First you can add text to any field either by declaring to be text (use the menu above) or by placing it between (* text text *)

Input press [shift]+[return]

```
In[•]:= 2 + 2
Out[•]= 4
```

Notice the In[] and Out[] labels. these can be used to refer back to these items. % refers back to the last output

```
In[•]:= % + 5
Out[•]= 9
```

Standard symbols work for multiplication, subtraction and division.

```
In[•]:= 5 + 4 * 5 - 4 / 6
```

Out[•]= $\frac{73}{3}$

Power is indicated by ^ symbol.

```
In[•]:= (5 - 3) ^ 2 / 3
```

Out[•]= $\frac{4}{3}$

There are a vast number of functions built in. Some of them are usual, and some require multiple inputs. Functions are invoked by [].

```
In[•]:= Min[3, 4]
Out[•]= 3

In[•]:= GCD[24, 28]
Out[•]= 4
```

Some irrational numbers are represented by symbols.

```
In[•]:= Sin[Pi / 2]
Out[•]= 1

In[•]:= Exp[1]
Out[•]= ⅇ
```

Be careful with brackets. Mathematica helps you by highlighting the related brackets.

```
In[•]:= (1 + Sqrt[5]) / (1 + 1)
```

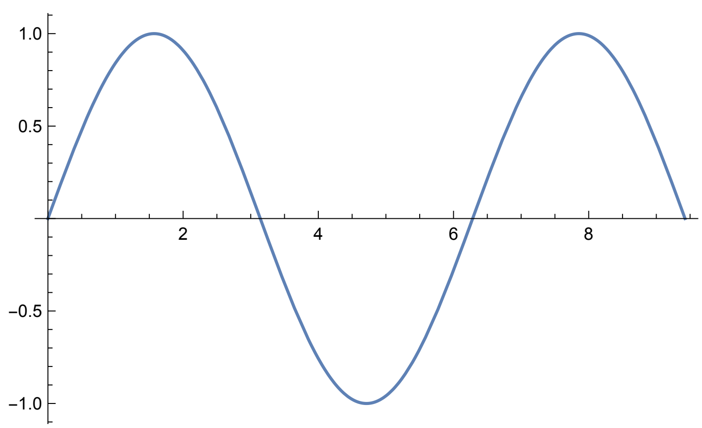
Out[•]= $\frac{1}{2}\left(1+\sqrt{5}\right)$

```
In[•]:= (1 + Sqrt[5] / 1 + 1)
```

Out[•]= $2+\sqrt{5}$

```
In[•]:= Plot[Sin[x], {x, 0, 3 Pi}]
```

Out[•]=



Data is represented in lists indicated by {...}. There can be lists of lists. They can contain numbers, variables, text, and even functions. There are many operations on lists.

```
In[]:= {1, 2, 3}
Out[]= {1, 2, 3}

In[]:= {1, 2, 3} + 4
Out[]= {5, 6, 7}
```

Get an element of a list. use [[ ]]

```
In[]:= {1, 2, 3}[[2]]
Out[]= 2
```

construct lists with simple commands

```
In[]:= Range[10]
Out[]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

In[]:= Table[1/i, {i, 1, 20}]
```

Out[]= $\left\{1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}, \frac{1}{7}, \frac{1}{8}, \frac{1}{9}, \frac{1}{10}, \frac{1}{11}, \frac{1}{12}, \frac{1}{13}, \frac{1}{14}, \frac{1}{15}, \frac{1}{16}, \frac{1}{17}, \frac{1}{18}, \frac{1}{19}, \frac{1}{20}\right\}$

Some rules with numbers. Exact input gets exact output. Decimal input will get approximate output.

In[]:= $\frac{1}{3} + \frac{2}{6}$

Out[]= $\frac{2}{3}$

```
In[]:= 0.3333 + 1/3
Out[]= 0.666633
```

N[..] function will convert exact output to approximate.
ScientificForm[..] will convert to scientific notation output.

In[∘]:= `N[1/3+1/2]`

Out[∘]= `0.833333`

In[∘]:= `ScientificForm[0.00082712/10132.0898]`

Out[∘]//ScientificForm=

8.16337×10^{-8}

You can use variables to represent almost anything. Usually lower case letters are used as variables. Upper cases for functions and built in constants.

In[∘]:= `(x+2)/5`

Out[∘]= $\frac{2+x}{5}$

Use space or * to indicate multiplication

In[∘]:= `(x y+55*x+14)/y^3`

Out[∘]= $\frac{14+55\,x+x\,y}{y^3}$

use /. and -> to make substitutions on the fly.

In[∘]:= `(x y+55*x+14)/y^3 /. {y→3, x→4}`

Out[∘]= $\frac{82}{9}$

variables can be assigned with = . Use ; to have multiple statements in one executation.

In[∘]:= `x = 4; y = 5; (x+y)/4`

Out[∘]= $\frac{9}{4}$

use Clear[] to clear assignment

```
In[•]:= Clear[x]; (x + y) / 4
```

Out[•]= $\frac{5+x}{4}$

Use := to create definitions for custom functions. Notice the x_ which means x is a pattern to be substituted. := means that arguments that are passed to f get substituted on the right.

```
In[•]:= f[x_] := Sin[x] * Exp[x];
        f[y]
```

Out[•]= e^5 Sin[5]

A little bit of Algebra.

First get used to the idea of the three different “equal” signs.

= means assignment. Right side is assigned to the symbol on the left
:= means a definition of a function to which arguments are passed
== really means a equal sign that is used in equations. The entire equation becomes an expression that either true or false.

Factor an equation, simplify, or make partial fractions separation.

```
In[•]:= Factor[x^2 + 2 x + 1]
```

Out[•]= $(1+x)^2$

```
In[•]:= Simplify[x^2 (1 - y^2) * x / (x^2 + 2 x + 1)]
```

Out[•]= $-\frac{24 x^3}{(1+x)^2}$

```
In[•]:= Apart[(x^2 - 1) / (x^2 + 2 x + 1)]
```

Out[•]= $1 - \frac{2}{1+x}$

```
In[•]:= 2 + 2 == 4
```

Out[•]= True

This is an equation

In[•]:=
```
1 + z == 5
```
Out[•]= $1+z==5$

How do we solve quadratic equations etc. Solve produces answers in the form of substitution rules. Nsolve gives numerical answers in case the answer is difficult to get.

In[•]:=
```
Solve[x^2 + 6 x - 6 == 0, x]
```
Out[•]= $\{\{x\to -3-\sqrt{15}\},\{x\to -3+\sqrt{15}\}\}$

In[•]:=
```
NSolve[x^3 + 6 x - 6 == 0, x]
```
Out[•]= $\{\{x\to -0.442311-2.5665\,i\},\{x\to -0.442311+2.5665\,i\},\{x\to 0.884622\}\}$

How about system of equations !

In[•]:=
```
Solve[{x^2 + z == 0, 8 x - 5 == z}, {x, z}]
```
Out[•]= $\{\{x\to -4-\sqrt{21},\ z\to -37-8\sqrt{21}\},\{x\to -4+\sqrt{21},\ z\to -37+8\sqrt{21}\}\}$

How about plotting functions ? The most basic command is plot. Plot has a huge number of options for making fancy looking pictures or overlapping plots.

In[•]:=
```
? Plot
```

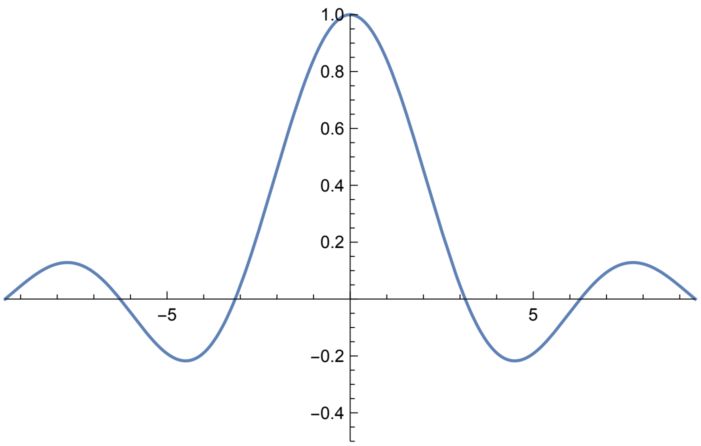
Plot[f, {x, x_{min}, x_{max}}] generates a plot of f as a function of x from x_{min} to x_{max}.
Plot[{f_1, f_2, …}, {x, x_{min}, x_{max}}] plots several functions f_i.
Plot[{…, $w[f_i]$, …}, …] plots f_i with features defined by the symbolic wrapper w.
Plot[…, {x} ∈ reg] takes the variable x to be in the geometric region reg. ≫

In[•]:=
```
? RegionPlot
```

RegionPlot[$pred$, {x, x_{min}, x_{max}}, {y, y_{min}, y_{max}}] makes a plot showing the region in which $pred$ is True. ≫

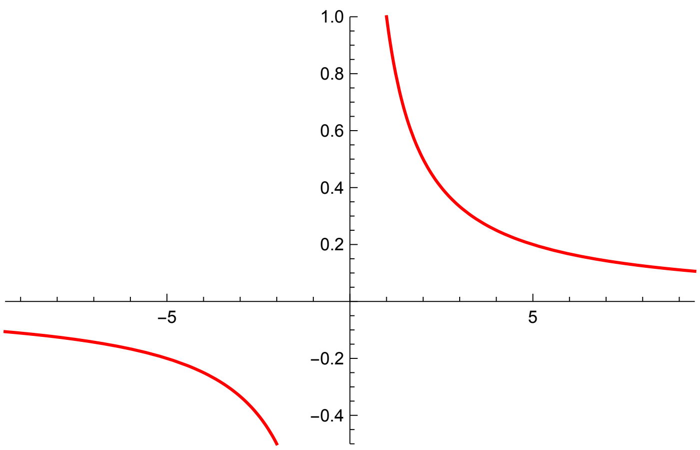
In[•]:=
```
f1 = Plot[Sin[x] / x, {x, -3 Pi, 3 Pi}, PlotRange → {{-3 Pi, 3 Pi}, {-0.5, 1.0}}]
```

Out[•]=



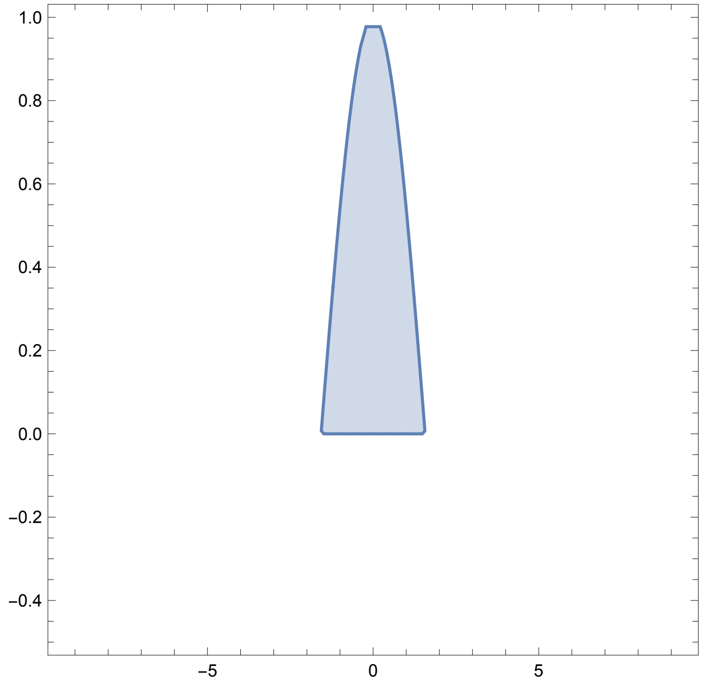
In[•]:=
```
f2 = Plot[1 / x, {x, -3 Pi, 3 Pi},
  PlotStyle → Red, PlotRange → {{-3 Pi, 3 Pi}, {-0.5, 1.0}}]
```

Out[•]=



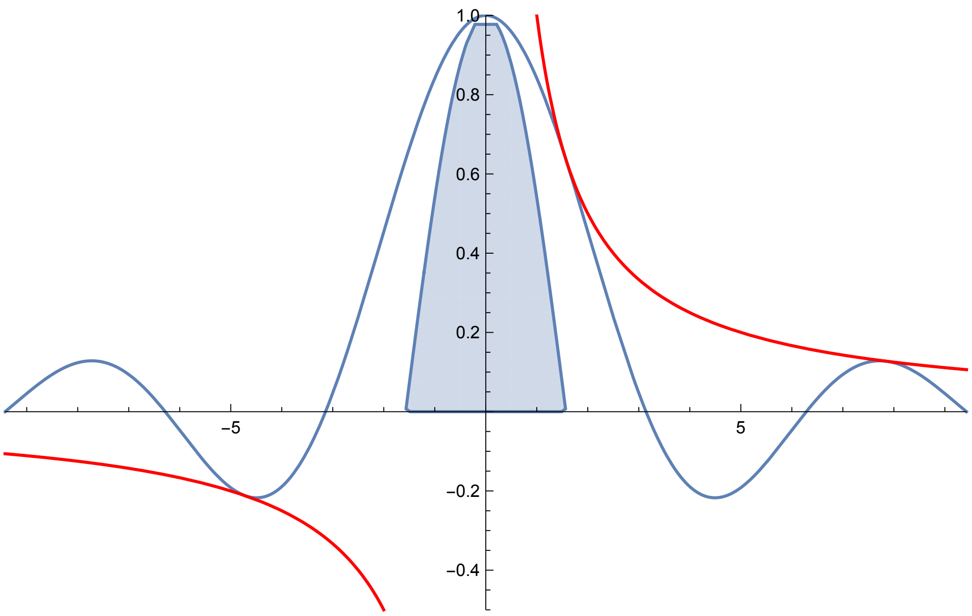
In[•]:=
```
f3 = RegionPlot[Abs[y] < Cos[x] && Abs[x] < Pi && y > 0, {x, -3 Pi, 3 Pi}, {y, -0.5, 1.}]
```

Out[•]=



In[•]:=
```
Show[{f1, f2, f3}]
```

Out[•]=



We will do more with plotting over the next several units. There is almost infinite flexibility and you can also write manually on the plots if you want to.