Mathematica for Mathematics

Omar Alsuhaimi
01/01/2018

A Quick Tour

Calculator

In[1]:=
```
23 + 10
```
Out[1]=
```
33
```

In[2]:=
```
13 * 50
```
Out[2]=
```
650
```

In[3]:=
```
Sin[0.5]
```
Out[3]=
```
0.479426
```

In[4]:= $e^{0.1}$

Out[4]=
```
1.10517
```

In[5]:=
```
6 !
```
Out[5]=
```
720
```

In[6]:= 5^3

Out[6]=
```
125
```

In[7]:= 5^3

Out[7]=
```
125
```

Solving Equations and Inequalities

In[8]:=

```
Solve[x^2 + 3 x - 4 == 0, x]
```

Out[8]=

```
{{x → -4}, {x → 1}}
```

Find a solution to $\cos(x) = x$ near $x = 0$:

In[9]:=

```
FindRoot[Cos[x] == x, {x, 0}]
```

Out[9]=

```
{x → 0.739085}
```

In[10]:=

```
Reduce[(x - 1) (x - 2) (x - 3) (x - 4) > 0, x]
```

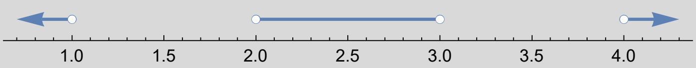
Out[10]=

```
x < 1 || 2 < x < 3 || x > 4
```

In[11]:=

```
NumberLinePlot[(x - 1) (x - 2) (x - 3) (x - 4) > 0, x]
```

Out[11]=



Computer Algebra

Expand polynomial expressions:

In[12]:=

```
Expand[(1 + x)^4]
```

Out[12]=

```
1 + 4 x + 6 x^2 + 4 x^3 + x^4
```

In[13]:=

```
Expand[(1 + x + y) (2 - x)^3]
```

Out[13]=

```
8 - 4 x - 6 x^2 + 5 x^3 - x^4 + 8 y - 12 x y + 6 x^2 y - x^3 y
```

Factor polynomials:

In[14]:=

```
Factor[1 + 2 x + x^2]
```

Out[14]=

```
(1 + x)^2
```

In[15]:=

```
Factor[8 - 4 x - 6 x^2 + 5 x^3 - x^4 + 8 y - 12 x y + 6 x^2 y - x^3 y]
```

Out[15]=

```
-(-2 + x)^3 (1 + x + y)
```

In[16]:=

```
Simplify[Sin[x]^2 + Cos[x]^2]
```

Out[16]=

```
1
```

In[17]:=
```
Cancel[(1 + x)/(1 - x^2)]
```
Out[17]=
$$\frac{1}{1-x}$$

Calculus

We can find the limiting value of *expression* when x approaches x_0.

$$\lim_{x\to 0} \frac{\sin(x)}{x}$$

In[18]:=
```
Limit[Sin[x]/x, x -> 0]
```
Out[18]=
```
1
```

In[19]:=
```
Limit[(1 - x)/(1 - x^2), x → 1]
```
Out[19]=
$$\frac{1}{2}$$

We can find the derivative of a function $2x^3 + 3x^2 - 5x$ with:

In[20]:=
```
D[2 x^3 + 3 x^2 - 5 x, x]
```
Out[20]=
$$-5 + 6x + 6x^2$$

We can find the indefinite integral $\int x \sin(x)\, dx$ with:

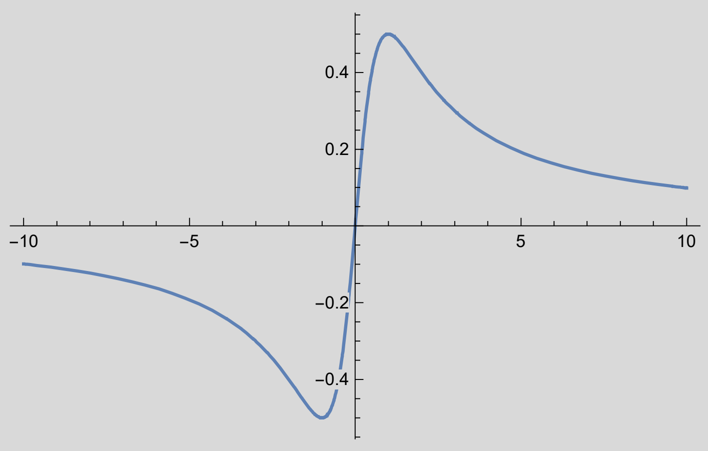
In[21]:=
```
Integrate[x Sin[x], x]
```
Out[21]=
```
-x Cos[x] + Sin[x]
```

Graphing in Plane

In[22]:=

$$\text{Plot}\left[\frac{x}{1+x^2}, \{x, -10, 10\}\right]$$

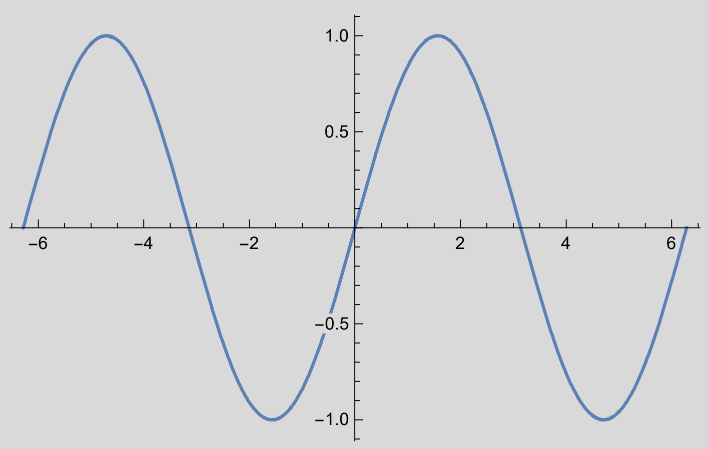
Out[22]=



In[23]:=

```
Plot[Sin[x], {x, -2 π, 2 π}]
```

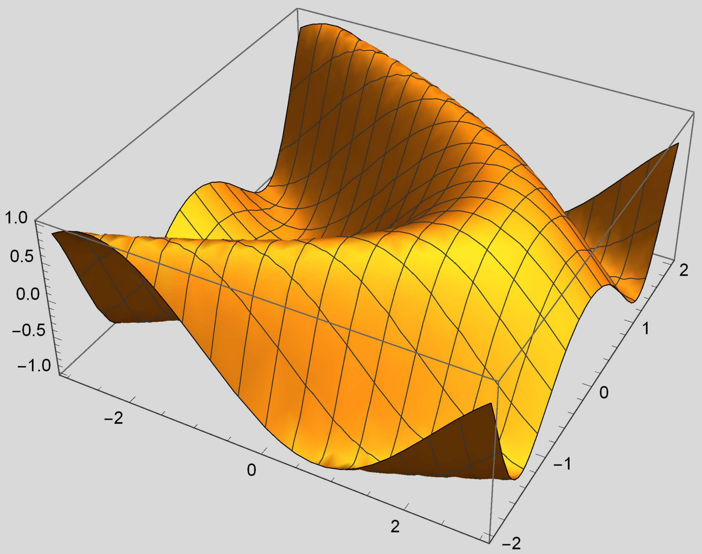
Out[23]=



Plotting in Space

In[24]:=
```
Plot3D[Sin[x + y^2], {x, -3, 3}, {y, -2, 2}]
```

Out[24]=



Entering Input

In a Wolfram notebook on the desktop , just type an input, then press (**SHIFT+ENTER** for Win, **SHIFT+Return** for Mac) to compute:

In[25]:=
```
2 + 2
```
Out[25]=
```
4
```

In[26]:=
```
100 - 40
```
Out[26]=
```
60
```

In[27]:=
```
1 + 2 + 3
```
Out[27]=
```
6
```

In[28]:=
```
5 + 2 * 3 - 7.5
```
Out[28]=
```
3.5
```

x^y or x^y gives x to the power y.

In[29]:=
```
((5 - 3) ^ (1 + 2)) / 4
```
Out[29]=
```
2
```

In[30]:= $\frac{(5-3)^{(1+2)}}{4}$

Out[30]= 2

In[31]:= (5 - 3) ^ ((1 + 2) / 4)

Out[31]= $2^{3/4}$

In[32]:= (1 + 4) (2 + 3)

Out[32]= 25

In[33]:= $\sqrt{9}$

Out[33]= 3

In[34]:= $\sqrt{16}$

Out[34]= 4

In[35]:= Sqrt[9]

Out[35]= 3

In[36]:= $\sqrt{12}$

Out[36]= $2\sqrt{3}$

In[37]:= Sqrt[12]

Out[37]= $2\sqrt{3}$

In[38]:= $N\left[\sqrt{12}\right]$

Out[38]= 3.4641

In[39]:= $\sqrt{12}$ // N

Out[39]= 3.4641

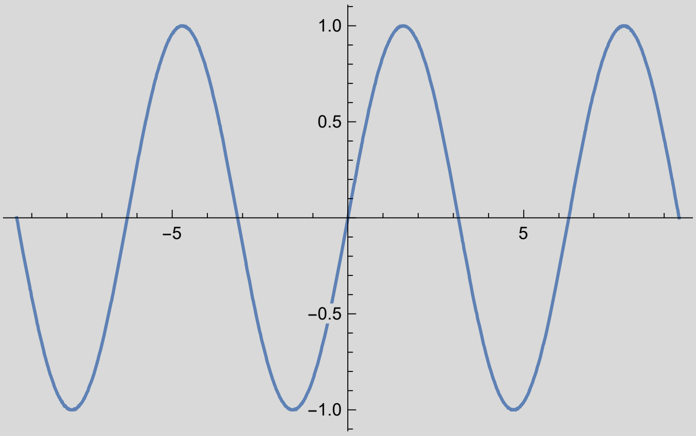
In[40]:= $\sqrt{2}+\sqrt{8}+\sqrt{18}$

Out[40]= $6\sqrt{2}$

In[41]:=
```
√-1
```
Out[41]=
```
ⅈ
```

In[42]:=
```
GCD[12, 15]
```
Out[42]=
```
3
```

In[43]:=
```
Plot[Sin[x], {x, -3 π, 3 π}]
```
Out[43]=



In[44]:=
```
{a, b, c, d}[[2]]
```
Out[44]=
```
b
```

In[45]:=
```
Range[10]
```
Out[45]=
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

Fractions & Decimals

In the Wolfram Language, exact input (like fractions) will provide exact output:

(Use **CTRL+ /** to enter fractions.)

In[46]:=

$$\frac{1}{3} + \frac{3}{5}$$

Out[46]=

$$\frac{14}{15}$$

In[47]:= $\frac{4}{6}$

Out[47]= $\frac{2}{3}$

In[48]:= $\frac{7}{8}$

Out[48]= $\frac{7}{8}$

Put fractions over their lowest common denominator with **Together**:

In[49]:=
$$\frac{1}{a}+\frac{1}{b}$$

Out[49]=
$$\frac{1}{a}+\frac{1}{b}$$

In[50]:=
$$\text{Together}\left[\frac{1}{a}+\frac{1}{b}\right]$$

Out[50]=
$$\frac{a+b}{a\,b}$$

Use N to get a numerical approximation of a result:

In[51]:=
$$N\left[\frac{1}{4}+\frac{1}{7}\right]$$

Out[51]=
```
0.392857
```

In[52]:=
$$\frac{1}{4}+\frac{1}{7}\;//\;N$$

Out[52]=
```
0.392857
```

Specify the accuracy to which your answer is displayed:

In[53]:=
$$N\left[\frac{1}{4}+\frac{1}{7},\,10\right]$$

Out[53]=
```
0.3928571429
```

Some numbers are better expressed in **ScientificForm**:

In[54]:=
```
ScientificForm[0.00039285]
```

Out[54]//ScientificForm=
$$3.9285\times 10^{-4}$$

How to Use Brackets and Braces Correctly in Mthematica

Parentheses ( ),**braces { }**, and **square brackets [ ]** all have different meanings in the Wolfram Language. The first two are sometimes called round brackets and curly brackets.

- **Parentheses ( )**
  You use parentheses ( ) in the Wolfram Language for grouping expressions and to determine the precedence of operations:

In[55]:=
```
1 + 2 / 3
```
Out[55]=

$\frac{5}{3}$

In[56]:=
```
(1 + 2) / 3
```
Out[56]=
```
1
```

In[57]:=
```
(x + 3) (y + 2)
```
Out[57]=
```
(3 + x) (2 + y)
```

- **braces { } :**
  A list in the Wolfram Language is represented by braces { } and is a collection of items referred to as elements.
  Create a list of the first five positive integers:

In[58]:=
```
{1, 2, 3, 4, 5}
```
Out[58]=
```
{1, 2, 3, 4, 5}
```

Anything in the Wolfram Language can be used in lists, including numbers, variables, typeset mathematical expressions, and strings:

In[59]:=
```
{1, b, 2, 3, 3 x == 12, Sqrt[9 + y], "hello"}
```
Out[59]=

$\left\{1, b, 2, 3, 3x == 12, \sqrt{9+y}, hello\right\}$

A function range:

In[60]:=
```
{x, 0, π}
```
Out[60]=
```
{x, 0, π}
```

A system of equations:

In[61]:=
```
Eqn = {x + y == 1, x - 2 y == 8}
```
Out[61]=
```
{x + y == 1, x - 2 y == 8}
```

- **Square Brackets [ ]**
  Square brackets are used in the Wolfram Language to enclose the arguments of functions.
  The functions Range, Sin, and N are used here with square brackets enclosing their arguments:

In[62]:=
```
Range[10]
```
Out[62]=
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

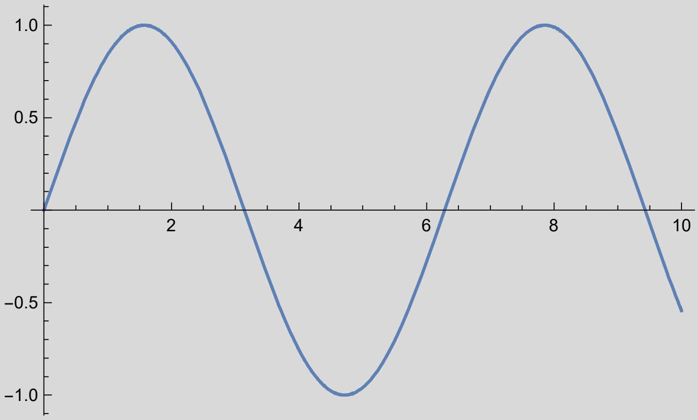
In[63]:=
```
Range[10]
```
Out[63]=
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In[64]:=
```
Sin[2]
```
Out[64]=
```
Sin[2]
```

In[65]:=
```
N[Sin[2]]
```
Out[65]=
```
0.909297
```

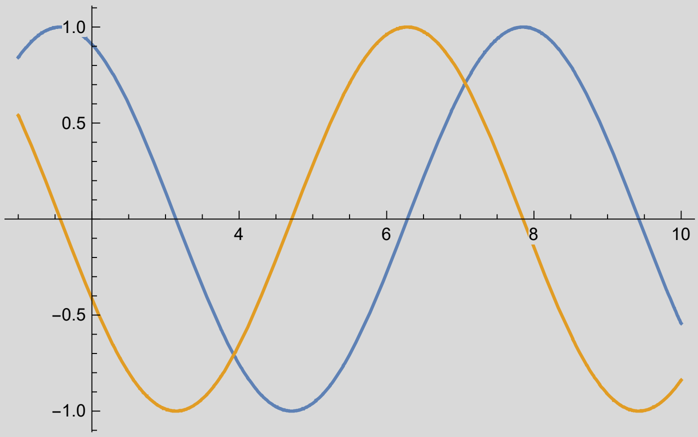
The various bracketing constructions can be used together.
Plot a function, with the range of the plot specified in a list:

In[66]:=
```
Plot[Sin[x], {x, 0, 10}]
```
Out[66]=



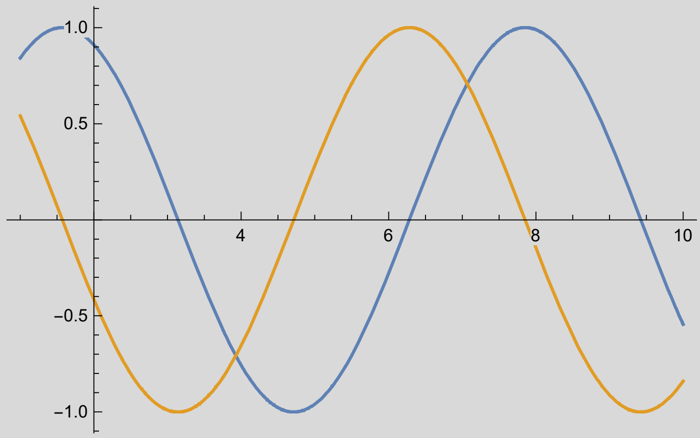
The ability to use functions and lists together is seamlessly integrated in the Wolfram Language.
Plot two functions together—the pair of functions is in a list:

In[67]:=
```
Plot[{Sin[x], Cos[x]}, {x, 1, 10}]
```
Out[67]=



In[68]:=
```
sincos = {Sin[x], Cos[x]};
```

In[69]:=
```
Plot[sincos, {x, 1, 10}]
```
Out[69]=



Variables & Functions

Variables:

Variables start with letters and can also contain numbers:

In[70]:=
```
x
```
Out[70]=
```
x
```

In[71]:=
```
x1
```
Out[71]=
```
x1
```

In[72]:=
```
mat123
```
Out[72]=
```
mat123
```

A space between two variables or numbers indicates multiplication:
(In other words, “a b” is a times b, whereas “ab” is the variable ab.)

In[73]:=
```
a b + 5 x x
```
Out[73]=

$a\ b + 5\ x^2$

Use **/. and →** to make substitutions in an expression:
(The “rule” → can be typed as ->.)

In[74]:=

$(1 + 2\ x + x^2)\ /.\ x \to 2$

Out[74]=
```
9
```

You can apply rules together by putting the rules in a list.

In[75]:=
```
(x + y) (x - y)^2 /. {x → 3, y → 1 - a}
```
Out[75]=
```
(4 - a) (2 + a)^2
```

Assign values using the = symbol:

In[76]:=
```
x = 2
```
Out[76]=
```
2
```

In[77]:=
```
x
```
Out[77]=
```
2
```

In[78]:=
```
x^2
```
Out[78]=
```
4
```

Use your variable in expressions and commands:

In[79]:=
```
1 + 2 x
```
Out[79]=
```
5
```

Important

Clear[*symbol*$_1$, *symbol*$_2$, …] clears values and definitions for the *symbol*$_i$.

Clear the assignment, and x remains unevaluated:

In[80]:=
```
Clear[x]
```

In[81]:=
```
x
```
Out[81]=
```
x
```

In[82]:=
```
x = 4
```
Out[82]=
```
4
```

In[83]:=
```
x =.
```

In[84]:=
```
1 + 2 x
```
Out[84]=
```
1 + 2 x
```

In[85]:=
```
t = 3; s = 5; r = 1;
```

In[86]:=
```
t
```
Out[86]=
```
3
```

In[87]:=
```
s
```
Out[87]=
```
5
```

In[88]:=
```
r
```
Out[88]=
```
1
```

In[89]:=
```
Clear[t, s, r]
```

In[90]:=
```
t
```
Out[90]=
```
t
```

Let a=2x+3y+5z, b=-x+6y-z, and c=5x-3y-3z. Compute the sum of a, b and c.

In[91]:=
```
Clear[a, b, c, x, y, z]
```

In[92]:=
```
a = 2 x + 3 y + 5 z;
b = -x + 6 y - z;
c = 5 x - 3 y - 3 z;
a + b + c
```

Out[95]=
```
6 x + 6 y + z
```

Functions:

Define your own functions with the construction **F[x_]:=**

F[*x*_] = *rhs* **(immediate assignment)**	*rhs* is evaluated when the assignment is made.
F[*x*_] := *rhs* **(delayed assignment)**	*rhs* is evaluated each time the value of *F*[*x*] is requested
F[*x*_, *y*_] = *rhs*	
F[*x*_, *y*_] := *rhs*	

x_ means that **x** is a pattern that can have any value substituted for it.
:= means that any argument passed to f is substituted into the right-hand side upon evaluation:

In[96]:=
```
Clear[f, h, g1, g2, x]
```

In[97]:=
```
f[x_] := 1 + 2 x
```

In[98]:=
```
f[2]
```

Out[98]=
```
5
```

In[99]:=
```
h[x_] = 1 + 2 x
```

Out[99]=
```
1 + 2 x
```

In[100]:=
```
h[2]
```

Out[100]=
```
5
```

In[101]:=
```
g1[x_] := Expand[(1 + x)^2]
```

In[102]:=
```
g2[x_] = Expand[(1 + x)^2]
```

Out[102]=
$1 + 2x + x^2$

In[103]:=
```
g1[y + 1]
```

Out[103]=
$4 + 4y + y^2$

In[104]:=
```
g2[y + 1]
```

Out[104]=
$1 + 2 (1 + y) + (1 + y)^2$

When difining piecewise functions, one must use :=. For example

$$g(x)=\begin{cases} x^2 & x \geqslant 0 \\ -x^2 & x \leq 0 \end{cases}$$

In[105]:=
```
Clear[g]
```

In[106]:=
```
g[x_] := x^2 /; x ≥ 0
g[x_] := -x^2 /; x <= 0
```

In[108]:=
```
g[3]
```

Out[108]=
```
9
```

In[109]:=
```
g[-4]
```

Out[109]=
```
-16
```

This is another way to define piecewise functions.

In[110]:=
```
G[x_] := Piecewise[{{x^2, x ≥ 0}, {-x^2, x <= 0}}]
```

In[111]:=
```
G[3]
```

Out[111]=
```
9
```

In[112]:=
```
G[-4]
```

Out[112]=
```
-16
```

We can use (**Basic Math Asistant)** to define piecewise functions as bellow.
To get (**Basic Math Asistant),** Palettes->**Basic Math Asistant.**

In[113]:=

$$H[x\_] := \begin{cases} x^2 & x \geq 0 \\ -x^2 & x < 0 \end{cases}$$

In[114]:=
```
H[3]
```

Out[114]=
```
9
```

In[115]:=
```
H[-4]
```

Out[115]=
```
-16
```

In[116]:=
```
P[x_, y_] = x y - x^2 + y^2
```

Out[116]=

$$-x^2 + x\,y + y^2$$

In[117]:=
```
P[1, 3]
```
Out[117]=
```
11
```

Algebra

Expressions

Transforming Algebraic Expressions

There are often many different ways to write the same algebraic expression. As one example, the expression $(1 + x)^2$ can be written as $1 + 2x + x^2$. The Wolfram System provides a large collection of functions for converting between different forms of algebraic expressions.

You can factor or expand algebraic expressions:
(Use CTRL+6 for typeset exponents.)

Factor[expr] write *expr* as a product of minimal factors
Expand[expr] multiply out products and powers, writing the result as a sum of terms

In[118]:=
```
Factor[x^2 + 2 x + 1]
```
Out[118]=
$(1 + x)^2$

In[119]:=
```
Expand[(1 + x)^3]
```
Out[119]=
$1 + 3 x + 3 x^2 + x^3$

In[120]:=
```
Expand[(1 + x + 3 y)^4]
```
Out[120]=
$1 + 4 x + 6 x^2 + 4 x^3 + x^4 + 12 y + 36 x y + 36 x^2 y + 12 x^3 y + 54 y^2 + 108 x y^2 + 54 x^2 y^2 + 108 y^3 + 108 x y^3 + 81 y^4$

In[121]:=
```
Factor[%]
```
Out[121]=
$(1 + x + 3 y)^4$

Expand[*expr*,*patt*] expand out *expr*, avoiding those parts which do not contain terms matching *patt*.

This avoids expanding parts which do not contain y

In[122]:=
```
Expand[(x + 1)^2 (y + 1)^2, x]
```
Out[122]=
$(1 + y)^2 + 2 x (1 + y)^2 + x^2 (1 + y)^2$

ExpandAll[*expr*] expands out all products and integer powers in any part of *expr*.
ExpandAll[*expr*,*patt*] expands out all products and integer powers in any part of *expr* avoiding those parts which do not contain terms matching *patt* (This avoids expanding parts which do not contain patt).

In[123]:=

$$r = \frac{(-1+x)^2\,(2+x)}{(-3+x)^2\,(1+x)}$$

Out[123]=

$$\frac{(-1+x)^2\,(2+x)}{(-3+x)^2\,(1+x)}$$

In[124]:=

```
Expand[r]
```

Out[124]=

$$\frac{2}{(-3+x)^2\,(1+x)} - \frac{3\,x}{(-3+x)^2\,(1+x)} + \frac{x^3}{(-3+x)^2\,(1+x)}$$

In[125]:=

```
ExpandAll[r]
```

Out[125]=

$$\frac{2}{9+3\,x-5\,x^2+x^3} - \frac{3\,x}{9+3\,x-5\,x^2+x^3} + \frac{x^3}{9+3\,x-5\,x^2+x^3}$$

ExpandNumerator[*expr*] expand numerators only.
ExpandDenominator[*expr*] expand denominators only

Expand the numerator of a fraction:

In[126]:=

```
ExpandNumerator[(x - 1) (x - 2) / ((x - 3) (x - 4))]
```

Out[126]=

$$\frac{2-3\,x+x^2}{(-4+x)\,(-3+x)}$$

Expand the denominator of a fraction:

In[127]:=

```
ExpandDenominator[(x - 1) (x - 2) / ((x - 3) (x - 4))]
```

Out[127]=

$$\frac{(-2+x)\,(-1+x)}{12-7\,x+x^2}$$

`PowerExpand[expr]`	expands all powers of products and powers.
`PowerExpand[`*expr*`, {`x_1`, `x_2`, …}]`	expands only with respect to the variables x_i
`PowerExpand[expr,Assumptions->assum]`	expand out expr assuming `assum`

In[128]:=
```
PowerExpand[(x y)^n]
```
Out[128]=
$x^n\ y^n$

In[129]:=
```
PowerExpand[Log[(x y)^n]]
```
Out[129]=
```
n (Log[x] + Log[y])
```

In[130]:=
```
PowerExpand[Log[x y]]
```
Out[130]=
```
Log[x] + Log[y]
```

Expand only with respect to `a` and `b`:

In[131]:=
```
Clear[a, b, c, d]
```

In[132]:=
```
PowerExpand[√(a b) + √(c d), {a, b}]
```
Out[132]=
$\sqrt{a}\ \sqrt{b} + \sqrt{c\ d}$

In[133]:=
```
PowerExpand[ArcTan[Cot[x]], Assumptions → 0 < x < π]
```
Out[133]=
$\frac{\pi}{2} - x$

Simplifying Algebraic Expressions

There are many situations where you want to write a particular algebraic expression in the simplest possible form. Although it is difficult to know exactly what one means in all cases by the “simplest form”, a worthwhile practical procedure is to look at many different forms of an expression, and pick out the one that involves the smallest number of parts.

`Simplify[expr]`	try to find the simplest form of *expr* by applying various standard algebraic transformations.
`FullSimplify[expr]`	try to find the simplest form by applying a wide range of transformations.

In[134]:=
```
Simplify[x^2 + 2 x + 1]
```
Out[134]=
$(1 + x)^2$

In[135]:=
```
FullSimplify[x^3 - 6 x^2 + 11 x - 6]
```
Out[135]=
```
(-3 + x) (-2 + x) (-1 + x)
```

In[136]:=
```
FullSimplify[Cosh[x] - Sinh[x]]
```
Out[136]=

e^{-x}

In[137]:=
```
Simplify[Cosh[x] - Sinh[x]]
```
Out[137]=
```
Cosh[x] - Sinh[x]
```

Together[expr] put all terms over a common denominator.

In[138]:=

$$\text{Together}\left[\frac{2}{9+3x-5x^2+x^3}-\frac{3x}{9+3x-5x^2+x^3}+\frac{x^3}{9+3x-5x^2+x^3}\right]$$

Out[138]=

$$\frac{2-3x+x^3}{(-3+x)^2\,(1+x)}$$

Apart[expr] separate into terms with simple denominator (partial fractions).

In[139]:=

$$\text{Apart}\left[\frac{(-1+x)^2\,(2+x)}{(-3+x)^2\,(1+x)}\right]$$

Out[139]=

$$1+\frac{5}{(-3+x)^2}+\frac{19}{4\,(-3+x)}+\frac{1}{4\,(1+x)}$$

Cancel[expr] cancel common factors between numerators and denominators.

In[140]:=

$$\text{Cancel}\left[\frac{x^2-1}{x-1}\right]$$

Out[140]=
```
1 + x
```

Collect[expr, x] group together powers of x.

This groups together terms in v that involve the same power of x.

In[141]:=

$$d = 9x^2 + 12x^3 + 4x^4 + 36xy + 48x^2y + 16x^3y + 36y^2 + 48xy^2 + 16x^2y^2;$$

In[142]:=
```
Collect[d, x]
```
Out[142]=
$4x^4 + 36y^2 + x^3 (12 + 16y) + x^2 (9 + 48y + 16y^2) + x (36y + 48y^2)$

This groups together terms in v that involve the same power of y.

In[143]:=
```
Collect[d, y]
```
Out[143]=
$9x^2 + 12x^3 + 4x^4 + (36x + 48x^2 + 16x^3) y + (36 + 48x + 16x^2) y^2$

FactorTerms[poly]	pull out any overall numerical factor
FactorTerms[expr,x]	pull out factors that do not depend on x

In[144]:=
```
FactorTerms[-4 + 12 x - 28 x^2 + 52 x^3 - 64 x^4 + 64 x^5 - 48 x^6 + 16 x^7]
```
Out[144]=
$4 (-1 + 3x - 7x^2 + 13x^3 - 16x^4 + 16x^5 - 12x^6 + 4x^7)$

This factors out the piece that does not depend on y.

In[145]:=
```
FactorTerms[d, y]
```
Out[145]=
$(9 + 12x + 4x^2) (x^2 + 4xy + 4y^2)$

Simplifying with Assumptions

Simplify[expr, assum]	simplify expr with assumptions

In[146]:=
```
Simplify[Sqrt[x^2], x > 0]
```
Out[146]=
x

In[147]:=
```
Simplify[Sqrt[x^2], x < 0]
```
Out[147]=
-x

In[148]:=
```
Clear[x]
```

In[149]:=
```
Simplify[ArcSin[Sin[x]], -π/2 < x < π/2]
```
Out[149]=
x

Some domains used in assumptions:

Element[x,dom]	state that x is an element of the domain dom
Element[{x1,x2,...},dom]	state that all the x_i are elements of the domain dom

Reals	real numbers
Integers	integers
Primes	prime numbers

In[150]:=
```
Simplify[Sqrt[x^2], Element[x, Reals]]
```
Out[150]=
```
Abs[x]
```

In[151]:=
```
Simplify[Sin[x + 2 n Pi], Element[n, Integers]]
```
Out[151]=
```
Sin[x]
```

Picking Out Pieces of Algebraic Expressions

Functions to pick out pieces of polynomials.

`Coefficient[expr, form]`	coefficient of form in expr
`CoefficientList[poly, var]` gives a list of coefficients of powers of *var* in *poly*, starting with power 0.	
`Exponent[expr, form]`	maximum power of form in `expr`
`Part[expr, n]` or `expr[[n]]`	n^{th} term of `expr`

In[152]:=
```
R = 1 + 6 x + 9 x^2 + 8 y^2 + 24 x y^2 + 16 y^4
```
Out[152]=
$1 + 6x + 9x^2 + 8y^2 + 24xy^2 + 16y^4$

This gives the coefficient of x in R.

In[153]:=
```
Coefficient[R, x]
```
Out[153]=
$6 + 24y^2$

This gives the coefficient of y^2 in R.

In[154]:=
```
Coefficient[R, y^2]
```
Out[154]=
$8 + 24x$

This gives the coefficient of x^0(**constant term**) in R.

In[155]:=
```
Coefficient[R, x, 0]
```
Out[155]=
$1 + 8y^2 + 16y^4$

This gives a list of coefficients of powers of x in R.

In[156]:=
```
CoefficientList[R, x]
```
Out[156]=
$\{1 + 8y^2 + 16y^4, 6 + 24y^2, 9\}$

This gives the highest power of `y` that appears in R.

In[157]:=
```
Exponent[R, y]
```
Out[157]=
4

This gives the fourth term in R.

In[158]:=
```
Part[R, 4]
```
Out[158]=
$8y^2$

Numerator[expr]	numerator of expr
Denominator[expr]	denominator of expr

In[159]:=
```
g = (1 + x) / (2 (2 - y))
```
Out[159]=
$$\frac{1 + x}{2\ (2 - y)}$$

In[160]:=
```
Numerator[g]
```
Out[160]=
```
1 + x
```

In[161]:=
```
Denominator[g]
```
Out[161]=
```
2 (2 - y)
```

Solving Equations

Combine algebraic expressions with == to represent an equation:

In[162]:=
```
1 + z == 15
```
Out[162]=
```
1 + z == 15
```

Solve[expr,vars] attempts to solve the system *expr* of equations or inequalities for the variables *vars*
Solve[expr,vars,dom] solves over the domain *dom*. Common choices of *dom* are ℝ,ℤ and ℂ.

Commands like **Solve** find exact solutions to equations:

In[163]:=
$$\text{Solve}\left[x^2 + 5\,x - 6 == 0,\ x\right]$$
Out[163]=
```
{{x → -6}, {x → 1}}
```

In[164]:=
```
Clear[a, b, c, x]
```

In[165]:=
$$\text{Solve}\left[a\,x^2 + b\,x + c == 0,\ x\right]$$
Out[165]=
$$\left\{\left\{x \to \frac{-b - \sqrt{b^2 - 4\,a\,c}}{2\,a}\right\},\ \left\{x \to \frac{-b + \sqrt{b^2 - 4\,a\,c}}{2\,a}\right\}\right\}$$

Pass in a system of equations as a list:

In[166]:=
```
Solve[{x + y == 3, x - y == 1}, {x, y}]
```
Out[166]=
```
{{x → 2, y → 1}}
```

In[167]:=
```
Solve[1 + 2 x + 2 x^2 + x^3 == 0, x, Reals]
```
Out[167]=
```
{{x → -1}}
```

You need to buy 100 birds for $100?
$1 = 1 pigeon.حمامة
$5 = 1 chicken.دجاجة
$1 = 20 sparrows.عصفور
You need to buy at least one of each.
So how many to buy so you spend only 100 but get 100 birds also, and get all birds too?

In[168]:=
```
Solve[{p + c + s == 100, p + 5 c + 1/20 s == 100, p > 0, c > 0, s > 0},
  Element[{p, c, s}, Integers]]
```
Out[168]=
```
{{p → 1, c → 19, s → 80}}
```

NSolve[expr,vars] attempts to find numerical approximations to the solutions of the system *expr* of equations or inequalities for the variables *vars*.
NSolve[expr,vars,Reals] finds solutions over the domain of real numbers.

In[169]:=
```
Solve[7 x^5 + 3 x - 5 == 0, x]
```
Out[169]=
```
{{x → Root[-5 + 3 #1 + 7 #1^5 &, 1]},
 {x → Root[-5 + 3 #1 + 7 #1^5 &, 2]}, {x → Root[-5 + 3 #1 + 7 #1^5 &, 3]},
 {x → Root[-5 + 3 #1 + 7 #1^5 &, 4]}, {x → Root[-5 + 3 #1 + 7 #1^5 &, 5]}}
```

In[170]:=
```
NSolve[7 x^5 + 3 x - 5 == 0, x]
```
Out[170]=
```
{{x → -0.791393 - 0.638892 i}, {x → -0.791393 + 0.638892 i},
 {x → 0.382867 - 0.835759 i}, {x → 0.382867 + 0.835759 i}, {x → 0.817051}}
```

In[171]:=
```
NSolve[7 x^5 + 3 x - 5 == 0, x, Reals]
```
Out[171]=
```
{{x → 0.817051}}
```

In[172]:=
```
Solve[-1 + 2 x + 2 x^2 + x^5 == 0, x] // N
```
Out[172]=
```
{{x → 0.364174}, {x → -1.00023 - 0.556621 i}, {x → -1.00023 + 0.556621 i},
 {x → 0.818145 - 1.19428 i}, {x → 0.818145 + 1.19428 i}}
```

In[173]:=
```
NSolve[-1 + 2 x + 2 x^2 + x^5 == 0, x]
```
Out[173]=
```
{{x → -1.00023 - 0.556621 i}, {x → -1.00023 + 0.556621 i},
 {x → 0.364174}, {x → 0.818145 - 1.19428 i}, {x → 0.818145 + 1.19428 i}}
```

In[174]:=

```
NSolve[-1 + 2 x + 2 x^2 + x^5 == 0, x, Reals]
```

Out[174]=

```
{{x → 0.364174}}
```

Solving Inequalities

Just as the *equation* `x^2 + 3 x == 2` asserts that `x^2 + 3 x` is equal to `2`, so also the *inequality* `x^2 + 3 x > 2` asserts that `x^2 + 3 x` is greater than `2`. In the Wolfram Language, Reduce works not only on equations, but also on inequalities.

The `Reduce` command reduces a set of inequalities into a simple form:

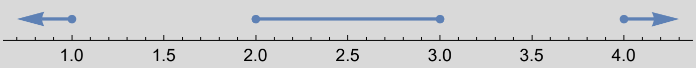
Reduce[expr, vars] reduces the statement *expr* by solving equations or inequalities for *vars* and eliminating quantifiers.
Reduce[expr, vars, dom] does the reduction over the domain *dom*. Common choices of *dom* are ℝ, ℤ and ℂ.

In[175]:=
```
Reduce[{0 < x < 2, 1 ≤ x ≤ 4}, x]
```
Out[175]=
```
1 ≤ x < 2
```

The reduced form may include multiple intervals:

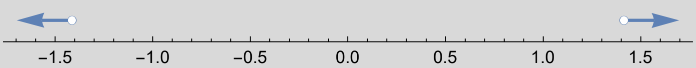
In[176]:=
```
Reduce[(x - 1) (x - 2) (x - 3) (x - 4) > 0, x]
```
Out[176]=
```
x < 1 || 2 < x < 3 || x > 4
```

NumberLinePlot[expr, vars] is a handy way to visualize the results of inequalities.

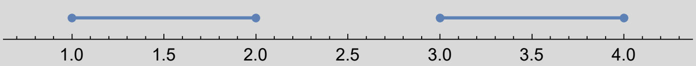
In[177]:=
```
NumberLinePlot[(x - 1) (x - 2) (x - 3) (x - 4) >= 0, x]
```
Out[177]=


Abs[x] or |x| is absolute value of x.

In[178]:=
```
Reduce[Abs[x^2 + 1] > Abs[x^2 - 5], Element[x, Reals]]
```
Out[178]=
$x < -\sqrt{2} \;||\; x > \sqrt{2}$

In[179]:=
```
NumberLinePlot[Abs[x^2 + 1] > Abs[x^2 - 5], x]
```
Out[179]=


In[180]:=
```
Reduce[(x - 1) (x - 2) (x - 3) (x - 4) <= 0, x]
```
Out[180]=
```
1 ≤ x ≤ 2 || 3 ≤ x ≤ 4
```

In[181]:=
```
NumberLinePlot[(x - 1) (x - 2) (x - 3) (x - 4) <= 0, x]
```
Out[181]=


This pair of inequalities reduces to a single inequality.

In[182]:=

```
Reduce[{0 < x < 2, 1 < x < 4}, x]
```

Out[182]=

```
1 < x < 2
```

These inequalities can never simultaneously be satisfied.

In[183]:=

```
Reduce[{x < 1, x > 3}, x]
```

Out[183]=

```
False
```

FindInstance[expr,vars] finds an instance of *vars* that makes the statement *expr* be True.

In[184]:=

```
FindInstance[x^2 + y^2 <= 1, {x, y}]
```

Out[184]=

```
{{x → 0, y → 0}}
```

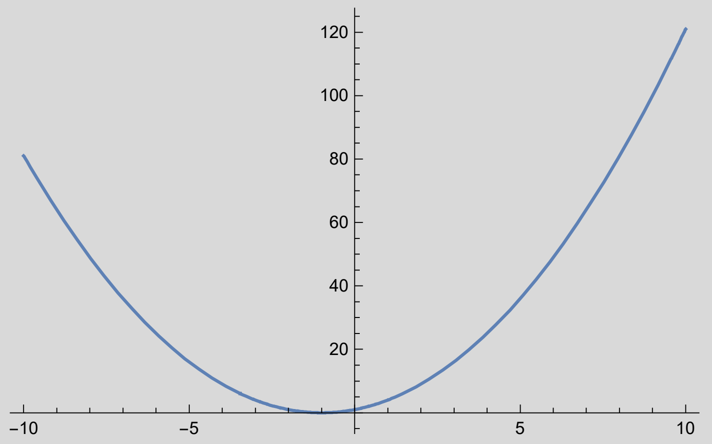
Plots in 2D

`Plot[f, {x, xmin, xmax}]` plot f as a function of x from x_{min} to x_{max}.
`Plot[{f1,f2,...},{x, xmin, xmax}]` plot several functions together.
`ContourPlot[f, {x, xmin, xmax},{y, ymin, ymax}]` generates a contour plot of f as a function of x and y.

In[185]:=

```
Plot[x^2 + 2 x + 1, {x, -10, 10}]
```

Out[185]=

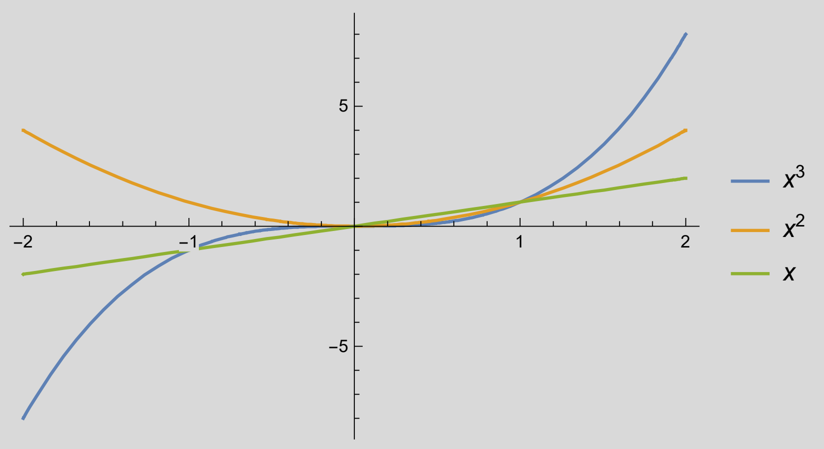


There are lots of useful options to customize visualizations, like adding legends:

In[186]:=

```
Plot[{x^3, x^2, x}, {x, -2, 2}, PlotLegends → "Expressions"]
```

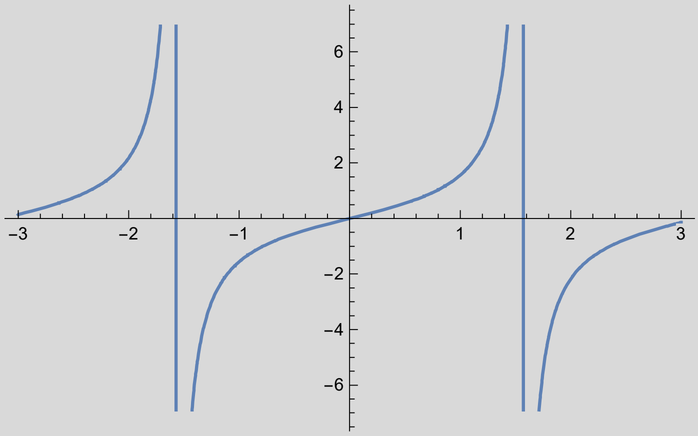
Out[186]=



In[187]:=

```
Plot[Tan[x], {x, -3, 3}]
```

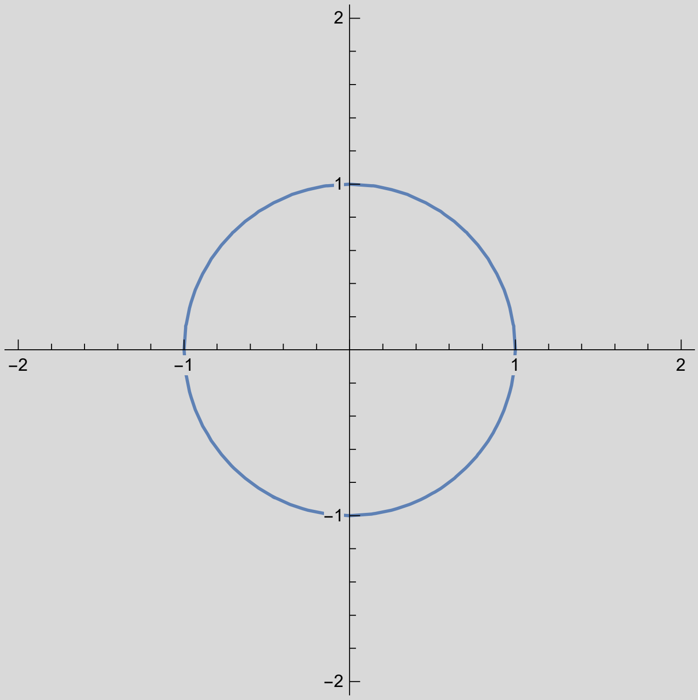
Out[187]=



In[188]:=

```
ContourPlot[x^2 + y^2 == 1, {x, -2, 2}, {y, -2, 2},
 Frame -> False, AxesOrigin → {0, 0}, Axes → True]
```

Out[188]=



Plots in 3D

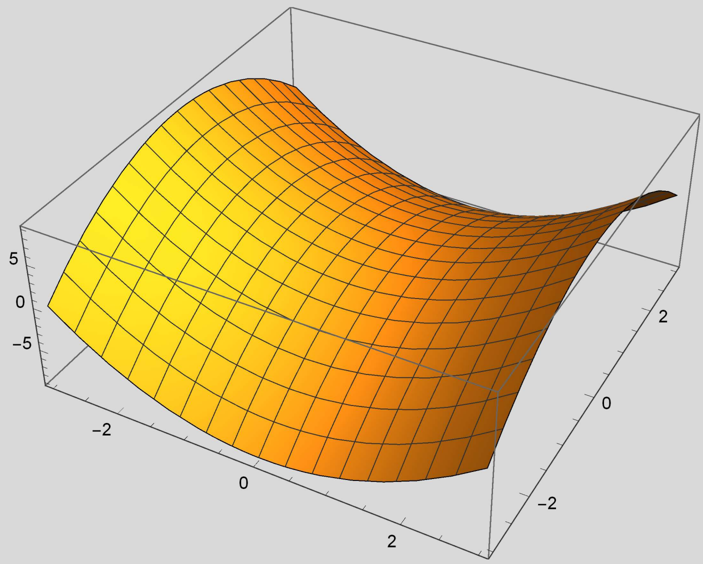
Plot3D[f, {x, x_{min}, x_{max}}, {y, y_{min}, y_{max}}] generates a three-dimensional plot of f as a function of x and y.
Plot3D[{f_1, f_2, ...}, {x, x_{min}, x_{max}},{y, y_{min}, y_{max}}] plot several functions together.

Plot3D will plot a **3D** Cartesian curve or surface:

In[189]:=

```
Plot3D[x^2 - y^2, {x, -3, 3}, {y, -3, 3}]
```

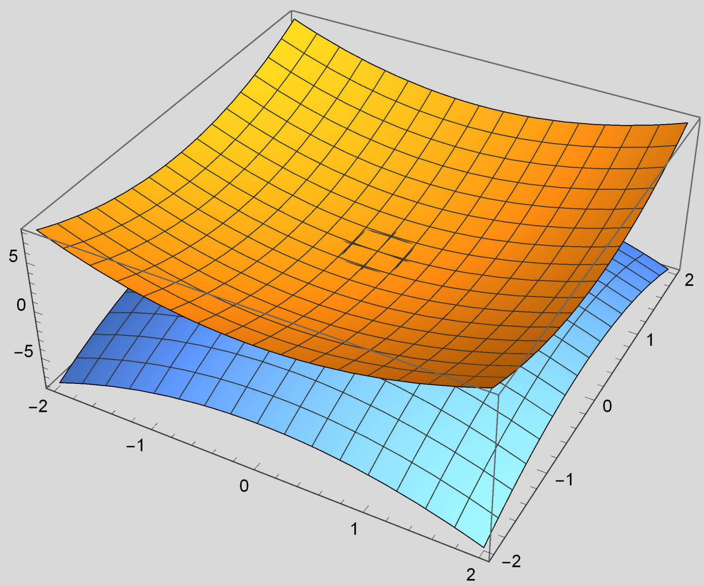
Out[189]=



In[190]:=

```
Plot3D[{x^2 + y^2, -x^2 - y^2}, {x, -2, 2}, {y, -2, 2}]
```

Out[190]=

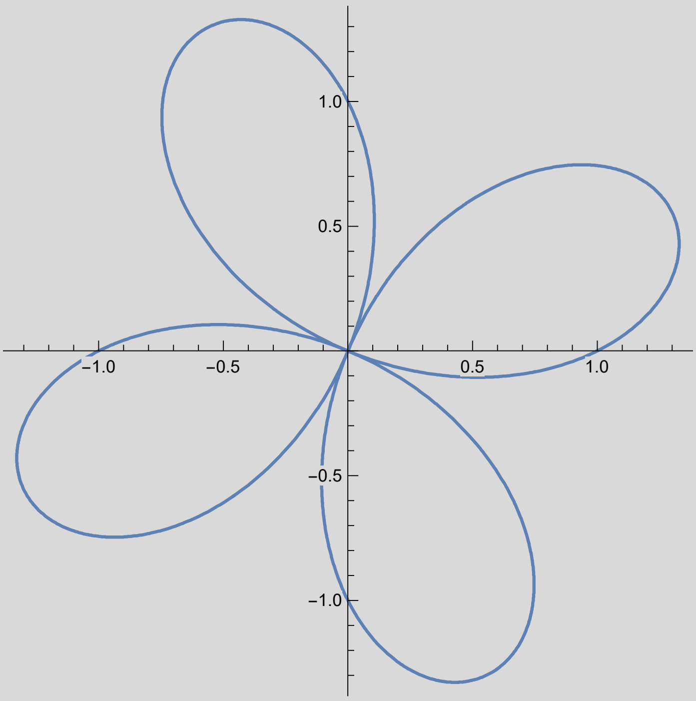


Polar Coordinates

PolarPlot[r, {θ, θ_{min}, θ_{max}}] generates a polar plot of a curve with radius r as a function of angle θ.

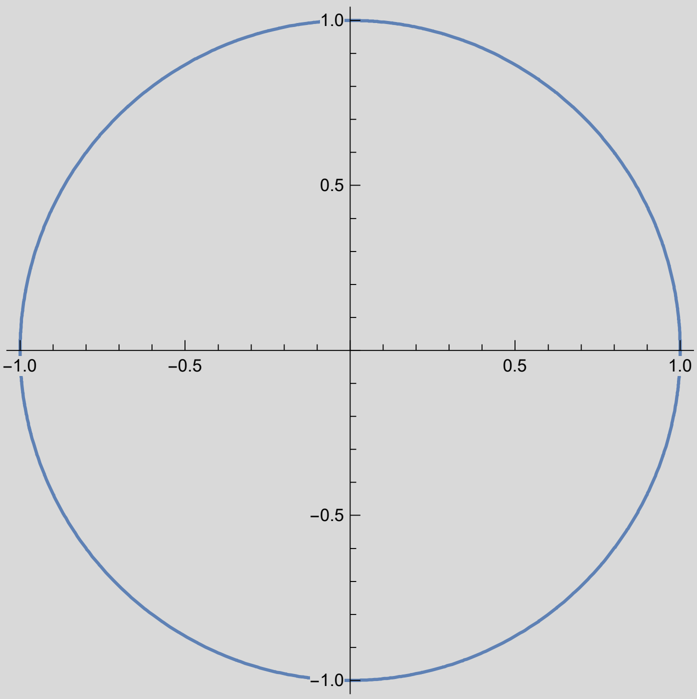
In[191]:=
```
PolarPlot[Sin[2 θ] + Cos[2 θ], {θ, 0, 2 Pi}]
```

Out[191]=



In[192]:=
```
PolarPlot[Sin[θ]^2 + Cos[θ]^2, {θ, 0, 2 Pi}]
```

Out[192]=



Convert Cartesian coordinates to polar:

In[193]:=
```
ToPolarCoordinates[{1, 1}]
```

Out[193]=
```
ToPolarCoordinates[{1, 1}]
```

Calculus

Limits

`Limit[expr, x -> x0]` finds the limiting value of *expr* when x approaches x_0.

Calculate the limiting value of an expression:

In[194]:=
```
Limit[(x^3 - 1)/(x - 1), x -> 1]
```
Out[194]=
```
3
```

Find the limit at ∞ :

In[195]:=
```
Limit[(2 x^3 - 1)/(5 x^3 + x + 1), x -> ∞]
```
Out[195]=
```
2/5
```

In[196]:=
```
Limit[(1 + 1/n)^n, n -> ∞]
```
Out[196]=
```
e
```

In[197]:=
```
E // N
```
Out[197]=
```
2.71828
```

You can also specify the limit's Direction.

A setting of 1 approaches the limit from the left:

In[198]:=
```
Limit[1/x, x -> 0, Direction -> 1]
```
Out[198]=
```
-∞
```

A setting of −1 approaches the limit from the right:

In[199]:=
```
Limit[1/x, x -> 0, Direction -> -1]
```
Out[199]=
```
∞
```

Derivatives

Calculate derivatives with the D command:

D[*f*, *x*] gives the partial derivative ∂f / ∂x.
D[*f*, {*x*, *n*}]
gives the multiple derivative ∂^n f / ∂ x^n.

In[200]:=
```
D[x^6, x]
```
Out[200]=
```
6 x^5
```

In[201]:=
```
∂_x x^6
```
Out[201]=
```
6 x^5
```

In[202]:=
```
∂_x Cos[x]
```
Out[202]=
```
-Sin[x]
```

In[203]:=
```
D[x^6, {x, 3}]
```
Out[203]=
```
120 x^3
```

Or use prime notation :

In[204]:=
```
Sin'[x]
```
Out[204]=
```
Cos[x]
```

Or use the ' symbol multiple times:

In[205]:=
```
Sin''[x]
```
Out[205]=
```
-Sin[x]
```

Differentiate user-defined functions:

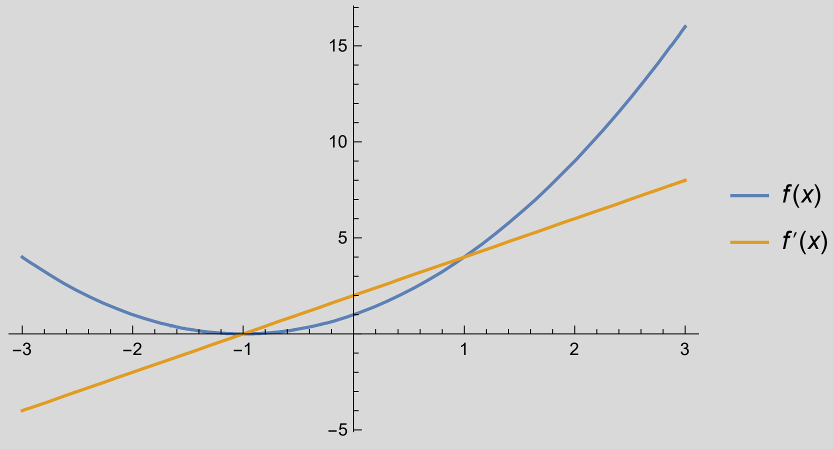
In[206]:=
```
f[x_] := x^2 + 2 x + 1
f'[x]
```
Out[207]=
```
2 + 2 x
```

Pass derivatives directly into a plot:

In[208]:=

```
Plot[{f[x], f'[x]}, {x, -3, 3}, PlotLegends → "Expressions"]
```

Out[208]=



Integrals

Integrate[*f*,*x*]	gives the indefinite integral .
Integrate[*f*,{*x*,*xmin*,*xmax*}]	gives the definite integral .
Integrate[*f*, {*x*, x_{min},x_{max}},{*y*,y_{min},y_{max}},...]	gives the multiple integral

Compute integrals with **Integrate**:

In[209]:=
```
Integrate[x^2, x]
```
Out[209]= $\frac{x^3}{3}$

In[210]:= $\int x^2 \, dx$

Out[210]= $\frac{x^3}{3}$

In[211]:= $\int \mathrm{Cos}[x]\,\mathrm{Sin}[x] \, dx$

Out[211]= $-\frac{1}{2}\mathrm{Cos}[x]^2$

In[212]:=
```
Integrate[x^2, {x, 0, 1}]
```
Out[212]= $\frac{1}{3}$

In[213]:= $\int_0^1 x^2 \, dx$

Out[213]= $\frac{1}{3}$

Use **NIntegrate** for a numeric approximation :

In[214]:=
```
NIntegrate[x^3 Sin[x] + 2 Log[3 x]^2, {x, 0, Pi}]
```
Out[214]= 28.1531

In[215]:=
```
Integrate[x^3 Sin[x] + 2 Log[3 x]^2, {x, 0, Pi}]
```
Out[215]= $\pi\left(-6+\pi^2+2\left(2+\mathrm{Log}[3]^2+\mathrm{Log}[9]\,(-1+\mathrm{Log}[\pi])+(-2+\mathrm{Log}[\pi])\,\mathrm{Log}[\pi]\right)\right)$

In[216]:=

```
N[∫_0^π (x^3 Sin[x] + 2 Log[3 x]^2) ⅆx]
```

Out[216]=

```
28.1531
```

In[217]:=

```
∫_0^π (x^3 Sin[x] + 2 Log[3 x]^2) ⅆx // N
```

Out[217]=

```
28.1531
```

Sequences

In the Wolfram Language, integer sequences are represented by lists.

`Table[expr, {imax}]`	generates a list of i_{max} copies of *expr*.

In[218]:=
```
Table[x^2, {5}]
```
Out[218]=
```
{x^2, x^2, x^2, x^2, x^2}
```

In[219]:=
```
Table[2, {7}]
```
Out[219]=
```
{2, 2, 2, 2, 2, 2, 2}
```

`Table[expr, {i, imax}]`	generates a list of the values of *expr* when i runs from 1 to i_{max}.

In[220]:=
```
Table[i, {i, 10}]
```
Out[220]=
```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

In[221]:=
```
Table[i^2, {i, 10}]
```
Out[221]=
```
{1, 4, 9, 16, 25, 36, 49, 64, 81, 100}
```

In[222]:=
```
Table[i^2/(1 + i^3), {i, 10}]
```
Out[222]=

$$\left\{\frac{1}{2}, \frac{4}{9}, \frac{9}{28}, \frac{16}{65}, \frac{25}{126}, \frac{36}{217}, \frac{49}{344}, \frac{64}{513}, \frac{81}{730}, \frac{100}{1001}\right\}$$

`Table[expr, {i, imin, imax}]`	starts with $i = i_{min}$.

In[223]:=
```
Table[i, {i, 3, 10}]
```
Out[223]=
```
{3, 4, 5, 6, 7, 8, 9, 10}
```

In[224]:=
```
Table[i^2, {i, 3, 10}]
```
Out[224]=
```
{9, 16, 25, 36, 49, 64, 81, 100}
```

In[225]:=
```
Table[x^2 - 2 x + 1, {x, 1, 7}]
```
Out[225]=
```
{0, 1, 4, 9, 16, 25, 36}
```

`Table[expr, {i, imin, imax, di}]`	uses steps *di*.

In[226]:=
```
list = Table[i, {i, 3, 20, 2}]
```
Out[226]=
```
{3, 5, 7, 9, 11, 13, 15, 17, 19}
```

Find the length of a list :

In[227]:=
```
Length[list]
```
Out[227]=
```
9
```

In[228]:=
```
Table[x^2 + Sin[x], {x, 1, 7, 0.5}]
```
Out[228]=
```
{1.84147, 3.24749, 4.9093, 6.84847, 9.14112, 11.8992,
 15.2432, 19.2725, 24.0411, 29.5445, 35.7206, 42.4651, 49.657}
```

Table[*expr*, {*i*, {i_1, i_2, ...}}] uses the successive values i_1, i_2,....

In[229]:=
```
Table[2 x^2 + 3 x - 5, {x, {1, -3, 5, 8, 100}}]
```
Out[229]=
```
{0, 4, 60, 147, 20 295}
```

Table[*expr*, {*i*, i_{min}, i_{max}}, {*j*, j_{min}, j_{max}}, ...] gives a nested list. The list associated with *i* is outermost.

Make a 4×3 matrix:

In[230]:=
```
A = Table[10 i + j, {i, 4}, {j, 3}]
```
Out[230]=
```
{{11, 12, 13}, {21, 22, 23}, {31, 32, 33}, {41, 42, 43}}
```

In[231]:=
```
MatrixForm[A]
```
Out[231]//MatrixForm=

$$\begin{pmatrix} 11 & 12 & 13 \\ 21 & 22 & 23 \\ 31 & 32 & 33 \\ 41 & 42 & 43 \end{pmatrix}$$

In[232]:=
```
MatrixForm[Table[10 i + j, {i, 4}, {j, 3}]]
```
Out[232]//MatrixForm=

$$\begin{pmatrix} 11 & 12 & 13 \\ 21 & 22 & 23 \\ 31 & 32 & 33 \\ 41 & 42 & 43 \end{pmatrix}$$

In[233]:=
```
A // MatrixForm
```
Out[233]//MatrixForm=

$$\begin{pmatrix} 11 & 12 & 13 \\ 21 & 22 & 23 \\ 31 & 32 & 33 \\ 41 & 42 & 43 \end{pmatrix}$$

Range[i_{max}]	generates the list {1, 2, …, i_{max}}.
Range[i_{min}, i_{max}]	generates the list {i_{min}, …, i_{max}}.
Range[i_{min}, i_{max}, di]	uses step di.

In[234]:=
```
Range[4]
```
Out[234]=
```
{1, 2, 3, 4}
```

In[235]:=
```
Range[7, 20]
```
Out[235]=
```
{7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
```

In[236]:=
```
Range[x, x + 4]
```
Out[236]=
```
{x, 1 + x, 2 + x, 3 + x, 4 + x}
```

Produce a geometric sequence:

In[237]:=
```
x^Range[5]
```
Out[237]=

$\{x, x^2, x^3, x^4, x^5\}$

A table with *i* running from 0 to 10 in steps of 2:

In[238]:=
```
Clear[g, x]
```

In[239]:=
```
g[x_] = x Sin[x] - 1;
```

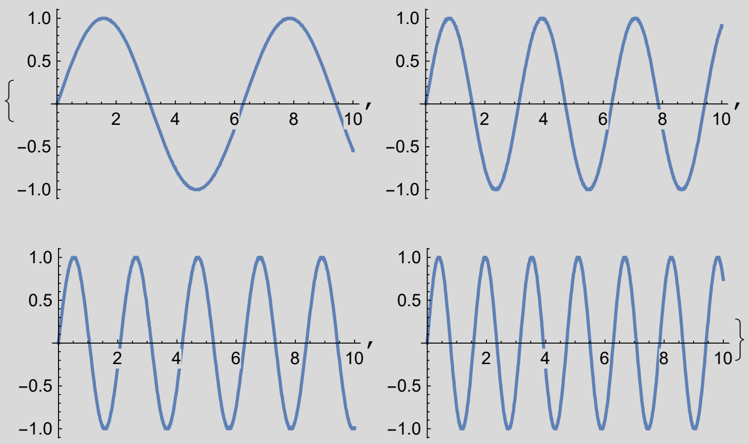
In[240]:=
```
Table[g[i], {i, 0, 1, 0.2}] // N
```

Out[240]=
```
{-1., -0.960266, -0.844233, -0.661215, -0.426115, -0.158529}
```

Make a table of graphics:

In[241]:=
```
Table[Plot[Sin[n x], {x, 0, 10}], {n, 4}]
```

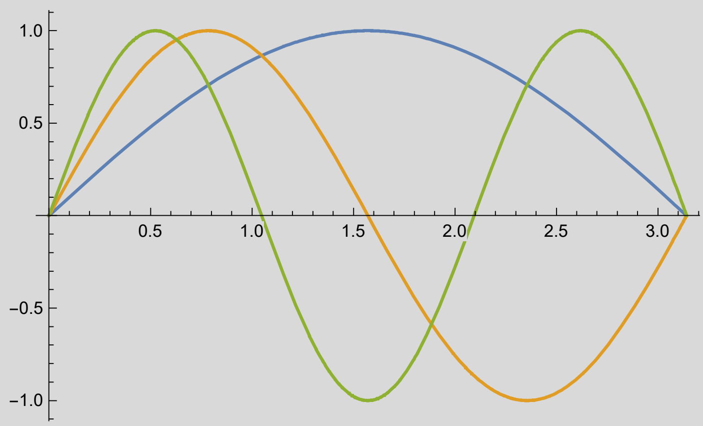
Out[241]=



The **Evaluate** is needed to force evaluation of the table before it is fed to **Plot**:

In[242]:=
```
Plot[Evaluate[Table[Sin[n x], {n, 3}]], {x, 0, π}]
```

Out[242]=



Sums

Compute the Sum of a sequence from its generating function:

Sum[f, {i, i_{max}}]   evaluates the sum $\sum_{i=1}^{i_{max}} f$.

In[243]:=
```
Sum[i (i + 1), {i, 10}]
```
Out[243]=
```
440
```

Sum[f, {i, i_{min}, i_{max}}]   starts with $i = i_{min}$.

In[244]:=
```
Sum[i (i + 1), {i, 3, 10}]
```
Out[244]=
```
432
```

In[245]:=

$$\sum_{i=3}^{10} i\ (i+1)$$

Out[245]=
```
432
```

Sum[f, {i, i_{min}, i_{max}, di}]   uses steps di.

In[246]:=
```
Sum[i (i + 1), {i, 3, 10, 2}]
```
Out[246]=
```
188
```

Sum[f, {i, {i_1, i_2, …}}] uses successive values i_1, i_2,

In[247]:=
```
Sum[i (i + 1), {i, {3, 4, 7, 9, 13}}]
```
Out[247]=
```
360
```

You can do indefinite and multiple sums :

Sum[f, {i, i_{min}, i_{max}}, {j, j_{min}, j_{max}}, …] evaluates the multiple sum $\sum_{i=i_{min}}^{i_{max}} \sum_{j=j_{min}}^{j_{max}} \ldots f$.

In[248]:=
```
Sum[i j, {i, 1, 10}, {j, 1, 10}]
```
Out[248]=
```
3025
```

In[249]:=

$$\sum_{i=1}^{10} \sum_{j=1}^{10} i\ j$$

Out[249]=
```
3025
```

In[250]:=
```
Sum[i j, {i, 1, n}, {j, 1, n}]
```
Out[250]=
$$\frac{1}{4} n^2 (1+n)^2$$

In[251]:=
$$\sum_{i=1}^{n}\sum_{j=1}^{n} i\ j$$
Out[251]=
$$\frac{1}{4} n^2 (1+n)^2$$

Calculate a generating function for a sequence :

In[252]:=
```
FindSequenceFunction[{1, 3, 5, 7}, n]
```
Out[252]=
$$-1+2\,n$$

In[253]:=
```
FindSequenceFunction[{2, 6, 12, 20, 30}, n]
```
Out[253]=
$$n+n^2$$

Convergent series may be automatically simplified:

In[254]:=
$$\sum_{n=0}^{\infty} 0.5^n$$
Out[254]=
```
2.
```

In[255]:=
$$\sum_{n=0}^{\infty} (-0.3)^n$$
Out[255]=
```
0.769231
```

Series

Series[f, {x, x_0, n}] generates a power series expansion for f about the point $x = x_0$ to order $(x - x_0)^n$
Series[f, {x, x_0, n}, {y, y_0, n}, …] successively finds series expansions with respect to x, then y, etc.

Generate power series approximations (**Taylor sereis**) to virtually any combination of built-in functions:

In[256]:=
```
S1 = Series[Exp[x^2], {x, 0, 8}]
```
Out[256]=
$$1 + x^2 + \frac{x^4}{2} + \frac{x^6}{6} + \frac{x^8}{24} + O[x]^9$$

$O[x]^9$ represents higher - order terms that have been omitted; use **Normal** to truncate this term :

In[257]:=
```
Normal[S1]
```
Out[257]=
$$1 + x^2 + \frac{x^4}{2} + \frac{x^6}{6} + \frac{x^8}{24}$$

In[258]:=
```
S1 // Normal
```
Out[258]=
$$1 + x^2 + \frac{x^4}{2} + \frac{x^6}{6} + \frac{x^8}{24}$$

In[259]:=
```
Series[F[x], {x, x0, 10}]
```
Out[259]=
$$F[x0] + F'[x0]\,(x - x0) + \frac{1}{2} F''[x0]\,(x - x0)^2 + \frac{1}{6} F^{(3)}[x0]\,(x - x0)^3 + \frac{1}{24} F^{(4)}[x0]\,(x - x0)^4 + \frac{1}{120} F^{(5)}[x0]\,(x - x0)^5 + \frac{1}{720} F^{(6)}[x0]\,(x - x0)^6 + \frac{F^{(7)}[x0]\,(x - x0)^7}{5040} + \frac{F^{(8)}[x0]\,(x - x0)^8}{40\,320} + \frac{F^{(9)}[x0]\,(x - x0)^9}{362\,880} + \frac{F^{(10)}[x0]\,(x - x0)^{10}}{3\,628\,800} + O[x - x0]^{11}$$

Power series in two variables:

In[260]:=
```
S2 = Series[Sin[x + y], {x, 0, 3}, {y, 0, 3}]
```

Out[260]=

$$\left(y - \frac{y^3}{6} + O[y]^4\right) + \left(1 - \frac{y^2}{2} + O[y]^4\right) x + \left(-\frac{y}{2} + \frac{y^3}{12} + O[y]^4\right) x^2 + \left(-\frac{1}{6} + \frac{y^2}{12} + O[y]^4\right) x^3 + O[x]^4$$

$$\left(y - \frac{y^3}{6} + O[y]^4\right) + \left(1 - \frac{y^2}{2} + O[y]^4\right) x + \left(-\frac{y}{2} + \frac{y^3}{12} + O[y]^4\right) x^2 + \left(-\frac{1}{6} + \frac{y^2}{12} + O[y]^4\right) x^3 + O[x]^4$$

$$\left(y - \frac{y^3}{6} + O[y]^4\right) + \left(1 - \frac{y^2}{2} + O[y]^4\right) x + \left(-\frac{y}{2} + \frac{y^3}{12} + O[y]^4\right) x^2 + \left(-\frac{1}{6} + \frac{y^2}{12} + O[y]^4\right) x^3 + O[x]^4$$

$$\left(y - \frac{y^3}{6} + O[y]^4\right) + \left(1 - \frac{y^2}{2} + O[y]^4\right) x + \left(-\frac{y}{2} + \frac{y^3}{12} + O[y]^4\right) x^2 + \left(-\frac{1}{6} + \frac{y^2}{12} + O[y]^4\right) x^3 + O[x]^4$$

$$\left(y - \frac{y^3}{6} + O[y]^4\right) + \left(1 - \frac{y^2}{2} + O[y]^4\right) x + \left(-\frac{y}{2} + \frac{y^3}{12} + O[y]^4\right) x^2 + \left(-\frac{1}{6} + \frac{y^2}{12} + O[y]^4\right) x^3 + O[x]^4$$

In[261]:=
```
S2 // Normal
```

Out[261]=

$$y - \frac{y^3}{6} + x\left(1 - \frac{y^2}{2}\right) + x^3\left(-\frac{1}{6} + \frac{y^2}{12}\right) + x^2\left(-\frac{y}{2} + \frac{y^3}{12}\right)$$

Multivariate Calculus

`D` works for partial derivatives—just specify which variable(s) to differentiate:

In[262]:=

$$D\left[x^3 z + 2 y^2 x + y z^3, y, z\right]$$

Out[262]=

$$3 z^2$$

Or use the ∂ symbol:

In[263]:=

$$\partial_{y,z}\left(x^3 z + 2 y^2 x + y z^3\right)$$

Out[263]=

$$3 z^2$$

Multiple integrals use the same notation as single integrals

In[264]:=

$$\int\int\int (x^\wedge 2 + y^\wedge 2 + z^\wedge 2)\, dy\, dx\, dz$$

Out[264]=

$$\frac{1}{3} x y z \left(x^2 + y^2 + z^2\right)$$

In[265]:=

$$\int\int\int (x^2+y^2+z^2)\, dx\, dy\, dz$$

Out[265]=

$$\frac{1}{3}\, x\, y\, z\, \left(x^2+y^2+z^2\right)$$

In[266]:=

$$\int\left(\int\left(\int (x^2+y^2+z^2)\, dx\right) dy\right) dz$$

Out[266]=

$$\frac{1}{3}\, x\, y\, \left(x^2\, z+y^2\, z+z^3\right)$$

In[267]:=

```
Solve[a x + y == 7 && b x - y == 1, {x, y}]
```

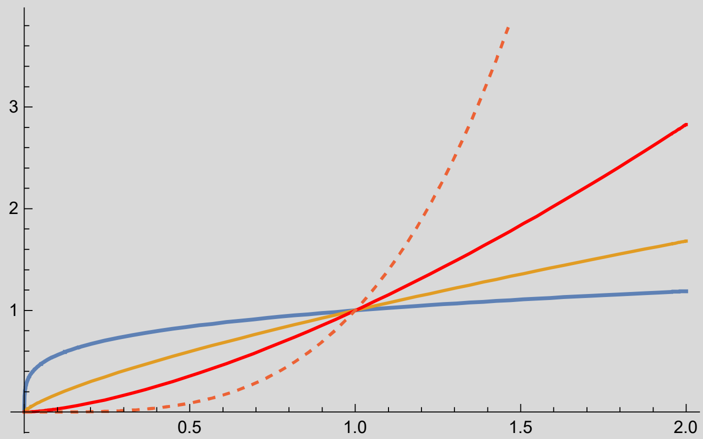
Out[267]=

$$\left\{\left\{x\to\frac{8}{a+b},\ y\to-\frac{a-7\,b}{a+b}\right\}\right\}$$

In[268]:=

```
Plot[{x^(1/4), x^(3/4), x^(3/2), x^(7/2)},
 {x, 0, 2}, PlotStyle → {Thick, Automatic, Red, Dashed}]
```

Out[268]=



In[269]:=

$$x^2 + 2\,x + 4 == 0$$

Out[269]=

$$4 + 2\,x + x^2 == 0$$

In[270]:=

$$\text{Reduce}\left[4 + 2\,x + x^2 == 0\right]$$

Out[270]=

$$x == -1 - i\,\sqrt{3}\ \ ||\ \ x == -1 + i\,\sqrt{3}$$

In[271]:=

$$\text{Solve}\left[4 + 2\,x + x^2 == 0,\ \{x\}\right]$$

Out[271]=

$$\left\{\left\{x\to -1 - i\,\sqrt{3}\right\},\ \left\{x\to -1 + i\,\sqrt{3}\right\}\right\}$$

In[272]:=

$$x^2 + 2\,x + 4$$

Out[272]=

$$4 + 2\,x + x^2$$

In[273]:=
```
FindMinimum[4 + 2 x + x^2, {{x, 1}}]
```
Out[273]=
```
{3., {x → -1.}}
```

In[274]:=
```
NMinimize[4 + 2 x + x^2, {x}]
```
Out[274]=
```
{3., {x → -1.}}
```

In[275]:=
```
Solve[4 + 2 x + x^2 == 0, x]
```
Out[275]=

$\{\{x \to -1 - i\sqrt{3}\}, \{x \to -1 + i\sqrt{3}\}\}$

In[276]:=

$\int (4 + 2x + x^2)\, dx$

Out[276]=

$4x + x^2 + \frac{x^3}{3}$

In[277]:=

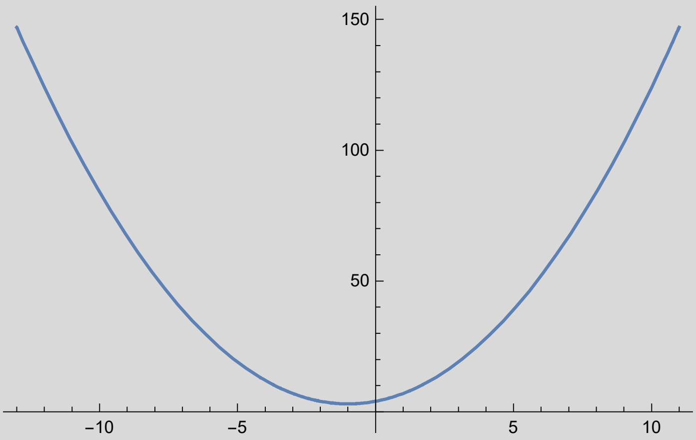
$\partial_x (4 + 2x + x^2)$

Out[277]=

$2 + 2x$

In[278]:=
```
FullSimplify[4 + 2 x + x^2]
```
Out[278]=
```
4 + x (2 + x)
```

In[279]:=
```
Plot[4 + 2 x + x^2, {x, -13., 11.}]
```
Out[279]=



Linear Algebra

Vectors

In the Wolfram Language, n-dimensional vectors are represented by lists of length n.

```
In[280]:= v1 = {1, 2, 3}
Out[280]= {1, 2, 3}
```

```
In[281]:= v2 = {a, b, c}
Out[281]= {a, b, c}
```

`v[[i]]` give the i^{th} element in the vector v.

```
In[282]:= v2[[2]]
Out[282]= b
```

c v (**space between c and v**) is scalar multiplication of c times the vector v

```
In[283]:= 2 v1
Out[283]= {2, 4, 6}
```

Calculate a vector’s norm:

```
In[284]:= Norm[v1]
```

Out[284]= $\sqrt{14}$

`Normalize[v]` gives the normalized form of a vector v.

```
In[285]:= Normalize[v1]
```

Out[285]= $\left\{\frac{1}{\sqrt{14}}, \sqrt{\frac{2}{7}}, \frac{3}{\sqrt{14}}\right\}$

Calculate the dot product of two vectors:

```
In[286]:= v1.v2
Out[286]= a + 2 b + 3 c
```

Calculate the cross product symbol:

In[287]:=
```
v1 × v2
```
Out[287]= $\{-3 b + 2 c, 3 a - c, -2 a + b\}$

Find the projection of a vector onto the x axis:

In[288]:=
```
Projection[{8, 6, 7}, {1, 0, 0}]
```
Out[288]= {8, 0, 0}

In[289]:=
```
Orthogonalize[{{8, 6, 7}, {1, 0, 0}}]
```
Out[289]= $\left\{\left\{\frac{8}{\sqrt{149}}, \frac{6}{\sqrt{149}}, \frac{7}{\sqrt{149}}\right\}, \left\{\sqrt{\frac{85}{149}}, -\frac{48}{\sqrt{12665}}, -\frac{56}{\sqrt{12665}}\right\}\right\}$

In[290]:=
```
VectorAngle[{1, 0}, {0, 1}]
```
Out[290]= $\frac{\pi}{2}$

Calculate the gradient of a vector:

In[291]:=
$\nabla_{\{x,y\}}\{x^2 + y, x + y^2\}$

Out[291]= {{2 x, 1}, {1, 2 y}}

Compute the divergence or curl of a vector field:

In[292]:=
```
Div[{f[x, y, z], g[x, y, z], h[x, y, z]}, {x, y, z}]
```
Out[292]= $h^{(0,0,1)}[x, y, z] + g^{(0,1,0)}[x, y, z] + f^{(1,0,0)}[x, y, z]$

Matrices

The Wolfram Language represents matrices as lists of lists A= $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$:

In[293]:=
```
A = {{1, 2}, {3, 4}}
```
Out[293]=
```
{{1, 2}, {3, 4}}
```

In[294]:=
```
MatrixForm[A]
```
Out[294]//MatrixForm=

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

You can construct a matrix with iterative functions :

In[295]:=
```
Table[x + y, {x, 1, 3}, {y, 0, 2}] // MatrixForm
```
Out[295]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

Compute the dot product of two matrices:

In[296]:=
```
{{1, 2}, {3, 4}}.{{a, b}, {c, d}}
```
Out[296]=

$$\left\{\left\{a + 2\,c,\ b + 2\left(9\,x^2 + 12\,x^3 + 4\,x^4 + 36\,x\,y + 48\,x^2\,y + 16\,x^3\,y + 36\,y^2 + 48\,x\,y^2 + 16\,x^2\,y^2\right)\right\},\ \left\{3\,a + 4\,c,\ 3\,b + 4\left(9\,x^2 + 12\,x^3 + 4\,x^4 + 36\,x\,y + 48\,x^2\,y + 16\,x^3\,y + 36\,y^2 + 48\,x\,y^2 + 16\,x^2\,y^2\right)\right\}\right\}$$

Find the determinant:

In[297]:=
```
Det[{{a, b}, {c, d}}]
```
Out[297]=

$$-b\,c + 9\,a\,x^2 + 12\,a\,x^3 + 4\,a\,x^4 + 36\,a\,x\,y + 48\,a\,x^2\,y + 16\,a\,x^3\,y + 36\,a\,y^2 + 48\,a\,x\,y^2 + 16\,a\,x^2\,y^2$$

Get the inverse of a matrix:

In[298]:=
```
Inverse[{{1, 1}, {0, 1}}]
```
Out[298]=
```
{{1, -1}, {0, 1}}
```

In[299]:=

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 2 & 2 \\ 5 & 1 & 7 \end{pmatrix}$$

Out[299]=
```
{{1, 2, 3}, {4, 2, 2}, {5, 1, 7}}
```

In[300]:=
```
Inverse[A] // MatrixForm
```
Out[300]//MatrixForm=

$$\begin{pmatrix} -\frac{2}{7} & \frac{11}{42} & \frac{1}{21} \\ \frac{3}{7} & \frac{4}{21} & -\frac{5}{21} \\ \frac{1}{7} & -\frac{3}{14} & \frac{1}{7} \end{pmatrix}$$

Trace

In[301]:=
```
Tr[A]
```
Out[301]=
```
10
```

Transpose

In[302]:=
```
Transpose[A] // MatrixForm
```
Out[302]//MatrixForm=

$$\begin{pmatrix} 1 & 4 & 5 \\ 2 & 2 & 1 \\ 3 & 2 & 7 \end{pmatrix}$$

Eigenvalues

In[303]:=
```
Eigenvalues[A] // N
```
Out[303]=
```
{9.76431, 2.19517, -1.95948}
```

Eigenvectors

In[304]:=
```
Eigenvectors[A] // N
```
Out[304]=
```
{{0.454513, 0.491744, 1.}, {-0.590403, -1.85282, 1.}, {-2.11901, 1.63558, 1.}}
```

gives a list `{values, vectors}` of the **eigenvalues** and **eigenvectors** of the square matrix m.

In[305]:=
```
Eigensystem[A] // N // MatrixForm
```
Out[305]//MatrixForm=

9.76431	2.19517	-1.95948
{0.454513, 0.491744, 1.}	{-0.590403, -1.85282, 1.}	{-2.11901, 1.63558, 1.}

`CharacteristicPolynomial[m,x]` gives the characteristic polynomial for the matrix m.

In[306]:=
```
CharacteristicPolynomial[{{a, b}, {c, d}}, x]
```
Out[306]=

$$-b\,c - a\,x + x^2 + 9\,a\,x^2 - 9\,x^3 + 12\,a\,x^3 - 12\,x^4 + 4\,a\,x^4 - 4\,x^5 + 36\,a\,x\,y - 36\,x^2\,y + 48\,a\,x^2\,y - 48\,x^3\,y + 16\,a\,x^3\,y - 16\,x^4\,y + 36\,a\,y^2 - 36\,x\,y^2 + 48\,a\,x\,y^2 - 48\,x^2\,y^2 + 16\,a\,x^2\,y^2 - 16\,x^3\,y^2$$

`A[[i]]` give the i^{th} row in the matrix A

In[307]:=
```
A[[1]]
```
Out[307]=
```
{1, 2, 3}
```

`A[[All, j]]` give the j^{th} column in the matrix A

In[308]:=
```
A[[All, 1]] // MatrixForm
```

Out[308]//MatrixForm=

$$\begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix}$$

`A[[i,j]]` give the i, j^{th} element in the matrix A

In[309]:=
```
A[[1, 3]]
```

Out[309]=
```
3
```

`Dimensions[`A`]` give the dimensions of a matrix represented by A

In[310]:=
```
Dimensions[A]
```

Out[310]=
```
{3, 3}
```

Complex Numbers

You can enter complex numbers in the Wolfram Language just by including the constant I, equal to $\sqrt{-1}$. Make sure that you type a capital I.

If you are using notebooks, you can also enter I as ⅈ by typing Esc *ii* Esc. The form ⅈ is normally what is used in output. Note that an ordinary i means a variable named *i*, not $\sqrt{-1}$.

This gives the imaginary number result 2 *i*.

In[311]:=
```
Sqrt[-4]
```
Out[311]=
```
2 ⅈ
```

This gives the ratio of two complex numbers.

In[312]:=
```
4 + 3 I
-------
 2 - I
```
Out[312]=
```
1 + 2 ⅈ
```

Here is the numerical value of a complex exponential.

In[313]:=
```
Exp[2 + 9 I] // N
```
Out[313]=
```
-6.73239 + 3.04517 ⅈ
```

In[314]:=
```
ⅇ^(2+9 ⅈ) // N
```
Out[314]=
```
-6.73239 + 3.04517 ⅈ
```

x + I y	the complex number $x + i\,y$.
Re[z]	real part
Im[z]	imaginary part
Conjugate[z]	complex conjugate z^* or $\bar{z}$
Abs[z]	absolute value |z|
Arg[z]	the argument φ in $\lvert z \rvert e^{i\varphi}$

Note:
*Abs[z] gives the phase angle of *z* in radians.
*The result from Arg[z] is always between $-\pi$ and $+\pi$.

AbsArg[z] gives the list {Abs[z], Arg[z]} of the number *z*.

Find the real part of a complex number:

In[315]:=
```
z = 2 + 3 I;
```

In[316]:=
```
Re[z]
```
Out[316]=
```
2
```

Find the imaginary part of a complex number:

In[317]:=
```
Im[z]
```
Out[317]=
```
3
```

In[318]:=
```
Conjugate[1 + I]
```
Out[318]=
```
1 - i
```

Use Esc conj Esc to conjugate expressions:

In[319]:=
```
(1 + I)*
```
Out[319]=
```
1 - i
```

Find complex conjugate of complex exponentials:

In[320]:=
```
Conjugate[Exp[I Pi / 4]]
```
Out[320]=

$e^{-\frac{i \pi}{4}}$

In[321]:=
```
Abs[z]
```
Out[321]=

$\sqrt{13}$

The result is given in radians:

In[322]:=
```
Arg[-1]
```
Out[322]=

π

In[323]:=
```
Arg[1 + I]
```
Out[323]=

$\frac{\pi}{4}$

The absolute value and argument of a complex number:

In[324]:=
```
AbsArg[1 + I]
```
Out[324]=
```
AbsArg[1 + i]
```

Differential Equations

The Wolfram Language can find solutions to ordinary, partial and delay differential equations (ODEs, PDEs and DDEs).

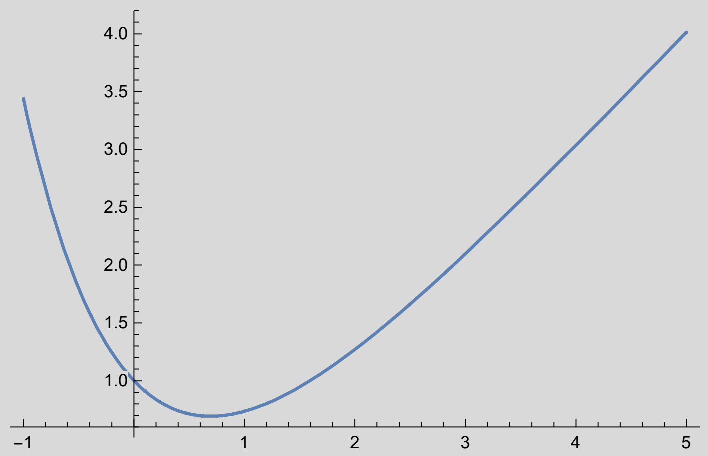
`DSolve[eqn,y[x],x]` solve a differential equation for y[x]
`DSolve[{eqn,y[x0]==y0},y[x],x]` solve a differential equation for y[x] with intial condition $y[x_0]==y_0$.
`DSolve[{eqn1,eqn2,…},{y1[x],y2[x],…},x]` solve a system of differential equations for $y_1[x]$, $y_1[x]$,....

Examples:

In[325]:=
```
DSolve[y'[x] + y[x] == x, y[x], x]
```
Out[325]=
$\{\{y[x] \to -1 + x + e^{-x} C[1]\}\}$

In[326]:=
```
sol = DSolve[{y'[x] + y[x] == x, y[0] == 1}, y[x], x]
```
Out[326]=
$\{\{y[x] \to e^{-x} (2 - e^{x} + e^{x} x)\}\}$

In[327]:=
```
y[x] /. sol
```
Out[327]=
$\{e^{-x} (2 - e^{x} + e^{x} x)\}$

In[328]:=
```
Plot[y[x] /. sol, {x, -1, 5}]
```
Out[328]=


In[329]:=
```
eqn = y''[x] + 4 y[x] == 0;
DSolve[eqn, y, x]
```
Out[330]=
$\{\{y \to Function[\{x\}, C[1] Cos[2 x] + C[2] Sin[2 x]]\}\}$

In[331]:=

```
eqns1 = {y'[t] == x[t] + y[t], x'[t] == x[t] - y[t]};
sol = DSolve[eqns1, {x, y}, t]
```

Out[332]=

```
{{x → Function[{t}, e^t C[1] Cos[t] - e^t C[2] Sin[t]],
  y → Function[{t}, e^t C[2] Cos[t] + e^t C[1] Sin[t]]}}
```

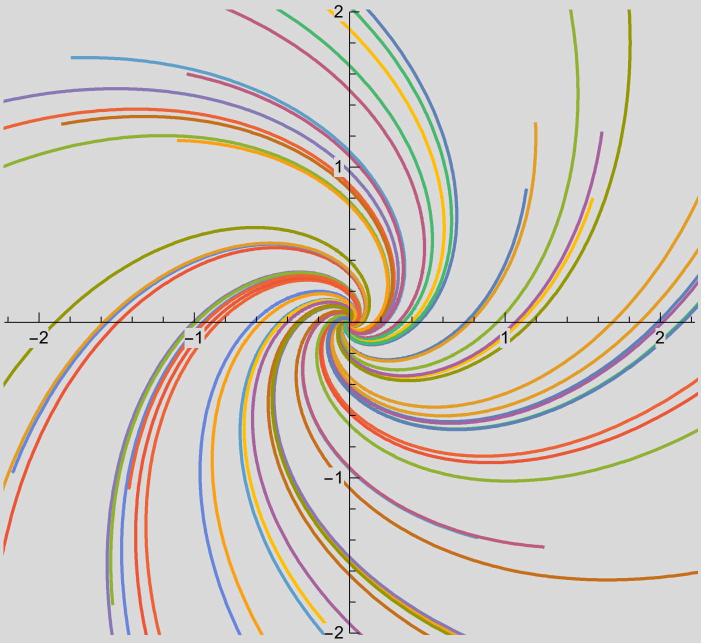
In[333]:=

```
particularsols =
  Partition[Flatten[Table[{x[t], y[t]} /. sol /. {C[1] → 1 / i, C[2] → 1 / j},
     {i, -20, 20, 6}, {j, -20, 20, 6}]], 2];
```

In[334]:=

```
ParametricPlot[Evaluate[particularsols], {t, -3, 3}, PlotRange → {-2, 2}]
```

Out[334]=



In[335]:=

```
Clear[x, y, z, t]
```

In[336]:=

```
linearsystem = {x'[t] == x[t] - 4 * y[t] + 1, y'[t] == 4 * x[t] + y[t], z'[t] == z[t]};
```

In[337]:=

```
initialvalues = {x[0] == 2, y[0] == -1, z[0] == 1};
```

In[338]:=

```
sol = DSolve[{linearsystem, initialvalues}, {x, y, z}, t]
```

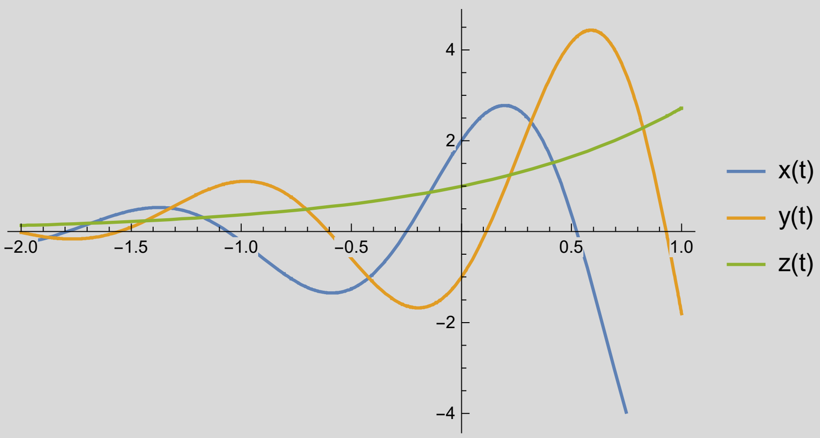
Out[338]=

$$\left\{\left\{x \to \text{Function}\left[\{t\}, \frac{1}{17}\left(35\, e^{t} \text{Cos}[4\,t] - \text{Cos}[4\,t]^2 + 21\, e^{t} \text{Sin}[4\,t] - \text{Sin}[4\,t]^2\right)\right], y \to \text{Function}\left[\{t\}, \frac{1}{17}\left(-21\, e^{t} \text{Cos}[4\,t] + 4\,\text{Cos}[4\,t]^2 + 35\, e^{t} \text{Sin}[4\,t] + 4\,\text{Sin}[4\,t]^2\right)\right], z \to \text{Function}[\{t\}, e^{t}]\right\}\right\}$$

In[339]:=

```
Plot[Evaluate[{x[t], y[t], z[t]} /. sol],
 {t, -2, 1}, PlotLegends → {"x(t)", "y(t)", "z(t)"}]
```

Out[339]=



DSolveValue[eqn,y[x],x] gives the value of **y[x]** determined by a symbolic solution to the ordinary differential equation *eqn* with independent variable x.
DSolveValue[{eqn,y[x_0]==y_0},y[x],x] solve a differential equation for y[x] with intial condition y[x_0]==y_0.
DSolveValue[{eqn_1,eqn_2,…},{y_1[x],y_2[x],…},x] solve a system of differential equations for y_1[x], y_1[x],….

In[340]:=
```
sol = DSolveValue[y'[x] + y[x] == x, y[x], x]
```
Out[340]=
```
-1 + x + e^-x C[1]
```

Use /. to replace the constant (C[1]=1):

In[341]:=
```
sol /. C[1] → 1
```
Out[341]=
```
-1 + e^-x + x
```

Or add conditions for a specific solution:

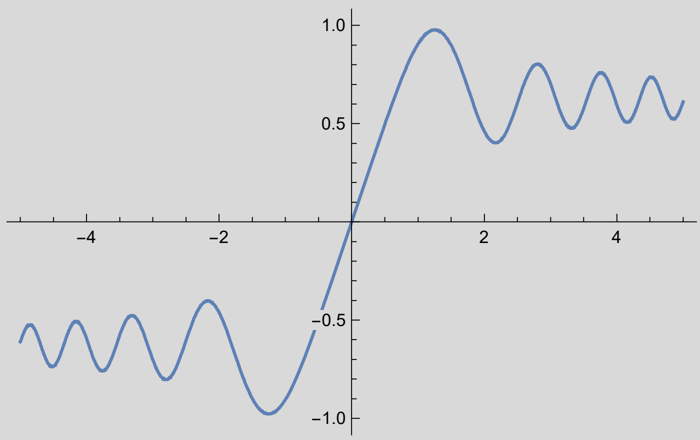
In[342]:=
```
DSolveValue[{y'[x] + y[x] == x, y[0] == -1}, y[x], x]
```
Out[342]=
```
-1 + x
```

NDSolveValue finds numerical solutions:

In[343]:=
```
sol1 = NDSolveValue[{y'[x] == Cos[x^2], y[0] == 0}, y[x], {x, -5, 5}]
```
Out[343]=
```
InterpolatingFunction[ Domain: {{-5., 5.}}  Output: scalar ][x]
```

You can plot this InterpolatingFunction directly:

In[344]:=
```
Plot[sol1, {x, -5, 5}]
```
Out[344]=



To solve systems of differential equations, include all equations and conditions in a list:
(Note that the line breaks have no effect.)

In[345]:=

```
{xsol, ysol} = NDSolveValue[{x'[t] == -y[t] - x[t]^2,
    y'[t] == 2 x[t] - y[t]^3, x[0] == y[0] == 1}, {x, y}, {t, 20}]
```

Out[345]=

```
{InterpolatingFunction[ Domain: {{0., 20.}}
                         Output: scalar ],
 InterpolatingFunction[ Domain: {{0., 20.}}
                         Output: scalar ]}
```

Visualize the solution as a parametric plot:

In[346]:=

```
ParametricPlot[{xsol[t], ysol[t]}, {t, 0, 20}]
```

Out[346]=