

Travaux Pratiques 1-2 – Etude du langage SQL

Présentation

Dans ces exercices, vous allez :

- + Utilisez l'environnement **APEX** sur le site <http://apex.oracle.com> pour examiner des objets de base de données ou **Oracle SQL Developer** dans le compte **HR** (déverrouillé préalablement). de votre base de données locale.
- + Tous les exercices écrits utilisent les objets de base de données sur le site **APEX** avec un navigateur Web
- + Vous pouvez aussi utiliser l'environnement de développement **Oracle SQL Developer**. Dans ce cas de figure, il vous faut également installer **Oracle Database XE 21c**. Vous pouvez aussi utiliser **SQL*Plus** en mode ligne de commandes avec la base de données que vous aurez installé en local sur votre machine.
- + **Important** : utilisez le « **Guide Exercices SQL** » qui vous a été fourni.

Avant de commencer les exercices, notez les points suivants :

- + Si vous utilisez le site <http://apex.oracle.com> vous devez charger préalablement le script `load_sample.sql` qui contient les tables et objets de démonstration qui sont utilisés dans les 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 ou dans le script `load_sample.sql`, cette dernière ayant été créée plus récemment.

Première partie

1.1 Exercice 1

Avec **APEX** :

1. Connectez-vous dans votre Workspace.
2. Cliquez sur l'icône « **SQL Workshop** » puis ouvrez « **Object Browser** »
3. Dans le cadre de navigation vertical de gauche vous pouvez consulter les tables du schéma de base de données qui ont été créées en chargeant le script `load_sample.sql` :

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

REGIONS

Si la table JOB_GRADES n'existe pas vous pouvez la charger avec le script qui vous a été fourni.

- Naviguez dans la structure de la table **EMPLOYEES**.
- Visualisez les données de la table **DEPARTMENTS**.

Prenez des captures d'écran pour les insérer dans votre rapport de TPs.

Avec **SQL*Developer**

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
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

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_grades;
```

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 Exercice 2

Sur APEX. Naviguer dans « **SQL Workshop** » puis cliquez sur « **SQL Commands** » pour entrer des instructions SQL en interactif. 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. Créer 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. Sauvegardez en cliquant sur « Save » votre instruction SQL sous le nom **ex_01_05.sql**. Prenez une capture d'écran et insérez la dans votre rapport.

	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. Créer une interrogation affichant tous les ID de postes uniques de la table **EMPLOYEES**.

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

Deuxième Partie

1.3 Liste des employés et leurs départements

Écrivez une requête SQL pour afficher le FIRST_NAME, LAST_NAME et DEPARTMENT_NAME de chaque employé.

1.4 Salaire moyen par département

Écrivez une requête SQL pour calculer le salaire moyen de chaque département. Affichez le DEPARTMENT_NAME et le salaire moyen arrondi à deux décimales.

1.5 Employés sans manager

Écrivez une requête SQL pour afficher les EMPLOYEE_ID, FIRST_NAME et LAST_NAME des employés qui n'ont pas de manager (c'est-à-dire dont le MANAGER_ID est NULL).

1.6 Top 5 des salaires les plus élevés

Écrivez une requête SQL pour afficher les cinq salaires les plus élevés parmi les employés, ainsi que les EMPLOYEE_ID, FIRST_NAME, LAST_NAME et SALARY correspondants.

1.7 Employés ayant le même job

Écrivez une requête SQL pour afficher les employés ayant le même JOB_ID, en regroupant les résultats par JOB_ID. Affichez le JOB_ID, FIRST_NAME et LAST_NAME pour chaque employé.

1.8 Liste des départements et de leurs managers

Écrivez une requête SQL pour afficher le DEPARTMENT_ID, DEPARTMENT_NAME, MANAGER_ID, FIRST_NAME et LAST_NAME de chaque manager de département.

1.9 Employés embauchés en 2020

Écrivez une requête SQL pour afficher les EMPLOYEE_ID, FIRST_NAME, LAST_NAME et HIRE_DATE des employés embauchés en 2020.

1.10 Augmentation de salaire pour un département spécifique

Écrivez une requête SQL pour mettre à jour le salaire de tous les employés d'un département spécifique (par exemple, DEPARTMENT_ID = 10) en augmentant leur salaire de 10 %. Affichez le EMPLOYEE_ID, FIRST_NAME, LAST_NAME et le nouveau SALARY de chaque employé concerné.

1.11 Propriétés des fonctions de groupe

1. 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

1.12 Utilisation basique des fonctions d'agrégation

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.

1.13 Utilisation de la clause GROUP BY

Afficher le salaire minimum, le salaire maximum, le salaire cumulé et le salaire moyen pour chaque type de poste (job_id).

1.14 Calculs avec les fonctions de groupe

Afficher le numéro de manager et le salaire de l'employé le moins payé sous les ordres de ce manager. Exclure toute personne pour laquelle le manager n'est pas connu. Exclure les groupes dans lesquels le salaire minimum est de 6 000 \$ ou moins. Triez les résultats par ordre décroissant sur la base du salaire.

1.15 Calculs avec les fonctions de groupe

Créez une matrice permettant d'afficher le poste (`job_id`), 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.