Cours les jointures entre les tables

1. **Contexte**

Dans une école, un étudiant écrit par un nom, prénom, email, montant, statut est inscrit dans une et une seule promotion connue par un designation et une salle. Une promotion prépare un et un seul diplôme décrit par un code, un libellé et un nombre d’années.

* **Modéliser ce système d’information.**



* **Elaboration du MLDR**

Diplôme (code, libellé, nbAnnees)

Promotion (idPromotion, désignation, salle, code#)

Etudiant (idEtudiant, nom, prénom, email, montant, statut, idPromotion#)

* **Script SQL de la base de données**

| drop database if exists Ecole\_284; create database Ecole\_284; use Ecole\_284; create table diplome ( code int (3) not null auto\_increment,  libelle varchar (50),  nbAnnees int (2),  primary key (code));create table promotion ( idPromotion int (3) not null auto\_increment,  designation varchar (50),  salle varchar(50),  code int(3) not null, primary key (idPromotion),  foreign key (code) references diplome (code)); create table etudiant ( idEtudiant int (3) not null auto\_increment,  nom varchar (50),  prenom varchar(50),  email varchar(50), montant float,  statut varchar (50),  idPromotion int(3) not null, primary key (idEtudiant),  foreign key (idPromotion) references promotion (idPromotion));  |
| --- |

* **Insertion des données**

| **insert into diplome values (null, "BTS SIO", 2), (null, "Bachelor", 3),** **(null, "Mastere", 5);** **insert into promotion values (null, "Promo 284", "Salle 1D",1),** **(null, "Promo 250", "Salle 4D", 2), (null, "Promo 266", "Salle 3G", 1);** **insert into etudiant values (null, "Allan", "Ismail", "a@gmail.com",** **5000, "initial", 1), (null, "Maria", "Asmae", "b@gmail.com",** **0, "Alternant", 2),(null, "Melissa", "Adel", "m@gmail.com",** **5000, "initial", 1), (null, "Paul", "Pierre", "p@gmail.com",** **0, "Alternant", 3);**  |
| --- |

1. **Jointures**
* **Quelle serait la requête pour afficher les noms et prénoms des étudiants avec la désignation et la salle de leurs promotions ?**

Cette requête doit afficher deux attributs de la table Etudiant (nom, prénom) et deux attributs de la table Promotion (désignation, salle). Elle accède à deux tables.

Pour réaliser cette requête, nous devons faire une jointure entre la table Etudiant et la table Promotion via les clés primaires et les clés étrangères.

Dans la table Etudiant, il y a la clé étrangère venant de la table Promotion qui est idPromotion.

L’écriture de la requête est la suivante :

Select Etudiant.nom, Etudiant.prenom, Promotion.designation, Promotion.salle

From Etudiant, Promotion

Where Promotion.idPromotion = Etudiant.idPromotion ;



On peut utiliser des alias pour les tables afin de réduire l’écriture de la requête :

Select e.nom, e.prenom, p.designation, p.salle

From Etudiant e , Promotion p

Where p.idPromotion = e.idPromotion ;

On peut donner des alias aux attributs surtout quand il y aura des ambiguités de noms :

Select e.nom as nomEtudiant, e.prenom as prenomEtudiant, p.designation as Promotion, p.salle as salleCours

From Etudiant e , Promotion p

Where p.idPromotion = e.idPromotion ;



* Lister les promotions (désignation) avec la liste des diplômes (libellé et nbAnnées) qu’elles préparent.

Select p.designation, d.libelle, d.nbAnnees

from promotion p, diplome d

where p.code = d.code ;



* Dans quelle promotion (désignation et salle) se trouve l’étudiant Allan ?

Select p.designation, p.salle

From promotion p, etudiant e

where e.idpromotion = p.idpromotion

and nom = **"**Allan**"** ;



* **Lister les étudiants (nom, prénom, statut) de la promotion Promo 284.**

Select e.nom, e.prenom, e.statut

From etudiant e, promotion p

where e.idpromotion = p.idpromotion

and p.designation = **"**Promo 284**"** ;



* Afficher les étudiants (nom, prénom et statut) avec le libellé des diplômes et le nombre d’années du diplôme préparé.

Select e.nom, e.prenom, e.statut, d.libelle, d.nbAnnees

From etudiant e, promotion p, diplome d

Where e.idpromotion = p.idpromotion

And p.code = d.code ;



* **Quels sont les étudiants (nom, prénom) qui préparent un diplôme de BTS SIO ?**

Select e.nom, e.prenom

From etudiant e , promotion p , diplome d

Where e.idpromotion = p.idpromotion

And p.code = d.code

And d.libelle = **"**BTS SIO**"**;



1. **LA CLAUSE GROUP BY**

On utilise le group by pour rassembler des résultats par tranche ou par ensemble d'enregistrements d’une table ou de plusieurs tables.

* **Utilisation du group by dans la requête count et sum.**

**Afficher le nombre d’étudiants par statut.**

select statut, count(\*) as nbEtudiants

from etudiant

group by statut ;



**Afficher le nombre d’étudiants par Promotion**

select p.designation, count(e.idetudiant) as nbEtudiants

From promotion p, etudiant e

where p.idpromotion = e.idpromotion

group by p.designation ;



**Afficher le total des montants par promotion :**

Select **p.designation,** sum(e.montant) as total

From promotion p, etudiant e

where p.idpromotion = e.idpromotion

group by **p.designation** ;



**Afficher le nombre de promotions par diplôme :**

Select d.libelle , count(p.idpromotion) as nbPromotions

From diplome d, promotion p

Where d.code = p.code

Group by d.libelle;

