

[View on GitHub](#)

audiosynth

JS Dynamic Audio Synth

[Download this project as a .zip file](#) [Download this project as a tar.gz file](#)

audiosynth

Dynamic waveform audio synthesizer, written in Javascript.

Generate musical notes dynamically them in your browser using the HTML5 Audio Element.

No static files required. (Besides the source, of course!)

Demo

To see a demo of audiosynth in action, visit <http://keithwhor.com/music/>

Installation

Assuming audiosynth.js is in your current directory, import package using:

```
<script src="audiosynth.js"></script>
```

Usage

audiosynth implements a singleton class, AudioSynth. By default, the global (window) variable Synth is the instance of the class.

Any attempt to instantiate new AudioSynth object will only create references to the original object.

```
Synth instanceof AudioSynth; // true
```

```
var testInstance = new AudioSynth;  
testInstance instanceof AudioSynth; // true
```

```
testInstance === Synth; // true
```

To use AudioSynth to generate .WAV files...

```
Synth.generate(sound, note, octave, duration);  
/*
```

Will generate a base64-encoded dataURI wavefile (.WAV) containing your data.

sound

a numeric index or string referring to a sound profile (by id or name, respectively)

note

the note you wish to play (A,B,C,D,E,F,G). Supports sharps (i.e. C#) but not flats. (Use the respective sharp!)

octave

the octave # of the note you wish to play

duration

```

    the duration (in seconds) of the note
*/

```

You can play notes instantly using...

```

/*
    Same arguments as Synth.generate,
    only this creates an HTML Audio element, plays it, and unloads it upon completion.
*/
Synth.play(sound, note, octave, duration);

```

You may also create individual instruments (objects that reference `.generate` and `.play`, bound to specific sounds).

```

var piano = Synth.createInstrument('piano');
piano.play('C', 4, 2); // plays C4 for 2s using the 'piano' sound profile

```

Sound Profiles

AudioSynth comes with four default sound profiles.

piano (id 0)

organ (id 1)

acoustic (id 2)

edm (id 3)

```

var acoustic = Synth.createInstrument('acoustic'); // play with your acoustic guitar!

```

Changing Settings

Poor performance? The default sampling rate for AudioSynth is 44100Hz (CD quality). This can be taxing on your browser.

To change the sampling rate, use `Synth.setSampleRate(n)` Please note that lower sampling rates will equate to poorer sound quality, especially for higher notes.

```

// Can only set values between 4000Hz and 44100Hz.
Synth.setSampleRate(20000); // sets sample rate to 20000Hz

```

```

Synth.getSampleRate(); // returns 20000

```

Volume a bit much? Adjust the volume of the sample similarly.

```

Synth.setVolume(1.00); // set volume to 100%
Synth.setVolume(0.40); // no, wait, 40%.
Synth.setVolume(0.1337); // even better.

```

```

Synth.getVolume(); // returns 0.1337

```

Advanced Usage

Additional sound profiles can be loaded using `Synth.loadSoundProfile()`

```

// Load a sound profile from an object...
Synth.loadSoundProfile({
  // name it
  name: 'my_sound',
  // WIP: return the length of time, in seconds, the attack lasts

```

```
attack: function(sampleRate, frequency, volume) { ... },
// WIP: return a number representing the rate of signal decay.
// larger = faster decay
dampen: function(sampleRate, frequency, volume) { ... },
// wave function: calculate the amplitude of your sine wave based on i (index)
wave: function(i, sampleRate, frequency, volume) {
  /*
  Here we have access to...
  this.module : an array of loaded frequency
  this.vars : any temporary variables you wish to keep track of
  */
}

});
```

A rough guide to waveform generation can be found at <http://keithwhor.com/music/>

Debugging

If you're hanging on note generation (for default or custom sound profiles), use `Synth.debug()` to enable debugging.

This will log note generation times in your console.

Credits and Acknowledgements

Special thanks to Albert Pham (<http://www.sk89q.com/>) for Dynamic .WAV file generation, the work off of which this is based (<http://www.sk89q.com/playground/jswav/>) and Hasen el Judy (<http://dev.hasenj.org/post/4517734448>) for information regarding Karplus-Strong String Synthesis.

Further Reading

.WAV Audio Files

http://en.wikipedia.org/wiki/.WAV_file

Sound Synthesis

http://www.acoustics.salford.ac.uk/acoustics_info/sound_synthesis/

"acoustic" sound profile generated using **Karplus-Strong String Synthesis**:

http://en.wikipedia.org/wiki/Karplus%E2%80%93Strong_string_synthesis
http://music.columbia.edu/cmc/musicandcomputers/chapter4/04_09.php

Contact

Feel free to e-mail me at keithwhor at gmail dot com

or follow me on Twitter, [@keithwhor](#)

If you like, feel free to share! :) Always appreciated.

audiosynth maintained by [keithwhor](#)

Published with [GitHub Pages](#)