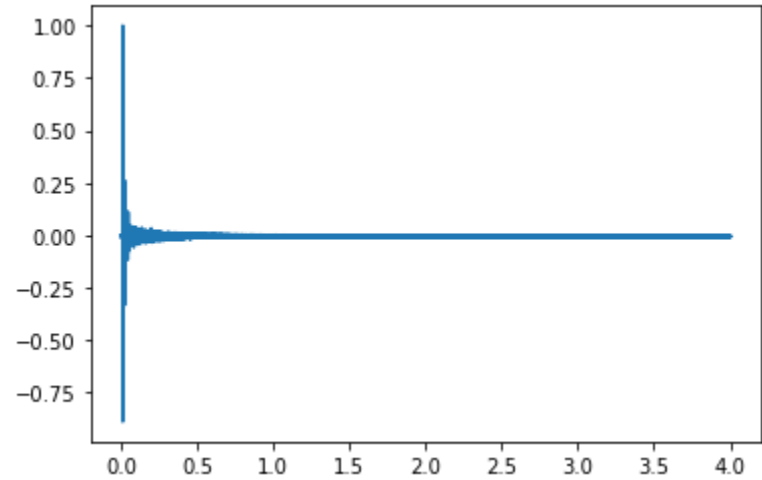


저는 터널안 환경 소리를 다운 받았습니다.

```
In [1]: import os
if not os.path.exists('../thinkdsp.py'):
    !wget https://github.com/AllenDowney/ThinkDSP/raw/master/code/thinkdsp.py

import numpy as np
import matplotlib.pyplot as plt
from thinkdsp import read_wave
from thinkdsp import decorate
```

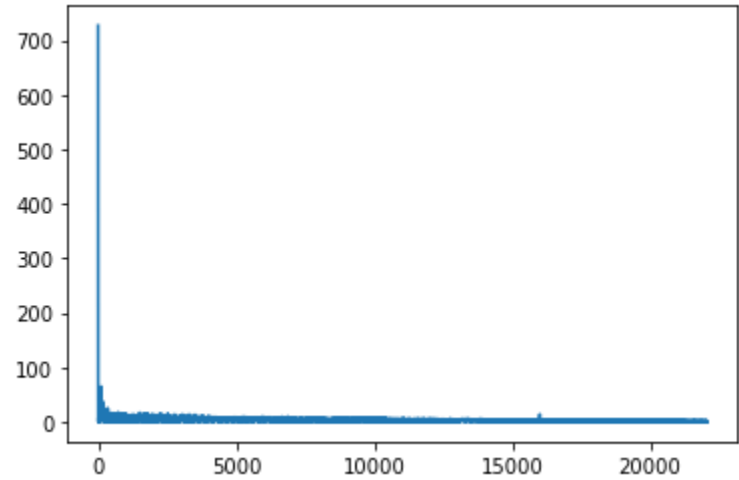
```
In [2]: response = read_wave('mono.wav')
duration = 4
response = response.segment(duration=duration)
response.normalize()
response.plot()
```



```
In [3]: response.make_audio()
```



```
In [4]: transfer = response.make_spectrum()
transfer.plot()
```

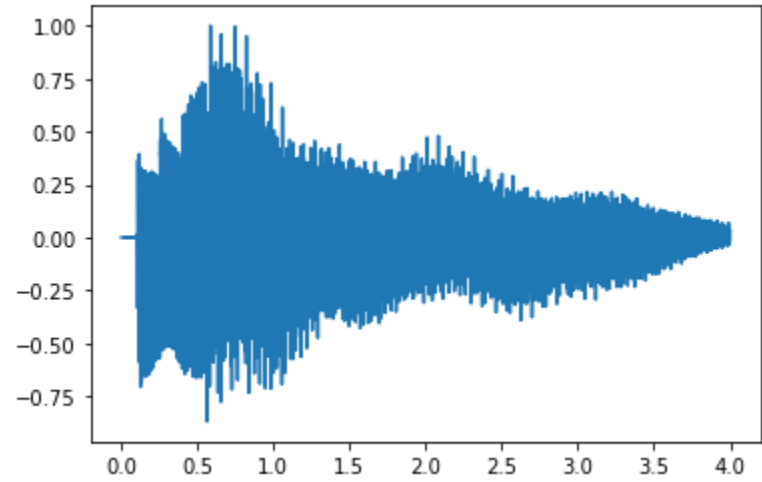


저는 기타 소리를 다운받았습니다

터널 안에서 기타치는 소리를 만들어 보겠습니다.

```
In [5]: wave = read_wave('aminopen.wav')
wave = wave.segment()

wave.truncate(len(response))
wave.normalize()
wave.plot()
```



```
In [6]: wave.make_audio()
```



```
In [7]: spectrum = wave.make_spectrum()
```

```
In [8]: len(spectrum.hs), len(transfer.hs)
```

```
Out[8]: (88201, 88201)
```

```
In [9]: spectrum.fs
```

```
Out[9]: array([0.000000e+00, 2.500000e-01, 5.000000e-01, ..., 2.204950e+04,
                2.204975e+04, 2.205000e+04])
```

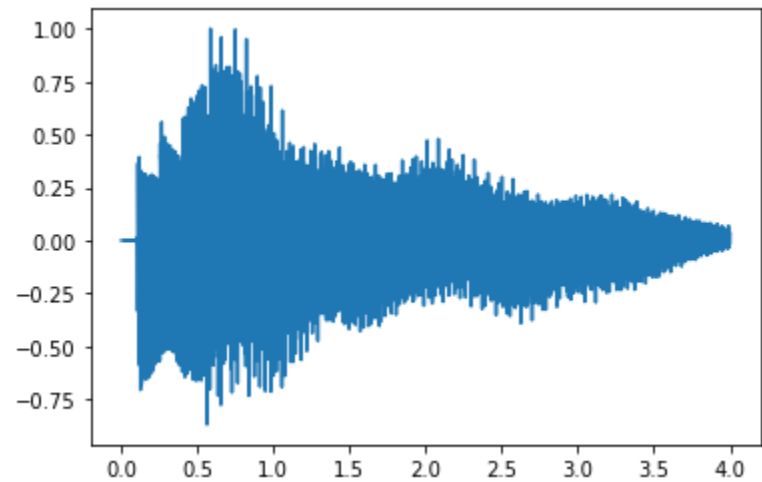
```
In [10]: transfer.fs
```

```
Out[10]: array([0.000000e+00, 2.500000e-01, 5.000000e-01, ..., 2.204950e+04,
                2.204975e+04, 2.205000e+04])
```

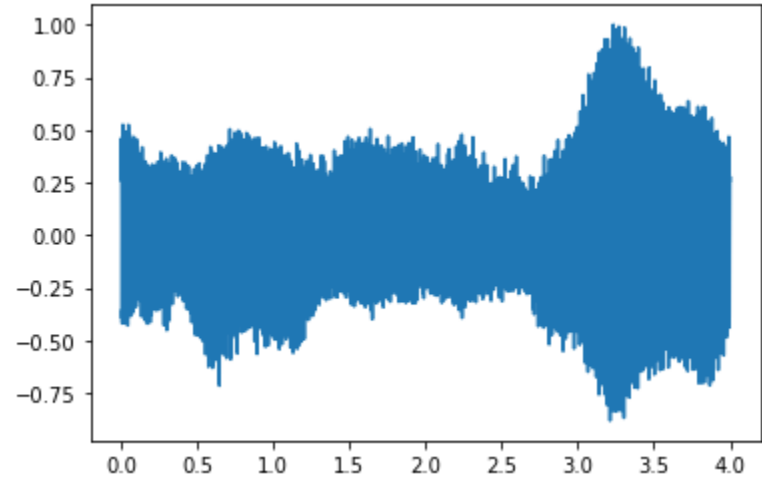
수업시간에 배운 공식을 사용해줍니다.

```
In [11]: output = (spectrum * transfer).make_wave()
output.normalize()
```

```
In [12]: wave.plot()
```



```
In [13]: output.plot()
```



동굴안에서 기타치는 소리가 납니다!

```
In [14]: output.make_audio()
```



컨볼루션 방식으로 만들어도 똑같은 소리가 납니다!

```
In [15]: convolved2 = wave.convolve(response)
convolved2.normalize()
convolved2.make_audio()
```

