

TP01 SQL – LMD (semaine 37-38 - 4h) :

Consultation d'une BD Oracle

Préalable

Si ce n'est pas déjà fait, créer **votre schéma** (c'est-à-dire de votre compte **ORACLE sur BD19**) sur <http://bd19> (pour vous connecter à Oracle Users Manager utiliser votre login/mdp du bureau virtuel).

Tester ensuite votre connexion sur **Oracle SQL Developer**

Lors du premier lancement de **SQL Developer**, il est nécessaire de créer une nouvelle connexion à bd19 (que vous utiliserez directement les fois suivantes).


Rappel : Les 4 boutons qui permettent respectivement

- d'exécuter l'instruction courante,
- d'exécuter le script complet,
- de valider (commit)
- d'annuler (rollback).



Exercice : Consultation de la base HOTEL

Remarque : Pour consulter les tables du schéma HOTEL vous devrez préfixer les tables par le nom du schéma (par ex. HOTEL.CLIENT)



Syntaxe générale d'une requête SQL :

```
SELECT [DISTINCT] ...  
FROM ...  
[WHERE ...]  
[ORDER BY ...] ;
```

Opérateurs logiques : OR, AND, NOT

Opérateurs:

```
[NOT] LIKE,  
BETWEEN .. AND ..,  
IN (... , ... , ...) ,  
IS [NOT] NULL  
=, !=, <>, <=, >=, <, >
```

Concaténation : ||

Requête 1 : Liste des chambres qui ne sont pas au rez-de-chaussée (étage rdc), qui communique avec une autre chambre (chb_communique renseigné) et qui sont entretenues par les agents (agt_nom) nommés Fechol et Boussel.

Tri par agent puis par étage. (5 lignes attendues, 3 chambres pour FECHOL, 2 Pour BOUSSEL)

Agent	Chambre	étage
-------	---------	-------

Où Agent : Nom-prénom de l'agent (Concaténation du nom en majuscules et du prénom)
Chambre : « Chambre » + Numéro de la chambre (chb_numero)

Requête 2 : Liste des clients (tit_lib, cli_nom et cli_prenom) de « REIMS » (adr_ville dans la table ADRESSE) ayant réservé une chambre du premier étage (étage « 1er ») entre le 20 et le 25 décembre 2007.

Client

Résultat attendu :

Client

Monsieur FAURE Alain
Mademoiselle DUHAMEL Evelyne
Monsieur DUBOIS Paul
Monsieur COULOMB Renaud

Utiliser la fonction **REPLACE** pour enlever les espaces inutiles du nom :
`REPLACE (ch_entrée,
 ch_à_replacer,
 ch_de_replacement)`

Ecrire 2 versions pour cette requête :

- l'une n'utilisant que des jointures (JOIN)
- l'autre utilisant des sous-requêtes chaque fois que cela est possible.

Requête 3 :

- Donner la liste des agents d'entretien (agt_nom, agt_prenom, agt_emb) en charge **d'au moins une chambre** et triés par date d'embauche (du plus ancien au plus récent). (il y en a 4)
- Idem mais donner le nom uniquement **des deux premiers** agents d'entretien **embauchés** par l'hôtel. Afficher uniquement l'année d'embauche.

Aide : On utilisera une sous-requête dans le FROM (Voir ci-après)

Agent	Embauche
-------	----------


Résultat attendu :

Agent

PEZIN Franck
BOUSSEL Annie

Embauche

2003
2004



La fonction **EXTRACT** permet d'extraire une partie donnée d'une date.
La fonction **EXTRACT** renvoie un nombre.
On l'utilise avec YEAR ou MONTH (l'un ou l'autre).
Syntaxe : **EXTRACT** (YEAR|MONTH FROM col_date)
Exemple : **EXTRACT** (MONTH FROM SYSDATE) → 09



La pseudo_colonne **ROWNUM** retourne un entier indiquant l'ordre séquentiel de chaque enregistrement extrait de la requête.

```
SELECT ... FROM ... WHERE ROWNUM < 5 ;
```

Pour obtenir les deux premiers enregistrements concernés par un tri il est nécessaire **d'effectuer la sélection sur les enregistrements préalablement triés.**

Pour cela, on a recours à une sous-requête dans le **FROM** :

```
SELECT ...  
FROM (SELECT ... FROM ... ORDER BY ...)  
WHERE ROWNUM < 5 ;
```

Requête 4 : Regroupements

- a) Donner le nombre de chambres réservées (inscrites au planning) et le nombre de clients différents pour le mois de novembre 2007.

Nb Réservations	Nb clients
-----------------	------------

Résultat attendu :

<u>Nb Reservations</u>	<u>Nb clients</u>
389	95

- b) Donner le nombre de chambres réservées (inscrites au planning), ainsi le nombre total (cumul) de personnes présentes (nb_pers) dans l'hôtel pour chaque jour du mois de novembre 2007.
Tri sur le jour.

Jour	Nb chambres	Nb personnes
------	-------------	--------------

Résultat attendu (extrait) : 30 lignes pour le mois de novembre car 30 jours en novembre

<u>Jour</u>	<u>Nb Chambres</u>	<u>Nb Personnes</u>
jeudi 01	16	34
vendredi 02	13	22
samedi 03	12	23
....		



La clause de regroupement **GROUP BY** permet de regrouper les lignes selon la valeur des expressions, colonnes, fonction, constantes ou calculs.

Les colonnes présentes dans le **SELECT** doivent apparaître dans le **GROUP BY**.

Seules des fonctions ou des expressions de regroupement (SUM, COUNT...) peuvent exister en plus dans le **SELECT**.

Les alias de colonnes ne peuvent pas être utilisés dans le **GROUP BY**.

Requête 5 : Liste des jours **de 2007** où l'hôtel a été complet (jour où toutes les chambres de l'hôtel sont réservées, c'est-à-dire où le nombre de chambres inscrites au planning est égal au nombre total de chambres dans l'hôtel) (division).

Jours Hôtel Complet

Résultat attendu :

<u>Jours Hôtel Complet</u>
Samedi 24 Mars 2007

Requête 6 :

- a) Donner le numéro (cli_id), le nom et le prénom des clients pour lesquels **plusieurs adresses mail** différentes ont été enregistrées. Tri des clients par ordre alphabétique.

CLI_ID	CLIENT	NB ADR.
--------	--------	---------

Résultat attendu :

CLI_ID	CLIENT	NB ADR.
9	BOYER Martine	3 adr. mail
98	DUQUESNAY Jacques	2 adr. mail
2	MARTIN Marc	2 adr. mail

- b) Donner le numéro (`cli_id`), le nom et le prénom des clients pour lesquels plusieurs adresses postales **et/ou** plusieurs adresses mails différentes ont été mémorisées (*opérateur ensembliste UNION*).

Tri des clients par ordre alphabétique.

Remarque : Si le client a à la fois plusieurs adresses mail et plusieurs adresses postales, il doit être affiché deux fois.

CLI_ID	CLIENT	PLUSIEURS ADR
--------	--------	---------------

Résultat attendu :

CLI_ID	CLIENT	PLUSIEURS ADR
9	BOYER Martine	3 adr. mail
49	COULOMB Renaud	2 adr. postales
68	DECONINCK Patricia	2 adr. postales
104	DIEUDONNE César	3 adr. postales
98	DUQUESNAY Jacques	2 adr. mail
98	DUQUESNAY Jacques	2 adr. postales
2	MARTIN Marc	2 adr. mail

Requête 7 : Liste des agents d'entretien qui ont entre 50 et 60 ans avec les chambres qu'ils ont en charge. Les agents ayant entre 50 et 60 ans **non affectés** à une chambre doivent également apparaître. (11 lignes)

Agent	Age	Chambre	Etage
-------	-----	---------	-------

Ecrire **2 versions** pour cette requête en utilisant les deux syntaxes différentes vues pour la jointure : syntaxe classique (jointure dans le `WHERE`) puis avec `JOIN`.

Extrait du résultat attendu :

Agent	Age	Chambre	Etage
M. Laurent SAUVAGE	52	N°12	1er
...			
Mme Isabelle BEVIERE	58	Aucune chambre	--
Mme Régine FECHOL	53	N°9	1er
...			



La fonction **NVL** permet de convertir une expression susceptible d'être nulle en une valeur réelle :

NVL(chp, 'Valeur Nulle') → affiche la valeur de **chp** si elle n'est pas nulle ou « *Valeur Nulle* » sinon

Attention les 2 paramètres de NVL doivent être de **mêmes types**.

La fonction **NVL2** s'utilise, elle, avec 3 paramètres :

NVL2(chp, val1, val2) → affiche la valeur *val1* si la valeur de **chp** n'est pas nulle ou *val2* sinon

Requête 8 : Liste des clients mémorisés dans la base (on leur a juste envoyé de la documentation) mais qui n'ont jamais effectué de réservation à l'hôtel (c'est-à-dire n'ont jamais été au planning de l'hôtel).

(4 clients sont concernés, n'afficher que l'adresse principale si le client en possède plusieurs : `adr_id = 1`).

Nom – Prénom	Adresse	Code postal	ville
--------------	---------	-------------	-------

Ecrire 4 versions pour cette requête :

- a) avec une jointure externe,
- b) avec une sous requête (opérateur =, IN, ou NOT IN)
- c) avec une sous-requête (opérateur EXISTS ou NOT EXISTS)
- d) avec un opérateur de la théorie des ensembles

Requête 9 : Sélection de groupes

Donner le Client ayant cumulé le plus grand nombre de jours passés à l'hôtel en 2007.

(On compte le nb de jours par client mais **attention** à ne pas compter deux fois le même jour dans le cas où le client a réservé plusieurs chambres le même jour).

Aide : On pourra commencer par afficher le nombre de jours distincts passés à l'hôtel pour chaque client.

Meilleur client 2007	Email	Nb jours
----------------------	-------	----------

Résultat à obtenir :

Meilleur client 2007	Email	Nb Jours
Monsieur LACOMBE Paul	rantanplan@aol.com	65

Requête 10 :

- a) Liste de tous les clients de **PARIS** avec leurs adresses mail.
On affichera **tous les clients de Paris** même s'ils n'ont pas d'adresse mail.
Aide : Il y a 24 clients à Paris en tout.

Nom	Prénom	Adr. Email
-----	--------	------------

- b) Liste de tous les clients de PARIS avec, s'ils en ont, leurs numéros de FAX (type_code = FAX) ainsi que leurs adresses mail. On affichera **tous** les clients de Paris même s'ils n'ont pas de numéro de FAX ou pas d'adresse mail.
Aide : Il y en a toujours 24 (Vérifier que vous affichez les clients qui n'ont pas de numéro de FAX).

Deux méthodes possibles :

- une méthode utilisant des jointures externes (+) et une sous-requête dans le FROM
- une méthode n'utilisant aucune sous-requête (uniquement LEFT/RIGHT JOIN)

Nom	Prénom	Numéro Fax	Adr. Email
-----	--------	------------	------------



Dans le cas d'une condition sur une table concernée par une jointure externe, il faut garder la main sur l'ordre d'exécution des conditions.

Deux solutions :

- Utiliser la syntaxe RIGHT/LEFT JOIN

```
SELECT ... FROM T1 LEFT JOIN T2 ON (jointure AND condition)
```

La condition sera alors exécutée **AVANT** la jointure contrairement aux conditions de la clause **WHERE**
- Si on utilise la syntaxe ORACLE des jointures externes utilisant (+), une sous-requête dans le FROM permet de créer une priorité dans les conditions

Requête 11 :

- a) Liste des chambres avec numéro des clients qui les ont réservées le **26 décembre 2007**.
Les chambres n'ayant pas été réservées ce jour-là devront malgré tout apparaître dans la liste.
(Aide : 20 lignes à obtenir car 20 chambres en tout)

A faire avec les 2 syntaxes connues.

Chambre	N°Client
---------	----------

- b) Modifier l'une des requêtes (au choix) pour qu'elle affiche les noms et prénoms des clients à la place des numéros.

Chambre	Client
Chambre N°1	NON RESERVEE
Chambre N°2	NON RESERVEE
Chambre N°3	NON RESERVEE
Chambre N°4	HESS Lucette
...	

Requête 12 : Liste des agents d'entretien de sexe féminin (agt_sx = 2) avec le nombre de chambres de plus de 2 couchages (chb_couchage) qu'elles ont en charge.

Afficher tous les agents d'entretien de sexe féminin, y compris ceux qui n'ont en charge aucune chambre de plus de 2 couchages. Utiliser 2 méthodes différentes.

Résultat Attendu :

Agent Entretien	Nb Chambres
SIMEON	Pas de Chambre de plus de 2 Couchages
BOUSSEL	7 Chambres de plus de 2 couchages
BEVIERE	Pas de Chambre de plus de 2 Couchages
FECHOL	Pas de Chambre de plus de 2 Couchages



En SQL, la commande **CASE ... WHEN ...** peut être utilisée comme les conditions de type "si / sinon" pour retourner un résultat disponible entre plusieurs possibilités. Le **CASE** peut être utilisé dans n'importe quelle instruction ou clause, telle que SELECT, UPDATE, DELETE, WHERE, ORDER BY ou HAVING.

Syntaxe :

CASE expression

WHEN v1 **THEN** expr1

[WHEN v2 **THEN** expr2]...

[ELSE expr_défaut]

END

SELECT nom **AS** pilote,

CASE sx

WHEN 1 **THEN** 'Homme'

WHEN 2 **THEN** 'Femme'

ELSE 'Inconnu'

END **AS** sexe

FROM pilote;

Requête 13 : Liste des clients qui ne sont pas de la marne (dont le code postal ne commence pas par 51) et qui n'ont pas séjourné à l'hôtel (Il y en 3). **A faire avec uniquement des jointures.**

Client	Ville	Téléphone
--------	-------	-----------

Où Ville : ville de l'adresse principale (adr_id =1) du client

Téléphone : numéro de téléphone fixe (typ_code = 'TEL') ou « Numéro Inconnu » le cas échéant.

CLIENT

DIEUDONNE

MAIRE

MERTON

VILLE

COMPIEGNE

TROYES

RUEIL MALMAISON

TELEPHONE

03-44-20-27-25

Numéro inconnu

Numéro inconnu

Pour ceux qui sont en avance :

Requête 6 b) : A faire avec des sous-requêtes synchronisées. Dans ce cas on ne pourra pas afficher le nombre d'adresses afficher uniquement « Mail » ou « Postale » dans la dernière colonne.

Requête 10 c) : Liste de tous les clients de Paris (nom-prénom) qui n'ont ni numéro de FAX et ni d'adresse mail. (Méthode au choix, avec sous-requêtes NOT IN étant la méthode la plus simple). (Aide : 12 lignes à obtenir)

Requête 13 : Trouver deux autres méthodes qui donnent le même résultat.