

Version en ligne  
**OFFERTE !**  
pendant 1 an

# Le langage DAX

Maîtrisez l'analyse et la modélisation  
de données dans Power BI et Excel

concepts clés, notions avancées et exercices pratiques

## En téléchargement



sources

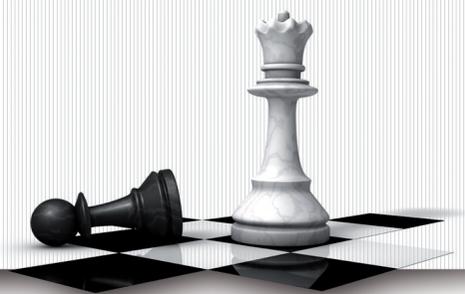


exemples Excel  
et Power BI



exercices  
et corrigés

André MEYER-ROUSSILHON



Solutions Business



## Chapitre 1

## Introduction

A. Introduction . . . . .	13
B. Le DAX en trois mots. . . . .	14
C. La présentation des fonctions . . . . .	15
D. Que faut-il attendre de cet ouvrage ? . . . . .	17
E. Les sources d'information sur le DAX. . . . .	17

## Chapitre 2

## Mise en route

A. Mise en route dans Excel. . . . .	21
1. Installation de Power Pivot. . . . .	21
2. Importation des tables. . . . .	22
3. Rôle et sens de la relation . . . . .	26
B. Mise en route dans Power BI . . . . .	28
1. Importation des tables. . . . .	28
2. Rôle et sens de la relation . . . . .	31
C. Modèle exemple . . . . .	33
D. Exercices . . . . .	36
1. Les relations et leur sens. . . . .	36
E. Corrigés . . . . .	37
1. Les relations et leur sens. . . . .	37
a. Dans Excel. . . . .	37
b. Dans Power BI . . . . .	38

## Chapitre 3

## Les principes fondamentaux du DAX

A. Introduction . . . . .	41
B. Les conventions d'écriture du DAX . . . . .	41
C. Les mesures et les colonnes . . . . .	43
1. Créer une mesure avec SUM, SUMX et RELATED . . . . .	44
2. Créer une colonne . . . . .	47
D. Le premier script DAX : créer une table du temps avec Power BI . . . . .	48
E. Créer une table de dates avec Power Pivot . . . . .	51

F.	Création de visuels avec Power BI	53
1.	Créer un tableau	53
2.	Créer un tableau croisé	54
3.	Créer une carte	55
4.	Créer un segment	56
G.	Création de visuels avec Excel et Power Pivot	57
1.	Créer un tableau croisé dynamique	57
2.	Créer une carte (donnée unique)	59
3.	Créer un segment	60
H.	Le contexte de filtre et le contexte de ligne	62
1.	Le contexte de ligne	62
2.	Le contexte de filtre	63
3.	Les relations à double sens et la propagation du filtre avec CROSSFILTER	67
4.	La transition de contexte avec RELATEDTABLE	70
5.	Exemple commenté de formule complexe	72
I.	La fonction CALCULATE	75
1.	La syntaxe	75
2.	Les trois façons de modifier le contexte de filtre (remplacer, ajouter, supprimer)	76
J.	L'éditeur de DAX de Power BI	79
K.	Exercices	81
1.	Les mesures, les colonnes	81
a.	Créer des mesures dans le fichier librairie	81
b.	Créer des mesures dans le fichier or	81
2.	Créer la table du temps avec Power BI	81
3.	Créer la table du temps avec Excel et Power Pivot	81
4.	Les conventions d'écriture du DAX	84
a.	Mettre en forme une formule DAX : exemple 1	84
b.	Mettre en forme une formule DAX : exemple 2	84
5.	Créer les visuels de base	85
a.	Dans Power BI	85
b.	Dans Excel	85
6.	CALCULATE	85
a.	Montant BD	85
b.	Différence 2019 / 2018	85
c.	Pourcentage	85
d.	Montant moyen	86

L. Corrigés .....	86
1. Les mesures, les colonnes .....	86
a. Créer des mesures dans le fichier librairie .....	86
b. Créer des mesures dans le fichier or .....	87
2. Créer la table du temps avec Power BI .....	87
3. Créer la table du temps avec Excel et Power Pivot .....	87
4. Les conventions d'écriture du DAX .....	88
a. Mettre en forme une formule DAX .....	88
b. Mettre en forme une formule DAX .....	89
5. Créer les visuels de base .....	89
a. Dans Power BI .....	89
b. Dans Excel .....	89
6. CALCULATE .....	89
a. Le montant pour la catégorie BD .....	89
b. Différence entre le montant 2019 et le montant 2018 .....	90
c. Pourcentage de chaque catégorie par rapport au total .....	90
d. Le montant moyen par commande .....	91

## Chapitre 4

### Utiliser les variables

A. Rôle et déclaration des variables avec VAR .....	95
B. Exemple de lisibilité facilitée et de performance améliorée .....	96
C. Correction de formules avec les variables .....	98
D. Les variables et le contexte de filtre .....	100
E. Déclarer une variable à l'intérieur de la formule .....	102
F. En résumé .....	103
G. Exercices .....	104
1. Renforcer la lisibilité et améliorer la performance .....	104
2. Débugger une formule .....	105
3. Jouer sur le contexte de filtre .....	106
H. Corrigés .....	107
1. Renforcer la lisibilité et améliorer la performance .....	107
2. Débugger une formule .....	109
3. Jouer sur le contexte de filtre .....	111

## Chapitre 5

## Agrégation et itération

A. Introduction .....	117
B. Pourquoi systématiquement agréger un nombre .....	117
C. Les fonctions d'agrégation (SUM, AVERAGE...) .....	120
1. Généralités .....	120
2. Une utilisation majeure de la fonction MAX .....	121
3. Compter avec COUNT et COUNTROWS .....	122
D. Les fonctions d'itération pour agréger (SUMX, AVERAGEX, ...) .....	124
1. Généralités .....	124
2. Calculer la moyenne mobile avec AVERAGEX .....	126
3. La fonction RANKX .....	130
4. La fonction CONCATENATEX .....	134
E. Exercices .....	137
1. Les fonctions d'agrégation .....	137
a. Calculer le cours de l'or à fin de semaine .....	137
b. Compter le nombre de ... .....	139
2. Les fonctions d'itération pour agréger .....	139
a. Calculer le montant moyen de la commande pour chaque mois. ....	139
b. Calculer le montant moyen des ventes pour les trois meilleurs livres de chaque catégorie .....	140
c. La fonction RANKX .....	140
d. La fonction CONCATENATEX .....	141
F. Corrigés .....	141
1. Les fonctions d'agrégation .....	141
a. Calculer le cours de l'or à fin de semaine .....	141
b. Compter .....	143
2. Les fonctions d'itération pour agréger .....	145
a. Calculer le montant moyen de la commande pour chaque mois. ....	145
b. Calculer le montant moyen des ventes pour les trois meilleurs livres de chaque catégorie .....	146
c. La fonction RANKX .....	147
d. La fonction CONCATENATEX .....	149

## Chapitre 6

## CALCULATE et les modifications du filtre

A. Introduction .....	153
B. Les principes de CALCULATE .....	153
1. La syntaxe .....	153
2. Comment CALCULATE modifie le contexte de filtre .....	155
C. Les arguments de filtres complexes (AND, OR) .....	156
1. Sur une colonne .....	156
2. Sur plusieurs colonnes .....	158
D. Les fonctions de filtre : FILTER et KEEPFILTERS .....	161
1. FILTER .....	161
2. KEEPFILTERS .....	163
E. Les fonctions ALL* .....	164
1. ALL, ALLEXCEPT .....	165
2. ALLSELECTED .....	166
F. Ordre des opérations dans CALCULATE .....	168
G. Exercices .....	170
1. Les principes de CALCULATE .....	170
a. Combien de livres par catégorie ? .....	170
b. Comprendre le contexte de filtre .....	170
2. Les arguments de filtres complexes .....	171
3. Les fonctions de filtre : FILTER .....	172
4. Les fonctions ALL, ALLEXCEPT et ALLSELECTED .....	173
5. Trois exercices sur CALCULATE .....	173
a. Les clients nantais .....	173
b. Le nombre de jours où il y a eu des commandes .....	174
c. Les quantités toutes années et toutes catégories .....	174
H. Corrigés .....	175
1. Les principes de CALCULATE .....	175
a. Combien de livres par catégorie ? .....	175
b. Comprendre le contexte de filtre .....	176
2. Les arguments de filtres complexes .....	177
3. Les fonctions de filtre : FILTER .....	178
4. Les fonctions ALL, ALLEXCEPT et ALLSELECTED .....	179
5. Trois exercices sur CALCULATE .....	180
a. Les clients nantais .....	180
b. Le nombre de jours où il y a eu des commandes .....	180
c. Les quantités toutes années et toutes catégories .....	181

## Chapitre 7

## Les fonctions logiques et les fonctions d'information

A. Introduction . . . . .	185
B. Le traitement conditionnel et les fonctions logiques . . . . .	186
1. La fonction IF et ses variantes . . . . .	186
2. La gestion des erreurs avec IFERROR . . . . .	190
3. COALESCE . . . . .	190
C. Les fonctions d'information . . . . .	192
1. ISBLANK, ISEMPTY . . . . .	192
2. FILTERS, VALUES . . . . .	194
3. ISFILTERED, ISCROSSFILTERED . . . . .	196
4. HASONEVALUE, SELECTEDVALUE, ISINSCOPE . . . . .	196
D. Exercices . . . . .	200
1. Le traitement conditionnel et les fonctions logiques . . . . .	200
2. Les fonctions d'information . . . . .	201
a. FILTERS, VALUES . . . . .	201
b. HASONEVALUE, SELECTEDVALUE, ISINSCOPE . . . . .	202
E. Corrigés . . . . .	202
1. Le traitement conditionnel et les fonctions logiques . . . . .	202
2. Les fonctions d'information . . . . .	204
a. FILTERS, VALUES . . . . .	204
b. HASONEVALUE, SELECTEDVALUE, ISINSCOPE . . . . .	205

## Chapitre 8

## Les fonctions de manipulation du modèle physique

A. Introduction . . . . .	209
B. Les fonctions de table . . . . .	210
1. FILTER, ALL . . . . .	211
2. CALCULATETABLE . . . . .	212
3. VALUES, DISTINCT . . . . .	213
4. ADDCOLUMNS . . . . .	216
5. SUMMARIZE, CROSSJOIN . . . . .	218
6. Un mot sur les tables d'agrégation . . . . .	221
7. SELECTCOLUMNS . . . . .	223
8. UNION / INTERSECT / EXCEPT . . . . .	224
9. GENERATESERIES . . . . .	229

C. Les fonctions de relation	230
1. Les types de relations	230
2. RELATED / RELATEDTABLE	233
3. USERRELATIONSHIP	234
4. CROSSFILTER	235
5. TREATAS	238
6. LOOKUPVALUE	242
D. Les fonctions de conversion	245
1. CONVERT	245
2. FORMAT	246
3. VALUE	247
4. DATE	247
5. TIME	247
E. Exercices	249
1. Les fonctions de table	249
a. Table des livres de littérature	249
b. Table des villes et quantité	250
c. Table des combinaisons ville-genre et montant	251
d. Agrégat année-moyen de paiement	252
e. Quels genres promouvoir ?	252
2. Les fonctions de relation	252
a. USERRELATIONSHIP	252
b. CROSSFILTER	254
c. TREATAS	255
F. Corrigés	255
1. Les fonctions de table	255
a. Table des livres de littérature	255
b. Table des villes et quantité	257
c. Table des combinaisons ville-genre et montant	258
d. Agrégat année-moyen de paiement	259
e. Quels genres promouvoir ?	259
2. Les fonctions de relation	260
a. USERRELATIONSHIP	260
b. CROSSFILTER	261
c. TREATAS	263

## Chapitre 9

## Les fonctions de date et la table du temps

A. Créer la table du temps .....	267
1. Faut-il toujours ajouter une table du temps ? .....	267
2. Une ou plusieurs tables ? .....	268
3. CALENDARAUTO, CALENDAR et le script DAX .....	270
B. Travailler avec les semaines (WEEKNUM) .....	272
1. Les normes de calcul de la semaine .....	272
2. Le calcul de la semaine en norme ISO 8601 .....	273
C. Travailler avec les heures .....	273
D. Les fonctions de date .....	276
1. DATEDIFF .....	276
2. NOW, TODAY .....	277
3. DAY, MONTH, QUARTER, YEAR .....	277
E. CALCULATE et les fonctions de Time Intelligence .....	277
1. Les briques de base de la Time Intelligence .....	278
2. Les fonctions de Time Intelligence pour cumuler (DATESYTD, TOTALYTD) .....	279
3. Le cas particulier du cumul hebdomadaire .....	280
4. La moyenne mobile avec AVERAGEX .....	281
5. La moyenne mobile avec DATESINPERIOD .....	282
6. La variation avec DATEADD .....	283
7. DATEADD, PARALLELPERIOD et SAMEPERIODLASTYEAR .....	286
8. La tendance (ou saisonnalité) .....	289
9. Les autres fonctions de Time Intelligence en deux mots .....	290
F. Exercices .....	291
1. Créer la table du temps .....	291
2. Les fonctions de date .....	291
a. DATEDIFF .....	291
3. CALCULATE et les fonctions de Time Intelligence .....	292
a. Calculer la quantité l'année précédente (A-1) .....	292
b. Calculer un cumul annuel de la quantité .....	292
c. Calculer un total trimestriel glissant de la quantité .....	293
d. Calculer un cumul perpétuel de la quantité .....	293
e. Calculer la variation par rapport au mois précédent .....	294
f. Le cumul annuel à date .....	294
g. Calculer la moyenne mobile des montants à 7 jours .....	295

G. Corrigés .....	295
1. Créer la table du temps .....	295
2. Les fonctions de date .....	298
a. DATEDIFF .....	298
3. CALCULATE et les fonctions de Time Intelligence .....	300
a. Calculer la quantité l'année précédente (A-1) .....	300
b. Calculer un cumul annuel de la quantité .....	301
c. Calculer un total trimestriel glissant de la quantité .....	301
d. Calculer un cumul perpétuel de la quantité .....	302
e. Calculer la variation par rapport au mois précédent .....	303
f. Le cumul annuel à date .....	304
g. Calculer la moyenne mobile des montants à 7 jours .....	304

## Chapitre 10

### Aperçu des autres familles de fonctions

A. Introduction .....	307
B. Les fonctions de manipulation du texte .....	308
C. Les fonctions d'analyse financière .....	310
D. Les fonctions mathématiques .....	313
E. Les fonctions statistiques .....	315

Conclusion .....	321
Index .....	323

## A. Introduction

SI vous souhaitez approfondir vos connaissances sur la fonction CALCULATE, je vous conseille mon précédent ouvrage, Power BI Desktop : Renforcer, approfondir, explorer, aux Éditions ENI, dont le chapitre 3 décrit notamment des motifs courants et des analyses fréquentes rendues possibles avec CALCULATE. Je ne reviendrai pas sur ces points, mais je résume les caractéristiques essentielles de cette fonction et j'ajoute ici d'autres aspects de son fonctionnement.

## B. Les principes de CALCULATE

Vous l'avez vu depuis le début de cet ouvrage, la fonction CALCULATE revient tout le temps. La raison en est simple, et fondamentale : CALCULATE a le pouvoir de modifier le contexte de filtre généré par le visuel, les segments, le rapport en général.

### 1. La syntaxe

Elle est très simple :

```
CALCULATE ( <Expression> [ , <Filtre1> ] [ , <Filtre2> ] [ , ... ] )
```

Mais ceci amène à de nombreuses remarques.

#### CALCULATE et la transition de contexte

Tout d'abord, CALCULATE peut s'utiliser avec pour seul argument l'expression qu'il s'agit de calculer. En effet, c'est un rôle second de la fonction : lorsqu'elle est invoquée dans un contexte de ligne (c'est-à-dire lors de la création d'une colonne ou lors de son utilisation dans une fonction itérative – SUMX par exemple), la formule CALCULATE(<expression>) déclenche une **transition de contexte**, et la ligne dans son intégralité devient un filtre, qui se propage donc aux tables du modèle.

Ce phénomène est loin d'être rare : en fait, il est essentiel de se rappeler qu'à chaque fois qu'une mesure est appelée dans une formule, un CALCULATE implicite est systématiquement ajouté.

Prenons un exemple. La mesure [Montant facturé] est définie par la formule :

```
[Montant facturé] = SUM(Ventes[Montant])
```

Si l'on souhaite connaître le montant moyen facturé chaque jour, la formule s'écrit :

```
[Montant moyen facturé] =
AVERAGEX (
    Date[Date] ,
    [Montant facturé]
)
```

Soit, en développant la formule en incluant le CALCULATE implicite :

```
[Montant moyen facturé] =
AVERAGEX (
    Date[Date] ,
    CALCULATE([Montant facturé])
)
```

CALCULATE étant invoqué dans le cadre d'un contexte de ligne (dû à la fonction AVERAGEX, itérateur), la transition de contexte est déclenchée, et la date est donc propagée à la table `Ventes`. Celle-ci est donc filtrée, il ne reste que les lignes de la table `Ventes` correspondant à la date du contexte de ligne. C'est ce qui permet le calcul de la moyenne, les montants restants sont additionnés, puis le résultat est divisé par le nombre de lignes.

Exercice : pour mettre en pratique, vous pouvez réaliser l'exercice *Combien de livres par catégories ?* à la fin de ce chapitre.

### Un nombre de filtres non limité

La seconde remarque tient au nombre de filtres de CALCULATE : vous pouvez en ajouter autant que vous le souhaitez. Les filtres sont tous traités simultanément (l'ordre des filtres n'a donc aucune importance) – mais nous reviendrons sur cette notion, lorsque nous distinguerons les filtres proprement dits et les modificateurs.

Un filtre dans CALCULATE peut être une table (c'est-à-dire le résultat d'une fonction de table, autrement dit une liste de valeurs), ou un prédicat booléen (Vrai/Faux).

Voici un exemple du premier cas, retournant une liste de valeurs :

```
calculate FILTER =
CALCULATE (
    [montant] ,
    FILTER (
        produits ,
        produits[couleur] = "Rouge"
    )
)
```

Voici un exemple du second, retournant vrai ou faux :

```
CALCULATE (
    [Montant] ,
    'Produit'[Couleur] = 'Rouge'
)
```

Il est utile et important de noter que cette dernière syntaxe est un raccourci, simple à écrire, qui est systématiquement développé par DAX en :

```
CALCULATE (
    [Montant] ,
    FILTER (
        ALL(Produit),
        'Produit'[Couleur] = 'Rouge'
    )
)
```

Le filtre booléen Vrai/Faux cache la fonction de table FILTER, dont le résultat est une liste de valeurs.

Attention, la première et la deuxième expression ne produisent pas toujours le même résultat. Un exercice vous permettra de creuser ce point.

## 2. Comment CALCULATE modifie le contexte de filtre

Je vous renvoie et je vous encourage à relire la section consacrée à ce point dans le chapitre Les principes fondamentaux du DAX du présent ouvrage (La fonction CALCULATE - Les trois façons de modifier le contexte de filtre (remplacer, ajouter, supprimer)). Il est essentiel que vous compreniez bien ces notions.

En résumé, CALCULATE agit de trois manières :

- ▶ Remplacer le contexte de filtre : lorsque le filtre explicite donné comme argument de la fonction CALCULATE porte sur la même colonne que celle qui définit le contexte implicite (celui que génère le rapport), CALCULATE remplace ce dernier par le premier.
- ▶ Ajouter au contexte de filtre : lorsque le filtre explicite porte sur une nouvelle colonne, le nouveau filtre est ajouté au filtre implicite.

- ▶ Supprimer le contexte de filtre : c'est le rôle de la fonction ALL et de ses variantes.

Exercice : pour mettre en pratique, vous pouvez réaliser l'exercice Les arguments de filtres complexes à la fin de ce chapitre.

## C. Les arguments de filtres complexes (AND, OR)



*Il existe de nombreux cas de figure : vous en trouverez à la fin de cette section un tableau récapitulatif simplifié.*

### 1. Sur une colonne

Par filtre complexe, il faut comprendre un filtre portant plusieurs fois sur la même colonne, pour encadrer des dates par exemple ou encore, indiquer plusieurs catégories.

Deux cas se présentent : celui où les deux conditions doivent s'appliquer simultanément et celui où l'une ou l'autre des conditions doit s'appliquer.

Lorsque la date doit être comprise entre deux bornes, alors les conditions doivent s'appliquer simultanément. C'est ce qui se passe lorsque vous indiquez plusieurs filtres dans CALCULATE :

```
montant janv-mars 2019 =
CALCULATE (
    [montant] ,
    Datum[Date] >= DATE ( 2019 , 01 , 01 ) ,
    Datum[Date] <= DATE ( 2019 , 03 , 31 )
)
```

Annee mois	montant	montant janv-mars 2019
201901	€ 8 621,00	24 243,00 €
201902	€ 8 278,00	24 243,00 €
201903	€ 7 344,00	24 243,00 €
201904	€ 5 420,00	24 243,00 €
201905	€ 5 539,00	24 243,00 €
201906	€ 7 345,00	24 243,00 €
201907	€ 7 329,00	24 243,00 €
201908	€ 5 535,00	24 243,00 €
201909	€ 5 823,00	24 243,00 €
201910	€ 5 039,00	24 243,00 €
201911	€ 5 315,00	24 243,00 €
201912		24 243,00 €
<b>Total</b>	<b>€ 71 588,00</b>	<b>24 243,00 €</b>



*Il n'est sans doute pas inutile de rappeler que la table **Datum** ayant été marquée comme table de dates, le contexte de filtre induit par le visuel (donc ici le champ **Annee mois**) est ramené à un filtre sur un ensemble de dates (donc sur le champ **Date**). Dans le **CALCULATE** ci-dessus, le filtre porte lui aussi sur le champ **Date**, il remplace donc le contexte de filtre existant. C'est la raison pour laquelle nous retrouvons le même montant sur toutes les lignes du tableau. Ceci explique également pourquoi la ligne 201912 apparaît : bien qu'il n'y ait pas de montant ce mois-là, puisqu'il y en a un pour janv-mars, cette ligne est rajoutée au tableau.*

Par défaut, les deux conditions dans **CALCULATE** sont liées par un ET (AND). La formule précédente pourrait aussi s'écrire à l'aide de l'opérateur **&&** :

```
montant janv-mars 2019 =
CALCULATE(
    [montant] ,
    Datum[Date] >= DATE(2019,01,01)
    && Datum[Date] <= DATE(2019,03,31)
)
```

Elle peut aussi s'écrire avec l'opérateur **AND** :

```
montant janv-mars 2019 =
CALCULATE(
    [montant] ,
    AND(
        Datum[Date] >= DATE(2019,01,01) ,
        Datum[Date] <= DATE(2019,03,31)
    )
)
```



*Attention, l'opérateur **AND** (de même que **OR**) n'accepte que deux arguments. Si votre filtre doit porter sur trois conditions ou plus, alors privilégiez **&&**.*

Le cas inverse est celui où l'une ou l'autre des conditions doit s'appliquer. Dans ce cas, c'est l'opérateur **OU** (**OR**) qui doit lier les conditions ( symbole **|** , appelé *double pipe*, c'est-à-dire les touches **AltGr** **6** ) :

```
montant produits verts ou turquoises =
CALCULATE(
    [montant] ,
    produits[couleur] = "Vert"
    || produits[couleur] = "Turquoise"
)
```

La formule précédente pourrait aussi s'écrire à l'aide de l'opérateur IN :

```
montant produits verts ou turquoises =
CALCULATE(
    [montant] ,
    produits[couleur] IN { "Vert" , "Turquoise" }
)
```



*Notez bien les accolades pour énumérer la liste des catégories.*

Elle peut aussi s'écrire avec l'opérateur OR :

```
montant produits verts ou turquoises =
CALCULATE(
    [montant] ,
    OR(
        produits[couleur] = "Vert" ,
        produits[couleur] = "Turquoise"
    )
)
```



*Attention, l'opérateur OR n'accepte que deux arguments. Si votre filtre doit porter sur trois conditions ou plus, alors privilégiez | |.*

## 2. Sur plusieurs colonnes

Lorsqu'il s'agit de faire porter les filtres sur plusieurs colonnes, de la même table ou de tables différentes, il faut à nouveau distinguer le cas où ils s'appliquent simultanément ou si l'un ou l'autre doit s'appliquer.

Dans le premier cas, l'écriture reste simple :

```
montant produits verts en 2019 =
CALCULATE(
    [montant] ,
    produits[couleur] = "Vert" ,
    Datum[Annee] = "2019"
)
```

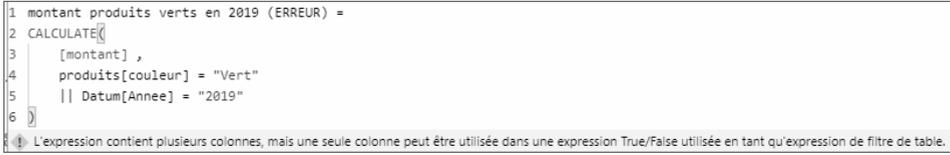
201810	€ 5 928,00	
201811	€ 6 338,00	
201812	€ 4 270,00	
201901	€ 8 621,00	924,00 €
201902	€ 8 278,00	386,00 €
201903	€ 7 344,00	330,00 €
201904	€ 5 420,00	258,00 €

En revanche, l'utilisation de && n'est plus possible. La formule ci-dessous génère une erreur :

```
montant produits verts en 2019 (ERREUR) =
CALCULATE(
    [montant] ,
    produits[couleur] = "Vert"
    && Datum[Annee] = "2019"
)
```

Lorsque l'une ou l'autre condition doit être appliquée (opérateur OU), la syntaxe simple n'est pas non plus possible, et génère la même erreur que précédemment :

```
montant produits verts ou 2019 (ERREUR) =
CALCULATE(
    [montant] ,
    produits[couleur] = "Vert"
    || Datum[Annee] = "2019"
)
```



```
1 montant produits verts en 2019 (ERREUR) =
2 CALCULATE(
3     [montant] ,
4     produits[couleur] = "Vert"
5     || Datum[Annee] = "2019"
6 )
L'expression contient plusieurs colonnes, mais une seule colonne peut être utilisée dans une expression True/False utilisée en tant qu'expression de filtre de table.
```

Dans ce cas, il est nécessaire de recourir à la syntaxe développée, complexe, notamment si les deux colonnes ne proviennent pas de la même table :

```
montant produits verts ou 2019 =
CALCULATE(
    [montant] ,
    FILTER(
        CROSSJOIN(
            ALL(produits[couleur]) ,
            ALL(Datum[Annee])
        ) ,
    produits[couleur] = "Vert"
    || Datum[Annee] = "2019"
)
```

Ici, la fonction CROSSJOIN retourne une table avec toutes les combinaisons possibles des deux colonnes.