

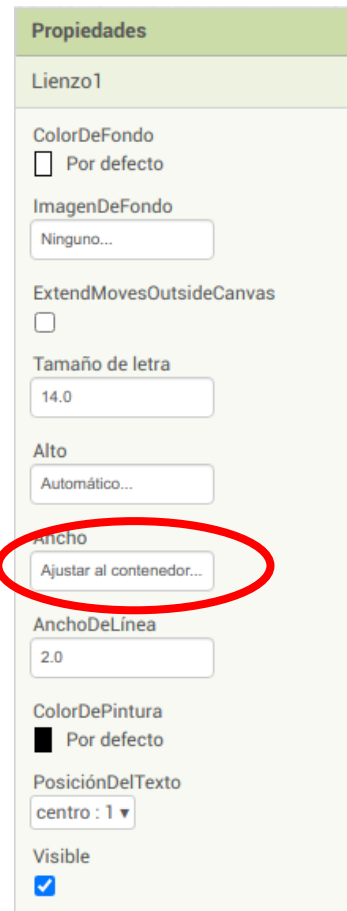


Lienzos y animaciones

547120 Introducción a ELN
mariomedina@udec.cl

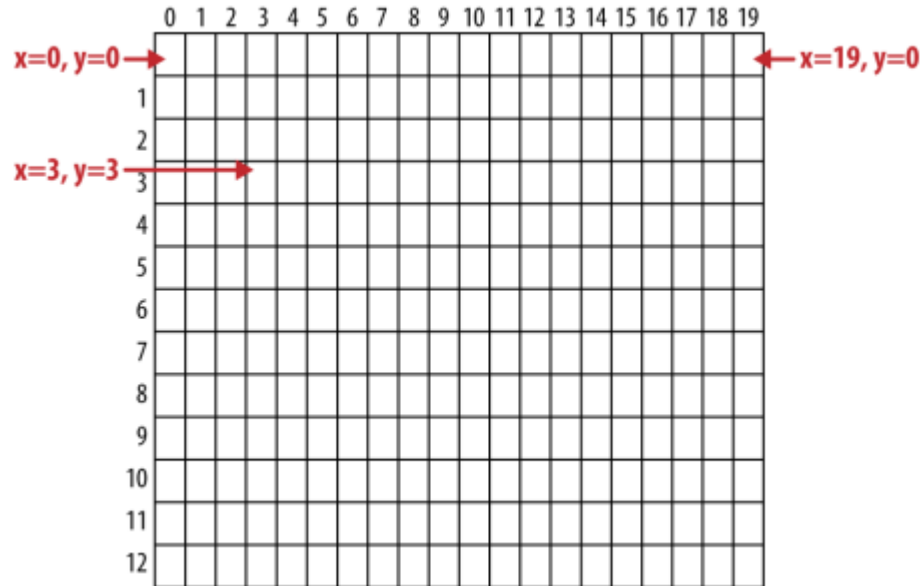
Lienzos (*Canvas*)

- Componente de la interfaz de usuario que se usa para dibujos y animaciones
 - Panel rectangular de dos dimensiones y sensible al tacto



Lienzos (*Canvas*)

- Sistema de coordenadas posiciona el punto (0,0) en la esquina superior izquierda de la pantalla





Animaciones

- Se pueden crear animaciones en *Lienzos* usando componentes *Pelota* y *SpriteImagen*
 - Un *Sprite* es una imagen bidimensional que se proyecta encima de otros objetos en un *Lienzo*
 - Puede ser leído desde un archivo PNG, GIF, JPG, WEBP
 - Una *Pelota* es un *Sprite* circular predefinido, con propiedades *Radio* y *ColorDePintura*

Posición de una *SpriteImagen*

- La posición de referencia de una *Pelota* o una *SpriteImagen* es la esquina superior izquierda
 - En el caso de la *Pelota*, se puede cambiar esta referencia al centro de ésta
- Si una *Pelota* o *SpriteImagen* se sale de la pantalla, ésta puede ser reubicada usando *MoverDentroDeLimites*

llamar Pelota1 ▾ .MoverDentroDeLimites



Animaciones

- Tanto para la *Pelota* como para *SpriteImagen*, se puede animar su movimiento usando las propiedades *Velocidad*, *Dirección* e *Intervalo*
 - *Direccion* usa las coordenadas de AppInventor
 - *Velocidad*: distancia en pixels a moverse en cada intervalo de tiempo
 - *Intervalo*: tiempo en milisegundos entre movimientos del objeto

Coordenadas en AppInventor

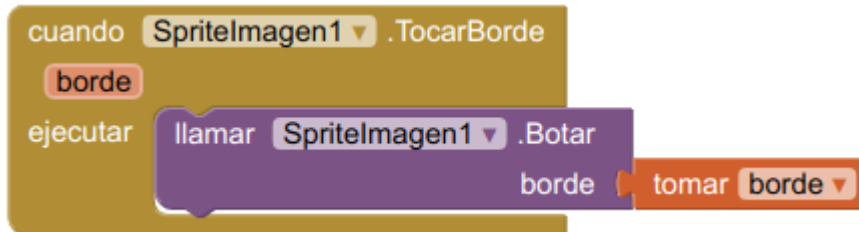
- Bordes

- 1: Norte
- 2: Noreste
- 3: Este
- 4: Sureste
- -1: Sur
- -2: Suroeste
- -3: Oeste
- -4: Noroeste



Rebotes

- *Sprites* y pelotas en AppInventor pueden rebotar en las paredes del lienzo
 - Procedimiento *Botar* hace que la pelota rebote en *borde*, donde *borde* es un entero entre -4 y 4, de acuerdo con las coordenadas anteriores



Detección de colisiones

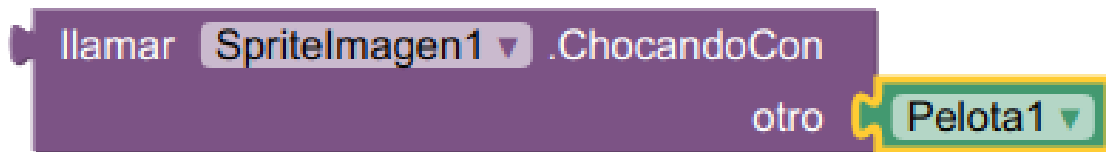
- AppInventor genera evento cuando un objeto choca con otro y también cuando estos dos objetos se separan después de haber chocado

cuando Spritelmagen1 ▾ .EnColisiónCon
otro
ejecutar poner Spritelmagen1 ▾ . Rota ▾ como cierto ▾

cuando Spritelmagen1 ▾ .FinDeColisiónCon
otro
ejecutar poner Spritelmagen1 ▾ . Rota ▾ como cierto ▾

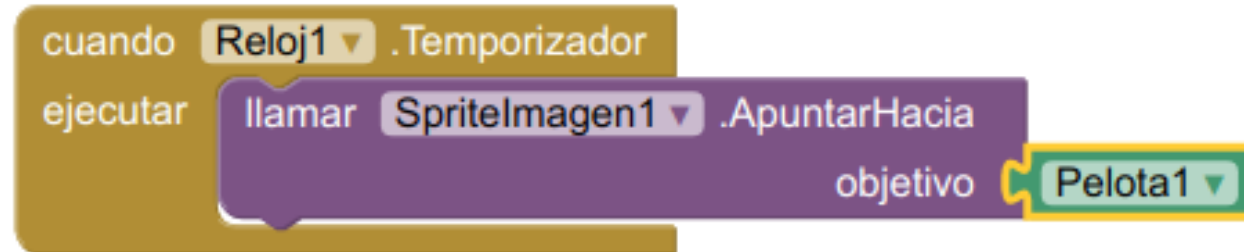
Detección de colisiones

- Bloque retorna verdadero si dos objetos *Pelota* o *SpriteImagen* están en colisión
- Objetos *Pelota* y *SpriteImagen* poseen referencia a sí mismo para permitir estas comparaciones



Animación con eventos de *Reloj*

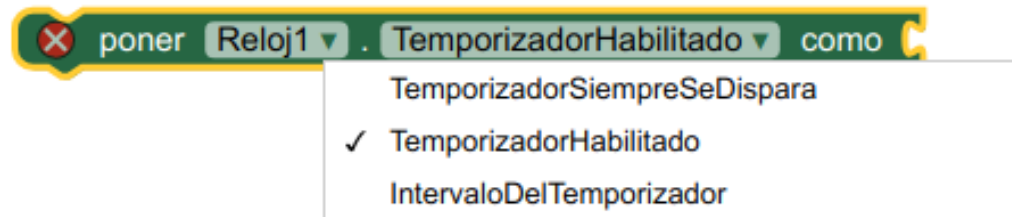
- Cambiar objetos en respuesta a eventos periódicos internos generados por un componente *Reloj*
 - Componente no visible de la UI
 - Usar un Temporizador



Sensores		
	Acelerómetro	?
	LectorCódigoDeBarras	?
	Barometer	?
	Reloj	?
	GyroscopeSensor	?
	Hygrometer	?
	LightSensor	?
	SensorDeUbicación	?
	MagneticFieldSensor	?
	NearField	?
	SensorDeOrientación	?
	Pedometer	?
	ProximitySensor	?
	Thermometer	?

Propiedades del *Reloj*

- *TemporizadorSiempreSeDispara* activa el temporizador aun cuando la app no esté en pantalla
- *TemporizadorHabilitado* activa el temporizador
- *IntervaloDelTemporizador* se da en milisegundos



Ejemplo usando Temporizador

- En el ejemplo, la pelota se mueve 5 pixels hacia la esquina inferior derecha cada 50 ms

