

SQL CREATE VIEW

Cours SQL	Commandes SQL
Manipulation de table	SQL Avancé
Syntaxe SQL	

SQL > Manipulation de table > Create View

Les vues peuvent être considérées comme des tables virtuelles. Généralement, une table contient un jeu de définitions et elle est destinée à stocker physiquement les données. Une vue a également un jeu de définitions, créé au-dessus des tables ou d'autres vues, et elle ne stocke pas physiquement les données.

La syntaxe pour la création d'une vue est comme suit :

CREATE VIEW "nom de vue" AS "instruction SQL";

"instruction SQL" peut être n'importe quelle instruction SQL que nous avons vue dans ce didacticiel.

À titre d'illustration, utilisons un exemple simple. Supposons que nous avons la table suivante :

Table **Customer**

Nom de Colonne	Type de Données
First_Name	char(50)
Last_Name	char(50)
Address	char(50)
City	char(50)
Country	char(25)
Birth_Date	datetime

et pour créer une vue appelée **V_Customer** contenant seulement les colonnes First_Name, Last_Name et Country de cette table, il faut saisir :

**CREATE VIEW V_Customer
AS SELECT First_Name, Last_Name, Country
FROM Customer;**

Nous avons à présent une vue appelée **V_Customer** avec la structure suivante :

View **V_Customer**

Nom de Colonne	Type de Données
----------------	-----------------

First_Name	char(50)
Last_Name	char(50)
Country	char(25)

Il est aussi possible d'utiliser une vue pour appliquer des jointures à deux tables. Dans ce cas, les utilisateurs ne verront qu'une vue au lieu de deux tables, disposant ainsi d'instructions SQL beaucoup plus simples. Supposons que nous avons les deux tables suivantes :

Table **Store_Information**

Store_Name	Sales	Txn_Date
Los Angeles	1500	05-Jan-1999
San Diego	250	07-Jan-1999
Los Angeles	300	08-Jan-1999
Boston	700	08-Jan-1999

Table **Geography**

Region_Name	Store_Name
East	Boston
East	New York
West	Los Angeles
West	San Diego

et pour créer une vue contenant des ventes par données de région, il faudra définir l'instruction SQL suivante :

```
CREATE VIEW V_REGION_SALES
AS SELECT A1.Region_Name REGION, SUM(A2.Sales) SALES
FROM Geography A1, Store_Information A2
WHERE A1.Store_Name = A2.Store_Name
GROUP BY A1.Region_Name;
```

Nous avons la vue **V_REGION_SALES**, définie pour stocker les ventes par enregistrements de région. Pour connaître le contenu de cette vue, il faut saisir :

```
SELECT * FROM V_REGION_SALES;
```

Résultat:

```
REGION SALES
East      700
West     2050
```

SQL CREATE INDEX >>

Copyright © 2021 [1keydata.com](https://www.1keydata.com) Tous droits réservés

[SQL SELECT](#)

[SQL DISTINCT](#)

[SQL WHERE](#)

[SQL AND OR](#)

[SQL IN](#)

[SQL BETWEEN](#)

[SQL LIKE](#)

[SQL ORDER BY](#)

[SQL Fonctions](#)

[SQL COUNT](#)

[SQL GROUP BY](#)

[SQL HAVING](#)

[SQL ALIAS](#)

[SQL Jointure](#)

[SQL Jointure Externe](#)

[SQL Concaténer](#)

[SQL SUBSTRING](#)

[SQL TRIM](#)

[SQL CREATE TABLE](#)

[SQL CONSTRAINT](#)

[SQL Clé primaire](#)

[SQL Clé étrangère](#)

[SQL CREATE VIEW](#)

[SQL CREATE INDEX](#)

[SQL ALTER TABLE](#)

[SQL DROP TABLE](#)

[SQL TRUNCATE TABLE](#)

[SQL INSERT INTO](#)

[SQL UPDATE](#)

[SQL DELETE FROM](#)

[SQL Avancé](#)

[SQL UNION](#)

[SQL UNION ALL](#)

[SQL INTERSECT](#)

[SQL MINUS](#)

[SQL Sous-requête](#)

[SQL EXISTS](#)

[SQL CASE](#)

[Plan du site](#)