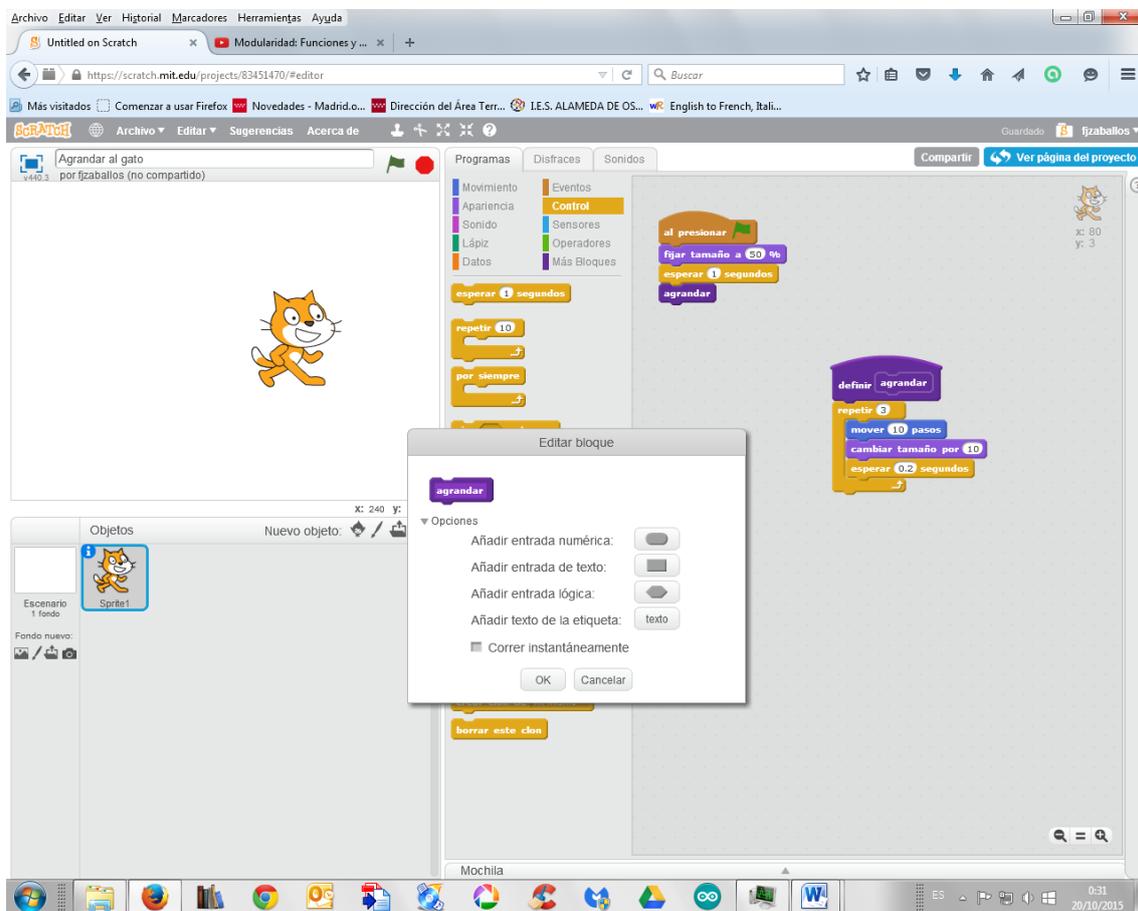
6. Incluye las siguientes instrucciones a tu bloque para agrandar al gato en 30, mientras va avanzando.



7. Programa en el objeto “Sprite1” el siguiente código. El gato empieza al 50% de su tamaño, esperamos 1 segundo y ejecutamos la función “agrandar”.



8. **Parámetros:** vamos a incluir un parámetro de entrada en nuestra función “agrandar”, para no fijar el dato en la función, sino como dato de entrada en la función. Haz clic con botón derecho en “definir agrandar”, y elige la opción “editar”.



9. Elige la opción “Añadir entrada numérica”. Aparece a la derecha el parámetro de entrada “number1”.
10. Sustituye el “3” de “repetir 3” por este parámetro de entrada.



11. Por último, mete distintos números en el programa del gato y prueba el efecto.



12. Muestra el proyecto a tu profesor.

PRÁCTICA 8. VARIABLES. PETICIÓN DE DATOS AL USUARIO.

TEORÍA.

Las **variables** nos permiten almacenar datos dentro del ordenador mientras el programa está funcionando. El valor de dichos datos puede variar durante la ejecución del programa. Las variables se caracterizan por una **etiqueta** (nombre que le damos a dicha variable, por ejemplo, *puntos*) y por un **valor** (por ejemplo, los puntos que vamos sumando en el juego). Es habitual que las variables puedan ser de varios tipos:

- **Numéricas**: el valor que contiene es un número.
- **Texto**: el valor que contiene es un carácter o cadena de caracteres.
- **Boolean**: el valor que contiene es “verdadero” (1) o “falso” (0).

En Scratch las variables son solo de tipo numérico. Existen también **listas**, que almacenan varias cadenas de caracteres.

En la práctica veremos cómo se utilizan las variables en Scratch.

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Abre el proyecto “Ratón que te pilla el gato” y crea una copia. Renómbrala como “Ratón que te pilla el gato v2”.

1ª PARTE DE LA PRÁCTICA: VARIABLES.

3. Vamos a crear la variable “Puntos”. Elige la pestaña “Programas”, categoría “Datos”, y pulsa en el botón “Crear una variable”. Dale el nombre “puntos” y pulsa “Ok”.



4. Verás que aparecen nuevas instrucciones en la ventana central. Vamos a hacer que la variable “puntos” se inicialice a 0 cuando comience el programa, y que sume un punto cada vez que el gato toque al ratón. Agrega las instrucciones necesarias para hacer esta programación (ver abajo).



5. Fíjate que ponemos la variable a 0 cada vez que comienza el programa y que va sumando un punto cada vez que el ratón y el gato se tocan. Pruébalo.

2ª PARTE DE LA PRÁCTICA: PETICIÓN DE DATOS AL USUARIO.

6. Vamos ahora a aprender a pedir datos al jugador, en este caso vamos a poder elegir el nivel del juego. Para ello tenemos que utilizar la categoría “sensores” de la pestaña “Programas”. Escribe las modificaciones en los programas del ratón para poder seleccionar el nivel.



7. Fíjate que hemos preguntado el nivel de juego (un número entre 1 y 5), y que la respuesta (en azul) la usamos para modificar el tiempo de espera del ratón entre un movimiento y el siguiente, de tal modo que a mayor número, menor tiempo de espera (y por lo tanto, más rápido se mueve el ratón). También hemos hecho que el programa contador de puntos no comience hasta que se elija el nivel. Pruébalo.
8. Habrás notado que el gato sigue al cursor. Arregla el programa del gato para que no comience hasta que hayamos metido el nivel en el que queremos jugar.
9. Muestra el proyecto a tu profesor.

PRÁCTICA 9.APLICACIÓN DE LO APRENDIDO.**TEORÍA.**

En esta práctica vas a aplicar los conceptos aprendidos hasta ahora (variable, petición de datos al usuario, etc.).

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto con los siguientes requisitos:
 - 2.1. El proyecto es un juego de preguntas y respuestas. Se le irá preguntando al usuario y se almacenarán los aciertos en una variable que dé la puntuación final.
 - 2.2. El programa tendrá al menos 5 preguntas.
 - 2.3. Puedes usar escenarios, objetos, sonidos (por ejemplo para acierto y error), etc.
 - 2.4. Cuanto más logrado esté el programa, mejor puntuado será.
3. Guarda el proyecto y muéstraselo a tu profesor.

PRÁCTICA 10. MELODÍAS. LÁPIZ.

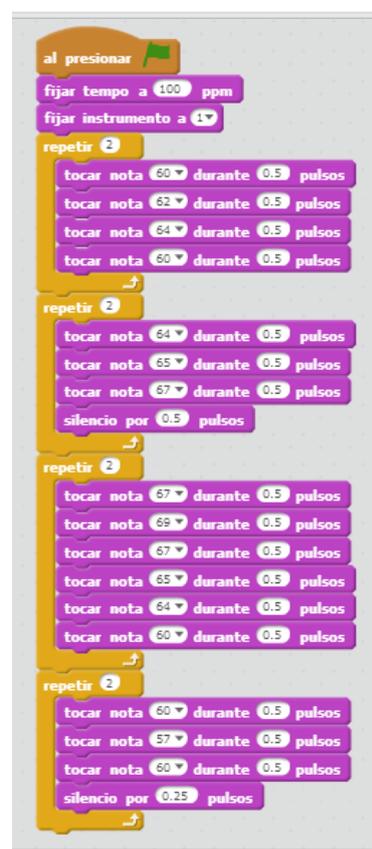
TEORÍA.

Scratch permite tocar melodías mediante la categoría “sonido” de la pestaña “programas”. Scratch nos permite elegir el instrumento y tocar notas con él. Las notas aparecen en nomenclatura inglesa, donde el “la” corresponde a “A”, “si” a “B”, “do” a “C” y así sucesivamente. Las notas vienen numeradas del 48 al 72, pero no estamos limitados por esto ya que podemos escribir otras notas (poniendo 76, por ejemplo, en la nota). También se pueden crear acordes, simplemente ejecutando dos notas simultáneamente.

Todos los personajes de Scratch tienen un lápiz oculto (herencia de Logo), que podemos bajar para dibujar. Así podremos dibujar distintas formas geométricas, por ejemplo, con nuestro gato.

PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto llamado “Música”.
3. Ve a la pestaña “programas”, categoría “sonido” y crea el programa que hay a tu derecha. Pruébalo.
4. Observa que fijamos el tiempo a 100. Cuando le decimos que la nota suene durante 0.5 pulsos, esos dos datos están relacionados. Prueba a cambiar el tiempo a 200 (más rápido) y a 50 (más lento) y Pruébalo.
5. Observa también que hemos elegido piano (“Fijar instrumento a 1”). Prueba la canción con otros instrumentos.
6. Guarda el proyecto.
7. Crea un nuevo proyecto, llámale “Dibujar cuadrado”.
8. Vamos a dibujar un cuadrado. Para ello, debes utilizar las funciones de la pestaña “Programa”, categoría “Lápiz”. Crea el siguiente programa:



9. Guarda el proyecto.
10. Crear un nuevo proyecto. Llámale “Dibujar Triángulo”.
11. Crea las instrucciones necesarias para dibujar un triángulo.
12. Guarda el proyecto y muestra los tres proyectos a tu profesor.

TAREAS DE AMPLIACIÓN (puedes hacer una o las tres).

13. Crea tu propia canción. Llama al proyecto “canción personal”.
14. Crea un programa para dibujar un cuadrado mediante movimientos polares y un bucle. Llama al proyecto “cuadrado bucle”.
15. Crea un programa para hacer un polígono regular de “n” lados mediante movimientos polares. El programa ha de pedir al usuario el número de lados al comenzar, y luego dibujarlo. Llama al proyecto “polígono n”.
16. Guarda los proyectos y muéstraselos a tu profesor.

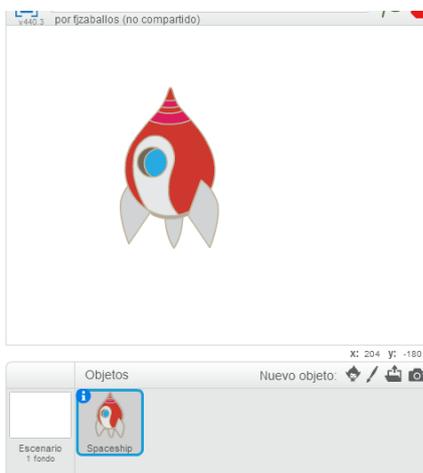
PRÁCTICA 11. SCROLL.

TEORÍA.

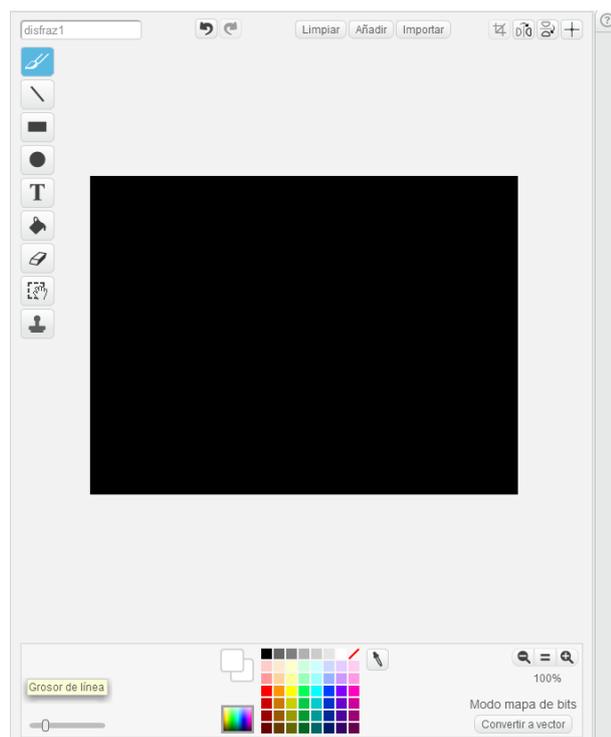
Podemos definir **scroll** como el desplazamiento del escenario donde se desarrolla un videojuego. Se le llama **scroll horizontal** cuando este desplazamiento se realiza horizontalmente, y **scroll vertical** cuando se realiza verticalmente. En esta práctica vamos a manejar este concepto. Para entender bien la práctica, conviene recordar que la pantalla de Scratch es de 480 x 360 píxeles.

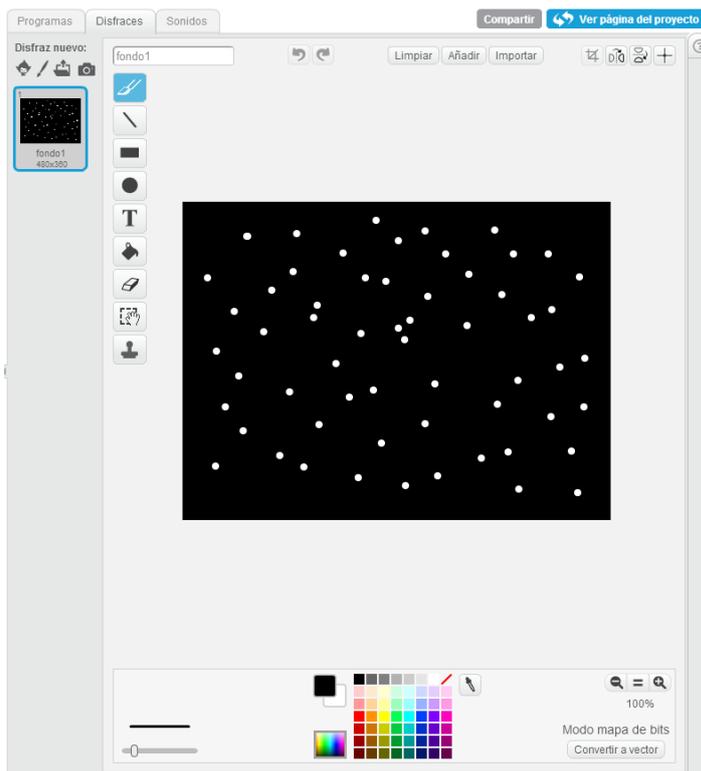
PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto llamado "Marcianitos v1".
3. Borra el objeto "gato" e importa el objeto "nave" de la biblioteca (en el tema "espacio").



4. Crea dos nuevos objetos que simulen el espacio. Llámales "fondo1" y "fondo2". Para crear los objetos debes usar el editor, con la opción "Dibujar nuevo objeto". Lo más fácil es dibujar un fondo negro y después elegir un grosor grande en el pincel para pintar las estrellas de blanco (el grosor de la línea se elige abajo, a la izquierda).
5. Te quedarán dos fondos parecidos al del ejemplo de la página siguiente.





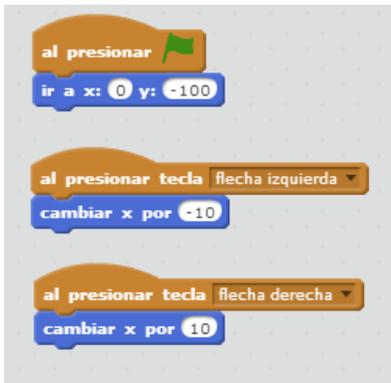
6. Ahora crea dos variables “scroll1” y “scroll2” para los fondos. Estas variables serán las que controlen la posición de cada uno de los objetos “fondo1” y “fondo2”.
7. Vamos a inicializar los objetos en dos posiciones distintas para conseguir el efecto de scroll vertical. Las variables de scroll se inicializan a 0 y se decrementan en 1 hasta llegar a (-345). En ese momento el objeto fondo vuelve a su posición inicial y scroll a 0. Escribe el programa para “fondo1”. Pruébalo, ocultando el “fondo2” (crea las instrucciones necesarias para ello).



8. Has visto que el “fondo1” va bajando por la pantalla. Vamos a hacer lo mismo con “fondo2” para que se solape con “fondo1”. Crea el siguiente programa para “fondo2”.



9. Fíjate que mientras que “fondo1” empieza con el valor “y” a 0, “fondo2” empieza con dicho valor a 345, es decir, más arriba. Al desplazarse los dos hacia abajo, el efecto es como si la nave estuviera avanzando. Prueba el programa.
10. Ya tenemos configurado el scroll. Por último, configura la nave para que se mueva a izquierda y derecha y aparezca en la parte baja de la pantalla, tal y como se muestra a continuación.



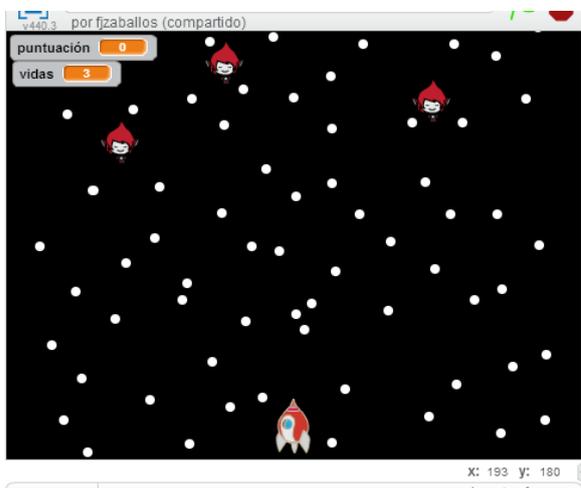
11. Guarda el proyecto y muéstraselo a tu profesor.

PRÁCTICA 12. CLONES.**TEORÍA.**

Scratch permite crear copias (clones) de los objetos, que pueden tener su propio comportamiento. Esto puede ser interesante para crear enemigos. Vamos a ver una aplicación práctica de ello.

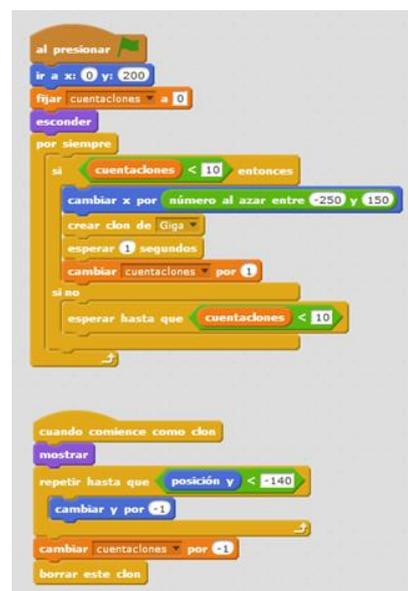
PRÁCTICA.

1. Entra en la web de Scratch con tu login y password.
2. Abre tu proyecto llamado “Marcianitos v1” y crea una copia. Renómbrala como “Marcianitos v2”.
3. Importa de la biblioteca un objeto que sea un marciano, por ejemplo “Giga”. Adecúa su tamaño, así como el de la nave, a la pantalla.



4. Crea la variable “cuentaclones”, que nos va a controlar el número de clones que hay en pantalla.
5. Escribe el siguiente programa para “Giga”.

6. Vamos a analizar un poco este programa, ya que es complejo. Fijamos el objeto “Giga” en un punto arriba de la pantalla y lo escondemos. También damos un valor 0 a la variable que nos va a contar el número de clones que hay. Después comprobamos que haya menos de 10 clones. Si hay 10 clones, el programa espera a que haya menos de 10. Si hay menos de 10, movemos a “Giga” horizontalmente entre los valores -250 y 180 (lo que ocupa la pantalla), y creamos un clon. Después, incrementamos la variable “cuentaclones” en 1, y esperamos 1 segundo para seguir creando clones. En la parte de abajo hemos programado el



comportamiento del clon. Cuando se crea un clon, lo mostramos y le vamos restando 1 a la “y” para que vaya bajando. Cuando llega a -140 lo borramos y decrementamos en 1 la variable “cuentaclones”. Pruébalo.

7. Vamos a introducir una interacción entre nave y marcianos. Importa el objeto “orange” de la biblioteca, y adecúa su tamaño al juego (debe ser pequeño, pues lo vamos a usar como bala).
8. Crea la variable “puntuación”, que sumará 10 puntos cuando matem os a un marciano.
9. Crea el siguiente código para el objeto “Orange”.
10. Analicemos el código: al iniciar el juego, posicionamos la naranja por la parte baja de la pantalla y la escondemos. Por otro lado, cada vez que pulsamos el cursor “Flecha arriba”, creamos un clon de nuestra bala. Cuando se crea el clon, éste va donde está la nave, se muestra y va subiendo hasta que llega al fin de la pantalla o toca a “Giga”. Si toca a “Giga”, sumamos 10 puntos. Al cumplirse una de las dos condiciones, borramos el clon.

11. Introduce las modificaciones para la interacción con el clon de “Giga”. Hemos modificado la condición de salida del bucle “repetir” para que salga de él si toca la bala.
12. Prueba el juego.
13. Guarda el proyecto y muéstraselo a tu profesor.

TAREA DE AMPLIACIÓN.

14. Crea una copia de “Marcianitos v2” y llámale “Marcianitos v3”.
15. Crea una variable “vidas” inicializada a 3 vidas. Si la nave toca al clon, pierde una vida. El juego acaba cuando el contador de vidas llega a 0.
16. Añade una vida extra cuando el contador de puntos llegue a 5000.
17. Guarda el proyecto y muéstraselo a tu profesor.

PRÁCTICA 13. PRÁCTICA FINAL.**PRÁCTICA.**

1. Entra en la web de Scratch con tu login y password.
2. Crea un nuevo proyecto combinando algunas de las cosas que has aprendido.
3. Una vez generado el programa, evalúa el código en la siguiente url:
<http://drscratch.programamos.es/>
4. Cuanto más logrado esté el proyecto, mejor puntuado será.
5. Guarda el proyecto y muéstraselo a tu profesor.