

Les fonctions DAX de base

Data : **SMMEC_Banque2.xlsx**

1. SUM – Calcul de la somme des montants

La fonction **SUM** permet de calculer la somme des valeurs numériques d'une colonne.

Exemple : Somme totale des montants de transactions

```
Total_Montant = SUM(SMMEC_Banque2[Montant])
```

Cette mesure retourne la somme totale de toutes les transactions effectuées.

Exemple de résultat :

Si la somme des montants des transactions est de **37 610**, alors la mesure retournera cette valeur.

2. COUNT – Nombre de transactions enregistrées

La fonction **COUNT** compte le nombre total de lignes où la colonne spécifiée n'est pas vide.

Exemple : Nombre total de transactions effectuées

```
Nombre_Transactions = COUNT(SMMEC_Banque2[Montant])
```

Cette mesure retourne le nombre total de transactions enregistrées.

Exemple de résultat :

Si le fichier contient **16 transactions**, alors la mesure retournera **16**.

3. DISTINCTCOUNT – Nombre de clients distincts

La fonction **DISTINCTCOUNT** compte le nombre de valeurs uniques dans une colonne.

Exemple : Nombre de clients uniques ayant effectué au moins une transaction

```
Nombre_Clients = DISTINCTCOUNT(SMMEC_Banque2[Client_ID])
```

Cette mesure retourne le nombre de clients distincts.

Exemple de résultat :

Si **7 clients** ont effectué des transactions, alors la mesure retournera **7**.

4. AVERAGE – Moyenne des montants des transactions

La fonction **AVERAGE** calcule la moyenne des valeurs numériques d'une colonne.

Exemple : Moyenne des montants des transactions

```
Moyenne_Montant = AVERAGE (SMMEC_Banque2 [Montant])
```

Cette mesure retourne la moyenne des montants des transactions.

Exemple de résultat :

Si la somme totale des transactions est **37 610** et qu'il y a **16 transactions**, alors la moyenne sera **2 350,63**.

5. MAX et MIN – Valeur maximale et minimale des montants

Les fonctions **MAX** et **MIN** permettent de trouver respectivement la valeur la plus grande et la plus petite dans une colonne.

Exemple : Montant maximum et minimum des transactions

```
Montant_Max = MAX (SMMEC_Banque2 [Montant])  
Montant_Min = MIN (SMMEC_Banque2 [Montant])
```

Ces mesures retournent respectivement la valeur la plus élevée et la plus faible des transactions.

Exemple de résultat :

- Si le montant le plus élevé est **4 641**, alors **Montant_Max = 4 641**.
- Si le montant le plus faible est **103**, alors **Montant_Min = 103**.

6. CALCULATE – Filtrer et modifier le contexte d'une mesure

La fonction **CALCULATE** permet d'appliquer des filtres à une mesure.

Exemple : Somme des montants des transactions de type 'Dépôt'

```
Total_Depots = CALCULATE (  
    SUM (SMMEC_Banque2 [Montant]) ,  
    SMMEC_Banque2 [Type_Transaction] = "Dépôt"  
)
```

Cette mesure retourne la somme des transactions où le type est "Dépôt".

Exemple de résultat :

Si les transactions de type "Dépôt" totalisent **19 096**, alors la mesure retournera **19 096**.

7. COUNTROWS – Nombre total de lignes

La fonction **COUNTROWS** compte le nombre total de lignes dans une table.

Exemple : Nombre total d'enregistrements dans le fichier

```
Total_Lignes = COUNTROWS (SMMEC_Banque2)
```

Cette mesure retourne le nombre total d'enregistrements dans la table.

Exemple de résultat :

Si le fichier contient **16 lignes**, alors la mesure retournera **16**.

8. FILTER – Appliquer un filtre sur un ensemble de données

La fonction **FILTER** permet de créer un sous-ensemble de la table.

Exemple : Nombre de transactions effectuées en 2024 uniquement

```
Transactions_2024 = COUNTROWS (
    FILTER (SMMEC_Banque2,
    YEAR (SMMEC_Banque2[Date_Transaction]) = 2024)
)
```

Cette mesure retourne le nombre de transactions effectuées en 2024.

Exemple de résultat :

Si **toutes les transactions sont en 2024**, alors la mesure retournera **16**.

9. RELATED – Lier des tables via une relation

La fonction **RELATED** est utilisée pour récupérer des valeurs à partir d'une autre table liée.

Si une table **Clients** contenant des informations supplémentaires est liée via **Client_ID**, on peut récupérer les noms des clients :

Exemple : Récupérer le nom du client

```
Nom_Client = RELATED (Clients[Nom])
```

Cette mesure retourne le nom du client correspondant à chaque transaction.

10. SWITCH – Remplacer des valeurs dans une colonne

La fonction **SWITCH** est utilisée comme une alternative à **IF** pour plusieurs conditions.

Exemple : Catégorisation des transactions par type

```
Categorie_Transaction = SWITCH(  
    SMMEC_Banque2[Type_Transaction],  
    "Dépôt", "Entrée",  
    "Retrait", "Sortie",  
    "Virement", "Interne",  
    "Autre"  
)
```

Cette mesure retourne :

- "Entrée" pour les **Dépôts**
- "Sortie" pour les **Retraits**
- "Interne" pour les **Virements**
- "Autre" pour tout autre type

11. IF – Condition simple

La fonction **IF** permet d'évaluer une condition et de retourner un résultat.

Exemple : Vérifier si une transaction est un retrait

```
Est_Retrait = IF(SMMEC_Banque2[Type_Transaction] = "Retrait",  
"Oui", "Non")
```

Cette mesure retourne "**Oui**" si la transaction est un retrait, sinon "**Non**".

12. ALL – Ignorer les filtres

La fonction **ALL** permet de supprimer les filtres appliqués.

Exemple : Calculer la part de chaque transaction dans le total

```
Part_Transaction =  
    DIVIDE(SUM(SMMEC_Banque2[Montant]),  
    CALCULATE(SUM(SMMEC_Banque2[Montant]),  
    ALL(SMMEC_Banque2)))
```

Cette mesure retourne la part d'une transaction par rapport à la somme totale.

Avec ces fonctions DAX :

1. **SUM, COUNT, DISTINCTCOUNT** → Pour des statistiques de base.
2. **AVERAGE, MAX, MIN** → Pour des analyses plus approfondies.
3. **CALCULATE, FILTER, COUNTROWS** → Pour des analyses conditionnelles.
4. **SWITCH, IF, RELATED** → Pour des traitements de données plus avancés.
5. **ALL** → Pour ignorer les filtres et comparer les données.