

# Amolar

El área «Control» contiene bloques que controlan si se están implementando otros bloques ubicados dentro de ellos. Existen dos tipos de bloques de control: **los bloques si-en caso contrario** (que se describen en una página independiente) y los bloques que controlan la frecuencia con la que se ejecutan sus elementos internos. Estos últimos se denominan bucles porque su interior, también conocido como cuerpo o cuerpo del bucle, se repite (probablemente) varias veces. Cada recorrido de un bucle se denomina iteración.

## Bloques para la creación de bucles

### repetir constantemente

El bloque **repetir constantemente** ejecuta el código en su cuerpo hasta que finaliza el programa.

### repetición

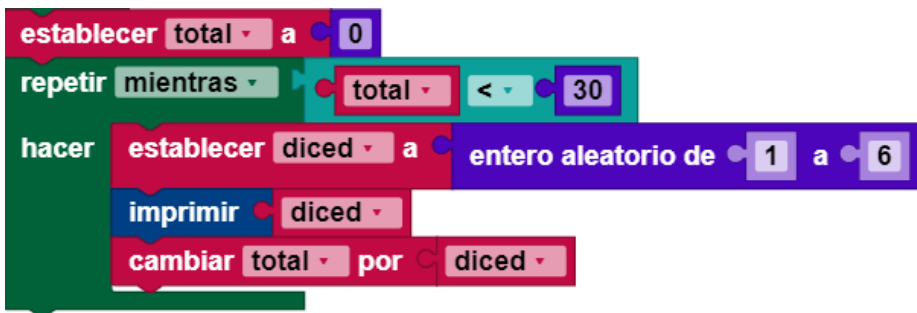
El bloque **repetición** ejecuta el código en su cuerpo según la frecuencia especificada. Por ejemplo, el siguiente bloque muestra «¡Hola!» diez veces:



### repetición-mientras que

Imagine un juego en el que un jugador lanza un dado y suma todos los valores obtenidos, siempre que el total sea inferior a 30. Los siguientes bloques implementan este juego:

1. Una variable denominada **en total** contiene un valor inicial de 0.
2. El bucle comprueba en primer lugar si **en total** es inferior a 30. Si es inferior, los bloques se implementan en el cuerpo.
3. Se genera un número aleatorio en el intervalo de 1 a 6 (para simular una tirada de dados) y se almacena en una variable denominada **tirada de dados**.
4. Se muestra el número obtenido.
5. La variable **en total** aumenta con el número de tiradas.
6. Cuando se llega al final del bucle, el control vuelve al paso



Una vez finalizado el bucle, se recorren todos los bloques posteriores (no mostrados). En el ejemplo, el recorrido de los bucles finaliza después de haberse mostrado un número determinado de cifras aleatorias en el intervalo de 1 a 6, y el valor de la variable **en total** tiene la suma de estos números que es, como mínimo, de 30.

## repetición-hasta

Los bucles **repetición-si** repiten su cuerpo **si** se cumple una condición. Los bucles **repetición hasta** son parecidos, con la diferencia de que repiten su cuerpo **hasta** que se cumple una condición determinada. Los bloques siguientes son equivalentes al ejemplo anterior, ya que el bucle se ejecuta hasta que **en total** es superior o igual a 30.



## contar-desde-hasta

El bucle **contar-desde-hasta** incrementa el valor de una variable, comenzando con un primer valor, terminando con un segundo valor y en incrementos de un tercer valor, ejecutando una vez el cuerpo por cada valor de la variable. Por ejemplo, el siguiente programa genera los números 1, 3 y 5.



Como muestran los dos bucles siguientes, que generan los números 5, 3 y 1, este primer valor puede ser superior al segundo. El comportamiento es el mismo, independientemente de que el valor incremental (tercer valor) sea positivo o negativo.



## para cada

El bloque **para cada** es similar al bucle **contar-desde-hasta**, solo que en lugar de usar la variable del bucle en orden numérico, utiliza los valores de una lista en orden. El siguiente programa genera cada elemento de la lista «alfa», «beta» y «gamma»:



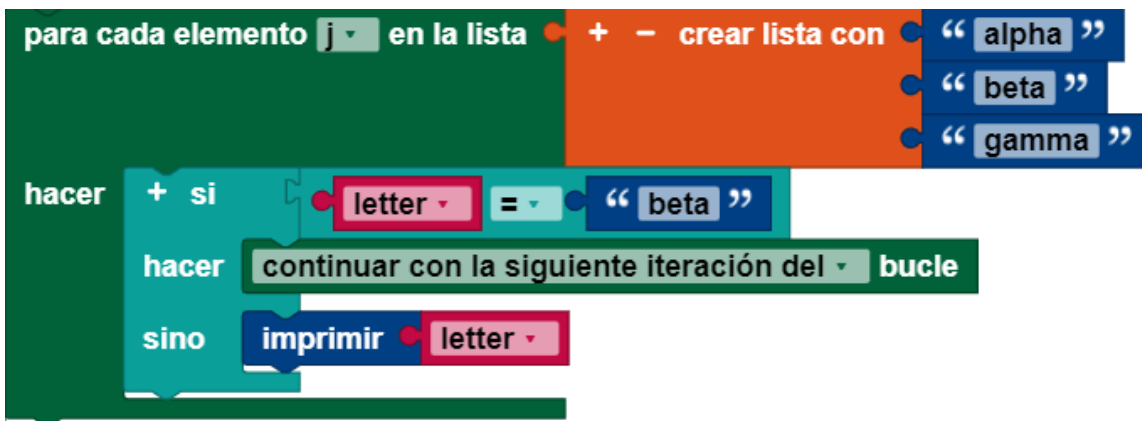
## Bloques de interrupción de bucles

La mayoría de los bucles se ejecutan hasta que se cumple la condición de interrupción (para los bloques de **repetición**) o hasta que se aceptan todos los valores de la variable del bucle (en el caso de los bloques **contar con** y **para cada**). Dos bloques de uso poco frecuente, pero ocasionalmente útiles, ofrecen opciones adicionales para controlar el comportamiento del bucle. Se pueden utilizar con cualquier tipo de bucle, aunque los siguientes ejemplos muestran su uso con el bucle **para cada**.

### continuar-con-la-siguiente-iteración

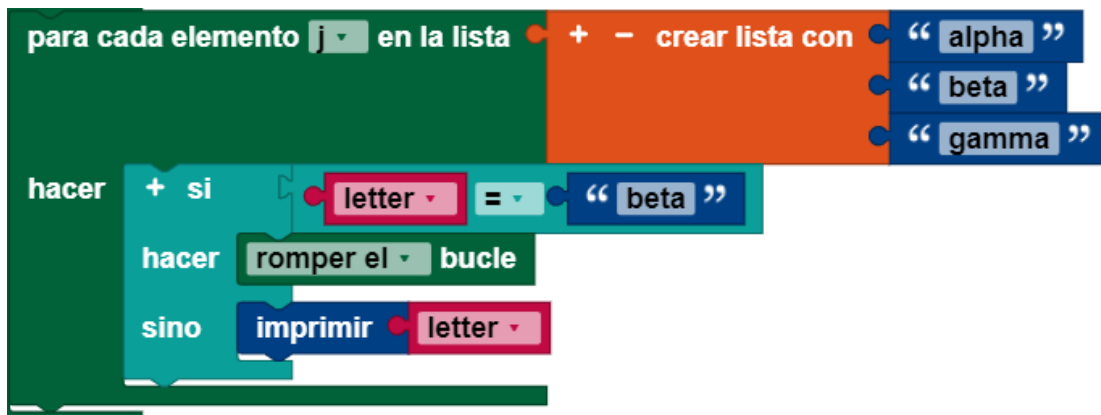
**continuar-con-la-siguiente-iteración** hace que se omitan los bloques restantes en el cuerpo del bucle y que comience la siguiente iteración del bucle.

El siguiente programa genera «alfa» en la primera iteración del bucle. En la segunda iteración se ejecuta el bloque **continuar con la siguiente iteración**, por lo que se omite la salida de «beta». En la última iteración se pulsa «gamma».



### Interrupción del bucle

El bloque **interrupción del bucle** permite salir de manera anticipada del bucle. El siguiente programa genera «alfa» en la primera iteración e interrumpe el bucle en la segunda iteración si la variable del bucle es igual a «beta». El tercer punto de la lista nunca se alcanza.



Revision #4

Created 17 November 2021 20:25:52 by Admin

Updated 18 February 2022 14:11:25 by Admin