Week 7 Tutorial 2

The purpose of this tutorial is to demonstrate plotting with multiple figures. There is also an example showing how to use fplot to see the benefits of anti-aliasing.

```
% Always clear workspace variables before a new tutorial or program.
clear
clc
close all  % closes all figure windows
```

Edit the code below and update the variable named **name** with your name for this tutorial in the code below.

```
name="";
fprintf("Output for Tutorial_07_2 run by %s.\n\n", name)
```

Figure 1

Fill in the missing x data to create the first plot figure. Notice the `figure(1)` command, this is telling all future code to apply plot related configurations to the "Figure 1" window.

```
% Define a vector x from 0 to 20 with increments of 5
x=;
% define vector y
y=10*cos(x);

figure(1)  % opens figure window 1
plot(x,y)  % Plot a graph of y vs x in Figure Window 1
xlabel('X')
ylabel('Y=10*cos(x)')
title('Extreme Aliasing:  Delta X = 5')
```

Figure 2

Fill in the missing x data to create the second plot figure. Notice the `figure(2)` command, this is telling all future code to apply plot related configurations to the "Figure 2" window.

```
% Define a vector x from 0 to 20 with increments of 1
x=;
y=10*cos(x);

figure(2)  % opens figure window 2
plot(x,y)  % Plot a graph of y vs x in Figure Window 2
xlabel('X')
ylabel('Y=10*cos(x)')
title('Moderate Aliasing:  Delta X = 1')
```

Figure 3

Fill in the missing x data to create the first plot figure.

```
% Define a vector x from 0 to 20 with increments of 0.5
```

```
x=;
y=10*cos(x);

figure(3)  % opens figure window 3
plot(x,y)  % Plot a graph of y vs x in Figure Window 3
xlabel('X')
ylabel('Y=10*cos(x)')
title('Minor Aliasing:  Delta X = 0.5')
```

Figure 4 Using fplot()

To avoid aliasing automatically, the `fplot` function should be used. `fplot` automatically adjusts the increment in x to avoid aliasing. Note that there is a prefex to the function @(x) which we have not seen before. This is referred to as a "function handle" and creates what is called an anonymous function. It's called anonymous because it's a function we are writing on the fly, there is no actual function or function file with `10*cos(x)` so we're telling MATLAB to create it for is in the background. To see more on this, refer to the function handle documentation.

```
figure(4)  % opens figure window 4

% Note that the function must be enclosed in apostrophes.
% [0 20] are [xmin xmax]
fplot(@(x)10*cos(x),[0 20])  % fplot of 10*cos(x) in Figure Window 4
xlabel('X')
ylabel('Y=10*cos(x)')
title('No Aliasing:  Function Plotted with fplot()')
```

Example Output:

Run this tutorial from the **Command Window** and ensure your output matches the following.

```
Output for Tutorial_07_2 run by Geoff Berl.
```

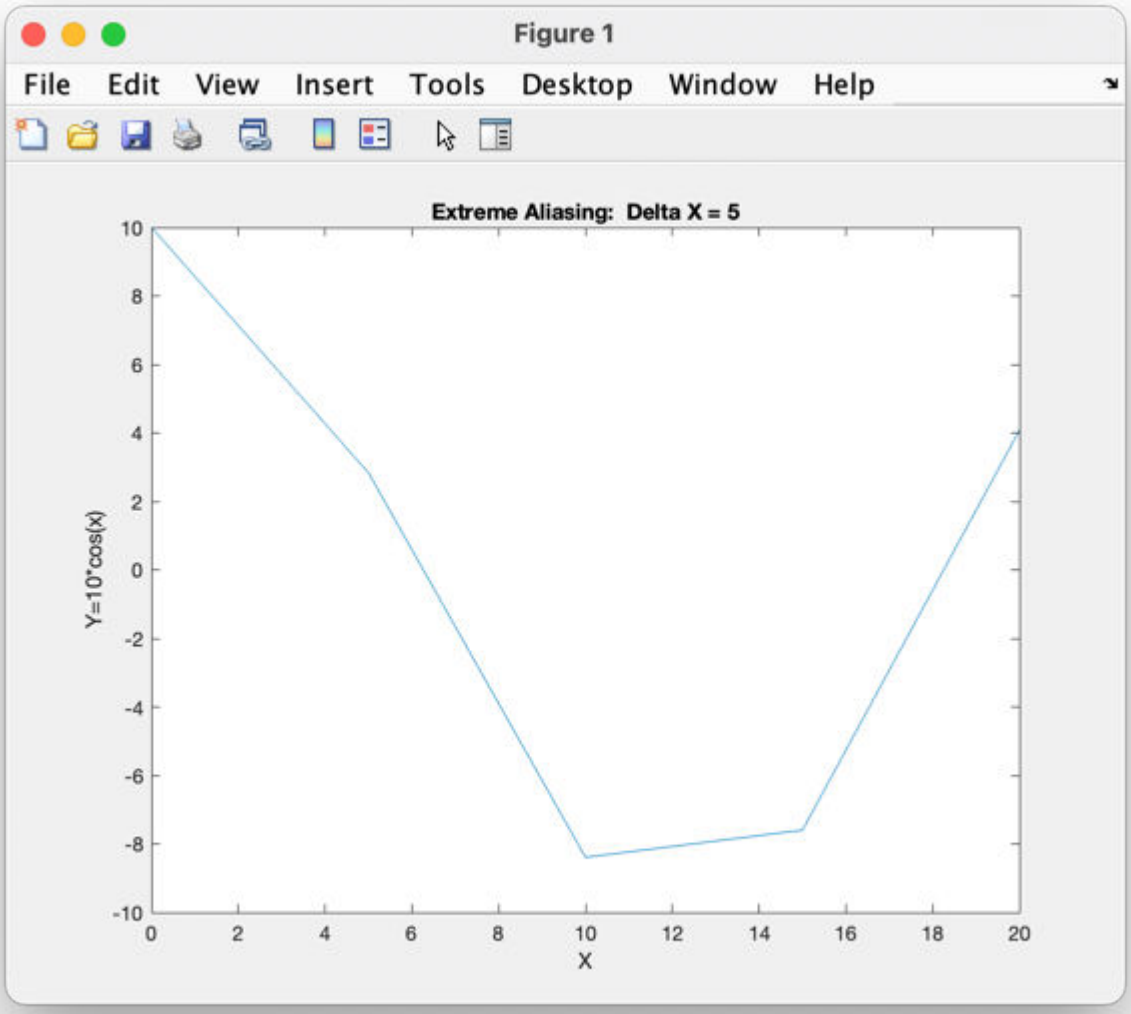
Figure 1
File Edit View Insert Tools Desktop Window Help
Extreme Aliasing: Delta X = 5
Y=10*cos(x)
X
10
8
6
4
2
0
-2
-4
-6
-8
-10
0
2
4
6
8
10
12
14
16
18
20

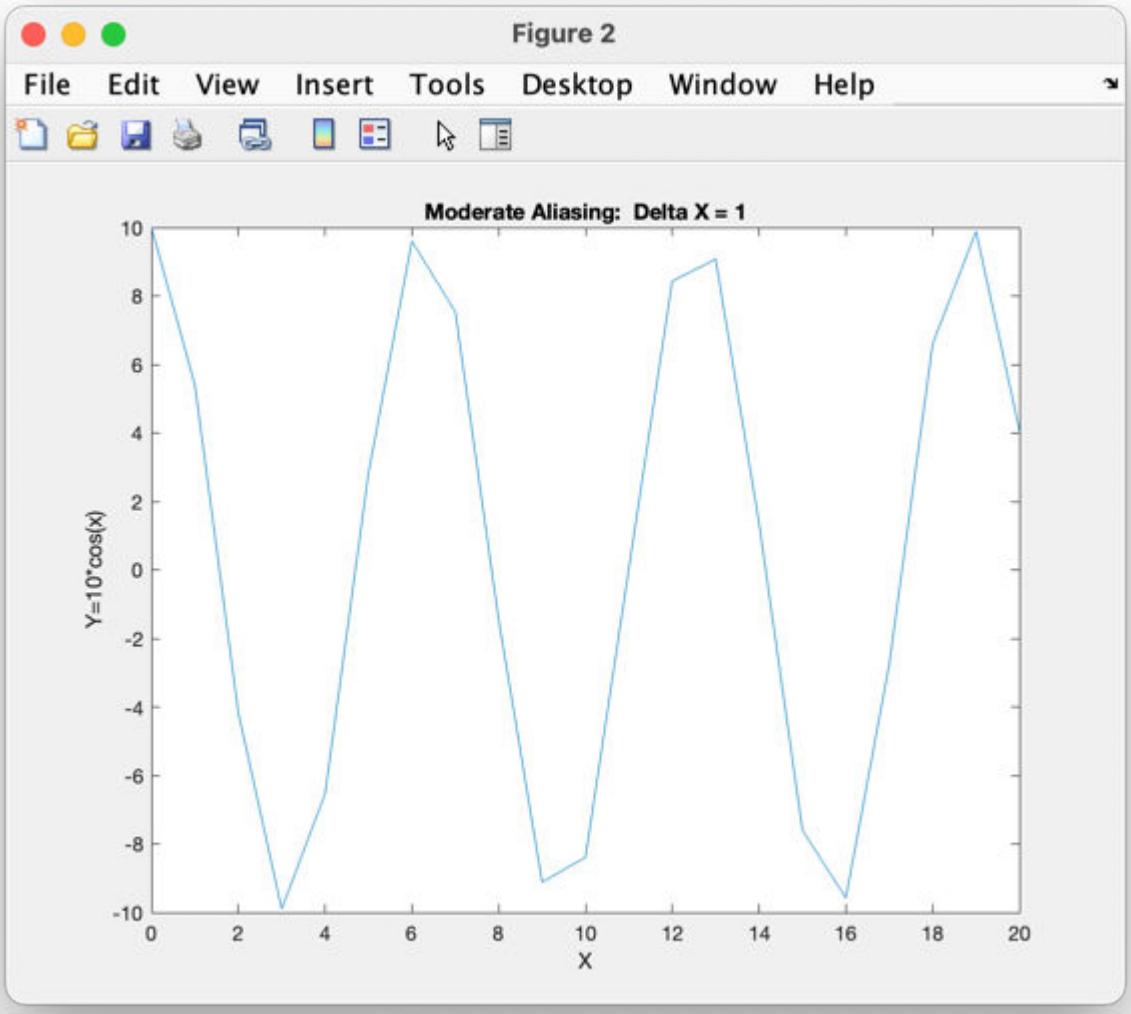
Figure 2
File Edit View Insert Tools Desktop Window Help
Moderate Aliasing: Delta X = 1
Y=10*cos(x)
X
10
8
6
4
2
0
-2
-4
-6
-8
-10
0 2 4 6 8 10 12 14 16 18 20

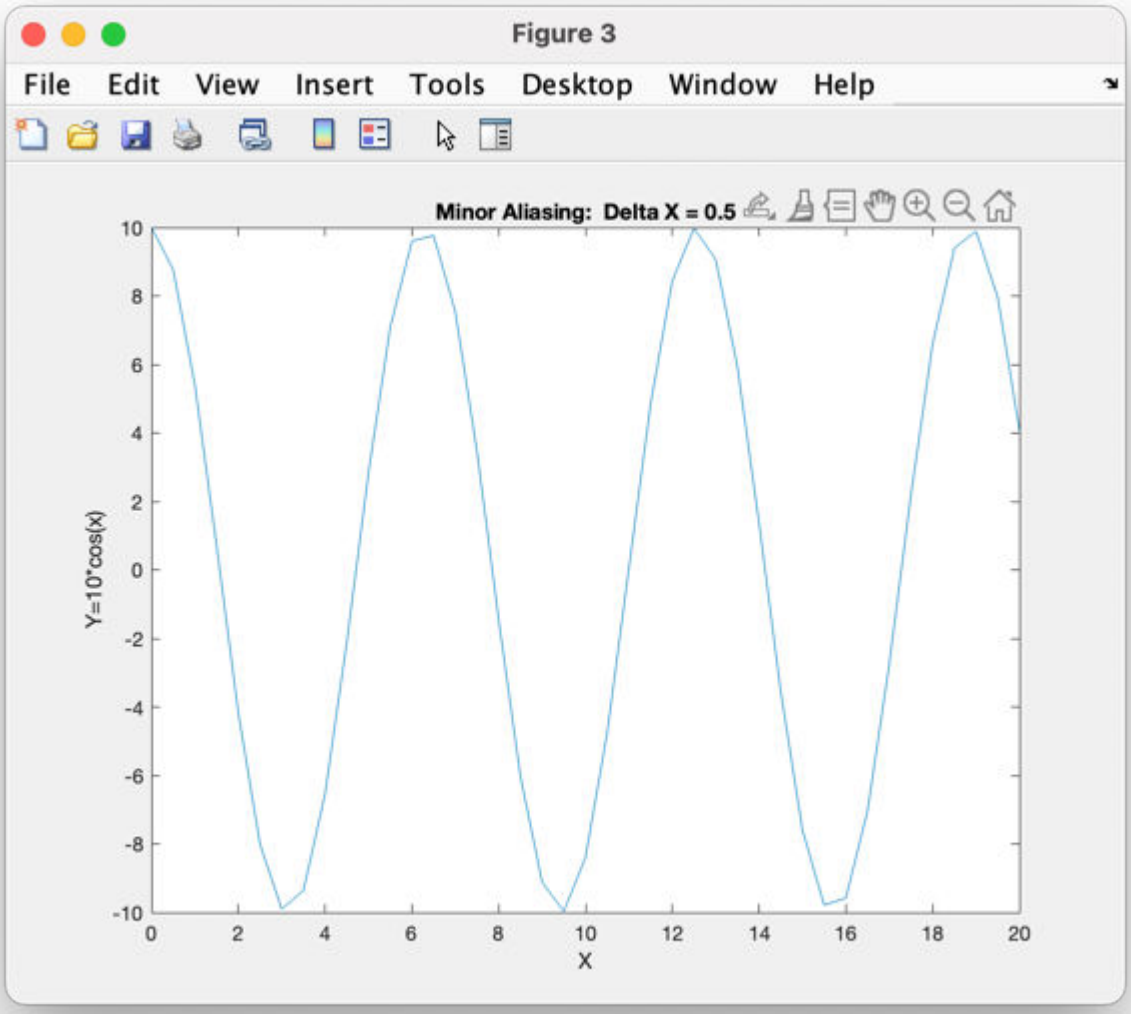
Figure 3
File Edit View Insert Tools Desktop Window Help
Minor Aliasing: Delta X = 0.5
Y=10*cos(x)
X
10
8
6
4
2
0
-2
-4
-6
-8
-10
0 2 4 6 8 10 12 14 16 18 20

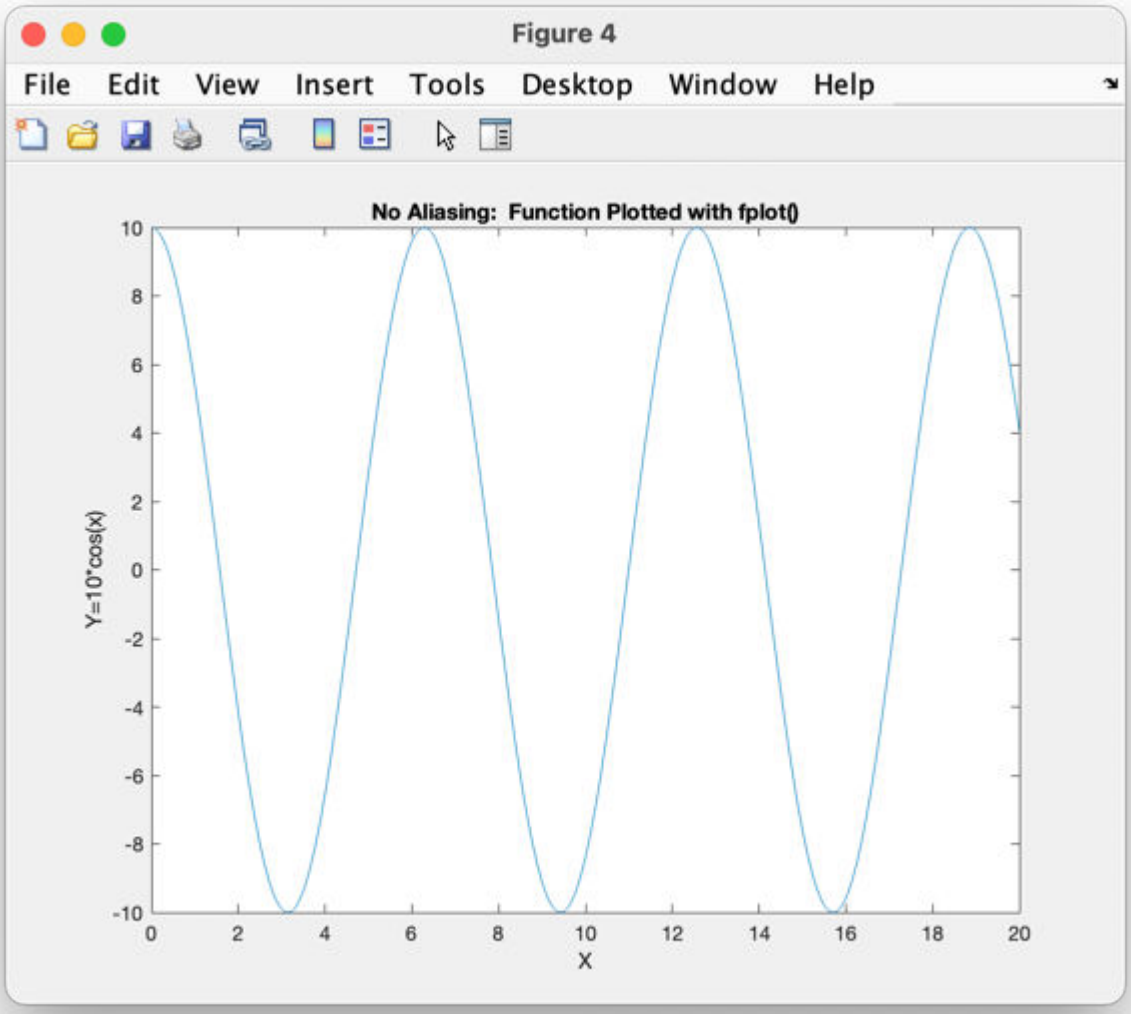
Figure 4
File Edit View Insert Tools Desktop Window Help
No Aliasing: Function Plotted with fplot()
Y=10*cos(x)
X
10
8
6
4
2
0
-2
-4
-6
-8
-10
0
2
4
6
8
10
12
14
16
18
20