Dynamic Audio for Digital Media
Tutorial 2 - TO BE COMPLETED FOR WEEK 3

1. Open the patch located in the *Examples* folder
2. Make sure that you understand the functionality and relationship of the following objects:

`array`
`soundfiler`
`openpanel`

Right-click on them to load the help file patches and properties windows if you have any questions about the objects.

3. Click on the `bang` above the `openpanel` and load the example soundfile *brahms.wav,* included in the folder.

4. Switch the toggle "Save contents" ON, on the `array` properties window.

5. Save the Pd patch as **wk2-tut2.pd** and close it. Reopen it and make sure that the *brahms.wav* soundfile is now stored in the two `arrays` (Left for the left channel, and Right for the right channel of the stereo file).

6. All you need in order to play back the contents of the two `arrays`, is an `expr`, a `phasor~`, two `tabread4~` (one for each array), and a `dac~` object. Create those and right-click on them to access their help files if needed.

7. The `tabread4~` objects take care of the playback by accessing our two `arrays.` Give our new `tabread4~` objects the arguments "Left" and "Right" (These are matching the `array` names, and it's basically how the `tabread4~` objects will know which `arrays` to access). Remember to leave a space between object name and argument. We now need an object to tell `tabread4~` *how* to read the `arrays'` data.

8. `phasor~` outputs audio rate linear ramps between 0 and 1. For this reason, it is often used as a means of reading data from tables, as we'll do so in this case. In order to make sure that the `arrays` will be read at the right rate, we need to ensure that **1)** the `phasor~` will be reading through the *whole* `arrays` and **2)** that the `phasor~` will be reading them at the right speed.

9. The output of `soundfiler` gives you, in samples, the length of the soundfile when you load it. Given that our sample rate is 44100, you will first have to perform the following division, using the `expr` object.

Sample Rate / Soundfile Length

10. This division's output will be the input to `phasor~`, as it gives us the correct playback rate (frequency).

11. Also, since we know that the `phasor~'s` output goes from 0 to 1, we'll have to multiply it with the length of the loaded soundfile in samples (Again, taking this from the output of `soundfiler`). Also, keep in mind that we're dealing with audio rate here so we should be using audio rate multiplication. `phasor~` will now be reading linearly through all the stored sample values of the two arrays through the `tabread4~` objects, at the correct playback rate (frequency).

12. Connect the `tabread4~ Left` object to the `dac~`'s left inlet and `tabread4~ Right` to the `dac~`'s right inlet. Turn the audio ON. You should now be able to hear the soundfile playing back. If you can't hear any sound, you might need to reload *brahms.wav* into the `soundfiler`, in order for its output to resend the soundfile's sample length - remember, the length is being outputted only when you first load the file. Every time you create new objects using that value, you will have to either reload the file, or store that value in a message, or `float` object, and get it from there.

13. Challenge: Can you find a way to implement a slider that controls the playback speed on the fly, from reverse to normal (-1 to 1) and everything in between? Hint: You will need to scale the slider's range (-1 to 1) with the `expr` division's output.