

9

Créer et gérer des tables

Objectifs

A la fin de ce chapitre, vous pourrez :

- **décrire les principaux objets de base de données**
- **créer des tables**
- **décrire les différents types de données utilisables pour les définitions de colonne**
- **modifier des définitions de table**
- **supprimer, renommer et vider des tables**

Objets de base de données

Objet	Description
Table	Unité de stockage élémentaire, composée de lignes et de colonnes
Vue	Représentation logique de sous-ensembles de données issus d'une ou de plusieurs tables
Séquence	Générateur de valeurs numériques
Index	Améliore les performances de certaines interrogations
Synonyme	Permet d'affecter un autre nom à un objet

Règles d'appellation

Les noms de table et de colonne :

- **doivent commencer par une lettre,**
- **peuvent comporter de 1 à 30 caractères,**
- **ne peuvent contenir que les caractères A à Z, a à z, 0 à 9, _, \$, et #,**
- **ne doivent pas être identiques à ceux d'un autre objet appartenant au même utilisateur,**
- **ne doivent pas correspondre à des mots réservés du serveur Oracle.**

Instruction CREATE TABLE

- Vous devez disposer :
 - du privilège CREATE TABLE,
 - d'un espace de stockage

```
CREATE TABLE [schema.]table  
              (column datatype [DEFAULT expr][, ...]);
```

- Vous devez indiquer :
 - le nom de la table,
 - le nom, le type de données et la taille des colonnes.

Référencer des tables appartenant à un autre utilisateur

- Les tables appartenant à d'autres utilisateurs ne se trouvent pas dans le schéma utilisateur.
- Le nom du propriétaire doit être placé devant le nom des tables.

Option DEFAULT

- Permet d'indiquer la valeur par défaut d'une colonne lors d'une insertion.

```
... hire_date DATE DEFAULT SYSDATE, ...
```

- Valeurs autorisées : valeurs littérales, expressions et fonctions SQL.
- Valeurs non autorisées : noms d'autres colonnes ou pseudo-colonnes.
- Le type de données par défaut doit correspondre à celui de la colonne.

Créer des tables

- **Créez la table.**

```
CREATE TABLE dept
      (deptno  NUMBER(2),
       dname    VARCHAR2(14),
       loc      VARCHAR2(13));
```

Table created.

- **Vérifiez la création de la table.**

```
DESCRIBE dept
```

Name	Null?	Type
DEPTNO		NUMBER(2)
DNAME		VARCHAR2(14)
LOC		VARCHAR2(13)

Tables de la base de données Oracle

- **Les tables utilisateur :**
 - constituent un ensemble de tables créées et gérées par l'utilisateur,
 - contiennent des informations relatives à l'utilisateur.
- **Le dictionnaire de données :**
 - constitue un ensemble de tables créées et gérées par le serveur Oracle,
 - contient des informations relatives à la base de données.

Interroger le dictionnaire de données

- Consultez le nom des tables appartenant à l'utilisateur.

```
SELECT table_name  
FROM user_tables ;
```

- Affichez les différents types d'objet appartenant à l'utilisateur.

```
SELECT DISTINCT object_type  
FROM user_objects ;
```

- Affichez les tables, vues, synonymes et séquences appartenant à l'utilisateur.

```
SELECT *  
FROM user_catalog ;
```

Types de données

Data Type	Description
VARCHAR2(<i>size</i>)	Données alphanumériques de longueur variable
CHAR(<i>size</i>)	Données alphanumériques de longueur fixe
NUMBER(<i>p, s</i>)	Données numériques de longueur variable
DATE	Valeurs de date et d'heure
LONG	Données alphanumériques d'une longueur variable pouvant atteindre 2 giga-octets
CLOB	Données alphanumériques de 4 giga-octets maximum
RAW et LONG RAW	Données binaires raw
BLOB	Données binaires de 4 giga-octets maximum
BFILE	Données binaires stockées dans un fichier externe de 4 giga-octets maximum
ROWID	Système numérique en base 64 représentant l'adresse unique d'une ligne dans sa table

Types de données d'horodatage

Améliorations des types de données d'horodatage dans Oracle9i :

- de nouveaux types de données d'horodatage,
- un nouveau système de stockage des types de données,
- améliorations des fuseaux horaires et des fuseaux locaux.

Type de données	Description
TIMESTAMP	Date avec partie décimale des secondes
INTERVAL YEAR TO MONTH	Données enregistrées sous la forme d'un intervalle en années et en mois
INTERVAL DAY TO SECOND	Données enregistrées sous la forme d'un intervalle en jours (heures, minutes et secondes)

Types de données d'horodatage

- Le type de données **TIMESTAMP** est une extension du type de données **DATE**.
- Il permet d'enregistrer non seulement l'année, le mois et le jour du type de données **DATE**, mais également les heures, les minutes, les secondes et la partie décimale des secondes.
- Ce type de données s'utilise de la façon suivante :

```
TIMESTAMP[(fractional_seconds_precision)]
```

Type de données **TIMESTAMP WITH TIME ZONE**

- **TIMESTAMP WITH TIME ZONE** est une variante de **TIMESTAMP** qui inclut un déplacement de fuseau horaire dans sa valeur.
- Le déplacement de fuseau horaire correspond à la différence, en heures et minutes, entre l'heure locale et l'heure UTC.

```
TIMESTAMP[(fractional_seconds_precision)]  
WITH TIME ZONE
```

Type de données **TIMESTAMP** **WITH LOCAL TIME**

- **TIMESTAMP WITH LOCAL TIME ZONE** est une autre variante de **TIMESTAMP** qui inclut un déplacement de fuseau horaire dans sa valeur.
- Les données stockées dans la base de données sont normalisées par rapport au fuseau horaire de la base.
- Le déplacement de fuseau horaire n'est pas stocké en tant qu'élément des données de colonne. Oracle renvoie les données dans le fuseau horaire de la session utilisateur locale.
- Le type de données **TIMESTAMP WITH LOCAL TIME ZONE** s'utilise de la façon suivante :

```
TIMESTAMP[(fractional_seconds_precision)]  
WITH LOCAL TIME ZONE
```

Type de données INTERVAL YEAR TO MONTH

- Le type de données INTERVAL YEAR TO MONTH permet d'enregistrer une période à l'aide des champs d'horodatage YEAR et MONTH.

```
INTERVAL YEAR [(year_precision)] TO MONTH
```

```
INTERVAL '123-2' YEAR(3) TO MONTH
```

Indique un intervalle de 123 années et 2 mois.

```
INTERVAL '123' YEAR(3)
```

Indique un intervalle de 123 années.

```
INTERVAL '300' MONTH(3)
```

Indique un intervalle de 300 mois.

```
INTERVAL '123' YEAR
```

Renvoie un message d'erreur, car la précision par défaut est 2 tandis que '123' comporte 3 chiffres.

Type de données INTERVAL

DAY TO SECOND

- **INTERVAL DAY TO SECOND** permet d'enregistrer une période en termes de jours, d'heures, de minutes et de secondes.

```
INTERVAL DAY [(day_precision)]  
            TO SECOND [(fractional_seconds_precision)]
```

```
INTERVAL '4 5:12:10.222' DAY TO SECOND(3)
```

Indique 4 jours, 5 heures, 12 minutes, 10 secondes et 222 millièmes de seconde.

```
INTERVAL '7' DAY
```

Indique 7 jours.

```
INTERVAL '180' DAY(3)
```

Indique 180 jours.

Type de données **INTERVAL** **DAY TO SECOND**

- **INTERVAL DAY TO SECOND** permet d'enregistrer une période en termes de jours, d'heures, de minutes et de secondes.

INTERVAL '4 5:12:10.222' DAY TO SECOND(3)

Indique 4 jours, 5 heures, 12 minutes, 10 secondes et 222 millièmes de seconde.

INTERVAL '4 5:12' DAY TO MINUTE

Indique 4 jours, 5 heures et 12 minutes.

INTERVAL '400 5' DAY(3) TO HOUR

Indique 400 jours et 5 heures.

INTERVAL '11:12:10.2222222' HOUR TO SECOND(7)

Indique 11 heures, 12 minutes et 10,2222222 secondes.

Créer une table à l'aide d'une syntaxe de sous-interrogation

- Créez une table et insérez des lignes en associant l'instruction **CREATE TABLE** et l'option **AS *subquery***.

```
CREATE TABLE table  
    [(column, column...)]  
AS subquery;
```

- Faites correspondre le nombre de colonnes indiquées au nombre de colonnes de la sous-interrogation.
- Définissez des colonnes à l'aide de noms de colonne et de valeurs par défaut.

Créer une table à l'aide d'une sous-interrogation

```
CREATE TABLE dept80
AS
SELECT  employee_id, last_name,
        salary*12 ANNSAL,
        hire_date
FROM    employees
WHERE   department_id = 80;
```

Table created.

```
DESCRIBE dept80
```

Name	Null?	Type
EMPLOYEE_ID		NUMBER(6)
LAST_NAME	NOT NULL	VARCHAR2(25)
ANNSAL		NUMBER
HIRE_DATE	NOT NULL	DATE

Instruction ALTER TABLE

L'instruction ALTER TABLE permet :

- d'ajouter une nouvelle colonne,
- de modifier une colonne existante,
- de définir une valeur par défaut pour une nouvelle colonne,
- de supprimer une colonne.

Instruction ALTER TABLE

L'instruction ALTER TABLE permet d'ajouter, de modifier ou de supprimer des colonnes.

```
ALTER TABLE table
ADD          (column datatype [DEFAULT expr]
             [, column datatype]...);
```

```
ALTER TABLE table
MODIFY       (column datatype [DEFAULT expr]
             [, column datatype]...);
```

```
ALTER TABLE table
DROP        (column);
```

Ajouter une colonne

DEPT80

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DATE
149	Zlotkey	126000	29-JAN-00
174	Abel	132000	11-MAY-96
176	Taylor	103200	24-MAR-98

Nouvelle colonne

JOB_ID

**"Ajoutez une
colonne à la
table
DEPT80."**

DEPT80

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DATE	JOB_ID
149	Zlotkey	126000	29-JAN-00	
174	Abel	132000	11-MAY-96	
176	Taylor	103200	24-MAR-98	

Ajouter une colonne

- La clause ADD permet d'ajouter des colonnes.

```
ALTER TABLE dept80
ADD      (job_id VARCHAR2(9));
Table altered.
```

- La nouvelle colonne est placée à la fin de la table.

EMPLOYEE_ID	LAST_NAME	ANNSAL	HIRE_DATE	JOB_ID
149	Zlotkey	126000	29-JAN-00	
174	Abel	132000	11-MAY-96	
176	Taylor	103200	24-MAR-98	

Modifier une colonne

- Vous pouvez modifier le type de données, la taille et la valeur par défaut d'une colonne.

```
ALTER TABLE dept80  
MODIFY (last_name VARCHAR2(30));  
Table altered.
```

- La modification d'une valeur par défaut ne s'applique qu'aux insertions ultérieures dans la table.

Supprimer une colonne

La clause **DROP COLUMN** permet de supprimer d'une table les colonnes qui ne sont plus utiles.

```
ALTER TABLE dept80  
DROP COLUMN job_id;  
Table altered.
```

Option SET UNUSED

- L'option SET UNUSED permet d'identifier une ou plusieurs colonnes comme non utilisées.
- L'option DROP UNUSED COLUMNS permet de supprimer les colonnes identifiées comme non utilisées.

```
ALTER TABLE table  
SET UNUSED (column);  
OR  
ALTER TABLE table  
SET UNUSED COLUMN column;
```

```
ALTER TABLE table  
DROP UNUSED COLUMNS;
```

Supprimer une table

- La structure et l'ensemble des données de la table sont supprimées.
- Toutes les transactions en cours sont validées.
- Tous les index sont supprimés.
- Vous ne *pouvez pas* annuler une instruction DROP TABLE.

```
DROP TABLE dept80;  
Table dropped.
```

Renommer un objet

- L'instruction RENAME permet de renommer une table, une vue, une séquence ou un synonyme.

```
RENAME dept TO detail_dept;  
Table renamed.
```

- Pour effectuer cette opération, vous devez être propriétaire de l'objet.

Vider une table

- L'instruction **TRUNCATE TABLE** :
 - supprime toutes les lignes d'une table,
 - libère l'espace de stockage utilisé par la table

```
TRUNCATE TABLE detail_dept;  
Table truncated.
```

- Vous ne pouvez pas annuler une instruction **TRUNCATE**.
- Vous pouvez également supprimer des lignes à l'aide de l'instruction **DELETE**.

Ajouter des commentaires à une table

- L'instruction **COMMENT** permet d'ajouter des commentaires à une table ou à une colonne.

```
COMMENT ON TABLE employees  
IS 'Employee Information';  
Comment created.
```

- Vous pouvez visualiser les commentaires grâce aux vues suivantes du dictionnaire de données :
 - **ALL_COL_COMMENTS**
 - **USER_COL_COMMENTS**
 - **ALL_TAB_COMMENTS**
 - **USER_TAB_COMMENTS**

Synthèse

Ce chapitre vous a permis d'apprendre à utiliser des instructions LDD pour créer, modifier, supprimer et renommer des tables.

Instruction	Description
CREATE TABLE	Crée une table
ALTER TABLE	Modifie la structure d'une table
DROP TABLE	Supprime les lignes et la structure d'une table
RENAME	Renomme une table, une vue, une séquence ou un synonyme
TRUNCATE	Supprime toutes les lignes d'une table et libère l'espace de stockage occupé par cette table
COMMENT	Ajoute des commentaires à une table ou à une vue

Présentation de l'exercice 9

Dans cet exercice, vous allez :

- créer des tables
- créer une table à l'aide de la syntaxe **CREATE TABLE AS**
- modifier des définitions de colonne
- vérifier l'existence des tables
- ajouter des commentaires aux tables
- supprimer des tables
- modifier des tables