

Taller: Crea tu primer videojuego en Thunkable

Antes del taller:

1. Recomendamos que completes el tutorial **Bola de Cristal**. Lo puedes encontrar en el siguiente link:
<https://docs.google.com/document/d/1Gqihog1nKZ1RgfxMNTYNOi62OYhWbzUBDTx5s4G4s44/edit?usp=sharing>
2. Si cuentas con celular o tablet Android o iOS descarga e instala Thunkable Live.
<http://thunkable.live/> Para hacer tus pruebas ahí. (Si no tienes dispositivo podrías participar en el taller haciendo pruebas en tu computadora).
3. Descargar las siguientes imágenes en tu computadora:

Topo:

<https://drive.google.com/file/d/1sezhuMkv6Pdb-loA2Dgdds7ZMOD8iLa/view?usp=sharing>

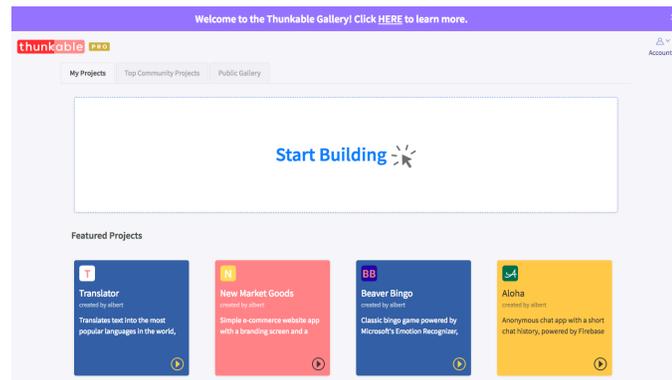
Agujero:

<https://drive.google.com/file/d/10STTi8Y4mnfrNXlALTWrp1xnJ4TlPy5Y/view?usp=sharing>

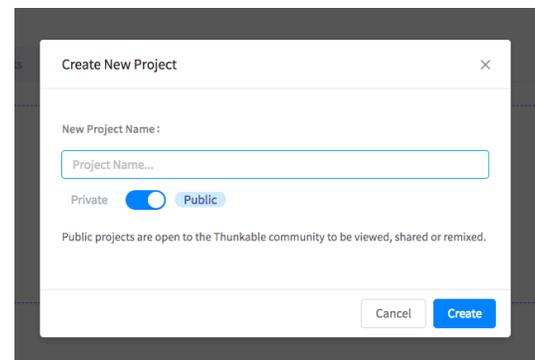


Tutorial: Aplasta Topo

Haz clic en el botón "Start Building" ("Comenzar a desarrollar")



Después, asigna un nombre a tu proyecto y configúralo como Privado o Público. Los proyectos públicos son visibles para toda la comunidad pública. Los proyectos privados solo son visibles para sus creadores, pero pueden tener un costo asociado con esta opción.



Nombre sugerido: "Aplasta Topo"
Public Haz clic en [Create](#).

Diseñando la Interfaz

Agrega los elementos principales del juego:

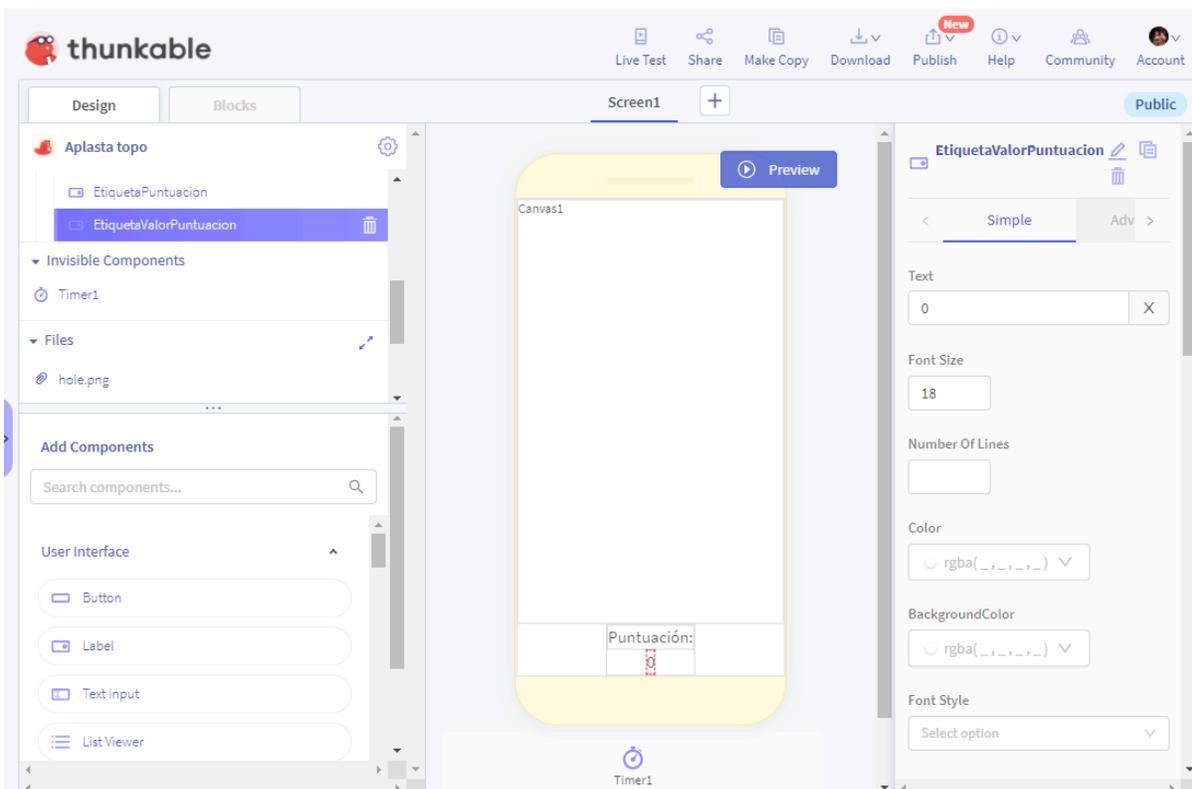
1. Del panel "Add Component" ("Agregar Componentes") arrastra el componente llamado "Canvas" ("Lienzo") a la pantalla.
2. Del panel "Add Component" ("Agregar Componentes") arrastra un componente tipo "Label" ("Etiqueta") a la pantalla.
3. En el panel Propiedades de "Label1" ("Etiqueta 1"), cambia el campo "Text" ("Texto") a "Puntuación:"
4. Arrastra otra etiqueta ("Label") y cambia su campo de texto a "0"

5. En el panel Propiedades “Label2” (“Etiqueta 2”) da clic en el icono de  y cambia el nombre del componente a “EtiquetaValorPuntuacion”



6. Puedes ajustar las propiedades de las etiquetas como el color y tamaño de texto.
7. Del panel “Add Component” (“Agregar Componentes”) en la sección de “Sensors” o (“sensores”) arrastra un componente tipo “Timer” (“Temporizador”) a la pantalla.
8. En el panel Propiedades de “Timer 1” (“Temporizador 1”) cambia “Interval” (“Intervalo”) -> “400”
9. En el panel Propiedades de “Timer 1” (“Temporizador 1”) cambia “CountUp” (“Contar Hasta”) -> “true” (“Verdadero”)

Hasta ahora tu interfaz debería verse de la siguiente manera:



Agrega los elementos del lienzo

- a. En la lista de componentes de tu app, da click en el componente “Stage1” para cambiar sus propiedades.
- b. En el panel Propiedades de “Stage1” (“Etapa1”), cambia:
 - i. “Background Color” (“Color de Fondo”)→ R:126,G:211,B:33
 - ii. “Height” (“altura”) → ”320”
 - iii. “Width” (“ancho”) → ”320”
- c. Selecciona el elemento Sprite_Type1 (“Sprite_Tipo1”)



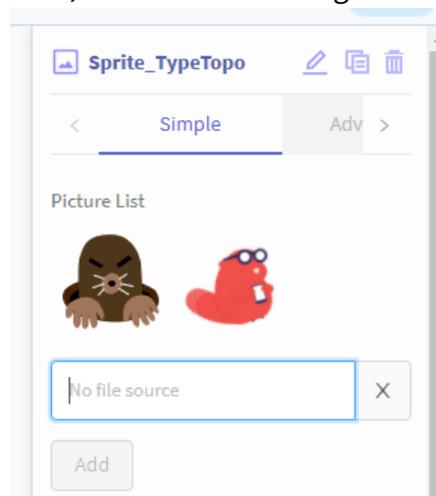
de la lista de componentes.

- d. En el panel Propiedades de Sprite_Type1 (“Sprite_Tipo1”):

- i. Cambia el nombre a “Sprite_TypeTopo”



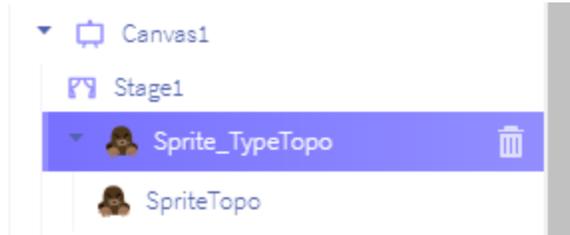
- ii. En la propiedad “Picture List” sube la imagen del topo que descargaste, para hacerlo da click en el recuadro que dice “No file source” y después en “Upload files”, ahora busca la imagen en tu computadora y da click en “Open” o "Abrir"



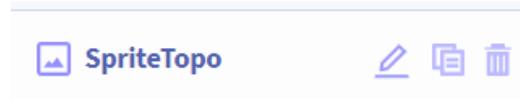
- e. Selecciona el elemento Sprite1
componentes.
Debería quedar así:



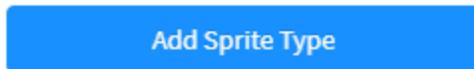
de la lista de



- f. En el panel Propiedades de Sprite1:



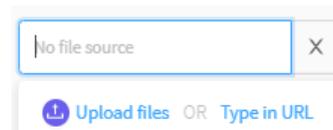
- i. Cambia el nombre a “SpriteTopo”.
- ii. Cambia la propiedad Z a “1”. Esto va a hacer que el topo aparezca frente a los agujeros.
- g. Del panel “Sprite Class” da click en el botón “Add Sprite Type”



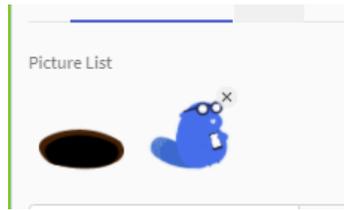
- h. En el panel Propiedades te aparecerá nuevo elemento llamado “Sprite_Type1”:



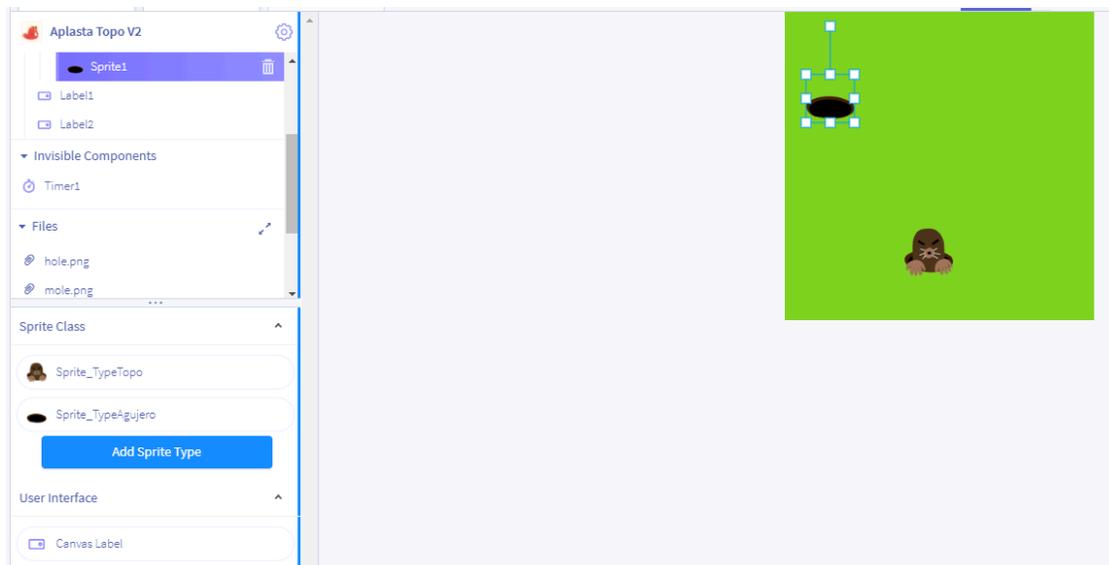
- i. Cambia el nombre a “Sprite_TypeAgujero”
- ii. Sube la imagen de agujero que descargaste:
- iii. Borra la imagen del topo azul seleccionandola y dando click en la cruz que aparece en la parte de arriba a la derecha:



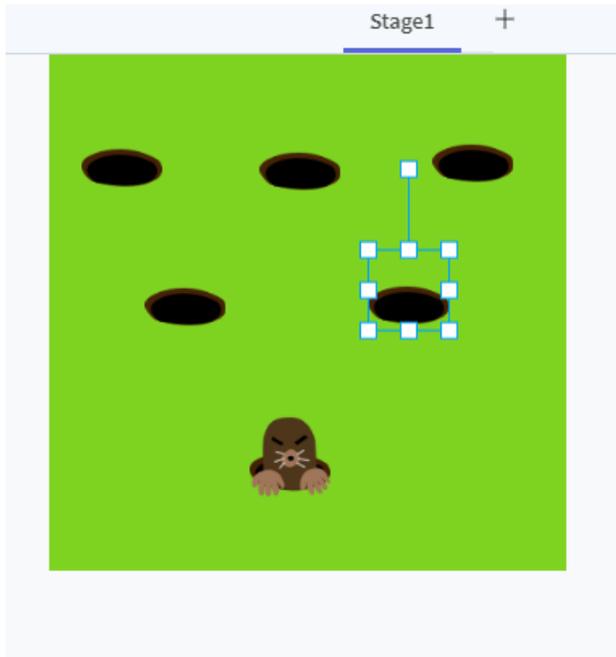
iv.



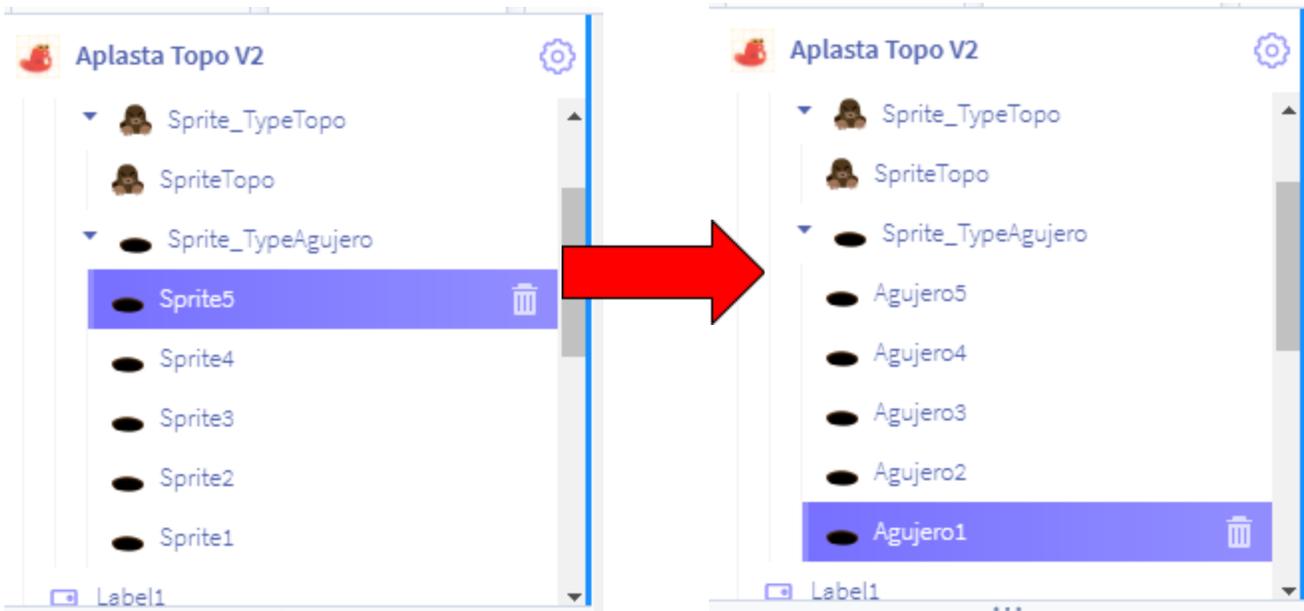
- i. En el panel “Sprite Class” selecciona “Sprite_TypeAgujero” y arrastralo a “Stage1”. Ahora vas a ver un agujero en tu pantalla principal del videojuego.



- j. Repite este paso cuatro veces más, hasta que tengas 5 agujeros en la pantalla Nota: (No te preocupes por la ubicación de los agujeros en la pantalla, eso lo arreglaremos más adelante):



- k. En el panel Propiedades de “Sprite 1” al “Sprite5” cambia el nombre a “Agujero1” hasta llegar al “Agujero5” respectivamente, como se muestra a continuación.



- l. En el panel Propiedades de cada uno de tus componentes “Agujero1” al “Agujero5”, cambia las propiedades “X” y “Y”, a como la siguiente tabla:

Componente	X	Y
Agujero1	40	60
Agujero2	150	60
Agujero3	260	60
Agujero4	95	140
Agujero5	210	140

Agregar comportamiento a los componentes

Aquí está una descripción general de lo que vamos a programar en los bloques (No empieces a programar aún):

1. Crear las variables global:
 - a. **agujeroActual**: Agujero dónde está parado el topo.
 - b. **X**: Las ubicaciones X de los cinco agujeros del juego.
 - c. **Y**: Las ubicaciones Y de los cinco agujeros del juego.
2. Cuando la app inicia:
 - a. Llamar el procedimiento **MoverTopo**.
3. Crear el procedimiento **MoverTopo** para que:
 - a. Cambie la variable local **agujeroActual** a un número aleatorio del 1 al 5 (por que tenemos 5 agujeros en nuestro juego).
 - b. Mueva el topo a la ubicación del **agujeroActual**.
 - c. Iniciar el **Timer1**
4. Hacer que **Timer1** llame al procedimiento **MoverTopo** cuando el temporizador se apaga.
5. Implementa un manejador que haga lo siguiente cuando el topo es tocado:
 - a. Sumar uno a la puntuación.
 - b. Hacer que el teléfono vibre un poco.
 - c. Llamar el procedimiento **MoverTopo**

Ahora vamos a programar. Para continuar cambia a Editor de Bloques.

Programando en “Blocks” (“Bloques”)

Cambia al interfaz de código dando click en Blocks:



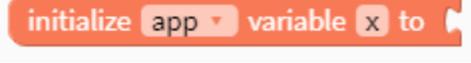
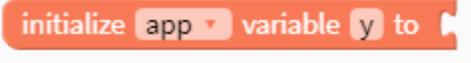
Del lado izquierdo de la pantalla hay una lista de secciones, al seleccionar alguna podrás ver los bloques disponibles. Identifica de qué sección de bloques debes tomar los bloques para el tutorial.



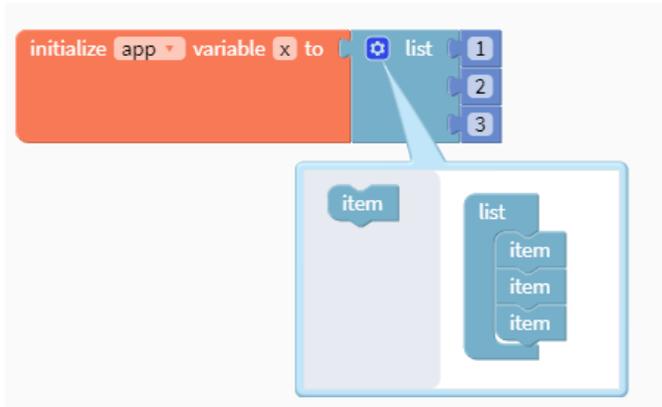
Creando variables

1. Crea una Variable llamada **agujeroActual**, Por ahora vamos a darle un valor inicial de 0; ya que vamos a poner el valor real en el manejador procedimiento **MoverTopo**, que se ejecuta cada vez que la app se carga en la pantalla.
2. Crea las variables **X** y **Y** para guardar las ubicaciones de los cinco agujeros del juego.

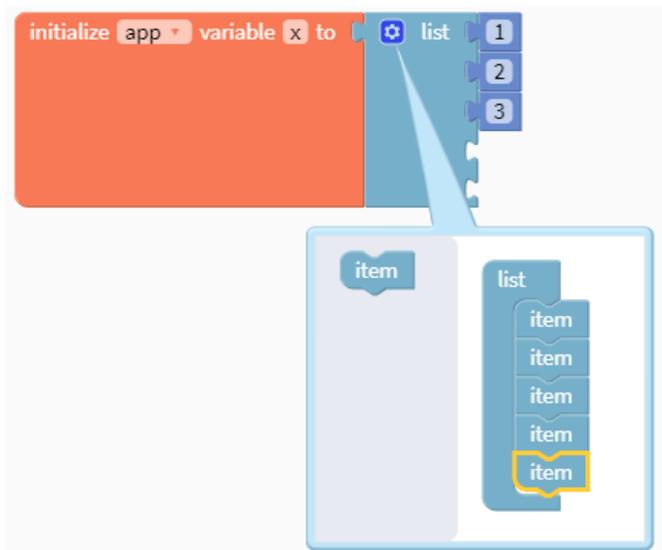
A continuación está la lista de los bloques que vas a necesitar:

Tipo de Bloque	Sección de bloques	Propósito
	Variables	Guardar la lista de agujeros.
	Math	Es un número, (Agrega los que sean necesarios)
	Variables	Guardar las cinco diferentes posiciones en X dónde se puede poner al Topo. (Son las posiciones X de los agujeros)
	List	Guardar una lista de números. (Vas a necesitar dos bloques de este tipo)
	Variables	Guardar las cinco diferentes posiciones en Y dónde se puede poner al Topo. (Son las posiciones Y de los agujeros)

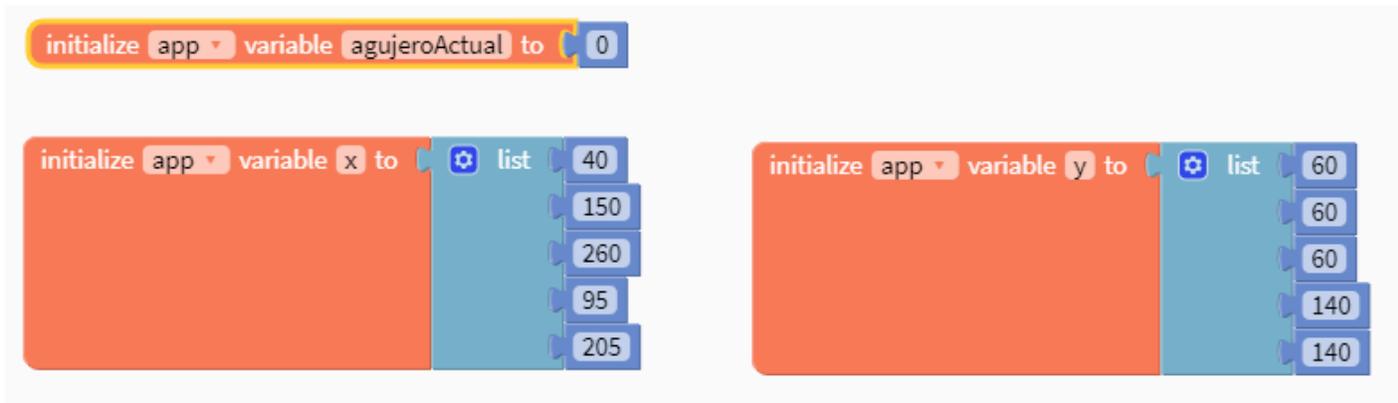
Al bloque de lista hay que agregarle dos elementos más para tener un total de cinco números en la lista, para eso da click en el engrane azul y agrega dos bloques “item” a la lista.



Quedará de la siguiente forma:



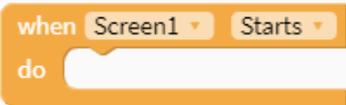
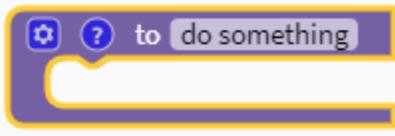
Ahora organiza todos los bloques de las variables y al finalizar deberás tener acomodados tus bloques de la siguiente manera:

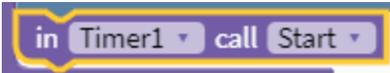


Iniciando la App

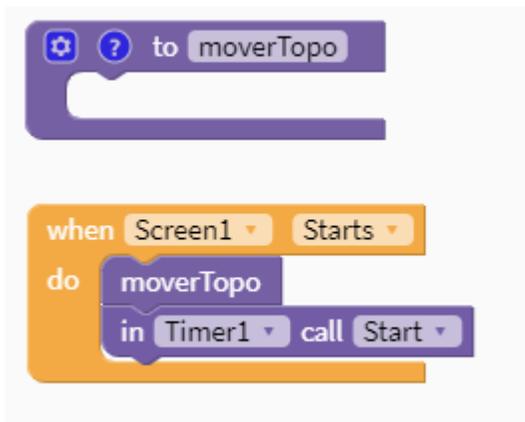
El primer evento que pasa cuando un programa se ejecuta es `Screen1.Starts`, por lo que vamos a poner el código de inicio en ese manejador. Específicamente, vamos a llamar a `MoverTopo`. Como aún no hemos programado `MoverTopo`, vamos a crear un procedimiento vacío con ese nombre que vamos a llenar después.

A continuación hay una tabla de los bloques que vas a crear.

Tipo de Bloque	Sección de bloques	Propósito
	Screen1	Especificar que debería pasar cuando la App inicia.
	Functions	Es un procedimiento o función, sigue los pasos indicados en el bloque.
	Functions	Procedimiento para mover el topo que vas a programar.

	Timer1	En el componente “Timer!” llamar el procedimiento “Iniciar”. (Esto iniciará el contador del temporizador)
---	--------	---

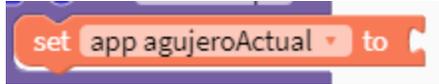
Acomoda los bloques de la siguiente forma. Para cambiar el nombre del procedimiento da click en “do something” (“Hacer algo”) y escribe “moverTopo”.

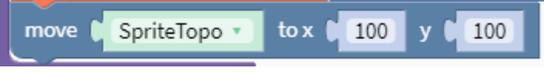
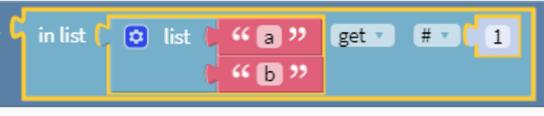
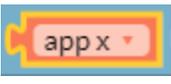
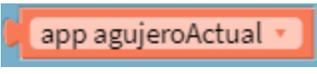
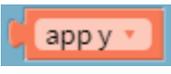
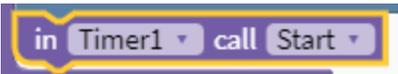


Moviendo el Topo

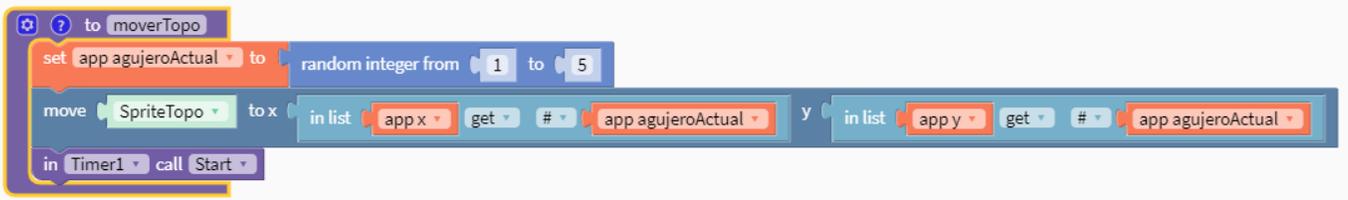
Ahora vamos a llenar el cuerpo del procedimiento **MoverTopo**, que vamos a llamar cuando inicie el programa, cuando el topo sea tocado, y cuando el temporizador se apague cada segundo. Lo que queremos hacer es escoger un agujero al azar y mover el topo sobre él.

Aquí está una imagen y tabla para los nuevos componentes.

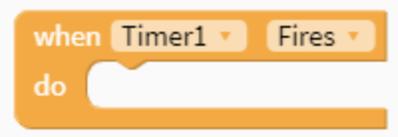
Tipo de Bloque	Sección de bloques	Propósito
	Variables	Cambiar el valor de la variable “agujeroActual” a un número aleatorio

	Math	Obtener un número aleatorio entre el 1 y el 100. Para proposito de esta practica será entre el 1 y el 5 ya que sólo tenemos 5 agujeros.
	Motion	Mover la posición X y Y de un objeto, en este caso "SpriteTopo" ya que queremos mover al Topo.
	Lists	Obtener el elemento en la posición 1 de la lista.(Necesitarás dos bloques como este)
	Variables	Obtener el valor de la variable "X". (Que es una lista de valores)
	Variables	Obtener el valor de la variable "agujeroActual"
	Variables	Obtener el valor de la variable "Y". (Que es una lista de valores)
	Timer1	En el componente "Timer!" llamar el procedimiento "Iniciar". (Esto iniciará el contador del temporizador)

Acomoda los bloques de la siguiente forma.



Ahora necesitamos especificar que **moverTopo** deberá ser llamado cada vez que el temporizador de **Timer1** se apague. Sólo necesitamos dos bloques para lograrlo:

Tipo de Bloque	Sección de bloques	Propósito
	Timer1	Cuando el temporizador se apaga
	Functions	Mover el Topo.

Los bloques quedarán de la siguiente forma:



Registrando los toques

Finalmente, necesitamos especificar qué pasa cuando el topo es tocado. Específicamente queremos que:

1. Se incremente la puntuación.
2. Hacer que el teléfono vibre brevemente.

3. Mover el Topo.

Podemos fácilmente traducirlo a bloques:

Tipo de Bloque	Sección de bloques	Propósito
	Events	Cuando el "SpriteTopo" es tocado o se hace click.
	EtiquetaValorPuntuacion	Cambiar el valor de la propiedad "Text" de "EtiquetaValorPuntuacion"
	Math	Hacer una suma
	EtiquetaValorPuntuacion	Obtener el valor de la propiedad "Text" del "EtiquetaValorPuntuacion"
	Device	Vibrar
	Functions	Mover el Topo.

Los bloques deberían quedar de la siguiente forma:

```

when SpriteTopo is clicked
  do
    from EtiquetaValorPuntuacion set Text to 1 + from EtiquetaValorPuntuacion get Text
    vibrate
    moverTopo
  
```

Programa Final

```

initialize app variable agujeroActual to 0

initialize app variable x to list 40 150 260 95 205
initialize app variable y to list 60 60 60 140 140

to moverTopo
  set app agujeroActual to random integer from 1 to 5
  move SpriteTopo to x in list app x get # app agujeroActual y in list app y get # app agujeroActual
  in Timer1 call Start
end to moverTopo

when Screen1 Starts
  do
    moverTopo
    in Timer1 call Start
  end do

when Timer1 Fires
  do
    moverTopo
  end do

when SpriteTopo is clicked
  do
    from EtiquetaValorPuntuacion set Text to 1 + from EtiquetaValorPuntuacion get Text
    vibrate
    moverTopo
  end do
  
```

Prueba tu app:

Para probar tu aplicación haz clic en “Live Test” (“Prueba en vivo”). Si el topo no se mueve, tócalo una vez, puede fallar en el Live Test.

Para probarlo en tu celular abre la app Thinkable y presiona: “Need a code? Click here” (“¿Necesitas un código? Haga clic aquí”) e ingresa el código en la pantalla de “Live Test” en tu computadora y da click en Connect.

Hackea tu app!

- Cambia el tipo de letra, colores, etc.
- Agregar el botón de Reiniciar para regresar la puntuación a 0.
- Hacer que la puntuación no sólo dependa del número de golpes sino también del número de fallos y topos que se escapan.
- Incrementando la velocidad del topo cuando el jugador está jugando bien y disminuirla si está jugando mal.
- Agregar un segundo topo con un temporizador distinto.

¿Qué aprendiste en este tutorial?

En este tutorial cubrimos cómo:

- Diseñar un juego básico usando el componente **Canvas**
- Poner un **Sprite** sobre otro **Sprite**, usando la propiedad **Z** para controlar cuál va al frente.
- Usar el componente **Timer** para controlar el juego.
- Crear un **procedimiento** y llamarlo desde varios lugares.