

Environnement

- MySQL Server
- MySQL Workbench

Directive

- Sauvegardez vos réponses texte et vos instructions SQL dans des fichiers aux noms module04_ex1.sql

Exercice 1 - Employés

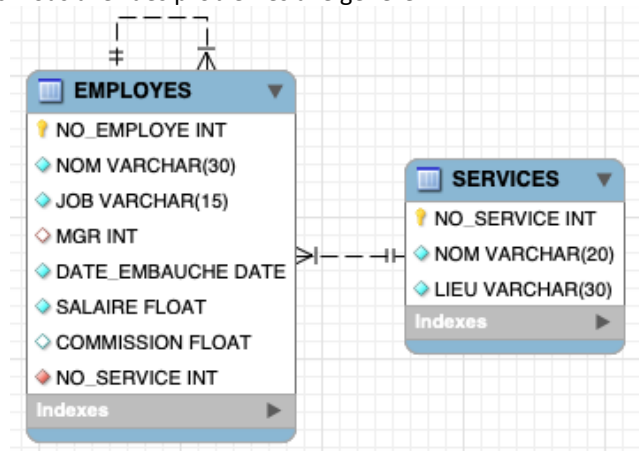
Pour cet exercice, vous allez mettre votre code SQL dans le fichier `module04_ex1.sql`.

Cet exercice traite d'une base de données d'employés. Les employés sont assignés à un service. Les employés ont un responsable qui est aussi un employé. Comme dans toute hiérarchie, au moins un des employés ne peut pas avoir de responsable car il est en fait de la pyramide.

Exercice 1.1 - Création de la base de données

1. Analysez les requêtes proposées dans le fichier "Code SQL" 41_module04_ex1.sql
2. Si le code vous semble sans danger, exécutez-le sur votre environnement.
3. Afin de bien analyser cette base de données, générez le diagramme

Schéma de la base de données si vous avez des problèmes à le générer :



4. Créez les enregistrements correspondants aux données demandées :

- Données de la table SERVICES

NO_SERVICE	NOM	LIEU
10	Comptabilité	Toronto
20	Recherche	Québec
30	Ventes	Montréal
40	Opérations	Québec

- Données de la table EMPLOYES

NO_EMPLOYE	NOM	JOB	MGR	DATE_EMBAUCHE	SALAIRE	COMMISSION	NO_SERVICE
7839	Musk	Président		1981-11-17	5000.00		10
7566	Murdock	Manager	7839	1981-04-02	2975.00		20
7902	Grim	Analyste	7566	1981-03-12	3000.00		20
7369	Baillargeon	Employe	7902	1980-07-12	800.00		20
7698	Pianetti	Manager	7839	1981-05-01	2850.00		30
7499	Franchi	Vendeur	7698	1981-02-20	1600.00	300.00	30
7521	Ouvrard	Vendeur	7698	1981-02-22	1250.00	500.00	30
7654	Martin	Vendeur	7698	1981-09-28	1250.00	1400.00	30
7782	Deleglise	Manager	7839	1981-06-09	2450.00		10
7788	Gorlin	Analyste	7566	1982-09-12	3000.00		20
7844	Hulk	Vendeur	7698	1981-09-08	1500.00	0.00	30
7876	Parker	Employe	7788	1983-12-01	1100.00		20
7900	Xavier	Employe	7698	1981-03-12	950.00		30
7934	Lee	Employe	7782	1982-01-23	1300.00		10

5. Expliquez pourquoi on ne peut pas insérer les données dans n'importe quel ordre (exemple : pourquoi ne peut-on pas insérer 'Musk' en dernier ?)

- Comment auriez-vous pu faire pour insérer les données dans l'ordre des NO_EMPLOYE ?
- Testez votre hypothèse en ajoutant l'employé 'Flash' puis 'Wells', 'Wells' étant le responsable de 'Flash' :

NO_EMPLOYE	NOM	JOB	MGR	DATE_EMBAUCHE	SALAIRE	COMMISSION	NO_SERVICE
8123	Flash	Analyste	8635	1979-11-23	1500.00		20
8635	Wells	Manager	7839	1981-04-02	4500.00		20

- Créez une requête pour afficher les données sous la forme d'un rapport (Attention aux noms de colonnes) comme ci-dessous (on veut seulement ces 8 lignes) :

No. employé	Nom	Role	Responsable	Date d'embauche	Salaire	Commission	Commission2	Numéro de service
7844	Hulk	Vendeur	7698	1981-09-08	1500	0	0	30
7499	Franchi	Vendeur	7698	1981-02-20	1600	300	300	30
7521	Ouvrard	Vendeur	7698	1981-02-22	1250	500	500	30
7654	Martin	Vendeur	7698	1981-09-28	1250	1400	1400	30
7839	Musk	Président	NULL	1981-11-17	5000	NULL	Aucune	10
7566	Murdock	Manager	7839	1981-04-02	2975	NULL	Aucune	20
8635	Wells	Manager	7839	1981-04-02	4500	NULL	Aucune	20
7698	Pianetti	Manager	7839	1981-05-01	2850	NULL	Aucune	30

Exercice 1.2 - Sélections simples

- Listez les noms et salaires des personnes travaillant dans le département (no service) 10
- Listez les noms et salaires des personnes travaillant dans les départements 10 ou 20
- Listez les noms, salaires et commissions des personnes qui ont reçues une commission. (Attention, validez que la commission n'est pas de 0)
- Listez les noms, salaires et commissions des vendeurs qui ont une commission indéfinie (NULL) ou = 0
- Listez les managers dont le salaire est supérieur à 2500
- Listez les personnes dont le salaire est compris entre 1500 et 2000 (bornes incluses)
- Vous vous êtes peut-être posé la question suivante : qu'est-ce qu'un manager ? Pensez-vous qu'il y a une ambiguïté entre la colonne JOB et MGR. Si oui, sans le mettre en oeuvre, proposez une piste de solution (La solution sera discutée plus loin dans la session)

Exercice 1.3 - Sélection avec opérateurs

Certains de ces opérateurs ont déjà été utilisés dans la section précédente. Nous allons nous focaliser sur les nouveautés dans les questions suivantes, à savoir les opérateurs BETWEEN, LIKE, CASE, WHEN, IN, IS, BINARY, COLLATE et REGEXP.

La signification des opérateurs ainsi que leur priorité est présentée ci-après, en partant de la priorité la plus basse vers la plus haute. Les opérateurs dans la même case ayant la même priorité.

Opérateur	Signification	Exemples
OR () XOR	OU logique inclusif OU logique exclusif	((nom :='MURDOCK')OR(no=1111) => MURDOCK
AND (&&)	ET logique	((nom :='MURDOCK')AND(no=1111) => rien
BETWEEN	Intervalle de deux valeurs	salaire BETWEEN 5000 AND 6000 => CHIRAC
CASE WHEN ... THEN ... ELSE ... WHEN END	Conditionnelle multiple Si... alors ...sinon	SELECT CASE jour WHEN 1 THEN 'LUNDI' WHEN 2 THEN 'MARDI' END
=, <=>, >=, >,<=, <,<>,! =	Comparaison	Présenté dans la section théorie
IS	Test booléen	1 IS TRUE => 1 (Car 1 est Vrai) Valeurs testées : TRUES, FALSE ou UNKNOWN
LIKE	Égalité incomplète avec caractère Joker	Nom LIKE 'MA%' => MARTIN Nom LIKE 'M_R%' => MARTIN + MURDOCK
REGEXP	Expression régulière	REGEXP '([7-9]{4})' -> Les nombres à 4 chiffres. Voir figure \ref{figure3}
-, +	Addition, soustraction<	colonne1 + colonne2 = Nouvelle valeur (Utilisez les alias (AS))
*, /	Multiplication, division	colonne1 / colonne2 = Nouvelle valeur (utiliser les alias)
DIV %, MOD	Division entière Modulo	5 DIV 2 => 2 5 MOD 2 => 1 (reste de 5 DIV 2)
NOT (!)	Négation	NOT (x=y) <=> X!=Y
BINRARY	Rend sensible à la casse et aux espaces	BINARY(Colonne1) = "abc"

1. Sélectionnez la liste des employés gagnant entre 1500 et 2000 (mais sans utiliser <= et >=)
2. Listez les employés dont le nom comporte un 'L'

3. Listez les employés dont le nom comporte deux 'L'
4. Sélectionnez les employés ayant un 'L' en troisième position dans leur nom
5. Listez les employés qui sont soit managers, soit vendeurs, soit employés
6. En utilisant l'opérateur CASE, affichez les noms et le libellé 'Cadre' pour les employés 'Manager' ou 'President' et 'Non cadre' pour les autres.
7. Affichez les noms, les salaires, plus une colonne "Haut salaire" à 1, si l'expression salaire >= 3000 est vraie, et une colonne "Bas salaire" à 1, si l'expression salaire < 3000 est vraie
8. En utilisant l'opérateur BINARY, affichez le salaire de la personne s'appelant 'DELEGLISE' (en respectant exactement la casse). Faîte de même pour 'Deleglise'
9. À l'aide de l'opérateur CASE, afficher les numéros de service des employés en toutes lettres (ex. trente, ..)
10. À l'aide de l'opérateur LIKE, trouver les employés dont le nom se termine par 'E'
11. À l'aide de l'opérateur REGEXP, trouver les employés dont le nom se termine par 'E'
12. Afficher le salaire annuel des vendeurs (sur 12 mois) incluant leur éventuelle commission. (Attention : la somme du salaire et de la commission peut être indéfinie. Si la commission est indéfinie, il faut lui affecter un 0 afin de ne pas rendre indéfini le résultat. On utilisera la fonction IFNULL pour ce faire)
13. Afficher les noms et salaires des vendeurs en simulant une augmentation de 15 %
14. Calculez le modulo 100 des salaires des vendeurs, sans utiliser l'opérateur % ni MOD (Voir DIV)
15. Affichez les employés par ordre alphabétique
16. Affichez les employés par ordre décroissant de salaire
17. Affichez les deux vendeurs qui ont les plus grosses commissions
18. Affichez le vendeur qui a la deuxième plus grosse commission

Exercice 1.4 - Un avant-goût de la suite

Les requêtes qui suivent seront expliquées en cours dans le module sur les jointures. Cet exercice a pour but de vous introduire à la notion de jointures et de vous les faire expérimenter un peu avant la théorie.

1. Analysez et testez la requête suivante :

```
SELECT
  e.NO_EMPLOYE AS 'Numero employé',
  e.NOM AS Nom,
  e.JOB AS Role,
  e.MGR AS 'Responsable',
  e.DATE_EMBAUCHE AS 'Date d''embauche',
  e.SALAIRE AS 'Salaire',
  IFNULL(e.COMMISSION, 'Aucune') AS Commission,
  s.NOM AS 'Service'
FROM   EMPLOYES AS e INNER JOIN SERVICES AS s
      ON e.NO_SERVICE = s.NO_SERVICE;
```

2. Expliquez, en vos mots, ce que fait cette requête

3. Modifiez la requête afin d'ajouter la colonne Ville qui affiche le LIEU du service pour chaque enregistrement.
4. Observez le résultat de la requête suivante :

```
SELECT
  e.NO_EMPLOYE AS 'Numero employé',
  e.NOM AS Nom,
  e.JOB AS Role,
  r.NOM AS 'Responsable',
  e.DATE_EMBAUCHE AS 'Date d''embauche',
  e.SALAIRE AS 'Salaire',
  IFNULL(e.COMMISSION, 'Aucune') AS Commission,
  s.NOM AS 'Service',
  s.LIEU AS 'Ville'
FROM
  EMPLOYES AS e
    INNER JOIN
  EMPLOYES AS r ON e.MGR = r.NO_EMPLOYE
    INNER JOIN
  SERVICES AS s ON e.NO_SERVICE = s.NO_SERVICE;
```

Manque-t-il un enregistrement ? Si oui, observez les données et essayer de déduire pourquoi vous obtenez ce résultat.

5. Testez cette nouvelle requête et observez s'il y a une différence de résultat par rapport à la requête précédente.

```
SELECT
  e.NO_EMPLOYE AS 'Numero employé',
  e.NOM AS Nom,
  e.JOB AS Role,
  r.NOM AS 'Responsable',
  e.DATE_EMBAUCHE AS 'Date d''embauche',
  e.SALAIRE AS 'Salaire',
  IFNULL(e.COMMISSION, 'Aucune') AS Commission,
  s.NOM AS 'Service',
  s.LIEU AS 'Ville'
FROM
  EMPLOYES AS e
    LEFT JOIN
  EMPLOYES AS r ON e.MGR = r.NO_EMPLOYE
    INNER JOIN
  SERVICES AS s ON e.NO_SERVICE = s.NO_SERVICE;
```

Personnes ayant contribué à la rédaction de ce document :

- Jean-Pierre Duchesneau : version initiale, révisions
- Pierre-François Léon : révisions
- Ali Awdé : révisions