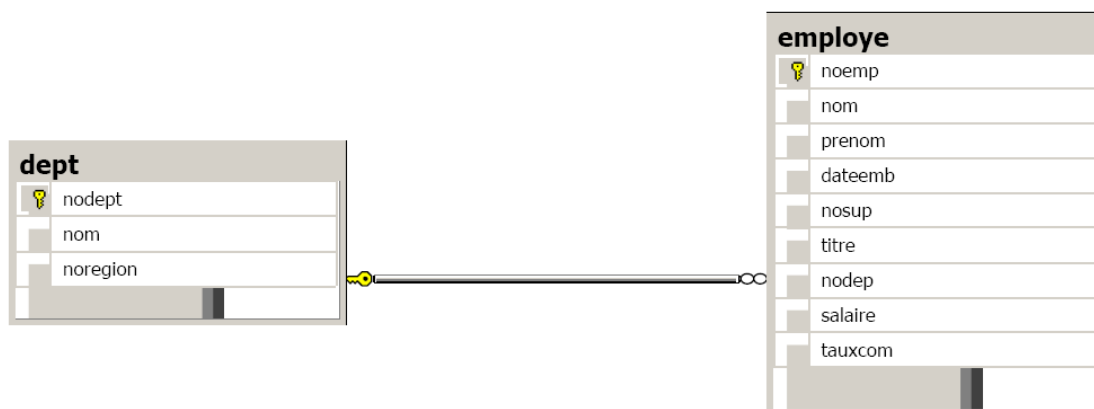


La base "exemple"

Cette base représente d'une manière simpliste (et remarquablement incomplète) le personnel d'une entreprise. Elle n'est pas destinée à être réaliste, mais seulement à être simple à comprendre pour une première approche de SQL.

La table `EMP` contient les informations sur les employés.

La table `DEPT` recense les différents départements de l'entreprise.



	noemp	nom	prenom	dateemb	nosup	titre	nodep	salaire	tauxcom
1	1	patamob	adhémar	2000-03-26 00:00:00.0...	NULL	président	50	50000	NULL
2	2	zeublouse	agathe	2000-04-15 00:00:00.0...	1	dir.distrib	41	35000	NULL
3	3	kuzbidon	alex	2000-05-05 00:00:00.0...	1	dir.vente	31	34000	NULL
4	4	locale	anasthas...	2000-05-25 00:00:00.0...	1	dir.finance	10	36000	NULL
5	5	teutmaronne	armand	2000-06-14 00:00:00.0...	1	dir.administr...	50	36000	NULL
6	6	zoudanlkou	debbie	2000-07-04 00:00:00.0...	2	chef entrepot	41	25000	NULL
7	7	rivenbusse	elsa	2000-07-24 00:00:00.0...	2	chef entrepot	42	24000	NULL
8	8	ardelpic	helmut	2000-08-13 00:00:00.0...	2	chef entrepot	43	23000	NULL
9	9	peursconla	humphrey	2000-09-02 00:00:00.0...	2	chef entrepot	44	22000	NULL
10	10	vrante	helena	2000-09-22 00:00:00.0...	2	chef entrepot	45	21000	NULL
11	11	melusine	enfaillite	2000-10-12 00:00:00.0...	3	representant	31	25000	10
12	12	eurktumeme	odile	2000-11-01 00:00:00.0...	3	representant	32	26000	12
13	13	hotdeugou	olaf	2000-11-21 00:00:00.0...	3	representant	33	27000	10
14	14	odlavieille	pacome	2000-12-11 00:00:00.0...	3	representant	34	25000	15
15	15	amartakaldi...	quentin	2000-12-21 00:00:00.0...	3	representant	35	23000	17
16	16	traibien	samira	2000-12-31 00:00:00.0...	6	secretaire	41	15000	NULL
17	17	fonfec	sophie	2001-01-10 00:00:00.0...	6	secretaire	41	14000	NULL
18	18	fairent	teddy	2001-01-20 00:00:00.0...	7	secretaire	42	13000	NULL
19	19	blaireur	terry	2001-02-09 00:00:00.0...	7	secretaire	42	13000	NULL
20	20	ajerre	tex	2001-02-09 00:00:00.0...	8	secretaire	43	13000	NULL
21	21	chmonfisse	thierry	2001-02-19 00:00:00.0...	8	secretaire	43	12000	NULL
22	22	phototetede...	thomas	2001-02-19 00:00:00.0...	9	secretaire	44	22500	NULL
23	23	kaecoute	xavier	2001-03-01 00:00:00.0...	9	secretaire	34	11500	NULL
24	24	adrouille-to...	yves	2001-03-11 00:00:00.0...	10	secretaire	45	11000	NULL
25	25	anchier	yvon	2001-03-21 00:00:00.0...	10	secretaire	45	10000	NULL

Détails sur les différents champs :

noemp :L'identifiant de la personne. Chaque employé a un **noemp**, tous les **noemp** sont différents. C'est une clef primaire (*primary key*).

nom :Le nom de la personne.

prenom :Le prénom de la personne.

dateemb :Sa date d'embauche.

nosup :Le **noemp** de son supérieur hiérarchique.

*Par exemple Sophie Fonfec (dont le **noemp** est 17) à 6 pour **nosup**, ce qui signifie que le **noemp** de son supérieur hiérarchique est 6, c'est-à-dire que son supérieur hiérarchique est Debbie Zoudanlkou (dont le propre supérieur hiérarchique est Agathe Zeublouze, etc.).*

titre: La fonction de l'employé (directeur, secrétaire, représentant, etc.).

nodep: Le numéro du département (dans l'entreprise) dont dépend l'employé. Ce numéro fait référence à la clef primaire de la table **DEPT**. C'est une clef étrangère (*foreign key*).

salaire: Salaire mensuel de l'employé.

tauxcom: Taux de commission, exprimé en pourcentage.

. : Exercice – la base exemple : .

Développer une application Client-Serveur – Développer des composants d'accès

Formaliser des requêtes SQL



	nodept	nom	noregion
1	10	finance	1
2	20	atelier	2
3	30	atelier	3
4	31	vente	1
5	32	vente	2
6	33	vente	3
7	34	vente	4
8	35	vente	5
9	41	distribution	1
10	42	distribution	2
11	43	distribution	3
12	44	distribution	4
13	45	distribution	5
14	50	administrati...	1

Détails sur les différents champs :

nodept : L'identifiant du département. Chaque département a un `nodept`, tous les `nodept` sont différents. C'est une clef primaire (*primary key*).

nom : Le nom du département. On peut remarquer que des départements différents peuvent avoir le même nom.

noregion : Référence à une région. Ce numéro fait référence à la clef primaire d'une table qui ne sera pas décrite ici^[1]. C'est une clef étrangère (*foreign key*).

SELECT

La commande de recherche est le verbe `SELECT` .

L'étude de la commande `SELECT` et des différentes clauses et fonctions est faite, pour commencer, en considérant une seule table.

Par la suite, l'utilisation de plusieurs tables dans le même `SELECT`, sera prise en compte.

La syntaxe de base est :

```
SELECT colonnes d'une ou plusieurs tables séparées par « , »  
FROM tables séparées par « , »  
WHERE conditions logiques séparées par « AND » ou « OR »  
ORDER BY colonnes séparées par « , »
```

Le caractère « * » permet de demander l'affichage de toutes les colonnes d'une table.

1. Afficher toutes les informations concernant les employés.

```
Select * from employe
```

2. Afficher toutes les informations concernant les départements.

A la place de « * » on peut donner la liste des colonnes souhaitées, dans l'ordre souhaité, en écrivant la requête sur une ou plusieurs lignes. L'écriture sur plusieurs lignes est conseillée en vue de rendre plus simple la relecture et la modification de la requête

3. Afficher le nom, la date d'embauche, le numéro du supérieur, le numéro de département et le salaire de tous les employés.

Le titre des colonnes obtenues par un `SELECT` sont les noms des champs.

On peut cependant changer ces titres en utilisant AS. La commande

```
SELECT nom AS Employe FROM emp ;
```

produira le même effet que

```
SELECT nom FROM emp ;
```

à la différence que la colonne ne sera pas intitulé `nom` mais `Employe`.

Si vous voulez utiliser un titre de colonne qui contient autre chose que des lettres non accentuées (des espaces, des ponctuations, des accents, etc.) il faut l'entourer par des guillemets (« " »). On écrira ainsi :

```
SELECT nom AS "Nom de l'employé" FROM emp ;
```

Suppression des doublons

Il peut être utile de supprimer les doublons, d'où utilisation de la clause `DISTINCT`. Elle a pour effet de n'afficher qu'une seule fois les lignes qui seraient semblables à l'affichage.

On l'utilise sous la forme `SELECT DISTINCT`

4. Afficher le titre de tous les employés.

5. Afficher les différentes valeurs des titres des employés.

Restrictions

Les clauses de restriction s'écrivent derrière `WHERE`. Elles permettent de sélectionner les lignes à afficher

6. Afficher le nom, le numéro d'employé et le numéro du département des employés dont le titre est « Secrétaire ».

7. Afficher le nom et le numéro de département dont le numéro de département est supérieur à 40.

Restriction en comparant les colonnes entre elles

La restriction peut mettre en jeu la comparaison de deux ou plusieurs colonnes entre elles. Il suffit de les appeler par leurs noms.

- 8. Afficher le nom et le prénom des employés dont le nom est alphabétiquement antérieur au prénom.**
- 9. Afficher le nom, le salaire et le numéro du département des employés dont le titre est « Représentant », le numéro de département est 35 et le salaire est supérieur à 20000.**
- 10. Afficher le nom, le titre et le salaire des employés dont le titre est « Représentant » ou dont le titre est « Président ».**
- 11. Afficher le nom, le titre, le numéro de département, le salaire des employés du département 34, dont le titre est « Représentant » ou « Secrétaire ».**
- 12. Afficher le nom, le titre, le numéro de département, le salaire des employés dont le titre est Représentant, ou dont le titre est Secrétaire dans le département numéro 34.**
- 13. Afficher le nom, et le salaire des employés dont le salaire est compris entre 20000 et 30000.**

Négation, recherche approchée

Les opérateurs arithmétiques sont niés avec « ! » : par exemple « non égal » s'écrit « != » (ou « <> »).

Les autres opérateurs sont niés par « NOT » : par exemple « pas dans » s'écrit « NOT IN ».

Les caractères *jokers* sont « % » pour une chaîne et « _ » pour un caractère. On les utilise avec l'opérateur `LIKE`.

15. Afficher le nom des employés commençant par la lettre « H ».

16. Afficher le nom des employés se terminant par la lettre « n ».

17. Afficher le nom des employés contenant la lettre « u » en 3^{ème} position.

18. Afficher le salaire et le nom des employés du service 41 classés par salaire croissant.

19. Afficher le salaire et le nom des employés du service 41 classés par salaire décroissant.

20. Afficher le titre, le salaire et le nom des employés classés par titre croissant et par salaire décroissant.

Valeurs non renseignées

En SQL il existe une valeur correspondant à la valeur *vide*. C'est l'équivalent de la valeur `Null` de certains langages de programmation (comme java ou python). C'est la valeur qu'ont les champs non renseignés. Elle s'appelle également `NULL`. Cette valeur n'est pas prise en compte dans les classements.

21. Afficher le taux de commission, le salaire et le nom des employés classés par taux de commission croissante.

Si on veut utiliser les valeurs non renseignées dans une restriction ce sera avec les clauses `IS NULL` ou `IS NOT NULL`.

22. Afficher le nom, le salaire, le taux de commission et le titre des employés dont le taux de commission n'est pas renseigné.

23. Afficher le nom, le salaire, le taux de commission et le titre des employés dont le taux de commission est renseigné.

24. Afficher le nom, le salaire, le taux de commission, le titre des employés dont le taux de commission est inférieur à 15.

25. Afficher le nom, le salaire, le taux de commission, le titre des employés dont le taux de commission est supérieur à 15.

Expressions arithmétiques

Des expressions arithmétiques peuvent être utilisées après : `SELECT`, `WHERE`, et `ORDER BY`.

26. Afficher le nom, le salaire, le taux de commission et la commission des employés dont le taux de commission n'est pas nul. (la commission est calculée en multipliant le salaire par le taux de commission)

27. Afficher le nom, le salaire, le taux de commission, la commission des employés dont le taux de commission n'est pas nul, classé par taux de commission croissant.

Concaténation

La fonction qui permet la concaténation de chaînes de caractères est [`CONCAT`](#)

28. Afficher le nom et le prénom (concaténés) des employés. Renommer les colonnes

Chaînes de caractères

La fonction extraction de chaîne de caractères `SUBSTRING(expr, p, n)` permet d'extraire de `expr` à partir de la position `p`, les `n` caractères.

La fonction de retour du rang d'une chaîne dans une autre chaîne, `LOCATE (arg1, arg2)` permet de retourner le rang de la chaîne `arg1` dans la chaîne `arg2`.

Les fonctions `UPPER(arg1)` et `LOWER(arg1)` permettent respectivement de forcer à la majuscule ou à la minuscule.

La fonction `LENGTH(arg1)` permet d'obtenir le nombre de caractères d'une chaîne de caractères.

29. Afficher les 5 premières lettres du nom des employés.

30. Afficher le nom et le rang de la lettre « r » dans le nom des employés.

31. Afficher le nom, le nom en majuscule et le nom en minuscule de l'employé dont le nom est *Vrante*.

32. Afficher le nom et le nombre de caractères du nom des employés.