

Scratch Intermedio

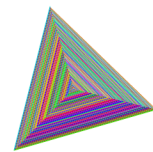
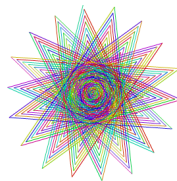
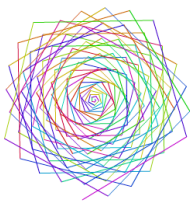


Herramienta Lápiz

Tarjeta 1 de 5

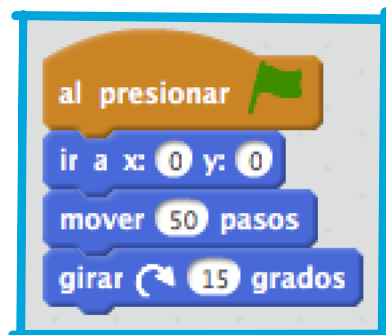
Estoy aprendiendo: Scratch

- 1 Ahora que has aprendido los conceptos básicos de Scratch (y si no lo has hecho, echa un vistazo las Tarjetas Sushi Scratch Principiante) y haz tu primer juego con Scratch. En esta serie vas a aprender unos cuantos trucos y a hacer uno de mis proyectos favoritos de Scratch: dibujar patrones de colores y, si se configura de la forma adecuada, puede ser algo muy vistoso.

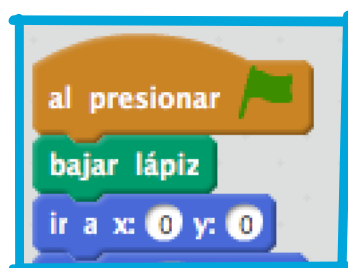


Todo depende de la herramienta **lápiz**, que es controlada por bloques en el grupo de **lápiz**. Estos bloques van trazando una línea detrás de un objeto mientras se mueve. Ahora vas a aprender cómo usarlo!

Crea un nuevo archivo de Scratch, selecciona el objeto del Gato Scratch y arrastra algunos de los bloques que ya conoces de las tarjetas de principiante, hasta que se parezca a esto:



- 2 ¡Genial! ¡Ahora, es momento de probar el lápiz! En la sección de lápiz, seleccione el bloque “bajar lápiz” y añadirlo al inicio del programa, como aquí:



Ahora haz clic en la bandera verde un par de veces y ver qué pasa!

Scratch Intermedio



Herramienta Lápiz

Tarjeta 1 de 5

Estoy aprendiendo: Scratch

3

Si puedes ver las líneas detrás el gato entonces el lápiz está funcionando y ya puedes comenzar a hacerlo dibujar patrones guays!

¡En primer lugar, tienes que deshacerse del gato! ¡Se está poniendo en medio del dibujo! Sólo tienes que añadir un bloque “esconder” de la sección **apariencia** al principio del programa y el gato desaparecerá.

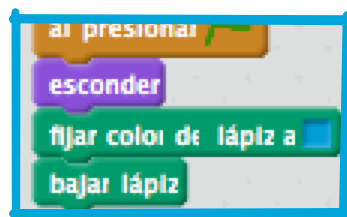


4

Ahora, puedes cambiar el color del lápiz con otro bloque de la sección del lápiz, pero este bloque es algo distinto a los que ya has visto. Es el bloque “fijar color de lápiz a” y se ve como esto:



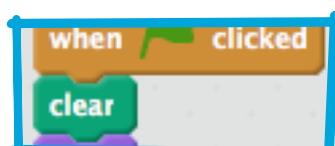
Arrastra uno hasta tu **panel de objetos** y encajalo encima del bloque “bajar lápiz”.



Ahora, haz clic en el cuadro de color (en el código anterior es el azul) y verás que el puntero del ratón se cambia por una mano. ¡Desplázate por la pantalla y observa cómo cambia el color en el cuadro coincidiendo con el color de lo que está por debajo del puntero! Al hacer clic, Scratch guardará ese color como el color de tu lápiz.

5

Si has hecho clic en la bandera verde para probar el código, te habrás fijado que no desaparecen los trazos del lápiz. ¡Agrega un bloque de “borrar” de la sección de la **lápiz** al comienzo de tu código para arreglarlo!



Scratch Intermedio

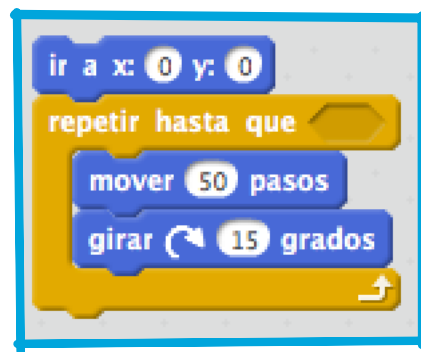


Dibujo de patrones

Tarjeta 2 de 5

Estoy aprendiendo: Scratch

- 1 Ahora tienes un programa que dibuja una línea, pero sólo una línea. ¿Es un poco aburrido, no? Puede utilizar un bucle, como “por siempre”, para dibujar una y otra vez. Por supuesto, si sólo utilizas “por siempre” entonces conseguirás dibujos que salgan del escenario!
Por lo que puedes utilizar un bucle distinto, que encontrarás en la misma sección de **control**, llamado “repetir hasta que” que va a hacer algo una y otra vez, hasta que una condición de verdadero/falso sea verdadero. Ponlo alrededor de tus bloques “mover” y “girar” así:

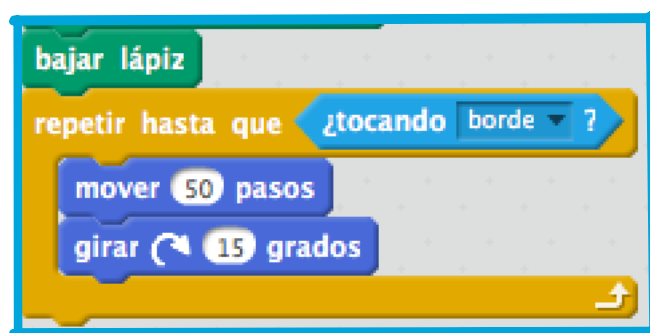


- 2 Ahora haz clic en la bandera verde para ejecutar el programa unas cuantas veces y verás qué pasa! Te darás cuenta de dos cosas: siempre empieza a dibujar una línea en el centro del **escenario** y no se detiene en el borde.

La primera es porque el primer bloque de **mover** que se ejecuta después de “bajar lápiz” es “ir a x: 0 y: 0”. Puedes solucionar este problema simplemente moviendo el bloque “ir a x: 0 y: 0” antes del bloque “bajar lápiz” y agregando justo al comienzo de tu código, un bloque de “subir lápiz” de la misma sección **lápiz**

La segunda es porque no todavía le has dicho lo que está comprobando, así que la condición nunca será verdadera. Básicamente, ahora mismo, está trabajando como un bucle “por siempre”.

- 3 Ahora toca ajustar tu bloque “repetir hasta”. Estás intentando buscar es saber si el objeto (invisible) toca el borde del **escenario**, por lo que necesitas un bloque de **sensores**. En este caso, el bloque “¿tocando?”. Encaja en tu “repetir hasta” y selecciona “borde”.



Scratch Intermedio



Dibujo de patrones

Tarjeta 2 de 5

Estoy aprendiendo: Scratch

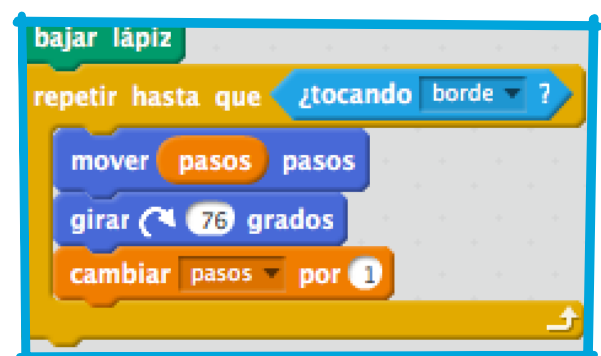
4 Cambia el número de pasos en tu programa a 5 y comprueba que coincide con éste:



5 Ahora, cuando ejecutes el programa, verás que se ha convertido en un programa para dibujar círculos! ¡Por lo menos se queda en el **escenario**! El problema aquí es que esos 15 grados se convierten 360, y el dibujo en un círculo completo. Lo que tienes que hacer ahora es los pasos un poco más largos cada vez, y así se convertirá en una espiral. Para esto, vas a necesitar una **variable**.

Ya has visto antes las **variables**, en la serie para principiantes. Básicamente son lugares que etiquetas para almacenar los números que te importan. Las puedes crear en la categoría de bloques de **datos** y también encontrarás allí los bloques relacionados.

Crea una **variable** llamada "pasos" y utiliza su valor en lugar de los 5 en el bloque "mover 5 pasos", a continuación añade "fijar pasos a 0" al inicio de tu programa y "cambiar pasos por 1" como parte de tu ciclo (¿importa donde lo pusiste?).



Ahora ejecútalo y trata de cambiar el número de girar grados (prueba 76 y 120)!

Scratch Intermedio



Solicitud de entrada

Tarjeta 3 de 5

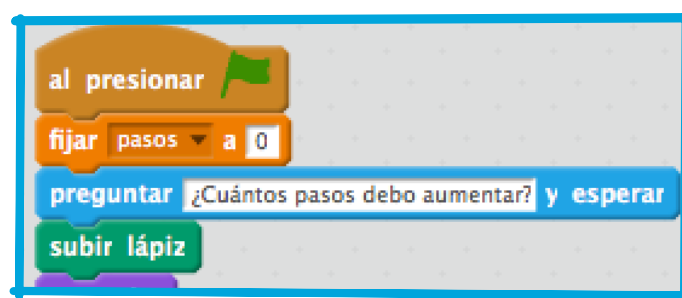
Estoy aprendiendo: Scratch

1 Ok, esto se está poniendo muy interesante, pero es un poco de dolor de cabeza tener que editar el código cada vez que quieras ver un patrón diferente. ¿No sería bueno conseguir que el programa preguntarse por esos valores? ¡Tu lo puedes hacer!

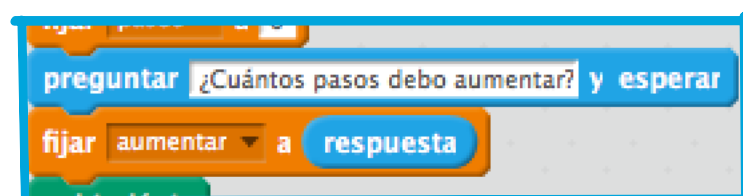
Primero, crea **variables** desde la sección de **datos** para “grados” y “aumentar” y añadelas a tu código como aquí:



2 Ahora necesitas pedir valores para estas dos **variables** y guardarlos. Para esto utiliza un bloque de sensores que se llama “preguntar y esperar”, que te permite escribir una pregunta dentro. Arrastra uno en tu **panel de objetos** y cambia la pregunta por “¿cuántos pasos debo aumentar?”, luego agrégalo a tu programa, justo después de fijar pasos a 0, así:



3 Una vez que tienes a Scratch preguntando, solo necesitas recordar la respuesta! Resulta que Scratch tiene una **variable** especial, llamada “respuesta”, donde pone la última respuesta que ha recibido. Puedes encontrarla entre los bloques de sensores. Utiliza de datos el bloque “fijar a”, que toma el valor de “respuesta” y se lo asigna a “aumentar”, así:



Scratch Intermedio



Solicitud de entrada

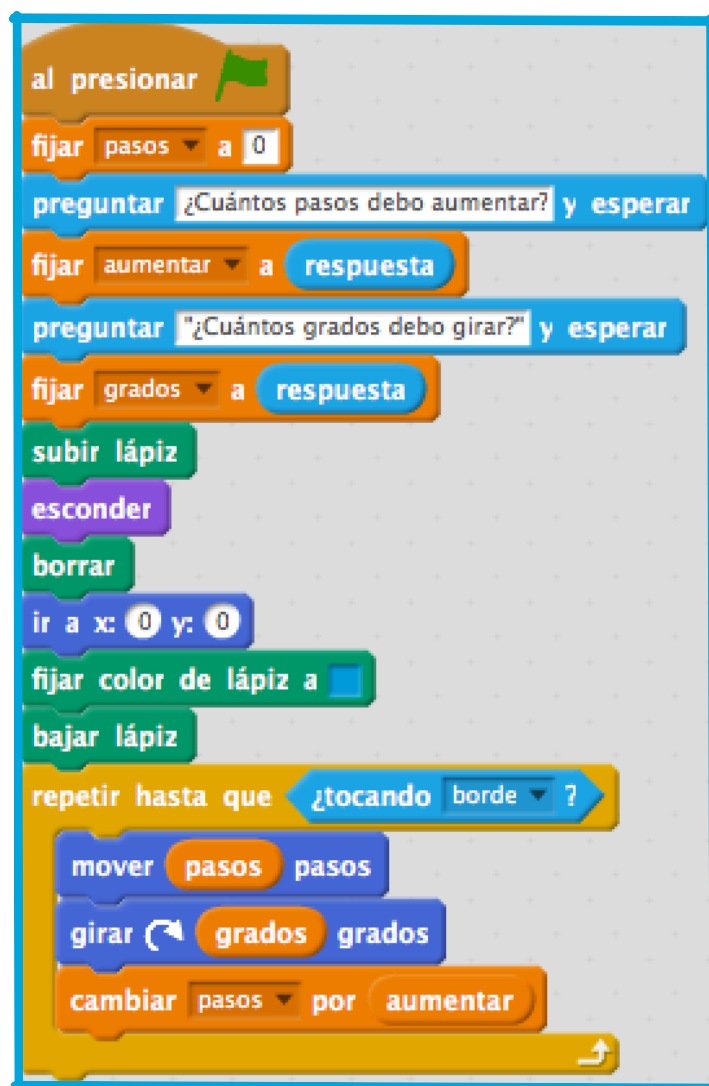
Tarjeta 3 de 5

Estoy aprendiendo: Scratch

- 4 Ahora, haz lo mismo con “grados”, pregunta “¿cuántos grados debo girar?” y guarda la respuesta en “grados”.



- 5 Comprueba que tu programa se parece al abajo y ejecútalo varias veces, prueba con diferentes números. Apunta las respuestas que hacen los dibujos mas guays. ¡Las necesitarás en una tarjeta más adelante!



Scratch Intermedio



Líneas guays

Tarjeta 4 de 5

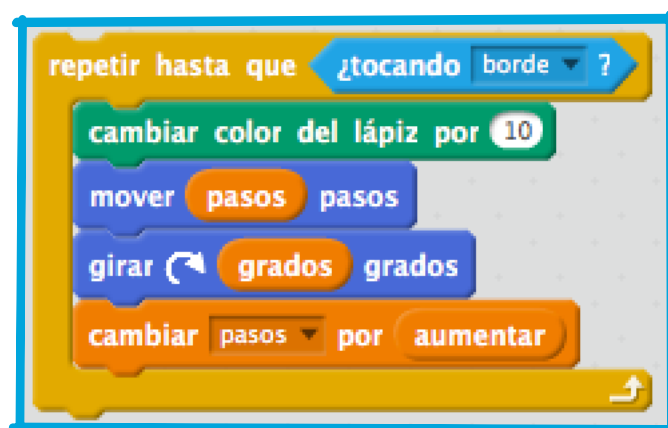
Estoy aprendiendo: Scratch

- 1 Es hora de añadir algo de color. Ahora mismo, tu línea es de un solo color, pero el lápiz tiene bloques que hacen cambiar su color. Con el bloque adecuado de la sección **operadores**, puedes cambiarlo al azar.

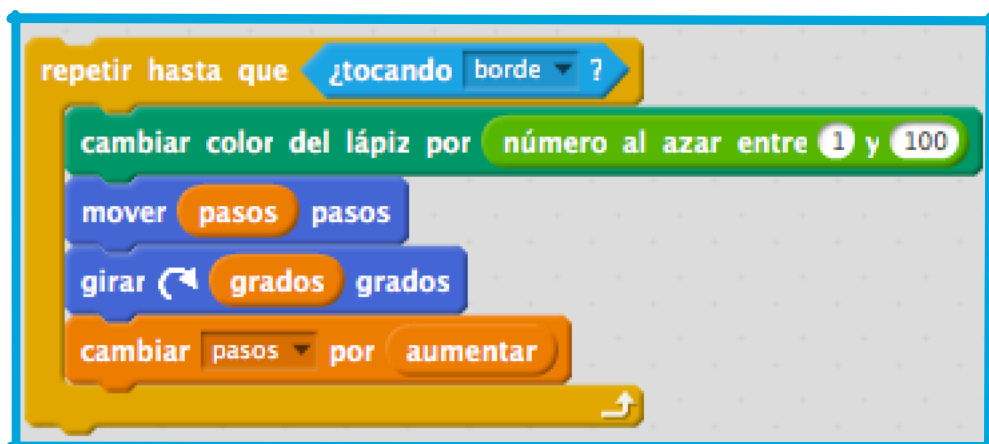
El bloque que cambia colores que estás buscando es “cambiar color del lápiz”:



Coge uno de esos y ponlo en tu control “repetir hasta”, como aquí:



- 2 Eso está guay, pero un poco predecible. Puedes hacerlo un poco más divertido si añades un número al azar, así el color también cambia al azar. Solo pon el operador de número al azar en el bloque “cambiar color del lápiz por” y selecciona los valores entre los que estará tu número. Yo probaría 1 y 100 para empezar.



Intenta ejecutarlo otra vez, verás un arco iris de colores al azar!

Scratch Intermedio



Líneas guays

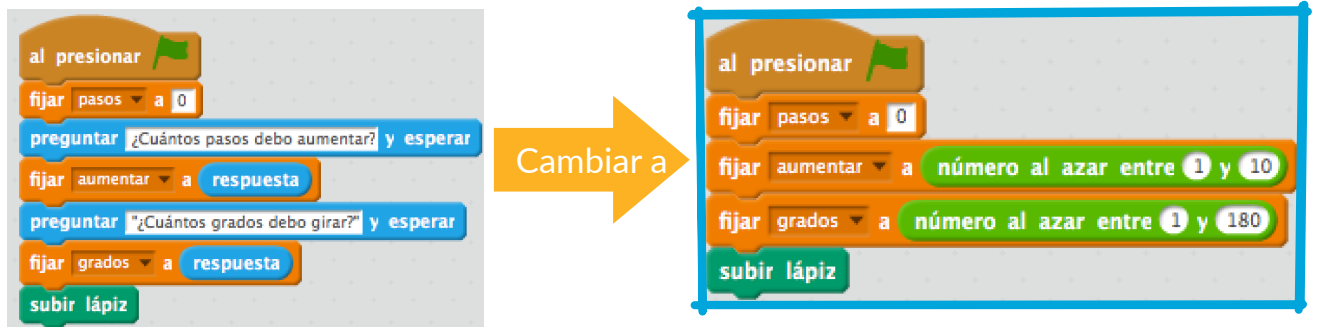
Tarjeta 4 de 5

Estoy aprendiendo: Scratch

3

Realmente puede utilizar números al azar para hacer que todo el programa se ejecute una y otra vez, cambiando el patrón cada vez! Se parecerá un poco a los protectores de pantalla en la década de 1990... que tu no recuerdas, pero preguntale a uno de tus mentores del Dojo!

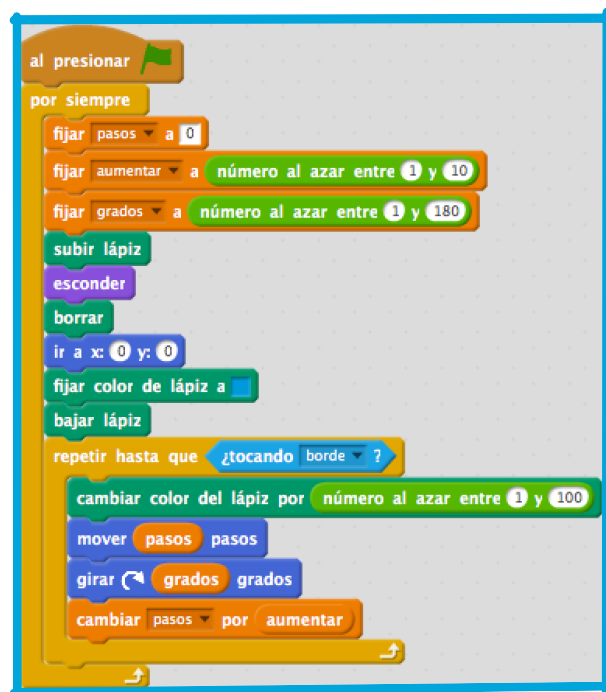
Necesitas hacer unos cambios para que esto ocurra. El primero es que es necesario establecer las variables de “aumento” y “grados”, en lugar de pedir las al usuario. Así que es necesario cambiar algunos de tus bloques de código:



4

Si se lo ejecutas, verás que el programa hace dibujos al azar, pero sólo una vez. ¿Por qué crees que puede ser?

Es porque el control sólo se ejecuta hasta que alcanza el borde del **escenario**. Necesitas otro control, que se ejecute por siempre (por lo que es un bloque “por siempre”) fuera del que estás usando ahora y así lo mantendrás ejecutándose una y otra vez! Sólo arrastra uno de la sección control y envuelve el código en él.



Scratch Intermedio



Ayudar a la computadora

Tarjeta 5 de 5

Estoy aprendiendo: Scratch

1 ¡Ahora tienes algo realmente impresionante a lo que estar mirando! Sin embargo, puedes notar que, de vez en cuando, la computadora dibuja algo que se parece bastante... malo. Esto es porque algunos números de algunas de esas variables son sólo malas decisiones, y algunas *combinaciones de esos números* también son malas decisiones.

¿Te acuerdas que unas tarjetas atrás cuando te dije que escribieras algunos de tus valores favoritos para “aumentar” “grados”, esos donde las imágenes tenían mejor pinta? Si no, no te preocupes, sólo tienes que ver el programa ejecutándose al azar durante un rato y escribir las combinaciones que dan mejores resultados.

Vas a enseñar a Scratch esas combinaciones de valores, así puede usarlas para hacer impresionantes dibujos!

Para hacer esto, necesitarás una **lista**. Las encontrarás con las **variables**, en **datos**. Al igual que con las variables, necesitarás crear tu lista.

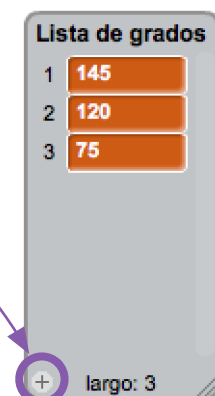
1 Haga clic en “Crear una lista”

2 Pon el nombre de “Lista de grados”

3 ¡Ya está en tu programa!

2 Crea otra lista llamada “Lista de Aumentos” y luego, pulsando en el más (+) en la parte inferior de la lista, escribe los pares de números.

¡Asegúrate de que el valor de la posición de los grados en la “Lista de Grados” coincide con la misma del valor de aumento en la “Lista de Aumentos”!



Scratch Intermedio



Ayudar a la computadora

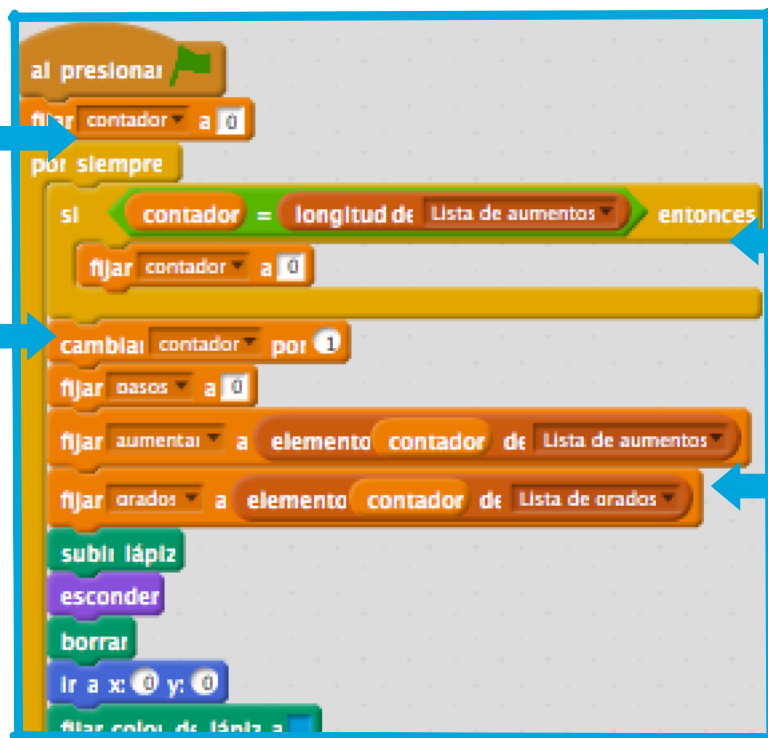
Tarjeta 5 de 5

Estoy aprendiendo: Scratch

3 Tienes listas, ahora sólo necesitas tener tu código para leerlas y control sobre ellas! Para hacer esto, vas a utilizar una nueva **variable** llamada “contador”, algún incremento y un bloque de **control** “Si entonces”. Actualiza tu código para que se parezca a este y tendrás:

Fijar contador a 0, fuera de todos los controles

Añade 1 a contador



Comprueba si contador es la longitud de la lista. Si es así, lo fijas a 0. Esto significa que contador siempre será una posición de un elemento en la lista.

Elige el contador-ésimo elemento de Aumentar la Lista y ponlo en la variable aumentar. Lo mismo para la Lista de Grados y la variable grados.

Si quieres entender lo que está sucediendo

Imagina tus listas sólo tienen dos pares de valores en ellas. Esto es lo que sucede:

1. Fijar contador a 0
2. Iniciar el control por siempre
3. Comprobar si el contador (0) es igual a la longitud de Lista de Aumentos (2). No es el caso
4. Cambio contador por 1. Ahora contador = 1
5. Fijar pasos a 0
6. Obtener el elemento en la posición del contador (1) en la lista Lista de Aumentos y guardarlo en aumentar
7. Obtener el elemento en la posición del contador (1) en la lista Lista de Grados y guardarlo en grados
8. Hacer todas las cosas relacionadas con la creación de los patrones
9. Iniciar el control por siempre
10. Comprobar si el contador (1) es igual a la longitud de la lista Lista de Aumentos (2). No es el caso
11. Cambio contador por 2. Ahora contador = 2
12. Fijar pasos a 0
13. Obtener el elemento en la posición del contador (2) en la lista Lista de Aumentos y guardarlo en aumentar
14. Obtener el elemento en la posición del contador (2) en la lista Lista de Grados y guardarlo en grados
15. Hacer todas las cosas relacionadas con la creación de los patrones
16. Iniciar el control por siempre
17. Comprobar si el contador (2) es igual a la longitud de Lista de Aumentos (2). Coinciden!
18. Fijar contador a 0
19. ¡Continúa desde el **paso 4** de esta lista, en un bucle interminable!