

Masque de Saisie sous Excel

1. Référencement des Cellules.

1.1 le Référencement L1C1.

Dans le style L1C1, Microsoft Excel indique l'emplacement d'une cellule par un « L » suivi du numéro de ligne puis un « C » suivi du numéro de colonne. Par exemple, la référence de cellule absolue L1C1 est équivalente à la référence absolue \$A\$1 dans le style de référence A1. Si la cellule active est A1, la référence de cellule relative L[1]C[1] fait référence à la cellule située une ligne en dessous et une colonne à droite, soit la cellule B2.

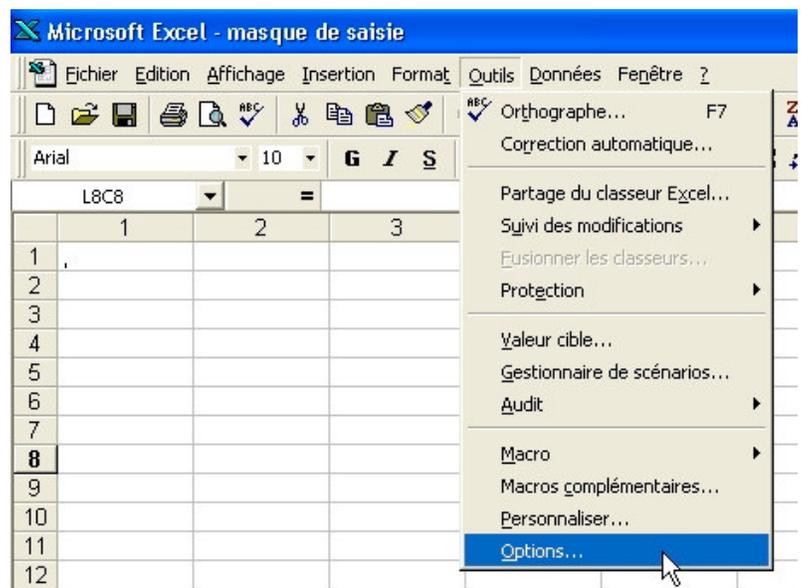
Les exemples suivants sont tous des références de style L1C1.

Référence	Signification
L(-2)C	Référence relative à la cellule située deux lignes au-dessus et dans la même colonne
L(2)C(2)	Référence relative à la cellule située deux lignes en dessous et deux colonnes à droite
L2C2	Référence absolue à la cellule située sur la seconde ligne et dans la seconde colonne
L(-1)	Référence relative à toute la ligne au-dessus de la cellule active
LC	Référence absolue à la cellule en cours

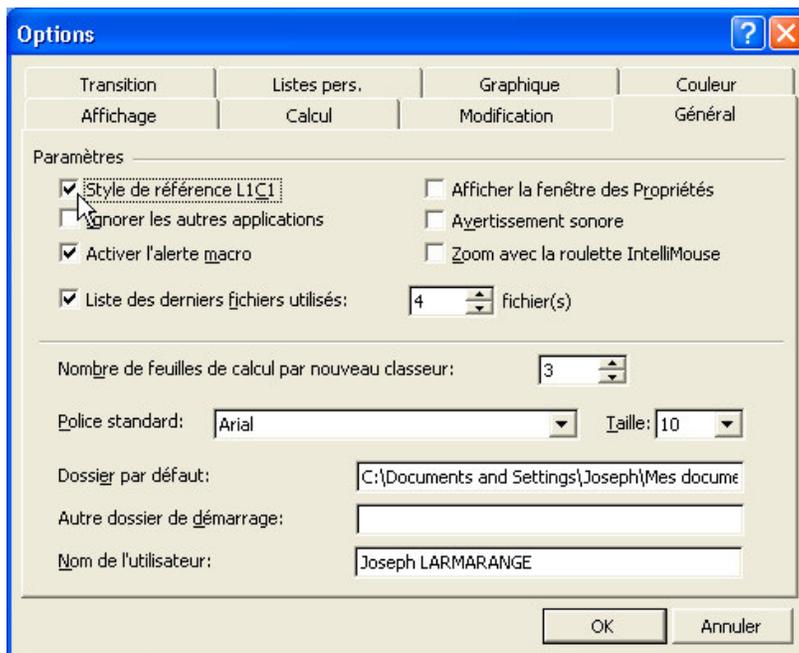
1.2 pour appliquer le Référencement L1C1 à Excel.

Par la suite, nous allons avoir besoin de ce type de référencement des cellules. Il faut donc indiquer à Excel que nous allons utiliser ce référencement.

Pour cela, il faut aller dans le menu Outils > Options...



Puis cocher la case *Style de référence L1C1* et cliquer sur *OK*.

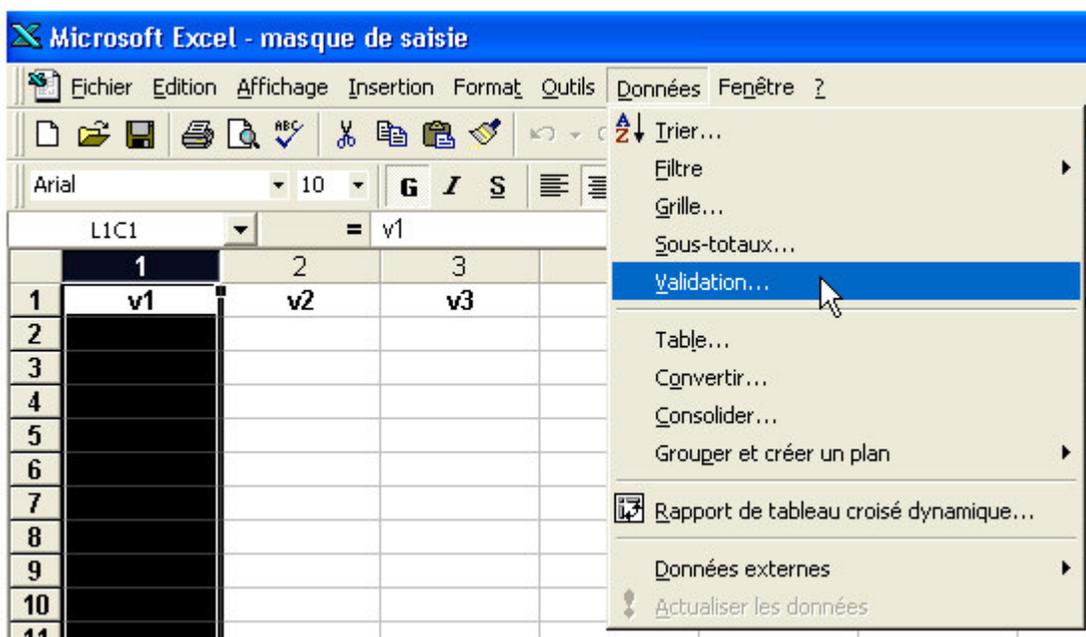


2. la Validation des Données.

2.1 le Principe général.

Excel dispose d'un système de validation des données qui permet de restreindre les données valides que l'on peut saisir dans une cellule. Cela permet également d'afficher un message à l'utilisateur pour lui indiquer comment saisir les données. Enfin, si la donnée saisie n'est pas valide, on peut choisir soit de simplement avertir l'utilisateur, soit d'interdire la saisie.

Pour accéder à la validation des données, il suffit de sélectionner les cellules sur lesquelles va porte cette validation et de cliquer sur *Données > Validation...*



Apparaît alors la boîte de dialogue *Validation des données* reproduite ci-contre.



2.2 le Volet *Options*.

Dans *Autoriser*, on va donner le type de critère de validation. Les différences options diffèrent selon ce qu'on a choisir dans *Autoriser*.

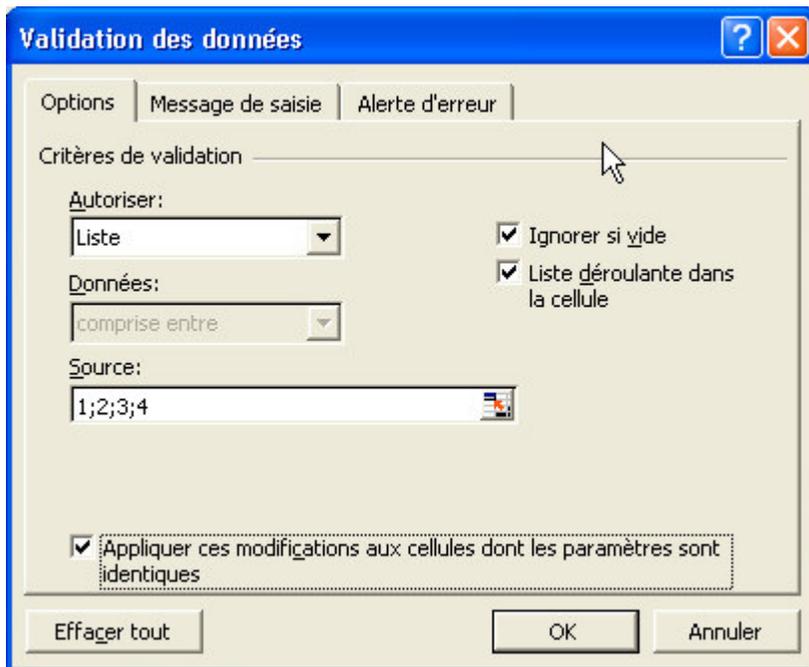
Type	Description et options
Tous	N'impose aucune restriction aux entrées valides. Utilisez ce paramètre si vous voulez afficher un message d'entrée sans vérifier la validité des entrées.
Personnalisé	Permet d'entrer une formule, d'utiliser une expression, ou de faire référence à un calcul d'une autre cellule pour déterminer les entrées valides.
Date	Indique que les entrées doivent être des dates.
Décimal	Indique que les entrées doivent être des nombres ou des fractions.
Liste	Permet de spécifier une liste d'entrées valides. <i>Liste déroulante dans cellule :</i> Sélectionnez cette case à cocher pour afficher une flèche déroulante permettant d'afficher la liste de sélection lorsque la personne tapant les données clique sur la cellule. Désactivez cette case à cocher pour ne pas afficher la liste déroulante.
Longueur du texte	Indique le nombre de caractères pour les entrées.
Heure	Indique que les entrées doivent être des heures.
Nombre entier	Indique que les entrées doivent être des nombres entiers.

Attention :

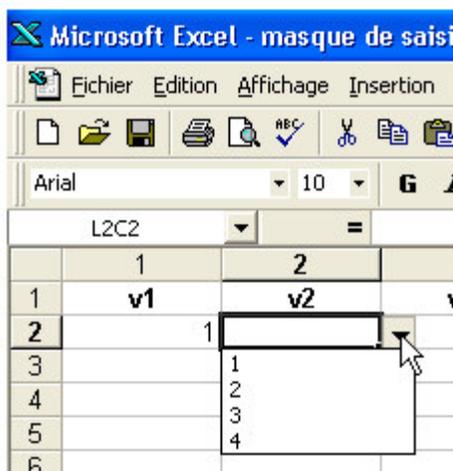
La case à cocher *Ignorer si vide* permet d'accepter permet de ne pas tenir compte des cellules vides et de les accepter telles quelles. Par la suite, nous travaillerons toujours avec cette case cochées.

Les différentes options de la cellule *Données* sont suffisamment explicites pour ne pas avoir à les détailler ici.

Lorsque vous avez créé une validation pour tout un ensemble de cellules, l'option *Appliquer ces modifications aux cellules dont les paramètres sont identiques* apparaît. Cela permet de modifier en une seule fois toutes les cellules qui avaient le même critère de validation.



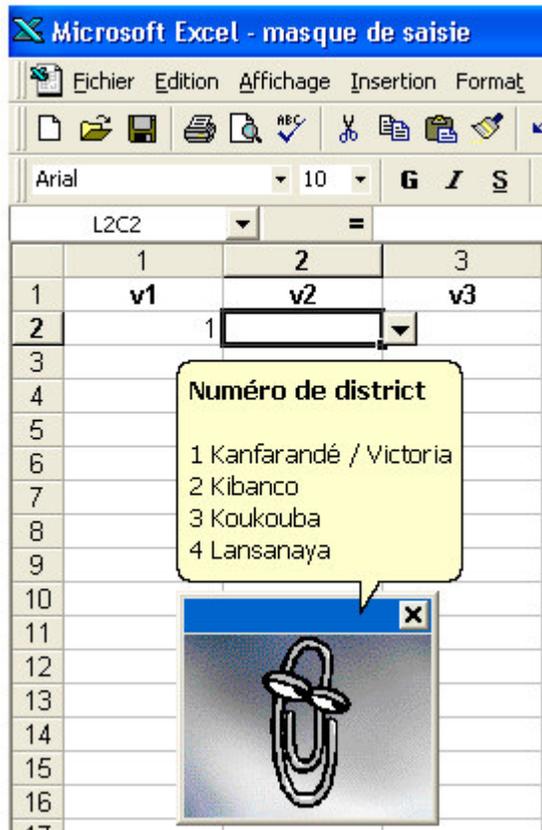
Si la case *Liste déroulante dans la cellule* est cochée, cela fera apparaître une petite liste où l'utilisateur pourra sélectionner la bonne valeur. Exemple :



2.3 le Volet *Message de Saisie*.



En cochant *Quand la cellule est sélectionnée*, on peut alors écrire en dessous un message qui apparaîtra lorsque l'utilisateur sera sur la cellule en question. Par exemple :



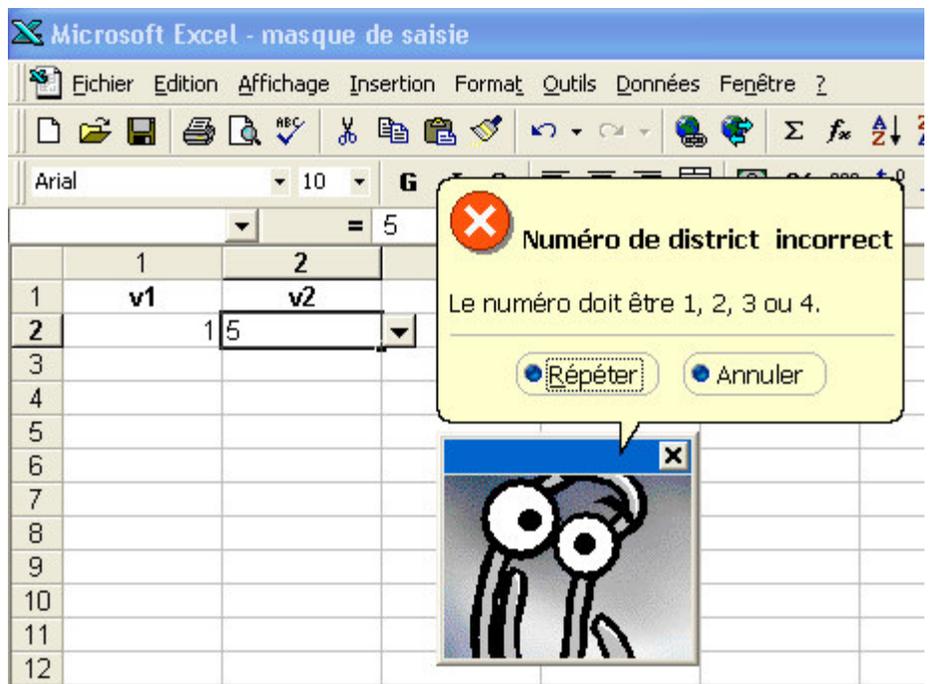
2.4 le Volet *Alerte d'Erreur*.

Quand la case à cocher *Quand des données non valides sont saisies* est cochée, alors il est possible de saisir un message d'erreur. Il est possible de choisir 3 styles de message d'erreurs :



Type d'erreur	Boutons	Effets
Information	OK	Entre les données non valides dans la cellule (par défaut).
	Annuler	Restitue à la cellule la valeur précédente.
Arrêt	Répéter	Renvoie à la cellule pour effectuer des modifications (par défaut).
	Annuler	Restitue à la cellule la valeur précédente.
Avertissement	Oui	Entre les données non valides dans la cellule.
	Non	Renvoie à la cellule pour effectuer des modifications (par défaut).
	Annuler	Restitue à la cellule la valeur précédente.

Dans notre cas, nous utiliserons systématiquement le type *Arrêt* qui permet d'interdire de saisir des données erronées. En cas de mauvaise saisie, un message d'erreur s'affiche et l'utilisateur doit saisir une valeur correcte.



2.5 la Validation Personnalisée.

Ce type de validation est très intéressante car elle permet de rentrer une formule. Ainsi, il nous est possible de procéder à des validations complexes.

Une fois *Personnalisée* choisi dans *Autoriser*, apparaît l'option *Formule*.

Une formule commence toujours par le signe =.

Ensuite, il faut écrire une formule qui renvoie deux valeurs : Vrai ou Faux, qui sont codées 1 et 0 par l'ordinateur. Si la valeur renvoyée est 1, alors la donnée saisie est validée, sinon elle est refusée.

Exemples de formule :

- =LC>12 renvoie VRAI si la valeur saisie est strictement supérieure à 12. LC fait référence à la cellule en cours, c'est-à-dire à la cellule testée.
- =LC=(LC(-1)+LC(-2)) renvoie VRAI si la valeur saisie est égale à la somme des valeurs saisies dans les deux cases précédentes.

2.6 Les Opérateurs de Comparaison

Opérateur de comparaison	Signification
= (signe égal)	Egal à
> (signe supérieur à)	Supérieur à
< (signe inférieur à)	Inférieur à
>= (signe supérieur ou égal à)	Supérieur ou égal à
<= (signe inférieur ou égal à)	Inférieur ou égal à
<> (signe différent)	Différent de

2.7 Les Fonctions Logiques

Les différentes fonctions logiques sont :

- ET Renvoie VRAI si tous ses arguments sont VRAI
- FAUX Renvoie la valeur logique FAUX
- SI Spécifie un test logique à effectuer
- NON Renvoie la logique de son argument
- OU Renvoie VRAI si un argument est VRAI
- VRAI Renvoie la valeur logique VRAI

2.7.1 La Fonction ET

Renvoie VRAI si tous les arguments sont VRAI; renvoie FAUX si au moins l'un des arguments est FAUX.

Syntaxe

ET(*valeur_logique1*; *valeur_logique2*;...)

valeur_logique1, *valeur_logique2*, ... représentent les 1 à 30 conditions que vous souhaitez tester et qui peuvent être soit VRAI, soit FAUX.

- Les arguments doivent être évalués à des valeurs logiques, telles que VRAI ou FAUX, ou doivent être des matrices ou des références contenant des valeurs logiques.
- Si une matrice ou une référence utilisée comme argument contient du texte ou des cellules vides, ces valeurs ne sont pas prises en compte.
- Si la plage spécifiée ne contient aucune valeur logique, ET renvoie la valeur d'erreur #VALEUR!

Exemples

ET(VRAI; VRAI) égale VRAI

ET(VRAI; FAUX) égale FAUX

ET(2+2=4; 2+3=5) égale VRAI

2.7.2 La Fonction OU

Renvoie la valeur VRAI si un argument est VRAI et FAUX si tous les arguments sont FAUX.

Syntaxe

OU(*valeur_logique1*; *valeur_logique2*,...)

valeur_logique1, *valeur_logique2*, ... sont de 1 à 30 conditions que vous souhaitez tester, et qui peuvent être soit VRAI, soit FAUX.

- Les arguments doivent être évalués à des valeurs logiques, telles que VRAI ou FAUX, ou dans des matrices ou références contenant des valeurs logiques.
- Si une matrice ou une référence tapée comme argument contient du texte, des nombres ou des cellules vides, ces valeurs ne sont pas prises en compte.
- Si la plage spécifiée ne contient aucune valeur logique, la fonction OU renvoie la valeur d'erreur #VALEUR!

Exemples

OU(VRAI) égale VRAI

OU(1+1=1,2+2=5) égale FAUX

Si A1:A3 contient les valeurs VRAI, FAUX et VRAI : OU(A1:A3) égale VRAI

2.7.3 La Fonction SI

Renvoie une valeur si la condition que vous spécifiez est VRAI et une autre valeur si cette valeur est FAUX. Utilisez la fonction SI pour effectuer un test conditionnel sur des valeurs et des formules.

Syntaxe

SI(*test_logique*; *valeur_si_vrai*; *valeur_si_faux*)

test_logique est toute valeur ou expression dont le résultat peut être VRAI ou FAUX.

valeur_si_vrai est la valeur qui est renvoyée si le test logique est VRAI. Si l'argument *test_logique* est VRAI et que l'argument *valeur_si_vrai* est omis, la fonction renvoie la valeur VRAI. L'argument *valeur_si_vrai* peut être une autre formule.

valeur_si_faux est la valeur qui est renvoyée si le test logique est FAUX. Si l'argument *test_logique* est FAUX et que l'argument *valeur_si_faux* est omis, la fonction renvoie la valeur FAUX. L'argument *valeur_si_faux* peut être une autre formule.

- Il est possible d'imbriquer jusqu'à sept fonctions SI comme arguments *valeur_si_vrai* et *valeur_si_faux* pour élaborer des tests plus complexes. Reportez-vous au dernier des exemples suivants.
- Lorsque les arguments *valeur_si_vrai* et *valeur_si_faux* sont évalués, la fonction SI renvoie la valeur transmise par l'exécution de ces instructions.
- Si l'un des arguments de la fonction SI est une matrice, chaque élément de la matrice est évalué lorsque l'instruction SI est exécutée. Si certains des arguments *valeur_si_vrai* et *valeur_si_faux* sont des fonctions exécutant une action, toutes ces actions sont exécutées.

Exemples

Dans l'exemple suivant, si la valeur contenue dans la cellule A10 est 100, l'argument *test_logique* est VRAI et la valeur totale de la plage B5:B15 est calculée. Sinon, l'argument *test_logique* est FAUX et du texte vide ("") est renvoyé, laissant vide la cellule qui contient la fonction SI.

SI(A10=100,SOMME(B5:B15),"")

3. Création d'un Masque de Saisie.

3.1 Préparation du Fichier.

Tout d'abord, il faut configurer Excel pour qu'il utilise le référencement L1C1 (cf. partie 1.1).

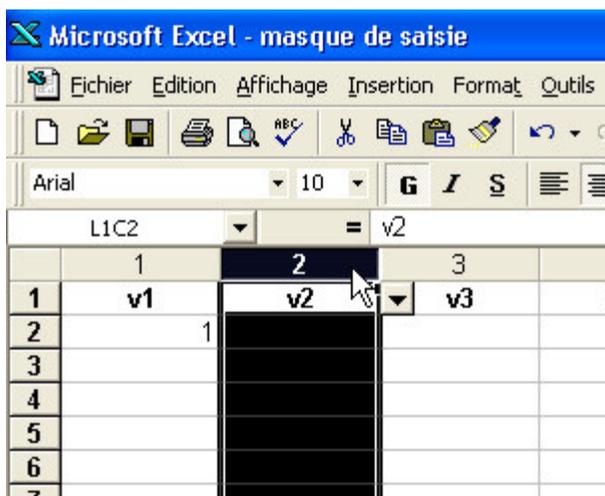
Puis nous allons saisir sur la première ligne le nom des différentes variables. La saisie se fera comme sous SPSS. En colonne les variables et en ligne les individus. La première ligne est consacrée au nom des différentes variables. Pour plus de visibilité, il est conseillé de mettre les noms en gras.

Par la suite, nous allons créer des critères de validation pour des colonnes entières. Il faut donc saisir les noms des variables avant les critères de validation, car après on ne pourra plus saisir le nom des variables, car il ne correspondront pas aux critères de validation retenus¹.

La procédure pour chaque variable va être la suivante :

- On sélectionne la colonne.
- On clique sur *Données > Validation...*
- On spécifie les critères de validation.
- Éventuellement spécifier un message de saisie et un message d'erreur.

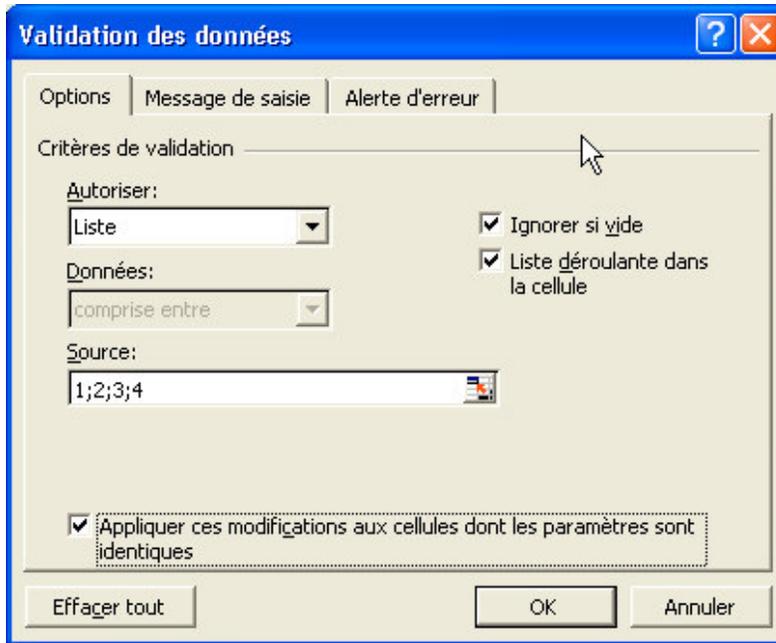
Pour sélectionner une colonne, cliquer sur le numéro de la colonne. La colonne entière doit se retrouver noircie :



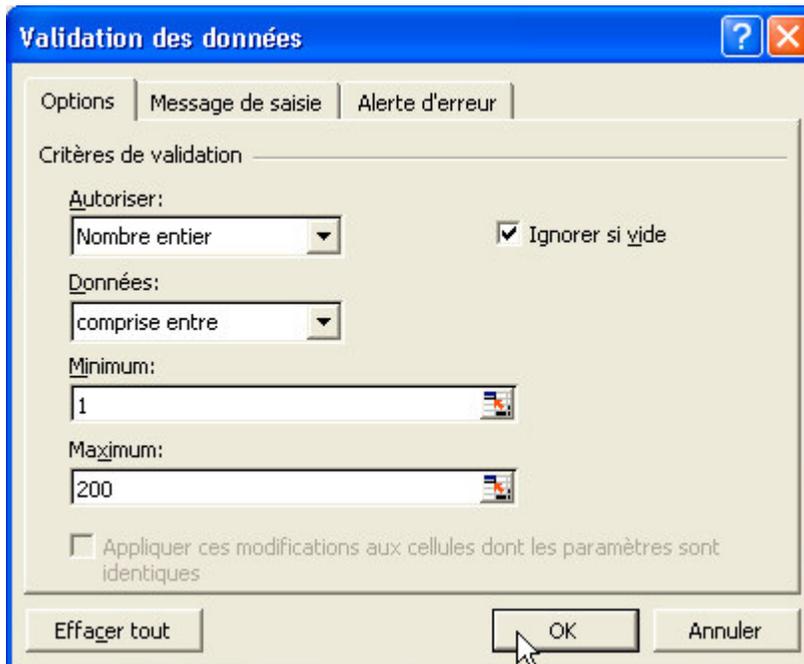
¹ Ou alors, il faudra supprimer les critères de validation pour la première ligne, en sélectionnant toute la ligne.

3.2 les Variables v1 à v6.

Pour v1 et v2, le plus simple est de choisir *Autoriser liste*, et de spécifier les valeurs possibles dans *Source*, en les séparant par un point-virgule. Pour v2, cela donne :



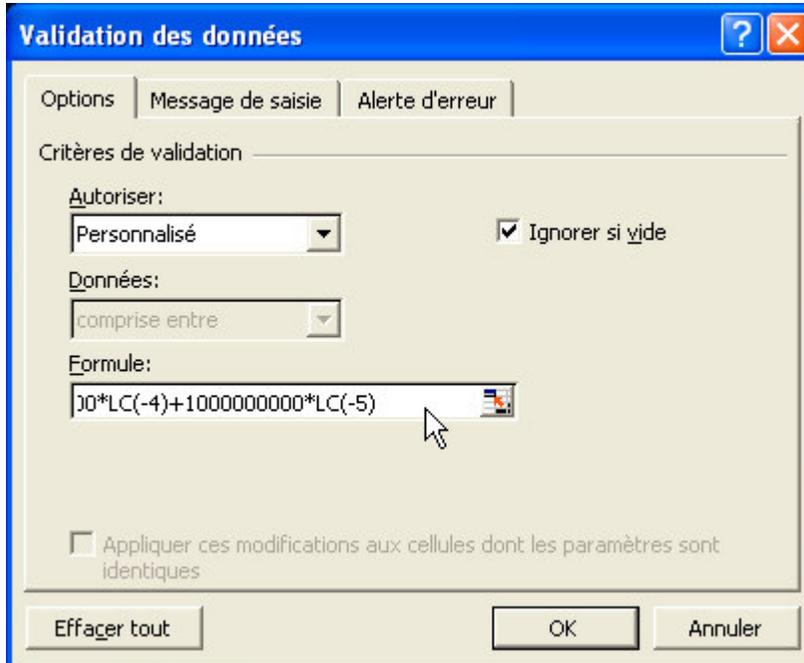
Pour v3, v4 et v5, le plus simple est d'autoriser les nombres entiers compris entre la valeur minimum et maximum de la variable. On obtient donc ceci pour v3 et v5.



L'identifiant v6 est la concaténation des variables v1 à v5. Comme ces variables sont des nombres, il est possible de calculer v6 avec la formule :

$$v6 = v5 + 1.000 v4 + 100.000 v3 + 100.000.000 v2 + 1.000.000.000 v1$$

Nous allons donc utiliser une validation *Personnalisée*.



La formule utilisée sera :

$$=LC=LC(-1)+1000*LC(-2)+100000*LC(-3)+100000000*LC(-4)+1000000000*LC(-5)$$

où LC est la case testée, LC(-1) renvoie à la valeur de la case située sur la même ligne dans la colonne précédente, soit la valeur saisie pour v5, LC(-2) la valeur saisie pour v4, etc.

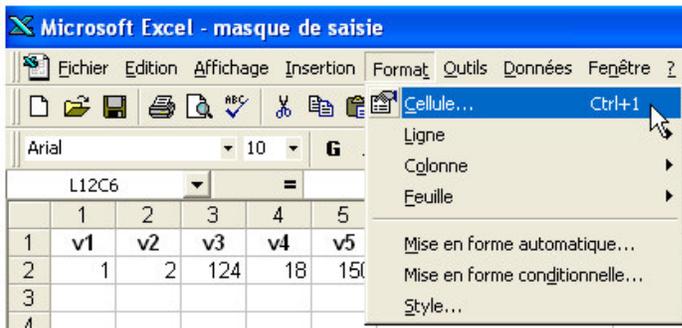
Il se peut que Excel affiche le message suivant :



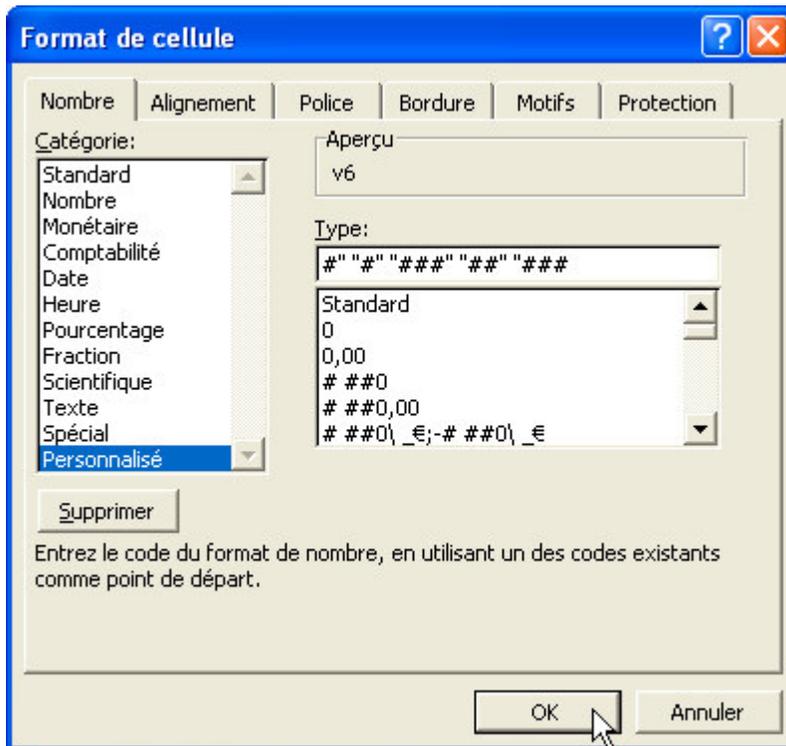
Cliquez sur *Oui*. Cela est dû au fait que le nom de la variable dans la première ligne de la colonne ne répond aux critères de validation.

On peut avoir intérêt à modifier l'affichage de telle manière qu'un identifiant comme 1212418150 apparaissent sous la forme 1 2 124 18 150. Pour cela :

- sélectionnez la colonne contenant v6,
- cliquez sur *Format > Cellule...* ,



- sélectionnez l'onglet *Nombre*,
- dans *Catégorie* choisissez *Personnalisé*,
- dans *Type*, rentrez la formule `"# " "# "###" "#" "###"`.

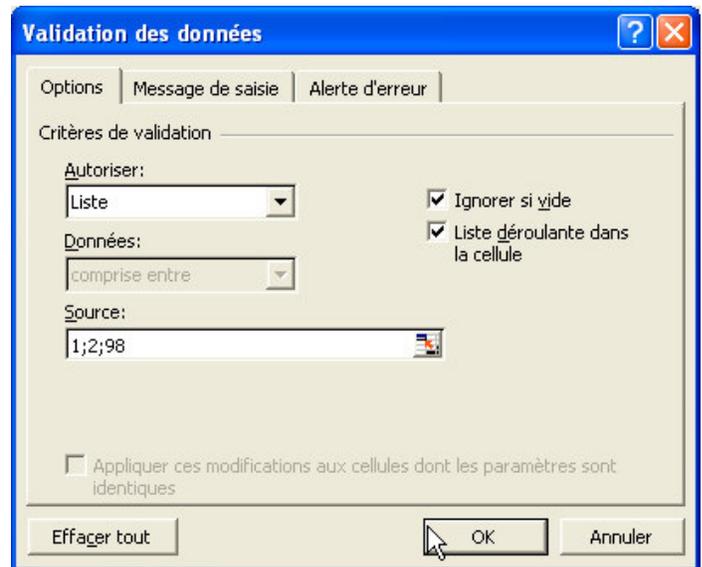


La formule est construite de la manière suivante : un # correspond à un chiffre, entre guillemets ("") le texte que l'on souhaite voir apparaître (ici un espace). On demande donc à l'ordinateur d'afficher un chiffre, puis un espace, puis un chiffre, puis un espace, puis trois chiffres, etc.

3.3 les Variables c1 à c14.

Pour c1, c'est relativement simple, il suffit de choisir *Liste* et de donner les différentes valeurs possibles en les séparant par un point-virgule.

Cela donne donc :



Pour c2, il faut tenir compte de la valeur de c1. Si c1 est différent de 2 ($LC(-1) \neq 2$) alors nous devons avoir $LC=97$. Sinon, c2 vaut 1, 2, 3, 4 ou 98. Cela peut se traduire avec la fonction OU : $OU(LC=1;LC=2;LC=3;LC=4;LC=98)$ en séparant les différents tests par un point-virgule.

Ainsi, nous allons utiliser une validation personnalisée avec la formule :

$$=SI(LC(-1) \neq 2; LC=97; OU(LC=1; LC=2; LC=3; LC=4; LC=98))$$

Nous avons utilisé une fonction SI dont le premier argument est une condition, le second le test à réaliser en si la condition est remplie et le troisième le test à réaliser si la condition n'est pas remplie.

Pour c3, le principe est le même. Si c2 vaut 2 ($LC(-1)=2$) on demandera seulement à ce que c3 soit différent de 97 ($LC \neq 97$), puisqu'il s'agit d'une saisie en toute lettre. Nous ferons donc une validation personnalisée avec la formule :

$$=SI(LC(-1)=2; LC \neq 97; LC=97)$$

Pour c4, on fait de même que pour c2, à la différence que la condition $c2 \neq 3$ s'exprime de la manière $LC(-2) \neq 3$ car la valeur donnée pour c2 est située deux colonnes avant la case utilisée pour c4. La formule est donc :

$$=SI(LC(-2) \neq 3; LC=97; OU(LC=1; LC=2; LC=3; LC=4; LC=98))$$

Pour c5, on procède comme pour c3 en adaptant la condition. La formule est donc :

$$=SI(LC(-3)=4; LC \neq 97; LC=97)$$

Pour c6, il nous faut accepter toutes les valeurs comprises entre 1850 et 1987 exclus (ET(LC>1850;LC<1987)). Mais nous voulons également que ces valeurs soient des nombres entiers (LC=ENT(LC)) (cf. Encadré 1 pour plus de renseignements sur la fonction ENT). Enfin, il se peut aussi que l'on rencontre 9999. Nous allons traduire tout cela dans une formule imbriquant une fonction ET dans une fonction OU soit :

=OU(LC=9999;ET(LC>1850;LC<1987;LC=ENT(LC)))

c7 doit être égal à 2002-c6, excepté si c6 vaut 9999, auquel cas c7 vaut 996. Cela va se traduire encore une fois par une fonction SI :

=SI(LC(-1)=9999;LC=999;LC=2002-LC(-1))

Encadré 1 Fonction Partie Entière ENT()

Fonction ENT()

Arrondit un nombre à l'entier immédiatement inférieur.

Syntaxe

ENT(*nombre*)

nombre représente le nombre réel que vous souhaitez arrondir au nombre entier immédiatement inférieur.

Exemples

ENT(8,9) égale 8

ENT(3,156) égale 3

Pour savoir si le nombre contenu dans la case LC(-1) est un nombre entier, il suffit de faire le test LC(-1)=ENT(LC(-1)) qui renverra VRAI si LC(-1) est entier et FAUX sinon.

Encadré 2 Variable géométrique.

Variable géométrique

Il y a deux manières principales de coder une question avec plusieurs réponses possibles. Soit on crée autant de variables Oui/Non que de modalités à la question, soit on a recours à une variable dite *géométrique*.

Dans ce second cas, on affecte chaque modalité d'une valeur correspondant à une puissance de 2 : 1, 2, 4, 8, 16, 32, 64, 128, 256, etc. On rentrera comme valeur la somme des puissances de 2 attribuées à chacune des modalités correspondant au sujet. On utilise le fait que tout nombre entier positif et non nul se décompose de manière unique comme la somme de puissances de deux.

Si notre variable comporte n modalités, ses valeurs seront comprises entre 1 et (2^n-1) inclus.

Lors d'une analyse, on peut avoir besoin de savoir si une modalité particulière a été validée pour un individu. Soit v la variable analysée et x une des modalités de cette variable. x est une puissance de 2. On cherche à savoir si x est contenu dans la décomposition en puissances de deux de v . Pour cela, il suffit de regarder $E(v/x)$, c'est-à-dire la partie entière de la division de v par x . Si le résultat est impair, x est contenu dans la décomposition en puissances de deux de v . Sinon, le résultat est pair.

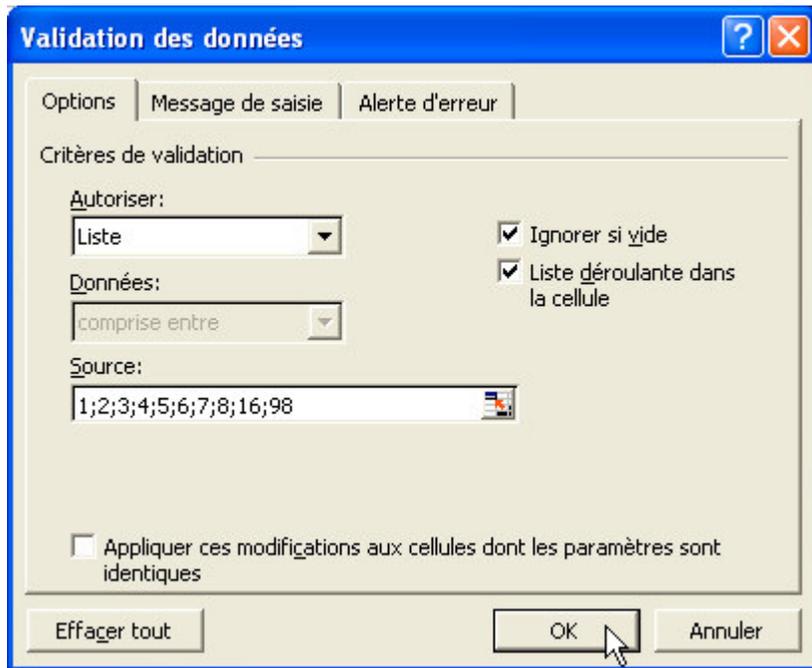
Sous Excel :

La formule `MOD(ENT(v/x);2)` renvoie 1 si x est « contenu » dans v et 0 sinon. Il faut penser à remplacer v et x par leur valeur ou par l'adresse des cellules qui les contiennent.

Sous SPSS :

Cette même formule s'écrit `MOD(TRUNC(v/x),2)`. Elle renvoie toujours 1 si x est « contenu » dans v et 0 sinon. Il faut penser à remplacer v par le nom de la variable analysée et x par la valeur de la modalité recherchée.

$c8$ est une variable dite *géométrique* (cf. Encadré 2). Mais les modalités *sans religion* et *athée* ne peuvent pas être associées avec d'autres modalités. Les valeurs possibles sont donc toutes les sommes possibles à partir de 1, 2 et 4, ainsi que les valeurs 8, 16 et 98. Le plus simple est d'autoriser une liste de nombre.



c9 est une variable géométrique à 8 modalités. Ses valeurs sont donc des nombres entiers ($LC=ENT(LC)$) compris entre 1 et $2^8-1=255$. Enfin, on peut aussi avoir la valeur 998 traduisant une absence d'information. Nous sommes dans une situation similaire à celle de c6.

$$=OU(LC=998;ET(LC>=1;LC<=255;LC=ENT(LC)))$$

Si c9 ne contient pas 16 ou si c9=998, alors c10=97. Nous allons donc avoir besoin de la formule donnée dans l'Encadré 2. On obtient donc :

$$=SI(OU(MOD(ENT(LC(-1)/16);2)=0;LC(-1)=998);LC=97;LC<>97)$$

Pour c11, être actuellement scolarisé, on considère qu'une personne âgée d'au moins 35 ans ne peut pas être encore scolarisée. Ainsi, si $c7 \geq 35$, alors c11=2. Cependant, il y a le cas où $c7=999$ et où on peut avoir néanmoins le renseignement concernant la scolarisation. Dans ce cas, si $c7 \geq 35$ et $c7 < 999$, alors c11=2 ou c11=98, sinon c11=1 ou c11=2 ou c11=98.

$$=SI(ET(LC(-4)>=35;LC(-4)<>999);OU(LC=2;LC=98);OU(LC=1;LC=2;LC=98))$$

c12 ne peut prendre que trois valeurs. Cependant, si c11=1, alors c12 vaut obligatoirement 1. La formule est donc la suivante :

$$=SI(LC(-1)=1;LC=1;OU(LC=1;LC=2;LC=98))$$

c13 vaut 97 si c12 est différent de 1, sinon c13 peut prendre les valeurs 1, 2 et 3.

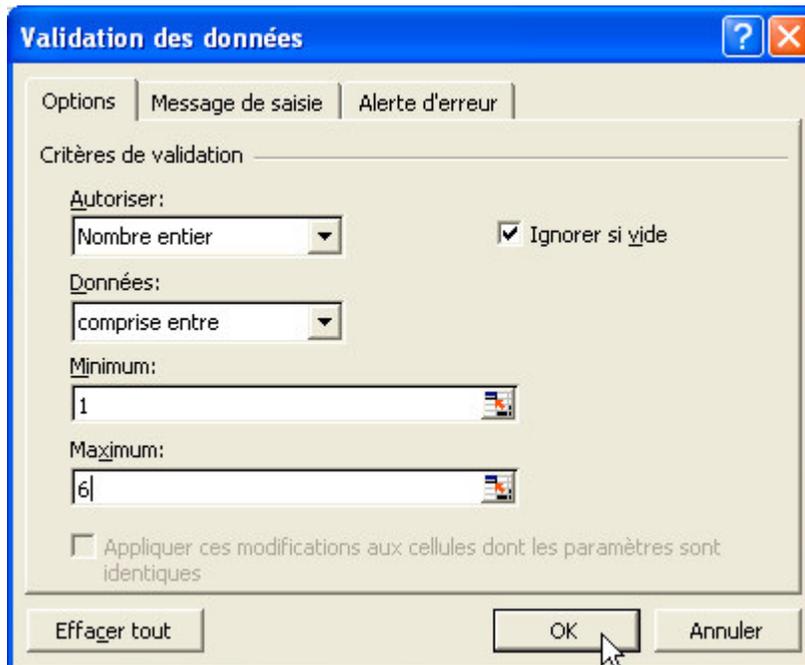
$$=SI(LC(-1)<>1;LC=97;OU(LC=1;LC=2;LC=3))$$

Pour c14, nous allons devoir imbriquer différentes fonction SI entre elles. En effet, si c13 vaut 1, alors c14 est compris entre 1 et 6, si c13 vaut 2, alors c14 est compris entre 7 et 12, si c13 vaut 3, alors c14 vaut 13, sinon c13 vaut 97. La formule va donc être :

$$=SI(LC(-1)=1;OU(LC=1;LC=2;LC=3;LC=4;LC=5;LC=6);SI(LC(-1)=2;OU(LC=7;LC=8;LC=9;LC=10;LC=11;LC=12);SI(LC(-1)=3;LC=13;LC=97)))$$

3.4 les Variables n1 à n6C.

Pour n1, il suffit d'autoriser une liste de nombre, ou bien les entiers compris entre 1 et 6 :



Si n1=1 ou n1=2, alors n2 vaut 0, sinon n2 est entier et compris entre 1 et 30. Cela se traduit sur Excel de la manière suivante :

$$=SI(OU(LC(-1)=1;LC(-1)=2);LC=0;ET(LC=ENT(LC);LC>=1;LC<=30))$$

Pour les questions n3 à n6, nous allons devoir distinguer le référencement relatif et absolu afin de pouvoir procéder à des copier / coller (cf. 1.1). Nous utiliserons un référencement absolu pour désigner les variables n1 et n2 qui sont respectivement dans les colonnes 21 et 22, un référencement relatif pour désigner les variables n3 à n6.

n3A vaut 97 si n2 < 1, c'est-à-dire si le nombre d'union est inférieur à 1. Sinon, n3A vaut 1 ou 2. La formule sera donc, en utilisant un référencement absolu pour désigner n2 :

=SI(LC22<1;LC=97;OU(LC=1;LC=2))

n4A vaut 997 si n3A vaut 97, sinon il s'agit d'un nombre entier compris entre 15 et c7, c7 étant l'âge au moment de l'enquête. Ainsi, on écrira en désignant c7 par LC13 car c7 est située dans la 13^{ème} colonne :

=SI(LC(-1)=97;LC=997;ET(LC=ENT(LC);LC>=15;LC<=LC13))

n5A est relativement simple :

=SI(LC(-2)=97;LC=97;OU(LC=1;LC=2))

n6A vaut 997 si n5A est différent de 2. Sinon, il est compris entre n4A et c7 (âge d'entrée en union et âge actuel). On aura donc :

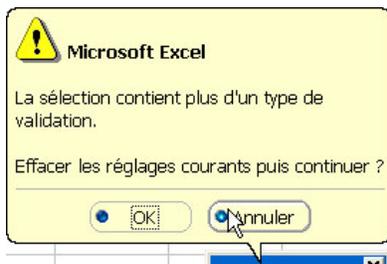
=SI(LC(-1)<>2;LC=997;ET(LC=ENT(LC);LC>=LC(-2);LC<=LC13))

On va ensuite utiliser la technique du copier / coller nous n3B à n6C :

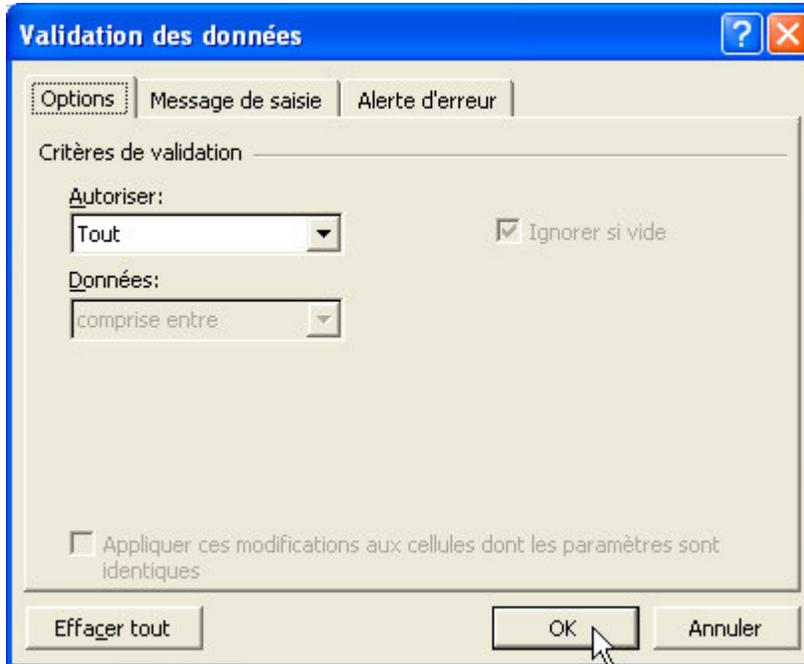
- sélectionner les colonnes 23 à 26 qui contiennent n3A à n6A,
- faire *Édition > Copier*,
- sélectionner la colonne 27 qui contient n3B,
- cliquer sur *Édition > Coller*,
- sélectionner la colonne 31 qui contient n3C,
- et cliquer sur *Édition > Coller*.

Il faut maintenant mettre à jour le nom des variables en remplaçant les A par la bonne lettre. Pour cela, il faut désactiver les critères de validation de la première ligne qui contient le nom des variables :

- sélectionner la première,
- cliquer sur *Données > Validation...*,
- le message suivant apparaît : cliquer sur OK.



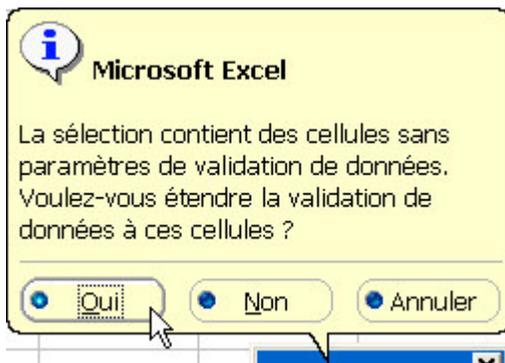
- Autoriser *Tout* et cliquer sur *OK*.



- Mettre à jour le nom des variables en remplaçant A par la lettre correspondante.

Il faut enfin mettre à jour n3B et n3C car un élément de la formule de validation doit être remplacé : 1 devient 2 pour n3B et 3 pour n3C dans LC22<1.

- Sélectionner la colonne qui contient n3B.
- *Données > Validation...*
- Le message suivant apparaît. Cliquer sur *OK*.



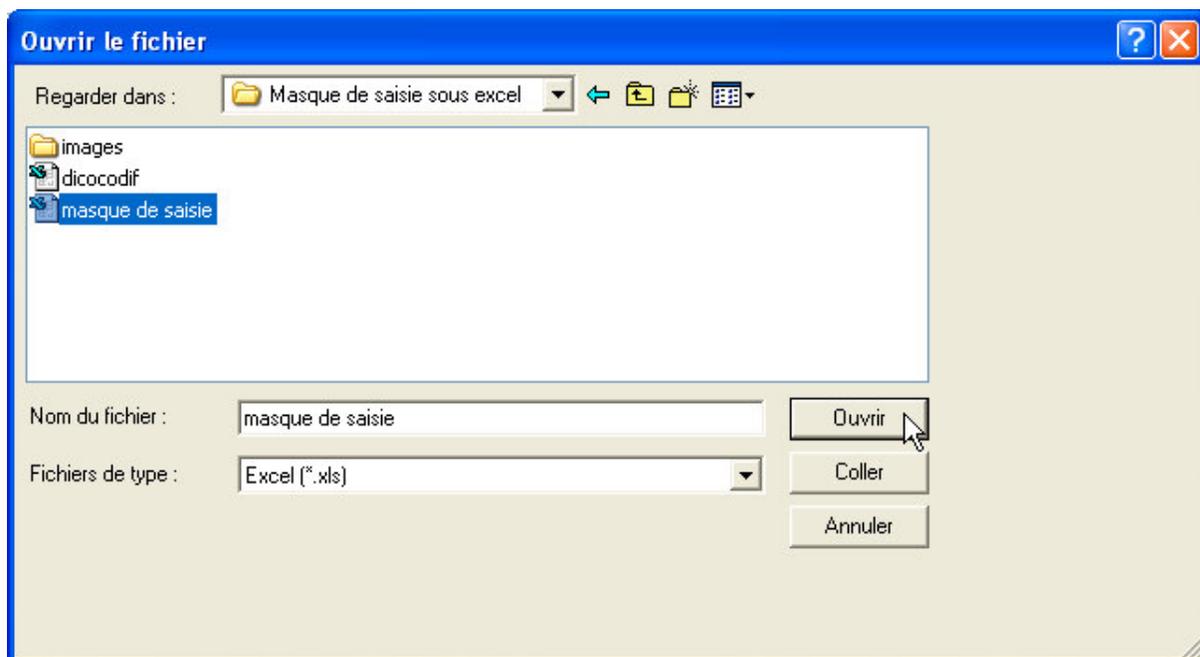
- Faire la modification dans la formule (remplacer LC22 < 1 par LC22 < 2).
- Procéder de même pour n3C.

On peut encore améliorer le masque de saisie en considérant que n4B est supérieur ou égal à n4A (dans le cas où les unions sont saisies dans l'ordre chronologique). Il suffit alors de modifier légèrement les formules de n4B et de n4D.

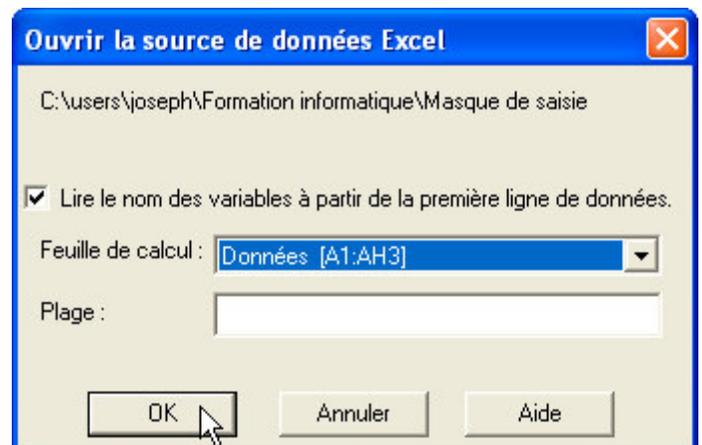
3.5 Récupérer les Données sous SPSS.

Une fois toutes les données saisies, il nous faut les récupérer sous SPSS. L'avantage est que SPSS lit sans problème les fichiers Excel.

- Ouvrir SPSS.
- Aller sur *Fichier > Ouvrir > Données...*
- Choisir *Excel* dans *Fichiers de type*.



- Sélectionner votre fichier et cliquer sur *Ouvrir*.
- La boîte de dialogue suivante apparaît.
- Sélectionner *Lire le nom des variables à partir de la première ligne de données*.
- Choisissez la feuille de calcul qui contient vos données.
- Cliquez sur OK.



Et voilà, le tour est joué.