

# TP 9 : Les requêtes imbriquées ( sous requêtes)

Il arrive parfois que l'on souhaite obtenir des informations pour lesquelles nous avons besoin au préalable d'autres informations.

Pour cela nous utilisons des sous-requête :

Une sous-requête est une requête **à l'intérieur** d'une autre requête. Avec le SQL, vous pouvez construire des requêtes imbriquées sur autant de niveaux que vous voulez. Vous pouvez également mélanger jointures et sous-requêtes. Tant que votre requête est correctement structurée, elle peut être aussi complexe que vous voulez.

## Sous-requête renvoyant une seule valeur

Par exemple, nous sommes capables d'afficher le prix moyen d'un article

"bio":

```
SELECT AVG(prix) AS 'prix moyen dun article bio'
FROM article
WHERE bio = True
```

Nous sommes également capables d'afficher les articles "bio" inférieurs ou égaux à un certain prix(x):

```
SELECT * FROM article
WHERE bio = TRUE AND prix <= x
```

Mais si on souhaite obtenir la liste des articles "bio" dont le prix est inférieur ou égal au prix moyen d'un article "bio", on va avoir besoin de mettre en œuvre une imbrication de requêtes :

- Pour cela, il faut commencer par faire la requête qui va nous permettre d'afficher tous les articles "bio" sans restriction de prix

```
SELECT * FROM article
WHERE bio = TRUE
```

- Ensuite faire la requête pour le prix moyen des articles bio

```
SELECT AVG(prix)
```

```
FROM article
WHERE bio = True
```

- Enfin choisir l'opérateur et assembler les deux requêtes

```
SELECT * FROM article
WHERE bio = TRUE
AND prix <= (SELECT AVG(prix)
             FROM article
             WHERE bio = TRUE)
```

Testez la requête

## Sous-requête renvoyant une liste de valeurs

Nous avons pu utiliser un opérateur de comparaison car notre sous-requête ne **renvoie** qu'une seule valeur.

Dans le cas où nous avons à utiliser une sous-requête renvoyant plusieurs lignes, il faut utiliser des **prédicats de listes ( le IN )** .

Nous souhaitons maintenant obtenir la liste des articles qui sont dans les mêmes rayons que "Maïs tendre" et "Café moulu":

Pour cela nous écrivons la requête permettant d'afficher la liste des

articles : `SELECT * FROM article`

Ensuite la requête permettant de ramener le numéro des rayons où sont disposés le « Mais tendre » et le « Café moulu »

```
SELECT idRayon FROM article
WHERE libelle IN ('Maïs tendre','Café moulu')
```

Enfin choisir le bon prédicat (**IN**) et assembler les requêtes :

```
SELECT * FROM article
WHERE idRayon IN (
    SELECT idRayon FROM article
    WHERE libelle IN ('Mais tendre','Café moulu'))
```

Testez la requête



Il est possible d'utiliser des jointures dans les sous-requêtes .

### **Exercice :**

Réaliser les requêtes permettant d'obtenir les informations ci-dessous.

- le ou les articles les moins chers. \*\*
- la liste des lots qui ont été livrés le même jour que le lot "L593078838". \*\*

· la liste des rayons qui ont été approvisionnés à la dernière livraison . \*\*\*\*

**Indice** : Il faut faire une sous requête dans la sous requête .

· la liste des rayons qui ont été approvisionnés à l'avant-dernière livraison. \*\*\*\*\*

**Indice** : Il faut faire une sous requête dans la sous requête de la sous requête..