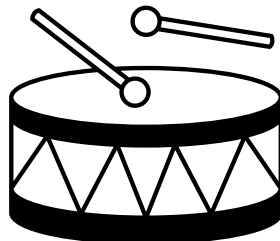_____

WILL BE

CODING LIKE A DJ

1.
SAMPLES

```
sample :drum_roll
sleep 6
sample :drum_splash_hard
```
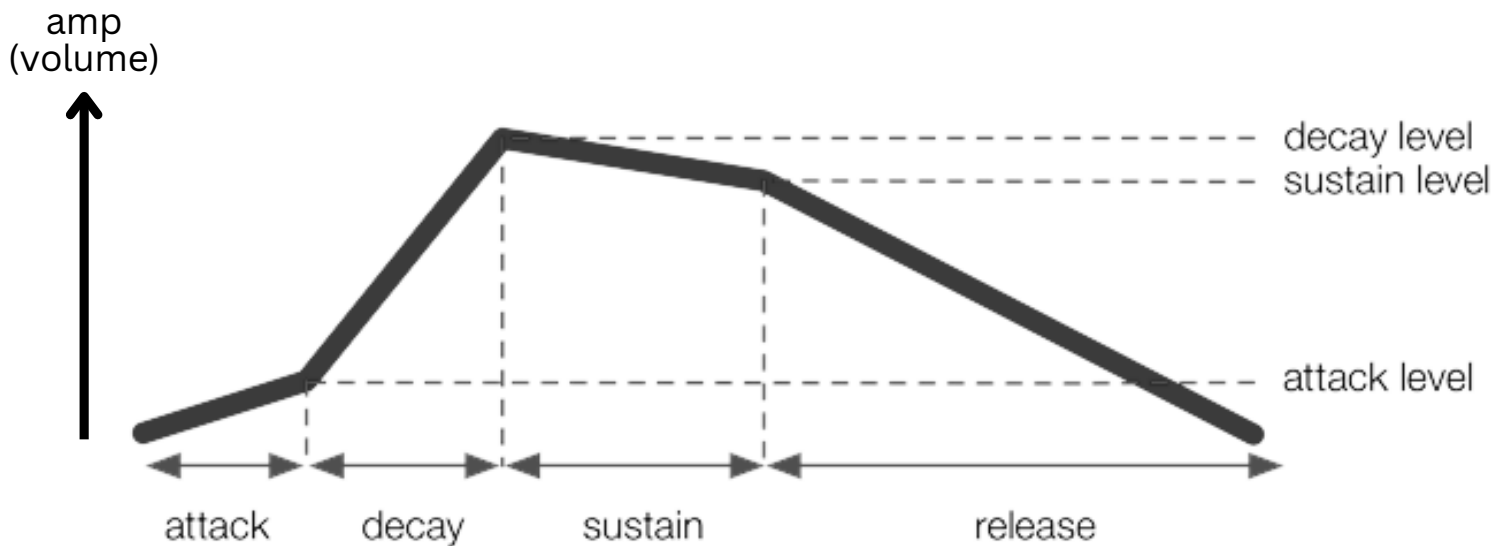
Add

```
sample :drum_roll, attack: 1,
sustain: 2
sleep 3
sample :drum_splash_hard,
amp: 1.5
```
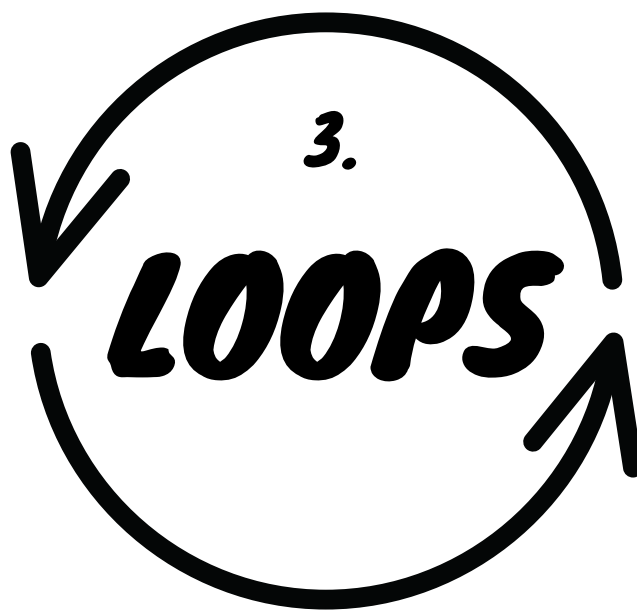
volume

2.

SYNTHS

amp
(volume)

attack   decay   sustain   release

decay level
sustain level

attack level

```
use_synth :fm
play :c
```

```
use_synth :fm
play :c, attack: 1, decay: 0,
sustain: 0, release: 1, amp:
0.5, pan: rrand(-0.5,0.5)
```
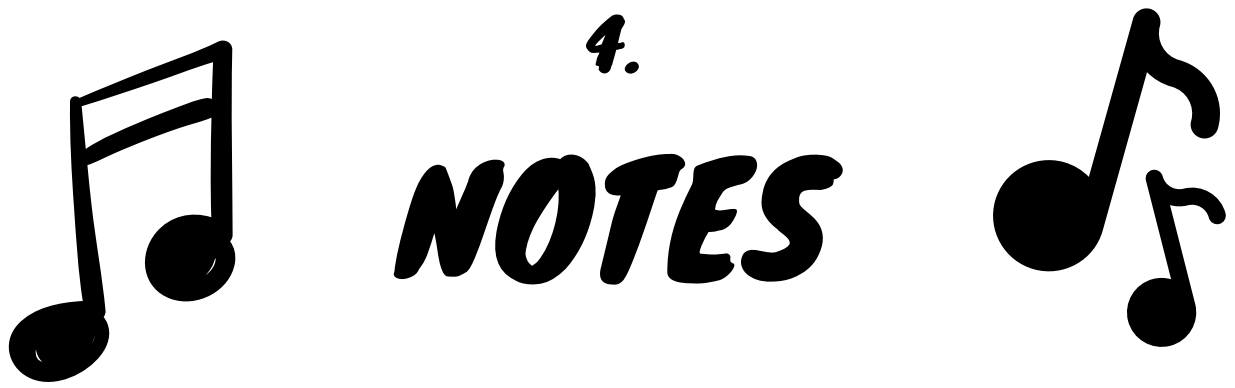
Add

3. LOOPS

```
live_loop :drums do
  4.times do
    sample :drum_bass_hard
    sleep 0.5
    sample :drum_cymbal_closed
    sleep 0.5
    sample :drum_snare_hard
    sleep 0.5
    sample :drum_cymbal_closed
    sleep 0.5
  end

  sample :drum_bass_hard
  sleep 0.75
  sample :drum_bass_hard
  sleep 0.25
end
```

4. NOTES

```
live_loop :bass do
  use_synth :tb303
  play :c2
  sleep 0.5
end

live_loop :riff do
  use_synth :pulse
  play :c, attack: 0.5, sustain:
0.5, release: 0.5
  sleep 1
  play :g, pan: rrand(-0.5, 0.5)
  sleep 0.5
  play :e
  sleep 0.5
end
```

RANDOMNESS

```
live_loop :my_random_beat do
  use_random_seed 220
  16.times do
    sample :elec_hi_snare if one_in 3
    sample :drum_cymbal_closed if one_in 3
    sample :drum_cymbal_pedal if one_in 5
    sample :bd_haus if one_in 3
    sleep 0.125
  end
end

live_loop :my_tune do
  use_synth :prophet
  notes = scale :c4, :minor_pentatonic
  use_random_seed 40
  6.times do
    play notes.choose
    sleep [0.25, 0.5].choose
  end
end
```
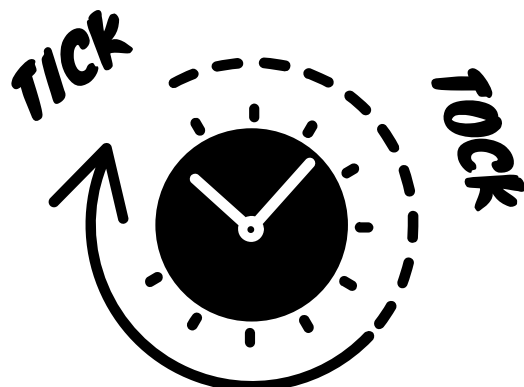
6.

TICK

```
use_synth :piano
play [:c, :e, :g]
sleep 1
play (chord :c, :major)
sleep 1
live_loop :play do
  play (chord :c, :major).tick
  sleep 0.5
  stop if look > 2
end
```
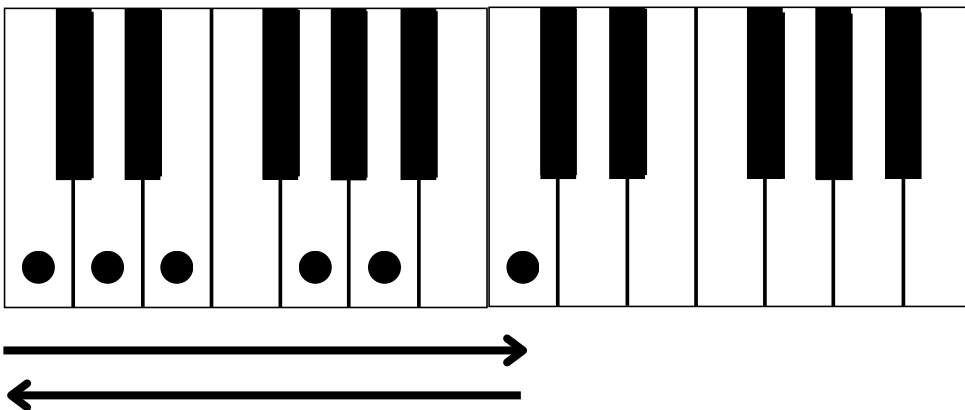
7.
SCALES

```
live_loop :up_and_down do
  use_synth :dsaw
  play (scale :c4,
:major_pentatonic).reflect.but
last.tick
  sleep 0.25
end
```
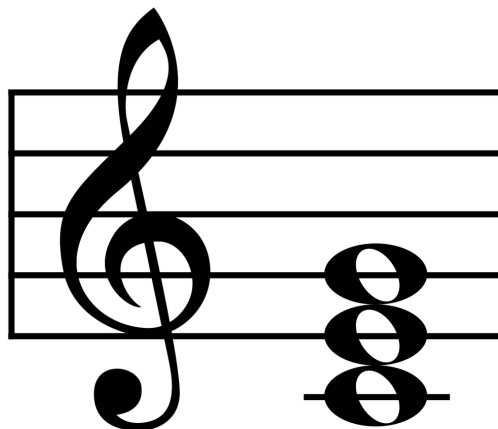
no spaces!

C major pentatonic scale

8.

CHORDS

```
use_synth :beep
live_loop :chords do
  [1, 3, 6, 4].each do |d|
    (range -3, 3).each do |i|
      play_chord (chord_degree
d, :c, :major, 3, invert: i)
      sleep 0.25
    end
  end
end
```
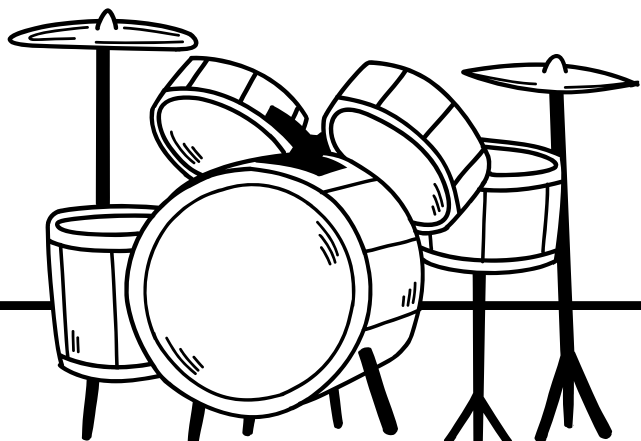
a group of notes

9.

DRUM PATTERNS

```
define :pattern do |p|
  return p.ring.look == 'x'
end


use_bpm 90  ←  90 beats
               per minute


live_loop :drums do
  tick
  sample :drum_bass_hard  if
pattern "x-x-----x-x--x--"
  sample :drum_snare_hard if
pattern "----x--x-x-xx---"
  sample :drum_cymbal_closed
  sleep 0.25
end
```
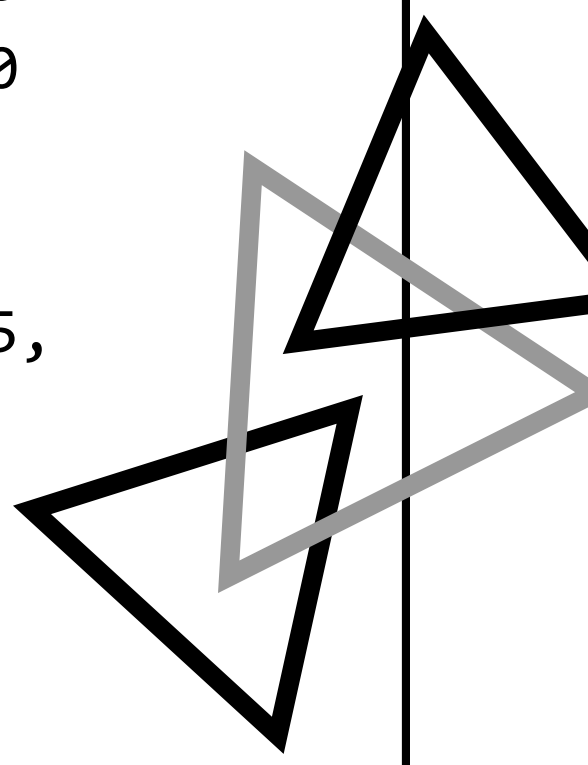
EFFECTS

```
with_fx :slicer do
  live_loop :breakbeat do
    sample :loop_breakbeat,
beat_stretch: 2, amp: 2
    sleep 2
  end
end
with_fx :wobble, phase: 0.25,
invert_wave: 1 do
  live_loop :melody do
    use_random_seed 20
    use_synth :fm
    6.times do
      play (chord :eb5,
:minor7).shuffle.tick
      sleep 0.5
    end
  end
end
```

CLASS NOTES
& MORE EXAMPLES

https://github.com/dariusf/sonicpi-workshop

including credits for all examples used

WE HOPE YOU HAD FUN!