Remarques importantes :

1) Ces résumés il faut les apprendre par coeur, mais il faut surtout apprendre à les utiliser sur des exemples, et apprendre à utiliser le help.

Comment utiliser le help ?

2) par exemple : si vous travaillez sur des matrices, tapez :
>?**matrix et valider (ou sélèctionner le mot matrix et aller dans le help et sélèctionner "help on matrix") ; un petit cours sur les matrices s'affichera, en bas de cette même page vous trouverez des exemples sur le sujet que vous étudiez sur le help.**

3) Si vous voulez directement des exemples sur "dsolve" (par exemple) vous tapez :
>???**dsolve et validez**

4) Si vous ne vous souvenez pas de la syntaxe exacte d'une instruction, cherchez dans le help, en utilisant le mot essentiel de l'instruction.

Algèbre linéaire

a) Généralités

>*with*(*linalg*); # Pour mettre en mémoire la bibliothèque de l'algèbre linéaire.
>$v := vector([1,2,3])$; # v est le vecteur $(1,2,3)$.
>$w := vector(5, x \text{ -> } f(x))$; # w est le vecteur $(f(1), f(2), f(3), f(4), f(5))$. (Exemples)
>$M := matrix([[1,2,3],[a,b,c]])$; # M est la matrice de lignes $[1,2,3]$ et $[a,b,c]$.
>$M := matrix(2,3,[1,2,3,a,b,c])$; # même chose que la ligne précèdente.
>$M := matrix(n,p,(i,j) \text{ -> } f(i,j))$; # c'est la matrice $(f(i,j))$ de type (n,p).
>$rowdim(M)$; $coldim(M)$; # nombre de lignes (resp colonnes) de M.
>$M := n \text{ -> } matrix(n,n,(i,j) \text{ -> } f(i,j))$; # Fonction matrice.
>$M(1)$; $M(3)$; # voir exemples.
>$M := (n,p) \text{ -> } matrix(n,p,(i,j) \text{ -> } f(i,j))$; # fonction matrice.
>$M(1,2)$; $M(2,5)$; $M(4,5)$; voir exemples.
>$diag(a,b,c,d,alpha,\ldots)$; # matrice diagonale...
>$diag(seq(f(i), i = 1..n))$; # $diag(f(1), f(2), \ldots, f(n))$.
>$band([1,2,3],5)$; # matrice tridiagonale d'ordre 5.
>$vandermonde([a,b,c,d])$; # Matrice de vandermonde.
>$Eq := [2*x+3*y-2*z = 3, x+y+z = 1, 5*x+4*y+2*z = 0]$;
>$genmatrix(Eq,[x,y,z])$; # matrice du système linéaire Eq.
>$M := matrix(\ldots)$; # Chargez une matrice de taille assez grande.
>$N := submatrix(M,[i1,i2,..,ip],[j1,j2,..,jq])$; # Matrice extraite de M en ne gardant que les lignes $i1, i2, \ldots, ip$ et les colonnes $j1, j2, \ldots, jq$.
>$N := submatrix(M, i1..ip, [j1,j2,..,jq])$; # si $i1, i2, .., ip$ sont successives.
>$N := submatrix(M,[i1,i2,..,ip], j1..jq)$; # si $j1, j2, .., jq$ sont successives.
>$P := delrows(M, i1..i2)$; # Matrice obtenue en supprimant à M les lignes $i1$ à $i2$.
>$Q := delcols(M, j1..jq)$; # Matrice obtenue en supprimant à M les colonnes $j1$ à $j2$.
>$concat(A,B)$; # juxtaposition horizontale des matrices (vecteurs) A et B. (Exemples)
>$stackmatrix(A,B)$; # juxtaposition verticale des matrices (vecteurs) A et B. (Exemples)

b) **Calculs vectoriels et matricielles**

>$scalarmul(U,x)$; # le vecteur (la matrice) $x.U$. (Exemples)
>$W := alpha * U + beta * V$; # Pour des matrices ou des vecteurs.
>$evalm(W)$; # valeur de W. (Exemples)
>$evalm(alpha * U + beta * V)$; # directement. (Exemples)
>$transpose(A)$; # matrice transposée de A. (Exemples)
>$transpose(U)$; # ça ne marche pas pour un vecteur. (Exemples)
>$trace(A)$; # pour une matrice carrée. (Exemples)
>$multiply(A,U)$; # produit de deux matrices ou d'une matrice par un vecteur. (Exemples)
>$inverse(A)$; # inverse de A lorsque A est inversible. (Exemples)
>$evalm(A\hat{}(-2))$; $evalm(A\hat{}(-3))$; # puissances négatives de A.
>$A\& * B$; # une autre façon d'écrire la multiplication des matrices. (NB: la notation $A * B$ pour les matrices conduit à des erreurs, voir $A * B - B * A$ et $P\hat{}(-1) * A * P$).

c) **Sous espaces vectoriels**

>$a := vector([1,2,3,4])$; $b := vector([1,3,5,7])$; $c := vector([1,1,2,3])$; $d := vector([1,4,6,8])$; # vecteurs à composantes rationnels.
>$Base := basis([a,b,c,d])$; # une base de $vect(a,b,c,d)$.
>$Base := basis(\{a,b,c,d\})$; # même chose que la ligne précèdente.
>$nops(Base)$; # dimension de $vect(a,b,c,d)$.
>$kernel(M)$; # une base du noyau de la matrice M.
>$kernel(M,k)$; k; # k prend la valeur de la dimension de $\ker(M)$.
>$colspace(M,'r')$; r; # donne une base de l'espace vectoriel engendré par les colonnes.
>$rowspace(M,'r')$; r; # r prend la valeur de la dimension
>$E1 := [vector([1,2,0]), vector([2,1,3])]$; # $E1 = vect((1,2,0),(2,1,3))$.
>$E2 := [vector([1,2,1]), vector([-1,1,1]), vector([-2,-1,0])]$;
>$intbasis(E1,E2)$; # une base de $E1 \cap E2$.
>$sumbasis(E1,E2)$; # une base de $E1 + E2$.

d) **Opérations élèmentaires**

>$swapcol(A,i,j)$; $swaprow(A,i,j)$; # permuter les colonnes (lignes) i et j de M. Exemples
>$mulcol(A,k,x)$; $mulrow(A,k,x)$; # multiplier la colonne (ligne) k par x. Exemples
>$addcol(A,i,j,x)$; $addrow(A,i,j,x)$; # $C_j \leftarrow C_j + xC_i$ ($L_j \leftarrow L_j + xL_i$) Exemples
>$A := matrix(4,3,(i,j) -> i + j - 1)$;
>$A1 := pivot(A,1,1)$; # éliminer les $A_{i,1}$ $i \neq 1$ par A_{11}.
>$A2 := pivot(A1,2,2)$; # éliminer les A_{i2} $i \neq 2$ par A_{22}.
>$C := pivot(A,1,1,2..4)$; pivot partiel par A_{11} sur les lignes 2 à 4.
>$gausselim(A)$; # retourne la réduite de Gauss de A.

e) **Valeurs propres et vecteurs propres**

>$det(M)$; $rank(M)$; # déterminant de M, rang de M. (Exemples)
>$P := charpoly(A,X)$; # $(-1)^n P_A(X)$. (Exemples)
>$Q := minpoly(A,X)$; # polynôme minimal de A.
>$eigenvals(M)$; # valeurs propres de M.
>$eigenvects(M)$; # vecteurs propres de M avec valeurs propres et leurs mutiplicités.
>$Eigenvals(M)$; $evalf(\%)$; # valeurs approchées des valeurs propres de M.
>$d := jordan(M,'P')$; $P := evalm(P)$; # La réduite de Jordan de M, avec P la matrice de passage (c'est à dire : $d = P^{-1}MP$). En particulier c'est une trigonalisation de M, et dans le cas où M est diagonalisable c'est une diagonalisation de M.

$>dn := evalm(map(x -> x * *n, d));$ # d^n quand d est diagonale.
$>Mn := multiply(P, dn, inverse(P));$ # Puissance $n^{ème}$ de M. Appliquer sur des exemples.
$>exponential(M);$ # $exp(M)$. Appliquer sur des exemples qui sont diagonalisables.
$>exponential(M, t);$ # $exp(tM)$. Appliquer sur des exemples qui sont diagonalisables et sur des systèmes linéaires homogènes avec et sans condition initiale.

f) **Orthogonalité**

$>with(linalg);$ # ...
$>u := vector([1, -1, 1]);\ v := vector([1, 2, 1]);$
$>dotprod(u, v);$ # Produit scalaire de u et v.
$>norm(u, 2);$ # norme euclidienne de u.
$>crossprod(u, v);$ # produit vectoriel de u et v.
$>U := vector([1, -1, 2, 1]);\ V := vector([1, 1, 1, 1]);\ W := vector([0, 1, 1, 0]);$
$>GramSchmidt([U, V, W]);$ # orthogonaliser par le procédé de Schmidt.
$>GramSchmidt([U, V, W], `normalized`);$ # orthonormaliser par le procédé de Schmidt.
$>A := matrix(2, 2, [sqrt(2)/2, sqrt(2)/2, -sqrt(2)/2, sqrt(2)/2]);$
$>orthog(A);$ # A est-elle orthogonale ? Essayer d'autres exemples