



# Exercise 1

Introduction to audio manipulation with Python

# Why Python

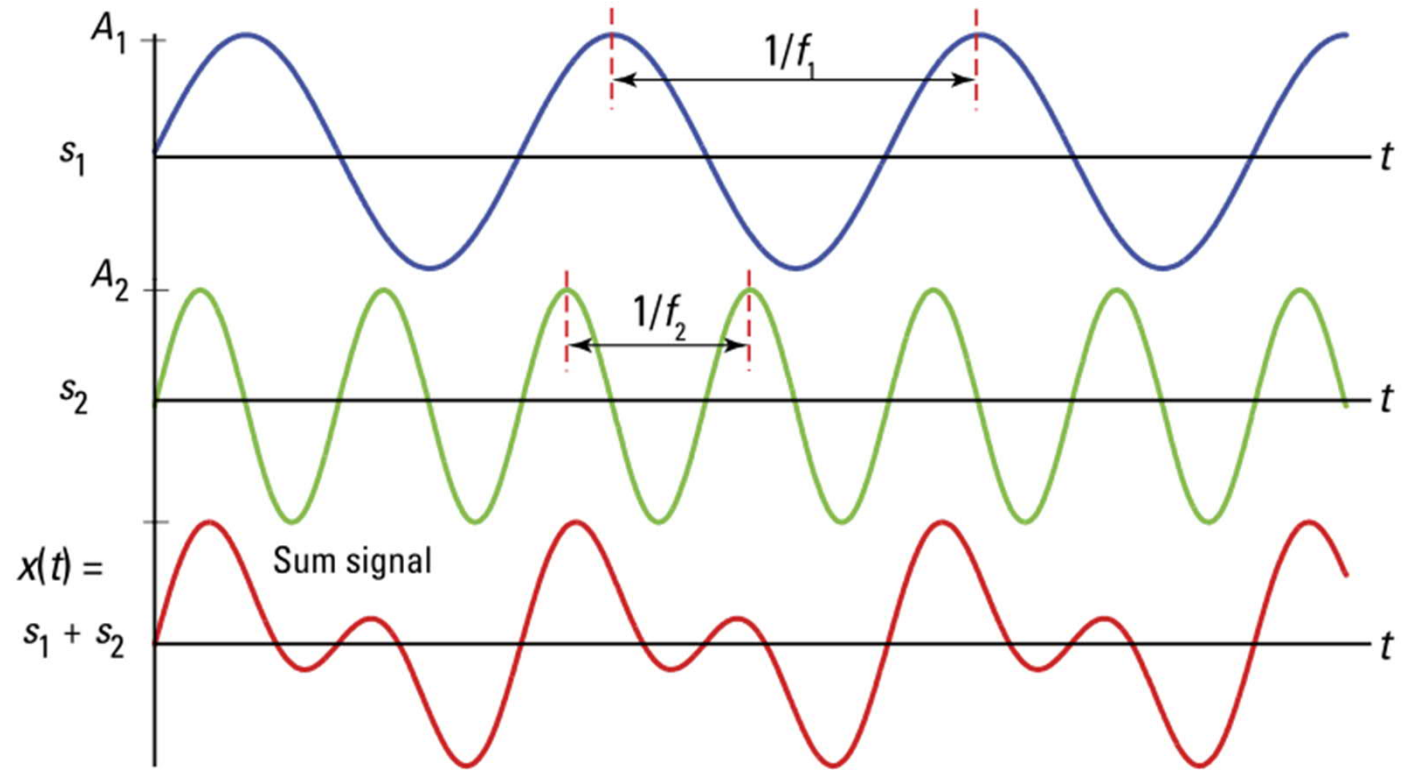
- **Easy** to get started.
- There is lots of **educational material** available online.
- Different libraries such as **Numpy**, **Scipy** and **matplotlib** form the bases of almost everything.
- It is **open source**.

Python 2.0 became unsupported in 2020.

# Problem 1

- Synthetic signal:  
Sum of sinusoids

import numpy as np



# Python audio manipulation

```
import sounddevice as sd
```

PortAudio wrapper providing realtime audio I/O with NumPy

Can be used to play audiofiles.

```
import soundfile as sf
```

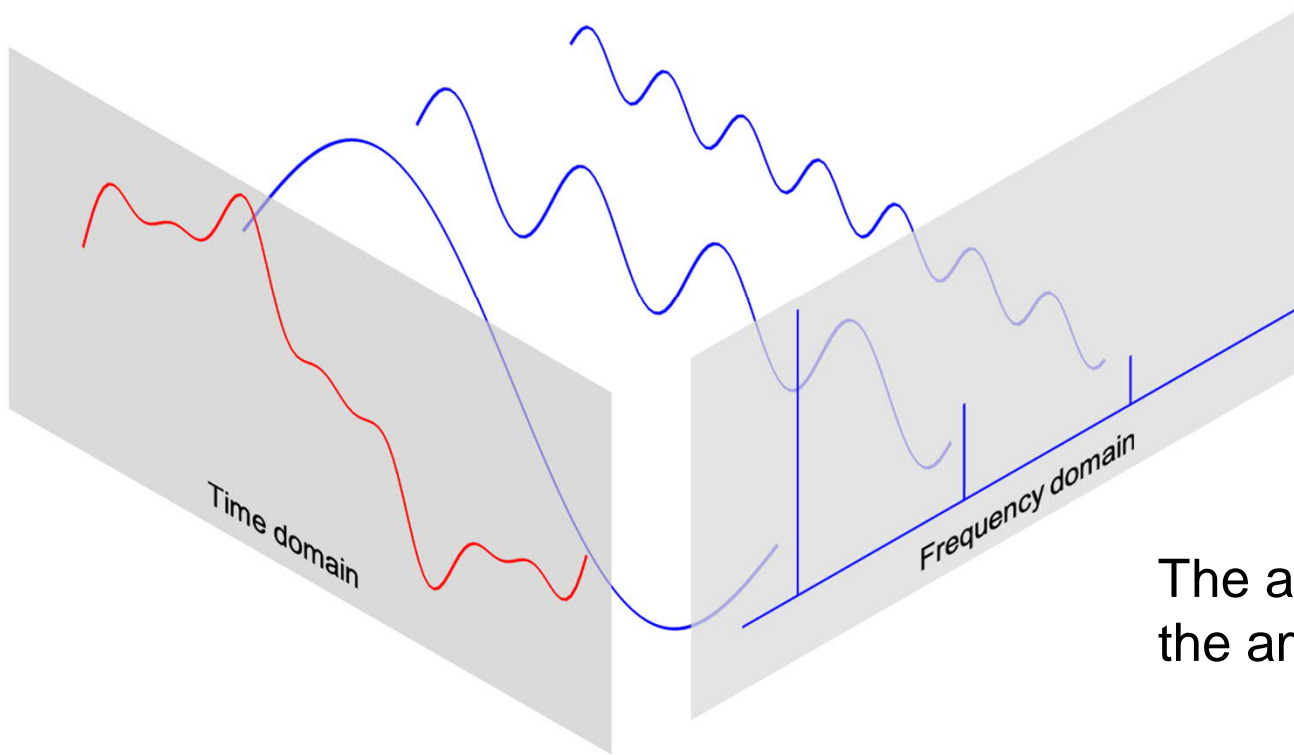
Can read and write sound files.

```
from scipy import signal
```

scipy is the core package for scientific routines in Python.

# Discrete Fourier Transform (DFT)

The DFT transforms a signal into the frequency domain.



The amplitudes returned by the DFT equal to the amplitudes of the signals fed into the DFT.

# Fast Fourier Transform

FFT fast algorithm to compute the Discrete Fourier Transform.

```
from scipy.fftpack import fft
```

it is faster than the one implemented in NumPy

# Notes

- YouTube [video](#) explaining **Sound and Waveforms**
- [Scipy](#) Lecture Notes
- Python programming [book](#)