DelVar

**DelVar** *Var*, deletes all members of the *Var* variable group (such as the statistics *stat.nn* results or variables created using the **LibShortcut()** function).

The dot (.) in this form of the **DelVar** command limits it to deleting a variable group; the simple variable *Var* is not affected.

```
DelVar aa.
getVarInfo()  aa.a "NUM" "[]"
             aa.b "NUM" "[]"
             aa.c "NUM" "[]"
```

```
DelVar aa.  Done
getVarInfo() "NONE"
```

delVoid()

**delVoid**(*List1*) ⇒ *list*

Returns a list that has the contents of *List1* with all empty (void) elements removed.

For more information on empty elements, see page 212.

```
delVoid()  delVoid({1, void, 3})  {1, 3}
```

derivative()

See **d()**, page 198.

deSolve()

**deSolve**(*1stOr2ndOrderODE, Var, depVar*) ⇒ a general solution

Returns an equation that explicitly or implicitly specifies a general solution to the 1st- or 2nd-order ordinary differential equation (ODE). In the ODE:

- Use a prime symbol (press **º**) to denote the 1st derivative of the dependent variable with respect to the independent variable.
- Use two prime symbols to denote the corresponding second derivative.

The prime symbol is used for derivatives within **deSolve()** only. In other cases, use **d()**.

The general solution of a 1st-order equation contains an arbitrary constant of the form c\(^k\), where k is an
integer suffix from 1 through 255. The solution of a 2nd-order equation contains two such constants.

Apply `solve( )` to an implicit solution if you want to try to convert it to one or more equivalent explicit solutions.

When comparing your results with textbook or manual solutions, be aware that different methods introduce arbitrary constants at different points in the calculation, which may produce different general solutions.

```
desolve(1stOrderODE and initCond, Var, depVar) => a particular solution

Returns a particular solution that satisfies 1stOrderODE and initCond. This is usually easier than determining a general solution, substituting initial values, solving for the arbitrary constant, and then substituting that value into the general solution.

initCond is an equation of the form:

depVar(initialIndependentValue) = initialDependentValue

The initialIndependentValue and initialDependentValue can be variables such as x0 and y0 that have no stored values. Implicit differentiation can help verify implicit solutions.

```
desolve(2ndOrderODE and initCond1 and initCond2, Var, depVar) => particular solution

Returns a particular solution that satisfies 2nd Order ODE and has a specified value of the dependent variable and its first derivative at one point.

For initCond1, use the form:

depVar(initialIndependentValue) = initialDependentValue

For initCond2, use the form:

depVar(initialIndependentValue) = initial1stDerivativeValue

```
desolve(2ndOrderODE and bndCond1 and