

Travaux Pratiques 1-2 – Etude du langage SQL

Présentation

11 exercices (20 pts)

Dans ces exercices, vous allez :

- + Utiliser :
 - Soit **Oracle SQL Developer** avec **Oracle XE** pour examiner des objets de base de données dans le compte **HR** (déverrouillé préalablement).
 - Soit **Oracle Application Express** sur le site <http://apex.oracle.com>
- + Tous les exercices écrits utilisent l'environnement de développement Oracle SQL Developer avec Oracle XE ou l'environnement APEX en ligne. Vous pouvez aussi utiliser **SQL*Plus** si vous avez installé Oracle XE sur votre station de travail.

Avant de commencer les exercices, notez les points suivants :

- + Enregistrez éventuellement vos fichiers d'exercices dans des scripts nommés **ex<numero>.sql** dans un répertoire créé préalablement, exemple: **c:\EXSQL**
- + Si vous utilisez SQL Developer, entrez vos instructions SQL dans une feuille de calcul SQL Worksheet. Pour enregistrer un script dans SQL Developer, vérifiez que la feuille de calcul SQL requise est active puis sélectionnez **File→Save As**. Vous pouvez aussi cliquer avec le bouton droit de la souris dans la feuille de calcul SQL Worksheet et sélectionner **Save file** pour enregistrer votre instruction SQL dans d'un script nommé **ex<numero>.sql**
- + Lorsque vous modifiez un script existant, veillez à utiliser l'option **Save As** pour l'enregistrer sous un autre nom.
- + Pour exécuter l'interrogation, cliquez sur l'icône **Execute Statement** dans la feuille de calcul SQL Worksheet. Vous pouvez également appuyer sur **[F9]**.
- + Pour les instructions **LMD** et **LDD**, utilisez l'icône **Run Script** ou appuyez sur **[F5]**.
- + Une fois l'interrogation exécutée, veillez à ne pas entrer l'interrogation suivante dans la même feuille de calcul. Ouvrez une nouvelle feuille.
- + Si vous utilisez l'environnement **APEX**, enregistrez le texte SQL dans un fichier au format texte ayant comme extension « .sql »

IMPORTANT

A la fin des séances de TP 01 et TP 02, vous envoyez un compte rendu ainsi qu'un **zip de vos scripts** à l'enseignant sur « Moodle. » :

- **TP 01 séances 1-2** – 5 premiers exercices
- **TP 02 séances 3-4** – 6 derniers exercices


Attention : Les données et le nom de certaines tables des exemples qui sont présentés dans ce document ne correspondent pas exactement à ce qui existe dans la base de données, cette dernière ayant été créée plus récemment.

Si vous utilisez l'environnement en ligne APEX vous devrez transposer et adapter dans ce document les énoncés décrits avec Oracle SQL Developer.

Exercice 1 (2 pt)

1.1 Première partie

1. Démarrez Oracle SQL Developer à l'aide de l'icône correspondante sur le bureau.
2. Créer une connexion de base de données pour Oracle SQL Developer pour le schéma HR (charger le script load_sample.sql si vous utilisez APEX)
3. Dans le navigateur de connexions, visualisez les objets à votre disposition sous le noeud Tables. Vérifiez que les tables suivantes sont présentes :

-  COUNTRIES
-  DEPARTMENTS
-  EMPLOYEES
-  JOB_GRADES
-  JOB_HISTORY
-  JOBS
-  LOCATIONS
-  REGIONS

4. Naviguez dans la structure de la table **EMPLOYEES**.
5. Visualisez les données de la table **DEPARTMENTS**.
6. Ouvrir une feuille de calcul SQL Worksheet (APEX SQL Workshop -> Commandes SQL)

Syntaxe -> Testez vos connaissances :

1. L'instruction SELECT suivante est-elle exécutée avec succès : (oui/non) ?

```
SELECT last_name, job_id, salary AS Sal  
FROM employees;
```

2. L'instruction SELECT suivante est-elle exécutée avec succès: (oui/non) ?

```
SELECT *  
FROM JOB_HISTORY;
```

3. L'instruction suivante présente quatre erreurs de codage. Pouvez-vous les identifier ?

```
SELECT employee_id, last_name  
sal x 12 ANNUAL SALARY  
FROM employees;
```

1.2 Deuxième Partie

Votre première tâche consiste à créer des états à partir des données des tables du département des ressources humaines.

Vous devez d'abord déterminer la structure et le contenu de la table **DEPARTMENTS**.

```
DESCRIBE departments
```

Name	Null	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

4 rows selected

Vous devez déterminer la structure de la table **EMPLOYEES**.

```
DESCRIBE employees
```

Name	Null	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

11 rows selected

1. Le département des ressources humaines souhaite une interrogation affichant le nom, l'ID de poste, la date d'embauche et l'ID d'employé de chaque employé, l'ID d'employé apparaissant en premier. Associez l'alias STARTDATE à la colonne HIRE_DATE.

Enregistrez votre instruction SQL dans un fichier nommé **ex_01_05.sql** afin de pouvoir fournir ce fichier au département des ressources humaines.
Testez l'interrogation enregistrée dans le fichier **ex_01_05.sql** afin de vérifier qu'elle s'exécute correctement.

Remarque : Une fois l'interrogation exécutée, veillez à ne pas entrer l'interrogation suivante dans la même feuille de calcul. Ouvrez une nouvelle feuille.

	EMPLOYEE_ID	LAST_NAME	JOB_ID	STARTDATE
1	100	King	AD_PRES	17-JUN-87
2	101	Kochhar	AD_VP	21-SEP-89
3	102	De Haan	AD_VP	13-JAN-93
4	103	Hunold	IT_PROG	03-JAN-90
5	104	Ernst	IT_PROG	21-MAY-91
6	107	Lorentz	IT_PROG	07-FEB-99
7	124	Mourgos	ST_MAN	16-NOV-99
8	141	Rajs	ST_CLERK	17-OCT-95
9	142	Davies	ST_CLERK	29-JAN-97
10	143	Matos	ST_CLERK	15-MAR-98

2. Le département des ressources humaines souhaite une interrogation affichant tous les ID de postes uniques de la table **EMPLOYEES**.

EMP_ID	JOB_ID
1	AC_ACCOUNT
2	AC_MGR
3	AD_ASST
4	AD PRES
5	AD_VP
6	IT_PROG
7	MK_MAN
8	MK_REP
9	SA_MAN
10	SA_REP
11	ST_CLERK
12	ST_MAN

3. Le département des ressources humaines souhaite des en-têtes de colonne plus descriptifs pour son état sur les employés. Copiez l'instruction enregistrée dans **ex_01_05.sql** dans une nouvelle feuille de calcul SQL Worksheet. Nommez les en-têtes de colonne respectivement : Emp #, Employee, Job et Hire Date. Exécutez à nouveau votre interrogation.

Le département des ressources humaines a demandé un état listant tous les employés avec leur ID de poste. Affichez le nom concaténé avec l'ID de poste (en séparant les deux par une virgule et un espace) et intitulez la colonne **Employee and Title**.

4. Pour vous familiariser avec le contenu de la table **EMPLOYEES**, créez une Interrogation affichant toutes les données de cette table. Séparez les colonnes de Résultat par une virgule. Attribuez le titre de colonne **THE_OUTPUT**.

Results	Script Output	Explain	Autotrace	DBMS Output	OWA Output
Results:					
THE_OUTPUT					
1	100,Steven,King,SKING,515.123.4567,AD_PRES,,17-JUN-87,24000,,90				
2	101,Neena,Kochhar,NKOCHHAR,515.123.4568,AD_VP,100,21-SEP-89,17000,,90				
3	102,Lex,De Haan,LDEHAAN,515.123.4569,AD_VP,100,13-JAN-93,17000,,90				
4	103,Alexander,Hunold,AHUNOLD,590.423.4567,IT_PROG,102,03-JAN-90,9000,,60				
5	104,Bruce,Ernst,BERNST,590.423.4568,IT_PROG,103,21-MAY-91,6000,,60				
6	107,Diana,Lorentz,DLORENTZ,590.423.5567,IT_PROG,103,07-FEB-99,4200,,60				
7	124,Kevin,Mourgos,KMOURGOS,650.123.5234,ST_MAN,100,16-NOV-99,5800,,50				
8	141,Trenna,Rajs,TRAJS,650.121.8009,ST_CLERK,124,17-OCT-95,3500,,50				
9	142,Curtis,Davies,CDAVIES,650.121.2994,ST_CLERK,124,29-JAN-97,3100,,50				
10	143,Randall,Matos,RMATOS,650.121.2874,ST_CLERK,124,15-MAR-98,2600,,50				

Présentation de l'exercice 2 (2 pts)

Dans cet exercice, vous allez générer de nouveaux états à l'aide d'instructions utilisant les clauses **WHERE** et **ORDER BY**. Vous allez inclure des variables avec esperluette d'interprétation afin de rendre les instructions SQL génériques et de pouvoir les réutiliser.

Exercice 2

Le département des ressources humaines a besoin de votre aide pour créer des interrogations.

1. Pour des raisons budgétaires, ce département a besoin d'un état affichant le nom et le salaire des employés qui gagnent plus de 12 000 \$. Enregistrez vos instructions SQL dans le fichier **ex_02_01.sql**. Exécutez votre interrogation.

	R 2	LAST_NAME	R 2	SALARY
1	King			24000
2	Kochhar			17000
3	De Haan			17000
4	Hartstein			13000

2. Ouvrez une nouvelle feuille de calcul SQL Worksheet. Créez un état affichant le nom et le numéro de département correspondant à l'ID d'employé 176. Exécutez l'interrogation.

	R 2	LAST_NAME	R 2	DEPARTMENT_ID
1	Taylor			80

3. Le département des ressources humaines a besoin de connaître les employés dont le salaire est élevé et ceux dont le salaire est faible. Modifiez le fichier **ex_02_01.sql** pour afficher le nom et le salaire des employés dont le salaire ne figure pas dans la plage de 5 000 \$ à 12 000 \$. Enregistrez votre instruction SQL dans le fichier **ex_02_03.sql**.

	R 2	LAST_NAME	R 2	SALARY
1	King			24000
2	Kochhar			17000
3	De Haan			17000
4	Lorentz			4200
5	Rajs			3500
6	Davies			3100
7	Matos			2600
8	Vargas			2500
9	Whalen			4400
10	Hartstein			13000

4. Créez un état affichant le nom, l'ID de poste et la date d'embauche des employés nommés Matos et Taylor. Triez les données par ordre croissant en fonction de la date d'embauche

	A Z	LAST_NAME	A Z	JOB_ID	HIRE_DATE
1		Matos		ST_CLERK	15-MAR-98
2		Taylor		SA_REP	24-MAR-98

5. Affichez le nom et le numéro de département de tous les employés du département 20 ou 50 par ordre alphabétique croissant, en fonction du nom.

	A Z	LAST_NAME	A Z	DEPARTMENT_ID
1		Davies		50
2		Fay		20
3		Hartstein		20
4		Matos		50
5		Mourgos		50
6		Rajs		50
7		Vargas		50

6. Modifiez le fichier **ex_02_03.sql** pour afficher le nom et le salaire des employés qui gagnent entre 5 000 \$ et 12 000 \$, et travaillent dans le département 20 ou 50. Intitulez respectivement les colonnes **Employee** et **Monthly Salary**. Enregistrez le fichier ex_02_03.sql sous le nom **ex_02_06.sql**. Exécutez l'instruction figurant dans **ex_02_06.sql**.

	A Z	Employee	A Z	Monthly Salary
1		Fay		6000
2		Mourgos		5800

7. Le département des ressources humaines a besoin d'un état affichant le nom et la date d'embauche de tous les employés embauchés en 1994.




	A Z	LAST_NAME	HIRE_DATE
1		Higgins	07-JUN-94
2		Gietz	07-JUN-94

8. Créez un état affichant le nom et l'intitulé de poste de tous les employés qui n'ont pas de manager.

	A Z	LAST_NAME	A Z	JOB_ID
1		King		AD_PRES

9. Créez un état affichant le nom, le salaire et la commission de tous les employés qui perçoivent des commissions. Triez les données par ordre décroissant en fonction du salaire et des commissions.



Utilisez la position numérique de la colonne dans la clause ORDER BY.

	 LAST_NAME	 SALARY	 COMMISSION_PCT
1	Abel	11000	0.3
2	Zlotkey	10500	0.2
3	Taylor	8600	0.2
4	Grant	7000	0.15

10. Les membres du département des ressources humaines souhaitent davantage de souplesse dans les interrogations que vous écrivez. Ils voudraient un état affichant le nom et le salaire des employés qui gagnent plus qu'un montant saisi par l'utilisateur en réponse à une invite (**&variable** (SQL Dev) ou **:variable**(APEX)).

Enregistrez cette interrogation dans le fichier

ex_02_10.sql. Si, à l'invite, vous indiquez 12000, l'état affiche les résultats suivants :

	 LAST_NAME	 SALARY
1	King	24000
2	Kochhar	17000
3	De Haan	17000
4	Hartstein	13000

Exercice 3 (2 pts)

1.3 Première partie

1. Ecrivez une interrogation permettant d'afficher la date système. Nommez la colonne Date.

Date
1 31-MAY-07

2. Le département des ressources humaines a besoin d'un état permettant d'afficher le numéro d'employé, le nom, le salaire et le salaire augmenté de 15,5 % (exprimé sous la forme d'un nombre entier) pour chaque employé. Nommez la colonne New Salary. Enregistrez l'instruction SQL dans le fichier **ex_03_02.sql**.

3. Exécutez l'interrogation figurant dans le fichier ex_03_02.sql.

	EMPLOYEE_ID	LAST_NAME	SALARY	New Salary
1	100	King	24000	27720
2	101	Kochhar	17000	19635
3	102	De Haan	17000	19635
4	103	Hunold	9000	10395
5	104	Ernst	6000	6930
6	107	Lorentz	4200	4851
7	124	Mourgos	5800	6699
8	141	Rajs	3500	4043
9	142	Davies	3100	3581
10	143	Matos	2600	3003

4. Modifiez l'interrogation ex_03_02.sql pour ajouter une colonne permettant de soustraire l'ancien salaire du nouveau. Nommez la colonne Increase. Enregistrez le contenu du fichier sous le nom ex_03_04.sql. Exécutez l'interrogation modifiée.

	EMPLOYEE_ID	LAST_NAME	SALARY	New Salary	Increase
1	100	King	24000	27720	3720
2	101	Kochhar	17000	19635	2635
3	102	De Haan	17000	19635	2635
4	103	Hunold	9000	10395	1395
5	104	Ernst	6000	6930	930

5. Ecrivez une interrogation permettant d'afficher le nom (la première lettre en majuscule et toutes les autres lettres en minuscules) et la longueur du nom de tous les employés dont le nom commence par les lettres "J", "A" ou "M". Attribuez à chaque colonne un libellé approprié. Triez les résultats en fonction du nom des employés.

	A Z	Name	A Z	Length
1		Abel		4
2		Matos		5
3		Mourgos		7

6. Réécrivez l'interrogation de sorte que l'utilisateur soit invité à saisir la lettre par laquelle le nom doit commencer. Par exemple, si l'utilisateur saisit "H" (en majuscule) à l'invite, le résultat doit afficher tous les employés dont le nom commence par la lettre "H".

	A Z	Name	A Z	Length
1		Hartstein		9
2		Higgins		7
3		Hunold		6

Avec APEX utiliser une « bind variable », entrer H suivi du caractère « % »

Variable attachée	Valeur
:START_LETTER	H%

7. Modifiez l'interrogation de sorte que la casse de la lettre saisie n'affecte pas le résultat. La lettre saisie doit être convertie en majuscule avant traitement par l'interrogation SELECT.

Enter Substitution Variable

START_LETTER:

h

OK

Cancel

	A Z	Name	A Z	Length
1		Hartstein		9
2		Higgins		7
3		Hunold		6

6. Le département des ressources humaines souhaite connaître l'ancienneté de chaque employé. Pour chacun d'eux, affichez le nom et calculez le nombre de mois entre la date du jour et la date d'embauche de l'employé. Nommez la colonne MONTHS_WORKED. Triez les résultats sur la base du nombre de mois d'ancienneté.

Arrondissez le nombre de mois au nombre entier supérieur le plus proche.

Remarque : Cette interrogation dépendant de sa date d'exécution, les valeurs de la colonne MONTHS_WORKED seront différentes dans votre cas.

	A Z	LAST_NAME	A Z	MONTHS_WORKED
1		Zlotkey		88
2		Mourgos		90
3		Grant		96
4		Lorentz		100
5		Vargas		107
6		Taylor		110
7		Matos		111
8		Fay		117
9		Davies		124
10		Abel		133
11		Hartstein		135
12		Rajs		139
13		Higgins		156
14		Gietz		156
15		De Haan		173
16		Ernst		192
17		Hunold		209
18		Kochhar		212
19		Whalen		236
20		King		239

Exercice 4 (2 pts)

1. Créez un état qui produit les éléments suivants pour chaque employé :
 <employee last name> earns <salary> monthly but wants
 <3 times salary.>. Intitulez la colonne Dream Salaries

	A Z	Dream Salaries
1	King	King earns \$24,000.00 monthly but wants \$72,000.00.
2	Kochhar	Kochhar earns \$17,000.00 monthly but wants \$51,000.00.
3	De Haan	De Haan earns \$17,000.00 monthly but wants \$51,000.00.
4	Hunold	Hunold earns \$9,000.00 monthly but wants \$27,000.00.
5	Ernst	Ernst earns \$6,000.00 monthly but wants \$18,000.00.

Utilisez le masque 'fm\$99,999.00' pour le salaire

2. Pour chaque employé, affichez le nom, la date d'embauche et la date de révision du salaire, soit le premier lundi après six mois d'ancienneté. Intitulez la colonne REVIEW. Affichez les dates sous la forme "Monday, the Thirty-First of July, 2000".

	A Z	LAST_NAME	HIRE_DATE	A Z	REVIEW
1	King	King	17-JUN-87		Monday, the Twenty-First of December, 1987
2	Kochhar	Kochhar	21-SEP-89		Monday, the Twenty-Sixth of March, 1990
3	De Haan	De Haan	13-JAN-93		Monday, the Nineteenth of July, 1993
4	Hunold	Hunold	03-JAN-90		Monday, the Ninth of July, 1990
5	Ernst	Ernst	21-MAY-91		Monday, the Twenty-Fifth of November, 1991

Utilisez le masque 'fmDay, "the" Ddspth "of" Month, YYYY' :

```
TO_CHAR(NEXT_DAY(ADD_MONTHS(hire_date, 6), 'MONDAY'),
'fmDay, "the" Ddspth "of" Month, YYYY')
```



Remplacer éventuellement MONDAY par LUNDI

3. Affichez le nom, la date d'embauche et le jour de la semaine où l'employé a commencé. Intitulez la colonne DAY. Triez les résultats en fonction du jour de la semaine, en commençant par le lundi (Monday).

```
SELECT last_name, hire_date,
TO_CHAR(hire_date, 'DAY') DAY
FROM employees
ORDER BY TO_CHAR(hire_date - 1, 'd');
```



	 LAST_NAME	HIRE_DATE	 DAY
1	Grant	24-MAY-99	MONDAY
2	Gietz	07-JUN-94	TUESDAY
3	Taylor	24-MAR-98	TUESDAY
4	Higgins	07-JUN-94	TUESDAY
5	Rajs	17-OCT-95	TUESDAY

4. Créez une interrogation qui affiche le nom et le montant de la commission de chaque employé. Si un employé ne perçoit pas de commission, indiquez "No Commission". Intitulez la colonne COMM.



	 LAST_NAME	 COMM
1	King	No Commission
2	Kochhar	No Commission
3	De Haan	No Commission
4	Hunold	No Commission
5	Ernst	No Commission
6	Lorentz	No Commission

5. A l'aide de la fonction DECODE, écrivez une interrogation qui affiche le niveau de tous les employés sur la base de la valeur de la colonne JOB_ID, à l'aide des données suivantes :

Poste Niveau
 AD_PRES A
 ST_MAN B
 IT_PROG C
 SA_REP D
 ST_CLERK E
 Autres 0

	 JOB_ID	 GRADE
1	AC_ACCOUNT	0
2	AC_MGR	0
3	AD_ASST	0
4	AD_PREP	A
5	AD_VP	0

6. Réécrivez l'instruction de l'exercice précédent à l'aide de la syntaxe CASE.

	 JOB_ID	 GRADE
1	AC_ACCOUNT	0
2	AC_MGR	0
3	AD_ASST	0
4	AD_PREP	A
5	AD_VP	0

Exercice 5 (2 pts)

Indiquez la validité des trois affirmations ci-après. Entourez la réponse correcte.

1. Les fonctions de groupe opèrent sur plusieurs lignes et produisent un résultat par groupe.

Vrai/Faux

2. Les fonctions de groupe prennent en compte les valeurs NULL dans les calculs.

Vrai/Faux

3. La clause WHERE restreint les lignes avant inclusion dans un calcul de groupe.

Vrai/Faux

Le département des ressources humaines a besoin des états suivants :

4. Déterminez le salaire le plus élevé, le salaire le plus bas, le salaire cumulé et le salaire moyen pour tous les employés. Intitulez respectivement les colonnes Maximum, Minimum, Sum et Average. Arrondissez les résultats à l'entier le plus proche.

Enregistrez l'instruction SQL dans le fichier ex_05_04.sql. Exécutez l'interrogation



	A Z	Maximum	A Z	Minimum	A Z	Sum	A Z	Average
1		24000		2500		175500		8775

5. Modifiez l'interrogation enregistrée dans le fichier **ex_05_04.sql** afin d'afficher le salaire minimum, le salaire maximum, le salaire cumulé et le salaire moyen pour chaque type de poste. Enregistrez ex_05_04.sql sous le nom **ex_05_05.sql**.



Exécutez l'instruction figurant dans **ex_05_05.sql**.

	A Z	JOB_ID	A Z	Maximum	A Z	Minimum	A Z	Sum	A Z	Average
1		IT_PROG		9000		4200		19200		6400
2		AC_MGR		12000		12000		12000		12000
3		AC_ACCOUNT		8300		8300		8300		8300
4		ST_MAN		5800		5800		5800		5800
5		AD_ASST		4400		4400		4400		4400
6		AD_VP		17000		17000		34000		17000
7		SA_MAN		10500		10500		10500		10500
8		MK_MAN		13000		13000		13000		13000
9		AD_PRES		24000		24000		24000		24000
10		SA_REP		11000		7000		26600		8867
11		MK_REP		6000		6000		6000		6000
12		ST_CLERK		3500		2500		11700		2925


Ecrivez une interrogation permettant d'afficher le nombre de personnes occupant le même poste.

	 JOB_ID	 COUNT(*)
1	AC_ACCOUNT	1
2	AC_MGR	1
3	AD_ASST	1
4	AD_PRES	1
5	AD_VP	2
6	IT_PROG	3
7	MK_MAN	1
8	MK_REP	1
9	SA_MAN	1
10	SA_REP	3
11	ST_CLERK	4
12	ST_MAN	1


6. Généralisez l'interrogation de sorte que l'utilisateur du département des ressources humaines soit invité à indiquer un intitulé de poste. Enregistrez le script sous le nom de fichier **ex_05_06.sql**. Exécutez l'interrogation. A l'invite, saisissez IT_PROG

	 JOB_ID	 COUNT(*)
1	IT_PROG	3

7. Déterminez le nombre de managers sans les répertorier. Intitulez la colonne Number of Managers. Indice : Utilisez la colonne MANAGER_ID pour déterminer le nombre de Managers

	 Number of Managers
1	8

8. Trouvez la différence entre le salaire le plus élevé et le salaire le plus bas. Intitulez la colonne DIFFERENCE.

	 DIFFERENCE
1	21500

9. Créez un état permettant d'afficher le numéro de manager et le salaire de l'employé le moins payé sous les ordres de ce manager. Excluez toute personne pour laquelle le manager n'est pas connu. Excluez les groupes dans lesquels le salaire minimum est inférieur ou égal à 6 000 \$. Triez les résultats par ordre décroissant sur la base du salaire

	MANAGER_ID	MIN(SALARY)
1	102	9000
2	205	8300
3	149	7000

10. Créez une interrogation permettant d'afficher le nombre total d'employés et, pour ce total, le nombre d'employés embauchés en 1995, 1996, 1997 et 1998 (2005, 2006, 2007, 2008). Créez les en-têtes de colonne appropriés.

	TOTAL	1995	1996	1997	1998
1	20	1	2	2	3

11. Créez une interrogation de matrice permettant d'afficher le poste, le salaire correspondant à ce poste sur la base du numéro de département et le salaire total correspondant à ce poste, pour les départements 20, 50, 80 et 90, en intitulant chaque colonne de façon appropriée

	Job	Dept 20	Dept 50	Dept 80	Dept 90	Total
1	IT_PROG	(null)	(null)	(null)	(null)	19200
2	AC_MGR	(null)	(null)	(null)	(null)	12000
3	AC_ACCOUNT	(null)	(null)	(null)	(null)	8300
4	ST_MAN	(null)	5800	(null)	(null)	5800
5	AD_ASST	(null)	(null)	(null)	(null)	4400
6	AD_VP	(null)	(null)	(null)	34000	34000
7	SA_MAN	(null)	(null)	10500	(null)	10500
8	MK_MAN	13000	(null)	(null)	(null)	13000
9	AD_PRES	(null)	(null)	(null)	24000	24000
10	SA_REP	(null)	(null)	19600	(null)	26600
11	MK_REP	6000	(null)	(null)	(null)	6000
12	ST_CLERK	(null)	11700	(null)	(null)	11700

Exercice 6 (2 pts)

1. Ecrivez pour le département des ressources humaines une interrogation produisant l'adresse de tous les départements. Utilisez les tables **LOCATIONS** et **COUNTRIES**.

Affichez dans les résultats l'ID de lieu, la rue, la ville, le département et le pays. Utilisez une jointure naturelle (NATURAL JOIN) pour produire les résultats.

	LOCATION_ID	STREET_ADDRESS	CITY	STATE_PROVINCE	COUNTRY_NAME
1	1400	2014 Jabberwocky Rd	Southlake	Texas	United States of America
2	1500	2011 Interiors Blvd	South San Francisco	California	United States of America
3	1700	2004 Charade Rd	Seattle	Washington	United States of America
4	1800	460 Bloor St. W.	Toronto	Ontario	Canada
5	2500	Magdalen Centre, The ...	Oxford	Oxford	United Kingdom

2. Le département des ressources humaines a besoin d'un état de tous les employés.

Ecrivez une interrogation permettant d'afficher le nom, ainsi que le numéro et le nom de département, pour tous les employés.

	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
1	Whalen	10	Administration
2	Hartstein	20	Marketing
3	Fay	20	Marketing
4	Davies	50	Shipping
5	Vargas	50	Shipping
6	Rajs	50	Shipping
7	Mourgos	50	Shipping
8	Matos	50	Shipping
9	Hunold	60	IT
10	Ernst	60	IT

3. Le département des ressources humaines a besoin d'un état des employés de Toronto.

Affichez le nom, le poste, ainsi que le numéro et le nom de département, pour tous les employés qui travaillent à Toronto.

	LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
1	Hartstein	MK_MAN	20	Marketing
2	Fay	MK_REP	20	Marketing

4. Créez un état permettant d'afficher le nom et le numéro des employés, ainsi que le nom et le numéro de leur manager. Intitulez respectivement les colonnes Employee, Emp#, Manager et Mgr#. Enregistrez l'instruction SQL sous le nom **ex_06_04.sql**. Exécutez l'interrogation.

	 Employee	 EMP#	 Manager	 Mgr#
1	Kochhar	101	King	100
2	De Haan	102	King	100
3	Hunold	103	De Haan	102
4	Ernst	104	Hunold	103
5	Lorentz	107	Hunold	103
6	Mourgos	124	King	100
7	Rajs	141	Mourgos	124
8	Davies	142	Mourgos	124
9	Matos	143	Mourgos	124
10	Vargas	144	Mourgos	124

Exercice 7 (2 pts)

1. Créez un état qui affiche le numéro d'employé, le nom et le salaire de tous les employés qui gagnent plus que le salaire moyen. Triez les résultats par ordre croissant sur la base du salaire.

	R 2	EMPLOYEE_ID	R 2	LAST_NAME	R 2	SALARY
1		103		Hunold		9000
2		149		Zlotkey		10500
3		174		Abel		11000
4		205		Higgins		12000
5		201		Hartstein		13000
6		101		Kochhar		17000
7		102		De Haan		17000
8		100		King		24000

2. Ecrivez une interrogation qui affiche le numéro d'employé et le nom de tous les employés qui travaillent dans un département comprenant un employé dont le nom contient la lettre "u". Enregistrez l'instruction SQL sous le nom ex_07_03.sql. Exécutez votre interrogation.

	R 2	EMPLOYEE_ID	R 2	LAST_NAME
1		124		Mourgos
2		141		Rajs
3		142		Davies
4		143		Matos
5		144		Vargas
6		103		Hunold
7		104		Ernst
8		107		Lorentz

3. Le département des ressources humaines a besoin d'un état qui affiche le nom, l'ID de département et l'ID de poste de tous les employés dont l'ID de lieu de département est 1700.

	LAST_NAME	DEPARTMENT_ID	JOB_ID
1	Whalen	10	AD_ASST
2	King	90	AD_PRES
3	Kochhar	90	AD_VP
4	De Haan	90	AD_VP
5	Higgins	110	AC_MGR
6	Gietz	110	AC_ACCOUNT

Modifiez l'interrogation de sorte que l'utilisateur soit invité à indiquer un ID de lieu. Enregistrez-la dans le fichier **ex_07_03.sql**.

4. Créez pour les ressources humaines un état affichant le nom et le salaire de tous les employés dont le manager est King.

	LAST_NAME	SALARY
1	Kochhar	17000
2	De Haan	17000
3	Mourgos	5800
4	Zlotkey	10500
5	Hartstein	13000

5. Créez pour les ressources humaines un état affichant l'ID de département, le nom et l'ID de poste de tous les employés du département Executive

	DEPARTMENT_ID	LAST_NAME	JOB_ID
1	90	King	AD_PRES
2	90	Kochhar	AD_VP
3	90	De Haan	AD_VP

6. Modifiez l'interrogation ex_07_03.sql pour afficher le numéro d'employé, le nom et le salaire de tous les employés qui gagnent plus que le salaire moyen et qui travaillent dans un département comprenant un employé dont le nom contient la lettre "u".

Enregistrez ex_07_03.sql sous le nom **ex_07_07.sql**. Exécutez l'instruction figurant dans ex_07_07.sql.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	103	Hunold	9000

Exercice 8 (2 pts)

1. Le département des ressources humaines a besoin de la liste des ID des départements qui ne contiennent pas l'ID de poste ST_CLERK. Utilisez les opérateurs ensemblistes pour créer cet état.

	DEPARTMENT_ID
1	10
2	20
3	60
4	80
5	90
6	110
7	190

2. Le département des ressources humaines a besoin de la liste des pays dans lesquels il n'existe aucun département. Affichez l'ID et le nom des pays. Utilisez les opérateurs ensemblistes pour créer cet état.

	COUNTRY_ID	COUNTRY_NAME
1	DE	Germany

3. Produisez la liste des postes des départements 10, 50 et 20, dans cet ordre. Affichez l'ID de poste et l'ID de département à l'aide des opérateurs ensemblistes.

	JOB_ID	DEPARTMENT_ID
1	AD_ASST	10
2	ST_MAN	50
3	ST_CLERK	50
4	MK_MAN	20
5	MK_REP	20

4. Créez un état répertoriant l'ID d'employé et l'ID de poste des employés dont l'intitulé de poste actuel est identique à l'intitulé de poste initial lors de leur embauche par l'entreprise. (Ces employés ont changé de poste, puis sont revenus à leur poste d'origine.)

	EMPLOYEE_ID	JOB_ID
1	176	SA_REP
2	200	AD_ASST

5. Le département des ressources humaines a besoin d'un état avec les spécifications suivantes :

- ✚ Nom et ID de département de tous les employés de la table EMPLOYEES, qu'ils appartiennent ou non à un département.
- ✚ ID et nom de tous les départements de la table DEPARTMENTS, qu'ils comptent des employés ou non.

Pour ce faire, écrivez une interrogation composée.

	A 2	LAST_NAME	A 2	DEPARTMENT_ID	A 2	TO_CHAR(NULL)
1		Abel		80		(null)
2		Davies		50		(null)
3		De Haan		90		(null)
4		Ernst		60		(null)
5		Fay		20		(null)
6		Gietz		110		(null)
7		Grant		(null)		(null)
8		Hartstein		20		(null)
9		Higgins		110		(null)
10		Hunold		60		(null)
11		King		90		(null)
12		Kochhar		90		(null)
13		Lorentz		60		(null)
14		Matos		50		(null)
15		Mourgos		50		(null)
16		Rajs		50		(null)
17		Taylor		80		(null)
18		Vargas		50		(null)

Exercice 9 (2 pts)

Le département des ressources humaines a besoin d'instructions SQL permettant d'insérer, de mettre à jour et de supprimer des données relatives aux employés. Vous allez tester ces instructions sur la table MY_EMPLOYEE avant de les livrer.

Remarque : Pour exécuter une interrogation comprenant des instructions LMD, utilisez l'icône **Run Script** (ou appuyez sur **[F5]**). Vous pouvez ainsi voir les messages renvoyés dans l'onglet **Script Output**. Pour les interrogations SELECT, continuez à utiliser l'icône **Execute Statement** ou à appuyer sur **[F9]** pour obtenir le résultat mis en forme dans l'onglet **Results**.

1. Créer un script **ex_09_01.sql** pour générer la création de la table MY_EMPLOYEE, avec la structure suivante :

DESCRIBE MY_EMPLOYEE		
Name	Null	Type

ID	NOT NULL	NUMBER(4)
LAST_NAME		VARCHAR2(25)
FIRST_NAME		VARCHAR2(25)
USERID		VARCHAR2(8)
SALARY		NUMBER(9,2)

2. Créez une instruction INSERT permettant d'ajouter à la table MY_EMPLOYEE la première ligne de données du tableau ci-après. N'énumérez pas les colonnes dans la clause INSERT. N'entrez pas encore toutes les lignes.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860
3	Biri	Ben	bbiri	1100
4	Newman	Chad	cnewman	750
5	Ropeburn	Audrey	aropebur	1550

3. Insérez dans la table MY_EMPLOYEE la deuxième ligne de données du tableau qui précède. Cette fois, énumérez les colonnes de façon explicite dans la clause INSERT.

4. Vérifiez les ajouts effectués dans la table.

	ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	1	Patel	Ralph	rpatel	895
2	2	Dancs	Betty	bdancs	860

5. Ecrivez dans un fichier script réutilisable et dynamique une instruction INSERT permettant de charger les lignes restantes dans la table MY_EMPLOYEE. Le script doit afficher une invite pour toutes les colonnes (ID, LAST_NAME, FIRST_NAME, USERID et SALARY). Enregistrez ce script sous le nom **ex_09_05.sql**.

6. Insérez dans la table les deux lignes suivantes du tableau de données fourni à l'étape 3 en exécutant l'instruction INSERT du script que vous avez créé.

7. Vérifiez les ajouts effectués dans la table.

	ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	1	Patel	Ralph	rpatel	895
2	2	Dancs	Betty	bdancs	860
3	3	Biri	Ben	bbiri	1100
4	4	Newman	Chad	cnewman	750

8. Rendez définitifs les ajouts de données.

Mettez à jour et supprimez des données dans la table MY_EMPLOYEE.

9. Remplacez le nom de l'employé 3 par Drexler.






10. Remplacez par 1 000 \$ le salaire de tous les employés qui ont un salaire inférieur à 900 \$.

10. Vérifiez les modifications apportées à la table.

	ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	1	Patel	Ralph	rpatel	1000
2	2	Dancs	Betty	bdancs	1000
3	3	Drexler	Ben	bbiri	1100
4	4	Newman	Chad	cnewman	1000

11. Supprimez Betty Dancs de la table MY_EMPLOYEE.




12. Vérifiez les modifications apportées à la table.

	 ID	 LAST_NAME	 FIRST_NAME	 USERID	 SALARY
1	1	Patel	Ralph	rpatel	1000
2	3	Drexler	Ben	bbiri	1100
3	4	Newman	Chad	cnewman	1000

13. Validez (commit) toutes les modifications en attente.

Contrôlez les transactions de données appliquées à la table MY_EMPLOYEE.

14. Vérifiez les ajouts effectués dans la table.

	 ID	 LAST_NAME	 FIRST_NAME	 USERID	 SALARY
1	1	Patel	Ralph	rpatel	1000
2	3	Drexler	Ben	bbiri	1100
3	4	Newman	Chad	cnewman	1000
4	5	Ropeburn	Audrey	aropebur	1550

15. Marquez un point intermédiaire dans le traitement de la transaction.

Exercice 10 (1 pts)

1. Créez la table DEPT conformément aux indications du tableau ci-après.

Enregistrez l'instruction dans un script nommé **ex_10_01.sql**, puis exécutez ce script pour créer la table. Vérifiez que la table a bien été créée.

Nom de la colonne	ID	NAME
Type de clé	Clé primaire	
NULL/Unique		
Table de clé étrangère		
Colonne de clé étrangère		
Type de données	NUMBER	VARCHAR2
Longueur	7	25

Name	Null	Type
-----	-----	-----
ID	NOT NULL	NUMBER(7)
NAME		VARCHAR2(25)

- Remplissez la table DEPT avec les données de la table DEPARTMENTS. Inclure uniquement les colonnes dont vous avez besoin.
- Créez la table EMP conformément aux indications du tableau ci-après. Enregistrez l'instruction dans un script nommé **ex_10_03.sql**, puis exécutez ce script pour créer la table. Vérifiez que la table a bien été créée.

Nom de la colonne	ID	LAST_NAME	FIRST_NAME	DEPT_ID
Type de clé				
NULL/Unique				
Table de clé étrangère				DEPT
Colonne de clé étrangère				ID
Type de données	NUMBER	VARCHAR2	VARCHAR2	NUMBER
Longueur	7	25	25	7

Name	Null	Type
-----	-----	-----
ID		NUMBER(7)
LAST_NAME		VARCHAR2(25)
FIRST_NAME		VARCHAR2(25)
DEPT_ID		NUMBER(7)

- Créez la table EMPLOYEES2 en utilisant la structure de la table EMPLOYEES. Inclure uniquement les colonnes EMPLOYEE_ID, FIRST_NAME, LAST_NAME, SALARY et DEPARTMENT_ID. Intitulez les colonnes de la nouvelle table respectivement ID, FIRST_NAME, LAST_NAME, SALARY et DEPT_ID.
- Placez la table EMPLOYEES2 en mode lecture seule.
- Essayez d'insérer la ligne suivante dans la table EMPLOYEES2 :

ID	FIRST_NAME	LAST_NAME	SALARY	DEPT_ID
34	Grant	Marcie	5678	10

Vous obtenez le message d'erreur suivant

```
Error starting at line 1 in command:
INSERT INTO employees2
VALUES (34, 'Grant','Marcie',5678,10)
Error at Command Line:1 Column:12
Error report:
SQL Error: ORA-12081: update operation not allowed on table "ORA16"."EMPLOYEES2"
12081. 00000 - "update operation not allowed on table \"%s\".\"%s\""
*Cause:      An attempt was made to update a read-only materialized view.
*Action:     No action required. Only Oracle is allowed to update a
              read-only materialized view.
```

7. Remettez la table EMPLOYEES2 en mode lecture-écriture. Essayez à nouveau d'insérer la même ligne. Vous devez obtenir les messages suivants :

```
ALTER TABLE employees2 succeeded.
1 rows inserted
```

8. Supprimez la table EMPLOYEES2.

Exercice 11 (1 pts)

1.4 Première partie

1. Le département des ressources humaines souhaite masquer certaines des données de la table EMPLOYEES. Il souhaite obtenir une vue nommée EMPLOYEES_VU, comprenant les ID et les noms d'employé ainsi que les ID de département de la table EMPLOYEES. L'en-tête EMPLOYEE doit être attribué à la colonne contenant les noms d'employé.

2. Vérifiez que la vue fonctionne. Affichez le contenu de la vue EMPLOYEES_VU.

	EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
1	100	King	90
2	101	Kochhar	90
3	102	De Haan	90
4	103	Hunold	60
5	104	Ernst	60

3. A l'aide de la vue EMPLOYEES_VU, écrivez pour le département des ressources humaines une interrogation permettant d'afficher les noms d'employé et les ID de département correspondants.

	EMPLOYEE	DEPARTMENT_ID
1	King	90
2	Kochhar	90
3	De Haan	90
4	Hunold	60
5	Ernst	60

4. Le département 50 a besoin d'accéder aux données concernant ses employés. Créez une vue nommée DEPT50 contenant l'ID d'employé, le nom et l'ID de département de tous les employés du département 50. Vous devez intituler les colonnes de la vue EMPNO, EMPLOYEE et DEPTNO. Pour des raisons de sécurité, n'autorisez pas le transfert d'un employé vers un autre département par l'intermédiaire de la vue.

5. Affichez la structure et le contenu de la vue DEPT50.

Name	Null	Type
EMPNO	NOT NULL	NUMBER(6)
EMPLOYEE	NOT NULL	VARCHAR2(25)
DEPTNO		NUMBER(4)

	EMPNO	EMPLOYEE	DEPTNO
1	124	Mourgos	50
2	141	Rajs	50
3	142	Davies	50
4	143	Matos	50
5	144	Vargas	50

6. Testez la vue. Tentez de transférer Matos vers le département 80.
 Deuxième partie

7. Vous avez besoin d'une séquence pouvant être utilisée avec la colonne PRIMARY KEY de la table DEPT. La séquence doit commencer à 200 et présenter une valeur maximum de 1 000.

La séquence doit croître par incréments de 10. Nommez-la DEPT_ID_SEQ.

8. Pour tester la séquence, écrivez un script permettant d'insérer deux lignes dans la table DEPT.

Nommez le script ex_11_08.sql. Veillez à utiliser la séquence que vous avez créée pour la colonne ID. Ajoutez deux départements : Education et Administration. Vérifiez vos ajouts.

Exécutez les commandes enregistrées dans le script.

9. Créez un index non unique sur la colonne NAME de la table DEPT.

10. Créez un synonyme pour la table EMPLOYEES. Nommez-le EMP.