



In this application, you will create a program that lets you control the sound coming from the speaker based on changing the brightness sensed by the light sensor and your own moving hand— hence, Hand Music!

#### Objectives:

- Write a program that converts brightness into sounds
- Review musical note frequencies and the twelfth root of 2 principle

Write a program that reads the BRIGHTNESS of a light sensor and plays a different sound depending on the brightness. There are two possible options for the sound:

- Play a frequency in the audible range (perhaps 100Hz – 1000Hz)
- Play a musical note (one of the specific harmonious sounds found on a piano or other musical instrument).

The first option would just play what sounds like ‘noise’. The second option will sound more like music, but the mathematics is a bit more complex.

This program makes the TI-Innovator™ Hub behave like a *theremin*.

#### Getting Started

1. Start a new program, and name it APPLIC3
2. Add **Disp**, add an opening and closing set of quotation marks, and type the text Hand Music! as shown.
3. Add a **While** loop to **READ** the **BRIGHTNESS** sensor and **Get** its value into a variable **b**.
4. Add the statement to play a sound.
  - Notice that we’re using the variable **B** for reading BRIGHTNESS and the variable **F** for playing the SOUND.

Your task is to complete the missing code that converts the BRIGHTNESS into an audible *sound* or a *musical note*.

For sound, use a frequency between 100 and 1000 (or two frequencies of your choice).

For musical notes, try a range starting with A1 (55Hz) and going up 50 notes. (You should refer to the activity studied in Unit 2, Skill Builder 3 (we called it prgmSOUND2) which played the 12 notes in an octave.

For the musical notes, you will need to convert your value to a whole number so that a note ‘number’ is correctly represented. You can use either the **int( )** function or the **round( ,0)** function.

**int(X)→X** gives the largest integer less than **X**.

**round(X,0)→X** rounds **X** to the nearest integer.

```

*applic3
Disp "Hand Music!"
b:=2
While b>1
  Send "READ BRIGHTNESS"
  Get b
  Send "SET SOUND eval(F)"
  Wait 0.2
EndWhile
EndPrgm
    
```

```

*applic3
Disp "Hand Music!"
2 0 55
2 0 55
2 0 55
2 0 55
2 0 55
2 0 55
2 0 55
    
```

**Teacher Tip:**

For just playing a sound, students need to convert B in (0,100) to F in (100,1000).

B	F
0	100
100	1000

Calculate the slope of the line containing these two points, and then write the equation for F in terms of B.

$$M = (1000-100)/(100-0) = 9$$

so

$$F = 9*B+100$$

For playing musical notes, recall the  $F \cdot 2^{(1/12)}$  property of note intervals. A1=55Hz, and we want 50 notes. So the first note is #0 and the last one is #49 (programmers always start counting with 0). Also, remember that the note number must be an integer, so we must use either `int()` or `round()`. We use a 2-step computation to make it clear:

$$N = \text{int}(49B/100) \quad \text{--note number}$$

$$F = 55 \cdot 2^{(N/12)} \quad \text{--note frequency}$$