



Ein Sonic Pi Workshop für Kinder

von Stefan Höhn, Irene Höppner und Matthias Malstädt

1a

Buffer 0

Verwende den Buffer 0 in Sonic Pi

1b

play 60
sleep 1

Spiele und warte

play 80	60	62	64	65	20
	67	69	71	72	
	:c4	:d4	:c5	:d5	

2a

play :c4
play :e4
play :g4
sleep 1

Spiele einen Akkord (Dreiklang)

c5 e5 g5 f4 a4 c5 g4 a4 d5

2b

BUFFER 1

Vewernde Buffer 1

3a

3b

play_chord [:c4, :e4, :g4]
sleep 1

Drei Töne gleichzeitig nennt man einen Akkord. So ist es einfacher als vorher. Hier ein C-Akkord.

c5 e5 g5 f4 a4 c5 g4 a4 d5

4a

play_chord chord(:e4, :major)
sleep 1

Major = Dur, Minor = Moll – erkennst Du den Unterschied?

:a4 :b4 :major7 :minor

Drei Akkorde mit einer Sekunde Abstand (nimm e, a und b als moll)

4b

BUFFER 2

Verwende Buffer 2

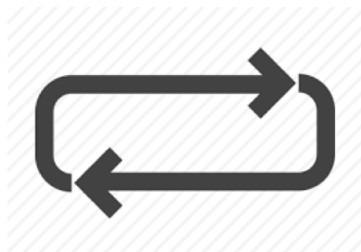
5a

Schleifen

5 mal

Für immer wieder

Zählen von 1 bis 10



5b

play_pattern (scale :c4, :major)

Spiele ein Muster (=Pattern) - hier eine Tonleiter (scale)

:major :major_pentatonic :minor_pentatonic :minor

6a

use_bpm 120

play_pattern (scale :e4, :minor)

Verwende ein andere Geschwindigkeit b p m = beats per minute = schläge pro minute

50 240 400 100 600
:major :major_pentatonic :minor_pentatonic :minor

6b

use_bpm 600

2mal

2.times do

play_pattern (scale :e4, :minor)

end

2 times = 2 mal. Wir nennen das eine Schleife

3.times

5.times

7a

live_loop :tonleiter do

Für
immer
wieder

use_bpm 120

play_pattern (scale :e4, :minor)

end

Wir nennen das Endlosschleife, die man während des Spielens aktualisieren kann.

Ändere auf 480. drücke RUN und höre, wann die Änderung kommt. Sofort?

7b

Füge den folgenden Befehl hinzu

`use_synth :saw`

Wir wäre es mit einem anderen Sound unseres Synthesizers?

`:dsaw`

`:mod_dsaw`

`:prophet`

`:piano`

`:blade`

`:tb303`

`:pluck`

`:dtri`

8a

`play_pattern (scale :e4, :minor)`

`play_pattern (scale :e4, :minor).reverse`

Und nun spielen wir die Tonleiter rückwärts

8b

BUFFER 3

Verwende Buffer 3

9a

```
live_loop :geblubber do
  use_bpm 240
  play_pattern (scale :e4, :minor).choose
  sleep 1
end
```

Choose heißt **wählen**. Dieser Befehl wählt **einen** zufälligen Ton aus der Tonleiter. Jedesmal einen anderen.

- *play* spielt **nur einen** Ton (*play_pattern* ein Muster, also viele)
- *choose* wählt einen beliebigen Ton

9b

BUFFER 8

Verwende Buffer 4

10a

Wenn / Dann

Zählen mit if / else

10b

meinton = 50
play meinton

Wir nennen meinton eine Variable

Was passiert hier und warum?

Was ist **meinton**? Was bedeutet das = ?

Was passiert wenn du in **meinton** = 60 schreibst?

11a

meinton = 50
play meinton
sleep 1
meinton = meinton + 20
play meinton

Welche Zahl hat meinton nach der vierten Zeile?

Welcher Ton wird in der vierten Zeile gespielt?

11b

meinton = 50

play meinton
sleep 1

meinton = meinton + 1
play meinton
sleep 1

meinton = meinton + 1
play meinton
sleep 1

meinton = meinton + 1
play meinton
sleep 1

Welche vier Töne werden hier gespielt?

12a

meinton = 50

live_loop :immerwieder do
 meinton = meinton + 1
 play meinton
 sleep 0.5
end



Immer
wieder

Was passiert nun?

Was ist nach einer Weile das Problem? (Tipp: bei ca. 125)

12b

```

meinton = 50
live_loop :immerwieder do
  meinton = meinton + 1
  play meinton
  sleep 0.5

  if (meinton==100) then
    meinton = 50
  end

end

```

Was passiert bei 100 und warum?

13a

```

meinton = 50
zaehleHoch = 1 # Zähle hoch = 1, Zö

live_loop :immerwieder do
  if (zaehleHoch == 1) then
    meinton = meinton + 1
  end
  if (zaehleHoch == 0) then
    meinton = meinton -1
  end

  play meinton
  sleep 1

  if (meinton==55) then
    zaehleHoch = 0
  end
  if (meinton==50) then
    zaehleHoch = 1
  end

end

```

Zum Knobeln für die Fortgeschrittenen

Was passiert hier?

13b

BUFFER 4

Verwende Buffer 4

14a

```
live_loop :schlagzeug do  
  sample :bd_haus  
  sleep 1  
  
end
```

Wir nennen das eine (Endlos-)Schleife

Füge ein weiteres sample *sn_zome* mit *sleep 1* hinzu

Mach' das Schlagzeug schneller (120)

```
:drum_bass_hard      :drum_snare_hard      :drum_tom_hi_hard
```

15a

BUFFER 5

Verwende Buffer 5

16a

16b


```
live_loop :melodie do
  sample :guit_em9
  sleep 2
end
```

Ein Gitarren-Sample

Probiere aus.

Danach kopiere das Schlagzeug (Buffer 4) und Melodie in Buffer 5 zusammen

17a

- Jetzt fügen wir in 6 alles zusammen
- Erst Buffer 5, dann Buffer 3 und Buffer 2
- Starte nach jedem weiteren Buffer neu
- Kopiere Buffer 1 und füge eine live_loop hinzu.

Benutze auch Size – und Size +, um den Text in den Größe zu verändern

```
use_bpm          use_synth :hollow          ,amp: 5
use_synth: hoover
```

17b

Weitere Ideen

- Mehr Schlagzeug
- Ein Musik-Programm
- Effekte
- Samples „loop_“ mit `sample_duration()`

Experimentiere

18a

18b

Spiele etwas mit dem Schlagzeug herum

use_bpm 120

live_loop :schlagzeug do

sample :drum_cymbal_closed, amp: 4, rate: 1

sleep 1

sample :drum_cymbal_closed, amp: 2

sample :drum_bass_soft, amp: 4

sleep 1

sample :drum_cymbal_closed, amp: 4

sleep 1

sample :drum_cymbal_closed, amp: 4

sample :drum_snare_hard, amp: 4, rate: 1.2

sleep 1

end

Experimentiere

1) Ändere die Lautstärken

2) Ändere die Rate: rate: 2 oder rate: -1

3) Ersetze das letzte Sample durch

4) Verändere die Geschwindigkeit

with_fx :reverb, room: 0.5 do

sample :drum_snare_hard, amp: 4, rate: 1.2

end

19a

Variablen und Bedingungen

Was geht hier vor?

If = falls / else = andernfalls

19b

Effekte

```
live_loop :mitHall do
  with_fx :reverb, room: 0.9 do
    play_pattern (scale :e4, :minor)
  end
end
```

So kann man Effekte verwenden

- Fx steht für „Effects“ = Effekte. Jeder Effekt kann auch „Parameter“ haben:
Hier die Größe des Raums für den Hall → room: 0.9
- Verwende eine andere Geschwindigkeit
- Verwende play und choose (siehe Seite 9b)
- Füge noch ein Effekt hinzu:
 - with_fx :krush do
- Probiere andere Effekte aus (siehe Fx im Spickzettel)

20a

Sample-Dauer

```
live_loop :endlos do
  sample :loop_amen
  sleep 4
end
```

Das Schlagzeug soll durchgängig spielen
aber die Pause ist zu lang. Versuche die
richtige Länge herauszufinden.

Probiere **sleep sample_duration(:loop_amen)** aus! Was passiert und warum?

Füge folgendes hinter dem Sample-Befehl ein: **, rate: 2** Was macht die Rate? Was passiert mit der Pause?

Wie müssen wir die Pause korrigieren?

Zum Spaß: Versuche mal eine **rate:-1**

20b

BUFFER 9

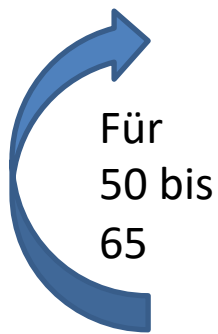
Verwende Buffer 4

21a

Schleifen

Alien-Geräusche für Spiele

21b



```
for meinton in 50..65
  play meinton
  sleep 1
end
```

Wir nennen das eine **Zähl-Schleife**

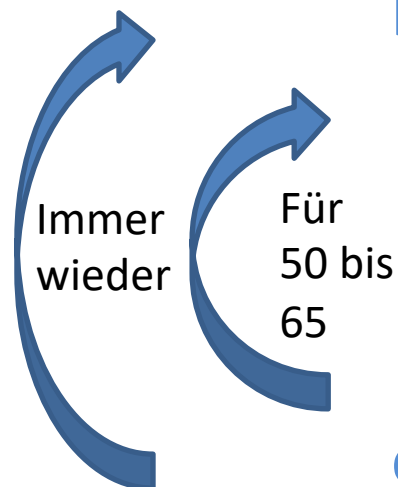
Was passiert hier und warum?

Was ist **meinton**? Was passiert mit **meinton** in der **Schleife**

Was macht **play** mit **meinton**?

Was passiert, wenn du **meinton** in deinen Namen änderst?

22a



```
live_loop :immerwieder do
  for meinton in 50..65
    play meinton
    sleep 1
  end
end
```

Eine Zählschleife in der Endlosschleife

Was passiert nun?

22b

use_bpm 1200

2400? 4800? 9600?

23a

for meinton in 55..70
 play meinton
 sleep 1
end

Füge diesen Teil noch hinzu. Vergleiche! Wie wird das klingen?
Höre es dir erst LANGSAM an (use_bpm 480)

23b

for meinton in 60..75

...

for meinton in 65..80

...

for meinton in 70..85

...

75.. , 80..

Lass es schneller laufen... Wie hört sich das an?

24a

Spickzettel

play 60

sleep 1

use_bpm 600

play :c4

play_chord [:c4, :e4, :g4]

```
play_chord chord(:e4, :major)
:major :major_pentatonic :minor_pentatonic :minor
```

```
play_pattern (scale :e4, :minor)
play_pattern (scale :e4, :minor).reverse
```

```
play (scale :e4, :minor).choose
```

```
use_synth :hollow → :saw, :hoover, :piano ...
```

```
live_loop :meineEndlosschleife do
  ...
end
```

```
2.times do
  ...
end
```

```
sample :bd_haus → :guit_em9 ...
```

Spickzettel

Befehle

ALT-R	Starten	ALT-A	Alles markieren
ALT-S	Stoppen	ALT-C	Kopieren
STRG-I	Hilfe für Befehl	ALT-V	Einfügen

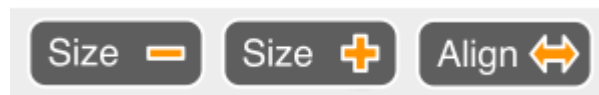
Knöpfe



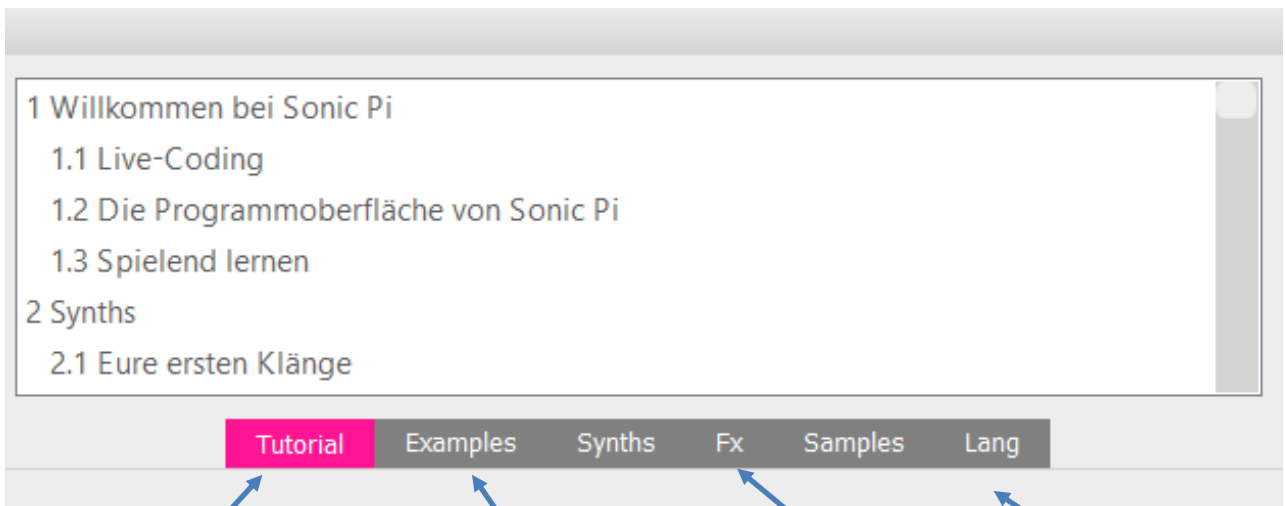
Starten Stoppen Aufnehmen



Speichern Laden



Text kleiner Text größer Text schön machen („ausrichten“)



Anleitung für zuhause

Cooler Beispiele

Effekte

Alle Befehle

Spickzettel

use_synth

[Tutorial](#)
[Examples](#)
[Synths](#)
[Fx](#)
[Samples](#)
[Lang](#)

:beep	:blade	:bnoise	:cnoise	:dark_ambience		
:dpulse	:dsaw	:dull_bell	:fm	:gnoise	:growl	
:hollow	:hoover					
:mod_beep	:mod_dsaw	:mod_fm	:chiplead	:chipbass	:chipnoise	
:mod_pulse	:mod_saw		:mod_sine	:mod_tri	:pule	
:noise	:piano	:pnoise	:pretty_bell	:prophet	:dtri	:pluck
:saw	:sine	:square	:subpulse	:tb303	:tri	:zawa

sample

[Tutorial](#)
[Examples](#)
[Synths](#)
[Fx](#)
[Samples](#)
[Lang](#)

:elec_triangle
 :elec_snare
 :elec_lo_snare
 :elec_hi_snare
 :elec_mid_snare
 :elec_cymbal
 :elec_soft_kick
 :elec_filt_snare
 :elec_fuzz_tom
 :elec_chime
 :elec_bong
 :elec_twang
 :elec_wood
 :elec_pop
 :elec_beep
 :elec_blip
 :elec_blip2
 :elec_ping
 :elec_bell
 :elec_flip
 :elec_tick
 :elec_hollow_kick
 :elec_twip
 :elec_plip
 :elec_blup

:misc_burp
 :perc_bell
 :perc_bell2
 :perc_snap
 :perc_snap2
 :perc_door
 :perc_impact1
 :perc_impact2

:bd_ada
 :bd_pure
 :bd_808
 :bd_zum
 :bd_gas
 :bd_sone
 :bd_haus
 :bd_zome
 :bd_boom
 :bd_klub
 :bd_fat
 :bd_tek
 :bd_mehackit

:bass_hit_c
 :bass_hard_c
 :bass_thick_c
 :bass_drop_c
 :bass_woodsyc
 :bass_voxy_c
 :bass_voxy_hit_c
 :bass_dnb_f

:ambi_soft_buzz
 :ambi_swoosh
 :ambi_drone
 :ambi_glass_hum
 :ambi_glass_rub
 :ambi_haunted_hum
 :ambi_piano
 :ambi_lunar_land
 :ambi_dark_woosh
 :ambi_choir

:ambi_soft_buzz
 :ambi_swoosh
 :ambi_drone
 :ambi_glass_hum
 :ambi_glass_rub
 :ambi_haunted_hum
 :ambi_piano
 :ambi_lunar_land
 :ambi_dark_woosh
 :ambi_choir
 :ambi_sauna

:drum_heavy_kick
 :drum_tom_mid_soft
 :drum_tom_mid_hard
 :drum_tom_lo_soft
 :drum_tom_lo_hard
 :drum_tom_hi_soft
 :drum_tom_hi_hard
 :drum_splash_soft
 :drum_splash_hard
 :drum_snare_soft
 :drum_snare_hard
 :drum_cymbal_soft
 :drum_cymbal_hard
 :drum_cymbal_open
 :drum_cymbal_closed
 :drum_cymbal_pedal
 :drum_bass_soft
 :drum_bass_hard
 :sn_dub
 :sn_dolf
 :sn_zome
 :sn_generic

:glitch_bass_g
 :glitch_perc1
 :glitch_perc2
 :glitch_perc3
 :glitch_perc4
 :glitch_perc5
 :glitch_robot1
 :glitch_robot2

:loop_industrial
 :loop_compus
 :loop_amen
 :loop_amen_full
 :loop_garzul
 :loop_mika
 :loop_breakbeat
 :loop_3d_printer
 :loop_drone_g_97
 :loop_electric
 :loop_mehackit1
 :loop_mehackit2

:drum_cowbell
 :drum_roll
 :misc_cros
 :misc_cineboom
 :perc_swash
 :perc_till
 :loop_safari
 :loop_tabla
 :loop_perc1
 :loop_perc2
 :loop_weirdo

:guit_harmonics
 :guit_e_fifths
 :guit_e_slide
 :guit_em9