

Piano

<http://www.keithwhor.com/music/>

Musical Keyboard - JS Dynamic Audio Synth

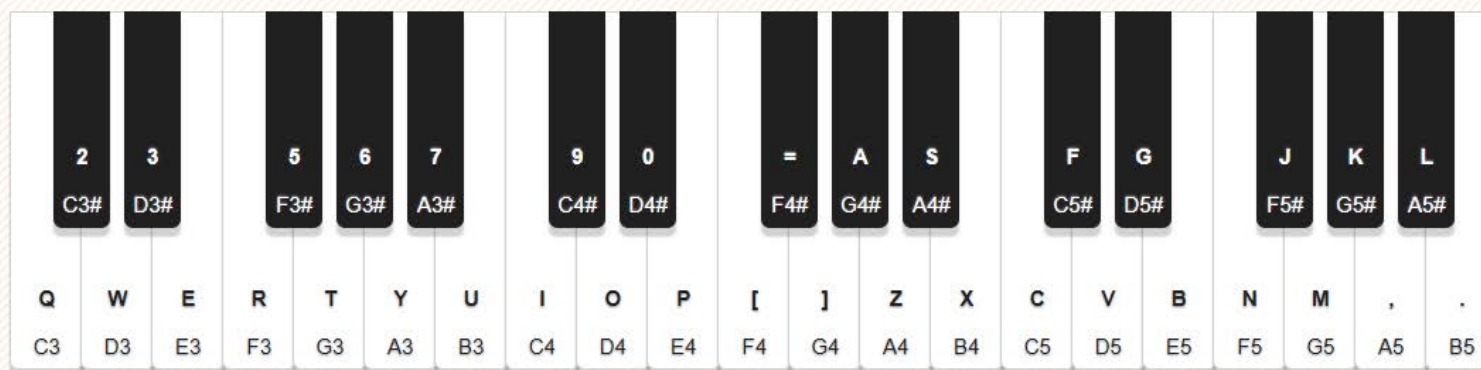
Fork me on GitHub

Created by [Keith William Horwood](#) © 2013

[\[Source Code and Usage \]](#)

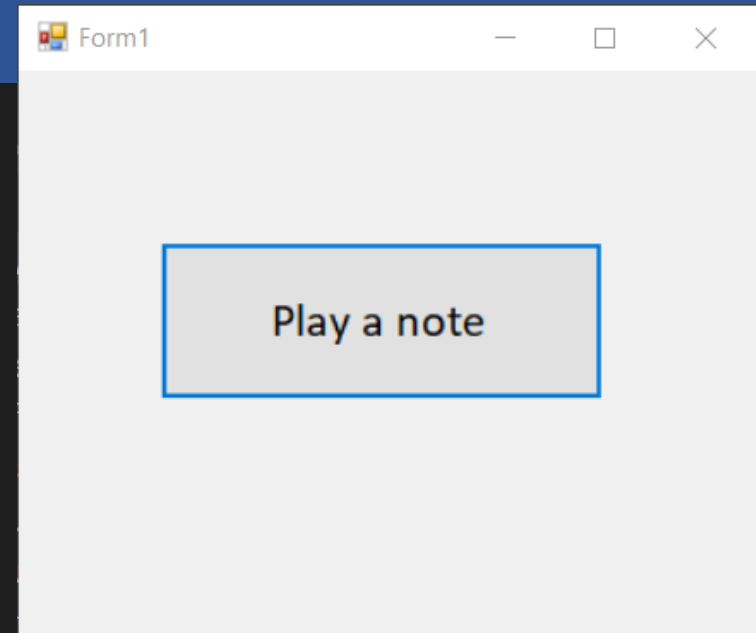
[\[Twitter \]](#)

Sound



Play a note

```
1  using System;
2  using System.Windows.Forms;
3  using System.Media;
4
5  namespace test
6  {
7      public partial class Form1 : Form
8      {
9          public Form1()
10         {
11             InitializeComponent();
12         }
13         private void buttonHello_Click(object sender, EventArgs e)
14         {
15             SoundPlayer player = new SoundPlayer("d:\\temp\\wav\\01c4.wav");
16             player.Play();
17         }
18     }
19 }
20
```



Mary had a little lamb

Mary had a little lamb

The musical notation is written on a single staff in 4/4 time. The key signature has one sharp (F#). The melody consists of the following notes: C4 (quarter), D4 (quarter), E4 (quarter), F#4 (quarter), G4 (quarter), A4 (quarter), B4 (half). The lyrics are: Ma - ry had a lit - tle lamb lit - tle lamb lit - tle lamb,.

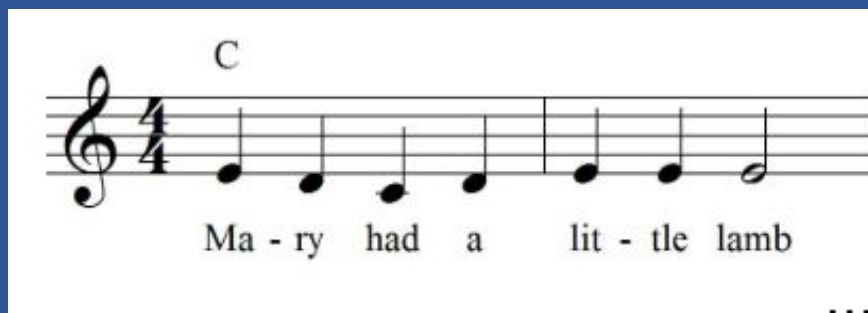
Play multiple notes

```
SoundPlayer player = new SoundPlayer("d:\\temp\\wav\\10A4.wav");
player.Play();
System.Threading.Thread.Sleep(800);
player = new SoundPlayer("d:\\temp\\wav\\08G4.wav");
player.Play();
System.Threading.Thread.Sleep(400);
player = new SoundPlayer("d:\\temp\\wav\\06F4.wav");
player.Play();
System.Threading.Thread.Sleep(800);
player = new SoundPlayer("d:\\temp\\wav\\08G4.wav");
player.Play();
System.Threading.Thread.Sleep(600);
player = new SoundPlayer("d:\\temp\\wav\\10A4.wav");
player.Play();
System.Threading.Thread.Sleep(600);
player = new SoundPlayer("d:\\temp\\wav\\10A4.wav");
player.Play();
System.Threading.Thread.Sleep(600);
```

Play using Array

```
// play chromatic scale
// use array and List to shorten code
string[] wavArray = {
    "01C4", "02Db4", "03D4", "04Eb4",
    "05E4", "06F4", "07Gb4", "08G4",
    "09Ab4", "10A4", "11Bb4", "12B4", "13C5"
};
List<SoundPlayer> playList = new List<SoundPlayer>();
// Create play list
foreach (string wa in wavArray)
    playList.Add(new SoundPlayer("d:\\temp\\" + wa + ".wav"));
// Play play list
foreach (SoundPlayer pl in playList)
    pl.PlaySync();
```

Music Note Symbols



♩ = **Whole note**: looks like a donut

♪ = **Half note**: we add a stem (vertical line) to the donut. The stem can go up or down. Look at any piece of music for examples.

♫ = **Quarter note**: we fill in the whole make it solid and keep the stem.

♫ = **Eighth note**: we add a flag to the stem.

♫ = **Sixteenth note**: we add two flags to the stem

Note with duration

```
public enum Duration
{
    WHOLE = 1600,
    HALF = WHOLE / 2,
    QUARTER = HALF / 2,
    EIGHTH = QUARTER / 2,
    SIXTEENTH = EIGHTH / 2,
}

public struct Note
{
    public string noteName;
    public Duration duration;
    public Note(string noteName, Duration duration)
    {
        this.noteName = noteName;
        this.duration = duration;
    }
}
```


Chromatic scale

The image displays a chromatic scale in treble clef, consisting of two staves. The first staff shows an ascending scale from C to C, and the second staff shows a descending scale from C to C. Each note is a half step (semitone) apart from the previous one. The notes are labeled with their names below the staff.

note names: C C# D D# E F F# G G# A A# B C

C B Bb A Ab G Gb F E Eb D Db C

Define Array of chromatic scale

```
string[] wavArray = {  
    "01C4", "02Db4", "03D4", "04Eb4",  
    "05E4", "06F4", "07Gb4", "08G4",  
    "09Ab4", "10A4", "11Bb4", "12B4", "13C5"  
};  
string[] noteName = {  
    "C4", "Db4", "D4", "Eb4",  
    "E4", "F4", "Gb4", "G4",  
    "Ab4", "A4", "Bb4", "B4", "C5"  
};
```

Dictionary of notes

```
// Create note list
List<SoundPlayer> noteList = new List<SoundPlayer>();
foreach (string wa in wavArray)
    noteList.Add(new SoundPlayer("d:\\temp\\" + wa + ".wav"));

// Create dic for all notes
Dictionary<string, SoundPlayer> myDic = new Dictionary<string, SoundPlayer>();
for (int i = 0; i <= wavArray.Length - 1; i++)
    myDic.Add(noteName[i], noteList[i]);
```

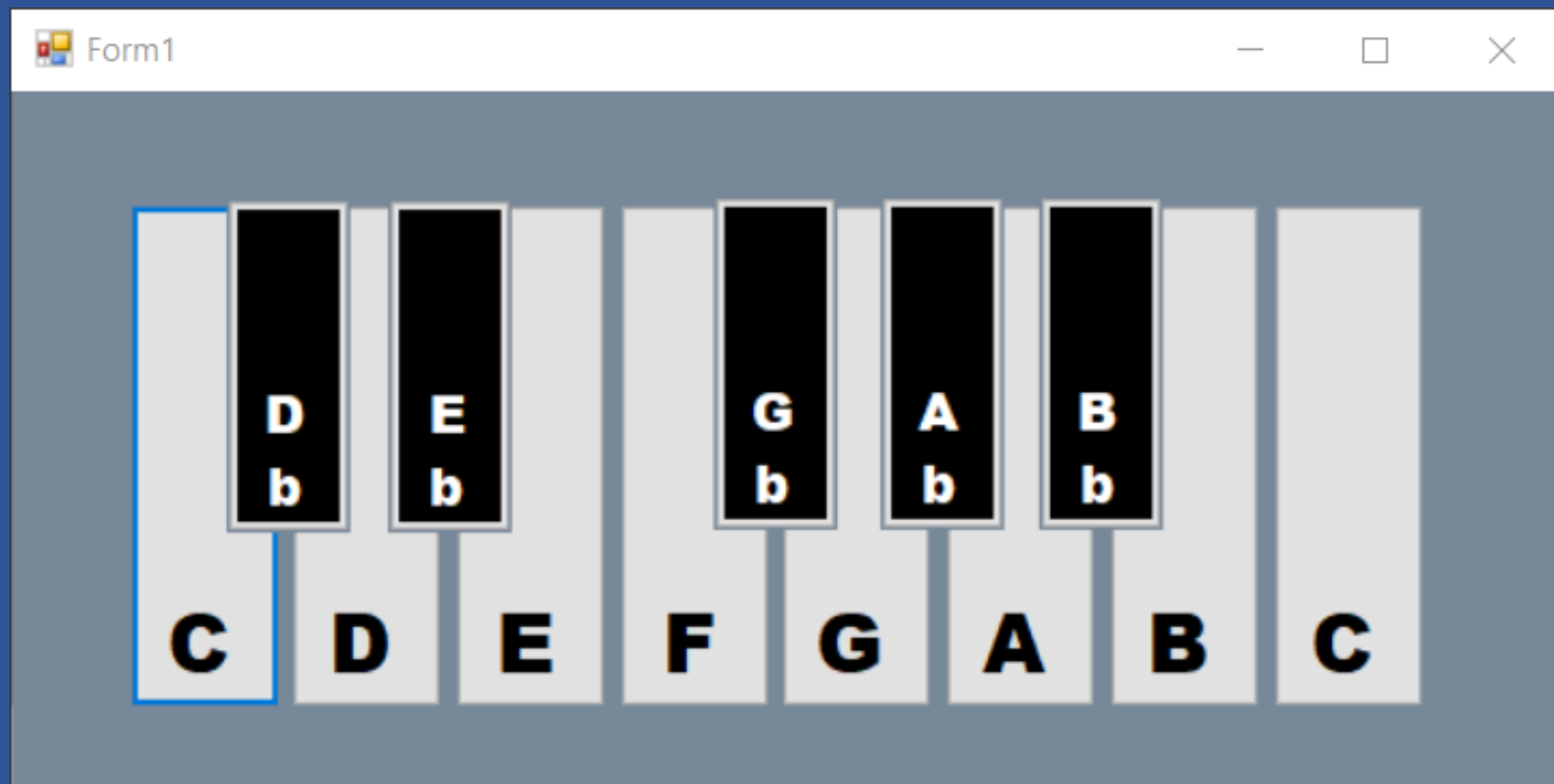
Create note list

```
// First few notes of the song, "Mary Had A Little Lamb".
List<Note> mySong = new List<Note>();
mySong.Add(new Note("B4", Duration.QUARTER));
mySong.Add(new Note("A4", Duration.QUARTER));
mySong.Add(new Note("G4", Duration.QUARTER));
mySong.Add(new Note("A4", Duration.QUARTER));
mySong.Add(new Note("B4", Duration.QUARTER));
mySong.Add(new Note("B4", Duration.QUARTER));
mySong.Add(new Note("B4", Duration.HALF));
mySong.Add(new Note("A4", Duration.QUARTER));
mySong.Add(new Note("A4", Duration.QUARTER));
mySong.Add(new Note("A4", Duration.HALF));
mySong.Add(new Note("B4", Duration.QUARTER));
mySong.Add(new Note("D4", Duration.QUARTER));
mySong.Add(new Note("D4", Duration.HALF));
```

Play song

```
// Play song
foreach (Note n in mySong)
{
    SoundPlayer myPlayer = myDic[n.noteName];
    myPlayer.Play();
    System.Threading.Thread.Sleep(Convert.ToInt32(n.duration));
}
```

Piano



```
1  using System;
2  using System.Media;
3  using System.Windows.Forms;
4
5  namespace _005_Piano
6  {
7      public partial class Form1 : Form
8      {
9          public Form1()...
13
14         private void buttonC4_Click(object sender, EventArgs e)...
19         private void buttonDb_Click(object sender, EventArgs e)...
24         private void buttonD_Click(object sender, EventArgs e)...
29         private void buttonEb_Click(object sender, EventArgs e)...
34         private void buttonE_Click(object sender, EventArgs e)...
39         private void buttonF_Click(object sender, EventArgs e)...
44         private void buttonGb_Click(object sender, EventArgs e)...
49         private void buttonG_Click(object sender, EventArgs e)...
54         private void buttonAb_Click(object sender, EventArgs e)...
59         private void buttonA_Click(object sender, EventArgs e)...
64         private void buttonBb_Click(object sender, EventArgs e)...
69         private void buttonB_Click(object sender, EventArgs e)...
74         private void buttonC5_Click(object sender, EventArgs e)...
79     }
80 }
```