

GUIDE DES CONTRAINTES SQL



Table des Matières

1. [Types de Contraintes](#)
 2. [Gestion des Contraintes](#)
 3. [Bonnes Pratiques](#)
-

Types de Contraintes {#types}

1 NOT NULL

Rôle: Empêche les valeurs NULL

```
/* Exemple 1 - Dans CREATE TABLE */
CREATE TABLE Employes (
    id NUMBER(4),
    nom VARCHAR2(50) NOT NULL, --  Nom obligatoire
    email VARCHAR2(100) NOT NULL --  Email obligatoire
);

/* Exemple 2 - Modifier une colonne existante */
ALTER TABLE Employes
MODIFY telephone NOT NULL;
```

2 UNIQUE

Rôle: Garantit des valeurs uniques

```
/* Exemple 1 - Niveau colonne */
CREATE TABLE Departements (
    code_dept VARCHAR2(5),
    nom VARCHAR2(50) CONSTRAINT uk_dept_nom UNIQUE
);

/* Exemple 2 - Niveau table */
CREATE TABLE Employes (
    id NUMBER(4),
    email VARCHAR2(100),
    CONSTRAINT uk_emp_email UNIQUE(email)
);
```

3 PRIMARY KEY

Rôle: Identifiant unique d'une table

```
/* Exemple 1 - Clé primaire simple */
CREATE TABLE Clients (
    id_client NUMBER(4),
    nom VARCHAR2(50),
    CONSTRAINT pk_client PRIMARY KEY(id_client)
);

/* Exemple 2 - Clé primaire composite */
CREATE TABLE Commandes (
    num_commande NUMBER(4),
    date_commande DATE,
    CONSTRAINT pk_commande PRIMARY KEY(num_commande)
);
```

4 FOREIGN KEY

Rôle: Établit des relations entre tables

```
/* Exemple 1 - Référence simple */
CREATE TABLE Commandes (
    id_commande NUMBER(4),
    id_client NUMBER(4),
    CONSTRAINT fk_client
    FOREIGN KEY (id_client)
    REFERENCES Clients(id_client)
);

/* Exemple 2 - Avec option CASCADE */
CREATE TABLE Details_Commande (
    id_commande NUMBER(4),
    id_produit NUMBER(4),
    CONSTRAINT fk_commande
    FOREIGN KEY (id_commande)
    REFERENCES Commandes(id_commande)
    ON DELETE CASCADE
);
```

5 CHECK

Rôle: Vérifie une condition

```
/* Exemple 1 - Contrôle simple */
CREATE TABLE Produits (
    id_produit NUMBER(4),
    prix NUMBER(10,2),
    CONSTRAINT chk_prix CHECK (prix > 0)
);

/* Exemple 2 - Contrôle multiple */
CREATE TABLE Employes (
    id NUMBER(4),
    age NUMBER(3),
    salaire NUMBER(8,2),
    CONSTRAINT chk_emp CHECK (age >= 18 AND salaire >= 1500)
);
```

⚙️ Gestion des Contraintes {#gestion}

1 Ajouter une Contrainte

```
/* Sur table existante */
ALTER TABLE Employes
ADD CONSTRAINT pk_emp PRIMARY KEY(id_emp);
```

2 Supprimer une Contrainte

```
/* Suppression simple */
ALTER TABLE Employes
DROP CONSTRAINT pk_emp;

/* Suppression avec CASCADE */
ALTER TABLE Employes
DROP PRIMARY KEY CASCADE;
```

3 Désactiver/Activer une Contrainte

```
/* Désactiver */  
ALTER TABLE Employes  
DISABLE CONSTRAINT pk_emp;  
  
/* Activer */  
ALTER TABLE Employes  
ENABLE CONSTRAINT pk_emp;
```

4 Vérifier les Contraintes

```
/* Liste des contraintes d'une table */  
SELECT constraint_name, constraint_type, search_condition  
FROM user_constraints  
WHERE table_name = 'EMPLOYES';
```

Bonnes Pratiques {#pratiques}

Conventions de Nommage

- **PK_**: Pour les Primary Keys
- **FK_**: Pour les Foreign Keys
- **UK_**: Pour les Unique Keys
- **CHK_**: Pour les Check Constraints

Points Importants

1. ☒ Toujours nommer explicitement vos contraintes
 2. ☒ Privilégier les contraintes au niveau table
 3. ☒ Documenter les contraintes complexes
 4. ☒ Vérifier l'impact avant de modifier/supprimer
 5. ☒ Utiliser CASCADE avec précaution
-

Exemples Pratiques

Cas d'Utilisation Complet

```
CREATE TABLE Categories (  
    id_categorie NUMBER(4),  
    nom VARCHAR2(50) NOT NULL,  
    description VARCHAR2(200),  
    CONSTRAINT pk_categorie PRIMARY KEY(id_categorie),  
    CONSTRAINT uk_cat_nom UNIQUE(nom)  
);  
  
CREATE TABLE Produits (  
    id_produit NUMBER(4),  
    nom VARCHAR2(50) NOT NULL,  
    prix NUMBER(10,2),  
    id_categorie NUMBER(4),  
    CONSTRAINT pk_produit PRIMARY KEY(id_produit),  
    CONSTRAINT fk_prod_cat  
        FOREIGN KEY (id_categorie)  
        REFERENCES Categories(id_categorie),  
    CONSTRAINT chk_prix CHECK (prix > 0)  
);
```