

# Interroger une base - A L'AIDE D'UNE VUE

Cours en ligne : <https://openclassrooms.com/fr/courses/1959476-administrez-vos-bases-de-donnees-avec-mysql/1973943-vues>

## 1 Introduction à la commande `CREATE VIEW`

### Introduction

Les vues sont des objets de la base de données, constitués d'un **nom** et d'une **requête de sélection**.

Une fois qu'une vue est définie, on peut l'utiliser comme on le ferait avec une table qui serait constituée des données sélectionnées par la requête définissant la vue.

### Syntaxe générale de la commande `CREATE VIEW`

C'est très exactement le principe d'une vue : **on stocke une requête `SELECT` en lui donnant un nom**, et on peut ensuite appeler directement la vue par son nom.

Quelques remarques importantes :

- Il s'agit bien d'objets de la base de données, **stockés de manière durable**, comme le sont les tables ou les procédures stockées.
- C'est donc bien différent des requêtes préparées, qui ne sont définies que le temps d'une session, et qui ont un tout autre but.
- Ce qui est stocké est la **requête**, et non pas les résultats de celle-ci. **On ne gagne absolument rien en termes de performances** en utilisant une vue plutôt qu'en faisant une requête directement sur les tables.

### Création

Pour créer une vue, on utilise tout simplement la commande `CREATE VIEW`, dont voici la syntaxe :

```
CREATE view_name [ ( column_name [ ,...n ] ) ]
    AS
    select_statement
    [WITH CHECK OPTION];
```

### WITH CHECK OPTION.

Oblige toutes les instructions de modification de données exécutées sur la vue à respecter les critères définis dans `select_statement`. Lorsqu'une ligne est modifiée par l'intermédiaire d'une vue, WITH CHECK OPTION vérifie que les données resteront visibles dans la vue après validation de la modification. CHECK OPTION s'applique uniquement aux mises à jour effectuées par l'intermédiaire de la vue. Elle ne s'applique pas aux mises à jour effectuées directement dans les tables sous-jacentes d'une vue.

### Exemples :

```
CREATE VIEW [Brazil Customers] AS
SELECT CustomerName, ContactName
FROM Customers
WHERE Country = 'Brazil';
```

```
CREATE VIEW [Products Above Average Price] AS
SELECT ProductName, Price
FROM Products
WHERE Price > (SELECT AVG(Price) FROM Products);
```

**Opérations sur les vues :****DROP VIEW**

Une fois la vue créée, il est bien évidemment possible de la supprimer avec la commande *DROP VIEW*.

```
DROP VIEW view_name;
```

**ALTER VIEW**

Une fois la vue créée, il est bien évidemment possible de la modifier avec la commande *ALTER VIEW*.

```
ALTER VIEW view_name;
```

**REPLACE VIEW**

Une fois la vue créée, il est bien évidemment possible de la remplacer avec la commande *REPLACE VIEW*.

```
REPLACE VIEW view_name;
```

**Les droits nécessaires**

Pour créer une vue l'utilisateur doit avoir le droit *CREATE VIEW*. Il faut également avoir la permission de sélectionner toutes les colonnes qui apparaissent dans la commande *SELECT* spécifiant ce qu'est la vue.

De plus si la clause *REPLACE* est utilisée, le droit *DROP* est également nécessaire.

Le droit *SHOW VIEW* donne la possibilité d'exécuter la commande *SHOW CREATE VIEW*. Cette commande permet d'obtenir les informations de création d'une vue. Une autre façon d'obtenir ces informations est d'interroger la table **view** du schéma **information\_schema**. Cette information sera exhaustive seulement pour les vues que vous avez créé.

Accorder les droits sur une vue pour un utilisateur

```
GRANT
  SELECT,
  DROP,
  CREATE VIEW,
  SHOW VIEW
ON `projet`.* TO 'secretaire'@'localhost';
```

**Restrictions**

Lors de la création d'une vue, certaines contraintes doivent être prises en compte :

- Il n'est pas possible de créer un index sur une vue.
- La vue ne peut pas contenir de sous-requêtes dans la clause *FROM* du *SELECT*.
- Il n'est pas possible d'utiliser de variables dans une vue.
- Les objets (tables et vues) nécessaires à la création de la vue doivent exister avant de la créer.
- Si un objet référencé par la vue est effacé, la vue n'est alors plus accessible.
- Une vue ne peut référencer une table temporaire (*TEMPORARY TABLE*).
- Il n'est pas possible de créer des vues temporaires.
- Il n'est pas possible d'associer un **trigger** à une vue.
- La définition d'une vue est "**gelée**" dans une requête préparée.

**Conseils :**

Préfixer le nom de la vue par "V\_". Il s'agit d'une convention que l'on conseille fortement de respecter. Cela permet de savoir au premier coup d'œil si l'on travaille avec une vraie table, ou avec une vue.