Travaux pratiques avec Excel 2007 et 2010

Fonctions avancées : traitement des données, travail collaboratif, Windows Live SkyDrive, VBA

Fabrice Lemainque

Copyright © 2012 Dunod.

Toutes les marques citées dans cet ouvrage sont des marques déposées par leurs propriétaires respectifs

Mise en pages réalisée par ARCLEMAX
Image de couverture :
© AshDesign-Fotolia.com

Le pictogramme qui figure ci-contre mérite une explication. Son objet est d'alerter le lecteur sur la menace que représente pour l'avenir de l'écrit, particulièrement dans le domaine de l'édition technique et universitaire, le développement massif du photocopillage.
Le Code de la propriété intellectuelle du 1er juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée dans les établissements d'enseignement supérieur, provoquant une baisse brutale des achats de livres et de revues, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.
Nous rappelons donc que toute reproduction, partielle ou totale, de la présente publication est interdite sans autorisation de l'auteur, de son éditeur ou du Centre français d'exploitation du droit de copie (CFC, 20, rue des Grands-Augustins, 75006 Paris).



© Dunod, Paris, 2012
ISBN 978-2-10-058044-6

Le Code de la propriété intellectuelle n'autorisant, aux termes de l'article L. 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).
Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

Copyright © 2012 Dunod.

Table des matières

Présentation des exercices VI

Avant-propos VIII

Outils avancés de traitement des données

TP 1 Installation du Solveur 3

TP 2 Utilisation du Solveur 5

TP 3 Affichage des solutions intermédiaires du Solveur 7

TP 4 Rapports du Solveur 9

TP 5 Modèle de problème 11

TP 6 Algorithmes de résolution du Solveur 15

TP 7 Gestionnaire de scénario 17

TP 8 Valeur cible 20

TP 9 Table de données 22

TP 10 Utilitaire d'analyse 24

Copyright © 2012 Dunod.

Travail en collaboration

TP 11 Partage de classeur et protection du partage (I) 28
TP 12 Partage de classeur et protection du partage (II) 30
TP 13 Gestion des commentaires 31
TP 14 Suivi et gestion des modifications des utilisateurs (I) 33
TP 15 Suivi et gestion des modifications des utilisateurs (II) 34
TP 16 Acceptation ou refus des modifications 35
TP 17 Inspection d'un classeur 36
TP 18 Finalisation d'un classeur 38
TP 19 Protection d'un classeur 39
TP 20 Protection de l'ensemble d'une feuille de calcul 41
TP 21 Protection d'une partie d'une feuille de calcul 43
TP 22 Afficher les relations entre formules et cellules 45
TP 23 Authentifier un classeur 47
TP 24 Envoi d'un classeur par messagerie électronique 49

Windows Live SkyDrive

TP 25 Windows Live : création d'un identifiant 52
TP 26 Windows Live : stocker un classeur 54
TP 27 Windows Live : ouvrir un classeur dans Excel Web App 56
TP 28 Windows Live : créer un classeur dans Excel Web App 57
TP 29 Windows Live : ouvrir le classeur dans Excel 58
TP 30 Windows Live : ajouter une connexion réseau vers SkyDrive 60
TP 31 Windows Live : partager un dossier 62
TP 32 Windows Live : partager un classeur 63

Copyright © 2012 Dunod.

TP 33 Prérequis au travail avec VBA 67
TP 34 Enregistrement d'une macro 69
TP 35 Macro absolue ou relative 72
TP 36 Modification des propriétés d'une macro 74
TP 37 Emplacement de stockage d'une macro 76
TP 38 Exécuter une macro lors de l'ouverture du classeur 79
TP 39 Édition d'une macro 81
TP 40 Éditeur Visual Basic : fenêtre Projet 83
TP 41 Rédaction d'une macro (I) 86
TP 42 Rédaction d'une macro (II) 87
TP 43 Rédaction d'une fonction personnalisée 88
TP 44 Création d'une procédure VBA pour un événement 90
TP 45 Liste des membres et Information rapide automatique 92
TP 46 Débogage 94
TP 47 Compilateur VBA 96
TP 48 Exécution pas à pas 98
TP 49 Points d'arrêt 101
TP 50 Fenêtre Espions 103
TP 51 Fenêtre Exécution 105
TP 52 Amélioration d'une fonction intégrée d'Excel 106
TP 53 Exportation et importation d'un module de code 108
TP 54 Création d'une interface utilisateur 110
TP 55 Astuces de programmation 116

Index 119

Copyright © 2012 Dunod.

Présentation des exercices

BLOC-NOTES
À chaque début d'exercice un bloc-notes vous indique le numéro du TP. Ce numéro est utile car il sert de référence pour d'autres exercices.

INTRODUCTION
L'introduction procure des indications sur l'exercice à venir et contient un encadré précisant les TP éventuellement prérequis et les fichiers d'exercice employés (à télécharger sur le site www.dunod.com). Le temps de réalisation est une estimation du nombre de minutes à accorder par exercice.

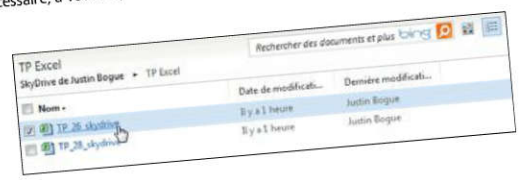
WINDOWS LIVE SKYDRIVE

TP 29 Windows Live : ouvrir le classeur dans Excel

Excel Web App ne dispose pas de toutes les fonctionnalités d'Excel Office. Vous pouvez toutefois ouvrir le classeur situé sur SkyDrive localement dans votre version d'Excel.

Prérequis : TP 25 à 28
Fichier employé : TP_26_skydrive.xlsx
Temps de réalisation : 5 minutes

1 Accédez, si nécessaire, à votre espace SkyDrive, cliquez sur le dossier *TP Excel*, puis sélectionnez le fichier *TP_26_skydrive.xlsx*.



2 Le fichier s'affiche dans Excel Web App. Dans l'onglet *Accueil*, dans le groupe *Office*, cliquez sur *Ouvrir dans Excel*.



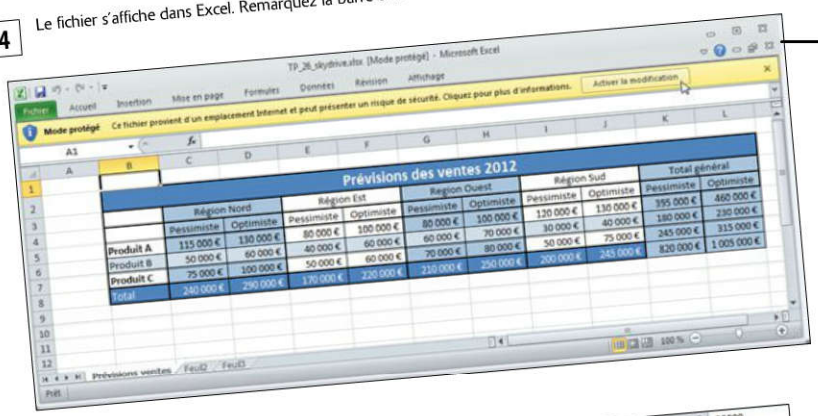
3 Dans la boîte de dialogue d'avertissement qui s'affiche, cliquez sur *OK*. Saisissez dans la boîte de dialogue suivante vos identifiants Windows Live, puis cliquez sur *OK*.



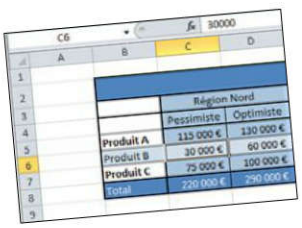
58

Copyright © 2012 Dunod.

4 Le fichier s'affiche dans Excel. Remarquez la barre d'avertissement. Cliquez dedans sur *Activer la modification*.



5 Dans Excel, cliquez sur la cellule *C6* et saisissez **30000**.



6 Dans l'onglet *Fichier*, cliquez sur *Enregistrer*. Le classeur est enregistré automatiquement dans le dossier SkyDrive. Fermez le classeur et Excel.



7 Revenez dans SkyDrive et affichez le fichier Excel dans votre navigateur, à l'aide d'Excel Web App : la modification apportée sous Excel sur votre ordinateur y est présente. Fermez le fichier TP_26_skydrive, puis fermez votre navigateur si vous ne poursuivez pas immédiatement avec l'exercice suivant.



59

WINDOWS LIVE SKYDRIVE

CAPTURES
Les copies d'écran présentées dans ce livre sont réalisées avec Excel 2010 sur un ordinateur exécutant Windows 7. Si vous possédez un autre système d'exploitation ou bien Excel 2007, ou si la fenêtre de votre programme est de taille différente, vous êtes susceptibles de remarquer certaines différences.

ZOOM
Lorsque cela est nécessaire, les captures d'écran sont recadrées sur la zone nécessaire à la compréhension.

PUCE
Les puces des pas à pas vous aident à suivre l'ordre de lecture de l'exercice.

Copyright © 2012 Dunod.

Avant-propos

Excel est une boîte à outils presque universelle à laquelle de nombreux utilisateurs ont recours dans de très nombreuses circonstances. Après avoir employé ce produit de façon traditionnelle, vous envisagez de passer à un emploi plus avancé : tel est le but de ce livre.

Tout au long de cet ouvrage, nous supposons que vous maîtrisez déjà les techniques de base de l'utilisation aussi bien de Windows que d'Excel, et qu'une version d'Excel 2007 ou 2010 est déjà installée sur votre ordinateur.

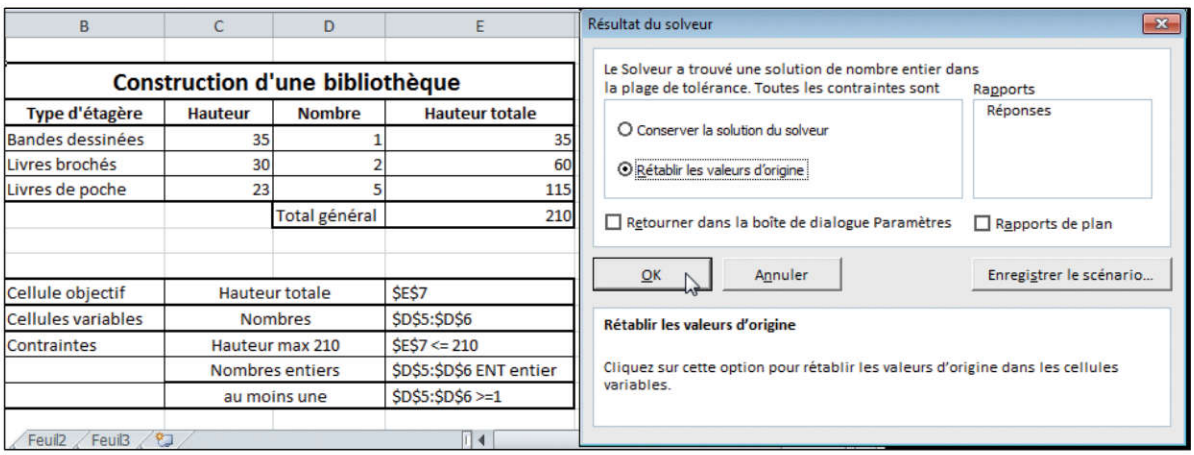
Vous trouverez sur le site Dunod.com les fichiers d'exercices employés dans ce livre, que vous devrez télécharger et installer sur votre ordinateur. Je vous recommande de les intégrer dans un dossier dédié dans *Documents*, par exemple TP.

Copyright © 2012 Dunod.

Outils avancés de traitement des données

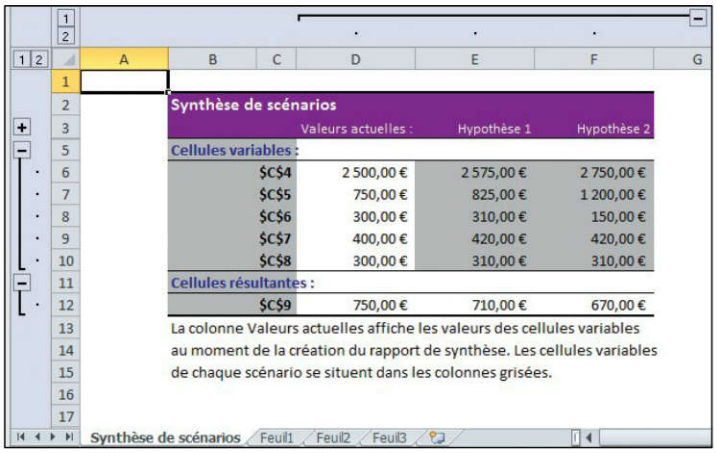
Excel propose plusieurs outils, souvent nommés outils d'analyses de simulation, pour réaliser facilement des simulations ou trouver la réponse nécessaire à la résolution de problèmes complexes.

Parmi ceux-ci, le Solveur permet de trouver une valeur optimale (maximale ou minimale) pour une formule dans une seule cellule, appelée cellule objectif (autrefois nommé « cellule cible »), en fonction de contraintes ou de limites appliquées aux valeurs d'autres cellules de la formule dans une feuille de calcul. Le Solveur utilise un groupe de cellules, appelées variables de décision ou simplement cellules variables, qui interviennent dans le calcul des formules contenues dans les cellules objectif et de contraintes. Le Solveur affine les valeurs des cellules variables de décision pour correspondre aux limites appliquées aux cellules de contraintes et produire le résultat souhaité pour la cellule objectif. Excel 2010 propose une nouvelle version du complément Solveur Excel, qui permet d'effectuer une analyse de scénarios sur vos données.



Le Gestionnaire de scénario permet, comme son nom l'indique, de comparer plusieurs scénarios, c'est-à-dire des ensembles de valeurs enregistrées par Excel. Vous pouvez créer et enregistrer les différents groupes de valeurs dans une feuille de calcul, puis passer d'un scénario à l'autre pour visualiser les différents résultats.

Copyright © 2012 Dunod.

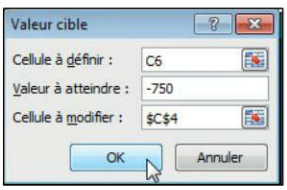


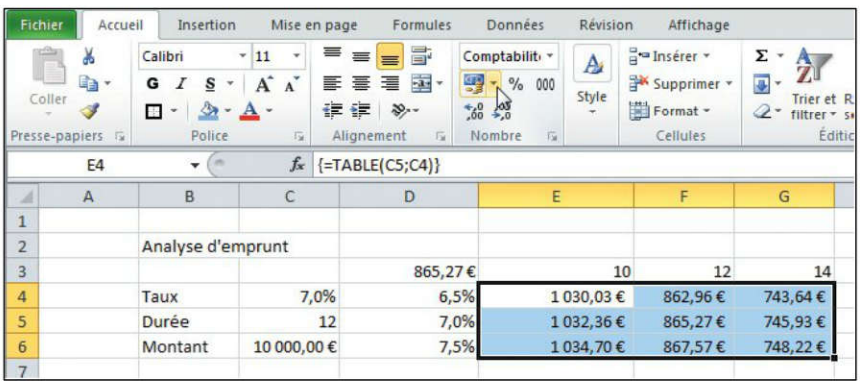
Synthèse de scénarios	Valeurs actuelles :	Hypothèse 1	Hypothèse 2
Cellules variables :			
$C$4	2 500,00 €	2 575,00 €	2 750,00 €
$C$5	750,00 €	825,00 €	1 200,00 €
$C$6	300,00 €	310,00 €	150,00 €
$C$7	400,00 €	420,00 €	420,00 €
$C$8	300,00 €	310,00 €	310,00 €
Cellules résultantes :			
$C$9	750,00 €	710,00 €	670,00 €

La colonne Valeurs actuelles affiche les valeurs des cellules variables au moment de la création du rapport de synthèse. Les cellules variables de chaque scénario se situent dans les colonnes grisées.

Synthèse de scénarios | Feuil1 | Feuil2 | Feuil3

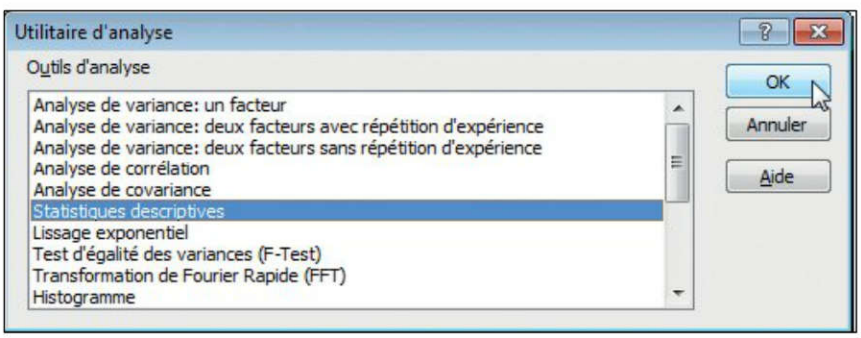
L'outil Valeur cible permet de déterminer une (et une seule) valeur permettant d'aboutir à un résultat recherché dans une (et une seule) autre cellule.





	Analyse d'emprunt		865,27 €	10	12	14
	Taux	7,0%	6,5%	1 030,03 €	862,96 €	743,64 €
	Durée	12	7,0%	1 032,36 €	865,27 €	745,93 €
	Montant	10 000,00 €	7,5%	1 034,70 €	867,57 €	748,22 €

Lorsqu'une formule a recours à un maximum de deux variables, ou si plusieurs formules se servent toutes d'une même variable, une table de données permet d'afficher tous les résultats en un emplacement unique.



Enfin, l'outil Utilitaire d'analyse, disponible dans le groupe *Analyse* de l'onglet *Données*, regroupe de nombreuses fonctions statistiques complexes que vous pouvez appliquer à vos données à l'aide d'assistants. Certains outils génèrent des graphiques.

Copyright © 2012 Dunod.

TP 1 Installation du Solveur

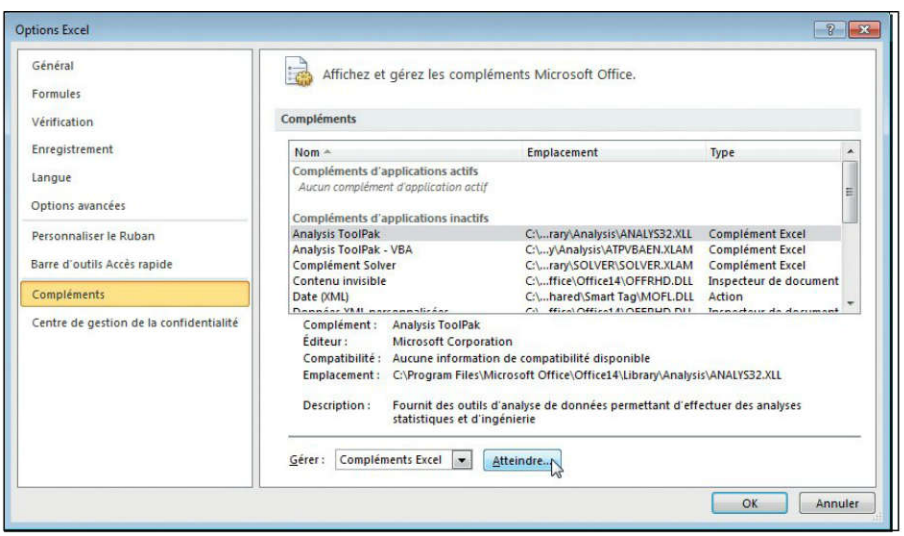
Le complément Solveur n'est pas installé par défaut. Vous devez le charger avant de pouvoir vous servir de ce précieux outil. Tel est l'objet de ce TP préliminaire.

Prérequis : néant
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Cliquez sur l'onglet *Fichier*, puis cliquez sur *Options*.

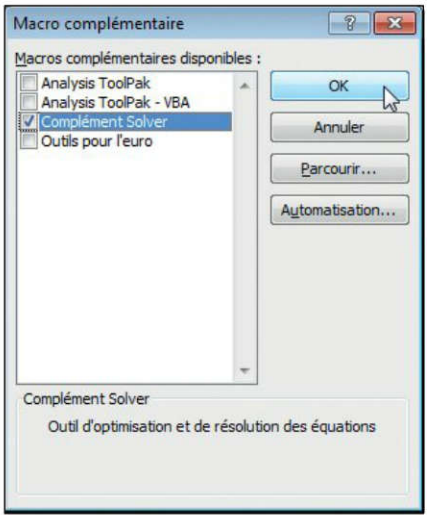


2 Cliquez à gauche sur la catégorie *Compléments*. Dans le volet de droite, en bas, dans la zone *Gérer*, cliquez sur *Compléments Excel*, puis sur *Atteindre*.



Copyright © 2012 Dunod.

3 Dans la zone *Macros complémentaires disponibles*, activez la case à cocher *Complément Solveur*, puis cliquez sur *OK*.



4 Cliquez sur l'onglet *Données* : vous voyez alors apparaître à droite dans le groupe *Analyse* le bouton *Solveur*. Vous pouvez désormais employer ce précieux outil à votre guise.



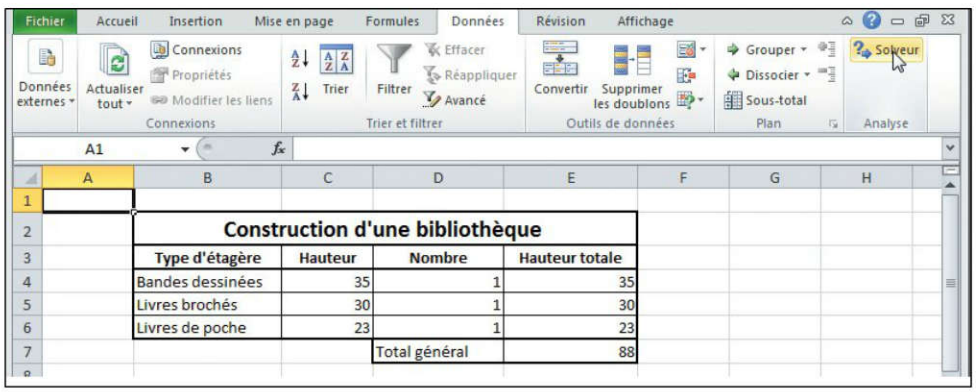
Copyright © 2012 Dunod.

TP 2 Utilisation du Solveur

Le solveur permet de résoudre des problèmes complexes. Dans ce TP, nous supposons que vous voulez placer des étagères pour créer une bibliothèque renfermant des bandes dessinées, des livres brochés et des livres de poche. Vous souhaitez pouvoir ranger un maximum de livres tout en minimisant la place perdue, mais en occupant tout de même le maximum de la hauteur disponible. Comment y parvenir ? Grâce au Solveur…

Prérequis : TP 1
Fichier employé : TP_02_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le fichier TP_02_début.xlsx et enregistrez-le sous le nom TP_02.xlsx. Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*.

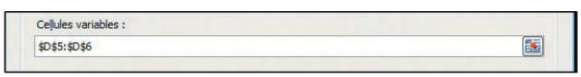


	A	B	C	D	E	F	G	H
1								
2		Construction d'une bibliothèque						
3		Type d'étagère	Hauteur	Nombre	Hauteur totale			
4		Bandes dessinées	35	1	35			
5		Livres brochés	30	1	30			
6		Livres de poche	23	1	23			
7				Total général	88			

2 Dans la boîte de dialogue *Paramètres du solveur*, dans la zone *Objectif à définir*, saisissez **$E$7.** Cliquez sur *Max*, pour indiquer que la valeur de la cellule objectif doit être aussi élevée que possible (dans d'autres cas, vous cliqueriez sur Min ou sur Valeur, en entrant dans le cas présent une valeur dans la zone).



3 Dans la zone *Cellules variables*, saisissez **$D$5:$D$6**.



4 Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur *Ajouter*, puis, dans la zone *Référence de cellule*, saisissez **$E$7**. Cliquez sur la relation <= et saisissez **210** dans la zone *Contrainte*.



5 Cliquez sur *Ajouter*. Dans la zone *Référence de cellule*, saisissez **$D$5:$D$6**. Cliquez sur la relation *ent* et entier s'affiche dans la zone *Contrainte*.

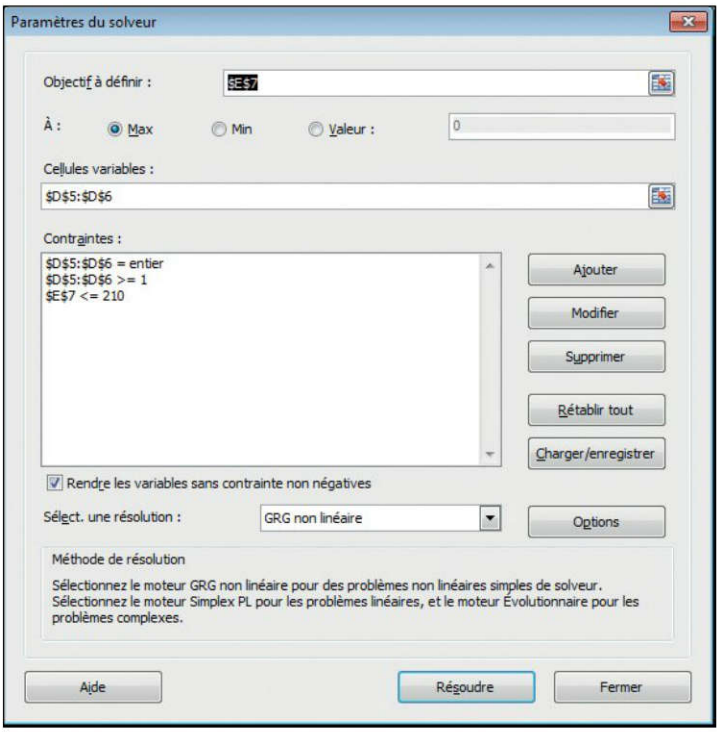


Copyright © 2012 Dunod.

6 Cliquez sur *Ajouter*. Dans la zone *Référence de cellule*, saisissez **$D$5:$D$6**. Cliquez sur la relation >= et saisissez **1** dans la zone *Contrainte*.

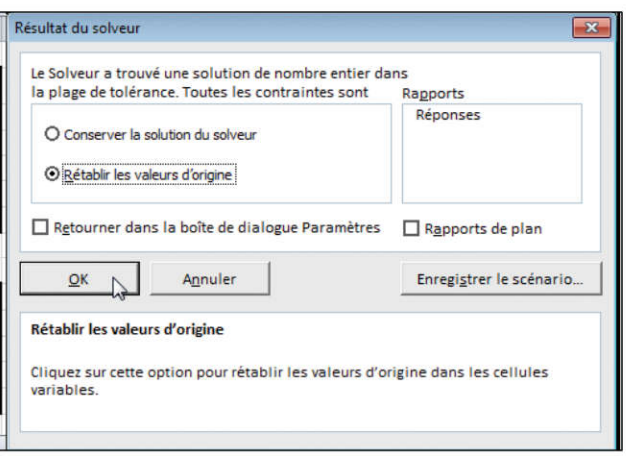
Vous pouvez à tout moment modifier ou supprimer une contrainte existante en cliquant sur celle-ci, puis en cliquant sur *Modifier* ou *Supprimer*.





7 Cliquez sur *OK* pour accepter la dernière contrainte et revenir à la boîte de dialogue *Paramètres du solveur*. Cliquez sur *Résoudre*.

	B	C	D	E
	Construction d'une bibliothèque			
	Type d'étagère	**Hauteur**	**Nombre**	**Hauteur totale**
	Bandes dessinées	35	1	35
	Livres brochés	30	2	60
	Livres de poche	23	5	115
			Total général	210
	Cellule objectif	Hauteur totale		$E$7
	Cellules variables	Nombres		$D$5:$D$6
	Contraintes	Hauteur max 210		$E$7 <= 210
		Nombres entiers		$D$5:$D$6 ENT entier
		au moins une		$D$5:$D$6 >=1



8 Observez les résultats puis, dans la boîte de dialogue *Résultat du solveur*, cliquez sur *Rétablir les valeurs d'origine*. Pour conserver les valeurs trouvées par le Solveur, il faudrait cliquer sur *Garder la solution du solveur*.
Enregistrez le classeur TP_02 et fermez-le si vous ne poursuivez pas avec le TP suivant.

Copyright © 2012 Dunod.

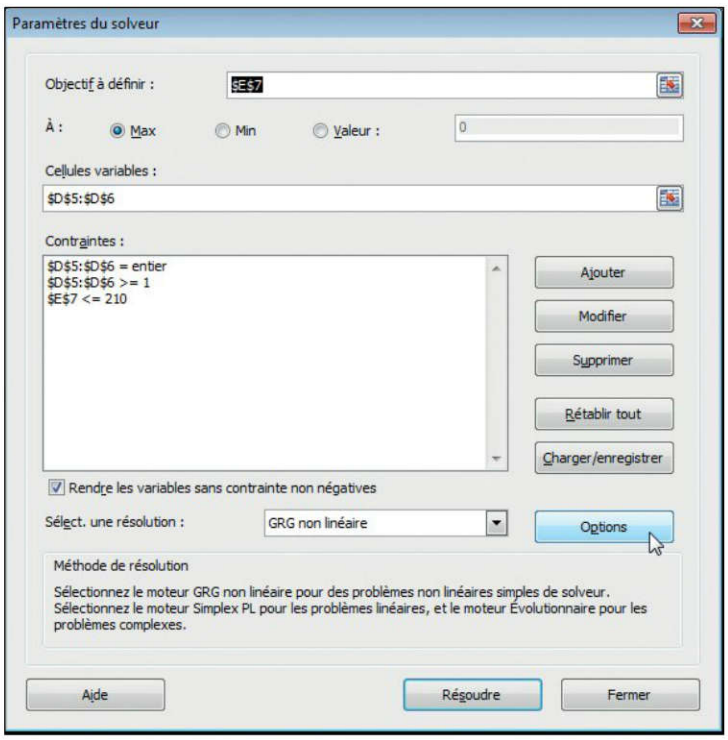
TP 3 Affichage des solutions intermédiaires du Solveur

Il est parfois intéressant d'examiner les solutions intermédiaires du Solveur afin de prendre exemple sur la façon dont celui-ci tente de résoudre le problème posé. C'est ce que nous allons voir dans ce TP.

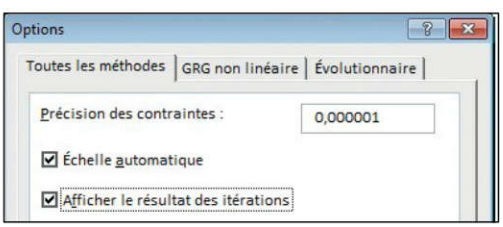
Prérequis : TP 1 et 2
Fichier employé : TP_03_début.xlsx
Temps de réalisation : 5 minutes

1 Si le classeur TP_02 n'est pas ouvert, ouvrez le fichier TP_03_début.xlsx. Enregistrez le classeur actif sous le nom TP_03.xlsx.

Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*, puis sur *Options*, dans la boîte de dialogue *Paramètres du solveur*.

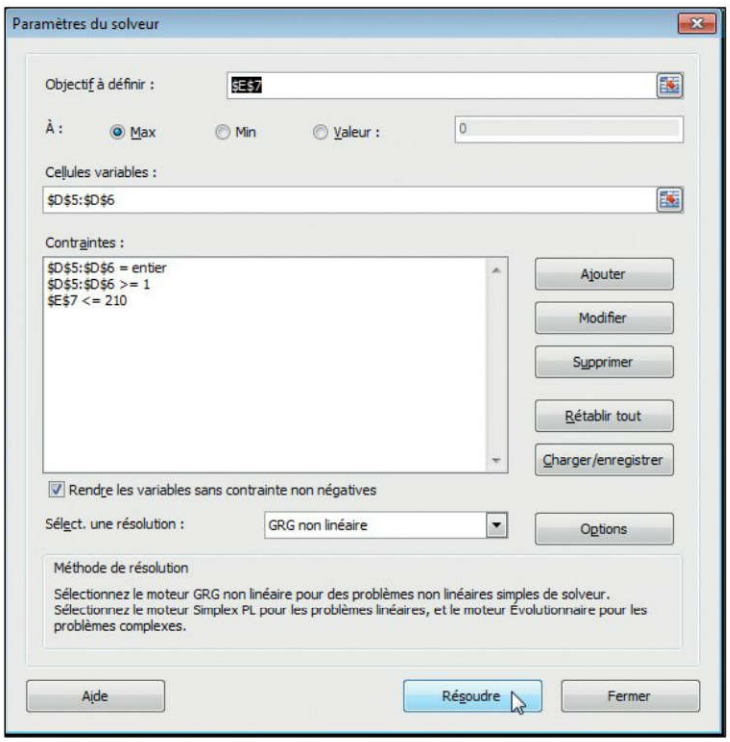


2 Dans la boîte de dialogue *Options*, cochez la case *Afficher le résultat des itérations*, puis cliquez sur *OK*.

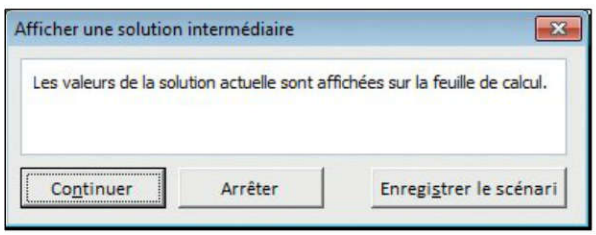


Copyright © 2012 Dunod.

3 Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur *Résoudre*.



4 La boîte de dialogue *Affichage d'une solution intermédiaire* qui apparaît signale que la solution intermédiaire actuelle est affichée sur la feuille de calcul. Examinez-la, puis cliquez sur *Continuer*. Répétez l'opération jusqu'à obtenir la boîte de dialogue *Résultat du solveur*. Cliquez sur *Rétablir les valeurs d'origine*, cliquez sur *OK* et fermez le classeur TP_03.



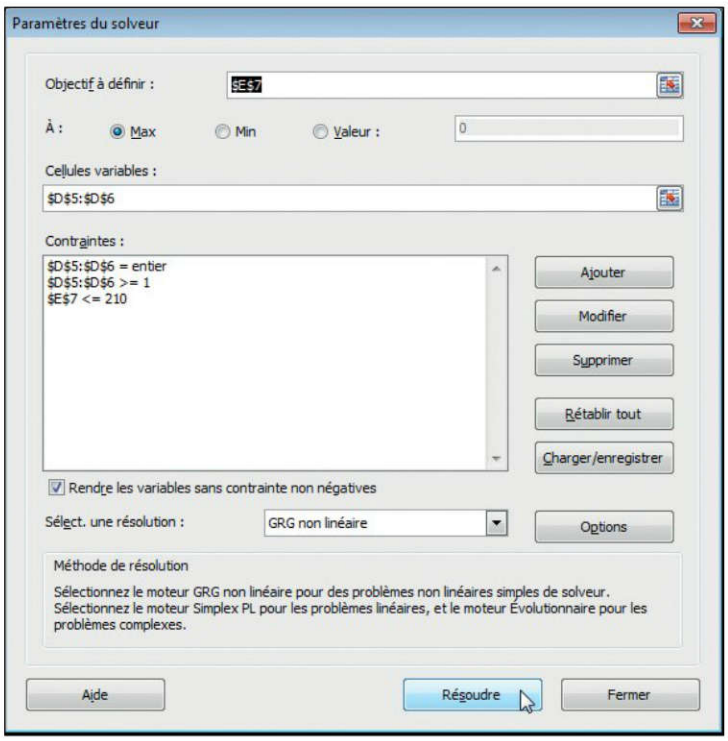
Copyright © 2012 Dunod.

TP 4 Rapports du Solveur

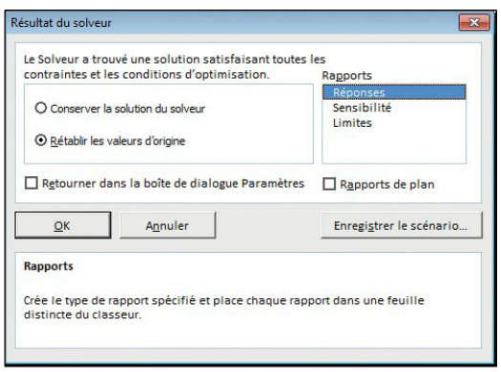
Vous pouvez générer un rapport fondé sur une solution identifiée par le Solveur. Selon le type de problème et les solutions identifiées, le Solveur peut proposer plusieurs types de rapports, apportant des informations complémentaires sur les résultats de l'outil.

Prérequis : TP 1 et 2
Fichier employé : TP_04_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_04_début.xlsx et enregistrez-le sous le nom TP_04.xlsx.
Cliquez sur la feuille *Tous rapports*. Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*. Cliquez sur *Résoudre*.

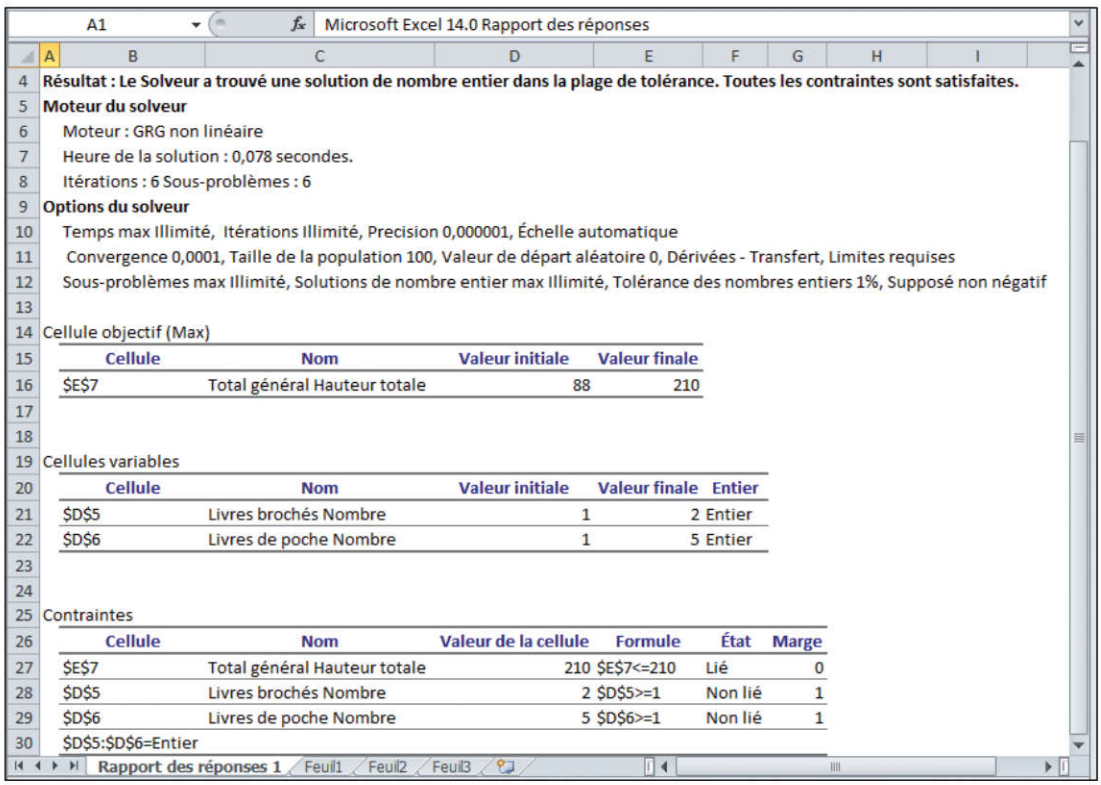


2 Dans la boîte de dialogue *Résultats du solveur*, dans la zone *Rapports*, trois rapports sont proposés : Réponses, Sensibilité et Limites. Cliquez sur *Réponses*, puis cliquez sur *OK*.



Copyright © 2012 Dunod.

3 Le rapport est créé dans une nouvelle feuille de calcul. Vous pouvez, si vous le souhaitez, tester les autres types de rapports, mais Réponses est généralement le plus utile.



A1 — Microsoft Excel 14.0 Rapport des réponses

Résultat : Le Solveur a trouvé une solution de nombre entier dans la plage de tolérance. Toutes les contraintes sont satisfaites.

Moteur du solveur

Moteur : GRG non linéaire
Heure de la solution : 0,078 secondes.
Itérations : 6 Sous-problèmes : 6

Options du solveur

Temps max Illimité, Itérations Illimité, Precision 0,000001, Échelle automatique
Convergence 0,0001, Taille de la population 100, Valeur de départ aléatoire 0, Dérivées - Transfert, Limites requises
Sous-problèmes max Illimité, Solutions de nombre entier max Illimité, Tolérance des nombres entiers 1%, Supposé non négatif

Cellule objectif (Max)

Cellule	Nom	Valeur initiale	Valeur finale
$E$7	Total général Hauteur totale	88	210

Cellules variables

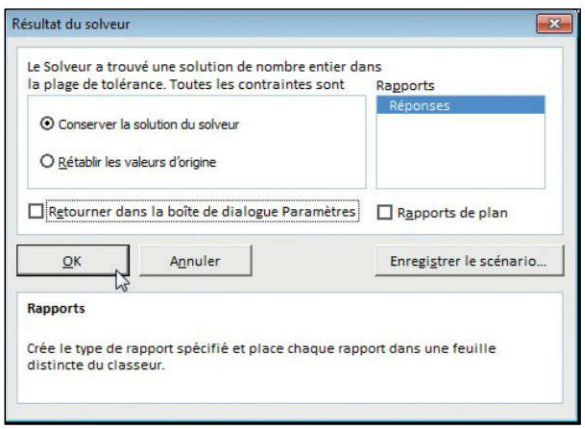
Cellule	Nom	Valeur initiale	Valeur finale	Entier
$D$5	Livres brochés Nombre	1	2	Entier
$D$6	Livres de poche Nombre	1	5	Entier

Contraintes

Cellule	Nom	Valeur de la cellule	Formule	État	Marge
$E$7	Total général Hauteur totale	210	$E$7<=210	Lié	0
$D$5	Livres brochés Nombre	2	$D$5>=1	Non lié	1
$D$6	Livres de poche Nombre	5	$D$6>=1	Non lié	1
$D$5:$D$6=Entier					

Rapport des réponses 1 | Feuil1 | Feuil2 | Feuil3

4 Cliquez sur la feuille *Un rapport*, puis répétez les étapes 1 et 2 : seul le rapport Réponses est proposé, en raison de la contrainte Entier appliquée aux cellules variables. Les rapports disponibles dépendent du type de problème. Si le Solveur ne trouve pas de solution, il se peut qu'aucun rapport ne soit disponible.
Enregistrez et fermez le classeur TP_04.xlsx.



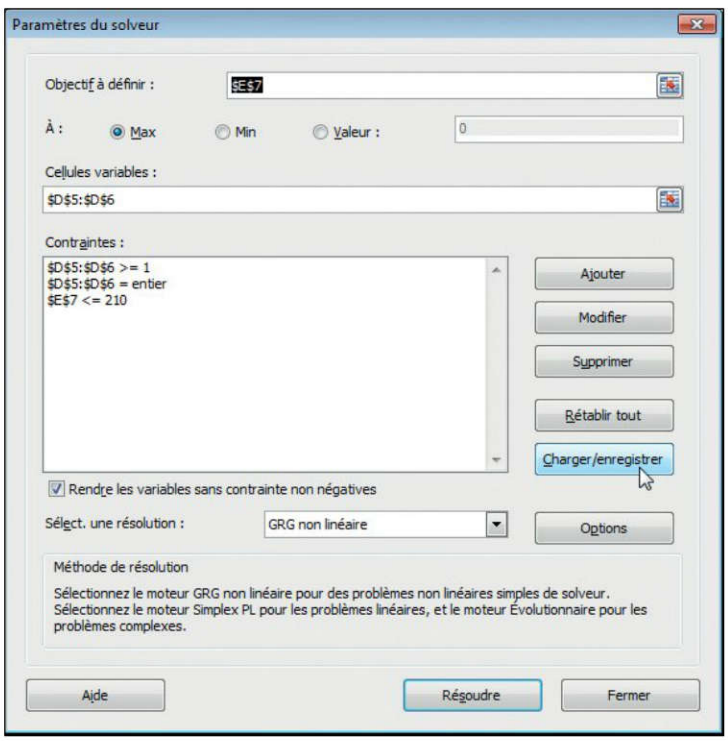
Copyright © 2012 Dunod.

TP 5 Modèle de problème

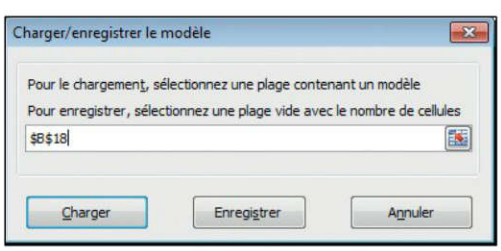
Il est parfois intéressant de pouvoir enregistrer un modèle de problème pour pouvoir le réemployer ultérieurement ou pour comparer des résultats obtenus en modifiant certains paramètres. Dans ce TP, nous allons émettre l'hypothèse que la hauteur totale occupée par la bibliothèque peut être changée.

Prérequis : TP 1 et 2
Fichier employé : TP_05_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le fichier TP_05_début.xlsx et enregistrez-le sous le nom TP_05.xlsx.
Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*. Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur *Charger/enregistrer*.

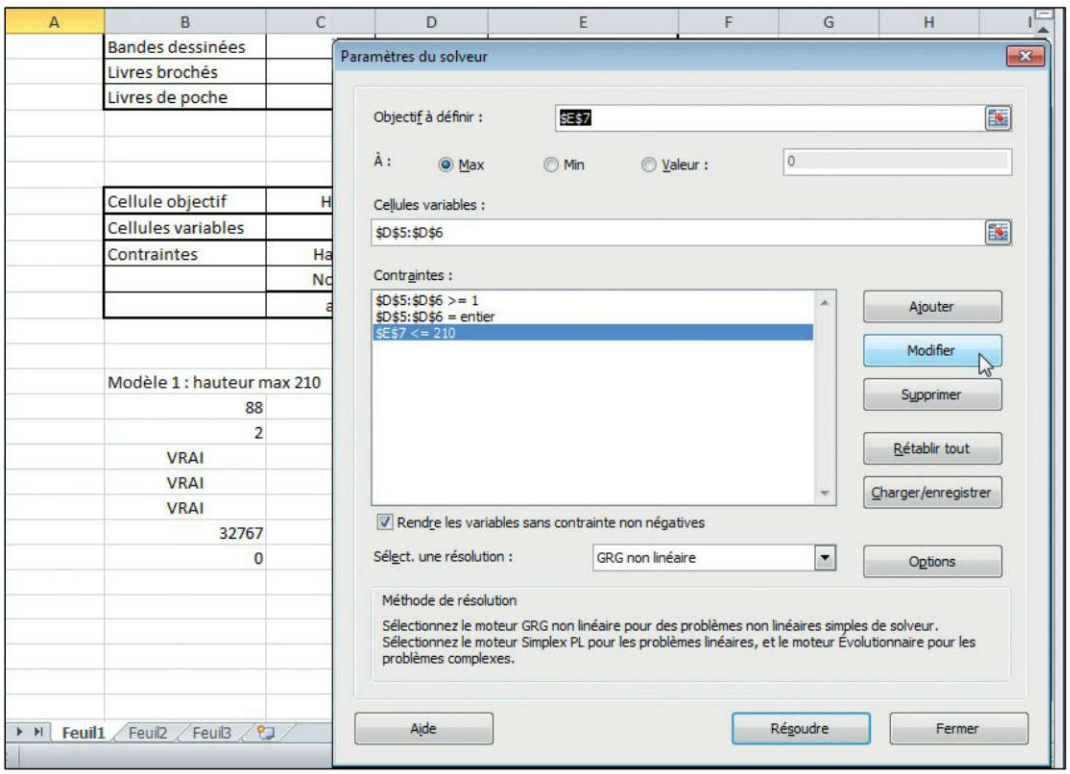


2 Dans la boîte de dialogue, dans la zone de modèle, saisissez **$B$18**, puis cliquez sur *Enregistrer*. Lorsque vous enregistrez un modèle, la référence doit être la première cellule d'une plage verticale de cellules vides.



Copyright © 2012 Dunod.

3 Le modèle apparaît dans votre feuille de calcul à l'emplacement spécifié. Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur la contrainte de la cellule *$E$17*, puis sur *Modifier*.



4 Modifiez la valeur en **225**.

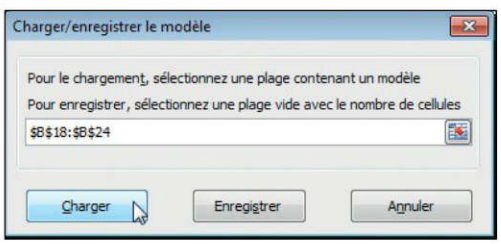


Copyright © 2012 Dunod.

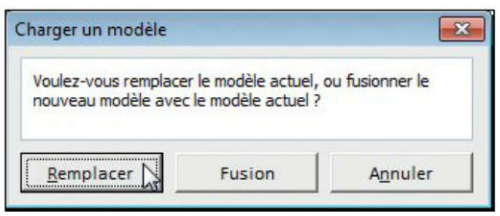
5 Cliquez sur *Charger/enregistrer*. Dans la boîte de dialogue, dans la zone de modèle, saisissez **$D$18**, puis cliquez sur *Enregistrer*. Le modèle apparaît dans votre feuille de calcul à l'emplacement indiqué. Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur *Résoudre* : vous obtenez la solution pour cette nouvelle hauteur.

	A	B	C	D	E	F
1						
2		**Construction d'une bibliothèque**				
3		**Type d'étagère**	**Hauteur**	**Nombre**	**Hauteur totale**	
4		Bandes dessinées	35	1	35	
5		Livres brochés	30	4	120	
6		Livres de poche	23	3	69	
7				Total général	224	
8						
9						
10		Cellule objectif	Hauteur totale		$E$7	
11		Cellules variables	Nombres		$D$5:$D$6	
12		Contraintes	Hauteur max 210		$E$7 <= 210	
13			Nombres entiers		$D$5:$D$6 ENT entier	
14			au moins une		$D$5:$D$6 >=1	
15						
16						
17		Modèle 1 : hauteur max 210		Modèle 2 : hauteur max 225		
18		224		224		
19		2		2		
20		VRAI		VRAI		
21		VRAI		VRAI		
22		FAUX		VRAI		
23		32767		32767		
24		0		0		

6 Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur *Charger/enregistrer*. Lorsque vous chargez un modèle, vous devez indiquer la plage de cellules contenant le modèle de problème. Dans la boîte de dialogue, dans la zone de modèle, saisissez **$B$18:$B$24**, puis cliquez sur *Charger*.

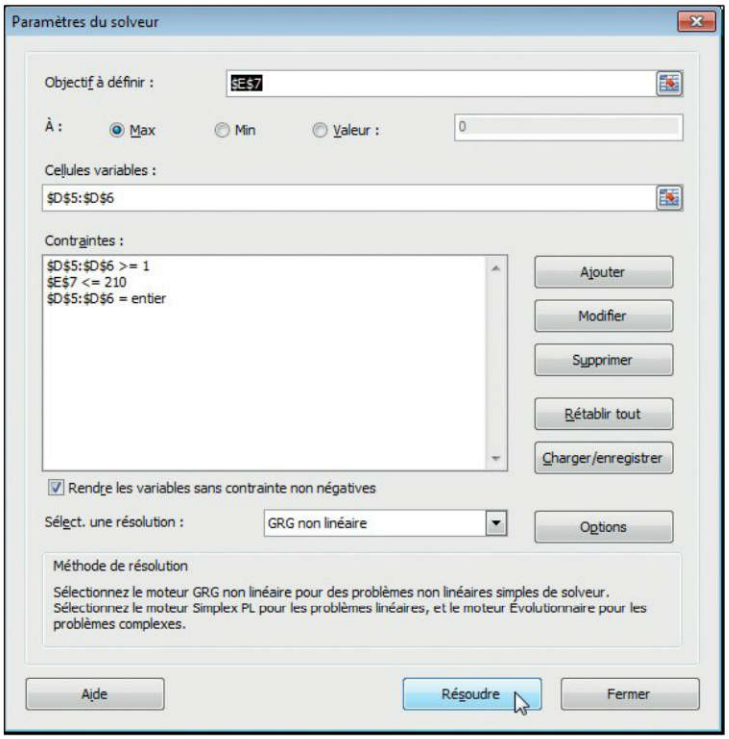


7 Une boîte de dialogue apparaît vous demandant si vous voulez remplacer ou fusionner le modèle chargé avec le modèle actuel. Cliquez sur *Remplacer* (Fusion pourrait être un moyen d'introduire des contraintes complémentaires, mais elles sont ici contradictoires).



Copyright © 2012 Dunod.

8 Vous voyez dans la boîte de dialogue *Paramètres du solveur* que l'ancienne contrainte est de nouveau présente. Cliquez sur *Résoudre*, puis sur *Conserver les résultats du solveur* pour revenir aux résultats du premier modèle.



Fermez et enregistrez le classeur TP_05.slsx.

Lors d'un enregistrement, une feuille de calcul comprend les dernières sélections effectuées dans la boîte de dialogue *Paramètres du solveur* en enregistrant le classeur. Chaque feuille de calcul d'un classeur peut posséder ses propres sélections de Solveur.

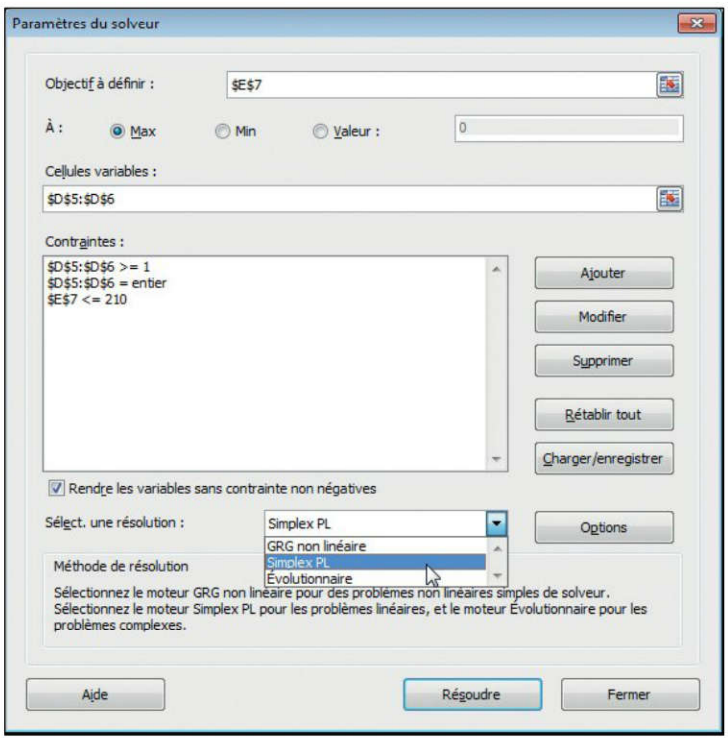
Copyright © 2012 Dunod.

TP 6 Algorithmes de résolution du Solveur

Le solveur dispose de trois algorithmes de résolution disponibles, à employer selon le type de problème à résoudre, que ce TP va présenter. Le choix de l'algorithme joue sur la pertinence de la solution identifiée, ainsi que sur le temps de résolution.

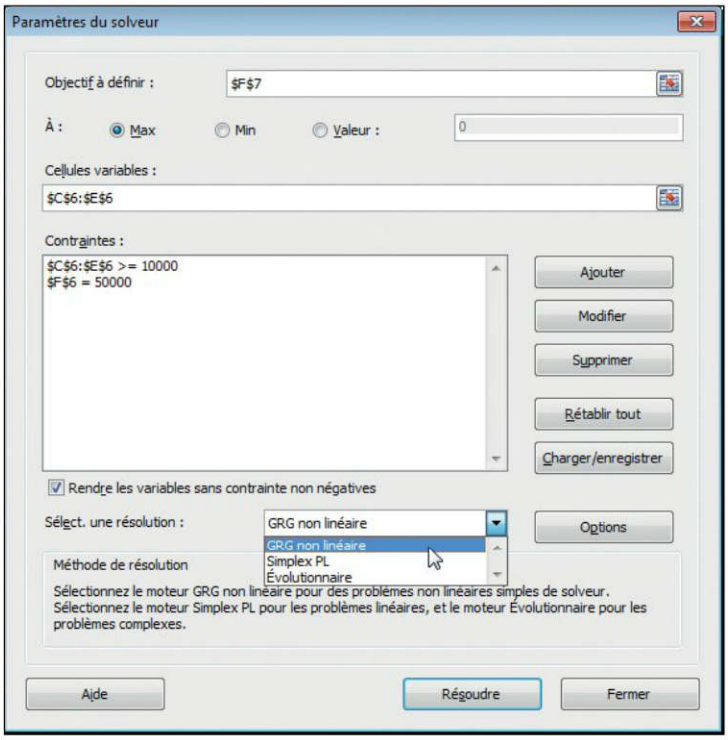
Prérequis : TP 1 et 2
Fichier employé : TP_05_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_06_début.xlsx et enregistrez-le sous le nom TP_06.xlsx.
Cliquez sur la feuille *Linéaire*. Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*. Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur la flèche située à côté de *Sélect. une résolution* pour ouvrir le menu déroulant. Le problème est ici de type linéaire (comme la formule de la cellule objectif) : choisissez *Simplex PL*, puis cliquez sur *Résoudre*.

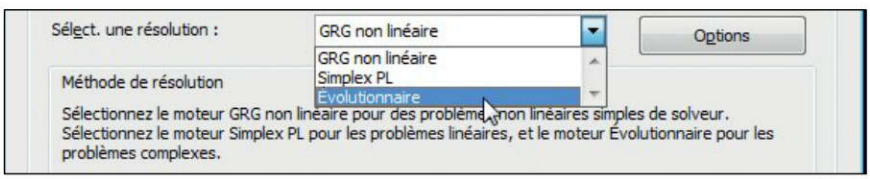


Copyright © 2012 Dunod.

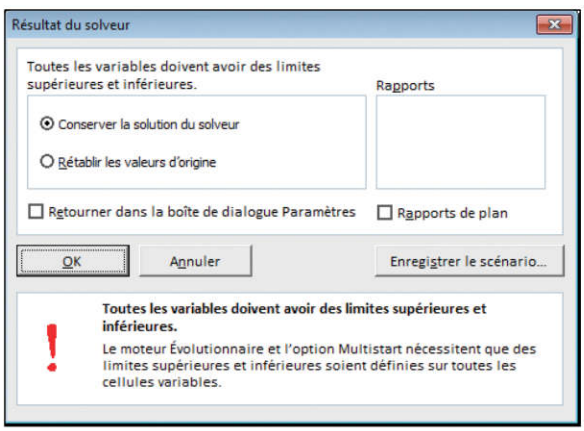
2 Cliquez sur la feuille *Non linéaire*. Sous l'onglet *Données*, cliquez sur *Solveur* dans le groupe *Analyse*. Dans la boîte de dialogue *Paramètres du solveur*, cliquez sur la flèche située à côté de *Sélect. une résolution* pour faire apparaître le menu déroulant. Le problème n'est pas cette fois-ci de type linéaire (des formules du tableau emploient des instructions SI0) : choisissez *GRG non linéaire*, puis cliquez sur *Résoudre*.



3 Cliquez sur la feuille *Évolutionnaire*. Choisissez à présent une résolution de type *Évolutionnaire*, puis cliquez sur *Résoudre*.



4 Une boîte de dialogue s'affiche signalant que ce type de résolution ne peut être employé qu'avec des cellules variables dotées de bornes maximales et minimales. Ajoutez la contrainte **$C$6:$E$6 <+50000**, puis cliquez sur *Résoudre*.
Le type *Évolutionnaire* est à réserver à des cas complexes, lorsque les autres types s'avèrent inadaptés ou sans résultat.



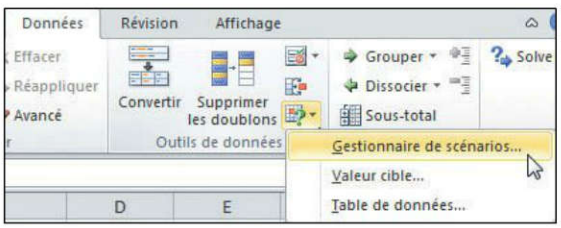
Copyright © 2012 Dunod.

TP 7 Gestionnaire de scénario

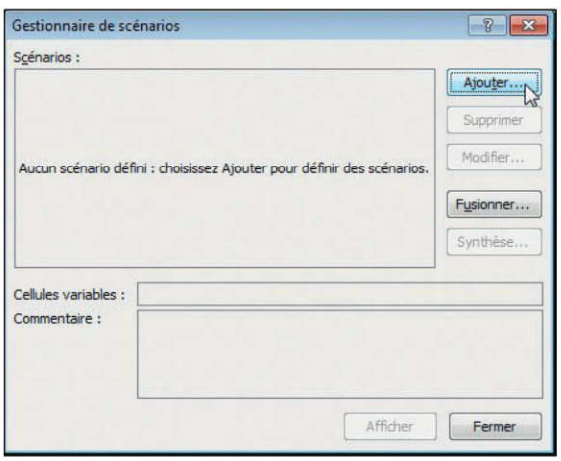
Un scénario est un ensemble de valeurs enregistrées par Excel. Vous pouvez créer et enregistrer différents groupes de valeurs dans une feuille de calcul, puis passer d'un scénario à l'autre pour visualiser les différents résultats. Un scénario accepte un maximum de 72 variables. Pour ce TP, nous allons comparer deux budgets prévisionnels : le maintien d'une situation actuelle, avec quelques augmentations de certains postes, et une promotion assortie d'un déménagement dans un domicile plus grand, situé plus près du lieu de travail, ce qui entraîne la variation de certains postes. Au final, quelle solution entraîne un accroissement des revenus disponibles ?

Prérequis : néant
Fichier employé : TP_07_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_07_début.xlsx et enregistrez-le sous le nom TP_07.xlsx.
Cliquez sur *Analyse scénarios* dans le groupe *Outils de données* de l'onglet *Données*, puis cliquez sur *Gestionnaire de scénarios*.

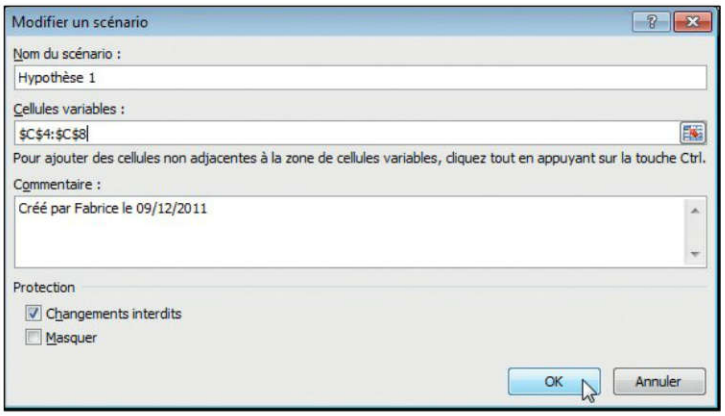


2 Dans la boîte de dialogue *Gestionnaire de scénarios* qui s'affiche, cliquez sur *Ajouter*.

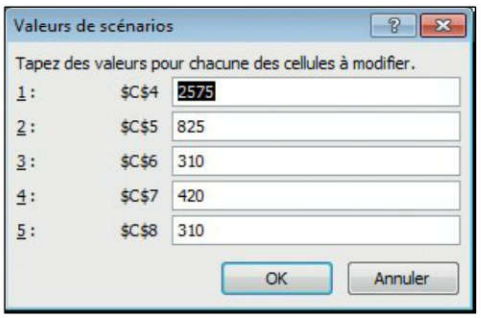


Copyright © 2012 Dunod.

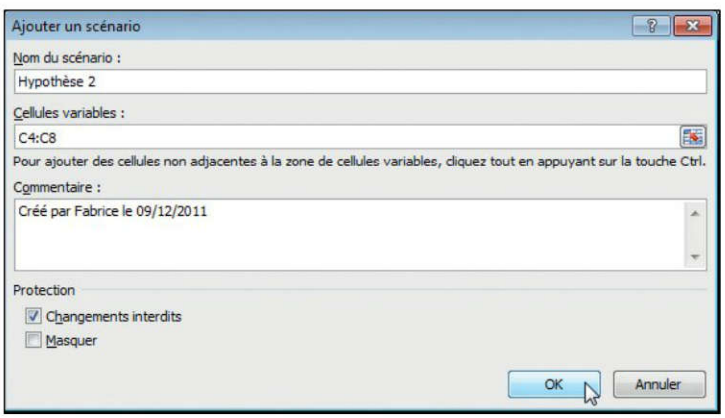
3 Dans la boîte de dialogue *Ajouter un scénario*, saisissez **Hypothèse 1** dans la zone *Nom du scénario*, puis saisissez **$C$4:$C$8** dans la zone *Cellules variables*. Cliquez sur *OK*.



4 Dans la boîte de dialogue *Valeurs de scénario*, vous voyez s'afficher les valeurs actuelles de chaque cellule variable. Saisissez les valeurs présentées dans la figure ci-dessus, puis cliquez sur *Ajouter*.

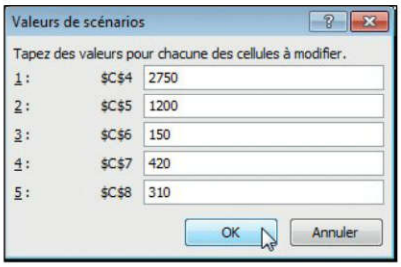


5 Dans la boîte de dialogue *Ajouter un scénario*, saisissez dans la zone **Hypothèse 2** *Nom du scénario*. La zone *Cellules variables* est déjà remplie. Cliquez sur *OK*.

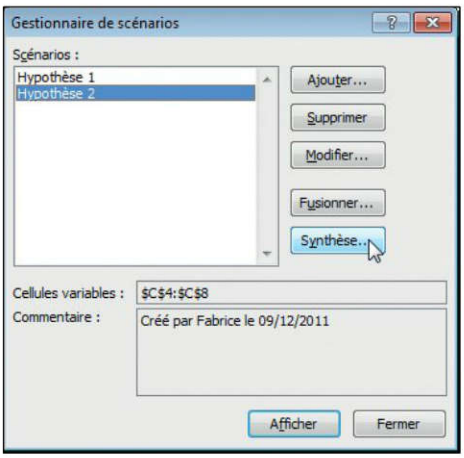


Copyright © 2012 Dunod.

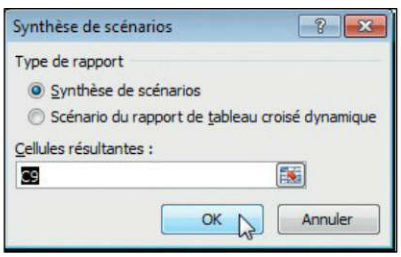
6 Dans la boîte de dialogue *Valeurs de scénario*, vous voyez toujours les valeurs actuelles de chaque cellule variable. Saisissez les valeurs présentées dans la figure ci-contre, puis cliquez sur *OK*.



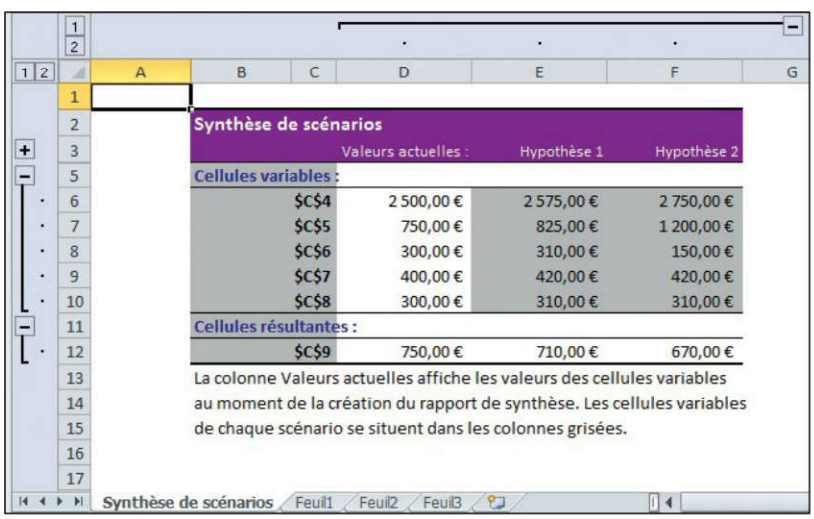
7 Vous revenez à la boîte de dialogue *Gestionnaire de scénarios*, présentant les noms de vos deux scénarios. À l'aide des boutons de droite, vous pouvez modifier, supprimer ou ajouter un autre scénario. Cliquez sur *Synthèse*.



8 Dans la boîte de dialogue *Synthèse de scénarios*, saisissez **C9** dans la zone *Cellule résultante*, puis cliquez sur *OK*.



9 La Synthèse de scénarios s'affiche dans une nouvelle feuille, comparant les différents scénarios par rapport à la situation originelle.
Enregistrez-la et fermez le classeur TP_07.



Synthèse de scénarios			
	Valeurs actuelles :	Hypothèse 1	Hypothèse 2
Cellules variables :			
$C$4	2 500,00 €	2 575,00 €	2 750,00 €
$C$5	750,00 €	825,00 €	1 200,00 €
$C$6	300,00 €	310,00 €	150,00 €
$C$7	400,00 €	420,00 €	420,00 €
$C$8	300,00 €	310,00 €	310,00 €
Cellules résultantes :			
$C$9	750,00 €	710,00 €	670,00 €

La colonne Valeurs actuelles affiche les valeurs des cellules variables au moment de la création du rapport de synthèse. Les cellules variables de chaque scénario se situent dans les colonnes grisées.

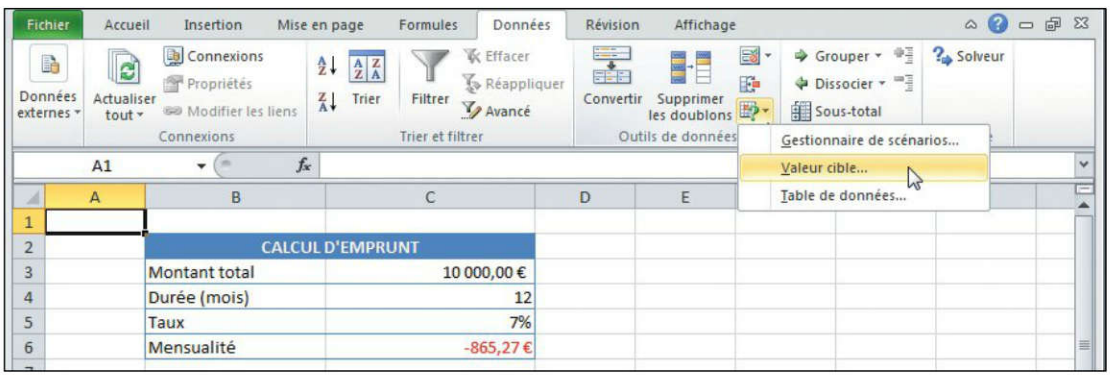
Copyright © 2012 Dunod.

TP 8 Valeur cible

L'outil Valeur cible permet de déterminer une (et une seule) valeur permettant d'aboutir à un résultat recherché dans une (et une seule) autre cellule. La cellule *résultat* peut renfermer une formule fondée sur de nombreuses cellules, mais, contrairement au Solveur, une seule cellule peut être ici considérée comme cellule variable. Dans ce TP, nous cherchons à savoir sur combien de temps nous devons emprunter 10 000 € au taux de 7 %, en tenant compte d'une possibilité de remboursement maximum mensuel de 750 €.

Prérequis : néant
Fichier employé : TP_08_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le fichier TP_08_début.xlsx et enregistrez-le sous le nom TP_08.xlsx.
Cliquez sur l'onglet *Données*. Dans le groupe *Outils de données*, cliquez sur *Analyse de scénarios* puis sur *Valeur cible*.



2 Dans la boîte de dialogue *Valeur cible*, saisissez **C6** dans la zone *Cellule à définir*, **-750** dans la zone *Valeur à atteindre* (négatif, car c'est un remboursement) et **C4** dans la zone *Cellule à modifier*. Cliquez sur *OK*.



Copyright © 2012 Dunod.

3 Une boîte de dialogue présente (parfois très brièvement) les différentes itérations avant d'afficher une autre boîte de dialogue indiquant le succès ou l'échec de l'opération.



4 Le résultat trouvé est affiché sur la feuille de calcul.
Enregistrez et fermez le classeur TP_08.xlsx.

A	B	C	D
	CALCUL D'EMPRUNT		
	Montant total	10 000,00 €	
	Durée (mois)	13,92085592	
	Taux	7%	
	Mensualité	-750,00 €	

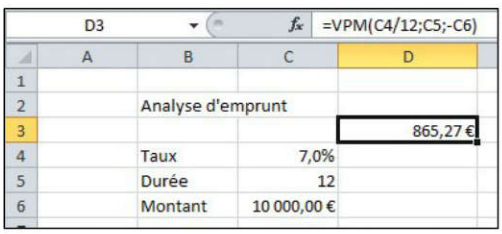
Copyright © 2012 Dunod.

TP 9 Table de données

Lorsqu'une formule a recours à un maximum de deux variables, ou si plusieurs formules utilisent toutes une même variable, une table de données permet d'afficher tous les résultats en un emplacement unique. Les différentes valeurs des variables sont présentées en lignes et colonnes, avec les différents résultats apparaissant ainsi dans le tableau automatiquement créé.

Prérequis : néant
Fichier employé : TP_09_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le fichier TP_09_début.xlsx et enregistrez-le sous le nom TP_09.xlsx.
Dans cet exemple, la cellule *D3* contient la formule **=VPM(C4/12,C5;-C6)**. Les valeurs variables sont celles des cellules *C4* et *C5*.



	A	B	C	D
1				
2		Analyse d'emprunt		
3				865,27 €
4		Taux	7,0%	
5		Durée	12	
6		Montant	10 000,00 €	

2 Saisissez dans la plage *D4:D6* les taux **0,065**, **0,07** et **0,75**.

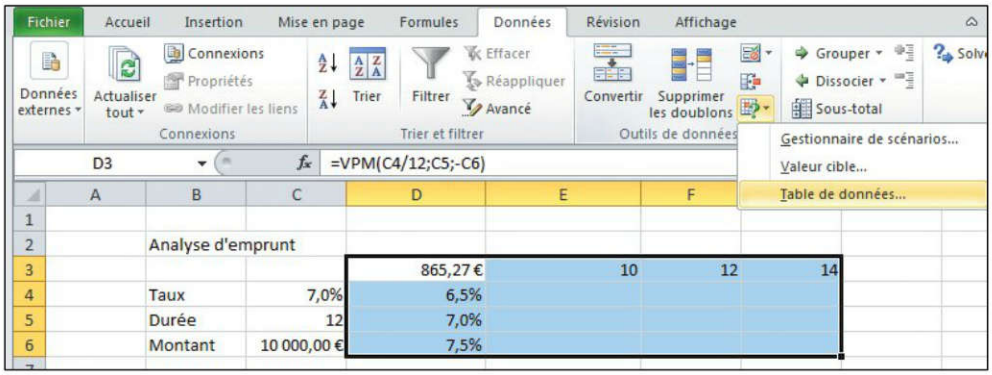
Analyse d'emprunt		
		865,27 €
Taux	7,0%	6,5%
Durée	12	7,0%
Montant	10 000,00 €	7,5%

3 Saisissez dans la plage *E4:G4* les différentes durées, soit **10**, **12** et **14**.

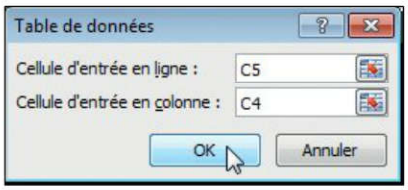
Analyse d'emprunt					
		865,27 €	10	12	14
Taux	7,0%	6,5%			
Durée	12	7,0%			
Montant	10 000,00 €	7,5%			

Copyright © 2012 Dunod.

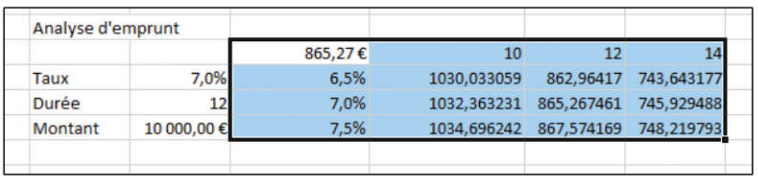
4 Sélectionnez la plage *D3:G6*. Dans le groupe *Outils de données* de l'onglet *Données*, cliquez sur *Analyse de scénarios*, puis sur *Table de données*.



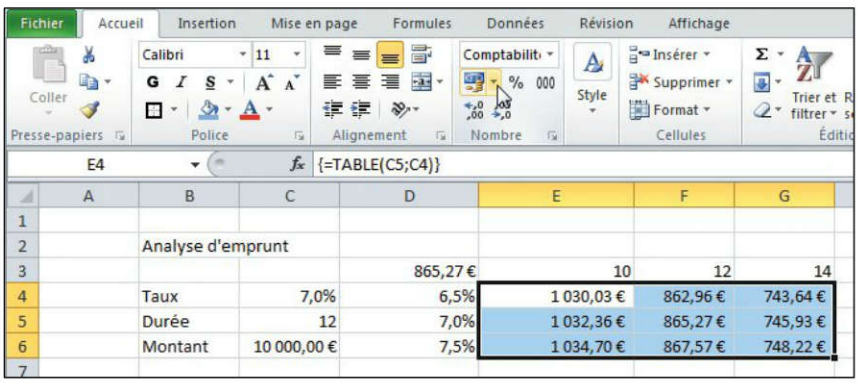
5 Dans la zone *Cellule d'entrée en lignes*, saisissez **C5**. Dans la zone *Cellule d'entrée en colonne*, saisissez **C4**. Cliquez sur *OK*.



6 La table de données affiche les résultats recherchés. Toutefois, le format de sortie n'est pas forcément le format adapté, comme c'est le cas ici.



7 Sélectionnez la plage *E4:G6*, puis, dans l'onglet *Accueil*, dans le groupe *Nombre*, cliquez sur *Format nombre comptabilité*.
Enregistrez puis fermez le classeur TP_09.xlsx.



Copyright © 2012 Dunod.

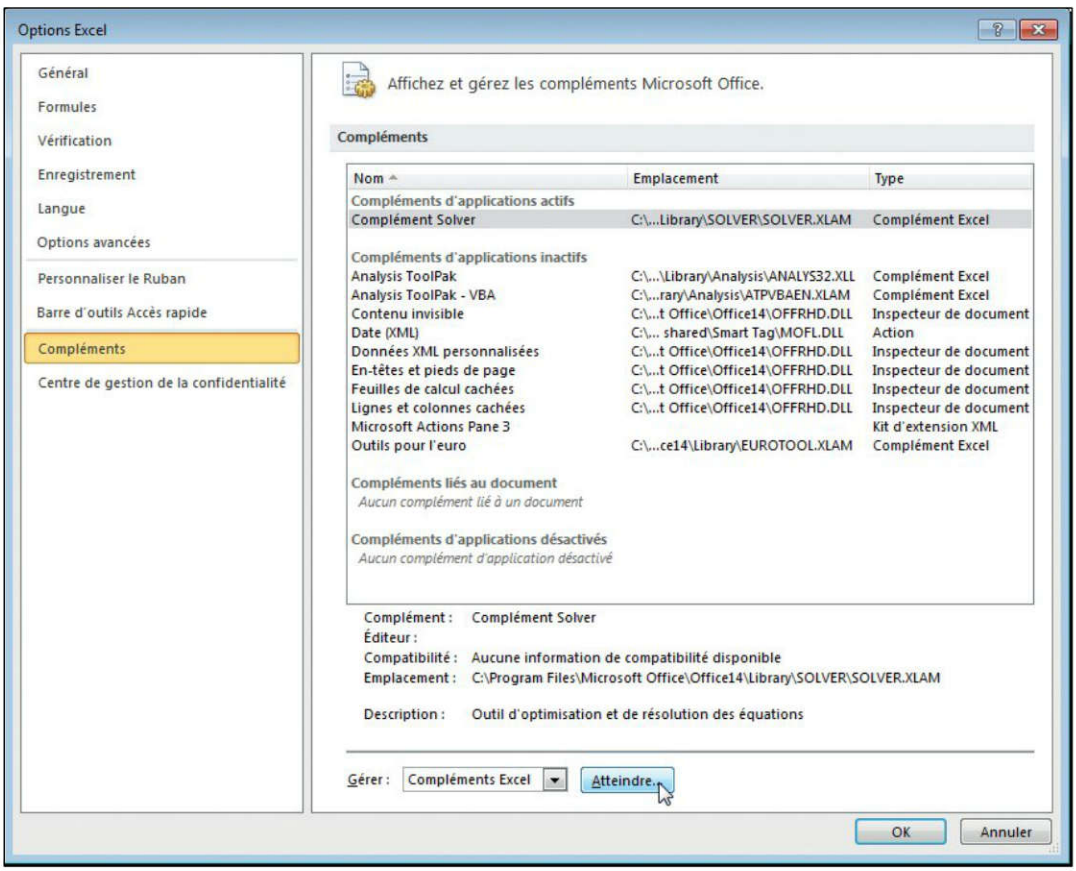
TP 10 Utilitaire d'analyse

L'utilitaire d'analyse, disponible dans le groupe *Analyse* de l'onglet *Données*, procure l'accès à de nombreuses fonctions statistiques parfois très complexes que vous pouvez appliquer à vos données à l'aide d'assistants. Certains outils génèrent des graphiques. Si vous êtes amenés à appliquer régulièrement des fonctions statistiques de haut niveau à vos données, cet utilitaire se révèlera d'une aide inestimable.

Dans ce TP, nous allons tester deux de ces outils, la moyenne mobile et les statistiques descriptives, appliqués à une consommation d'électricité suivie depuis plusieurs années.

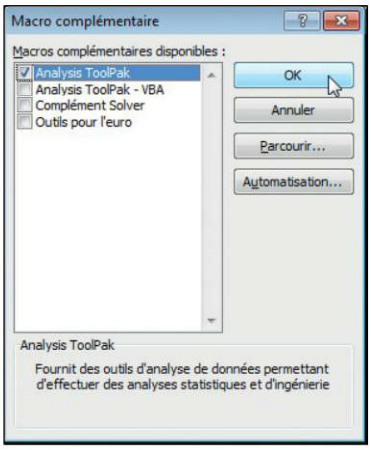
Prérequis : néant
Fichier employé : TP_10_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le classeur TP_10_début.xlsx et enregistrez-le sous le nom TP_10.xlsx.
Cliquez sur l'onglet *Fichier*, cliquez sur *Options*, puis sur la catégorie *Compléments*. Dans la zone *Gérer*, sélectionnez *Compléments Excel*, puis cliquez sur *Atteindre*.

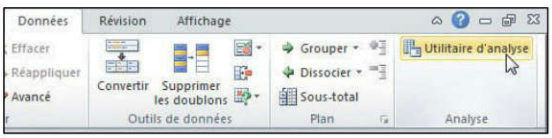


Copyright © 2012 Dunod.

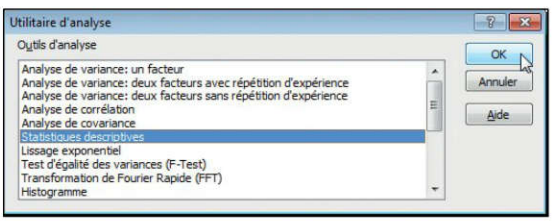
2 Dans la zone *Macros complémentaires disponibles*, cochez **Analysis ToolPack**. Si vous souhaitez employer ces fonctions dans du code VBA, cochez également *Analysis ToolPack -VBA*. Cliquez sur *OK*.



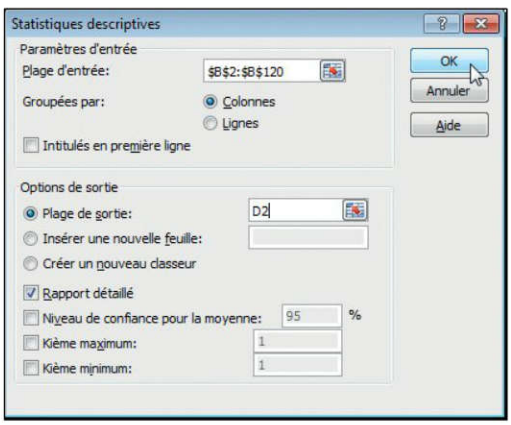
3 Cliquez sur l'onglet *Données*. Dans le groupe *Analyse*, cliquez sur *Utilitaire d'analyse*.



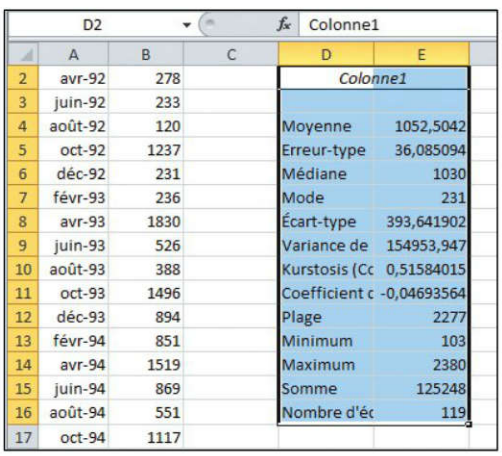
4 La boîte de dialogue *Utilitaire d'analyse* s'affiche. Choisissez *Statistiques descriptives*, puis cliquez sur *OK*.



5 Dans la boîte de dialogue *Statistiques descriptives*, sélectionnez dans la zone *Plage d'entrée* la plage **$B$2:$B$120**, définissez dans la zone *Plage de sortie* **D2**, et cochez l'option *Rapport détaillé*. Cliquez sur *OK*.



6 Les résultats s'affichent dans la feuille de calcul. Dans le groupe *Analyse* de l'onglet *Données*, cliquez à nouveau sur *Utilitaire d'analyse*. Sélectionnez cette fois dans la boîte de dialogue *Utilitaire d'analyse* l'outil *Moyenne mobile*, puis cliquez sur *OK*.

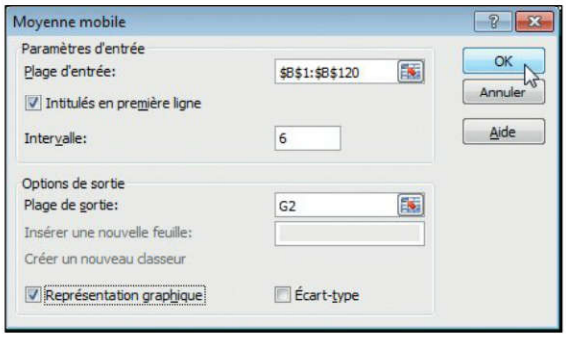


D2 | Colonne1

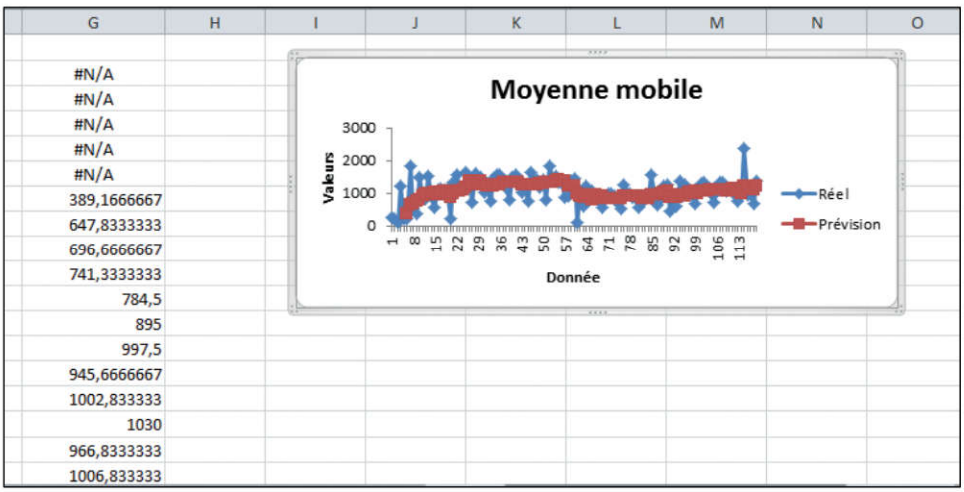
	A	B	C	D	E
2	avr-92	278		Colonne1	
3	juin-92	233			
4	août-92	120		Moyenne	1052,5042
5	oct-92	1237		Erreur-type	36,085094
6	déc-92	231		Médiane	1030
7	févr-93	236		Mode	231
8	avr-93	1830		Écart-type	393,641902
9	juin-93	526		Variance de	154953,947
10	août-93	388		Kurstosis (Cc	0,51584015
11	oct-93	1496		Coefficient c	-0,04693564
12	déc-93	894		Plage	2277
13	févr-94	851		Minimum	103
14	avr-94	1519		Maximum	2380
15	juin-94	869		Somme	125248
16	août-94	551		Nombre d'éc	119
17	oct-94	1117			

Copyright © 2012 Dunod.

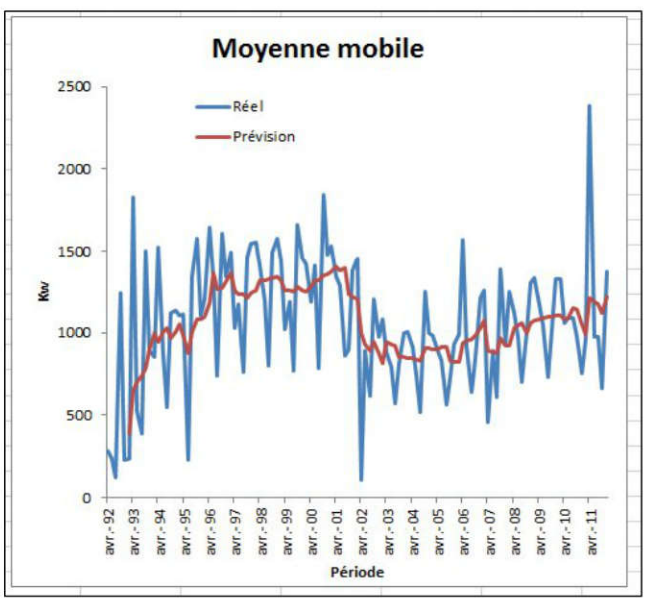
7 Dans la boîte de dialogue *Moyenne Mobile*, sélectionnez dans la zone *Plage d'entrée* la plage **$B$2:$B$120,** spécifiez un intervalle de **6**, définissez **G2** dans la zone *Plage de sortie*, et cochez l'option *Représentation graphique*. Cliquez sur *OK*.



8 Les résultats, ainsi qu'un graphique, s'affichent dans la feuille désignée. Le graphique n'est toutefois ici pas très lisible.



9 Modifiez le graphique à votre goût. Vous pourriez, par exemple, opter pour une présentation comme celle figurant dans cette copie d'écran.



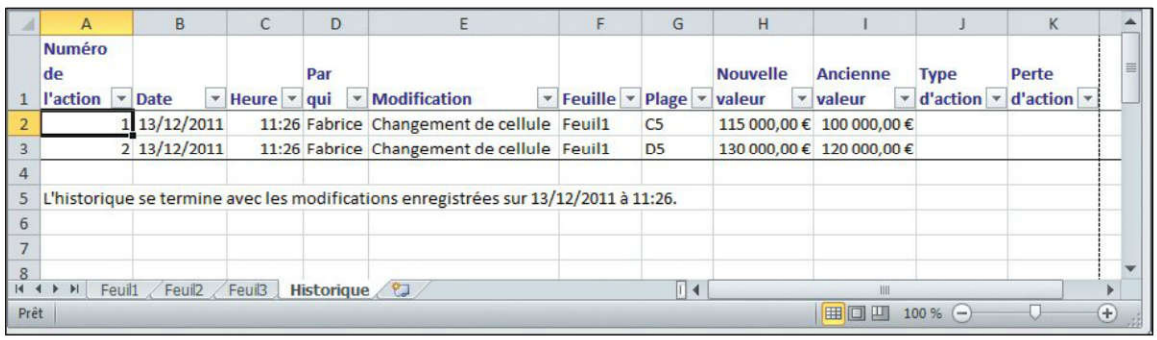
Enregistrez et fermez le classeur TP_10.xlsx.

Chacun des outils de l'Utilitaire d'analyse propose généralement des assistants. La place manque dans cet ouvrage pour décrire en détail ces puissants outils : reportez-vous à la documentation si nécessaire.

Copyright © 2012 Dunod.

Travail en collaboration

Même si une seule personne peut être chargée de gérer les données d'une entreprise, d'un organisme ou d'une association, plusieurs personnes sont concernées lorsqu'il s'agit d'établir des projections. Il est possible d'enrichir les données d'un classeur partagé en ajoutant des commentaires qui mettent en lumière des informations. Si le classeur qui héberge ces projections et commentaires est disponible sur un réseau ou dans un Intranet, plusieurs utilisateurs accèdent simultanément au classeur en activant le partage de classeur. Une fois le classeur partagé avec vos collaborateurs, annotez-le et enregistrez-y tous les changements apportés. L'administrateur du classeur opte alors pour les changements à accepter et à refuser.



	A	B	C	D	E	F	G	H	I	J	K
1	Numéro de l'action	Date	Heure	Par qui	Modification	Feuille	Plage	Nouvelle valeur	Ancienne valeur	Type d'action	Perte d'action
2	1	13/12/2011	11:26	Fabrice	Changement de cellule	Feuil1	C5	115 000,00 €	100 000,00 €		
3	2	13/12/2011	11:26	Fabrice	Changement de cellule	Feuil1	D5	130 000,00 €	120 000,00 €		
4											
5	L'historique se termine avec les modifications enregistrées sur 13/12/2011 à 11:26.										
6											
7											
8											

Feuil1 | Feuil2 | Feuil3 | Historique

Prêt — 100 %

Si vous préférez limiter le nombre d'utilisateurs susceptibles d'afficher et de modifier vos classeurs, ajoutez un mot de passe au classeur, à la feuille de calcul, à la plage de données ou même à une cellule individuelle. Vous évitez ainsi que des éléments cruciaux soient modifiés. Il est également possible de masquer des formules.



Si vous travaillez dans un environnement comprenant des collaborateurs itinérants avec lesquels vous échangez fréquemment des fichiers, utilisez une signature numérique pour vérifier que vos classeurs et les macros qu'ils contiennent proviennent d'une source de confiance.



Copyright © 2012 Dunod.

Partage de classeur et protection du partage (I)

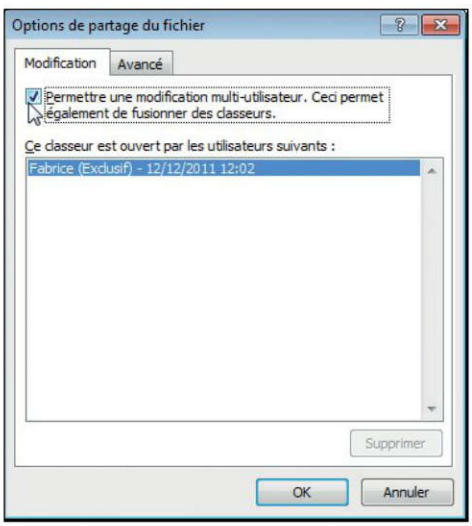
Pour autoriser plusieurs utilisateurs à modifier simultanément un classeur, vous devez activer le partage de classeur. Vous pouvez ensuite interdire la désactivation du suivi des modifications d'un classeur partagé en protégeant le partage : ainsi, l'historique des modifications apportées au classeur sera toujours visible. Il est impossible de partager un classeur contenant un tableau Excel : vous devrez auparavant convertir ce tableau en plage de cellules standard.

Prérequis : néant
Fichier employé : TP_11_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_11_début.xlsx, puis enregistrez-le sous le nom TP_11.xlsx.
Cliquez sur l'onglet *Révision*, puis, dans le groupe *Modifications*, cliquez sur *Partager le classeur*.

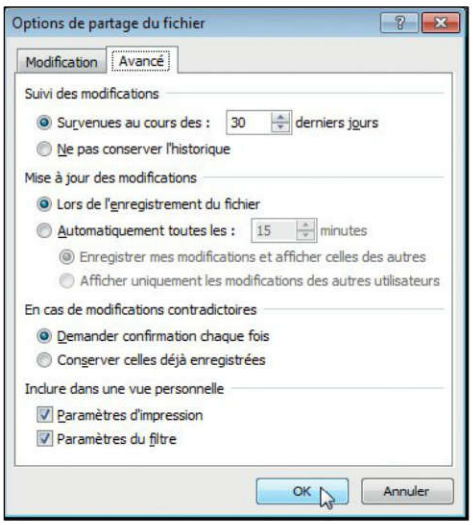


2 Dans la boîte de dialogue *Options de partage du fichier*, cochez la case *Permettre une modification multi-utilisateur*.
Cliquez sur l'onglet *Avancé* de la boîte de dialogue *Options de partage du fichier*.

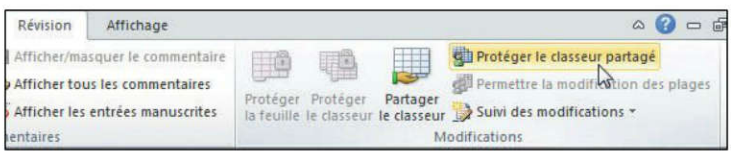


Copyright © 2012 Dunod.

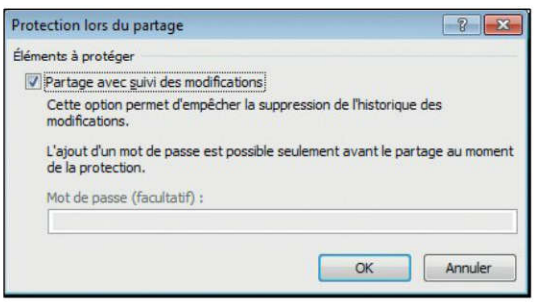
3 Cet onglet renferme plusieurs paramètres importants. Le *Suivi des modifications* définit si Excel conserve l'historique des changements apportés au classeur et la durée de cet enregistrement. La valeur par défaut est de 30 jours. La *Mise à jour des modifications* définit le moment de mise à jour du suivi et la possibilité de suivre ou non vos propres modifications. Le paramètre *En cas de modifications contradictoires* concerne l'attitude à adopter en pareil cas. Si vous êtes confrontés à ce risque, cochez la case *Demander confirmation chaque fois pour choisir la valeur à conserver*. Cliquez sur *OK*.



4 Une boîte de message s'affiche. Cliquez sur *OK*. Excel enregistre et partage le classeur. Cliquez sur l'onglet *Révision*, puis, dans le groupe *Modifications*, cliquez *sur Protéger le classeur partagé*.



5 Dans la boîte de dialogue *Protection lors du partage*, cochez l'option *Partage avec suivi des modifications*. Cela rend impossible la suppression du suivi des modifications par l'utilisateur.
Enregistrez et fermez le classeur TP_11.



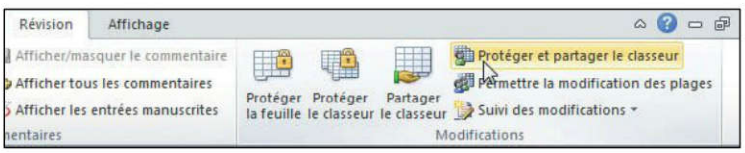
Copyright © 2012 Dunod.

Partage de classeur et protection du partage (II)

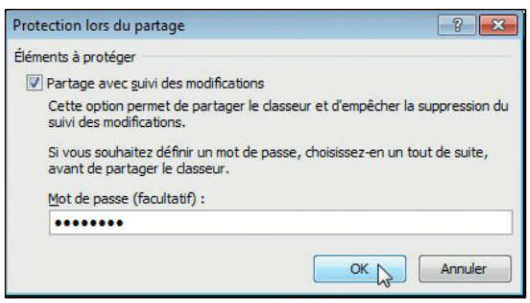
La méthode précédente présente l'inconvénient de laisser à un utilisateur la possibilité de supprimer facilement cette protection, puis de désactiver le suivi des modifications. Pour l'interdire totalement, vous devez accomplir les deux démarches précédentes en une seule opération, ce qui permet en outre de définir un mot de passe pour outrepasser éventuellement la protection.

Prérequis : TP 11
Fichier employé : TP_12_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_12_début et enregistrez-le sous le nom TP_12.xlsx.
Dans le groupe *Modifications* de l'onglet *Révision*, cliquez sur *Protéger et partager le classeur*.

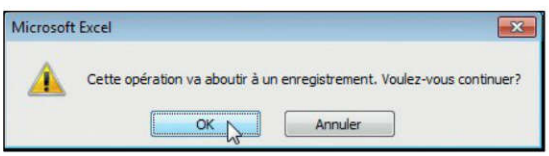


2 Dans la boîte de dialogue *Protection lors du partage*, cochez l'option *Partage avec suivi des modifications*. Dans la zone *Mot de passe*, saisissez **motpasse** (ou le mot de passe de votre choix) et cliquez sur *OK*. Dans la boîte de dialogue *Confirmer le mot de passe*, saisissez le mot de passe que vous avez choisi et cliquez sur *OK*.





3 Une boîte de message s'affiche. Cliquez sur *OK*. Excel enregistre et partage le classeur TP_12.xlsx.
Fermez le classeur TP_12.xlsx.



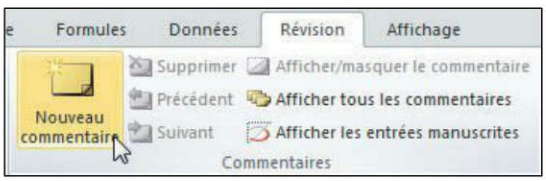
Copyright © 2012 Dunod.

TP 13 Gestion des commentaires

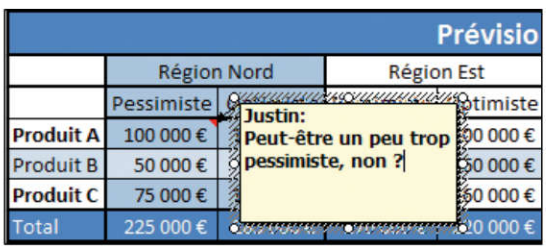
L'insertion d'un commentaire dans les cellules d'un classeur peut améliorer la compréhension des données. Une cellule comportant un commentaire affiche un indicateur dans l'angle supérieur droit de la cellule. Lorsque vous survolez avec la souris une telle cellule, le commentaire s'affiche à proximité de celle-ci, avec le nom de l'utilisateur connecté à l'ordinateur au moment de la création du commentaire.

Prérequis : néant.
Fichier employé : TP_13_début.xlsx.
Temps de réalisation : 5 minutes

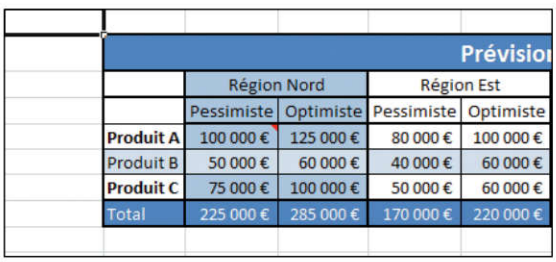
1 Ouvrez le classeur TP 13_début.xlsx, puis enregistrez-le sous le nom TP_13.xlsx.
Cliquez sur la cellule *C5*. Dans l'onglet *Révision*, groupe *Commentaires*, cliquez sur *Nouveau commentaire*.



2 Un indicateur de commentaire rouge s'affiche dans la cellule *C5* et une zone de commentaire apparaît à proximité de la cellule. Dans la zone de commentaire, tapez **Peut-être un peu trop pessimiste, non ?**

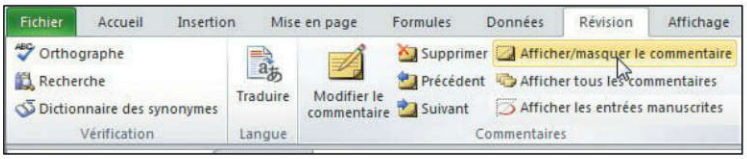


3 Cliquez sur une cellule en dehors de la zone de commentaire. Cette dernière disparaît. Seul subsiste l'indicateur rouge.

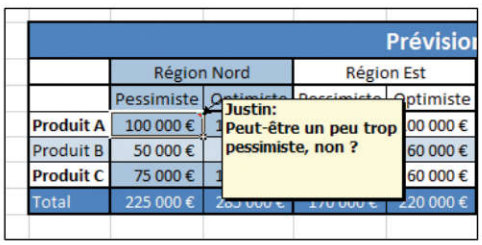


Copyright © 2012 Dunod.

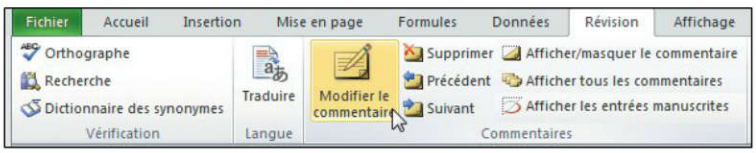
4 Cliquez sur la cellule *C5*, puis sur l'onglet *Révision* et enfin, dans le groupe *Commentaires*, cliquez sur *Afficher/masquer le commentaire*.



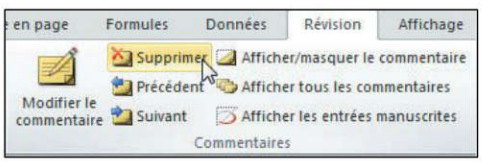
5 Le commentaire apparaît de façon permanente. Cliquez à nouveau sur le bouton *Afficher/masquer le commentaire* pour le faire disparaître.
Si le classeur comportait plusieurs commentaires, vous les afficheriez tous en cliquant sur *Afficher tous les commentaires*.



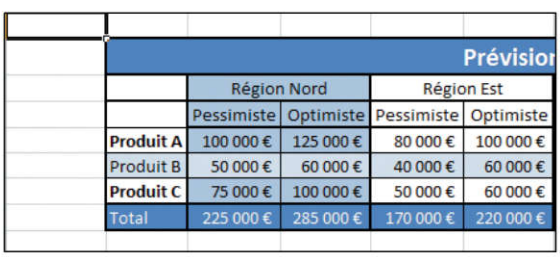
6 Cliquez sur la cellule *C5*. Dans l'onglet *Révision*, dans le groupe *Commentaires*, cliquer sur le bouton *Modifier le commentaire* : vous pouvez modifier le commentaire.



7 Cliquez sur la cellule *A1*, puis sur la cellule *C5*. Cliquez sur *Supprimer* dans le groupe *Commentaires* de l'onglet *Révision*.



8 Excel supprime le commentaire de la cellule. L'indicateur rouge a disparu.
Fermez le classeur TP_13.xlsx.



Copyright © 2012 Dunod.

TP 14 Suivi et gestion des modifications des utilisateurs (I)

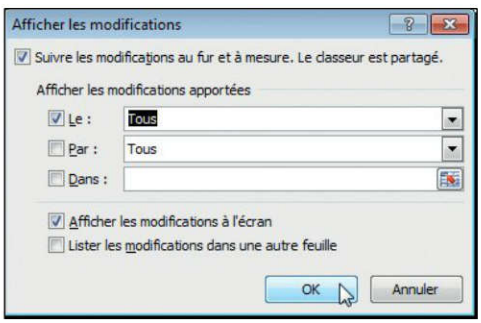
Dès que vous collaborez avec plusieurs utilisateurs pour générer ou modifier un document, il est pertinent de suivre les modifications apportées. Une fois le suivi des modifications activé, tout changement apporté au classeur est surligné dans la couleur attribuée à l'utilisateur concerné. En cas de doute sur un changement, vous identifiez rapidement son auteur et pourrez juger de la pertinence de la modification apportée.

Prérequis : néant
Fichier employé : TP_14_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_14_début et enregistrez-le sous le nom TP_14.xlsx.
Dans le groupe *Modifications* de l'onglet *Révision*, cliquez sur *Suivi des modifications*, puis sur *Afficher les modifications*.



2 Dans la boîte de dialogue *Afficher les modifications* qui s'ouvre, sélectionnez *Suivre les modifications au fur et à mesure*, puis vérifiez que l'option *Le* est fixée à *Tous*. Cliquez sur *OK*.



3 Une boîte de dialogue signale qu'Excel va enregistrer le classeur. Cliquez sur *OK*. Dans la cellule *C5*, tapez **115000** et appuyez sur *Entrée*.

	A	B	C	D
1				
2				
3			Région Nord	
4			Pessimiste	Optimiste
5		**Produit A**	115 000 €	120 000 €
6		Produit B	50 000 €	60 000 €
7		**Produit C**	75 000 €	100 000 €
8		Total	240 000 €	280 000 €
9				

4 Un indicateur bleu apparaît dans l'angle supérieur gauche de la cellule *C5*. Dans la cellule *D5*, tapez **130000** et appuyez sur *Entrée*.
Enregistrez et fermez le classeur TP_14.xlsx si vous ne poursuivez pas immédiatement avec l'exercice suivant.

	A	B	C	D	E	F
1						
2						Prévisio
3			Région Nord		Région Est	
4			Pessimiste	Optimiste	Pessimiste	Optimiste
5		**Produit A**	110 000 €	130 000 €	80 000 €	100 000 €
6		Produit B	50 000 €	60 000 €	40 000 €	60 000 €
7		**Produit C**	75 000 €	100 000 €	50 000 €	60 000 €
8		Total	235 000 €	290 000 €	170 000 €	220 000 €
9						

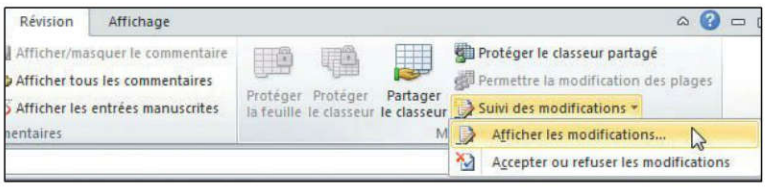
Copyright © 2012 Dunod.

TP 15 Suivi et gestion des modifications des utilisateurs (II)

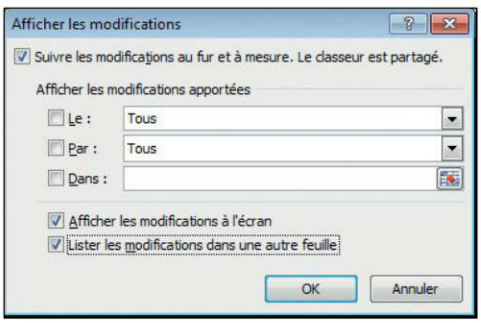
Il existe un moyen d'obtenir rapidement une synthèse de toutes les modifications apportées.

Prérequis : TP 14
Fichier employé : TP_15_début.xlsx
Temps de réalisation : 5 minutes

1 Si le classeur TP_14.xlsx n'est pas ouvert, ouvrez le classeur TP_15_début.xlsx. Enregistrez le classeur actif sous le nom TP_15.xlsx.
Dans l'onglet *Révision*, groupe *Modifications*, cliquez sur *Suivi des modifications* puis sur *Afficher les modifications*.



2 Dans la boîte de dialogue *Afficher les modifications*, cochez la case *Lister les modifications dans une autre feuille*, décochez l'option *Le* et cliquez sur *OK*.



3 Excel crée et affiche une feuille intitulée Historique qui contient la liste de toutes les modifications effectuées depuis la dernière fois qu'un utilisateur a accepté ou refusé des changements.

	A	B	C	D	E	F	G	H	I	J	K
1	Numéro de l'action	Date	Heure	Par qui	Modification	Feuille	Plage	Nouvelle valeur	Ancienne valeur	Type d'action	Perte d'action
2	1	13/12/2011	11:26	Fabrice	Changement de cellule	Feuil1	C5	115 000,00 €	100 000,00 €		
3	2	13/12/2011	11:26	Fabrice	Changement de cellule	Feuil1	D5	130 000,00 €	120 000,00 €		
4											
5	L'historique se termine avec les modifications enregistrées sur 13/12/2011 à 11:26.										
6											
7											
8											

Feuil1 Feuil2 Feuil3 Historique
Prêt 100 %

4 Cliquez sur le bouton *Enregistrer* de la barre d'outils rapide. Excel supprime la feuille Historique et enregistre le classeur.
Laissez le classeur ouvert si vous poursuivez immédiatement avec le TP suivant. Sinon, fermez-le.



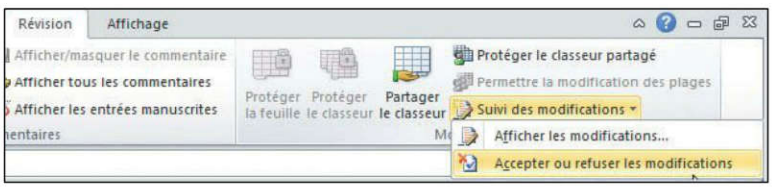
Copyright © 2012 Dunod.

TP 16 Acceptation ou refus des modifications

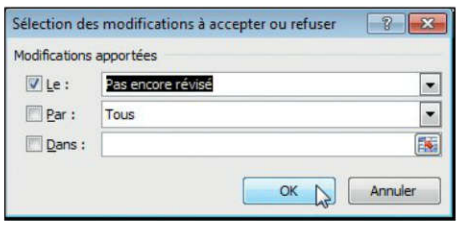
Après avoir activé le suivi des modifications, vous pouvez les accepter ou les refuser.

Prérequis : TP 14 et 15
Fichier employé : TP_16_début.xlsx
Temps de réalisation : 5 minutes

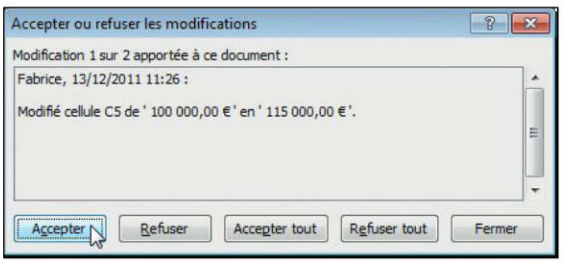
1 Si le classeur TP_15.xlsx n'est pas ouvert, ouvrez le classeur TP_16_début.xlsx. Enregistrez le classeur actif sous le nom TP_16.xlsx.
Dans l'onglet *Révision*, groupe *Modifications*, cliquez sur *Suivi des modifications* puis sur *Accepter ou refuser les modifications*.



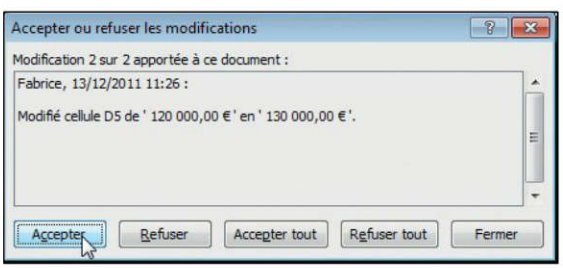
2 La boîte de dialogue *Sélection des modifications à accepter ou refuser* s'affiche. Cliquez sur *OK*.



3 La boîte de dialogue *Accepter ou refuser les modifications* présente le premier changement. Cliquez sur *Accepter*. Excel accepte la modification et affiche la suivante.



4 Cliquez sur *Accepter*. Excel accepte la modification. La boîte de dialogue *Accepter ou refuser les modifications* disparaît. Si elle était affichée, la feuille Historique serait supprimée.
Enregistrez et fermez le classeur TP_16.xlsx.



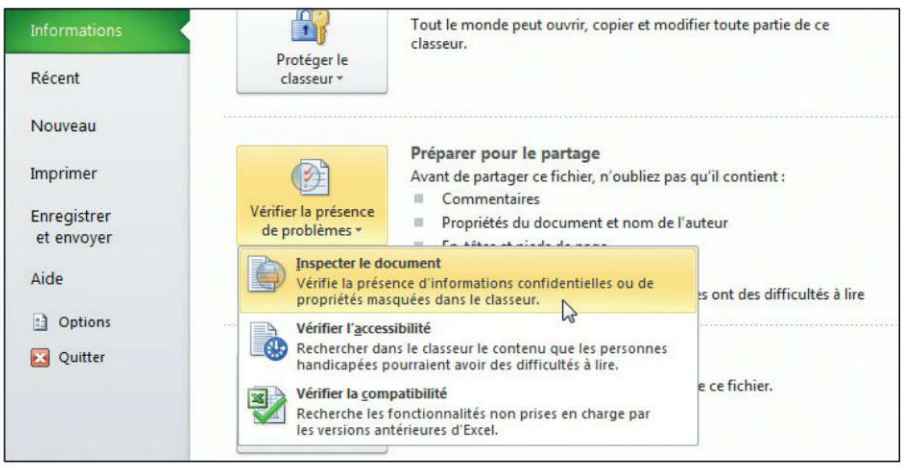
Copyright © 2012 Dunod.

TP 17 Inspection d'un classeur

Distribuer un classeur à d'autres personnes n'est pas sans danger : le classeur pourrait renfermer des informations privées que vous ne souhaitez pas partager. Grâce à l'Inspecteur de document, vous pouvez rapidement repérer commentaires et notes, propriétés du document et informations personnelles, données XML personnalisées, en-têtes et pieds de page, colonnes et lignes masquées, feuilles de calcul masquées et contenu invisible. Vous pouvez facilement supprimer toute information identifiée par l'Inspecteur de document.

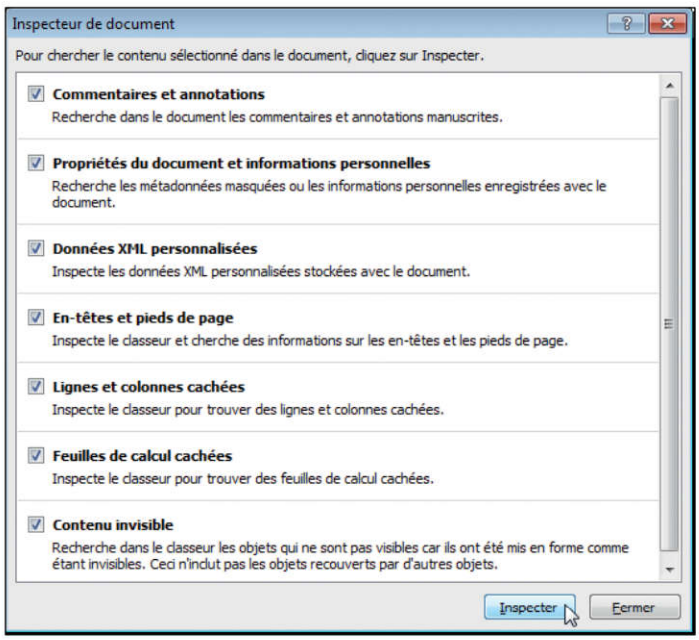
Prérequis : néant
Fichier employé : TP_17_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_17_début.xlsx et enregistrez-le sous le nom TP_17.xlsx.
Cliquez sur l'onglet *Fichier*. Sur la page *Informations,* cliquez sur *Vérifier la présence de problèmes*, puis sur *Inspecter le document*.



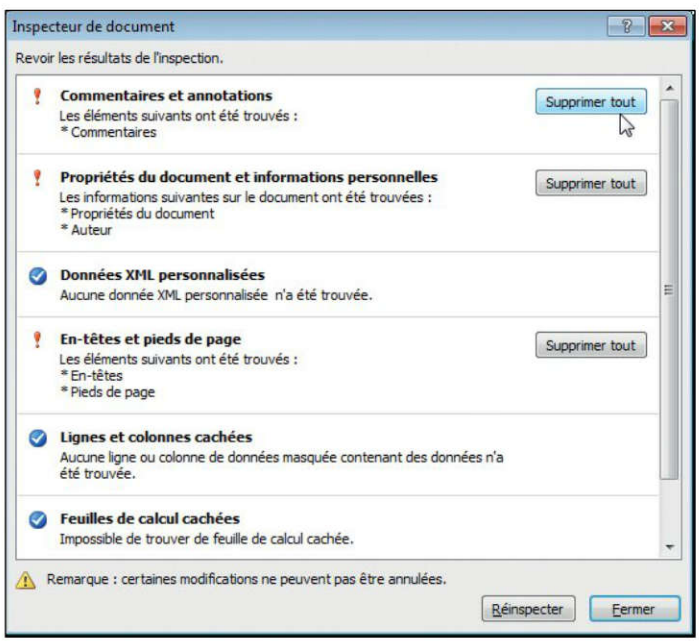
Copyright © 2012 Dunod.

2 Dans la fenêtre de l'Inspecteur de document, vous décochez tous les types de contenu qui doivent rester dans le document. Laissez-les ici tous cochés, puis cliquez sur *Inspecter*.



3 Dans la liste des résultats de l'inspection, cliquez successivement sur le bouton *Supprimer tout* situé à droite des catégories de données que vous voulez supprimer. Une fois les données indésirables supprimées, cliquez sur *Fermer*.

Enregistrez et fermez le classeur TP_17.xlsx.



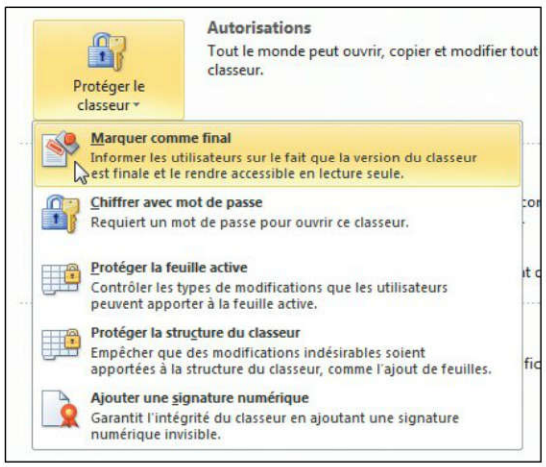
Copyright © 2012 Dunod.

TP 18 Finalisation d'un classeur

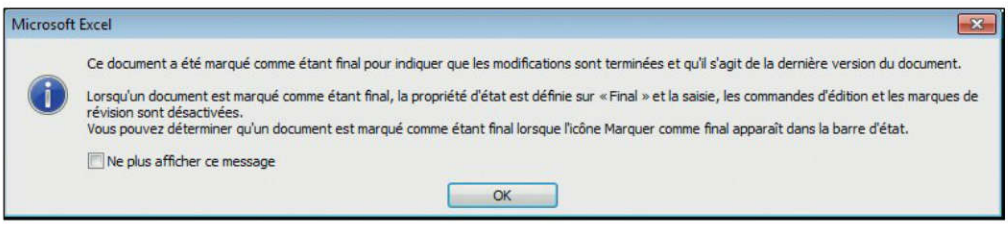
Marquer un classeur comme final fixe sa propriété d'état à Final et empêche toute modification ultérieure effectuée par inadvertance.

Prérequis : néant
Fichier employé : TP_18_début.xlsx.
Temps de réalisation : 5 minutes

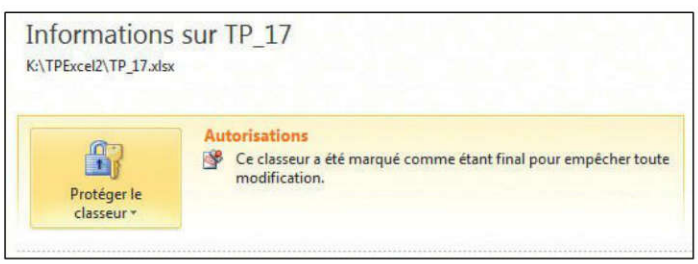
1 Ouvrez le classeur TP_18_début.xlsx et enregistrez le sous le nom TP_18.xlsx.
Cliquez sur l'onglet *Fichier*, puis, sur la page *Informations,* cliquez sur *Protéger le classeur*, et enfin sur *Marquer comme final*.



2 Dans la boîte de message indiquant que le classeur va être marqué comme final puis enregistré, cliquez sur *OK*.
Une nouvelle boîte de message s'affiche pour confirmer votre choix. Si tel est le cas, cliquez sur *OK*.



3 Cliquez sur l'onglet *Fichier*, puis sur la page *Informations* : vous voyez que le classeur est marqué comme final.
Pour restaurer les fonctionnalités d'un classeur marqué comme tel, cliquez sur l'onglet *Fichier* puis, dans la page *Informations*, cliquez sur *Protéger le classeur* puis à nouveau sur *Marquer comme final* pour modifier son état.
Fermez le classeur TP_18.xlsx.



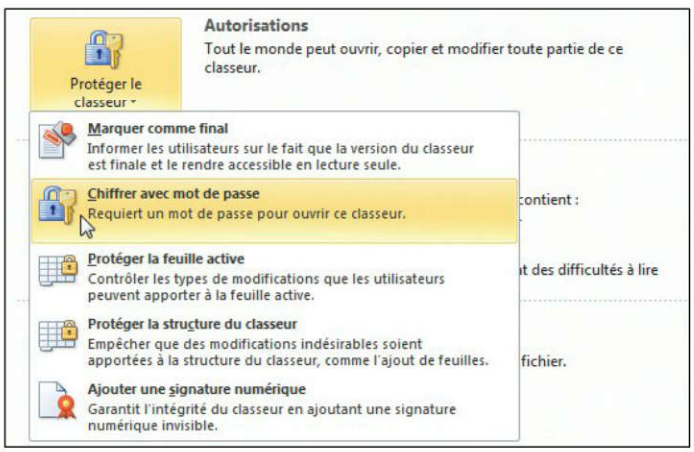
Copyright © 2012 Dunod.

TP 19 Protection d'un classeur

Vous limitez l'accès aux classeurs ou à ses éléments à l'aide d'un mot de passe. Ainsi, un utilisateur qui souhaite ouvrir un classeur protégé devra saisir un mot de passe dans une boîte de dialogue avant de pouvoir le déverrouiller. En l'absence de ce mot de passe, l'accès est refusé.

Prérequis : néant
Fichier employé : TP_19_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_19_début.xlsx, puis enregistrez-le sous le nom TP_19.xlsx.
Cliquez sur l'onglet *Fichier* et, si nécessaire, choisissez *Informations*. Cliquez sur le bouton *Protéger le classeur* puis sur *Chiffrer avec mot de passe*.



2 La boîte de dialogue *Chiffrer un document* s'affiche. Saisissez un mot de passe (comme **motpasse**, ceci n'est qu'un exemple) dans la zone *Mot de passe pour la lecture*, puis cliquez sur *OK*. Dans la zone *Mot de passe* de la boîte de dialogue *Confirmer le mot de passe*, saisissez à nouveau le mot de passe, puis cliquez sur *OK*.



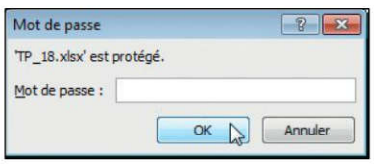


Copyright © 2012 Dunod.

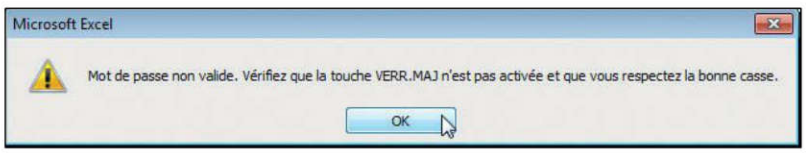
3 Cliquez sur *Fichier*, puis sur *Informations* : il est indiqué que le classeur est protégé.



4 Cliquez sur *Accueil*, fermez le classeur : une boîte de dialogue vous demande de confirmer les modifications. Cliquez sur *OK*, puis ouvrez à nouveau le classeur : une boîte de dialogue vous demande de saisir le mot de passe. Ne saisissez rien et cliquez sur *OK*.



5 Une boîte de dialogue vous signale que le mot de passe est invalide. Sans le mot de passe, vous ne pouvez pas ouvrir le classeur.



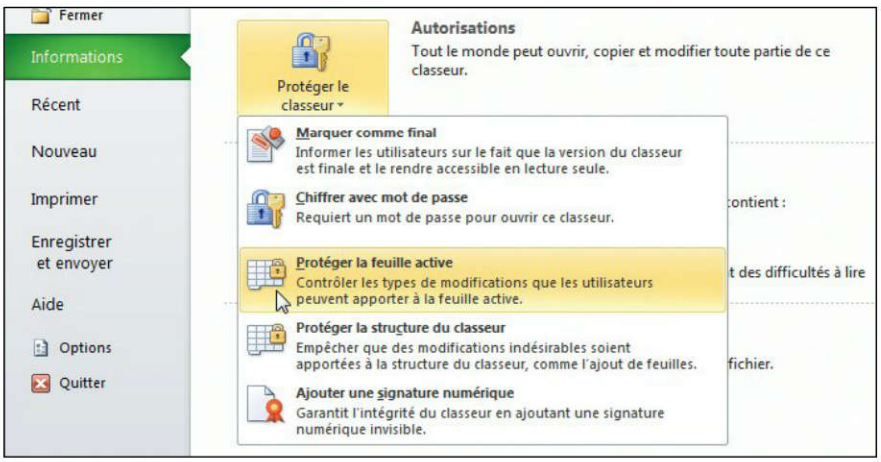
Copyright © 2012 Dunod.

TP 20 Protection de l'ensemble d'une feuille de calcul

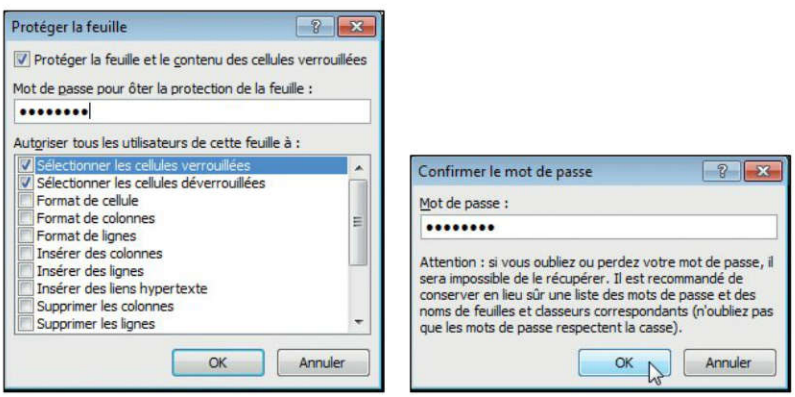
Que l'ouverture d'un classeur soit protégée ou non, vous pouvez protéger toute une feuille de calcul contre toute modification.

Prérequis : TP 19
Fichier employé : TP_20_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_20_début.xlsx, puis enregistrez-le sous le nom TP_20.xlsx.
Cliquez sur l'onglet *Fichier* et, si nécessaire, choisissez *Informations*. Cliquez sur le bouton *Protéger le classeur* puis *Protéger la feuille active*.



2 Dans la boîte de dialogue *Protéger la feuille*, laissez cochées les options et saisissez un mot de passe (**motpasse**, par exemple), puis cliquez sur *OK*. Dans la zone *Mot de passe*, de la boîte de dialogue *Confirmer le mot de passe*, saisissez à nouveau le mot de passe, puis cliquez sur *OK*.



Copyright © 2012 Dunod.

3 Cliquez sur *Fichier*, puis sur *Informations* : il est indiqué que la feuille de calcul est protégée.



4 Cliquez sur *Accueil*, sélectionnez la cellule *D6*, saisissez **140000** puis appuyez sur *Entrée* : une boîte de dialogue signale que la feuille est protégée et que toute modification est impossible. Cliquez sur *OK*.
Enregistrez et fermez le classeur TP_19.xlsx.



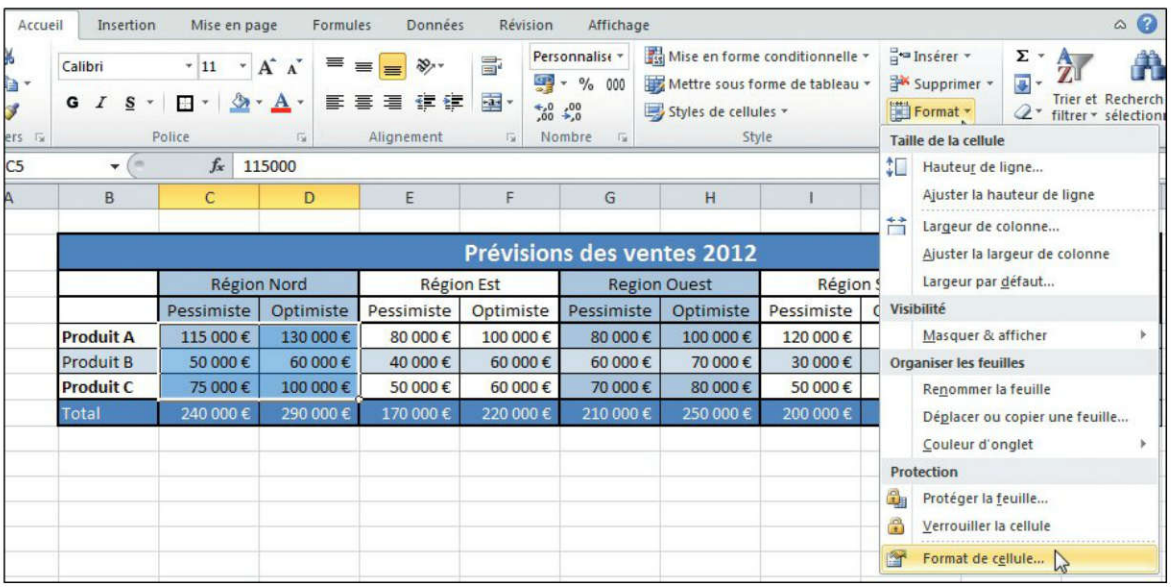
Copyright © 2012 Dunod.

TP 21 Protection d'une partie d'une feuille de calcul

Pour protéger seulement une partie d'une feuille de calcul, vous protégez toute la feuille après avoir enlevé les protections sur les zones modifiables.

Prérequis : TP 19 et 20
Fichier employé : TP_21_début.xlsx
Temps de réalisation : 5 minutes

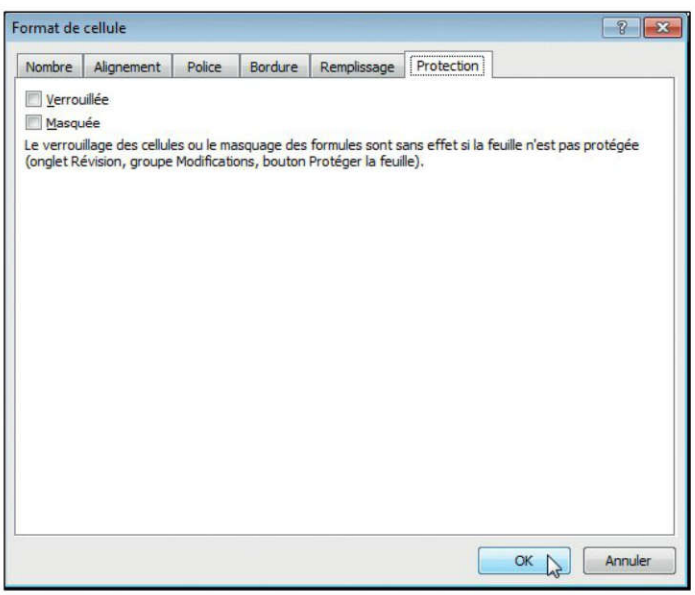
1 Ouvrez le classeur TP_21_début.xlsx et enregistrez-le sous le nom TP_21.xlsx.
Sélectionnez la zone *C5:D7*, puis, dans le groupe *Format* de l'onglet *Accueil,* cliquez sur *Format de cellule*.



Prévisions des ventes 2012

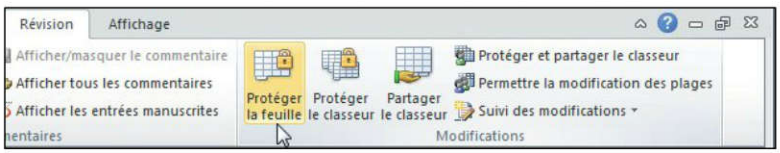
	Région Nord		Région Est		Region Ouest		Région
	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste
Produit A	115 000 €	130 000 €	80 000 €	100 000 €	80 000 €	100 000 €	120 000 €
Produit B	50 000 €	60 000 €	40 000 €	60 000 €	60 000 €	70 000 €	30 000 €
Produit C	75 000 €	100 000 €	50 000 €	60 000 €	70 000 €	80 000 €	50 000 €
Total	240 000 €	290 000 €	170 000 €	220 000 €	210 000 €	250 000 €	200 000 €

2 Dans la boîte de dialogue *Format de cellule*, cliquez sur l'onglet *Protection*. Dans celui-ci, décochez l'option *Verrouillée*, puis cliquez sur *OK*.

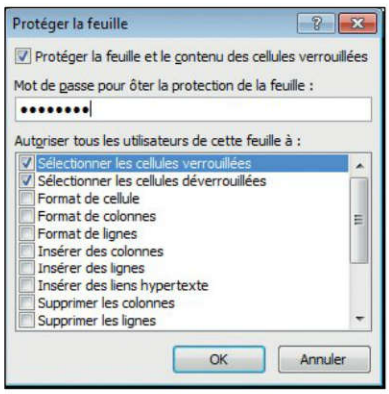


Copyright © 2012 Dunod.

3 Dans le groupe *Modifications* de l'onglet *Révision*, cliquez sur *Protéger la feuille*.

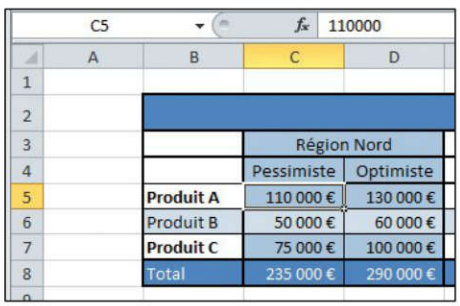


4 Dans la boîte de dialogue *Protéger la feuille*, saisissez **motpasse** (c'est un exemplet) et cliquez sur *OK*. Dans la boîte de dialogue *Confirmer le mot de passe*, saisissez **motpasse** (ou le mot de passe choisi) et cliquez sur *OK*.





5 Sélectionnez la cellule *C5* et saisissez **110000** : la valeur de la cellule est modifiée.



6 Cliquez sur la cellule *E5*. Essayez de saisir une nouvelle valeur : la boîte de dialogue signalant que la cellule est protégée s'affiche. Seules les cellules de la plage initialement sélectionnée peuvent être modifiées. Enregistrez et fermez le classeur TP_21.xlsx.



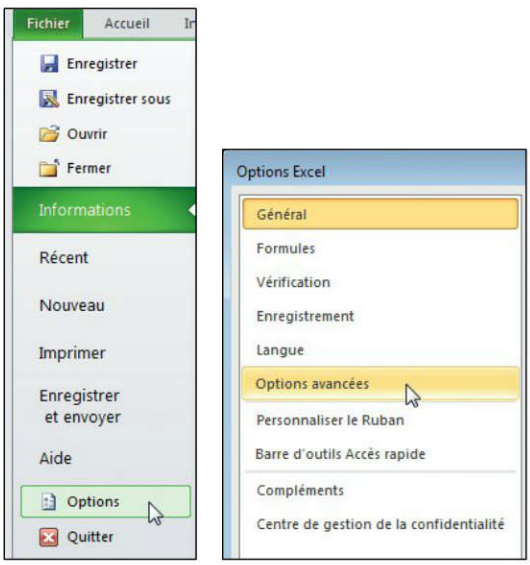
Copyright © 2012 Dunod.

TP 22 Afficher les relations entre formules et cellules

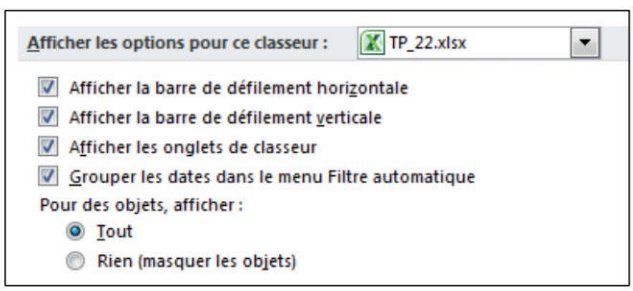
Il est parfois difficile de vérifier l'exactitude d'une formule ou de détecter la source d'une erreur. Excel permet d'examiner pour chaque cellule les cellules intervenant dans son contenu (les antécédents) ou celles affectées par le contenu de la cellule active (les dépendants).

Prérequis : néant
Fichier employé : TP_22_début.xlsx
Temps de réalisation : 10 minutes

1 Ouvrez le classeur TP_22_début.xlsx et enregistrez-le sous le nom TP_22.xlsx.
Cliquez sur l'onglet *Fichier* puis sur *Options*. Dans la boîte de dialogue *Options Excel*, cliquez sur *Options avancées*.

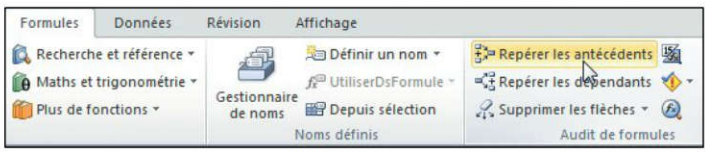


2 Dans la section *Afficher les options pour ce classeur*, sélectionnez le classeur voulu, puis vérifiez que l'option *Pour des objets, afficher* est fixée à *Tous*.



Copyright © 2012 Dunod.

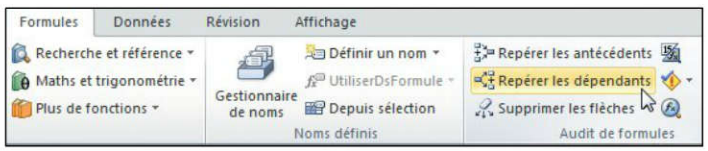
3 Sélectionnez la cellule *K5*. Dans l'onglet *Formules*, dans le groupe *Audit de formules*, cliquez sur *Repérer les antécédents*.



4 Excel affiche des flèches d'audit pour les antécédents, soit chaque cellule fournissant des données à la cellule active. Les flèches sont bleues pour une cellule exempte d'erreurs, rouges si celle-ci génère une erreur.

Prévisions des ventes 2012										
	Région Nord		Région Est		Region Ouest		Région Sud		Total général	
	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste
Produit A	115 000 €	130 000 €	80 000 €	100 000 €	80 000 €	100 000 €	120 000 €	130 000 €	395 000 €	460 000 €
Produit B	50 000 €	60 000 €	40 000 €	60 000 €	60 000 €	70 000 €	30 000 €	40 000 €	180 000 €	230 000 €
Produit C	75 000 €	100 000 €	50 000 €	60 000 €	70 000 €	80 000 €	50 000 €	75 000 €	245 000 €	315 000 €
Total	240 000 €	290 000 €	170 000 €	220 000 €	210 000 €	250 000 €	200 000 €	245 000 €	820 000 €	1 005 000 €

5 Sélectionnez la cellule *I7*. Dans l'onglet *Formules*, dans le groupe *Audit de formules*, cliquez sur *Repérer les dépendants*.



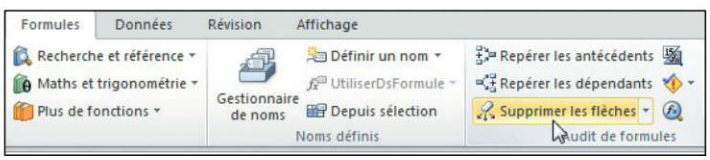
6 Excel affiche des flèches d'audit pour les dépendants, soit chaque cellule référençant la cellule active. Les flèches sont bleues pour une cellule exempte d'erreurs, rouges si celle-ci génère une erreur.

Prévisions des ventes 2012										
	Région Nord		Région Est		Region Ouest		Région Sud		Total général	
	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste
Produit A	115 000 €	130 000 €	80 000 €	100 000 €	80 000 €	100 000 €	120 000 €	130 000 €	395 000 €	460 000 €
Produit B	50 000 €	60 000 €	40 000 €	60 000 €	60 000 €	70 000 €	30 000 €	40 000 €	180 000 €	230 000 €
Produit C	75 000 €	100 000 €	50 000 €	60 000 €	70 000 €	80 000 €	50 000 €	75 000 €	245 000 €	315 000 €
Total	240 000 €	290 000 €	170 000 €	220 000 €	210 000 €	250 000 €	200 000 €	245 000 €	820 000 €	1 005 000 €

7 Dans l'onglet *Formules*, dans le groupe *Audit de formules*, cliquez sur *Supprimer les flèches*. Toutes les flèches d'audit disparaissent.

C'est également le cas si vous modifiez la formule vers laquelle elles pointent, si vous insérez ou supprimez des colonnes ou des lignes, ou si vous supprimez ou encore déplacez des cellules.

Enregistrez et fermez le classeur TP_22.xlsx.



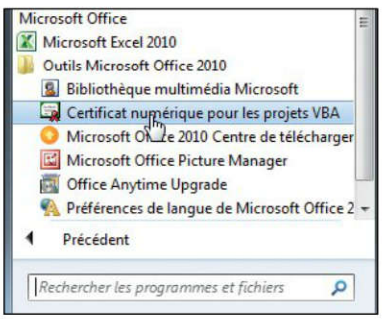
Copyright © 2012 Dunod.

TP 23 Authentifier un classeur

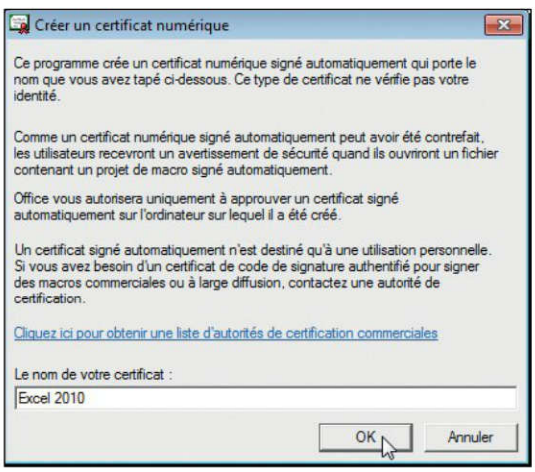
Lorsque vous échangez des fichiers sur des réseaux, et en particulier Internet, il est important de s'assurer de l'origine des fichiers. Cela est possible en authentifiant chaque classeur à l'aide d'une signature numérique. Celle-ci se définit comme une chaîne de caractères qui combine mathématiquement un fichier de signature numérique secrète et unique au contenu d'un classeur, qui permet de vérifier l'identité de l'utilisateur qui a signé le fichier. Les détails techniques et la procédure de gestion des certificats numériques dépassent la portée de cet ouvrage. Pour ce TP, nous utilisons l'accessoire du système Microsoft Office selfcert.exe pour générer un certificat. Ce dernier convient pour cette démonstration, mais les autres utilisateurs ne le considéreront pas comme valide.

Prérequis : néant
Fichier employé : TP_23_début.xlsx
Temps de réalisation : 5 minutes

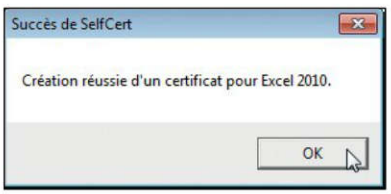
1 Ouvrez le classeur TP_23_début.xlsx et enregistrez-le sous le nom TP_23.xlsx.
Cliquez sur le bouton *Démarrer*, puis sur *Tous les programmes*, et enfin, sur *Microsoft Office*, *Outils Microsoft Office 2010* et sélectionnez *Certificat numérique pour les projets VBA*.



2 La boîte de dialogue *Créer un certificat numérique* s'affiche. Dans la zone *Le nom de votre certificat*, tapez **Excel 2010** et cliquez sur *OK*.

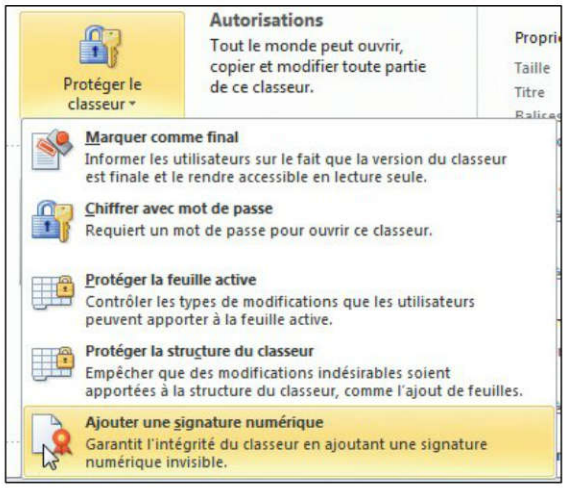


3 Cliquez sur *OK* pour fermer la boîte de message indiquant que le programme a créé avec succès le certificat.

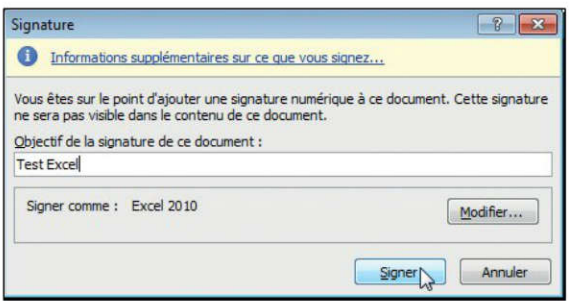


Copyright © 2012 Dunod.

4 Cliquez sur l'onglet *Fichier*, puis, si nécessaire, sur *Informations*. Cliquez sur *Protéger le classeur* puis sur *Ajouter une signature numérique*.



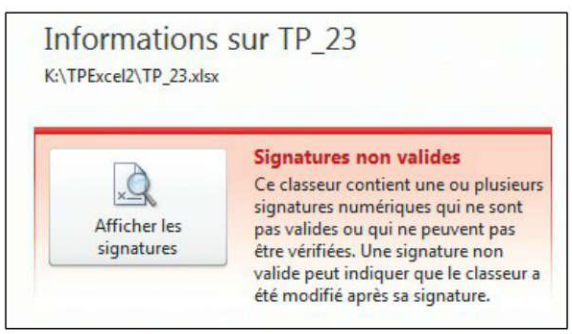
5 Cliquez sur *OK* pour fermer la boîte de message qui offre la possibilité d'accéder aux services de signatures d'Office MarketPlace. Dans la boîte de dialogue *Signature*, dans la zone *Objectif de la signature de ce document*, saisissez **Test Excel**. Vérifiez que le certificat Excel 2010 apparaît dans la zone *Signer comme* de la boîte de dialogue puis cliquez sur *Signer*.



6 La boîte de dialogue *Confirmation de signature* s'affiche et le classeur est marqué comme final. Cliquez sur *OK*.



7 Dans la page *Informations*, il est signalé que le classeur est signé avec une signature invalide : un certificat ainsi créé n'a pas de réelle valeur. Pour en obtenir un vrai, vous devez l'acheter auprès d'un organisme agréé.
Enregistrez et fermez le classeur TP_23.xlsx.



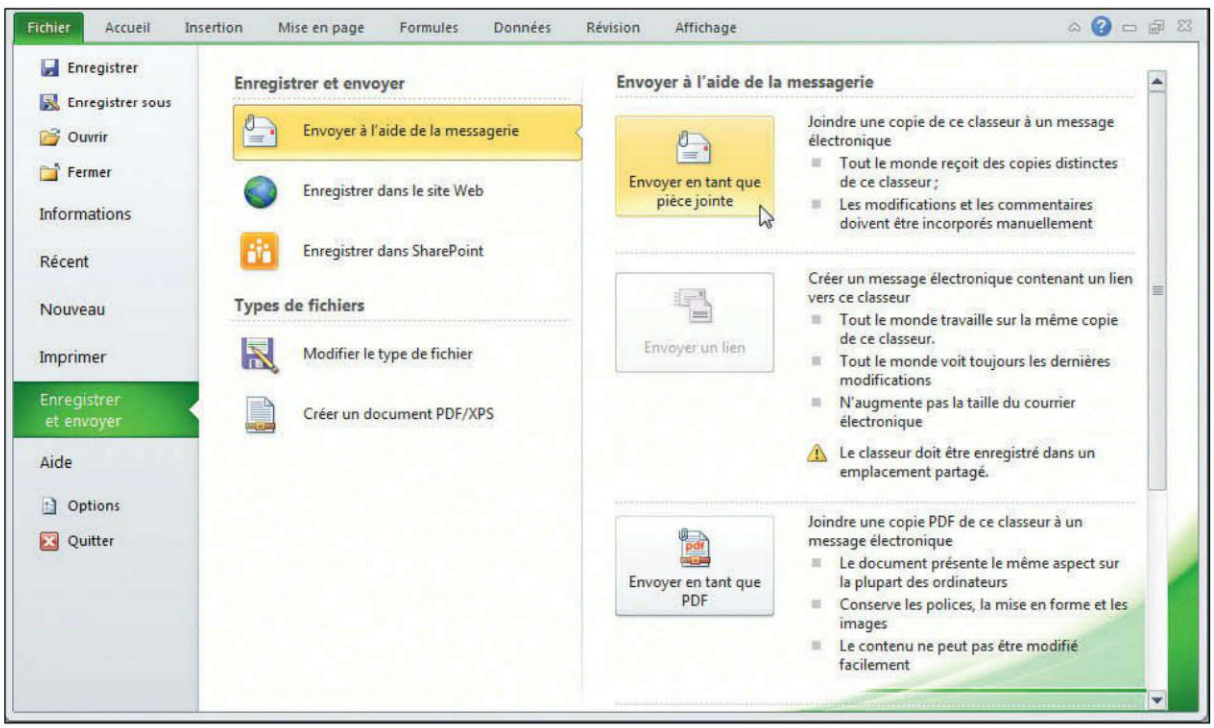
Copyright © 2012 Dunod.

TP 24 Envoi d'un classeur par messagerie électronique

Un classeur partagé doit être distribué vers ses destinataires. Excel permet l'envoi direct d'un classeur par messagerie électronique : vous partez du classeur au lieu de le rechercher depuis votre logiciel de messagerie électronique.

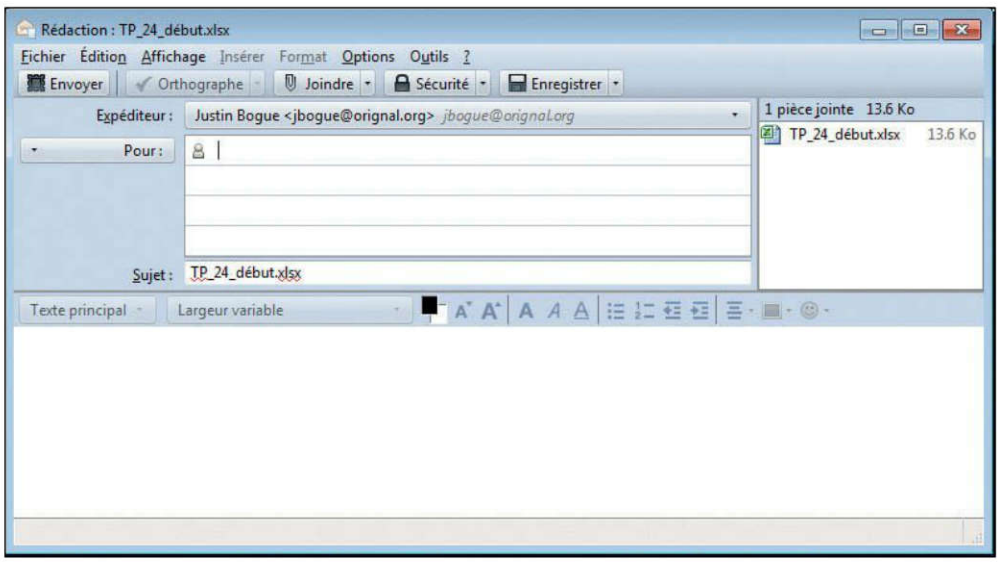
Prérequis : TP 11 et 12
Fichier employé : TP 24_début.xlsx
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_24_début.xlsx.
Cliquez sur l'onglet *Fichier*, cliquez sur *Enregistrer et envoyer*, ensuite sur *Envoyer à l'aide de la messagerie*, et enfin, sur *Envoyer en tant que pièce jointe*.

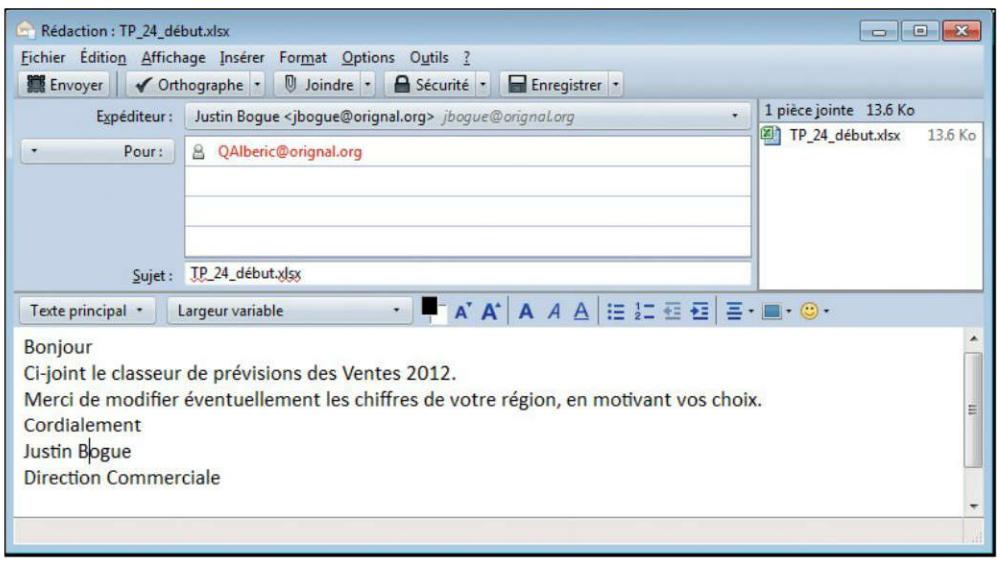


Copyright © 2012 Dunod.

2 Votre logiciel de messagerie s'ouvre. Le nouveau courrier électronique qui s'affiche contient le classeur partagé.



3 Tapez une adresse dans la zone *À*, saisissez le texte du message, puis cliquez sur *Envoyer*. Le programme de messagerie envoie le message. Si Excel a ouvert votre programme de messagerie, ce dernier se ferme. Fermez le classeur TP_24_debut.xlsx.

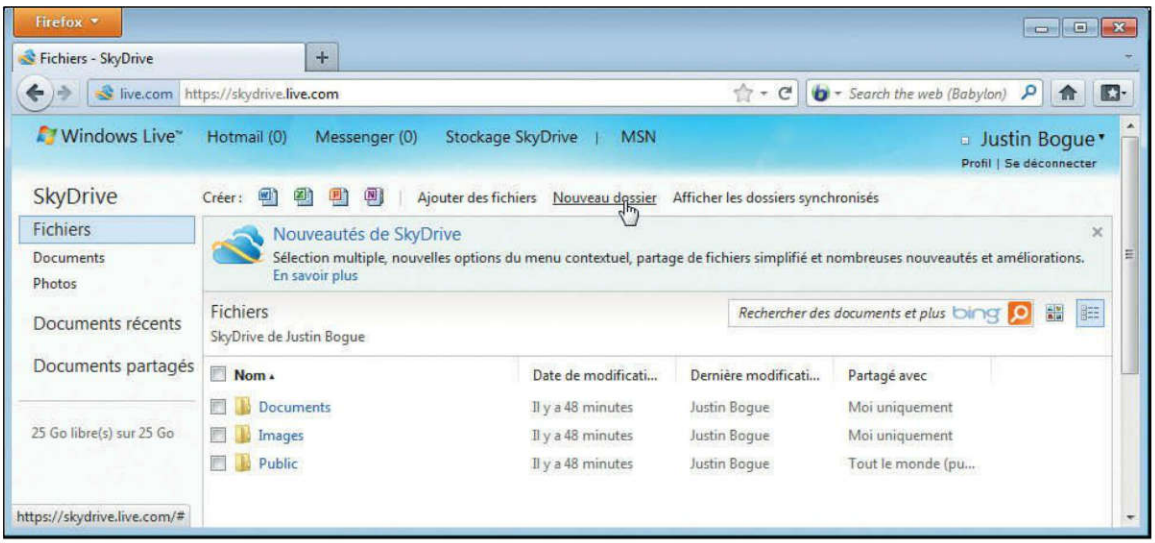


Copyright © 2012 Dunod.

Windows Live SkyDrive

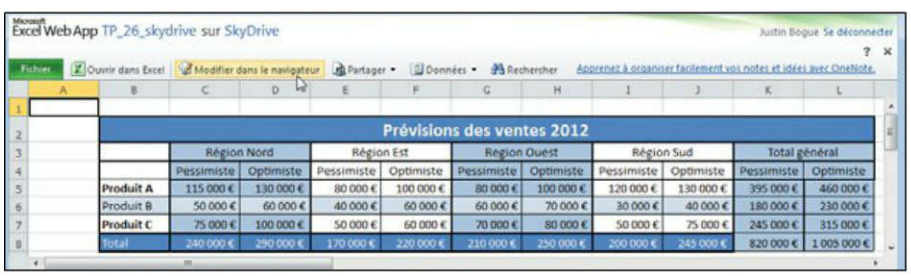
Vous avez certainement entendu parler du *cloud computing*, nommé en français informatique en nuage ou infonuagique. Grossièrement, ce concept consiste à déporter sur des serveurs distants des fichiers et applications de traitement traditionnellement situés sur des serveurs locaux ou sur l'ordinateur de l'utilisateur.

Microsoft propose une telle structure grâce à Windows Live SkyDrive : vous pouvez stocker, organiser et télécharger des fichiers sur les serveurs Windows Live et y accéder, vous ou bien les personnes que vous aurez autorisées, à partir de n'importe quel ordinateur équipé d'une connexion Internet. Ce n'est pas un site FTP nécessitant un logiciel client FTP. Cette possibilité d'accès à distance à partir de n'importe quel navigateur (ou presque) à des fichiers stockés en un emplacement unique élimine bien des problèmes de transfert et de synchronisation.



Windows Live SkyDrive offre la particularité de proposer une forme particulière des applications Office, nommée Web App. Celles-ci permettent de travailler depuis le navigateur. Il en existe pour tous les produits Office majeurs.

Cette partie s'intéresse aux données de base de SkyDrive et uniquement à l'emploi d'Excel Web App. Reportez-vous à la documentation pour découvrir toutes les autres possibilités de SkyDrive : ce livre n'y suffirait pas à lui seul.



Même si Excel Web App ne dispose pas de toutes les fonctionnalités d'Excel, il constitue déjà un produit très complet. Vous pouvez en outre télécharger un fichier SkyDrive vers votre ordinateur, le travailler dans Excel puis l'enregistrer à nouveau sur SkyDrive.

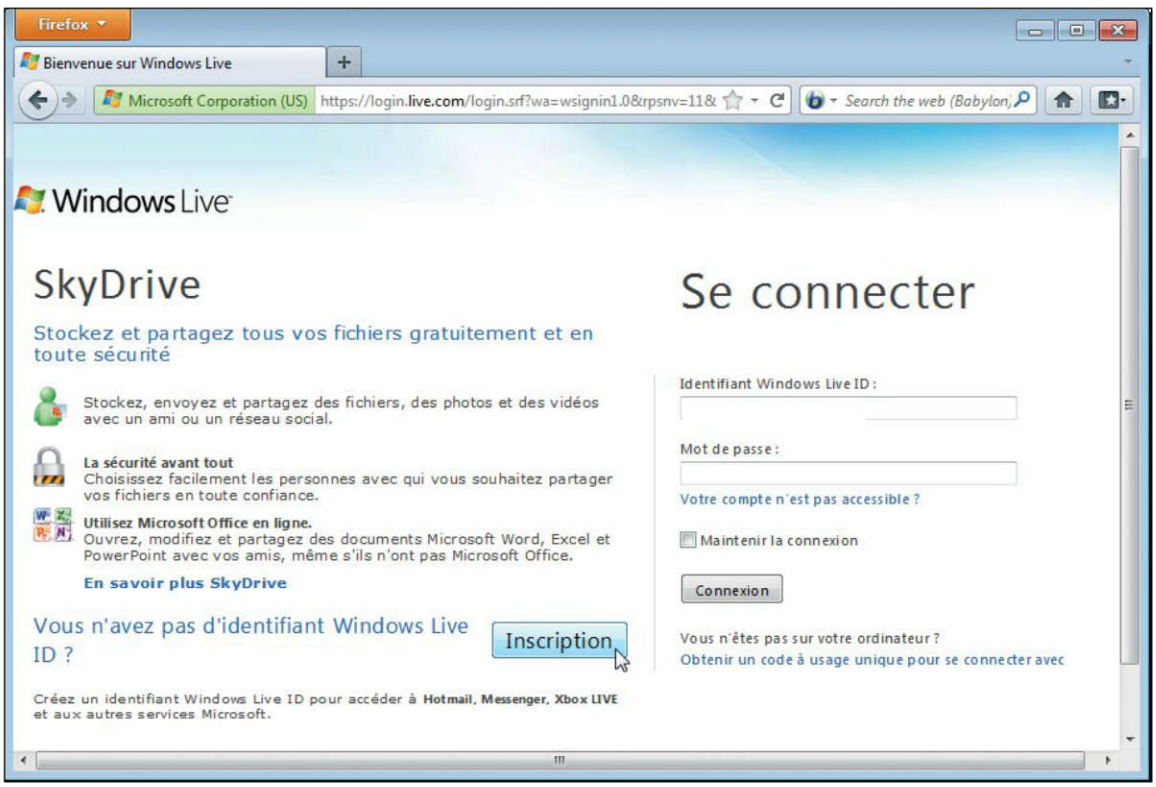
Copyright © 2012 Dunod.

Windows Live : création d'un identifiant

Pour pouvoir employer Windows Live SkyDrive, vous devez disposer d'un identifiant Windows Live. Si ce n'est pas encore le cas, voici comment en obtenir un.

Prérequis : néant
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Accédez à Windows Live sur http://skydrive.live.com. Au-dessous de *Vous n'avez pas encore d'identifiant Windows Live ID ?*, cliquez sur *Inscription*.

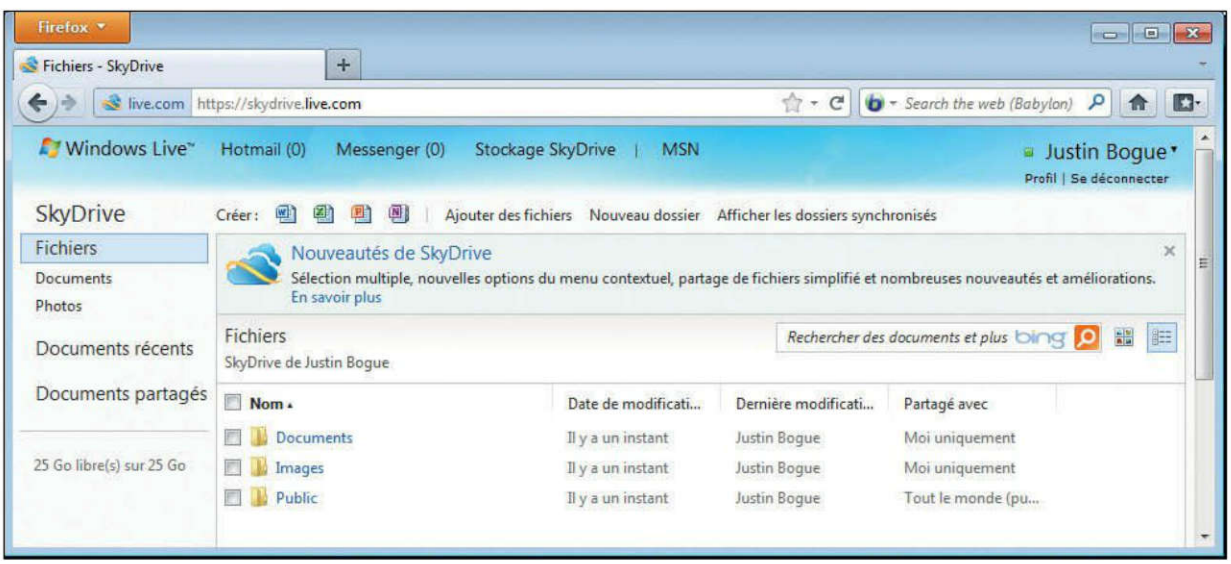


Copyright © 2012 Dunod.

2 Remplissez le formulaire pour créer votre identifiant Windows Live ID.



3 Vous accédez à votre espace SkyDrive.



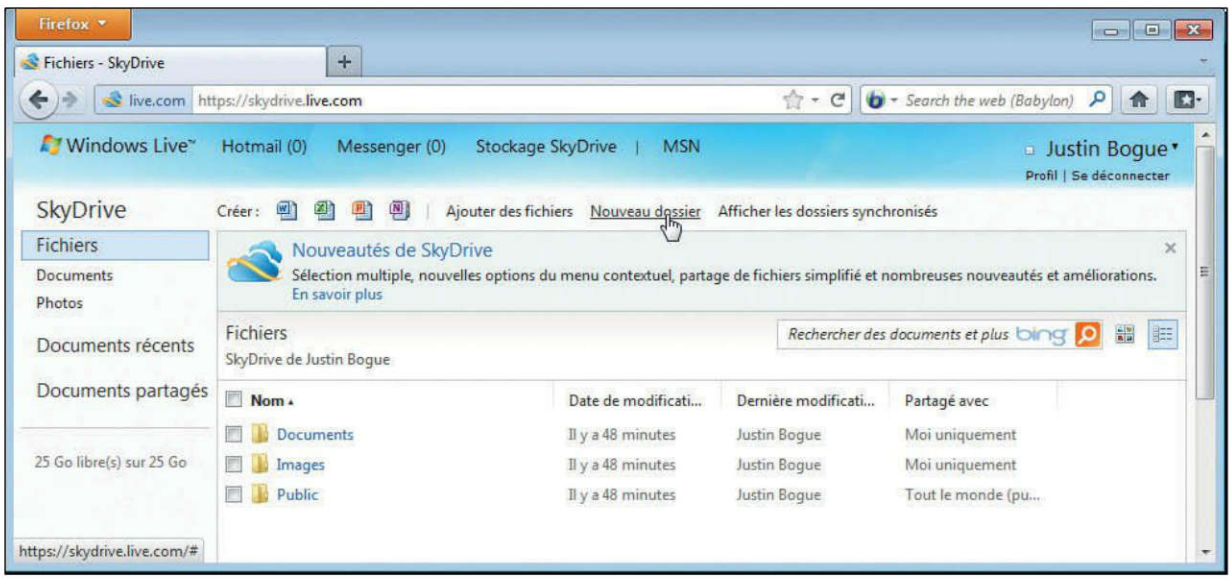
Copyright © 2012 Dunod.

TP 26 Windows Live : stocker un classeur

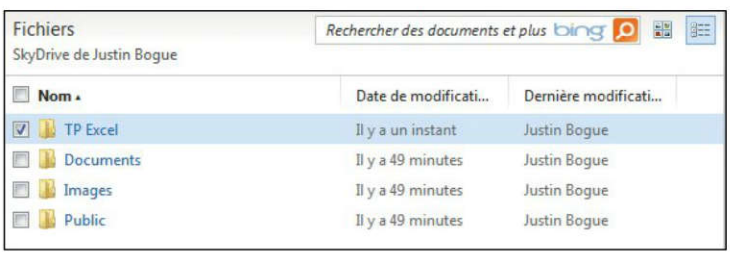
Une fois votre compte Windows Live SkyDrive ouvert, vous pouvez facilement y stocker des classeurs Excel.

Prérequis : TP 25
Fichier employé : TP_26_skydrive.xlsx
Temps de réalisation : 5 minutes

1 Accédez à votre espace SkyDrive à l'aide de votre identifiant Windows Live ID, puis cliquez sur *Nouveau dossier*.



2 Saisissez comme nom **TP Excel**. Le nouveau dossier apparaît dans votre espace.

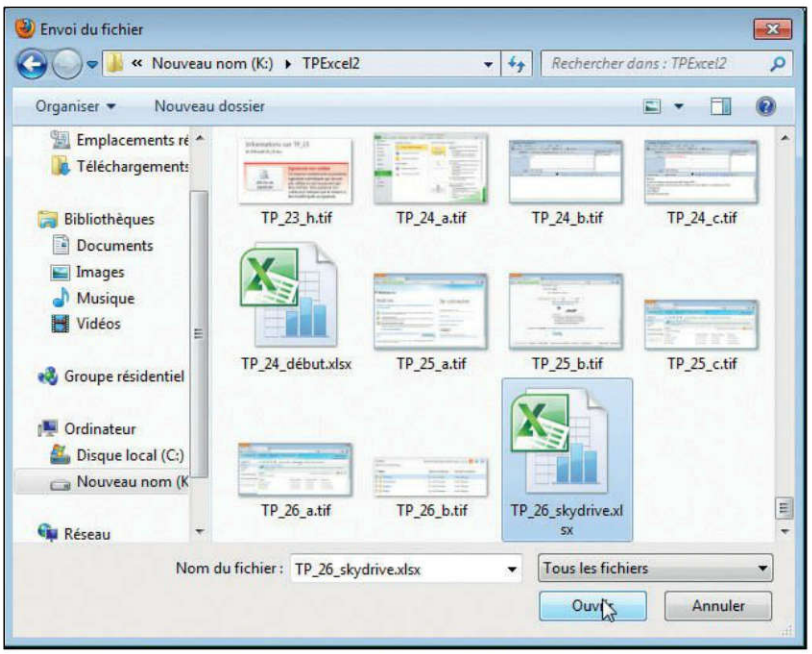


Copyright © 2012 Dunod.

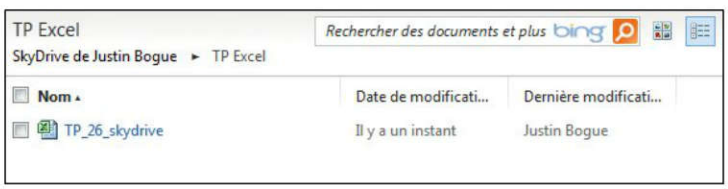
3 Cliquez sur le dossier *TP Excel*, puis sur *Ajouter des fichiers*.



4 Naviguez jusqu'au fichier *TP_26_skydrive.xlsx*, sélectionnez le fichier puis cliquez sur *Ouvrir*.



5 Une boîte de progression indique le chargement du fichier. Une fois le chargement terminé, fermez-la : le classeur est désormais présent dans votre espace SkyDrive.



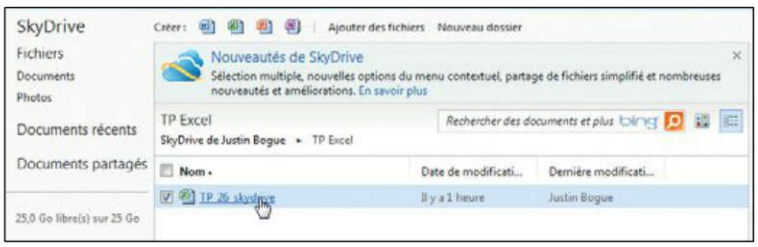
Copyright © 2012 Dunod.

TP 27 Windows Live : ouvrir un classeur dans Excel Web App

Il est très simple d'examiner des classeurs stockés sur SkyDrive.

Prérequis : TP 25 et 26
Fichier employé : TP_26_skydrive.xlsx (sur SkyDrive)
Temps de réalisation : 5 minutes

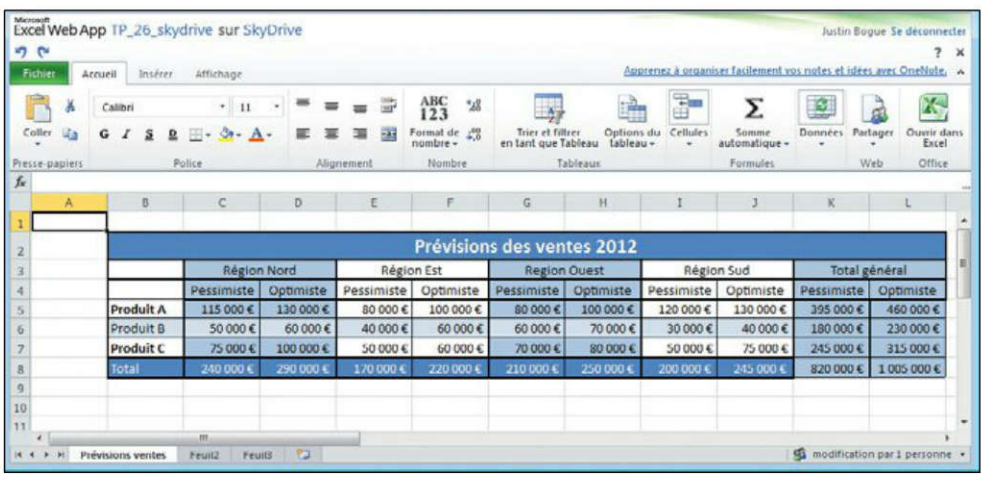
1 Accédez, si nécessaire, à votre espace SkyDrive. Cliquez sur le dossier *TP Excel*, puis double-cliquez sur le fichier *TP_26_SkyDrive.xlsx*.



2 Le fichier s'ouvre dans Excel Web App. Cliquez sur *Modifier dans le navigateur*.



3 Vous voyez apparaître l'interface familière d'Excel. Il s'agit en réalité d'Excel Web App, une application fondée sur un navigateur qui permet d'afficher et d'utiliser des classeurs Excel depuis un navigateur. Toutes les fonctionnalités Excel n'y sont pas disponibles ou ne fonctionnent pas de la même façon. Il n'est pas nécessaire qu'Office soit installé sur l'ordinateur employé.



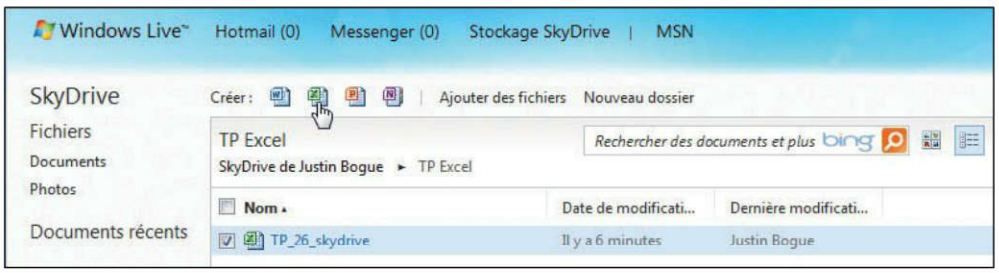
Copyright © 2012 Dunod.

TP 28 Windows Live : créer un classeur dans Excel Web App

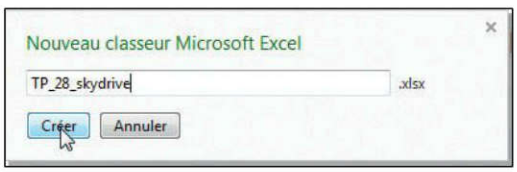
En plus d'examiner et de modifier des classeurs dans SkyDrive, vous pouvez directement y créer un nouveau classeur.

Prérequis : TP 25, 26 et 27
Fichier employé : néant
Temps de réalisation : 5 minutes

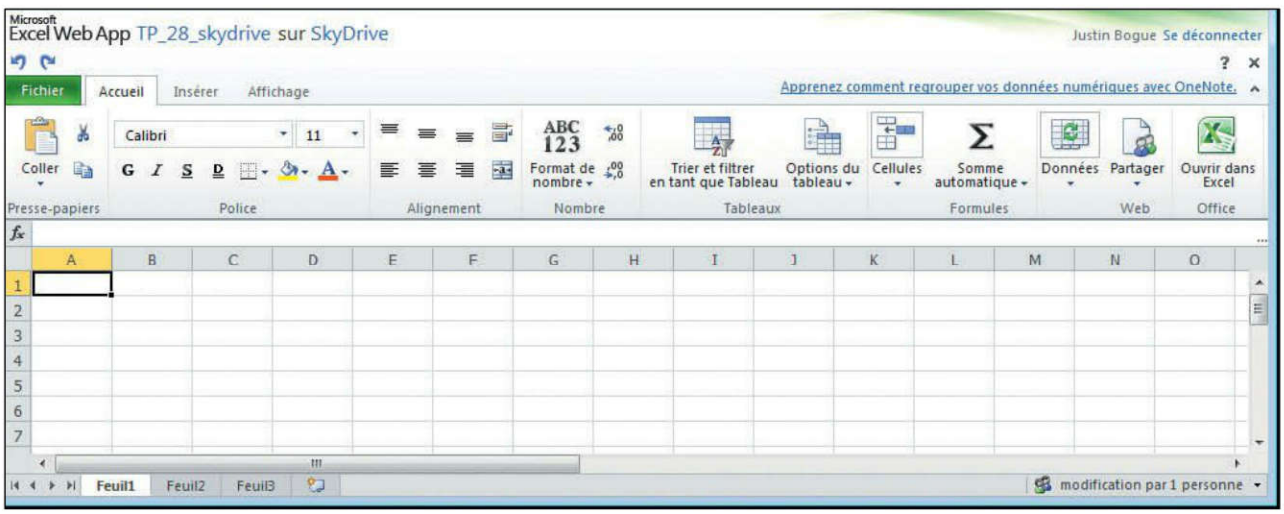
1 Accédez, si nécessaire, à votre espace SkyDrive, cliquez sur le dossier *TP Excel*, puis sur le bouton *Créer un classeur Excel*.



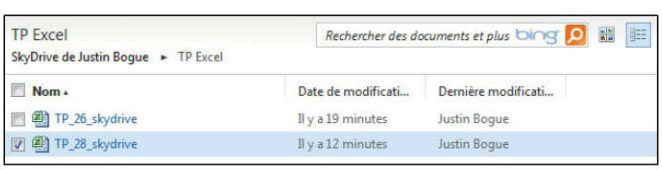
2 Dans la boîte de dialogue, saisissez comme nom **TP_28_skydrive** puis cliquez sur *OK*.



3 Le classeur s'ouvre dans Excel Web App. Fermez-le. Il n'est pas nécessaire de l'enregistrer dans la mesure où Excel Web App se charge immédiatement et automatiquement de conserver en mémoire les modifications apportées.



4 Vous pouvez dès lors voir le nouveau classeur présent dans votre espace SkyDrive.



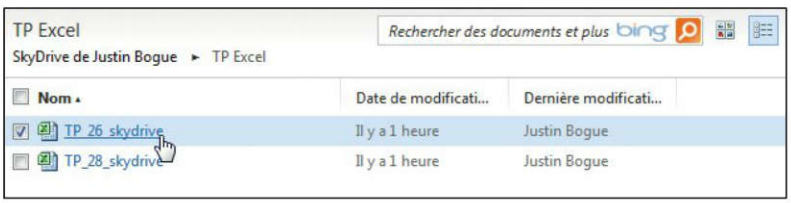
Copyright © 2012 Dunod.

TP 29 Windows Live : ouvrir le classeur dans Excel

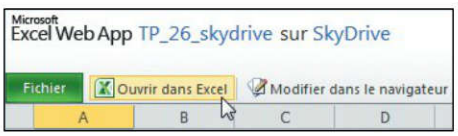
Excel Web App ne dispose pas de toutes les fonctionnalités d'Excel Office. Vous pouvez toutefois ouvrir le classeur situé sur SkyDrive localement dans votre version d'Excel.

Prérequis : TP 25 à 28
Fichier employé : TP_26_skydrive.xlsx
Temps de réalisation : 5 minutes

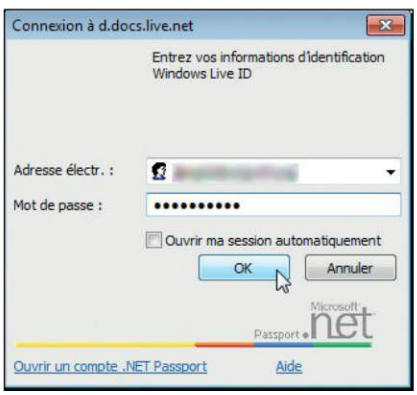
1 Accédez, si nécessaire, à votre espace SkyDrive, cliquez sur le dossier *TP Excel*, puis sélectionnez le fichier *TP_26_skydrive.xlsx*.



2 Le fichier s'affiche dans Excel Web App. Dans l'onglet *Accueil*, dans le groupe *Office*, cliquez sur *Ouvrir dans Excel*.

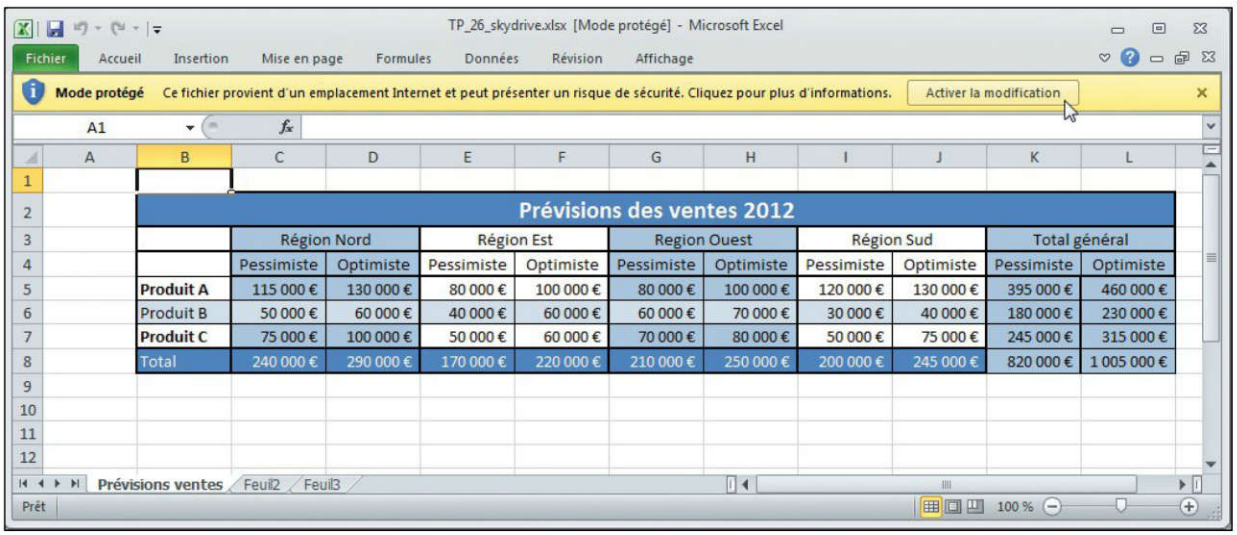


3 Dans la boîte de dialogue d'avertissement qui s'affiche, cliquez sur *OK*. Saisissez dans la boîte de dialogue suivante vos identifiants Windows Live, puis cliquez sur *OK*.



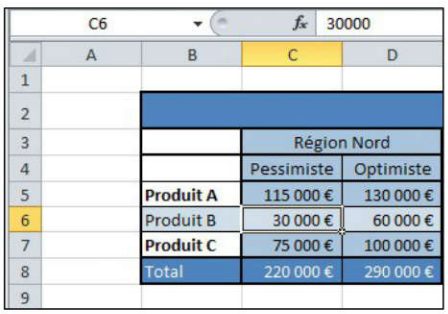
Copyright © 2012 Dunod.

4 Le fichier s'affiche dans Excel. Remarquez la barre d'avertissement. Cliquez dedans sur *Activer la modification*.



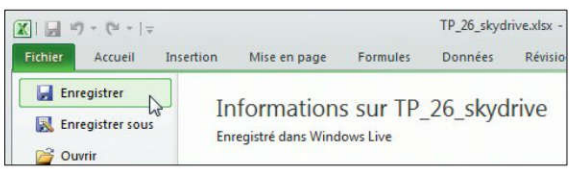
Prévisions des ventes 2012											
	Région Nord		Région Est		Region Ouest		Région Sud		Total général	
	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste	Pessimiste	Optimiste
Produit A	115 000 €	130 000 €	80 000 €	100 000 €	80 000 €	100 000 €	120 000 €	130 000 €	395 000 €	460 000 €
Produit B	50 000 €	60 000 €	40 000 €	60 000 €	60 000 €	70 000 €	30 000 €	40 000 €	180 000 €	230 000 €
Produit C	75 000 €	100 000 €	50 000 €	60 000 €	70 000 €	80 000 €	50 000 €	75 000 €	245 000 €	315 000 €
Total	240 000 €	290 000 €	170 000 €	220 000 €	210 000 €	250 000 €	200 000 €	245 000 €	820 000 €	1 005 000 €

5 Dans Excel, cliquez sur la cellule *C6* et saisissez **30000**.

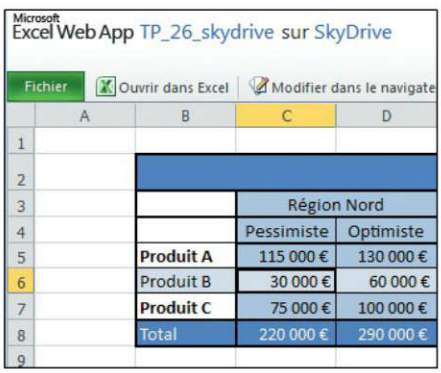


	Région Nord	
	Pessimiste	Optimiste
Produit A	115 000 €	130 000 €
Produit B	30 000 €	60 000 €
Produit C	75 000 €	100 000 €
Total	220 000 €	290 000 €

6 Dans l'onglet *Fichier*, cliquez sur *Enregistrer*. Le classeur est enregistré automatiquement dans le dossier SkyDrive. Fermez le classeur et Excel.



7 Revenez dans SkyDrive et affichez le fichier Excel dans votre navigateur, à l'aide d'Excel Web App : la modification apportée sous Excel sur votre ordinateur y est présente.
Fermez le fichier TP_26_skydrive, puis fermez votre navigateur si vous ne poursuivez pas immédiatement avec l'exercice suivant.



	Région Nord	
	Pessimiste	Optimiste
Produit A	115 000 €	130 000 €
Produit B	30 000 €	60 000 €
Produit C	75 000 €	100 000 €
Total	220 000 €	290 000 €

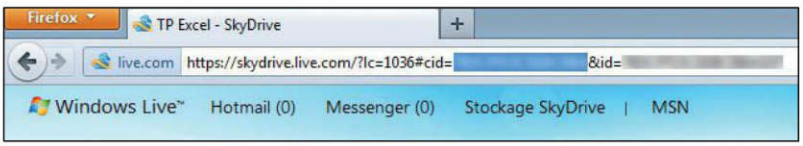
Copyright © 2012 Dunod.

Windows Live : ajouter une connexion réseau vers SkyDrive

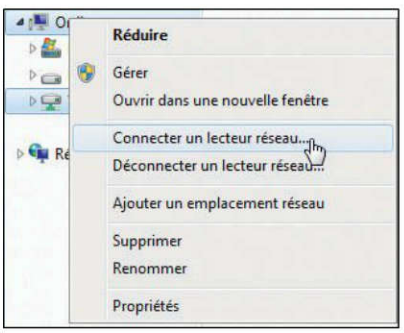
Pour gérer plus facilement les dossiers stockés dans votre espace SkyDrive, il peut être intéressant de créer une connexion réseau sur votre ordinateur vers votre espace SkyDrive.

Prérequis : TP 25 à 29
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Ouvrez votre navigateur et saisissez dans la barre d'adresse, l'adresse suivante : **http://skydrive.live.com**. Votre espace SkyDrive s'ouvre. Cliquez sur *TP Excel*. La barre d'adresse affiche une adresse de type *https://skydrive.live.com/?lc=1036#cid=xxxxxxxxxxxxx&id=nnnnnnnnnnn*, où *xxxx…* et *nnn…* sont des séquences alphanumériques. Ces séquences sont masquées dans la copie d'écran pour raison de confidentialité. Sélectionnez la partie *xxxx* dans la barre d'adresse, et placez-la dans le Presse-papiers en appuyant sur *Ctrl+C*.

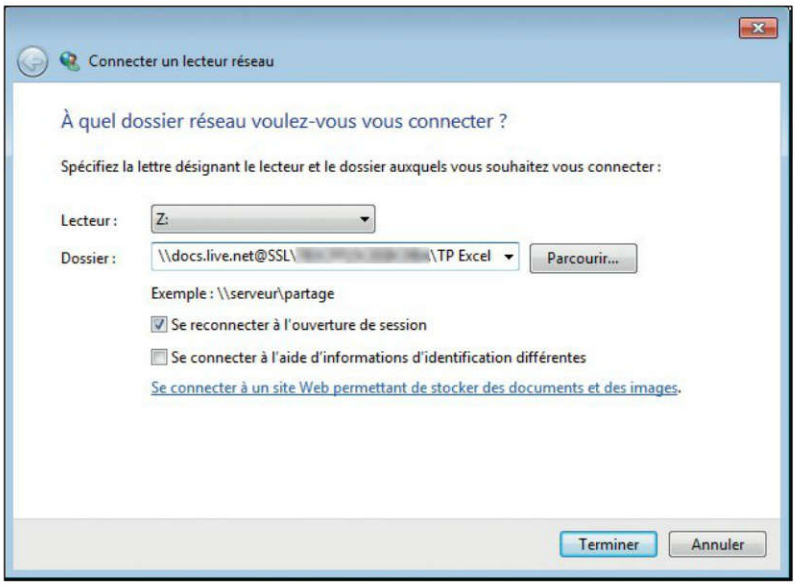


2 Ouvrez l'explorateur Windows. Dans la barre de gauche, effectuez un clic droit sur *Ordinateur* et choisissez *Connecter un lecteur réseau*.



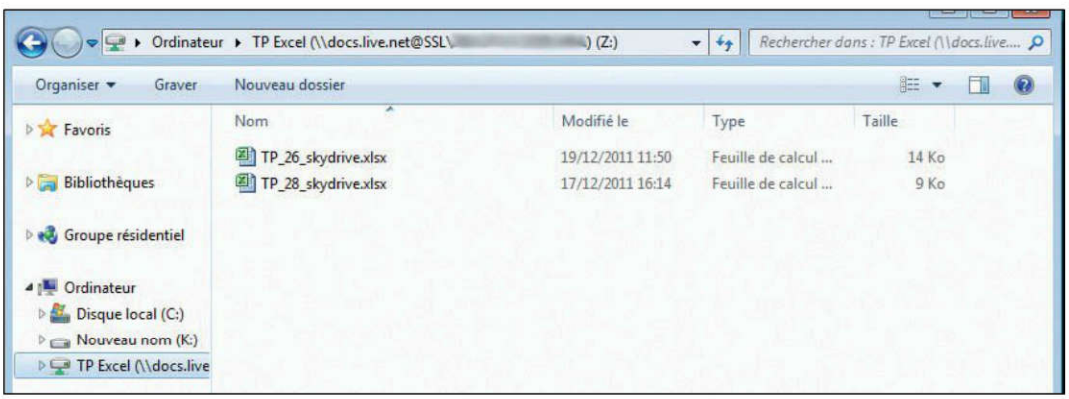
Copyright © 2012 Dunod.

3 Dans la boîte de dialogue, choisissez une lettre pour le lecteur réseau, puis saisissez **\\docs.live.net@SSL**. Collez le contenu du Presse-papiers par *Ctrl+V*, puis saisissez le nom du répertoire précédé d'un slash, ici **\TP Excel**.



4 Une boîte de dialogue signale que l'ordinateur tente d'établir une connexion. Attendez. Dans la boîte de dialogue qui s'affiche ensuite, saisissez votre identifiant Windows Live et votre mot de passe. La connexion réseau s'établit, et vous voyez directement le contenu de votre espace SkyDrive. En cliquant sur un classeur, vous l'ouvrez dans votre version d'Excel.

Cette connexion permet également de gérer directement le contenu de votre espace SkyDrive : ajout, suppression ou modification de nom.



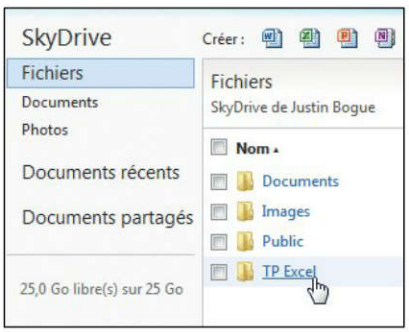
Copyright © 2012 Dunod.

TP 31 Windows Live : partager un dossier

Vous pouvez collaborer simultanément avec d'autres personnes sur des classeurs ou autres fichiers situés dans un même dossier. Un dossier partagé ne doit donc contenir que des fichiers pouvant être partagés.

Prérequis : TP 25 à 29
Fichier employé : néant
Temps de réalisation : 5 minutes

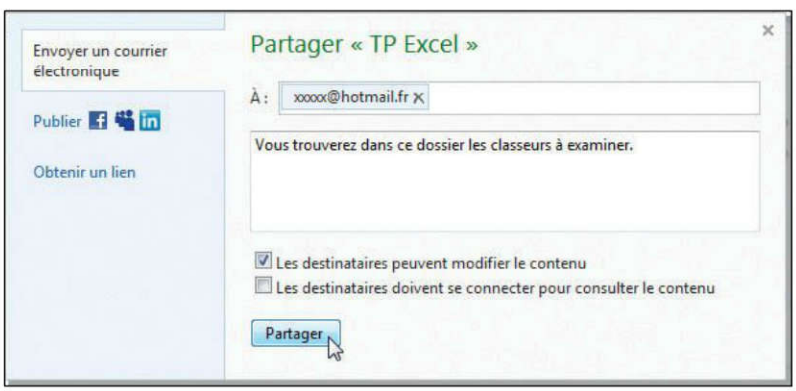
1 Connectez-vous, si nécessaire, à Windows Live, puis cliquez sur le dossier *TP Excel*.



2 Dans le menu de droite, cliquez sur *Partager le dossier*.



3 Dans la boîte de dialogue qui s'ouvre, saisissez l'adresse de messagerie de la personne avec qui vous voulez partager le dossier, saisissez un message facultatif et cliquez sur *Partager*. Un message électronique va lui être envoyé. Il se pourrait qu'un message de sécurité demande de confirmer. Saisissez alors le code présenté au-dessus dans la zone, puis cliquez sur *Continuer*. Dans la partie droite de SkyDrive, votre dossier est signalé comme partagé.



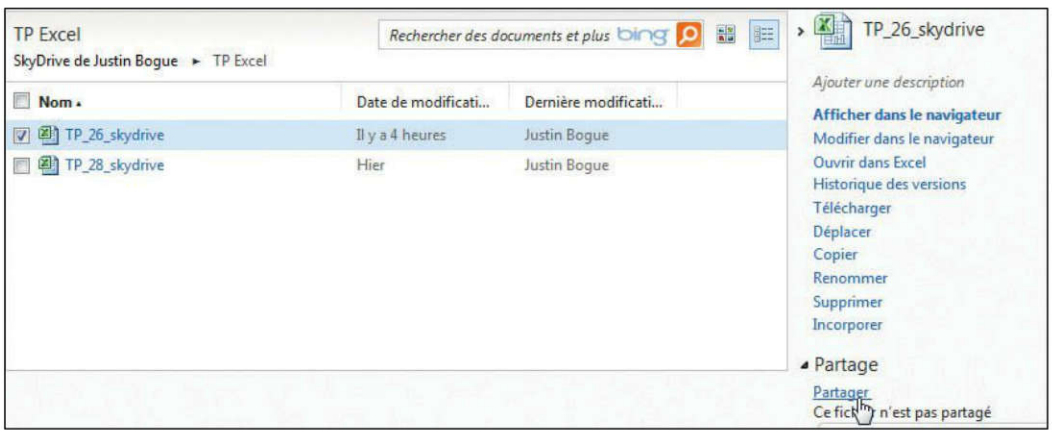
Copyright © 2012 Dunod.

TP 32 Windows Live : partager un classeur

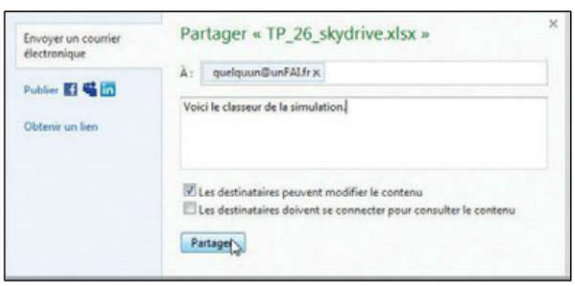
Plutôt que de partager tout un dossier, vous pouvez ne partager qu'un seul classeur.

Prérequis : TP 25 à 29, TP 31
Fichier employé : néant
Temps de réalisation : 5 minutes

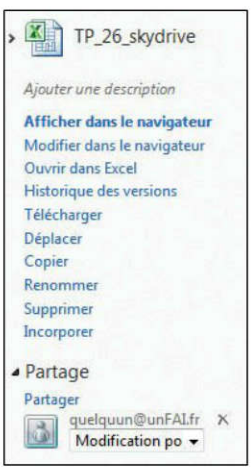
1 Connectez-vous, si nécessaire, à Windows Live, cliquez sur le dossier *TP Excel*, puis cochez la case à côté du fichier *TP_26_skydrive.xlsx*. Dans le menu de droite, cliquez sur *Partager*.



2 Dans la boîte de dialogue qui s'ouvre, saisissez l'adresse de messagerie de la personne avec qui vous voulez partager le fichier (cela peut être ici une adresse fictive), saisissez un message facultatif, et cliquez sur *Partager*. Un message électronique va lui être envoyé. Comme précédemment, vous pourriez voir apparaître un message de sécurité demandant de confirmer. Saisissez dans ce cas le code présenté au-dessus dans la zone, puis cliquez sur *Continuer*.



3 Dans la partie droite de SkyDrive, votre dossier est signalé comme partagé. Cliquez sur la croix rouge à côté du nom de la personne avec qui vous avez partagé le dossier pour annuler le partage.



Copyright © 2012 Dunod.

Copyright © 2012 Dunod.

Macros et VBA (Visual Basic for Applications)

À l'origine, une macro, abrégé de macrocommande, permettait d'effectuer rapidement une série de frappes répétitives : des raccourcis pour des commandes de l'interface utilisateur. Excel a introduit un langage de programmation à part entière, finalement devenu Visual Basic pour Applications, ou VBA, qui permettait d'exécuter des tâches autrement complexes. L'habitude est toutefois restée de nommer macro toute procédure (routine ou fonction) écrite en VBA.

VBA n'est pas spécifique à Excel, mais peut être utilisé avec les autres produits de la famille Office. C'est une excellente introduction à la programmation en Visual Basic.

Néanmoins, le fait est que c'est probablement avec Excel que VBA se révèle le plus intéressant. Ses possibilités d'emploi sont innombrables : personnalisation de fonctions intégrées, création de nouvelles fonctions personnalisées, construction d'une interface personnalisée pour accomplir en un clic des tâches répétitives parfois très complexes, etc.

VBA est un langage de programmation orienté objet. Cela signifie simplement que chaque composant d'Excel (classeur, feuille de calcul, plage de cellules ou cellule, graphique ou point dans un graphique) est un objet qui peut être manipulé par VBA.

Tout classeur se compose d'un ensemble d'objets qui peuvent être manipulés par programmation. Le classeur, les feuilles de calcul et les lignes sont autant d'objets que vous manipulez lors du travail avec un tableur. Leur liste complète est impressionnante : vous en découvrirez un certain nombre peu à peu dans ce chapitre. VBA, langage de programmation orienté objet, est capable de les manipuler.

Certains d'entre eux appartiennent à une collection. Une collection est un ensemble d'objets similaires pouvant être manipulés simultanément : une collection est elle-même un objet. La collection des classeurs en est un exemple. Il est constitué de chacun des objets classeur que vous avez à votre disposition. Vous vous référez généralement à un élément d'une collection en mettant son nom entre parenthèses, comme dans l'exemple suivant :

```
Collection("Nom_élément")
```

Les principales collections d'objets du tableur Excel sont `WorkBooks` (classeur), `Sheets` (feuille de calcul) et `Range` (cellules). Elles sont utilisables selon une hiérarchie descendante :

`Workbooks("Ventes 2009").Sheets("Janvier").Range("B2")` désigne la cellule B2 de la feuille Janvier du classeur Ventes 2009 ;

`Sheets("Janvier").Range("B2")` désigne la cellule B2 de la feuille Janvier du classeur actif (ouvert et affiché) ;

`Range("B2")` désigne la cellule B2 de la feuille active.

Veillez à ne pas confondre collection et objet membre d'une collection. Le nom d'une collection se termine par un « s ». Ainsi, `Sheets` est la collection de toutes les feuilles d'un classeur, `Sheet` est une feuille.

Une méthode est une action qui peut être effectuée par un objet. Un objet de classeur possède des méthodes. L'objet classeur (`Workbook`) possède par exemple une méthode `Open` (ouvrir) et une méthode `Close` (fermer). Dans une instruction VBA, vous employez la méthode d'un objet en faisant

Copyright © 2012 Dunod.

suivre son nom de celui de la méthode, séparés par un point :

```
Collection("Nom_élément").Nom_méthode
```

Les propriétés décrivent un objet. Certaines sont en lecture seule, d'autres en lecture/écriture. Une propriété en lecture seule ne peut être modifiée. Ainsi, un objet classeur possède une propriété `Path` (chemin), qui décrit son chemin d'accès complet. Elle est en lecture seule car vous ne pouvez pas modifier le chemin d'un classeur sans l'enregistrer dans un autre dossier.

Une propriété possède donc une valeur. Vous modifiez la valeur d'une propriété en lecture/écriture en lui affectant une nouvelle valeur. Vous spécifiez une propriété comme une méthode :

```
Collection ("Nom_élément").Propriété = Valeur
```

Si vous n'affectez pas de valeur, l'instruction indique (ou renvoie) la valeur courante. Parmi les propriétés classiques figurent `Visible` (à utiliser avec `True` et `False` pour afficher ou masquer), `Value` (valeur d'une cellule), `Count` (nombre de cellules d'une page, de feuilles de classeur, etc.).

Les propriétés et méthodes d'un objet portent le nom de membres. L'éditeur VBA propose d'ailleurs deux dispositifs très intéressants, la liste des membres et l'Information rapide automatique, qui simplifient considérablement le travail de rédaction de code. Nous y reviendrons ultérieurement.

Une procédure est un ensemble d'instructions de programmation qui effectue une tâche spécifique ou renvoie un résultat. Il en existe de deux sortes :

Une *sous-routine* effectue une tâche spécifique. Les macros enregistrées directement sont des sous-routines.

Une *fonction* est conçue pour renvoyer un résultat.

Lorsque vous écrivez du code, vous modifiez ou créez une procédure. Le processus est identique qu'il s'agisse d'une sous-routine ou d'une fonction.

Il est impossible dans cet ouvrage d'entrer dans le détail de la programmation orientée objet : nous nous limiterons à vous montrer des exemples de mise en œuvre de procédures VBA afin d'améliorer votre confort et votre productivité. Pour en savoir plus, vous devrez vous en reporter à un ouvrage plus spécialisé.

Copyright © 2012 Dunod.

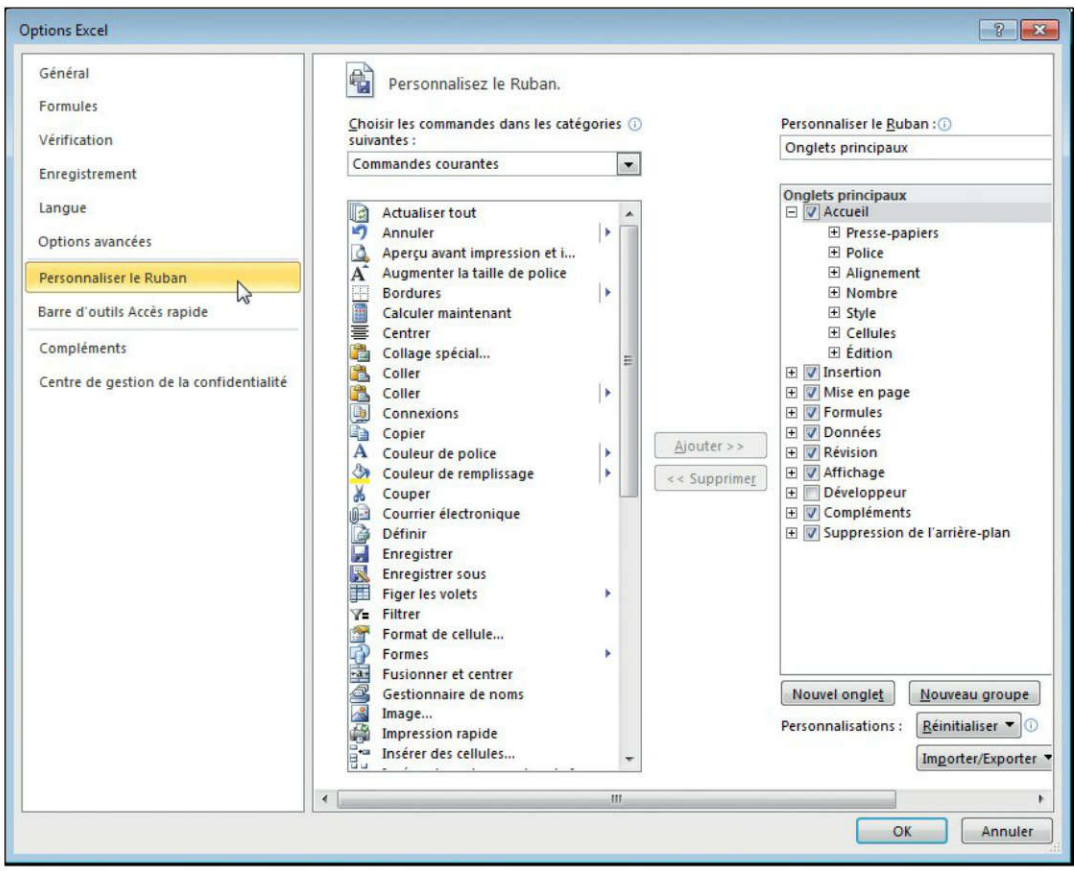
TP 33

Prérequis au travail avec VBA

L'onglet Développeur, indispensable au travail avec les macros (et donc VBA), n'est pas affiché par défaut. Si un groupe Macros est disponible à l'extrême droite de l'onglet Affichage, même en l'absence de l'onglet Développeur, ses fonctionnalités restent plus limitées que celles de l'onglet Développeur : mieux vaut rendre visible cet onglet lors du travail avec les macros.

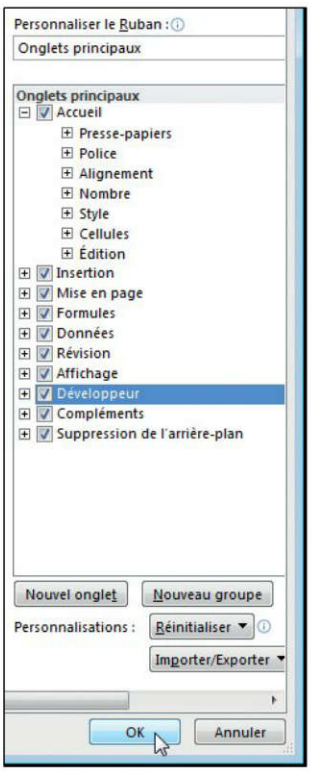
Prérequis : néant
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Cliquez sur l'onglet *Fichier*. Cliquez sur *Options*, puis sur la catégorie *Personnaliser le Ruban*.



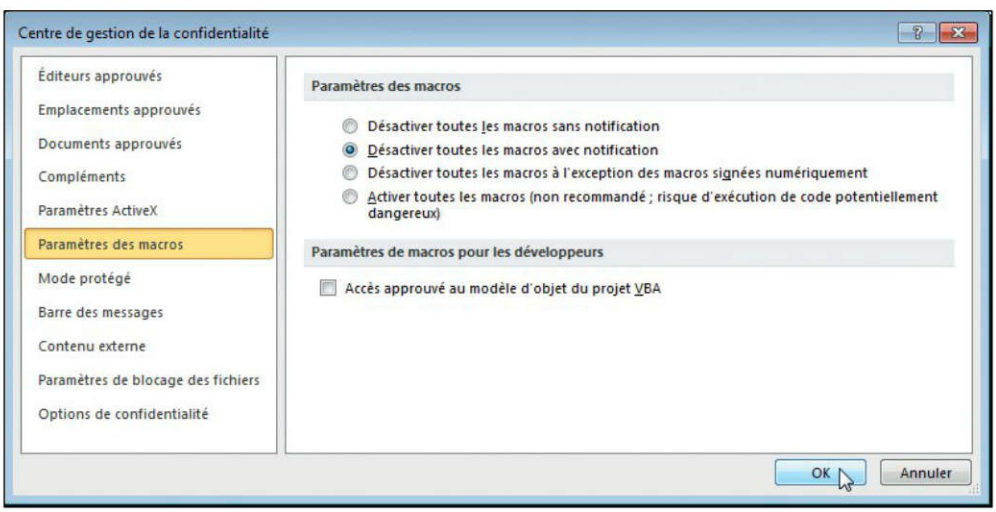
Copyright © 2012 Dunod.

2 Si l'onglet *Développeur* n'apparaît pas dans la liste des onglets principaux, sous *Personnaliser le Ruban*, dans la liste *Onglets principaux*, cliquez sur *Développeur*, puis cliquez sur *Ajouter*. S'il y est déjà présent, cochez sa case. Cliquez sur *OK*.



3 Dans le ruban, cliquez sur le nouvel onglet *Développeur*. Dans le groupe *Code*, cliquez sur l'option *Sécurité des macros*. Conservez le réglage par défaut, *Désactiver toutes les macros avec notification*, qui désactive les macros tout en activant les alertes de sécurité. Vous pourrez ainsi choisir d'activer ces macros au cas par cas. Cliquez sur *OK*.

Ces modifications sont apportées à Excel de façon générale, et non pas à un classeur particulier. Elles resteront actives tant que vous ne les changerez pas.



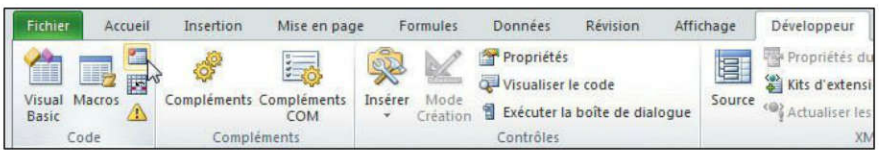
Copyright © 2012 Dunod.

TP 34 Enregistrement d'une macro

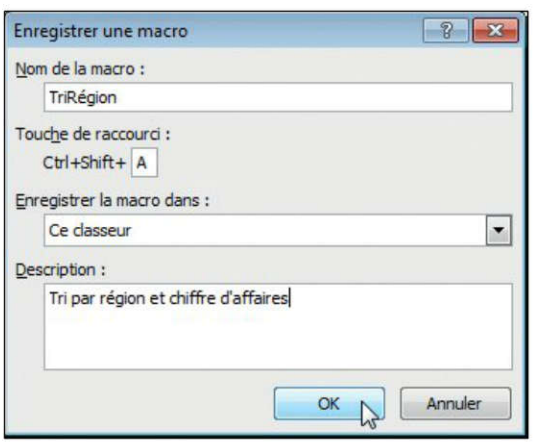
Nous allons dans ce TP enregistrer une macro simple : c'est le meilleur moyen de comprendre le fonctionnement de VBA, tout en offrant un excellent exemple de la façon dont les macros peuvent vous simplifier la vie. Le classeur de cet exercice présente la liste des commerciaux d'une entreprise fictive. Il indique leur nom, leur région d'activité et leur chiffre d'affaires. Cette liste pourrait notamment être créée automatiquement à partir d'un fichier texte, et s'avère peu exploitable sous sa forme actuelle. Vous souhaitez pouvoir la trier de différentes façons, afin d'interpréter plus rapidement les données. Comme vous devrez effectuer fréquemment ces opérations, vous désirez créer des macros.

Prérequis : TP 33
Fichier employé : TP_34_début.xls
Temps de réalisation : 15 minutes

1 Ouvrez le classeur TP_34_début.xlsx et enregistrez-le sous le nom TP_34.xlsx.
Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Enregistrer une macro*.

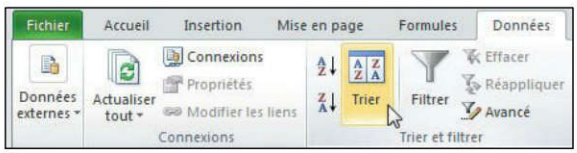


2 Dans la boîte de dialogue *Enregistrer une macro*, saisissez dans la zone *Nom de la macro* **TriRégion**. Dans la zone *Touche de raccourci* précédée de Ctrl+, appuyez sur la touche *MAJ* puis sur la touche *A*. Vous voyez apparaître Ctrl+Shift, avant la lettre A dans la zone. Dans la zone de la liste déroulante *Enregistrer la macro dans*, laissez la valeur par défaut, *Ce classeur*. Saisissez le bref descriptif suivant dans la boîte *Description* : **Tri par région et chiffre d'affaires**. Cliquez sur *OK*.
Un nom de macro peut contenir des lettres, des chiffres, le caractère de soulignement, mais jamais d'espace. Il doit commencer par une lettre et sa longueur ne doit pas dépasser 64 caractères. Préférez des noms courts et évocateurs.

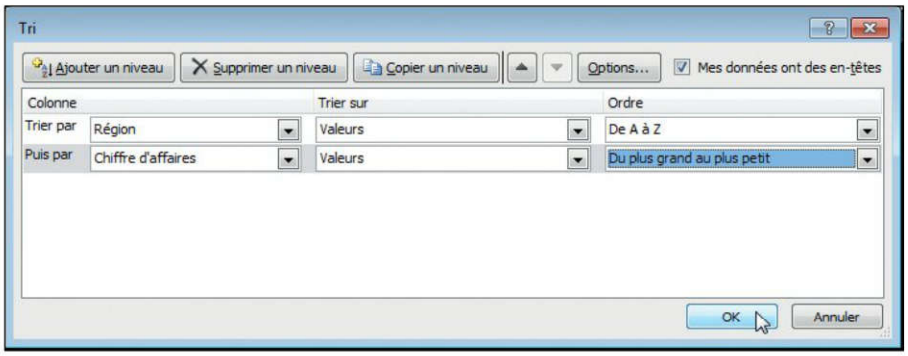


Copyright © 2012 Dunod.

3 Appuyez sur *Ctrl+A* pour sélectionner la totalité de la feuille de calcul, puis, dans le menu *Données*, cliquez sur *Trier*.

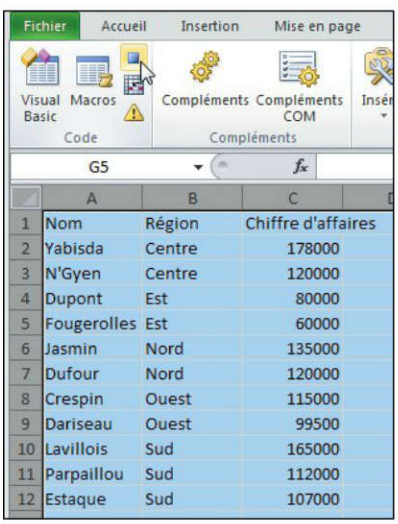


4 Dans la boîte de dialogue *Tri*, choisissez dans la zone *Trier par* **Région**. Cliquez sur *Ajoutez un niveau*, choisissez dans la zone *Trier par* **Chiffre d'affaires**, puis fixez l'ordre de tri du plus grand au plus petit. Cliquez sur *OK*.

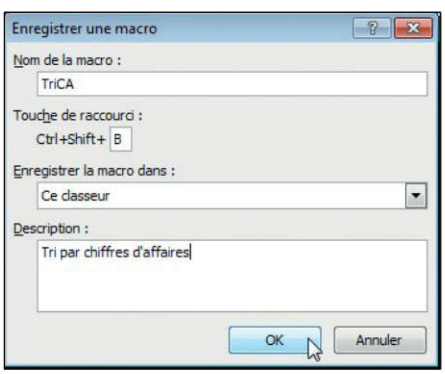


5 La feuille de calcul est triée selon vos critères. Cliquez sur l'onglet *Développeur*, puis cliquez sur *Arrêter l'enregistrement* dans le groupe *Code*.

Il est facile d'oublier qu'un enregistrement est en cours. Celui-ci se poursuivant jusqu'à son arrêt explicite, veillez à toujours arrêter un enregistrement dès que les tâches sont accomplies. Si vous avez oublié d'arrêter l'enregistrement d'une macro à temps, vous n'avez pas à tout enregistrer à nouveau : il suffira d'éditer la macro et de supprimer les actions superflues. Nous verrons par la suite comment éditer une macro.



6 Cliquez à nouveau sur *Enregistrer une macro*. Dans la boîte de dialogue *Enregistrer une macro*, saisissez dans la zone *Nom de la macro* **TriCA**. Dans la zone *Touche de raccourci* précédée de Ctrl+, appuyez sur la touche *Maj* puis sur la touche *B*. Dans la zone de liste déroulante *Enregistrer la macro dans*, laissez la valeur par défaut, *Ce classeur*. Saisissez dans la boîte *Description* le bref descriptif suivant : **Tri par chiffres d'affaires**. Cliquez sur *OK*.



Copyright © 2012 Dunod.

7 Appuyez sur *Ctrl+A* pour sélectionner la totalité de la feuille de calcul, puis, dans le menu *Données*, cliquez sur *Trier*. Dans la boîte de dialogue *Tri*, cliquez sur *Supprimer un niveau* pour conserver uniquement le tri par chiffre d'affaires. Cliquez sur *OK*.

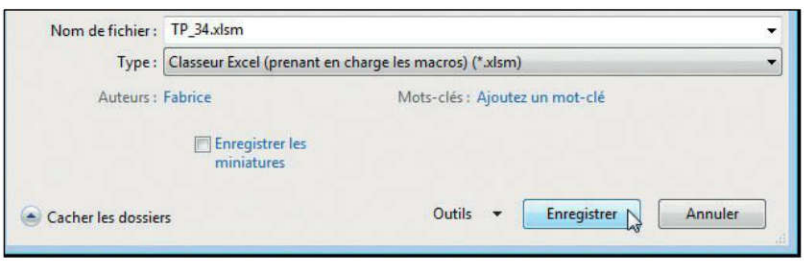


8 La feuille de calcul est triée selon les nouveaux critères. Cliquez sur le bouton d'arrêt de l'enregistrement, en bas à gauche de la barre d'état : c'est l'autre façon d'arrêter un enregistrement. Mieux vaut toujours enregistrer un classeur qui contient une macro avant de la tester. Si celle-ci contient des erreurs, vous risqueriez en effet de perdre des données lors de son exécution. Si tel est le cas, fermez le classeur sans l'enregistrer, puis rouvrez-le, ce qui restaurera les données d'origine. Cliquez sur le bouton *Enregistrer*, dans la barre d'outils d'accès rapide.



9 Dans la boîte de dialogue qui s'affiche signalant que si vous enregistrez le classeur sous ce format, la macro sera perdue, cliquez sur *Non*. Dans la boîte de dialogue *Enregistrer sous* qui s'ouvre, sélectionnez le type de fichier *Classeur Excel (prenant en charge les macros (*.xslm)*, puis cliquez sur *Enregistrer*.

Pour vous prémunir contre les virus susceptibles d'être présents dans une macro, il est bon de savoir que seul un fichier dont l'extension de nom de fichier est .xlsm peut contenir des macros.



10 Appuyez sur *Ctrl+Maj+A* : la liste est classée par région, puis par chiffre d'affaires. Appuyez sur *Ctrl+Maj+B* : la liste est triée uniquement par chiffre d'affaire. Vous pouvez ainsi basculer très rapidement entre les deux vues, plutôt que d'avoir à utiliser la boîte de dialogue Tri. Enregistrez et fermez le fichier TP_34.xlsm.

	A	B	C
1	Nom	Région	Chiffre d'affaires
2	Yabisda	Centre	178000
3	N'Gyen	Centre	120000
4	Dupont	Est	80000
5	Fougerolles	Est	60000
6	Jasmin	Nord	135000
7	Dufour	Nord	120000
8	Crespin	Ouest	115000
9	Dariseau	Ouest	99500
10	Lavillois	Sud	165000
11	Parpaillou	Sud	112000
12	Estaque	Sud	107000

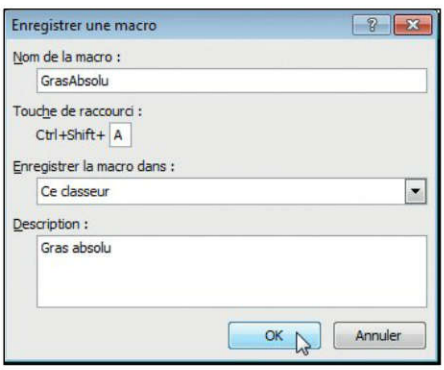
Copyright © 2012 Dunod.

TP 35 Macro absolue ou relative

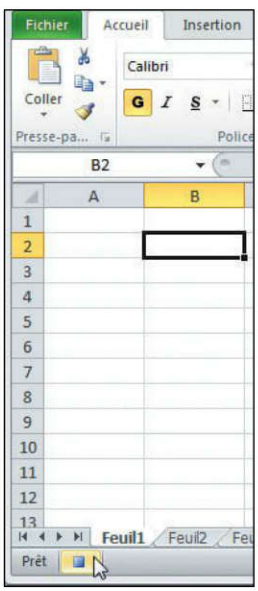
Le résultat d'une macro peut grandement différer selon qu'elle soit relative ou absolue. Ce TP va présenter les distinctions et particularités.

Prérequis : TP 33 et 34
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Ouvrez un classeur vierge et enregistrez-le comme classeur prenant en charge les macros sous le nom TP_35.xlsm.
Cliquez dans le groupe *Code* de l'onglet *Développeur* sur *Enregistrer une macro*. Nommez-le **GrasAbsolu**, comme raccourci **Ctrl+Maj+A** et comme description **Gras absolu**.

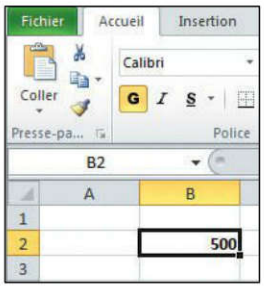


2 Cliquez sur la cellule *B2*. Dans l'onglet *Accueil*, groupe *Police*, cliquez sur le bouton *Gras*, puis arrêtez l'enregistrement (par le bouton de la barre des tâches ou celui de l'onglet *Développeur*).



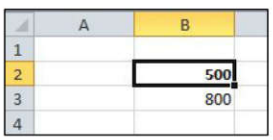
Copyright © 2012 Dunod.

3 Saisissez **500** dans la cellule *B2*. Le contenu apparaît en gras, suite à l'opération précédente. Sélectionnez la cellule à nouveau puis cliquez sur *Gras* dans l'onglet *Accueil* pour supprimer la mise en forme. Appuyez sur *Ctrl+Maj+A* : la cellule revient en gras.

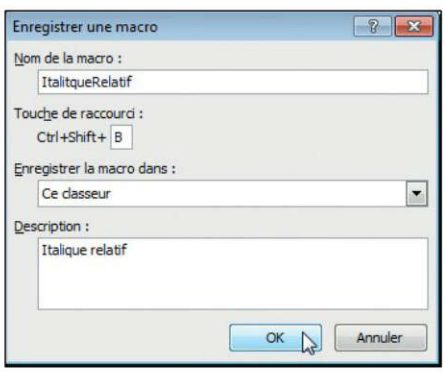


4 Saisissez **800** dans la cellule *B3*. Appuyez sur *Ctrl+Maj+A* : il ne se passe rien, si ce n'est que le curseur sélectionne à nouveau la cellule *B2*. Vous avez créé ici une macro absolue.

En sélectionnant une zone ou une cellule précise **après** avoir lancé l'enregistrement, la macro appliquera vos instructions uniquement à ou à partir de cette cellule ou plage précise, sans modification possible. Procédez ainsi lorsque vous voulez absolument que la macro affecte une cellule ou une plage précise.



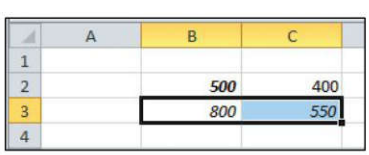
5 Cliquez dans le groupe *Code* de l'onglet *Développeur* sur *Enregistrer une macro*. Donnez-lui comme nom **ItaliqueRelatif**, comme raccourci **Ctrl+Maj+B** et comme description **Italique relatif**.



6 Saisissez **400** dans la cellule *C3*, **550** dans la cellule *C4*. Sélectionnez la cellule *C3*, puis appuyez sur *Ctrl+Maj+B* : la cellule passe en italique. Sélectionnez la plage *B4:C4*, puis appuyez sur *Ctrl+Maj+B* : la plage passe en italique. La macro s'applique à toute cellule ou plage de cellules sélectionnée.

En sélectionnant comme vous l'avez fait ici une cellule ou une plage avant de lancer l'enregistrement, le processus de sélection n'est pas enregistré dans la macro. Celle-ci s'exécute sur ou à partir de la zone active : c'est une macro relative. Les macros relatives sont souvent plus polyvalentes.

Enregistrez et fermez le classeur TP_35.xlsm.



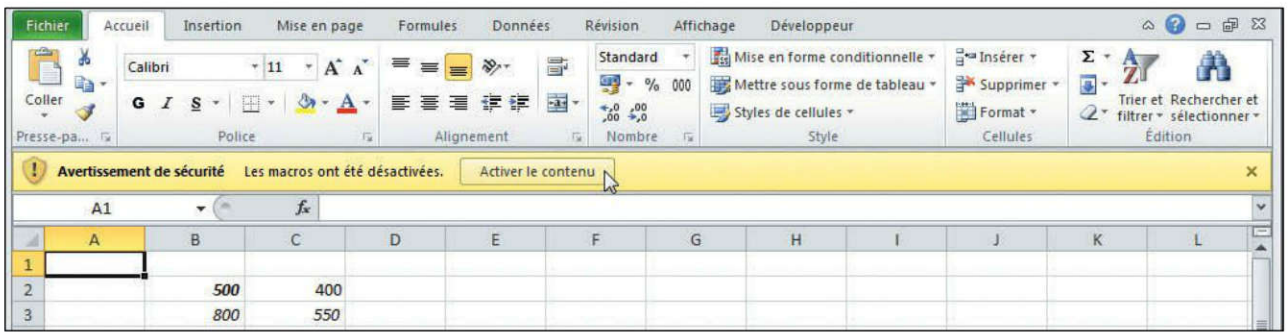
Copyright © 2012 Dunod.

TP 36 Modification des propriétés d'une macro

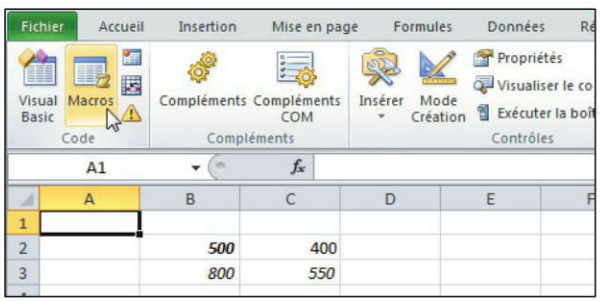
Après avoir enregistré ou créé une macro, vous pourriez vouloir modifier ses propriétés fondamentales, comme le nom, le raccourci ou la description (mais pas encore le code). C'est ce qu'explique ce TP.

Prérequis : TP 33 et 34
Fichier employé : TP_36_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_36_début.xlsm. Puisqu'il s'agit d'un classeur renfermant des macros et qu'il dépend des réglages de sécurité que nous avons retenu, un bandeau d'avertissement signale un contenu actif. Cliquez dans le bandeau sur *Activer le contenu*, puis enregistrez le fichier sous le nom TP_36.xlsm.

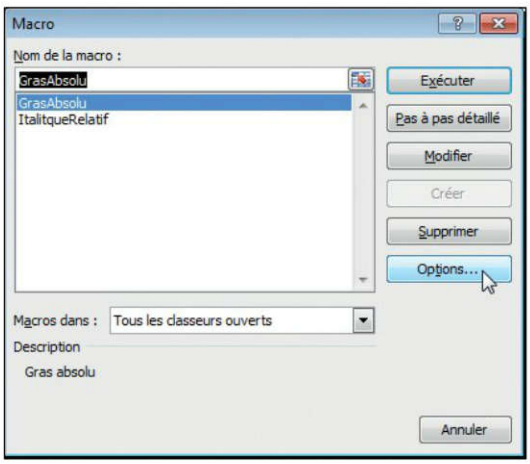


2 Cliquez sur l'onglet *Développeur*, puis cliquez sur le bouton *Macro*.

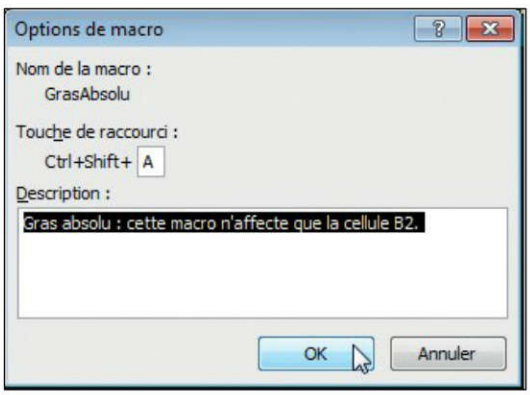


Copyright © 2012 Dunod.

3 La boîte de dialogue, *Macro* permet d'effectuer de nombreuses opérations, dont entre autres l'exécution d'une macro. Cliquez ici sur *Options*.



4 Dans la zone *Description*, complétez le texte avec : **Gras absolu : cette macro n'affecte que la cellule B2**. Vous pourriez également modifier le nom et le raccourci. Cliquez sur *OK*, puis sur *Annuler*.
Enregistrez et fermez le classeur TP_36.xlsm.



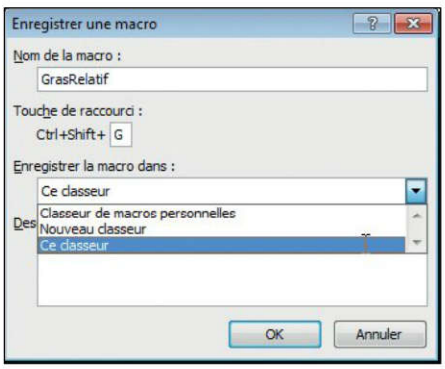
Copyright © 2012 Dunod.

TP 37 Emplacement de stockage d'une macro

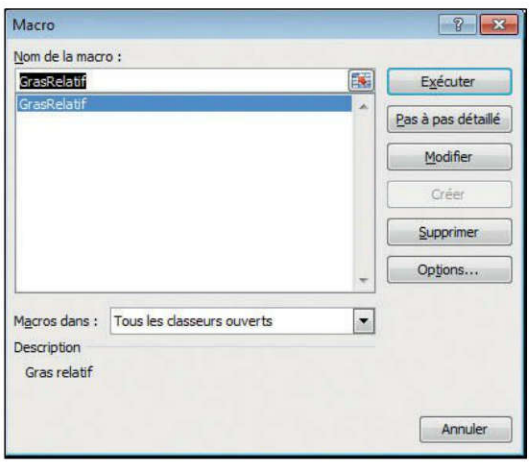
Il est possible de stocker une macro (enregistrée ou créée manuellement) en différents endroits, selon le but et la portée de celle-ci. Ce TP présente les différences existantes entre les multiples lieux de stockage. Les macros employées n'ont guère d'intérêt, ce sont les conséquences du choix du lieu de stockage qui nous intéressent.

Prérequis : TP 33 à 36
Fichier employé : TP_36_début.xlsm
Temps de réalisation : 10 minutes

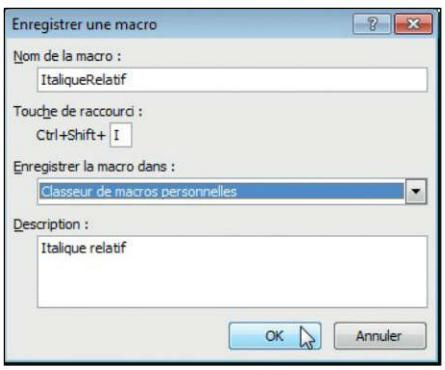
1 Ouvrez un classeur vierge et enregistrez-le sous le nom TP_37.xlsm.
Dans l'onglet *Développeur*, cliquez sur le bouton *Enregistrer une macro*. Dans la boîte de dialogue *Enregistrer une macro*, saisissez comme nom **GrasRelatif**, comme raccourci **Ctrl+Maj+G**, comme description **Gras relatif** puis cliquez sur la flèche déroulante sous *Enregistrer cette macro dans*.



2 La première possibilité, *Ce classeur*, enregistre la macro dans le classeur actif. Sélectionnez ce choix et cliquez sur *OK*. Cliquez ensuite sur *Accueil*, puis sur le bouton *Gras* du groupe *Police* et arrêtez la macro. Cliquez ensuite sur l'onglet *Développeur*, puis sur *Macros* : vous voyez dans la liste une seule macro disponible, *GrasRelatif*.

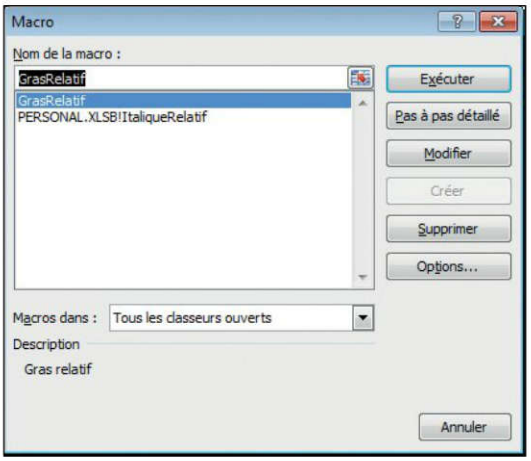


3 Cliquez sur *Annuler*. Dans l'onglet *Développeur*, cliquez sur le bouton *Enregistrer une macro*. Dans la boîte de dialogue *Macro*, saisissez comme nom **ItaliqueRelatif**, comme raccourci **Ctrl+Maj+I**, comme description **Italique relatif** puis cliquez sur la flèche déroulante sous *Enregistrer cette macro dans*, en choisissant cette fois *Classeur de macros personnelles*. Cliquez sur *OK*.

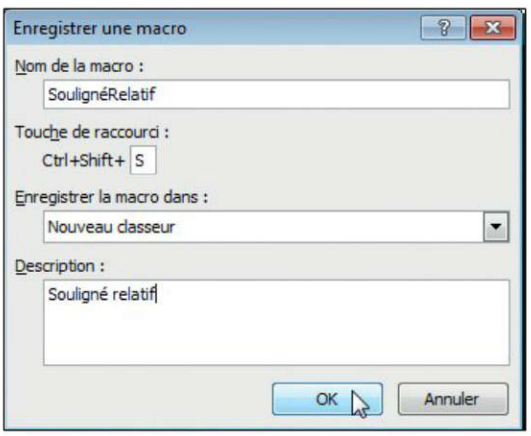


Copyright © 2012 Dunod.

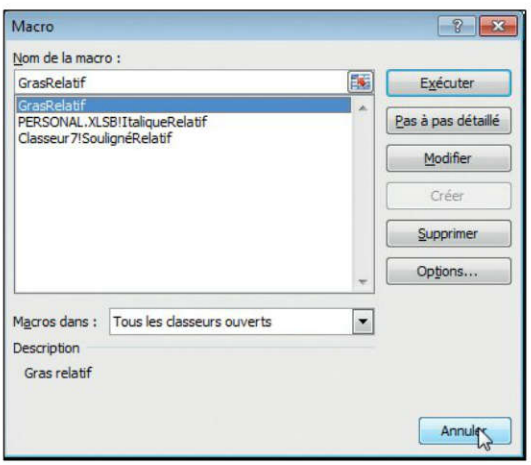
4 Cliquez sur *Accueil*, puis sur le bouton *Italique* du groupe *Police* et arrêtez la macro. Cliquez ensuite sur l'onglet *Développeur*, puis sur *Macros* : vous voyez dans la liste que deux macros sont disponibles : GrasRelatif et ItaliqueRelatif. Remarquez que la seconde macro est identifiée sous le nom *PERSONAL.XLSB!ItaliqueRelatif*.



5 Dans l'onglet *Développeur*, cliquez sur le bouton *Enregistrer une macro*. Dans la boîte de dialogue, saisissez comme nom **SoulignéRelatif**, comme raccourci **Ctrl+Maj+S**, comme description **Souligné relatif** puis cliquez sur la flèche déroulante sous *Enregistrer cette macro dans*, en choisissant cette fois *Nouveau classeur*. Cliquez sur *OK*.

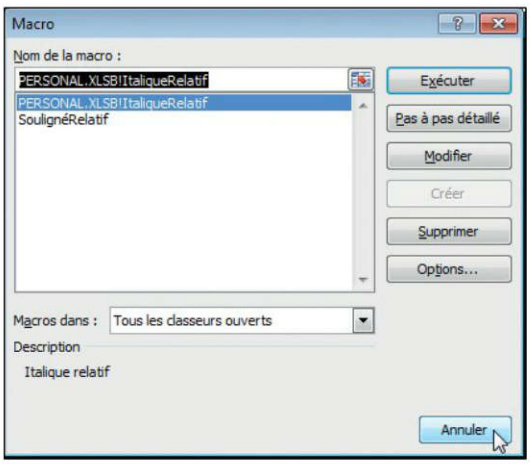


6 Cliquez sur *Accueil*, puis sur le bouton *Souligné* du groupe *Police* et arrêtez la macro. Cliquez ensuite sur l'onglet *Développeur*, puis sur *Macros* : vous voyez dans la liste que trois macros sont disponibles : GrasRelatif, ItaliqueRelatif et SoulignéRelatif. Remarquez que la nouvelle macro est identifiée sous le nom *ClasseurN!SoulignéRelatif*, où *N* est un chiffre dépendant du nombre de nouveaux classeurs que vous avez ouverts jusqu'ici avec cette instance d'Excel (7 dans le cas de la capture d'écran).



Copyright © 2012 Dunod.

7 Enregistrez et fermez le classeur TP_37.xslm. Vous voyez apparaître le nouveau classeur. Cliquez sur l'onglet *Développeur*, puis sur *Macros* : vous voyez dans la liste que seules deux macros sont disponibles : ItaliqueRelatif et SoulignéRelatif. Cette dernière n'est plus précédée d'un nom de classeur. Cliquez sur *Annuler*.



Ce TP démontre plusieurs choses :
- Une macro n'est disponible que dans le classeur qui la contient ou dans un classeur également ouvert sans être le classeur actif.
- En choisissant *Classeur de macros personnelles*, Excel crée, s'il n'existe pas déjà, un classeur particulier masqué nommé *Personal.xlsb*. Ce classeur est situé avec Windows Vista et Windows 7 dans le dossier *C:\Users\nom_utilisateur\AppData\Local\Microsoft\Excel\XLStart*, et avec Windows XP, dans le dossier *C:\Documents and Settings\nom_utilisateur\Application Data\Microsoft\Excel\XLStart*. Les classeurs situés dans ce dossier XLStart sont automatiquement ouverts à chaque démarrage d'Excel. De ce fait, Personnal.xlsb est toujours chargé et les macros qu'il contient sont toujours disponibles dans n'importe quel classeur Excel.

Fermez le classeur actif sans l'enregistrer.

Copyright © 2012 Dunod.

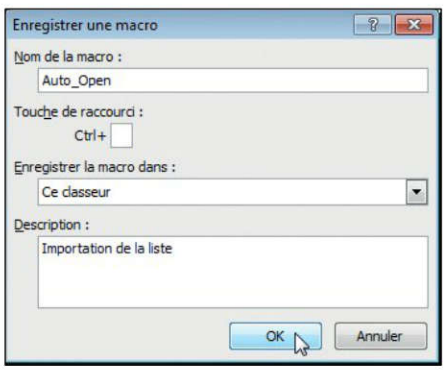
TP 38 Exécuter une macro lors de l'ouverture du classeur

Il est fréquent de vouloir exécuter automatiquement une procédure à l'ouverture d'un classeur. Souvenez-vous du TP 34 : nous avions créé deux macros pour effectuer le tri d'une liste de commerciaux. Imaginons que cette liste soit initialement un fichier texte qu'il faille importer dans Excel. En stockant toujours le fichier actualisé au même endroit, nous pourrions automatiquement l'importer à l'ouverture du classeur. C'est ce que nous allons faire ici.

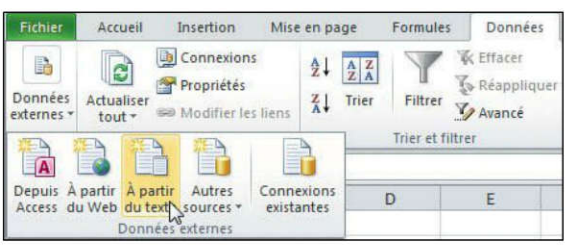
Prérequis : TP 33 et 34
Fichier employé : TP_38_début.xlsm et TP_38.txt
Temps de réalisation : 10 minutes

1 Ouvrez le classeur TP_38_début.xlsm, activez le contenu et enregistrez-le comme classeur prenant en charge les macros sous le nom TP_38.xlsm.

Dans l'onglet *Développeur*, dans le groupe *Code*, cliquez sur *Enregistrer une macro*. Dans la zone *Nom de la macro*, tapez **Auto_Open**. Ne saisissez pas de touche de raccourci, laissez le paramètre *Enregistrer la macro dans* à sa valeur par défaut, *Ce classeur* et saisissez comme description **Importation de la liste**. Cliquez sur *OK*.

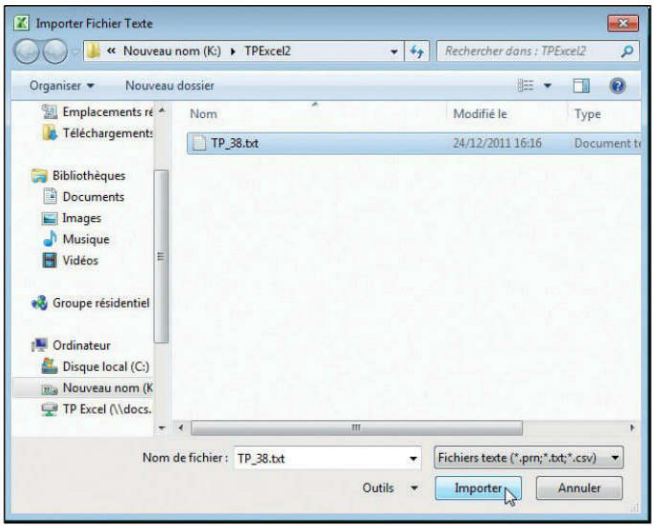


2 Dans le groupe *Données*, cliquez sur *Données externes*, puis sur *À partir d'un fichier texte*.

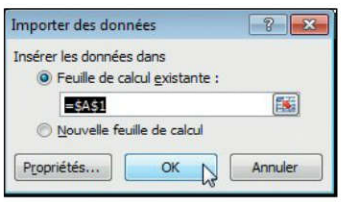


Copyright © 2012 Dunod.

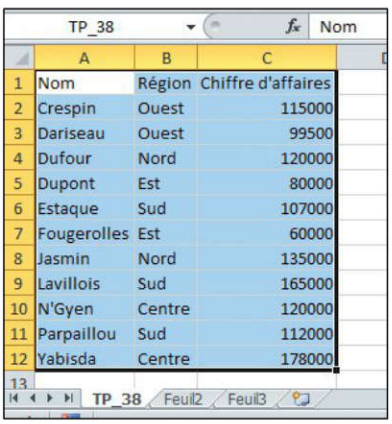
3 Dans la boîte de dialogue *Importer Fichier Texte*, naviguez si nécessaire jusqu'à votre dossier d'exercices et sélectionnez le fichier TP_38.txt. Cliquez sur *Importer*.



4 Cliquez immédiatement sur *Terminer* dans la boîte de dialogue (les paramètres par défaut sont idéals ici). Dans la boîte de dialogue suivante, choisissez d'importer à partir de la cellule $A$1. Cliquez sur *OK*.



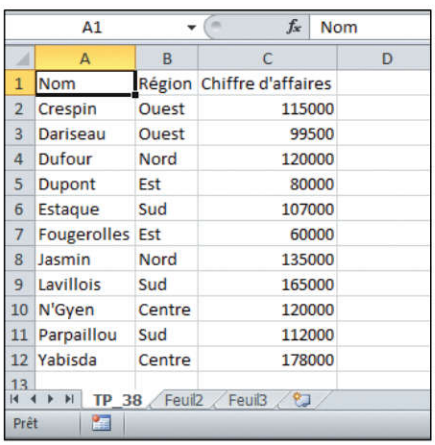
5 La liste des commerciaux et de leurs chiffres d'affaires apparaît dans la feuille de calcul. Arrêtez l'enregistrement de la macro.



6 Sélectionnez l'ensemble de la zone de la liste, puis appuyez sur *Suppr* pour la supprimer. Enregistrez puis fermez le classeur TP_38.xlsm, puis ouvrez-le à nouveau. Activez le contenu : le classeur apparaît avec la liste. La macro Auto_Open s'est automatiquement exécutée à l'ouverture du classeur. Vous pouvez utiliser ses deux autres macros pour trier cette liste à votre guise.

Si un classeur dans lequel vous enregistrez une macro Auto_Open contient déjà une procédure VBA dans son événement *Open*, cette dernière est prioritaire. En outre, une macro Auto_Open s'exécute avant l'ouverture de tout autre classeur.

Fermez le classeur TP_38.xlsm. Répondez *non* au message qui signale que le classeur a été modifié et propose de l'enregistrer.



Copyright © 2012 Dunod.

TP 39 Édition d'une macro

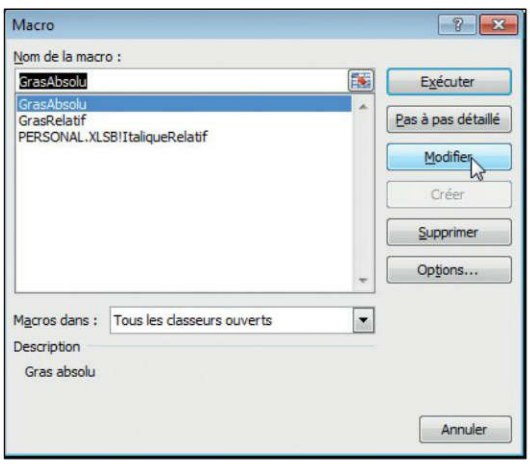
Il est toujours intéressant d'examiner une macro enregistrée pour éventuellement la modifier, ou simplement pouvoir reproduire telle ou telle commande dans une macro que vous souhaitez rédiger. Jusque-là, nous avons toujours travaillé uniquement dans l'interface d'Excel. L'édition d'une macro va vous faire découvrir un nouveau composant d'Excel, l'éditeur Visual Basic, que nous allons employer dans tous les TP suivants.

Prérequis : TP 33 et 34
Fichier employé : TP_39_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP39_début.xlsm, choisissez d'en activer le contenu, puis enregistrez-le sous le nom TP_39.xlsm. Choisissez dans le groupe *Code* de l'onglet *Développeur* l'option *Macros*. Vous pouvez également utiliser le raccourci *Alt+F8*.

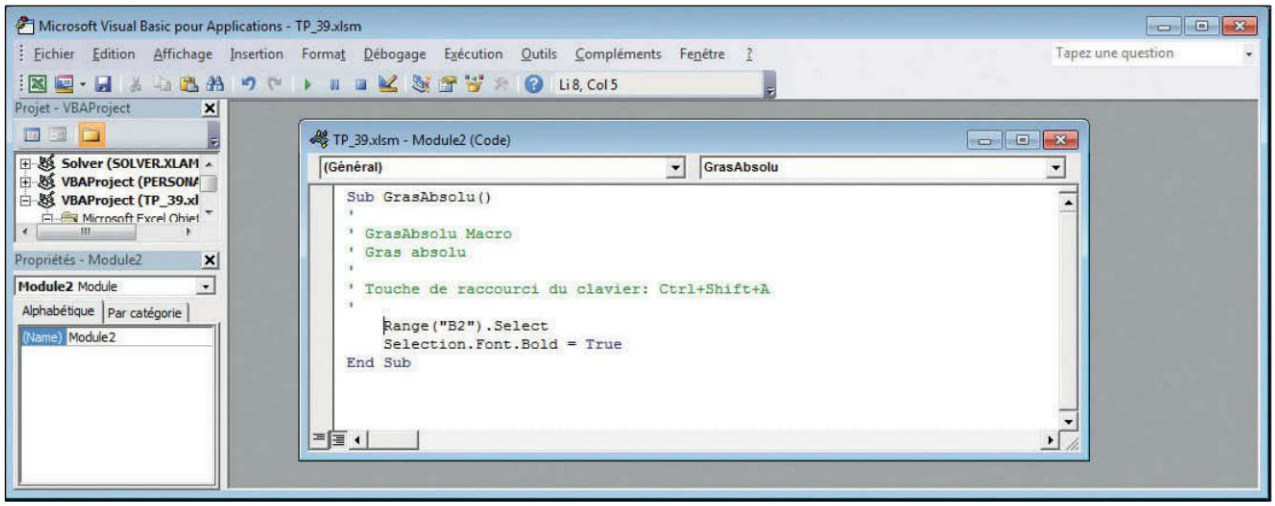


2 Dans la boîte de dialogue *Macro*, sélectionnez la macro *GrasAbsolu* dans la liste *Nom de la macro*, puis cliquez sur *Modifier*.



Copyright © 2012 Dunod.

3 La macro sélectionnée s'ouvre dans l'Éditeur Visual Basic.Vous pouvez afficher directement l'Éditeur Visual Basic en cliquant dans l'onglet *Développeur* sur *Visual Basic* ou en appuyant sur les touches *Alt+F11*.
La macro apparaît dans le volet principal de l'éditeur, la fenêtre *Code*. C'est cette fenêtre que vous utiliserez le plus souvent dans l'éditeur. Nous examinerons certains des autres volets dans le TP suivant.



4 Examinez le code de la macro. Toute macro VBA est composée d'instructions. Celles-ci sont en anglais, contrairement aux fonctions intégrées employées depuis une feuille de calcul.
Une macro enregistrée débute par l'instruction `Sub`, suivi du nom de la macro, lui-même suivi de deux parenthèses, et se termine par `End Sub`. `Sub` qui est l'abréviation de sous-routine.
Viennent ensuite s'ajouter quelques lignes de commentaire (nous y reviendrons), qui reprennent le nom de la macro, sa description et le raccourci clavier. Automatiquement insérées par Excel, elles apparaissent de couleur verte et débutent par une apostrophe.
La macro sélectionne la plage concernée par la totalité des cellules (`Range("B2").Select`), puis applique à la propriété gras (`bold`) de l'objet `Font` sélectionné la valeur vraie (`Selection.Font.Bold = True`), puis s'achève.

```
Sub GrasAbsolu()
'
' GrasAbsolu Macro
' Gras absolu
'
' Touche de raccourci du clavier: Ctrl+Shift+A
'
    Range("B2").Select
    Selection.Font.Bold = True
End Sub
```

5 Cette macro était une macro absolue. Quelle serait la différence pour une macro relative ? Fermez l'Éditeur Visual Basic : vous revenez à Excel. Dans l'onglet *Développeur*, cliquez sur *Macros*. Dans la boîte de dialogue *Macro*, sélectionnez la macro *GrasRelatif*, puis cliquez sur *Modifier*. Le code de la macro GrasRelatif s'affiche dans l'Éditeur Visual Basic. La seule différence avec la macro GrasAbsolu est l'absence de l'instruction de sélection : la macro s'applique à la sélection active, quelle qu'elle soit. Vous venez de découvrir la seule différence réelle entre le code d'une macro absolue et celui d'une macro relative, ce qui vous permet de convertir facilement l'une l'autre et réciproquement.

```
Sub GrasRelatif()
'
' GrasRelatif Macro
' Gras relatif
'
' Touche de raccourci du clavier: Ctrl+Shift+G
'
    Selection.Font.Bold = True
End Sub
```

Si vous ne poursuivez pas immédiatement avec le TP suivant, fermez l'Éditeur Visual Basic, puis fermez Excel (inutile d'enregistrer le fichier).

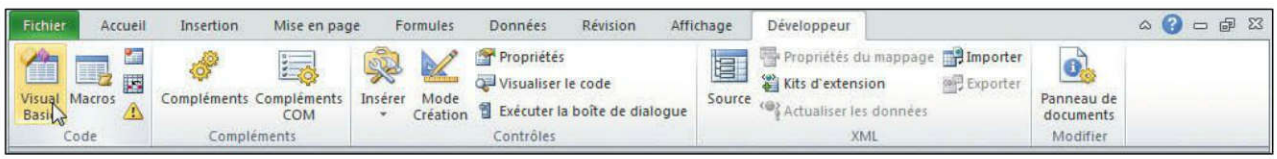
Copyright © 2012 Dunod.

TP 40 Éditeur Visual Basic : fenêtre Projet

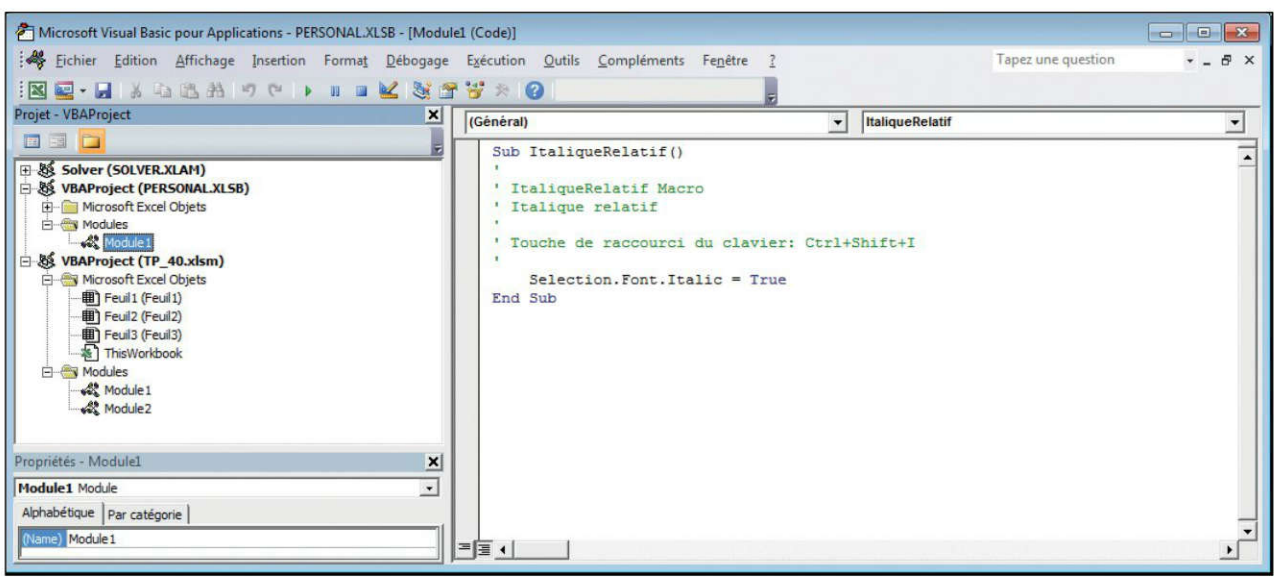
Il est utile de se familiariser un peu avec le second composant le plus important de l'interface de l'Éditeur Visual Basic, la fenêtre Projet. Vous apprendrez dans ce TP à parcourir les macros disponibles dans les différents modules, ainsi qu'à déplacer une macro d'un module à l'autre.

Prérequis : TP 33 à 39
Fichier employé : TP_39.xslm ou TP_40_début.xlsm
Temps de réalisation : 5 minutes

1 Si le classeur TP_39 n'est pas ouvert, ouvrez le classeur TP_40_début.xlsm et activez le contenu. Enregistrez le classeur actif sous le nom TP_40.xlsm.
Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Visual Basic* ou appuyez sur *Alt+F11*.

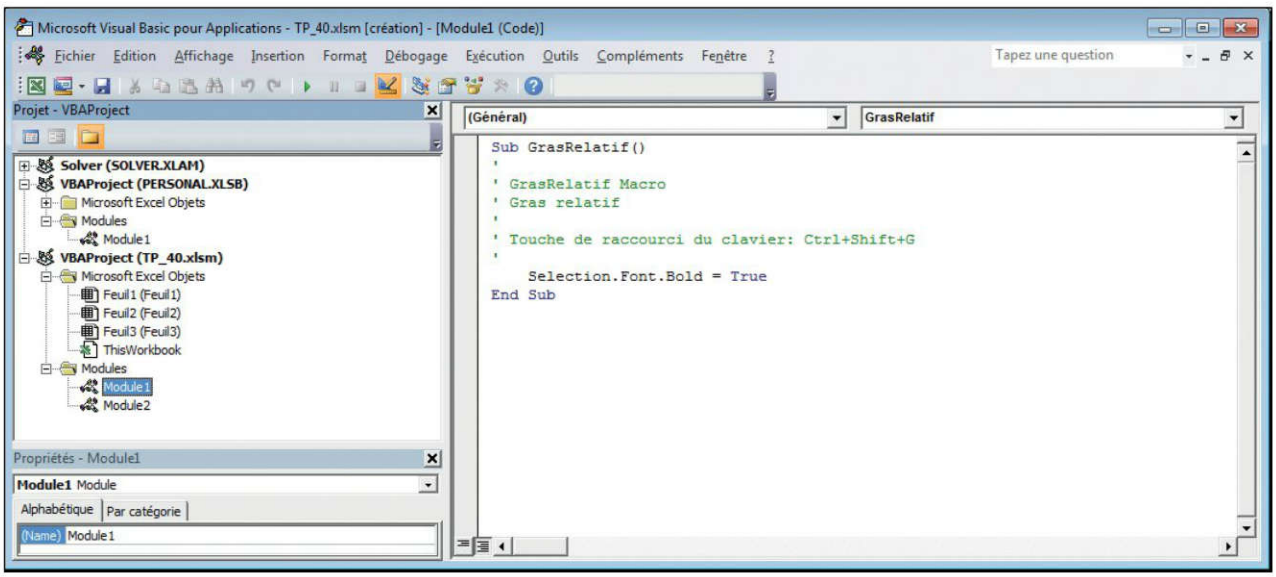


2 Dans la fenêtre de l'Éditeur Visual Basic, redimensionnez la fenêtre *Projet* (en haut à gauche), de façon à en voir tout le contenu. On y distingue trois groupes : *Solver (SOLVER.XLAM)*, le composant Solveur que vous avez installé au début de ce livre, *VBAPROJECT (PERSONAL.XLSB)*, votre classeur de macros personnelles, et *VBAPROJECT (TP_40.XLSM)*, le classeur actif. Cliquez sur le signe + à côté de *VBAPROJECT (PERSONAL.XLSB)* pour développer l'arborescence, puis sur le signe + à côté de *Modules*, puis double-cliquez sur *Module 1* : vous voyez apparaître dans la fenêtre *Code* la macro ItaliqueRelatif, que vous aviez stockée dans votre classeur de macros personnelles. En effet, les macros sont toujours stockées dans des modules, associés aux classeurs correspondants.

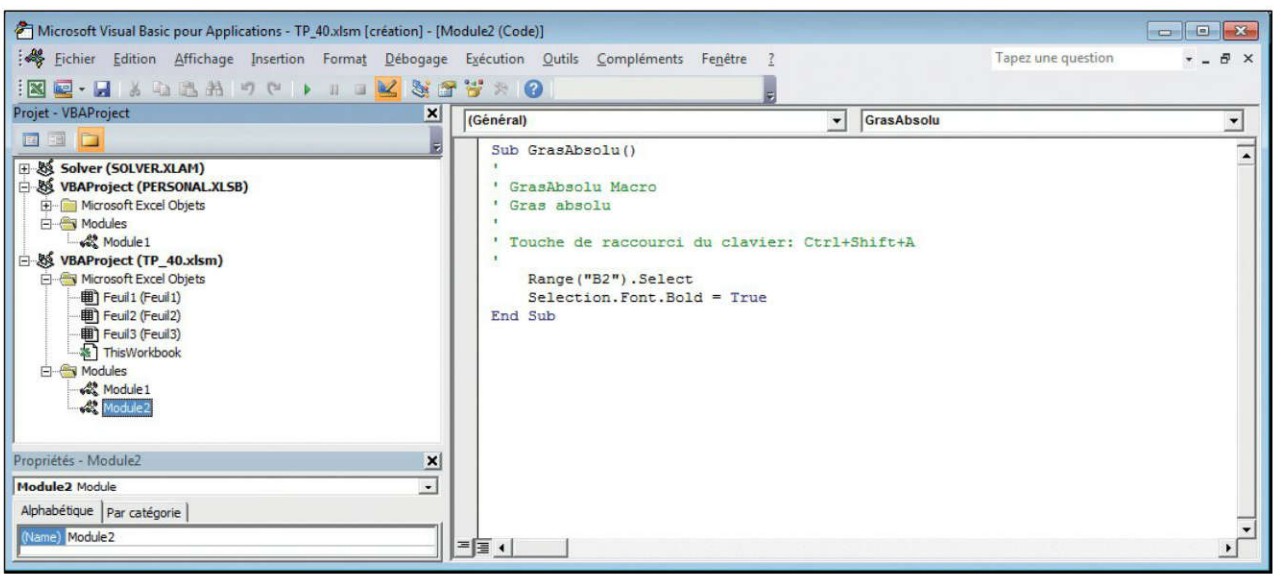


Copyright © 2012 Dunod.

3 Développez, si nécessaire, l'arborescence *VBAPROJECT (TP_40.XLSM)*, puis celle de ses modules, et double-cliquez sur *Module1* : vous voyez apparaître la macro GrasRelatif.

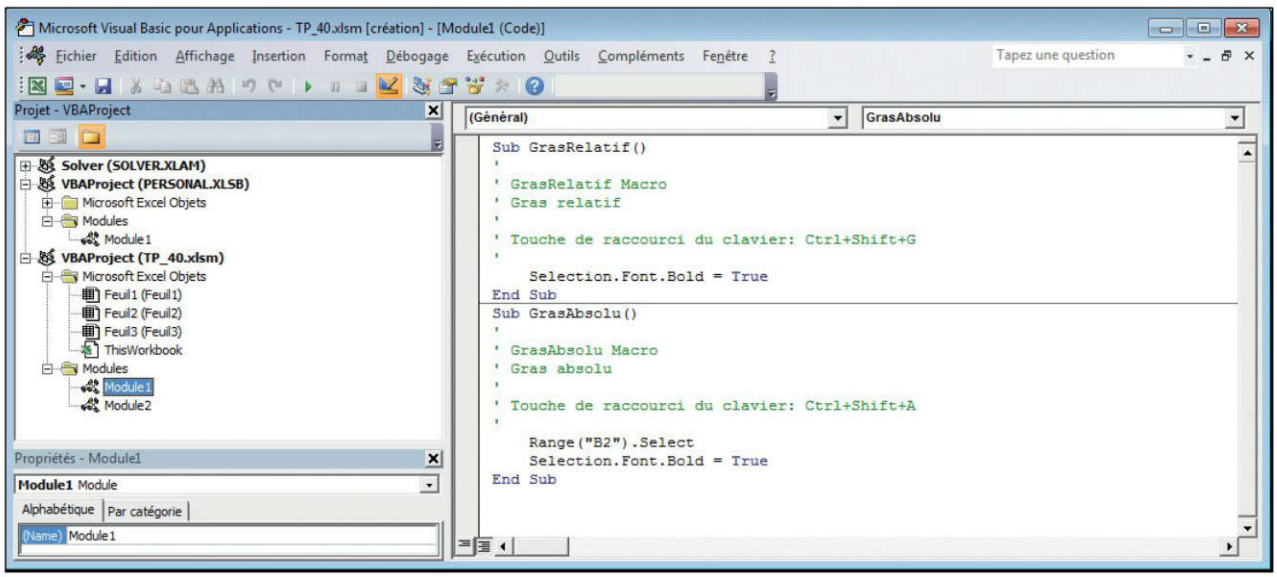


4 Double-cliquez sur *Module2* : vous voyez dans la fenêtre *Code* la macro *GrasAbsolu*.

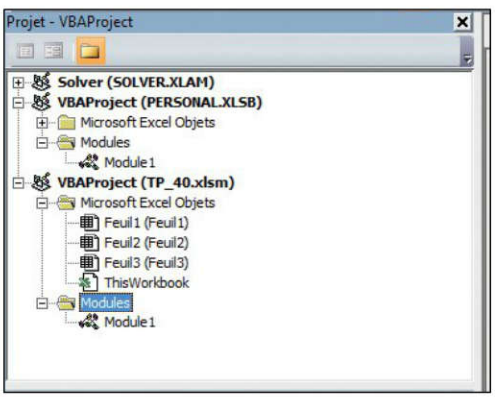


Copyright © 2012 Dunod.

5 Sélectionnez la totalité du code de la macro GrasAbsolu. Appuyez sur *Ctrl+C*, double-cliquez sur *Module1*, placez-vous en dessous de l'instruction `End Sub` de la macro GrasRelatif, puis appuyez sur *Ctrl+V* : la macro GrasAbsolu se trouve sous la macro précédente, séparée par un trait horizontal.



6 Effectuez un clic-droit sur *Module2* et choisissez *Supprimer Module2*. Une boîte de dialogue vous propose de l'exporter avant suppression. Cliquez sur *Non* : Excel supprime le module 2. Ce dernier était superflu puisque vous avez copié la macro qu'il contenait dans *Module1*. Lorsque vous enregistrez des macros, cela crée souvent des modules supplémentaires, ce qui complique ensuite un peu la recherche d'une macro : mieux vaut remettre un peu d'ordre, comme ici, et rassembler les macros d'un même classeur dans un seul et unique module, sauf pour des besoins particuliers.



En effectuant un clic-droit sur *Module1*, vous pourriez, en choisissant *Insertion* > *Module,* ajouter un nouveau module à votre projet. Enregistrez et fermez le classeur TP_40 si vous ne poursuivez pas immédiatement avec le prochain TP.

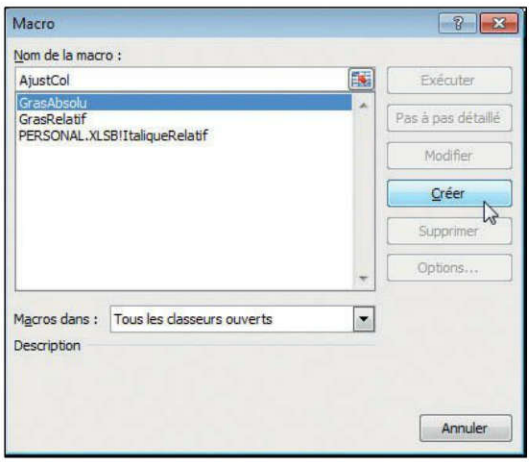
Copyright © 2012 Dunod.

Rédaction d'une macro (I)

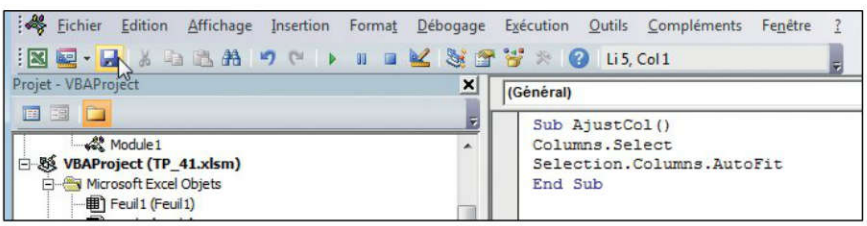
Nous avons jusqu'à présent examiné des macros enregistrées. Certes, enregistrer une macro est toujours un excellent point de départ, mais vous pourriez tout aussi bien vous fonder sur une macro préexistante.

Prérequis : TP 33 à 40
Fichier employé : TP_40.xslm ou TP_41_début.xlsm
Temps de réalisation : 5 minutes

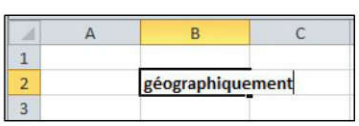
1 Si le classeur TP_40 n'est pas ouvert, ouvrez le classeur TP_41_début.xlsm et activez le contenu. Enregistrez le classeur actif sous le nom TP_41.xlsm.
Cliquez dans le groupe *Code* de l'onglet *Développeur* sur *Macros*, ou appuyez sur *Alt+F8*. Dans la zone *Nom de la macro*, nommez cette procédure **AjustCol**, puis cliquez sur *Créer*.



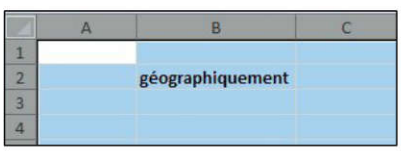
2 L'éditeur de macro s'ouvre, affichant la nouvelle procédure. Vous voyez dans la fenêtre *Code* l'instruction `Sub AjustCol` immédiatement suivie sur la ligne suivante de l'instruction `End Sub`. Saisissez `Columns.Select` entre ces deux instructions, appuyez sur *Entrée*, puis saisissez `Selection.Columns.Autofit`. Enregistrez ensuite le classeur : il vaut toujours mieux enregistrer un classeur avant d'exécuter une macro que l'on a créée, afin d'éviter toute éventuelle perte de données. Fermez l'Éditeur Visual Basic.



3 Saisissez dans la cellule *B2* le mot **géographiquement** : ce texte excède la largeur de la cellule.



4 Dans l'onglet *Développeur*, cliquez sur *Macros*, sélectionnez la macro AjustCol et cliquez sur *Exécuter* : la largeur de la colonne s'ajuste automatiquement.
Enregistrez et fermez le classeur TP_41 si vous ne poursuivez pas immédiatement avec le prochain TP.



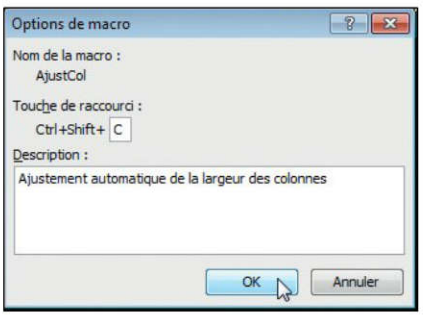
Copyright © 2012 Dunod.

TP 42 Rédaction d'une macro (II)

La macro rédigée lors du TP précédent fonctionne, mais est dépourvue de tout commentaire explicatif et n'utilise aucune touche de raccourci. Nous allons y remédier dans ce TP.

Prérequis : TP 33 à 41
Fichier employé : TP_41.xlsm ou TP_42_début.xlsm
Temps de réalisation : 5 minutes

1 Si le classeur TP_41 n'est pas ouvert, ouvrez le classeur TP_42_début.xlsm et activez le contenu. Enregistrez le classeur actif sous le nom TP_42.xlsm.
Cliquez dans le groupe *Code* de l'onglet *Développeur* sur *Macros*, ou appuyez sur *Alt+F8*. Dans la zone *Nom de la macro*, sélectionnez la macro AjustCol, puis cliquez sur *Options*. Saisissez comme raccourci **Maj+C**, puis saisissez comme description **Ajustement automatique de la largeur des colonnes**. Cliquez sur *OK*. Cette démarche vous est familière puisque vous l'aviez déjà effectuée dans le TP 36.



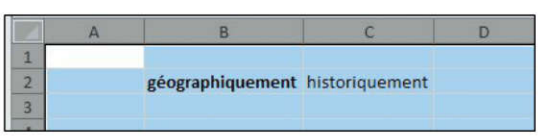
2 Cliquez dans le groupe *Code* de l'onglet *Développeur* sur *Macros*, ou appuyez sur *Alt+F8*. Dans la zone *Nom de la macro*, sélectionnez la macro *AjustCol*, puis cliquez sur *Modifier*. La macro s'ouvre dans l'Éditeur de macro. Sous l'instruction `Sub AjustCol()`, appuyez sur *Entrée* pour créer une nouvelle ligne, saisissez une **apostrophe** suivie d'un **espace** et du texte **Ajustement automatique de la largeur des colonnes**. Dès que vous déplacez le curseur sur la ligne suivante, la ligne ajoutée devient verte : c'est un commentaire, utile à la compréhension d'une macro mais non exploitée comme code au sens strict.

```
Sub AjustCol()
' Ajustement automatique de la largeur des colonnes
Columns.Select
Selection.Columns.AutoFit
End Sub
```

3 Placez-vous à la fin de l'instruction `Select.Columns`. Ajoutez suffisamment d'espaces pour que le curseur se trouve sous le *l* du mot *la* du commentaire précédent, puis saisissez **' Sélection de toutes les colonnes**. Posez votre curseur à la fin de l'instruction suivante, ajoutez des espaces de la même façon et saisissez **' Activation de l'ajustement automatique**.
Vous pouvez ajouter des commentaires après une ligne d'instructions pour en expliciter le sens.

```
Sub AjustCol()
' Ajustement automatique de la largeur des colonnes
Columns.Select              ' Sélection de toutes les colonnes
Selection.Columns.AutoFit   ' Activation de l'ajustement automatique
End Sub
```

4 Dans l'éditeur Visual Basic, cliquez sur *Enregistrer*, puis fermez l'éditeur. Dans la feuille de calcul, saisissez dans la cellule *C2* le mot **historiquement**, puis appuyez sur *Ctrl+Maj+C* : les colonnes s'ajustent automatiquement.
Enregistrez et fermez le classeur TP_42.xlsm.



Copyright © 2012 Dunod.

TP 43 Rédaction d'une fonction personnalisée

Outre une routine, vous pouvez écrire une fonction personnalisée, disponible ensuite dans n'importe quelle feuille de calcul du classeur qui la contient (ou dans tous les classeurs si vous avez enregistré la fonction dans votre Classeur de macros personnelles). Une routine débute par `Sub` et se termine par `End Sub`. Une fonction commence par `Function` et s'achève par `End Function`. La principale différence, hormis ce nom, est qu'une fonction renvoie à un résultat, contrairement à une routine. L'intérêt d'une fonction personnalisée est de réduire le temps de saisie nécessaire si elle doit être employée répétitivement, et qu'un copier-coller n'est pas facile à réaliser (la formule se trouve dans un autre classeur, par exemple). Nous allons concevoir ici une formule qui répartit des commerciaux en classes selon leur chiffre d'affaires.

Prérequis : TP 33 à 42
Fichier employé : TP_43_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_43_début.xlsm, activez le contenu et enregistrez-le sous le nom TP_43.xlsm.
Choisissez dans l'onglet *Développeur* l'icône *Visual Basic* ou appuyez sur *Alt+F11* pour ouvrir l'éditeur Visual Basic.
Dans la fenêtre Code, saisissez **Function TEST(a)**, puis appuyez sur *Entrée*. L'éditeur ajoute automatiquement une ligne vierge, puis en dessous l'instruction `End Function`.

```
Function Test(a)

End Function
```

2 Saisissez entre les deux instructions la (longue) consigne suivante :
`Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, IIf(a > 100000, 4, IIf(a > 80000, 5, 6)))))`
Remarquez que dans cette instruction, l'expression de formule *SI(condition, si vrai ;si faux)* devient *IIf(condition, si vrai, si faux)*. Comme cela a déjà été signalé, toutes les commandes sont en anglais.
Si vous commettez une faute de frappe, l'Éditeur Visual Basic vous le signalera. Vérifiez votre saisie. Faites attention en particulier au nombre de parenthèses fermantes.
Cette fonction effectue des tests conditionnels successifs sur la cellule référencée comme paramètre. Selon sa valeur, la fonction renvoie un nombre compris entre 1 et 6. Le nom de la fonction est TEST, tandis qu'entre parenthèses figure son unique paramètre : la cellule testée.

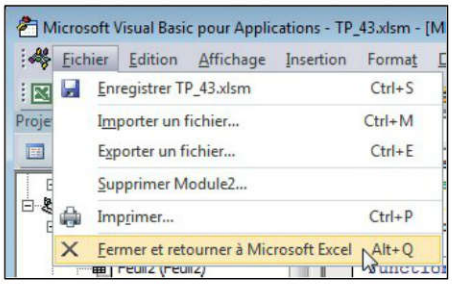
```
Function Test(a)
Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, IIf(a > 100000, 4, IIf(a > 80000, 5, 6))))
End Function
```

3 La ligne est trop longue pour être facilement lisible. Placez-vous juste avant le quatrième `Iif` (mais après l'espace qui suit la virgule), insérez un trait de soulignement et appuyez sur *Entrée*. La combinaison espace trait de soulignement (_) sert à indiquer la suite d'une ligne d'instruction.

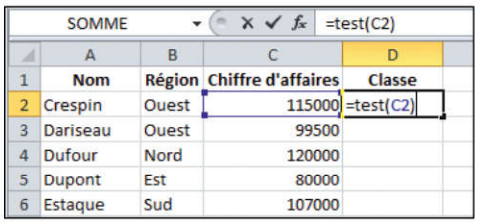
```
Function Test(a)
Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, _
IIf(a > 100000, 4, IIf(a > 80000, 5, 6)))))
End Function
```

Copyright © 2012 Dunod.

4 Dans l'Éditeur Visual Basic, cliquez sur *Enregistrer*, puis choisissez dans le menu *Fichier* l'option *Fermer et retourner à Microsoft Excel*.



5 Dans la cellule *D2*, saisissez **=TEST(C2)**.



	A	B	C	D
1	**Nom**	**Région**	**Chiffre d'affaires**	**Classe**
2	Crespin	Ouest	115000	=test(C2)
3	Dariseau	Ouest	99500	
4	Dufour	Nord	120000	
5	Dupont	Est	80000	
6	Estaque	Sud	107000	

6 Le chiffre 4 apparaît dans la cellule *D2*. Copiez la formule dans la plage *D3:D12*. Vous voyez apparaître dans les cellules de la plage les classes respectives des commerciaux.

	A	B	C	D
1	**Nom**	**Région**	**Chiffre d'affaires**	**Classe**
2	Crespin	Ouest	115000	4
3	Dariseau	Ouest	99500	5
4	Dufour	Nord	120000	4
5	Dupont	Est	80000	6
6	Estaque	Sud	107000	4
7	Fougerolles	Est	60000	6
8	Jasmin	Nord	135000	3
9	Lavillois	Sud	165000	2
10	N'Gyen	Centre	120000	4
11	Parpaillou	Sud	112000	4
12	Yabisda	Centre	178000	1
13				

Une fois une fonction créée dans l'éditeur de macros, vous pouvez l'employer exactement comme une des fonctions intégrées d'Excel, et vous épargner ainsi une saisie laborieuse.
Remarquez que cette fonction personnalisée a recours à un codage « en dur » des seuils. Vous pourriez facilement, en les transformant en paramètres, construire une fonction bien plus souple de la forme TEST(a, b, c, d, e, f), où b à f feraient référence aux valeurs seuils que vous souhaitez utiliser.
Enregistrez et fermez le classeur TP_43.xlsm.

Copyright © 2012 Dunod.

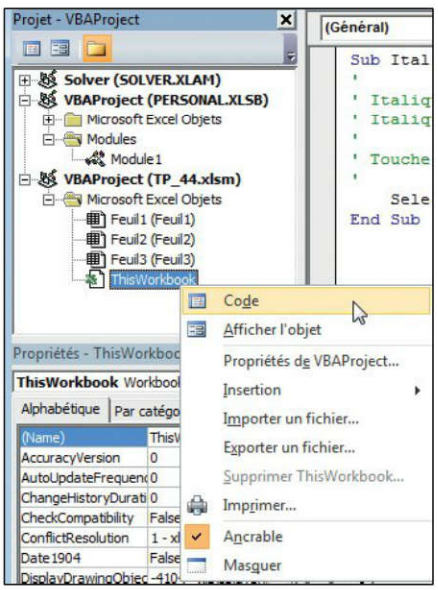
Création d'une procédure VBA pour un événement

Un des aspects les plus intéressants des procédures VBA est la possibilité d'écrire une macro pour un événement. Un événement est une situation ou une action précise à laquelle vous pouvez attacher du code : ce dernier s'exécutera dès que l'occurrence se produira. Toutefois, seuls certains événements relatifs à certains objets peuvent être associés à une macro. De façon générale, une procédure associée à un événement porte obligatoirement le nom Sub nomobjet_nomévénement. L'exemple ci-dessous utilise l'événement Open pour exécuter une macro lors de l'ouverture du classeur actif.

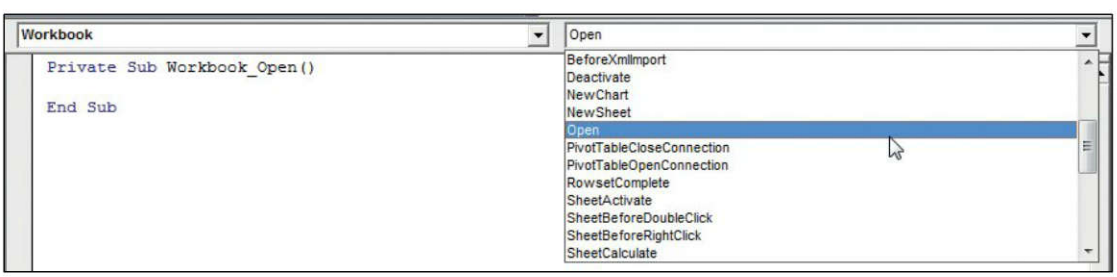
Prérequis : TP 33 à 43
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Ouvrez un nouveau classeur et enregistrez-le comme classeur prenant en charge les macros sous le nom TP_44.xlsm.

Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Visual Basic*. Dans la fenêtre *Projet*, cliquez avec le bouton droit sur l'objet *ThisWorkbook*, puis sur *Code*. Si la fenêtre *Projet* n'est pas visible, dans le menu *Affichage*, cliquez sur *Explorateur de projets*. Les termes fenêtre Projet et Explorateur de projet sont synonymes.



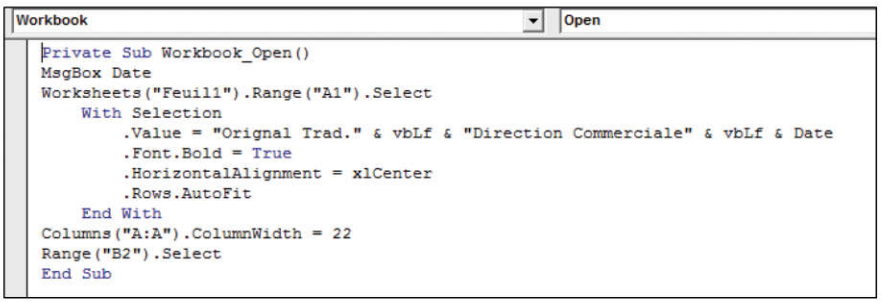
2 Dans la liste *Objet* au-dessus de la fenêtre *Code*, sélectionnez *Workbook*. Cette opération crée automatiquement une procédure vide pour l'événement *Open*. Par simple curiosité, cliquez sur la flèche située sur la droite, à côté de *Open* : vous voyez les très nombreux événements de l'objet *Workbook* auxquels il serait possible d'attacher une macro. Laissez *Open* sélectionné.



Copyright © 2012 Dunod.

3 Insérez le code suivant entre les instructions `Sub` et `End Sub` :

```
MsgBox Date
Worksheets("Feuil1").Range("A1").Select
With Selection
    .Value = "Orignal Trad." & vbLf & "Direction Commerciale" & vbLf & Date
    .Font.Bold = True
    .HorizontalAlignment = xlCenter
    .Rows.AutoFit
End With
Columns("A:A").ColumnWidth = 22
Range("B2").Select
```



4 Basculez vers Excel et enregistrez le classeur, puis fermez-le. Ouvrez à nouveau le classeur TP_44.xlsm. Dès lors que vous en avez activé le contenu, Excel exécute la procédure « Workbook_Open » qui affiche la date courante dans un message. Cliquez sur *OK* dans la boîte de dialogue de ce dernier.



5 La cellule *A1* de la feuille 1 affiche automatiquement un logo mis en forme, tandis que le curseur se place inéluctablement sur la cellule *B2*.



Cette procédure mérite quelques explications.
La première instruction, `MsgBox Date`, crée une boîte de message renfermant la date.
La seconde instruction débutant par `Worksheets` sélectionne la cellule *A1* de *Feuil1*.
Les lignes suivantes agissent toutes sur la sélection active : on utilise l'instruction `With` qui permet de modifier simultanément plusieurs propriétés de la sélection active. Ici, une chaîne de caractères est affectée à la valeur de la cellule (remarquez l'emploi du caractère de concaténation `&`, ainsi que celui du caractère spécial `vbLf`, une constante Visual Basic équivalent à un retour à la ligne). Puis sont appliqués une mise en gras, un centrage et l'ajustement de la hauteur de la ligne.
La ligne suivante ajuste la largeur de la colonne tandis que la dernière instruction place le curseur sur la cellule *B2*.
Fermez le classeur TP_44 sans l'enregistrer.

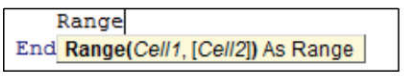
Copyright © 2012 Dunod.

Liste des membres et Information rapide automatique

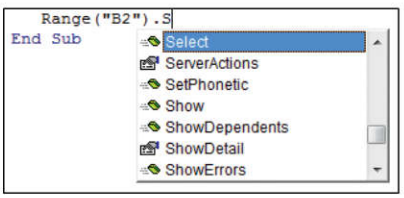
Excel propose plusieurs précieux outils pour faciliter la rédaction de macros.

Prérequis : néant
Fichier employé : TP_45_début.xlsm
Temps de réalisation : 5 minutes

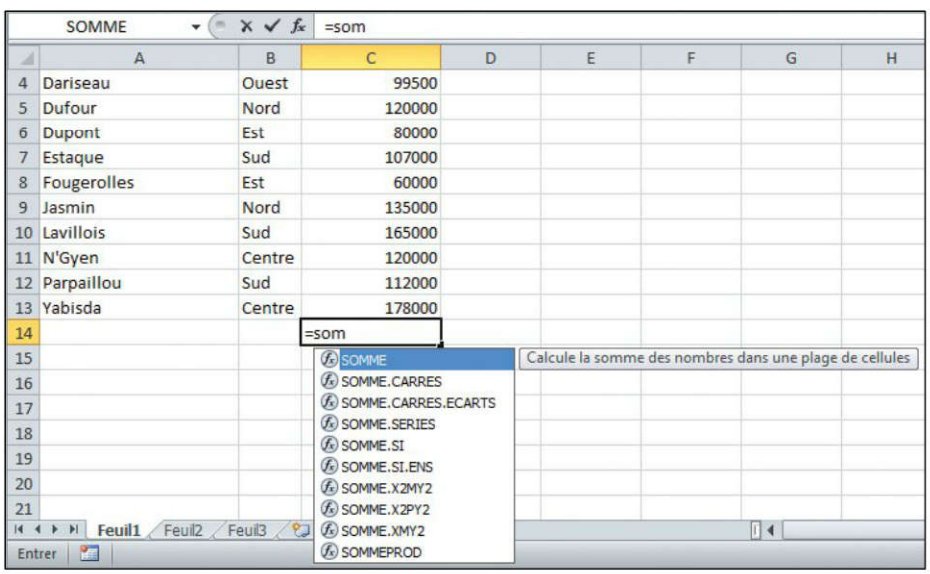
1 Ouvrez le classeur TP_45_début, activez le contenu, puis enregistrez-le sous le nom TP_45.xlsm. Ouvrez l'éditeur VBA, puis placez-vous dans la sous routine *Workbook_Open()*. Saisissez au-dessus de l'instruction `End Sub` l'instruction `Range`, effectuez un clic-droit puis choisissez *Informations Express*. Vous voyez apparaître la syntaxe de l'instruction `Range`.



2 Saisissez **("B2").S**. Vous voyez apparaître une zone de liste déroulante, correspond à la liste des membres, qui propose les méthodes et propriétés de l'objet qui précède le point. `Select` est présélectionné : double-cliquez dessus pour insérer l'instruction dans le code.

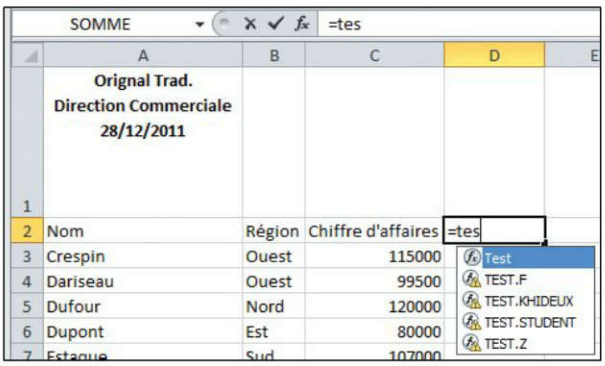


3 Dans le menu *Fichier* de l'Éditeur Visual Basic, choisissez *Fermer et retourner à Microsoft à Excel*. Dans la cellule *C14*, saisissez **=som**. L'Information rapide automatique propose les fonctions débutant par som.



Copyright © 2012 Dunod.

4 Cliquez sur *Somme* : une parenthèse est automatiquement ajoutée. Saisissez la plage **C3:C13**, puis fermez la parenthèse et appuyez sur *Entrée*. Dans la cellule *D2*, saisissez **=Tes** : l'information rapide automatique est disponible même pour des fonctions personnalisées créées par vous-même, et propose la fonction Test. Complétez l'instruction en référençant la cellule *B3*.
Enregistrez et fermez le classeur TP_45.xlsm.



Copyright © 2012 Dunod.

TP 46 Débogage

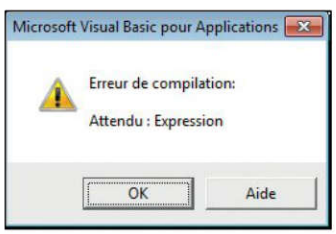
Hélas ! L'expérience le prouve, même la macro la plus soigneusement écrite peut aboutir à des conséquences inattendues. Pour obtenir un fonctionnement correct, vous devez découvrir les erreurs de programmation qui s'y cachent : ce processus est appelé débogage. Toute programmation peut entraîner plusieurs types d'erreurs. Dans ce TP, vous allez apprendre à les connaître et à les éliminer.

Prérequis : TP 33 à 45
Fichier employé : TP_46_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_46_début.xlsm et enregistrez-le sous le nom TP_46.xlsm. Cliquez n'importe où dans Feuil1.
Ouvrez l'Éditeur Visual Basic et examinez la procédure de fonction Test, située dans le Module 1 : vous voyez qu'une partie du code est affichée en rouge, ce qui indique une erreur de syntaxe ou de compilation. Ce type d'erreur est le plus fréquent et se produit lors d'une faute de frappe ou du non-respect de la succession de mots d'une ou de plusieurs instructions.

```
Function Test(a)
Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, _
IIf(a > 100000, 4, If(a > 80000, 5, 6)))))
End Function
```

2 Placez-vous sur le `If`, ajoutez puis supprimez un i, puis placez-vous sur l'instruction `End Sub` : une boîte de dialogue signale une erreur de compilation. L'Éditeur VBA est équipé d'une fonction de vérification automatique de la syntaxe qui contrôle la validité d'une ligne de code dès que vous la quittez. S'il détecte une erreur, il affiche cette boîte de message et place la ligne erronée en rouge. Fermez la boîte de dialogue.



3 L'instruction fautive est affichée en rouge, la partie erronée étant placée en surbrillance. Ajouter le i manquant et placez-vous sur l'instruction `End Sub` : toute la ligne est désormais affichée en noir. Il n'y a plus d'erreur évidente. Fermez l'Éditeur Visual Basic pour revenir à Excel.

```
Function Test(a)
Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, _
IIf(a > 100000, 4, If(a > 80000, 5, 6)))))
End Function
```

Copyright © 2012 Dunod.

4 Dans l'onglet *Développeur*, cliquez sur *Macros*. Choisissez la macro *ImportListe*, puis cliquez sur *Exécuter* : une boîte de dialogue signale une erreur d'exécution. C'est le second type d'erreur qui se produit lors de l'exécution du code. Même si la syntaxe est correcte, le programme peut tenter d'effectuer une tâche impossible, comme ici charger un fichier depuis un dossier inexistant. La boîte de dialogue affiche généralement la nature de l'erreur. Fermez-la. (Vous pouvez facilement corriger cette macro : modifiez le chemin d'accès en choisissant celui qui mène vers le dossier contenant le fichier texte TP_38.txt).



5 Cliquez sur la feuille *Liste*, puis dans le groupe *Code* de l'onglet *Développeur* sur *Macros*, choisissez *TriRégion* et cliquez sur *Exécuter*. La macro fonctionne parfaitement, mais le résultat n'est pas celui escompté : il se produit ici une erreur logique. Dans le cas présent, le tri ne s'effectue pas sur les bonnes cellules.

	A	B	C	D
1				
2		Crespin	Ouest	Chiffre d'affaires
3		Dariseau	Ouest	115000
4		Dufour	Nord	99500
5		Dupont	Est	120000
6		Estaque	Sud	80000
7		Fougerolles	Est	107000
8		Jasmin	Nord	60000
9		Lavillois	Sud	135000
10		N'Gyen	Centre	165000
11		Nom	Région	120000
12		Parpaillou	Sud	112000
13		Yabisda	Centre	178000

Ce type d'erreurs est souvent le plus complexe à corriger. Celles-ci peuvent même parfois passer inaperçues et aboutir à votre insu à un résultat erroné. Heureusement, l'éditeur VBA dispose d'outils permettant de venir à bout de tous ces types d'erreurs : c'est ce que nous allons voir dans les prochains TP.
Enregistrez et fermez le classeur TP_46.xlsm.

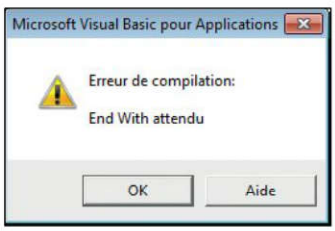
Copyright © 2012 Dunod.

TP 47 Compilateur VBA

Le compilateur VBA vérifie la syntaxe des lignes de code au fur et à mesure de leur saisie, comme nous l'avons vu dans le TP précédent. Il détecte les erreurs ligne à ligne au moment où elles se produisent. Des erreurs de syntaxe peuvent toutefois couvrir plusieurs lignes, auquel cas elles ne sont pas immédiatement identifiables.

Prérequis : TP 33 à 46
Fichier employé : TP_47.xslm
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_47_début.xlsm et enregistrez-le sous le nom TP_47.xlsm.
Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Visual Basic*. Dans l'Éditeur Visual Basic, cliquez dans le menu *Débogage* sur *Compiler VBAProject*. Une boîte de dialogue signale une erreur de compilation, ici l'absence d'une instruction `End With`. Fermez la boîte de dialogue.



2 Examinez la procédure *MiseEnForme*. Aucune ligne n'est affichée en rouge : de prime abord, la syntaxe est correcte. Remarquez toutefois la présence d'une apostrophe devant le premier `End With`, transformant cette instruction en commentaire : c'est la cause de l'erreur. Dans la réalité, cette instruction serait probablement totalement absente, compliquant l'identification du problème.
La boîte de dialogue signalant l'erreur de compilation indique le type de celle-ci ainsi que la ligne où elle s'est produite. Après l'avoir corrigé, vous devez recompiler le projet, et cela chaque fois qu'il en découvre une.

```
Sub MiseEnForme()
'
' Macro MiseEnForme
'

' Ajout de l'étiquette de colonne Classe
    Range("E2").Select
    ActiveCell.FormulaR1C1 = "Classe"

' Affectation de l'instruction Test() et copie dans la colonne.
    Range("E3").Select
    ActiveCell.FormulaR1C1 = "=Test(RC[-1])"
    Range("E3").Select
    Selection.Copy

    Range("E3:E13").Select
    ActiveSheet.Paste
    Application.CutCopyMode = False
    Selection.HorizontalAlignment = xlCenter

'Mise en forme des chiffres d'affaires en monétaire sans décimale
    Range("D3:D13").Select
    Selection.Style = "Currency"
    Selection.NumberFormat = "_-* #,##0 $_-;-* #,##0 $_-;_-* ""-""?? $_-;_-@_-"

'Tri du tableau par région et chiffre d'affaires
    Range("B2:E13").Select
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Add Key:=Range("C3:C13") _
        , SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Add Key:=Range("D3:D13") _
        , SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal
    With ActiveWorkbook.Worksheets("Feuil1").Sort
        .SetRange Range("B2:E13")
        .Header = xlYes
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
'   End With
```

Copyright © 2012 Dunod.

3 Enlevez la marque de commentaire, puis choisissez à nouveau dans le menu *Débogage* l'option *Compiler VBAProject*. Le compilateur ne décèle plus d'erreur de compilation. Enregistrez le fichier puis cliquez dans *Fichier* sur *Fermer et retourner dans Microsoft Excel*. Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Macros*, choisissez *MiseEnForme* puis cliquez sur *Exécuter*: Votre tableau est parfaitement mis en forme.
Enregistrez et fermez le classeur TP_47.xlsm.

	A	B	C	D	E	F
1	**Orignal Trad. Direction Commerciale 29/12/2011**					
2		**Nom**	**Région**	**Chiffre d'affaires**	**Classe**	
3		**Yabisda**	Centre	178 000 €	1	
4		**N'Gyen**	Centre	120 000 €	4	
5		**Dupont**	Est	80 000 €	6	
6		**Fougerolles**	Est	60 000 €	6	
7		**Jasmin**	Nord	135 000 €	3	
8		**Dufour**	Nord	120 000 €	4	
9		**Crespin**	Ouest	115 000 €	4	
10		**Dariseau**	Ouest	99 500 €	5	
11		**Lavillois**	Sud	165 000 €	2	
12		**Parpaillou**	Sud	112 000 €	4	
13		**Estaque**	Sud	107 000 €	4	
14						
15						

Copyright © 2012 Dunod.

TP 48 Exécution pas à pas

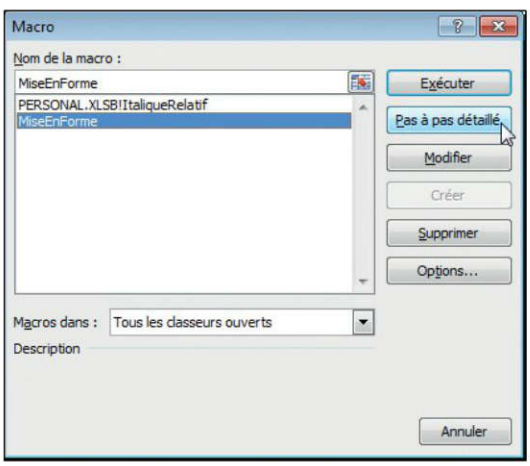
Parfois, le fait d'identifier l'endroit où se cache une erreur de logique ou d'exécution n'est pas chose évidente. L'exécution d'une procédure avec un arrêt à chaque ligne, nommée exécution pas à pas, permet de ralentir l'application de la procédure et d'examiner la façon dont elle traite chaque ligne. Cette stratégie met souvent en évidence un dysfonctionnement.

Prérequis : TP 33 à 47
Fichier employé : TP_48_début.xlsm
Temps de réalisation : 15 minutes

1 Ouvrez le classeur TP_48_début.xlsm et activez le contenu.
Dans le groupe *Code* de l'onglet *Développeur*, choisissez *Macros*. Sélectionnez la macro *MiseEnForme* et cliquez sur *Exécuter* : la macro fonctionne parfaitement, mais le résultat n'est pas celui attendu.

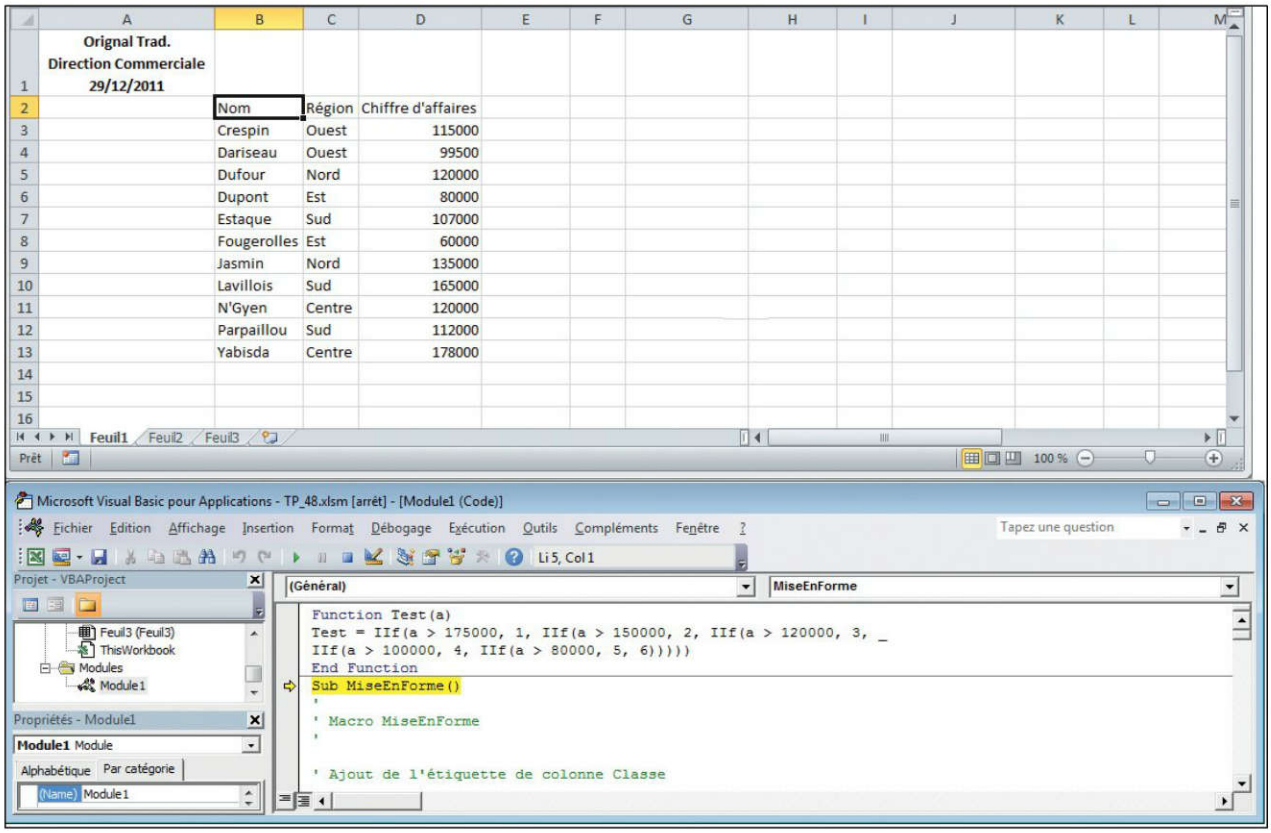
Nom	Région	Chiffre d'affaires	Classe
Yabisda	Centre	178 000 €	1
N'Gyen	Centre	120 000 €	4
Dupont	Est	80 000 €	6
Fougerolles	Est	60 000 €	6
Jasmin	Nord	135 000 €	3
Dufour	Nord	120 000 €	4
Crespin	Ouest	115 000 €	4
Dariseau	Ouest	99 500 €	5
Lavillois	Sud	165 000 €	2
Parpaillou	Sud	112 000 €	4
Estaque	Sud	107 000 €	4

2 Fermez le classeur sans l'enregistrer. Ouvrez-le à nouveau, enregistrez-le sous le nom TP_48.xlsm et activez le contenu. Dans le groupe *Code* de l'onglet *Développeur*, choisissez *Macros*. Sélectionnez la macro *MiseEnForme* et cliquez sur *Pas à pas détaillé*.

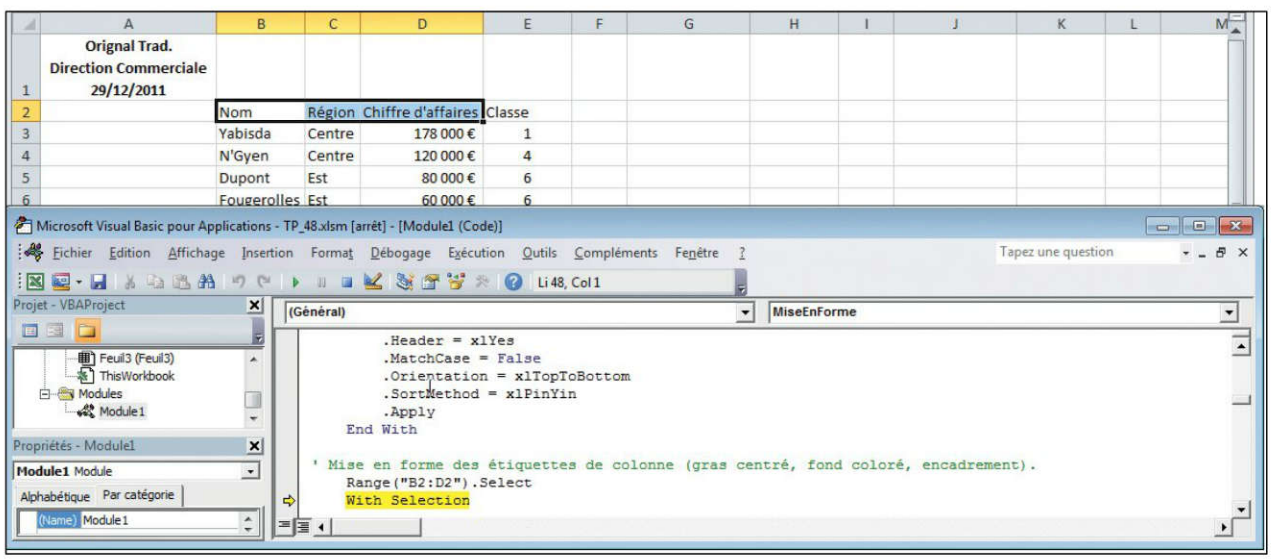


Copyright © 2012 Dunod.

3 L'Éditeur Visual Basic s'ouvre sur le code. La première instruction de la macro est surlignée en jaune. Organisez les fenêtres de l'Éditeur Visual Basic et d'Excel de façon à les avoir l'une au-dessus de l'autre et pouvoir suivre facilement les effets de la macro sur les données.

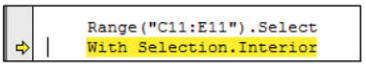


4 Appuyez sur la touche *F8* pour passer à la seconde ligne de la procédure. Cette touche fonctionne aussi longtemps que le point d'insertion se trouve quelque part dans la routine. Observez bien ce que fait la procédure pendant que vous l'examinez pas à pas. Ayant abouti (au bout d'un temps certain) à la section *' Mise en forme des étiquettes de colonne (gras centré, fond coloré, encadrement)*, vous vous apercevez que la sélection est incorrecte. L'instruction `Range("B2:D2").Select` est erronée, la bonne plage est *B2:E2*.



Copyright © 2012 Dunod.

5 Notez cette ligne, puis continuez à appuyer sur *F8*. Tout se déroule normalement, jusqu'au dernier groupe d'instructions de la macro : l'instruction `Range("C11:E11").Select` est aussi erronée, ce devrait être `Range("C12:E12").Select`. Poursuivez la vérification jusqu'à la fin de la procédure.



6 Modifiez les deux instructions fautives : `Range("B2:D2").Select` en **Range("B2:E2").Select** et `Range("C11:E11").Select` en **Range("C12:E12").Select**. Cela étant fait, enregistrez le code, revenez à Excel puis relancez la macro *MiseEnForme* : cette fois, le tableau est correctement mis en forme.

	A	B	C	D	E
1	**Orignal Trad. Direction Commerciale 29/12/2011**				
2		**Nom**	**Région**	**Chiffre d'affaires**	**Classe**
3		**Yabisda**	Centre	178 000 €	1
4		**N'Gyen**	Centre	120 000 €	4
5		**Dupont**	Est	80 000 €	6
6		**Fougerolles**	Est	60 000 €	6
7		**Jasmin**	Nord	135 000 €	3
8		**Dufour**	Nord	120 000 €	4
9		**Crespin**	Ouest	115 000 €	4
10		**Dariseau**	Ouest	99 500 €	5
11		**Lavillois**	Sud	165 000 €	2
12		**Parpaillou**	Sud	112 000 €	4
13		**Estaque**	Sud	107 000 €	4
14					

Vous auriez probablement pu repérer ces erreurs sans recourir à chaque étape détaillée, mais c'était un exemple. Enregistrez et fermez le fichier TP_48.xlsm.

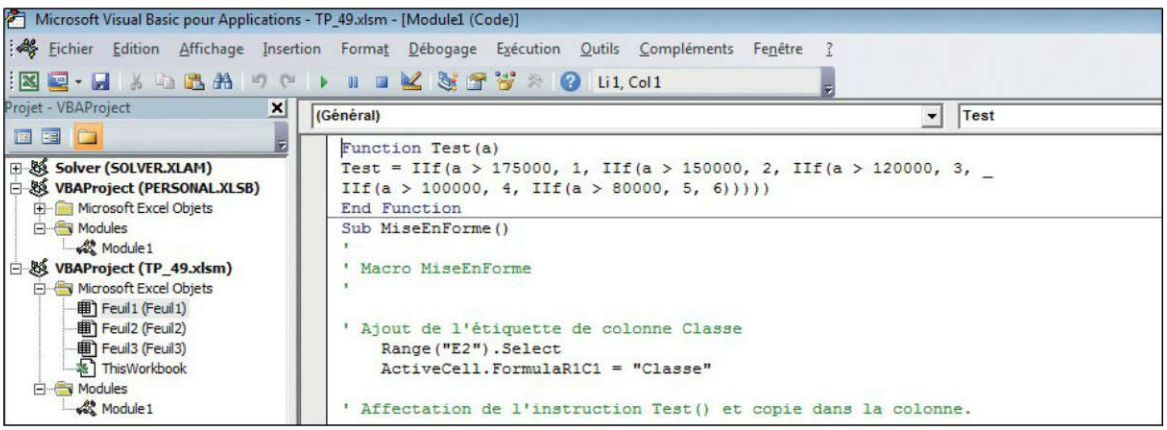
Copyright © 2012 Dunod.

TP 49 Points d'arrêt

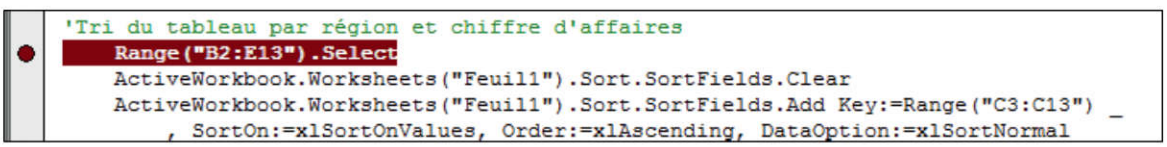
Si la procédure qui contient des erreurs est longue, vous risquez de passer beaucoup de temps à parcourir les lignes de code avant d'arriver à l'endroit qui mérite d'être examiné, surtout si vous corrigez les erreurs au fur et à mesure. C'est là qu'un point d'arrêt peut vous aider. Il identifie une ligne de code spécifique, arrêtant le programme lorsqu'il atteint ce point. Vous pouvez ensuite continuer pas à pas.

Prérequis : TP 33 à 48
Fichier employé : TP_49_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_49_début.xlsm et enregistrez-le sous le nom TP_49.xslm. Activez le contenu. Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Visual Basic* pour afficher l'Éditeur Visual Basic. Atteignez la procédure *Sub MiseEnForme*.



2 Redimensionnez les deux fenêtres pour voir simultanément l'Éditeur et Excel. Placez-vous sous le commentaire `'Tri du tableau par région et chiffre d'affaires` et cliquez dans la marge à gauche de l'instruction `Range("B2:E13").Select` : un point rouge apparaît dans la marge, tandis que la ligne est surlignée en rouge. Vous pourriez tout aussi bien cliquer sur *Débogage*, *Basculer le point d'arrêt*, ou appuyer sur la touche *F9* pour parvenir au même résultat.



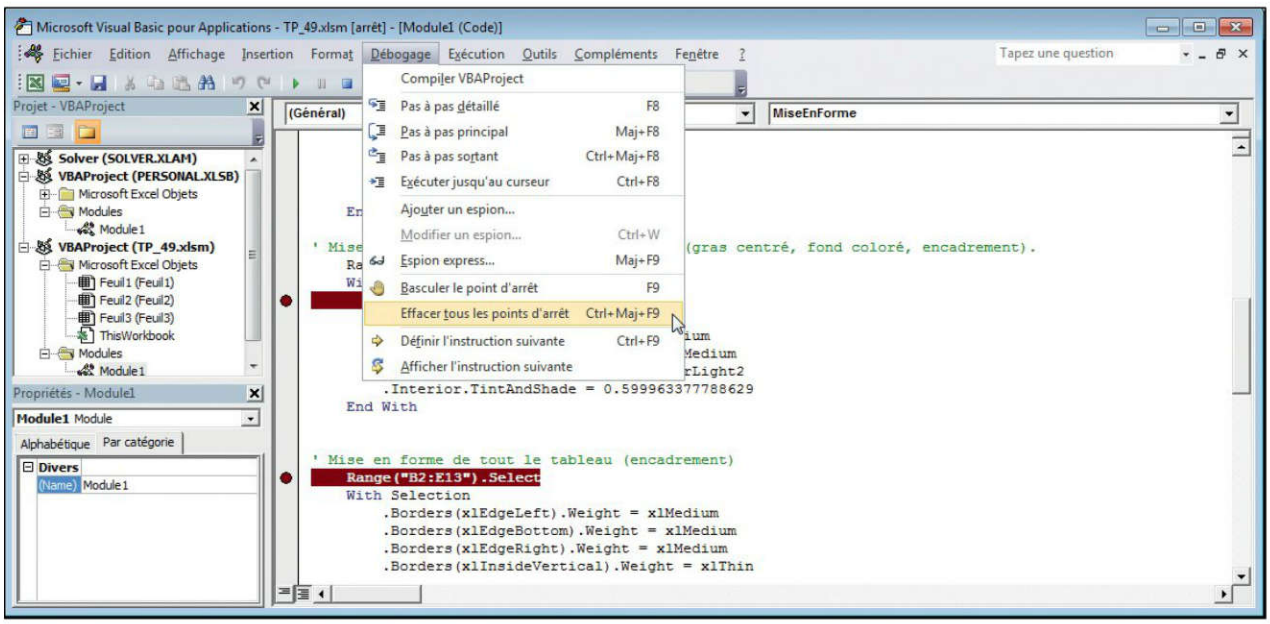
Copyright © 2012 Dunod.

3 Exécutez la procédure, soit depuis Excel, soit depuis l'Éditeur Visual Basic en cliquant sur le bouton *Exécuter* de la barre d'outils, en choisissant *Exécuter Sub/userForm* dans le menu *Exécution* ou en appuyant sur *F5*. La procédure s'exécute jusqu'au point d'arrêt, la ligne concernée affichant un surlignement rouge et jaune.

```
'Tri du tableau par région et chiffre d'affaires
    Range("B2:E13").Select
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Add Key:=Range("C3:C13") _
        , SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
```

4 Appuyez sur *F8* pour continuer progressivement l'exécution du reste de la procédure. La commande de point d'arrêt est une bascule arrêt/marche. Pour ôter un point d'arrêt, suivez les mêmes étapes que pour le placer. Une procédure peut comporter plusieurs points d'arrêts. Pour tous les supprimer, choisissez *Supprimer tous les points d'arrêt* dans le menu *Débogage*.

Lorsque vous fermez un classeur contenant des procédures dotées de points d'arrêt, ces derniers sont automatiquement supprimés. Fermez le classeur TP_49.xlsm.



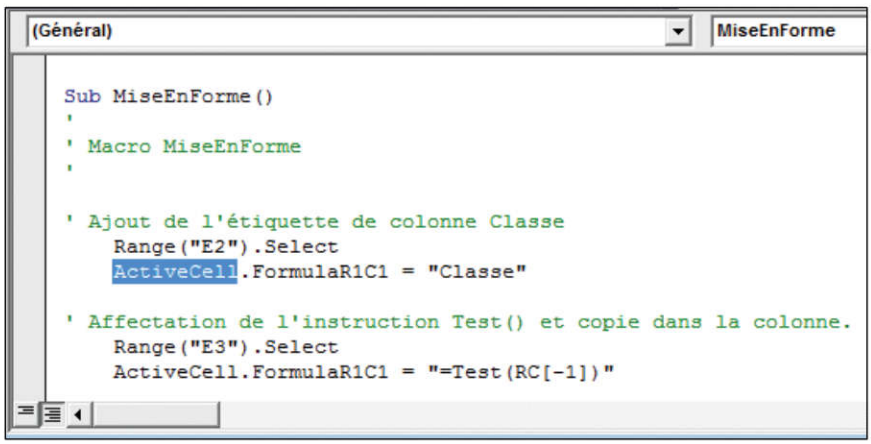
Copyright © 2012 Dunod.

TP 50 Fenêtre Espions

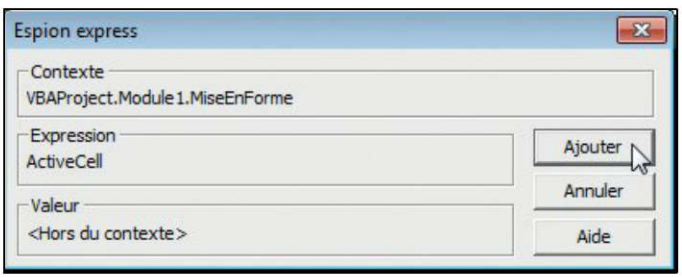
Il est parfois nécessaire, pendant un débogage, de connaître la valeur d'une variable ou d'une propriété. Vous pouvez surveiller celle-ci afin de suivre ses évolutions pendant l'exécution de la procédure.

Prérequis : TP 33 à 49
Fichier employé : TP50_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_50_début.xlsm et enregistrez-le sous le nom TP_50.xslm. Activez le contenu. Ouvrez l'Éditeur Visual Basic. Dans la fenêtre *Code* de l'éditeur, dans la procédure *MiseEnForme*, sélectionnez sur la troisième ligne d'instruction `ActiveCell`.

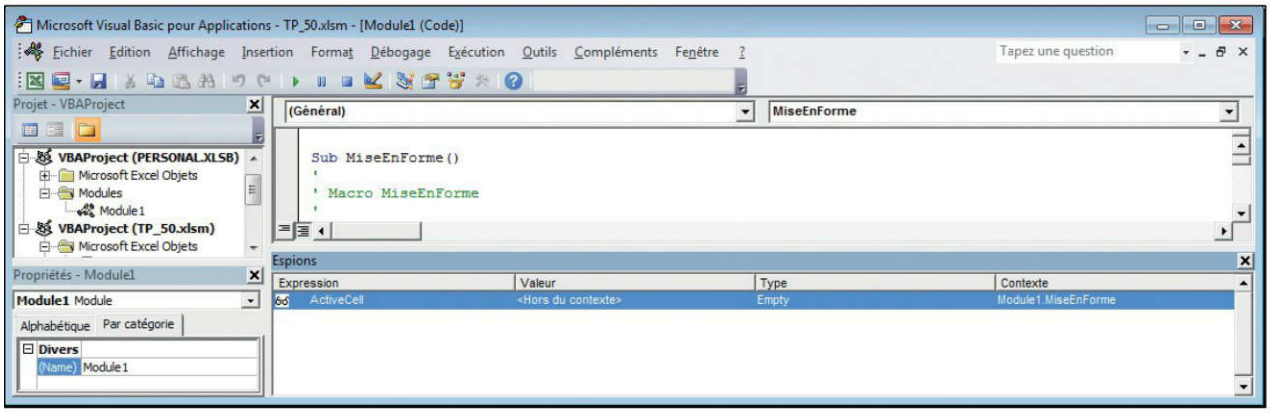


2 Dans le menu de l'Éditeur, choisissez *Débogage*, *Espion* express, ou appuyez sur *Maj+F9*. La boîte de dialogue Espion express apparaît. Cliquez sur *Ajouter*.

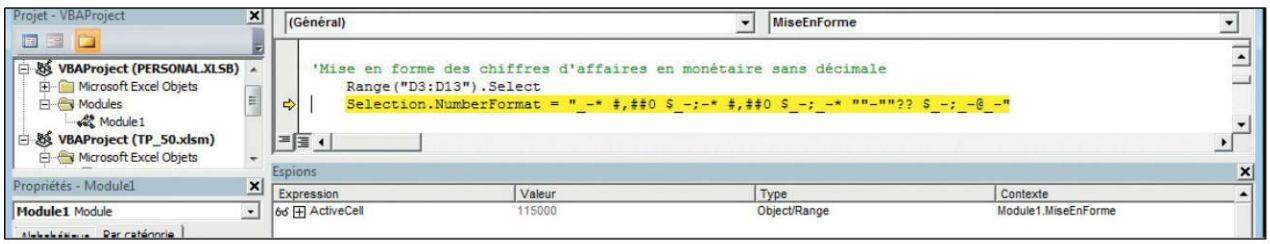


Copyright © 2012 Dunod.

3 L'expression ActiveCell apparaît dans la fenêtre Espions.



4 Placez-vous sur l'instruction `Sub MiseEnForme`, puis, dans le menu *Débogage,* choisissez *Pas à Pas détaillé*. Appuyez de façon répétitive sur *F8* pour parcourir la procédure espionnée en mode pas à pas et observez les modifications de la valeur de la variable espionnée (ici, la valeur de la cellule active) pendant l'exécution de la procédure.



Vous pouvez ainsi suivre n'importe quelle variable ou propriété. Vous pouvez espionner plusieurs variables simultanément. C'est un outil précieux en cas d'erreur de logique non apparente.
Enregistrez et fermez le classeur TP_50.xlsm.

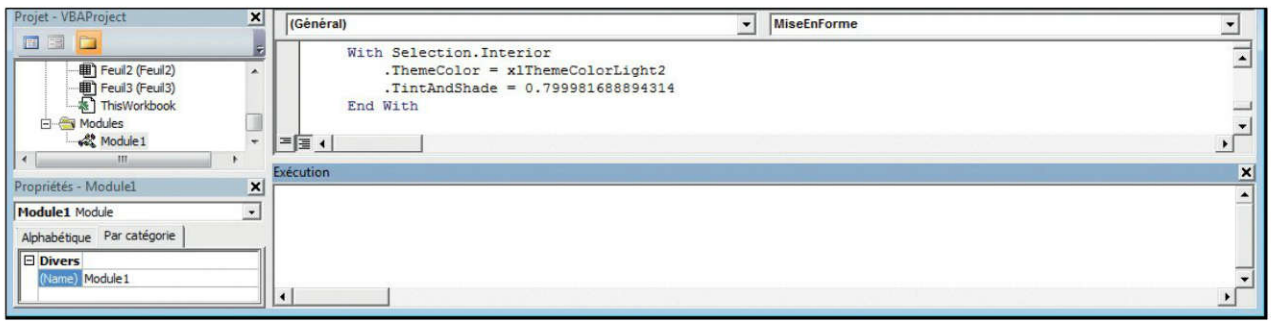
Copyright © 2012 Dunod.

TP 51 Fenêtre Exécution

La fenêtre Exécution de l'Éditeur Visual Basic affiche le résultat d'une fonction ou d'une propriété.

Prérequis : TP 33 à 50
Fichier employé : TP_51_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le classeur TP_51_début.xlsm et enregistrez-le sous le nom TP_51.xslm. Activez le contenu. Ouvrez l'Éditeur Visual Basic. Dans le menu *Affichage*, cliquez sur *Fenêtre Exécution*. Vous pouvez également appuyer sur *Ctrl+G*. La fenêtre apparaît dans l'Éditeur.



2 Dans Excel, sélectionnez la cellule *D2*. Dans l'Éditeur Visual Basic, dans la fenêtre *Exécution*, saisissez **?ActiveCell.Interior.TintAndShade**, puis appuyez sur *Entrée* : la valeur actuelle de la propriété apparaît en dessous.

```
Exécution
 ?ActiveCell.Interior.TintAndShade
  0,599963377788629
```

3 Dans la fenêtre *Exécution*, saisissez **?Activeworkbook.Path**, puis appuyez sur *Entrée* : le chemin du classeur actif apparaît (le vôtre sera probablement différent de celui de la capture d'écran).

```
Exécution
 ?ActiveCell.Interior.TintAndShade
  0,599963377788629
 ?Activeworkbook.Path
K:\TPExcel2
```

4 Cette fenêtre Exécution permet également de tester des fonctions personnalisées. Saisissez dans la fenêtre Exécution **?Test(190000)**. Vous obtenez la réponse correcte, 1. Cette fonction a été programmée correctement. Si la réponse est fausse, vous saurez que vous devrez vous remettre à l'ouvrage.

```
Exécution
 ?Activeworkbook.Path
K:\TPExcel2
 ?Test(190000)
  1
```

Ce type d'essai permet de mieux vous familiariser avec les objets et leurs propriétés et de tester des fonctions personnalisées aux abords des valeurs frontières susceptibles de poser problème.
Enregistrez et fermez le classeur TP_51.xlsm.

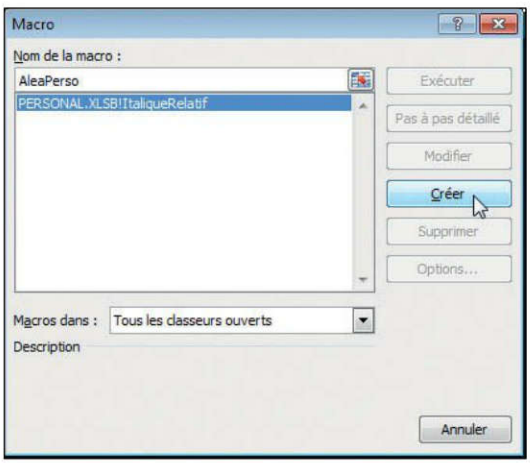
Copyright © 2012 Dunod.

TP 52 Amélioration d'une fonction intégrée d'Excel

Excel dispose d'un nombre impressionnant de fonctions intégrées. Parfois, cependant, celles-ci pourraient ne pas correspondre exactement à vos besoins. Il est alors souvent facile de créer une fonction personnalisée qui réponde mieux à vos attentes. Par exemple, la fonction ALEA d'Excel génère un nombre aléatoire compris entre 0 et 1. Vous pourriez souhaiter disposer d'une fonction renvoyant une valeur comprise entre deux bornes (ce qu'effectue en fait déjà la fonction Excel ALEA.ENTRE.BORNES), mais avec un nombre de décimales paramétrable, et vouloir en outre que la fonction soit « résistante aux erreurs », c'est-à-dire que l'absence d'un ou de plusieurs paramètres n'entraîne pas d'erreur.

Prérequis : TP 33 à 51
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Ouvrez un nouveau classeur et enregistrez-le comme classeur prenant en charge les macros sous le nom TP_52.xlsm.
Dans le groupe *Code* de l'onglet *Développeur*, cliquez sur *Macros*. Nommez-la nouvelle macro **AleaPerso** et cliquez sur *Créer*.



2 Dans la fenêtre *Code* de l'Éditeur Visual Basic, modifiez l'instruction `Sub` en **Function**. L'instruction `End Sub` change automatiquement en `End Function`. Corrigez l'instruction `Function AleaPerso()` en **Function AleaPerso(min, max, dec)**. Vous venez de doter votre fonction de paramètres.

```
Function AleaPerso(min, max, dec)

End Function
```

3 Saisissez maintenant le code de la fonction entre les deux instructions : **AleaPerso = Rnd * (max – min) + min**.
`Rnd` est la fonction Visual Basic correspondant à la fonction Excel `ALEA`. Cette valeur étant comprise entre 0 et 1, vous la multipliez par l'étendue souhaitée (max–min) et lui ajoutez la valeur minimale.

```
Function AleaPerso(min, max, dec)
AleaPerso = Rnd * (max - min) + min
End Function
```

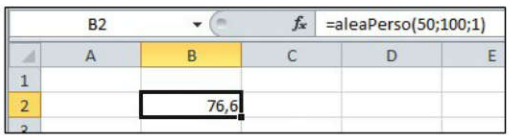
Copyright © 2012 Dunod.

4 Pour gérer le nombre de décimale, complétez comme suit la fonction : **AleaPerso = int(Rnd * (max – min) +-min*10^dec)/10^dec**.

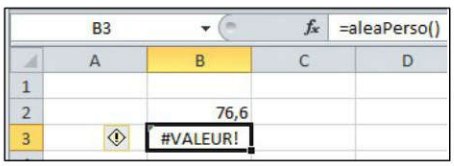
Vous multipliez le résultat obtenu par 10 à la puissance dec (soit 1 lorsque dec est égal à 0) ; vous le transformez en entier puis le divisez par 10 à la puissance dec : le nombre de décimales sera bien égal à dec.

```
Function AleaPerso(min, max, dec)
AleaPerso = Int((Rnd * (max - min) + min) * 10 ^ dec) / 10 ^ dec
End Function
```

5 Enregistrez le classeur, revenez à Excel, puis saisissez dans la cellule *B2* **= AleaPerso(50 ;100 ;1)**, puis appuyez sur *Entrée*. Vous obtenez une valeur comprise entre 50 et 100 avec une décimale. Le chiffre étant aléatoire, il devrait être différent de celui de la capture d'écran.

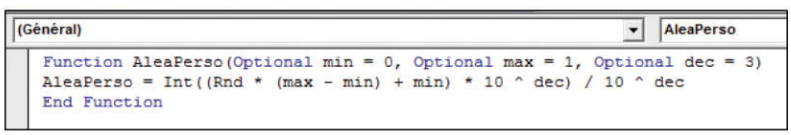


6 Dans la cellule *B3*, saisissez dans la cellule *B2* **= AleaPerso()** : Excel affiche le message d'erreur #VALEUR. Pour cette fonction, les arguments sont actuellement obligatoires.

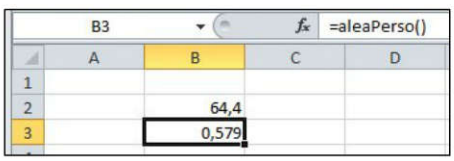


7 Pour corriger ceci, affichez la fonction dans l'éditeur Visual Basic et modifiez-la en **Function AleaPerso(optional min = 0 optional max = 1, optional dec = 3)**.

Le mot-clé `optional` signale que le paramètre est facultatif. Il lui est affecté une valeur par défaut. Dans le cas présent, si aucun paramètre n'est fourni par l'utilisateur, le nombre obtenu sera un nombre compris entre 0 et 1 doté de 3 décimales.



8 Enregistrez le classeur, revenez à Excel, puis saisissez dans la cellule *B3* **= AleaPerso()**, puis appuyez sur *Entrée*. Vous obtenez une valeur de la forme attendue.



Lorsque vous déclarez des paramètres facultatifs, ceux-ci ne peuvent être suivis que de paramètres également facultatifs. Dans une déclaration, vous devez donc placer en premier les paramètres obligatoires, puis les paramètres facultatifs.

Enregistrez et fermez le classeur TP_52.xslm.

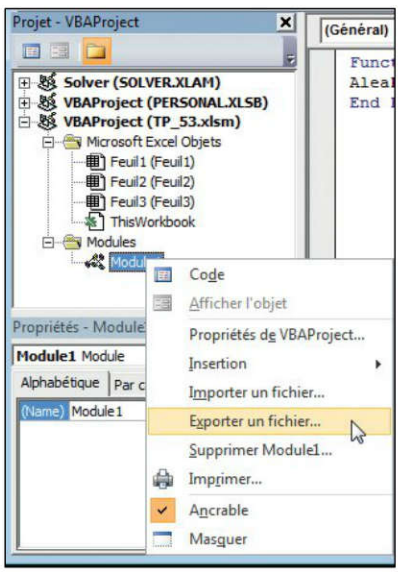
Copyright © 2012 Dunod.

Exportation et importation d'un module de code

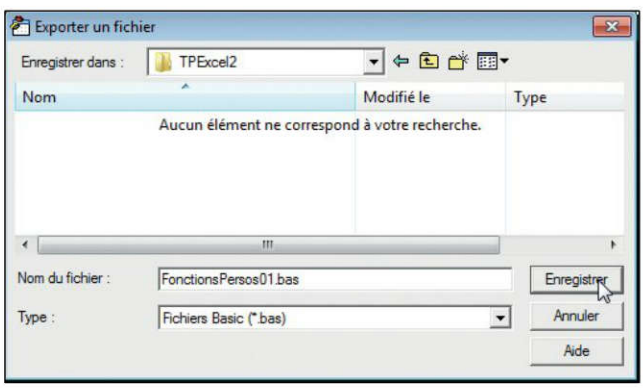
Plutôt que de copier une procédure du module d'un classeur dans le module d'un autre classeur, il est souvent intéressant d'importer la totalité d'un module existant dans un nouveau classeur. Pour ce faire, vous devez d'abord exporter le module concerné.

Prérequis : TP 33 à 52
Fichier employé : TP_53_début.xlsm
Temps de réalisation : 5 minutes

1 Ouvrez le fichier TP_53_début.xlsm et enregistrez-le sous le nom TP_53.xlsm.
Ouvrez l'Éditeur Visual Basic. Dans l'*explorateur de projet*, effectuez un clic-droit sur *Module1*, puis choisissez *Exporter un fichier*.

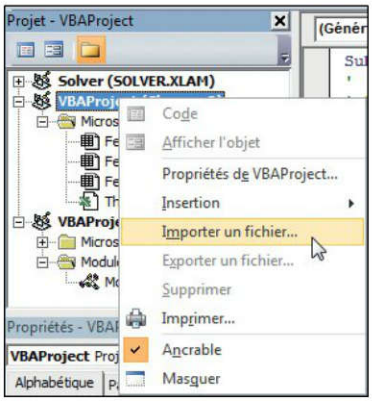


2 Dans la boîte de dialogue, saisissez comme nom du module **FonctionsPersos01**. Remarquez que le fichier possède une extension *.bas*. Naviguez jusqu'au dossier des exercices, puis cliquez sur *Enregistrer*.

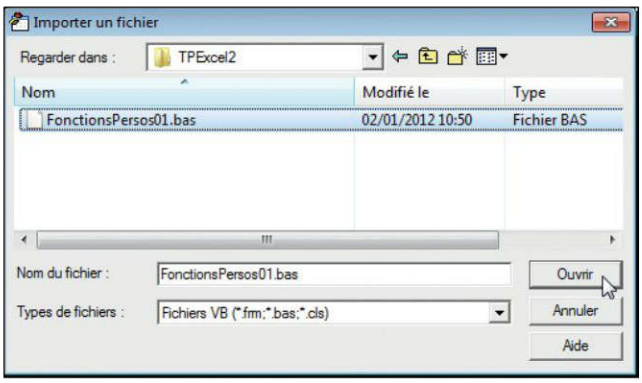


Copyright © 2012 Dunod.

3 Fermez le classeur TP_53.xlsm. Ouvrez un classeur vierge. Ouvrez l'Éditeur Visual Basic. Dans l'*Explorateur de projet*, effectuez un clic-droit sur la ligne *VBAProject (classeur2)* (le nom du classeur peut différer), puis choisissez *Importer un fichier*.



4 Dans la boîte de dialogue, naviguez jusqu'au dossier d'exercices, sélectionnez le fichier *FonctionsPersos01.bas* puis cliquez sur *Ouvrir* : une arborescence *Modules* apparaît, avec le module 1 contenant les procédures du module importé. Celles-ci sont désormais disponibles dans votre nouveau classeur.



S'il est facile de copier-coller une procédure d'un module dans un autre lorsque deux classeurs différents sont ouverts dans Excel, le fait d'exporter et d'importer des modules comprenant vos routines favorites est un excellent moyen de gagner du temps.
Fermez le nouveau classeur sans l'enregistrer.

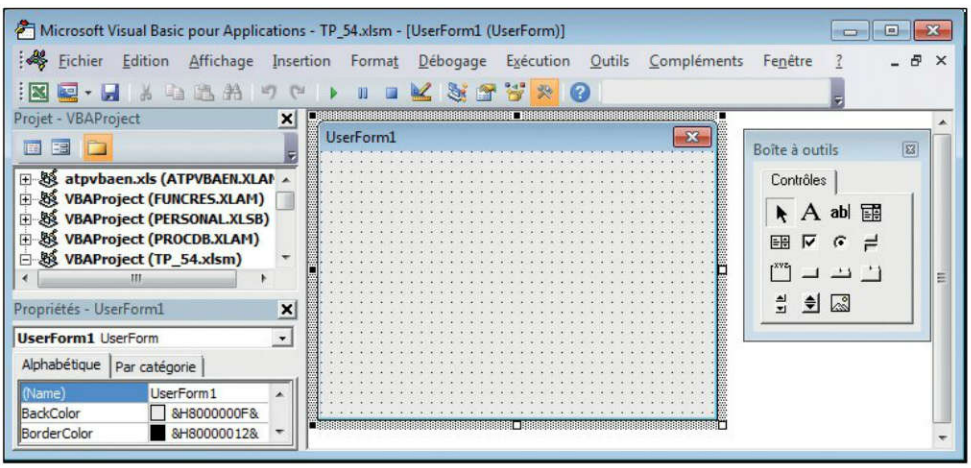
Copyright © 2012 Dunod.

Création d'une interface utilisateur

Tout au long de ces derniers TP, vous avez vu comment employer des macros ou procédures Visual Basic afin de faciliter votre travail (ou celui de vos collaborateurs). Il est toutefois possible d'aller beaucoup plus loin, en proposant une interface utilisateur dédiée à un classeur qui regroupe la majorité (sinon la totalité) des actions susceptibles d'être effectuées par un utilisateur. Il existe deux façons de procéder : placer sur une feuille de calcul des contrôles ActiveX qui agissent sur la feuille à l'aide de procédures VBA, ou créer une interface utilisateur totalement distincte, dotée de contrôles. Une telle interface se nomme UserForm (formulaire utilisateur) avec Excel, et c'est ce que nous allons analyser dans ce dernier TP. L'interface créée aura pour but de gérer le fichier des ventes des commerciaux que nous avons déjà rencontré. Vous avez déjà examiné la majorité des procédures qui vont être employées : seules les nouvelles seront détaillées ici.

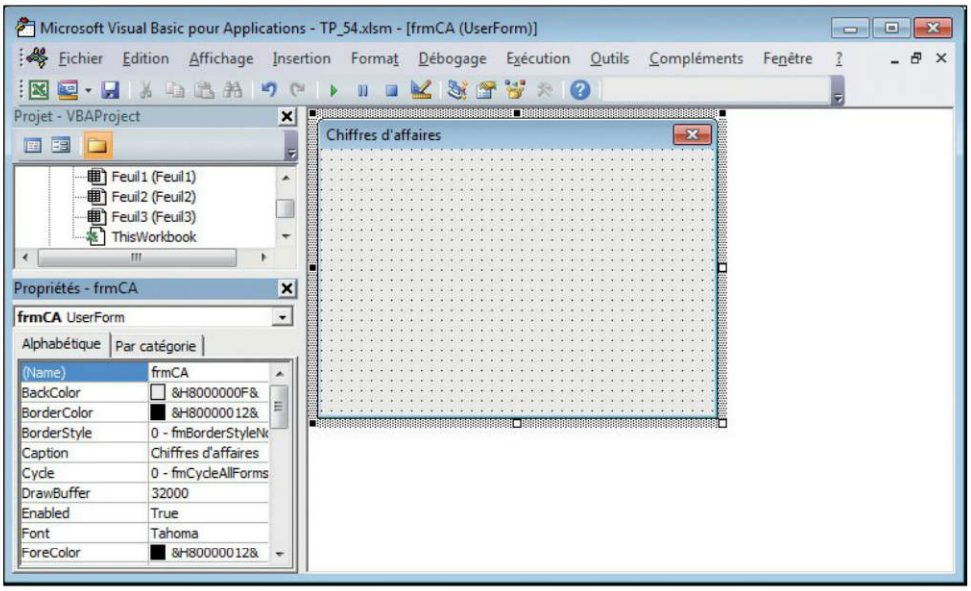
Prérequis : TP 33 à 53
Fichier employé : TP_54_début.xlsm
Temps de réalisation : 20 minutes

1 Ouvrez le fichier TP_54_début.xlsm et enregistrez-le sous le nom TP_54.xlsm. Activez le contenu.
Ouvrez l'Éditeur Visual Basic. Dans le menu *Insertion*, choisissez *UserForm*. Un formulaire utilisateur vierge s'affiche dans l'Éditeur, ainsi que la fenêtre flottante de la *Boîte à outils*.

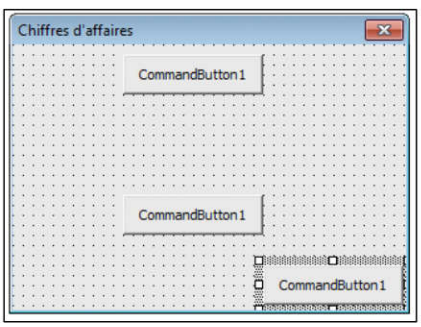


Copyright © 2012 Dunod.

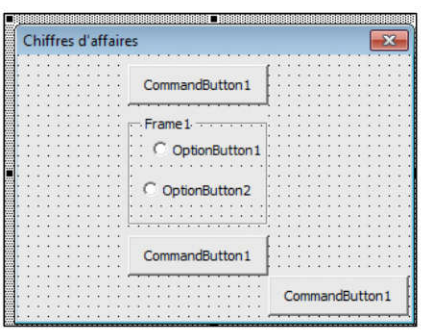
2 Dans la fenêtre *Propriétés* de l'Éditeur Visual Basic, en bas à gauche, modifiez la propriété *Caption* de *UserForm1* en **Chiffres d'affaires**. L'intitulé se modifie dans la barre de titre du formulaire utilisateur. Changez ensuite la valeur de la propriété *Name* en **frmCA**. *frm* est l'abréviation de formulaire, et attribuer un nom plus explicite facilitera une référence à ce formulaire dans une procédure.



3 Vous allez maintenant construire le contenu de l'interface utilisateur. Dans la *Boîte à outils*, cliquez sur l'outil *Bouton de commande*, puis cliquez en haut au tiers du formulaire. Appuyez sur la touche *Ctrl* et, en la maintenant enfoncée, glissez-déposer le bouton juste en dessous du précédent pour y créer une copie. Répétez l'opération pour placer une dernière copie en bas à droite du formulaire. Ne vous préoccupez pas de les placer parfaitement, nous le ferons ultérieurement.



4 Il reste un élément à ajouter. Dans la *Boîte à outils*, cliquez sur *Cadre*, puis placez un contrôle cadre entre les deux boutons centraux. Double-cliquez ensuite dans la *Boîte à outils* sur *Bouton d'option*, cliquez 2 fois dans le cadre puis cliquez à nouveau sur *Bouton d'option* dans la *Boîte à outils* pour désactiver cet outil.

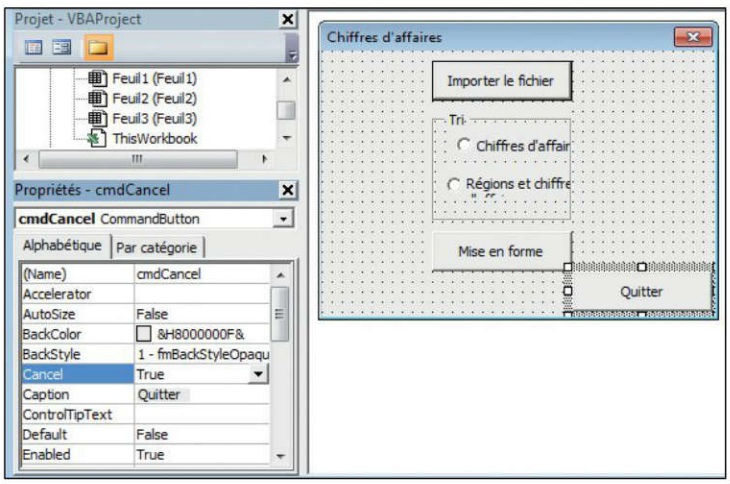


Copyright © 2012 Dunod.

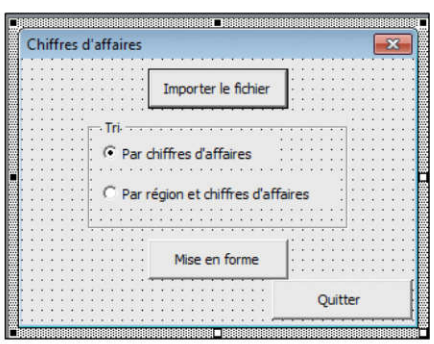
5 Nous allons maintenant nommer les contrôles et définir leurs intitulés. Tout cela va s'effectuer dans la fenêtre *Propriétés*. Choisissez dans la zone du haut le nom du contrôle (sa propriété *Name* actuelle), puis cherchez la propriété à corriger dans la liste située en dessous. Modifiez comme précisé ci-après les propriétés :

Nom contrôle	Propriété Name	Propriété Caption	Autre propriété (nom : valeur)
Button1	btImport	Import du fichier	Default : True
Frame1	grpTri	Tri	
OptionButton1	optCA	Par chiffres d'affaires	Value : True
OptionButton2	optReg	Par régions et chiffres d'affaires	
Button2	btFormat	Mise en forme	
Button3	btCancel	Quitter	Cancel : True

Ceci demande probablement un peu d'explications. La propriété *Name* est le nom du contrôle, tel qu'il sera référencé dans le code. La propriété *Caption* est le libellé du contrôle. La propriété *Default*, fixée à True pour un seul contrôle, définit celui qui sera actif si l'utilisateur appuie immédiatement sur *Entrée*. La propriété *Value* d'un bouton d'option, fixée à *True* ou *False*, définit si celui-ci est coché ou non par défaut. Enfin, la propriété *Cancel* fixée à *True* du bouton *Quitter* lui attribue son effet.



6 Il faut maintenant ajuster la mise en forme du formulaire. Le cadre n'est pas assez large pour afficher correctement les libellés des boutons d'option : agrandissez-le en le sélectionnant et en agissant sur les flèches de sa bordure. Agrandissez ensuite la largeur du deuxième bouton d'option, jusqu'à afficher son libellé sur une seule ligne. Alignez maintenant les contrôles sur le formulaire. Appuyez sur la touche *Ctrl*, puis sélectionnez successivement sans relâcher la touche sur le bouton du haut, le cadre et celui du bas. Dans la barre de menu, choisissez *Format* > *Aligner sur la feuille* > *Horizontalement*. Cliquez sur le formulaire en dehors de tout contrôle, appuyez à nouveau sur *Ctrl* et en maintenant la touche *Ctrl* enfoncée, sélectionnez les deux boutons d'options du cadre. Dans la barre de menu, choisissez *Format* > *Aligner* > *À gauche*. Les contrôles sont maintenant parfaitement disposés sur le formulaire.

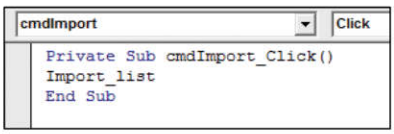


Copyright © 2012 Dunod.

7 L'interface semble correcte. Vérifiez-le en appuyant sur *F5* : l'aspect est effectivement satisfaisant, mais cliquer sur n'importe lequel contrôle ne donne aucun résultat : il faut doter les contrôles de procédures gérant leurs événements. Cliquez sur la croix rouge située en haut à droite pour revenir à l'Éditeur Visual Basic.



8 En principe, il faudrait écrire ces procédures. Heureusement, nous les avons déjà rédigées dans les TP précédents et elles sont déjà présentes dans le module 1 (après parfois de légères modifications). Il suffira donc d'y faire appel depuis les procédures événementielles des contrôles. Double-cliquez sur le premier bouton de commande, *Importer*. Une nouvelle procédure *Sub btnImporter* apparaît dans la fenêtre *Code*. Saisissez simplement entre les deux instructions `Sub… End Sub` l'instruction `Import_List` : c'est un appel à la routine existante. Attention : en l'état actuel, la procédure `Import_List` ne fonctionne pas pour vous. Vous devez ajuster le chemin d'accès selon l'emplacement réel de stockage du fichier à importer sur votre configuration, ce qui est signalé par un commentaire dans la procédure. Le cas échéant, reportez-vous à votre classeur TP_38.xlsm pour copier-coller le bon chemin d'accès.



9 Double-cliquez sur le bouton de commande *Mise en forme*. Saisissez entre les deux instructions `Sub… End Sub` l'instruction `MiseEnForme` : là aussi, un appel à une routine existante. Avant de mettre en forme le tableau importé, il faut le trier selon l'option retenue dans le cadre Tri et les deux boutons d'option. Juste en dessous de l'instruction `Sub cmdFormat()`, saisissez l'instruction :

```
If Me.optCA.Value = True Then TriCA Else TriRegion.
```

Si le bouton d'option *Tri par chiffres d'affaires* est coché, il est effectué un tri par chiffres d'affaires. Sinon, c'est un tri par région et chiffres d'affaires.

```
Private Sub cmdFormat Click()
    If Me.optCA.Value = True Then TriCA Else TriRégion
    MiseEnForme
End Sub
```

10 Double-cliquez sur le bouton de commande *Quitter*. Saisissez entre les deux instructions `Sub… End Sub` l'instruction `Unload frmCA`. Cette instruction supprime de la mémoire le formulaire utilisateur, ne laissant que le classeur actif affiché (et l'Éditeur Visual Basic, si vous aviez travaillé à partir de celui-ci).

```
Private Sub cmdCancel_Click()
Unload frmCA
End Sub
```

Copyright © 2012 Dunod.

11 Testez maintenant le formulaire. Enregistrez le classeur, fermez l'Éditeur Visual Basic pour revenir à Excel. Dans le formulaire, cliquez une seule fois successivement sur les boutons *Importer la liste*, puis *Mise en forme* : la liste est importée et mise en forme après un tri par chiffres d'affaires. Cochez la case *Tri par région et chiffres d'affaires*, puis cliquez à nouveau sur *Mise en forme* : le tableau est trié par région et chiffres d'affaires et est mis en forme de façon adéquate. Cliquez sur *Quitter* : le formulaire disparaît, ne laissant que le classeur Excel visible (ou l'Éditeur Visual Basic).

	Nom	Région	Chiffre d'affaires	Classe
3	Yabisda	Centre	178 000 €	1
4	N'Gyen	Centre	120 000 €	4
5	Dupont	Est	80 000 €	6
6	Fougerolles	Est	60 000 €	6
7	Jasmin	Nord	135 000 €	3
8	Dufour	Nord	120 000 €	4
9	Crespin	Ouest	115 000 €	4
10	Dariseau	Ouest	99 500 €	5
11	Lavillois	Sud	165 000 €	2
12	Parpaillou	Sud	112 000 €	4
13	Estaque	Sud	107 000 €	4

12 L'interface utilisateur fonctionne à la perfection et pourtant, elle présente certains défauts. Affichez l'Éditeur Visual Basic, cliquez sur *F5* pour exécuter le formulaire, puis cliquez sur *Importer la liste* : une nouvelle liste apparaît à gauche du tableau existant, décalant celui-ci.

	Nom	Région	Chiffre d'affaires	Nom	Région	Chiffre d'affaires	Classe
3	Crespin	Ouest	115000	Yabisda	Centre	178 000 €	1
4	Dariseau	Ouest	99500	N'Gyen	Centre	120 000 €	4
5	Dufour	Nord	120000	Dupont	Est	80 000 €	6
6	Dupont	Est	80000	Fougerolles	Est	60 000 €	6
7	Estaque	Sud	107000	Jasmin	Nord	135 000 €	3
8	Fougerolles	Est	60000	Dufour	Nord	120 000 €	4
9	Jasmin	Nord	135000	Crespin	Ouest	115 000 €	4
10	Lavillois	Sud	165000	Dariseau	Ouest	99 500 €	5
11	N'Gyen	Centre	120000	Lavillois	Sud	165 000 €	2
12	Parpaillou	Sud	112000	Parpaillou	Sud	112 000 €	4
13	Yabisda	Centre	178000	Estaque	Sud	107 000 €	4

13 Pour y remédier, il faut (et suffit) de vider la feuille de calcul avant toute importation de liste. Ouvrez la procédure *cmdImport_Click* et ajoutez au début ce qui suit :

```
Feuil1.Rows("1:16").Select
Selection.Delete Shift:=xlUp
```

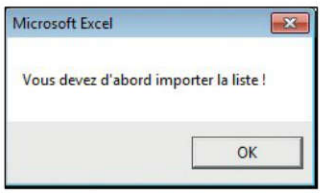
```
Private Sub cmdImport_Click()
    Feuil1.Rows("1:14").Select
    Selection.Delete Shift:=xlUp
    Import_list
End Sub
```

Copyright © 2012 Dunod.

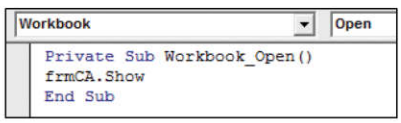
14 Enregistrez le classeur, placez-vous sur le formulaire, appuyez sur *F5* puis appuyez à plusieurs reprises sur *Importer la liste* : l'importation se déroule désormais parfaitement. De la même façon, il est possible d'appliquer un tri et une mise en forme avant d'avoir importé la liste. Cela n'est pas plus acceptable. Ouvrez la procédure *cmdFormat* et ajoutez au début ce qui suit :

```
If feuil1.range("B2").value = ""then
   MsgBox ("Vous devez d'abord importer la liste !")
   exit sub
    else
```

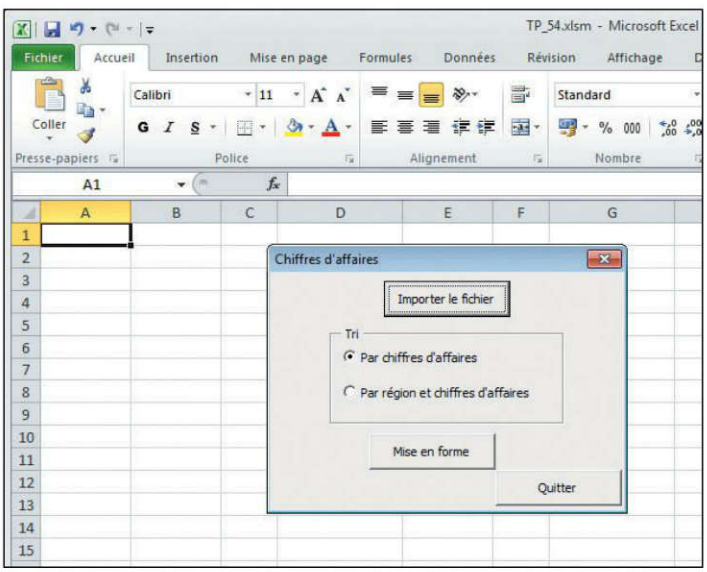
Ajoutez ensuite une instruction `End If` au-dessus de `End Sub`.
La procédure ne pourra désormais s'exécuter que si la cellule *B2* contient une valeur. Enregistrez le classeur, supprimez la liste actuelle puis testez le bouton *Mise en forme* : la procédure affiche une boîte de message signalant l'omission.



15 Il reste une dernière chose à mettre en place : l'affichage et le lancement automatique du formulaire lors du chargement du classeur. Dans l'Éditeur Visual Basic, dans la fenêtre *Code*, sélectionnez dans la colonne de gauche *Workbook* et saisissez, comme corps de la procédure *WorkBook_Open,* l'instruction `frmCA.Show`.



16 Enregistrez le classeur, fermez l'éditeur, puis fermez le classeur TP_54. Ouvrez-le à nouveau et activez le contenu : le formulaire s'affiche et vous permet d'importer et de mettre en forme la liste à votre guise.



Il serait possible d'améliorer encore cette interface et ses procédures. Par exemple, pour le moment, la taille de la liste est fixe : si le nombre de commerciaux ou de régions change, la mise en forme sera défectueuse. Comme le fichier doit toujours être importé avant d'être mis en forme, l'importation manuelle est superflue et pourrait s'effectuer automatiquement dans la procédure du bouton *Mise en forme*, ce qui éviterait d'ailleurs d'avoir à vérifier si elle a bien été exécutée. La feuille de calcul pourrait être datée automatiquement, et une possibilité d'impression automatique, avec en-tête et pied de page, proposée. Je laisse ces possibilités d'amélioration à votre sagacité. Si besoin est, vous trouverez dans les exemples un fichier TP_54_fini qui devrait correspondre à ce que vous devriez avoir obtenu à la fin de ce long TP.

Copyright © 2012 Dunod.



Astuces de programmation

Ce livre va s'achever sur quelques astuces de programmation, qui rendront votre code plus lisible et aisément compréhensible. Si l'éditeur VBA est doté d'options bien utiles, les autres démarches ne relèvent que de vous.

Prérequis : TP 33 à 54
Fichier employé : néant
Temps de réalisation : 5 minutes

1 Lorsqu'Excel enregistre une macro, il place avec pertinence des indentations là où cela est nécessaire. Elles servent généralement à indiquer ce qui appartient à un même ensemble (ou sous-ensemble). Cela contribue grandement à faciliter la lecture du code. Excel ne tient pas compte des indentations lorsqu'il exécute une procédure qui en contient. Lorsque vous rédigez vous-même le code d'une procédure, c'est à vous seul d'insérer les indentations adéquates. Utilisez la touche *TAB* pour ajouter une indentation à une ligne, et de la combinaison *Maj+TAB* pour supprimer une indentation. De même, servez-vous du caractère de saut de ligne (_) pour expliciter une procédure.

```
Function Test(a)
Test = IIf(a > 175000, 1, IIf(a > 150000, 2, IIf(a > 120000, 3, _
IIf(a > 100000, 4, IIf(a > 80000, 5, 6)))))
End Function
```

```
Function Test(a)
Test = IIf(a > 175000, 1, _
           IIf(a > 150000, 2, _
               IIf(a > 120000, 3, _
                   IIf(a > 100000, 4, _
                       IIf(a > 80000, 5, 6) _
                       ) _
                   ) _
               ) _
           )
End Function
```

2 Si l'enregistreur de macros insère automatiquement quelques commentaires, affichés en vert, c'est à vous de commenter votre propre code. Les instructions facilitent la compréhension du code tant pour vous-même que pour les autres. Ils sont également ignorés pendant l'exécution d'une procédure.
Pour créer un commentaire, dans la fenêtre *Code*, placez une apostrophe au début de la ligne de texte. Lorsque le curseur quitte cette ligne, celle-ci devient verte. Vous pouvez également placer des remarques en fin de ligne, ou à la fin de la procédure.

```
Sub MiseEnForme()
    Range("E2").Select
    ActiveCell.FormulaR1C1 = "Classe"
    Range("E3").Select
    ActiveCell.FormulaR1C1 = "=Test(RC[-1])"
    Selection.Copy
    Range("E3:E13").Select
    ActiveSheet.Paste
```

```
Sub MiseEnForme()
'
' Macro MiseEnForme
' Auteur Justin Bogue

' Ajout de l'étiquette de colonne Classe
    Range("E2").Select                    ' Sélection de la plage
    ActiveCell.FormulaR1C1 = "Classe"     ' Affectation de la valeur
```

Copyright © 2012 Dunod.

3 Condensez les instructions. Très souvent, vous modifiez les propriétés d'un même objet. Une fois celui-ci sélectionné, servez-vous de l'instruction `With Selection… End With` pour économiser de la saisie et rendre le code plus clair. Il est souvent possible de rassembler ainsi des instructions éparses.

```
'Tri du tableau par région et chiffre d'affaires
    Range("B2:E13").Select
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Add Key:=Range("C3:C13") _
        , SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
    ActiveWorkbook.Worksheets("Feuil1").Sort.SortFields.Add Key:=Range("D3:D13") _
        , SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal
    With ActiveWorkbook.Worksheets("Feuil1").Sort
        .SetRange Range("B2:E13")
        .Header = xlYes
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
```

```
'Tri du tableau par région et chiffre d'affaires
    Range("B2:E13").Select
    With ActiveWorkbook.Worksheets("Feuil1").Sort
        .SortFields.Clear
        .SortFields.Add Key:=Range("C3:C13"), SortOn:=xlSortOnValues, _
            Order:=xlAscending, DataOption:=xlSortNormal
        .SortFields.Add Key:=Range("D3:D13"), SortOn:=xlSortOnValues, _
            Order:=xlDescending, DataOption:=xlSortNormal
        .SetRange Range("B2:E13")
        .Header = xlYes
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
```

Enfin, dernier point et sans doute le plus important : n'hésitez pas à abuser de l'enregistreur de macros. Il est inutile de réinventer la roue. Très souvent, si vous ne vous souvenez pas de la façon d'effectuer une action particulière, enregistrez celle-ci dans une macro et servez-vous du code ainsi généré. Vous devrez souvent l'épurer en le débarrassant des instructions superflues, mais cela demeure un excellent moyen de vous familiariser facilement avec Visual Basic et d'en découvrir les membres et propriétés. Parfois, le seul examen du code généré procure la réponse à votre question…

Il était impossible d'aborder ici plus en détail la programmation VBA : un livre entier de cette taille serait à peine suffisant. Pour plus d'informations, reportez-vous à la documentation d'Excel et de Visual Basic ou à des sources extérieures, comme D. Maniez, *Formation à VBA Office 2010*, Dunod, 2011.

Copyright © 2012 Dunod.

Copyright © 2012 Dunod.

Index

A

Accepter ou refuser les modifications, boîte de dialogue 35
antécédents 46

B

boîte de message 115
Bouton de commande, contrôle ActiveX 111
Bouton d'option, contrôle ActiveX 111

C

Cadre,, contrôle ActiveX 111
Caption, propriété 111
certificat numérique 47
classeur
  authentifier 47
  créer dans Excel Web App 57
  envoyer par messagerie électronique 49
  exécuter une macro à l'ouverture 79
  inspecter 36
  marquer comme final 38
  ouvrir dans Excel 56
  ouvrir dans Excel depuis Skydrive 58
  partager 28
  partager dans Skydrive 63
  protéger 39
  stocker sur Skydrive 54
collection 65
commentaire
  Excel 31
  VBA 116
connexion réseau vers Skydrive 60

D

dépendants 46

E

Éditeur Visual Basic
  afficher 82
  Boîte à outils 110
  fenêtre Espions 103
  fenêtre Exécution 105
enregistrement
  arrêter 70, 71
  lancer 70
Espion express 103
Excel Web App
  créer un classeur 57
  ouvrir un classeur 56
exécution
  fenêtre 105
  pas à pas détaillé 98
  point d'arrêt 102

F

feuille de calcul
  protéger 41
  protéger une partie de 43
fichier texte, importer 79
fonction 66
  intégrée, améliorer 106
  paramètres facultatifs 107
formules
  afficher les relations 45
  antécédents 46
  dépendants 46

G

Gestionnaire de scénario 1, 17

H

Historique des modifications, feuille 35

I

identifiant Windows Live 52
If Then Else, instruction 113
indentation 116
Inspecteur de document 37
interface utilisateur, créer 110

M

macro
  absolue 73
  éditer 81

Copyright © 2012 Dunod.

emplacement de stockage 76
enregistrer 69
exécuter à l'ouverture d'un classeur 79
propriétés, modifier 74
membre 66
méthode 65
modifications des utilisateurs 33, 34
accepter ou refuser 35
module 83
importer et exporter 108
supprimer ou ajouter 85
mot de passe 39
MsgBox, instruction 115

N

Name, propriété 111

O

onglet Développeur, afficher 68

P

Paramètres du solveur, boîte de dialogue 5
partage de classeur 28
protéger 29, 30
pas à pas détaillé 98
point d'arrêt 102
procédure 66
programmation
commentaire 116
indentation 116
propriété 66
protéger
classeur 39
feuille de calcul 41
partie de feuille de calcul 43

S

scénario
gestionnaire de 17
Solveur 11
Show, méthode 115
signature numérique 48
Skydrive
connexion réseau vers 60
ouvrir un classeur dans Excel 58
partager un classeur 63
partager un dossier 62
stocker un classeur 54
Solveur 1
algorithmes de résolution 15
Cellules variables 5
employer 5
installer 3
Modèle de problème 11
rapports 9
solutions intermédiaires, afficher 7
sous-routine 66

T

table de données 2, 22

U

Unload, instruction 113
UserForm 110
Utilitaire d'analyse 2
Moyenne mobile 25
Statistiques descriptives 25

V

Valeur cible 2, 20