Arduino Tutorial 11 - Music by Tone and Pulse

Precis Arduino has a function (or command) called – TONE.

It can play notes to a speaker – for example an 8 ohm speaker with a 100 ohm resistor is series. The speaker earth is connected to Arduino ground. One end of the 100 ohm resistor is connected to the speaker and the other to one of the Arduino pins.

How to use : The function takes the form tone(pin_Number,Note, Duration);

The "pin_Number" can be any of the pins. (Avoid pin 3 and pin 11 as it interferes with PWM)
Use a variable to store the pin – eg int speaker = 8;

The "Note" is the frequency in hertz, and is usually contained in an array such as :

```
int melody[] = {
  NOTE_C4, NOTE_G3,NOTE_G3, NOTE_A3, NOTE_G3,0, NOTE_B3, NOTE_C4};
```

The "Duration" is how long the tone is played for – and is also contained in an array, such as :

```
int noteDurations[] = {
    4, 8, 8, 4,4,4,4,4 };
```

To play the melody, you use a loop – with a counter for the number of notes.

The number of notes can be stored in a variable – num_notes, and calculated as :

```
int  num_notes = sizeof(melody) / sizeof(int);  // calculates the number of notes
```

NOTE – the function sizeof returns the number of **bytes** the array occupies in memory.
Since each element of our "melody" arrray is an integer, we must divide by the size of an integer to arrive at the number of ELEMENTS (or notes) in the array.
A sample code to play a small melody is included below.

pitches.h To play the notes, you need to include a file called pitches.h
This file has the frequency of all the notes.
This is NOT a library file. It is added to your sketch in an extra TAB.
Since this file is part of your Arduino sketch, the command is
#include "pitches.h"
NOTE - As this is NOT a library file, the name is in inverted commas.
If it was a library file, it would need #include <pitches.h>

HOW to add pitches.h

After making your main sketch, you will notice on the right hand side of the IDE a small down arrow. This is used to add extra TABS, very similar to extra tabs in your web browser.

Click the down arrow (on the right hand side of the IDE) – click NEW TAB -
It asks for a NAME – type pitches.h then press enter.
COPY the data for pitches file from below (or from the main Arduino site), and paste it into the tab.
When you SAVE your sketch, an extra file will be created called pitches.h

```
//  Sample Code for Arduino using the NOTE command.
//  Connect an 8 ohm speaker in series with a 100 ohm resistor to pin 5.
// connect the speaker ground to the Arduino ground pin.

 #include "pitches.h"

  int melody[] = {
        NOTE_C4, NOTE_G3,NOTE_G3, NOTE_A3, NOTE_G3,0, NOTE_B3, NOTE_C4};

int noteDurations[] = {
                 4, 8, 8, 4,4,4,4,4 };

int num_notes = sizeof(melody) / sizeof(int);    // calculate the number of notes.
int speaker = 5;                                 //  pin that our speaker is connected to
void setup()
{
   for (int thisNote = 0; thisNote < num_notes; thisNote++)
   {
                // to calculate the note duration, take one second  divided by the note type.
                //e.g. quarter note = 1000 / 4, eighth note = 1000/8, etc.
        int noteDuration = 1000/noteDurations[thisNote];
        tone(speaker, melody[thisNote],noteDuration);

                // Allow delay between notes - 30%  works well:
        int pauseBetweenNotes = noteDuration * 1.30;
        delay(pauseBetweenNotes);
                // stop the tone playing:
        noTone(speaker);
  }
}

void loop()
{      // nothing needed here – we only want to play the tune once – hence  the code is in setup.
}
```

NOTE – you can change the pause between notes, by making it **variable**.
How – insert an extra “note” of 0 between each note in the melody array, and add a corresponding time into the note_duration array for the 0 note.

pitches.h file data

```
/*
Definition for Arduino pitches.h  file used with NOTE command.
*/

#define NOTE_B0  31
 #define NOTE_C1  33
 #define NOTE_CS1 35
 #define NOTE_D1  37
 #define NOTE_DS1 39
 #define NOTE_E1  41
 #define NOTE_F1  44
 #define NOTE_FS1 46
 #define NOTE_G1  49
 #define NOTE_GS1 52
 #define NOTE_A1  55
 #define NOTE_AS1 58
 #define NOTE_B1  62
 #define NOTE_C2  65
 #define NOTE_CS2 69
 #define NOTE_D2  73
 #define NOTE_DS2 78
 #define NOTE_E2  82
 #define NOTE_F2  87
 #define NOTE_FS2 93
 #define NOTE_G2  98
 #define NOTE_GS2 104
 #define NOTE_A2  110
 #define NOTE_AS2 117
 #define NOTE_B2  123
 #define NOTE_C3  131
 #define NOTE_CS3 139
 #define NOTE_D3  147
 #define NOTE_DS3 156
 #define NOTE_E3  165
 #define NOTE_F3  175
 #define NOTE_FS3 185
 #define NOTE_G3  196
 #define NOTE_GS3 208
 #define NOTE_A3  220
 #define NOTE_AS3 233
 #define NOTE_B3  247
 #define NOTE_C4  262
 #define NOTE_CS4 277
 #define NOTE_D4  294
 #define NOTE_DS4 311
 #define NOTE_E4  330
 #define NOTE_F4  349
 #define NOTE_FS4 370
 #define NOTE_G4  392
 #define NOTE_GS4 415
 #define NOTE_A4  440
 #define NOTE_AS4 466
```

```
 #define NOTE_B4  494
 #define NOTE_C5  523
 #define NOTE_CS5 554
 #define NOTE_D5  587
 #define NOTE_DS5 622
 #define NOTE_E5  659
 #define NOTE_F5  698
 #define NOTE_FS5 740
 #define NOTE_G5  784
 #define NOTE_GS5 831
 #define NOTE_A5  880
 #define NOTE_AS5 932
 #define NOTE_B5  988
 #define NOTE_C6  1047
 #define NOTE_CS6 1109
 #define NOTE_D6  1175
 #define NOTE_DS6 1245
 #define NOTE_E6  1319
 #define NOTE_F6  1397
 #define NOTE_FS6 1480
 #define NOTE_G6  1568
 #define NOTE_GS6 1661
 #define NOTE_A6  1760
 #define NOTE_AS6 1865
 #define NOTE_B6  1976
 #define NOTE_C7  2093
 #define NOTE_CS7 2217
 #define NOTE_D7  2349
 #define NOTE_DS7 2489
 #define NOTE_E7  2637
 #define NOTE_F7  2794
 #define NOTE_FS7 2960
 #define NOTE_G7  3136
 #define NOTE_GS7 3322
 #define NOTE_A7  3520
 #define NOTE_AS7 3729
 #define NOTE_B7  3951
 #define NOTE_C8  4186
 #define NOTE_CS8 4435
 #define NOTE_D8  4699
 #define NOTE_DS8 4978
//  end of data for   pitches.h   file
```

Christmas – Jingle Bells

```
//  To keep the melody array small, we have replaced  NOTE_  with  N in both the main sketch
//  and in the pitches.h   code.

int melody[] =
{
        NG4,NE5,ND5,NC5,NG4,NG4,NE5,ND5,NC5,NA4,
        NA4,NF5,NE5,ND5,NB4,NG5,NG5,NF5,ND5,NE5,
        NG4,NE5,ND5,NC5,NG4,NG4,NE5,ND5,NC5,NA4,
        NA4,NF5,NE5,ND5,NG5,NG5,NG5,NG5,NG5,NA5,NG5,NF5,ND5,NC5,NG5,
        NE5,NE5,NE5,NE5,NE5,NE5,NE5,NG5,NC5,ND5,NE5,
        NF5,NF5,NF5,NF5,NF5,NF5,NE5,NE5,NE5,NE5,NE5,ND5,ND5,NE5,ND5,NG5,
        NE5,NE5,NE5,NE5,NE5,NE5,NE5,NG5,NC5,ND5,NE5,
        NF5,NF5,NF5,NF5,NF5,NF5,NE5,NE5,NE5,NE5,NG5,NG5,NF5,ND5,NC5,
};


int noteDurations[] =
{
            8,8,8,8,2,8,8,8,8,2, // 10
            8,8,8,8,2,8,8,8,8,2, // 10
            8,8,8,8,2,8,8,8,8,2, // 10
            8,8,8,8,8,8,8,16,16,8,8,8,8,4,4, // 15
            8,8,4,8,8,4,8,8,8,8,2, // 11
            8,8,8,16,16,8,8,8,16,16,8,8,8,8,4,4, // 16
            8,8,4,8,8,4,8,8,8,8,2, // 11
            8,8,8,16,16,8,8,8,16,16,8,8,8,8,2, // 15
 };
```