

Mémento SQL

Bases relationnelles

Christian BONNIN

DBSPACE – TABLE – INDEX – VUE – SYNONYME – GROUPE –
JONCTION

DBSPACE (CRÉATION-MODIFICATION-SUPPRESSION)

ACQUIRE { PUBLIC
PRIVATE } DBSPACE NAMED [Propriétaire●] Nom-DBspace
([NHEADER = { 8
Entier-1 }] [PAGES = { 128
Entier-2 }]
[PCTINDEX = { 33
Entier-3 }] [PCTFREE = { 15
Entier-4 }]
[LOCK = { PAGE
ROW
DBSPACE }])

ALTER DBSPACE [Propriétaire●] Nom-DBspace

([PCTFREE = Entier] [LOCK = { PAGE
ROW
DBSPACE }])

DROP DBSPACE [Propriétaire●] Nom-DBspace

2^e édition

EYROLLES

Dix-septième tirage 2008
61, Bd St Germain - 75005 PARIS

TABLE (CRÉATION-MODIFICATION-SUPPRESSION)

CREATE TABLE [Créateur●] Nom-de-Table

(Nom-Colonne-1 Format-1 [**NOT NULL**]

[, Nom-Colonne-2 Format-2 [**NOT NULL**]] ...)

[**IN** [Propriétaire●] Nom-DBspace]

ALTER TABLE [Créateur●] Nom-de-Table **ADD** Nom-Colonne Format

DROP TABLE [Créateur●] Nom-de-Table

INDEX (CRÉATION-SUPPRESSION)

CREATE [**UNIQUE**] **INDEX** Nom-Index **ON** [Créateur●] Nom-de-Table

(Nom-Colonne-1 [**ASC**
DESC] [, Nom-Colonne-2 [**ASC**
DESC]] ...)

[**PCTFREE** = { 10
Entier }]

DROP INDEX [Créateur●] Nom-Index

VUES (CRÉATION-SUPPRESSION)

CREATE VIEW Nom-de-Vue (Nom-Colonne-1 [, Nom-Colonne-2] ...) **AS**

SELECT ... **FROM** ... [**WHERE** ...]

DROP VIEW Nom-de-Vue

SYNONYMES (CRÉATION-SUPPRESSION)

CREATE SYNONYM Nom-Synonyme **FOR** [Créateur●] { Nom-de-Table
Nom-de-Vue }

GROUPES (DÉCLARATION)

DECLARE Nom-Curseur **CURSOR FOR**

SELECT ... **FROM** ... [**WHERE** ...]

GROUP BY Nom-Colonne-1 [, Nom-Colonne-2] ...

[**HAVING** Conditions] [**ORDER BY** Specifs]

JONCTIONS (DÉCLARATION)

DECLARE Nom-Curseur **CURSOR FOR**

SELECT ... **FROM** Nom-de-Table-1

[, Nom-de-Table-2] ... [**WHERE** ...]

LES MANIPULATIONS DE DONNÉES

L'INSTRUCTION SELECT

SELECT $\left[\begin{array}{l} \text{ALL} \\ \text{DISTINCT} \end{array} \right] \left\{ \begin{array}{l} \text{Critères de Sélection} \\ * \end{array} \right\}$

INTO : Host-Variable-1 [, Host-Variable-2] ...

FROM [Créateur●] Nom-de-Table

[**WHERE** Conditions]

Les opérateurs logiques : AND, OR, NOT

Les opérateurs de comparaison : = Egal, > Supérieur, < Inférieur, \neg Non,

\neg = Non égal, > = Supérieur ou égal, < = Inférieur ou égal●

LES CURSEURS

DECLARE Nom-Curseur **CURSOR FOR**

SELECT ... **FROM** ... [**WHERE** ...]

$\left[\text{ORDER BY Specifs-1} \left\{ \begin{array}{l} \text{ASC} \\ \text{DESC} \end{array} \right\}, \text{Specifs-2} \left\{ \begin{array}{l} \text{ASC} \\ \text{DESC} \end{array} \right\} \dots \right]$

OPEN Nom-Curseur

CLOSE Nom-Curseur

FETCH Nom-Curseur **INTO** : Host-Variable-1 [, Host-Variable-2] ...

● Déclaration pour UPDATE

DECLARE Nom-Curseur **CURSOR FOR**

SELECT ... **FROM** ... [**WHERE** ...]

UPDATE OF Nom-Colonne-1 [, Nom-Colonne-2] ...

- UPDATE de Colonne spécifique

UPDATE Nom-de-Table **SET** Nom-Colonne-1 = Expression
 [, Nom-Colonne-2 = Expression] ...
WHERE CURRENT OF Nom-Curseur

- Delete d'une ligne de table

DELETE FROM Nom-de-Table **WHERE CURRENT OF** Nom-Curseur

- Déclaration pour INSERT

DECLARE Nom-Curseur **CURSOR FOR INSERT INTO** Nom-de-Table
 [(Nom-Colonne-1 [, Nom-Colonne-2] ...)]
VALUES (Valeur-1 [, Valeur-2] ...)

- Insertion de ligne dans une table

PUT Nom-Curseur

PROGRAMMER SANS CURSEUR

- Mise à jour de données

UPDATE Nom-de-Table **SET** Nom-Colonne-1 = Expression
 [, Nom-Colonne-2 = Expression] ...
 [**WHERE** Conditions]

- Suppression de lignes de tables

DELETE FROM Nom-de-Table [**WHERE** Conditions]

- Insertion d'une seule ligne de table

INSERT INTO Nom-de-Table
 [(Nom-Colonne-1 [, Nom-Colonne-2] ...)]
VALUES (Valeur-1 [, Valeur-2] ...)

- Insertion de lignes à partir d'une autre table

INSERT INTO Nom-de-Table
 [(Nom-Colonne-1 [, Nom-Colonne-2] ...)]
SELECT ... **FROM** ... [**WHERE** ...]

• Format Général d'utilisation

SELECT Fonction-1 (Argument-1) [, Fonction-2 (Argument-2)] ...
INTO Host-Variable-1 [, Host-Variable-2] ...
FROM Nom-de-Table [**WHERE** Conditions]

• Les fonctions

NULL ⇒ Pas de valeur réelle.

Variables Indicateurs ⇒ s'écrivent : Host-variable : Variable-Indicateur

0 ⇒ Host-variable correcte non NULL

< 0 ⇒ Host-variable NULL

> 0 ⇒ Host-variable tronquée

AVG ⇒ Moyenne arithmétique

SUM ⇒ Somme arithmétique

MAX ⇒ Valeur maximum

MIN ⇒ Valeur minimum

COUNT ({ **DISTINCT** Nom-de-Colonne }
 *)

[**NOT**] **BETWEEN** Expression-1 **AND** Expression-2

⇒ Recherche entre 2 valeurs

[**NOT**] **IN** (liste de Constantes ou Host-Variables)

⇒ Recherche d'une valeur d'une liste

[**NOT**] **LIKE** (chaîne de Caractères ou Host-Variable)

⇒ Recherche d'une valeur similaire

GRANT POUR TABLES ET VUES

GRANT {
ALTER
DELETE
INDEX
INSERT
SELECT
UPDATE [(Nom-Colonne-1 [, Nom-Colonne-2] ...)]
ALL [PRIVILEGES]

ON [Créateur•] {
 Nom-de-Table
 Nom-de-Vue }

TO {
PUBLIC
 Utilisateur-1 [, Utilisateur-2] ... }

[WITH GRANT OPTION]

GRANT POUR EXÉCUTION (RUN)

GRANT RUN ON [Créateur•] Nom-de-Programme

TO {
PUBLIC
 Utilisateur-1 [, Utilisateur-2] ... }

[WITH GRANT OPTION]

GRANT POUR AUTORISATIONS SPÉCIALES

GRANT {
CONNECT
DBA
RESOURCE

TO Utilisateur-1 [, Utilisateur-2] ...

[IDENTIFIED BY Mot-Passe-1 [, Mot-Passe-2] ...]

01	SQLCA●	
05	SQLCAID	PIC X(8)●
05	SQLCABC	PIC S9(9) COMPUTATIONAL●
05	SQLCODE	PIC S9(9) COMPUTATIONAL●
05	SQLERRM●	
49	SQLERRML	PIC S9(4) COMPUTATIONAL●
49	SQLERRMC	PIC X(70)●
05	SQLERRP	PIC X(8)●
05	SQLERRD	OCCURS 6 TIMES PIC S9(9) COMPUTATIONAL●
05	SQLWARN●	
10	SQLWARNO	PIC X(1)●
10	SQLWARN1	PIC X(1)●
10	SQLWARN2	PIC X(1)●
10	SQLWARN3	PIC X(1)●
10	SQLWARN4	PIC X(1)●
10	SQLWARN5	PIC X(1)●
10	SQLWARN6	PIC X(1)●
10	SQLWARN7	PIC X(1)●
10	SQLWARN8	PIC X(1)●
10	SQLWARN9	PIC X(1)●
10	SQLWARNA	PIC X(1)●
05	SQLEXT	PIC X(5)●

WHENEVER = DÉCLARATIVES SQL DE TRAITEMENT D'ERREURS

EXEC SQL WHENEVER { SQLERROR
SQLWARNING }

{ STOP
CONTINUE
GOTO Adresse } END-EXEC●

EXEC SQL WHENEVER NOT FOUND { CONTINUE
GOTO Adresse } END-EXEC●

VALEURS DE SQL CODE

ZERO : Tout s'est bien passé

NÉGATIF : Erreur Grave

POSITIF : Warning – SQL CODE = 100 ⇒ Fin de table

MISE EN EXPLOITATION DE SQL

INITIALISATION D'UNE MACHINE UTILISATEUR

SQLINIT [D (Nom-DB / SQLDBA)]

LA PRÉCOMPILATION DES INSTRUCTIONS SQL

SQLPREP { COBOL
PLI
ASM
FORTTRAN } PREPARM (PREPNAME = Nom-Programme

[, USERID = Nom-Utilisateur [/ Mot-de-Passe]

[, NOCHECK] [, GRAPHIC] [, NOPRINT]
[, CHECK] [, NOGRAPHIC] [, PRINT]

[, QUOTE] [, LINE COUNT (Nombre-Entier)]
[, APOST]

[, ISOLATION (CS / USER / RR)] { , BLOCK }
{ , NOBLOCK }

[, SYSIN (Fichier-1)] [, SYSPRINT (Fichier-2)]

[, SYSPUNCH (Fichier-3)]

[, DBNAME (Nom-Base-de-Données)]

APPEL DE SQL DANS UN PROGRAMME COBOL

EXEC SQL CONNECT : Nom-Utilisateur

IDENTIFIED BY Mot-de-Passe

END-EXEC•

DÉCONNEXION DU SYSTÈME SQL

EXEC SQL { COMMIT
ROLLBACK } WORK END-EXEC.

MISE AU POINT AVEC ISQL

● Connexion à ISQL ⇒ IPL CMS puis ISQL
ou SQLINIT DBNAME (Nom-de-Table)

● Création DBSPACE ⇒

ACQUIRE { PUBLIC
PRIVATE } DBSPACE ...

● Modification DBSPACE ⇒

ALTER DBSPACE Nom-DBSpace ...

● Modification de Table ⇒

ALTER TABLE [Créateur●] Nom-de-Table ...

● Visualisation arrière ⇒

BACKWARD [Entier
MAX]

● Sabordage d'une commande ⇒ CANCEL

● Modification de commande ISQL ⇒

CHANGE / Chaîne-de-Caractères-1 [/ Chaîne-de-Charatères-2]

● Affichage d'une colonne à l'écran ⇒ COLUMN [n]

● Commitment de fin de travail ⇒ COMMIT WORK

● Connexion à SQL ⇒ CONNECT Nom-Utilisateur

IDENTIFIED BY Mot-de-Passe

● Création d'Index ⇒ CREATE [UNIQUE] INDEX Nom-Index ...

● Création de Table ⇒ CREATE TABLE [Créateur●] Nom-de-Table ...

● Création de Vue ⇒ CREATE VIEW Nom-de-Vue ...

● Suppression d'une ligne dans une table ⇒

DELETE FROM Nom-de-Table ...

● Affichage à l'écran ⇒ DISPLAY

● Suppression de DBSPACE ⇒

DROP DBSPACE [Propriétaire●] Nom-DBSpace

● Suppression d'INDEX ⇒

DROP INDEX [Créateur●] Nom-Index

● Suppression de Programme ⇒

DROP PROGRAM [Créateur●] Nom-de-Programme

- Suppression d'une Table ⇒

DROP TABLE [Créateur●] Nom-de-Table

- Suppression d'une Vue ⇒

DROP VIEW Nom-de-Vue

- Fin d'affichage ⇒ **END**

- Suppression de commande ISQL ⇒

ERASE Nom-Commande-1 [Nom-Commande-2] ...

- Achèvement de commande ISQL ⇒ **EXIT**

- Formattage de la sortie d'une interrogation ⇒

FORMAT {

BTITLE { Chaîne de caractères
ERASE }

COLUMN { Nom-de-Colonne
N°-de-Colonne }

{ **DPLACE** Entier-1
NAME Chaîne-de-Caract.
WIDTH Entier-2
ZERO { **ON** }
 { **OFF** } }

EXCLUDE [**ALL BUT**] (liste de Colonnes)

GROUP { **EXCEPT** (liste de Colonnes)
ERASE }

INCLUDE (liste de Colonnes)

NULL Chaîne de Caractères

{ **ON** }
{ **OFF** }

OUTLINE { **ON** }
 { **OFF** }

SEPARATOR { Entier-3 **BLANCK** }
 { Chaîne de Caractères }

SUBTOTAL { [**EXCEPT**] (liste de Colonnes)
ERASE }

TTITLE { Chaîne de Caractères
ERASE }

VARCHAR Entier-4

TYPE SQL	PICTURE COBOL	DCL PL/I	FORMAT INTERNE
INTEGER	S9(9) COMP	BIN FIXED (31)	Entier binaire de 31 bits plus le signe.
SMALLINT	S9(4) COMP	BIN FIXED (15)	Entier binaire de 15 bits plus le signe.
DECIMAL (m, [n])	S9(p)V9(q) COMP-3	FIXED DECIMAL (m, n)	Décimal condensé sur 8 octets maximum avec un nombre pair d'octets. m = nombre maxi de chiffres dont n après la virgule. m = p + q et q = n.
FLOAT	COMP-2	BIN FLOAT (53)	Virgule flottante double précision sur 8 octets.
CHAR (n)	X(n) DISPLAY	CHAR (n)	Chaîne de n caractères de longueur fixe avec $n \leq 254$.
VARCHAR (n)	—	CHAR (n) VARYING	Chaîne de longueur variable où $n \leq 254$ représente la longueur maximum.
LONG VARCHAR	—	CHAR (n) VARYING	Chaîne de longueur variable où n peut atteindre 32 767 caractères.

UNLOAD-RELOAD DE TABLE ET DBSPACE

UNLOAD { DBSPACE ([Créateur●] Nom-DBspace) }
 { TABLE ([Créateur●] Nom-de-Table) }

OUTFILE (Nom-DD-de-Fichier)

RELOAD TABLE ([Créateur●] Nom-de-Table)

{ NEW ([Créateur●] Nom-DBspace) }
 { PURGE }

[INTABLE ([Créateur●] Nom-de-Table)]

INFILE (Nom-DD-de-Fichier)

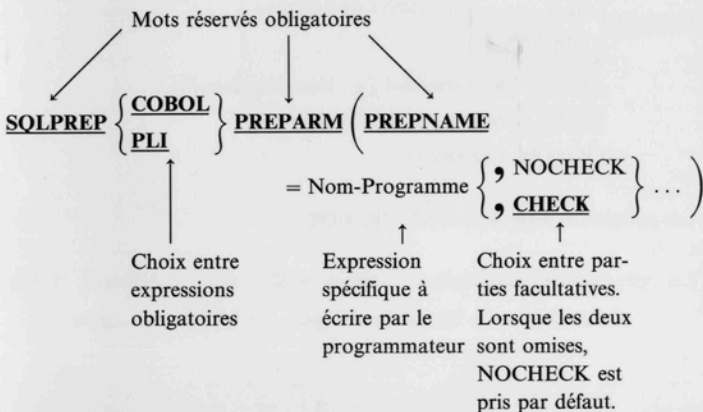
RELOAD DBSPACE ([Créateur●] Nom-DBspace) { NEW }
 { PURGE }

INFILE (Nom-DD-de-Fichier)

Les notations utilisées pour décrire les instructions SQL, sont les expressions couramment admises pour les langages de programmation, c'est-à-dire :

- Les mots réservés SQL sont imprimés en lettres majuscules **SELECT** et les mots obligatoires dans une expression sont également en lettres majuscules mais en gras **DECLARE**.
- Les noms symboliques ou expressions spécifiques à écrire par le programmeur sont imprimés en caractères minuscules.
- Les expressions facultatives sont éditées entre crochets []. Lorsqu'on peut choisir entre plusieurs expressions facultatives, elles sont regroupées entre crochets.
- Lorsqu'on peut choisir entre plusieurs expressions mais que, au moins l'une d'elles est obligatoire, elles sont regroupées entre accolades { }.

Exemple de Notations :



CHEZ LE MÊME ÉDITEUR

C. SOUTOU. – **Apprendre SQL avec MySQL.**

N° 11915, 2006, 398 pages.

C. SOUTOU, O. TESTE. – **SQL pour Oracle.**

N° 11697, 2^e édition, 2005, 500 pages + DVD-Rom.

R. BIZOÏ. – **SQL pour Oracle 10g.**

N° 12055, 2007, 620 pages.

R. BIZOÏ. – **PL/SQL pour Oracle 10g.**

N° 12056, 2007, 340 pages.

M. KOFLER. – **MySQL 5. Guide de l'administrateur et du développeur.**

N° 11633, 2005, 672 pages.

C. SOUTOU. – **UML 2 pour les bases de données.**

N° 12091, 2007, 314 pages.

Imprimerie nouvelle
45800 Saint-jean-de-Braye
n° 423437X
Dépôt légal : Novembre 2007
N° d'éditeur : 5412



9 782212 062588
ISBN : 978-2-212-06258-8