

14

Les Contraintes

Objectifs

A la fin de ce chapitre, vous saurez :

- **Définir les contraintes**
- **Créer des contraintes et les maintenir**

Objectifs

Dans ce chapitre, vous allez apprendre à implémenter des règles de gestion en utilisant des contraintes d'intégrité.

Les Contraintes

- **Les contraintes contrôlent des règles de gestion au niveau d'une table.**
- **Les contraintes empêchent la suppression d'une table lorsqu'il existe des dépendances.**
- **Types de contraintes valides dans Oracle :**
 - NOT NULL
 - UNIQUE
 - PRIMARY KEY
 - FOREIGN KEY
 - CHECK

14-3

Copyright © Oracle Corporation, 1998. Tous droits réservés. **ORACLE®**

Contraintes

Oracle8 Server fait appel à des *contraintes* pour empêcher l'entrée de données incorrectes dans des tables.

Vous pouvez utiliser des contraintes pour :

- Appliquer des règles au niveau d'une table chaque fois qu'une ligne est insérée, mise à jour ou supprimée dans cette table. La contrainte doit être satisfaite pour que l'opération réussisse.
- Empêcher la suppression d'une table si il y a des dépendances avec d'autres tables.
- Fournir des règles pour des outils Oracle comme Developer 2000.

Contraintes d'Intégrité des Données

| Contrainte | Description |
|-------------|--|
| NOT NULL | Spécifie que cette colonne ne doit pas contenir de valeur null |
| UNIQUE | Spécifie une colonne ou une combinaison de colonnes dont les valeurs doivent être uniques pour toutes les lignes de la table |
| PRIMARY KEY | Identifie chaque ligne de la table de manière unique |
| FOREIGN KEY | Etablit et contrôle une relation de clé étrangère entre la colonne et une colonne de la table référencée |
| CHECK | Spécifie une condition qui doit être vraie |



Pour plus d'informations, reportez-vous à
Oracle8 Server SQL Reference, Release 8.0, "CONSTRAINT Clause".

Conventions Applicables aux Contraintes

- Si vous ne nommez pas une contrainte, Oracle8 Server créera un nom au format **SYS_Cn**.
- Vous pouvez créer une contrainte :
 - En même temps que la création de la table
 - Une fois que la table est créée
- Définissez une contrainte peut être définie au niveau table ou colonne.
- Consulter le dictionnaire de données pour retrouver une contrainte.

14-4

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

Conventions Applicables aux Contraintes

Toutes les contraintes sont stockées dans le dictionnaire de données. Elles seront très faciles à manipuler si vous leur donnez un nom parlant. Les noms de contraintes sont soumis aux conventions de dénomination des objets standard. Si vous ne nommez pas une contrainte, Oracle8 génère un nom dont le format est SYS_Cn, où *n* est un entier permettant de créer un nom de contrainte unique.

Il est possible de définir les contraintes au moment de la création de la table ou plus tard.

Vous pouvez retrouver les contraintes définies pour une table spécifique en consultant la table USER_CONSTRAINTS du dictionnaire de données.

Les Contraintes

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

```
CREATE TABLE emp(
  empno  NUMBER(4),
  ename  VARCHAR2(10),
  ...
  deptno NUMBER(2) NOT NULL,
  CONSTRAINT emp_empno_pk
             PRIMARY KEY (EMPNO));
```

14-5

Copyright © Oracle Corporation, 1998. Tous droits réservés. **ORACLE®**

Les Contraintes

La diapositive montre la syntaxe utilisée pour définir des contraintes pendant la création d'une table.

Dans cette syntaxe :

| | |
|--------------------------|--|
| <i>schema</i> | nom du propriétaire de la table |
| <i>table</i> | nom de la table |
| DEFAULT <i>expr</i> | valeur par défaut à utiliser si une valeur est omise dans l'ordre INSERT |
| <i>column</i> | nom de la colonne |
| <i>datatype</i> | type de données et longueur de la colonne |
| <i>column_constraint</i> | contrainte d'intégrité incluse dans la définition de la colonne |
| <i>table_constraint</i> | contrainte d'intégrité incluse dans la définition de la table |



Pour plus d'informations, reportez-vous à
Oracle8 Server SQL Reference, Release 8.0, "CREATE TABLE."

Les Contraintes

• Contrainte au niveau colonne

```
column [CONSTRAINT constraint_name] constraint_type,
```

• Contrainte au niveau table

```
column,...  
  [CONSTRAINT constraint_name] constraint_type  
  (column, ...),
```

Les Contraintes (suite)

En général, on crée les contraintes en même temps que la table. Il est cependant possible d'ajouter des contraintes ou d'en désactiver dans une table déjà créée.

Il existe deux niveaux de contraintes :

| Niveau de contrainte | Description |
|----------------------|--|
| Colonne | Référence une seule colonne et se définit dans la spécification de cette colonne ; applicable à n'importe quel type de contrainte d'intégrité |
| Table | Référence une ou plusieurs colonnes et se définit indépendamment de la définition des colonnes ; applicable à toutes les contraintes sauf NOT NULL |

Syntaxe :

| | |
|------------------------|----------------------|
| <i>constraint_name</i> | nom de la contrainte |
| <i>constraint_type</i> | type de contrainte |

La Contrainte NOT NULL

Interdit la présence de valeurs NULL dans la colonne

EMP

| EMPNO | ENAME | JOB | ... | COMM | DEPTNO |
|-------|-------|-----------|-----|------|--------|
| 7839 | KING | PRESIDENT | | | 10 |
| 7698 | BLAKE | MANAGER | | | 30 |
| 7782 | CLARK | MANAGER | | | 10 |
| 7566 | JONES | MANAGER | | | 20 |
| ... | | | | | |

Contrainte NOT NULL
(aucune ligne ne peut avoir de valeur NULL dans cette colonne)

Absence de contrainte NOT NULL
(toute ligne peut avoir une valeur NULL dans cette colonne)

Contrainte NOT NULL

La contrainte NOT NULL

La contrainte NOT NULL interdit la présence de valeurs NULL dans la colonne à laquelle elle s'applique. Par défaut, les colonnes qui ne sont pas associées à la contrainte NOT NULL peuvent contenir des valeurs NULL.

La Contrainte NOT NULL

Se définit au niveau colonne

```
SQL> CREATE TABLE emp(  
2      empno      NUMBER(4),  
3      ename      VARCHAR2(10) NOT NULL,  
4      job        VARCHAR2(9),  
5      mgr        NUMBER(4),  
6      hiredate   DATE,  
7      sal        NUMBER(7,2),  
8      comm       NUMBER(7,2),  
9      deptno     NUMBER(2) NOT NULL);
```

14-8

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

La contrainte NOT NULL (suite)

La contrainte NOT NULL ne peut être définie qu'au niveau colonne, pas au niveau table.

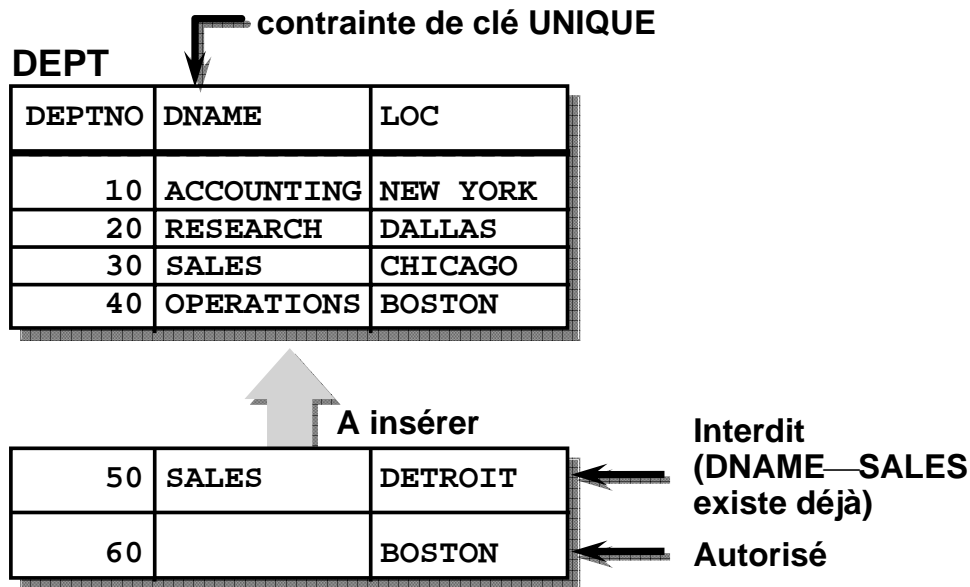
L'exemple ci-dessus applique la contrainte NOT NULL aux colonnes ENAME et DEPTNO de la table EMP. Comme ces contraintes ne sont pas nommées, Oracle8 Server créera des noms pour elles.

Vous pouvez spécifier le nom de la contrainte pendant que vous la définissez.

```
... deptno NUMBER(2)  
  
    CONSTRAINT emp_deptno_nn NOT NULL;
```

Remarque : toutes les contraintes citées en exemple dans ce chapitre ne sont pas nécessairement présentes dans les tables fournies pour les exercices de ce cours. Le cas échéant, il est possible de les ajouter aux tables.

La Contrainte de Clé UNIQUE



La Contrainte de Clé UNIQUE

Une contrainte d'intégrité de type clé UNIQUE exige que chaque valeur dans une colonne ou dans un ensemble de colonnes (la clé) soit unique, c'est-à-dire qu'elle n'existe pas dans plusieurs lignes pour la colonne ou l'ensemble de colonnes spécifiés. La colonne ou l'ensemble de colonnes indiqués dans la définition de la contrainte UNIQUE constituent la *clé unique*. Si la clé comprend plusieurs colonnes, le groupe de colonnes est appelé *clé unique composée*.

Les contraintes UNIQUE autorisent la saisie de valeurs NULL, à moins que vous ne définissiez également des contraintes NOT NULL sur les mêmes colonnes. En fait, lorsque des colonnes n'ont pas de contrainte NOT NULL, un nombre quelconque de lignes peuvent contenir des valeurs NULL puisque celles-ci n'équivalent à rien. Une valeur NULL dans une colonne (ou dans toutes les colonnes appartenant à une clé unique composée) satisfait toujours une contrainte de clé UNIQUE.

Remarque : étant donné le mécanisme de recherche pour les contraintes de clé UNIQUE sur plusieurs colonnes, il ne peut y avoir de valeurs identiques dans les colonnes non NULL d'une contrainte de clé unique composée partiellement NULL.

La Contrainte de Clé UNIQUE

Se définit au niveau table ou colonne

```
SQL> CREATE TABLE dept(  
  2     deptno    NUMBER(2),  
  3     dname     VARCHAR2(14),  
  4     loc       VARCHAR2(13),  
  5     CONSTRAINT dept_dname_uk UNIQUE(dname));
```

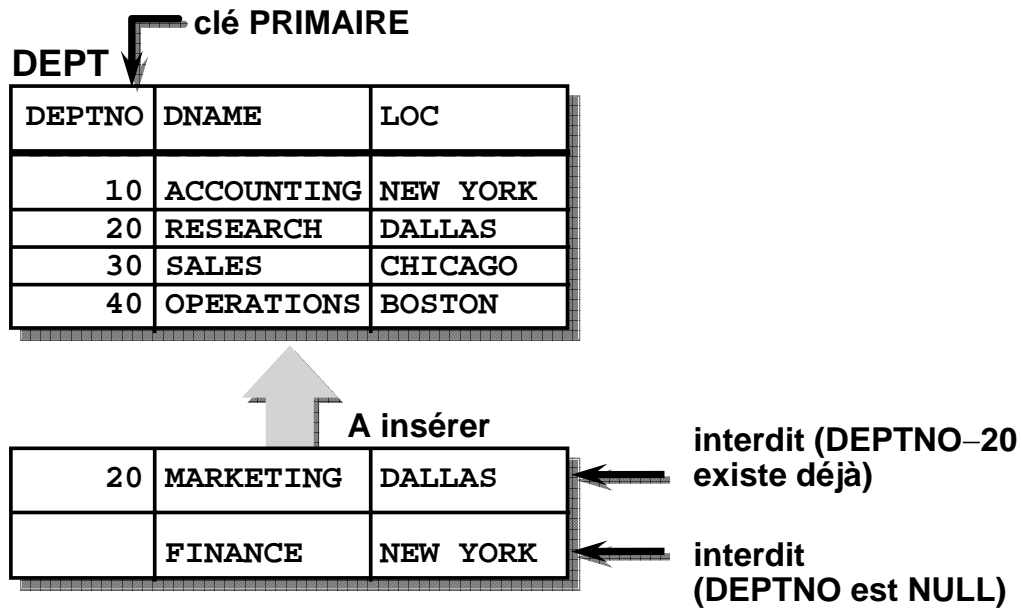
La Contrainte de Clé UNIQUE (suite)

Les contraintes de clé UNIQUE se définissent au niveau colonne ou table. Une clé unique composée est créée à l'aide de définitions de niveau table.

L'exemple de la diapositive applique une contrainte de clé UNIQUE à la colonne DNAME de la table DEPT. Le nom de cette contrainte est DEPT_DNAME_UK.

Remarque : Oracle Server contrôle la contrainte UNIQUE en créant implicitement un index unique sur la clé.

La Contrainte PRIMARY KEY



La Contrainte PRIMARY KEY

Une contrainte PRIMARY KEY crée une clé primaire pour la table. Une seule clé primaire peut être créée par table. La contrainte PRIMARY KEY est une colonne ou un ensemble de colonnes qui identifie de manière unique chaque ligne d'une table. Elle établit une règle d'unicité de la colonne ou d'une combinaison de colonnes et garantit qu'aucune colonne faisant partie de la clé primaire ne contient de valeur NULL.

La Contrainte PRIMARY KEY

Se définit au niveau table ou colonne

```
SQL> CREATE TABLE dept(  
2     deptno      NUMBER(2),  
3     dname       VARCHAR2(14),  
4     loc         VARCHAR2(13),  
5     CONSTRAINT dept_dname_uk UNIQUE (dname),  
6     CONSTRAINT dept_deptno_pk PRIMARY KEY(deptno));
```

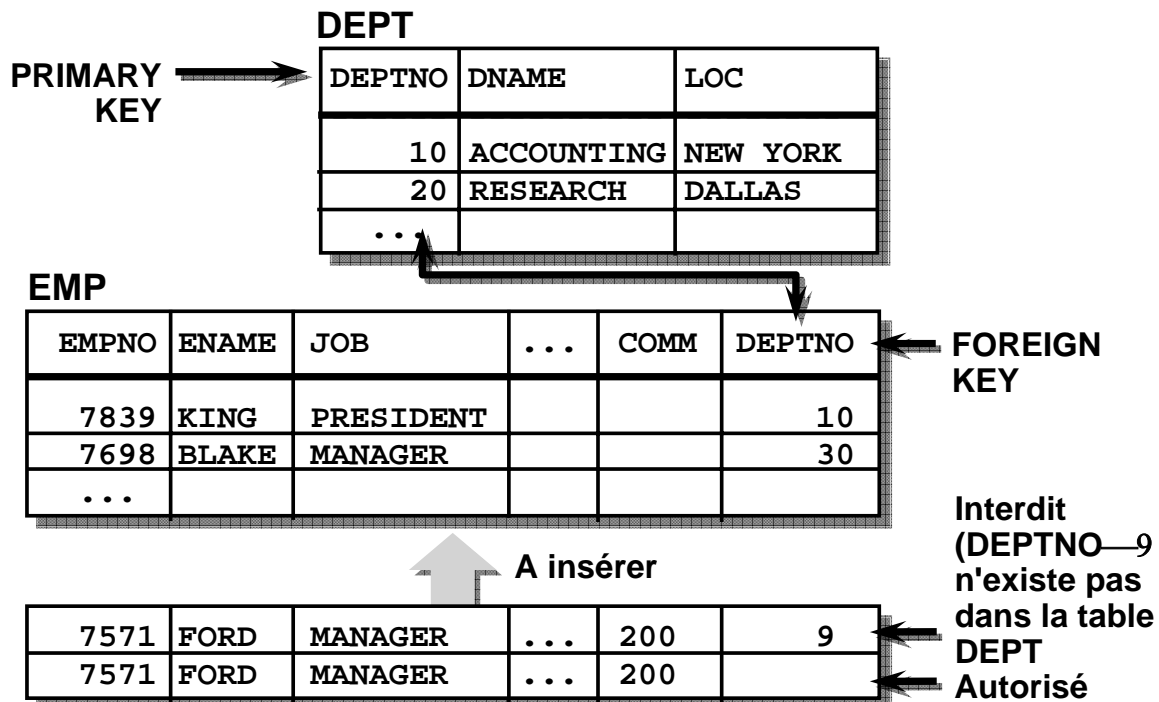
La Contrainte PRIMARY KEY (suite)

Les contraintes PRIMARY KEY peuvent être définies au niveau colonne ou table. Une clé primaire composée se crée à l'aide de la définition de niveau table.

L'exemple de la diapositive définit une contrainte PRIMARY KEY pour la colonne DEPTNO de la table DEPT. Le nom de la contrainte est DEPT_DEPTNO_PK.

Remarque : un index UNIQUE est automatiquement créé pour une colonne PRIMARY KEY.

La Contrainte FOREIGN KEY



14-13

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

La Contrainte FOREIGN KEY

La contrainte FOREIGN KEY, ou contrainte d'intégrité référentielle, désigne une colonne ou une combinaison de colonnes comme étant une clé étrangère et établit une relation avec une clé primaire ou une clé unique de la même table ou d'une table différente. Dans l'exemple ci-dessus, la colonne DEPTNO a été définie comme clé étrangère dans la table EMP (table dépendante ou enfant) ; elle fait référence à la colonne DEPTNO de la table DEPT (table de référence ou parent).

Une valeur de clé étrangère doit obligatoirement correspondre à une valeur existante de la table maître ou être NULL.



Les clés étrangères sont basées sur des valeurs de données et sont des pointeurs purement logiques et non physiques.

La Contrainte FOREIGN KEY

Se définit au niveau table ou colonne

```
SQL> CREATE TABLE emp(  
 2      empno      NUMBER(4),  
 3      ename      VARCHAR2(10) NOT NULL,  
 4      job        VARCHAR2(9),  
 5      mgr        NUMBER(4),  
 6      hiredate   DATE,  
 7      sal        NUMBER(7,2),  
 8      comm       NUMBER(7,2),  
 9      deptno     NUMBER(2) NOT NULL,  
10      CONSTRAINT emp_deptno_fk FOREIGN KEY (deptno)  
11              REFERENCES dept (deptno));
```

La Contrainte FOREIGN KEY (suite)

Les contraintes FOREIGN KEY peuvent être définies au niveau table ou colonne. Une clé étrangère composée se crée au moyen de la définition de niveau table.

L'exemple ci-dessus définit une contrainte FOREIGN KEY pour la colonne DEPTNO de la table EMP. Le nom de la contrainte est EMP_DEPTNO_FK.

Mots-clés Associés à la Contrainte FOREIGN KEY

- **FOREIGN KEY**
 - Définit la colonne dans la table détail dans une contrainte de niveau table
- **REFERENCES**
 - Identifie la table et la colonne de la table maître
- **ON DELETE CASCADE**
 - Autorise la suppression d'une ligne dans la table maître et des lignes dépendantes dans la table détail

La Contrainte FOREIGN KEY (suite)

La clé étrangère est définie dans la table détail et la table contenant la colonne référencée est la table maître. Pour définir la clé étrangère, on utilise une combinaison des mots-clés suivants :

- FOREIGN KEY définit une colonne de la table détail dans une contrainte de niveau table.
- REFERENCES identifie la table et la colonne dans la table maître.
- ON DELETE CASCADE indique que si une ligne est supprimée de la table maître, les lignes dépendantes de la table détail seront également supprimées.

En l'absence de l'option ON DELETE CASCADE, la ligne de la table maître ne peut pas être supprimée si elle est référencée dans la table détail.

La Contrainte CHECK

- Définit une condition que chaque ligne doit obligatoirement satisfaire
- Expressions interdites :
 - Références aux pseudo-colonnes CURRVAL, NEXTVAL, LEVEL et ROWNUM
 - Appels aux fonctions SYSDATE, UID, USER et USERENV
 - Requêtes faisant référence à d'autres valeurs dans d'autres lignes

```
..., deptno  NUMBER(2),
      CONSTRAINT emp_deptno_ck
      CHECK (DEPTNO BETWEEN 10 AND 99),...
```

14-16

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

La contrainte CHECK

La contrainte CHECK définit une condition que chaque ligne doit obligatoirement satisfaire. La condition peut utiliser les mêmes constructions que les conditions d'une requête, aux exceptions près suivantes :

- Références aux pseudo-colonnes CURRVAL, NEXTVAL, LEVEL et ROWNUM
- Appels aux fonctions SYSDATE, UID, USER et USERENV
- Requêtes faisant référence à d'autres valeurs dans d'autres lignes

Plusieurs contraintes CHECK peuvent être définies sur la même colonne. Le nombre de contraintes CHECK pouvant être associées à une colonne est illimité.

Les contraintes CHECK peuvent être définies au niveau colonne ou au niveau table.

Ajout d'une Contrainte

```
ALTER TABLE table
ADD [CONSTRAINT constraint] type (column);
```

- Vous pouvez ajouter ou supprimer une contrainte, mais pas la modifier
- Vous pouvez activer ou désactiver des contraintes
- Pour ajouter une contrainte NOT NULL, utilisez la clause MODIFY

14-17

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

Ajout d'une Contrainte

Vous pouvez ajouter des contraintes dans une table existante en utilisant l'ordre ALTER TABLE avec la clause ADD.

Syntaxe :

| | |
|-------------------|---|
| <i>table</i> | nom de la table |
| <i>constraint</i> | nom de la contrainte |
| <i>type</i> | type de contrainte |
| <i>column</i> | nom de la colonne concernée par la contrainte |



Il est recommandé de préciser le nom de la contrainte, même si ce n'est pas obligatoire. A défaut, le système générera lui-même des noms de contraintes.

Conseils

- Vous pouvez ajouter, supprimer, activer ou désactiver une contrainte, mais il est impossible d'en modifier la structure.
- Vous pouvez ajouter une contrainte NOT NULL à une colonne existante en utilisant la clause MODIFY à la place de la clause ADD de l'ordre ALTER TABLE.

Remarque : Vous pouvez définir une colonne NOT NULL seulement si la table est vide, ou bien si vous spécifiez une valeur par défaut, celle-ci sera alors automatiquement attribuée à toutes les lignes existantes de la table.

Ajout d'une Contrainte

Ajouter une contrainte FOREIGN KEY à la table EMP précisant qu'un manager doit déjà exister dans la table EMP en tant qu'employé valide.

```
SQL> ALTER TABLE      emp
      2  ADD CONSTRAINT emp_mgr_fk
      3              FOREIGN KEY(mgr) REFERENCES emp(empno);
Table altered.
```

Ajout d'une Contrainte

L'exemple ci-dessus crée une contrainte FOREIGN KEY dans la table EMP. Cette contrainte vérifie qu'un manager existe en tant qu'employé valide dans la table EMP.

Suppression d'une Contrainte

- Supprimer de la table EMP la contrainte concernant le manager.

```
SQL> ALTER TABLE          emp
      2 DROP CONSTRAINT emp_mgr_fk;
Table altered.
```

- Supprimer la contrainte PRIMARY KEY de la table DEPT, ainsi que la contrainte FOREIGN KEY associée définie sur la colonne EMP.DEPTNO.

```
SQL> ALTER TABLE          dept
      2 DROP PRIMARY KEY CASCADE;
Table altered.
```

14-19

Copyright © Oracle Corporation, 1998. Tous droits réservés. ORACLE®

Suppression d'une Contrainte

Si vous souhaitez supprimer une contrainte, vous pouvez en retrouver le nom dans les vues du dictionnaire de données USER_CONSTRAINTS et USER_CONS_COLUMNS. Utilisez ensuite l'ordre ALTER TABLE avec la clause DROP. L'option CASCADE de la clause DROP provoque également la suppression de toutes les contraintes associées.

Syntaxe

```
ALTER TABLE table
DROP   PRIMARY KEY | UNIQUE (column) |
      CONSTRAINT constraint [CASCADE];
```

où :

| | |
|-------------------|---|
| <i>table</i> | représente le nom de la table |
| <i>column</i> | représente le nom de la colonne concernée par la contrainte |
| <i>constraint</i> | représente le nom de la contrainte |



Lorsque vous supprimez une contrainte d'intégrité, elle n'est plus contrôlée par Oracle8 Server et n'existe plus dans le dictionnaire de données.

Désactivation de Contraintes

- Pour désactiver une contrainte d'intégrité, utiliser la clause **DISABLE** de l'ordre **ALTER TABLE**.
- Pour désactiver les contraintes d'intégrité dépendantes, ajouter l'option **CASCADE**.

```
SQL> ALTER TABLE          emp
      2  DISABLE CONSTRAINT  emp_empno_pk CASCADE;
Table altered.
```

14-20

Copyright © Oracle Corporation, 1998. Tous droits réservés. **ORACLE®**

Désactivation d'une Contrainte

Vous pouvez désactiver une contrainte sans la supprimer (ce qui évite d'avoir à la recréer) en utilisant l'ordre **ALTER TABLE** avec la clause **DISABLE**.

Syntaxe

```
ALTER TABLE  table
DISABLE  CONSTRAINT constraint [CASCADE];
```

où : *table* représente le nom de la table
 constraint représente le nom de la contrainte

Conseils

- Vous pouvez utiliser la clause **DISABLE** aussi bien dans l'ordre **CREATE TABLE** que **ALTER TABLE**.
- La clause **CASCADE** désactive des contraintes d'intégrité dépendantes.

Activation de Contraintes

- Pour activer une contrainte d'intégrité actuellement désactivée dans la définition de la table, utiliser la clause

ENABLE

```
SQL> ALTER TABLE          emp
      2  ENABLE CONSTRAINT    emp_empno_pk;
Table altered.
```

- Si vous activez une contrainte **UNIQUE** ou **PRIMARY KEY**, un index correspondant est automatiquement créé.

14-21

Copyright © Oracle Corporation, 1998. Tous droits réservés. **ORACLE®**

Activation d'une Contrainte

Vous pouvez activer une contrainte, sans la supprimer et la recréer, en utilisant l'ordre ALTER TABLE avec la clause ENABLE.

Syntaxe

```
ALTER TABLE  table
ENABLE  CONSTRAINT constraint;
```

où : *table* représente le nom de la table
 constraint représente le nom de la contrainte

Conseils

- Si vous activez une contrainte, celle-ci s'applique à toutes les données de la table. Les données de la table doivent donc toutes respecter la contrainte.
- Si vous activez une contrainte **UNIQUE** ou **PRIMARY KEY**, un index unique est automatiquement créé.
- Vous pouvez utiliser la clause **ENABLE** aussi bien dans l'ordre **CREATE TABLE** que **ALTER TABLE**.

Vérification des Contraintes

Pour afficher les définitions et noms de toutes les contraintes, interrogez la table USER_CONSTRAINTS.

```
SQL> SELECT constraint_name, constraint_type,
2         search_condition
3 FROM   user_constraints
4 WHERE  table_name = 'EMP';
```

| CONSTRAINT_NAME | C | SEARCH_CONDITION |
|-----------------|---|--------------------|
| ----- | - | ----- |
| SYS_C00674 | C | EMPNO IS NOT NULL |
| SYS_C00675 | C | DEPTNO IS NOT NULL |
| EMP_EMPNO_PK | P | |
| ... | | |

Vérification des Contraintes

Une fois que vous avez créé une table, vous pouvez en contrôler l'existence en exécutant la commande DESCRIBE. Cependant, la seule contrainte que vous pouvez vérifier de cette manière est NOT NULL. Si vous voulez afficher toutes les contraintes de votre table, interrogez la table USER_CONSTRAINTS.

L'exemple ci-dessus affiche toutes les contraintes de la table EMP.

Remarque : les contraintes qui ne sont pas explicitement nommées par le propriétaire de la table se voient attribuer un nom par le système. Dans la colonne décrivant le type de contrainte, la lettre C signifie CHECK, P représente PRIMARY KEY, R intégrité référentielle et U, UNIQUE. A noter que la contrainte NULL est en réalité une contrainte CHECK.

Affichage des Colonnes Associées aux Contraintes

**Affichez les colonnes associées aux noms
de contraintes au moyen de la vue
USER_CONS_COLUMNS view**

```
SQL> SELECT  constraint_name, column_name
2  FROM      user_cons_columns
3  WHERE     table_name = 'EMP';
```

| CONSTRAINT_NAME | COLUMN_NAME |
|-----------------|-------------|
| EMP_DEPTNO_FK | DEPTNO |
| EMP_EMPNO_PK | EMPNO |
| EMP_MGR_FK | MGR |
| SYS_C00674 | EMPNO |
| SYS_C00675 | DEPTNO |

Affichage des Contraintes

Vous pouvez afficher le nom des colonnes associées à des contraintes en interrogeant la vue USER_CONS_COLUMNS du dictionnaire de données. Cette vue est particulièrement utile pour les contraintes ayant un nom attribué par le système.

Résumé

- **Vous pouvez créer des contraintes de type :**
 - **NOT NULL**
 - **UNIQUE**
 - **PRIMARY KEY**
 - **FOREIGN KEY**
 - **CHECK**
- **Utilisez la table USER_CONSTRAINTS pour afficher les noms et définitions de toutes les contraintes.**

Résumé

Oracle8 Server empêche la saisie de données incorrectes dans des tables par l'intermédiaire de contraintes.

Les types de contraintes utilisés par Oracle8 sont les suivants :

- NOT NULL
- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK

Vous pouvez rechercher les noms et définitions de toutes les contraintes dans la table USER_CONSTRAINTS.

Présentation des Exercices

- **Ajout de contraintes à des tables existantes**
- **Ajout de colonnes supplémentaires à une table**
- **Affichage d'informations des vues du dictionnaire de données**

Présentation des Exercices

Dans les exercices qui suivent, vous allez ajouter des contraintes et des colonnes supplémentaires dans une table en utilisant les ordres étudiés dans ce chapitre.

Exercices 14

1. Ajoutez une contrainte PRIMARY KEY de niveau table dans la table EMPLOYEE en utilisant la colonne ID.
La contrainte devrait être activée dès sa création.
2. Créez une contrainte PRIMARY KEY sur la table DEPARTMENT en utilisant la colonne ID. La contrainte devrait être activée dès sa création.
3. Ajoutez une clé étrangère dans la table EMPLOYEE qui permettra de contrôler que l'employé n'est pas associé à un département inexistant.
4. Vérifiez que les contraintes ont été ajoutées en les recherchant dans USER_CONSTRAINTS. Prenez note des types et des noms des contraintes. Enregistrez votre ordre dans un fichier nommé *p14q4.sql*.

| CONSTRAINT_NAME | C |
|---------------------|----|
| ----- | -- |
| DEPARTMENT_ID_PK | P |
| EMPLOYEE_ID_PK | P |
| EMPLOYEE_DEPT_ID_FK | R |

5. Recherchez le nom et le type des objets dans la vue USER_OBJECTS du dictionnaire de données correspondant aux tables EMPLOYEE et DEPARTMENT. Vous pouvez mettre les colonnes en forme pour qu'elles soient plus lisibles. Remarquez que pour chaque nouvelle table un nouvel index a été créé.

| OBJECT_NAME | OBJECT_TYPE |
|------------------|-------------|
| ----- | ----- |
| DEPARTMENT | TABLE |
| DEPARTMENT_ID_PK | INDEX |
| EMPLOYEE | TABLE |
| EMPLOYEE_ID_PK | INDEX |

Si vous avez le temps, faites l'exercice suivant :

6. Modifiez la table EMPLOYEE. Ajoutez une colonne SALARY dont le type de donnée est NUMBER, avec une longueur 7.