# Objects Covered Cheatsheet

## Part 2

| Object [] | What it does | Example Arguments (where applicable) |
|---|---|---|
| **[midiin]** | Listens to a MIDI port and outputs the raw data. Double click to select the port. Keep in mind that raw MIDI data is **not human readable**. | |
| **[midiparse]** | Parses raw MIDI data into Note/Velocity pairs, Poly Pressure, CC's, Program Changes, Aftertouch, Pitch Bend, MIDI Channel, and midievent (for [vst~] only). | |
| **[notein]** | Listens to a MIDI port and outputs pitch, velocity, and channel. Double click to select the port. | |
| **[ctlin]** | Listens to a MIDI port and outputs CC information. Double click to select the port. You can specify the CC number as an argument. | [ctlin 64] |
| **[mtof]** | Converts MIDI to frequency in Hz. Always operates as a float. Can receive decimals - e.g. (60.5) to add cents (the decimals) to a given MIDI note. You can set the tunning (default 440) with @base | [mtof @base 435] |
| **[pack]** | Creates a list and outputs it when it receives a value in the left inlet. Arguments are the type of values accepted and the respective inlets, which are dynamically allocated. | [pack f i f] |
| **[unpack]** | Takes a list and breaks it into individual messages output from the left to right. Arguments are the type of values accepted and the respective outlets, which are dynamically allocated. | [unpack i 0. 0.] |
| **[pak]** | Creates a list and outputs it whenever it receives a value in any inlet. Arguments are the type of values accepted and the respective inlets, which are dynamically allocated. | [pak i 0. sym] |
| **[join]** | Joins multiple untyped items into a single list. Set the number of items with the argument and use @triggers to set which inlets are hot. | [join 3 @triggers 0 1] |
| **[unjoin]** | Breaks a list into messages of any type sending each out of an individual outlet. Argument is the number of individual outlets to create (dynamically allocated). It also creates an additional, right-most outlet to dump any remaining elements to. | [unjoin 3] - note that this will have 4 outlets, with the last being the dump |
| **[makenote]** | Generates a note-on pair and a note-off pair after a specified amount of time. Arguments are default velocity and duration of the note. | [makenote 64 500] |
| **[adsr~]** | Generates an ADSR envelope based on the arguments or inlets when it receives a 1 or note-on. Releases on a 0 or note-off. Arguments are A, D, S, and R respectively. All are floats, with Sustain having a range or 0-1. It also outputs mute messages for [thispoly~] to control voice allocation. | [adsr~ 10 80 0.75 250] |
| **[!-]** | Subtraction, but inlets are reversed! Subtracts the left from the right or argument. | Specify for floats - [!-6.] |
| **[cycle~]** | Antialiased sinusoidal oscillator. Generates a cosine wave by default (can be changed with [buffer~]). Argument is the default frequency. | [cycle~ 220] |

| | | |
|---|---|---|
| **[rect~]** | Antialiased rectangular oscillator. Generates a PWM wave. Arguments are frequency and pulse width (floats 0-1, 0.5 is a square wave). | [rect~ 440. 0.5] |
| **[saw~]** | Antialiased sawtooth oscillator. Generates a sawtooth wave. Argument is the initial frequency. | [saw~ 660] |
| **[tri~]** | Antialiased triangle oscillator. Generates a triangle wave. Arguments are the initial frequency and the duty cycle which alters the slope of the wave (floats 0-1, 0.5 is a triangle). | [tri~ 20 0.5] |
| **[phasor~]** | Generates sawtooth waves for sample rate control (e.g. LFO's or other modulation, or to drive other objects). **DO NOT USE FOR AUDIO.** Argument is the initial frequency. | [phasor~ 23] |
| **[triangle~]** | Triangle ramp wavetable. Must be driven by a phasor that controls the rate. Outputs a triangle wave for sample rate control from -1 to 1 (e.g. LFO's or other modulation, or to drive other objects). **DO NOT USE FOR AUDIO.** Argument is the duty cycle (floats 0-1, 0.5 is a triangle). | [triangle~ 0.6] |
| **[sig~]** | Converts a number into a steady DC signal. Use to drive other objects. **DO NOT USE FOR AUDIO.** | |
| **[cos~]** | Generates a signal rate cosine that outputs from 0-1 for sample rate control (e.g. LFO's or other modulation, or to drive other objects). **DO NOT USE FOR AUDIO.** | |
| **[noise~]** | Generates white noise. | |
| **[pink~]** | Generates pink noise. | |
| **[scope~]** | Visualizes an audio signal. LOTS of customizable options in the Inspector. Check the help file for details. | |
| **[spectroscope~]** | Displays a spectrogram or sonogram of an audio signal. LOTS of customizable options in the Inspector. Check the help file for details. | |
| **[attrui]** | Inspects and sets attribute of a connected object as a menu. | |
| **[ezdac~]** | Audio output (channels 1 and 2) and on/off button. | |
| **[dac~]** | Audio output. The arguments are the specific channel (or channels) to use. | [dac~ 5 7] |