Sonic Pi Walk-up Workshop

Time: Activity 1: 5 minutes
Activity 2: 7 minutes
Activity 3: 2 minutes (optional)
Activity 4: 4 minutes (optional)

All participants should be able to run the 'Get Lucky' file in Activity 1 and make some alterations to choice of synth, tempo, choice of samples.

Some participants may then want to carry on to Activity 2 and work through the example of how to code up your own "band".

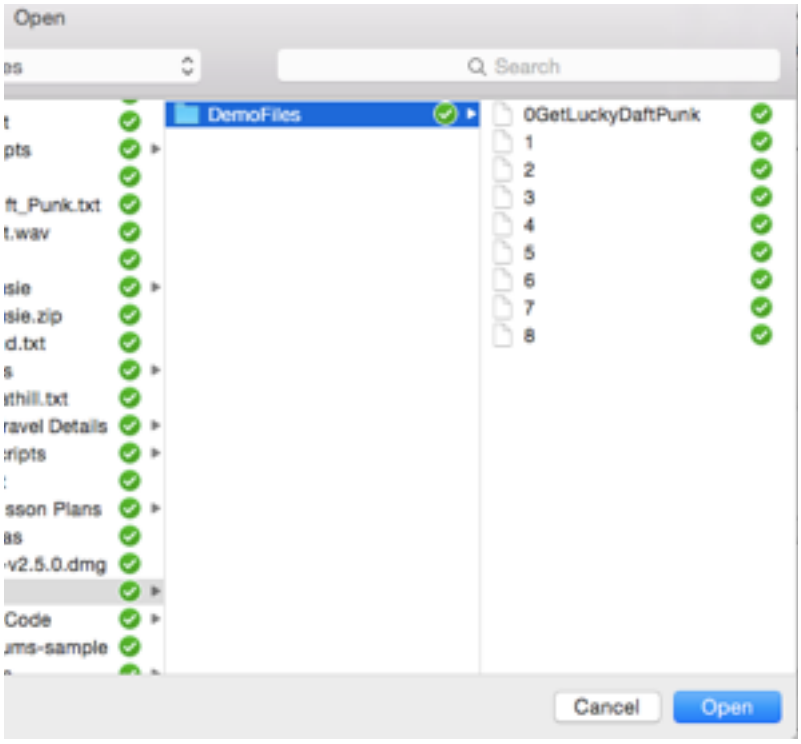
A few participants may want to share their work from Activity 1 or 2 via the PA system at the workstation.

A few participants may want to find out more about live-coding performances (from Sam Aaron - creator of Sonic Pi who'll be with us on both days).

Setup/Reset Between Participants

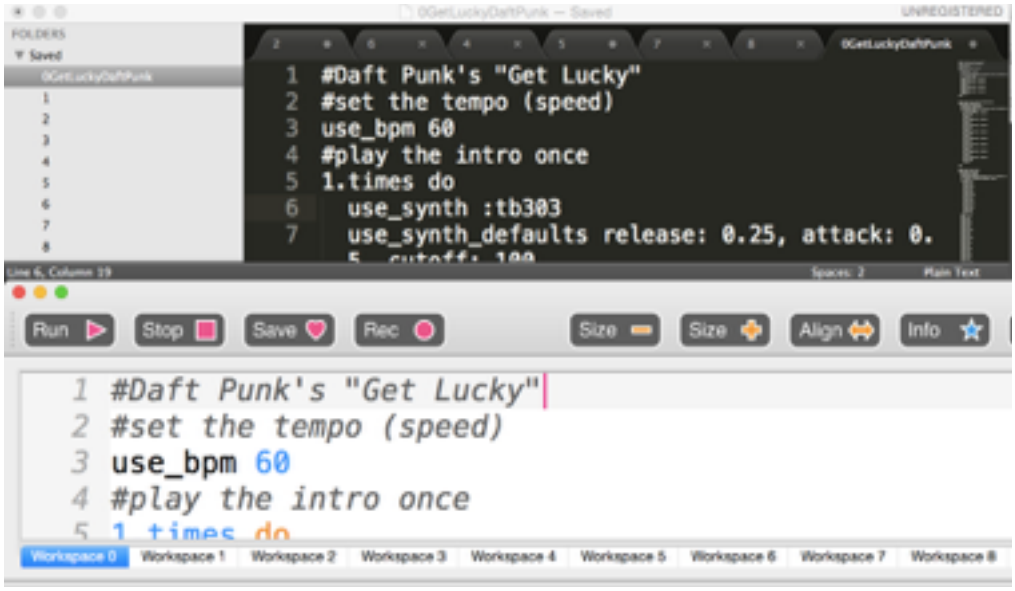
In the Sublime text editor go to the **File** menu and select **Open**. Select the DemoFiles folder inside the SonicPiResources folder on the Desktop and click the **Open** button.

This will open all the files in that folder.



Click into the first file (**0GetLuckyDaftPunk**), select and copy the text inside (pressing **cmd+a** will select all, **cmd+c** will copy).

Then in the Sonic Pi window click on **Workspace 0** and press **cmd +v** to copy the text into that workspace.



Repeat this for the other files in the DemoFiles folder.

Pasting the text in file 1 into Workspace 1, file 2 into Workspace 2, etc.

Activity 1 - Remix 'Get Lucky' by Daft Punk

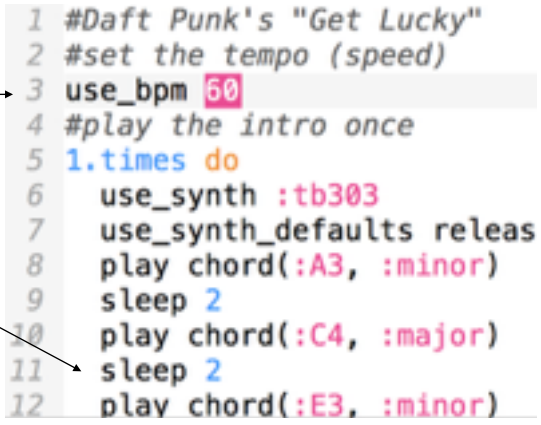
Sonic Pi is a program that lets you make music by writing code, rather than pressing keys on a keyboard, hitting drums, or plucking guitar strings. Let's see what you can do with it by listening to a piece of music we've already written.

Workspace 0

Click into Workspace 0 in Sonic Pi and hit **Run** to play "Get Lucky" by Daft Punk.
How do you think we could change how it sounds?

Change the tempo (speed)

- use a different value for `use_bpm` (highlighted in pink in this screenshot).
  Higher number = faster, lower number = slower
- change the values for the `sleep` commands. (`sleep` tells the computer how long to wait before playing the next sound).



Edit the code to see what effect different values have.

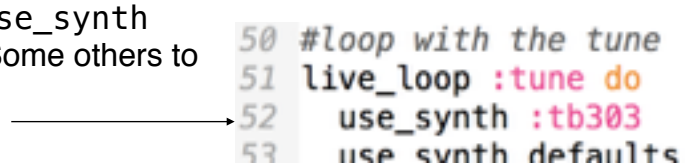
Change the sounds.

There are **two types** of sound used in Sonic Pi: **synths** and **samples.**

Change the synth

Synths (short for synthesisers) are what the computer creates a note electronically as you play it. If you've ever played an electronic keyboard that let you make sounds like other instruments (e.g. piano, violin, flute) you've used a synth. Bands/musicians who use a lot of synths include: Lady Gaga, Daft Punk, Calvin Harris, Keisha, Nicki Minaj. **Can you think of others?**

- try using a different synth with the `use_synth` command on lines 6, 21, and 52 . Some others to try are



- `:prophet` `:beep` `:supersaw` `:pretty_bell` `:mod_sine`

What do you think of the new sounds you've made? What's your favourite synth of the ones you've tried?

Change the sample

Samples are small sound clips that have been made and recorded earlier.
The `:loop_amen` sample is a couple of seconds of someone playing the drums.
The `:ambi_lunar_landing` sounds like a spaceship landing on the moon.

Samples can be part of other songs or sounds from the real world, songs that use songs from other songs include: One Direction's "Steal My Girl" (samples "Faithfully" by Journey), Lady Gaga's "Born this Way" (samples "Express Yourself" by Madonna). Can you think of others?

- **Change the sample** (recorded sound clip)
  - try using a different sample with the sample command on line 107 in the :drums section of the code

```
107 live_loop :drums do
108   3.times do
109     sample :drum_heavy_kick
110     sleep 0.5
111     sample :drum_snare_hard
112     sleep 0.5
113     sample :drum_heavy_kick
114     sleep 0.5
115     sample :drum_snare_hard
```

Some other samples to try are:
`:ambi_choir` `:bd_boom` `:bass_drop_c`
`:drum_cymbal_open` `:guit_e_fifths` `:loop_amen`

Activity 2 - Code your own band

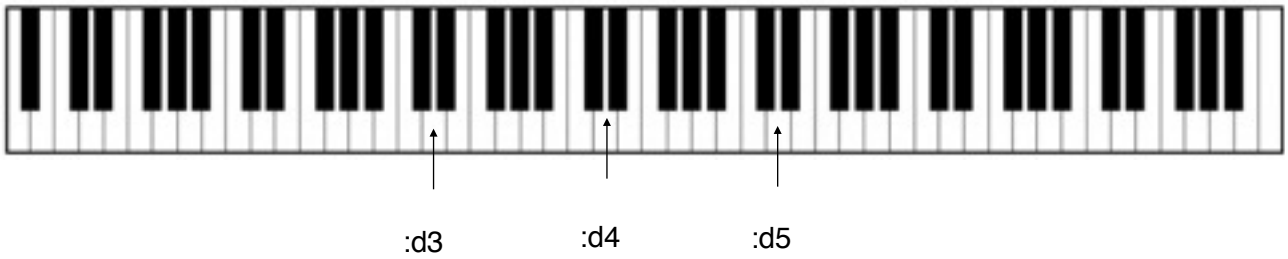
How do you build up your own tune? How can you code your own band? Let's take it a step at a time.

Workspace 1

Start by choosing a synth and playing a couple of notes using it:

```
use_synth :beep
play :d4
sleep 2
play :a4
sleep 2
```

The command `play :d4` plays the note D. The 4 means that it's the 4th D you'd find on a piano. Try changing the note name (a,b,c,d,e,f,g) or how far along the keyboard it is (1..7).



The command `sleep 2` means the computer waits 2 beats before doing the next command. What happens if you make this number smaller (between 0.05 and 1.5)? Or larger (between 3 and 10)?

Workspace 2

Most bands have a drummer. So let's add a drum sound using a sample.

```
sample :bd_haus, rate: 1
sleep 0.5
```

You can change the sample if you like. Or add another one that plays after the first.

The rate: setting controls the speed the sample is played at. What happens if you change the number after `rate:` ?

- a higher number makes it play faster (and with a higher pitch)
- a lower number (between 0 and 1) makes it play slower (and with a lower pitch).

This is like when you fast forward through a CD and the singer sounds like a chipmunk.

Try setting the rate to -1. What is happening? (The sample is played backwards).

Workspace 3

Most bands also have a guitarist. So let's add a guitar sample.

```
sample :guit_e_fifths, rate: 1
sleep 1
```

Try adjusting the rate like you did with the drum sample.

Workspace 4

Now we've got all the parts of our band: a tune, a drummer, and a guitarist. But it's a really short song! We can use something called a loop to make our band keep on playing. A loop is a command that does something over and over again. It saves a lot of typing!

```
live_loop :tune do
    use_synth :beep
    play :d4
    sleep 2
    play :a4
    sleep 2
end
```

`live_loop` is the name of the command - it's a loop that lets you edit it while its playing. Just hit Run to update the sound as it plays.

`:tune` is the name of this loop. It could be any name - like Fred or Bob - but it's a good idea to give it a name that tells you a bit about what it's doing.

Everything between the words `do` and `end` will be repeated over and over until you click **Stop**.

Workspace 5

Lets add a `live_loop` to our drum 5 sound and call it `:drums`

```
live_loop :drums do
  sample :bd_haus, rate: 1
  sleep 0.5
end
```

Workspace 6

Lets add a `live_loop` to our guitar sound and call it `:guitar`

```
live_loop :guitar do
  sample :guit_e_fifths, rate: 1
  sleep 1
end
```

Workspace 7

We've now got all three members of our imaginary band playing. But they're each in a room by themselves! Lets start bringing them together by putting our `:tune` loop in the same workspace as our `:drums` loop. What happens when you press play now? (Both loops play at the same time).

Workspace 8

Finally, let's get our guitarist to join in by adding the `:guitar` loop to the other two. What happens when you press play now? (All 3 loops play at the same time).

Activity 3 - Sharing your work

Are you really pleased with the way your remix of "Get Lucky" or the piece you built up with loops turned out? Do you want to share it with other people at the event? We can plug it into the screens and loudspeakers and play it to the public.

(Plug the Thunderbolt connector from the screen into the port on the right hand side of the Mac to transfer the picture and sound to the screens and hit Run).

Activity 4 - Live Coding

Live-coding involves coding 'live' as the music plays- as you would with any other musical instrument in a performance. If you'd like to learn more about live-coding Dr Sam Aaron, who created Sonic Pi, will be doing some demo sessions that you can watch. He'll also be available to chat to about how you can get started with this.

Troubleshooting

- Open the **Help** panel/tutorial by pressing **Cmd + i** (there's a bug in this version of Sonic Pi for Mac that crashes the program if you click on Help).
- If the program keeps crashing or gives you an error like "Can't restart Sonic Pi: Port 4885 already in use", **restart the machine** by clicking on the apple at the top left of the screen and selecting **Restart** from the drop-down menu.

Complete example of a Sonic Pi “Band”

```
live_loop :tune do
  use_synth :sine
  play :d4
  sleep 2
  play :a4
  sleep 2
end

live_loop :drums do
  sample :bd_haus, rate: 1
  sleep 0.5
end

live_loop :guitar do
  sample :guit_e_fifths, rate: 1
  sleep 1
end
```