Plotting Data in MATLAB

Graphics commands:

Command	Action
`plot(x,y,symbol)`	creates a pop up window that displays the (*x*,*y*) data points specified on linearly-scaled axes with the symbol (and color) specified in the *string variable* **`symbol`**. The data points are supplied as separate **`x`** and **`y`** vectors. MATLAB automatically scales the axes to fit the data.
`semilogy(x,y,symbol)`	creates a pop up window that displays the (*x*,*y*) data points specified on a graph with the *y*-axis scaled in powers of 10 and the *x*-axis scaled linearly with the symbol (and color) specified in the *string variable* **`symbol`**. The data points are supplied as separate **`x`** and **`y`** vectors. MATLAB automatically scales the axes to fit the data.
`loglog(x,y,symbol)`	creates a pop up window that displays the (*x*,*y*) data points specified on a graph with both the x- and *y*-axes scaled in powers of 10 with the symbol (and color) specified in the *string variable* **`symbol`**. The data points are supplied as separate **`x`** and **`y`** vectors. MATLAB automatically scales the axes to fit the data.
`xlabel(xname)`	adds the text in the *string variable* **`xname`** below the *x*-axis.
`ylabel(yname)`	adds the text in the *string variable* **`yname`** below the *y*-axis.
`title(graphname)`	adds the text in the *string variable* **`graphname`** above the plot.
`axis('equal')`	forces equal-scaling on the *x*- and *y*-axes

Graph symbol options:

Symbol Options			
Symbol Color		Symbol	
y	yellow	.	point
m	magenta	o	circle
c	cyan	x	x-mark
r	red	+	plus
g	green blue	*	star
b	blue	s	square
w	white	d	diamond
k	black	v	triangle (down)
		^	triangle (up)
		<	triangle (left)
		>	triangle (right)
		p	pentagram
		h	hexagram

Example:

```
x = -5:10; % values of the argument
y = x.^2 - 20; % values of the function

figure
plot(y)

figure
plot(x,y)

figure
plot(x,y,'k.-')
```

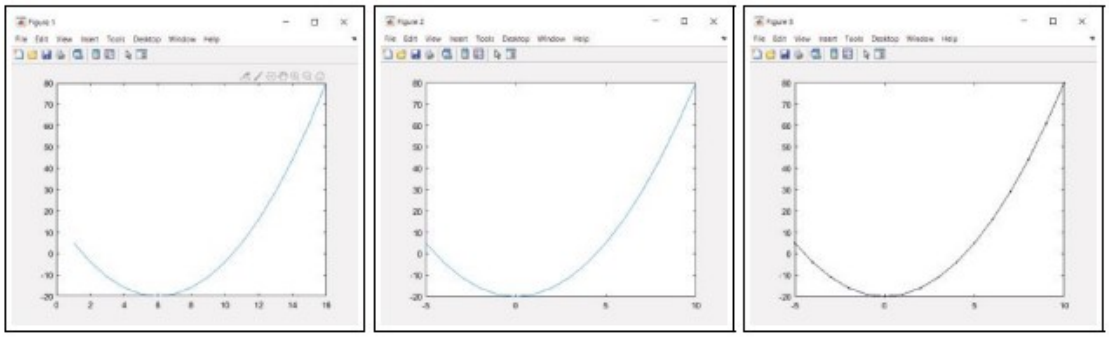
Plots only the function versus the index: 1,2,3...

Plots the function versus the argument: -5,-4,-3...

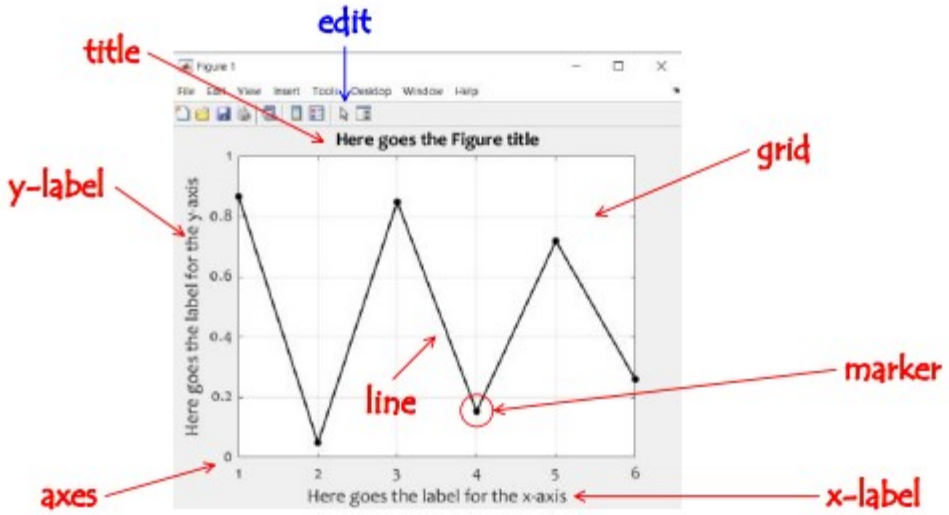
Plots the function versus the argument with a black line and a black dot marker

```
x=-5:10;
y=x.^2 - 20;
plot(y)
plot(x,y)
plot(x,y,'k.-')
```

Output Graph:



graph sample:



Data display:

The graphical display has several elements that are common to all graphical data displays:

• Data shown as discrete points

• Symbol (and color) used to display data

• Appropriate range and domain for data display

• x-axis label (with units)

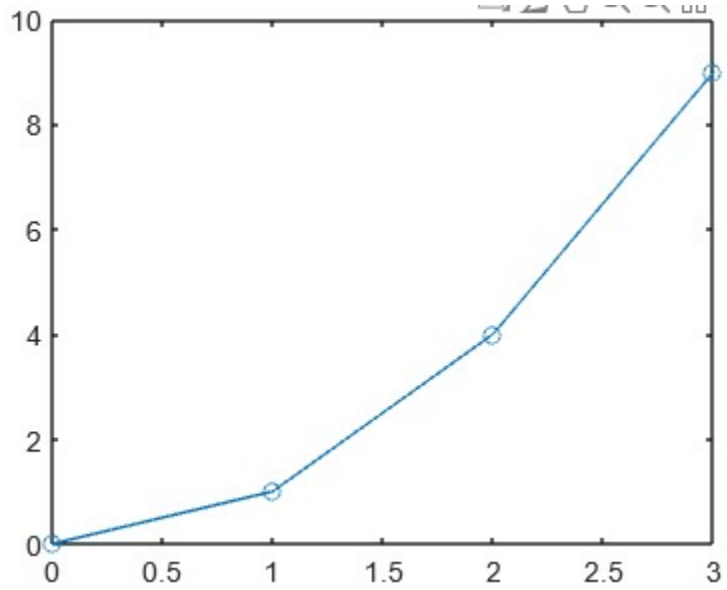
• y-axis label (with units)

• Graph title

Example1:

```
x = [0,1,2,3];
y = [0,1,4,9];
%If the x and y vectors are different lengths, you will get an error!
% plot(x,y) simple plot
plot(x,y,'o-')
```

OUTPUT:



Note: We could create the same x-vector by typing:

x = 0:1:3

That is, x is the vector obtained by starting at 0, stepping by 1, and ending at 3. Since we determined the step size is too large,

we can instead step by 0.1 using the following code:

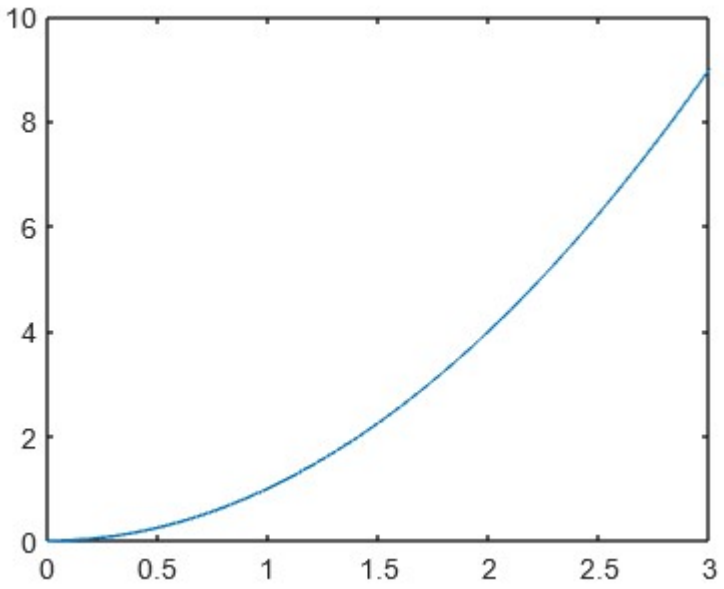
x = 0:0.1:3

Example2:

Plot the graph for $y = x^2$ on the interval [0, 3]

```
x = 0:0.1:3; % Don't forget the semicolon!
y = x.^2; % The period before the ^ indicates ELEMENT-WISE operation
plot(x,y)
```

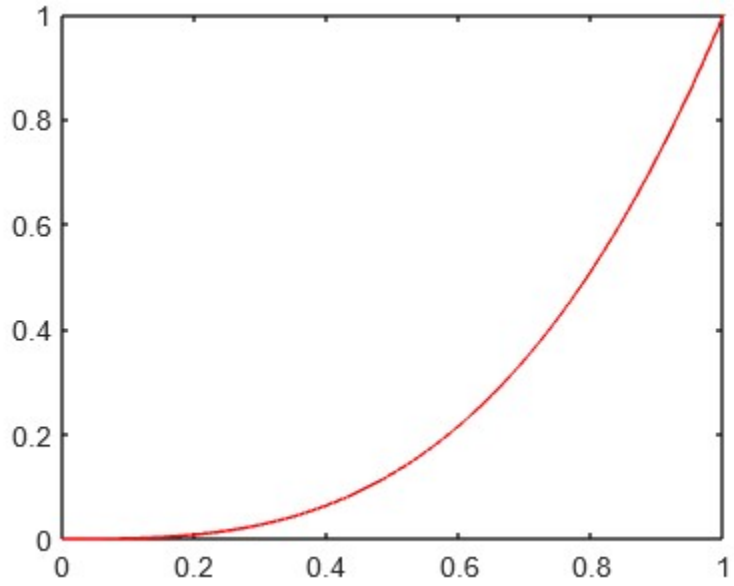
OUTPUT:



Multiple Plots on Single Axes:

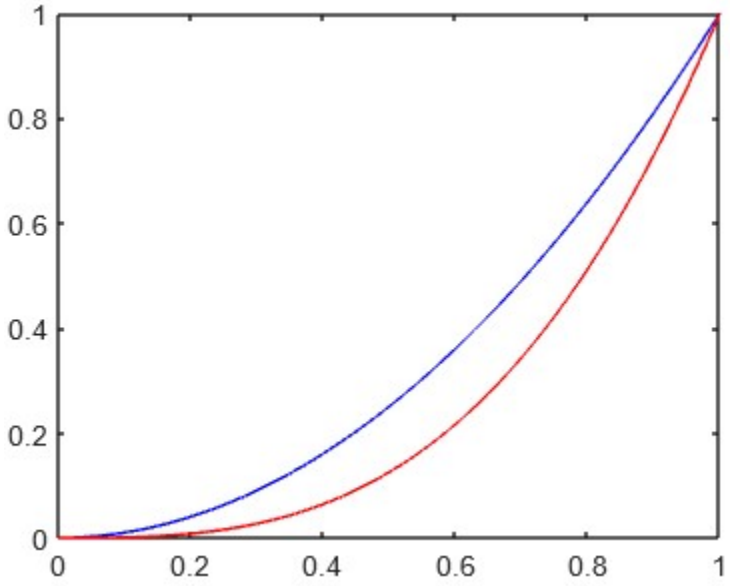
Example 1:

```
x = 0:0.01:1;
y1 = x.^2;
y2 = x.^3;
plot(x,y1,'blue')
plot(x,y2,'red')
```



Example 2:

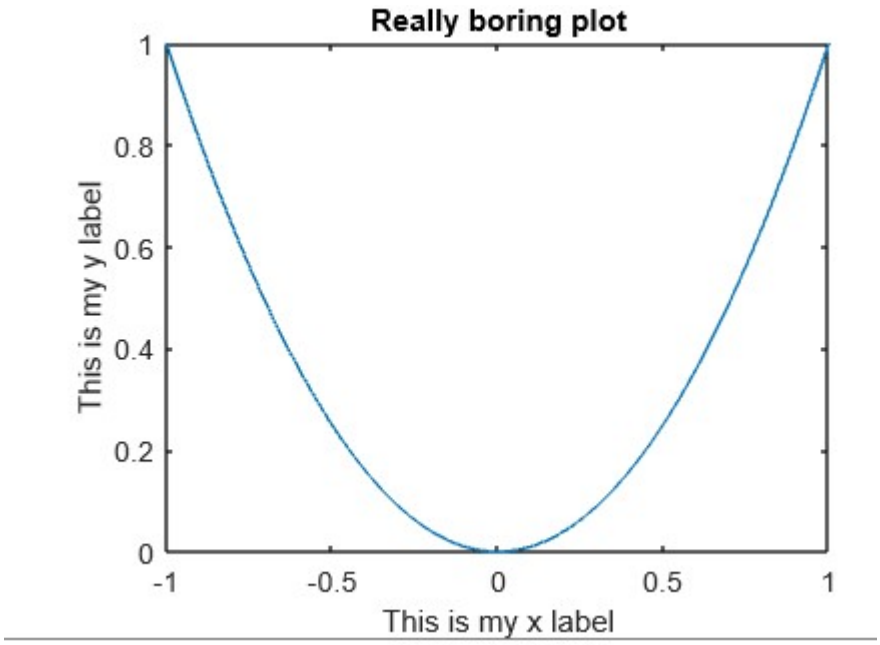
```
x = 0:0.01:1;
y1 = x.^2;
y2 = x.^3;
%plot(x,y1,'blue')
%plot(x,y2,'red')
%.............................................
hold on
plot(x,y1,'blue')
plot(x,y2,'red')
hold off
```



Making Plots Pretty:

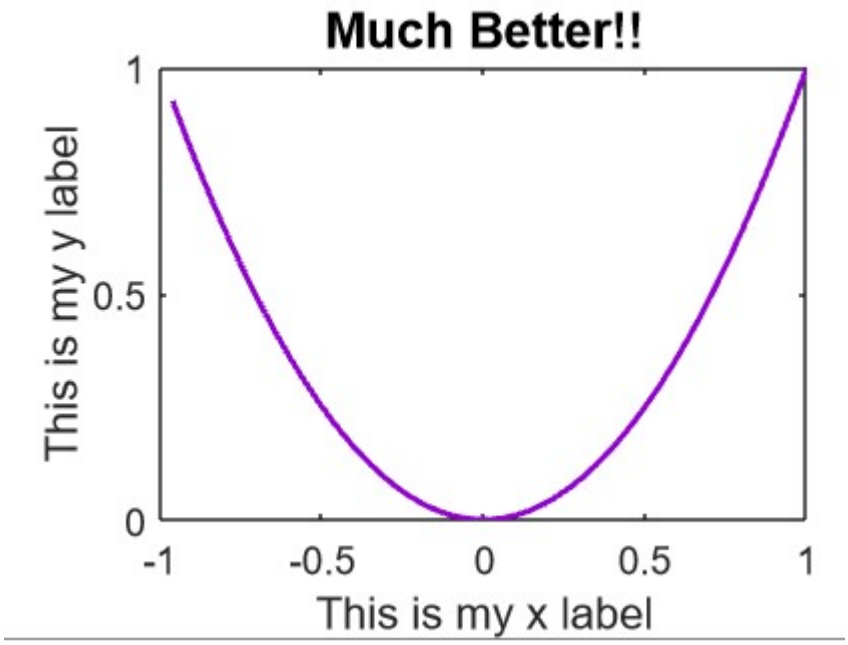
Example 1:

```
x = linspace(-1,1,50); % 50 evenly spaced points between -1 and 1
y = x.^2;
plot(x,y)
xlabel('This is my x label')
ylabel('This is my y label')
title('Really boring plot')
```



Example 2:

```
x = linspace(-1,1,50);
y = x.^2;
plot(x,y,'Color',1/255*[148 0 211],'LineWidth',2)
set(gca,'FontSize',14)
xlabel('This is my x label','FontSize',16)
ylabel('This is my y label','FontSize',16)
title('Much Better!!','FontSize',18)
```



Fill Colour:

fill(x,y,colour)

x is array of x-co-ordinate

y is array of y-co-ordinate

colour is one of the following

'k' black 'r' red 'g' green 'b' blue 'w' white 'm' magenta 'y' yellow 'c' cyan

Example:

```
figure, hold on
t = linspace(0,2*pi,50); % 50 angles from 0 to 2*pi (in radians)

fill(sin(t),cos(t),[0.9137, 0.5882, 0.4784]) % circle centred at [0,0]
% with radius 1, filled with dark salmon colour.

fill(0.5*sin(t)+1.5,0.5*cos(t),[1 0.9412 0.9608])
% circle centred at [1.5,0] with radius 0.5, filled with lavender bush
% salmon colour.

fill(0.2*sin(t)+2.2,0.2*cos(t),[0.5020 0 0])
% circle centred at [2.2,0] with radius 0.2, filled with maroon colour.

grid on
axis equal
```

Output: