```matlab
clc; clear all; close all;

%% Plot the original signal
[y1, Fs1] = audioread('music1.wav');
S1 = y1';
t1 = (1:length(S1))/Fs1;
% % figure(1)
% % plot(t1, S1);
% % xlabel('Time [s]');
% % ylabel('Amplitude');
% % title('Mary had a little lamb (piano)');
% p81 = audioplayer(S1, Fs1);
% playblocking(p81);

[y2, Fs2] = audioread('music2.wav');
S2 = y2';
t2 = (1:length(S2))/Fs2;
% % figure(2)
% % plot(t2, S2);
% % xlabel('Time [sec]');
% % ylabel('Amplitude');
% % title('Mary had a little lamb (recorder)');
% p82 = audioplayer(S2, Fs2);
% playblocking(p82);


%% Setup the time and frequency domain
n1 = length(t1);
L1 = t1(n1);
k1 = (2*pi/L1)*[0:n1/2-1 -n1/2:-1];
ks1 = fftshift(k1);

n2 = length(t2);
L2 = t2(n2);
k2 = (2*pi/L2)*[0:n2/2-1 -n2/2:-1];
ks2 = fftshift(k2);

%% FFT the signals
S1t = fft(S1);

S2t = fft(S2);

%% AUDIO 1 Compute spectrograms using Gaussian window
tslide1 = linspace(0, t1(n1), 80);
```

```
S1gt_spec = zeros(80, n1);
a1 = 80;
k01 = zeros(1, 80);
tau = 0.5;
S1gtf_spec = zeros(80, n1);
for i = 1 : length(tslide1)
    g1 = exp(-a1*(t1 - tslide1(i)).^2);
    S1g = g1.*S1;
    S1gt = fft(S1g);
    [M, I] = max(fftshift(abs(S1gt)));
    k01(i) = abs(ks1(I));
    gt1 = exp(-tau*(ifftshift(ks1) - k01(i)).^2);
    S1gtf = gt1.*S1gt;
    S1gtf_spec(i, :) =
fftshift(abs(S1gtf))/max(abs(S1gtf));
    S1gt_spec(i, :) = fftshift(abs(S1gt))/max(abs(S1gt));
    if i == 40
        figure(3)
        plot(t1, S1);
        hold on;
        plot(t1, g1);
        xlabel('Time [s]');
        ylabel('S1(t), g1(t)');
        title('Audio signal 1 and Gaussian window');
        saveas(gcf, 'P2_1.jpg');
    end
end
figure(4)
plot(tslide1, k01);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
title('Center frequencies of audio signal 1 throughout
time');
saveas(gcf, 'P2_1_CF.jpg');

figure(5)
pcolor(tslide1, ks1, S1gt_spec.');
shading interp;
colormap(hot);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
title('Spectrogram of audio signal 1');
axis([0 L1 200*2*pi 400*2*pi]);
saveas(gcf, 'P2_1_Spec.jpg');
```

```
figure(6)
pcolor(tslide1, ks1, S1gtf_spec.');
shading interp;
colormap(hot);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
title('Spectrogram of the filtered audio signal 1');
axis([0 L1 200*2*pi 400*2*pi]);
saveas(gcf, 'P2_1_Spec_Filter.jpg');

%% AUDIO 2 Compute spectrograms using Gaussian window
tslide2 = linspace(0, t2(n2), 80);
S2gt_spec = [];
a2 = 70;
k02 = zeros(1, 80);
tau = 0.2;
S2gtf_spec = zeros(80, n2);
for i = 1 : length(tslide2)
    g2 = exp(-a2*(t2 - tslide2(i)).^2);
    S2g = g2.*S2;
    S2gt = fft(S2g);
    [M, I] = max(fftshift(abs(S2gt)));
    k02(i) = abs(ks2(I));
    gt2 = exp(-tau*(ifftshift(ks2) - k02(i)).^2);
    S2gtf = gt2.*S2gt;
    S2gtf_spec(i, :) =
fftshift(abs(S2gtf))/max(abs(S2gtf));
    S2gt_spec(i, :) = fftshift(abs(S2gt))/max(abs(S2gt));
    if i == 40
        figure(3)
        plot(t2, S2);
        hold on;
        plot(t2, g2);
        xlabel('Time [s]');
        ylabel('S1(t), g1(t)');
        title('Audio signal 2 and Gaussian window');
        saveas(gcf, 'P2_2.jpg');
    end
end
figure(4)
plot(tslide2, k02);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
```

```
title('Center frequencies of audio signal 2 throughout time');
saveas(gcf, 'P2_2_CF.jpg');

figure(5)
pcolor(tslide2, ks2, S2gt_spec.');
shading interp;
colormap(hot);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
title('Spectrogram of audio signal 2');
axis([0 L1 700*2*pi 1100*2*pi]);
saveas(gcf, 'P2_2_Spec.jpg');

figure(6)
pcolor(tslide2, ks2, S2gtf_spec.');
shading interp;
colormap(hot);
xlabel('Time [s]');
ylabel('Frequency [\omega]');
title('Spectrogram of the filtered audio signal 2');
axis([0 L1 700*2*pi 1100*2*pi]);
saveas(gcf, 'P2_2_Spec_Filter.jpg');
```