

## **Código Pi** **Educación Secundaria**

### **Producciones musicales**

**APRENDER  
CONECTADOS**



Ministerio de Educación,  
Cultura, Ciencia y Tecnología  
Presidencia de la Nación



# Autoridades

**Presidente de la Nación**

Mauricio Macri

**Jefe de Gabinete de Ministros**

Marcos Peña

**Ministro de Educación, Cultura, Ciencia y Tecnología**

Alejandro Finocchiaro

**Secretario de Gobierno de Cultura**

Pablo Avelluto

**Secretario de Gobierno de Ciencia, Tecnología e  
Innovación Productiva**

Lino Barañao

**Titular de la Unidad de Coordinación General del  
Ministerio de Educación, Cultura, Ciencia y Tecnología**

Manuel Vidal

**Secretaria de Innovación y Calidad Educativa**

Mercedes Miguel

**Directora Nacional de Innovación Educativa**

María Florencia Ripani

ISBN en trámite

Este material fue producido por el Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación en el marco del Plan Aprender Conectados.

# Índice

Ficha técnica .....	5
1. Inicio.....	7
2. Desarrollo .....	8
3. Cierre.....	14

## Ficha técnica

Nivel educativo	Educación Secundaria.
Año	2do/3ero.
Área del conocimiento	Matemática / Educación artística.
Tema de la clase	Creación de una estructura de programación para generar sonidos, utilizando estructuras repetitivas y vinculando la actividad con los conceptos de matemáticas relacionados.
NAP de matemática relacionados	En relación con el número y las operaciones: <ul style="list-style-type: none"><li>• El reconocimiento y uso de los números racionales en situaciones problemáticas que requieran:<ul style="list-style-type: none"><li>° interpretar, registrar, comunicar y comparar números.</li></ul></li></ul>
NAP de educación artística relacionados	En relación con las prácticas musicales y su producción: <ul style="list-style-type: none"><li>• El dominio inicial de los modos de producción en realizaciones musicales grupales sean propias y/o de otros, donde las estrategias compositivas impliquen el uso de la repetición, la reaparición, la variación y el cambio en torno a sus relaciones y búsquedas de sentido estético.</li></ul>

## APRENDER CONECTADOS

---

Habilidad de Programación y robótica relacionada:	<ul style="list-style-type: none"><li>• Resolver problemas a partir de su descomposición en partes pequeñas y aplicando diferentes estrategias, utilizando entornos de programación.</li></ul>
Duración	2 clases.
Materiales	Una computadora por grupo.
Desafíos pedagógicos	<ul style="list-style-type: none"><li>• Abordar situaciones que presenten desafíos: analizar el problema, establecer relaciones, ensayar posibles soluciones, poniendo en juego formas de pensamiento crítico y creativo.</li></ul>
Resumen de la actividad	Cada grupo deberá crear una producción musical. Se trabajará sobre el concepto de notas MIDI y su conversión a notas musicales, utilización de sonidos pregrabados y la reproducción de dos melodías al mismo tiempo.

---

## Inicio

¡Seguimos creando con Sonic Pi!

En esta actividad vamos hacer que las melodías suenen más interesantes utilizando distintas notas, cambiando los sonidos del sintetizador, haciendo variaciones en los bucles e incorporando nuevos conceptos básicos de programación.

## 2. Desarrollo

Para reproducir una nota con Sonic Pi sólo tenemos que poner una instrucción, como por ejemplo:

```
play :C5
```

Esta instrucción ¿sólo contiene la información acerca de la nota que reproducirá?

Los valores numéricos que estuvimos usando hasta ahora después de la instrucción `play` representan notas MIDI.

Ahora bien, ¿De qué se trata este término MIDI?

El sistema MIDI lleva mensajes de eventos que especifican notación musical, tono y velocidad; señales de control para parámetros musicales. Estos datos también pueden ser grabados en un hardware o software llamado secuenciador, el cual permite editar la información y reproducirla posteriormente. (Fuente: Wikipedia)

Estas notas manejan valores entre los números entre 0 y 127. De esta forma será más fácil probar rápidamente tus notas y ajustarlas bajando su valor (bajando la nota) o incrementándose (haciendo que la afinación sea más alta).

Recordemos que podemos repetir algo infinitamente usando `live_loop`. Por ejemplo:

```
live_loop :escalera
do
  play :C5
  sleep 1
  play :D5
end
sleep 1
play :E5
sleep 1
play :D5
sleep 1
```

Utilizando la instrucción **amp** y un valor pequeño el sonido se oirá en un volumen más bajo, y con la instrucción **amp** y un valor más alto se oirá amplificado:

```
live_loop :escalera
do
  play :C5, amp:1
  sleep 1
  play :D5, amp:0.5
  sleep 1
end
  play :E5, amp:2
  sleep 1
  play :D5, amp:0.5
  sleep 1
```

## Usando samples

No solamente podés crear música en Sonic Pi a partir de notas individuales, sino que también podés crear música a partir de samples. Los samples son sonidos pre-grabados que podés incluir en tu programación. ¡Es una manera realmente simple de hacer que tu música suene increíble!

Con los *samples* podrás incluir dentro de la estructura que creaste sonidos como: ambiente, percusión, de base, etc.

Para usar un *sample* tenés que agregar la línea de código

**sample: nombre del sample**

en la secuencia de tu programa donde querés que se reproduzca.

Debajo de la pantalla a la izquierda encontrarás los *samples* disponibles



## APRENDER CONECTADOS

Y a su lado, la forma en que se escribe la instrucción con sus diferentes variantes:

```
sample :bd_sone  
sample :bd_haus  
sample :bd_zome  
sample :bd_boom  
sample :bd_klub  
sample :bd_fat  
sample :bd_tek  
sample :bd_mehackit
```

En este ejemplo, `loop_amen` es el nombre del sample:

```
2.times do  
  sample :loop_amen  
  sleep 1.753  
end
```

También podemos usar un *sample* de redoble de tambor (`drum_roll`) para crear un poco de suspenso:

```
2.times do  
  sample :drum_roll  
  sleep 1  
end
```

## Un poco de ritmo

Ahora vamos a intentar que suenen varias cosas a la vez en nuestra canción. ¿Qué tal una batería?

```
live_loop :escalera do
  sample :loop_amen, rate: 0.5
  play :C5
  sleep 0.5
  play :D5
  sleep 1
  play :E5
  sleep 2
end
```

Veamos el código para escuchar la batería sola:

```
live_loop :bateria do
  sample :loop_amen, rate: 0.5
  sleep 3.5
end
```

Podemos dejar a la batería trabajando sola, y en otro bucle live\_loop diferente crear una melodía, buscando que coordine bien el ritmo:

```
live_loop :bateria do
  sample :loop_amen, rate: 0.5
  sleep 3.5
end

live_loop :notas do
  play :C5
  sleep 3.5
end
```

## APRENDER CONECTADOS

Para modificar la velocidad de reproducción del *sample* `live_loop`, usamos el `rate: 0.5` haciendo que se reproduzca a la mitad de velocidad (lo estamos “ralentizando”)

Cuando especificamos `rate: 1` el *sample* se reproduce a la velocidad normal. Si usáramos `rate: 1.5`, estaríamos acelerando un poco la reproducción del *sample*.

Podemos no usar `rate:` ya que es opcional, y por defecto toma el valor es 1.

Para pensar:

En el ejemplo, hacemos sonar una nota y luego utilizamos `sleep 3.5`

¿Qué ocurre si en ese segundo *bucle* ponemos `sleep 3`? ¿O si usamos `sleep 4`?

¡No van al mismo ritmo!

¿Será necesario que los tiempos de la percusión y de la melodía sean exactamente iguales?

Si todavía no encontraste la respuesta, veamos qué sucede en el próximo ejemplo.

En este otro ejemplo tenemos un bombo de batería en un *bucle* infinito, mientras una melodía se repite dos veces:

```
live_loop :bata do
  sleep 0.4
  sample :drum_heavy_kick
end
2.times do
  play 83
  sleep 0.4
  play 84
  sleep 0.8
  play 83
  sleep 0.4
  play 82
  sleep 0.8
end
```

## APRENDER CONECTADOS

Atención: ¿podés anticipar qué pasaría si en el segundo *bucle*, el de la melodía, en lugar de “sleep 0.4” y “sleep 0.8” utilizáramos “sleep 0.5”? Verificá tu respuesta modificando los valores en el código.

¿Qué ocurre si el tiempo del segundo *bucle* no coincide con el primero?

Importante: para que la melodía y la percusión estén en el mismo ritmo, será necesario que los valores de sus tiempos sean **múltiplos**.

En el ejemplo anterior el bombo se reproduce infinitamente con silencios de tiempo 0.4, mientras que las notas de la melodía se reproducen en una secuencia con silencios de tiempos 0.4 - 0.8 - 0.4 - 0.8, dos veces.

Tenemos mucho más para investigar: escribiendo la palabra *sample* y pulsando la barra espaciadora se desplegarán las distintas opciones. Probemos con aquellos que comiencen con “*drum*” para explorar los instrumentos de percusión.



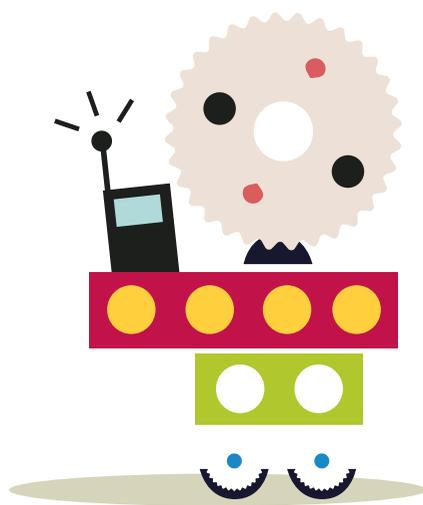
```
Run ▶ Stop ■ Rec ● Save ♥ Load 📁
1 2.times do
2  sample
3  sleep 1
4 end
5
```

- :drum\_cowbell
- :drum\_cymbal\_closed
- :drum\_cymbal\_hard
- :drum\_cymbal\_open
- :drum\_cymbal\_pedal
- :drum\_cymbal\_soft
- :drum\_heavy\_kick
- :drum\_roll
- :drum\_snare\_hard

### 3. Cierre

En esta actividad estuvimos trabajando varios elementos nuevos: la amplificación, que nos brinda matices de volumen, y la percusión con el uso de los samples. También aprendimos que podemos dejar trabajando un *bucle*, mientras en otro hacemos una melodía, y que para ello es necesario valores múltiples en los tiempos.

Ahora con estos potentes recursos, te proponemos que sigas explorando para lograr nuevas producciones musicales, por ejemplo creando una pista de bajos usando las notas que quieras, y para hacerlo más interesante, cambiando el sintetizador.



**APRENDER  
CONECTADOS**



Ministerio de Educación,  
Cultura, Ciencia y Tecnología  
Presidencia de la Nación