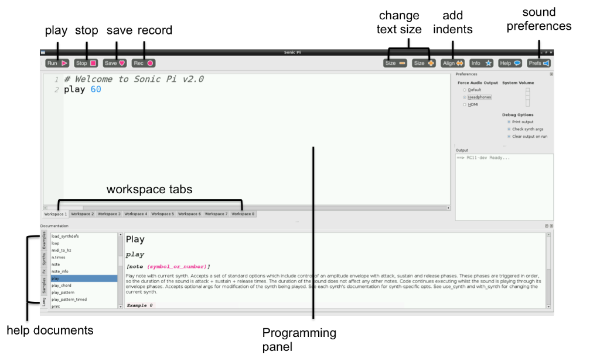
**GETTING STARTED MAKING MUSIC WITH SONIC PI V2.0**

Ever wanted to make funky music but you're not sure how to hold a cello, let alone play it? Sonic Pi v2.0 could get you there.

**STEP 1: FIRST SOUNDS WITH SONIC PI V2.0**



This is the Sonic Pi v2.0 application interface; it has three main windows. The largest one is for writing your code, and we call it the Programming Panel. There is also an output panel that displays information about your program as it runs. When you click on the **help** button at the top of the window, the third panel appears along the bottom displaying help documentation. This contains information about different code you can try and use, as well as different synth sounds, samples, and much more.

1. Launch Sonic Pi by clicking on the Main Menu, then Education, and then Sonic Pi v2.0.
2. Select **Workspace 1** and type:

play 60

1. Click on the **play** icon at the top of the screen. What happens?
2. What happens if you type play 60 and click on the play icon?

This is an example of a bug in your code. In later activities, if the error panel displays text you will know that you have a bug that you need to fix. It could be that you have misspelt a word like play.

1. Now type:

play 60

play 67

play 69

1. Click on the play icon at the top of the screen. What happens?
2. The computer is playing each note in sequence (one after the other), but it is happening so fast that to us they sound like they are playing at the same time.

We need to tell the computer to pause between each note. We can do this by typing the following after each play:

sleep1

The value entered after the word sleep represents time in seconds. Using the value 1 represents one second. What would you type for half a second?

1. Now write a sequence of play and sleep to make a cool-sounding tune!

**STEP 2: LOOP A TUNE**

Now you have mastered the basics of Sonic Pi, let's code a tune!

1. Select Workspace 2.
2. Type the following code:

play 60

sleep 0.5

play 62

sleep 0.5

play 64

sleep 0.5

play 60

sleep 0.5

1. Now click on the play icon at the top of the screen and it will play the first part of a tune. Can you tell what it is?

This first section plays twice. How could you repeat it? You could type the same section out again, or we could start to introduce loops to your code.

1. At the top of your code, above the first play 60, type:

2.times do

1. And at the bottom of your code, below sleep 0.5, type:

end

1. Click on the play icon at the top of the screen. What happens?

Let's play this part in Sonic Pi.

In the example below, you can see that some lines of code are indented. This makes it easier to read your code, and check for any bugs if it does not work when you press the play button. You can press the space bar twice to indent a line of code.

2.times do

play 60

sleep 0.5

play 62

sleep 0.5

play 64

sleep 0.5

play 60

sleep 0.5

end

Looping notes for a set number of times is certainly useful, but what if you want to loop your tune forever?

Instead of using 2.times do and end you can use loop do and end, like this:

loop do

play 60

sleep 0.5

end

**STEP 3: MIDI NOTES AND MUSIC NOTES**

The values that you have been typing after the word play represent notes; in fact, they are MIDI note numbers. This means we can translate songs played on a piano into Sonic Pi using a table like so:

C D E C or 60 62 64 60 in MIDI notes.

**Music Notes to MIDI Note Values**

| **C** | **D** | **E** | **F** | **G** | **A** | **B** |
| --- | --- | --- | --- | --- | --- | --- |
| 60 | 62 | 64 | 65 | 67 | 69 | 71 |

This is quite a long process if you know the notes of the song you are trying to play. With Sonic Pi 2 you are able to use standard sheet music notation too.

1. In a new workspace tab type:

play :c4

sleep 0.5

play :d4

sleep 0.5

play :e4

sleep 0.5

play :c4

sleep 0.5

1. Press **play** to hear your tune. Does it sound the same as when you used MIDI notes?

**STEP 4: CHANGE THE SOUNDS**

It's time to make your tune sound more interesting! We can do this by changing the synthesizer sounds it is using. The default Sonic Pi synth is called beep.

To use a different synth, you need to add the code use\_synth :name of synth above the sequence of code you want to use it in.

In this example, fm is the name of the synth:

use\_synth :fm

2.times do

play 60

sleep 0.5

play 67

sleep 0.5

end

**SYNTHS TO TRY**

There are lots of cool-sounding synths included with Sonic Pi 2. To find the names of them, click on the **help** icon at the top of the screen so that the help documents window appears. Then select**Synths** from the tabs along the left hand side of the help window. Click on any of the synth names to get more information on how to use it.

**STEP 5: USE SAMPLES**

Not only can you create music in Sonic Pi 2 using single notes, you can also create music with samples. Samples are pre-recorded sounds or tunes that you can bring into your music. This is a really simple way to make your music sound amazing!

To use a sample, you need to add the code sample :name of sample in the sequence of your music program where you want it to play.

In this example, loop\_amen is the name of the sample:

2.times do

sample :loop\_amen

sleep 1.753

end

**SAMPLES TO TRY**

There are lots of samples included with Sonic Pi 2. To find the names of them, click on **help** followed by **samples** on the left hand side of the help window. Click on any of the sample names to get more information on how to use it.

**STEP 6: PLAYING TWO TUNES AT THE SAME TIME**

Music often has a repeating backing track, with a separate melody played over the top. So far in Sonic Pi you have played one tune. Let's try playing two tunes at the same time!

1. Click on a new workspace tab.
2. The code we use to play two tunes at the same time needs to be between in\_thread do andend.
3. Underneath in\_thread do, type your tune. Here I've used a sample for my backing track:

in\_thread do

loop do

sample :loop\_amen

sleep 1.753

end

end

This first 'thread' will act as the melody of your music. Underneath, you can type the code for your backing track or baseline.

1. Type:

in\_thread do

16.times do

play 75

sleep 1.753

play 74

sleep 0.25

end

end

1. Now press **play** and you should hear both threads playing at the same time.

**STEP 7: LIVE CODE!**

Sonic Pi v2.0 has been developed to be a platform for the live coding of music, so that the code can be manipulated, changed and adapted in real time; this means coders can perform their code rather than playing pre-written programs.

1. In a new workspace tab type:

define :play\_my\_synth do

use\_synth :prophet

play 50, attack: 0.2, release: 1.3

sleep 0.5

end

loopdo

play\_my\_synth

end

1. Press **play** to start the program.
2. Whilst the tune is playing, comment out the last three lines by adding a # symbol to the start of each line like this:

# loop do

# play\_my\_synth

# end

1. Next change some of code in the function, and press **play** again.

**TAKING SONIC PI 2 FURTHER**

Sonic Pi 2 is so awesome that there are too many features to mention in just one tutorial!

**STEP 1: ADDING EFFECTS**

Modern synthesizers have the ability to add effects to sounds. Sonic Pi 2 is no different, you are able to add studio effects such as reverb, echo and distortion. Of course you have to use code to add the effects!

1. In a new worksheet find a sample that you like for example sample :guit\_e\_fifths
2. Wrap the sample in an effects block like this:

with\_fx :reverb do

sample :guit\_e\_fifths

end

1. You can add effects on top of effects like this:

with\_fx :reverb do

with\_fx :distortion do

sample :guit\_e\_fifths

end

end

1. Play around with some effects and add them to your music. Remember that a complete list of effects can be found in the help section of Sonic Pi 2 under **FX**.

**STEP 2: MODIFYING PARAMETERS**

On occasion, you might like to make sounds play for a longer time or at a different rate. This can be achieved easily by modifying the parameters of the code you are using.

Take play 60, for example.

1. Click on **help** to open the help documents, then select **lang** on the left hand side, and scroll down to**play**. You will see some examples of its use. So far you have used play without any parameters; let's use some now.
2. In a new worksheet type:

play 60, attack: 1, release: 3

1. Press the **play** button to hear how that one note sounds. Attack and release control the amplitude of a note over time.
2. Now change the values for attack and release to see how those parameters affect the note.

There are lots of parameters that can also change the way samples or synths sound. Try changing the values for cutoff:, pan:, rate: or amp:. For a full list of parameters for each sample click on the **Help** icon, followed by **Samples**, select a sample and scroll down to see a full explanation for each type of paramter that can be used with that sample. Same goes for synths!

For a full list of parameters for each sample, click on the **Help** icon, followed by **Samples**. Select a sample and scroll down to see a full explanation for each type of parameter that can be used with that sample. The same applies for synths!

**STEP 3: USING RRAND**

Sonic Pi includes a number of functions that can add more interesting elements to your music. A really fun function is rrand, which will return a value between two specified numbers. For a cool effect, use rrand to make the cutoff bounce around.

1. In a blank worksheet type:

loop do

play chord(:a3, :minor).choose, attack: 0, release: 0.3, cutoff: 80

sleep 0.2

end

1. Instead of passing a number like 80 to the cutoff value, try rrand(40, 120) like this:

loop do

play chord(:a3, :minor).choose, attack: 0, release: 0.3, cutoff: rrand(40, 120)

sleep 0.2

end

1. Then you can start to experiment using rrand with other parameters. For example, add pan: rrand(-1, 1) to the play chord line and then press **play**.