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.

1) Fonctions usuelles

>exp(x); # exponentielle de x.
>ln(x); # logarithme népérien de x.
>log[10](x); # logarithme décimal de x.
>log[b](x); # logarithme de base b de x.
>sin(x);cos(x); tan(x); cot(x); # sinus, cosinus, tangente, et cotangente trigonométriques.
>sinh(x); cosh(x); tanh(x); coth(x); # sinus, cosinus, tangente, et cotangente hyperboliques.
>arcsin(x); arccos(x); arctan(x); arccot(x); # reciproques des fonctions trigonométriques.
>arctan(y,x); # retourne l'argument de $(x+iy)$.
>arcsinh(x); # Fonction argument sinus hyperbolique, c'est la fonction reciproque de sinh(x) sur $\mathbb{R}$. $[\forall x \in \mathbb{R} \quad \arcsin h(x) = \arg sh(x) = \ln(x+\sqrt{x^2+1})]$.
>arccosh(x); # Fonction argument cosinus hyperbolique, c'est la fonction reciproque de cosh(x) sur $[1+\infty[$. $[\forall x \in [1+\infty[\quad \operatorname{arccosh}(x) = \arg ch(x) = \ln(x+\sqrt{x^2-1})]$.
>arctanh(x); # Fonction argument tangente hyperbolique, c'est la fonction reciproque de tanh(x) sur $]-1,1[$. $[\forall x \in]-1,1[\quad \operatorname{arctanh}(x) = \arg th(x) = \frac{1}{2}\ln(\frac{1+x}{1-x})]$.
>arccoth(x); # Fonction argument cotangente hyperbolique, c'est la fonction reciproque de coth(x) sur $\mathbb{R}\backslash[-1,1]$. $[\forall x \in \mathbb{R}\backslash[-1,1] \quad \operatorname{arccoth}(x) = \arg\coth(x) = \frac{1}{2}\ln(\frac{x+1}{x-1})]$.

2) Dérivation des fonctions d'une variable réelle

>*diff*($f(x),x$); # dérivée première de f au point x. (exemples)
>*diff*($f(x),x,x$); # dérivée seconde de f au point x. //
>*diff*($f(x),x,x,x$); # dérivée troisième de f au point x. //
>*diff*($f(x),x$\$4); # dérivée quatrième de f au point x. //
>*diff*($f(x),x$\$$n$); # dérivée $n^{ème}$ de f au point x. //
>(D@@n)(f); # la fonction dérivée $n^{ème}$ de la fonction f. //
>f := x -> 1/(1 + x^2);
>g := (D@@3)(f); # Appliquer à d'autres exemples.

3) Équations différentielles

>$E1 := 2 * diff(y(x),x) + y(x) = cos(x)$; # équation linéaire du premier ordre..
>$y1(x) := dsolve(E1, y(x))$; # solutions.
>$z1(x) := dsolve(\{E1, y(0) = 0\}, y(x))$; # solution avec la condition initiale. (Exemples)
>$E2 := a * diff(y(x),x,x) + b * diff(y(x),x) + c * y(x) = f(x)$; # équation linéaire du secod ordre avec second membre à coefficients constants.
>$y2(x) := dsolve(E2, y(x))$; # Solution.
>$z2(x) := dsolve(\{E2, y(u1) = v1, y(u2) = v2\}, y(x))$; # Solution avec la condition initiale. (Exemples)
>$z2(x) := dsolve(\{E2, y(u) = v, D(y)(u) = w\}, y(x))$; # Solution avec la condition initiale. (Exemples)
># **ATTENTION** : **Maple tient mal compte des intervalles**, **et la solution qu'il donne n'est souvent valable que sur l'un des intervalles de définition de l'équation**.
>$E3 := y(x) * diff(y(x),x) + x - 1$; # $(E3)\ yy' + x - 1 = 0$.
>$y3(x) := dsolve(E3, y(x))$;
>$z3(x) := dsolve(\{E3, y(1) = 1\}, y(x))$;
>$E4 := diff(y(x),x) = -y(x)/sqrt(4 - y(x)^2)$;
>$y4 := dsolve(E4, y(x))$; # ça ne donne peut etre rien.
>$z4 := dsolve(\{E4, y(0) = 1\}, y(x), numeric)$; # resolution numérique (approchée) de $E4$. ça demande tj des conditions initiales
>$z4(4)$; $z4(3)$; valeurs particuliéres approchées
>$f := t -> subs(z4(t), y(x))$; # charger la solution particulière dans une fonction.
>$plot(f, 3..10, y = 0..2)$; courbe intégrale de l'équation $E4$ sur $[3, 10]$.
>$Eq := diff(y(x),x,x) + sin(y(x)) = 0$; # équation du pendule.
>$S := dsolve(\{Eq, y(0) = 0, D(y)(0) = 0.5\}, y(x), numeric)$;
>$Y := t -> subs(S(t), y(x))$; $Y1 := t -> subs(S(t), diff(y(x),x))$;
>$plot([Y, Y1], 0..10, color = [red, blue])$; # Position et vitesse du pendule.
># voir plus de détails sur les équations différentielles sur le help.

4) Champs de vecteurs

>$with(linalg)$;
>$F := [f(x,y), g(x,y)]$; $U := [x,y]$; # une fonction à deux variables.
>$diverge(F, U)$; # divegence de F. Exemples.
>$M := jacobian(F, U)$; # matrice jacobienne de la fonction F. càd : $\begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} \end{pmatrix}$.
>$det(M)$; # le jacobian de$F$. càd : $\frac{D(f(x,y),g(x,y))}{D(x,y)}$.
>$restart$; # **refaire ce qui précède** (**diverge**,**jacobian**) **sur des exemples en dimension** 3.
>$f := arctan(y/x)$;
>$G := grad(f, [x,y])$; # gradient de f au point (x,y).
>$F := [2 * x + y + z, x + 2 * y + z, x + y + 2 * z]$;
>$potential(F, [x,y,z], `V`)$; # F dérive t-il d'un potentiel scalaire ?
>V; # Le potentiel scalaire quand il existe. Exemples.
>$restart$; # équations aux dérivées partielles.
>$E1 := x * diff(f(x,y),x) - y * diff(f(x,y),y) = 1$;
>$pdsolve(Edp, f(x,y))$;# faires d'autres exemples d'équations aux dérivées partielles.

5) Les surfaces

a) **Surfaces implicites d'équation** : $z = f(\mathbf{x}, \mathbf{y})$

>$f := (x,y) -> f(x,y)$;
>$plot3d(f(x,y), x = a..b, y = c..d)$; # Exemples.

>*plot3d*(*f*(*x*,*y*),*x* = *a*..*b*,*y* = *c*..*d*,*axes* = *frame*);
>*plot3d*(*f*(*x*,*y*),*x* = *a*..*b*,*y* = *c*..*d*,*axes* = *boxed*);
>*plot3d*(*f*(*x*,*y*),*x* = *a*..*b*,*y* = *c*..*d*,*view* = *e*..*f*,*axes* =…,*style* = *patchcontour*);
>*f* := (*x*,*y*) -> *cos*(*x* ∗ ∗2) + *cos*(*y*);
>*plot3d*(*f*(*x*,*y*),*x* = 1..3,*y* = −3..3,*axes* = *frame*,*style* = *point*); # voir aussi pour *axes* = *boxed* et les styles : *wireframe* ; *patchnogrid*)

b) **Surfaces implicites d'équation** : $f(\mathbf{x},\mathbf{y},\mathbf{z}) = \mathbf{0}$

>*with*(*plots*,*implicitplot3d*); # Avant toute utilisation des surfaces implicites.
>*implicitplot3d*(*x* ∗ ∗2 + *y* ∗ ∗2 − *z* ∗ ∗2 − 1,*x* = −2..2,*y* = −2..2,*z* = −2..2);
>*f* := (*x*,*y*,*z*) -> *x* ∗ ∗2 + *y* ∗ ∗2 − *z* ∗ ∗2 − 1;
>*implicitplot3d*(*f*,−2..2,−2..2,−2..2); # Une autre façon de faire la même chose.
> # faire la tracé pour différents styles et axes. Autres exemples.

c) **Nappes paramétrées en cartésien**

>*p* := *u* ∗ *cos*(*v*) − *u* ∗ *sin*(*v*) + *u*;
>*q* := *u* ∗ *cos*(*v*) + *u* ∗ *sin*(*v*) − *u*;
>*r* := *u* ∗ *sin*(*v*) + *u*;
>*plot3d*([*p*,*q*,*r*],*u* = 0..2 ∗ *Pi*,*v* = 0..2 ∗ *Pi*,*axes* =…,*style* =…); # faire la tracé pour différents styles et axes. Autres exemples.

d) **Nappes paramétrées en cylindrique**

>*p* := *z* + *cos*(*theta*);
>*plot3d*(*p*,*theta* = 0..2 ∗ *Pi*,*z* = 0..5,*coords* = *cylindrical*); # faire la tracé pour différents styles et axes. Autres exemples.
>*r* := 2 ∗ *v* − sin(*theta*);
>*z* := *v* ∗ *cos*(*theta*);
>*plot3d*([*r*,*theta*,*z*],*theta* = 0..2 ∗ *Pi*,*v* = −1..1,*coords* = *cylindrical*); # faire la tracé pour différents styles et axes. Autres exemples.

e) **Nappes paramétrées en sphérique**

>*r* := 1 + *cos*(*theta*) ∗ *cos*(*phi*);
>*plot3d*([*r*,*theta*,*phi*],*theta* = 0..2 ∗ *Pi*,*phi* = 0..2 ∗ *Pi*,*coords* = *spherical*); # faire la tracé pour différents styles et axes. Autres exemples.