

Progress/ Fails & Success:

<https://photos.app.goo.gl/BDyUj3jauvgwQG3Z8>

Final Product:

<https://photos.app.goo.gl/jM5DsWhScinyWccn7>

Code

```
/*  
Melody
```

```
Plays a melody on a Circuit Playground.
```

```
Adapted from:
```

```
https://www.arduino.cc/en/Tutorial/BuiltInExamples/toneMelody
```

```
*/
```

```
#include <Adafruit_CircuitPlayground.h>
```

```
// A few music note frequencies as defined in this tone example:
```

```
// https://www.arduino.cc/en/Tutorial/toneMelody
```

```
#define NOTE_C4 262
```

```
#define NOTE_D4 294
```

```
#define NOTE_E4 330
```

```
#define NOTE_F4 349
```

```
#define NOTE_G4 392
```

```
#define NOTE_A4 440
```

```

#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_D5 587
#define NOTE_E5 659
#define NOTE_F5 699
#define NOTE_G5 784
#define NOTE_A5 932

//-----Modify The Code Below Only-----
// Notes for the song. A space represents a rest
// Add the notes needed using the note sheet. Spaces acts like breaks
char song[] = " f a c e c a c a e d b ab ab b d c a ga ga agabc b ab ab b d c a a
dffdgg a d ffdfg a ebe beg e ";

// beats per note
// Add the beats needed to complete the song.
// The amount of beats should match the amount of notes and spaces
int beats[] = { 1, 1, 1, 1/2, 1, 1, 1, 1, 1, 3, 1, 1/2, 2, 1/2, 2, 1, 1,
1, 1, 1/2, 2, 1/2, 2, 1, 1/2, 1, 1, 1, 1/2, 1, 1/2, 1, 1, 1, 3, 2, 1, 1/2,
1, 1/2, 1, 1/2, 1, 1, 1, 1, 1/2, 1, 1/2, 1, 1/2, 1, 2, 1, 1/2, 1, 1/2,
1, 1/2, 1, 1/2, 1, 1/2, 1, 1, 1 };
//-----Modify ONLY The Code Above To Update the Melody -----

//----- Do Not Edit The Code Below-----
// speed of song
int tempo = 300;
// convenience notes means we don't need Pitches.h
// names of the notes
char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
// corresponding tones for C4 - B4, C5 (looked up values from Pitches.h)
int tones[] = { NOTE_C4, NOTE_D4, NOTE_E4, NOTE_F4, NOTE_G4, NOTE_A4, NOTE_B4,
NOTE_C5, NOTE_D5, NOTE_E5, NOTE_F5, NOTE_G5 };

```

```

// play the song using the variables: song, songLength, beats
void playSong()
{
    delay( 5000 );

    // for each note in the melody:
    for (int noteIndex = 0; noteIndex < sizeof(song); noteIndex++)
    {
        // if it's a space
        if (song[noteIndex] == ' ')
        {
            // rest
            delay( beats[noteIndex] * tempo/5);
        }
        // otherwise it's the name of a note
        else
        {
            // play the note at that index for the specified time
            playNoteByName(song[noteIndex], beats[noteIndex] * tempo);
        }
        // pause between notes
        delay(tempo);
    }
}

// play the tone corresponding to the note name
void playNoteByName(char noteName, int duration)
{
    // loop through the names
    for (int i = 0; i < sizeof(names); i++)
    {
        // if we found the right name
        if (names[i] == noteName)
        {
            // play the tone at the same index
            playNote( tones[i], duration);
            // don't bother looking through the rest of the names
            break;
        }
    }
}

```

```

    }
}

int lightValue;

// Play a note of the specified frequency and for the specified duration.
// No tones will play if the slide switch is in the -/off position.
void playNote(int frequency, int duration) {
    lightValue = CircuitPlayground.lightSensor();
    Serial.print("Light Sensor: ");
    Serial.println(lightValue);
    // Check if the slide switch is off
    if ((!CircuitPlayground.slideSwitch()) || (lightValue < 20)) {
        // stop immediately without playing anything.
        return;
    }

    else {
        //Play the note for the specified duration.
        CircuitPlayground.playTone(frequency, duration, false);
        // wait for the note to play
        delay(duration+ duration/16);
    }
}

//-----Modify ONLY The Code Below To Respond to Light -----

/**
 * The code in this special method is executed
 * once when the microcontroller is turned on
 * (or the program is uploaded).
 */

void setup()
{
    // this code is executed once as the board turns on

```

```

    // we need to initialize it
    Serial.begin(9600);
    CircuitPlayground.begin();

}

/**
 * The code in this special method is constantly executed
 * after setup has occurred.
 */
void loop() {
    lightValue = CircuitPlayground.lightSensor();
    Serial.print ("Light Sensor: ") ;
    Serial.println (lightValue) ;
    // if there is light
    if ( lightValue > 30 )
        // turn off playNote
    // else
        // otherwise, playNote
    playSong();
    // the slide switch can (un)mute the melody, so we can leave it in the loop
    delay(70);
}

```