

Le TABLEUR de ClarisWork (mode d'emploi résumé)

Ce tableur, appelé *feuille de calcul*, est très voisin des tableurs classiques (Lotus, Excel) ; il possède cependant quelques particularités, par exemple dans la fonction de **copie multiple**.

Comme les autres, il crée des **tableaux dynamiques**, comprenant des valeurs initiales ou *données* (équivalentes aux variables dans les logiciels de calcul en langage évolué) et de valeurs calculées à partir des précédentes grâce à des *formules*. Le changement des *données* initiales fait recalculer le tableau tout entier.

Noter que la *casse* n'est pas significative, hormis dans les textes : il est indifférent d'écrire en majuscules ou en minuscules les fonctions, les adresses ou les noms des cases (Claris les remet en minuscules).

Définitions

Le tableau est formée de **lignes** et de **colonnes**. Les lignes – *horizontales* – sont repérées par des numéros (de 1 à 501). Les colonnes – *verticales* – sont repérées par des lettres (A, B, C, ... Y, Z, AA, AB, AC ...).

On appellera **rangée** indifféremment une ligne ou une colonne.

On appellera **case** la *cellule* élémentaire du tableau. Située à l'intersection d'une ligne et d'une colonne, elle a pour **adresse** le repère de sa colonne joint à celui de sa ligne (par exemple, la première case s'appelle A1).

(On verra qu'exprimée sous cette forme, il s'agit en fait d'une **adresse relative**).

Présentation

On peut dilater à l'écran le tableau en modifiant le nombre situé en bas à gauche de la fenêtre (effet non imprimable).

Pour dilater des *rangées*, étirer leur frontières, ou bien via le menu *Format, largeur* ou *hauteur...* (effets imprimables).

Contenu des cases

Texte. Les cases peuvent contenir du texte (avec polices, corps et enrichissements de Windows). Le texte peut déborder de la case, il sera néanmoins correctement imprimé.

Nombre. Toute case peut contenir un nombre, dont on peut ajuster la taille et la police comme pour le texte, ainsi que le format via le menu *Format, Nombre* (sélectionner *Fixe, précision 2* pour les comptes financiers).

Les **nombres réels** s'écrivent avec une virgule, ou le signe % (ex., 20% équivaut à 0,2).

Formule. Une case peut recevoir une formule comme il sera détaillé ci-dessous.

Nom : on peut *affecter* un nom à une case quelconque via l'icône ▼ (à gauche de *fx*) : *Définir un bloc*, donner un nom (il se comportera comme une adresse absolue et ne s'utilise que dans des formules).

Une case est dite **active** quand on l'a sélectionnée (avec les touches flèches ou la souris). Elle est alors encadrée.

Ligne d'état. Modifications

La ligne de menus située au-dessus des lettres repère des colonnes est la *ligne d'état*. On y voit l'adresse de la *case active*, puis son contenu. C'est à cet endroit qu'il est préférable d'effectuer des modifications dans le **contenu de la case active**.

Pour modifier **une rangée**, il faut d'abord la sélectionner en cliquant sur son repère (ou en-tête). Puis,

- effacer une rangée : menu contextuel ou *.Edition, effacer* (on n'efface que le contenu).

- détruire une rangée : menu *Options, supprimer* (on supprime la rangée, les suivantes se décalent).

- insérer une rangée : menu *Options, Insérer* (attention aux décalages des rangées suivantes).

Formules

Elles permettent de faire effectuer des calculs automatiques.

Une formule commence toujours par le signe = (c'est ce qui la distingue du *texte*), suivi d'une *opération*.

L'**opération** s'écrit avec des *opérateurs*, des *fonctions*, des nombres, des adresses ou des noms de case.

On a déjà défini ce qu'étaient les adresses de case, les noms de case, les nombres.

Le tableur **remplace les adresses par le contenu de la case citée**, calcule la formule et affiche le résultat dans la case (sauf en cas de réglage contraire, via le menu *Feuille, afficher, formules.*).

Opérateurs

Ce sont les symboles classiques : + (addition) ; - (soustraction) ; * (multiplication) ; / (division), plus ceux de groupement (les parenthèses), celui d'exponentiation (^) et celui de l'inversion algébrique (-).

Ex : si on écrit 5,5 dans A1, puis 3 dans A2 et enfin $=2*-A1+A2$ dans A4, la case A4 affichera -8 soit $3 - (2*5,5)$.

Fonctions (voir liste en annexe)

Un grand nombre de fonctions sont disponibles. On peut écrire leur nom à la main ou bien le sélectionner dans le menu déroulant *fx*. Puis écrire ensuite les **arguments** corrects (nombres, adresses...) entre les parenthèses de la fonction.

Blocs

Certaines fonctions exigent des *blocs* (ensemble de cases) comme arguments (la fonction *somme* par exemple).

On désigne un bloc par les adresses de ses deux cases extrêmes séparées par deux points. Exemples :

B4..B8 (bloc colonne) désigne l'ensemble des 5 cases B4, B5, B6, B7 et B8.

D3 .. G3 (bloc ligne) désigne l'ensemble des 4 cases D3, E3, F3 et G3.

B2..D5 (bloc multiligne) désigne l'ensemble des 12 cases B2, C2, D2, B3, C3, D3, B4, C4, D4, B5, C5, D5.

Opérateurs logiques

Une **expression logique** est une expression ou opération ne pouvant avoir que les valeurs *vrai* ou *faux*.

On l'obtient avec des fonctions logiques ou avec des opérateurs de comparaison : <, >, <=, >=, <> (différent de).

Copie

C'est une **fonction essentielle** dans les tableurs. Il faut bien comprendre son fonctionnement.

On sélectionne une *case source* ; on place son contenu dans le *presse-papier*, comme à l'ordinaire ; puis on demande sa copie (son *collage*) dans une *case cible*.

Si la source ne contient pas d'*adresse*, tout se passe comme dans la copie ordinaire.

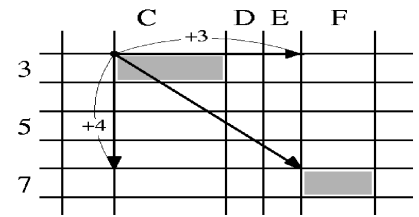
Si la source contient des adresses, il faut bien distinguer *adresses relatives* et *adresses absolues*.

Les **adresses absolues** (ainsi que les **noms de cases**) sont recopiées telles quelles (pas de transposition).

Les **adresses relatives** sont **transposées**, de façon à conserver la même configuration d'adresses entre la source et la cible.

Exemple : si C3 contient D4 et si on copie C3 (case source) en F7 (case cible), il apparaîtra G8 en F7 au lieu de D4.

($F7 = C3 + [3,4]$, $D4 + [3,4] = G8$). On dit qu'on conserve la translation, ou plutôt le vecteur de translation $[3,4]$.



Adresse relative et adresse absolue

L'adresse relative de la case Xn s'écrit Xn (exemple : A2, G8, B101). Dans une copie, elle est toujours **transposée**.

L'adresse absolue de la case Xn s'écrit \$X\$ ou \$X\$ ou X\$. Le symbole \$ indique que le repère qui suit ne doit pas être transposé lors d'une copie (utiliser un nom de case assure également la fixité de son adresse).

L'emploi d'un nom pour désigner une case entraîne également, en cas de copie, la fixité de l'adresse.

Copie multiple

On peut également recopier une fraction de rangée dans une autre (ligne vers le bas, ou colonne vers la gauche).

La copie des adresses obéit alors aux mêmes règles que ci-dessus.

On peut aussi copier une case ou un bloc dans un bloc cible **plus étendu** que le bloc source.

Sélectionner d'abord la source (case ou bloc) et étendre la sélection vers le bas ou vers la droite.

Puis menu *Options, Remplir vers la droite, ... vers le bas*

Tout se passe comme si on avait opéré plusieurs copies successives. Il y a là aussi translation des adresses relatives.

Graphique

On obtient facilement un graphique rudimentaire traduisant le contenu d'un bloc de cases.

Sélectionner un ou plusieurs blocs (selon le type de graphique), puis menu *Feuille, créer un graphique, ...*

Choisir le type de graphique désiré (cartésien, à barres, nuages de points, camembert...)

On peut ajouter quelques améliorations, via le menu contextuel ou *Edition*, puis la branche *Infos graphiques*.

Impression

Bien vérifier les polices, leur corps, les enrichissements, la hauteur des lignes, la largeur des colonnes.

Le quadrillage pointillé visible à l'écran ne sera pas imprimé (ni les repères des rangées).

On peut ajouter un quadrillage ou bien entourer d'un *filet* seulement certaines cases (sélection , puis outils à droite).

Sélectionner la partie du tableau à imprimer, puis menu *Fichier, Imprimer*.

Fonctions disponibles

Au nombre de 103. Voir en annexe la description de la plupart d'entre elles.

Exemples de tableaux (fichiers indépendants)

1. *Budget.cwk*. Calcul d'un budget mensuel moyen familial (très simple)

On donne le montant des versements obligatoires avec leur fréquence dans l'année. On obtient un budget mensuel.

2. *CalcInt.cwk*. Remboursement de prêt (avec intérêt composé, utilisation de la fonction SI).

On donne le taux d'intérêt annuel, le montant du prêt, celui du remboursement mensuel, le tableau calcule la durée du prêt.

Le tableau n'est pas fini, étendre les lignes vers le bas. S'arrêter quand tout le prêt est remboursé.

3. *CalcPi.cwk*. Calcul du nombre π par statistiques (exercice simple, mais méthode ardue, dite de *Monte-Carlo*).

Si on fait tomber un point au hasard dans un cercle de rayon 1 inscrit dans un carré de côté 2, sa probabilité de tomber à l'intérieur du cercle plutôt qu'à l'extérieur est égale au rapport de la surface du cercle divisée par celle du carré, soit $\pi/4$.

Le point est ici l'extrémité d'un vecteur de composantes aléatoires x et y comprises entre 0 et 1. On admet qu'il *tombe* dans le cercle si le carré de son module x^2+y^2 est inférieur ou égal à 1. En opérant un très grand nombre de fois, le nombre de coups favorables tend vers $\pi/4$, si la fonction utilisée (ALEA) est bien aléatoire (or ce n'est pas vrai, comme le montre facilement un graphique sur la suite des valeurs obtenues). Étendre le tableau vers le bas.

4. *CalcImpo.cwk*. Calcul de l'impôt IRPP (pour des contribuables ayant une situation ordinaire).

On se borne à la déclaration des salaires et pensions, on tient compte des tranches et du quotient familial.

Les 103 fonctions du tableur Claris-Work

Abréviations utilisées

nb = nombre. resp. : respectivement.
 x = nombre réel. n = nombre entier. α = angle exprimé en radians.
 ch = chaîne de caractères car = caractère arg = argument quelconque
liste signifie une suite (avec le séparatif ;) de valeurs, d'adresses, de noms de cases, de bloc(s)..
Le nom des fonctions peut être écrit aussi bien avec des majuscules que des minuscules.

Conversions numériques

Arrondi ($x ; n$) : arrondi de x avec n zéros avant la virgule (si $n < 0$) ou après la virgule (si $n > 0$).
Ent (x) : partie entière n de x ($n \leq x$, mais si $x < 0$, $|n| \geq |x|$). **Frac** (x) : partie fractionnaire de x .
Tronque (x) : $\rightarrow x$ diminué de sa partie fractionnaire ($|n| \leq |x|$, même si $x < 0$).
Signe (x) : $\rightarrow 1, 0$ ou -1 selon que x est positif, nul ou négatif.
Base10 ($n ; na$) : convertit en décimal le nb n exprimé dans le système à base na .
Basen ($n ; na ; ns$) : convertit le nb décimal n en un nb de base na avec au minimum ns chiffres.
Degres (α) : convertit en degrés l'angle α exprimé en radians.
Pi() : \rightarrow valeur de π . **Radians** (x) : convertit en radians la valeur x .

Algèbre

Abs (x) : valeur absolue $|x|$ de x
Mod ($n1 ; n2$) : $\rightarrow n1$ modulo $n2$; reste de la division $n1/n2$ (le signe est celui de $n1$).
Produit (*liste*) : \rightarrow résultat du produit de tous les nombres contenus dans la *liste*.
Racine (x) : racine carrée de x ($x \geq 0$).
Exp (x) : exponentielle e^x . **Fact** (n) : factorielle $n!$ (n entier > 0)
Ln (x), **Log10** (x) : logarithme de x , resp. népérien et décimal, $x \geq 0$
Log ($x ; b$) : logarithme de x en base b ($b > 1, x \geq 0$; si b est omis, il est pris égal à 10)

Trigonométrie

Sin (α), **Cos** (α), **Tan** (α) : resp. sinus, cosinus, tangente de α (α en radians). Valeurs de sin et cos entre -1 et 1 .
Asin (x) $\rightarrow \alpha$ ($-\pi/2 \leq \alpha \leq \pi/2$), **Acoss** (x) $\rightarrow \alpha$ ($0 \leq \alpha \leq \pi$) : resp. Arcsin et Arccos de x ($-1 \leq x \leq 1$), α en radians.
Atan (x) : $\rightarrow \alpha =$ arctangente de x ($-\pi/2 \leq \alpha \leq \pi/2$), α en radians.
Atan2 ($x ; y$) $\rightarrow \alpha =$ arctangente de y/x ($-\pi \leq \alpha \leq \pi$), α en radians.

Logique et conditions

Une variable ou une expression logique n'a que 2 valeurs possibles, vrai ou faux. Ex.: ($3 > 1$) est vrai, ($4 <> (2 * 2)$) est faux.
Un nombre peut être considéré comme une variable logique, de valeur vrai s'il n'est nul, fausse s'il est nul.
SI (*cond ; exp1 ; exp2*) : si la condition *cond* est vraie, l'expression *exp1* est exécutée, sinon c'est *exp2*.
ET ($l1 ; l2 ; \dots$) : intersection des variables logiques $l1, l2, \dots \rightarrow$ Vrai si tous les $l1, l2, \dots$ sont vrais.
OU ($l1 ; l2 ; \dots$) : réunion des variables logiques $l1, l2, \dots \rightarrow$ Vrai si l'un au moins des $l1, l2, \dots$ est vrai.
NON (*expr*) : inverse de la valeur logique *expr*.
Egal ($ch1 ; ch2$) : vrai ou faux selon que les chaînes *ch1* et *ch2* sont identiques ou non (casses prises en compte)

Identité

Type (*expr*) : $\rightarrow 1$ si *expr* vide, 2 si l'*expr* est à valeur logique, 3 si c'est un nombre, 4 si c'est une chaîne.
EstNum (*arg*) : \rightarrow vrai si *arg* est un nombre, sinon faux. **EstChaîne** (*arg*) : \rightarrow vrai seulement si *arg* est une chaîne.
EstLogique (*arg*) : \rightarrow vrai seulement si *arg* est une expression à valeur logique.
EstErr(*expr*) ; **EstNd** (*expr*) ; **EsrVide** (*expr*) : \rightarrow vrai si l'expression *exp* est erronée, non disponible, vide (voir Aide).

De chaîne (ou de texte)

(NB. le premier caractère d'une chaîne a pour numéro 1).

Longueur (*ch*) : \rightarrow nombre de caractères contenus dans *ch*.
Majus (*ch*) : met la chaîne *ch* en majuscules. **Minus** (*ch*) : met la chaîne *ch* en minuscules.
NomPropre (*ch*) : met en majuscules le premier caractère de chaque mot de *ch*.
SupprEspace (*ch*) : supprime les espaces redondants dans *ch* (dont tous ceux de tête).
Debut ($ch ; n$) : $\rightarrow n$ premiers caractères de la chaîne *ch*. **Fin** ($ch ; n$) : $\rightarrow n$ derniers caractères de la chaîne *ch*.
Concat ($ch1 ; ch2 ; \dots$) : concatène les chaînes en argument. **Repete** ($ch ; n$) : concatène *ch* n fois sur elle-même.
Car (n) : \rightarrow caractère ASCII de numéro n . Ex.: **car** (65) = 'A'. **Code** (*ch*) : $\rightarrow n^{\circ}$ ASCII du 1er caractère de *ch*.
Chaîne (x) : \rightarrow chaîne représentant le nombre x (entier, réel, etc...)
Valeur (*ch*) : tente de constituer une valeur numérique à partir du contenu de *ch* (ignore les caractères non numériques).
Remplace ($ch1 ; n ; n1 ; ch2$) : à partir du $n^{\text{ième}}$ caractère de *ch1* compris, supprime $n1$ caractères et insère *ch2* à leur place.
Recherche ($ch1 ; ch2 ; n$) : position de la chaîne *ch1* dans la chaîne *ch2* à partir de son $n^{\text{ième}}$ caractère. ????

Statistique, tri, recherche

Max (*liste*) : → le plus grand nombre dans la *liste*.

Min (*liste*) : → le plus petite nombre dans la *liste*.

Somme (*liste*) : → somme des nombres de la *liste*.

Moyenne (*liste*) : → moyenne des nombres de la *liste*.

Ecartech (*liste*) : écart-type ϵ de l'ensemble des valeurs x_i ; $\epsilon = \text{racine}(\sum(x_i-x_0)^2/(n-1))$; x_0 est la valeur moyenne.

Varech (*liste*) : → variance de l'ensemble des valeurs dans la liste (carré de l'écart-type ?).

Nombre (*liste*) : → nombre d'arguments dans la *liste* (chaînes, nombres, adresses).

Nombre2 (*val* ; *liste*) : recherche combien de fois la donnée *val* (case, nb, chaîne) est contenue dans la *liste*.

Index (*bloc* ; *nl* ; *nc*) : → contenu de la case du *bloc*, ligne *nl*, colonne *nc*, comptées à partir du début du *bloc*.

Extrait (*ch* ; *n0* ; *n*) : extrait *n* caractères de *ch* à partir de *n0* (i.e. du caractère de rang *n0*).

Alea () : → *x* aléatoire ($0 < x < 1$).

Alea (*m*) : → *n* aléatoire ($0 \leq n \leq m$, *n* et *m* entiers).

Cherche (*expr* ; *bloc1* ; *bloc2* ; *typ*) : convertit *expr* en nombre *n*, recherche dans *bloc1* la plus grande valeur $\leq n$ (si *typ*=1 et si *bloc1* est à valeurs croissantes), la plus petite $\geq n$ (si *typ* = -1 et si *bloc1* est à valeurs décroissantes) ; retourne la valeur de *bloc2* correspondant à celle trouvée. Si *typ*=0, cherche le premier nombre égal à *expr*.

Chercheh (*expr* ; *bloc* ; *dec* ; *typ*) : comme Cherche, mais seulement sur la 1ère ligne du bloc *bloc* ; retourne la valeur de la case située dans *bloc* à *dec* positions verticales de la case trouvée.

Cherchev (*expr* ; *bloc* ; *dec* ; *typ*) : comme Chercheh, mais dans la 1ère colonne de *bloc* ; le décalage *dec* est horizontal

Choix (*n* ; *ch1* ; *ch2* ; *ch3* ; ...) : → le contenu de la *nième* composante de la liste en argument.

Dates

La date numérique *da* est un nb **réel** donnant le nombre de jours et la fraction de jour écoulés depuis T_0 (1er janvier 1904).

Date (*na* ; *nm* ; *nj*) → date numérique traduisant celle donnée par les numéros *na*, *nm*, *nj* de l'année, du mois, du jour.

Heure (*nh* ; *nm* ; *ns*) → date numérique ($da \leq 1$) traduisant le temps *nh*, *nm*, *ns* en heures, minutes et secondes.

DateNum (*ch*) : convertit la chaîne *ch*, si elle représente bien une date, en *da*, date numérique.

HeureNum (*ch*) : comme Heure(), mais l'argument *ch* doit être une chaîne donnant une heure avec les séparatifs :

DateChaîne (*da* ; *typ*) : convertit la date numérique *da* (partie entière seule) en chaîne indiquant la date "en clair"

(Ex.: *typ*=1 → 9 Avr 2001 ; 2 → 9 Avril ... ; 3 → Dim 9 Avr ... ; 4 → Dimanche 9 Avril ... ; 0 → 20/04/01).

HeureChaîne (*da* ; *typ*) : convertit la partie fractionnaire de *da* en chaîne indiquant l'heure "en clair"

(Ex. avec $da = 1234,9692$: *typ* = 0 → 11:15 ; 1 → 11:15:35 ; 2 → 23:15 ; 3 → 23:15:35).

Année (*da*), **Mois** (*da*), → numéro de l'année, numéro du mois correspondant à la date *da*.

Heures (*da*), **Minutes** (*da*), **Secondes** (*da*) : → le nb entier resp. d'heures, de minutes, de secondes contenues dans *da*

Jour (*da*), **JourAnnée** (*da*), **JourSemaine** (*da*) : *n*° du jour resp. dans le mois, l'année, la semaine (dimanche=1).

NomJour (*n*) : nom du jour de la semaine ($n \leq 7$, 1→dimanche).

NomMois (*n*) : idem, pour le mois ($n \leq 12$).

SemaineAnnee (*da*) : → *n*° de la semaine dans la (dernière) année correspondant à la date numérique *da*.

Maintenant () : → date actuelle lue sur l'horloge du PC présentée sous un format dépendant de celui de la case.

Tableau

Nombre (*liste*) : → nombre de valeurs dans la *liste* en argument.

Colonne (*case*), **Rangee** (*case*) : → *n*° absolu de la colonne, resp. de la ligne, contenant la case *case*.

Index (*bloc* ; *nl* ; *nc*) : → contenu de la case du bloc *bloc* située à *nc* colonnes et *nl* lignes du début du bloc.

Position (*val* ; *bloc* ; *typ*) : → position dans *bloc* (ordonné dans le sens compatible) de la valeur la plus proche de *val* (juste inférieure si *typ*=-1, supérieure si *typ*=1) ou la première valeur égale à *val* si *typ*=0.

RubriqueFusion (), **Macro** (*nom* ; *typ*) : pour mémoire, voir l'*Aide-en-ligne*.

Dialogue

Alerte (*message* ; *typ*) : écrit le *message*, → 1 si on clique sur 'OK' (*typ*=1) ou 'Oui' (*typ*=2) et 0 sur 'Annuler' ou 'Non'.

Bip () : émet un signal sonore.

Err () : fait afficher '#Erreur!'

Nd () : voir l'*Aide en ligne*.

Financières (pour mémoire, voir les détails dans le menu Aide)

Periodes (*taux*, *vers*, *val*) : → nb de versements *vers* nécessaires pour rembourser un prêt au taux *taux* de montant *val*.

Rembour () : → valeur du versement nécessaire pour rembourser un prêt selon le taux et sa durée.

Taux (*y*, *x*, *n*) : → taux d'intérêt nécessaire pour qu'un placement *x* atteigne la valeur *y* au bout de *n* périodes.

Tri () : taux de rendement interne

Trim () : taux de rendement interne modifié.

Va () : valeur initiale d'un placement

Van () : valeur actualisée nette d'un placement.

Vc () : valeur capitalisée d'un placement.