Entrepôt de données

Universite Lille 1 | Master 1 MIAGE FA

Rapport TP1

LANDSCHOOT ludovic

2016

Table des matières

[Schéma des tables 2](#_Toc467017076)

[Table Temps 2](#_Toc467017077)

[Table Client 2](#_Toc467017078)

[Table Produit 2](#_Toc467017079)

[Table Ventes 2](#_Toc467017080)

[Exercice 1 : Requête de classement sous Oracle 2](#_Toc467017081)

[Question a 2](#_Toc467017082)

[Question b 2](#_Toc467017083)

[Question c 3](#_Toc467017084)

[Question d 3](#_Toc467017085)

[Question e 3](#_Toc467017086)

[Question f 4](#_Toc467017087)

[Question g 4](#_Toc467017088)

[Exercice 2 : OLAP/ROLAP sous Oracle 5](#_Toc467017089)

[Question a 5](#_Toc467017090)

[Question b 5](#_Toc467017091)

[Question c 5](#_Toc467017092)

[Question d 6](#_Toc467017093)

[Question e 6](#_Toc467017094)

[Question f 7](#_Toc467017095)

[Question g 7](#_Toc467017096)

[Question h 7](#_Toc467017097)

# Schéma des tables

## Table Temps

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TID** | Année | Trimestre | Mois | Jour |
| Numéric (4) | Numéric (4) | Numéric (2) | Numéric (2) | Numéric (2) |

## Table Client

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CL\_ID** | CL\_NAME | CL\_CITY | CL\_R | CL\_STATE |
| Numéric (2) | Varchar (50) | Varchar (30) | Varchar (30) | Varchar (30) |

## Table Produit

|  |  |  |  |
| --- | --- | --- | --- |
| **PID** | PNAME | CATEGORY | SUBCAT |
| Numéric (2) | Varchar (50) | Varchar (40) | Varchar (30) |

## Table Ventes

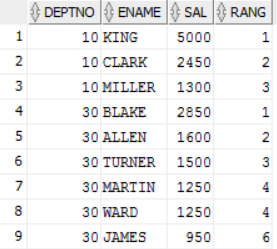
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TID** | **PID** | **CID** | QTE | PU |
| Numéric (4) | Numéric (2) | Numéric (2) | Numéric (3) | Numéric (6,2) |

# Exercice 1 : Requête de classement sous Oracle

## Question a

**Question :** *Quel est le classement des salaires des employés par département pour les départements 10 et 30 ?*

**Résultat :**

**Requête :**

**SELECT** DEPTNO**,** ENAME**,** SAL**,** **RANK()** OVER **(PARTITION** **BY** DEPTNO **ORDER** **BY** SAL **DESC)** **AS** RANG

**FROM** EMP

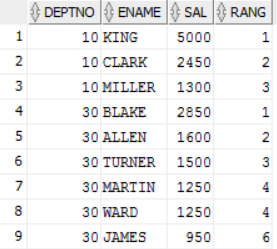
**WHERE** DEPTNO **=** 10 **OR** DEPTNO **=** 30**;**

**Commentaire :**

## Question b

**Question :** *Idem en ôtant les trous dans le classement ?*

**Résultat :**

**Requête :**

**SELECT** DEPTNO**,** ENAME**,** SAL**,** **DENSE\_RANK()** OVER **(PARTITION** **BY** DEPTNO **ORDER** **BY** SAL **DESC)** **AS** RANG

**FROM** EMP

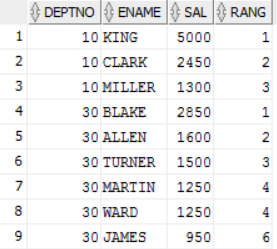
**WHERE** DEPTNO **=** 10 **OR** DEPTNO **=** 30**;**

**Commentaire :**

## Question c

**Question :** *Quel est le classement décroissant des salaires différents par département pour les départements 10 et 20 ?*

**Résultat :**

**Requête :**

**SELECT** **DISTINCT** DEPTNO**,** SAL**,** **DENSE\_RANK()** OVER **(PARTITION** **BY** DEPTNO **ORDER** **BY** SAL**)** **AS** RANG

**FROM** EMP

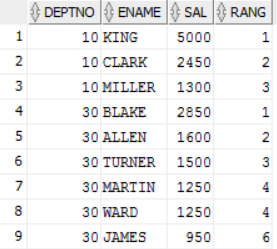
**WHERE** DEPTNO **=** 10 **OR** DEPTNO **=** 20

**ORDER** **BY** SAL **DESC;**

**Commentaire :**

## Question d

**Question :** *Quel est le salaire total versé par profession (répondre avec deux méthodes possibles avec et sans « group by ») ?*

**Résultat :**

**Requête :**

**SELECT** JOB**,** **SUM(**SAL**)** **AS** TOT\_SAL\_JOB

**FROM** EMP

**GROUP** **BY** JOB**;**

**SELECT** **DISTINCT** JOB**,** **SUM(**SAL**)** OVER **(PARTITION** **BY** JOB**)** **AS** TOT\_SAL\_JOB

**FROM** EMP**;**

**SELECT** **DISTINCT** JOB**, (SELECT** **SUM(**SAL**)** **FROM** EMP E2 **WHERE** E1**.**JOB**=**E2**.**JOB**)**

**AS** TOT\_SAL\_JOB

**FROM** EMP E1**;**

**Commentaire :**

## Question e

**Question :** *Quelle est la différence entre un « group by » et un « partition by » dans une requête SQL ?*

**Réponse :**

Group by :

* Réduis le nombre d’enregistrement en renvoyant qu’une seule ligne par groupe.
* Dans le select il est possible d’utiliser seulement les éléments qui compose le « group by ».
* En filtre, pour les conditions sur le « group by » il faut utiliser une autre clause que « where ».

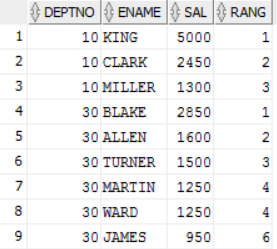
Partition by :

* Découpe en partition mais ne diminue pas le nombre d’enregistrement, il les retourne tous avec une colonne supplémentaire.
* Dans le select il est possible d’afficher n’importe quelles données.
* Nous pouvons utiliser le where pour filtrer n’importe quelle condition.

## Question f

**Question :** *Quel est le montant total des salaires versés, tous départements et jobs confondus, par département et par département et job ?*

**Résultat :**

**Requête :**

**SELECT** DEPTNO**,** JOB**,** **SUM(**SAL**)**

**FROM** EMP

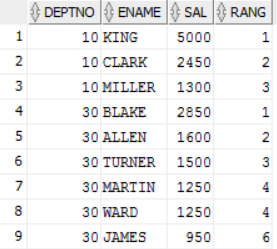
**GROUP** **BY** **ROLLUP** **(**DEPTNO**,** JOB**);**

**Commentaire :**

## Question g

**Question :** *Idem en ôtant toute confusion avec les valeurs nulles (répondre avec deux méthodes possibles).*

**Résultat :**

**Requête :**

**SELECT** **NVL(to\_char(**DEPTNO**),**'TousDep'**),** **NVL(to\_char(**JOB**),**'TousEmployes'**),** **SUM(**SAL**)**

**FROM** EMP

**GROUP** **BY** **ROLLUP** **(**DEPTNO**,** JOB**)**

**ORDER** **BY** DEPTNO**,** JOB **DESC;**

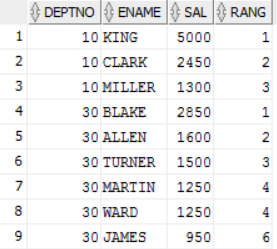
**Commentaire :**

# Exercice 2 : OLAP/ROLAP sous Oracle

## Question a

**Question :** *Donnez la moyenne des ventes par année, région, catégorie ; par année, région et par année, pour les années 2009 et 2010.*

**Résultat :**

**Requête :**

**SELECT** t**.**annee**,** c**.**CL\_R**,** p**.**CATEGORY**,** **AVG(**v**.**QTE**\***v**.**PU**)** **AS** CA\_MOYEN

**FROM** VENTES v

**JOIN** TEMPS t **ON** t**.**TID**=**v**.**TID

**JOIN** CLIENTS c **ON** v**.**CID**=**c**.**CL\_ID

**JOIN** PRODUITS p **ON** v**.**PID**=**p**.**PID

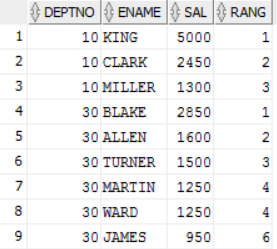
**WHERE** t**.**annee**=**2009 **OR** t**.**annee**=**2010

**GROUP** **BY** **ROLLUP** **(**annee**,** CL\_R**,** CATEGORY**);**

**Commentaire :**

## Question b

**Question :** *Donnez la moyenne des ventes par année, selon les dimensions région et catégorie pour les années 2009 et 2010.*

**Résultat :**

**Requête :**

**SELECT** t**.**annee**,** c**.**CL\_R**,** p**.**CATEGORY**,** **AVG(**v**.**QTE**\***v**.**PU**)** **AS** CA\_MOYEN

**FROM** VENTES v

**JOIN** TEMPS t **ON** t**.**TID**=**v**.**TID

**JOIN** CLIENTS c **ON** v**.**CID**=**c**.**CL\_ID

**JOIN** PRODUITS p **ON** v**.**PID**=**p**.**PID

**WHERE** t**.**annee**=**2009 **OR** t**.**annee**=**2010

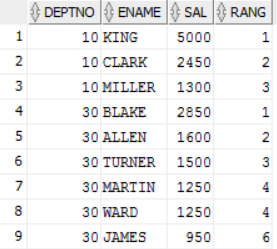
**GROUP** **BY** **CUBE** **(**annee**,** CL\_R**,** CATEGORY**);**

**Commentaire :**

## Question c

**Question***Quel est le produit qui fait le plus grand CA par année et par catégorie (cf. RANK) ?*

**Résultat :**

**Requête :**

**SELECT** annee**,** CATEGORY**,** PNAME

**FROM** **(**

**SELECT** t**.**annee**,** p**.**CATEGORY**,** p**.**PNAME**,** **RANK()** OVER

**(PARTITION** **BY** t**.**annee**,** p**.**CATEGORY **ORDER** **BY** **SUM(**v**.**QTE**\***v**.**PU**)** **DESC)** **AS** rang

**FROM** VENTES v

**NATURAL** **JOIN** temps t

**NATURAL** **JOIN** produits p

**GROUP** **by** **(**annee**,** CATEGORY**,** p**.**PNAME**)**

**)**

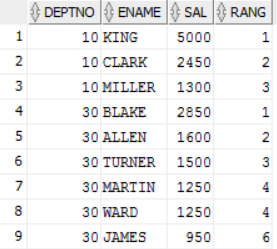
**WHERE** RANG**=**1**;**

**Commentaire :**

## Question d

**Question :** *Pour chaque année, donner le total du CA ainsi que le total du CA par catégorie. On ne veut pas le résultat pour le total des années (utiliser la fonction GROUPING\_ID).*

**Résultat :**

**Requête :**

**SELECT** annee**,** category**,** **SUM(**qte**\***pu**)**

**FROM** ventes

**NATURAL** **JOIN** temps

**NATURAL** **JOIN** produits

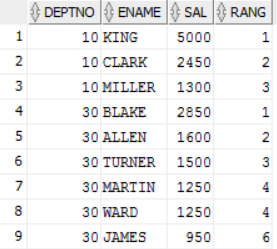
**GROUP** **BY** **CUBE(**annee**,** category**)**

**HAVING** GROUPING\_ID**(**annee**)** **<** 1**;**

**Commentaire :**

## Question e

**Question :** *Quel est le meilleur mois de vente du produit "Sirop d’érable" pour chacune des années ?*

**Résultat :**

**Requête :**

**SELECT** annee**,** mois**,** CA\_TOTAL

**FROM** **(**

**SELECT** annee**,** mois**,** **SUM(**qte**\***pu**)** **AS** CA\_TOTAL**,** **RANK()** OVER

**(PARTITION** **BY** annee **ORDER** **BY** **SUM(**qte**\***pu**)** **DESC)** **AS** rang

**FROM** ventes

**NATURAL** **JOIN** temps

**NATURAL** **JOIN** produits

**WHERE** pname **=** 'Sirop d érable'

**GROUP** **BY** annee**,** mois

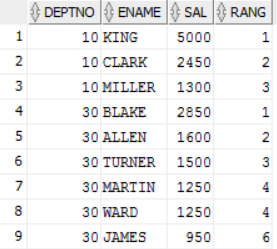
**)** tmp

**WHERE** tmp**.**rang **=** 1**;**

**Commentaire :**

## Question f

**Question :** *Donnez tous les totaux de ventes par année selon la dimension catégorie, ainsi que selon la dimension cl\_name (nom du client). Utilisez la commande GROUPING\_SETS.*

**Résultat :**

**Requête :**

**SELECT** annee**,** cl\_name**,** category**,** **SUM(**qte**\***pu**)**

**FROM** ventes

**NATURAL** **JOIN** temps

**NATURAL** **JOIN** produits

**NATURAL** **JOIN** clients

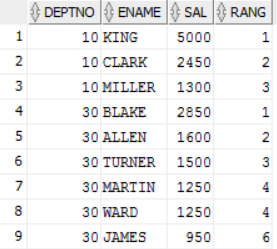
**GROUP** **BY** annee**,**

**GROUPING** **SETS(**category**,** cl\_name**);**

**Commentaire :**

## Question g

**Question :** *Quelle est la répartition par tiers des catégories selon leurs quantités totales vendues en 2010 ?*

**Résultat :**

**Requête :**

**SELECT** category**,** **SUM(**qte**)** **AS** QTE\_VENDU\_2010**,** NTILE**(**3**)**

OVER**(ORDER** **BY** **SUM(**qte**)** **DESC)** **AS** TIERS

**FROM** ventes

**NATURAL** **JOIN** temps

**NATURAL** **JOIN** produits

**WHERE** annee **=** 2010

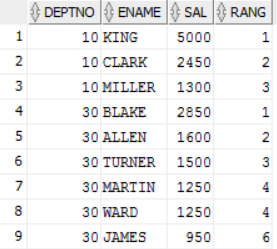
**GROUP** **BY** category**;**

**Commentaire :**

## Question h

**Question :** *Quelle est la quantité de produits vendus pour chaque catégorie, les 5 premiers jours de chaque mois de 2010 ? On peut imaginer plusieurs solutions, selon que l’on interprète « les 5 premiers jours de chaque mois » par « un jour compris entre 1 et 5 » ou « les 5 premiers jours de vente ».*

**Résultat :**

**Requête :**

**SELECT** category**,** mois**,** **SUM(**qte**)** **AS** qte\_vendue

**FROM** ventes

**NATURAL** **JOIN** temps

**NATURAL** **JOIN** produits

**WHERE** annee **=**2010 **AND** jour **<=**5

**GROUP** **BY** CATEGORY**,** annee**,** mois

**ORDER** **BY** CATEGORY**,** mois**;**

**Commentaire :**