

Morgado-Samagaio Jonathan

**IBD / TP15 Oracle**

## Table des matières

Table des matières .....	2
1. Introduction.....	3
2. Requêtes.....	3
3. Conclusion .....	9

## 1. Introduction

Dans ce TP, nous allons revoir tous les types de requêtes que nous avons vu depuis le début.

## 2. Requêtes

- 1) Nous commençons par de la projection/sélection en cherchant les montagnes de plus de 6000 mètres d'altitude de la chaîne : Cordillère des Andes. On utilise la commande suivante :

```
SELECT nm, nom, altitude
FROM Montagne
WHERE altitude >= 6000
AND chaine = 'Cordillère des
```

C'est une commande basique avec l'utilisation de l'opérateur = et >= avec un ORDER BY pour trier. On obtient :

NM	NOM
52	Aconcagua
53	Nevado Ojos del Salado
54	Monte Pissis
55	Huascarán Sur
56	Cerro Bonete Chico
57	Tres Cruces Sur
58	Llullaillaco
59	Mercedario
60	Huascarán Norte
61	Walter Penck I
62	Yerupajá
63	Tres Cruces Central
64	Nevado de Incahuasi

- 2) On passe maintenant aux jointures relationnelles. On cherche les Montagnes de plus 8000 mètres d'altitude soit en Inde soit au Népal. On utilise la commande suivante :

```
SELECT DISTINCT M.nm, M.nom, M.
FROM Montagne M, Localiser L
WHERE M.altitude > 8000
AND M.nm = L.nm
AND (L.nomp = 'Inde'
OR L.nomp = 'Népal')
```

On peut voir la jointure des deux tables suivant une colonne commune. On voit aussi l'utilisation du DISTINCT pour ne pas avoir de doublons. On obtient :

NM	NOM
99	Everest
101	Kangchenjunga
102	Lhotse
103	Makalu
104	Cho Oyu
105	Dhaulagiri I
106	Manaslu
108	Annapurna I

- 3) On continue sur les jointures relationnelles en cherchant les pays avec des montagnes de plus de 5000 mètres d'altitude et des fleuves de plus de 2000 mètres de longueurs. On utilise la commande suivante :

```
COLUMN nom FORMAT A32;
COLUMN capitale FORMAT A15;
COLUMN nomc FORMAT A10;
SELECT DISTINCT P.nom, P.capitale, P.superficie,
FROM Pays P, Montagne M, Localiser L, Fleuve F,
WHERE M.altitude > 5000 AND F.longueur > 2000
AND M.nm = L.nm AND T.nf = F.nf
```

Nous avons utilisé les COLUMN FORMAT pour pouvoir formater l'affichage du résultat. Cette requête était une version plus difficile de la dernière donc elles sont similaire. On obtient :

NOM	CAPITALE	SUPERFIC
Tanzanie	Dodoma	9450
Mexique	Mexico	19643
États-Unis	Washington D.C.	98335
Chine	Pékin	95969
Pakistan	Islamabad	8819
Ouganda	Kampala	2415
Canada	Ottawa	99846
Pérou	Lima	12852
Russie	Moscou	171251
Kenya	Nairobi	5803
République démocratique du Congo	Kinshasa	23454
Argentine	Buenos Aires	27918

- 4) Nous passons maintenant à l'auto-jointure. On cherche le numéro des montagnes localisées en Argentine et au Chili. On utilise la commande suivante :

```
SELECT L1.nm
FROM Localiser L1, Localiser L2
WHERE L1.nomp = 'Argentine' AND L2.nomp
AND L1.nm = L2.nm;
```

Nous pouvons voir l'utilisation de deux instance d'une même table. On obtient :

N°
5.
5'
5)
6.

- 5) Nous passons maintenant à la théta-jointure. On cherche les fleuves plus longs que Le Mékong. On utilise la commande :

```
SELECT F1.*
FROM Fleuve F1, Fleuve F2
WHERE F2.nom = 'Le Mékong'
AND F1.longueur > F2.longueur
```

Nous pouvons voir l'utilisation du > qui fait de cette requête une théta-jointure. On obtient :

N°	NOM	LONGUEUR	EMBOUCHURE
21	Le Nil	6700	mer Méditerranée
38	L'Amazone	6437	océan Atlantique
87	Le fleuve Jaune	5646	mer de Chine

- 6) Nous passons aux jointures externes. On cherche les pays du continent Américain qui n'ont pas de fleuves. On utilise la commande :

```
SELECT P.nom, P.capitale, P.superficie
FROM Pays P, Traverser T
WHERE P.nomc = 'Amérique'
AND P.nom = T.nomp(+)

```

On peut voir le (+) qui fait de cette requête une jointure externe et le IS NULL qui permet de récupérer les colonnes vides. On obtient :

NOM	CAPITALE
Costa Rica	San José
Belize	Belmopan
Trinité-et-Tobago	Port-d'Espagne
Bolivie	La Paz
Saint-Vincent-et-les-Grenadines	Kingstown
Panama	Panama
République dominicaine	Saint-Domingue
Antigua-et-Barbuda	Saint John's
Chili	Santiago
Sainte-Lucie	Castries
Haïti	Port-au-Prince
Jamaïque	Kingston
Barbade	Bridgetown
Bahamas	Nassau
Grenade	Saint-Georges
Saint-Christophe-et-Niévès	Basseterre
Cuba	La Havane

- 7) On passe aux agrégations. On cherche le nombre de fleuve et la longueur du fleuve le plus long par continent. On utilise la commande :

```
SELECT P.nomc, COUNT(F.nf) AS NBF, MAX(F.longue
FROM Pays P, Traverser T, Fleuve F
WHERE P.nom = T.nomp AND T.nf = F.nf
```

On obtient :

NOMC	NBF
Océanie	12
Asie	78
Amérique	53
Afrique	83

- 8) On cherche à présent les pays ayant plus de 10 montagnes. On utilise la commande :

```
SELECT nomp, COUNT(L.nm)
FROM Localiser L
GROUP BY nomp
```

On voit l'utilisation du HAVING pour rajouter notre condition. On obtient :



NOMP
Russie
Papouasie-Nouvelle-Guinée
États-Unis
Pakistan
Indonésie
Canada
Népal
Suisse
Chine

- 9) On cherche maintenant le pays ayant le plus de montagnes. On veut faire deux versions de cette requête.
- 10) Nous passons aux requêtes ensemblistes. On cherche à faire la requête de la question 3 mais d'une manière ensembliste. La commande est :

```
SELECT P.nom, P.capitale, P.superficie, P.nomc
FROM Pays P, Localiser L, Montagne M
WHERE L.nomp = P.nom AND L.nm = M.nm AND M.alti
INTERSECT
SELECT P.nom, P.capitale, P.superficie, P.nomc
FROM Pays P, Traverser T, Fleuve F
```

On voit INTERSECT qui fait de cette requête une requête ensembliste. On obtient :

NOM	CAPITALE	SUPERFIC
Argentine	Buenos Aires	27918
Canada	Ottawa	99846
Chine	Pékin	95969
Inde	New Delhi	32872
Kenya	Nairobi	5803
Mexique	Mexico	19643
Ouganda	Kampala	2415
Pakistan	Islamabad	8819
Pérou	Lima	12852
Russie	Moscou	171251
République démocratique du Congo	Kinshasa	23454
Tanzanie	Dodoma	9450

- 11) On passe aux jointures procédurales. On cherche les pays avec des montagnes de plus de 6000 mètres d'altitude. On utilise la commande :

```

SELECT nom, capitale, superficie, nomc
FROM Pays
WHERE nom IN (SELECT nomp
               FROM Localiser
               WHERE nm IN (SELECT nm
                           FROM Montagne

```

On peut voir plusieurs requêtes imbriquées avec des IN. On obtient :

NOM	CAPITALE	SUPERFIC
-----	-----	-----
Argentine	Buenos Aires	27918
Pakistan	Islamabad	8819
Népal	Katmandou	1475
Pérou	Lima	12852
Inde	New Delhi	32872
Bolivie	La Paz	10985
Chine	Pékin	95969
Chili	Santiago	7552
Bhoutan	Thimphou	383
-----	-----	-----

- 12) On cherche maintenant le fleuve le plus long. On veut deux versions de cette requête. On utilise les commandes :

```

SELECT *
FROM Fleuve
WHERE longueur = (SELECT MAX(longueur)
                  FROM Fleuve);

SELECT *
FROM Fleuve
WHERE longueur >= ALL (SELECT NVL (longueur, 0)

```

Nous pouvons voir le NVL dans la deuxième requête qui nous permet d'enlever les valeurs null. On obtient :

NF NOM	LONGUEUR	EMBOUCHURE
-----	-----	-----
21 Le Nil	6700	mer Méditerranée



### 3. Conclusion

Dans ce TP, nous avons revu tous ce que nous avons vu pendant le semestre sur les requête : Les projections/sélection, les jointures relationnelles, les auto et théta-jointures, les jointures externes, les agrégations, les requêtes ensembliste et les jointures procédurales.