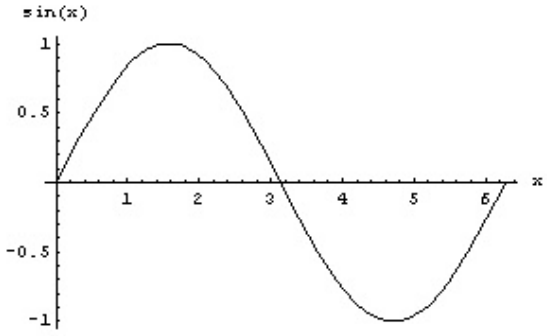
Mathematica (5)

Mathematica has a number of plotting options. To label axes, use the AxesLabel command and to label the plot use the PlotLabel command. The syntax of the AxesLabel command can be seen in the following example. The command "points to" (dash, greater than sign) a list of quoted quantities. The first element in this list is the x quantity and the second is y quantity. (If there is only one element in the list, it is assumed to be the y quantity.)

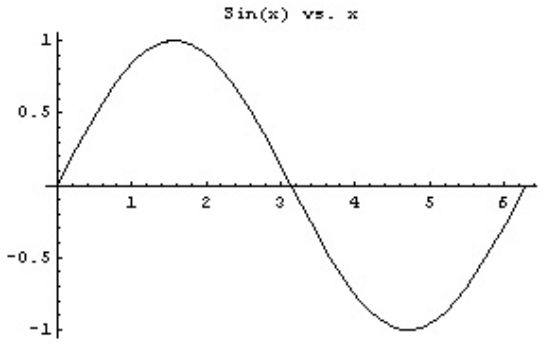
To label the plot, use the PlotLabel command. It points to the quoted value of the plot label. (Note that these commands may be combined in a single Plot[] command.)

```
Plot[Sin[x], {x, 0, 2 Pi}, AxesLabel → {"x", "sin(x)"}]
```



- Graphics -

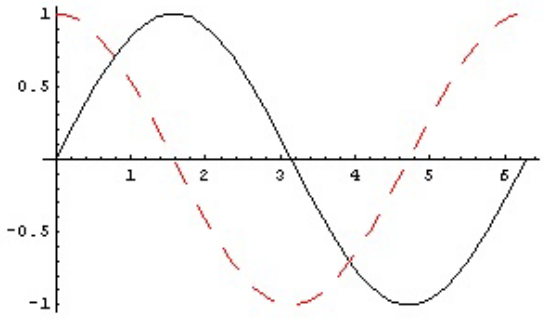
```
Plot[Sin[x], {x, 0, 2 Pi}, PlotLabel → "Sin(x) vs. x"]
```



- Graphics -

Mathematica can also draw multiple curves on the same plot and give them different colors or line characteristics. This is done with the PlotStyle command illustrated below:

```
Plot[{Sin[x], Cos[x]}, {x, 0, 2 Pi},
 PlotStyle → {{Dashing[{}]}, {RGBColor[1, 0, 0], Dashing[{0.05, 0.05}]}}]
```




```
- Graphics -
```

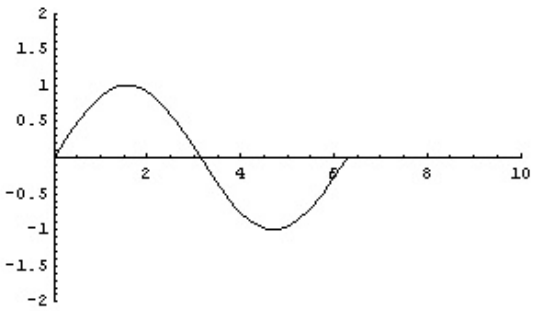
This command points to a list of two lists. The first list gives the characteristics of the first curve, and the second list gives the characteristics of the second curve. (Note that a list can have one or more elements.)

The Dashing[ ] command uses lists of values to determine how to generated a dashed curve. If the list is empty {}, the curve is a continuous line. If the list has two quantities, {0.05, 0.05} the curve will be drawn with equal segments on and off (blank parts). If the list has four quantities {0.01, 0.05, 0.05, 0.05} the curve will be drawn with a short on part, an off part, an on part equal to the off part, and another off part, i.e. a dot – dashed curve.

The RGBColor command determines the color of the curve. One can use different numerical values of Red, Green, and Blue between 0 and 1 to change the color of the curve. In this case, the second curve is pure red.

One can also change the x and y ranges on a plot. The following example shows how to do this:

```
Plot[Sin[x], {x, 0, 2 Pi}, PlotRange → {{0, 10}, {-2, 2}}]
```



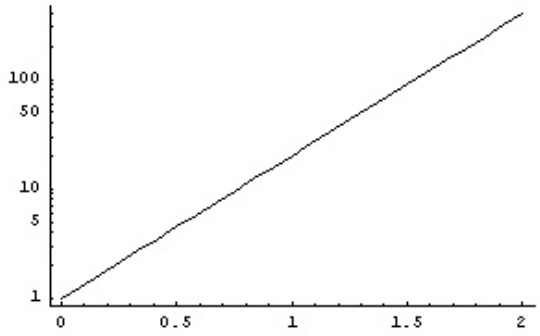

```
- Graphics -
```

Here the PlotRange command is used and points to a list of two lists. The first list contains the beginning and ending values of x, and the second list contains the beginning and ending values of y. If PlotRange points to a single list of two values, these are assumed to be the beginning and ending values of y.

One can also use Mathematica to generate semilog plots (Log plots) or log-log plots. To do this, you need to load the graphics package with the command <<Graphics`

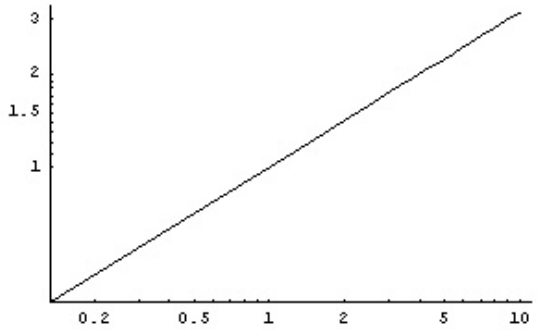
```
<< Graphics`

LogPlot[Exp[3 x], {x, 0, 2}]
```




```
- Graphics -

LogLogPlot[x^(0.5), {x, 0, 10}]
```




```
- Graphics -
```

These plotting command have options that are similar to those in the basic Plot[] command.