# Exercices sur les jonctions de tables

Pour les numéros suivants, utilisez les tables « departement » et « employe »

### Exercice #1

Faire afficher pour chaque employé son nom, son numéro d’employé et la ville dans laquelle il travaille.

SELECT employe.nom, employe.id, ville

FROM employe, departement

WHERE employe.id\_departement = departement.id

### Exercice #2

Soit l’énoncé suivant :

SELECT

\*

FROM

employe,

departement

WHERE

employe.id\_departement = departement.id;

Quel est l’effet d’inverser l’ordre des tables (entre le FROM et le WHERE) dans cet énoncé ?

L’ordre d’affichage des colonnes est inversé

Inverser l’ordre des champs dans le WHERE dans l’énoncé du numéro précédent. Quel est l’effet de cette inversion ?

Aucun

Modifier la requête afin d’ajouter aux résultats les employés qui n’appartiennent à aucun département.

SELECT employe.nom, employe.id, departement.ville

FROM employe LEFT JOIN department ON employe.id\_departement = departement.id

Modifier la requête afin d’ajouter aux résultats non seulement les employés qui n’appartiennent à aucun département, mais les départements qui n’ont aucun employé.

SELECT employe.nom, employe.id, ville

FROM employe

FULL JOIN departement ON employe.id\_departement = departement.id;

### Exercice #3

Afficher la liste des employés qui travaillent dans une ville autre que Montréal. Inclure dans la réponse le nom de l'employé et le nom de la ville. Afficher en ordre de nom d’employé.

SELECT employe.nom, ville

FROM employe, departement

WHERE employe.id\_departement = departement.id AND

ville != 'Montreal'

ORDER BY employe.nom;

Ajouter à la requête la liste des employés qui n’appartiennent à aucun département. Inclure dans la réponse le nom de l'employé et le nom de la ville.

SELECT employe.nom, ville

FROM employe LEFT JOIN departement ON employe.id\_departement = departement.id

WHERE Ville != 'Montreal' OR Ville IS NULL

ORDER BY employe.nom;

Même question, mais afficher en plus les villes dans lesquelles sont les départements qui n’ont aucun employé.

SELECT employe.nom, ville

FROM employe **FULL** JOIN departement ON employe.id\_departement = departement.id

WHERE Ville != 'Montreal'

OR Ville IS NULL

ORDER BY employe.nom;

### Exercice #4

Afficher la liste des employés qui ont un patron qui n'est pas dans le même département qu’eux. Pour chaque employé fournir son nom et le nom de son patron.

SELECT E.nom, P.nom

FROM employe E, employe P

WHERE (E.id\_patron = P.id)

AND E.id\_departement != P.id\_departement;

-- Pour sortir Roy : ( e.id\_departement != p.id\_departement or p.id\_departement is null)

### Exercice #5

Afficher la liste des employés et leur patron ainsi que le nom du département de chacun.

1 – Commencer par trouver l'employé et son département

SELECT e1.nom, d1.nom

FROM employe e1 LEFT JOIN departement d1 ON e1.id\_departement = d1.id

2- Le lier avec son patron (s'il en a un)

SELECT e1.nom, d1.nom, e2.nom

FROM employe e1 LEFT JOIN departement d1 ON e1.id\_departement = d1.id

LEFT JOIN employe e2 ON e1.id\_patron = e2.id

3- Lier le patron avec son département

SELECT e1.nom, d1.nom, e2.nom, d2.departement

FROM employe e1 LEFT JOIN departement d1 ON e1.id\_departement = d1.id

LEFT JOIN employe e2 ON e1.id\_patron = e2.id

LEFT JOIN departement d2 ON e2.id\_departement = d2.id

### Exercice #6

Écrire l'énoncé nécessaire pour trouver toutes les paires de valeurs <employé,employé> tel que le deuxième employé gagne un salaire plus élevé que le premier. Pour chaque employé, fournir son nom et son salaire.

SELECT e1.nom, e1.salaire, e2.nom, e2.salaire

FROM employe e1, employe e2

WHERE e1.salaire < e2.salaire

### Exercice #7

Ajouter l’employé suivant dans la table employe : Demers, id 1234, département 20, id patron : 7782.

Écrire l'énoncé nécessaire pour trouver le nom des départements qui ont un employé qui porte le même nom qu'un employé d'un autre département.

Ici on veut afficher 3 colonnes : Le nom des deux départements et le nom de l’employé

1- Trouver un empoye et son departement

SELECT e1.nom, d1.nom

FROM

employe e1,

departement d1

WHERE

e1.id\_departement = d1.id

2- Trouver un employe du meme nom

SELECT e1.nom, d1.nom

FROM

employe e1,

departement d1,

employe e2

WHERE

e1.id\_departement = d1.id AND

e1.nom = e2.nom

3- Trouver le département du deuxième employé

SELECT e1.nom, d1.nom, d2.nom

FROM

employe e1,

departement d1,

employe e2,

departement d2

WHERE

e1.id\_departement = d1.id AND

e2.id\_departement = d2.id AND

e1.nom = e2.nom AND

d1.id != d2.id;

(Retourne deux lignes : les deux Demers)

## Interrogation de la base de données « FPA »

Pour les exercices suivants, utilisez la base de données FPA.

Cette base de données contient 4 tables:

* Produit : Liste de produits
* Fournisseur : Liste de fournisseurs créant les produits
* Atelier : Atelier où sont livrés les produits
* FPA : Lien entre les trois premières tables : Le fournisseur X créé un produit Y pour l'atelier Z

### Exercice #1

Trouver le numéro des produits fournis à un atelier de Montréal par un fournisseur de Montréal.

SELECT Fpa.Nop

FROM Fpa, Atelier, Fournisseur

WHERE Fpa.Noa = Atelier.Noa

AND Fpa.Nof = Fournisseur.Nof

AND Atelier.Ville = 'Montreal'

AND Fournisseur.Ville = 'Montreal';

(Rép : P6)

### Exercice #2

Trouver le numéro des ateliers qui reçoivent au moins un produit d'un fournisseur situé dans la même ville que celle de l'atelier.

SELECT DISTINCT Fpa.Noa

FROM Fpa, Atelier, Fournisseur

WHERE Fpa.Noa = Atelier.Noa

AND Fpa.Nof = Fournisseur.Nof

AND Atelier.Ville = Fournisseur.Ville

(Rép : A1, A4, A7)

### Exercice #3

21.Trouver toutes les valeurs <ville,produit,ville> tel qu'un fournisseur de la première ville fournisse le produit à un atelier de la deuxième ville. Par exemple :

1 Montreal crayon Quebec

2 Montreal crayon Chicoutimi

3 Quebec carton Quebec

4 Quebec carton Rimouski

5 Quebec carton Chicoutimi

………..

SELECT Fournisseur.Ville,Produit.NomP,Atelier.Ville

FROM Fpa, Atelier, Fournisseur, Produit

WHERE Fpa.Noa = Atelier.Noa

AND Fpa.Nof = Fournisseur.Nof

AND Fpa.Nop = Produit.Nop

(Résultat : 24 lignes)

Même question, mais avec la contrainte suivante : les deux villes doivent être différentes

SELECT Fournisseur.Ville, Produit.NomP, Atelier.Ville

FROM Fpa

JOIN Atelier USING (Noa)

JOIN Fournisseur USING (Nof)

JOIN Produit USING (Nop)

WHERE Atelier.Ville <> Fournisseur.Ville

## Interrogation de la base de données « Commandes »

Pour les exercices suivants, utilisez la base de données « Commandes ».

### Exercice #1

Faire afficher le nom et le prénom des employés (table employe**S**) qui ont effectué une vente (table commandes) à des clients de Paris. On veut aussi le nom de la société des clients de Paris.

SELECT DISTINCT Nom, Prenom, Societe

FROM Employes A, Commandes B, Clients C

WHERE A.No\_Employe = B.No\_Employe

AND

B.Code\_Client = C.Code\_Client

AND

Ville = ‘Paris’ ;

### Exercice #2

Faire afficher le nom de la catégorie du produit, la société fournisseur, le nom du produit et le code da la catégorie uniquement pour les produits des catégories 1,4 et 7.

SELECT Nom\_Categorie, Societe, Nom\_Produit, Categories.Code\_Categorie

FROM Categories, Produits, Fournisseurs

WHERE Produits.Code\_Categorie = Categories.Code\_Categorie

AND Produits.No\_Fournisseur = Fournisseurs.No\_Fournisseur

AND Produits.Code\_Categorie IN (1,4,7) ;

### Exercice #3

Faire afficher la société cliente, la société fournisseur et leur ville pour les clients qui sont localisés dans une ville d’un fournisseur (Il s’agit d’une jonction entre la table Clients et Fournisseurs). Ajouter des ALIAS pour Société clients et Société Fournisseurs.

SELECT Fournisseurs.Societe "Société fournisseurs", Clients.Societe "Société Clients", Fournisseurs.Ville

FROM Fournisseurs, Clients

WHERE Fournisseurs.Ville = Clients.Ville ;

### Exercice #4

Faire afficher le nom des sociétés clientes qui ont commandé le produit ‘Chai’. Faire aussi afficher la ville dans laquelle est la société cliente.

SELECT Societe, Ville, Produits.Nom\_Produit

FROM Clients, Commandes, Details\_Commandes, Produits

WHERE Clients.Code\_Client = Commandes.Code\_Client

AND

Commandes.No\_Commande = Details\_Commandes.No\_Commande

AND

Details\_Commandes.Ref\_Produit = Produits.Ref\_Produit

AND

Produits.Nom\_Produit = 'Chai' ; (Résultat : 38 lignes)