

420-KB6-LG

Introduction aux bases de données

Gestion de l'information hiérarchisée

Ordre du jour

Requête hiérarchique

Syntaxe : START WITH et CONNECT BY

Pseudocolonne LEVEL


Conclusion

Exercices

Données hiérarchiques

Des données hiérarchiques sont des données stockées dans une table avec une **relation récursive** (relation sur la même table ou entité) dans une **cardinalité maximale est 1** (un subordonné a un seul supérieur, mais un supérieur peut avoir plusieurs subordonnés) .


```
SELECT * FROM syemp;
```



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	80-12-17	800	(null)	20
7499	ALLEN	SALESMAN	7698	81-02-20	1600	300	30
7521	WARD	SALESMAN	7698	81-02-22	1250	500	30
7566	JONES	MANAGER	7839	81-04-02	2975	(null)	20
7654	MARTIN	SALESMAN	7698	81-09-28	1250	1400	30
7698	BLAKE	MANAGER	7839	81-05-01	2850	(null)	30
7782	CLARK	MANAGER	7839	81-06-09	2450	(null)	10
7788	SCOTT	ANALYST	7566	87-04-19	3000	(null)	20
7839	KING	PRESIDENT	(null)	81-11-17	5000	(null)	10
7844	TURNER	SALESMAN	7698	81-09-08	1500	0	30
7876	ADAMS	CLERK	7788	87-05-23	1100	(null)	20
7900	JAMES	CLERK	7698	81-12-03	950	(null)	30
7902	FORD	ANALYST	7566	81-12-03	3000	(null)	20
7934	MILLER	CLERK	7782	82-01-23	1300	(null)	10

Données hiérarchiques

Pour implémenter une relation récursive on a besoin de deux colonnes :
une colonne de **clé primaire (clé parent)** et une colonne de **clé étrangère (clé enfant)** qui réfère à la clé primaire.



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	80-12-17	800	(null)	20
7499	ALLEN	SALESMAN	7698	81-02-20	1600	300	30
7521	WARD	SALESMAN	7698	81-02-22	1250	500	30
7566	JONES	MANAGER	7839	81-04-02	2975	(null)	20
7654	MARTIN	SALESMAN	7698	81-09-28	1250	1400	30
7698	BLAKE	MANAGER	7839	81-05-01	2850	(null)	30
7782	CLARK	MANAGER	7839	81-06-09	2450	(null)	10
7788	SCOTT	ANALYST	7566	87-04-19	3000	(null)	20
7839	KING	PRESIDENT	(null)	81-11-17	5000	(null)	10
7844	TURNER	SALESMAN	7698	81-09-08	1500	0	30
7876	ADAMS	CLERK	7788	87-05-23	1100	(null)	20
7900	JAMES	CLERK	7698	81-12-03	950	(null)	30
7902	FORD	ANALYST	7566	81-12-03	3000	(null)	20
7934	MILLER	CLERK	7782	82-01-23	1300	(null)	10

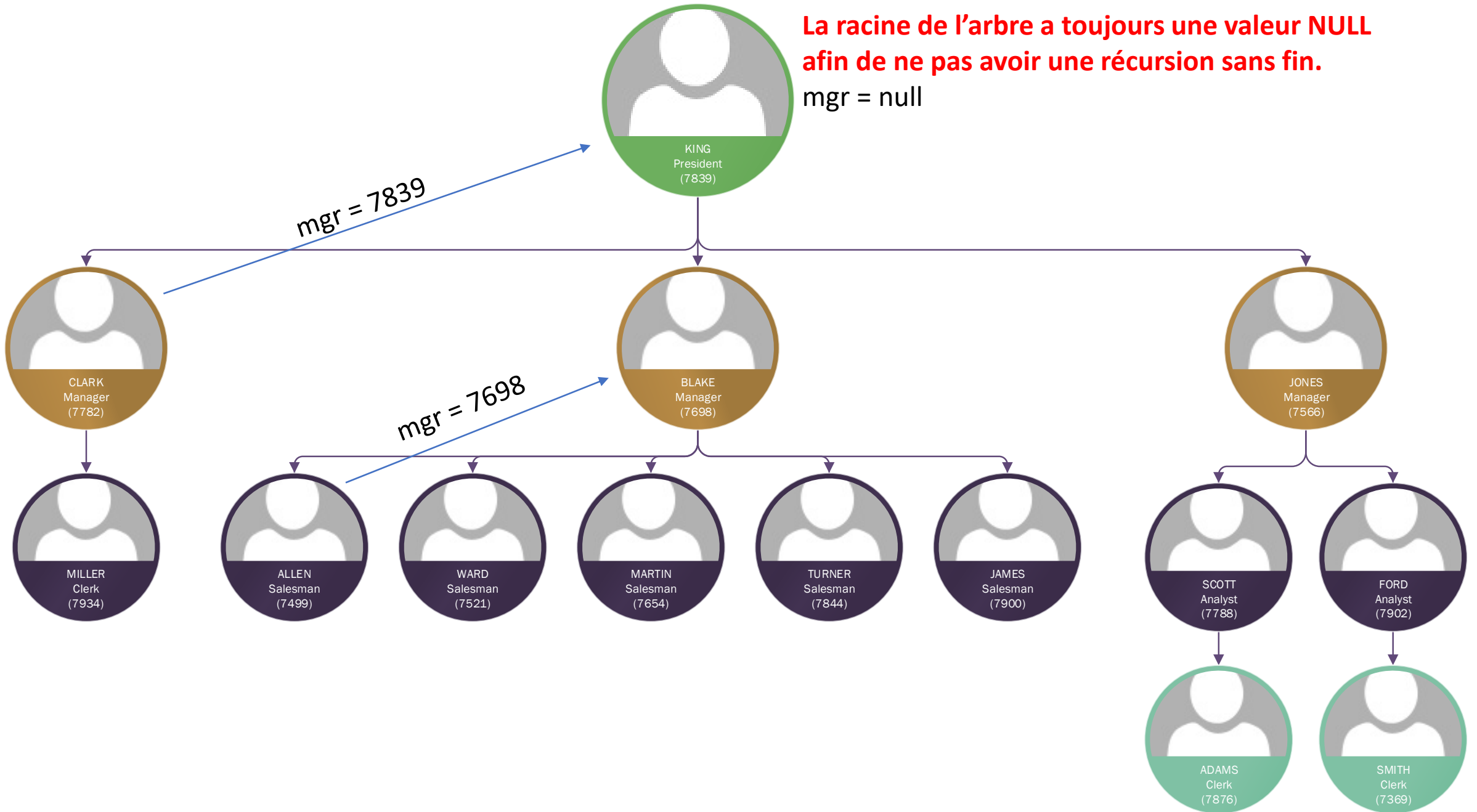
Données hiérarchiques

En utilisant une arborescence, on souhaite mettre en évidence la hiérarchisation entre les employés (employés avec les responsables hiérarchiques).

La page suivante représente l'organigramme des employés enregistrés dans la table précédente.

La racine de l'arbre a toujours une valeur NULL afin de ne pas avoir une récursion sans fin.

mgr = null



Qu'est-ce qu'une requête hiérarchique ?

Il est parfois nécessaire d'interroger les données suivant un ordre hiérarchique bien précis. Pour cela, Oracle met à notre disposition des commandes, **CONNECT BY** et **START WITH**, pour extraire des données en utilisant un ordre.

L'exécution de la requête se fait comme suit:

1. Oracle sélectionne la racine de l'arbre, ce sont les lignes qui satisfont la condition indiquée dans la clause **START WITH**.
2. Oracle sélectionne les enfants de chaque parent. Tous doivent satisfaire la condition indiquée dans la clause **CONNECT BY**.
3. Tous les enfants sont sélectionnés les uns après les autres.
4. Si la requête contient une clause **WHERE**, Oracle supprime de la hiérarchie toutes les lignes qui ne respectent pas la condition indiquée par cette clause.

Syntaxe de la requête hiérarchique

```
SELECT <liste_colonnes>, [ LEVEL ]  
FROM <nom_table>  
[ START WITH <condition(s)> ]  
CONNECT BY PRIOR <condition(s)>;
```

Entre [] signifie optionnel.

START WITH : indique pour quelle occurrence on doit sélectionner les subordonnées ou les supérieurs.

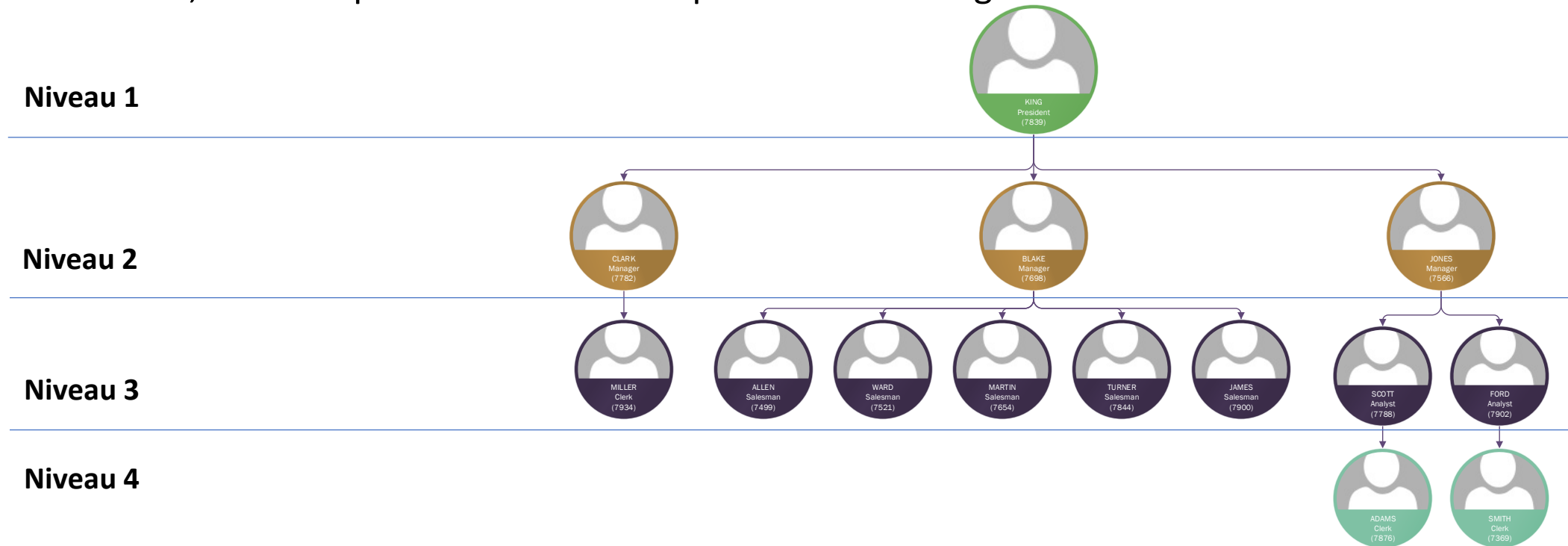
CONNECT BY : connecte la colonne clé primaire avec la colonne clé enfant (clause obligatoire).

PRIOR : indique le sens du parcours de la hiérarchie (ou de l'arbre).

LEVEL : nom de la pseudocolonne indiquant le niveau de hiérarchie entre les enregistrements.

Pseudocolonne LEVEL

LEVEL : nom de la colonne indiquant le niveau de hiérarchie entre les enregistrements.
C'est-à-dire, savoir à quel niveau hiérarchique se situe l'enregistrement.



Utilisation de la clause START WITH

Exemple:

Afficher tous les employés sous l'employé dont le empno est 7698.

```
SELECT empno, ename, mgr, LEVEL  
FROM syemp  
START WITH empno = 7698  
CONNECT BY PRIOR empno = mgr;
```

Ici, on se positionne sur le noeud 7698 dans l'arbre et donc le niveau est 1 (c'est la racine de la requête). Un seul niveau hiérarchique existe sous l'employé 7698.

	EMPNO	ENAME	MGR	LEVEL
1	7698	BLAKE	7839	1
2	7499	ALLEN	7698	2
3	7521	WARD	7698	2
4	7654	MARTIN	7698	2
5	7844	TURNER	7698	2
6	7900	JAMES	7698	2

Utilisation de la clause START WITH

Exemple:

Afficher tous les employés sous l'employé dont le ename est SCOTT.

```
SELECT empno, ename, mgr, LEVEL  
FROM syemp  
START WITH ename = 'SCOTT'  
CONNECT BY PRIOR empno = mgr;
```

Ici, on se positionne sur le noeud 'SCOTT' dans l'arbre et donc le niveau est 1 (c'est la racine de la requête). Un seul employé a comme supérieur immédiat SCOTT.

EMPNO	ENAME	MGR	LEVEL
7788	SCOTT	7566	1
7876	ADAMS	7788	2

Parcours de la hiérarchie

PRIOR : indique le sens du parcours de la hiérarchie (ou de l'arbre).

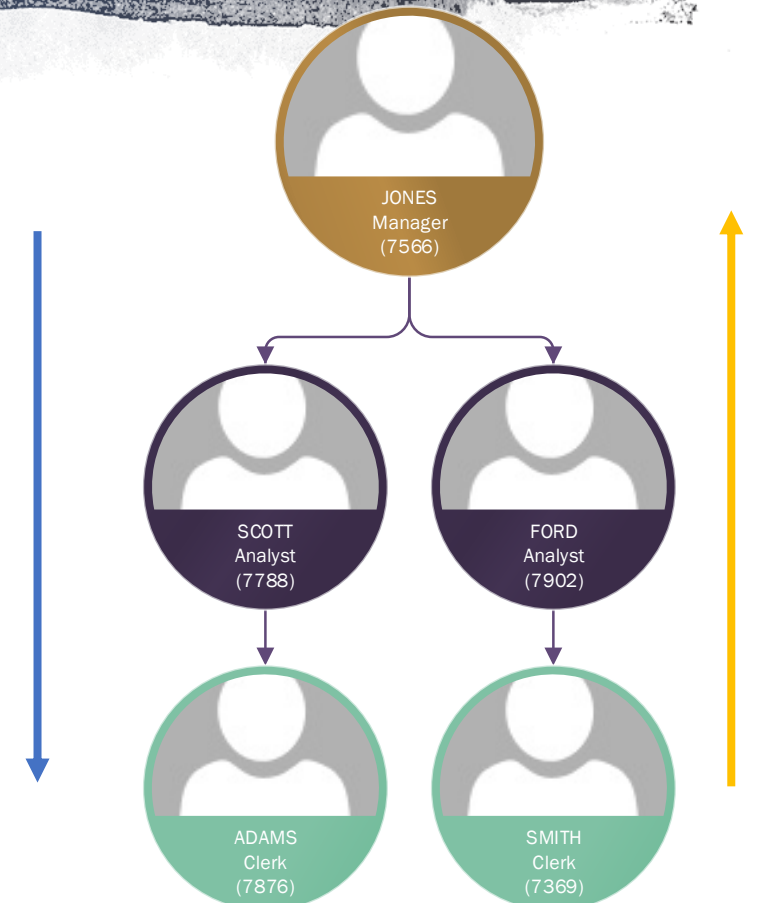
Si l'on place **PRIOR** à **gauche**, on parcourt l'arbre vers le bas (on extrait les subordonnés)

CONNECT BY PRIOR empno = mgr

ou

Si l'on place **PRIOR** à **droite**, on parcourt l'arbre vers le haut (on extrait les supérieurs)

CONNECT BY empno = PRIOR mgr



Parcours de la hiérarchie

Parcours vers le bas:

Les **subordonnés** de FORD

```
SELECT empno, ename, mgr, LEVEL
FROM syemp
START WITH ename = 'FORD'
CONNECT BY PRIOR empno = mgr;
```

EMPNO	ENAME	MGR	LEVEL
7902	FORD	7566	1
7369	SMITH	7902	2

Parcours vers le haut

Les **supérieurs** de FORD

```
SELECT empno, ename, mgr, LEVEL
FROM syemp
START WITH ename = 'FORD'
CONNECT BY empno = PRIOR mgr;
```

EMPNO	ENAME	MGR	LEVEL
7902	FORD	7566	1
7566	JONES	7839	2
7839	KING	(null)	3

Conclusion

- Les requêtes hiérarchiques permettent d'afficher une relation hiérarchique existante entre des lignes d'une table.
- Les données hiérarchique peuvent être vues comme un arbre à l'envers (page 9).
- L'arbre a toujours une racine de valeur NULL au sommet (voir page 6).
- CONNECT BY est une clause obligatoire qui se compare à une jointure puisqu'elle lie la clé primaire et la clé étrangère.
- Il est possible de déterminer la direction (PRIOR) et le point de départ du parcours (START WITH).

Exercices

Disponibles sur le site du cours.

Questions

