In [1]:

```
import os
import librosa
import librosa.display
import IPython.display as ipd
import numpy as np
import matplotlib.pyplot as plt
```

Loading audio files with Librosa

In [2]:

```
scale_file = "audio/scale.wav"
debussy_file = "audio/debussy.wav"
redhot_file = "audio/redhot.wav"
duke_file = "audio/duke.wav"
```

In [3]:

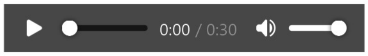
```
ipd.Audio(scale_file)
```

Out[3]:

In [4]:

```
ipd.Audio(debussy_file)
```

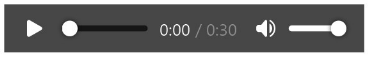
Out[4]:



In [5]:

```
ipd.Audio(redhot_file)
```

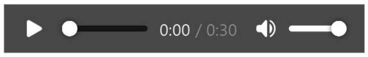
Out[5]:



In [6]:

```
ipd.Audio(duke_file)
```

Out[6]:



In [7]:

```
# load audio files with librosa
scale, sr = librosa.load(scale_file)
debussy, _ = librosa.load(debussy_file)
redhot, _ = librosa.load(redhot_file)
duke, _ = librosa.load(duke_file)
```

Extracting Short-Time Fourier Transform

In [8]:

```
FRAME_SIZE = 2048
HOP_SIZE = 512
```

In [9]:

```
S_scale = librosa.stft(scale, n_fft=FRAME_SIZE, hop_length=HOP_SIZE)
```

In [11]:

```
S_scale.shape
```

Out[11]:

```
(1025, 342)
```

In [13]:

```
type(S_scale[0][0])
```

Out[13]:

```
numpy.complex64
```

Calculating the spectrogram

In [14]:

```
Y_scale = np.abs(S_scale) ** 2
```

In [15]:

```
Y_scale.shape
```

Out[15]:

```
(1025, 342)
```

In [17]:

```
type(Y_scale[0][0])
```

Out[17]:

```
numpy.float32
```

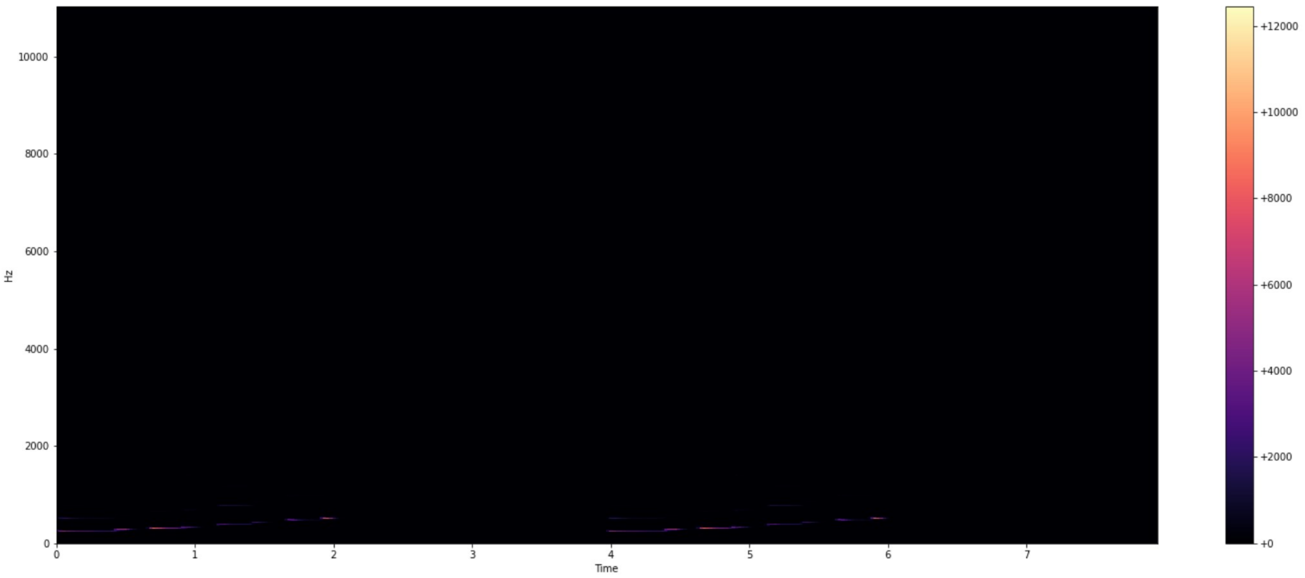
Visualizing the spectrogram

In [18]:

```
def plot_spectrogram(Y, sr, hop_length, y_axis="linear"):
    plt.figure(figsize=(25, 10))
    librosa.display.specshow(Y,
                             sr=sr,
                             hop_length=hop_length,
                             x_axis="time",
                             y_axis=y_axis)
    plt.colorbar(format="%+2.f")
```

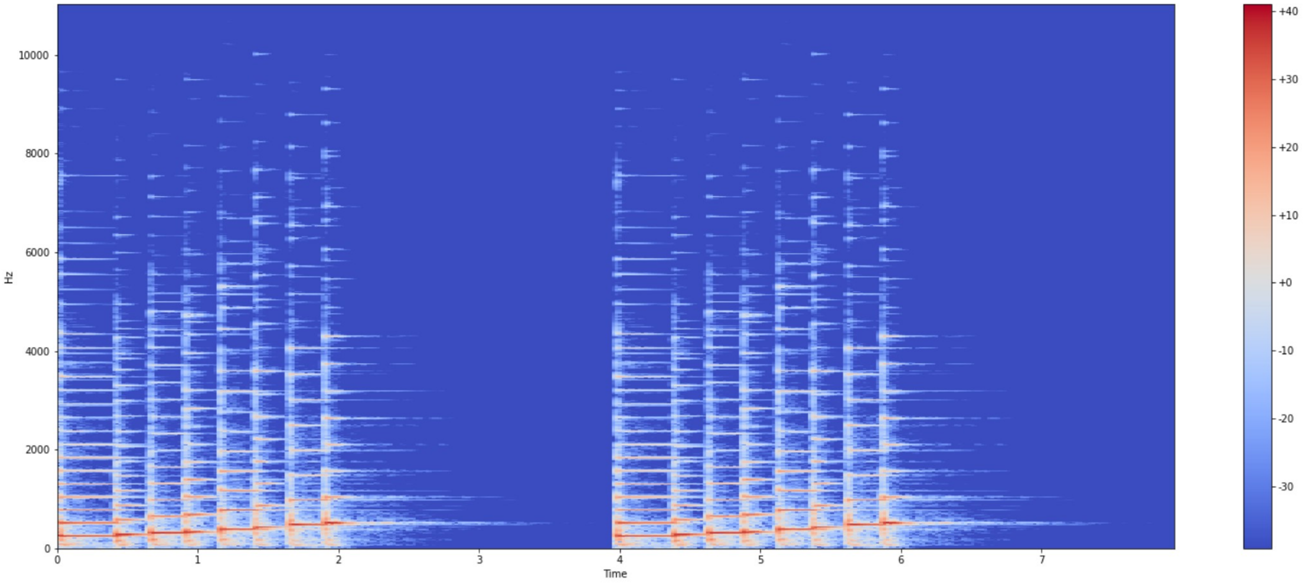
In [19]:

```
plot_spectrogram(Y_scale, sr, HOP_SIZE)
```



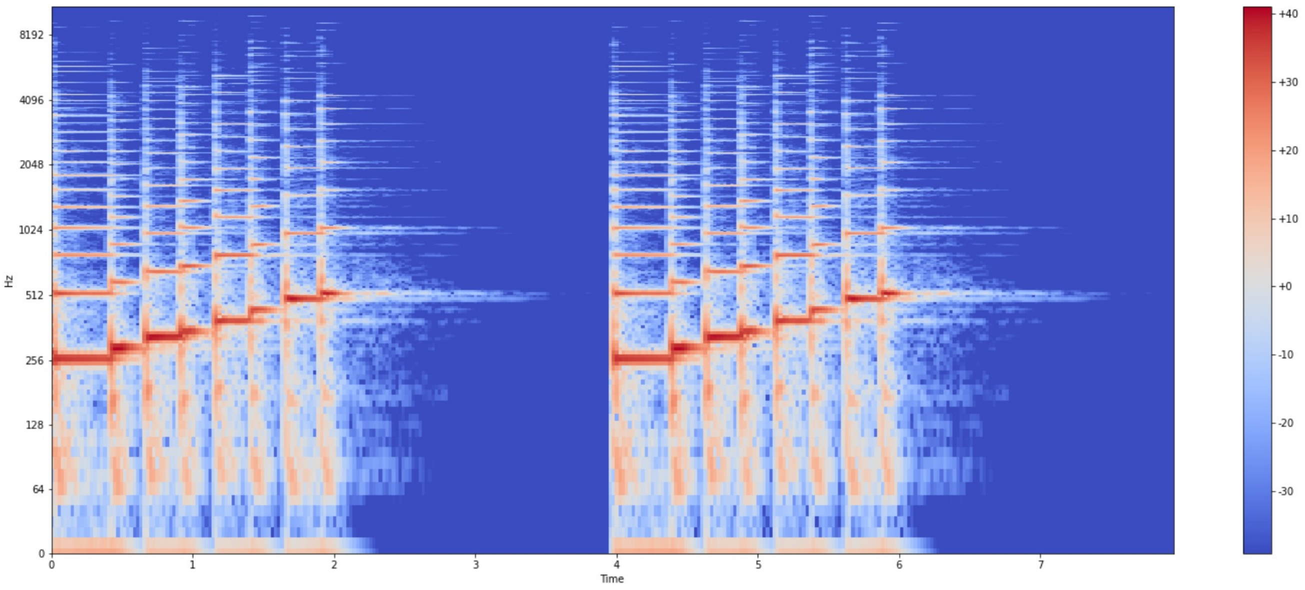
Log-Amplitude Spectrogram

In [21]:
```
Y_log_scale = librosa.power_to_db(Y_scale)
plot_spectrogram(Y_log_scale, sr, HOP_SIZE)
```



Log-Frequency Spectrogram

In [22]:
```
plot_spectrogram(Y_log_scale, sr, HOP_SIZE, y_axis="log")
```

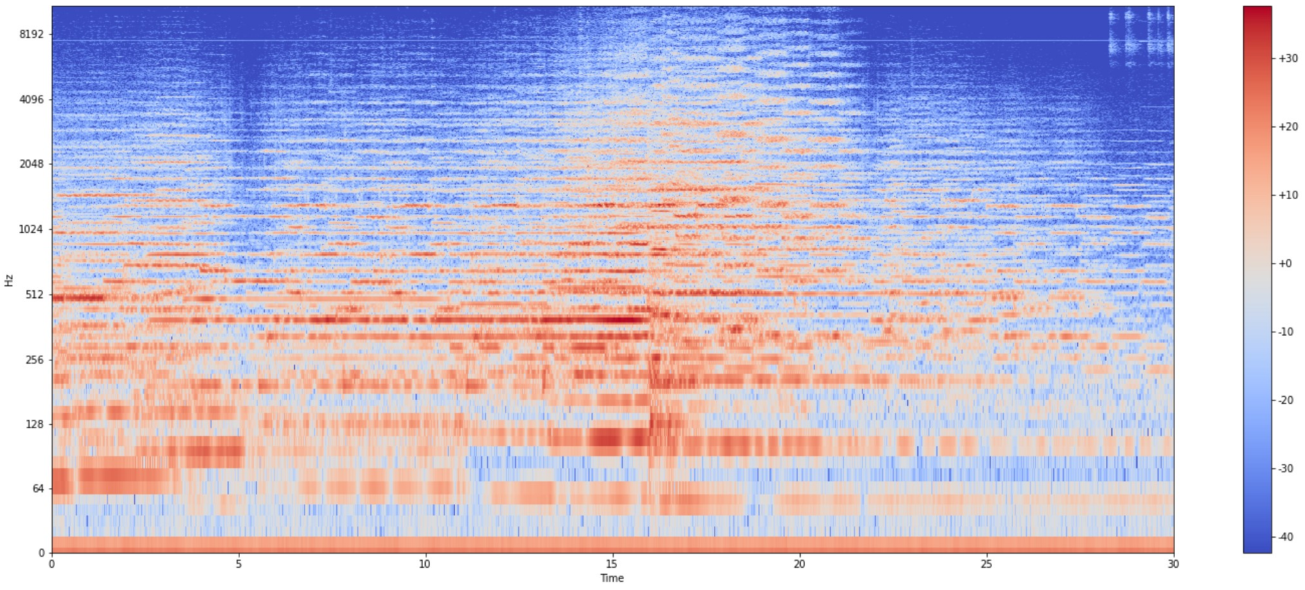


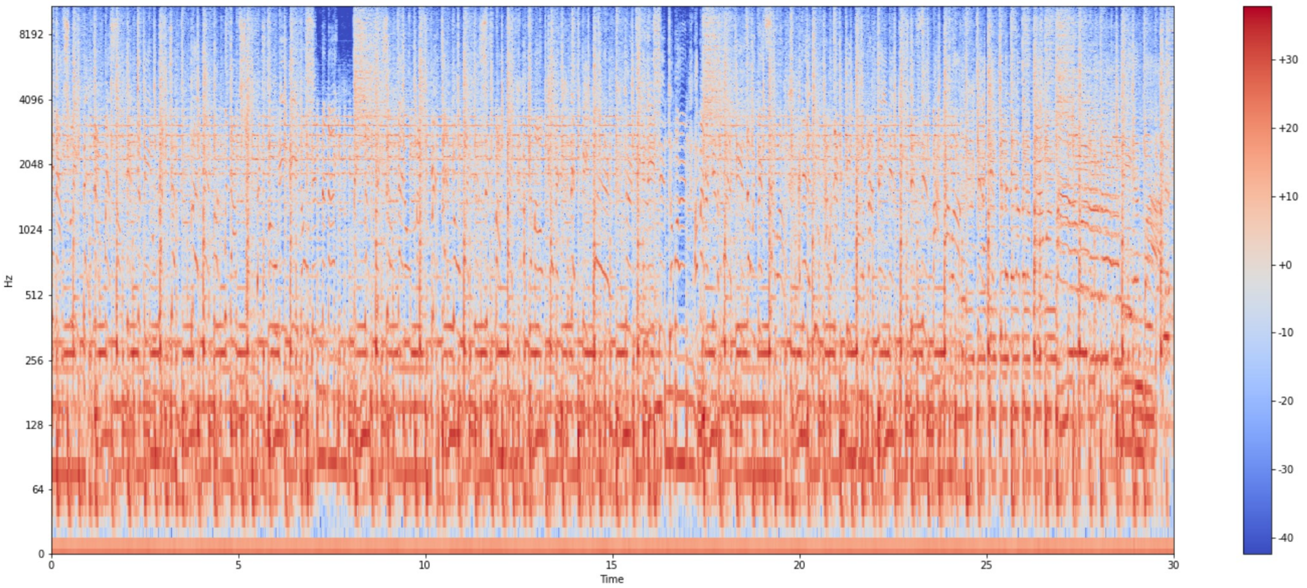
Visualising songs from different genres

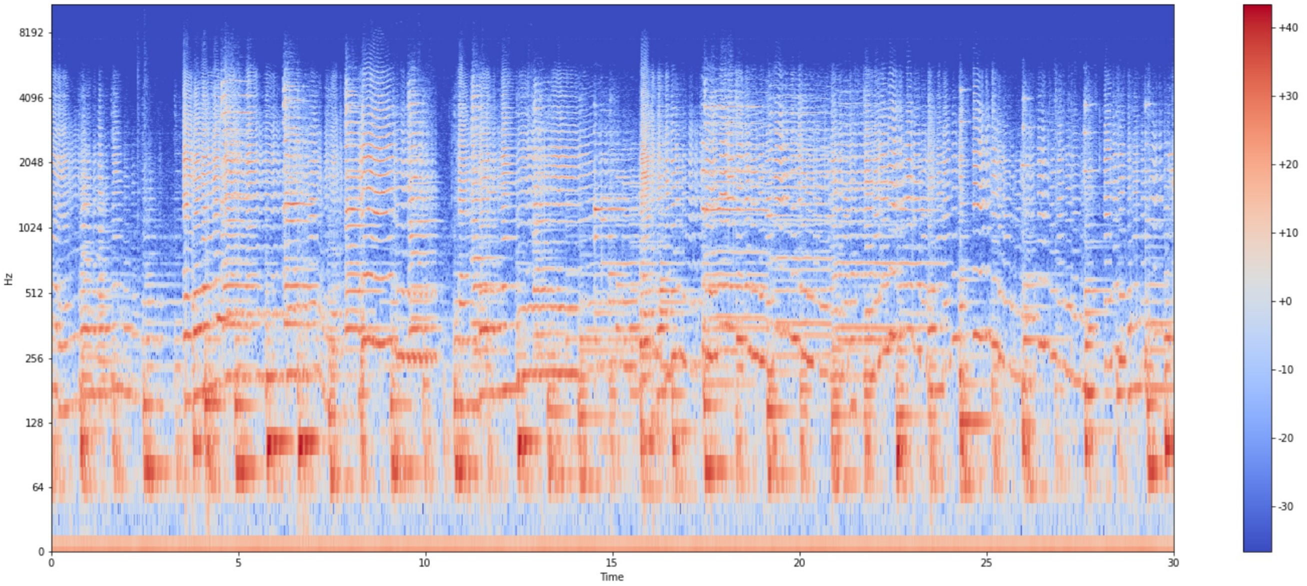
```
In [23]: S_debussy = librosa.stft(debussy, n_fft=FRAME_SIZE, hop_length=HOP_SIZE)
         S_redhot = librosa.stft(redhot, n_fft=FRAME_SIZE, hop_length=HOP_SIZE)
         S_duke = librosa.stft(duke, n_fft=FRAME_SIZE, hop_length=HOP_SIZE)


         Y_debussy = librosa.power_to_db(np.abs(S_debussy) ** 2)
         Y_redhot = librosa.power_to_db(np.abs(S_redhot) ** 2)
         Y_duke = librosa.power_to_db(np.abs(S_duke) ** 2)

         plot_spectrogram(Y_debussy, sr, HOP_SIZE, y_axis="log")
         plot_spectrogram(Y_redhot, sr, HOP_SIZE, y_axis="log")
         plot_spectrogram(Y_duke, sr, HOP_SIZE, y_axis="log")
```




8192
4096
2048
1024
512
256
128
64
0
Hz
0
5
10
15
20
25
30
Time
+30
+20
+10
+0
-10
-20
-30
-40


8192
4096
2048
1024
512
256
128
64
0
Hz
0
5
10
15
20
25
30
Time
+40
+30
+20
+10
+0
-10
-20
-30