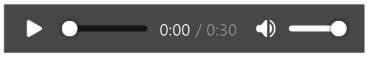
In [1]:
```
import librosa
import librosa.display
import IPython.display as ipd
import matplotlib.pyplot as plt
import numpy as np
```

Loading audio files with Librosa

In [2]:
```
audio_file = "audio/debussy.wav"
```

In [3]:
```
ipd.Audio(audio_file)
```

Out[3]:



In [4]:
```
# load audio files with librosa
signal, sr = librosa.load(audio_file)
```

Extracting MFCCs

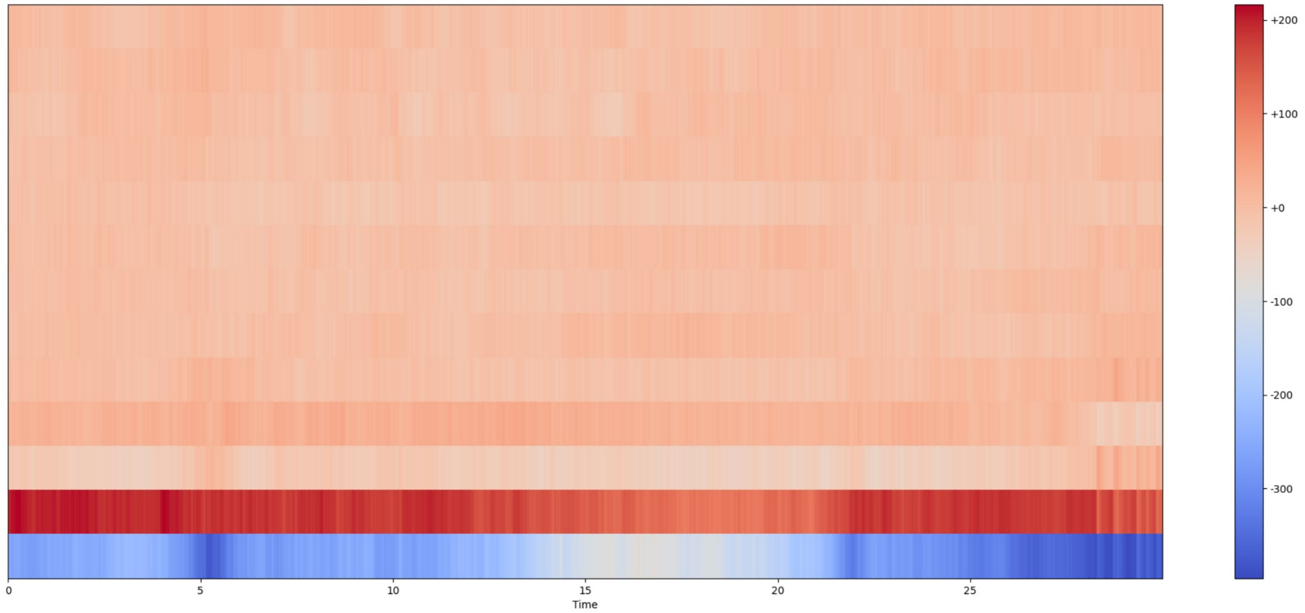
In [5]:
```
mfccs = librosa.feature.mfcc(y=signal, n_mfcc=13, sr=sr)
```

In [6]:
```
mfccs.shape
```

Out[6]:
```
(13, 1292)
```

Visualising MFCCs

In [7]:
```
plt.figure(figsize=(25, 10))
librosa.display.specshow(mfccs,
                         x_axis="time",
                         sr=sr)
plt.colorbar(format="%+2.f")
plt.show()
```



Computing first / second MFCCs derivatives

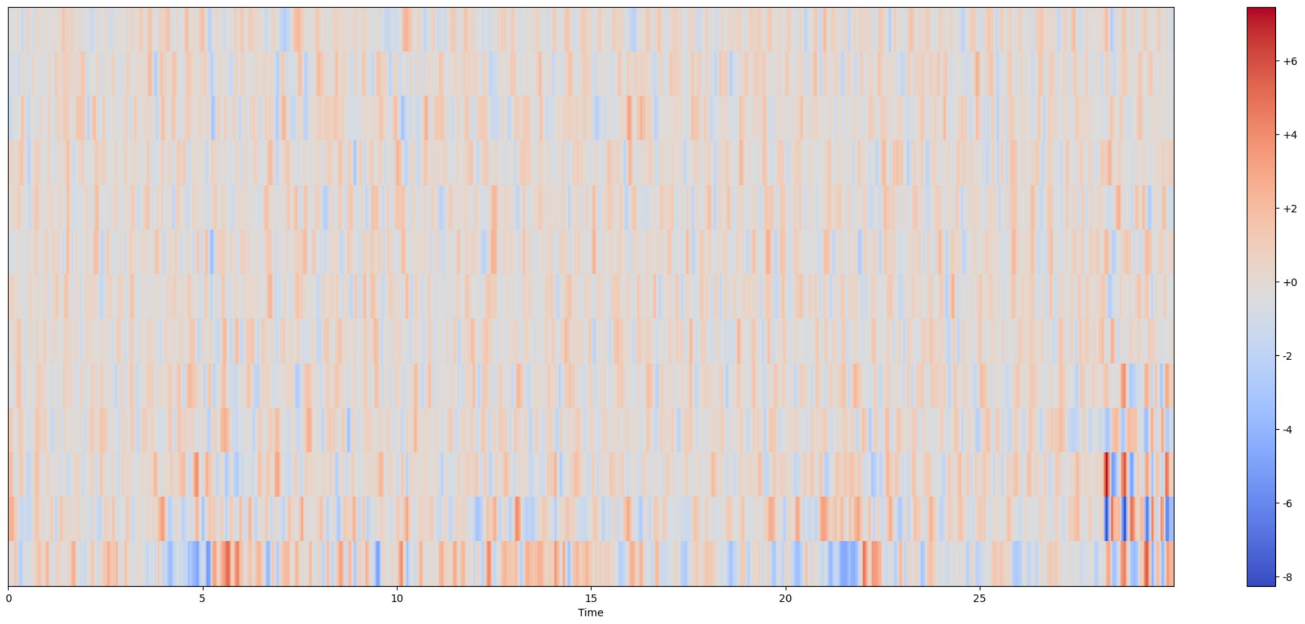
In [8]:
```
delta_mfccs = librosa.feature.delta(mfccs)
```

In [9]:
```
delta2_mfccs = librosa.feature.delta(mfccs, order=2)
```

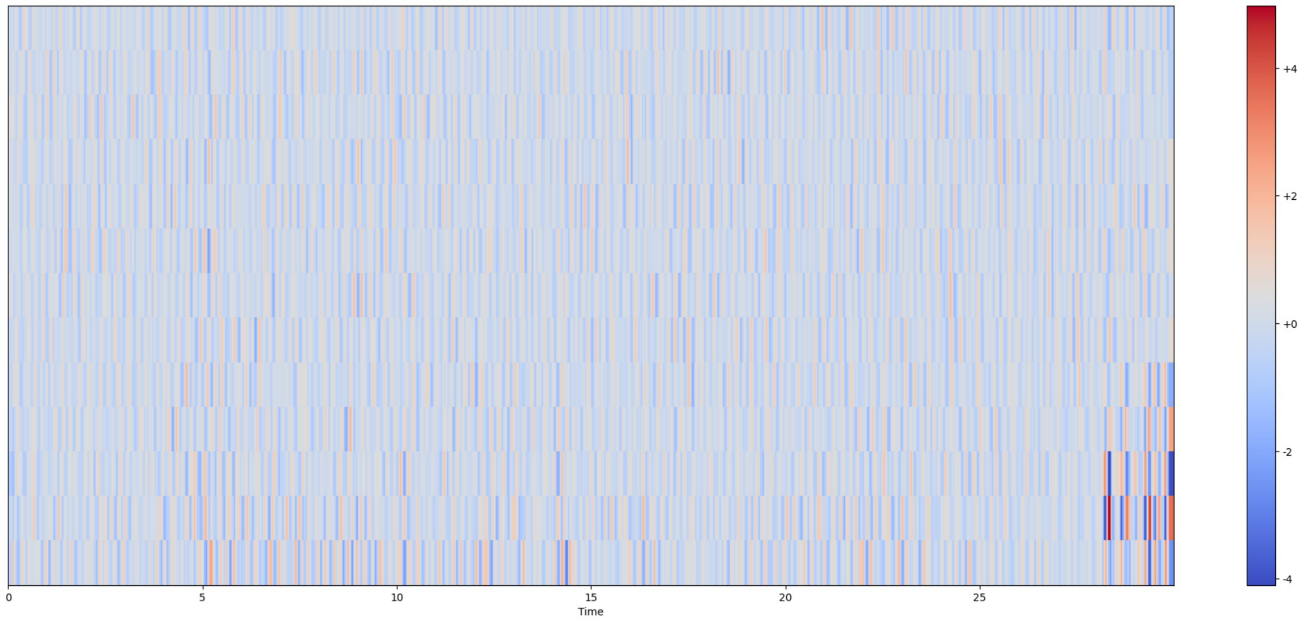
In [10]:
```
delta_mfccs.shape
```

Out[10]:
```
(13, 1292)
```

In [11]:
```
plt.figure(figsize=(25, 10))
librosa.display.specshow(delta_mfccs,
                         x_axis="time",
                         sr=sr)
plt.colorbar(format="%+2.f")
plt.show()
```



In [12]:
```
plt.figure(figsize=(25, 10))
librosa.display.specshow(delta2_mfccs,
                         x_axis="time",
                         sr=sr)
plt.colorbar(format="%+2.f")
plt.show()
```



In [13]:
```
mfccs_features = np.concatenate((mfccs, delta_mfccs, delta2_mfccs))
```

In [14]:
```
mfccs_features.shape
```

Out[14]:
```
(39, 1292)
```

In [ ]: