

# 420-KHG-LG, Semaine 03- Séance2

## Retour sur la commande CREATE TABLE.

### Requêtes avec jointure

Une jointure est une opération relationnelle qui sert à chercher des lignes ou des enregistrements à partir de deux ou plusieurs tables disposant d'un ensemble de valeur communes, en général les clés primaires.

#### Produit cartésien

Le produit cartésien est une requête de sélection qui met en jeux plusieurs tables. Pour deux tables, la sélection consiste à afficher la première ligne de la première table avec toutes les lignes de la deuxième table, puis la deuxième ligne de la première table avec toutes les lignes de la deuxième table et ainsi de suite. Ce type de sélection implique beaucoup de redondances.

Exemple

```
SELECT NOM, PRENOM, NOMPROG  
FROM ETUDIANTS,PROGRAMME;
```

| NOM     | PRENOM   | NOMPROG        |
|---------|----------|----------------|
| PATOCHE | ALIAN    | INFORMATIQUE   |
| PATOCHE | ALIAN    | ADMINISTRATION |
| PATOCHE | ALIAN    | ELECTRONIQUE   |
| FAFAR   | CHANTALE | INFORMATIQUE   |
| FAFAR   | CHANTALE | ADMINISTRATION |
| FAFAR   | CHANTALE | ELECTRONIQUE   |
| MARTIN  | RACHA    | INFORMATIQUE   |
| MARTIN  | RACHA    | ADMINISTRATION |
| MARTIN  | RACHA    | ELECTRONIQUE   |

.... (il y a une suite à la sortie de la requête)

### Jointure simple:

Une jointure simple consiste est un produit cartésien avec un INNER JOIN faisant ainsi une restriction sur les lignes. La restriction est faite sur l'égalité de la valeur de deux champs (cas de deux tables) qui sont généralement les clés primaires.

Exemple

```
SELECT NOM, PRENOM, NOMPROG
FROM ETUDIANTS E INNER JOIN PROGRAMME P
ON E.CODEPRG =P.CODEPRG;
```

| R | NOM     | R | PRENOM   | R | NOMPROG        |
|---|---------|---|----------|---|----------------|
|   | PATOCHE |   | ALIAN    |   | INFORMATIQUE   |
|   | FAFAR   |   | CHANTALE |   | INFORMATIQUE   |
|   | MARTIN  |   | RACHA    |   | ADMINISTRATION |
|   | PLOUFFE |   | STEVEN   |   | ELECTRONIQUE   |
|   | ALLARD  |   | MATHIEU  |   | ADMINISTRATION |
|   | VIENS   |   | NATHALIE |   | ADMINISTRATION |
|   | DANIS   |   | SAMUEL   |   | ELECTRONIQUE   |
|   | FAVRE   |   | NICOLAS  |   | INFORMATIQUE   |
|   | JACOB   |   | YANICK   |   | INFORMATIQUE   |

Autre exemple

```
SELECT nom,prenom, description, note
FROM ((etudiant E INNER JOIN NOTE ON E.numad = R.numad)
INNER JOIN cours C ON C.code_cours = R.code_cours);
```

### Important

- Lorsqu'un champ (colonne) sélectionné est présent dans plus d'une table alors il faut le précéder du nom de la table à partir de laquelle on désire l'extraire.
- Vous pouvez donner un alias aux noms de tables afin de faciliter la référence aux tables. Cependant si un alias est donné alors, il faudra utiliser l'alias à la place du nom de la table.

- Les attributs qui apparaissent dans la jointure ne sont pas nécessairement dans le SELECT.
- Toutes les tables dont les champs apparaissent dans la clause SELECT ou dans la clause WHERE doivent apparaître dans la clause FROM.

#### Exemple

```
SELECT J.NUMJOUEUR, NOM, PRENOM FROM JOUEURS J INNER JOIN
STATISTIQUES S
ON J.NUMJOUEUR = S.NUMJOUEUR
WHERE NBBUTS >0
GROUP BY J.NUMJOUEUR, NOM, PRENOM;
```

#### Jointure externe

Jointure externe droite: Dans la jointure externe droite, des enregistrements de table de la première table seront ramenés même si ceux-ci n'ont pas d'occurrences dans la deuxième table.

Jointure externe gauche: Dans la jointure externe gauche des enregistrements de table de deuxième table seront ramenés même si ceux-ci n'ont pas d'occurrences dans la première table.

Dans le cas d'une jointure externe, il faut faire suivre la colonne pour laquelle il n'est pas obligatoire d'avoir des lignes correspondant à l'égalité par l'opérateur LEFT OUTER JOIN ou RIGHT OUTER JOIN

#### Exemple

Cette requête ramène tous les étudiants y compris ceux qui ne sont pas inscrits dans un programme

```
SELECT NOM, PRENOM, NOMPROG
FROM ETUDIANTS E LEFT OUTER JOIN PROGRAMME P ON
E.CODEPRG=P.CODEPRG;
```

| RESPONSABLE | EMPLOYE   |
|-------------|-----------|
| alibaba     | Martin    |
| alibaba     | Patoche   |
| alibaba     | Cristophe |
| Patoche     | FAFAR     |
| Cristophe   | ROY       |
| Patoche     | SIMPSON   |

## IMPORTANT:

Dans le cas d'une jointure externe, il faut faire suivre la colonne pour laquelle il n'est pas obligatoire d'avoir des lignes correspondant à l'égalité par l'opérateur LEFT OUTER JOIN ou RIGHT OUTER JOIN

Lorsqu'un champ sélectionné est présent dans plus d'une table alors il faut le précéder du nom de la table à partir de laquelle on désire l'extraire.

Exemple

Vous pouvez donner un alias aux noms de tables afin de faciliter la référence aux tables. Cependant si un alias est donné alors, il faudra utiliser l'alias à la place du nom de la table.

## Exercice

En utilisant les tables que vous avez créés au dernier lab, c'est-à-dire : Etudiants, programmes, cours et resultats, écrire les requêtes suivantes :

1. Liste des étudiants (nom, prenom, nomprogramme)
2. Liste des étudiants du programme informatique
3. Liste des cours des programmes informatiques
4. Les notes des étudiants (nom et prénom) en informatique dans le cours  
Intégration des bases de données,
5. Liste des étudiants (nom, prenom, titre Cours, note) ayant une notes plus élevée 70
6. Liste des étudiants numad, nom, prenom ayant une note plus petite que 70 dans le cours Intégration des bases de données.

7. Donner le nombre d'étudiants par programme (utilisez le nomprogramme au lieu du codeprogramme)
8. Quelle est la moyenne des notes par cours ?
  - a. Utilisez le code cours
  - b. Utilisez le titre du cours
  - c. Compléter la requête précédente pour quelle affiche la moyenne des notes du cours «Intégration des bases de données »
9. Lise de tous les étudiants nom et prénoms avec le nom du programme même les étudiants qui sont dans aucun programme.
10. Lise de tous les programmes (nomprogramme) avec les étudiants qui y sont inscrits. On doit afficher même les programmes où il n'y a aucun étudiant.

# Modification de la définition d'une table : La commande ALTER TABLE

Il est parfois nécessaire de modifier la structure d'une table, la commande ALTER TABLE sert à cela. Cette commande change la structure de la table mais pas son contenu.

Les types de modifications acceptées sont les suivants:

- ✓ Ajout d'une nouvelle colonne à la table avec ses contraintes
- ✓ Augmente ou diminue la largeur d'une colonne existante
- ✓ Changer la catégorie d'une colonne, d'obligation à optionnelle ou vice versa (NOT NULL à NULL ou vice versa)
- ✓ Spécification d'une valeur par défaut pour une colonne existante
- ✓ Changer le type de données d'une colonne existante
- ✓ Spécification d'autres contraintes pour une colonne existante
- ✓ Activer ou désactiver une contrainte
- ✓ Détruire une contrainte.

## L'option ADD

Cette option permet d'ajouter une colonne ou une contrainte à une table existante.

Attention!!

Si la table contient des valeurs, alors la colonne ajoutée doit être mise à jour.

Si une contrainte est ajoutée à une colonne alors que celle-ci contient déjà des données qui ne correspondent pas à la contrainte, alors la modification de la structure de la table sera refusée.

Exemples :

Voici la commande CREATE initiale pour la table Employes

```
CREATE TABLE Employes(NumEmp number, nom varchar2(15), prenom  
varchar2(20));
```

```
ALTER TABLE Employes ADD (Salaire NUMBER (8,2));
```

Permet d'ajouter la colonne Salaire à la table Employes

```
ALTER TABLE Employes ADD CONSTRAINT emppk PRIMARY KEY  
(NumEmp);
```

Permet d'ajouter une contrainte de clé primaire sur la colonne Numemp de la table Employes

### **L'option MODIFY:**

Cette option permet de modifier le type de données, la valeur par défaut et la contrainte de NOT NULL sur une table déjà existante. Il est impossible de raccourcir la taille d'une colonne (la longueur des données) si celle-ci contient des données.

```
ALTER TABLE Employes MODIFY (nom NOT NULL);
```

L'option ENABLE /DISABLE

Cette option sert à activer ou désactiver une contrainte.

```
ALTER TABLE Employes DISABLE Primary Key;
```

### **L'option DROP**

Cette option sert à supprimer une contrainte sur une table déjà existante. Lorsque

```
ALTER TABLE Employes DROP Primary Key;
```

Ou

```
ALTER TABLE Employes DROP CONSTRAINT emppk;
```

```
ALTER TABLE Employes DROP COLUMN nom;
```

```
ALTER TABLE Employes RENAME COLUMN Salaire TO SalaireEmp;
```

## **Supprimer une table : la commande <DROP TABLE>**

La commande DROP permet de supprimer un objet de la base de données. (Table, indice, synonyme..)

```
DROP TABLE personne;
```

# Renommer une table : la commande <RENAME>

Permet de renommer une table ou un objet de la base de données

Syntaxe

```
RENAME <Ancien_nom> TO <Nouveau_nom>;
```

```
RENAME Employes TO EmployesInfo;
```

## Exercice 1 : Travailler avec la commande ALTER TABLE

1. Ajouter la colonne CODEEQUIPE de type CHAR(3) à votre table JOUEURS ;
2. Modifier la table pour que le CODEEQUIPE prenne ses valeurs dans (MTL, OTT, TOR, VAN)
3. Faire en sorte que le CODEEQUIPE soit par défaut à MTL.
4. Effectuer une insertion pour vérifier.
5. Mettre à jour les valeurs de cet attribut pour toute la table.
6. Changer le nom de la colonne CODEEQUIPE par CODEVILLE.
7. Écrire la requête qui permet de supprimer la colonne CODEVILLE.
8. Quelle la requête qui vous permet de détruire la contrainte sur le salaire.
9. Quelle est la requête qui supprime la table JOUEURS. Supprimer la table JOUEURS.