

Oracle Database 10g : Les fondamentaux du langage SQL (II)

Exercices complémentaires et solutions

D17111FR10

Édition 1.0

Juin 2004

D39296

ORACLE®

Auteur

Priya Vennapusa

**Révisions et
contributions techniques**

Nancy Greenberg
Priya Nathan
Andrew Brannigan
Angelika Krupp
Brian Boxx
Christopher Lawless
Joel Goodman
Malika Marghadi
Marjolein Dekkers
Stefan Grenstad
Zarko Cesljas
Rosita Hanoman
Ruediger Steffan

Editeur

Joseph Fernandez

Copyright © 2004, Oracle. Tous droits réservés.

Cette documentation contient des informations qui sont la propriété d'Oracle Corporation. Fournie sous accord de licence, elle est soumise à des restrictions relatives à l'utilisation et à la publication et est protégée par la loi relative aux droits d'auteur. Il est interdit de rechercher le secret de fabrication du logiciel. Restrictions applicables au gouvernement américain :

If this documentation is delivered to a U.S. Government Agency of the Department of Defense, then it is delivered with Restricted Rights and the following legend is applicable :

Restricted Rights Legend

Use, duplication or disclosure by the Government is subject to restrictions for commercial computer software and shall be deemed to be Restricted Rights software under Federal law, as set forth in subparagraph (c)(1)(ii) of DFARS 252.227-7013, Rights in Technical Data and Computer Software (October 1988).

If this documentation is delivered to a U.S. Government Agency not within the Department of Defense, then it is delivered with "Restricted Rights," as defined in FAR 52.227-14, Rights in Data-General, including Alternate III (June 1987).

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Oracle est une marque déposée, les autres marques Oracle sont des marques d'Oracle Corporation.

Tout autre nom de produit ou de société est utilisé à titre indicatif seulement et peut être une marque de son propriétaire.

Sommaire

Préface

I Introduction

Objectifs 1-2

Objectifs du cours 1-3

Présentation du cours 1-4

Synthèse 1-6

1 Controlling User Access

Objectifs 1-2

Contrôler l'accès des utilisateurs 1-3

Privilèges 1-4

Privilèges système 1-5

Créer des utilisateurs 1-6

Privilèges système utilisateur 1-7

Accorder des privilèges système 1-8

Qu'est-ce qu'un rôle ? 1-9

Créer un rôle et accorder des privilèges à un rôle 1-10

Changer le mot de passe 1-11

Privilèges objet 1-12

Accorder des privilèges objet 1-14

Transmettre des privilèges 1-15

Vérifier les privilèges accordés 1-16

Révoquer des privilèges objet 1-17

Synthèse 1-19

Présentation de l'exercice 1 1-20

2 Gérer les objets de schéma

Objectifs 2-2

L'instruction ALTER TABLE 2-3

Ajouter une colonne 2-5

Modifier une colonne 2-6

Supprimer une colonne 2-7

L'option SET UNUSED 2-8

Ajouter une contrainte 2-10

ON DELETE CASCADE 2-12

Différer des contraintes 2-13

Supprimer une contrainte 2-14

Désactiver des contraintes 2-15

Activer des contraintes 2-16

Contraintes en cascade 2-18

Présentation des index 2-20

Instruction `CREATE INDEX` avec `CREATE TABLE` 2-21

Index basés sur une fonction 2-23

Supprimer un index 2-25

`DROP TABLE ...PURGE` 2-26

L'instruction `FLASHBACK TABLE` 2-27

Tables externes 2-29

Créer un répertoire pour la table externe 2-31

Créer une table externe 2-33

Créer une table externe avec `ORACLE_LOADER` 2-35

Interroger des tables externes 2-37

Synthèse 2-38

Présentation de l'exercice 2 2-39

3 Manipuler des jeux de données volumineux

Objectifs 3-2

Utiliser des sous-interrogations pour manipuler les données 3-3

Copier des lignes à partir d'une autre table 3-4

Insérer des données en utilisant une sous-interrogation comme cible 3-5

Extraire des données avec une sous-interrogation comme source 3-7

Mettre à jour deux colonnes avec une sous-interrogation 3-8

Mettre à jour des lignes en fonction d'une autre table 3-9

Supprimer des lignes en fonction d'une autre table 3-10

Utiliser le mot-clé `WITH CHECK OPTION` dans les instructions LMD 3-11

Présentation de la fonction de valeur par défaut explicite 3-12

Utiliser des valeurs par défaut explicites 3-13

Présentation des instructions `INSERT` multitable 3-14

Types d'instruction `INSERT` multitable 3-16

Instructions `INSERT` multitable 3-17

Instruction `INSERT ALL` inconditionnelle 3-19

Instruction `INSERT ALL` conditionnelle 3-20

Instruction `INSERT FIRST` conditionnelle 3-22

Instruction `INSERT` de transformation 3-24

L'instruction `MERGE` 3-27

Syntaxe de l'instruction `MERGE` 3-28

Fusionner des lignes 3-29

Effectuer le suivi des modifications apportées aux données 3-31

Exemple d'utilisation de la fonctionnalité Flashback Version Query 3-33

La clause `VERSIONS BETWEEN` 3-35

Synthèse 3-36

Présentation de l'exercice 3 3-37

4 Générer des états en regroupant des données liées

Objectifs 4-2

Révision des fonctions de groupe 4-3

Révision de la clause GROUP BY 4-5
Révision de la clause HAVING 4-6
Fonction GROUP BY avec les opérateurs ROLLUP et CUBE 4-7
Opérateur ROLLUP 4-8
Opérateur ROLLUP : exemple 4-9
Opérateur CUBE 4-10
Opérateur CUBE : exemple 4-12
Fonction GROUPING 4-13
Fonction GROUPING : exemple 4-15
GROUPING SETS 4-16
GROUPING SETS : exemple 4-18
Colonnes composites 4-20
Colonnes composites : exemple 4-22
Regroupements concaténés 4-24
Regroupements concaténés : exemple 4-25
Synthèse 4-26
Practice 4: Overview 4-27

5 Gérer les données de différents fuseaux horaires

Objectifs 5-2
Fuseaux horaires 5-3
Paramètre de session TIME_ZONE 5-4
CURRENT_DATE, CURRENT_TIMESTAMP et LOCALTIMESTAMP 5-5
CURRENT_DATE 5-6
CURRENT_TIMESTAMP 5-7
LOCALTIMESTAMP 5-8
DBTIMEZONE et SESSIONTIMEZONE 5-9
Type de données TIMESTAMP 5-10
Types de données TIMESTAMP 5-11
Champs TIMESTAMP 5-12
Différence entre DATE et TIMESTAMP 5-13
Type de données TIMESTAMP WITH TIMEZONE 5-14
TIMESTAMP WITH TIMEZONE : exemple 5-15
TIMESTAMP WITH LOCAL TIMEZONE 5-16
TIMESTAMP WITH LOCAL TIMEZONE : exemple 5-17
Types de données INTERVAL 5-18
Champs INTERVAL 5-20
Type de données INTERVAL YEAR TO MONTH 5-21
INTERVAL YEAR TO MONTH : exemple 5-22
Type de données INTERVAL DAY TO SECOND 5-23
Type de données INTERVAL DAY TO SECOND : exemple 5-24
EXTRACT 5-25

TZ_OFFSET 5-26

Conversion TIMESTAMP avec FROM_TZ 5-28

Convertir en TIMESTAMP avec TO_TIMESTAMP et TO_TIMESTAMP_TZ 5-29

Conversion d'intervalle temporel avec TO_YMINTERVAL 5-30

Utiliser TO_DSINTERVAL : exemple 5-31

Heure d'été 5-32

Synthèse 5-34

Présentation de l'exercice 5 5-35

6 Extraire des données à l'aide de sous-interrogations

Objectifs 6-2

Sous-interrogations multicolonne 6-3

Comparaisons de colonnes 6-4

Sous-interrogation avec comparaison par groupe de colonnes 6-5

Sous-interrogation avec comparaison colonne par colonne 6-6

Expressions de sous-interrogation scalaire 6-7

Sous-interrogations scalaires : exemples 6-9

Sous-interrogations corrélées 6-11

Sous-interrogations corrélées 6-12

Utiliser des sous-interrogations corrélées 6-13

Utiliser l'opérateur EXISTS 6-16

Rechercher les employés qui sont manager d'au moins un employé 6-17

Rechercher tous les départements qui ne comportent aucun employé 6-18

Opération UPDATE corrélée 6-19

Effectuer une opération UPDATE corrélée 6-20

Opération DELETE corrélée 6-22

Effectuer une opération DELETE corrélée 6-23

La clause WITH 6-24

Clause WITH : exemple 6-25

Synthèse 6-27

Présentation de l'exercice 6 6-29

7 Extraction hiérarchique

Objectifs 7-2

Echantillon de données de la table EMPLOYEES 7-3

Arborescence naturelle 7-4

Interrogations hiérarchiques 7-5

Parcourir l'arborescence 7-6

Parcourir l'arborescence : de bas en haut 7-8

Parcourir l'arborescence : de haut en bas 7-9

Classer des lignes avec la pseudo-colonne LEVEL 7-10

Formater des états hiérarchiques avec LEVEL et LPAD 7-12

Éliminer des branches 7-14

Synthèse 7-15

Présentation de l'exercice 7 7-16

8 Prise en charge des expressions régulières

- Objectifs 8-2
- Présentation des expressions régulières 8-3
- Métacaractères 8-4
- Utiliser des métacaractères 8-5
- Fonctions d'expression régulière 8-7
- Syntaxe de la fonction REGEXP 8-8
- Effectuer des recherches élémentaires 8-9
- Déterminer la présence d'un modèle 8-10
- Exemple d'extraction de sous-chaînes 8-11
- Remplacer des modèles 8-12
- Expressions régulières et contraintes CHECK 8-13
- Synthèse 8-14
- Présentation de l'exercice 8 8-15

Annexe A : Solutions des exercices

Annexe B : Description et données des tables

Annexe C : Ecrire des scripts avancés

- Objectifs C-2
- Utiliser du code SQL pour générer des instructions SQL C-3
- Créer un script de base C-4
- Contrôler l'environnement C-5
- Vue d'ensemble C-6
- Transférer le contenu d'une table dans un fichier C-7
- Générer un prédicat dynamique C-9
- Synthèse C-11

Annexe D : Composants de l'architecture Oracle

- Objectifs D-2
- Architecture de la base de données Oracle : présentation D-3
- Architecture physique de la base de données D-4
- Fichiers de contrôle D-5
- Fichiers de journalisation D-6
- Tablespaces et fichiers de données D-7
- Segments, extents et blocs D-8
- Gestion des instances Oracle D-9
- Structures mémoire Oracle D-10
- Processus Oracle D-12
- Autres structures physiques essentielles D-13
- Traitement d'une instruction SQL D-14

Connexion à une instance D-15
Traitement d'une interrogation D-17
Zone de mémoire partagée D-18
Cache de tampons de la base D-20
Mémoire PGA D-21
Traiter une instruction LMD D-22
Tampon de journalisation D-24
Segment d'annulation D-25
Traitement des opérations COMMIT D-26
Synthèse D-28

Index

Exercices complémentaires

Solutions des exercices complémentaires

Alioune DIOP (aadiop@hbo-technology.com) has a non-transferable license to use this Student Guide.

Exercices complémentaires

Alioune DIOP (aadiop@hbo-technology.com) has a non-transferable
license to use this Student Guide.

Exercices complémentaires

Les exercices complémentaires suivants peuvent être effectués après l'étude des instructions LMD et LDD dans les chapitres 2 et 3.

Remarque : Exécutez les scripts `lab_ap_cre_special_sal.sql`, `lab_ap_cre_sal_history.sql` et `lab_ap_cre_mgr_history.sql` du dossier labs pour créer les tables `SPECIAL_SAL`, `SAL_HISTORY` et `MGR_HISTORY`.

1. Le département Human Resources souhaite obtenir la liste des employés sous-payés (sur la base d'une enquête réalisée), l'historique du salaire des employés, ainsi que l'historique du salaire des managers. Il vous a donc été demandé d'effectuer les opérations suivantes :

Ecrivez une instruction pour :

- Extraire l'ID, la date d'embauche, le salaire et l'ID du manager des employés dont l'ID est supérieur ou égal à 200 dans la table `EMPLOYEES`.
- Si le salaire est inférieur à 5 000 \$, insérez l'ID et le salaire de l'employé dans la table `SPECIAL_SAL`.
- Insérer l'ID, la date d'embauche et le salaire des employés dans la table `SAL_HISTORY`.
- Insérer l'ID, l'ID du manager et le salaire des employés dans la table `MGR_HISTORY`.

2. Interrogez les tables `SPECIAL_SAL`, `SAL_HISTORY` et `MGR_HISTORY` afin d'afficher les enregistrements insérés.

SPECIAL_SAL

EMPLOYEE_ID	SALARY
200	4400

SALARY_HISTORY

EMPLOYEE_ID	HIRE_DATE	SALARY
201	17-FEB-96	13000
202	17-AUG-97	6000
203	07-JUN-94	6500
204	07-JUN-94	10000
205	07-JUN-94	12000
206	07-JUN-94	8300

6 rows selected.

Exercices complémentaires

MGR_HISTORY

EMPLOYEE_ID	MANAGER_ID	SALARY
201	100	13000
202	201	6000
203	101	6500
204	101	10000
205	101	12000
206	205	8300

6 rows selected.

3. Le DBA vous demande de créer une table comportant une contrainte de clé primaire, mais il souhaite que le nom de l'index soit différent du nom de la contrainte. Créez la table LOCATIONS_NAMED_INDEX à partir des informations ci-dessous. Nommez LOCATIONS_PK_IDX l'index de la colonne PRIMARY KEY.

Column Name	Deptno	Dname
Primary Key	Yes	
Data Type	Number	VARCHAR2
Length	4	30

4. Interrogez la table USER_INDEXES afin d'afficher le nom INDEX_NAME pour la table LOCATIONS_NAMED_INDEX.

INDEX_NAME	TABLE_NAME
LOCATIONS_PK_IDX	LOCATIONS_NAMED_INDEX

Exercices complémentaires

Les exercices complémentaires suivants peuvent être effectués après l'étude des améliorations de la clause GROUP BY.

5. Le département Human Resources a besoin d'états sur certains départements. Les besoins sont les suivants :

Ecrivez une interrogation pour afficher les informations suivantes pour les départements dont l'ID est supérieur à 80 :

- Le salaire total de chaque poste d'un département
- Le salaire total
- Le salaire total des villes dans lesquelles se trouvent les départements
- Le salaire total de chaque poste, quel que soit le département
- Le salaire total de chaque département, quelle que soit la ville
- Le salaire total des villes dans lesquelles se trouvent les départements
- Le salaire total des départements, quels que soient le poste et la ville

CITY	DNAME	JOB	SUM(E.SALARY)
			129900
		AD_VP	34000
		AC_MGR	12000
		FI_MGR	12000
		AD_PRES	24000
		AC_ACCOUNT	8300
		FI_ACCOUNT	39600
	Finance		51600
	Finance	FI_MGR	12000
	Finance	FI_ACCOUNT	39600
...	Executive		58000
CITY	DNAME	JOB	SUM(E.SALARY)
Seattle	Finance	FI_MGR	12000
Seattle	Finance	FI_ACCOUNT	39600
Seattle	Executive		58000
Seattle	Executive	AD_VP	34000
Seattle	Executive	AD_PRES	24000
Seattle	Accounting		20300
Seattle	Accounting	AC_MGR	12000
Seattle	Accounting	AC_ACCOUNT	8300

32 rows selected.

Exercices complémentaires

6. Le département Accounting nécessite l'analyse du salaire maximum et du salaire minimum par département, poste et manager. Il vous a donc été demandé d'effectuer les opérations suivantes :

Ecrivez une interrogation pour afficher les regroupements suivants :

- ID de département, ID de poste
- ID de poste, ID de manager

L'interrogation doit calculer le salaire maximum et le salaire minimum de chacun de ces groupes.

DEPARTMENT_ID	JOB	MANAGER_ID	MAX(SALARY)	MIN(SALARY)
	AD_VP	100	17000	17000
	AC_MGR	101	12000	12000
	FI_MGR	101	12000	12000
	HR_REP	101	6500	6500
	MK_MAN	100	13000	13000
	MK_REP	201	6000	6000
	PR_REP	101	10000	10000
	PU_MAN	100	11000	11000
	SA_MAN	100	14000	10500
	SA_REP	145	10000	7000
	SA_REP	146	10000	7000
	SA_REP	147	10500	6200
	SA_REP	148	11500	6100

DEPARTMENT_ID	JOB	MANAGER_ID	MAX(SALARY)	MIN(SALARY)
100	FI_MGR		12000	12000
100	FI_ACCOUNT		9000	6900
110	AC_MGR		12000	12000
110	AC_ACCOUNT		8300	8300

52 rows selected.

Exercices complémentaires

Les exercices complémentaires suivants peuvent être effectués après l'étude des fonctions date-heure.

Vous travaillez dans une entreprise internationale et le nouveau vice-président souhaite connaître les différents fuseaux horaires de toutes les succursales de l'entreprise. Il vous a demandé les informations suivantes :

7. Modifiez la session afin d'affecter au paramètre `NLS_DATE_FORMAT` la valeur `DD-MON-YYYY HH24:MI:SS`.
8. a. Ecrivez des interrogations afin d'afficher le décalage horaire (`TZ_OFFSET`) pour les fuseaux horaires suivants :
Australia/Sydney

<code>TZ_OFFSET('AUSTRALIA/</code>
<code>+11:00</code>

Chile/Easter Island

<code>TZ_OFFSET('CHILE/EAST</code>
<code>-05:00</code>

- b. Modifiez la session afin d'affecter au paramètre `TIME_ZONE` le décalage horaire de la zone Australia/Sydney.
- c. Affichez les valeurs `SYSDATE`, `CURRENT_DATE`, `CURRENT_TIMESTAMP` et `LOCALTIMESTAMP` de cette session.

Remarque : La sortie peut être différente, en fonction de la date à laquelle la commande est exécutée.

<code>SYSDATE</code>	<code>CURRENT_DATE</code>	<code>CURRENT_TIMESTAMP</code>	<code>LOCALTIMESTAMP</code>
19-FEB-2004 09:32:44	20-FEB-2004 02:32:44	20-FEB-04 02.32.44.466163 AM +10:00	20-FEB-04 02.32.44.466163 AM

- d. Modifiez la session afin d'affecter au paramètre `TIME_ZONE` le décalage horaire de la zone Chile/Easter Island.

Remarque : Les résultats de la question précédente sont basés sur une date différente ; dans certains cas, ils ne correspondront pas aux résultats réels obtenus par les stagiaires. En outre, le décalage horaire des différents pays peut varier en fonction de l'heure d'été.

- e. Affichez les valeurs `SYSDATE`, `CURRENT_DATE`, `CURRENT_TIMESTAMP` et `LOCALTIMESTAMP` de cette session.

Remarque : La sortie peut être différente, en fonction de la date à laquelle la commande est exécutée.

<code>SYSDATE</code>	<code>CURRENT_DATE</code>	<code>CURRENT_TIMESTAMP</code>	<code>LOCALTIMESTAMP</code>
19-FEB-2004 09:33:37	19-FEB-2004 10:33:38	19-FEB-04 10.33.37.906944 AM - 06:00	19-FEB-04 10.33.37.906944 AM

Exercices complémentaires

- f. Modifiez la session afin d'affecter au paramètre NLS_DATE_FORMAT la valeur DD-MON-YYYY.

Remarque

- Notez que, dans la question précédente, les valeurs CURRENT_DATE, CURRENT_TIMESTAMP et LOCALTIMESTAMP dépendent toutes du fuseau horaire de la session. En revanche, SYSDATE ne dépend pas du fuseau horaire de la session.
- Les résultats de la question précédente sont basés sur une date différente ; dans certains cas, ils ne correspondront pas aux résultats réels obtenus par les stagiaires. En outre, le décalage horaire des différents pays peut varier en fonction de l'heure d'été.

9. Le département Human Resources souhaite obtenir la liste des employés qui seront évalués en janvier ; il vous a donc été demandé d'effectuer les opérations suivantes :
Ecrivez une interrogation permettant d'afficher le nom, le mois d'embauche et la date d'embauche des employés qui ont été embauchés en janvier, quelle que soit l'année.

LAST_NAME	EXTRACT(MONTHFROMHIRE_DATE)	HIRE_DATE
De Haan	1	13-JAN-1993
Hunold	1	03-JAN-1990
Landry	1	14-JAN-1999
Davies	1	29-JAN-1997
Partners	1	05-JAN-1997
Zlotkey	1	29-JAN-2000
Tucker	1	30-JAN-1997
King	1	30-JAN-1996
Marvins	1	24-JAN-2000
Fox	1	24-JAN-1998
Johnson	1	04-JAN-2000
Taylor	1	24-JAN-1998
Sarchand	1	27-JAN-1996
Grant	1	13-JAN-2000

14 rows selected.

Exercices complémentaires

Les exercices complémentaires suivants peuvent être effectués après l'étude des sous-interrogations avancées.

10. Le PDG a besoin d'un état sur les trois plus gros salaires de l'entreprise, pour l'intéressement aux bénéfices. Il vous a demandé de lui fournir une liste.

Ecrivez une interrogation permettant d'afficher les trois plus gros salaires dans la table EMPLOYEES. Affichez leur nom et leur salaire.

LAST_NAME	SALARY
King	24000
Kochhar	17000
De Haan	17000

11. Les charges de l'état de Californie ont changé suite à une ordonnance locale. Le responsable des payes vous a demandé de lui fournir la liste des personnes concernées. Ecrivez une interrogation permettant d'afficher l'ID et le nom des employés qui travaillent dans l'état de Californie.

Indice : Utilisez des sous-interrogations scalaires.

EMPLOYEE_ID	LAST_NAME
120	Weiss
121	Fripp
122	Kauffling
123	Vollman
124	Mourgos
125	Nayer
126	Mikkilineni
127	Landry
128	Markle
129	Bissot
...	
190	Gates
191	Perkins
192	Bell
193	Everett
194	McCain
195	Jones
196	Walsh
197	Feeney
198	OConnell
199	Grant

45 rows selected.

Exercices complémentaires

12. Le DBA souhaite supprimer les anciennes informations de la base de données. Il estime que les enregistrements des postes précédents sont inutiles. Il vous a donc demandé d'effectuer les opérations suivantes :

Ecrivez une interrogation permettant de supprimer la ligne JOB_HISTORY la plus ancienne d'un employé en recherchant dans la table JOB_HISTORY la valeur MIN(START_DATE) de l'employé. Supprimez les enregistrements des *seuls* employés ayant changé au moins deux fois de poste.

Indice : Utilisez une commande DELETE corrélée.

13. Le vice-président du département Human Resources a besoin des enregistrements complets des postes occupés par les employés, pour son allocation annuelle. Il vous appelle afin de vous demander de ne pas effectuer les opérations souhaitées par le DBA. Annulez la transaction.
14. La situation économique contraint la direction à réduire les coûts. Le PDG souhaite revoir à la baisse les salaires les plus importants de l'entreprise. Il vous a demandé une liste basée sur les éléments suivants :
- Ecrivez une interrogation permettant d'afficher l'ID des postes dont le salaire maximum se situe au-delà de la moitié du salaire maximum de l'entreprise. Utilisez la clause WITH pour écrire cette interrogation. Nommez l'interrogation MAX_SAL_CALC.

JOB_TITLE	JOB_TOTAL
President	24000
Administration Vice President	17000
Sales Manager	14000
Marketing Manager	13000

Les exercices complémentaires suivants peuvent être effectués après l'étude de l'extraction hiérarchique.

15. Lex De Haan quitte l'entreprise. Son remplaçant souhaite des états concernant les employés directement sous sa responsabilité.
- Ecrivez une instruction SQL permettant d'afficher le numéro, le nom, la date d'embauche et le salaire des employés, indiquant :
- a. Les employés directement sous la responsabilité de De Haan

EMPLOYEE_ID	LAST_NAME	HIRE_DATE	SALARY
103	Hunold	03-JAN-1990	9000

- b. L'organigramme sous De Haan (numéro d'employé 102)

EMPLOYEE_ID	LAST_NAME	HIRE_DATE	SALARY
103	Hunold	03-JAN-1990	9000
104	Ernst	21-MAY-1991	6000
105	Austin	25-JUN-1997	4800
106	Pataballa	05-FEB-1998	4800
107	Lorentz	07-FEB-1999	4200

Exercices complémentaires

16. Ecrivez une interrogation hiérarchique afin d'afficher le numéro, le numéro de manager et le nom de tous les employés situés deux niveaux sous De Haan (numéro d'employé 102). Affichez également le niveau de chaque employé.

EMPLOYEE_ID	MANAGER_ID	LEVEL	LAST_NAME
104	103	3	Ernst
105	103	3	Austin
106	103	3	Pataballa
107	103	3	Lorentz

17. Le PDG souhaite un état hiérarchique concernant tous les employés. Ses besoins sont les suivants :

Générez un état hiérarchique afin d'afficher le numéro, le numéro de manager, la pseudo-colonne LEVEL et le nom des employés. Pour chaque ligne de la table EMPLOYEES, vous devez afficher une structure arborescente qui indique l'employé, son manager, le manager de son manager, etc. Utilisez des indentations pour la colonne NAME.

EMPLOYEE_ID	MANAGER_ID	LEVEL	LAST_NAME
100		1	King
101	100	1	Kochhar
100		2	__King
102	100	1	De Haan
100		2	__King
103	102	1	Hunold
102	100	2	__De Haan
100		3	___King
...			
EMPLOYEE_ID	MANAGER_ID	LEVEL	LAST_NAME
205	101	2	__Higgins
101	100	3	___Kochhar
100		4	____King

315 rows selected.

Remarque : Le résultat illustré n'est qu'un exemple. Toutes les lignes du résultat réel ne sont pas incluses.

Solutions des exercices complémentaires

Alioune DIOP (aadiop@hbo-technology.com) has a non-transferable
license to use this Student Guide.

Solutions des exercices complémentaires

Les exercices complémentaires suivants peuvent être effectués après l'étude des instructions LMD et LDD dans les chapitres 2 et 3.

Remarque : Exécutez les scripts `lab_ap_cre_special_sal.sql`, `lab_ap_cre_sal_history.sql` et `lab_ap_cre_mgr_history.sql` du dossier labs pour créer les tables `SPECIAL_SAL`, `SAL_HISTORY` et `MGR_HISTORY`.

1. Le département Human Resources souhaite obtenir la liste des employés sous-payés (sur la base d'une enquête réalisée), l'historique du salaire des employés, ainsi que l'historique du salaire des managers. Il vous a donc été demandé d'effectuer les opérations suivantes :

Ecrivez une instruction pour :

- Extraire l'ID, la date d'embauche, le salaire et l'ID du manager des employés dont l'ID est supérieur ou égal à 200 dans la table `EMPLOYEES`.
- Si le salaire est inférieur à 5 000 \$, insérez l'ID et le salaire de l'employé dans la table `SPECIAL_SAL`.
- Insérer l'ID, la date d'embauche et le salaire des employés dans la table `SAL_HISTORY`.
- Insérer l'ID, l'ID du manager et le salaire des employés dans la table `MGR_HISTORY`.

```
INSERT ALL
WHEN SAL < 5000 THEN
  INTO special_sal VALUES (EMPID, SAL)
ELSE
  INTO sal_history VALUES (EMPID, HIREDATE, SAL)
  INTO mgr_history VALUES (EMPID, MGR, SAL)
SELECT employee_id EMPID, hire_date HIREDATE,
       salary SAL, manager_id MGR
FROM employees
WHERE employee_id >= 200;
```

2. Interrogez les tables `SPECIAL_SAL`, `SAL_HISTORY` et `MGR_HISTORY` afin d'afficher les enregistrements insérés.

```
SELECT * FROM special_sal;
SELECT * FROM sal_history;
SELECT * FROM mgr_history;
```

3. Le DBA vous demande de créer une table comportant une contrainte de clé primaire, mais il souhaite que le nom de l'index soit différent du nom de la contrainte. Créez la table `LOCATIONS_NAMED_INDEX` à partir des informations ci-dessous. Nommez `LOCATIONS_PK_IDX` l'index de la colonne `PRIMARY KEY`.

Solutions des exercices complémentaires

Column Name	Deptno	Dname
Primary Key	Yes	
Data Type	Number	VARCHAR2
Length	4	30

```
CREATE TABLE LOCATIONS_NAMED_INDEX  
(location_id NUMBER(4) PRIMARY KEY USING INDEX  
(CREATE INDEX locations_pk_idx ON  
LOCATIONS_NAMED_INDEX(location_id)),  
location_name VARCHAR2(20));
```

4. Interrogez la table USER_INDEXES afin d'afficher le nom INDEX_NAME pour la table LOCATIONS_NAMED_INDEX.

```
SELECT INDEX_NAME, TABLE_NAME  
FROM USER_INDEXES  
WHERE TABLE_NAME = 'LOCATIONS_NAMED_INDEX';
```

Les exercices complémentaires suivants peuvent être effectués après l'étude des améliorations de la clause GROUP BY.

5. Le département Human Resources a besoin d'états sur certains départements. Les besoins sont les suivants :

Ecrivez une interrogation pour afficher les informations suivantes pour les départements dont l'ID est supérieur à 80 :

- Le salaire total de chaque poste d'un département
- Le salaire total
- Le salaire total des villes dans lesquelles se trouvent les départements
- Le salaire total de chaque poste, quel que soit le département
- Le salaire total de chaque département, quelle que soit la ville
- Le salaire total des villes dans lesquelles se trouvent les départements
- Le salaire total des départements, quels que soient le poste et la ville

Solutions des exercices complémentaires

```
COLUMN      city FORMAT A25 Heading CITY
COLUMN      department_name FORMAT A15 Heading DNAME
COLUMN      job_id FORMAT A10 Heading JOB
COLUMN      SUM(salary) FORMAT $99,99,999.00 Heading
              SUM(SALARY)

SELECT      l.city,d.department_name, e.job_id,
              SUM(e.salary)
FROM        locations l, employees e, departments d
WHERE       d.location_id=l.location_id
AND         e.department_id = d.department_id
AND         e.department_id > 80
GROUP BY CUBE( l.city, d.department_name, e.job_id);
```

6. Le département Accounting nécessite l'analyse du salaire maximum et du salaire minimum par département, poste et manager. Il vous a donc été demandé d'effectuer les opérations suivantes :

Ecrivez une interrogation pour afficher les regroupements suivants :

- ID de département, ID de poste
- ID de poste, ID de manager

L'interrogation doit calculer le salaire maximum et le salaire minimum de chacun de ces groupes.

```
SELECT
  department_id,job_id,manager_id,max(salary),
  min(salary)
FROM   employees
GROUP BY GROUPING SETS
  ((department_id,job_id), (job_id,manager_id));
```

Les exercices complémentaires suivants peuvent être effectués après l'étude des fonctions date-heure.

Vous travaillez dans une entreprise internationale et le nouveau vice-président souhaite connaître les différents fuseaux horaires de toutes les succursales de l'entreprise. Il vous a demandé les informations suivantes :

7. Modifiez la session afin d'affecter au paramètre NLS_DATE_FORMAT la valeur DD-MON-YYYY HH24:MI:SS.

```
ALTER SESSION
SET NLS_DATE_FORMAT = 'DD-MON-YYYY HH24:MI:SS';
```

Solutions des exercices complémentaires

8. a. Ecrivez des interrogations afin d'afficher le décalage horaire (TZ_OFFSET) pour les fuseaux horaires suivants :

- Australia/Sydney

```
SELECT TZ_OFFSET ('Australia/Sydney') from dual;
```

- Chile/Easter Island

```
SELECT TZ_OFFSET ('Chile/EasterIsland') from dual;
```

- b. Modifiez la session afin d'affecter au paramètre TIME_ZONE le décalage horaire de la zone Australia/Sydney.

```
ALTER SESSION SET TIME_ZONE = '+10:00';
```

- c. Affichez les valeurs SYSDATE, CURRENT_DATE, CURRENT_TIMESTAMP et LOCALTIMESTAMP de cette session.

Remarque : La sortie peut être différente, en fonction de la date à laquelle la commande est exécutée.

```
SELECT SYSDATE, CURRENT_DATE, CURRENT_TIMESTAMP,  
LOCALTIMESTAMP FROM DUAL;
```

- d. Modifiez la session afin d'affecter au paramètre TIME_ZONE le décalage horaire de la zone Chile/Easter Island.

Remarque : Les résultats de la question précédente sont basés sur une date différente ; dans certains cas, ils ne correspondront pas aux résultats réels obtenus par les stagiaires. En outre, le décalage horaire des différents pays peut varier en fonction de l'heure d'été.

```
ALTER SESSION SET TIME_ZONE = '-06:00';
```

- e. Affichez les valeurs SYSDATE, CURRENT_DATE, CURRENT_TIMESTAMP et LOCALTIMESTAMP de cette session.

Remarque : La sortie peut être différente, en fonction de la date à laquelle la commande est exécutée.

```
SELECT SYSDATE, CURRENT_DATE, CURRENT_TIMESTAMP,  
LOCALTIMESTAMP FROM DUAL;
```

- f. Modifiez la session afin d'affecter au paramètre NLS_DATE_FORMAT la valeur DD-MON-YYYY.

```
ALTER SESSION SET NLS_DATE_FORMAT = 'DD-MON-YYYY';
```

Solutions des exercices complémentaires

Remarque

- Notez que, dans la question précédente, les valeurs CURRENT_DATE, CURRENT_TIMESTAMP et LOCALTIMESTAMP dépendent toutes du fuseau horaire de la session. En revanche, SYSDATE ne dépend pas du fuseau horaire de la session.
 - Les résultats de la question précédente sont basés sur une date différente ; dans certains cas, ils ne correspondront pas aux résultats réels obtenus par les stagiaires. En outre, le décalage horaire des différents pays peut varier en fonction de l'heure d'été.
9. Le département Human Resources souhaite obtenir la liste des employés qui seront évalués en janvier ; il vous a donc été demandé d'effectuer les opérations suivantes :
Ecrivez une interrogation permettant d'afficher le nom, le mois d'embauche et la date d'embauche des employés qui ont été embauchés en janvier, quelle que soit l'année.

```
SELECT last_name, EXTRACT (MONTH FROM HIRE_DATE),  
HIRE_DATE FROM employees  
WHERE EXTRACT (MONTH FROM HIRE_DATE) = 1;
```

Les exercices complémentaires suivants peuvent être effectués après l'étude des sous-interrogations avancées.

10. Le PDG a besoin d'un état sur les trois plus gros salaires de l'entreprise, pour l'intéressement aux bénéfices. Il vous a demandé de lui fournir une liste.
Ecrivez une interrogation permettant d'afficher les trois plus gros salaires dans la table EMPLOYEES. Affichez leur nom et leur salaire.

```
SELECT last_name, salary  
FROM employees e  
WHERE 3 > (SELECT COUNT (*)  
FROM employees  
WHERE e.salary < salary);
```

11. Les charges de l'état de Californie ont changé suite à une ordonnance locale. Le responsable des payes vous a demandé de lui fournir la liste des personnes concernées.
Ecrivez une interrogation permettant d'afficher l'ID et le nom des employés qui travaillent dans l'état de Californie.

Indice : Utilisez des sous-interrogations scalaires.

Solutions des exercices complémentaires

```
SELECT employee_id, last_name
FROM employees e
WHERE ((SELECT location_id
        FROM departments d
        WHERE e.department_id = d.department_id)
        IN (SELECT location_id
            FROM locations l
            WHERE state_province = 'California'));
```

12. Le DBA souhaite supprimer les anciennes informations de la base de données. Il estime que les enregistrements des postes précédents sont inutiles. Il vous a donc demandé d'effectuer les opérations suivantes :

Ecrivez une interrogation permettant de supprimer la ligne JOB_HISTORY la plus ancienne d'un employé en recherchant dans la table JOB_HISTORY la valeur MIN(START_DATE) de l'employé. Supprimez les enregistrements des *seuls* employés ayant changé au moins deux fois de poste.

Indice : Utilisez une commande DELETE corrélée.

```
DELETE FROM job_history JH
WHERE employee_id = (SELECT employee_id
                    FROM employees E
                    WHERE JH.employee_id = E.employee_id
                    AND START_DATE = (SELECT MIN(start_date)
                                      FROM job_history JH
                                      WHERE JH.employee_id = E.employee_id)
                    AND 3 > (SELECT COUNT(*)
                              FROM job_history JH
                              WHERE JH.employee_id = E.employee_id
                              GROUP BY EMPLOYEE_ID
                              HAVING COUNT(*) >= 2));
```

13. Le vice-président du département Human Resources a besoin des enregistrements complets des postes occupés par les employés, pour son allocation annuelle. Il vous appelle afin de vous demander de ne pas effectuer les opérations souhaitées par le DBA. Annulez la transaction.

```
ROLLBACK;
```

Solutions des exercices complémentaires

14. La situation économique contraint la direction à réduire les coûts. Le PDG souhaite revoir à la baisse les salaires les plus importants de l'entreprise. Il vous a demandé une liste basée sur les éléments suivants :

Ecrivez une interrogation permettant d'afficher l'ID des postes dont le salaire maximum de situe au-delà de la moitié du salaire maximum de l'entreprise. Utilisez la clause WITH pour écrire cette interrogation. Nommez l'interrogation MAX_SAL_CALC.

```
WITH
MAX_SAL_CALC AS (SELECT job_title, MAX(salary) AS
job_total
FROM employees, jobs
WHERE employees.job_id = jobs.job_id
GROUP BY job_title)
SELECT job_title, job_total
FROM MAX_SAL_CALC
WHERE job_total > (SELECT MAX(job_total) * 1/2
FROM MAX_SAL_CALC)
ORDER BY job_total DESC;
```

Les exercices complémentaires suivants peuvent être effectués après l'étude de l'extraction hiérarchique.

15. Lex De Haan quitte l'entreprise. Son remplaçant souhaite des états concernant les employés directement sous sa responsabilité.

Ecrivez une instruction SQL permettant d'afficher le numéro, le nom, la date d'embauche et le salaire des employés, indiquant :

- a. Les employés directement sous la responsabilité de De Haan

```
SELECT employee_id, last_name, hire_date, salary
FROM employees
WHERE manager_id = (SELECT employee_id
FROM employees
WHERE last_name = 'De Haan');
```

- b. L'organigramme sous De Haan (numéro d'employé 102)

```
SELECT employee_id, last_name, hire_date, salary
FROM employees
WHERE employee_id != 102
CONNECT BY manager_id = PRIOR employee_id
START WITH employee_id = 102;
```

Solutions des exercices complémentaires

16. Ecrivez une interrogation hiérarchique afin d'afficher le numéro, le numéro de manager et le nom de tous les employés situés deux niveaux sous De Haan (numéro d'employé 102). Affichez également le niveau de chaque employé.

```
SELECT employee_id, manager_id, level, last_name
FROM employees
WHERE LEVEL = 3
CONNECT BY manager_id = PRIOR employee_id
START WITH employee_id = 102;
```

17. Le PDG souhaite un état hiérarchique concernant tous les employés. Ses besoins sont les suivants :

Générez un état hiérarchique afin d'afficher le numéro, le numéro de manager, la pseudo-colonne LEVEL et le nom des employés. Pour chaque ligne de la table EMPLOYEES, vous devez afficher une structure arborescente qui indique l'employé, son manager, le manager de son manager, etc. Utilisez des indentations pour la colonne NAME.

```
COLUMN name FORMAT A25
SELECT employee_id, manager_id, LEVEL,
LPAD(last_name, LENGTH(last_name) + (LEVEL*2) - 2, '_ ')
LAST_NAME
FROM employees
CONNECT BY employee_id = PRIOR manager_id;
COLUMN name CLEAR
```

Alioune DIOP (aadiop@hbo-technology.com) has a non-transferable
license to use this Student Guide.