



Extrait du Environnement iSeries

<http://xdocs400.com/spip.php?article203>

Aide mémoire SQL sur les instructions de base.

- Les articles -



Date de mise en ligne : mercredi 27 juillet 2011

Date de parution : 5 octobre 2004

Description :

Un mémento sur les instructions de base sql.

Environnement iSeries

Peut être êtes vous comme moi, et n'estimez pas nécessaire de tout retenir par coeur.

Oui, mais dans ce cas, il faut avoir les bons outils, comme des aides mémoires par exemple.

Celui ci vous propose :

- un récapitulatif des instructions de base disponibles pour SQL.
- iSeries, les nouveautés SQL à partir de la V4R5.
- iSeries, les nouveautés SQL à partir de la V5R1.
- iSeries, les nouveautés SQL à partir de la V5R2.
- les liens vers les principaux aides mémoires publiés ou référencés sur le site, comme par exemple la liste exhaustive des fonctions scalaires disponibles pour DB2 iSeries.
- les liens vers les principales astuces SQL publiées sur le site.

Annotations

L'opérateur de concaténation est CONCAT. Le double pipe (deux barres verticales) est souvent utilisé pour remplacer CONCAT. Sur AS400, vous trouverez plus souvent le double point d'exclamation pour remplacer le CONCAT.

L'utilisation du symbole Pipe (barre verticale) est nuisible à la portabilité du code entre les différents produits base de données IBM. Il est préférable d'utiliser l'opérateur CONCAT plutôt que le double Pipe. Consulter la [note d'information IBM](#) sur les caractères.

Rappel des instructions sql de base

Sélectionner les données

- SELECT FROM
- SELECT * FROM t1 (toutes les colonnes)
- SELECT c1,c2 FROM t1 (sélection des colonnes c1 et c2)
- SELECT DISTINCT c1 FROM t1 (élimine les doublons : ne ramène qu'une valeur pour la colonne c1)
- SELECT c1 AS "colonne1" FROM t1 (renommer une colonne) [\[1\]](#)

Restreindre la sélection

- SELECT * FROM t1 WHERE <condition(s)>
- SELECT * FROM t1 WHERE c1 IN ('01','02','04')
- SELECT * FROM t1 WHERE c2 NOT BETWEEN 10 AND 15
- SELECT * FROM t1 WHERE c3 IS NULL
-

SELECT * FROM t1 WHERE c3 IS NOT NULL

- >, >=, <, <=, =, <>, (comparateur arithmétiques)
- AND, OR, NOT, (comparateur logique)
- % (n'importe quelle séquence de car.)
- _ (soulignement) (n'importe quel caractère)

Trier et présenter les résultats

- SELECT * FROM t1 ORDER BY c1 (tri ascendant par défaut)
- SELECT * FROM t1 ORDER BY c2,c4 (tri par c2 puis tri par c4)
- SELECT * FROM t1 ORDER BY c1 ASC, c3 DESC (tri ascendant ou descendant)

Exprimer les jointures

- SELECT * FROM t1,t2 (jointure sans qualification = produit cartésien)
- SELECT * FROM t1,t2 WHERE t1.c1 = t2.c2 (jointure avec égalité)
- SELECT * FROM t1 a,t2 b,t3 c WHERE a.c1=b.c2 AND b.c2=c.c3 (jointures en cascades)

Manipuler les données

- SELECT c1,c2*3.25 AS "PRIX" FROM t1
- YEAR, MONTH, DATE (date)
- SUBSTRING, UPPER, LOWER, CHARACTER_LENGTH (manipulation de chaînes de car.)

Les fonctions statistiques

- AVG (moyenne)
- COUNT (nombre d'éléments)
- MAX (maximum)
- MIN (minimum)
- SUM (somme)
- SELECT COUNT(*)FROM t1
- SELECT SUM(c1) FROM t2

Regroupements

- SELECT * FROM t1 GROUP BY c1

Sous-requêtes SQL

- SELECT * FROM t1 WHERE c1 > (SELECT MIN(c1) FROM t2)
- SELECT * FROM t1 WHERE c2 NOT IN (SELECT c2 FROM t2)
- SELECT * FROM t1 WHERE c1 > ALL (SELECT c2 FROM t2) (sup. à ttes les valeurs)
- SELECT * FROM t1 WHERE c1 > ANY (SELECT c2 FROM t2) (sup. à au moins 1)

Opérateurs ensemblistes

Ils s'intercalent entre deux sélections

- UNION (sans les doublons) ou UNION ALL (y compris les doublons)
- INTERSECT à partir de la v5r3
- EXCEPT à partir de la v5r3

Insérer des enregistrements

- INSERT INTO t1 VALUES ('abc',5,7) (toutes les valeurs doivent être renseignées)
- INSERT INTO t1(c1,c2) VALUES (1, 'ROUGE') (on ne renseigne que les colonnes indiquées, les colonnes non précisées sont mises à NULL ou à la valeur par défaut si elle est précisée)
- INSERT INTO t1 SELECT * FROM t2

Mises à jour d'enregistrement

- UPDATE t1 SET c2='ROUGE' WHERE c1=1 Supprimer des enregistrements
- DELETE FROM t1 WHERE c1=1
- DELETE FROM t1 WHERE c1 IN (SELECT c2 FROM t2)
- DELETE FROM t1 (supprime tous les enregistrements de la table t1)

iSeries : Nouveautés SQL en V4R5

Limites SQL

- nombre de lignes maximales dans une table = 4 Go
- nombre de tables accédées dans une instruction SQL : 256
- taille maximale d'un récepteur de journaux : 500 fois la capacité d'un journal V4R4
- nombre d'enregistrement verrouillés par une transaction : 500 millions

Apparition de nouvelles fonctions scalaires

- BIGINT(expression) : renvoi d'un "gros entier" ; fonctionnement identique à un INTEGER
- CEILING(expression) : renvoi de l'entier immédiatement supérieur à l'expression ; abréviation CEIL
- SIGN(expression) : renvoi -1, 0 ou 1 selon que l'expression est négative, nulle ou positive
- ROUND(expression1, expression2) : renvoi l'expression 1 arrondie (ROUND (873,726, 2) = 873,730)
- TRUNCATE(expression1, expression2) : renvoi l'expression 1 tronquée (TRUNCATE (873,726, 2) = 873,720)
- DIFFERENCE(expression1, expression 2) : renvoi de 0 à 4 selon le degré de proximité du son de 2 expressions (4 si les sons sont quasi identiques) ... pour mélomanes...
- SOUNDEX(expression) : renvoi un code de 4 caractères représentant le son de l'expression
- RAND(entier) : renvoi un nombre aléatoire entre 0 et 1 de type FLOAT
- RADIANS(expression numérique) : renvoi de la valeur en radians de type FLOAT d'une expression fournie en degré ...on sait jamais...
- ATAN2(X, Y) : renvoi de la valeur en radians d'un angle de coordonnées x, y

Procédures et fonctions SQL

➤

GET DIAGNOSTICS variable=ROW_COUNT : permet d'obtenir le nombre de lignes concernées par l'instruction précédente DELETE, INSERT, UPDATE, PREPARE (estimation)

- GOTO étiquette
- SIGNAL : pour positionner un SQLSTATE et un libellé de message (SIGNAL SQLSTATE 'II001' SET MESSAGE_TEXT='Trop long')
- RESIGNAL : pour substituer un SQLSTATE à un autre
- création d'une procédure stockée en JAVA

Nouveaux formats SQL :

- BIGINT : entier signé sur 8 octets (de - à + 9 milliards de milliards)
- COUNT_BIG : idem COUNT, mais résultat dans un DECIMAL(31, 0)

iSeries : Nouveautés SQL en V5R1

Limitation possible du nombre de lignes retournées par un SELECT, pour ne "ramener" que quelques lignes.

```
SELECT * FROM Txxx
WHERE ...
FETCH FIRST nn ROWS ONLY
```

Utilisation du prédicat LIKE dans un ordre de concaténation.

```
SELECT * FROM Txxx
WHERE NOM LIKE '%' CONCAT PRENOM LIKE '%'...
```

Définition d'une jointure externe droite : renvoi de toutes les lignes de la table de droite et les lignes de la table de gauche vérifiant la condition.

```
RIGHT [OUTER] JOIN
```

Les prédicats IN, BETWEEN, LIKE et IS [NOT] NULL sont désormais acceptés sur une condition de jointure.

Création d'une table dupliquée par SQL.

```
CREATE TABLE Tzzz
LIKE Txxx
```

- les caractéristiques de clé ne sont pas dupliquées
- les lignes ne sont pas copiées

Enrichissement des SUBSELECT : il est désormais possible de citer un SELECT dans la liste des colonnes d'un autre SELECT, à condition que ce premier SELECT ne renvoie qu'une valeur.

- SELECT (zone1, SELECT MAX(zone2) FROM Tyyy) FROM Txx

Il est possible de faire un tri sur une zone qui ne figure pas dans la clause select, sans avoir de code avertissement dans la log.

- SELECT c1, c2 from t1 order by c1, c3

Procédure SQL de création de collection

- CREATE_SQL_SAMPLE(nom de la collection)

Nouvelles fonctions scalaires

- DAYOFWEEK_ISO(date/horodate) : renvoi de 1 (lundi) à 7 (dimanche)
- JULIAN_DAY(date/horodate) : renvoi du nombre de jours depuis le 1er janvier -4712
- MIDNIGHT_SECONDS(time/horodate) : renvoi du nombre de secondes depuis le minuit précédent
- TIMESTAMPDIFF(expression1, expression2) : expression 1 est une valeur entière signifiant 1=fraction de seconde, 2=secondes, 4=minutes, 8=heures, 16=jours, 32=semaines, 64=mois, 128=trimestres, 256=années et expression 2 est une chaîne de longueur 22 représentant la différence entre 2 horodatages
- WEEK_ISO(date/horodate) : fournit le numéro de la semaine (la semaine 1 contient le 4 janvier)
- GRAPHIC(expression alpha, [longueur], [CCSID]) : transforme une chaîne en DBCS ou UCS-2 (Unicode)
- PI() : fournit la valeur de PI en format FLOAT
- SPACE(expression numérique) : pour aller dans la LUNE... fournit une chaîne VARCHAR du nombre d'espaces indiqué

iSeries : Nouveautés SQL en V5R2

Créer un fichier à partir d'un select

- Create table TABTEMP as (select codart, libart from TABART) with data pour créer un fichier avec les données résultantes
- Create table TABTEMP as (select codart, libart from TABART) with no data pour créer un fichier sans les données résultantes
- La clause with no data est prise par défaut si vous ne l'indiquez pas.

Aides mémoire SQL publiés sur le site

- [Messages d'erreurs pour SQL DB2 iSeries](#)
- [Les fonctions scalaires de SQL pour DB2 iSeries.](#)
- [Aide mémoire sur les jointures en SQL.](#)
- [Aide mémoire sur les instructions INSERT en SQL.](#)
- [Equivalence des types de données entre SQL et RPG](#)

Les astuces SQL publiées sur le site

- [Des couleurs dans le résultat d'une requête SQL ?](#)
- [Comment trouver les clés en double avec SQL/400.](#)
- [Gestion des dates avec SQL](#)
- [Récupérer les numéros de séquence libre avec SQL.](#)

- [Limite de DB2 iSeries](#)
- [Générer un script sql create table à partir de l'objet table existant.](#)
- [SQL dynamique, marquer l'emplacement d'une variable par un " ?"](#)
- [Comment gérer dans une requêtes SQL, des notions qui ne figurent pas dans une table ?](#)
- [Tester le contenu d'une variable mémoire par une requête SQL](#)
- [Exécuter une requête SQL, stockée dans le membre d'un fichier source.](#)
- [Comment mettre à jour une partie seulement d'un champ de type caractère en SQL](#)
- [Comment arrêter la journalisation en SQL](#)
- [Analyse des fichiers systèmes.](#)
- [SQL Communication area SQLCA](#)
- [Recadrer une zone alpha, avec un zero significatif en sql](#)
- [Déterminer le nombre d'enregistrement concernés par une requête sql](#)
- [Ajouter un code retour chariot dans une variable, en SQL](#)
- [Comment lister toutes les zones d'un fichier plus quelque chose d'autre avec SQL](#)
- [Caractères non-affichables \(Erreur E/S\)](#)
- [Identifier une rupture de numéros de séquences en SQL](#)
- [Exécuter une requête SQL stockées sur l'IFS via QSH et la commande db2](#)
- [Exécuter une commande AS400 via SQL](#)
- [Créer une table temporaire sous SQL](#)
- [Tester si la valeur d'une colonne alphanumérique est numérique en SQL](#)
- [Formater une requête SQL.](#)
- [Compter les occurrences de caractères dans un champ](#)
- [Conserver un seul espace entre les mots](#)

[1] Sur certain système le mot clé AS est facultatif