Experiment 2

Aim:

1. To generate Siner signal and sample it at different sampling rates.
2. To reconstruct the original signal using signals sampled at different sampling rates.
3. To plot the original signal, sampled signals and their energy spectrum.
4. To observe the classical notes change with different sampling rates.
5. To observe the change in an audio with reduced sampling rates.

Observations and Code:

Q1)

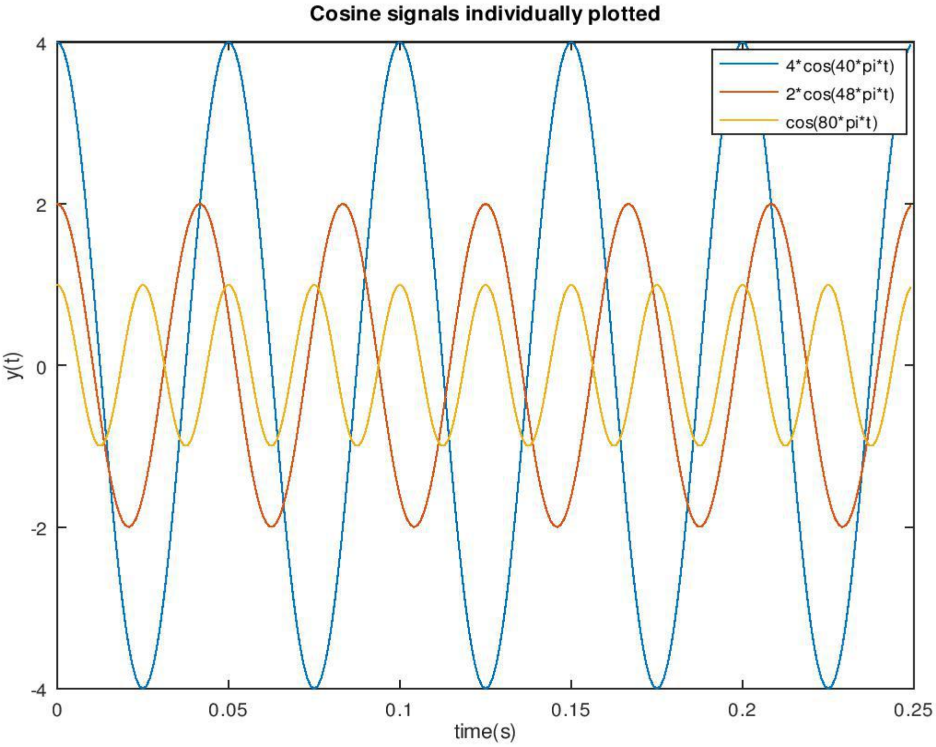
Code -

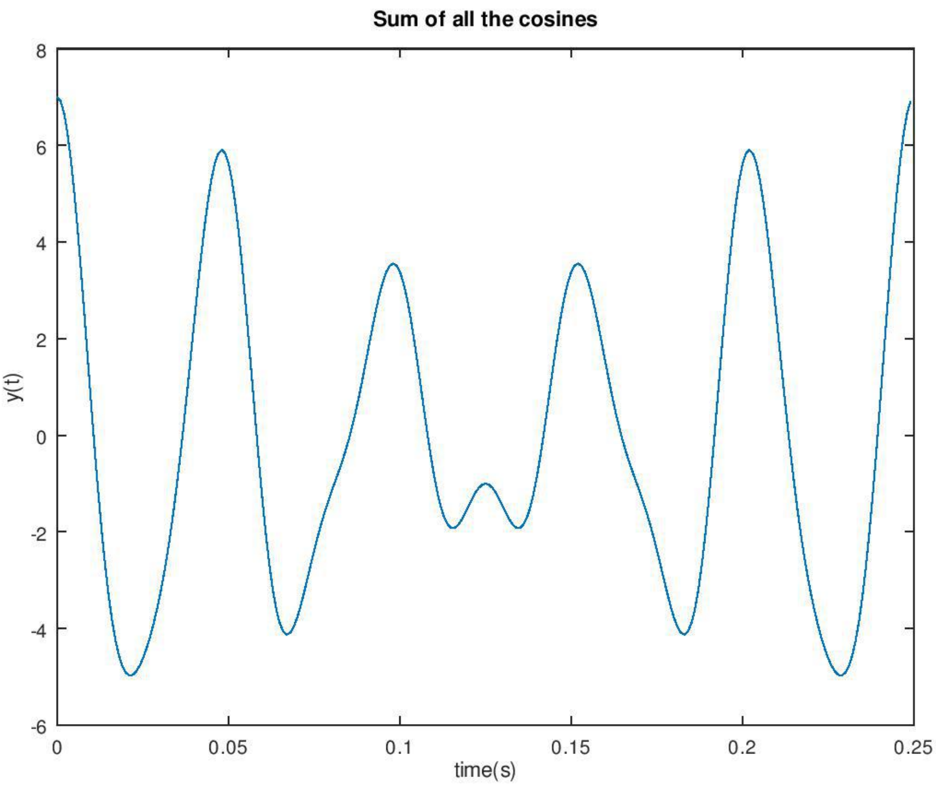
```
# a = 4

# Q1 - i

# setting dt
dt = 0.001;
t = [0:dt:1/4-dt];

# a part
y1 = 4*cos(2*pi*5*4*t);

# b part
y2 = 2*cos(2*pi*6*4*t);

# c part
y3 = 1*cos(2*pi*10*4*t);

figure();
plot(t,y1,t,y2,t,y3);
title('Cosine signals individually plotted');
legend(['4*cos(40*pi*t)';'2*cos(48*pi*t)';'cos(80*pi*t)']);
xlabel('time(s)');
ylabel('y(t)');


# Q1 - i

# total signal by adding all three cosines

y_sum = y1+y2+y3;
figure();
```

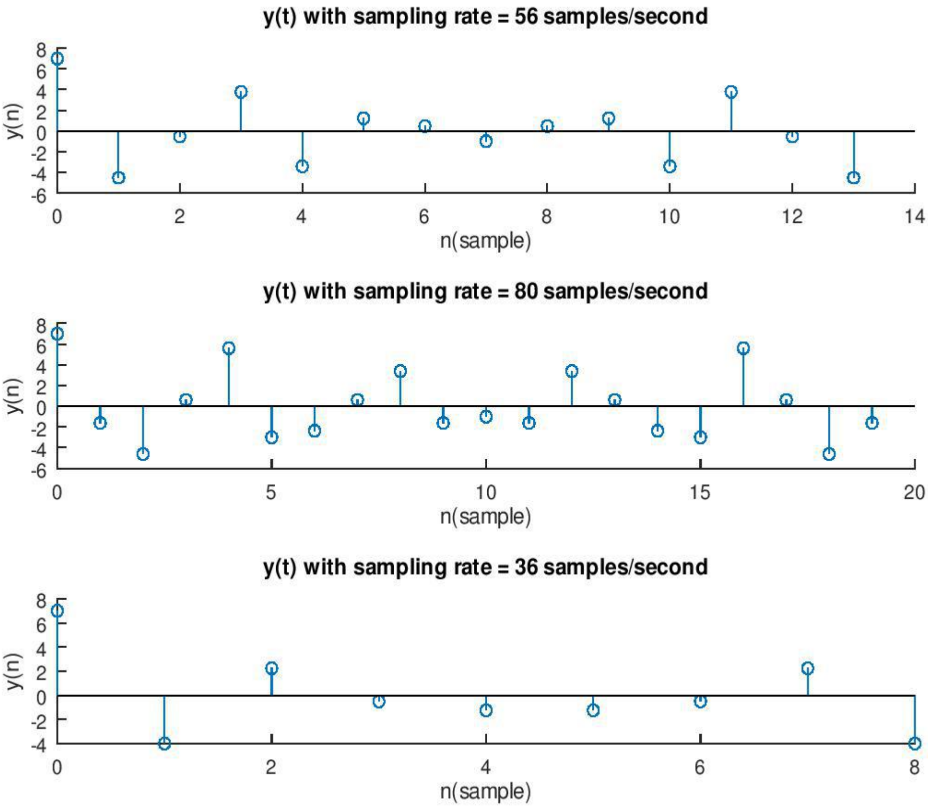
```
plot(t,y_sum);
title('Sum of all the cosines');
xlabel('time(s)');
ylabel('y(t)');

# Q1 - ii

# sampling signals at 56, 80 and 36 samples per second

Fs = 56;
dt = 1/Fs;
t = [0:dt:0.25-dt];
y_sum1 = 4*cos(2*pi*5*4*t) + 2*cos(2*pi*6*4*t) + 1*cos(2*pi*10*4*t);

Fs = 80;
dt = 1/Fs;
t = [0:dt:0.25-dt];
y_sum2 = 4*cos(2*pi*5*4*t) + 2*cos(2*pi*6*4*t) + 1*cos(2*pi*10*4*t);

Fs = 36;
dt = 1/Fs;
t = [0:dt:0.25-dt];
y_sum3 = 4*cos(2*pi*5*4*t) + 2*cos(2*pi*6*4*t) + 1*cos(2*pi*10*4*t);

figure();
subplot(3,1,1);
stem([0:length(y_sum1)-1],y_sum1);
title('y(t) with sampling rate = 56 samples/second');
xlabel('n(sample)');
ylabel('y(n)');

subplot(3,1,2);
stem([0:length(y_sum2)-1],y_sum2);
title('y(t) with sampling rate = 80 samples/second');
xlabel('n(sample)');
ylabel('y(n)');

subplot(3,1,3);
stem([0:length(y_sum3)-1],y_sum3);
title('y(t) with sampling rate = 36 samples/second');
xlabel('n(sample)');
ylabel('y(n)');


# Q1 - iii

# interpolating signals of different SR
```

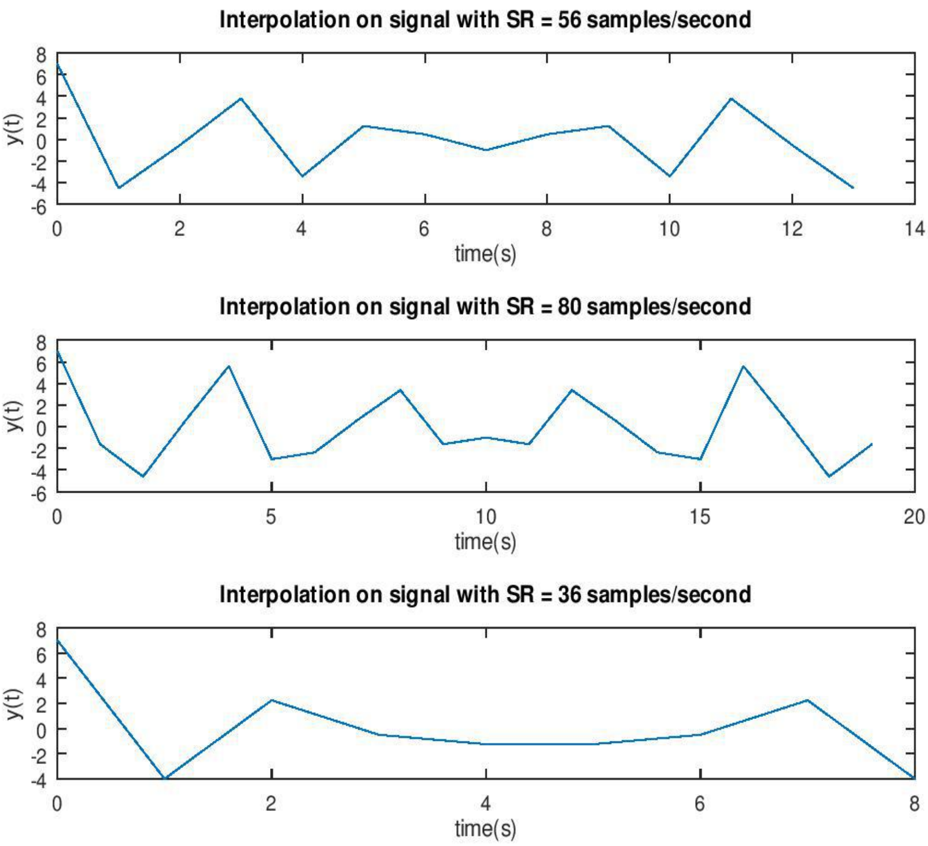
```
figure();
subplot(3,1,1);
plot([0:length(y_sum1)-1],y_sum1);
title('Interpolation on signal with SR = 56 samples/second');
xlabel('time(s)')
ylabel('y(t)');

subplot(3,1,2);
plot([0:length(y_sum2)-1],y_sum2);
title('Interpolation on signal with SR = 80 samples/second');
xlabel('time(s)')
ylabel('y(t)');

subplot(3,1,3);
plot([0:length(y_sum3)-1],y_sum3);
title('Interpolation on signal with SR = 36 samples/second');
xlabel('time(s)')
ylabel('y(t)');

# Q1 - iv

# calculating energies of signals
E1 = abs(fft(y_sum1)).^2;
E2 = abs(fft(y_sum2)).^2;
E3 = abs(fft(y_sum3)).^2;

figure();
subplot(3,1,1);
bar([0:length(E1)-1],E1);
title('Energy spectrum of signal sampled at 56 samples/second');
xlabel('n');
ylabel('Energy');

subplot(3,1,2);
bar([0:length(E2)-1],E2);
title('Energy spectrum of signal sampled at 80 samples/second');
xlabel('n');
ylabel('Energy');

subplot(3,1,3);
bar([0:length(E3)-1],E3);
title('Energy spectrum of signal sampled at 36 samples/second');
xlabel('n');
ylabel('Energy');
```

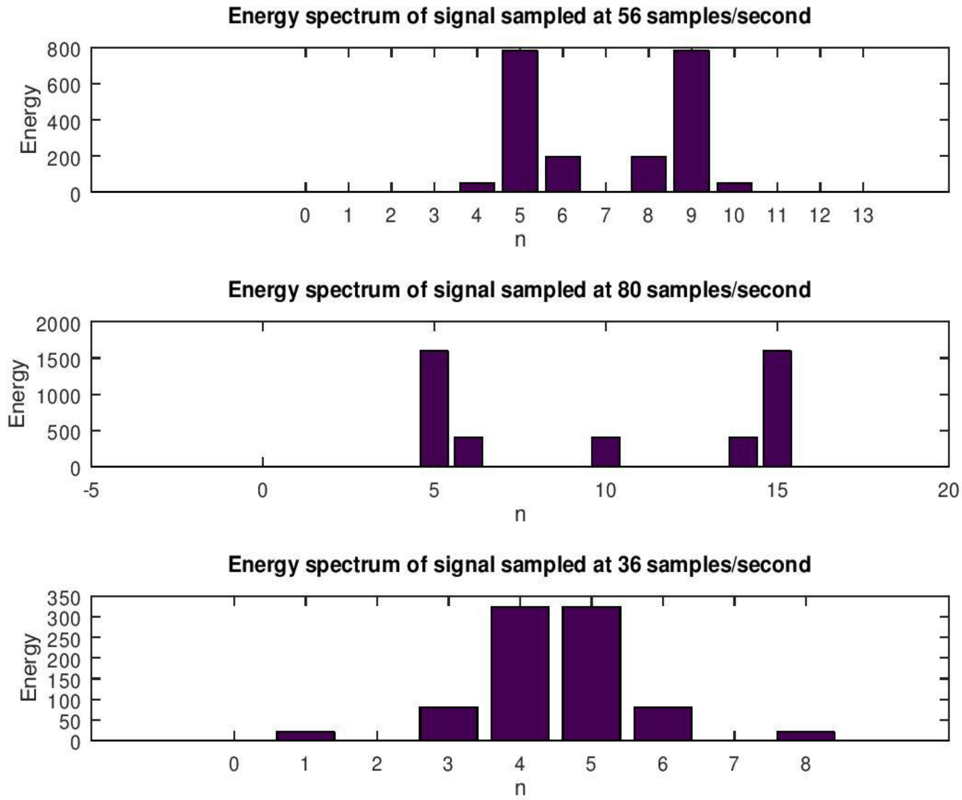
Graphs -


Cosine signals individually plotted
4*cos(40*pi*t)
2*cos(48*pi*t)
cos(80*pi*t)
y(t)
time(s)
4
2
0
-2
-4
0
0.05
0.1
0.15
0.2
0.25


Sum of all the cosines
y(t)
time(s)
8
6
4
2
0
-2
-4
-6
0
0.05
0.1
0.15
0.2
0.25


y(t) with sampling rate = 56 samples/second
y(n)
n(sample)
y(t) with sampling rate = 80 samples/second
y(n)
n(sample)
y(t) with sampling rate = 36 samples/second
y(n)
n(sample)


Interpolation on signal with SR = 56 samples/second
y(t)
time(s)
Interpolation on signal with SR = 80 samples/second
y(t)
time(s)
Interpolation on signal with SR = 36 samples/second
y(t)
time(s)



Observations:

1. The original signal consists cosines of frequencies 20 Hz, 24 Hz and 40 Hz. When sampled at rates below Nyquist rate which is 80 Hz, aliases are created and information is lost. So, the signal reconstructed using signals obtained by sampling at 56 samples per second and 36 samples per second are not as accurate as that constructed using the signal obtained by sampling at 80 samples per second.
2. As a result of the above, the second interpolated (SR = 80 samples/second) graph matches the original signal more. The peaks and valleys occur at the same time in both the cases. Also, it is able to correctly identify the harmonics present in the signal. The harmonics are $5\omega_0$, $6\omega_0$, and $10\omega_0$ with $\omega_0 = 4$ Hz, ie. 20 Hz, 24 Hz, 40 Hz.
3. From the energy spectrum, following can be seen:
   a) Signal obtained by sampling at 56 samples/second has aliases - 40 Hz to 16 Hz, 36 Hz to 20 Hz, 32 Hz to 24 Hz.
   b) Signal obtained by sampling at 36 samples/second has aliases - 32 Hz to 4 Hz, 24 Hz to 12 Hz, 20 Hz to 16 Hz.

Q2)

Code -

```
# Notes with SR = 1000 samples/second
```

```
Fs = 1000;
dt = 1/Fs;
t = [0:dt:0.5-dt];

sa = sin(2*pi*525*t);
re = sin(2*pi*590*t);
ga = sin(2*pi*664*t);
ma = sin(2*pi*704*t);
pa = sin(2*pi*790*t);
da = sin(2*pi*885*t);
ni = sin(2*pi*995*t);
sa2 = sin(2*pi*1055*t);

y = [sa re ga ma pa da ni sa2];
wavwrite(y,Fs,"audio1.wav");

# Notes with SR = 1600 samples/second
Fs = 1600;
dt = 1/Fs;
t = [0:dt:0.5-dt];

sa = sin(2*pi*525*t);
re = sin(2*pi*590*t);
ga = sin(2*pi*664*t);
ma = sin(2*pi*704*t);
pa = sin(2*pi*790*t);
da = sin(2*pi*885*t);
ni = sin(2*pi*995*t);
sa2 = sin(2*pi*1055*t);

y = [sa re ga ma pa da ni sa2];
wavwrite(y,Fs,"audio2.wav");

# Notes with SR = 2200 samples/second
Fs = 2200;
dt = 1/Fs;
t = [0:dt:0.5-dt];

sa = sin(2*pi*525*t);
re = sin(2*pi*590*t);
ga = sin(2*pi*664*t);
ma = sin(2*pi*704*t);
pa = sin(2*pi*790*t);
da = sin(2*pi*885*t);
ni = sin(2*pi*995*t);
sa2 = sin(2*pi*1055*t);
```

```
y = [sa re ga ma pa da ni sa2];
wavwrite(y,Fs,"audio3.wav");
```

Observations:
1. For the audio generated using 1000 samples/second, all the notes are aliased. So it is difficult to identify them.
2. For the audio signal generated using 1600 samples/second, all the notes till pa (frequency = 790 Hz) can be identified but beyond that, the other notes are aliased.
3. For the audio signal generated using 2200 samples/second, all the notes can be easily identified as their frequencies are less than 1100 Hz. In other words, sampling rate of the signal is greater than Nyquist rate.

Q3)

Code -

```
# a = 4;
[x,Fs] = audioread('Track004.wav');

# SR = Fs/2
x2 = x(1:2:length(x));
# SR = Fs/3
x3 = x(1:3:length(x));
# SR = Fs/4
x4 = x(1:4:length(x));
# SR = Fs/5
x5 = x(1:5:length(x));
# SR = Fs/6
x6 = x(1:6:length(x));

audiowrite('PAudio2.wav',x2,Fs/2);
audiowrite('PAudio3.wav',x3,Fs/3);
audiowrite('PAudio4.wav',x4,Fs/4);
audiowrite('PAudio5.wav',x5,Fs/5);
audiowrite('PAudio6.wav',x6,Fs/6);
```

Observations:
It seen that with decreasing sampling rate of the audio signals, there is a decrease in the quality of the audio. This is because, the higher frequencies get aliased to lower frequencies.

Conclusions:
Q1) The original signal was sampled at different frequencies and the resulting aliases were identified using energy spectrum. Also the original signal was reconstructed using the signal obtained by sampling at Nyquist rate.

Q2) The eight notes in an octet were generated using different sampling rates and aliases were noted.

Q3) The audio signal quality decreases with decreasing sampling rate as long as the sampling rate is below Nyquist frequency.