Derivatives are easy: diff(f, x, n)
arg1 is expression or function
arg2 is the variable
arg3 is the order of the derivative (defaults to 1 if missing)

```
(%i1) diff(x^5, x);
      diff(x^5, x, 2);
```

(%o1) $5\,x^4$

(%o2) $20\,x^3$

```
(%i3) ff(x) := sin(x)^2;
```

(%o3) $\mathrm{ff}\left(x\right):=\sin\left(x\right)^2$

```
(%i4) diff(ff(x), x);
      diff(ff(x), x, 5);
```

(%o4) $2\,\cos\left(x\right)\,\sin\left(x\right)$

(%o5) $32\,\cos\left(x\right)\,\sin\left(x\right)$

```
(%i6) trigreduce(%);
```

(%o6) $16\,\sin\left(2\,x\right)$

Multiple partial derivatives have same notation, with each variable followed by its order.

```
(%i7) g(x,y) := sin(x)^2 * (x^2+y^2)^2;
      diff1 : diff(g(x,y), x, 2, y, 1);
      diff2 : diff(g(x,y), y, 1, x, 2);
```

(%o7) $g\left(x,y\right):=\sin\left(x\right)^2\,\left(x^2+y^2\right)^2$

(%o8) $-8\,\sin\left(x\right)^2\,y\left(y^2+x^2\right)+8\,\cos\left(x\right)^2\,y\left(y^2+x^2\right)+8\,\sin\left(x\right)^2\,y+32\,x\,\cos\left(x\right)\,\sin\left(x\right)\,y$

(%o9) $-8\,\sin\left(x\right)^2\,y\left(y^2+x^2\right)+8\,\cos\left(x\right)^2\,y\left(y^2+x^2\right)+8\,\sin\left(x\right)^2\,y+32\,x\,\cos\left(x\right)\,\sin\left(x\right)\,y$

is(exp1 = exp2) returns true if two expressions are the same, false otherwise.
The below expression verifies that partial derivatives evaluated in opposite order are identical.

```
(%i10) is(diff1 = diff2);
```

(%o10) true

Take the nth derivative of exp(-x^2/2).
The term in () is proportional to the nth order Hermite polynomial (within factor of -1).

```
(%i11) diff(exp(-x^2), x, 10);
       %, factor;
```

(%o11) $1024\,x^{10}\,\%e^{-x^2}-23040\,x^8\,\%e^{-x^2}+161280\,x^6\,\%e^{-x^2}-403200\,x^4\,\%e^{-x^2}+302400\,x^2\,\%e^{-x^2}-30240\,\%e^{-x^2}$

(%o12) $32\left(32\,x^{10}-720\,x^8+5040\,x^6-12600\,x^4+9450\,x^2-945\right)\,\%e^{-x^2}$

```
(%i13) hermite(10,x);
```

STYLE-WARNING: redefining MAXIMA::SIMP-UNIT-STEP in DEFUN
STYLE-WARNING: redefining MAXIMA::SIMP-POCHHAMMER in DEFUN

(%o13) $-30240\left(-\frac{32\,x^{10}}{945}+\frac{16\,x^8}{21}-\frac{16\,x^6}{3}+\frac{40\,x^4}{3}-10\,x^2+1\right)$

We can also calculate Legendre polynomials directly from the derivative

```
(%i14) 1/2^10/10! * diff( (x^2-1)^10, x, 10), factor;
```

$$(\%o14)\ \frac{46189\,x^{10}-109395\,x^{8}+90090\,x^{6}-30030\,x^{4}+3465\,x^{2}-63}{256}$$

Compare this answer to stored Legendre function legendre_p(n,x)

```
(%i15) legendre_p(10,x), expand, factor;
```

$$(\%o15)\ \frac{46189\,x^{10}-109395\,x^{8}+90090\,x^{6}-30030\,x^{4}+3465\,x^{2}-63}{256}$$

Integration is easy, too. You can do definite and indefinite integrals. Definite integrals require two limits.

```
(%i16) integrate(x^2, x);
       integrate(x^2, x, 0, 2);
```

$$(\%o16)\ \frac{x^{3}}{3}$$

$$(\%o17)\ \frac{8}{3}$$

```
(%i18) integrate(x^2*exp(-5*x), x);
       integrate(x^2*exp(-5*x), x, 0, inf);
```

$$(\%o18)\ -\frac{\left(25\,x^{2}+10\,x+2\right)\,\%e^{-5\,x}}{125}$$

$$(\%o19)\ \frac{2}{125}$$

Other interesting integrals can be readily evaluated.
Use inf for infinity and -inf or minf for -infinity

```
(%i20) integrate(exp(-x^2), x, -inf, inf);
       integrate(x^2 * exp(-x^2), x, -inf, inf);
       integrate(sin(x)/x, x, minf, inf);
       integrate(1 / (x^2 + 4), x, minf, inf);
```

$$(\%o20)\ \sqrt{\pi}$$

$$(\%o21)\ \frac{\sqrt{\pi}}{2}$$

$$(\%o22)\ \pi$$

$$(\%o23)\ \frac{\pi}{2}$$

Expressions with unspecified constants have to be done carefully because wxMaxima sometimes needs to know whether the constant is real, positive, integer, etc.
Parameters default to being real, but noninteger.

Here's a dialog where an integral is requested and the user is asked to respond to questions about parameter. The second integral has two questions.

```
(%i24) integrate(exp(-a*x^2), x, -inf, inf);
Is  a  positive, negative, or zero?pos;
```

$$(\%o24)\ \frac{\sqrt{\pi}}{\sqrt{a}}$$

```
(%i25) integrate(1/(x^2+a^2), x, -inf, inf);
Is  a  zero or nonzero?nonzero;
Is  a  positive or negative?pos;
```

$$(\%o25)\ \frac{\pi}{a}$$

Use the assume() function to specify these things.
The arguments are called predicates and have the form a>0, b<a, etc.
Allowed operators are >, >=, <, <=, equal() notequal().
Note that equal and notequal are functions taking two arguments. Uugghh...

```
(%i26) assume(a>0, b>0, c>b, c>a, notequal(g,a), equal(b,e) );
       integrate(exp(-a*x^2), x, -inf, inf);
       integrate(1/(x+b)^2, x, 0, inf);
```

$$(\%o26)\ [a>0,\ b>0,\ c>b,\ c>a,\ \mathrm{notequal}(g,a),\ \mathrm{equal}(b,e)]$$

$$(\%o27)\ \frac{\sqrt{\pi}}{\sqrt{a}}$$

$$(\%o28)\ \frac{1}{b}$$

facts() lists all the assumptions you have specified

```
(%i29) facts();
```

$$(\%o29)\ [a>0,\ b>0,\ c>b,\ c>a,\ \mathrm{notequal}(g,a),\ \mathrm{equal}(b,e)]$$

functions lists all the functions you have specified

```
(%i30) functions;
```

$$(\%o30)\ [\mathrm{ff}(x),\ g(x,y)]$$

forget() tells wxmaxima to forget the specified assumptions, which are
specified the same way as assume(). This allows you to make wxMaxima
forget specific assumptions.

```
(%i31) forget(b < c, equal(b,e));
       facts();
```

$$(\%o31)\ [c>b,\ \mathrm{equal}(b,e)]$$

$$(\%o32)\ [a>0,\ b>0,\ c>a,\ \mathrm{notequal}(g,a)]$$

```
(%i33) forget(a>0);
       facts();
```

$$(\%o33)\ [a>0]$$

$$(\%o34)\ [b>0,\ c>a,\ \mathrm{notequal}(g,a)]$$

Use the is() function to test expressions

```
(%i35) is(5>0);
       is(a^2 > 0);
       factor(x^2-y^2);
       is( equal((x^2-z^2) , ( (x-z)*(x+z) ) ) );
       is(equal(x^2-y^2, -y^2+x^2) );
```

$$(\%o35)\ \mathrm{true}$$

$$(\%o36)\ \mathit{unknown}$$

$$(\%o37)\ -(y-x)(y+x)$$

$$(\%o38)\ \mathrm{true}$$

$$(\%o39)\ \mathrm{true}$$

You can declare a variable to be real, integer, complex, etc.
New variables are assumed to be real when defined.

```
(%i40) declare(n, integer);
       sin(n*%pi);
       cos(n*%pi);
(%o40) done
(%o41) 0
(%o42) (-1)^n
```

Now declare it to be noninteger

```
(%i43) declare(m, noninteger);
       sin(m*%pi);
       cos(m*%pi);
(%o43) done
(%o44) sin(%pi m)
(%o45) cos(%pi m)
```

You can calculate power series easily with the taylor function. Let's do a couple we already know

```
(%i46) taylor(sin(x), x, 0, 5);
```

$$(\%o46)/T/\quad x-\frac{x^3}{6}+\frac{x^5}{120}+\ldots$$

```
(%i47) taylor(log(x), x, 1, 7);
```

$$(\%o47)/T/\quad x-1-\frac{(x-1)^2}{2}+\frac{(x-1)^3}{3}-\frac{(x-1)^4}{4}+\frac{(x-1)^5}{5}-\frac{(x-1)^6}{6}+\frac{(x-1)^7}{7}+\ldots$$

Now let's expand a function that would take a *long* time by hand

```
(%i48) taylor(exp(sin(x)), x, 0, 13);
```

$$(\%o48)/T/\quad 1+x+\frac{x^2}{2}-\frac{x^4}{8}-\frac{x^5}{15}-\frac{x^6}{240}+\frac{x^7}{90}+\frac{31\,x^8}{5760}+\frac{x^9}{5670}-\frac{2951\,x^{10}}{3628800}-\frac{x^{11}}{3150}+\frac{181\,x^{12}}{14515200}+\frac{2417\,x^{13}}{48648600}+\ldots$$

Check the accuracy of the series approximation. First make a function from the series.
The construction ''(...) evaluates everything inside the (), and the := then assigns the
expression to the function. If you write df(x) := taylor(f(x), x, a, n), you end up
with a function set to taylor(...) which cannot be evaluated for constant arguments because
the constant is substituted for the variable everywhere.

```
(%i49) expsin(x) := ''( taylor(exp(sin(x)), x, 0, 13) );
```

$$(\%o49)/T/\quad \mathrm{expsin}(x):=1+x+\frac{x^2}{2}-\frac{x^4}{8}-\frac{x^5}{15}-\frac{x^6}{240}+\frac{x^7}{90}+\frac{31\,x^8}{5760}+\frac{x^9}{5670}-\frac{2951\,x^{10}}{3628800}-\frac{x^{11}}{3150}+\frac{181\,x^{12}}{14515200}+\frac{2417\,x^{13}}{48648600}+\ldots$$

It's much easier to define this function using define().

```
(%i50) define(expsin2(x), taylor(exp(sin(x)), x, 0, 13));
```

$$(\%o50)/T/\quad \mathrm{expsin2}(x):=1+x+\frac{x^2}{2}-\frac{x^4}{8}-\frac{x^5}{15}-\frac{x^6}{240}+\frac{x^7}{90}+\frac{31\,x^8}{5760}+\frac{x^9}{5670}-\frac{2951\,x^{10}}{3628800}-\frac{x^{11}}{3150}+\frac{181\,x^{12}}{14515200}+\frac{2417\,x^{13}}{48648600}+\ldots$$

Check that these functions are identical using is(exp1 = exp2). You can test many
conditions this way, e.g. is(a < b), is(a > b), is(a # b) (i.e., not equal)

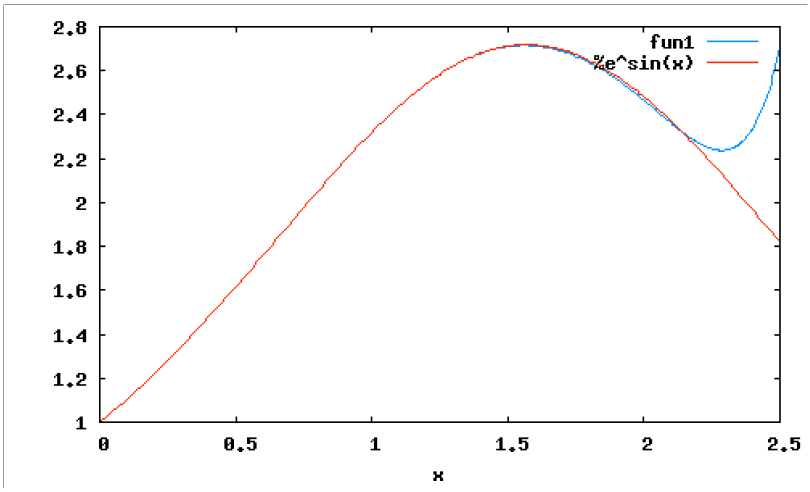
```
(%i51) is(expsin(x) = expsin2(x));
(%o51) true
```

Check the approximation against the exact value for x = 1.0

```
(%i52) float( expsin(1.0) - exp(sin(1.0)) );
rat: replaced 2.319767564771905 by 26225369/11305171 = 2.319767564771908
rat: replaced -2.31977682471585 by -26237170/11310213 = -2.31977682471586
(%o52) -9.259943948008587 10^-6
```

Let's plot the function and the series approximation. A growing divergence is apparent for x > 1.5

```
(%i53) wxplot2d([expsin(x), exp(sin(x))], [x, 0, 2.5]);
```



(%t53)

(%o53)

```
(%i54) float(expsin(2.0) - exp(sin(2.0)));
rat: replaced 2.46548496770719 by 14992799/6081075 = 2.46548496770719
rat: replaced -2.482577728015 by -22323329/8991996 = -2.482577728015
(%o54) -.01709276030781364
```

taylor can handle Laurent series too

```
(%i55) taylor(1/sin(x)^3, x, 0, 5);
```

$$(\%o55)/T/\quad \frac{1}{x^3}+\frac{1}{2\,x}+\frac{17\,x}{120}+\frac{457\,x^3}{15120}+\frac{3287\,x^5}{604800}+\ldots$$

```
(%i56) taylor(1/log(x), x, 1, 4);
```

$$(\%o56)/T/\quad \frac{1}{x-1}+\frac{1}{2}-\frac{x-1}{12}+\frac{(x-1)^2}{24}-\frac{19\,(x-1)^3}{720}+\frac{3\,(x-1)^4}{160}+\ldots$$

You can kill the definition of some variables

```
(%i57) a: 24;
       kill(a);
       a;
(%o57) 24
(%o58) done
(%o59) a
```

```
(%i60) ff(x) := x^2;
       b : 10;
       G : x^2 - x;
(%o60) ff(x):=x^2
(%o61) 10
(%o62) x^2-x
```

Get a list of all user defined functions, user defined values and user defined predicates

```
(%i63) functions;
       values;
       facts();
(%o63) [ff(x),g(x,y),expsin(x),expsin2(x)]
(%o64) [diff1,diff2,b,G]
(%o65) [b>0,c>a,notequal(g,a),kind(n,integer),kind(m,noninteger)]
```

Kill selective user assignments

```
(%i66) kill(ff);
       kill(b);
       kill(n);
       functions;
       values;
       facts();
(%o66) done
(%o67) done
(%o68) done
(%o69) [g(x,y),expsin(x),expsin2(x)]
(%o70) [diff1,diff2,G]
(%o71) [b>0,c>a,notequal(g,a),kind(m,noninteger)]
```

Kill all user assignments with kill(all). Nothing is remembered.

```
(%i72) kill(all);
       ff(x);
       b;
       g;
       functions;
       values;
       facts();
(%o0) done
(%o1) ff(x)
(%o2) b
(%o3) g
(%o4) []
(%o5) []
(%o6) []
```