

Source Code 1<sup>st</sup> VER:

```
import numpy as np
import matplotlib.pyplot as plt
import librosa

# 1. Load any audio file (mp3, wav, flac, etc.)
filename = "Dewa - Dua Sedjoli.mp3" # <<<< input file here
data, fs = librosa.load(filename, sr=None, mono=True)
# 'sr=None' = keep original sampling rate
# 'mono=True' = convert to mono

# 2. Normalize
data /= np.max(np.abs(data))

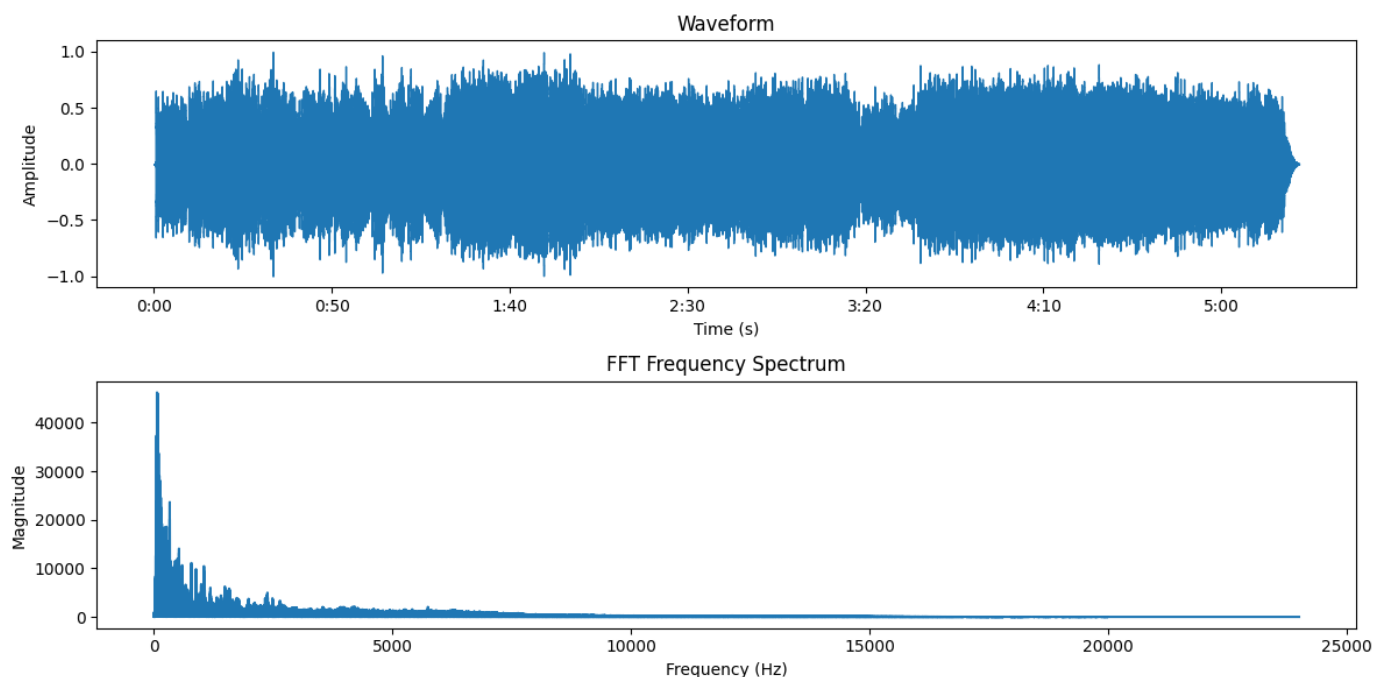
# 3. Fast fourier transform
N = len(data)
fft_data = np.fft.fft(data)
freqs = np.fft.fftfreq(N, 1/fs)

# Set the figure size
plt.figure(figsize=(12,6))

# 4. Plot waveform
plt.subplot(2,1,1)
librosa.display.waveshow(data, sr=fs)
plt.title("Waveform")
plt.xlabel("Time (s)")
plt.ylabel("Amplitude")

# 5. Plot magnitude spectrum (only positive frequencies)
plt.subplot(2,1,2)
plt.plot(freqs[:N//2], np.abs(fft_data[:N//2]))
plt.title("FFT Frequency Spectrum")
plt.xlabel("Frequency (Hz)")
plt.ylabel("Magnitude")

# Set the window title after the figure is created
plt.gcf().canvas.manager.set_window_title("FFT 1st ver.py")
plt.tight_layout()
plt.show()
```



Source Code 2<sup>nd</sup> VER:

```
import numpy as np
import matplotlib.pyplot as plt
import soundfile as sf

# === 1. Load audio ===
filename = "Dewa - Dua Sedjoli.mp3" # <<<< input file here
data, fs = sf.read(filename, dtype='float32')

# If stereo, convert to mono
if data.ndim > 1:
    data = data.mean(axis=1)

# === Normalize ===
data /= np.max(np.abs(data))

# === 2. Time axis ===
N = len(data)
time = np.linspace(0, N/fs, N)

# === 3. FFT ===
fft_data = np.fft.fft(data)
freqs = np.fft.fftfreq(N, 1/fs)

# === Set figure size ===
plt.figure(figsize=(12, 6))

# === 4. Plot waveform ===
plt.subplot(2,1,1)
plt.plot(time, data, linewidth=0.8)
plt.title("Waveform")
plt.xlabel("Time (s)")
plt.ylabel("Amplitude")
plt.xlim(0, time[-1]) # start at 0, stop at end

# === 5. Plot FFT spectrum ===
plt.subplot(2,1,2)
plt.plot(freqs[:N//2], np.abs(fft_data[:N//2]), linewidth=0.8)
plt.title("FFT Frequency Spectrum")
plt.xlabel("Frequency (Hz)")
plt.ylabel("Magnitude")
plt.xlim(0, freqs[N//2 - 1]) # start at 0, stop at Nyquist

# === Set window title ===
plt.gcf().canvas.manager.set_window_title("FFT 2nd ver.py")
plt.tight_layout()
plt.show()
```

